

# Bausteine für die Simatic S7 – TIA und die Profibus-Anschaltung SERVOSTAR® 300/400/600 und S700

## Inhaltsverzeichnis

Hinweise für das TIA-Projekt „Sv14_v3c_V14_SP1_1516“ für Servostar® 300/400/600 & S700.....	2
TIA Gerätekonfiguration.....	6
Servostar Einstellungen mit der DriveGui .....	8
Bausteinübersicht .....	10
Anmerkungen .....	10
Eingang- und Ausgangsschnittstelle vom Axis_01_(FB14).....	11
Axis_01_FB (FB14).....	11
IF_DB (DB70) .....	11
Symbolische Adressierung von dem Hardware-Interface an das Axis_01_FB ( FB14) - Interface.....	12
Gesamtübersicht der Schnittstelle vom Axis_01_FB (FB14) .....	14
Programmbeispiele zur Umrechnung der reglerinternen Einheiten in Benutzer/SI-Einheiten.....	18
GenCycleOB (OB1) - Network 11: "IF_DB".ActualPosition.....	18
GenCycleOB (OB1) – Netzwerk 12: "IF_DB".ActualSpeed.....	18
GenCycleOB (OB1) – Netzwerk 13: "IF_DB".ActualCurrent .....	18
GenCycleOB (OB1) – Netzwerk 20: "IF_DB".ActualPressure .....	19
GenCycleOB (OB1) – Netzwerk 21: "IF_DB".ActualFlow .....	19
Inbetriebnahme Axis_01_FB (FB14).....	20
S7-PLC - AUS/EIN-Schalten.....	20
S7-PLC - Run/Stop-Schalter .....	20
iAck =1 .....	20
Initialisierung .....	20
Betriebsart setzen .....	20
Betrieb freigeben.....	20
Anforderungen für die Betriebsart Positionieren .....	21
Tippbetrieb.....	21
Referenzfahrt (Homing).....	21
Start eines EEPROM oder RAM Fahrauftrags ( MotionTask).....	21
Start von dem Direktfahrauftrag ( RAM und hat die Nummer 0 ) .....	21
Betriebsart Drehzahl digital .....	23
Betriebsart Drehmoment digital .....	23
Betriebsart Elektrisches Getriebe .....	23
Betriebsart Servopumpe digital – Drehzahl .....	24
Makro-Programm.....	25
Parameter schreiben .....	28
Parameter lesen .....	28
Zusätzliche Profibus-Funktionen .....	30
Beobachtungs- und Forcetabelle .....	30
Beispiel für eine Schrittkette „Initialisierung und Enablen Servostar“ .....	31
Beispiel für eine Schrittkette „Starte Fahrsätze im Servostar“ .....	33
Beispiel für den “RAM MotionTask 201 im Servostar“ .....	36
Bausteine für Servostar-Parameter.....	38
Axis_01_WriteDataDB (DB20).....	38
Axis_01_ReadDataDB (DB21).....	38
Axis_01_Write (FB16).....	39
Axis_01_Read (FB17) .....	40
Axis_01_Compare (FB18).....	41
Tipps und Infos: .....	42
Fehlermeldungen und Warnungen von den Bausteinen .....	42
Override über Profibus .....	42
Aktivieren der Trajektorie/Profil Beschleunigung $\sin^2$ bei Direktfahrsatz Nr. 0 .....	43
Bitcodierung der Fehler – PNU 1001 - ERRCODE .....	46
Bitcodierung der Warnungen und Herstellerspezifisches Statusregister – PNU 1002 - DRVSTAT .....	47
Switch ON and Start MotionTask timing diagram.....	48
Kill MotionTask timing diagram.....	49

## Hinweise für das TIA-Projekt „Sv14\_v3c\_V14\_SP1\_1516“ für Servostar<sup>®</sup> 300/400/600 & S700

### Bisher erschienene Ausgaben:

Revision/Ausgabe	Comment/Bemerkungen	Author/Autor
A, 04/2018	first edition / Erstausgabe	KeH

### Warenzeichen

- Servostar ist ein eingetragenes Warenzeichen der Kollmorgen Corporation
- Simatic, Step7, TIA-Portal sind geschützte Marken der Siemens AG
- Profibus und ProfiNet sind geschützte Marken der PROFIBUS and PROFINET International(PI)

Dieses Dokument ist gültig für die PROFIBUS-DP Anbindung der Kollmorgen Servoverstärker Servostar<sup>®</sup> 300/400/600 und S700 an eine Simatic S7-1500 PLC Steuerung.

Im folgenden Text werden die Servoverstärker Servostar<sup>®</sup> 300/400/600 & S700 kurz als „Servostar“ bezeichnet.



**Verwenden Sie das S7-Projekt „Sv14\_v3c\_V14\_SP1\_1516“ niemals unverändert in einer Anwendung.  
Das S7-Projekt „Sv14\_v3c\_V14\_SP1\_1516“ ist ein Beispiel wie der Servostar in ein S7-Projekt integriert werden kann. Dieses Projektbeispiel muss immer an die vorhandene Anwendung angepasst werden.**



**KOLLMORGEN Europe GmbH haftet nicht für Schäden und schliesst alle Ansprüche aus, die sich durch den Einsatz des S7-Projekts „Sv14\_v3c\_V14\_SP1\_1516“ oder Programmteilen daraus ergeben könnten.**



**Beachten und befolgen Sie unbedingt die Sicherheits- und Warnhinweise der Hersteller zu den jeweiligen Komponenten.**

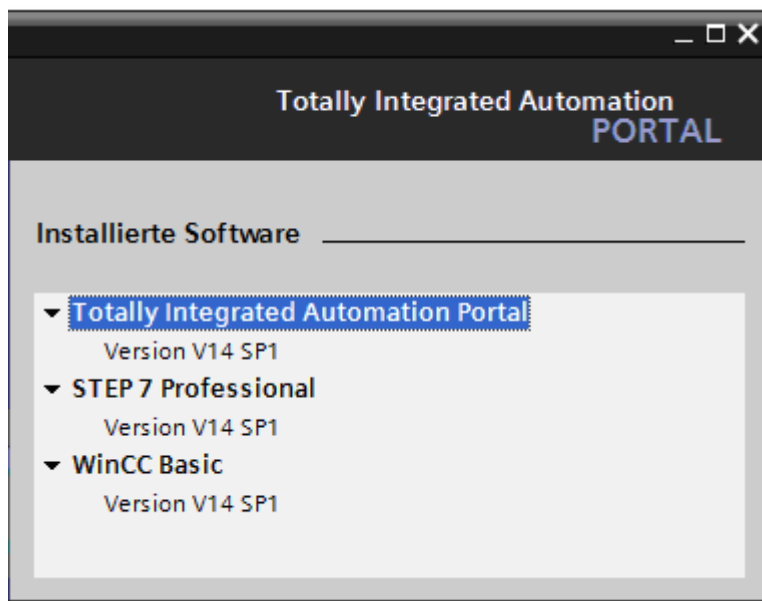
Gute Kenntnisse in Simatic TIA, der Steuerungs- und Antriebstechnik werden vorausgesetzt.

Ein einwandfreies Verhalten der kompletten Servoachse bestehend aus S7-1500 PLC und Servostar und Motor erfordert eine korrekt durchgeführte Inbetriebnahme !!!

Alle Verweise auf das Handbuch beziehen sich auf die technische Beschreibung „srprof\_e.pdf / srprof\_d.pdf“

Die verwendeten Begriffe und Abkürzungen entsprechen der Servostar Setup Software DriveGui bzw. dem Handbuch in der Sprache „English“.

Das TIA Projekt „Sv14\_v3c\_V14\_SP1\_1516“ ist erstellt in der Sprache English mit Mnemonik International und der Version:



Das S7-Projekt „Sv14\_v3c\_V14\_SP1\_1516“ besteht aus den folgenden Komponenten:

#### PLC\_1 ( CPU1516-3 PN/DP )

- Geräte- und Netzwerkkonfiguration für eine Simatic S7-CPU1516 – 6ES7 516-3AN00-0AB0
- Programmbausteine
- PLC-Variablen
- PLC-Datentypen
- Beobachtungs- und Forcetabellen

#### Nicht gruppierte Geräte ( Servostar )

- Slave\_1

The screenshot shows the Siemens TIA Portal interface for a project named "Sv14\_v3c\_V14\_SP1\_1516". The main window displays a network diagram with two devices: "PLC\_1" (CPU 1516-3 PN/DP) and "Slave\_1" (Servostar). They are connected via a "PROFIBUS\_1" network. The left sidebar shows the project navigation tree, and the right sidebar shows the device catalog. The bottom panel displays the properties of the "Slave\_1" device.

Gerät	Typ	Adresse im Sub...
S71500/ET200MP-Station_1	S71500/ET200MP station	
PLC_1	CPU 1516-3 PN/DP	
PROFINET-Schnittstell...	PROFINET interface	192.168.0.1
Port_1	Port	
Port_2	Port	
PROFINET-Schnittstell...	PROFINET interface	192.168.1.1
Port_1	Port	
DP-Schnittstelle_1	DP interface	2
GSD-Gerät_1	GSD device	
Slave_1	Servostar	12

Properties of Slave\_1 (Module):

- Kurzbezeichnung: Servostar
- Beschreibung: (koll045d.gsd)
- Artikel-Nr.:
- Firmware-Version: Z01
- GSD-Datei: koll045d.gsd

## System- und Taktmerker – Verwendung der System / Taktmerker – PLC-Variablen sind default

The screenshot shows the Siemens TIA Portal interface for a project named 'Sv14\_v3c\_V14\_SP1\_1516'. The main window displays the 'Geräteübersicht' (Device Overview) for 'PLC\_1 [CPU 1516-3 PN/DP]'. The 'System- und Taktmerker' (System and Edge Triggers) configuration is shown in the 'Eigenschaften' (Properties) pane.

**System- und Taktmerker Configuration:**

- Systemmerkerbits (System Edge Triggers):**
  - ☒ Verwendung des Systemmerkerbytes aktivieren (Enable use of system edge trigger byte)
  - Adresse des Systemmerkerbytes (MBx): 1
  - Erster Zyklus: %M1.0 (FirstScan)
  - Diagnosestatus geändert: %M1.1 (DiagStatusUpdate)
  - Immer 1 (high): %M1.2 (AlwaysTRUE)
  - Immer 0 (low): %M1.3 (AlwaysFALSE)
- Taktmerkerbits (Edge Triggers):**
  - ☒ Verwendung des Taktmerkerbytes aktivieren (Enable use of edge trigger byte)
  - Adresse des Taktmerkerbytes (MBx): 0
  - Takt 10 Hz: %M0.0 (Clock\_10Hz)
  - Takt 5 Hz: %M0.1 (Clock\_5Hz)
  - Takt 2.5 Hz: %M0.2 (Clock\_2.5Hz)
  - Takt 2 Hz: %M0.3 (Clock\_2Hz)
  - Takt 1.25 Hz: %M0.4 (Clock\_1.25Hz)

The screenshot shows the 'PLC-Variablen' (PLC Variables) configuration in the Siemens TIA Portal. The 'Variablen' (Variables) tab is selected, showing a list of variables for 'PLC\_1 [CPU 1516-3 PN/DP]'.

Name	Variablen-tabelle	Datentyp	Adresse	Rema...	Erreic...	Schrei...	Sichtb...	Überwach...	Kommen...
1 System_Byte	Standard-Variab...	Byte	%MB1						
2 Clock_Byte	Standard-Variablen...	Byte	%MB0						
3 FirstScan	Standard-Variablen...	Bool	%M1.0						
4 DiagStatusUpdate	Standard-Variablen...	Bool	%M1.1						
5 AlwaysTRUE	Standard-Variablen...	Bool	%M1.2						
6 AlwaysFALSE	Standard-Variablen...	Bool	%M1.3						
7 Clock_10Hz	Standard-Variablen...	Bool	%M0.0						
8 Clock_5Hz	Standard-Variablen...	Bool	%M0.1						
9 Clock_2.5Hz	Standard-Variablen...	Bool	%M0.2						
10 Clock_2Hz	Standard-Variablen...	Bool	%M0.3						
11 Clock_1.25Hz	Standard-Variablen...	Bool	%M0.4						
12 Clock_1Hz	Standard-Variablen...	Bool	%M0.5						
13 Clock_0.625Hz	Standard-Variablen...	Bool	%M0.6						
14 Clock_0.5Hz	Standard-Variablen...	Bool	%M0.7						
15 System_1_4	Standard-Variablen...	Bool	%M1.4						
16 System_1_5	Standard-Variablen...	Bool	%M1.5						
17 System_1_6	Standard-Variablen...	Bool	%M1.6						
18 System_1_7	Standard-Variablen...	Bool	%M1.7						
19 <Hinzufügen>									



## TIA Gerätekfiguration

Unter anderem bitte beachten:

- \* symbolische Systemkonstanten – Hardware-Kennung
  - siehe auch PLC Variablen > Alle Variablen anzeigen > Systemkonstanten
- \* Eingang-Adresse / Ausgang-Adresse
- \* GSD-file-Version
- \* Subnetz, Adresse und Übertragungsgeschwindigkeit
  - Der Servostar stellt sich automatisch auf die Übertragungsgeschwindigkeit ( Baudrate ) ein.
- \* Ansprechüberwachung Watchdog

### Symbolische Systemkonstanten – Hardware-Kennung

The screenshot shows the Siemens TIA Portal interface for configuring a PLC and a Slave device. The main window displays a network diagram with a PLC\_1 and a Slave\_1 connected via PROFIBUS\_1. The right pane shows the 'Netzübersicht' (Network Overview) table, and the bottom pane shows the 'Systemkonstanten' (System Constants) table for the device.

**Netzübersicht (Network Overview):**

Gerät	Typ	Adresse im Sub...
S71500/ET200MP-Station_1	S71500/ET200MP station	
PLC_1	CPU 1516-3 PN/DP	
PROFINET-Schnittstell...	PROFINET interface	192.168.0.1
Port_1	Port	
Port_2	Port	
PROFINET-Schnittstell...	PROFINET interface	192.168.1.1
Port_1	Port	
DP-Schnittstelle_1	DP interface	2
GSD-Geraet_1	GSD device	
Slave_1	Servostar	12

**GSD-Geraet\_1 [Device] - Systemkonstanten:**

Name	Typ	HW-Kennung	Verwendet von	Kommentar
Slave_1-DPSlave	Hw_DpSlave	259	PLC_1	
Slave_1-Head	Hw_Interface	258	PLC_1	
Slave_1-4_Wort_AE_AA_PKW	Hw_SubModule	261	PLC_1	
Slave_1-6_Wort_AE_AA_PZD	Hw_SubModule	262	PLC_1	

## Eingang Adresse / Ausgang Adresse und GSD-file-Version

**Geräteübersicht**

Modul	Baugr...	Steck...	E-Adresse	A-Adres...	Typ
Slave_1	0	0	256...263	256...263	Servostar
4 Wort AE_AA_PKW	0	1	256...263	256...263	4 Wort AE/AA/Konsistenz ...
6 Wort AE_AA_PZD	0	2	264...275	264...275	6 Wort AE/AA/Konsistenz ...

**Eigenschaften**

**Allgemein**

Kataloginformation

PROFIBUS-Adresse

Allgemeine DP-Parameter

Ansprechüberwachung

SYNC/FREEZE

HW-Kennung

Kurzbezeichnung: Servostar

Beschreibung: (koll045d.gsd)

Artikel-Nr.:

Firmware-Version: Z01

GSD-Datei: koll045d.gsd

Installationszeitpunkt: Mittwoch, 13. September 2017 12:18

## Subnetz, Adresse und Übertragungsgeschwindigkeit

**PROFIBUS-Adresse**

Schnittstelle vernetzt mit

Subnetz: PROFIBUS\_1

Parameter

Adresse: 12

Höchste Adresse: 126

Übertragungsgeschwindigkeit: 1.5 Mbit/s

## Ansprechüberwachung Watchdog

**Ansprechüberwachung**

☒ Watchdog aktiviert

## Servostar Einstellungen mit der DriveGui

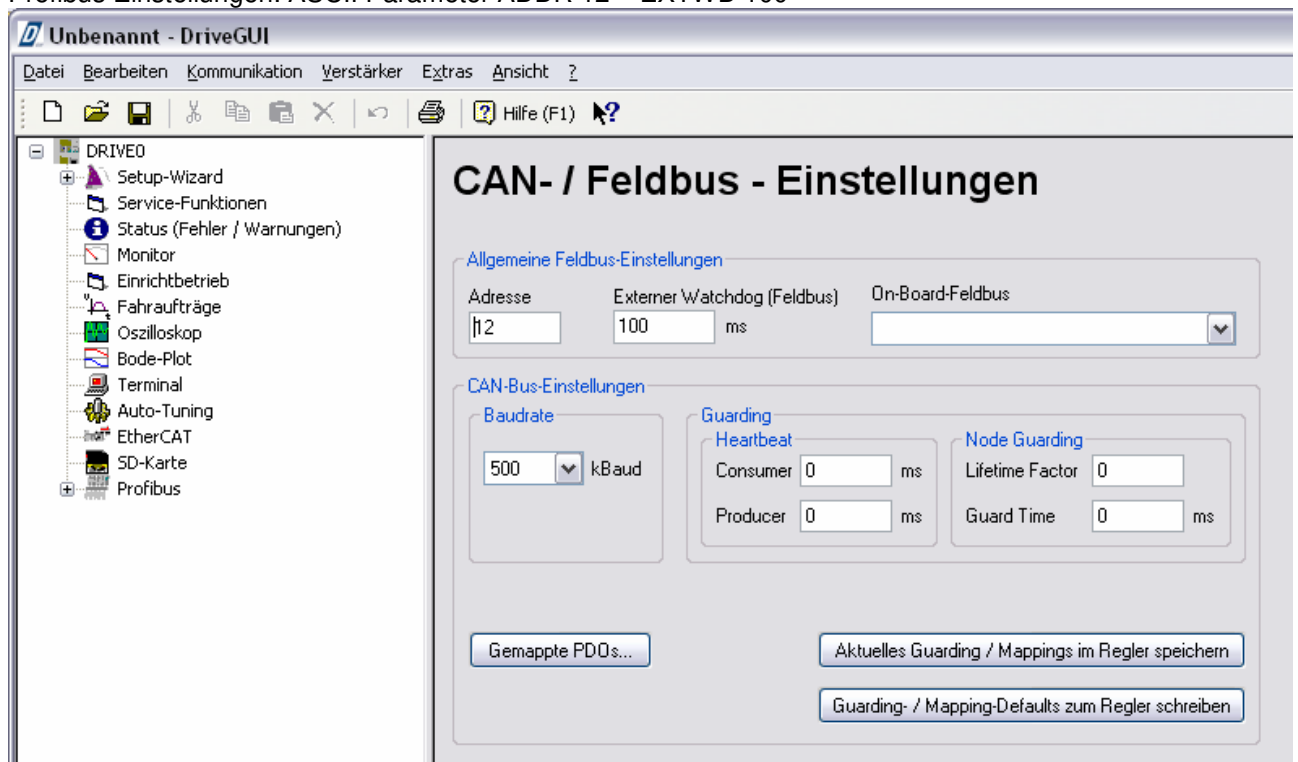
Für die Inbetriebnahme für den Servostar S400 / S600 wird die Software Drive benutzt.  
Dies wird hier im Detail nicht näher beschrieben.



Für die Inbetriebnahme für den Servostar S300 / S700 wird die Software DriveGui benutzt.  
DriveGui-Version:



Profibus Einstellungen: ASCII Parameter ADDR 12 – EXTWD 100





## Profibus – Daten – Eingang / Ausgang oder Send / Receive – PKW und PZD - Interface

**Profibus**

Baudrate: 1500 kBaud  
Adresse: 12  
PNO-Identnr.: H045D  
PPO-Typ: 2

Verstärker  
Leistungs-  
teil  
Rege-  
lung  
Profibus-  
Interface

Profibus-Master  
Profibus  
Ausgang  
Eingang

**PROFIBUS-Interface-Zustände**  
Watchdog-Status: Baudr.-Suche — Prüft. d. Baudr. — DP Mode — Kommunikation  
DP-Status: Warten, Param. — Warten, Konfig. — Datenaustausch — OK

☐ Ansprechüberwachung ignorieren

**Ein- / Ausgangsbuffer**

PKW			PZD					
PKE	IND	PWE	STW	HSW	PZD3	PZD4	PZD5	PZD6
0000	0000	0000 0000	0034	0000	0000	0000	0000	0000
PKE	IND	PWE	ZSW	HIW	PZD3	PZD4	PZD5	PZD6
0000	0000	0000 0000	0270	0000	0000	0000	0000	0000

Zustandsmaschine

Bereit S706 (DRIVE0) @ COM1 (38,4 kBaud) Online No SW

## Profibus – Steuerwort STW und Zustandswort ZSW - Interface

**Profibus - Zustandsmaschine**

**Steuerwort (STW)**

0	0	Einschalten
1	0	Spannung sperren
2	1	Schnellh. in Einsch.
3	0	Betrieb freigegeben
4	1	Schnellhalt (HLG sperr.)
5	1	Zwisch. (HLG stop)
6	0	Sollwertfreigabe
7	0	Reset Fehler
8	0	Tippen (Ein/Aus)
9	0	reserviert
10	0	PZD (Freig./Sperren)
11	0	Start Refer. (Flanke)
12	0	herstellerspezifisch
13	0	herstellerspezifisch
14	0	herstellerspezifisch
15	0	herstellerspezifisch

**Zustandsmaschine**

```

graph TD
    Start --> 0
    0 --> Nicht_einschaltbereit[Nicht einschaltbereit]
    Nicht_einschaltbereit -- 1 --> Einschaltsperr[Einschaltsperr]
    Einschaltsperr -- 2 --> Einschaltbereit[Einschaltbereit]
    Einschaltbereit -- 3 --> Betriebsbereit[Betriebsbereit]
    Betriebsbereit -- 4 --> Betrieb_freigegeben[Betrieb freigegeben]
    Betrieb_freigegeben -- 5 --> Schnellhalt[Schnellhalt]
    Schnellhalt -- 11 --> Betrieb_freigegeben
    Betrieb_freigegeben -- 16 --> Schnellhalt
    Schnellhalt -- 12 --> Fehler[Fehler]
    Fehler -- 14 --> Fehlerreaktion_aktiv[Fehlerreaktion aktiv]
    Fehlerreaktion_aktiv -- 15 --> Einschaltsperr
    Fehlerreaktion_aktiv -- 10 --> Einschaltbereit
    Fehlerreaktion_aktiv -- 9 --> Einschaltbereit
    Fehlerreaktion_aktiv -- 17 --> Einschaltbereit
    
```

**Zustandswort (ZSW)**

0	0	Einschaltbereit
1	0	Eingeschaltet
2	0	Betrieb freigegeben
3	0	Fehler
4	1	Spannung gesperrt
5	1	Schnellhalt
6	1	Einschaltsperr
7	0	Warnung
8	0	SollHstwertüberwach.
9	1	Remote
10	0	Sollwert erreicht
11	0	Begrenzung aktiv
12	0	Abh. v. d. Betriebsart
13	0	Abh. v. d. Betriebsart
14	0	herstellerspezifisch
15	0	herstellerspezifisch

Bereit S706 (DRIVE0) @ COM1 (38,4 kBaud) Online No SW

## Bausteinübersicht

Axis_01_FB	FB14/DB14	Hantierungsbaustein zwischen S7-PLC und Servostar
Axis_01_Write	FB16/DB16	Baustein für den FB14 zum Daten schreiben in den Servostar
Axis_01_WriteData	DB20	Enthält die Daten, die vom FB16 in den Servostar geschrieben werden
Axis_01_Read	FB17/DB17	Baustein für den FB14 zum Daten lesen aus dem Servostar
Axis_01_ReadData	DB21	Enthält die Daten, die vom FB17 aus dem Servostar gelesen werden
Axis_01_Compare	FB18/DB18	Baustein zum Datenvergleich vom z. Bsp. DB20 und DB21
IF_DB	DB70	Interface-Datenbaustein z. Bsp. für ein Operator Panel /Touch Panel

Alle Bausteine können bei Bedarf umbenannt bzw. umnummeriert werden.

Anschließend muss "Übersetzen -> Software (komplett übersetzen)" durchgeführt werden.

Daraus resultierende Konflikte müssen von Hand in den Bausteinen korrigiert werden.

Der Axis\_01\_FB (FB14) nutzt die folgenden Bausteine:

D\_ACT\_DP – Deaktivieren und Aktivieren von DP Slaves / ProfiNet IO devices

DPRD\_DAT – Konsistente Daten lesen eines DP Slaves / ProfiNet IO devices

DPWR\_DAT – Konsistente Daten schreiben eines DP Slaves / ProfiNet IO devices

BLKMOV – Speicherbereich kopieren

FILL – Speicherbereich vorbesetzen

und den Systembaustein:

BITSUM ( FC99 ) – Anzahl der gesetzten Bits zählen

Der Axis\_01\_Write (FB16), Axis\_01\_Read (FB17), Axis\_01\_Compare (FB18) benutzt die Systembausteine:

ReadFromArrayDB ( FC901 ) – Aus Array Datenbaustein lesen

WriteToArrayDB ( FC902 ) – In Array Datenbaustein schreiben

### Anmerkungen

- Der Axis\_01\_FB (FB14) ist als Multi-Instanz programmiert und kann somit in einem FB mehrmals als Unterprogramm-Baustein ohne eigenen Instanz-Datenbaustein aufgerufen werden.
- Wenn der Axis\_01\_FB (FB14) nicht als Multi-Instanz implementiert wird, dann muss der Axis\_01\_FB (FB14) für jeden Antrieb mit einem eigenen Instanz-Datenbaustein aufgerufen werden.
- Der Axis\_01\_FB (FB14) ist in FUP ( Funktionsplan ) mit der Mnemonik International und der Symbolik mit Kommentaren in ENGLISCH programmiert.
- Die PLC-Datentypen ermöglichen eine einheitliche Datenbasis und eine objektorientierte Programmierung in TIA.
- Die STRUCT-Datentypen ermöglichen ganze Datenbereiche einfach mittels Pointer zu adressieren.
- Das S7-Projekt „Sv14\_v3c\_V14\_SP1\_1516“ enthält ein komplett funktionsfähiges S7-PLC Programm für den Servostar - Profibus.
- Das S7-Projekt „Sv14\_v3c\_V14\_SP1\_1516“ wurde von dem Step7-Projekt „Sv14\_v3c“ in das Simatic TIA- Portal migriert und dann komplett in FUP umprogrammiert.
- Für die S7-1200 PLC gibt es das komplett funktionsfähige Projekt „Sv14\_v3c\_V14\_SP1\_1214C“.

Mit einer S7-1500 PLC und einem Servostar mit Profibus-Optionskarte und Motor kann die komplette Servoachse schnell und einfach mit der Watch table\_1 in Betrieb genommen werden.

## Eingang- und Ausgangschnittstelle vom Axis 01 (FB14)

Die Schnittstelle vom FB14, FB16, FB17, FB18 und DB20, DB21, DB70 verwendet PLC-Datentypen.

M\_Axis\_01\_Request  
 M\_Axis\_01\_MaMsg – Machine messages / Error messages  
 M\_Axis\_01\_State  
 M\_Axis\_01\_InData  
 M\_Axis\_01\_OutData  
 M\_Axis\_01\_PnuStruct  
 M\_Axis\_01\_RcvSend - Receive and Send Profibus

Die PLC-Datentypen stellen sicher dass die Datenstruktur in dem S7-Projekt eine einheitliche gemeinsame Datenbasis hat. Eine Änderung an der gesamten Datenstruktur erfolgt einmalig im PLC- Datentyp. Die gesamte Datenstruktur vom S7-Projekt wird dann mit "Übersetzen -> Software (komplett übersetzen)" automatisch aktualisiert.

Bitte danach prüfen dass die Anfangswerte von den PLC-Datentypen in die Aktualwerte von dem DB transferiert wurden. Wenn nicht dann ist für diesen DB eine separate Initialisierung oder eine korrekte Einstellung der Bausteineigenschaften erforderlich.

## Axis 01 FB (FB14)

Eingangsvariablen haben ein „i“ ( Input ) und Ausgangsvariablen ein „o“ ( Output ) vorangestellt. Damit ist der Programmcode leichter lesbar.

VAR_INPUT		
iData	M_Axis_01_InData	
iRequest	M_Axis_01_Request	
iPnu	Struct	
Write	M_Axis_01_PnuStruct	
..Read	M_Axis_01_PnuStruct	
...		
VAR_OUTPUT		
oMaMsg	M_Axis_01_MaMsg	
oState	M_Axis_01_State	
oData	M_Axis_01_OutData	
oAxis	M_Axis_01_RcvSend	
...		

## IF DB (DB70)

ToMachine	IfGroupToUnit	
FromMachine	IfUnitToGroup	
Request	M_Axis_01_Request	
State	M_Axis_01_State	
...		
iData	M_Axis_01_InData	
oData	M_Axis_01_OutData	
oMaMsg	M_Axis_01_MaMsg	
oAxis	M_Axis_01_RcvSend	
...		

## Symbolische Adressierung von dem Hardware-Interface an das Axis\_01\_FB (FB14) - Interface

TIA generiert automatisch symbolische System constants – Hardware identifiers  
– siehe ebenfalls PLC Variablen > Alle Variablen anzeigen > Systemkonstanten

Die numerischen Adressen können unerwartet durch TIA nach "Übersetzen -> Hardware (komplett übersetzen)" modifiziert werden. Deshalb im Programm immer nur die symbolische Adresse benutzen !!!

Die Eingangsadresse / Ausgangsadresse wird in der Geräte und Netzwerkkonfiguration eingestellt.  
Siehe Kapitel TIA-Gerätekonfiguration

Mit drag&drop oder copy&paste wird die symbolische Adresse einfach von den Systemkonstanten oder von der Gerätekonfiguration auf den Input von dem MOVE geschrieben.

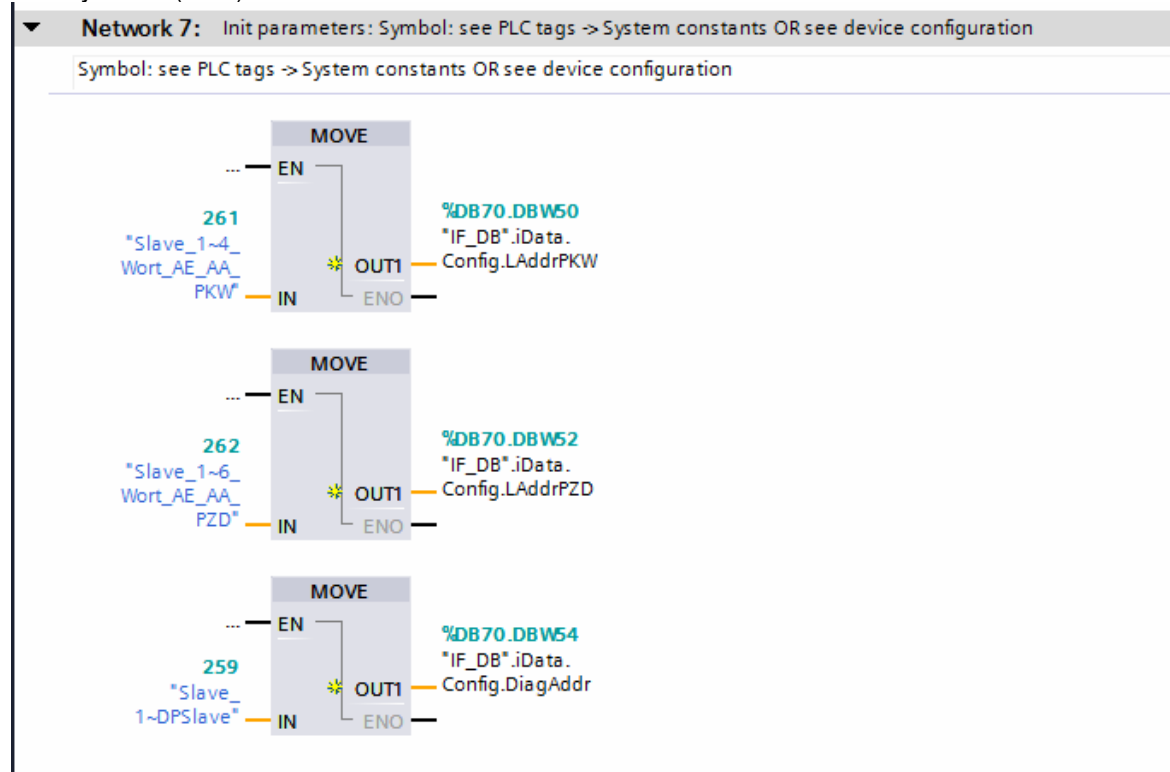
Jeder Servostar benötigt die Zuordnung von seiner Gerätekonfiguration zu dem Axis\_01\_FB (FB14) durch den Datentyp HardwareInterface.

Hinweis: Bei Copy&Paste von Projektteilen in anderes Projekt werden die symbolischen Adressen nicht immer identisch übernommen. Die TIA Gerätekonfiguration generiert dann z.Bsp. eine Erweiterung xxxPKW\_1, xxxPZD\_1, xxxSlave\_1.

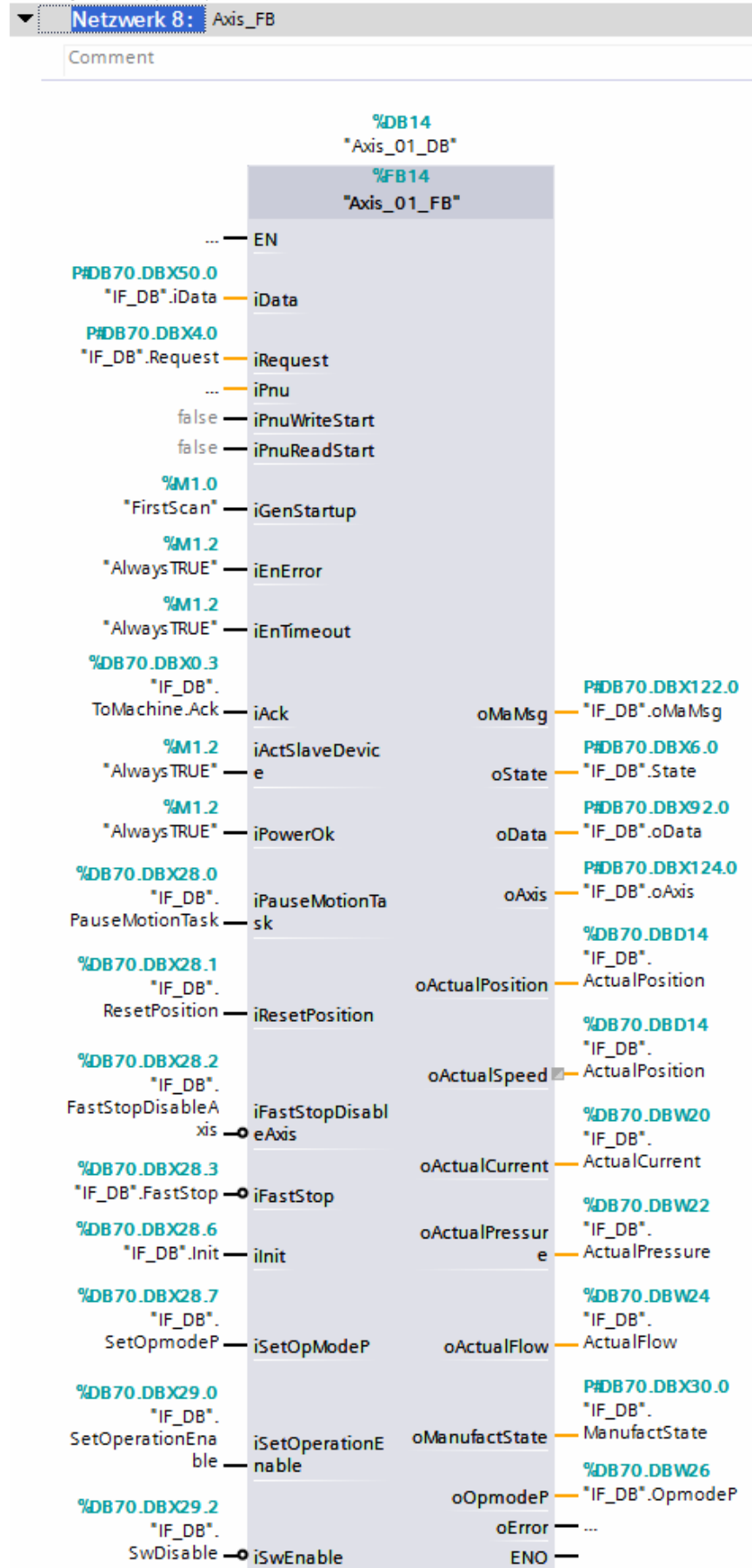
Deshalb immer unter PLC tags > show all tags > System constants prüfen dass die Symbole eindeutig und einmalig vergeben sind. In der TIA device configuration oder im Programm kann dann die Zuordnung berichtigt werden und das nicht benutzte Symbol sollte anschliessend gelöscht werden.

Ist die Zuordnung korrekt durchgeführt kommunizieren die S7-PLC und der Servostar über den Profibus und der Axis\_01\_FB (FB14) meldet oState.CommunicationOk =1

### GenCycleOB (OB1)



## GenCycleOB (OB1)





## Gesamtübersicht der Schnittstelle vom Axis 01 FB (FB14)

VAR_INPUT		
iData	STRUCT	Data: HWConfig, OpMode, RefJogSpeed, MotionTask
Config	STRUCT	
LAddrPKE	HW_Inf	StartAddressIn in HW-Config periphery dez
LAddrPZD	HW_Inf	StartAddressOut in HW-Config periphery dez
DiagAddr	HW_Inf	DiagnosticAddress in HW-Config periphery hex
TO_Reference	Time	Timeout referencing
TO_Position	Time	Timeout positioning
OpModeP	DWORD	PNU 930 (2 =MotionTask, 1 =VelocityDigital, -2 =TorqueDigital ... )
JogSpeed	WORD	PNU 1889
RefSpeed	WORD	PNU 1896
MotionTask	STRUCT	
Number	WORD	0 =DirectMotionTask
DirectSpeed	DWORD	PNU 1791
DirectPosition	DWORD	PNU 1790
DirectType	WORD	PNU 1785
DigitalSpeed	STRUCT	
Ncmd	WORD	PNU1886 - Ncommand ( Ncmd16 = Ncmd * 2 <sup>15</sup> / VOSPD )
DigitalTorque	STRUCT	
Icmd	WORD	PNU1870 - Icommand ( I[A] = Icmd * IpeakAmplifier[A] / 3280 )
DigitalPump	STRUCT	
QPRcmd	WORD	QPR command ( pressure 1 <-> 10 mbar )
QFcmd	WORD	QF command ( flow 1 <-> 0,1 l/min )
iRequest	STRUCT	Requests: Ref Pos StartStopCancelMotionTask Jog
StartRef	BOOL	Start reference movement, static 1
StartIcmd	BOOL	Start I command digital torque, P4, static 1
StartMotionTask	BOOL	Start motion task ( direct motion task =0 ), P4
StopMotionTask	BOOL	Stop motion task, P4->P3
CancelMotionTask	BOOL	Cancel motion task, P3
StartNcmd	BOOL	Start N command digital speed, P4, static 1
JogPlus	BOOL	Jog positive, static 1
JogMinus	BOOL	Jog negative, static 1
MacroInput	BOOL	MacroProgramInput, static 1, PROSTW & 0x200
StartQcmd	BOOL	Start Q command digital speed, P4, static 1
Res_1_2	BOOL	
Res_1_3	BOOL	
Res_1_4	BOOL	
Res_1_5	BOOL	
Res_1_6	BOOL	
Res_1_7	BOOL	
iPnu	STRUCT	
Write	STRUCT	
Number	WORD	
Index	WORD	
Value	DWORD	
Read	STRUCT	
Number	WORD	
Index	WORD	
Value	DWORD	
iPnuWriteStart	BOOL	Request PnuWriteStart pulse 0->1
iPnuReadStart	BOOL	Request PnuReadStart static 1
iGenStartup	BOOL	Startup cycle after PLC startup
iEnError	BOOL	Enable error messages
iEnTimeout	BOOL	Enable timeout movement referencing and positioning
iAck	BOOL	Acknowledge WarningsErrors
iActSlaveDevice	BOOL	Activate Slave or Device
iPowerOk	BOOL	All powersupplies are ok
iPauseMotionTask	BOOL	Pause for Motion Task, P4
iResetPosition	BOOL	Reset position, set ActualPosition to RefPosition ( ROFFS )
iFastStopDisableAxis	BOOL	FastStop with disable axis, P4->P1
iFastStop	BOOL	FastStop without disable axis, P4->P11
iInit	BOOL	Initialize axis with disable axis
iSetOpModeP	BOOL	Set operating mode Profibus
iSetOperationEnable	BOOL	Set axis state machine to P4_OperationEnabled
iSwEnable	BOOL	Software enable axis

VAR_OUTPUT		
oMsg	STRUCT	Error messages
ErrTO_Ref	BOOL	Error timeout reference
ErrTO_Pos	BOOL	Error timeout positioning
ErrNoReferenceSet	BOOL	Error if reference is not set and request
ErrWrongOpMode	BOOL	Error wrong operation mode selected and request
ErrActSlaveDevice	BOOL	Error activating slave or device
ErrCfgInput	BOOL	Error configuration input
ErrRcv	BOOL	Error receiving data
ErrCfgOutput	BOOL	Error configuration output
ErrSend	BOOL	Error sending data
ErrAxis	BOOL	Error from axis
ErrReadWrite	BOOL	Error request Read and Write together
ErrNotEnabled	BOOL	Error if not enabled and request
ErrRes_1_4	BOOL	
ErrRes_1_5	BOOL	
ErrRes_1_6	BOOL	
ErrCmd	BOOL	Error more than one request command active
oState	STRUCT	AxisState
ReferencingActive	BOOL	Axis is referencing
JoggingActive	BOOL	Axis is jogging
VelocityIsZero	BOOL	Axis velocity is zero
MotionTaskActive	BOOL	Axis motion task is active
InPosition	BOOL	Axis is in position
Res_0_5	BOOL	
Res_0_6	BOOL	
Res_0_7	BOOL	
ReferenceOk	BOOL	Axis is referenced
CommunicationOk	BOOL	Axis communication Profibus is ok
InitOK	BOOL	Axis initialisation is ok
InitError	BOOL	Axis initialisation error
OpModeP_Ok	BOOL	Axis opmode Profibus is ok
OpModeP_Error	BOOL	Axis opmode Profibus error
WarningActive	BOOL	Axis warning active
WarnPositionError	BOOL	Axis warning position error
P0_NotReadySwitchOn	BOOL	state diagram
P1_SwitchOnInhibited	BOOL	state diagram
P2_ReadyForSwitchOn	BOOL	state diagram
P3_ReadyForOperation	BOOL	state diagram
P4_OperationEnabled	BOOL	state diagram
P11_FastStopActive	BOOL	state diagram
P13_ErrorReaction	BOOL	state diagram
P14_ErrorActive	BOOL	state diagram
PnuWriteOk	BOOL	Pnu write done and ok
PnuWriteError	BOOL	Pnu write not done and error
PnuReadOk	BOOL	Pnu read done and ok
PnuReadError	BOOL	Pnu read not done and error
Res_3_4	BOOL	
Res_3_5	BOOL	
Res_3_6	BOOL	
Res_3_7	BOOL	
ResponseTelegram_PKW_PWE	DWORD	Axis response telegram after Pnu Rcv or Send
oData	STRUCT	AxisData
Canceled	STRUCT	
DirectMotionTask	STRUCT	
STW	WORD	
Speed	DWORD	
Position	DWORD	
TaskType	WORD	
ActualSpeed	INT	
ActualPosition	DINT	
StartPosition	DINT	
PnuRead	STRUCT	
Number	WORD	
Index	WORD	
Value	DWORD	

oAxis	STRUCT	
Rcv	STRUCT	
PKW		
PKE	WORD	
IND	WORD	
PWE1	WORD	
PWE2	WORD	
PZD	STRUCT	
ZSW	STRUCT	
SetpointActualValMonitor	BOOL	only in OpMode POSITION: Following error
Remote	BOOL	not working, set to 1
SetpointReached	BOOL	only in OpMode POSITION: At Position
LimitActive	BOOL	at the moment not working
ModeDependentx	BOOL	used in ASCII-Modus
ModeDependenty	BOOL	used in ASCII-Modus
ModeDependentz	BOOL	used in ASCII-Modus
Reserved	BOOL	reserved
ReadyForSwitchOn	BOOL	
SwitchedOn	BOOL	
OperationEnabled	BOOL	
Error	BOOL	see ASCII-Kommando ERRCODE
VoltageInhibit	BOOL	
FastStop	BOOL	
SwitchOnInhibit	BOOL	
Warning	BOOL	see ASCII-Kommando STATCODE
HIW	WORD	
PZD3	WORD	
PZD4	WORD	
PZD5	WORD	
PZD6	WORD	
Send	STRUCT	
PKW	STRUCT	
PKE	WORD	
IND	WORD	
PWE1	WORD	
PWE2	WORD	
PZD	STRUCT	
STW	STRUCT	
JoggingOnOff	BOOL	OpMode dependent
Reserved	BOOL	
PZDenableInhibit	BOOL	
StartHomingRun	BOOL	OpMode dependent
ResetPosition	BOOL	
AckWarning	BOOL	Acknowledge warnings
MoTaskDirectOrMoTaskNr	BOOL	Only in OpModes Position: 0=MotionTaskNumber 1=DirectMotionTask
DigitalRevolutionSpeed	BOOL	OpMode dependent, digital velocity
SwitchOn	BOOL	
InhibitVoltage	BOOL	
FastStopSwitchOn	BOOL	1>0Axis FastStopWithEmgyRamp, AxisWillDisabled- STOPMODEECDIS
OperationEnabled	BOOL	
FastStopWithEmgyRamp	BOOL	1>0 Axis fast stop with emergency ramp
PauseStopRfg	BOOL	OpMode dependent, 1>0 Axis stop
SetpointEnable	BOOL	OpMode dependent
ResetFault	BOOL	Reset errors
HSW	WORD	
PZD3	WORD	
PZD4	WORD	
PZD5	WORD	
PZD6	WORD	

oActualPosition	DINT	Actual position, valid if PZD channel is active 1*)
oActualSpeed	INT	Actual speed, valid if PZD channel is active, OpmodePb 1,2
oActualCurrent	INT	Actual current, valid if PZD channel is active, OpmodePb -2
oActualPressure	INT	Actual pressure, valid if PZD channel is active, OpmodeP -7
oActualFlow	INT	Actual flow, valid if PZD channel is active, OpmodeP -7
oManufactState	STRUCT	comment valid for OpMode positioning
Pos3reached	BOOL	
Pos4reached	BOOL	
AxisInternalInitReady	BOOL	
x0_3	BOOL	
VelocityIsZero	BOOL	
SafetyRelayOpen	BOOL	
AxisEnabled	BOOL	
AxisErrorExist	BOOL	
MotionTaskActive	BOOL	
ReferenceDoneAndOK	BOOL	
ReferenceSwitchOn	BOOL	
InPositionWindow	BOOL	
LatchPositionDone	BOOL	
x1_5	BOOL	
Pos1reached	BOOL	
Pos2reached	BOOL	
oOpmodeP	WORD	Actual OpmodeP ProfibusDP ( Pnu 930)
oError	BOOL	Error is active

1\*) Die aktuelle Positon wird in den Prozessdaten nur in inkrementellen reglerinternen Einheiten (  $2^{20}$  Inkr. pro Motorumdrehung ) zur S7-PLC übertragen. Die Umrechnung von den reglerinternen Einheiten in Benutzereinheiten kann in der S7-PLC erfolgen und ist abhängig von der mit der Bediensoftware im Regler eingestellten Positonsauflösung PRBASE.

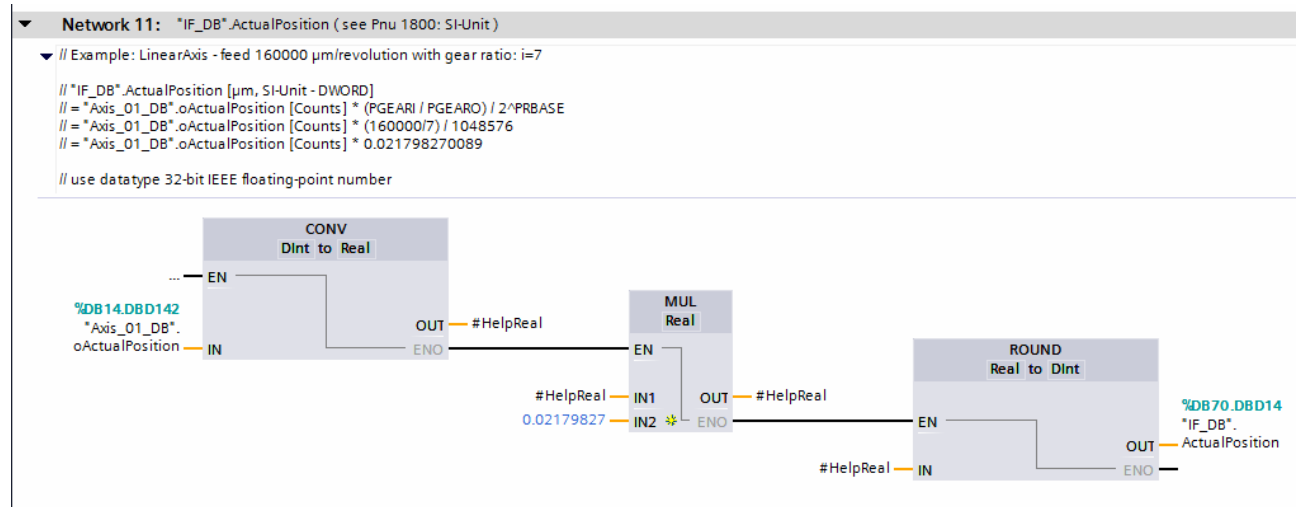
Beispiel mit PRBASE 20:

Auflösung = 5000 Inkr / 3 Umdrehungen:

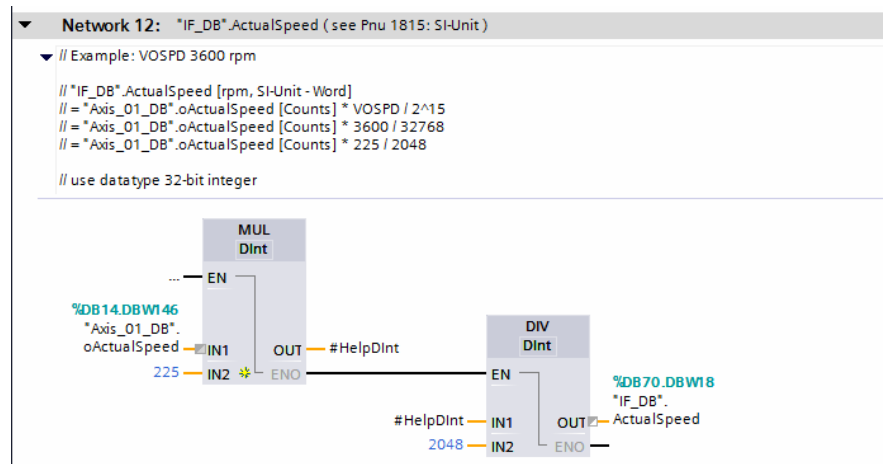
=> Position in Benutzereinheiten = Actual\_Position x 5000 / (3 x  $2^{20}$ )

## Programmbeispiele zur Umrechnung der reglerinternen Einheiten in Benutzer/SI-Einheiten

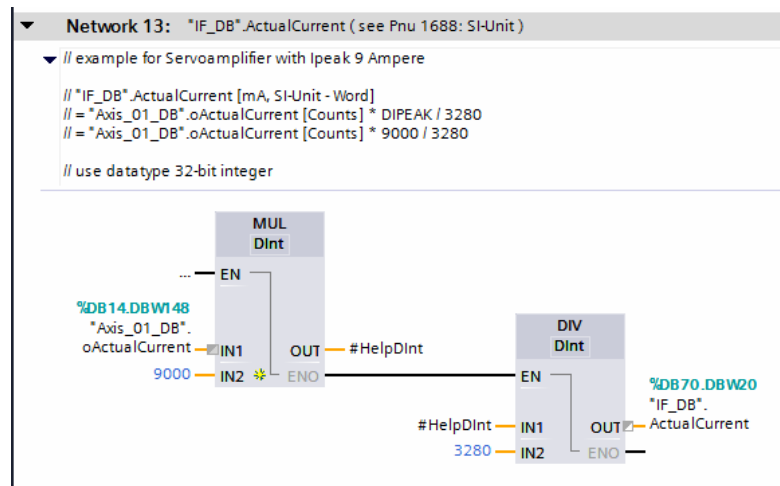
### GenCycleOB (OB1) - Network 11: "IF\_DB".ActualPosition



### GenCycleOB (OB1) – Netzwerk 12: "IF\_DB".ActualSpeed

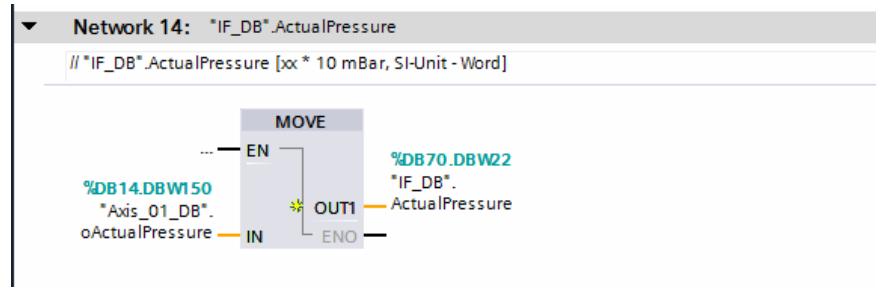


### GenCycleOB (OB1) – Netzwerk 13: "IF\_DB".ActualCurrent

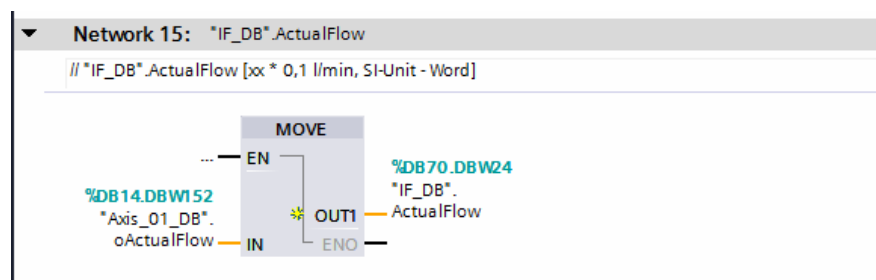




### GenCycleOB (OB1) – Netzwerk 20: "IF\_DB".ActualPressure



### GenCycleOB (OB1) – Netzwerk 21: "IF\_DB".ActualFlow



## Inbetriebnahme Axis 01 FB (FB14)

### S7-PLC - AUS/EIN-Schalten

Fällt der S7-PLC Profibus-Master aus dann startet im Servostar der Watchdog-Timeout (EXTWD) und generiert die Warnung „n04 – Ansprechüberwachung aktiv ( watch dog )“.

Die S7-PLC macht nach dem Anlauf einen Software-Reset (FirstScan - M1.0) und löscht alle Warnungen und Fehlermeldungen und Zustände im PLC-Programm, aber nicht im Servostar.

Bootet der Servostar schneller als die S7-PLC dann wird im Servostar die Warnung n04 generiert. Mit ASCII-Parameter S1DLY kann die Boot-Zeit verlängert werden um z.Bsp. die Warnung n04 zu vermeiden. Erst nach Quittieren dieser Warnung n04 kann der Servostar in den Betrieb gehen.

### S7-PLC - Run/Stop-Schalter

Im Zustandswechsel von STOP -> RUN macht die S7-PLC einen Software-Reset (FirstScan - M1.0) und löscht alle Warnungen und Fehlermeldungen und Zustände im PLC-Programm, aber nicht im Servostar.

### iAck =1

Setzt die Warnungen und Fehler zurück im Servostar und im S7-PLC Programm in den Bausteinen Axis\_01\_FB (FB14), Axis\_01\_Write (FB16), Axis\_01\_Read (FB17), Axis\_01\_Compare (FB18)

### Initialisierung

Bei der Initialisierung sendet die S7-PLC ein Null-Telegramm (Send.PKW und Send.PZD) an den Servostar. Der Servostar wird sofort disabled und die Zustandsmaschine wechselt in oState.P1\_SwitchOnInhibited =1

- Setze iInit =1 ( intern wird ein Impuls generiert )
- Warte bis oState.InitOk =1  
Wenn oState.InitOk nicht =1 oder oState.InitError =1 dann war die Initialisierung nicht erfolgreich.  
Fehlerursache siehe Receive-Telegramm Servostar.
- Die Initialisierung wird mit einem Timeout von 1 sec überwacht.

### Betriebsart setzen

Programmierte Betriebsarten: Position Fahrsätze, Drehzahl digital, Drehmoment digital, Elektrisches Getriebe, Servopumpe digital

Nach 24VDC EINschalten ist der Servostar immer in der sicheren internen Betriebsart -126 und „gesperrt“. Über Profibus findet eine Kommunikation zwischen der S7-PLC und dem Servostar nur von dem STW und ZSW statt und die RCV - PZD2, PZD3, PZD4, PZD5, PZD6 ( receive telegram ) enthalten alle den Wert "0". Die S7-PLC muss zuerst die Betriebsart ändern und anschliessend prüfen.

- Schreibe die gewünschte Betriebsart in iData.OpModeP  
( siehe Handbuch: Positionieren: 2 ; Drehzahl digital: 1 ; Drehmoment digital: -2 ;  
Elektrisches Getriebe: -4 , Servopumpe digital: -7 )
- Setze iSetOpModeP =1 ( intern wird ein Impuls generiert )
- Warte bis oState.OpModeP\_Ok = 1  
Die aktivierte Betriebsart wird angezeigt in oOpmodeP  
Wenn oState.OpModeP\_Ok nicht =1 oder oState.OpModeP\_Error =1 oder oOpModeP ist nicht der angeforderte OpModeP dann war Betriebsart setzen nicht erfolgreich.  
Fehlerursache siehe Receive-Telegramm Servostar.
- Jetzt werden auch die RCV - PZD2, PZD3, PZD4, PZD5, PZD6 angezeigt abhängig von der Betriebsart mit Werten für z. Bsp. oActualPosition, oActualSpeed, oManufactState.
- Die Betriebsart setzen wird mit einem Timeout von 1 sec überwacht.

### Betrieb freigeben

Nach 24VDC EINschalten ist der Servostar in oState.P1\_SwitchOnInhibited =1 ( Einschaltsperrung ). iFastStop und iFastStopDisableAxis müssen =0 und iSwEnable muss =1 sein.

Am Servostar muss HardwareEnable =1 und falls vorhanden muss AS/STO-Enable =1 sein.

Nur S700: Eine eingebaute Safetycard muss im Zustand „RUN“ sein.

Die Zwischenkreisspannung wird angelegt und im Display vom Servostar wird „Pxx“ angezeigt ohne Warnungen oder Fehler.

- Setze iSetOperationEnabled =1 ( intern wird ein Impuls generiert )
- Warte bis oState.P4\_OperationEnabled =1  
Wenn oState.P4\_OperationEnabled nicht =1 dann war Betriebsart setzen nicht erfolgreich.

Der Servostar ist nun enabled und kann eine Kraft / Drehmoment generieren und einen Motor bewegen.

Anforderungen für die Betriebsart Positionieren

- Der Betrieb ist freigegeben (oState.P4\_OperationEnabled =1)
- AS/STO- und Hardware-Enable ist vorhanden
- Keine Warnungen und Fehler und das Display vom Servostar zeigt „Exx“
- Die Betriebsart Position (oState.OpModeP\_Ok = 1 und oOpModeP = 2 dec ) ist aktiviert

Tippbetrieb

- Schreibe die gewünschte Tippgeschwindigkeit in „iData.Config.JogSpeed“
- Setze iRequest.JogPlus =1 -> Der Antrieb bewegt sich positiv und oState.JoggingActive =1
- Setze iRequest.JogMinus =1 -> Der Antrieb bewegt sich negativ und oState.JoggingActive =1

Referenzfahrt (Homing)

Der Referenzfahrttyp wird normalerweise mit der DriveGui eingestellt und im Servostar gespeichert.

- Schreibe die gewünschte Referenzfahrtgeschwindigkeit in „iData.Config.RefSpeed“
- Setze iRequest.StartRef =1 -> Der Antrieb bewegt sich und oState.ReferenceActive =1
- Warte bis oState.ReferenceOk =1 und oState.ReferenceActive wieder =0
- Setze iStartRef = 0 – Der Antrieb ist referenziert

Mit iEnTimeout =1 und iData.Config.TO\_Reference xxx ms [TIME] wird die Referenzfahrt mit einem Timeout überwacht.

Start eines EEPROM oder RAM Fahrauftrags ( MotionTask)

Zusätzliche Voraussetzung: Der Servostar ist referenziert.

Mit der DriveGui werden Fahraufträge angelegt und im Servostar EEPROM gespeichert.

Nur die Parameter der RAM-Fahraufträge können mit der S7-PLC im Servostar sogar im Zustand “P4\_OperationEnabled” und während der Motor sich bewegt geändert werden (siehe ASCII - MTMUX).

- Schreibe die Nummer des Fahrauftrags in „iData.MotionTask.Number
- Setze iStartMotionTask =1 ( intern wird ein Impuls generiert )
- Warte bis oState.InPosition =0 und oState.MotionTaskActive =1
- Warte bis oState.InPosition =1 und oState.MotionTaskActive =0

Der Servostar hat den Fahrauftrag ausgeführt. Mit iEnTimeout =1 und iData.Config.TO\_Position xxx ms [TIME] wird der Fahrauftrag mit einem Timeout überwacht.

Istposition, Istdrehzahl (16bit) und Hersteller-Status werden dabei zyklisch im RCV.PZD-Kanal übertragen.

Start von dem Direktfahrauftrag ( RAM und hat die Nummer 0 )

Voraussetzung: Der Servostar ist referenziert.

- Schreibe 0 in iData.MotionTask.Number
- Schreibe Zielposition in iData.MotionTask.DirectPosition
- Schreibe Sollgeschwindigkeit in iData.MotionTask.DirectSpeed
- Schreibe Fahrauftragsart in iData.MotionTask.DirectType
- Setze iRequest.StartMotionTask =1 ( intern wird ein Impuls generiert )
- Warte bis oState.InPosition =0 und oState.MotionTaskActive =1
- Warte bis oState.InPosition =1 und oState.MotionTaskActive =0

Der Servostar hat den Fahrauftrag ausgeführt. Mit iEnTimeout =1 und iData.Config.TO\_Position xxx ms [TIME] wird der Fahrauftrag mit einem Timeout überwacht.

Istposition, Istdrehzahl (16bit) und Hersteller-Status werden dabei zyklisch im RCV.PZD-Kanal übertragen.

Während der MotionTask läuft kann bereits der nächste MotionTask mit seinem Datensatz an iData.MotionTask kopiert werden und mit Setze iRequest.StartMotionTask =1 ( Impuls ) dieser dann sofort gestartet werden.

Mit dem BLKMOV und Pointer kann der komplette Data.MotionTask [STRUCT] einfach kopiert werden.

Siehe “Switch ON and Start MotionTask timing diagram”

Ein gestarteter Fahrauftrag kann mit `iPauseMotionTask = 1` angehalten werden.  
Der Servostar bleibt in `oState.P4_OperationEnabled = 1`.

Ein gestarteter Fahrauftrag kann mit `iRequest.StopMotionTask = 1` gestoppt werden.  
Der Servostar wechselt in `oState.P3_ReadyForOperation = 1`.  
Ein gestoppter Fahrauftrag bleibt im Regler weiterhin aktiv, d.h. wenn der Regler wieder in den `oState.P4_OperationEnabled = 1` geschaltet wird, wird der Fahrauftrag fortgesetzt.  
Mit `iSwEnable = 0` wird der gestoppte Fahrauftrag gekillt, d.h. wenn der Regler nun wieder in den `oState.P4_OperationEnabled = 1` geschaltet wird, wird der Fahrauftrag nicht mehr fortgesetzt.  
Siehe: "Kill MotionTask timing diagram"

In dem GenCycleOB (OB1) ist ein Beispiel um Teile von einem Fahrauftrag zu parametrieren und mit `Axis_01_WRITE` (FB16) von der S7-PLC in den Servostar mit dem Parameter Kanal zu übertragen.

Hinweis: Mit dem ASCII-Parameter `INPT0 PNU1904` wird die Zeit in ms eingestellt werden, für die das Signal `oManufactState.InPositionWindow` nach dem Fahrsatzstart mindestens zurückgesetzt wird (vgl. Handbuch Kap. VII.1).  
So ist das Interface Zeitverhalten zwischen der S7-PLC und dem Servostar immer identisch.

#### Hinweis zu Positionen und Geschwindigkeiten – Fahrauftragsart (MotionTaskType) O C – PNU 1785

`0x0000` hex (bit 13 = 0 )  
Absolute Positionierung mit Vorgabe der Geschwindigkeit und Position in inkrementellen reglerinternen 32 Bit und 16 Bit Einheiten und Fahrprofil Trapez.

`0x2000` hex (bit 13 = 1 )  
Absolute Positionierung mit Vorgabe der Geschwindigkeit und Position in Benutzereinheiten und Fahrprofil Trapez.

`0x2003` hex (bits 0, 1, 13 = 1 )  
Relative Positionierung mit Vorgabe der Geschwindigkeit und Position in Benutzereinheiten und Fahrprofil Trapez.

`0x12000` hex ( bit 13 = 1 und bit 16 = 1 )  
Absolute Positionierung mit Vorgabe der Geschwindigkeit und Position in Benutzereinheiten und Fahrprofil Sinus<sup>2</sup>.

`0x10000` hex ( bit 16 )  
Das Bit 16 kann nicht direkt im PZD-Kanal gesetzt werden.  
Siehe: Aktivieren der Trajektorie/Profil Beschleunigung Sin<sup>2</sup> bei Direktfahrsatz Nr. 0

#### Hinweis zu Feedback mit Wake&Shake (W&S)

Die Istposition, Istgeschwindigkeit und Hersteller-Status werden zyklisch in den RCV.PZD Kanal geschrieben nachdem Betriebsart setzen ( `oState.OpModeP_Ok = 1` und `oOpModeP = xxx dec` ) und die Wake&Shake Prozedur ausgeführt wurde.

Bemerkung: `n14 = 1` Ermittlung von MPHASE (z.B. `FBTYPE=7`)  
Wird gesetzt beim Einschalten des Verstärkers. Wird gelöscht wenn die Endstufe freigegeben und MPHASE mit Wake&Shake ermittelt wurde.

Falls ein Fehler beim Betriebsart setzen auftritt dann bitte zuerst bei dem Servostar den Betrieb freigegeben ( `iSetOperationEnabled = 1` ) und damit wird die Wake&Shake Prozedur ausgeführt.  
Erst danach die Betriebsart setzen durchführen.

### Betriebsart Drehzahl digital

Die Betriebsart Drehzahl digital (oState.OpModeP\_Ok = 1 und oOpmodeP = 1 dez ) ist aktiviert und der Betrieb ist freigegeben (oState.P4\_OperationEnabled =1).

- Schreibe Sollgeschwindigkeit in iData.DigitalSpeed.Ncmd ( 16bit )
- Setze iRequest.StartNcmd =1  
Der Antrieb verfährt mit der vorgegebenen Sollgeschwindigkeit.  
Die Sollgeschwindigkeit kann dabei jederzeit verändert werden.
- Setze iRequest.StartNcmd =0
- Der Antrieb bremsst über die eingestellten Drehzahlrampen bis zum Stillstand ab.

Istposition, Istgeschwindigkeit (16bit ) und Hersteller-Status werden dabei zyklisch im RCV.PZD-Kanal übertragen.

Der Drehzahlwert ( 16bit ) wird dabei nach folgender Formel berechnet:

PNU1886 - Ncommand ( Ncmd16 = Ncmd \* 2<sup>15</sup> / VOSPD )

### Betriebsart Drehmoment digital

Die Betriebsart Drehmoment digital (oState.OpModeP\_Ok = 1 und oOpmodeP = -2 dez ) ist aktiviert und der Betrieb ist freigegeben (oState.P4\_OperationEnabled =1).

- Schreibe Sollstrom in iData.DigitalTorque.lcmd
- Setze iRequest.Startlcmd =1
- Der Antrieb prägt den vorgegebenen Sollstrom ein.
- Der Sollstrom kann dabei jederzeit verändert werden.
- Setze iRequest.Startlcmd =0 und der Antrieb gibt den Sollstrom 0 Ampere aus.

Istposition, Iststrom und Hersteller-Status werden dabei zyklisch im RCV.PZD-Kanal übertragen.

Der Stromwert wird dabei nach folgender Formel berechnet:

PNU1870 - Icommand ( I[A] = lcmd \* IpeakAmplifier[A] / 3280 )

Mit dem ASCII-Parameter ICMDVLIM - PNU 1989 kann die Drehzahl des Motors auf einen Maximalwert begrenzt werden. Damit wird ein Durchgehen des Motors bei zu geringer Belastung verhindert.

### Betriebsart Elektrisches Getriebe

Die Betriebsart Elektrisches Getriebe (oState.OpModeP\_Ok = 1 und oOpmodeP = -4 dez ) ist aktiviert und der Betrieb ist freigegeben (oState.P4\_OperationEnabled =1).

Istposition, Iststrom und Hersteller-Status werden dabei zyklisch im RCV.PZD-Kanal übertragen.



### Betriebsart Servopumpe digital – Drehzahl

nur für S300/S700 - siehe „Applikationsschrift – Servopumpe - s700\_servopumpe\_d.pdf“

Die Betriebsart Servopumpe digital (oState.OpModeP\_Ok = 1 und oOpmodeP = -7 dez ) ist aktiviert und der Betrieb ist freigegeben (oState.P4\_OperationEnabled =1).

Hinweis: Beim Aktivieren der Betriebsart -7 wird automatisch QENA auf 1 gesetzt. Beim Umschalten auf eine andere Betriebsart als -7, wird die Servopumpe abgeschaltet (QENA=0).

- Schreibe Solldruck in iData.DigitalPump.QPFRcmd
- ODER
- Schreibe Sollvolumenstrom ( Solldurchfluss ) in iData.DigitalPump.QFcmd
- Setze iRequest.StartQcmd =1
- Der Antrieb prägt den vorgegebenen Sollwert Druck / Volumenstrom ein.
- Der Sollwert Druck / Volumenstrom kann dabei jederzeit verändert werden.
- Setze iRequest.StartQcmd =0 und der Servostar gibt den Sollwert Druck / Volumenstrom 0 aus.

Istdruck, Istdurchfluss, Iststrom und Istposition werden dabei zyklisch im RCV.PZD-Kanal übertragen.

Mit diesen Betriebsarten sind die meisten Anforderungen in einer Maschine realisierbar.

Weitere Betriebsarten sind derzeit nicht im Axis\_01\_FB (FB14) implementiert, diese können jedoch einfach nachprogrammiert werden.

### Makro-Programm

Ein Makroprogramm kann im Servostar weitere Funktionen ausführen.

Zur Programmierung wird die Software MacroStar verwendet.

Im Parameterkanal können für den Datenaustausch zwischen der S7-PLC und dem Servostar die ASCII Parameter DPRVARxx und DPVxx für Macro- und PLC-Programme verwendet werden.

z. Bsp. DPRVAR1 - PNU 2022 (dec) IND = 1 (dec)

Im Echtzeitkanal gibt es zusätzlich iRequest.MacroInput

– siehe Axis\_01\_FB (FB14) – Network 88: #Axis.Send.PZD.STW.MacroInput

Für besondere Anwendungen kann eine Änderung und Erweiterung z.Bsp. vom Axis\_01\_FB (FB14) erforderlich sein.

Die gesamten SEND und RECEIVE Daten können ebenfalls vom Makroprogramm verwendet werden.

Damit der Compiler fehlerfrei durchläuft muss im MacroStar in der Datei variables.cfg eingetragen sein:

```
PROSTW,2,""  
PROZSW,2,""  
PROFIN0,2,""  
PROFIN1,2,""  
PROFIN2,2,""  
PROFIN3,2,""  
PROFIN4,2,""  
PROFIN5,2,""  
PROFIN6,2,""  
PROFIN7,2,""  
PROFIN8,2,""  
PROFIN9,2,""  
PROFOUT0,2,""  
PROFOUT1,2,""  
PROFOUT2,2,""  
PROFOUT3,2,""  
PROFOUT4,2,""  
PROFOUT5,2,""  
PROFOUT6,2,""  
PROFOUT7,2,""  
PROFOUT8,2,""  
PROFOUT9,2,""
```

Beispiel zur S7-PLC und Makro-Programmierung im S300/S700:

Der Motor soll endlos drehen und anschliessend direkt in einer definierten Position ruckfrei mit weicher  $\sin^2$ -Bremsrampe anhalten. Die Zielposition und VJOG Drehzahl wird entweder von der S7-PLC als DirectMotionTask-Parameter übergeben oder kann auch als DPRVAR1 und DPRVAR2 im S300/S700-EEPROM abgespeichert werden.

Der komplette Bewegungsablauf ist an der Maschine mit dem DriveGui-Oszilloskop zu kontrollieren.  
( Position, Schleppfehler, Iststrom, Istdrehzahl )

S300/S700-Parameter:

```
OPMODE 8 ( oOpmodeP = 2 dez )
PGEARI 3600
PGEARO 1
POSCNFG 1 (Axis type MODULO )
DREF 16
SRND 0
ERND 36000
DPRVAR1 18500
DPRVAR2 3000
DPRVAR3 73728
DPRVAR4 50
DPRVAR5 50
```

S7-PLC Program:

```
// Start iAck
// Start iInit
// Start iSetOpModeP =2
// Start iOpEnable
// Start Reference run
// Start DirektMotionTask to TargetPosition with VJOG-Velocity
```

Aktiviere das Macro programm mit Tag\_20 ( M200.0 )



## Makro program:

```

PROGRAM PLCINIT

LONG INP1:=0;
LONG INP2:=0;
LONG INP3:=0;
LONG VALUE1:= 1048576;
LONG VALUE2:= 1;
LONG VALUE3:= 35999;
LONG RESULT:= 1;
LONG TEMPVAR1:= 0;
LONG TEMPVAR2:= 0;

END_PROGRAM

//*****
PROGRAM PLCMAIN

END_PROGRAM

//*****

PROGRAM PLC250

// Profi-STW - Bit 9 - MacroInput
// Bit 9 0->1: JOGMOVE with VJOG-Velocity: iData.MotionTask.Direct.Speed or DPRVAR2
// Bit 9 1->0: STOP at TargetPosition: iData.MotionTask.Direct.Position or DPRVAR1 < ERND !!!
// O_ACC with DPRVAR4
// O_DEC with DPRVAR5

// O_P:= DPRVAR1; // TargetPosition
// O_V:= DPRVAR2; // VJOG-Velocity
O_C:= O_C | 0x12000; // DPRVAR3; // 73728dez <-> 0x12000hex: Sin^2 and SI-Units
O_ACC:= DPRVAR4;
O_DEC:= DPRVAR5;

IF O_V <> 0 THEN
    TEMPVAR1:= 7179; // GO
ELSE
    TEMPVAR1:= 8888; // XX
END_IF;

IF TEMPVAR1 = 7179 THEN
    TEMPVAR2:= PROSTW&0x200;
    IF TEMPVAR2 = 0x200 THEN
        INP2:= 0;
        INP3:= 0;
        IF INP1 = 0 THEN
            INP1:= 1;
            VJOGIO:=O_V; // Set VJOG speed
            SETPTR(TRJ,G_MOVEJOGIO); //Start moving
        END_IF;
    ELSE
        IF INP1 = 1 THEN
            INP2:= 1;
            VALUE2:= O_P;
            MULDIV(VALUE1,VALUE2,VALUE3,RESULT);
        END_IF;
        INP1:= 0;
        IF INP2 = 1 THEN
            IF INP3 = 0 THEN
                IF PFB < RESULT THEN
                    INP3:= 1;
                END_IF;
            END_IF;
            IF INP3 = 1 THEN
                IF PFB >= RESULT THEN
                    INP2:= 0;
                    INP3:= 0;
                    MOVEP_NR:=0;
                    SETPTR(TRJ,G_STARTMOVE);
                END_IF;
            END_IF;
        END_IF;
    END_IF;
END_IF;

END_PROGRAM

```

## Parameter lesen und schreiben:

Alle Parameter und Kommandos des Servostars sind über die Profibus PNU-Nummer ansprechbar. Diese werden in der ASCII Kommandoliste Objektreferenz der Reihe nach angezeigt. Außerdem findet sich die Nummer bei der Beschreibung der Kommandos/ Parameter im Feld „Profibus PNU“. Zusätzlich befindet sich im Handbuch eine Liste ausgewählter Parameternummern.

ASCII Objektreferenz		ASCII - Kommando		ACC	
Sprachumschaltung nach <b>englisch</b>		Syntax Senden		ACC [Data]	
sortiert nach:		Syntax Empfangen		ACC <Data>	
Kommandos Gruppen Objekt-Idr.		Type		Variable rw	
\		ASCII Format		Integer32	
ACC		DIM		>> ACCUNIT	
ACC_X		Bereich		3 ..126000	
ACCR		Default		31400	
ACCUNIT		Opmode		0, 1	
ACTFAULT		Verstärker Status		-	
ACTIVE		ab Firmware		1.0	
ACTRS232		Konfiguration		Nein	
ADC0_15		Funktionsgruppe		velocity loop	
ADDR		Kurzbeschreibung		Beschleunigungsrampe Drehzahlregelung	
AEIA		Beschreibung		Das Kommando ACC definiert die Beschleunigungsrampe des Drehzahlreglers. Die Beschleunigungsrampe ACC wird nur bei Sollwertsprüngen benutzt, die eine Erhöhung der Drehzahl nach sich ziehen (Beschleunigungsvorgang). Für den Bremsvorgang gilt der Parameter DEC.	
ALIAS		Vorhanden in		Setup Software	
AIH11NR		CANBus Objektnummer		3501 (hex)	
AIH11RANGE		PROFIBUS PIU		1601 (dec) IND = 1 (dec)	
AIH1TRIG		DPR		1 (dec)	
AIH2TRIG		Datentyp Bus/DPR		Integer16	
AICHFG		Wichtung			
ANDB		Letzte Bearbeitung der Seite REV		2.7	
AIIH1		EEPROM		Ja	

## Parameter schreiben

Voraussetzung: Parameterkanal PKW wird nicht bereits benutzt

- Schreibe Nummer in iPnu.Write.Number
  - Schreibe Index in iPnu.Write.Index
  - Schreibe Wert in iPnu.Write.Value
  - Setze iPnuWriteStart =1 ( Impuls )
  - Warte bis oState.PnuWriteOK
- Wenn oState.PnuWriteOk nicht =1 oder oState.PnuWriteError =1 dann war Parameter schreiben nicht erfolgreich. Fehlerursache siehe Receive-Telegramm Servostar.

## Parameter lesen

Voraussetzung: Parameterkanal PKW wird nicht bereits benutzt

- Schreibe Nummer in iPnu.Read.Number
  - Schreibe Index in iPnu.Read.Index
  - Schreibe Wert in iPnu.Write.Value
  - Setze iPnuReadStart =1
  - Warte bis oState.PnuReadOK
- Wenn oState.PnuReadOk nicht =1 oder oState.PnuReadError =1 dann war Parameter lesen nicht erfolgreich. Fehlerursache siehe Receive-Telegramm Servostar.
- Die gelesenen Daten werden ausgegeben in
    - oData.PnuRead.Number
    - oData.PnuRead.Index
    - oData.PnuRead.Value
- Damit kann verglichen werden ob die gelesenen Daten auch die angeforderten Daten sind.
- Parameter lesen kann ständig aktiviert sein z.Bsp. für einen zu überwachenden Istwert ( z.Bsp Iststrom PNU 1688 zur Ermittlung des Drehmoments ).

Im GenCycleOB (OB1) sind einige Beispiele für verschiedene PNUs.

Hinweis: Gleichzeitige Read und Write Anforderung am Axis\_01\_FB (FB14) generiert den Fehler MaMsg.ErrReadWrite



Die S7-PLC kann den Servostar parametrieren und diese Daten mit SAVE und COLDSTART ins EEPROM speichern.

Bei einigen Parametern ist nach einer Änderung ebenfalls ein SAVE und anschliessend COLDSTART erforderlich.

Diese Änderungen bitte immer im sicheren Zustand von der Servoachse durchführen.

Disable zuerst den Servostar:

Setze iSwEnable =0

Prüfe dass oState.P1\_SwitchOnInhibited =1 ( Einschaltsperrung ).

Setze den Hardware-Enable Eingang =0

Setze den Hardware-Eingang AS/STO-Enable =0

Starte dann eine Initialisierung ( Null-Telegramm )

Setze ilnit =1 ( intern wird ein Impuls generiert )

Warte bis oState.InitOk =1

Ändere nun die Parameter: WRITE ausführen

Anschliessend SAVE – Kommando - über einen READ ausführen.

Siehe ASCII-Parameter SAVE - PNU 1835

Hinweis: Die maximal erlaubte Anzahl der SAVE Schreibzyklen in ein EEPROM ist begrenzt !!!

Warte 5 Sekunden – Die Parameter werden nun in das Servostar-EEPROM gespeichert.

COLDSTART – Kommando - über einen READ ausführen.

Der Servostar bootet und macht einen Neustart / Wiederanlauf.

Die S7-PLC meldet dabei einen „Slave-Servostar – Kommunikationsfehler“

Warte bis der Servostar wieder im Profibus läuft: oState.CommunicationOK =1

Setze nun die Warnungen und Fehler zurück im Servostar und im S7-PLC Programm in den Bausteinen

iAck =1 (Acknowledge )

Nun mit READ und COMPARE überprüfen dass der Servostar die Werte in das EEPROM korrekt übernommen hat.

Im Servostar muss nun wieder die Betriebsart setzen und der Betrieb freigegeben durchgeführt werden.

## Zusätzliche Profibus-Funktionen

Der DP-Slave Servostar wird im Axis\_01\_FB (FB14) mit dem D\_ACT\_DP bei laufender PLC aktiviert bzw. deaktiviert.

- iActSlaveDevice =1, Servostar ist aktiviert und tauscht Daten über den Profibus aus.
  - iActSlaveDevice =0, Servostar ist deaktiviert und der Datenaustausch ist beendet. Die Ansprechüberwachung ( watchdog ) startet im Servostar (EXTWD) und generiert die Warning „n04 – Ansprechüberwachung aktiv ( watch dog )“.
- Der Servostar kann nun vom Profibus getrennt werden ohne eine Fehlermeldung an der S7-PLC.

Zudem wird die Profibuskommunikation mit DPRD\_DAT und DPWR\_DAT überwacht auf

- Konfigurationsfehler
- Slaveausfall
- Telegrammfehler

Die gesamten SEND und RCV-Datentelegramme werden im oAxis [STRUCT] ausgegeben  
Die Servostar-Zustandsmaschine wird im oState [STRUCT] ausgegeben

Bei relativer Positionierung können bei einem Abbruch eines Fahrauftrags die Daten in oData.Canceled [STRUCT] weggespeichert werden. Zu einem späteren Zeitpunkt ( z. Bsp. nach NOTAUS – manueller Eingriff - TürAUF / TürZU ) kann dann die S7-PLC damit die Fahrsatz-Daten korrigieren und den relativen Fahrauftrag mit korrigierten Daten zu Ende fahren.

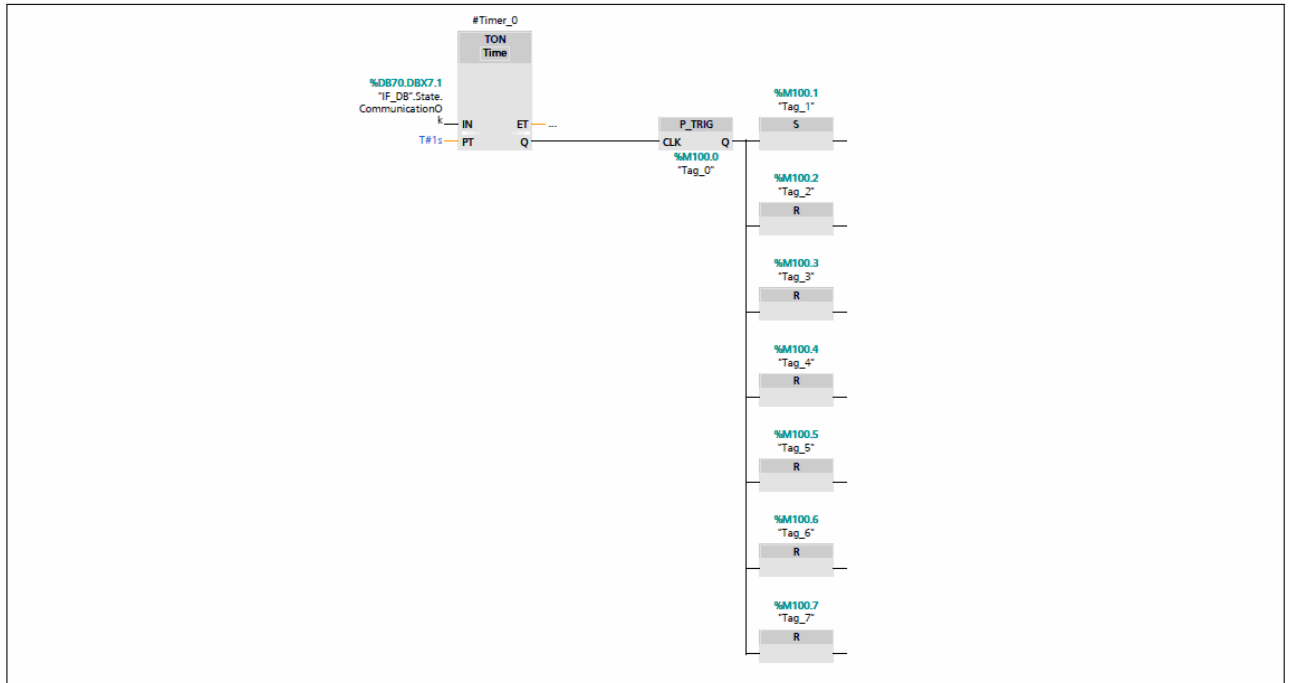
## Beobachtungs- und Forcetable

Mit der Watch table 1 können die Bausteine beobachtet und gesteuert werden.

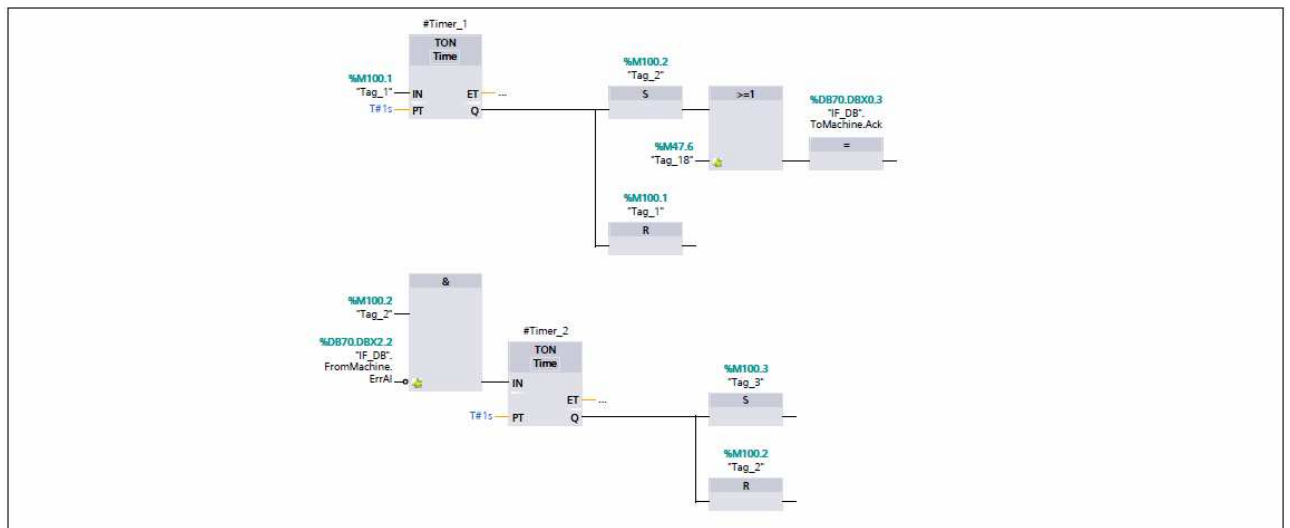
Sv14_v3c_V14_SP1_1516 ▶ PLC_1 [CPU 1516-3 PN/DP] ▶ Watch and force tables ▶ Watch table_1									
	Name	Adresse	Beobachtungswert	Anzeigeformat	Steuernwert		C...	Kommentar	
1	"IF_DB".ToMachine.Ack	%DB70.DBX0.3	Bool	<input type="checkbox"/> FALSE	FALSE	<input checked="" type="checkbox"/>	!	Acknowledge error	
2	"IF_DB".Init	%DB70.DBX28.6	Bool	<input type="checkbox"/> FALSE	FALSE	<input checked="" type="checkbox"/>	!		
3	"IF_DB".SetOpModeP	%DB70.DBX28.7	Bool	<input type="checkbox"/> FALSE	FALSE	<input checked="" type="checkbox"/>	!		
4	"IF_DB".SetOperationEnable	%DB70.DBX29.0	Bool	<input type="checkbox"/> FALSE	FALSE	<input checked="" type="checkbox"/>	!		
5	"IF_DB".Request.StartRef	%DB70.DBX4.0	Bool	<input type="checkbox"/> FALSE	FALSE	<input checked="" type="checkbox"/>	!	Start reference movement, static 1	
6	"IF_DB".Request.StartCmd	%DB70.DBX4.1	Bool	<input type="checkbox"/> FALSE	FALSE	<input checked="" type="checkbox"/>	!	Start I command digital torque, P4, static 1	
7	"IF_DB".Request.StartMotionTask	%DB70.DBX4.2	Bool	<input type="checkbox"/> FALSE	FALSE	<input checked="" type="checkbox"/>	!	Start motion task ( direct motion task =0 )...	
8	"IF_DB".Request.StopMotionTask	%DB70.DBX4.3	Bool	<input type="checkbox"/> FALSE	FALSE	<input checked="" type="checkbox"/>	!	Stop motion task, P4->P3, pulse 0->1	
9	"IF_DB".Request.CancelMotionTask	%DB70.DBX4.4	Bool	<input type="checkbox"/> FALSE	FALSE	<input checked="" type="checkbox"/>	!	Cancel motion task, P3, pulse 0->1	
10	"IF_DB".Request.StartNcmd	%DB70.DBX4.5	Bool	<input type="checkbox"/> FALSE	FALSE	<input checked="" type="checkbox"/>	!	Start N command digital speed, P4, static 1	
11	"IF_DB".Request.JogPlus	%DB70.DBX4.6	Bool	<input checked="" type="checkbox"/> TRUE	TRUE	<input checked="" type="checkbox"/>	!	Jog positive, static 1	
12	"IF_DB".Request.JogMinus	%DB70.DBX4.7	Bool	<input type="checkbox"/> FALSE	FALSE	<input checked="" type="checkbox"/>	!	Jog negative, static 1	
13	"IF_DB".Request.MacroInput	%DB70.DBX5.0	Bool	<input type="checkbox"/> FALSE	FALSE	<input checked="" type="checkbox"/>	!	MacroProgramInput, static 1, PROSTW & 0...	
14	"IF_DB".Request.StartQcmd	%DB70.DBX5.1	Bool	<input type="checkbox"/> FALSE	FALSE	<input checked="" type="checkbox"/>	!	Start Q command digital speed, P4, static 1	
15	"IF_DB".PauseMotionTask	%DB70.DBX28.0	Bool	<input type="checkbox"/> FALSE	FALSE	<input checked="" type="checkbox"/>	!		
16	"IF_DB".FastStopDisableAxis	%DB70.DBX28.2	Bool	<input type="checkbox"/> FALSE	FALSE	<input checked="" type="checkbox"/>	!		
17	"IF_DB".FastStop	%DB70.DBX28.3	Bool	<input type="checkbox"/> FALSE	FALSE	<input checked="" type="checkbox"/>	!		
18	"IF_DB".SwDisable	%DB70.DBX29.2	Bool	<input type="checkbox"/> FALSE	FALSE	<input checked="" type="checkbox"/>	!		
19						<input type="checkbox"/>			
20	"IF_DB".iData.Config.OpModeP	%DB70.DBX64	DEC+/-	2	2	<input checked="" type="checkbox"/>	!	PNU 930 (2 =MotionTask, 1 =DigitalSpeed...	
21	"IF_DB".iData.Config.JogSpeed	%DB70.DBW68	DEC	100	100	<input checked="" type="checkbox"/>	!	PNU 1889	
22	"IF_DB".iData.Config.RefSpeed	%DB70.DBW70	DEC	100	100	<input checked="" type="checkbox"/>	!	PNU 1896	
23	"IF_DB".iData.MotionTask.Number	%DB70.DBW72	DEC	0	0	<input checked="" type="checkbox"/>	!	0 =DirectMotionTask	
24	"IF_DB".iData.MotionTask.DirectSpeed	%DB70.DBW74	DEC	100	100	<input checked="" type="checkbox"/>	!	PNU 1791	
25	"IF_DB".iData.MotionTask.DirectPosition	%DB70.DBW78	DEC+/-	12345	12345	<input checked="" type="checkbox"/>	!	PNU 1790	
26	"IF_DB".iData.MotionTask.DirectType	%DB70.DBW82	Bin	2#0010_000...	2#0010_00...	<input checked="" type="checkbox"/>	!	PNU 1785	
27	"IF_DB".iData.DigitalSpeed.Ncmd	%DB70.DBW84	DEC+/-	547	547	<input checked="" type="checkbox"/>	!	PNU1886 - Ncommand ( Ncmd16 = Ncm...	
28	"IF_DB".iData.DigitalTorque.Icmd	%DB70.DBW86	DEC+/-	200	200	<input checked="" type="checkbox"/>	!	PNU1870 - Icommand ( I[A] = Icmd * Ipea...	
29	"IF_DB".iData.DigitalPump.QPRcmd	%DB70.DBW88	DEC+/-	43	43	<input checked="" type="checkbox"/>	!	1 <=> 10 mBar	
30	"IF_DB".iData.DigitalPump.QFcmd	%DB70.DBW90	DEC+/-	12	12	<input checked="" type="checkbox"/>	!	1 <=> 0,1 l/min	
31						<input type="checkbox"/>			
32	"IF_DB".FromMachine.ErrAl	%DB70.DBX2.2	Bool	<input type="checkbox"/> FALSE		<input type="checkbox"/>		Error or alarm at unit active	
33	"IF_DB".FromMachine.Warning	%DB70.DBX2.3	Bool	<input type="checkbox"/> FALSE		<input type="checkbox"/>		Warning at unit active	
34						<input type="checkbox"/>			
35	"IF_DB".OpModeP	%DB70.DBW26	DEC+/-	2		<input type="checkbox"/>			
36	"IF_DB".ActualPosition	%DB70.DBW14	DEC+/-	412908		<input type="checkbox"/>			
37	"IF_DB".ActualSpeed	%DB70.DBW18	DEC+/-	602		<input type="checkbox"/>			
38	"IF_DB".ActualCurrent	%DB70.DBW20	DEC+/-	0		<input type="checkbox"/>			
39	"IF_DB".ActualPressure	%DB70.DBW22	DEC+/-	0		<input type="checkbox"/>			
40	"IF_DB".ActualFlow	%DB70.DBW24	DEC+/-	0		<input type="checkbox"/>			

# Beispiel für eine Schrittkette „Initialisierung und Enablen Servostar“

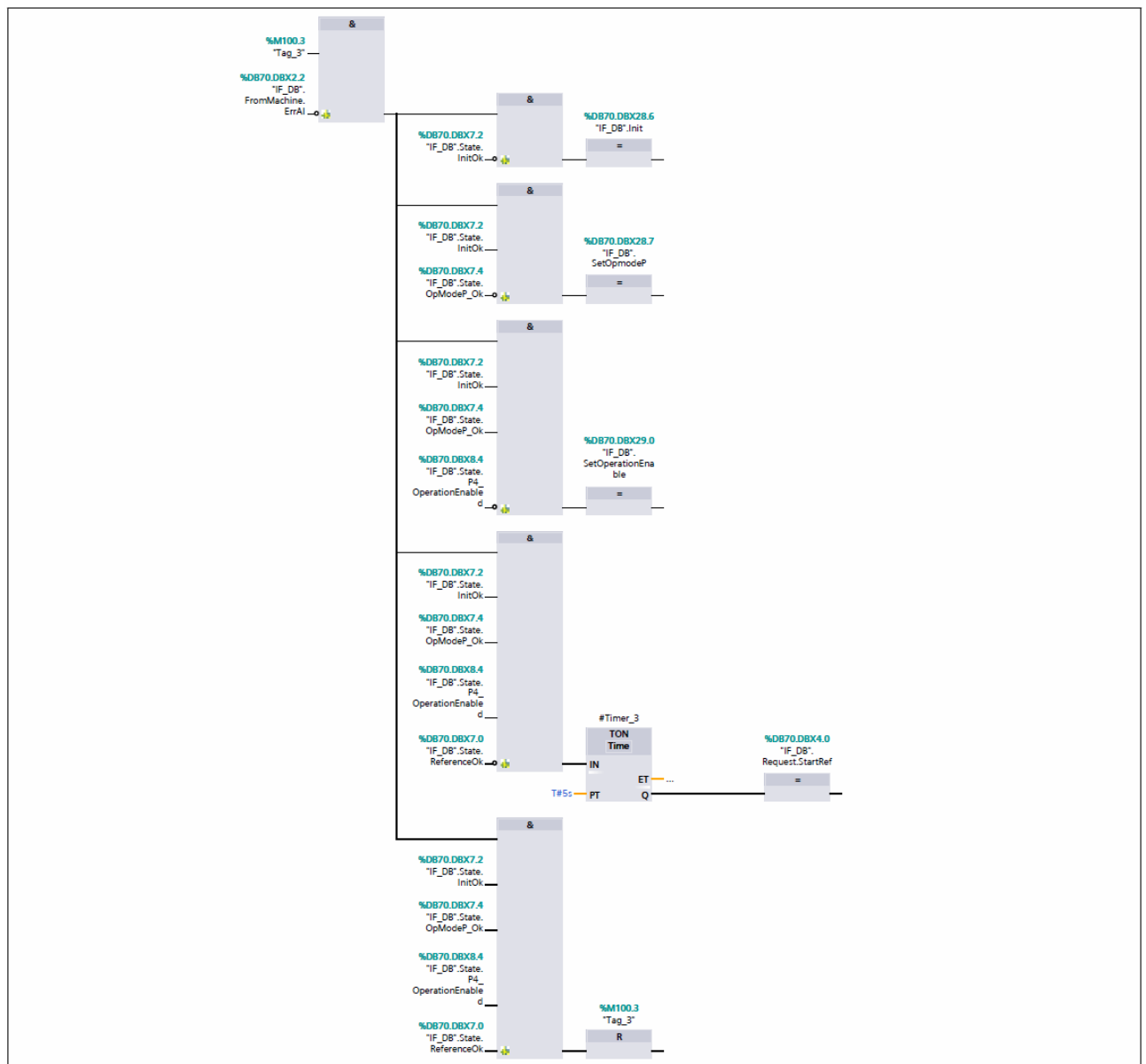
Network 1: Initialisation Sequence Enable and Referencing



Network 2: "IF\_DB".ToMachine.Ack



### Network 3: Sequence Enable and Referencing



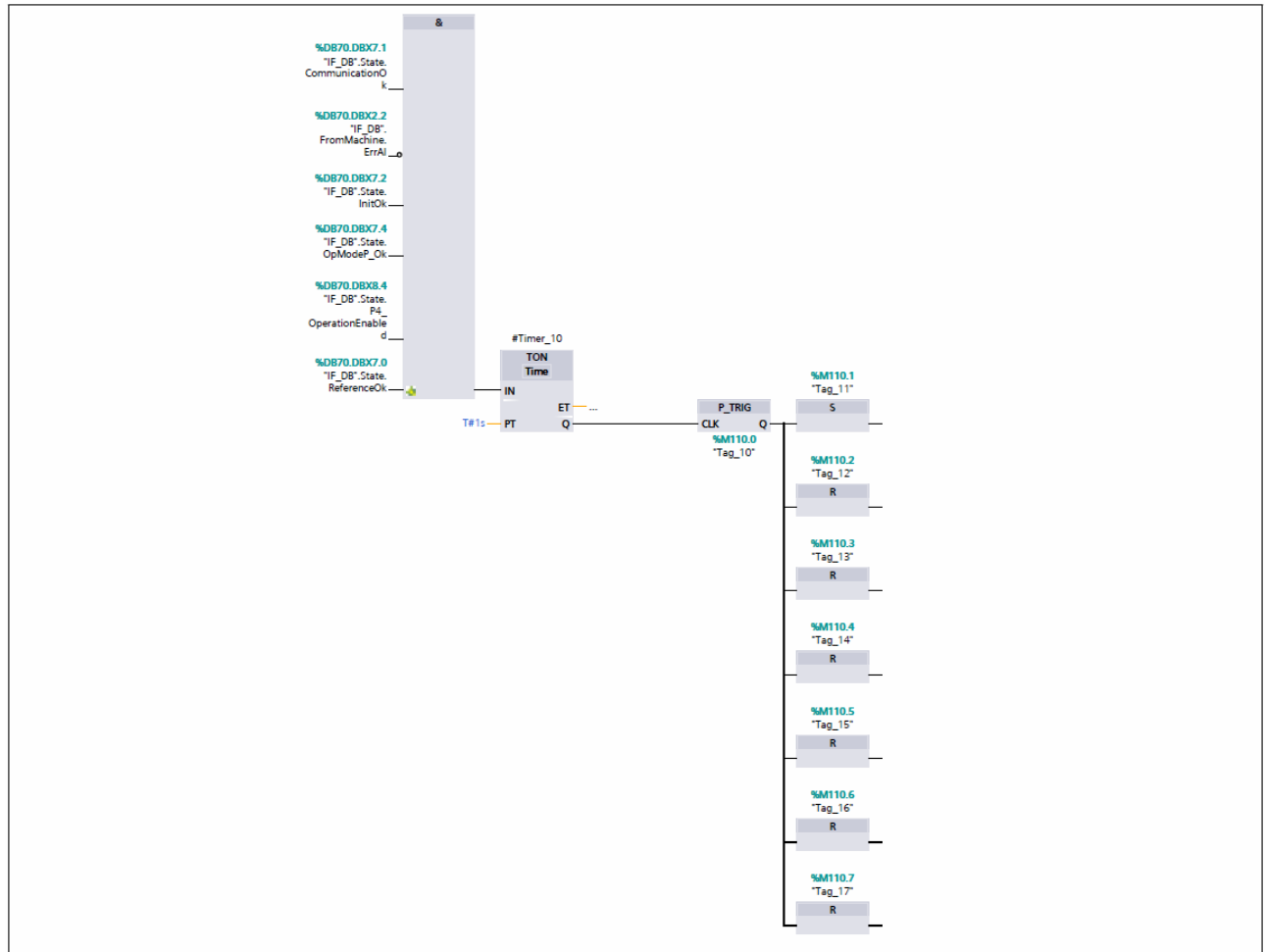
Timer\_3: Verzögerung ist erforderlich für z.Bsp. solange bis die Wake&Shake Prozedur beendet ist

Siehe "Switch ON and Start MotionTask timing diagram"

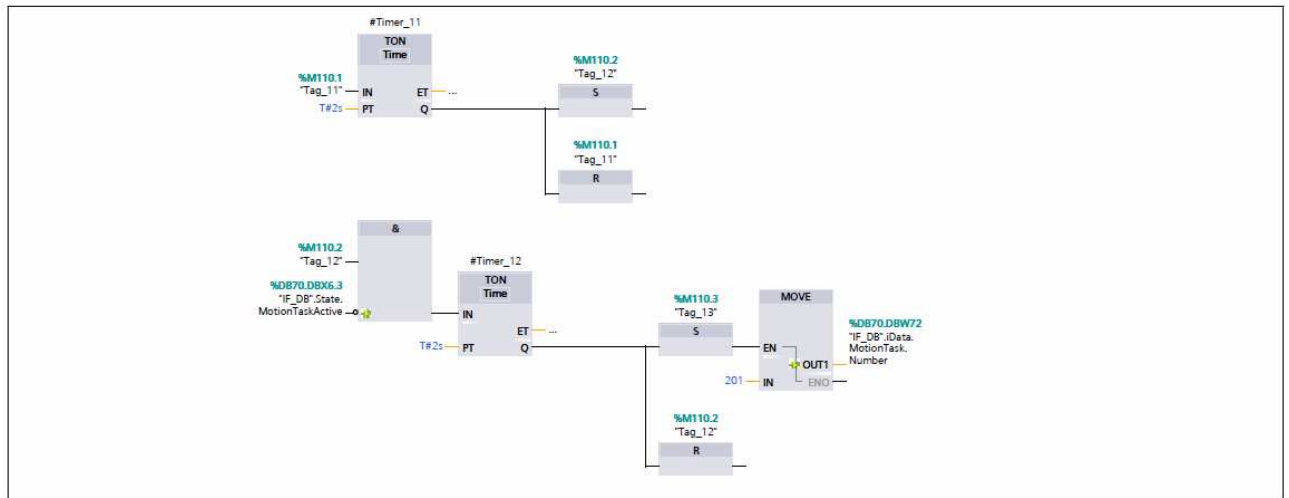
## Beispiel für eine Schrittkette „Starte Fahrsätze im Servostar“

```
// IF Ready then Write Data from PLC to Servostar with pulse from M110.1 and FB16
// and then Start RAM_MotionTask 201,210 and 211
// remark: the RAM_MotionTask 201 has a following RAM_MotionTask 202
```

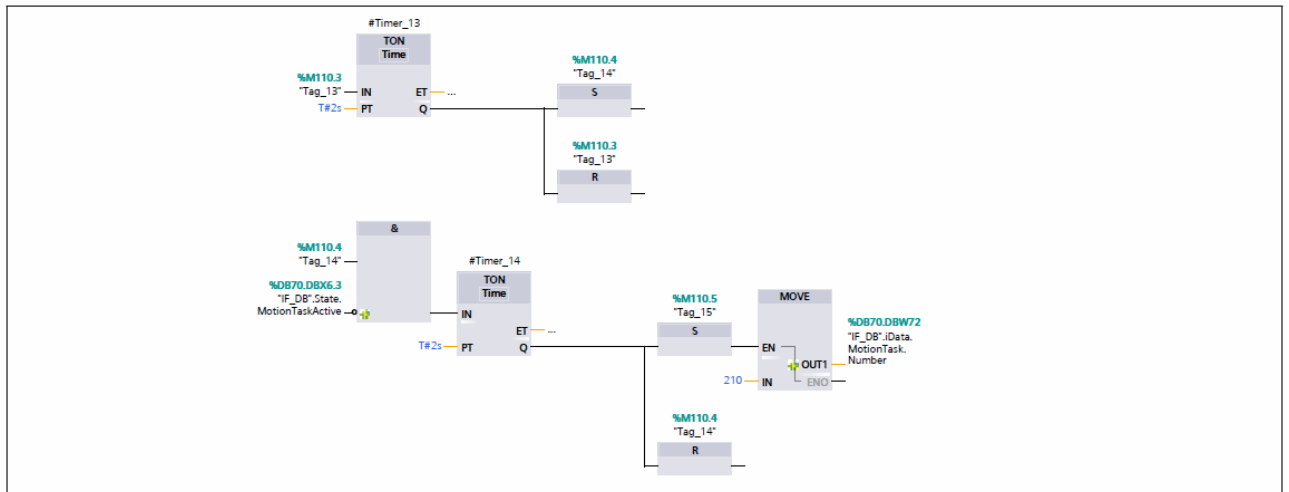
Network 4: Initialization Sequence Start Motion Tasks



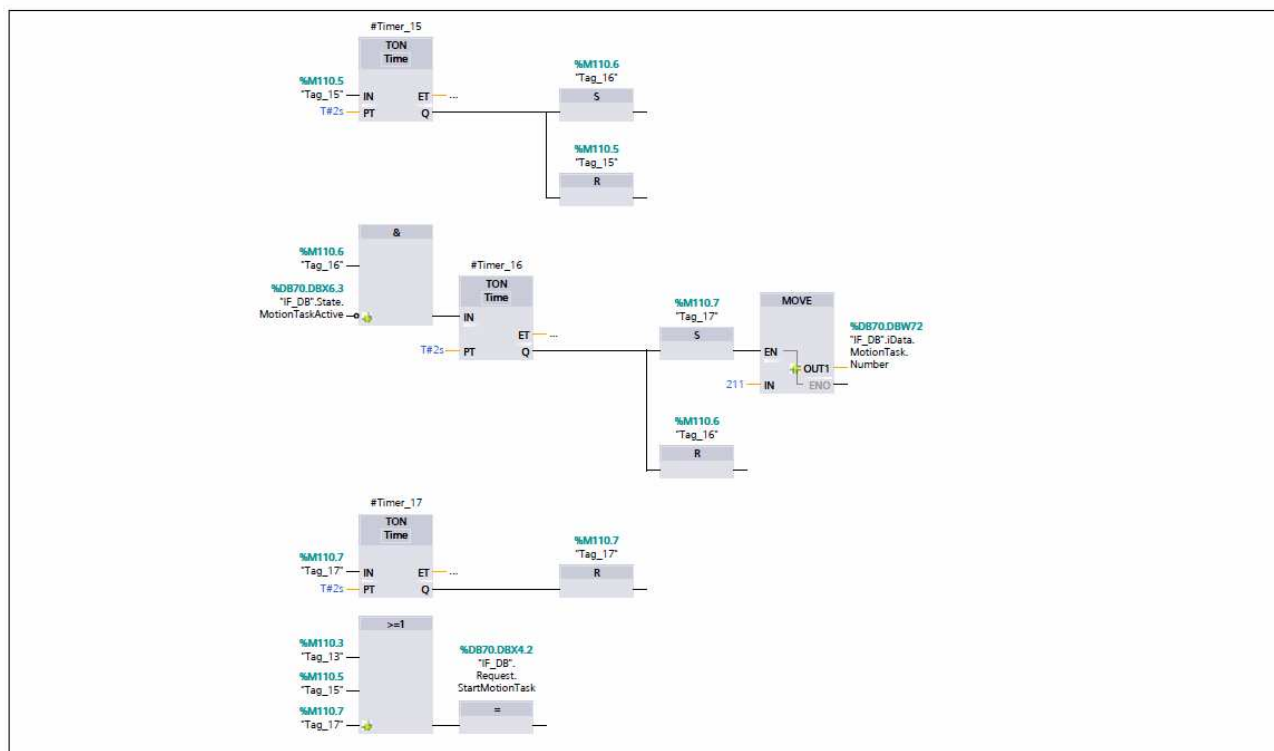
Network 5: "IF\_DB".Request.StartMotionTask No 201



Network 6: "IF\_DB".Request.StartMotionTask No 210



# Network 7: "IF\_DB".Request.StartMotionTask No 211



Siehe "Switch ON and Start MotionTask timing diagram"



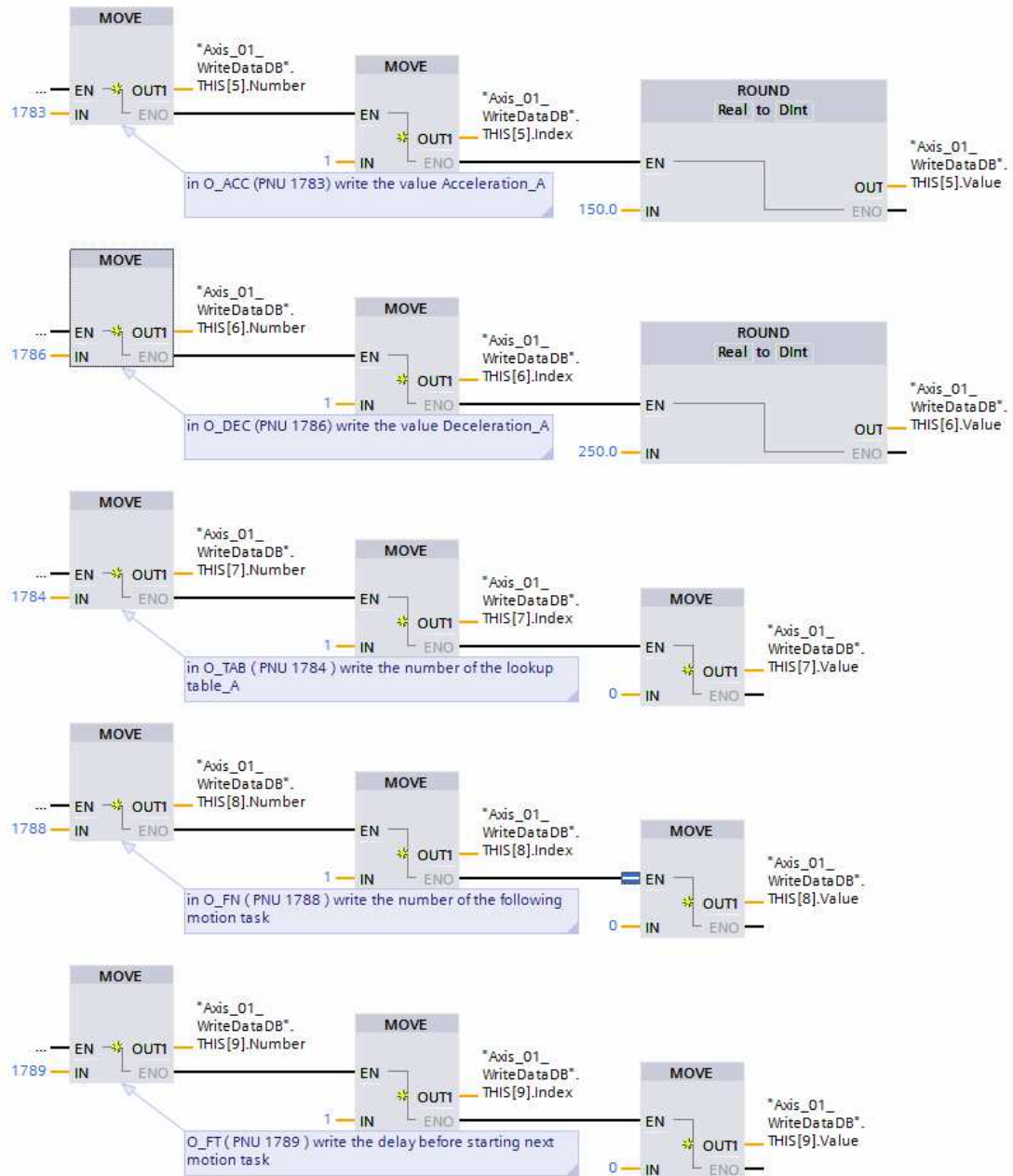
```
// ORDER 201 - RAM MotionTask
```

```
// PNU 1947 Index 1 - MTMUX
```

Figure 10 illustrates the structure of the MotionTask\_A program, showing four parallel task blocks, each corresponding to a specific PNU address (1947, 1790, 1791, and 1785). Each block contains a sequence of logic blocks:

- MOVE Block (Number):** Takes the PNU address as input and outputs the 'Number' value to the 'Axis\_01\_WriteDataDB'.
- MOVE Block (Index):** Takes the 'Number' value as input and outputs the 'Index' value to the 'Axis\_01\_WriteDataDB'.
- ROUND Block (Value):** Takes the 'Index' value as input and outputs the 'Value' to the 'Axis\_01\_WriteDataDB'. The 'Value' is also used to write to the PNU address (e.g., 'in O\_P (PNU 1790) write the value Position\_A').

The diagram shows the flow of data between these blocks and the external database, ensuring that the correct values are written to the PNU addresses for each task.



Hinweis zu ASCII Parameter MTMUX - PNU 1947 – Beispiel im GenCycleOB (OB1):

Mit der DriveGui sind manchmal nicht alle Werte sichtbar in der MotionTask Tabelle No. 201,210,211  
Dann einfach mit dem DriveGui Terminal die Werte überprüfen:

--> ORDER 201

201 -400 1000 -1 -1 -1 -1 -1 -1 400 -1

--> ORDER 210

210 -200 1500 -1 100 50 -1 -1 -1 -1 -1

--> ORDER 211

211 -500 850 -1 150 250 -1 -1 -1 -1 -1

-->

## Bausteine für Servostar-Parameter

Das S7-Projekt „Sv14\_v3c\_V14\_SP1\_1516“ enthält noch weitere Funktionsbausteine und Datenbausteine die ganze Datenbereiche in den Servostar schreiben, aus dem Servostar lesen und in der S7-PLC vergleichen.

Ein Datensatz besteht aus 3 Parametern:

- PNU - Parameternummer – gibt an um welchen Parameter es sich handelt  
z.Bsp PNU 1783 Anfahrzeit O\_ACC1.
- Index gibt an um was es sich bei dem übertragenen Wert handelt  
z.Bsp Index=1 - Istwert oder Index=3 oberer Grenzwert.
- Value – enthält den übertragenen Wert.

### Axis\_01\_WriteDataDB (DB20)

enthält die Daten zum schreiben mit dem Axis\_01\_Write (FB16) und dem Axis\_01\_FB (FB14) von der S7-PLC über den nicht echtzeitfähigen PKW-Parameterkanal im Profibus in den Servostar.

#### Deklaration vom Axis\_01\_WriteDataDB (DB20) – für 100 Werte

Sv14_v3c_V14_SP1_1516 > PLC_1 [CPU 1516-3 PN/DP] > Program blocks > Axis_01_WriteDataDB [DB20]								
Axis_01_WriteDataDB								
	Name	Data type	Start value	Retain	Accessible f...	Writa...	Visible in ...	Setpoint
1	Axis_01_WriteDataDB	Array[0..100] of *M_Axis_01_PnuStruct*		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

### Axis\_01\_ReadDataDB (DB21)

enthält die Daten zum lesen mit dem Axis\_01\_Read (FB17) und dem Axis\_01\_FB (FB14) von dem Servostar über den nicht echtzeitfähigen PKW-Parameterkanal im Profibus in die S7-PLC.

#### Deklaration vom Axis\_01\_ReadDataDB (DB21) – für 100 Werte

Sv14_v3c_V14_SP1_1516 > PLC_1 [CPU 1516-3 PN/DP] > Program blocks > Axis_01_ReadDataDB [DB21]								
Axis_01_ReadDataDB								
	Name	Data type	Start value	Retain	Accessible f...	Writa...	Visible in ...	Setpoint
1	Axis_01_ReadDataDB	Array[0..100] of *M_Axis_01_PnuStruct*		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Siehe GenCycleOB (OB1) und Watch table\_1

## Axis\_01\_Write (FB16)

steuert den Axis\_01\_FB (FB14) zum Daten schreiben von der S7-PLC in den Servostar

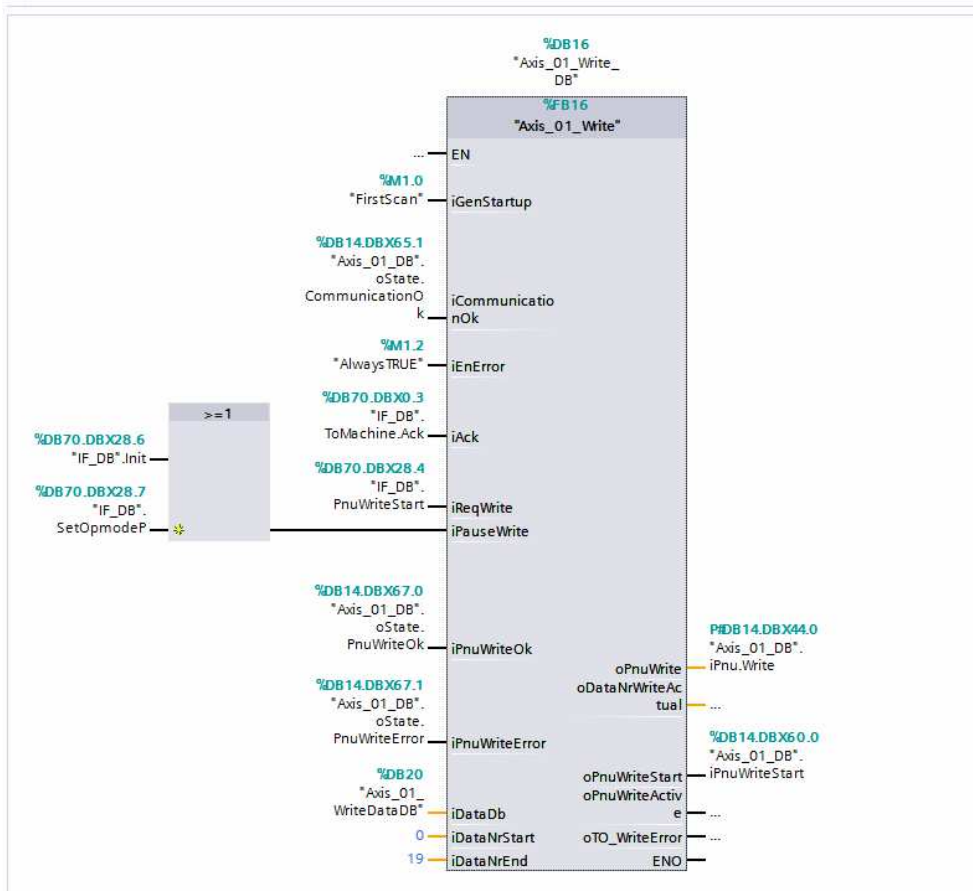
### Eingang- und Ausgang-Interface des Axis\_01\_Write (FB16)

VAR_INPUT		
iGenStartup	BOOL	Startup cycle after PLC startup
iCommunicationOk	BOOL	Axis communication Profibus is OK
iEnError	BOOL	Enable error messages
iAck	BOOL	Acknowledge WarningsErrors
iReqWrite	BOOL	RequestWrite
iPauseWrite	BOOL	PauseWrite (necessary for SetOpmode or InitAxis)
iPnuWriteOk	BOOL	PnuWriteOk =1 succesful
iPnuWriteError	BOOL	PnuWriteError =1 not succesful
iDataDBNr	WORD	WriteDataDBNumber
iDataNrStart	INT	DataNumberStart - first number from data to write
iDataNrEnd	INT	DataNumberEnd - last number from data to write

VAR_OUTPUT		
oPnuWrite	STRUCT	PnuWrite to Axis_FB actual with Number, Index, Value
Number	WORD	
Index	WORD	
Value	DWORD	
oDataNrWriteActual	INT	Data number is writing actual
oPnuWriteStart	BOOL	Request write start for Axis_FB
oPnuWriteActive	BOOL	Writing DataBlock is active
oTO_WriteError	BOOL	Timeout writing is active but not working

#### Network 3: Axis\_Write

Write: "Axis\_01\_WriteDataDB" from Data[0] until Data[19]



## Axis\_01\_Read (FB17)

steuert den Axis\_01\_FB (FB14) zum Daten lesen vom Servostar in die S7-PLC

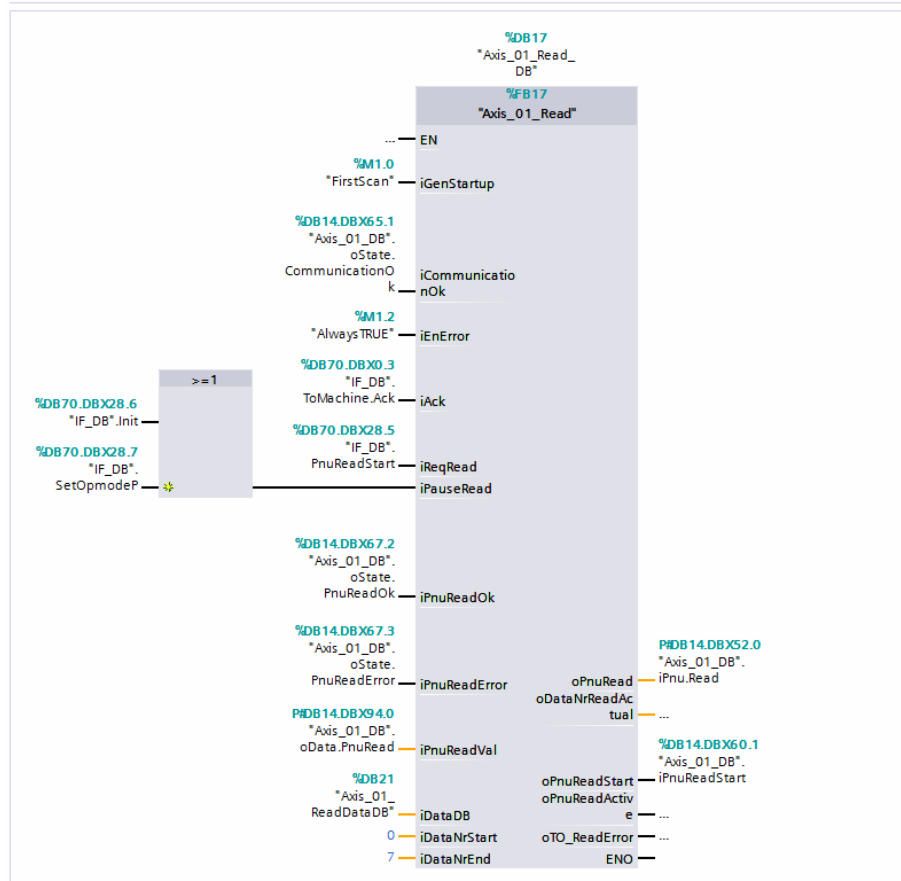
### Eingang- und Ausgang-Interface des Axis\_01\_Read (FB17)

VAR_INPUT		
iGenStartup	BOOL	Startup cycle after PLC startup
iCommunicationOk	BOOL	Axis communication Profibus is OK
iEnError	BOOL	Enable error messages
iAck	BOOL	Acknowledge WarningsErrors
iRequestRead	BOOL	RequestRead
iPauseWrite	BOOL	PauseRead (necessary for SetOpmode or InitAxis)
iPnuWriteOk	BOOL	PnuReadOk =1 succesful
iPnuWriteError	BOOL	PnuReadError =1 not succesful
iPnuReadVal	STRUCT	PnuRead from Axis_FB actual with Number, Index, Value
Number	WORD	
Index	WORD	
Value	DWORD	
iDataDBNr	WORD	ReadDataDBNumber
iDataNrStart	INT	DataNumberStart - first number from data to read
iDataNrEnd	INT	DataNumberEnd - last number from data to read

VAR_OUTPUT		
oPnuRead	STRUCT	PnuRead from Axis_FB actual with Number, Index
Number	WORD	
Index	WORD	
oDataNrReadActual	INT	Data number is reading actual
oPnuReadStart	BOOL	Request read start for Axis_FB
oPnuReadActive	BOOL	Reading DataBlock is active
oTO_ReadError	BOOL	Timeout reading is active but not working

#### Network 5: Axis\_Read

Read: "Axis\_01\_ReadDataDB" from Data[0] until Data[7]



Hinweis: Gleichzeitige Anforderung Read und Write am Axis\_01\_FB (FB14) generiert den Fehler oMaMsg.ErrReadWrite

## Axis\_01\_Compare (FB18)

vergleicht einen Datenbereich von den geschriebenen Daten vom Axis\_01\_WriteDataDB (DB20) mit den gelesenen Daten vom Axis\_01\_ReadDataDB (DB21).

Nur eine PNU [STRUCT] wird in einem S7-PLC Zyklus verglichen.

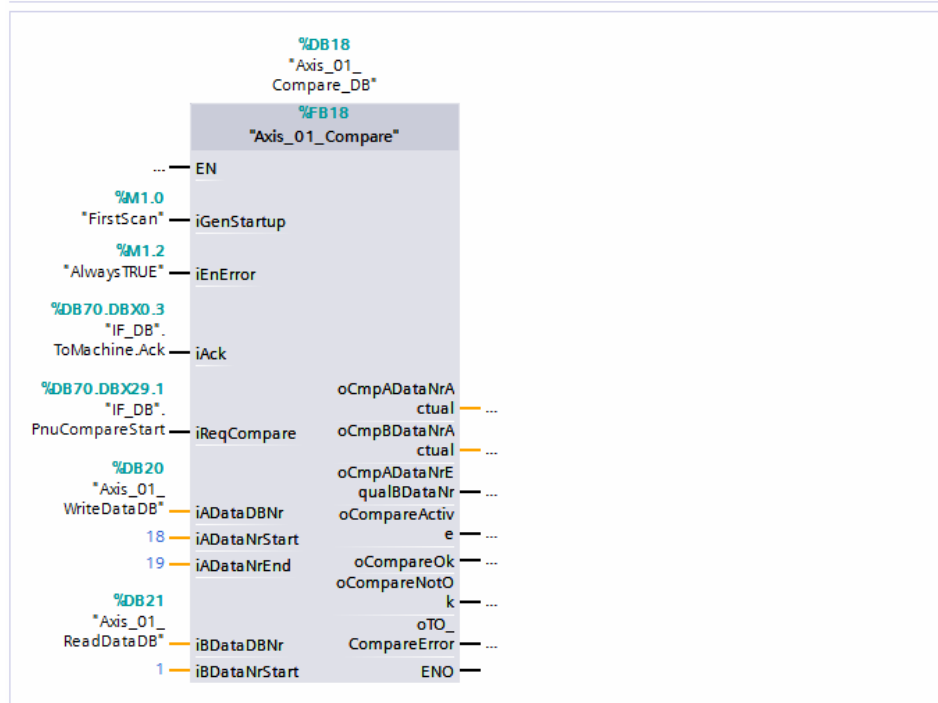
### Eingang- und Ausgang-Interface des Axis\_01\_CompareDB (FB18)

VAR_INPUT		
iGenStartup	BOOL	Startup cycle after PLC startup
iEnError	BOOL	Enable error messages
iAck	BOOL	Acknowledge WarningsErrors
iReqCompare	BOOL	Request compare
iADaDBNr	WORD	ADaDBNumber
iADaNrStart	INT	ADaNumberStart - first number from data to compare
iADaNrEnd	INT	ADaNumberEnd - last number from data to compare
iBdaDBNr	WORD	BdaDBNumber
iBdaNrStart	INT	BdaNumberStart - first number from data to compare

VAR_OUTPUT		
oCmpADaNrActual	INT	Compare A Data number is actual
oCmpBdaNrActual	INT	Compare B Data number is actual
oCmpADaNrEqualBdaNr	BOOL	Compare A Data number is equal B Data number
oCompareActive	BOOL	comparing DataBlocks is active
oCompareOk	BOOL	Compare is ok, datas are equal
oCompareNotOk	BOOL	Compare is not ok, datas are not equal
oTO_CompareError	BOOL	Timeout compare is active but not working

#### Network 6: Axis\_Compare

▼ iADaDBNr: "Axis\_01\_WriteDataDB" from Data[18] until Data[19]  
iBdaDBNr: "Axis\_01\_ReadDataDB" from Data[1] until Data[19-18+1=2]



Mit der Watch table\_1 können die Bausteine Axis\_01\_FB (FB14), Axis\_01\_Read (FB16), Axis\_01\_Write (FB17) und Axis\_01\_Compare (FB18) angesteuert und beobachtet werden.

## Tipps und Infos:

### Fehlermeldungen und Warnungen von den Bausteinen

sind in das Programm zu implementieren damit das PLC-Programm darauf reagieren kann.

oMaMsg	STRUCT	Error messages
ErrTO_Ref	BOOL	Error timeout reference
ErrTO_Pos	BOOL	Error timeout positioning
ErrNoReferenceSet	BOOL	Error if reference is not set and request
ErrWrongOpMode	BOOL	Error wrong operation mode selected and request
ErrActSlave	BOOL	Error activating slave
ErrCfgInput	BOOL	Error configuration input
ErrRcv	BOOL	Error receiving data
ErrCfgOutput	BOOL	Error configuration output
ErrSend	BOOL	Error sending data
ErrAxis	BOOL	Error from axis
ErrReadWrite	BOOL	Error request Read and Write together
ErrNotEnabled	BOOL	Error if not enabled and request
ErrRes_1_4	BOOL	
ErrRes_1_5	BOOL	
ErrRes_1_6	BOOL	
ErrCmd	BOOL	Error more than one request command active
oTO_WriteError	BOOL	Timeout writing is active but not working
oTO_ReadError	BOOL	Timeout reading is active but not working
oTO_CompareError	BOOL	Timeout reading is active but not working

### Override über Profibus

Siehe ASCII Parameter OVERRIDE

Mit Override-Funktion kann die Geschwindigkeit eines Fahrsatzes, der Referenzfahrt und des Tippbetriebes beeinflusst werden.

OVRIDE=0 Override-Funktion abgeschaltet

OVRIDE=3 Profibus Schnittstelle für die digitale Override-Funktion aktiviert.

Siehe ASCII Parameter DOVRIDE:

Beim Aktivieren einer digitalen OVERRIDE-Funktion wird mit Hilfe dieses Parameters der digitale Override-Faktor vorgegeben. Dabei gilt folgende Normierung:

DOVRIDE=0 Fahrsatzgeschwindigkeit = 0 %

DOVRIDE=8192 Fahrsatzgeschwindigkeit = 100 %

Die OVERRIDE – Funktion ist nicht möglich bei  $\sin^2$  – Rampen !



## Aktivieren der Trajektorie/Profil Beschleunigung $\sin^2$ bei Direktfahrsatz Nr. 0

Das Fahrprofil  $\sin^2$  ermöglicht ein weiches, ruckfreies Anfahren und Abbremsen. Damit wird die Mechanik geschont (Zahnräder, Spindeln). Pendelbewegungen etc. werden unterdrückt.

Bei Servostar S400/S600 siehe ASCII-Liste, dies wird hier nicht näher beschrieben.

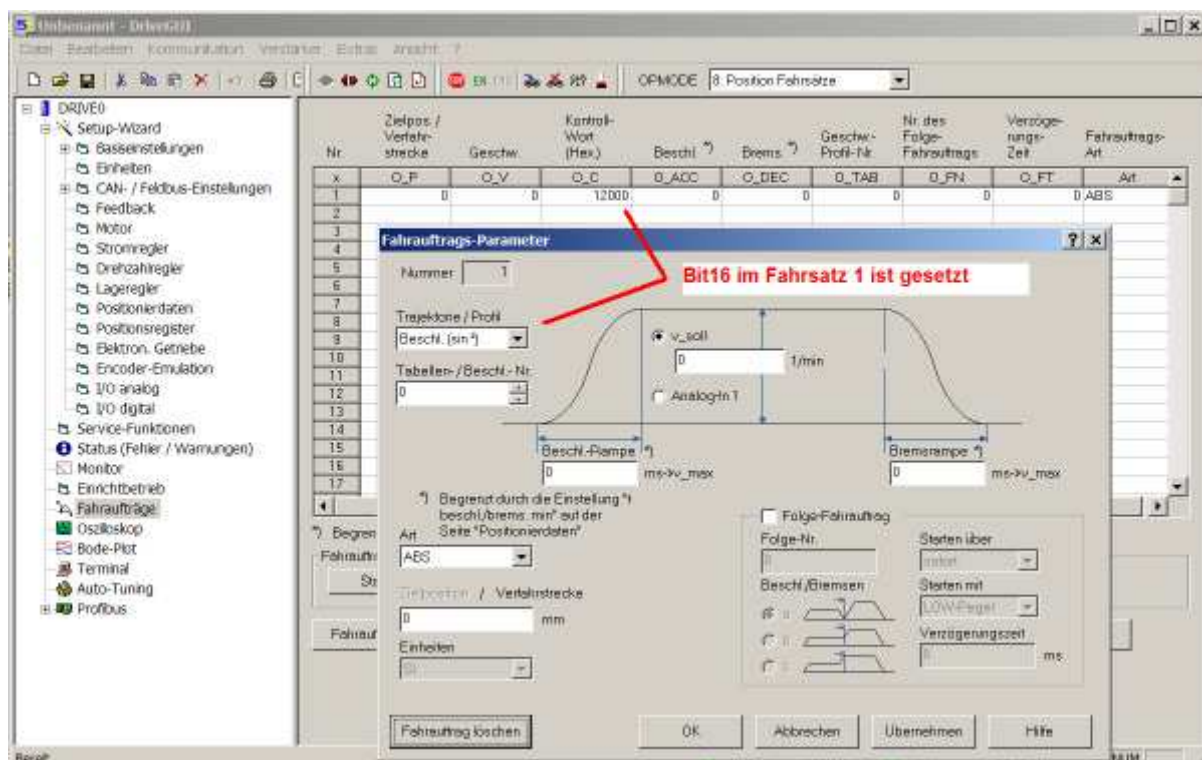
Bei Servostar S300/S700 – Profibus siehe ASCII-Parameter O\_C

Das Kommando O\_C ( Profibus PNU 1785 ) definiert die Fahrauftragsart für den Direktfahrsatz Nr. 0. Ist das Bit 16 von O\_C gesetzt, dann hat der Fahrsatz das  $\sin^2$ -Profil. Das Bit 9 muss auf 0 gesetzt werden. Von dem O\_C werden nur die Bits 0 bis 15 direkt im DirectMotionTaskType (PZD5) adressiert, deshalb muss das Bit 16 anderweitig im Servostar S300/S700 gesetzt werden.

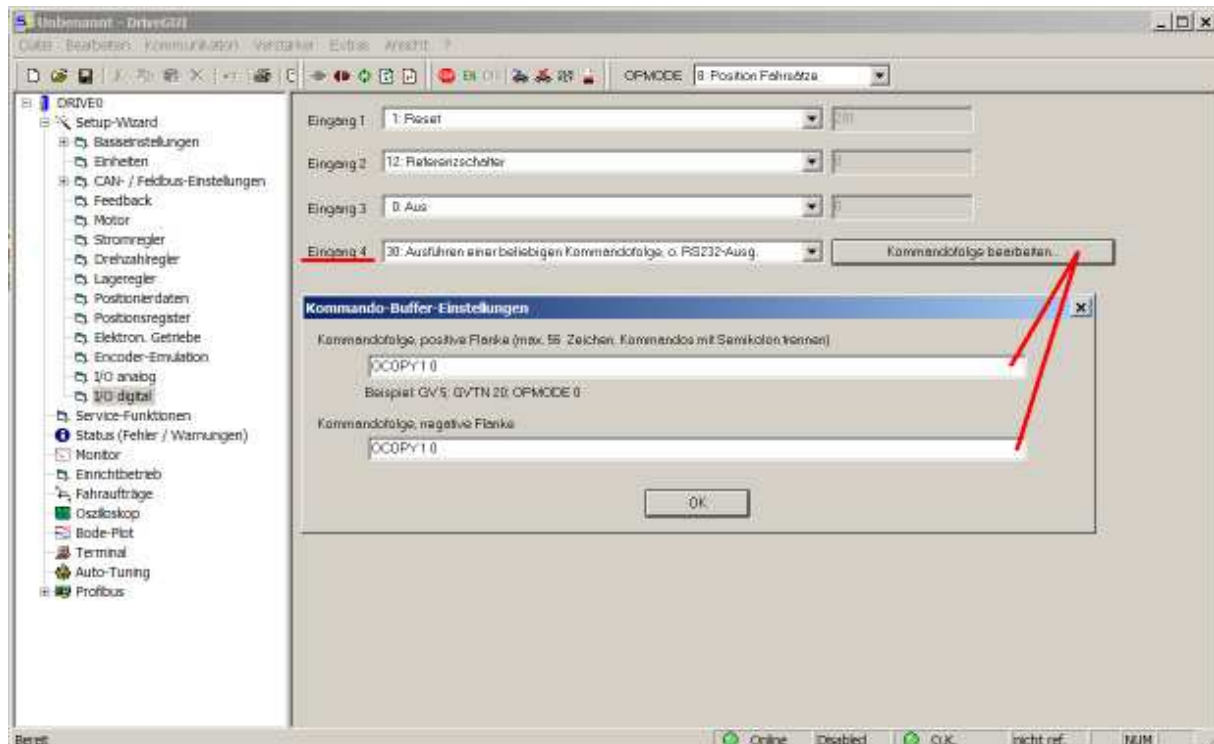
Das Kommando O\_C ist ein RAM-Parameter und kann deshalb nicht im Servostar gespeichert werden.

O\_C - Bit16 kann auf diese Weise gesetzt werden:

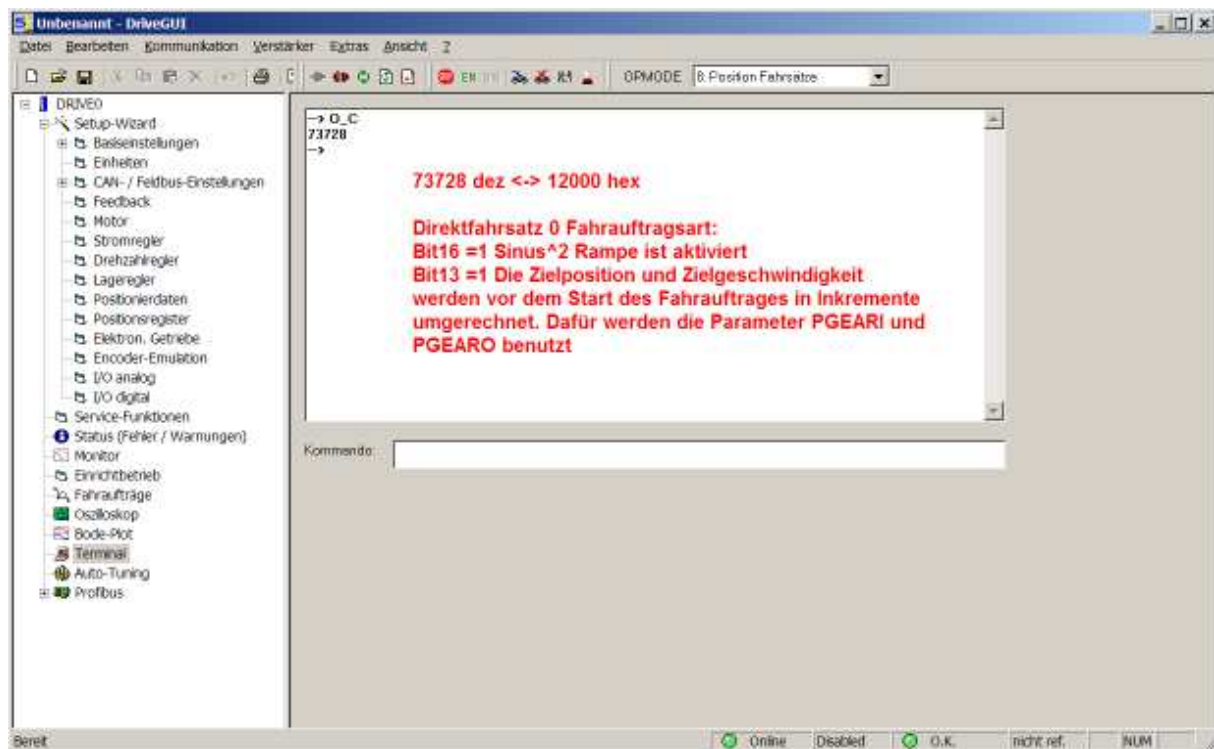
- 1.) Schreibe über den Parameterkanal PKW in die PNU 1785 den Wert 10000hex in den S300/S700.
- 2.) Nach dem EINSchalten der 24VDC – Versorgungsspannung kopiert sich der S300/S700 automatisch das Bit16 von einem EEPROM-Fahrsatz z.Bsp. Nr.1 in den Direktfahrsatz Nr. 0. Dadurch bleibt das S7-Programm unverändert und PNU 1785 Parameterdownload ist nicht erforderlich. Zusätzlich können auch die Beschleunigungs- O\_ACC und Bremsrampe O\_DEC für eine weiche Bewegung eingestellt werden. Bei Wert 0 gilt der Wert PTMIN bzw. jeweils die grösste Rampe.



Einem nicht benötigten Eingang ( nicht verdrahtet ) wird die OCOPY-Funktion von EEPROM Fahrsatz Nr.1 in den Direktfahrsatz Nr.0 zugewiesen.



SAVE und COLDSTART und Überprüfung mit DriveGui -Terminal den Inhalt von O\_C



### 3.) ASCII Kommandofolge

Mit dem Kommando ASCII INxHCMD / INxLCMD kann eine ASCII-Befehlsfolge definiert werden. Diese Befehlsfolge wird immer dann ausgeführt, wenn eine steigende Flanke auf dem mit der Funktion INxMODE=30 konfigurierten Eingang erkannt wird. Eine Befehlsfolge besteht aus einzelnen ASCII-Kommandos, die mit einem Semikolon (;) getrennt sind.  
Die maximale Länge dieser Befehlsfolge beträgt 56 Zeichen.

Mit DriveGui-Terminal:

```
IN4MODE 30
IN4HCMD O_C 73728; O_ACC 100; O_DEC 200
IN4LCMD O_C 73728; O_ACC 100; O_DEC 200
SAVE
COLDSTART
```

Überprüfung mit DriveGui-Terminal:

```
ORDER 0
0 0 0 73728 100 200 0 0 0 0
```

### 4. ) Makro Programm

Beispiel:

```
O_C:= O_C | 0x12000; // DPRVAR3; // 73728dez <-> 0x12000hex: Sin^2 and SI-Units
```

Überprüfung mit with DriveGui-Terminal:

```
ORDER 0
0 0 0 73728 0 0 0 0 0 0
```

Hinweis: Bei der Überprüfung mit dem DriveGui-Terminal kann auch dieses Ergebnis mit derselben Funktionalität angezeigt werden:

```
ORDER 0
0 -1 -1 73728 -1 -1 -1 -1 -1 -1
```

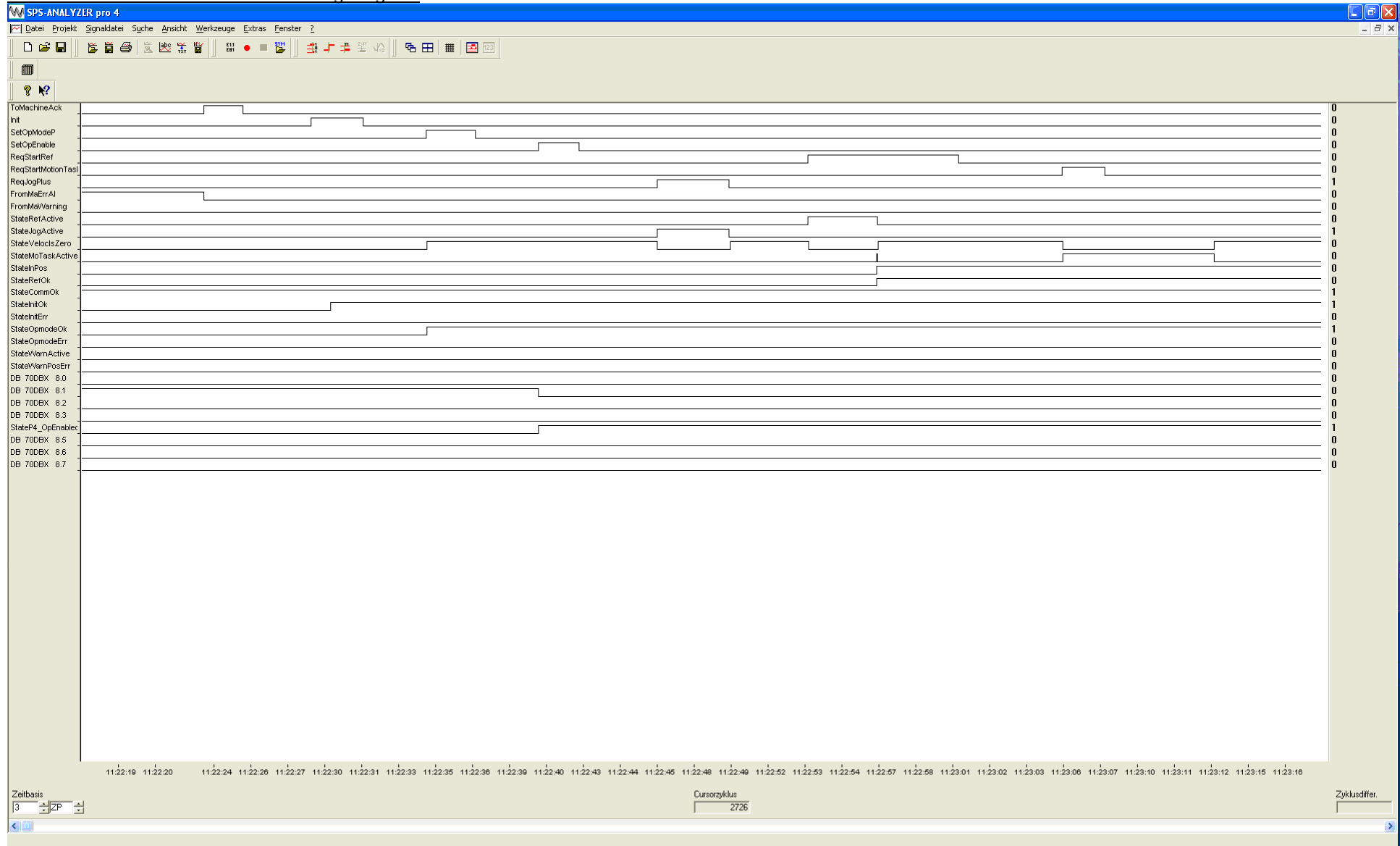
### Bitcodierung der Fehler – PNU 1001 - ERRCODE

Axis_PNU1001	MD 1000	DWORD	Fehler
Axis_PNU1001_Bit31_F32	M 1000.7	BOOL	Systemfehler
Axis_PNU1001_Bit30_F31	M 1000.6	BOOL	Fehler SafetyCard
Axis_PNU1001_Bit29_F30	M 1000.5	BOOL	Fehler Notstop Time-out
Axis_PNU1001_Bit28_F29	M 1000.4	BOOL	Slotkartenfehler
Axis_PNU1001_Bit27_F28	M 1000.3	BOOL	Fehler Synchronisation EtherCAT
Axis_PNU1001_Bit26_F27	M 1000.2	BOOL	Fehler STO
Axis_PNU1001_Bit25_F26	M 1000.1	BOOL	Fehler bei Referenzfahrt - Hardware-Endschalter
Axis_PNU1001_Bit24_F25	M 1000.0	BOOL	Kommutierungsfehler
Axis_PNU1001_Bit23_F24	M 1001.7	BOOL	Fehler Warnung in Fehler gewandelt
Axis_PNU1001_Bit22_F23	M 1001.6	BOOL	Fehler in der CAN – Kommunikation
Axis_PNU1001_Bit21_F22	M 1001.5	BOOL	Fehler reserviert
Axis_PNU1001_Bit20_F21	M 1001.4	BOOL	Fehler Handling Error
Axis_PNU1001_Bit19_F20	M 1001.3	BOOL	Fehler Slot-Error
Axis_PNU1001_Bit18_F19	M 1001.2	BOOL	Fehler Einbruch der Zwischenkreisspannung
Axis_PNU1001_Bit17_F18	M 1001.1	BOOL	Fehler Ballast ( defekter Ballasttransistor )
Axis_PNU1001_Bit16_F17	M 1001.0	BOOL	Fehler A/D-Wandler
Axis_PNU1001_Bit15_F16	M 1002.7	BOOL	Fehler Netz-BTB
Axis_PNU1001_Bit14_F15	M 1002.6	BOOL	Fehler I2tmax überschritten
Axis_PNU1001_Bit13_F14	M 1002.5	BOOL	Fehler Endstufe: Erdschluss, Motorkurzschluss oder Ballastkurzschluss
Axis_PNU1001_Bit12_F13	M 1002.4	BOOL	Fehler Umgebungstemperatur
Axis_PNU1001_Bit11_F12	M 1002.3	BOOL	Fehler reserviert
Axis_PNU1001_Bit10_F11	M 1002.2	BOOL	Fehler Bremse
Axis_PNU1001_Bit9_F10	M 1002.1	BOOL	Kabelbruch ROD Schnittstelle oder Zeitproblem Reglerbooten Master Slave
Axis_PNU1001_Bit8_F09	M 1002.0	BOOL	Fehler EEPROM
Axis_PNU1001_Bit7_F08	M 1003.7	BOOL	Fehler Überdrehzahl
Axis_PNU1001_Bit6_F07	M 1003.6	BOOL	Fehler interne Versorgungsspannungen
Axis_PNU1001_Bit5_F06	M 1003.5	BOOL	Fehler Motortemperatur
Axis_PNU1001_Bit4_F05	M 1003.4	BOOL	Fehler Unterspannung
Axis_PNU1001_Bit3_F04	M 1003.3	BOOL	Feedback-Fehler
Axis_PNU1001_Bit2_F03	M 1003.2	BOOL	Schleppfehler bei Ausführung der externen Trajektorie
Axis_PNU1001_Bit1_F02	M 1003.1	BOOL	Fehler Überspannung
Axis_PNU1001_Bit0_F01	M 1003.0	BOOL	Fehler Kühlkörpertemperatur

# Bitcodierung der Warnungen und Herstellerspezifisches Statusregister – PNU 1002 - DRVSTAT

Axis_PNU1002	MD 1004	DWORD	Warnungen UND herstellerspezifisches Statusregister
Axis_PNU1002_Bit31	M 1004.7	BOOL	Fehler steht an
Axis_PNU1002_Bit30	M 1004.6	BOOL	Endstufe freigegeben
Axis_PNU1002_Bit29	M 1004.5	BOOL	Sicherheitsrelais hat angesprochen (STO)
Axis_PNU1002_Bit28	M 1004.4	BOOL	Drehzahl = 0
Axis_PNU1002_Bit27	M 1004.3	BOOL	-
Axis_PNU1002_Bit26	M 1004.2	BOOL	Initialisierung beendet ( interne Initialisierung Verstärkers abgeschlossen)
Axis_PNU1002_Bit25	M 1004.1	BOOL	Position 4 erreicht (s.o.)
Axis_PNU1002_Bit24	M 1004.0	BOOL	Position 3 erreicht (s.o.)
Axis_PNU1002_Bit23	M 1005.7	BOOL	Position 2 erreicht
Axis_PNU1002_Bit22	M 1005.6	BOOL	Position 1 erreicht
Axis_PNU1002_Bit21	M 1005.5	BOOL	-
Axis_PNU1002_Bit20	M 1005.4	BOOL	Positions latch erfolgt
Axis_PNU1002_Bit19	M 1005.3	BOOL	In Position
Axis_PNU1002_Bit18	M 1005.2	BOOL	Aktuelle Position = Home Position ( Referenzschalter ist belegt)
Axis_PNU1002_Bit17	M 1005.1	BOOL	Referenzpunkt gesetzt ( nach einer Referenzfahrt bzw. Absolutwertgeber )
Axis_PNU1002_Bit16	M 1005.0	BOOL	Fahrauftrag aktiv - Fahrsatz, Tippbetrieb, Referenzfahrt)
Axis_PNU1002_Bit15_n16	M 1006.7	BOOL	Warnung 16: Reserve
Axis_PNU1002_Bit14_n15	M 1006.6	BOOL	Warnung 15: Geschwindigkeits-Strom Tabelle INXMODE 35 Fehler
Axis_PNU1002_Bit13_n14	M 1006.5	BOOL	Warnung 14: SinCos Kommutierung nicht vollzogen
Axis_PNU1002_Bit12_n13	M 1006.4	BOOL	Warnung 13: Erweiterungskarte arbeitet nicht ordnungsgemäß
Axis_PNU1002_Bit11_n12	M 1006.3	BOOL	Warnung 12: HIPERFACE® oder EnDat®: Motordefaultwerte wurden geladen
Axis_PNU1002_Bit10_n11	M 1006.2	BOOL	Warnung 11: Endschalter NSTOP betätigt
Axis_PNU1002_Bit9_n10	M 1006.1	BOOL	Warnung 10: Endschalter PSTOP betätigt
Axis_PNU1002_Bit8_n09	M 1006.0	BOOL	Warnung 9: Beim Fahrauftrag-Start war kein Referenzpunkt gesetzt
Axis_PNU1002_Bit7_n08	M 1007.7	BOOL	Warnung 8: Ein fehlerhafter Fahrauftrag wurde gestartet
Axis_PNU1002_Bit6_n07	M 1007.6	BOOL	Warnung 7: Software-Endschalter 2 überschritten
Axis_PNU1002_Bit5_n06	M 1007.5	BOOL	Warnung 6: Software-Endschalter 1 überschritten
Axis_PNU1002_Bit4_n05	M 1007.4	BOOL	Warnung 5: Netzphase fehlt
Axis_PNU1002_Bit3_n04	M 1007.3	BOOL	Warnung 4: Ansprechüberwachung (Feldbus) aktiv
Axis_PNU1002_Bit2_n03	M 1007.2	BOOL	Warnung 3: eingestelltes Schleppfehler-Fenster überschritten
Axis_PNU1002_Bit1_n02	M 1007.1	BOOL	Warnung 2: eingestellte Bremsleistung erreicht
Axis_PNU1002_Bit0_n01	M 1007.0	BOOL	Warnung 1: I <sup>2</sup> t-Meldeschwelle überschritten

## Switch ON and Start MotionTask timing diagram



Kill MotionTask timing diagram

