

Kurzdokumentation

Integration i550 in Siemens Step 7 und Siemens TIA Portal

an Profibus und Profinet

Verwendete Softwareversionen:

Lenze Easy Starter V1.10.2

i550, Profibus Standard IO mit FW 03.00.02, Profinet Standard IO mit FW 03.00.02

Siemens TIA Portal V14 Update 1

Siemens Step 7 V5.5 SP4

Version	Anmerkungen	Autor / Datum
V1.0	Ersterstellung	Dittmann / 26.10.2015
V1.1	Update Profinet	Dittmann / 22.02.2016
V2.0	Update TIA-Portal V14	Dittmann / 06.12.2016

Inhaltsverzeichnis:

1	i550 am Profibus	2
1.1	Parametrierung mit dem EASY Starter	2
1.1.1	Einstellung der Profibusadresse	2
1.1.2	Verbindung zum i550 via USB-Diagnoseinterface aufbauen	2
1.1.3	Einstellung der nötigen Grundparameter	3
1.2	Integration unter Step - 7	3
1.2.1	Hardwarekonfiguration	4
1.2.2	Einbinden des Funktionsbausteins „LCB-ActuatorSpeed_V2_1 in das Steuerungsprogramm der S7	7
1.3	Integration unter dem Siemens TIA-Portal.....	10
1.3.1	Hardwarekonfiguration im TIA-Portal mit S7-300 Steuerung	11
1.3.2	Einbinden des Funktionsbausteins „LCB-ActuatorSpeed_V2_1 in das Steuerungsprogramm der S7	16
1.3.3	Besonderheiten bei S7-1200 und S7-1500 Steuerungen.....	20
2	i550 am Profinet	22
2.1	Parametrierung mit dem EASY Starter	22
2.1.1	Einstellung der Profinetadresse und des Stationsnamens.....	22
3	Parameterdatenkommunikation via DP-V1	22
3.1	Beispiel lesen / schreiben eines Parameters mit einer S7-1500 CPU	23
3.1.1	Bestimmung der Diagnoseadresse bei einer S7-300.....	23
3.1.2	Bestimmung der HW-Kennung für den Parametertransfer für S7-1200 und S71500	23
3.1.3	Einbinden der Bausteine im SPS Programm	24

1 i550 am Profibus

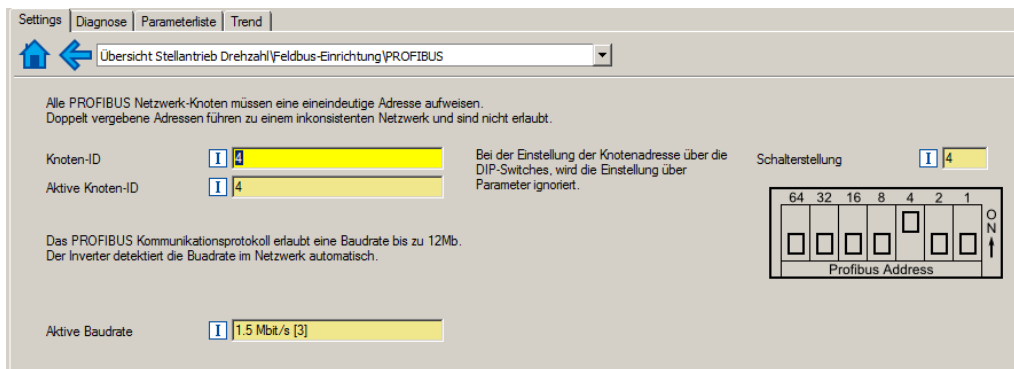
Diese Dokumentation beschreibt die Integration des i550 am Profibus mit den Lenze S7-AppSample Bibliotheken und der Applikation Stellantrieb Drehzahl.

1.1 Parametrierung mit dem EASY Starter

Am i550 müssen für die grundsätzliche Ansteuerung über Profibus nur wenige Codestellen über das Keypad bzw. den EASY Starter eingestellt werden. Die grundsätzlichen Antriebsparameter wie Eckfrequenz oder Maximalstrom werden ebenso über den EASY Starter parametrieren.

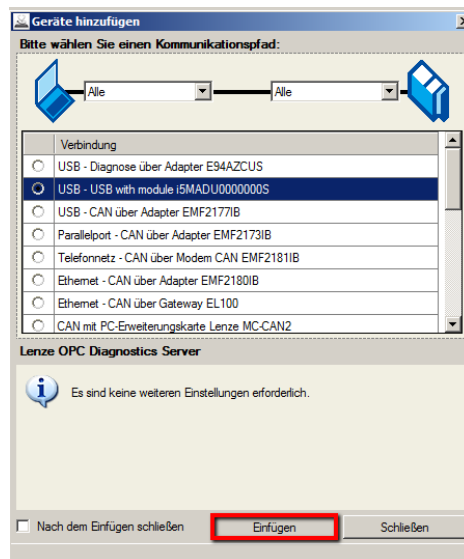
1.1.1 Einstellung der Profibusadresse

Die Profibusadresse kann über die Dipschalter an der Frontseite des i550 eingestellt werden. Wird die Adresse nicht über die Dipschalter eingestellt so kann diese über den Menüpunkt „Feldbus-Einrichtung\Profibus“ eingestellt werden (Index 2341:001).



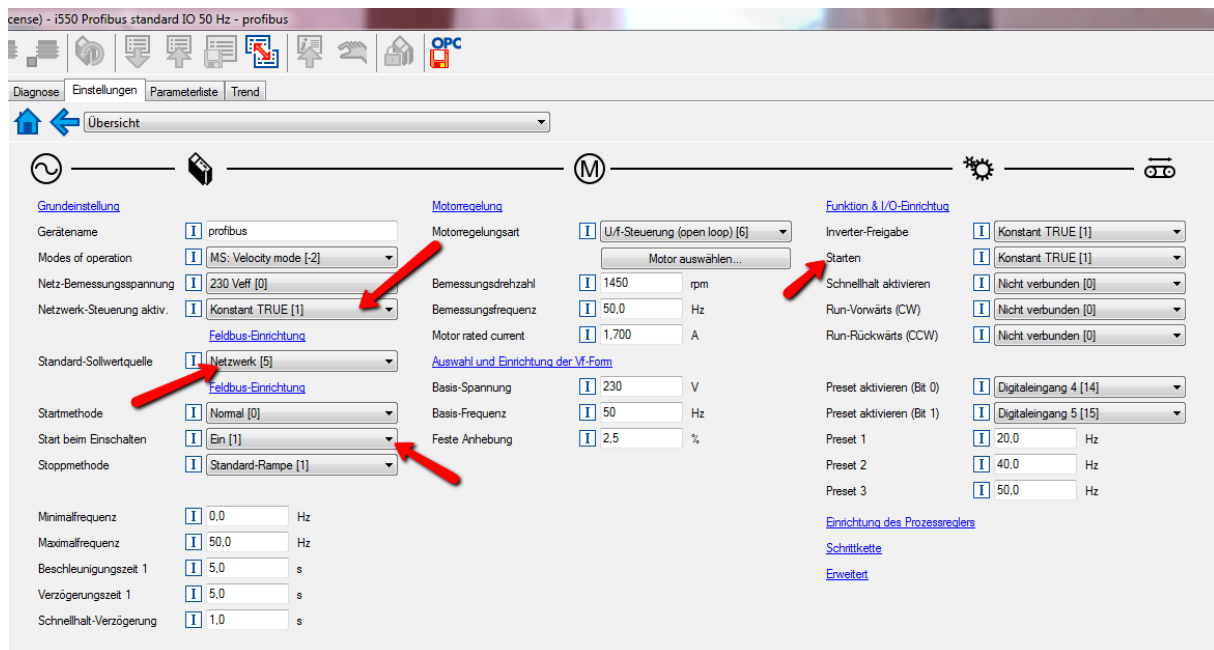
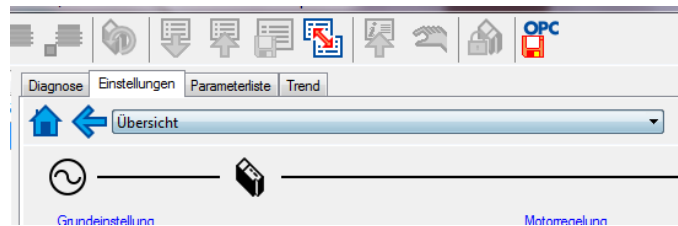
Im Beispiel wird die Adresse 4 verwendet.

1.1.2 Verbindung zum i550 via USB-Diagnoseinterface aufbauen



1.1.3 Einstellung der nötigen Grundparameter

Über das Schnellzugriff – Dropdown Menü den Punkt „Übersicht“ anwählen:

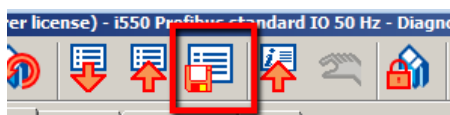


Netzwerk Steuerung aktiv (Index 0x2631:037) auf „Konstant True“ setzen.

Standard Sollwertquelle (Index 0x2860:001) auf Netzwerk [5] einstellen:

Je nach Anforderung, die Start-Freigabe (Index 2631:002) auf „Konstant True[1]“ stellen. Dies hat zur Folge dass keine Hardware Reglerfreigabe erforderlich ist. Ebenso kann noch das Startverhalten beim Einschalten festgelegt werden Start beim Einschalten (Index 0x2838:002).

Abschließend den Parametersatz im i550 abspeichern.



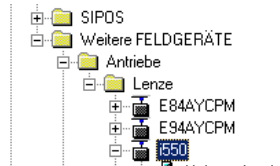
1.2 Integration unter Step - 7

Für die Einbindung unter Step-7 wird die GSD Datei des i550 benötigt. Diese muss über den Hardwaremanager in den Hardwarekatalog von Step-7 eingebunden werden.

Um den Baustein „LCB_ActuatorSpeed_V2_1“ aus der Lenze Beispielbibliothek zu verwenden muss diese noch eingebunden werden.

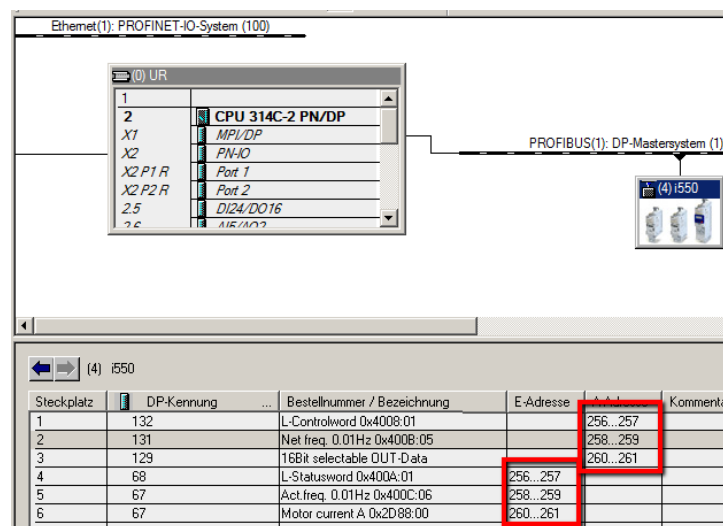
1.2.1 Hardwarekonfiguration

Auswahl des i550 unter „Profibus-DP/Weitere Feldgeräte/Antriebe/Lenze“ den i550 auswählen.



Anschließend die Profibusadresse einstellen (Im Beispiel: 4)

Wird der i550 per „Drag and Drop“ in die Hardwarekonfiguration gezogen so wird ist die Standard IO-Konfiguration der Prozessdaten bereits angelegt. Sie muss nicht mehr manuell angelegt werden, ist aber editierbar bzw. erweiterbar.



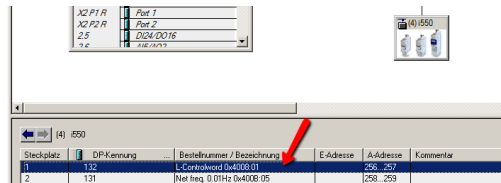
Für den Baustein „LCB_ActuatorSpeed_V2_1“ wird eine Datenlänge von jeweils 3 Worten Eingangsdaten und Ausgangsdaten benötigt. **Damit der Baustein einwandfrei arbeitet ist darauf zu achten das die Adressen ohne Sprünge direkt aufeinander folgen, auch müssen die Anfangsadressen des E/A Bereiches identisch sein (256).** Die Grundkonfiguration muss für die Basisfunktionalität nicht mehr angepasst werden. Individuelle Anpassungen sind jedoch möglich.

1.2.1.1 Anpassung des Prozessdaten-Mappings

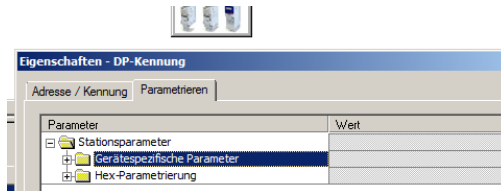
Das Prozessdaten Mapping kann nur auf der Steuerungsseite im Hardwaremanager geändert werden. Die Siemens-Steuerung schreibt dieses bei jedem Neustart auf den i550 via Profibus herunter.

Die Belegung der Freien Eingangsbits (xFreeCtrl_1-4) des Lenze FB kann direkt im Hardwaremanager im Controlword geändert werden.

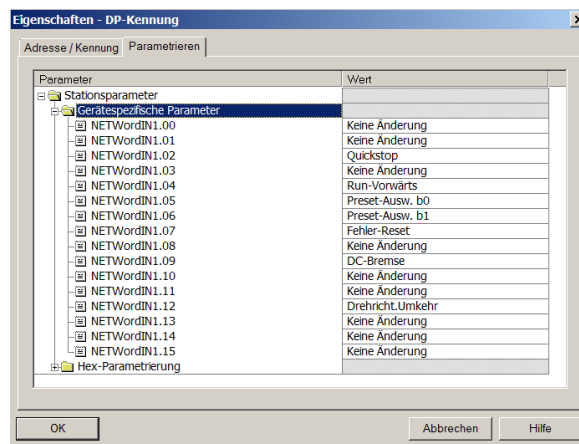
Doppelklick auf das Controlword öffnet den Parametrierdialog:



Anwahl Parametrieren:



Gerätespezifische Parameter erweitern:



Die freien Controlbits sind wie folgt zugeordnet:

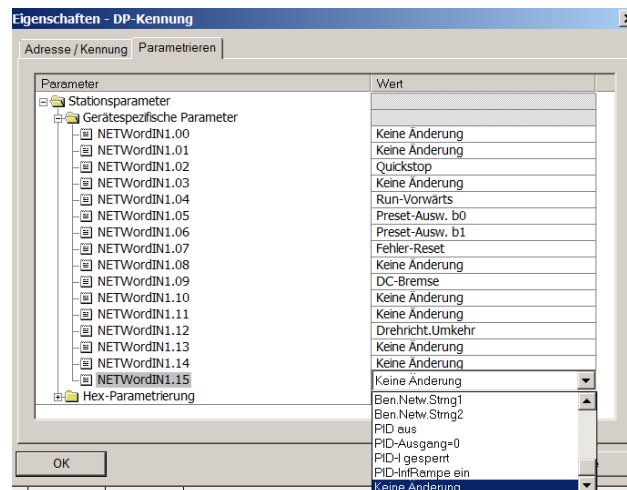
xFreeCtrl_1 <-> NetWordIn1.11

xFreeCtrl_2 <-> NetWordIn1.13

xFreeCtrl_3 <-> NetWordIn1.14

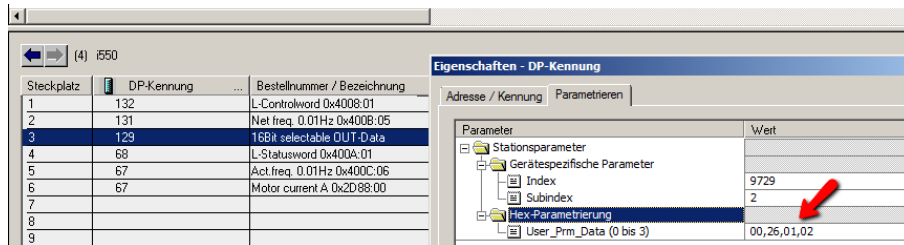
xFreeCtrl_4 <-> NetWordIn1.15

Für jedes einzelne Bit kann über das Drop-Down Menü eine neue Funktion verknüpft werden:



Das freie Eingangswort wFreeCtrl_1 des Bausteins ist mit dem 3. Ausgangswort des Prozessabbildes verknüpft (16Bit selectable Out-Data) und kann ebenfalls nur in der Hardwarekonfiguration auf S7 Seite verändert werden:

Öffnet man den Parametrierdialog durch Doppelklick so kann man selektieren welcher Index im Regler geschrieben werden soll:



Unter Gerätespezifische Parameter steht der Index und Subindex in Dezimaler Darstellung, unter Hex-Parametrierung sieht man direkt den HEX-Index: (0x2601:02 Defaultbelegung):

Adresse	Display Code	Bezeichnung	Wert
0x2601:000	PAR0202:000	highest sub-index supported (generated)	3
0x2601:001	PAR0202:001	Keypad-Sollwertbefehle: Frequenz-Sollwert	20.0
0x2601:002	PAR0202:002	Keypad-Sollwertbefehle: Prozessregler	0.00
0x2601:003	PAR0202:003	Keypad-Sollwertbefehle: Drehmoment-Sollwert	100.0

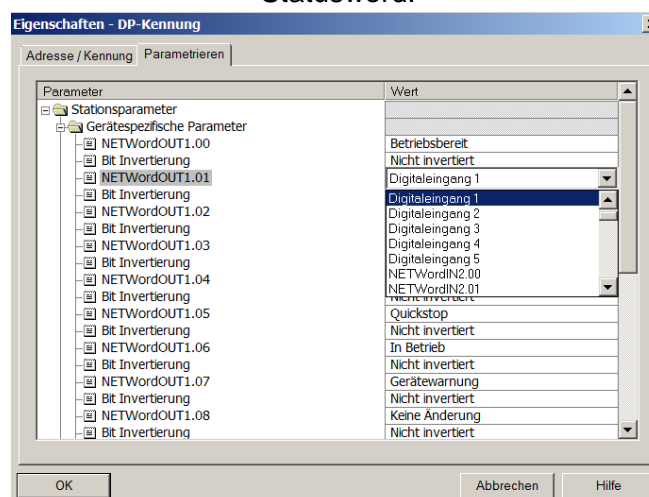
Soll ein anderer Sollwert übertragen werden so muss dieser Index die Eigenschaft „mapable“ besitzen, dies kann der Parameterbeschreibung in der Dokumentation entnommen werden.

0x2601	Keypad-Sollwerte			
	001: Frequenz-Sollwert	20.0 Hz	0.0 ... 599.0 Hz	U16 / 10 / P / r
	002: Prozessregler-Sollwert	0.00 PUnit	-300.00 ... 300.00 PUnit	116 / 100 / P / r

Das kleine „r“ steht für Receive-Mapping erlaubt.

Auf analogem Wege sind auch die Statusdaten des i550 im Prozessabbild änderbar:

Statusword:

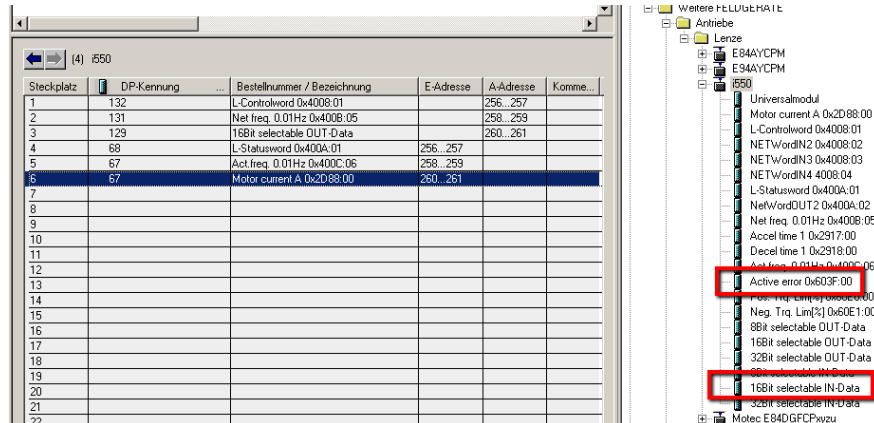


Hier kann zusätzlich noch jedes Bit des Statuswortes einzeln invertiert werden. (Soll für das Contolword auch noch implementiert werden).

Die Zuordnung der Bits zu den freien Ausgängen (xFreeState_1-4) des Bausteins ist folgende:
 xFreeState_1 <-> NetWordOut1.01

xFreeState_2 <-> NetWordOut1.04
 xFreeState_3 <-> NetWordOut1.08
 xFreeState_4 <-> NetWordOut1.09

Das freie Statuswort des Bausteins ist mit dem 3 Eingangswort verknüpft und per Default mit dem Aktuellen Motorstrom in Ampere verknüpft (Index 0x2D88:000). Hier könnte auch ein anderer Wert aus der Auswahlliste genommen werden oder aber über das Modul „16Bit selectable IN-Data“ ein anderer Index verwendet werden.



1.2.2 Einbinden des Funktionsbausteins „LCB-ActuatorSpeed_V2_1 in das Steuerungsprogramm der S7

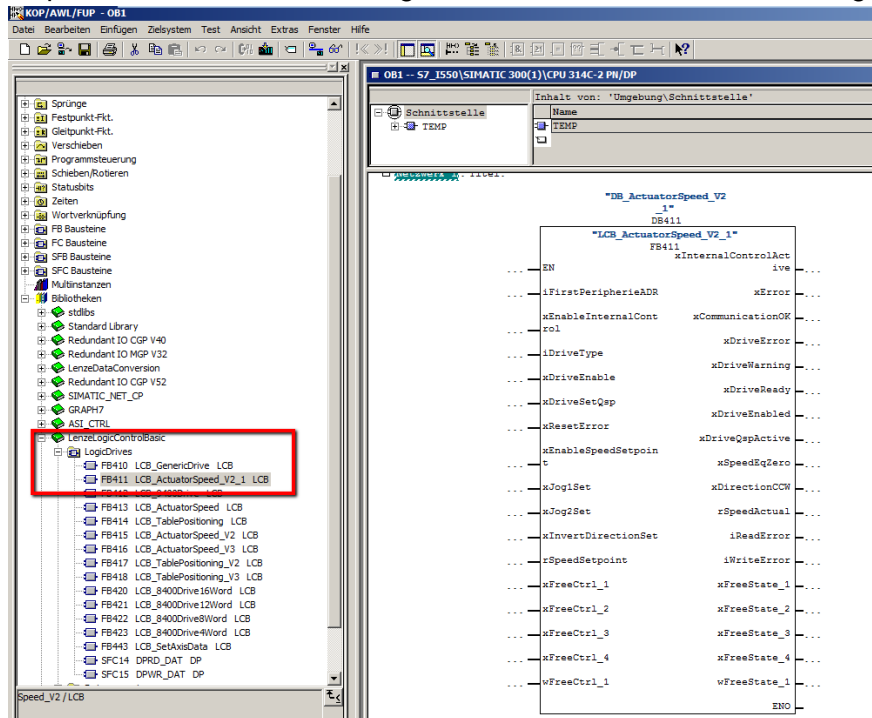
Zur genauen Funktionalitätsbeschreibung des Bausteins sowie der Ein- / Ausgangsdaten sei hier auf die Dokumentation „S7_Lenze_Application_Sample_V3-0 / V4.0_DE.PDF“ verwiesen; in dieser Kurzanleitung geht es lediglich um die prinzipielle Vorgehensweise.

Da der i550 auch Diagnosemeldungen (z.B.: Zwischenkreisunterspannung) an die S7 Steuerung versendet empfiehlt es sich folgende Organisationsbausteine in das S7 Projekt zu implementieren. Ohne diese würde die Steuerung ansonsten in Stop gehen sobald eine Diagnosemeldung auftritt.

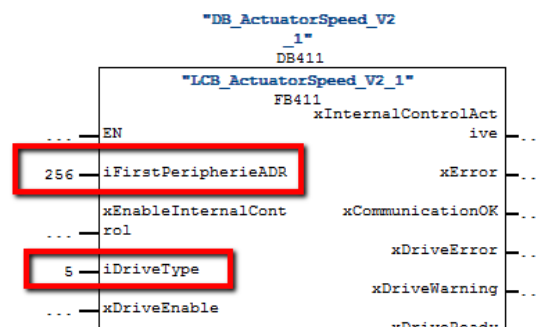
OB80	CYCL_FLT	AWL
OB82	I/O_FLT1	AWL
OB83	I/O_FLT2	AWL
OB86	RACK_FLT	AWL
OB121	PROG_ERR	FUP
OB122	MOD_ERR	FUP

Die Bausteine müssen keinen weiteren Programmcode enthalten.

Im Beispiel wird im OB1 der benötigte Baustein aus der Bibliothek eingefügt:



Für einen ersten Test über die in der Bibliothek enthaltene Variablentabelle sollten nur folgende Eingänge beschaltet werden:



Die Startadresse (iFirstPeripherieADR) kann aus der Hardwarekonfiguration entnommen werden:

Steckplatz	DP-Kennung	Bestellnummer / Bezeichnung	E-Adresse	A-Adresse	Komme...
1	132	L-Controlword 0x4008:01		256...257	
2	131	Net freq. 0.01Hz 0x400B:05		258...259	
3	129	16Bit selectable OUT-Data		260...261	
4	68	L-Statusword 0x400A:01	256...257		
5	67	Act.freq. 0.01Hz 0x400C:06	258...259		
6	67	Motor current A 0x2D88:00	260...261		

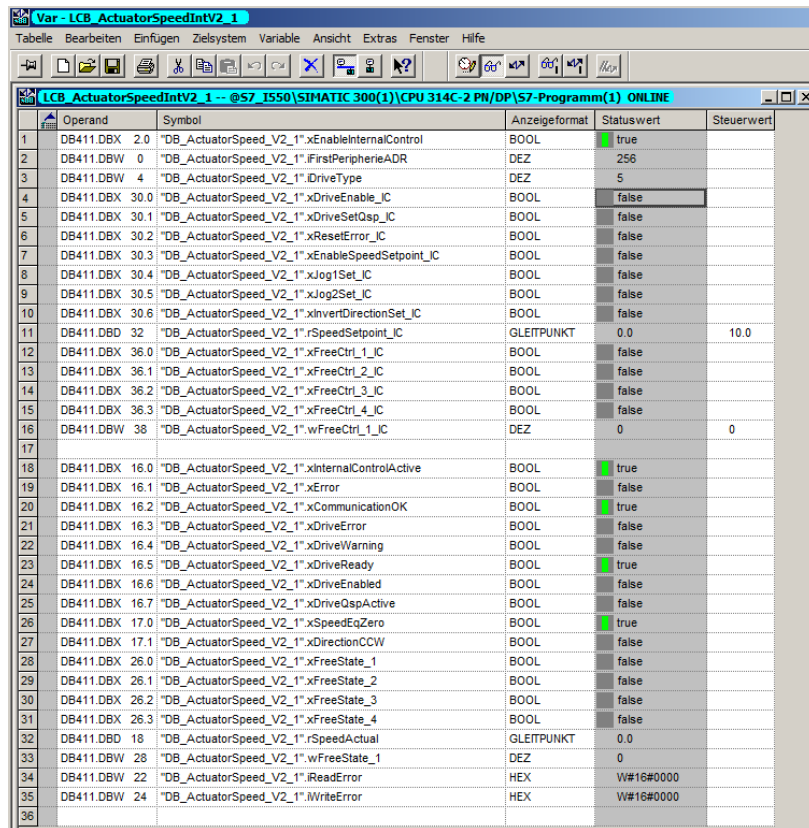
Über iDriveType = 5 wird dem Baustein mitgeteilt das ein i550 angesteuert wird.

Die weiteren Ein- und Ausgänge müssen im Anwenderprogramm entsprechend der Maschinenfunktion verschaltet werden.

Alles übersetzen und Hardwarekonfiguration und Bausteine in die Steuerung laden.

Aktuell muss nach der erstmaligen Parametrierung einmalig ein „Netzschalten“ für PLC und i550 durchgeführt werden damit die Parametrierung übernommen wird.

Anschließend kann über die Variablentabelle „VAR – LCB_ActuatorSpeedIntV2_1“ der Antrieb angesteuert und getestet werden:



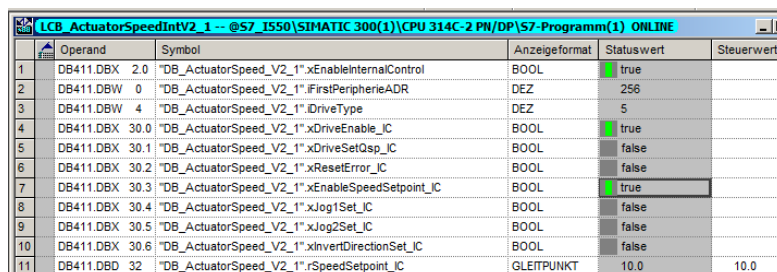
Operand	Symbol	Anzeigeformat	Statuswert	Steuerwert
DB411.DBX 2.0	"DB_ActuatorSpeed_V2_1".xEnableInternalControl	BOOL	true	
DB411.DBW 0	"DB_ActuatorSpeed_V2_1".rFirstPeripherieADR	DEZ	256	
DB411.DBW 4	"DB_ActuatorSpeed_V2_1".rDriveType	DEZ	5	
DB411.DBX 30.0	"DB_ActuatorSpeed_V2_1".xDriveEnable_IC	BOOL	false	
DB411.DBX 30.1	"DB_ActuatorSpeed_V2_1".xDriveSetQsp_IC	BOOL	false	
DB411.DBX 30.2	"DB_ActuatorSpeed_V2_1".xResetError_IC	BOOL	false	
DB411.DBX 30.3	"DB_ActuatorSpeed_V2_1".xEnableSpeedSetpoint_IC	BOOL	false	
DB411.DBX 30.4	"DB_ActuatorSpeed_V2_1".xJog1Set_IC	BOOL	false	
DB411.DBX 30.5	"DB_ActuatorSpeed_V2_1".xJog2Set_IC	BOOL	false	
DB411.DBX 30.6	"DB_ActuatorSpeed_V2_1".xInvertDirectionSet_IC	BOOL	false	
DB411.DBD 32	"DB_ActuatorSpeed_V2_1".rSpeedSetpoint_IC	GLEITPUNKT	0.0	10.0
DB411.DBX 36.0	"DB_ActuatorSpeed_V2_1".xFreeCtrl_1_IC	BOOL	false	
DB411.DBX 36.1	"DB_ActuatorSpeed_V2_1".xFreeCtrl_2_IC	BOOL	false	
DB411.DBX 36.2	"DB_ActuatorSpeed_V2_1".xFreeCtrl_3_IC	BOOL	false	
DB411.DBX 36.3	"DB_ActuatorSpeed_V2_1".xFreeCtrl_4_IC	BOOL	false	
DB411.DBW 38	"DB_ActuatorSpeed_V2_1".wFreeCtrl_1_IC	DEZ	0	0
DB411.DBX 16.0	"DB_ActuatorSpeed_V2_1".xInternalControlActive	BOOL	true	
DB411.DBX 16.1	"DB_ActuatorSpeed_V2_1".xError	BOOL	false	
DB411.DBX 16.2	"DB_ActuatorSpeed_V2_1".xCommunicationOK	BOOL	true	
DB411.DBX 16.3	"DB_ActuatorSpeed_V2_1".xDriveError	BOOL	false	
DB411.DBX 16.4	"DB_ActuatorSpeed_V2_1".xDriveWarning	BOOL	false	
DB411.DBX 16.5	"DB_ActuatorSpeed_V2_1".xDriveReady	BOOL	true	
DB411.DBX 16.6	"DB_ActuatorSpeed_V2_1".xDriveEnabled	BOOL	false	
DB411.DBX 16.7	"DB_ActuatorSpeed_V2_1".xDriveQspActive	BOOL	false	
DB411.DBX 17.0	"DB_ActuatorSpeed_V2_1".xSpeedEqZero	BOOL	true	
DB411.DBX 17.1	"DB_ActuatorSpeed_V2_1".xDirectionCCW	BOOL	false	
DB411.DBX 26.0	"DB_ActuatorSpeed_V2_1".xFreeState_1	BOOL	false	
DB411.DBX 26.1	"DB_ActuatorSpeed_V2_1".xFreeState_2	BOOL	false	
DB411.DBX 26.2	"DB_ActuatorSpeed_V2_1".xFreeState_3	BOOL	false	
DB411.DBX 26.3	"DB_ActuatorSpeed_V2_1".xFreeState_4	BOOL	false	
DB411.DBD 18	"DB_ActuatorSpeed_V2_1".rSpeedActual	GLEITPUNKT	0.0	
DB411.DBW 28	"DB_ActuatorSpeed_V2_1".wFreeState_1	DEZ	0	
DB411.DBW 22	"DB_ActuatorSpeed_V2_1".rReadError	HEX	W#16#0000	
DB411.DBW 24	"DB_ActuatorSpeed_V2_1".rWriteError	HEX	W#16#0000	

Durch das setzen des Bits „xEnableInternalControl“ werden alle Eingänge des Bausteins vom Steuerungsprogramm abgekoppelt und nur die Sollwertvorgaben der Variablentabelle sind wirksam.

Folgende Controlbits müssen in der Variablentabelle gesetzt sein damit der Antrieb dreht:

xDriveEnable_IC = Antriebsfreigabe

xEnableSpeedSetpoint_IC = Sollwertfreigabe, der Antrieb dreht mit dem unter „rSpeedSetpoint_IC vorgegeben Sollwert. Nur positive Werte sind hier zulässig, Richtungsumkehr über das Controlbit xInvertDirectionSet_IC.

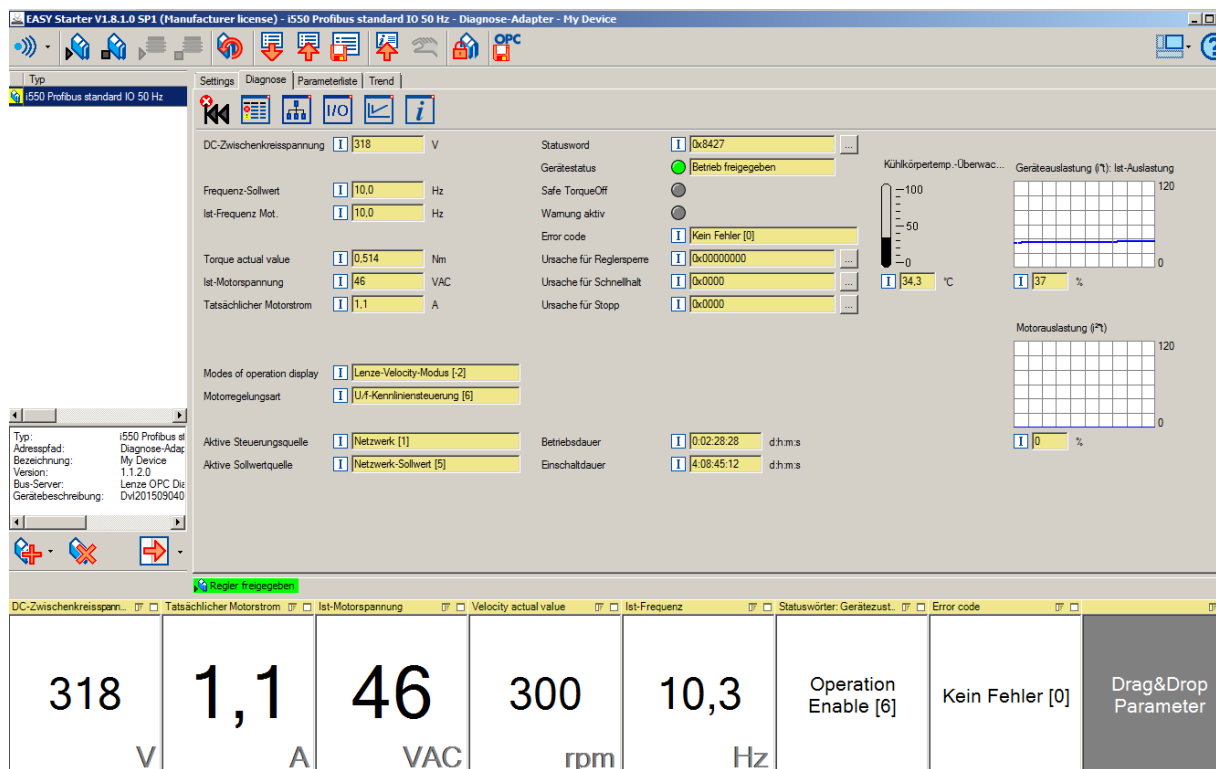


Operand	Symbol	Anzeigeformat	Statuswert	Steuerwert
DB411.DBX 2.0	"DB_ActuatorSpeed_V2_1".xEnableInternalControl	BOOL	true	
DB411.DBW 0	"DB_ActuatorSpeed_V2_1".rFirstPeripherieADR	DEZ	256	
DB411.DBW 4	"DB_ActuatorSpeed_V2_1".rDriveType	DEZ	5	
DB411.DBX 30.0	"DB_ActuatorSpeed_V2_1".xDriveEnable_IC	BOOL	true	
DB411.DBX 30.1	"DB_ActuatorSpeed_V2_1".xDriveSetQsp_IC	BOOL	false	
DB411.DBX 30.2	"DB_ActuatorSpeed_V2_1".xResetError_IC	BOOL	false	
DB411.DBX 30.3	"DB_ActuatorSpeed_V2_1".xEnableSpeedSetpoint_IC	BOOL	true	
DB411.DBX 30.4	"DB_ActuatorSpeed_V2_1".xJog1Set_IC	BOOL	false	
DB411.DBX 30.5	"DB_ActuatorSpeed_V2_1".xJog2Set_IC	BOOL	false	
DB411.DBX 30.6	"DB_ActuatorSpeed_V2_1".xInvertDirectionSet_IC	BOOL	false	
DB411.DBD 32	"DB_ActuatorSpeed_V2_1".rSpeedSetpoint_IC	GLEITPUNKT	10.0	10.0

Statuswerte:

18	DB411.DBX	16.0	"DB_ActuatorSpeed_V2_1".xInternalControlActive	BOOL	true
19	DB411.DBX	16.1	"DB_ActuatorSpeed_V2_1".xError	BOOL	false
20	DB411.DBX	16.2	"DB_ActuatorSpeed_V2_1".xCommunicationOK	BOOL	true
21	DB411.DBX	16.3	"DB_ActuatorSpeed_V2_1".xDriveError	BOOL	false
22	DB411.DBX	16.4	"DB_ActuatorSpeed_V2_1".xDriveWarning	BOOL	false
23	DB411.DBX	16.5	"DB_ActuatorSpeed_V2_1".xDriveReady	BOOL	false
24	DB411.DBX	16.6	"DB_ActuatorSpeed_V2_1".xDriveEnabled	BOOL	true
25	DB411.DBX	16.7	"DB_ActuatorSpeed_V2_1".xDriveQspActive	BOOL	false
26	DB411.DBX	17.0	"DB_ActuatorSpeed_V2_1".xSpeedEqZero	BOOL	false
27	DB411.DBX	17.1	"DB_ActuatorSpeed_V2_1".xDirectionCCW	BOOL	false
28	DB411.DBX	26.0	"DB_ActuatorSpeed_V2_1".xFreeState_1	BOOL	false
29	DB411.DBX	26.1	"DB_ActuatorSpeed_V2_1".xFreeState_2	BOOL	false
30	DB411.DBX	26.2	"DB_ActuatorSpeed_V2_1".xFreeState_3	BOOL	false
31	DB411.DBX	26.3	"DB_ActuatorSpeed_V2_1".xFreeState_4	BOOL	false
32	DB411.DBX	18	"DB_ActuatorSpeed_V2_1".rSpeedActual	GLEITPUNKT	10.0
33	DB411.DBW	28	"DB_ActuatorSpeed_V2_1".wFreeState_1	DEZ	11
34	DB411.DBW	22	"DB_ActuatorSpeed_V2_1".iReadError	HEX	W#16#0000
35	DB411.DBW	24	"DB_ActuatorSpeed_V2_1".iWriteError	HEX	W#16#0000

Status im EASY Starter:



1.3 Integration unter dem Siemens TIA-Portal

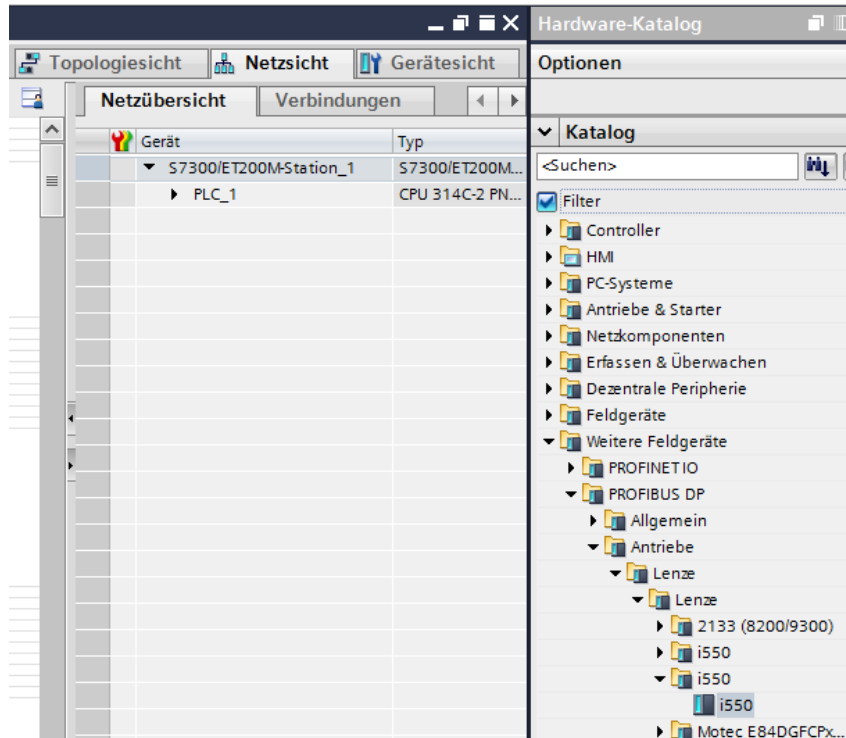
Für die Einbindung unter TIA wird die GSD Datei des i550 benötigt. Diese muss über den Hardwaremanager in den Hardwarekatalog des TIA-Portals eingebunden werden.

Um den Baustein „LCB_ActuatorSpeed_V2_1“ aus der Lenze Beispielbibliothek zu verwenden muss diese noch eingebunden werden.

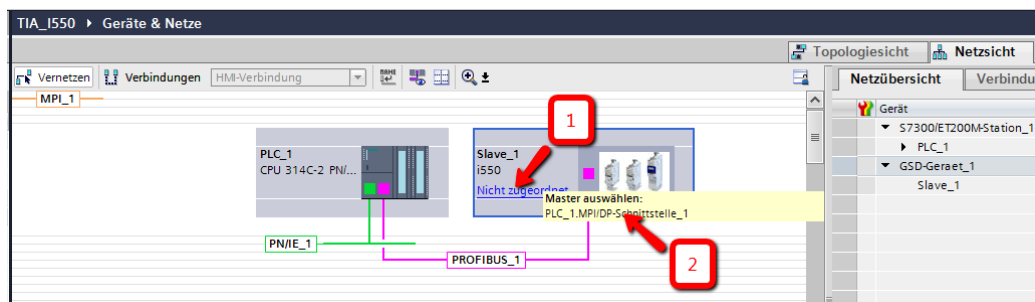
1.3.1 Hardwarekonfiguration im TIA-Portal mit S7-300 Steuerung

In der Gerätekonfiguration des TIA Portals die Netzansicht auswählen

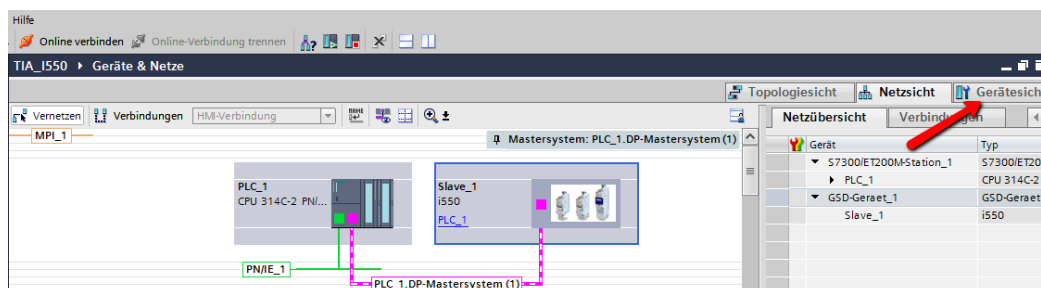
Auswahl des i550 im Hardwarekatalog unter „Weitere Feldgeräte/Profibus-DP/Antriebe/Lenze“ den i550 auswählen.



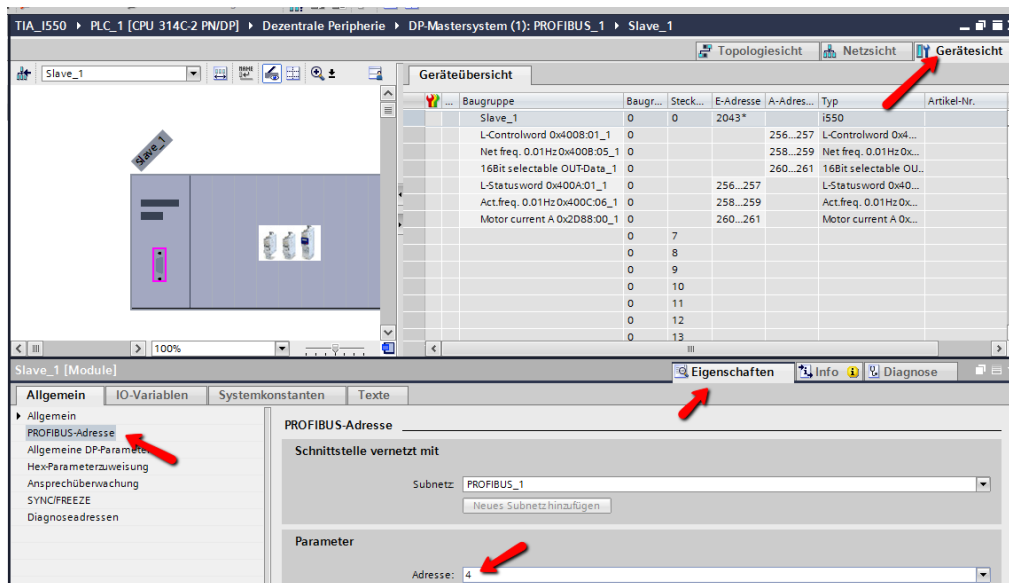
Wird der i550 per „Drag and Drop“ in die Netzansicht auf den Profibus gezogen, wird die Profibusverbindung zur SPS zwar dargestellt. Der i550 ist jedoch noch nicht der SPS zugeordnet. Dies erfolgt indem man mit der Maus auf den Text „Nicht zugeordnet“ in dem Symbol für den i550 klickt. In dem sich dann öffnenden Fenster wird dann die gewünschte SPS ausgewählt.



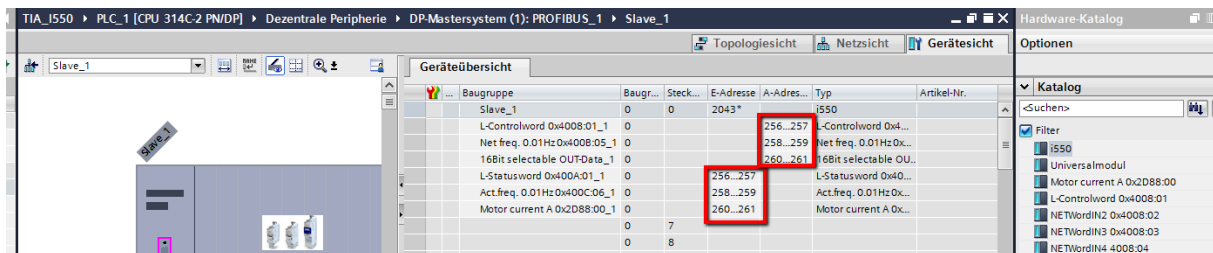
Nun den i550 markieren und in die Geräteansicht wechseln:



Unter den Geräteeigenschaften zunächst die Profibusadresse einstellen, im Beispielprojekt die Adresse 4:



Da in der GSD Datei des i550 eine Standardbelegung hinterlegt ist, ist die Standard IO-Konfiguration der Prozessdaten bereits angelegt. Sie muss nicht mehr manuell angelegt werden, ist aber editierbar bzw. erweiterbar.



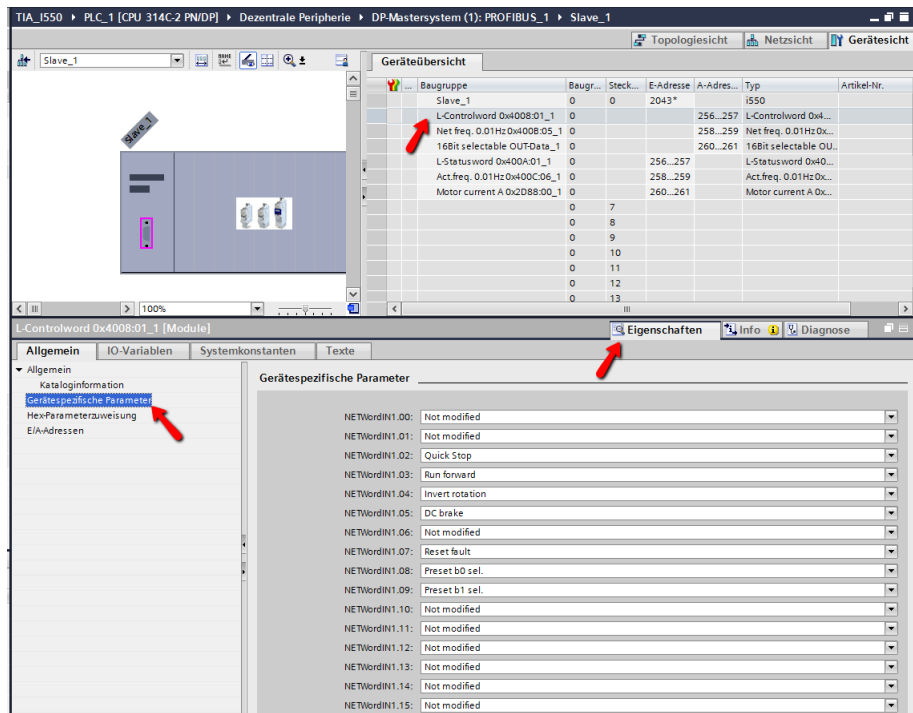
Für den Baustein „LCB_ActuatorSpeed_V2_1“ wird eine Datenlänge von jeweils 3 Worten Eingangsdaten und Ausgangsdaten benötigt. **Damit der Baustein einwandfrei arbeitet ist darauf zu achten das die Adressen ohne Sprünge direkt aufeinander folgen, auch müssen die Anfangsadressen des E/A Bereiches identisch sein (hier z.B.: 256).** Die Grundkonfiguration muss für die Basisfunktionalität nicht mehr angepasst werden. Individuelle Anpassungen sind jedoch möglich.

1.3.1.1 Anpassung des Prozessdaten-Mappings

Das Prozessdaten Mapping kann nur auf der Steuerungsseite im Hardwaremanager geändert werden. Die Siemens-Steuerung schreibt dieses bei jedem Neustart auf den i550 via Profibus herunter.

Die Belegung der Freien Eingangsbits (xFreeCtrl_1-4) des Lenze FB kann direkt im Eigenschaftendialog der Geräteansicht im Controlword geändert werden.

Controlword markieren und unter Eigenschaften Gerätespezifische Parameter anwählen:



Die freien Controlbits sind wie folgt zugeordnet:

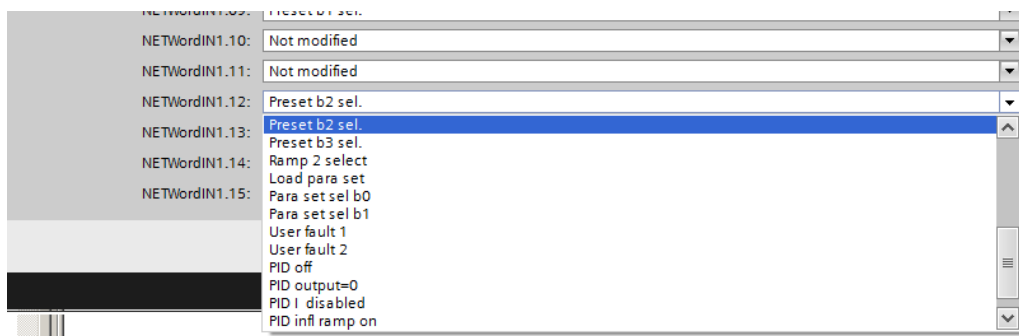
xFreeCtrl_1 <-> NetWordIn1.11

xFreeCtrl_2 <-> NetWordIn1.13

xFreeCtrl_3 <-> NetWordIn1.14

xFreeCtrl_4 <-> NetWordIn1.15

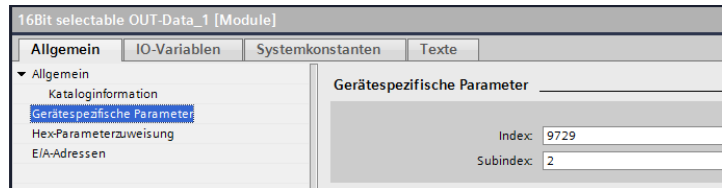
Für jedes einzelne Bit kann über das Drop-Down Menü eine neue Funktion verknüpft werden:



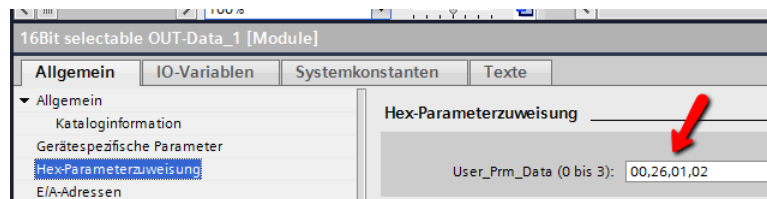
Das freie Eingangswort wFreeCtrl_1 des Bausteins ist mit dem 3 Ausgangswort des Prozessabbildes verknüpft (16Bit selectable Out-Data) und kann ebenfalls nur im Eigenschaftendialog der Geräteansicht auf S7 Seite verändert werden:

Im Eigenschaftendialog kann selektiert werden welcher Index im Regler geschrieben werden soll:

Unter Gerätespezifische Parameter steht der Index und Subindex in Dezimaler Darstellung,



unter Hex-Parameterzuweisung sieht man direkt den HEX-Index: (0x2601:02 Defaultbelegung):



Adresse	Display Code	Bezeichnung	Wert
0x2601:000	PAR0202:000	highest sub-index supported (generated)	3
0x2601:001	PAR0202:001	Keypad-Sollwertbefehle: Frequenz-Sollwert	20,0
0x2601:002	PAR0202:002	Keypad-Sollwertbefehle: Prozessregler	0,00
0x2601:003	PAR0202:003	Keypad-Sollwertbefehle: Drehmoment-Sollwert	100,0

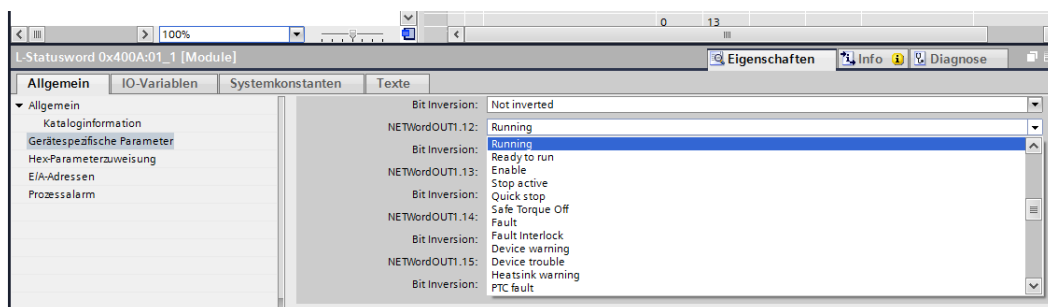
Soll ein anderer Sollwert übertragen werden so muss dieser Index die Eigenschaft „mappable“ besitzen, dies kann der Parameterbeschreibung in der Dokumentation entnommen werden.

0x2601	Keypad-Sollwerte			
	001: Frequenz-Sollwert	20.0 Hz	0.0 ... 599.0 Hz	U16 / 10 / P / r
	002: Prozessregler-Sollwert	0.00 PUnit	-300.00 ... 300.00 PUnit	I16 / 100 / P / r

Das kleine "r" steht für Receive-Mapping erlaubt.

Auf analogem Wege sind auch die Statusdaten des i550 im Prozessabbild änderbar:

Statusword:



Hier kann zusätzlich noch jedes Bit des Statuswortes einzeln invertiert werden. (Soll für das Contolword auch noch implementiert werden).

Die Zuordnung der Bits zu den freien Ausgänge (xFreeState_1-4) des Bausteins ist folgende:

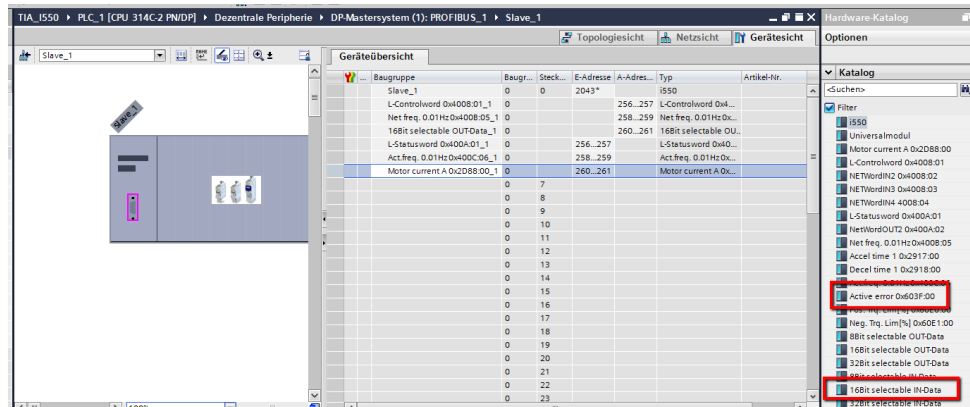
xFreeState_1 <-> NetWordOut1.01

xFreeState_2 <-> NetWordOut1.04

xFreeState_3 <-> NetWordOut1.08

xFreeState_4 <-> NetWordOut1.09

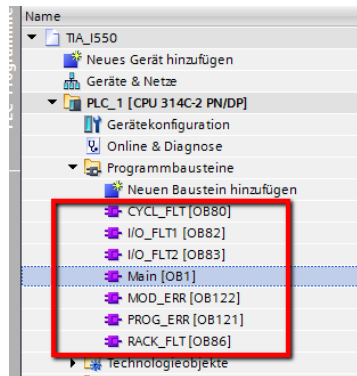
Das freie Statuswort des Bausteins ist mit dem 3 Eingangswort verknüpft und per Default mit dem Aktuellen Motorstrom in Ampere verknüpft (Index 0x2D88:000). Hier könnte auch ein anderer Wert aus der Auswahlliste genommen werden oder aber über das Modul „16Bit selectable IN-Data“ ein anderer Index verwendet werden.



1.3.2 Einbinden des Funktionsbausteins „LCB-ActuatorSpeed_V2_1 in das Steuerungsprogramm der S7

Zur genauen Funktionsitätsbeschreibung des Bausteins sowie der Ein- / Ausgangsdaten sei hier auf die Dokumentation „S7_Lenze_Application_Sample_V4-0_DE.PDF“ verwiesen; in dieser Kurzanleitung geht es lediglich um die prinzipielle Vorgehensweise.

Da der i550 auch Diagnosemeldungen (z.B.: Zwischenkreisunterspannung) an die S7 Steuerung versendet empfiehlt es sich folgende Organisationsbausteine in das S7 Projekt zu implementieren. Ohne diese würde die Steuerung ansonsten in Stop gehen sobald eine Diagnosemeldung auftritt.



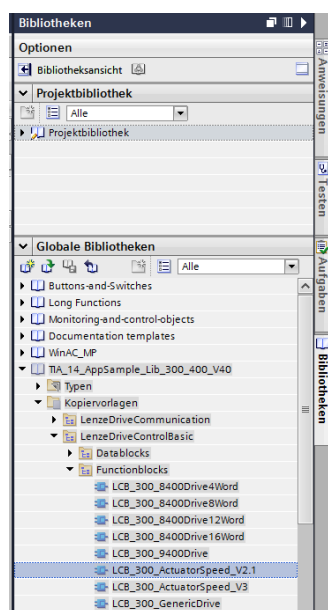
Die Bausteine müssen keinen weiteren Programmcode enthalten.

Aus der dem SPS Typen entsprechenden Bibliothek folgende Baustein - Kopiervorlagen in das Projekt kopieren:

„LCB_300_ActuatorSpeed_V2.1“

„LCB_300_ActuatorSpeed_V2.1_DB“

und die Beobachtungstabelle „LCB_300_Actuator_Speed_V2.1“ zum steuern des Antriebes.



Im Beispiel wird im OB1 der benötigte Baustein aus der Bibliothek eingefügt.

TIA_I550 > PLC_1 [CPU 314C-2 PN/DP] > Programmbausteine > Main [OB1]

Main

	Name	Datentyp	Offset	Defaultwert
1	Temp			
2	OB1_EV_CLASS	Byte	0.0	
3	OB1_SCAN_1	Byte	1.0	

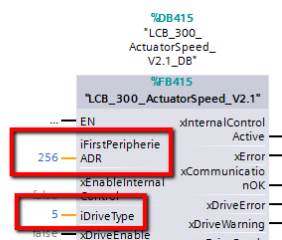
& >= 1 [??] -> -o| -> [-=]

%DB415
*LCB_300_
ActuatorSpeed_
V2.1_DB*

%FB415
LCB_300_ActuatorSpeed_V2.1

...	EN	xInternalControl	Active	...
256	iFirstPeripherieADR	xError
false	xEnableInternalControl	xCommunicationOK
5	iDriveType	xDriveError
false	xDriveEnable	xDriveWarning
false	xDriveSetQsp	xDriveReady
false	xResetError	xDriveEnabled
		xDriveQspActive
false	xEnableSpeedSetpoint	xSpeedEqZero
false	xlog1Set	xDirectionCCW
false	xlog2Set	rSpeedActual
	xInvertDirection	iReadError
false	Set	iWriteError
0.0	rSpeedSetpoint	xFreeState_1
false	xFreeCtrl_1	xFreeState_2
false	xFreeCtrl_2	xFreeState_3
false	xFreeCtrl_3	xFreeState_4
false	xFreeCtrl_4	wFreeState_1
16#0	wFreeCtrl_1	ENO

Für einen ersten Test über die in der Bibliothek enthaltene Beobachtungstabelle sollten nur folgende Eingänge beschaltet werden:



Über iDriveType = 5 wird dem Baustein mitgeteilt das ein i550 angesteuert wird.

Die Startadresse (iFirstPeripherieADR) kann aus der Hardwarekonfiguration entnommen werden:

Geräteübersicht						
Baugruppe	Baugr...	Steck...	E-Adress...	A-Adress...	Typ	Artike...
Slave_1	0	0	2043*	256...257	i550	
L-Controlword 0x4008:01_1	0			258...259	L-Controlword 0x4...	
Net freq. 0.01Hz 0x4008:05_1	0			260...261	Net freq. 0.01Hz 0x...	
16Bit selectable OUT-Data_1	0				16Bit selectable OU...	
L-Statusword 0x400A:01_1	0				L-Statusword 0x40...	
Act.freq. 0.01Hz 0x400C:06_1	0				Act.freq. 0.01Hz 0x...	
Motor current A 0x2D88:00_1	0				Motor current A 0x...	
	0	7				
	0	8				

Die weiteren Ein- und Ausgänge müssen im Anwenderprogramm entsprechend der Maschinenfunktion verschaltet werden.

Alles übersetzen und Hardwarekonfiguration und Bausteine in die Steuerung laden.

Aktuell muss nach der erstmaligen Parametrierung einmalig ein „Netzschalten“ für PLC und i550 durchgeführt werden damit die Parametrierung übernommen wird.

Anschließend kann über die Variablentabelle „VAR – LCB_ActuatorSpeedIntV2_1“ der Antrieb angesteuert und getestet werden:

TIA_I550 > PLC_1 [CPU 314C-2 PN/DP] > Beobachtungs- und Forcetabellen > LCB_300_Actuator_Speed_V2.1						
	Name	Adresse	Anzeigeformat	Beobachtungswert	Steuerwert	
1	*LCB_300_ActuatorSpeed_V2.1_DB*.xEnableInternalCont...	%DB415.DBX2.0	BOOL	<input checked="" type="checkbox"/> TRUE	TRUE	
2	*LCB_300_ActuatorSpeed_V2.1_DB*.iFirstPeripherieADR	%DB415.DBW0	DEZ +/-	256		
3	*LCB_300_ActuatorSpeed_V2.1_DB*.iDriveType	%DB415.DBW4	DEZ +/-	5		
4	*LCB_300_ActuatorSpeed_V2.1_DB*.xDriveEnable_IC	%DB415.DBX32.0	BOOL	<input type="checkbox"/> FALSE	FALSE	
5	*LCB_300_ActuatorSpeed_V2.1_DB*.xDriveSetQsp_IC	%DB415.DBX32.1	BOOL	<input type="checkbox"/> FALSE	FALSE	
6	*LCB_300_ActuatorSpeed_V2.1_DB*.xResetError_IC	%DB415.DBX32.2	BOOL	<input type="checkbox"/> FALSE	FALSE	
7	*LCB_300_ActuatorSpeed_V2.1_DB*.xEnableSpeedSetpoint_I	%DB415.DBX32.3	BOOL	<input type="checkbox"/> FALSE	FALSE	
8	*LCB_300_ActuatorSpeed_V2.1_DB*.xlog1Set_IC	%DB415.DBX32.4	BOOL	<input type="checkbox"/> FALSE	FALSE	
9	*LCB_300_ActuatorSpeed_V2.1_DB*.xlog2Set_IC	%DB415.DBX32.5	BOOL	<input type="checkbox"/> FALSE	FALSE	
10	*LCB_300_ActuatorSpeed_V2.1_DB*.xInvertDirectionSet_IC	%DB415.DBX32.6	BOOL	<input type="checkbox"/> FALSE	FALSE	
11	*LCB_300_ActuatorSpeed_V2.1_DB*.rSpeedSetpoint_IC	%DB415.DBX32.7	Gleitpunktzahl	0.0	-10.0	
12	*LCB_300_ActuatorSpeed_V2.1_DB*.xFreeCtrl_1_IC	%DB415.DBX38.0	BOOL	<input type="checkbox"/> FALSE	TRUE	
13	*LCB_300_ActuatorSpeed_V2.1_DB*.xFreeCtrl_2_IC	%DB415.DBX38.1	BOOL	<input type="checkbox"/> FALSE	FALSE	
14	*LCB_300_ActuatorSpeed_V2.1_DB*.xFreeCtrl_3_IC	%DB415.DBX38.2	BOOL	<input type="checkbox"/> FALSE	TRUE	
15	*LCB_300_ActuatorSpeed_V2.1_DB*.xFreeCtrl_4_IC	%DB415.DBX38.3	BOOL	<input type="checkbox"/> FALSE	FALSE	
16	*LCB_300_ActuatorSpeed_V2.1_DB*.wFreeCtrl_1_IC	%DB415.DBW40	DEZ	0		
17	*LCB_300_ActuatorSpeed_V2.1_DB*.wOUT	%DB415.DBW56	Hex	16#0000		
18	*LCB_300_ActuatorSpeed_V2.1_DB*.xInternalControlActive	%DB415.DBX16.0	BOOL	<input checked="" type="checkbox"/> TRUE		
19	*LCB_300_ActuatorSpeed_V2.1_DB*.xError	%DB415.DBX16.1	BOOL	<input type="checkbox"/> FALSE		
20	*LCB_300_ActuatorSpeed_V2.1_DB*.xCommunicationOK	%DB415.DBX16.2	BOOL	<input checked="" type="checkbox"/> TRUE		
21	*LCB_300_ActuatorSpeed_V2.1_DB*.xDriveError	%DB415.DBX16.3	BOOL	<input type="checkbox"/> FALSE		
22	*LCB_300_ActuatorSpeed_V2.1_DB*.xDriveWarning	%DB415.DBX16.4	BOOL	<input type="checkbox"/> FALSE		
23	*LCB_300_ActuatorSpeed_V2.1_DB*.xDriveReady	%DB415.DBX16.5	BOOL	<input checked="" type="checkbox"/> TRUE		
24	*LCB_300_ActuatorSpeed_V2.1_DB*.xDriveEnabled	%DB415.DBX16.6	BOOL	<input type="checkbox"/> FALSE		
25	*LCB_300_ActuatorSpeed_V2.1_DB*.xDriveQspActive	%DB415.DBX16.7	BOOL	<input type="checkbox"/> FALSE		
26	*LCB_300_ActuatorSpeed_V2.1_DB*.xSpeedEqZero	%DB415.DBX17.0	BOOL	<input checked="" type="checkbox"/> TRUE		
27	*LCB_300_ActuatorSpeed_V2.1_DB*.xDirectionCCW	%DB415.DBX17.1	BOOL	<input type="checkbox"/> FALSE		
28	*LCB_300_ActuatorSpeed_V2.1_DB*.xFreeState_1	%DB415.DBX26.0	BOOL	<input type="checkbox"/> FALSE	TRUE	
29	*LCB_300_ActuatorSpeed_V2.1_DB*.xFreeState_2	%DB415.DBX26.1	BOOL	<input type="checkbox"/> FALSE	TRUE	
30	*LCB_300_ActuatorSpeed_V2.1_DB*.xFreeState_3	%DB415.DBX26.2	BOOL	<input type="checkbox"/> FALSE		
31	*LCB_300_ActuatorSpeed_V2.1_DB*.xFreeState_4	%DB415.DBX26.3	BOOL	<input type="checkbox"/> FALSE		
32	*LCB_300_ActuatorSpeed_V2.1_DB*.rSpeedActual	%DB415.DBX18	Gleitpunktzahl	0.0		
33	*LCB_300_ActuatorSpeed_V2.1_DB*.wFreeState_1	%DB415.DBW28	Hex	16#0000		
34	*LCB_300_ActuatorSpeed_V2.1_DB*.iReadError	%DB415.DBW22	DEZ +/-	0		
35	*LCB_300_ActuatorSpeed_V2.1_DB*.iWriteError	%DB415.DBW24	DEZ +/-	0		

Durch das setzen des Bits „xEnableInternalControl“ werden alle Eingänge des Bausteins vom Steuerungsprogramm abgekoppelt und nur die Sollwertvorgaben der Variablentabelle sind wirksam.

Folgende Controlbits müssen in der Variablentabelle gesetzt sein damit der Antrieb dreht:

xDriveEnable_IC = Antriebsfreigabe

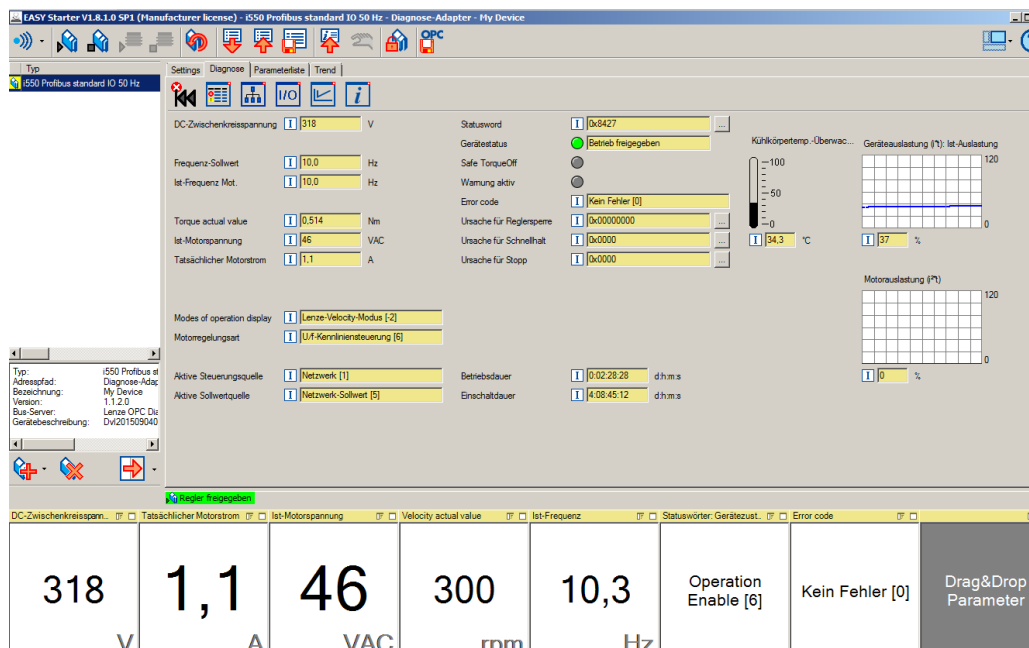
xEnableSpeedSetPoint_IC = Sollwertfreigabe, der Antrieb dreht mit dem unter „rSpeedSetpoint_IC vorgegeben Sollwert. Nur positive Werte sind hier zulässig, Richtungsumkehr über das Controlbit xInvertDirectionSet_IC.

TIA_I550 ▶ PLC_1 [CPU 314C-2 PN/DP] ▶ Beobachtungs- und Forcetabellen ▶ LCB_300_Actuator_Speed_V2.1						
	Name	Adresse	Anzeigeformat	Beobachtungswert	Steuerwert	
1	"LCB_300_ActuatorSpeed_V2.1_DB".xEnableInternalControl	%DB415.DBX2.0	BOOL	<input checked="" type="checkbox"/> TRUE	TRUE	
2	"LCB_300_ActuatorSpeed_V2.1_DB".iFirstPeripherieADR	%DB415.DBW0	DEZ+/-	256		
3	"LCB_300_ActuatorSpeed_V2.1_DB".iDriveType	%DB415.DBW4	DEZ+/-	5		
4	"LCB_300_ActuatorSpeed_V2.1_DB".xDriveEnable_IC	%DB415.DBX32.0	BOOL	<input checked="" type="checkbox"/> TRUE	TRUE	
5	"LCB_300_ActuatorSpeed_V2.1_DB".xDriveSetQsp_IC	%DB415.DBX32.1	BOOL	<input type="checkbox"/> FALSE	FALSE	
6	"LCB_300_ActuatorSpeed_V2.1_DB".xResetError_IC	%DB415.DBX32.2	BOOL	<input type="checkbox"/> FALSE		
7	"LCB_300_ActuatorSpeed_V2.1_DB".xEnableSpeedSetpoint_IC	%DB415.DBX32.3	BOOL	<input checked="" type="checkbox"/> TRUE	TRUE	
8	"LCB_300_ActuatorSpeed_V2.1_DB".xLog1Set_IC	%DB415.DBX32.4	BOOL	<input type="checkbox"/> FALSE	FALSE	
9	"LCB_300_ActuatorSpeed_V2.1_DB".xLog2Set_IC	%DB415.DBX32.5	BOOL	<input type="checkbox"/> FALSE	FALSE	
10	"LCB_300_ActuatorSpeed_V2.1_DB".xInvertDirectionSet_IC	%DB415.DBX32.6	BOOL	<input type="checkbox"/> FALSE	FALSE	
11	"LCB_300_ActuatorSpeed_V2.1_DB".rSpeedSetpoint_IC	%DB415.DBD34	Gleitpunktzahl	10.0	10.0	

Statuswerte:

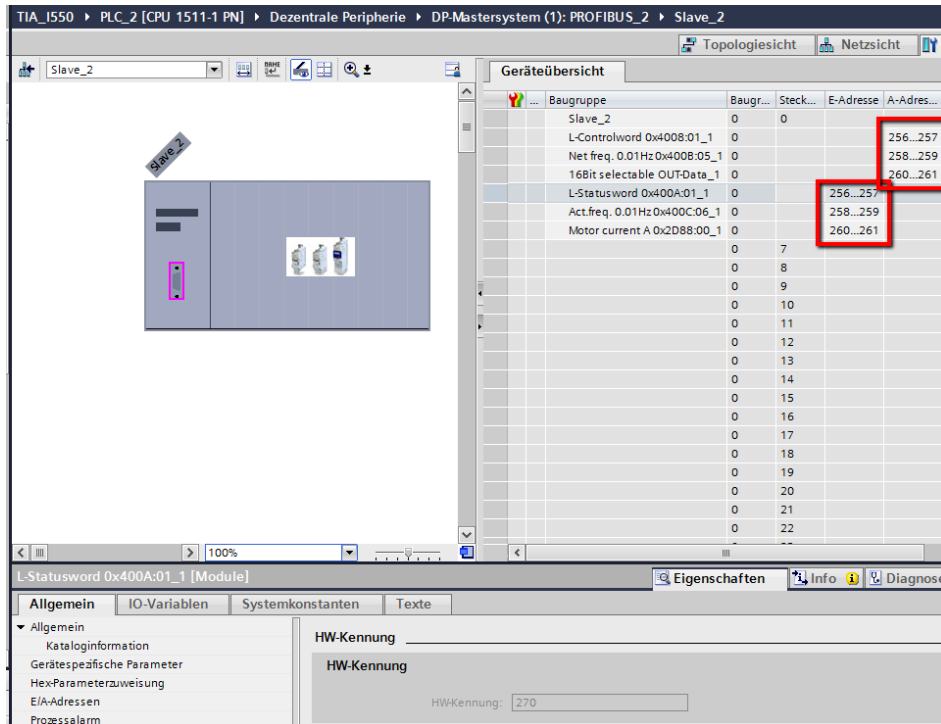
18	"LCB_300_ActuatorSpeed_V2.1_DB".xInternalControlActive	%DB415.DBX16.0	BOOL	<input checked="" type="checkbox"/> TRUE		
19	"LCB_300_ActuatorSpeed_V2.1_DB".xError	%DB415.DBX16.1	BOOL	<input type="checkbox"/> FALSE		
20	"LCB_300_ActuatorSpeed_V2.1_DB".xCommunicationOK	%DB415.DBX16.2	BOOL	<input checked="" type="checkbox"/> TRUE		
21	"LCB_300_ActuatorSpeed_V2.1_DB".xDriveError	%DB415.DBX16.3	BOOL	<input type="checkbox"/> FALSE		
22	"LCB_300_ActuatorSpeed_V2.1_DB".xDriveWarning	%DB415.DBX16.4	BOOL	<input type="checkbox"/> FALSE		
23	"LCB_300_ActuatorSpeed_V2.1_DB".xDriveReady	%DB415.DBX16.5	BOOL	<input type="checkbox"/> FALSE		
24	"LCB_300_ActuatorSpeed_V2.1_DB".xDriveEnabled	%DB415.DBX16.6	BOOL	<input checked="" type="checkbox"/> TRUE		
25	"LCB_300_ActuatorSpeed_V2.1_DB".xDriveQspActive	%DB415.DBX16.7	BOOL	<input type="checkbox"/> FALSE		
26	"LCB_300_ActuatorSpeed_V2.1_DB".xSpeedEqZero	%DB415.DBX17.0	BOOL	<input type="checkbox"/> FALSE		
27	"LCB_300_ActuatorSpeed_V2.1_DB".xDirectionCCW	%DB415.DBX17.1	BOOL	<input type="checkbox"/> FALSE		
28	"LCB_300_ActuatorSpeed_V2.1_DB".xFreeState_1	%DB415.DBX26.0	BOOL	<input type="checkbox"/> FALSE	TRUE	
29	"LCB_300_ActuatorSpeed_V2.1_DB".xFreeState_2	%DB415.DBX26.1	BOOL	<input type="checkbox"/> FALSE	TRUE	
30	"LCB_300_ActuatorSpeed_V2.1_DB".xFreeState_3	%DB415.DBX26.2	BOOL	<input type="checkbox"/> FALSE		
31	"LCB_300_ActuatorSpeed_V2.1_DB".xFreeState_4	%DB415.DBX26.3	BOOL	<input type="checkbox"/> FALSE		
32	"LCB_300_ActuatorSpeed_V2.1_DB".rSpeedActual	%DB415.DBD18	Gleitpunktzahl	10.0		
33	"LCB_300_ActuatorSpeed_V2.1_DB".wFreeState_1	%DB415.DBW28	DEZ	11		
34	"LCB_300_ActuatorSpeed_V2.1_DB".iReadError	%DB415.DBW22	DEZ+/-	0		
35	"LCB_300_ActuatorSpeed_V2.1_DB".iWriteError	%DB415.DBW24	DEZ+/-	0		

Status im EASY Starter:

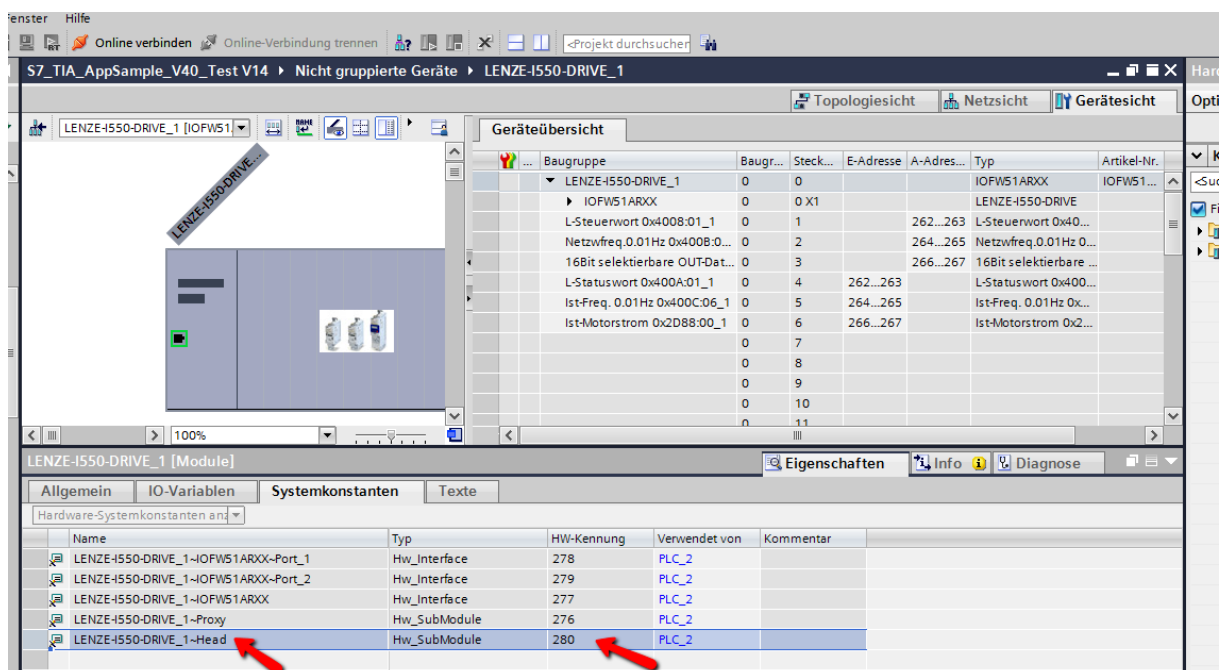


1.3.3 Besonderheiten bei S7-1200 und S7-1500 Steuerungen

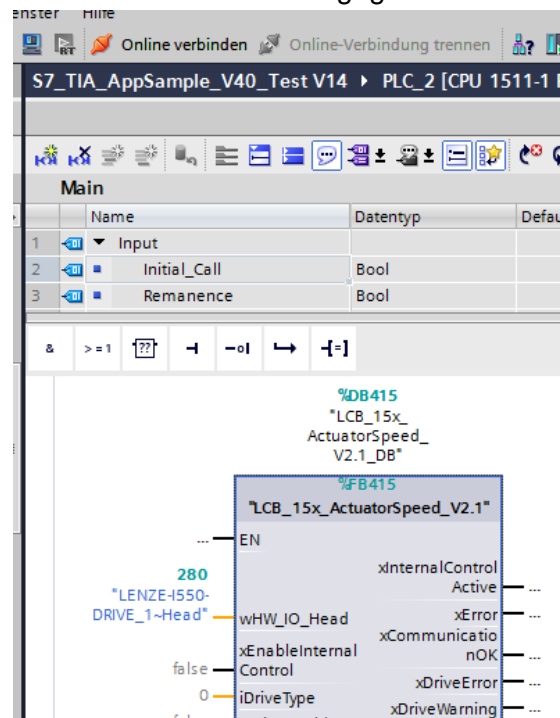
Die Grundsätzliche Konfiguration ist identisch mit dem bei einer Steuerung des Typs S7-300/400. Jedoch werden die Prozessdaten im Normalfall nicht mehr direkt über Adressen angesprochen sondern über die sogenannte „HW-Kennung“. Unabhängig davon müssen die I/O Adressen bei der gleichen Startadresse beginnen und ohne Lücken aufeinander folgen.



Alle Bausteine in den Lenze Bibliotheken für S7-1200 und S7-1500 arbeiten ab TIA-Portal V14 mit der „HEAD HW-Kennung“:



Am Baustein LCB_15x_ActuatorSpeed_V2.1 wird dann die HW_IO_Head (280) als Startadresse vorgegeben.



Da ein konsistenter Zugriff mit den Siemens Funktionen SFC14 und SFC15 über 3 Slots nicht möglich ist werden die 3 Slots im Lenze FB jeweils einzeln ausgewertet. Sollte dabei ein Fehler auftreten so wird er in den Diagnosemeldungen der CPU als Peripherie Zugriffsfehler eingetragen. Das Ausgangsbit „xCommunicationOK“ des Lenze FB ist bei dem Umrichter i550 ohne Funktion.

2 i550 am Profinet

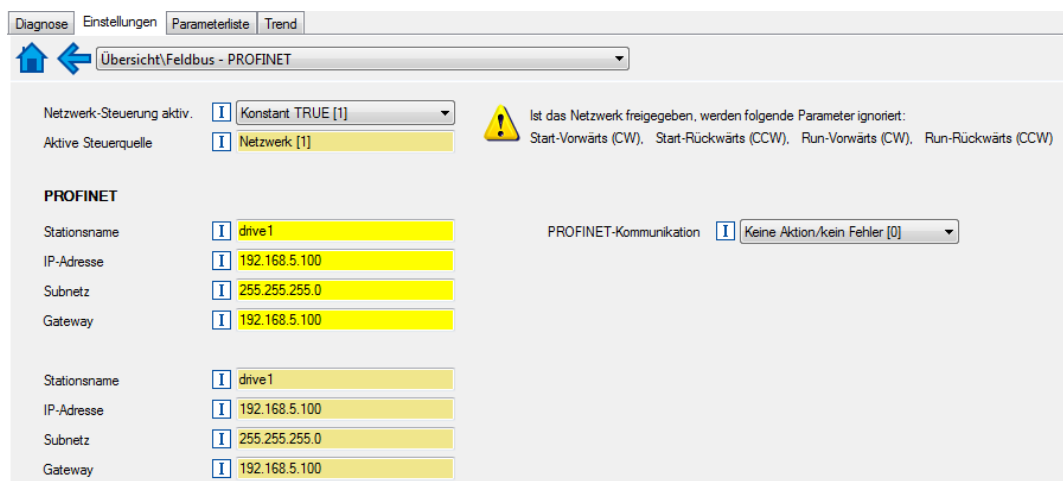
2.1 Parametrierung mit dem EASY Starter

Am i550 müssen für die grundsätzliche Ansteuerung über Profinet nur wenige Codestellen über das Keypad bzw. den EASY Starter eingestellt werden. Die grundsätzlichen Antriebsparameter wie Eckfrequenz oder Maximalstrom werden ebenso über den EASY Starter parametriert.

2.1.1 Einstellung der Profinetadresse und des Stationsnamens

Die Profinetadresse kann über das Keypad oder .

Die Adresse und der Stationsname kann über Menüpunkt „Übersicht\Feldbus-Profinet“ eingestellt werden (Index 2381:001, 2381:002, 2381:004).



Alle weiteren Einstellungen im Easystarter und die Parametrierung im Hardwaremanager der S7, bzw. des TIA-Portals sind identisch.

3 Parameterdatenkommunikation via DP-V1

Die Parameterdatenkommunikation funktioniert genauso wie bei 8400 und 9400. Sowohl über Profibus als auch Profinet. Beim i550 wurde jedoch die interne Parameterstruktur geändert. Es gibt keine Array - Parameter mehr, also eine Codestelle mit x-Subcodestellen des gleichen Typs (z.B.: Jog-Drehzahlen, Rampen). Der i550 besitzt zwar auch noch Subcodestellen aber diese wurden nicht als Array angelegt, daher muss jede Subcodestelle separat ausgelesen / geschrieben werden.

Folgende Funktionsbausteine aus der Lenze Bibliothek sind mit dem i550 verwendbar:

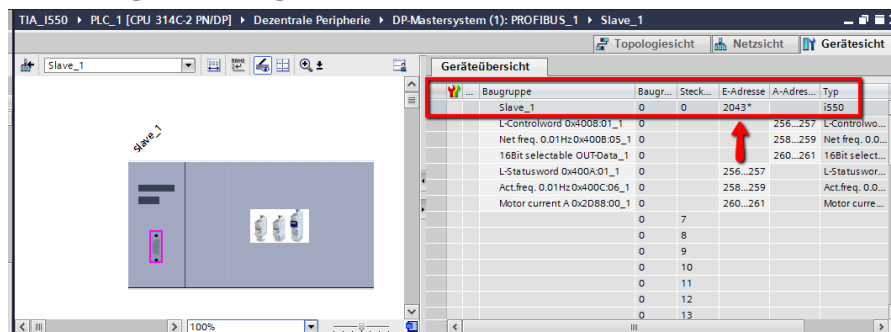
- ReadDriveParameter – Lesen eines einzelnen Parameters
- WriteDriveParameter – Schreiben eines einzelnen Parameters
- ReadDriveParamString – Lesen eines String - Parameters
- R_W_n_DrivePar – Lesen / Schreiben von bis zu 32 Parametern

Alle Bausteine für Arrays sind nicht verwendbar.

3.1 Beispiel lesen / schreiben eines Parameters mit einer S7-1500 CPU

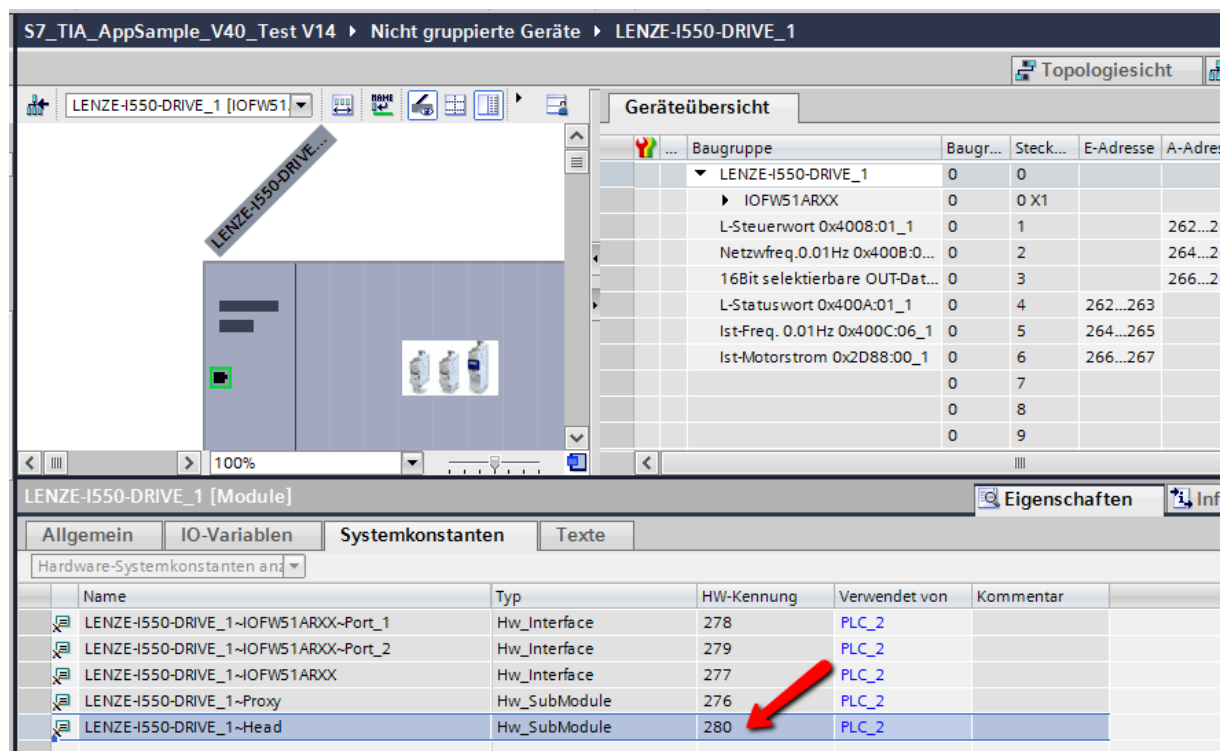
Dann wurden beim i550 die Codestellenbezeichnungen geändert, es gilt nicht mehr die alte Lenze Nummerierung. Es werden direkt die HEX Indizes angegeben, daher muss der Eingang „xUseIndexAsParam“ an den Bausteinen auf False gesetzt werden und der Index direkt im Hexformat übergeben werden.

3.1.1 Bestimmung der Diagnoseadresse bei einer S7-300



Diagnoseadresse des i550 Slave ist die 2043.

3.1.2 Bestimmung der HW-Kennung für den Parametertransfer für S7-1200 und S71500



HW-Kennung (Diagnoseadresse) des i550 Slave ist 280.

Im Beispiel wird der Index 0x2916 Subindex 0 (Maximalfrequenz) gelesen und geschrieben,

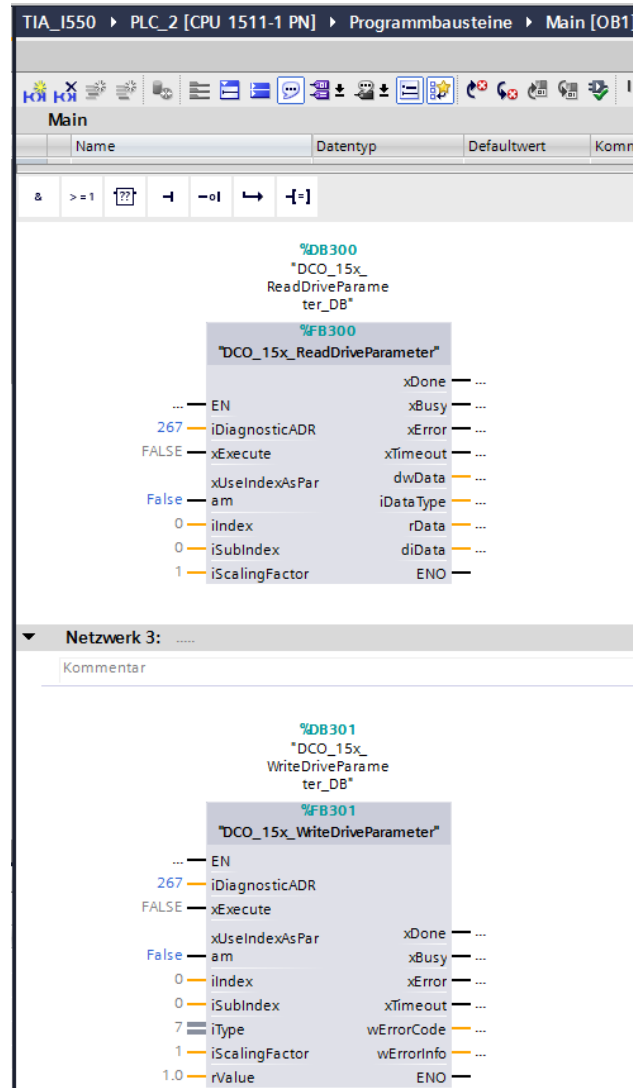
Datentyp (6 = Unsigned Integer) und Skalierungsfaktor (10) können der Parameter-Attributliste aus dem „Referenzhandbuch i550 Cabinet“ entnommen werden:



Anhang
Parameter-Attributliste

Adresse	Name / Subindex: Name	Voreinstellung	Einstellbereich	D / F / A / M
0x2916	Maximalfrequenz	50.0 Hz	0.0 ... 599.0 Hz	U16 / 10 / P / -

3.1.3 Einbinden der Bausteine im SPS Programm



Lesen des Parameters mit der Beobachtungstabelle:

TIA_I550 > PLC_2 [CPU 1511-1 PN] > Beobachtungs- und Forcetabellen > DCO_Read/WriteDriveParameter

	Name	Adre...	Anzeig...	Beobachtungswert	Steuwert
1	"DCO_15x_ReadDriveParameter_DB".iDiagnosticADR	%DB...	DEZ +/-	267	
2	"DCO_15x_ReadDriveParameter_DB".xExecute	%DB...	BO...	<input checked="" type="checkbox"/> TRUE	TRUE
3	"DCO_15x_ReadDriveParameter_DB".iIndex	%DB...	Hex	16#2916	16#2916
4	"DCO_15x_ReadDriveParameter_DB".iSubindex	%DB...	DEZ +/-	0	0
5	"DCO_15x_ReadDriveParameter_DB".xUseIndexAsParam	%DB...	BOOL	<input type="checkbox"/> FALSE	
6	"DCO_15x_ReadDriveParameter_DB".iScalingFactor	%DB...	DEZ +/-	10	10
7					
8	"DCO_15x_ReadDriveParameter_DB".xDone	%DB...	BOOL	<input checked="" type="checkbox"/> TRUE	
9	"DCO_15x_ReadDriveParameter_DB".xBusy	%DB...	BOOL	<input type="checkbox"/> FALSE	
10	"DCO_15x_ReadDriveParameter_DB".xError	%DB...	BOOL	<input type="checkbox"/> FALSE	
11	"DCO_15x_ReadDriveParameter_DB".dwData	%DB...	Hex	16#0000_01F4	
12	"DCO_15x_ReadDriveParameter_DB".iDataType	%DB...	DEZ +/-	6	
13	"DCO_15x_ReadDriveParameter_DB".dData	%DB...	Gleitpu...	50.0	
14	"DCO_15x_ReadDriveParameter_DB".diData	%DB...	DEZ +/-	500	

Settings | Diagnose | Parameterliste | Trend

← Übersicht Stellantrieb Drehzahl

Grundeinstellung

Netz-Bemessungsspannung Minimalsfrequenz Hz

Rotationsbeschränkung Maximalsfrequenz Hz

Schreiben des Parameters mit der Beobachtungstabelle:

17	"DCO_15x_WriteDriveParameter_DB".iDiagnosticADR	%DB...	DEZ +/-	267	
18	"DCO_15x_WriteDriveParameter_DB".xExecute	%DB...	BO...	<input checked="" type="checkbox"/> TRUE	TRUE
19	"DCO_15x_WriteDriveParameter_DB".xUseIndexAsParam	%DB...	BOOL	<input type="checkbox"/> FALSE	
20	"DCO_15x_WriteDriveParameter_DB".iIndex	%DB...	Hex	16#2916	16#2916
21	"DCO_15x_WriteDriveParameter_DB".iSubindex	%DB...	DEZ +/-	0	0
22	"DCO_15x_WriteDriveParameter_DB".iType	%DB...	DEZ +/-	6	6
23	"DCO_15x_WriteDriveParameter_DB".iScalingFactor	%DB...	DEZ +/-	10	10
24	"DCO_15x_WriteDriveParameter_DB".rValue	%DB...	Gleitpu...	60.0	60.0
25	"DCO_15x_WriteDriveParameter_DB".scWriteRecord.Paramet...	%DB...	Hex	16#0258_0000	
26					
27	"DCO_15x_WriteDriveParameter_DB".xDone	%DB...	BOOL	<input checked="" type="checkbox"/> TRUE	
28	"DCO_15x_WriteDriveParameter_DB".xBusy	%DB...	BOOL	<input type="checkbox"/> FALSE	
29	"DCO_15x_WriteDriveParameter_DB".xError	%DB...	BOOL	<input type="checkbox"/> FALSE	

Settings | Diagnose | Parameterliste | Trend

← Übersicht Stellantrieb Drehzahl

Grundeinstellung

Netz-Bemessungsspannung Minimalsfrequenz Hz

Rotationsbeschränkung **Maximalsfrequenz Hz**

Weiterführende Informationen zu den Funktionsbausteinen sind der Dokumentation der S7-Appsamples zu entnehmen.