

Bausteine für die Simatic S7 und den S300/S400/S600/S700

Inhaltsverzeichnis

| | |
|--|---|
| Hinweise für das S7-Projekt „Sv14_Awl“ für den S300/S400/S600/S700 | 2 |
| Bausteinübersicht | 6 |
| Anmerkungen | 6 |
| Zusätzliche Dateien | Fehler! Textmarke nicht definiert. |
| Eingang- und Ausgangsschnittstelle des FB14 - Axis_01_FB | 7 |
| FB14 - Axis_01_FB | 7 |
| DB 70 - IF_TP170 - Interface DB Touch Panel TP170 | 7 |
| Gesamtübersicht der Schnittstelle vom FB14 | 8 |
| Inbetriebnahme FB14 Axis_01_FB | 12 |
| S7-SPS - AUS/EIN-Schalten | 12 |
| S7-SPS - Start/Stop-Schalten | 12 |
| iAck =1 | 12 |
| Initialisierung | 12 |
| Betriebsart setzen (Der FB14 ist für die Betriebsart Positionieren programmiert) | 12 |
| Betrieb freigeben | 12 |
| Tippbetrieb | 12 |
| Referenzfahrt | 12 |
| Start eines EEPROM oder RAM Fahrauftrags | 13 |
| Start eines Direktfahrauftrags (hat die Nummer 0) | 13 |
| Fahrauftragsarten (PNU 1785 - häufige Werte vgl. Handbuch Kap. IV 2.5.3) | 13 |
| Parameter schreiben | 14 |
| Parameter lesen | 14 |
| Zusätzliche Profibus-Funktionen | 14 |
| Beispiel für eine Schrittkette „Initialisierung und Enablen Servostar“ | 15 |
| Beispiel für eine Schrittkette „Starte Fahrsätze im Servostar“ | 16 |
| Bausteine für Servostar-Parameter | 18 |
| DB 20 - Axis_01_WriteDataDB | 18 |
| DB 21 - Axis_01_ReadDataDB | 18 |
| FB16 Axis_01_Write | 19 |
| FB17 Axis_01_Read | 20 |
| FB18 Axis_01_CompareDB | 21 |

Hinweise für das S7-Beispielprojekt „Sv14_Awl“ für den S300/S400/S600/S700

Das S7-Beispielprojekt „Sv14_Awl“ ist lediglich ein Beispiel, wie der S300/S400/S600/S700 in ein S7-Projekt integriert werden kann.

Ein Anspruch auf Vollständigkeit und Richtigkeit im S7-Projekt „Sv14_Awl“ und der Dokumentation wird nicht gewährleistet.

Änderungen durch Danaher Motion können jederzeit ohne Ankündigung erfolgen.

Für die Umsetzung der Applikation in eine reale Anwendung ist der Nutzer allein verantwortlich.

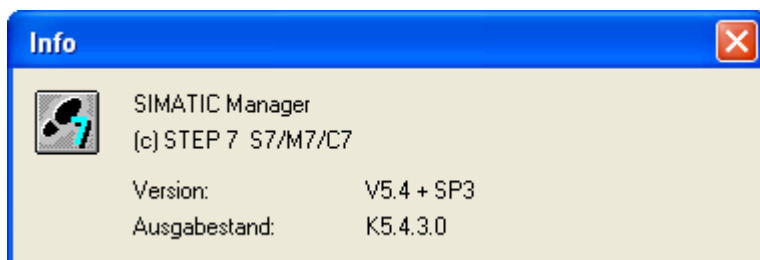
Gute Kenntnisse in S7, der Steuerungs- und Antriebstechnik werden vorausgesetzt.

Alle Verweise auf das Handbuch beziehen sich auf die technische Beschreibung

„Kommunikationsprofil PROFIBUS DP für SERVOSTAR® 300/400/600/700“.

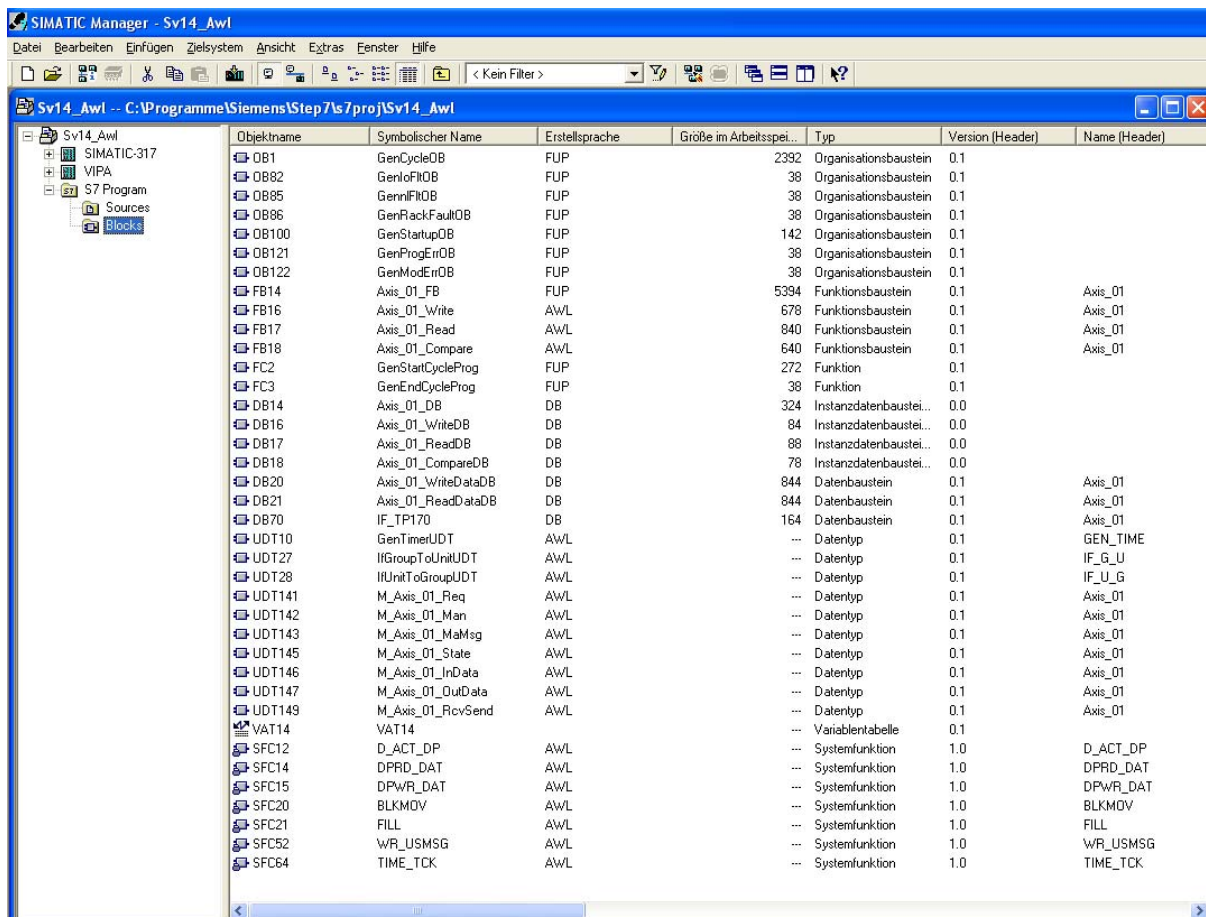
Die verwendeten Begriffe und Abkürzungen entsprechen der Drive-Software bzw. dem Handbuch in der Sprache „English“.

Das S7-Projekt „Sv14_Awl“ ist erstellt in der Sprache English mit Mnemonik deutsch und der Version:



Das S7-Projekt „Sv14_Awl“ besteht aus den folgenden Komponenten:

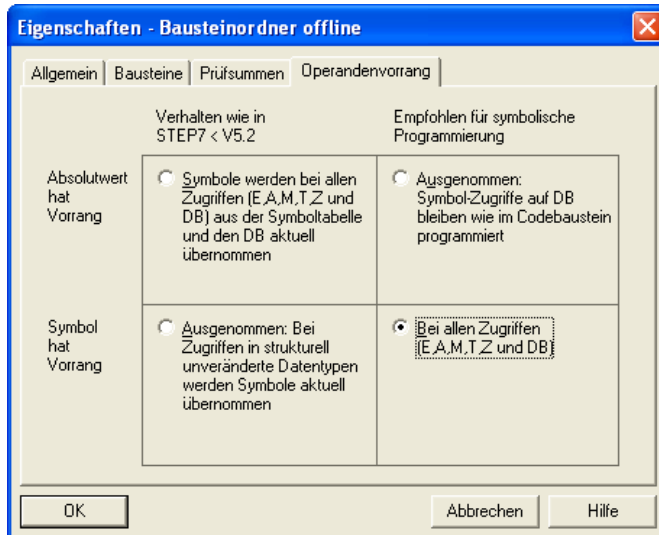
- Simatic-317 – enthält nur die Hardwarekonfiguration (system data) für eine S7-CPU317-2DP
- VIPA – enthält nur die Hardwarekonfiguration (system data) für eine VIPA-CPU315-SB
- S7-Program – enthält die Bausteine und Symbolik



Die Eigenschaften des Baustein-Ordners Offline – Operandenvorrang sind:

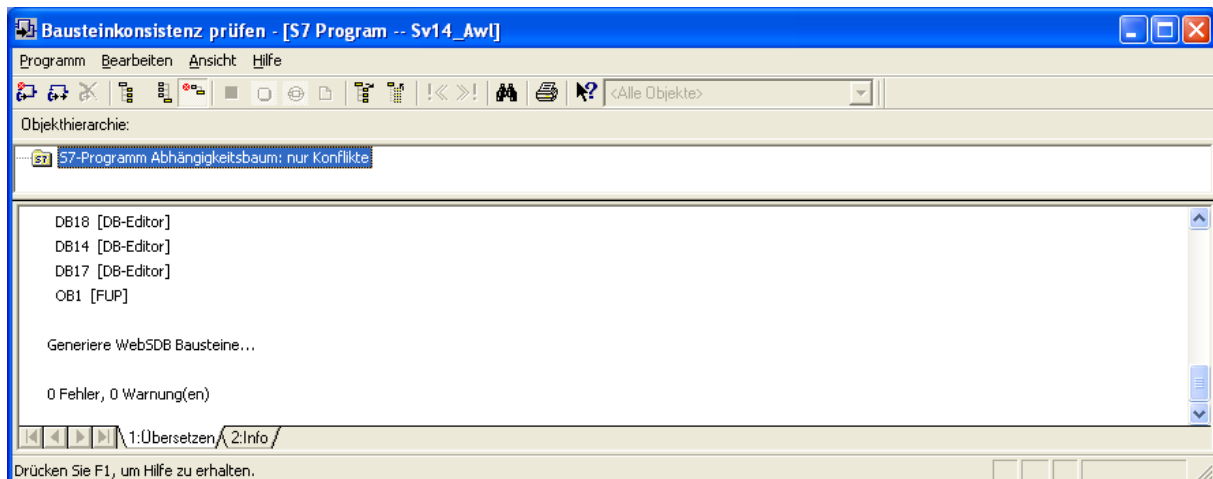
„Symbol hat Vorrang bei allen Zugriffen (E,A,M,T,Z und DB)“.

Damit kann problemlos die Datenstruktur eines FBs, UDTs oder DBs geändert und erweitert werden. Step7 behält dabei im S7-Projekt die symbolische Adresse bei und ändert automatisch dazu passend die absolute Adresse.

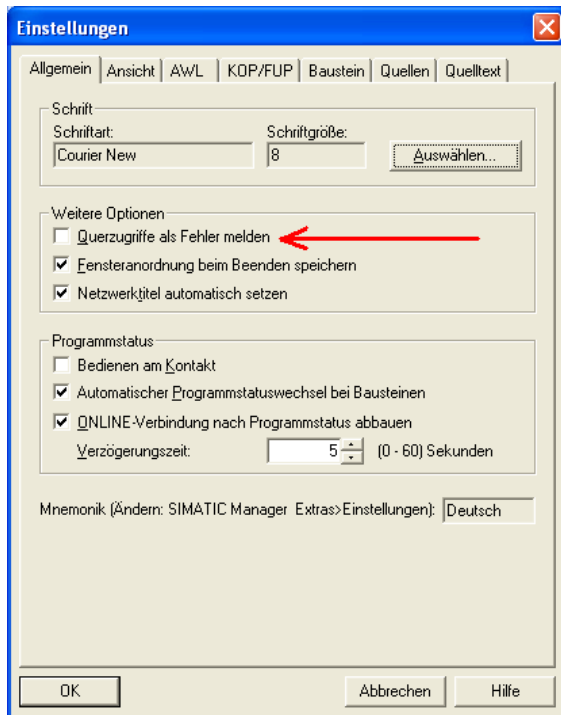


Die gesamte Datenstruktur vom S7-Projekt wird mit „Bausteinkonsistenz prüfen“ und „Alles übersetzen“ automatisch aktualisiert. Wird anschliessend die Meldung: „0 Fehler, 0 Warnungen“ nicht angezeigt dann enthält das S7-Programm in den Bausteinen noch manuell zu behebende Adresskonflikte. Dies ist manchmal bei Multi-Instanzen der Fall.

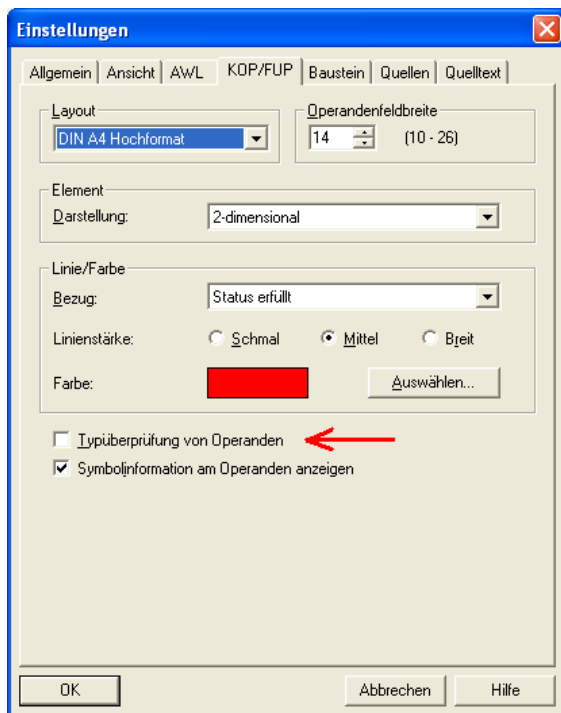
Diese Adresskonflikte werden dann im KOP/AWL/FUP-Editor manuell in den Bausteinen mit Datei – Zugriffe prüfen und aktualisieren behoben.



Werden Zugriffe auf die Instanzdaten eines DBs weiterhin ROT bzw. als Konflikt dargestellt, dann ist im KOP/AWL/FUP-Editor: EXTRAS\Einstellungen\Allgemein „Querzugriffe als Fehler melden“ zu deaktivieren. Das ist auch die Voreinstellung vom S7-Manager. Anschliessend ist das Programm nochmals zu übersetzen.



Zur vollständigen FUP-Darstellung der Netzwerke muss im KOP/AWL/FUP-Editor: EXTRAS\Einstellungen\Kop/FUP die „Typüberprüfung von Operanden“ deaktiviert sein.



Bausteinübersicht

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

Alle Bausteine können bei Bedarf umbenannt bzw. umnummeriert werden.
Anschließend muss „Bausteinkonsistenz prüfen“ mit „Alles übersetzen“ durchgeführt werden.
Daraus resultierende Konflikte müssen gegebenenfalls von Hand in den Bausteinen korrigiert werden.

Der FB14 nutzt die folgenden S7 Systembausteine:

SFC12 – D_ACT_DP – Deaktivieren und Aktivieren eines DP-Normslaves
SFC14 – DPRD_DAT – Konsistente Daten eines DP-Normslaves lesen
SFC15 – DPWR_DAT – Konsistente Daten eines DP-Normslaves schreiben
SFC20 – BLKMOV – Speicherbereich kopieren
SFC21 – FILL – Speicherbereich vorbesetzen

Im FC2 werden folgende S7 Systembausteine benutzt:

SFC52 – WR_USMSG – Anwenderdefiniertes Diagnoseereignis in S7-SPS-Diagnosepuffer schreiben
SFC64 – TIME_TCK – Systemzeit lesen (damit werden im Programm die Timer realisiert)

Anmerkungen

Der FB14 ist als Multi-Instanz programmiert und kann somit in einem FB mehrmals als Unterprogramm-Baustein ohne eigenen Instanz-Datenbaustein aufgerufen werden.

Wenn der FB14 nicht als Multi-Instanz implementiert wird, dann muss der FB14 für jeden Antrieb mit einem eigenen Instanz-DatenBaustein aufgerufen werden.

Der FB14 ist in AWL und FUP (Funktionsplan) mit der Mnemonik DEUTSCH und der Symbolik mit Kommentaren in ENGLISCH programmiert.

Der Datentyp UDT (UserDefinedDatatype) ermöglicht eine einheitliche Datenbasis und eine objektorientierte Programmierung in Step7.

Der Datentyp STRUCT ermöglicht ganze Datenbereiche einfach mittels Pointer (P#DB.DBX Byte) zu adressieren.

Das S7-Projekt „Sv14_Awl“ enthält ein komplett funktionsfähiges SPS-Programm für den Servostar. Mit einer S7-CPU und einem Servostar mit Profibuskarte und Motor kann mit der VAT14 (Variablentabelle) der komplette Servoantrieb schnell und einfach in Betrieb genommen werden. Der DB70 kann auch einfach als Interfacedatenbaustein für ein Operator/TouchPanel/Visualisierung benutzt werden.

Eingang- und Ausgangschnittstelle des FB14 - Axis_01_FB

In der Schnittstelle vom FB14 werden UDT – Anwenderdefinierte Datentypen verwendet.

UDT 141 - M_Axis_01_Req – Request
 UDT 143 - M_Axis_01_MaMsg – Machine message
 UDT 145 - M_Axis_01_State
 UDT 146 - M_Axis_01_InData
 UDT 147 - M_Axis_01_OutData
 UDT 149 - M_Axis_01_RcvSend - Receive and Send Profibus

Diese UDT werden auch noch im DB 70 - IF_TP170 - Interface DB Touch Panel TP170 verwendet. Dadurch ist sichergestellt dass die Datenstruktur in dem S7-Projekt eine einheitliche gemeinsame Datenbasis hat. Eine Änderung an der Datenstruktur erfolgt einmalig im UDT. Die gesamte Datenstruktur vom S7-Projekt wird dann mit „Bausteinkonsistenz prüfen“ und „Alles übersetzen“ automatisch aktualisiert. Zeitstempelkonflikte werden ebenfalls beseitigt.

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

FB14 - Axis_01_FB

| | | |
|------------|--------|--|
| VAR_INPUT | | |
| iData | UDT146 | |
| iReq | UDT141 | |
| ... | | |
| | | |
| VAR_OUTPUT | | |
| oMaMsg | UDT143 | |
| oState | UDT145 | |
| oData | UDT147 | |
| oAxis | UDT149 | |
| ... | | |
| | | |
| | | |

DB 70 - IF_TP170 - Interface DB Touch Panel TP170

| | | |
|-------------|--------|--|
| ToMachine | UDT27 | |
| FromMachine | UDT28 | |
| Request | UDT141 | |
| State | UDT145 | |
| ... | | |
| | | |
| iData | UDT146 | |
| oData | UDT147 | |
| oMaMsg | UDT143 | |

Gesamtübersicht der Schnittstelle vom FB14

| | | |
|----------------------|--------|---|
| VAR_INPUT | | |
| iData | STRUCT | Data: HWConfig, OpMode, RefJogSpeed, MotionTask, PNU |
| Config | STRUCT | |
| LAddrIn | WORD | StartAddressIn in HW-Config periphery dez |
| LAddrOut | WORD | StartAddressOut in HW-Config periphery dez |
| DiagAddr | WORD | DiagnosticAddress in HW-Config periphery hex |
| TO_Reference | Time | Timeout referencing |
| TO_Position | Time | Timeout positioning |
| OpModePb | 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 |
| PnuWrite | STRUCT | |
| Number | WORD | |
| Index | WORD | |
| Value | DWORD | |
| PnuRead | STRUCT | |
| Number | WORD | |
| Index | WORD | |
| iReq | STRUCT | Requests: Ref Pos StartStopCancelMotionTask Jog |
| StartRef | BOOL | Start reference movement |
| Res_0_1 | BOOL | |
| StartMotionTask | BOOL | Start motion task (direct motion task =0), P4 |
| StopMotionTask | BOOL | Stop motion task, P4->P3 |
| CancelMotionTask | BOOL | Cancel motion task, P3 |
| Res_0_5 | BOOL | |
| JogPlus | BOOL | Jog positive |
| JogMinus | BOOL | Jog negative |
| Res_1_0 | BOOL | |
| Res_1_1 | BOOL | |
| Res_1_2 | BOOL | |
| Res_1_3 | BOOL | |
| Res_1_4 | BOOL | |
| Res_1_5 | BOOL | |
| Res_1_6 | BOOL | |
| Res_1_7 | BOOL | |
| iGenSysTime | TIME | System time at beginning of OB1 |
| iGenStartup | BOOL | Startup cycle after PLC startup |
| iGenClock2000 | BOOL | Periode clock 2000ms |
| iEnError | BOOL | Enable error messages |
| iEnTimeout | BOOL | Enable timeout movement referencing and positioning |
| iAck | BOOL | Acknowledge WarningsErrors |
| iActSlave | BOOL | Activate DP-slave |
| iPowerOk | BOOL | All powersupplies are ok |
| iPauseMotionTask | BOOL | Pause for Motion Task, P4 |
| iResetPosition | BOOL | Reset position, set ActualPosition to RefPosition =0 |
| iFastStopDisableAxis | BOOL | FastStop with disable axis, P4->P1 |
| iFastStop | BOOL | FastStop without disable axis, P4->P11 |
| iPnuWriteStart | BOOL | Request PnuWriteStart pulse 0->1 |
| iPnuReadStart | BOOL | Request PnuReadStart static 1 |
| iInit | BOOL | Initialize axis with disable axis |
| iSetOpModePb | BOOL | Set operating mode Profibus |
| iSetOperationEnable | BOOL | Set axis state machine to P4_OperationEnabled |
| iSwEnable | BOOL | Software enable axis |

| | | |
|--------------------------|--------|--|
| VAR_OUTPUT | | |
| oMamsg | STRUCT | Error messages |
| ErrTO_Ref | BOOL | Error timeout reference |
| ErrTO_Pos | BOOL | Error timeout positioning |
| ErrRes_0_2 | BOOL | |
| ErrRes_0_3 | BOOL | |
| 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 |
| ErrRes_1_3 | BOOL | |
| 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 DP communication is ok |
| InitOK | BOOL | Axis initialisation is ok |
| InitError | BOOL | Axis initialisation error |
| OpModePbOk | BOOL | Axis opmode Profibus is ok |
| OpModePbError | 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 OpMode Position:0=MotionTaskNumber/1=DirectMotionTask |
| DigitalRevolutionSpeed | BOOL | OpMode dependent, digital velocity |
| SwitchOn | BOOL | |
| InhibitVoltage | BOOL | |
| FastStopSwitchOn | BOOL | 1>0Axis FastStopWithEmgyRamp, AxisWillDisabled-STOPMODEDECDIS |
| 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 |
| 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 | |
| oError | BOOL | Error is active |

^{1*)} Die aktuelle Positon wird in den Prozessdaten nur in reglerinternen Einheiten (2^{20} Inkr. pro Motorumdrehung) übertragen. Die Umrechnung in Benutzereinheiten kann in der SPS erfolgen, abhängig von der mit der Bediensoftware im Regler eingestellten Positonsauflösung.

Beispiel: Auflösung = 5000 Inkr/ 3 Umdrehungen

$$\Rightarrow \text{Position in Benutzereinheiten} = \text{Actual_Position} \times 5000 / (3 \times 2^{20})$$

Inbetriebnahme FB14 Axis_01_FB

S7-SPS - AUS/EIN-Schalten

Fällt der Profibusmaster S7-SPS aus dann startet im Servostar der Watchdog-Timeout.
Die S7-SPS macht nach dem Anlauf einen Software-Reset (GenStartUp - M1.2) und löscht alle Fehlermeldungen und Zustände im SPS-Programm.

S7-SPS - Start/Stop-Schalten

macht einen Software-Reset (GenStartUp - M1.2) und löscht alle Fehlermeldungen und Zustände im SPS-Programm. Der Servostar bleibt enabled und ein aktiver Fahrauftrag wird weiterhin ausgeführt.

iAck =1

setzt Warnungen und Fehler im Servostar und in den Hantierungsbausteinen zurück.

Initialisierung

Bei der Initialisierung wird ein Null-Telegramm im PKW und PZD gesendet. Die Achse wird disabled.

- 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 (Der FB14 ist für die Betriebsart Positionieren programmiert)

Nach dem Einschalten ist der Servostar immer in der sicheren Betriebsart -126

- Schreibe die gewünschte Betriebsart in iData.OpMode
(siehe Handbuch z.B. 2 für „Positionieren“)
- Setze iSetOpModePb =1 (intern wird ein Impuls generiert)
- Warte bis oState.OpModeOkOK = 1
Wenn oState.OpModePbOk nicht =1 oder oState.OpModePbError =1 dann war Betriebsart setzen nicht erfolgreich. Fehlerursache siehe Receive-Telegramm Servostar.
- Jetzt werden auch oActualPosition, oActualSpeed und oManufactState mit Werten angezeigt.
- Die Betriebsart setzen wird mit einem Timeout von 1 sec überwacht.

Betrieb freigeben

Nach dem Einschalten ist der Servostar im Zustand „Switch-On inhibited / Einschaltsperrung“.

iFastStop und iFastStopDisableAxis müssen =0 und iSwEnable muss =1 sein.

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

- 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.
Fehlerursache siehe Receive-Telegramm Servostar.

Tippbetrieb

Voraussetzung: Der Antrieb ist im Zustand „Operation enabled / Betrieb freigegeben“ und Hardware-Enable ist vorhanden und ist fehlerfrei.

- Schreibe die gewünschte Tippgeschwindigkeit in „iData.Config.JogSpeed“
- Setze iReq.JogPlus =1 -> Der Antrieb bewegt sich in die positive Richtung
- Setze iReq.JogMinus =1 -> Der Antrieb bewegt sich in die negative Richtung
oState.JoggingActive ist =1 wenn der Antrieb sich bewegt.

Referenzfahrt

Die Art der Referenzfahrt ist mit der Bediensoftware DRIVEGUI / DRIVE einzustellen.

Voraussetzung: Der Antrieb ist im Zustand „Operation enabled / Betrieb freigegeben“ und Hardware-Enable ist vorhanden und ist fehlerfrei.

- Schreibe die gewünschte Referenzfahrtgeschwindigkeit in „iData.Config.RefSpeed“
- Setze iReq.StartRef =1 -> Der Antrieb bewegt sich und oState.ReferencingActive =1
- Warte bis oState.ReferenceOk =1 und oState.ReferencingActive 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

Mit der Bediensoftware DRIVEGUI / DRIVE werden Fahraufträge angelegt.

Nur die Parameter der RAM-Fahraufträge können im Antrieb im Zustand „Operation enabled / Betrieb freigegeben“ geändert werden (siehe ASCII-Kommando MTMUX)

Voraussetzung: Der Antrieb ist im Zustand „Operation enabled / Betrieb freigegeben“ und referenziert.

- Schreibe die Nummer des zu startenden 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 Antrieb hat den Fahrauftrag ausgeführt

Mit iEnTimeout =1 und iData.Config.TO_Position xxx ms [TIME] wird der Fahrauftrag mit einem Timeout überwacht.

Start eines Direktfahrauftrags (hat die Nummer 0)

Voraussetzung: Der Antrieb ist im Zustand „Operation enabled / Betrieb freigegeben“ und referenziert.

- Schreibe 0 in iData.MotionTask.Number
- Schreibe Sollgeschwindigkeit in iData.MotionTask.DirectSpeed
- Schreibe Zielposition in iData.MotionTask.DirectPosition
- Schreibe Fahrauftragsart in iData.MotionTask.DirectType
- Setze iReq.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 Antrieb hat den Fahrauftrag ausgeführt

Anmerkung: Mittels Pointer (P#DB.DBX Byte) kann der iData.MotionTask mit einem SFC20-BLKMOV einfach beschrieben werden.

Mit iEnTimeout =1 und iData.Config.TO_Position xxx ms [TIME] wird der Fahrauftrag mit einem Timeout überwacht.

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

Ein gestarteter MotionTask kann mit iPauseMotionTask =1 angehalten werden.

Ein gestarteter MotionTask kann mit iReq.StopMotionTask =1 gestoppt werden.

Ein gestoppter MotionTask bleibt im Regler weiterhin aktiv, d.h. wenn der Regler wieder in den oState.P4_OperationEnabled geschaltet wird, wird der MotionTask fortgesetzt.

Mit iSwEnable =0 wird der gestoppte MotionTask gekillt, d.h wenn der Regler wieder in den oState.P4_OperationEnabled geschaltet wird, wird der MotionTask nicht mehr fortgesetzt.

Hinweis: Mit dem Parameter INPT kann im Terminalfenster der Bediensoftware (oder mit der PNU1904) die Zeit in ms eingestellt werden, für die das Signal oManufactState.InPositionWindow nach dem Fahrsatzstart mindestens zurückgesetzt wird (vgl. Handbuch Kap. VII.1).

Hinweis zu Positionen und Geschwindigkeiten: Zielposition und Sollgeschwindigkeit können in den im Lageregler eingestellten Einheiten vorgegeben werden, wenn in der Fahrauftragsart PNU1785 das Bit 13 gesetzt ist (vgl. Handbuch Kap. IV.2.5.3).

Fahrauftragsarten (PNU 1785 - häufige Werte vgl. Handbuch Kap. IV 2.5.3)

0x2000 (Bit 13 gesetzt) Absolute Positionierung mit Vorgabe der Geschwindigkeiten und Positionen in Benutzereinheiten.

0x2003 (Bits 0, 1, 13 gesetzt) Positionierung „relativ Soll“ mit Vorgabe von Geschwindigkeiten und Positionen in Benutzereinheiten.

Parameter lesen und schreiben:

Alle Parameter und Kommandos des SERVOSTAR® sind über die Profibus PNU-Nummer ansprechbar. In der ASCII Kommandoliste können die Nummern der Reihe nach gelistet angezeigt werden. Außerdem findet sich die Nummer bei der Beschreibung der Kommandos/ Parameter im Feld „Profibus PNU“. Zusätzlich befindet sich im Handbuch eine Liste ausgewählter Parameternummern.

Parameter schreiben

Voraussetzung: Parameterkanal PKW wird nicht bereits benutzt

- Schreibe Nummer in iData.PnuWrite.Number
 - Schreibe Index in iData.PnuWrite.Index
 - Schreibe Wert in iData.PnuWrite.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 iData.PnuRead.Number
 - Schreibe Index in iData.PnuRead.Index
 - Schreibe Wert in iData.PnuWrite.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 PNU 1688 Effektivstrom zur Ermittlung des Drehmoments).

Zusätzliche Profibus-Funktionen

Mit iActSlave =1 wird im FB14 mit dem SFC12 – D_ACT_DP der DP-Slave Servostar bei laufender SPS aktiviert bzw. mit iActSlave =0 deaktiviert.

Zudem wird die Profibuskommunikation auf

- Konfigurationsfehler
- Slaveausfall
- und Telegrammfehler überwacht

Die gesamten 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-SPS damit die Fahrsatz-Daten korrigieren und den relativen Fahrauftrag mit korrigierten Daten zu Ende fahren.

Mit der Variablentabelle VAT14 - können die Bausteine im Servostar angesteuert und beobachtet werden.

Beispiel für eine Schrittkette „Initialisierung und Enablen Servostar“

```

U      "Axis_01_DB".oState.CommunicationOk
L      S5T#1S
SE     T      100
U      T      100
FP     M      100.0
S      M      100.1
R      M      100.2
R      M      100.3
R      M      100.4
R      M      100.5
R      M      100.6
R      M      100.7

U      M      100.1
=      "IF_TP170".ToMachine.Ack

U      M      100.1
L      S5T#1S
SE     T      101
U      T      101
S      M      100.2
R      M      100.1

U      M      100.2
UN     "Axis_01_DB".oError
L      S5T#1S
SE     T      102
U      T      102
S      M      100.3
R      M      100.2

// -----

U      M      100.3
UN     "Axis_01_DB".oError
UN     "Axis_01_DB".oState.InitOk
=      "IF_TP170".Init

U      M      100.3
UN     "Axis_01_DB".oError
U      "Axis_01_DB".oState.InitOk
UN     "Axis_01_DB".oState.OpModePbOk
=      "IF_TP170".SetOpmodePb

U      M      100.3
UN     "Axis_01_DB".oError
U      "Axis_01_DB".oState.InitOk
U      "Axis_01_DB".oState.OpModePbOk
UN     "Axis_01_DB".oState.P4_OperationEnabled
=      "IF_TP170".SetOperationEnable

U      M      100.3
UN     "Axis_01_DB".oError
U      "Axis_01_DB".oState.InitOk
U      "Axis_01_DB".oState.OpModePbOk
U      "Axis_01_DB".oState.P4_OperationEnabled
UN     "Axis_01_DB".oState.ReferenceOk
=      "IF_TP170".Request.StartRef

U      M      100.3
UN     "Axis_01_DB".oError
U      "Axis_01_DB".oState.InitOk
U      "Axis_01_DB".oState.OpModePbOk
U      "Axis_01_DB".oState.P4_OperationEnabled
U      "Axis_01_DB".oState.ReferenceOk
R      M      100.3

```

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
```

```

U      "Axis_01_DB".oState.CommunicationOk
UN     "Axis_01_DB".oError
U      "Axis_01_DB".oState.InitOk
U      "Axis_01_DB".oState.OpModePbOk
U      "Axis_01_DB".oState.P4_OperationEnabled
U      "Axis_01_DB".oState.ReferenceOk
L      S5T#1S
SE     T      110
U      T      110
FP     M      110.0
S      M      110.1
R      M      110.2
R      M      110.3
R      M      110.4
R      M      110.5
R      M      110.6
R      M      110.7

U      M      110.1
L      S5T#2S
SE     T      111
U      T      111
S      M      110.2
R      M      110.1

U      M      110.2
UN     "Axis_01_DB".oState.MotionTaskActive
L      S5T#2S
SE     T      112
U      T      112
S      M      110.3
R      M      110.2

U      M      110.3
L      S5T#2S
SE     T      113
U      T      113
S      M      110.4
R      M      110.3

U      M      110.4
UN     "Axis_01_DB".oState.MotionTaskActive
L      S5T#2S
SE     T      114
U      T      114
S      M      110.5
R      M      110.4

U      M      110.5
L      S5T#2S
SE     T      115
U      T      115
S      M      110.6
R      M      110.5

U      M      110.6
UN     "Axis_01_DB".oState.MotionTaskActive

L      S5T#2S
SE     T      116
U      T      116
S      M      110.7
R      M      110.6

U      M      110.7
L      S5T#2S
SE     T      117
U      T      117
R      M      110.7

```



```
// -----
```

```

      U      M      110.3
      SPBN   m201
      L      201
      T      "IF_TP170".iData.MotionTask.Number
m201: NOP    0

      U      M      110.5
      SPBN   m210
      L      210
      T      "IF_TP170".iData.MotionTask.Number
m210: NOP    0

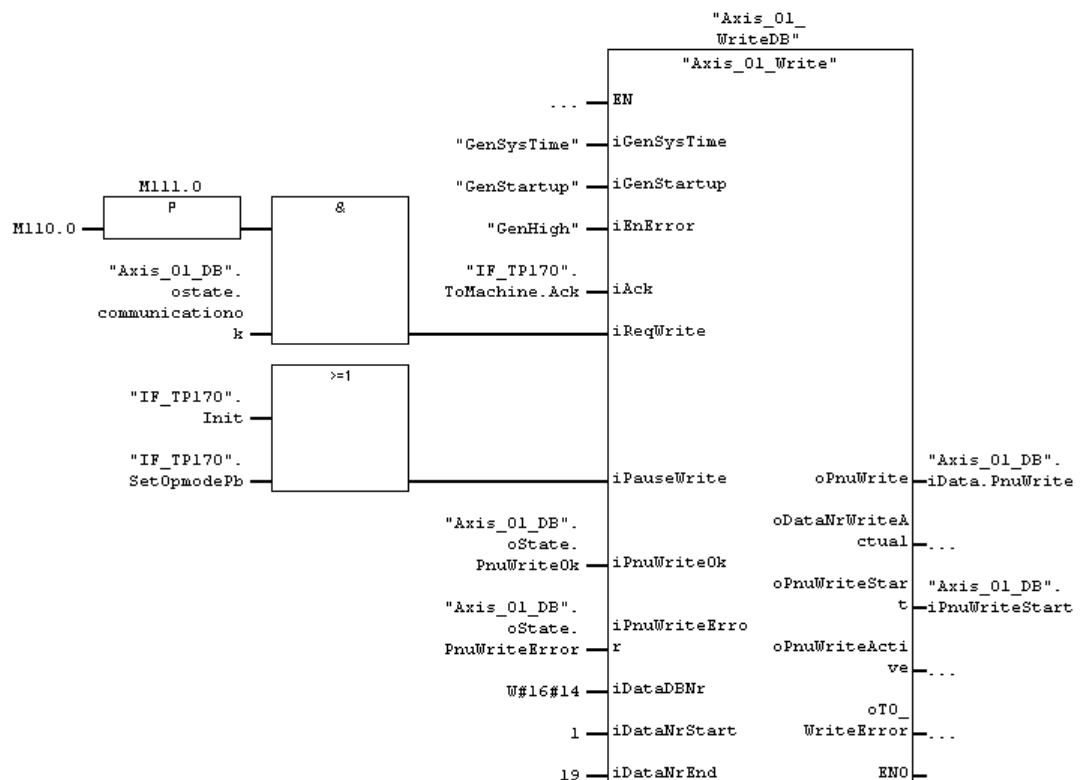
      U      M      110.7
      SPBN   m211
      L      211
      T      "IF_TP170".iData.MotionTask.Number
m211: NOP    0

      CLR
      O      M      110.3
      O      M      110.5
      O      M      110.7
      =      "IF_TP170".Request.StartMotionTask

```

Network 5: Axis_Write

```
iDataDENr: DB20 -> W#16#14
from Data[1] until Data[8]
```



Bausteine für Servostar-Parameter

Das S7-Projekt „Sv14_Awl“ enthält noch weitere Funktionsbausteine und Datenbausteine die ganze Datenbereiche in den Servostar schreiben, aus dem Servostar lesen und in der S7-SPS 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.

DB 20 - Axis_01_WriteDataDB

enthält die Daten zum schreiben mit dem FB16 Axis_01_Write und dem FB14 Axis_01 von der S7-SPS über den über den nicht echtzeitfähigen PKW-Parameterkanal im Profibus in den Servostar.

Deklaration vom DB20 – für 100 Werte

KOP/AWL/FUP - [DB20 -- "Axis_01_WriteDataDB" -- Sv14_Awl\S7 Program\...DB20]

Datei Bearbeiten Einfügen Zielsystem Test Ansicht Extras Fenster Hilfe

| Adresse | Name | Typ | Anfangswert | Kommentar |
|---------|---------|---------------|-------------|--|
| 0.0 | | STRUCT | | |
| +0.0 | Res_0_7 | ARRAY[0..7] | | |
| *1.0 | | BYTE | | |
| +8.0 | Data | ARRAY[1..100] | | StartAddress for DataNumber 1 is 8 !!! |
| *0.0 | | STRUCT | | |
| +0.0 | Number | WORD | W#16#0 | |
| +2.0 | Index | WORD | W#16#0 | |
| +4.0 | Value | DWORD | DW#16#0 | |
| =8.0 | | END_STRUCT | | |
| =808.0 | | END_STRUCT | | |

DB 21 - Axis_01_ReadDataDB

enthält die Daten zum lesen mit dem FB17 Axis_01_Read und dem FB14 Axis_01 von dem Servostar über den über den nicht echtzeitfähigen PKW-Parameterkanal im Profibus in die S7-SPS.

Deklaration vom DB21 – für 100 Werte

KOP/AWL/FUP - [DB21 -- "Axis_01_ReadDataDB" -- Sv14_Awl\S7 Program\...DB21]

Datei Bearbeiten Einfügen Zielsystem Test Ansicht Extras Fenster Hilfe

| Adresse | Name | Typ | Anfangswert | Kommentar |
|---------|---------|---------------|-------------|--|
| 0.0 | | STRUCT | | |
| +0.0 | Res_0_7 | ARRAY[0..7] | | |
| *1.0 | | BYTE | | |
| +8.0 | Data | ARRAY[1..100] | | StartAddress for DataNumber 1 is 8 !!! |
| *0.0 | | STRUCT | | |
| +0.0 | Number | WORD | W#16#0 | |
| +2.0 | Index | WORD | W#16#0 | |
| +4.0 | Value | DWORD | DW#16#0 | |
| =8.0 | | END_STRUCT | | |
| =808.0 | | END_STRUCT | | |

FB16 Axis_01_Write

steuert den FB14 an zum Daten schreiben von der SPS in den Servostar

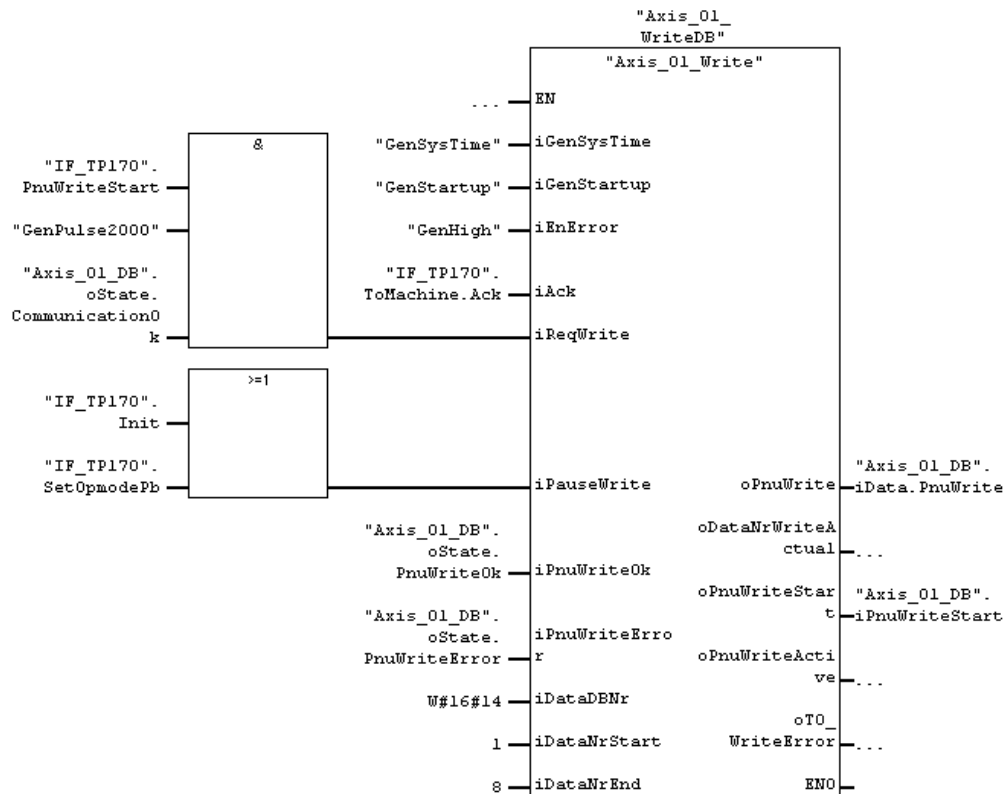
Eingang- und Ausgangsschnittstelle des FB16 - Axis_01_Write

| VAR_INPUT | | |
|----------------|------|--|
| iGenSysTime | TIME | System time at beginning of OB1 |
| iGenStartup | BOOL | Startup cycle after PLC startup |
| 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 | DataSetNumberStart - first number from data to write |
| iDataNrEnd | INT | DataSetNumberEnd - 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 | |
| oDataSetNrWriteActual | BOOL | 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 |

Netzwerk 5 : Axis_Write

iDataDBNr: DB20 -> W#16#14
from Data[1] until Data[8]



FB17 Axis_01_Read

steuert den FB14 an zum Daten lesen vom Servostar in die SPS

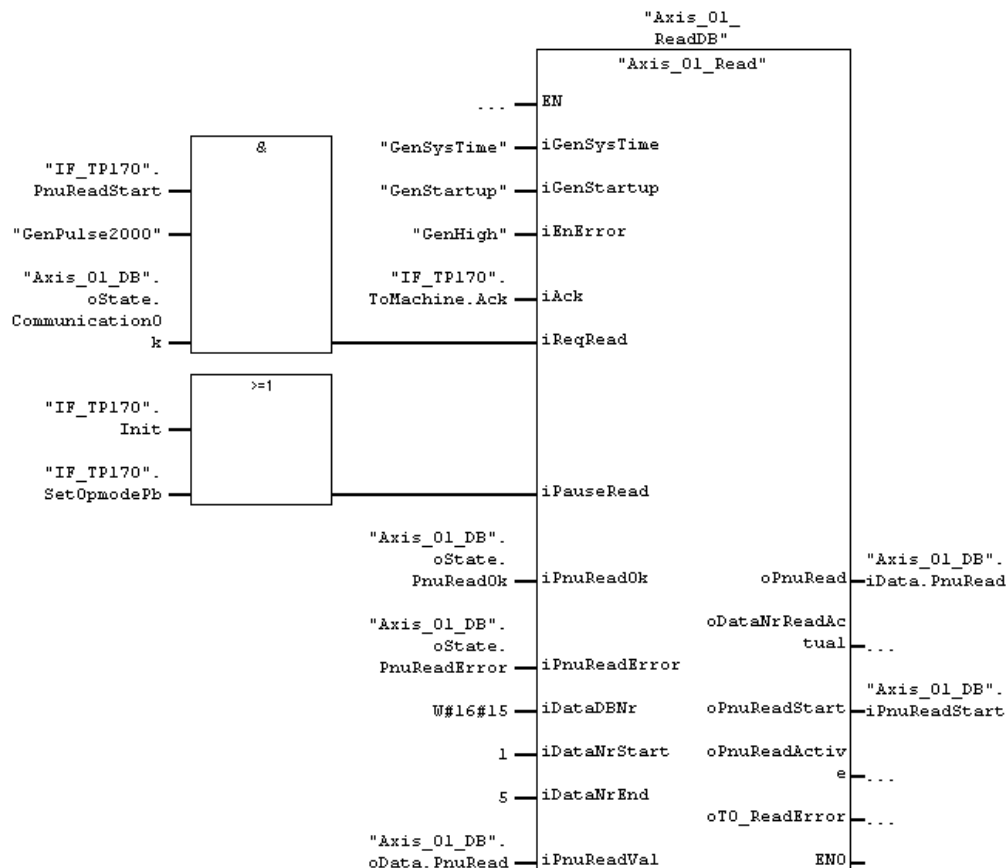
Eingang- und Ausgangschnittstelle des FB17 - Axis_01_Read

| VAR_INPUT | | |
|----------------|--------|---|
| iGenSysTime | TIME | System time at beginning of OBL |
| iGenStartup | BOOL | Startup cycle after PLC startup |
| iEnError | BOOL | Enable error messages |
| iAck | BOOL | Acknowledge WarningsErrors |
| iReqRead | BOOL | RequestRead |
| iPauseWrite | BOOL | PauseRead (necessary for SetOpmode or InitAxis) |
| iPnuWriteOk | BOOL | PnuReadOk =1 succesful |
| iPnuWriteError | BOOL | PnuReadError =1 not succesful |
| iDataDBNr | WORD | ReadDataDBNumber |
| iDataNrStart | INT | DataSetNumberStart - first number from data to read |
| iDataNrEnd | INT | DataSetNumberEnd - last number from data to read |
| iPnuReadVal | STRUCT | PnuRead from Axis_FB actual with Number, Index, Value |
| Number | WORD | |
| Index | WORD | |
| Value | DWORD | |

| VAR_OUTPUT | | |
|-------------------|--------|--|
| oPnuRead | STRUCT | PnuRead from Axis_FB actual with Number, Index |
| Number | WORD | |
| Index | WORD | |
| oDataNrReadActual | WORD | 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 |

Netzwerk 8 : Axis_Read

iDataDBNr: DB21 -> W#16#15
from Data[1] until Data[5]



Hinweis:

Bei gleichzeitiger Anforderung Read und Write am FB14 generiert der FB14 den Fehler
oMaMsg.ErrReadWrite

FB18 Axis_01_CompareDB

vergleicht einen Datenbereich von den geschriebenen Daten vom DB20 Axis_01_WriteDataDB mit den gelesenen Daten vom DB21 Axis_01_ReadDataDB.
Pro SPS-Zyklus wird nur ein Datensatz verglichen.

Eingang- und Ausgangsschnittstelle des FB18 Axis_01_CompareDB

| VAR_INPUT | | |
|---------------|------|--|
| iGenStartup | BOOL | Startup cycle after PLC startup |
| 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 |
| iBDataDBNr | WORD | BDataDBNumber |
| iBDataNrStart | INT | BDataNumberStart - first number from data to compare |

| VAR_OUTPUT | | |
|-----------------------|------|--|
| oCmpADaNrActual | WORD | compare A Data number is actual |
| oCmpBDataNrActual | WORD | compare B Data number is actual |
| oCmpADaNrEqualBDataNr | BOOL | compare A Data number is equal B Data number |
| oCompareActive | WORD | comparing DataBlocks is active |
| oCompareOk | BOOL | compare is ok, datas are equal |
| oCompareNotOk | BOOL | compare is not ok, datas are not equal |

Netzwerk 10 : Axis_Compare

iADaDBNr: DB20 -> W#16#14 from Data[7] until Data[8]
iBDataDBNr: DB21 -> W#16#15 from Data[1] until Data[8-7=2]

