



PDL

Process Device Library for TIA

Engineering Guidelines

V4.2



Inhaltsverzeichnis

Einleitung	1-5
1 Einführung	8
1.1	Was ist neu in V4.2?.....	8
1.2	-VB Skript Was ist neu in V4.1.2?.....	8
1.3	Was ist neu in V4.1.1?.....	8
1.4	Was ist neu in V4.1?.....	8
1.5	Zweck des Dokuments.....	10
1.6	Basisprojekt.....	11
1.6.1	Beispielprogramm „Devices V2“.....	12
1.6.2	Beispielprogramm „Devices V3“.....	12
1.6.3	Beispielprogramm „Devices V4.2“.....	13
1.6.4	Beispielprogramm Tank 1.....	14
1.7	Projektbibliothek.....	15
1.7.1	Typen.....	15
1.7.2	Kopiervorlagen.....	15
1.8	Baustein Hilfe Aufrufen.....	16
1.9	Systemübersicht.....	17
1.10	Bausteinumfang.....	18
1.10.1	PDL V02.00.....	18
1.10.2	PDL V03.00.....	19
1.10.3	PDL V04.00.....	19
1.10.4	PDL V04.10.....	20
1.10.5	PDL V04.20.....	20
1.11	Aufbau eines Device.....	21
1.12	Bibliothek Mechanismus Übersicht.....	23
2	Manuelles Anlegen eines neuen Devices am Beispiel eines MOT_FC.....	24
2.1	SPS Teil.....	24
2.2	HMI Teil.....	26
3	Programmstruktur.....	32
3.1	Programmstruktur allgemein.....	32
3.2	Programmierung.....	33
3.3	Objektorientierung / Statusübergabe.....	34
3.4	Weitere Programmierrichtlinien.....	35
4	Bibliotheksupdate.....	36
4.1	Bibliotheksupdate einzelnes Bibliothekselement (FACEP, TYPICAL, UDT).....	36
4.2	Bibliothek- / Projektupdate von ganzer Bibliothek von V1.0/V2.0/3.0 nach V4.2.....	39
5	System.....	50
5.1	Systemvoraussetzungen Hardware.....	50
5.2	Systemvoraussetzungen Software.....	50
5.3	Standard.....	50
5.4	Systemmerker.....	51

5.5	OB1 min. Zykluszeit für Regler	52
5.6	Hardwareaufbau	52
5.7	Uhrzeit / Zeitsynchronisation.....	52
5.8	Zeitsynchronisation.....	53
5.9	Optimierte Bausteine	53
5.10	GMP Einstellungen.....	53
5.11	Anlaufverhalten der Devices.....	55
5.12	Sm@rtserver Einstellungen	55
5.13	Panel Tauschen.....	55
5.14	CPU Tauschen	56
5.15	PLC Sim Advanced	56
5.16	PLC Meldungen Verriegelungen aktivieren.....	57
5.17	Technology Objekte - PID Regler Version	57
5.18	HMI Meldungen Runtime Einstellungen.....	57
5.19	Sprachumschaltung.....	58
5.19.1	Sprachumschaltung auf HMI.....	58
5.19.2	Einstellungen im TIA.....	58
5.19.3	Projekttexte projektieren.....	61
5.19.3.1	Projekttexte Exportieren	62
5.19.3.2	Projekttexte Importieren.....	62
6	HMI PDL Generator SiVArc.....	64
6.1	Einleitung.....	64
6.2	Arbeiten mit SiVArc	64
6.2.1	Kopiervorlage in Programm kopieren.....	64
6.2.2	Automatiklogik Programmierung	66
6.2.3	Texte für SiVArc eingeben.....	66
6.2.4	Übersetzten und Generieren.....	67
6.2.5	Symbole anordnen	67
7	HMI WinCC Classic V7.4.....	68
7.1	Einleitung.....	68
7.2	Anlegen des WinCC Projekts.....	68
7.2.1	Projekt entpackt.....	68
7.2.2	WinCC for PDL ins Projekt implementieren	68
7.2.3	Vorlagebild @PDLTypicals	69
7.2.4	Benutzerverwaltung.....	69
7.2.5	Neue Verbindung anlegen	69
7.3	Anlegen eines neuen Typicals	70
7.3.1	Variablen einlesen	70
7.3.2	Alarmer einlesen.....	71
7.3.3	TAG-Logging	71
7.3.4	Typical anlegen	72
7.4	Projekt duplizieren	72
7.5	Updaten.....	73
7.6	Zusatzinfos	74

1 Einführung

1.1 Was ist neu in V4.2?

- Multilingual / Sprachumschaltung

SPS Bausteine und Kommentare	Visualisierung (Faceplates, Typicals & Panels)	Programmiersprache
Deutsch English	Deutsch English Spanisch Französisch (Frankreich) Französisch (Schweiz) Niederländisch Italienisch	Deutsch English Spanisch Französisch (Frankreich) Französisch (Schweiz) Niederländisch Italienisch

- Die Sprache kann im Panel umgestellt werden
- Es können auch Multiinstanz Devices via SiVArc generiert werden. Dafür wurden die SiVArc Regeln aller Typicals und Faceplates angepasst.
- PID
 - Anzeige Fehler von Stellwert Y bei LOCK und TRACK korrigiert
- Neues Device MOT_2S «Motor mit 2 Geschwindigkeiten
- Bug Fix am Device «LS».
- Baustein @Version interniert. In diesem ist die aktuelle PDL Version abgelegt und wird automatisch auf dem Panel unter «Info» angezeigt. Damit man einfach erkennen kann welche PDL Version verbaut ist.

1.2 Was ist neu in V4.1.2?

- -VB Skript in allen WinCC Faceplates mit analogen Eingabefeldern korrigiert. Die VB Abfrage auf «1» oder «true» funktioniert ab WinCC 7.4 SP1 Update 3 und höher nicht zuverlässig. Dies hatte zur Folge, dass das Userlogin nicht mehr funktioniert hat und wurde korrigiert.
 - Da alle Tests mit WinCC 7.4 SP1 Update 9 durchgeführt wurden, muss dieses Update oder ein höher verwendet werden.

1.3 Was ist neu in V4.1.1?

- ANA_ALARM_4LIM
 - Übernahme Problem der INOUT Parameter ALA_LL und LIM_LL behoben
- ANA_ALARM
 - Übernahme Problem der INOUT Parameter ALA_LL behoben

1.4 Was ist neu in V4.1?

- Neue Bausteine
 - DRV_BI - Treiber für Digitales Eingangsbyte mit Diagnose für Namurkarten
 - DRV_DI - Treiber für Digitales Eingangsbit mit Diagnose für Namurkarten
 - UNIT – Unitbaustein ohne Steuerungsmöglichkeit
- Der Status «Quittierter Alarm anstehend» kann am Unit Baustein ausgelesen werden. Dafür wurden folgende Bausteine angepasst:
 - VAL, VAL_MOT, MOT_1D ; MOT_2D, MOT_FC, CTRL_VOL, DOS, LS; DIG_WARN, DIG_ALARM

- UNIT Typical zeigt kommenden Alarm mit «blinken» an.
- Der Status QAUTO wird im Lokalmodus rückgesetzt. Dies wurde beifolgenden Devices angepasst.
 - VAL, VAL_ANA, VAL_MOT, MOT_1D, MOT_2D, MOT_FC, LS
- MOT_2D blinket während der Umschaltzeit.
- Beim ALA_ALARM_4LIM wurde der Fehler des MODE_L_H behoben. In WinCC können die nicht verwendeten Alarme oder Warnungen können mit @ ausgeblendet werden
- Der Updateleitfaden wurde angepasst.
- Globale Bibliothek für die PDL V4.1 um Bibliothekupdate zu vereinfachen.
- Die Steuerungen 1512-SP (Bauform von ET200SP) und 1515-SP (PC mit CPU und Runtime Advanced) wurden in die Library integriert.

1.5 Zweck des Dokuments

Dieses Dokument beschreibt ein Regelwerk zur Erstellung von Software-Programmen mit Simatic TIA (ab V14 SP1), mit der Actemium Process Device Library (PDL) für TIA. Zudem wird der Mechanismus der Bibliothek erklärt.

Mit der Version V4.2 wurde eine Sprachumschaltung realisiert. Bitte lesen Sie Kapitel 5.19 bevor sie starten.

Es wird das Arbeiten mit dem PDL Generator via SiV Arc erklärt.

Es werden Vorgaben bezüglich Programmstruktur, Bezeichnungen und Programmaufbau gemacht.

Das Arbeiten mit WinCC V7.x (Ab Version V7.4 SP1 Update 9).

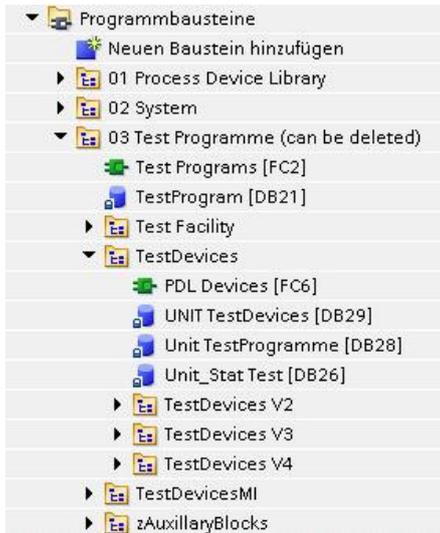
Die strikte Einhaltung dieser Regeln verbessert den Prozess der Software-Programmierung mit folgenden Zielen:

- ✓ Vereinheitlichung des Quell-Code-Stils und dessen Darstellung.
- ✓ Vereinfachter Einstieg für neu ins Projektteam hinzukommende Programmierer, die sich mit vorhandenem Code vertraut machen müssen. Die Richtlinien geben ihnen klare Regeln für die Modifizierung und Neuerstellung von Programmen.
- ✓ Eliminierung von Schwachstellen einer Programmiersprache.
- ✓ Zeitoptimierung bei der Entwicklung.
- ✓ Die Wartbarkeit von Programmen sicherstellen.
- ✓ Die Wiederverwendbarkeit von Programmen ermöglichen.

1.6 Basisprojekt

Als Basis soll das Projekt „PDL_V4.2“ verwendet werden. Hierin ist die Grundinstallation mit einer CPU S7-1511 und einem TP700 Panel vorhanden. Falls die Geräte nicht passen, müssen dies in der Gerätekonfiguration getauscht oder können aus der Bibliothek unter Kopiervorlagen eingefügt werden.

In diesem Basisprojekt sind auch zwei Beispiel Programme programmiert.



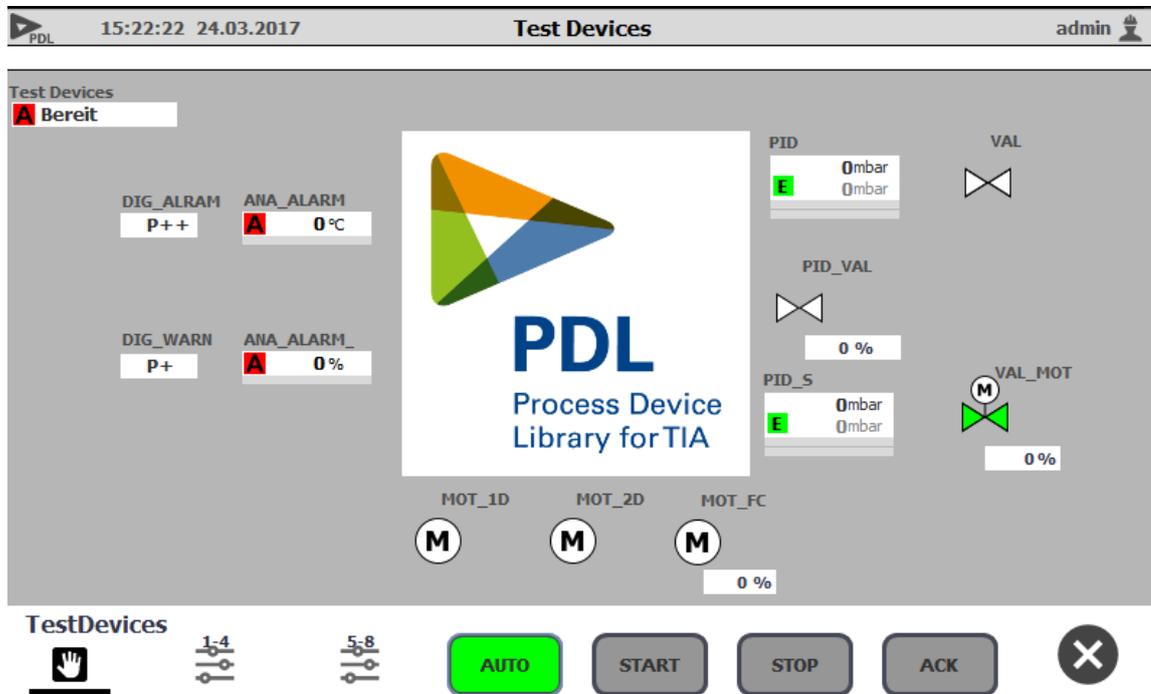
Die Programme «Test Devices V2», «Test Devices V3» und «Test Devices V4» zeigen die Basis Verschaltung jedes Devices. Das Programm «Tanklager Ost» zeigt eine einfache Demoapplikation, in der die Funktionen der PDL gezeigt werden.

Diese Beispielprogramme können gelöscht werden.

Unter TestDevicesMI sind dieselben Programmbeispiele mit Multiinstanzen realisiert.

1.6.1 Beispielprogramm „Devices V2“

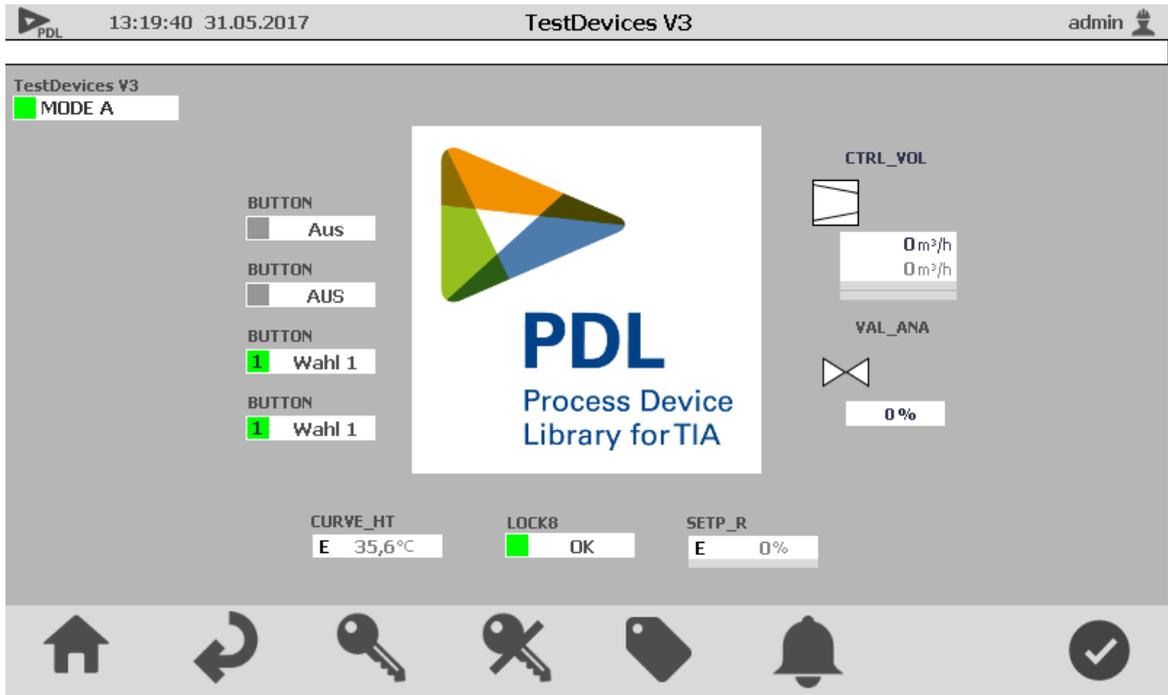
Es gibt ein Bild mit Namen „Devices“ in dem alle Devices mit dem Test Programm verbunden sind. Es handelt sich um die Bausteine welche bereits in V2.0 enthalten waren.



Die Handbedienung kann so ausprobiert werden.

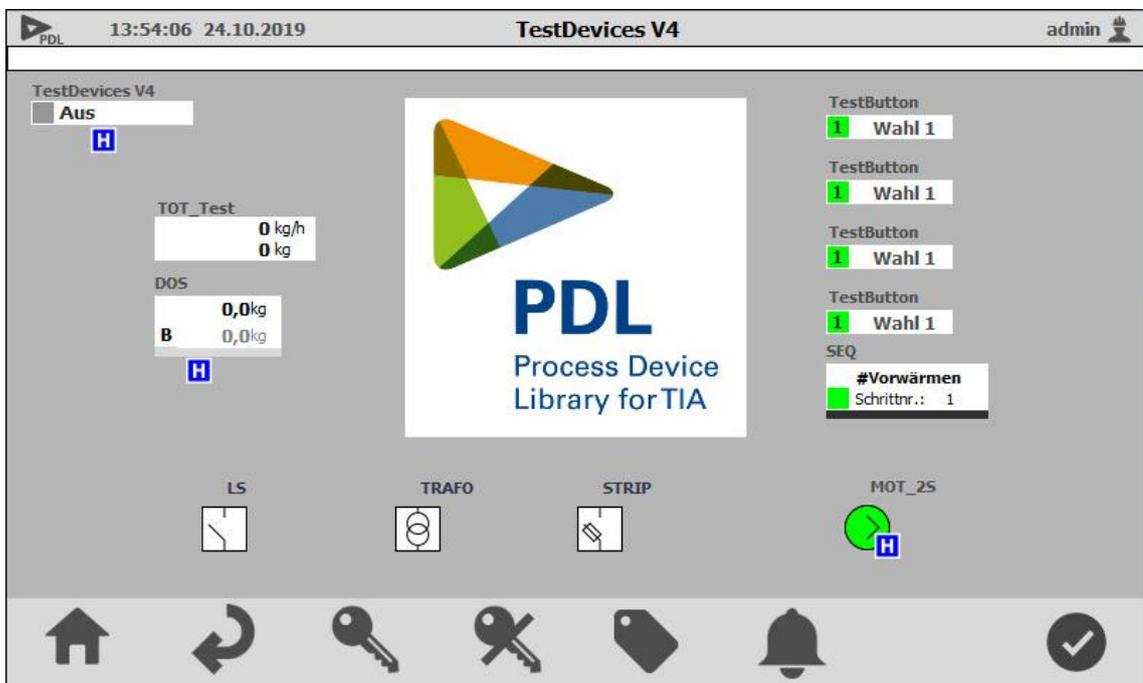
1.6.2 Beispielprogramm „Devices V3“

Es gibt ein Bild mit Namen „Devices“ in dem alle Devices mit dem Test Programm verbunden sind. Es handelt sich um die Bausteine welche neu in V3.0 enthalten sind.



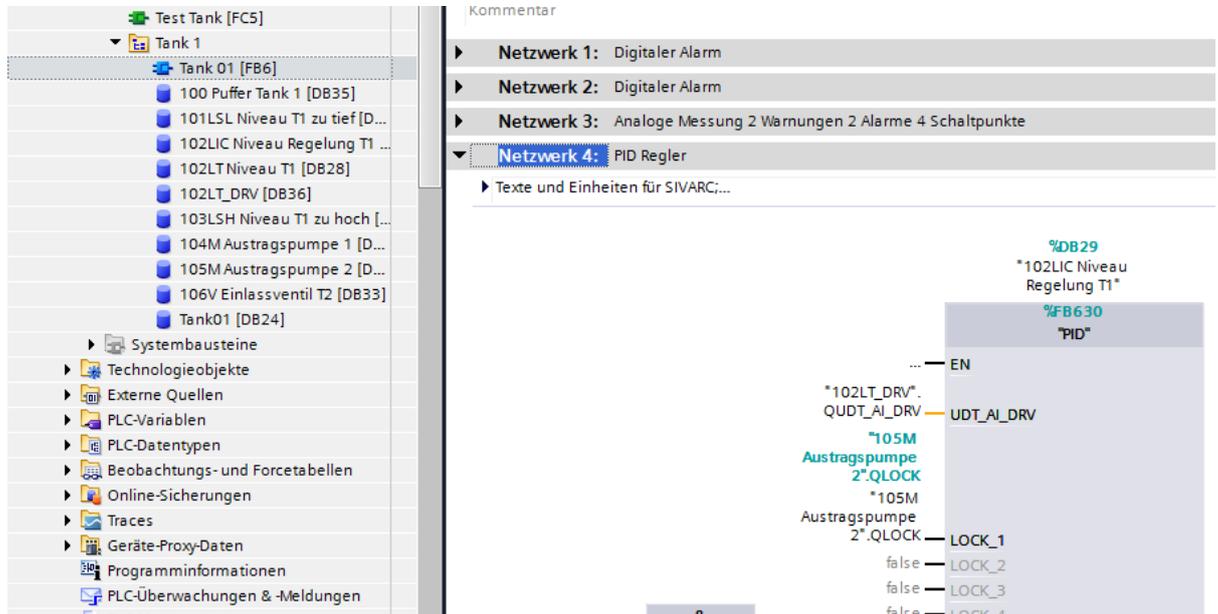
1.6.3 Beispielprogramm „Devices V4.2“

Es gibt ein Bild mit Namen „Devices“ in dem alle Devices mit dem Test Programm verbunden sind. Es handelt sich um die Bausteine welche neu in V4.0 enthalten sind.



1.6.4 Beispielprogramm Tank 1

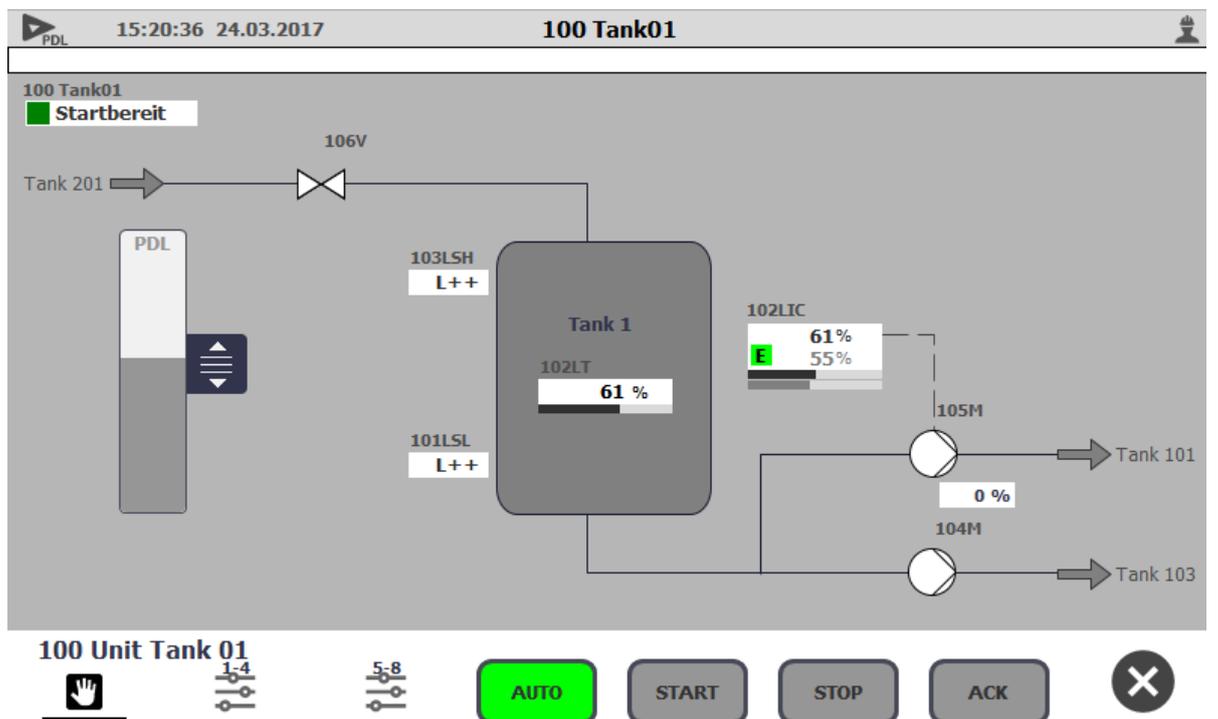
Es wurde ein Demo Programm entwickelt, welches die Mechanismen und die Programmierung der PDL erklärt.



The screenshot shows the project tree on the left with 'Tank 1' expanded. On the right, the 'Netzwerk 4: PID Regler' is selected, showing the variable declaration for the PID controller:

```

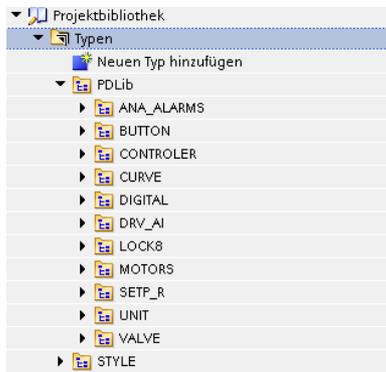
%DB29
*102LIC Niveau
Regelung T1*
%FB630
"PID"
... EN
*102LT_DRV*.
QUDT_AI_DRV  UDT_AI_DRV
*105M
Austragspumpe
2*.QLOCK
*105M
Austragspumpe
2*.QLOCK  LOCK_1
false  LOCK_2
false  LOCK_3
false  LOCK_4
    
```



Mit dem Unit Baustein kann die Anlage eingeschaltet werden. Mit dem Auto Button können alle Devices auf Auto geschaltet werden. Das Niveau des Tanks kann mit dem «Schieber» simuliert werden.

1.7 Projektbibliothek

1.7.1 Typen



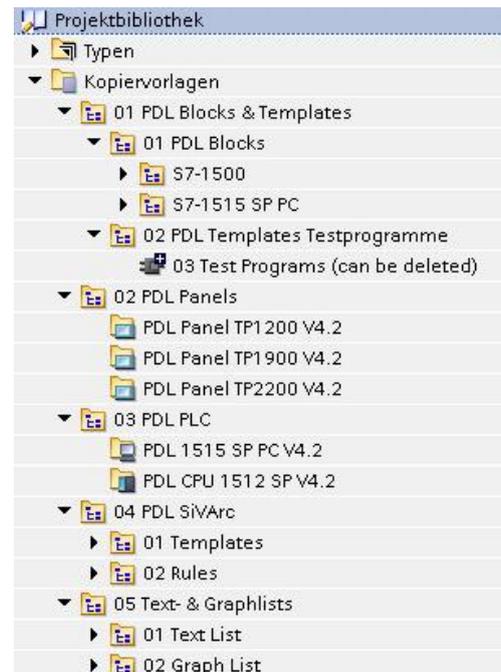
Hier sind alle UDT's, Faceplates und Typicals abgelegt. Zudem die Styles der PDL.

1.7.2 Kopiervorlagen

Hier sind die FB's der PDL für S7-1500 und S7 1515SP (PC Basiert) abgelegt. Zudem auch Programmvorlagen, Panels mit dem PDL Framework in allen Comfortpanel Grössen von 7 bis 22 Zoll. Aus dem TP700 kann per «Gerät tauschen» ein TP900 gemacht werden da es die gleiche Auflösung hat. Dasselbe gilt für TP1200 nach TP1500.

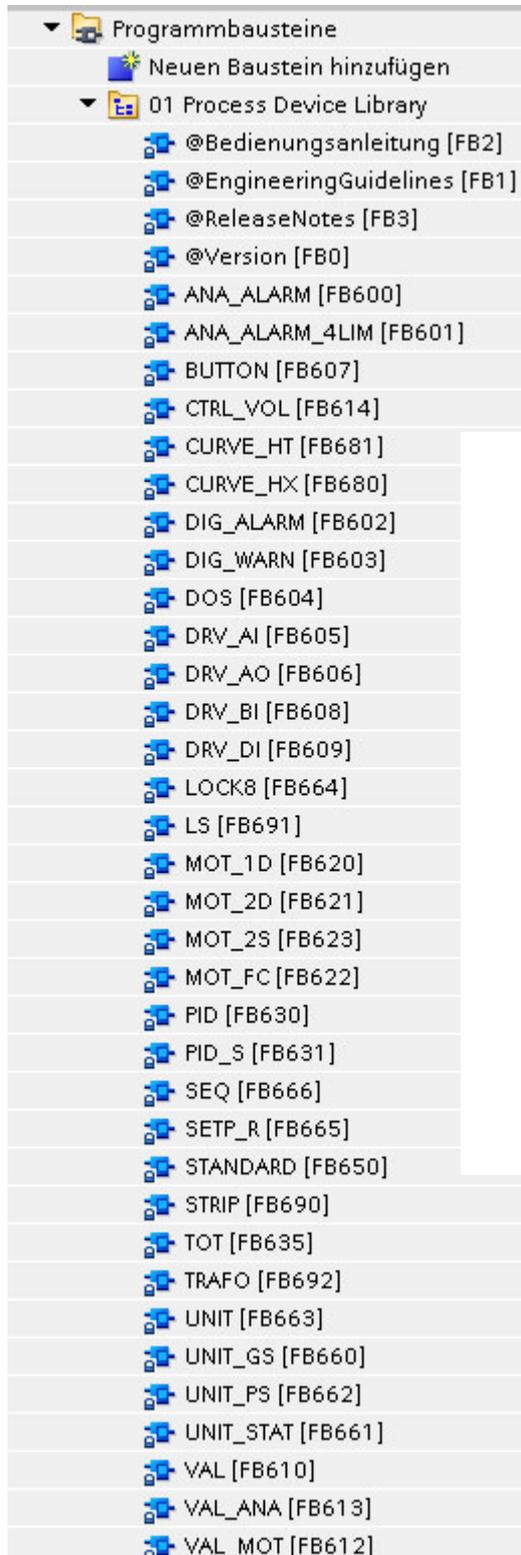
Es sind auch PDL Vorlagen für die S7-1511/12 SP und die CP 1515 SP (PC basiert) abgelegt. Bitte immer die Steuerungen und Panel aus dem Demoprojekt oder der Bibliothek verwenden, da diese für die PDL konfiguriert sind.

Es sind auch die SiVArc Templates und Regeln enthalten, welche für das generieren der Visualisierung gebraucht wird.



1.8 Baustein Hilfe Aufrufen

Die Bausteinhilfe kann mit SHIFT & F1 beim Anklicken eines Bausteins aufgerufen werden.



PDL
Process Device
Library for TIA

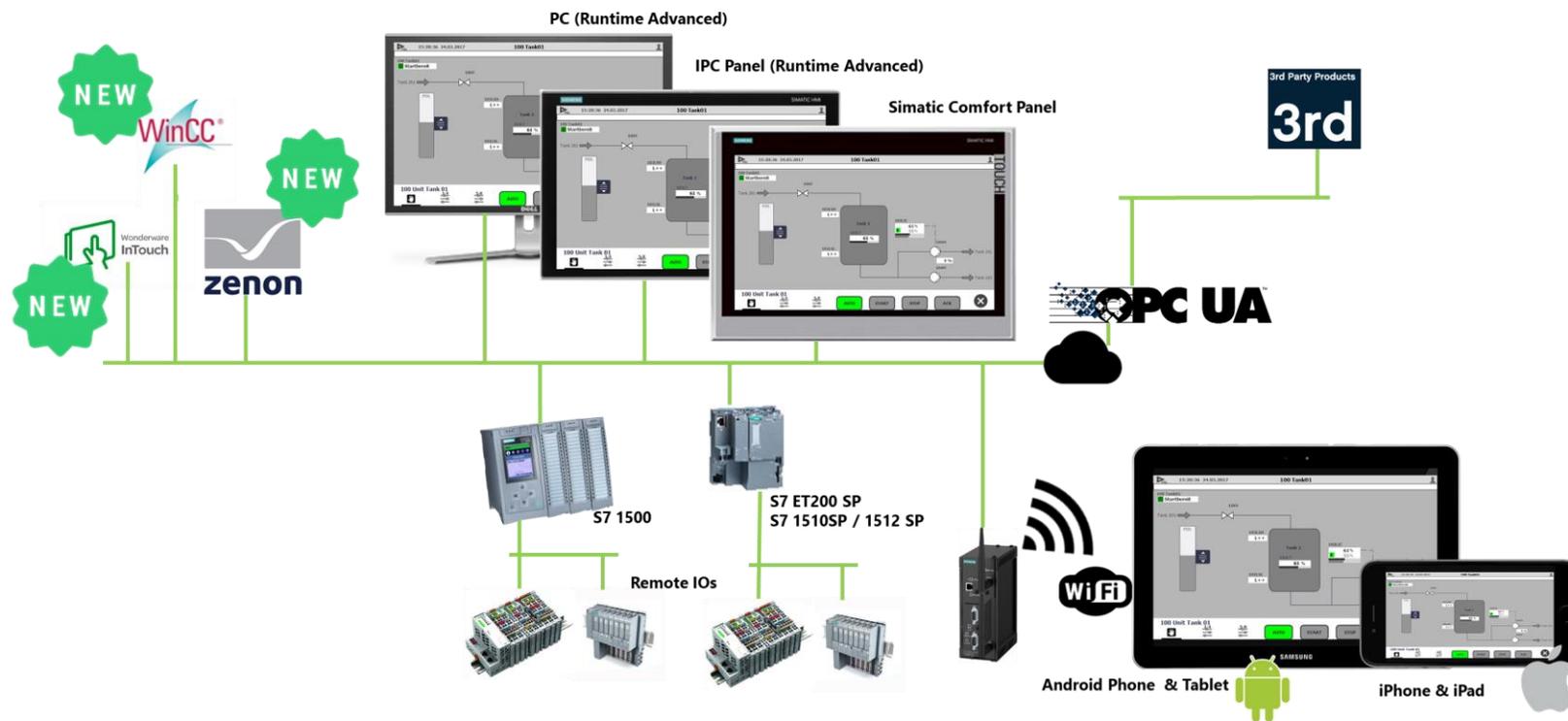
ANA_ALARM_4LIM
Analoger Alarm mit Grenzwerten

V3.1



1.9 Systemübersicht

Die PDL Bibliothek unterstützt die Simatic S7 15XX CPU's und die CPU für die ET200 SP (15XXSP). Neu wird auch WinCC 7.4 SP1 < Update 9 unterstützt. Es werden auch InTouch und Zenon unterstützt.



1.10 Bausteinumfang

1.10.1 PDL V02.00

Function Block Name	Typical Name	Faceplate Name	UDT Name	Name
ANA_ALARM	TYPICAL_ANA_ALARM	FACEP_ANA_ALARM	UDT_ANA_ALARM	4 Analogue Alarms
ANA_ALARM_4LIM	TYPICAL_ANA_ALARM_4LIM	FACEP_ANA_ALARM_4LIM	UDT_ANA_ALARM_4LIM	4 Analogue Alarms 4 Limits
DIG_ALARM	TYPICAL_DIG_ALARM	FACEP_DIG_ALARM	UDT_DIG_ALARM	1 Digital Alarm
DIG_WARN	TYPICAL_DIG_WARN	FACEP_DIG_ALARM	UDT_DIG_ALARM	1 Digital Warning
DRV_AI	-	-	UDT_DRV_AI	Analogue Input Driver
MOT_1D	TYPICAL_MOT_1D	FACEP_MOT_1D	UDT_MOT_1D	Motor 1 Direction
MOT_2D	TYPICAL_MOT_2D	FACEP_MOT_2D	UDT_MOT_2D	Motor 2 Directions
MOT_FC	TYPICAL_MOT_FC	FACEP_MOT_FC	UDT_MOT_FC	Motor Frequency Controlled
PID	TYPICAL_PID TYPICAL_PID_VAL	FACEP_PID	UDT_PID	PID Controller
PID_S	-	FACEP_PID_S	UDT_PID_S	STEP PID Controller
STANDARD	-	-	UDT_STANDARD	System Standard
UNIT_GS	TYPICAL_UNIT OVERVIEW_UNIT	FACEP_UNIT	UDT_UNIT	Unit Controller
VAL	TYPICAL_VAL	FACEP_VAL	UDT_VAL	Valve with FB open / closed
VAL_MOT	TYPICAL_VAL_MOT	FACEP_VAL_MOT	UDT_VAL_MOT	Motor valve

1.10.2 PDL V03.00

Function Block Name	Typical Name	Faceplate Name	UDT Name	Name
BUTTON	TYPICAL_BUTTON	TYPICAL_BUTTON	UDT_BUTTON	Button, Switch, 1 of 3 Selection, 1 of 4 Selection
CTRL_VOL	TYPICAL_CTRL_VOL	FACEP_CTRL_VOL	UDT_CTRL_VOL	Volume Flow Controller
CURVE_HT	TYPICAL_CURVE_HT	FACEP_CURVE_HT	UDT_CURVE_HT	Curve HT
CURVE_HX	-	-	-	Curve HX
DRV_AO	-	-	-	Analogue Output Driver
LOCK8	TYPICAL_LOCK8	FACEP_LOCK8	UDT_LOCK8	Interlock 8
SETP_R	TYPICAL_SETP_R	FACEP_SETP_R	UDT_SETP_R	Setpoint Real
UNIT_PS	TYPICAL_UNIT OVERVIEW_UNIT	FACEP_UNIT	UDT_UNIT	Unit Parameter Switch
VAL_ANA	TYPICAL_VAL_ANA	FACEP_VAL_ANA	UDT_VAL_ANA	Analogue Valve

1.10.3 PDL V04.00

Function Block Name	Typical Name	Faceplate Name	UDT Name	Name
TOT	TYPICAL_TOT	TYPICAL_TOT	UDT_TOT	Totalizer, Heat Meter, Operation Hours
SEQ	TYPICAL_SEQ	FACEP_SEQ	UDT_SEQ	Sequence
DOS	TYPICAL_DOS	FACEP_DOS	UDT_DOS	Dosage
TRAFO	TRAFO	-	UDT_ALARM	Trafo
LS	LS	FACEP_LS	UDT_ALARM	Power Switch
STRIP	TYPICAL_STRIP	-	UDT_ALARM	Fuse Rails

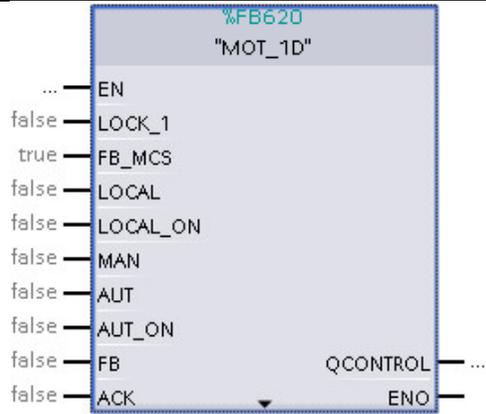
1.10.4 PDL V04.10

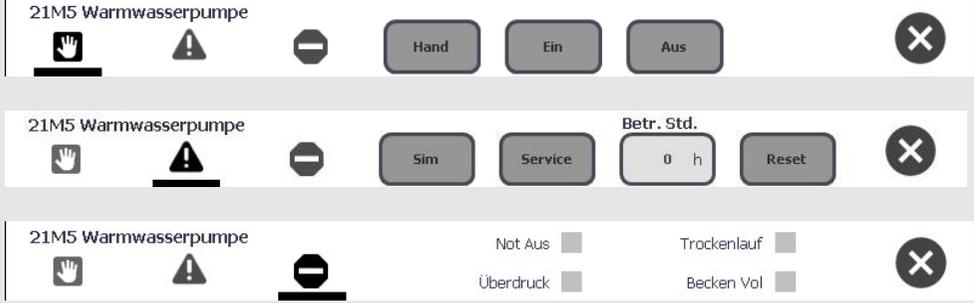
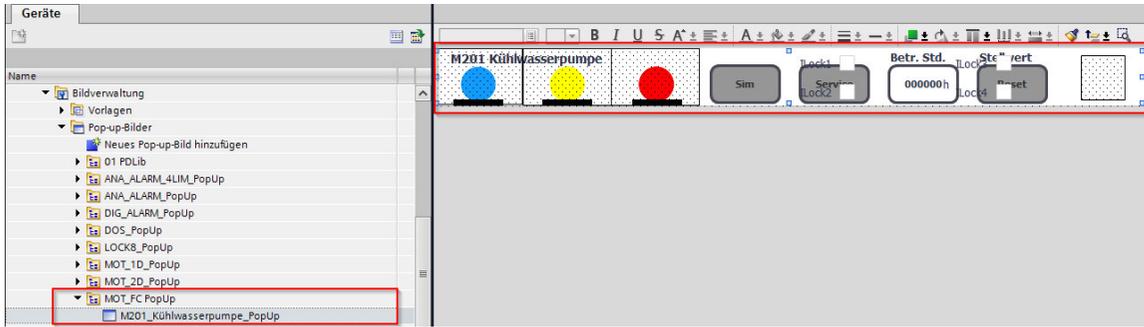
Function Block Name	Typical Name	Faceplate Name	UDT Name	Name
DRV_BI	-	-	-	Digital Input Driver (Byte)
DRV_BI	-	-	-	Digital Input Driver
UNIT	TYPICAL_UNIT_BASIC		UDT_UNIT	Unit

1.10.5 PDL V04.20

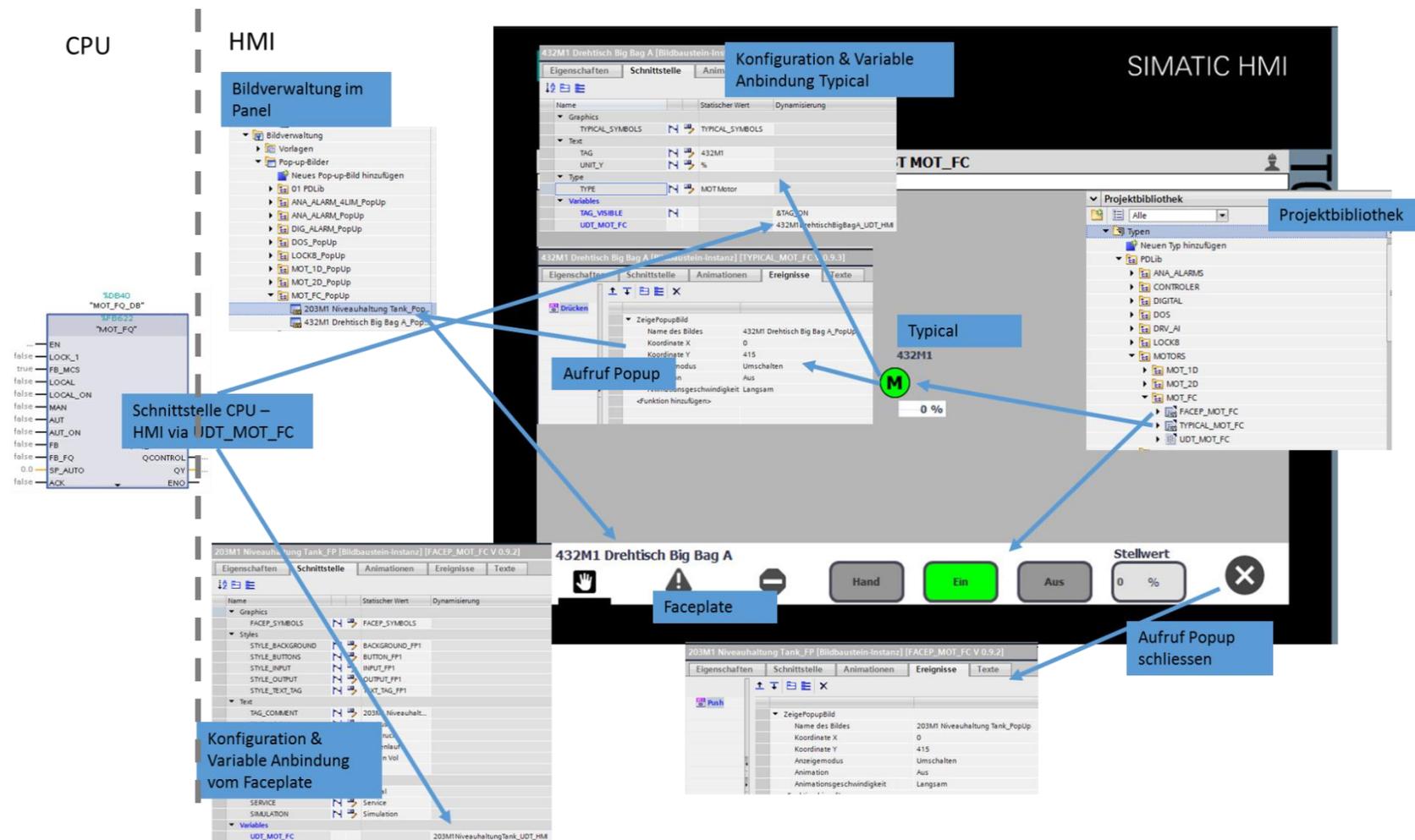
Function Block Name	Typical Name	Faceplate Name	UDT Name	Name
MOT_2S	TYPICAL_MOT_2S	FACEP_MOT_2S	UDT_MOT_2S	Motor 2 Speed

1.11 Aufbau eines Device

Typ	Objekt																				
<p>Funktionsbaustein</p> <p>Dieser Baustein enthält die Logik des jeweiligen Devices. Pro Device gibt es eine UDT Struktur welche mit dem HMI kommuniziert. Die ganze Kommunikation läuft über diese UDT.</p>																					
<p>UDT</p> <p>Die UDT ist eine Struktur von Variablen, welche mit dem HMI kommunizieren. Als erster ist immer der Status mit für die Statusanzeige auf dem HMI und die Steuerbefehle vom HMI. Anschließend sind die andere Variablen, welche angezeigt werden wie Betriebsstunden usw.</p>	<table border="1"> <thead> <tr> <th colspan="5">UDT_MOT_1D</th> </tr> <tr> <th></th> <th>Name</th> <th>Datentyp</th> <th>Defaultwert</th> <th>Ein</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Status</td> <td>DWord</td> <td>16#0</td> <td></td> </tr> <tr> <td>2</td> <td>OP_HRS</td> <td>Dint</td> <td>0</td> <td></td> </tr> </tbody> </table>	UDT_MOT_1D						Name	Datentyp	Defaultwert	Ein	1	Status	DWord	16#0		2	OP_HRS	Dint	0	
UDT_MOT_1D																					
	Name	Datentyp	Defaultwert	Ein																	
1	Status	DWord	16#0																		
2	OP_HRS	Dint	0																		
<p>Typical</p> <p>Das Symbol zur Visualisierung auf dem HMI</p>																					

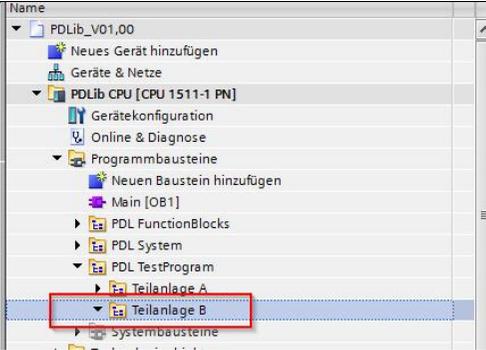
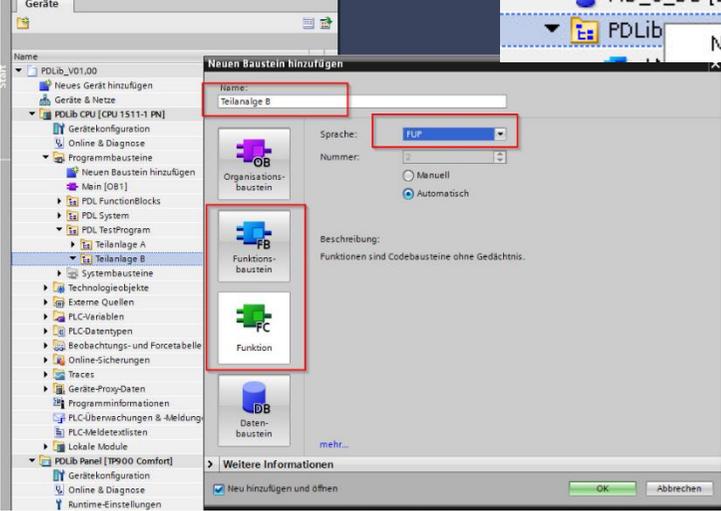
<p>Facepalte</p> <p>Bim Berühren des Typicals öffnet sich das dazugehörige PopUp auf welchem das Faceplate liegt. Dieses dient zum Steuern des Devices. Es besteht aus 1-3 Registerkarten.</p>	
<p>PopUp</p> <p>Das Faceplate wird auf das PopUp gelegt, damit es bei einem „Touch“ auf das jeweilige Device erscheint.</p>	

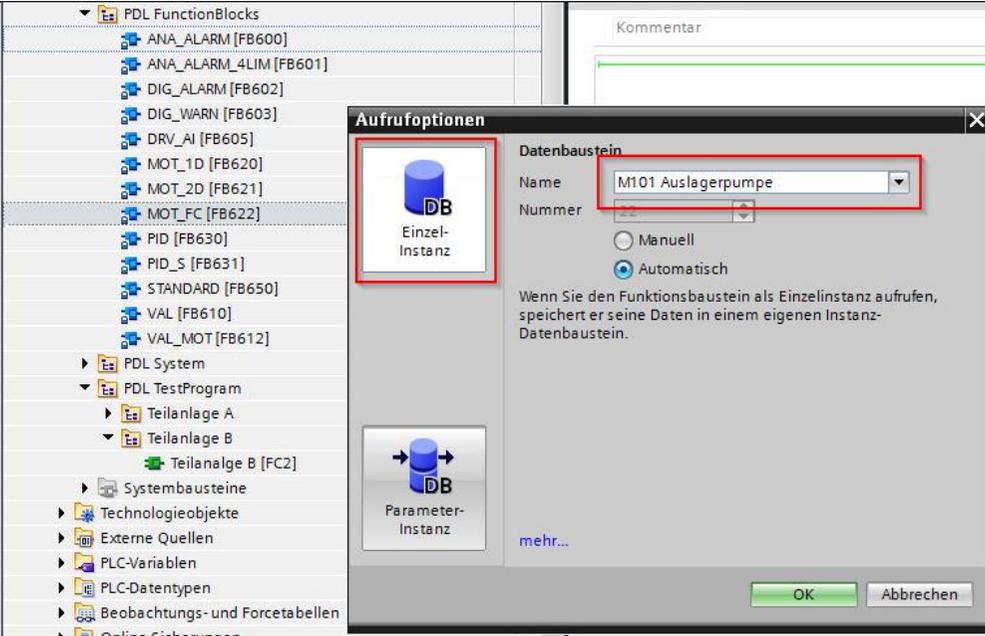
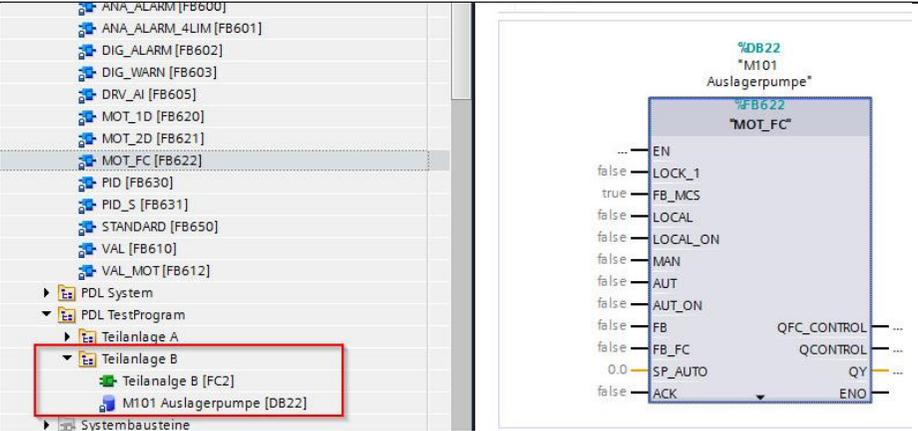
1.12 Bibliothek Mechanismus Übersicht

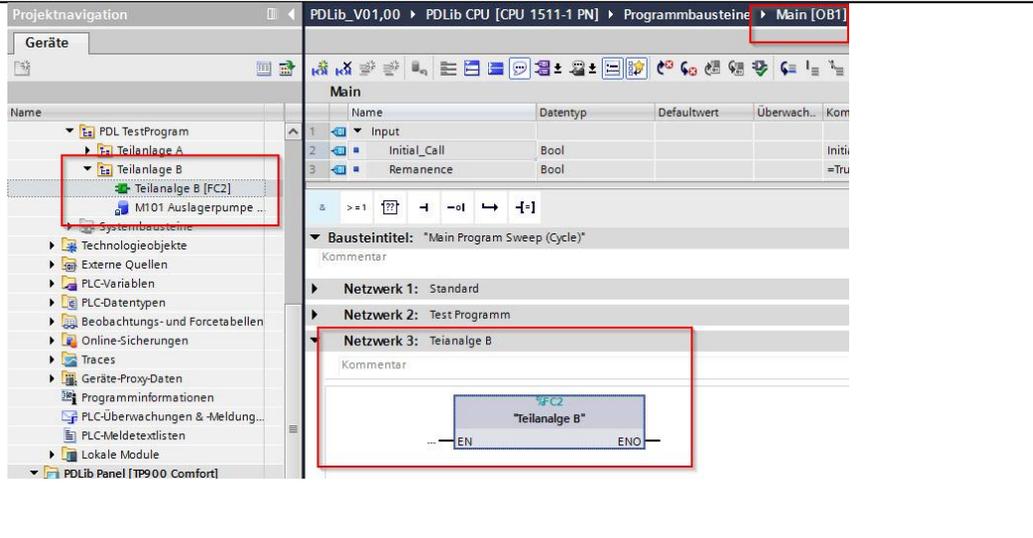


2 Manuelles Anlegen eines neuen Devices am Beispiel eines MOT_FC

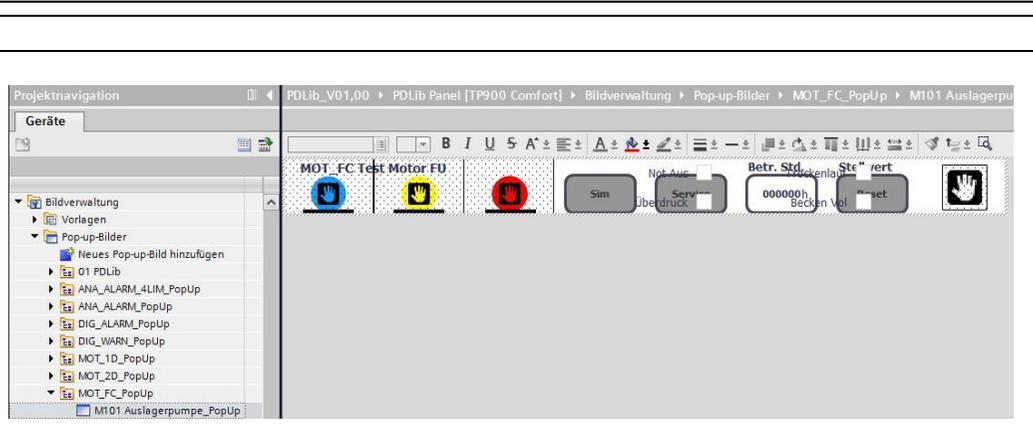
2.1 SPS Teil

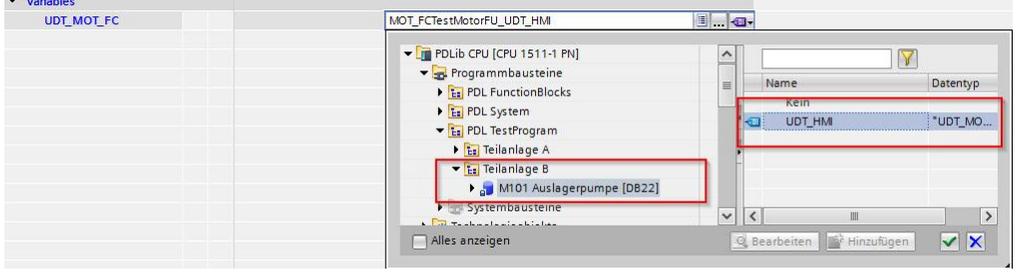
	Typ	Objekt
1	<p>Einen Hierarchieordner anlegen für die Teilanlage in unserem Beispiel „Teilanlage B“.</p> <p>Mit rechts Klick auf einen bereits vorhandenen Hierarchieordner dann auf „<i>Neue Gruppe hinzufügen</i>“, wird ein neuer Hierarchieordner erstellt.</p>	
2	<p>Einen FB oder FC erzeugen in welcher die „Teilanlage B“ programmiert wird. Die Programmiersprache kann frei gewählt werden, jedoch wird FUP empfohlen.</p> <p>Mit Doppel Klick auf „<i>Neuen Baustein hinzufügen</i>“ unter Programmbausteine, erscheint das Auswahlfenster und es können die verschiedenen Bausteine erstellt werden.</p>	

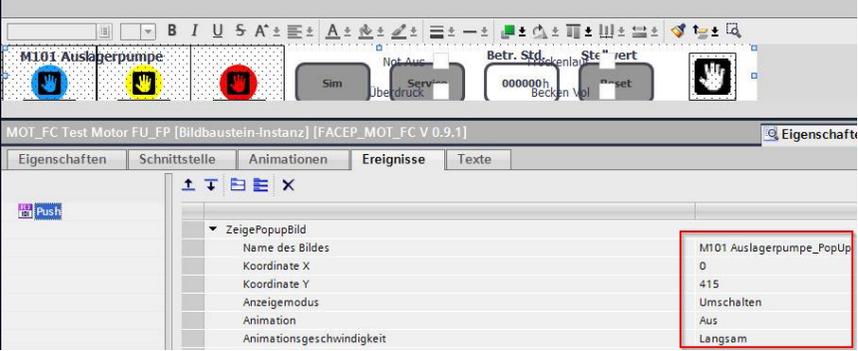
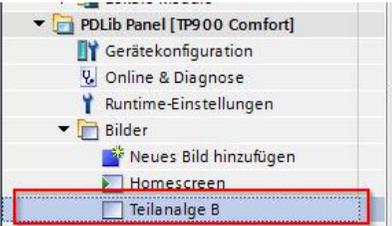
3	Typ	Objekt
	<p>Vom Ordner „PDL FunctionBlocks“ den MOT_FC per Drag & Drop in den FC Teilanlage B ziehen.</p> <p>Den Baustein mit <Tagname + Leerzeichen und Kommentar> bezeichnen. Aus diesem Namen werden die Alarmer des Devices abgeleitet. In unserem Beispiel „M101 Auslagerpumpe“</p>	
4	<p>Alle Funktionen (FC) und Instanzbausteine (DB) werden im Ordner der Teilanlage abgelegt. Bei größeren macht es Sinn die die Bausteine noch hierarchisch zu strukturieren.</p>	

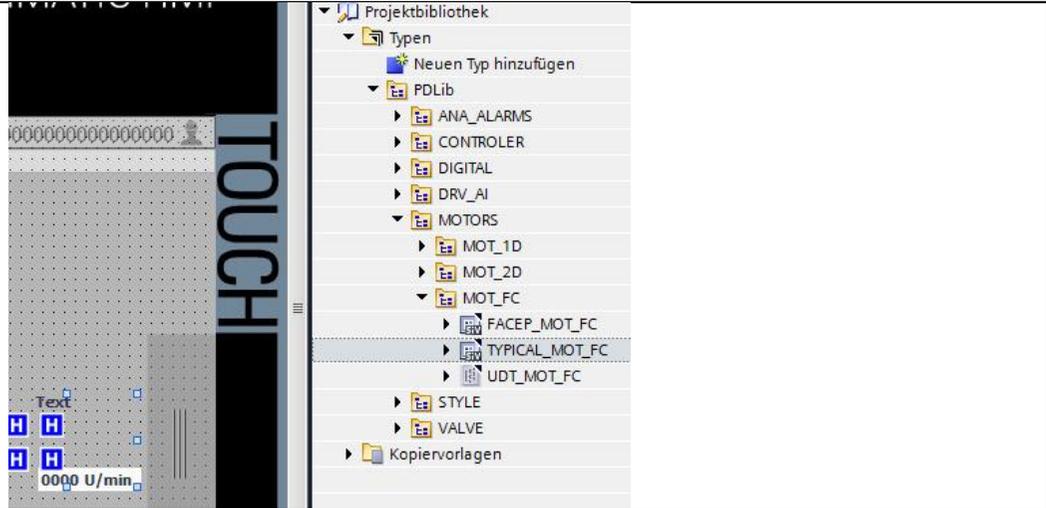
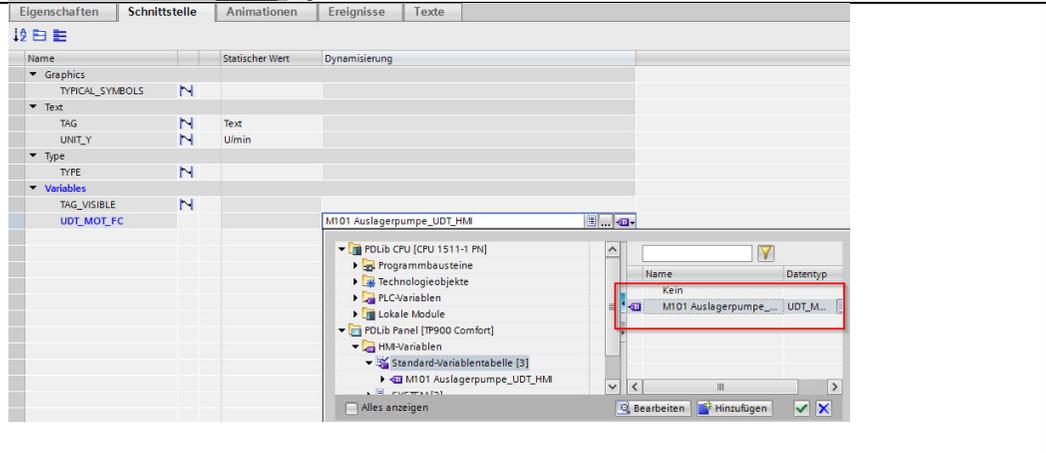
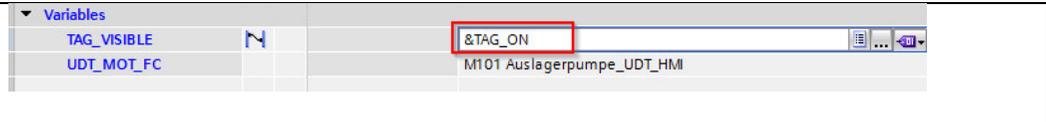
Typ	Objekt
<p>5</p> <p>Den FC Teilanlage im OB1 aufrufen.</p>	

2.2 HMI Teil

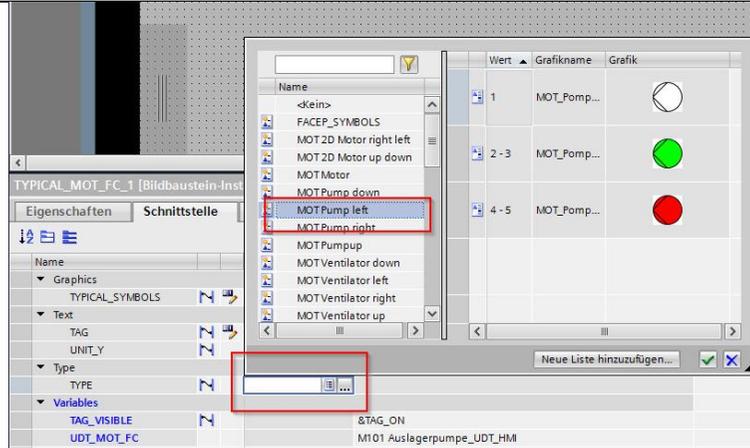
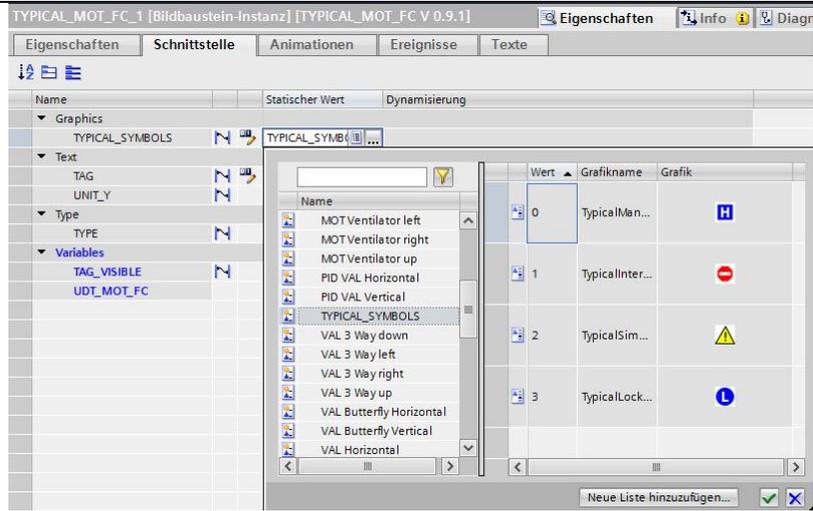
Typ	Objekt
<p>1.</p> <p>Unter Bildverwaltung / Pop-up-Bilder / MOT_FC_PopUp das „MOT_FC Test Motor FU_PopUp“ kopieren und umbenennen zu „M101 Auslagerpumpe_PopUp“.</p>	

Typ	Objekt																																												
<p>2. Doppelklick auf das Facepalte dann unter Schnittstelle \ Variables  die UDT vom Instanzbaustein M101 Auslagerpumpe auswählen.</p>																																													
<p>3. Unter Schnittstelle alle Attribute wie Tagname usw. Eintragen. Weitere Details zur Konfiguration findet man in der Hilfe zum Baustein.</p>	<table border="1"> <thead> <tr> <th>Attribute</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td>Graphics</td> <td></td> </tr> <tr> <td>FACEP_SYMBOLS</td> <td>FACEP_SYMBOLS</td> </tr> <tr> <td>Styles</td> <td></td> </tr> <tr> <td>STYLE_BACKGROUND</td> <td>BACKGROUND_FP1</td> </tr> <tr> <td>STYLE_BUTTONS</td> <td>BUTTON_FP1</td> </tr> <tr> <td>STYLE_INPUT</td> <td>INPUT_FP1</td> </tr> <tr> <td>STYLE_OUTPUT</td> <td>OUTPUT_FP1</td> </tr> <tr> <td>STYLE_TEXT_TAG</td> <td>TEXT_TAG_FP1</td> </tr> <tr> <td>Text</td> <td></td> </tr> <tr> <td>TAG_COMMENT</td> <td>MOT_FC Test Motor FU</td> </tr> <tr> <td>TEXT_ILOCK1</td> <td>Not Aus</td> </tr> <tr> <td>TEXT_ILOCK2</td> <td>Überdruck</td> </tr> <tr> <td>TEXT_ILOCK3</td> <td>Trockenlauf</td> </tr> <tr> <td>TEXT_ILOCK4</td> <td>Becken Vol</td> </tr> <tr> <td>UNIT_Y</td> <td>%</td> </tr> <tr> <td>UserRights</td> <td></td> </tr> <tr> <td>MANUAL</td> <td>Manual</td> </tr> <tr> <td>SERVICE</td> <td>Service</td> </tr> <tr> <td>SIMULATION</td> <td>Simulation</td> </tr> <tr> <td>Variables</td> <td></td> </tr> <tr> <td>UDT_MOT_FC</td> <td>M101 Auslagerpumpe_UDT_HMI</td> </tr> </tbody> </table>	Attribute	Value	Graphics		FACEP_SYMBOLS	FACEP_SYMBOLS	Styles		STYLE_BACKGROUND	BACKGROUND_FP1	STYLE_BUTTONS	BUTTON_FP1	STYLE_INPUT	INPUT_FP1	STYLE_OUTPUT	OUTPUT_FP1	STYLE_TEXT_TAG	TEXT_TAG_FP1	Text		TAG_COMMENT	MOT_FC Test Motor FU	TEXT_ILOCK1	Not Aus	TEXT_ILOCK2	Überdruck	TEXT_ILOCK3	Trockenlauf	TEXT_ILOCK4	Becken Vol	UNIT_Y	%	UserRights		MANUAL	Manual	SERVICE	Service	SIMULATION	Simulation	Variables		UDT_MOT_FC	M101 Auslagerpumpe_UDT_HMI
Attribute	Value																																												
Graphics																																													
FACEP_SYMBOLS	FACEP_SYMBOLS																																												
Styles																																													
STYLE_BACKGROUND	BACKGROUND_FP1																																												
STYLE_BUTTONS	BUTTON_FP1																																												
STYLE_INPUT	INPUT_FP1																																												
STYLE_OUTPUT	OUTPUT_FP1																																												
STYLE_TEXT_TAG	TEXT_TAG_FP1																																												
Text																																													
TAG_COMMENT	MOT_FC Test Motor FU																																												
TEXT_ILOCK1	Not Aus																																												
TEXT_ILOCK2	Überdruck																																												
TEXT_ILOCK3	Trockenlauf																																												
TEXT_ILOCK4	Becken Vol																																												
UNIT_Y	%																																												
UserRights																																													
MANUAL	Manual																																												
SERVICE	Service																																												
SIMULATION	Simulation																																												
Variables																																													
UDT_MOT_FC	M101 Auslagerpumpe_UDT_HMI																																												

Typ	Objekt
<p>4. Unter Ereignisse wird das Schließen des Pop-Ups konfiguriert.</p> <p>Unter „Name des Bildes“ und  wird das „Pop-Up Bild MOT_FC“ in seinem Ordner angewählt und so das Schliessen ermöglicht.</p>	
<p>5. Ein neues Bild mit dem Namen „Teilanlage B“ anlegen, unter Bilder mit Doppel Klick auf „<i>Neues Bild hinzufügen</i>“.</p>	

Typ	Objekt
<p>6. Per Drag & Drop das Typical „TYPICAL_MOT_FC“ in das Bild „Teilanlage B“ ziehen.</p>	
<p>7. Doppelklick auf das Typical unter „Schnittstelle“, „Variables“ „UDT_MOT_FC“ beim Ordner „HMI-Variablen“ und unter „Standard Variablen“ die entsprechende UDT Struktur / HMI Variable anwählen und so verknüpfen.</p>	
<p>8. Unter Variables / TAG_VISIBLE die Variable TAG_ON auswählen. Diese dient zum ein und ausblenden eines Tags.</p>	

Typ	Objekt
9.	Unter Text den Tagname und die Einheit des Stellwerts eingeben.
10.	Unter Graphics / TYPICAL_SYMBOLE die Typical Symbole auswählen.
11.	Unter TYPE die Art des Motors oder Pumpe Auswählen.



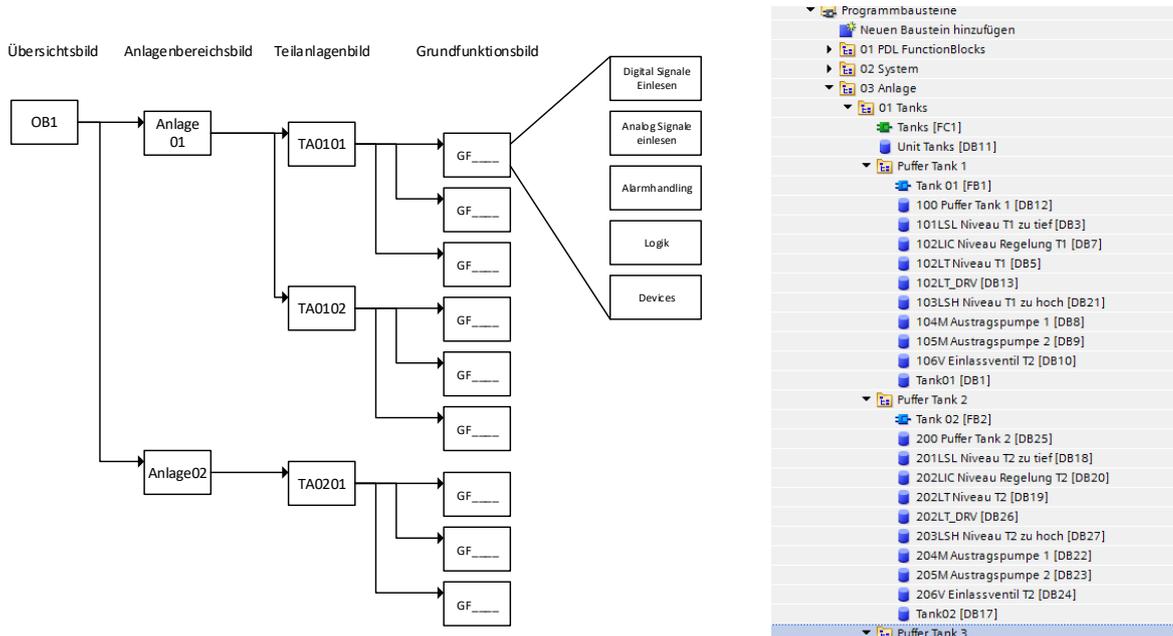
Typ	Objekt
12.	<p>In der Registerkarte Ereignisse den Pop-Up Aufruf konfigurieren.</p> 

3 Programmstruktur

3.1 Programmstruktur allgemein

Alle Programmteile sollten aus dem OB1 aufgerufen werden. Auch Regler werden im OB1 programmiert. Für zeitkritische Aufgaben erfolgt der Funktionsaufruf aus einem Zeit-OB. Aus dem OB1 wird die oberste Ebene aufgerufen

Im TIA Portal müssen die entsprechenden Datenbausteine und evtl. Funktionsbausteine in Gruppen, die der Programmstruktur entsprechen, sortiert werden. Dies vereinfacht die Programmierung, sobald eine komplette Teilanlage kopiert wird. Die Schachteltiefe kann je nach Anlage gewählt werden. Es wird empfohlen mindestens 2 Hierarchieebenen zu erstellen.



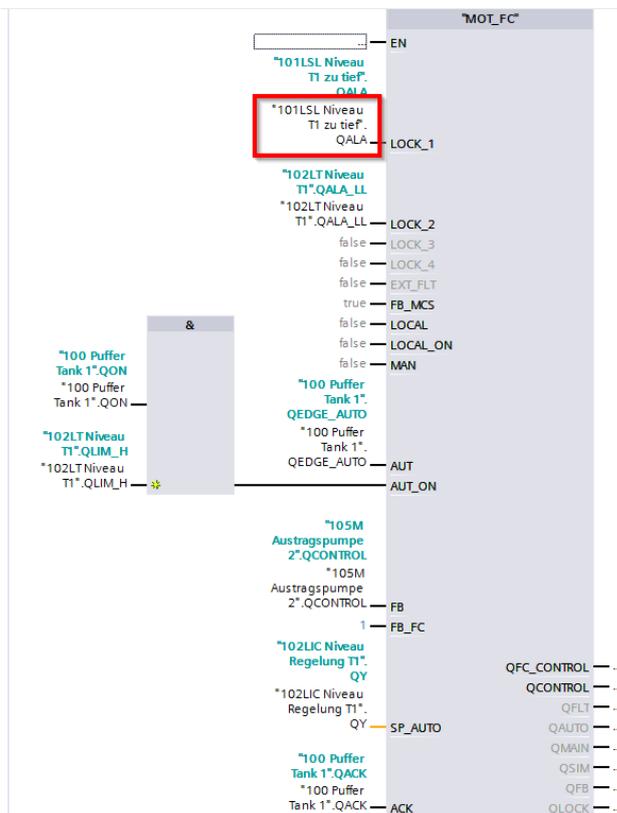
Als Erstsprache des Steuerungsprogramms sollte soweit wie möglich FUP zum Einsatz kommen. Wiederkehrende komplexe Funktionalitäten sollen in SCL geschrieben und gekapselt werden. Die FC's, FB und Instanzbausteinnummern können vom System automatisch vergeben werden. Da symbolisch programmiert wird. Alle FB von 600-699 sind für die PDL Bibliothek reserviert.

3.2 Programmierung

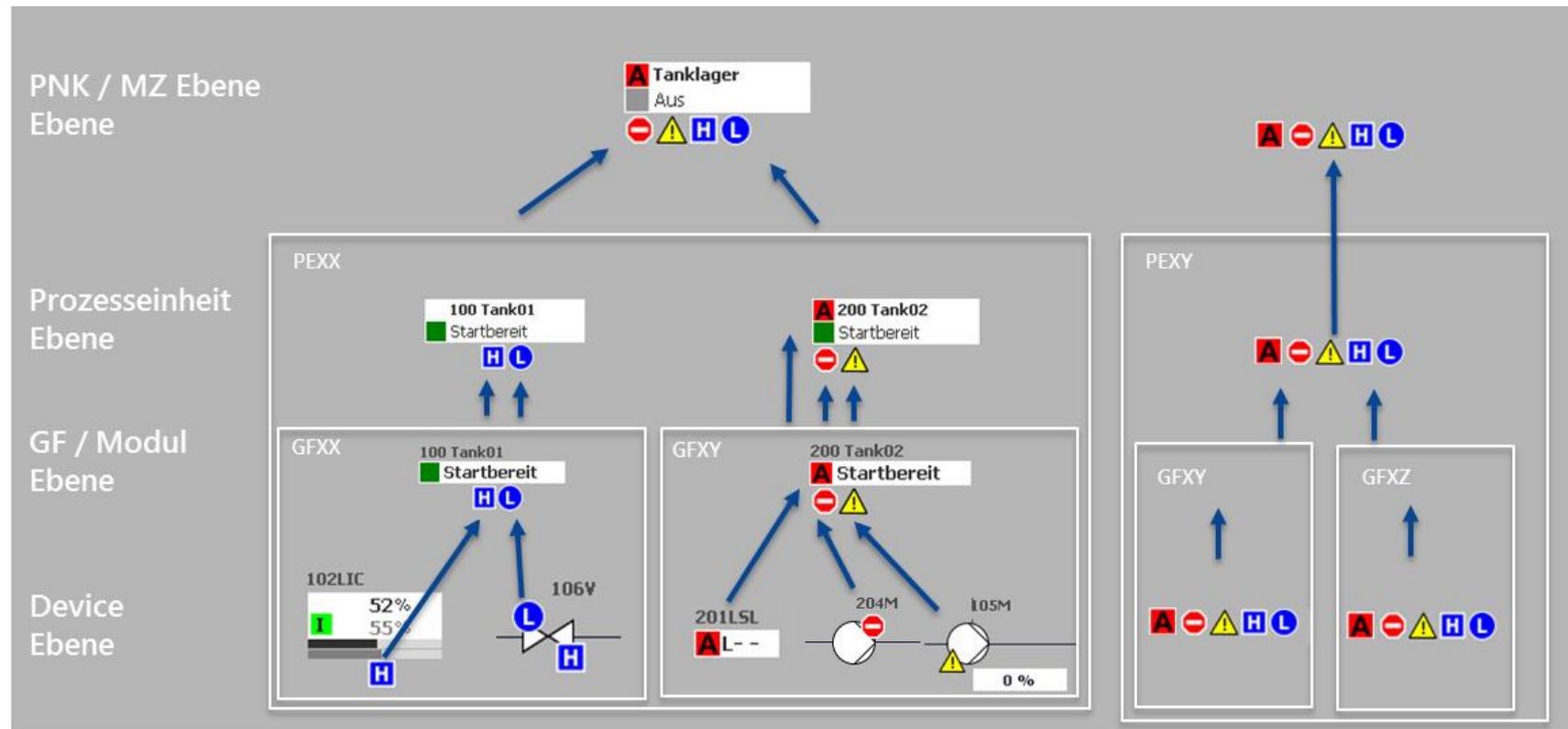
Alle Aggregate, Messungen usw. die zu einem Anlagenteil gehören, müssen in dem entsprechenden Baustein (FC oder FB) aufgerufen werden. Ebenso die Bausteine mit der Programmlogik für diesen Anlagenteil. (Datenkapselung)

Die Namen der Instanz-DB's entsprechen dem AKZ und dem Kommentar, (<Tagname> + <Kommentar>). dass später im Panel angezeigt wird. Sind spezifische Texte für das zugehörige Symbol im Panel vorhanden, werden diese im Kommentar gepflegt.

Zur Programmierung sollen nur die Bausteinanschlüsse verwendet werden z. B für eine Verriegelung von *einem* MOT_FC «101LSL Niveau T1 tief.QALA». Es sollen keine Merker verwendet werden. Ist ein Variable notwendig muss diese im Kopf des Bausteins als STAT-Variable angelegt werden.



3.3 Objektorientierung / Statusübergabe

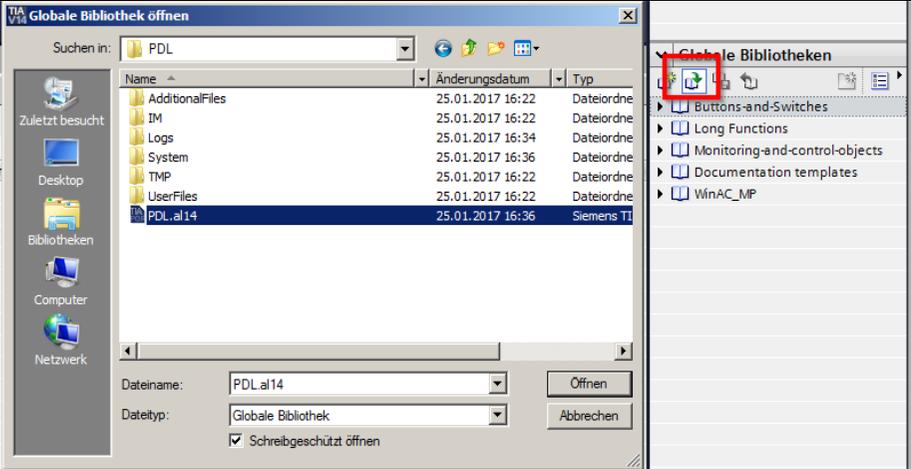


3.4 Weitere Programmierrichtlinien

- ✓ Es soll nach dem EVA (Eingabe Verarbeitung Ausgabe) Prinzip programmiert werden / Englisch IPO für Input Process Output.
- ✓ Instanzbausteinennamen müssen wie folgt benannt werden: aus <Tagname>, «Leerschlag» und <Kommentar>.
- ✓ Jeder Ausgang darf nur an einer Stelle beschrieben werden.
- ✓ Timer müssen in Bausteinen aufgerufen werden und sind als Instanz im FB anzulegen.
- ✓ Die Zykluszeit einer SPS sollte 50 ms nicht überschreiten.
- ✓ Die Programmlogik sollte in FUP erstellt werden.
- ✓ Wenn möglich statisch programmieren, wenig Flanken und Speicher verwenden.
- ✓ E/A-Liste sollte mit der E/A-Liste des CAD-Programms übereinstimmen.
- ✓ Ein Netzwerk sollte nicht größer als eine Bildschirmseite sein.
- ✓ Es sollen keine Merker verwendet werden
- ✓ Es soll symbolisch programmiert werden

4 Bibliotheksupdate

4.1 Bibliotheksupdate einzelnes Bibliothekselement (FACEP, TYPICAL, UDT)

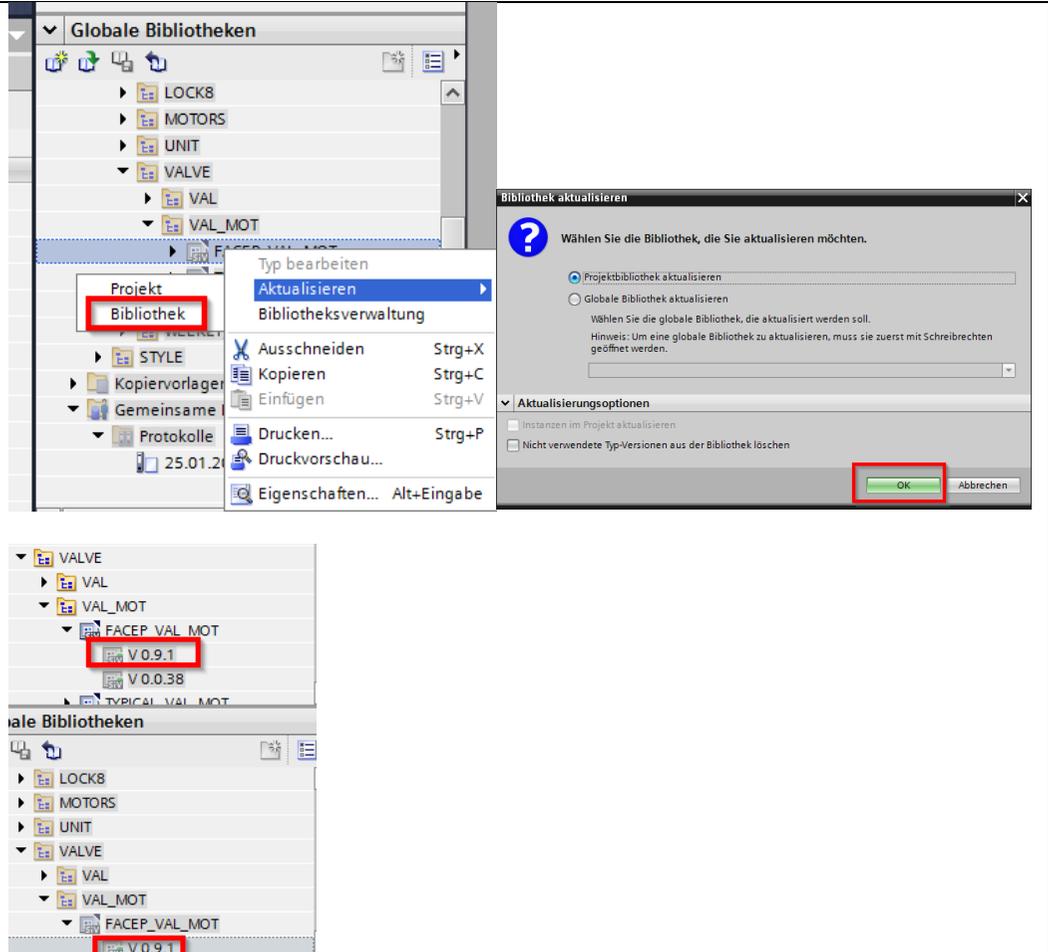
<p>1</p>	<p>Das Projekt welches „upgedated“ werden soll öffnen.</p> <p>Die im PDL VX.X enthaltene globale Bibliothek öffnen.</p>		
----------	-------------------------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------	--

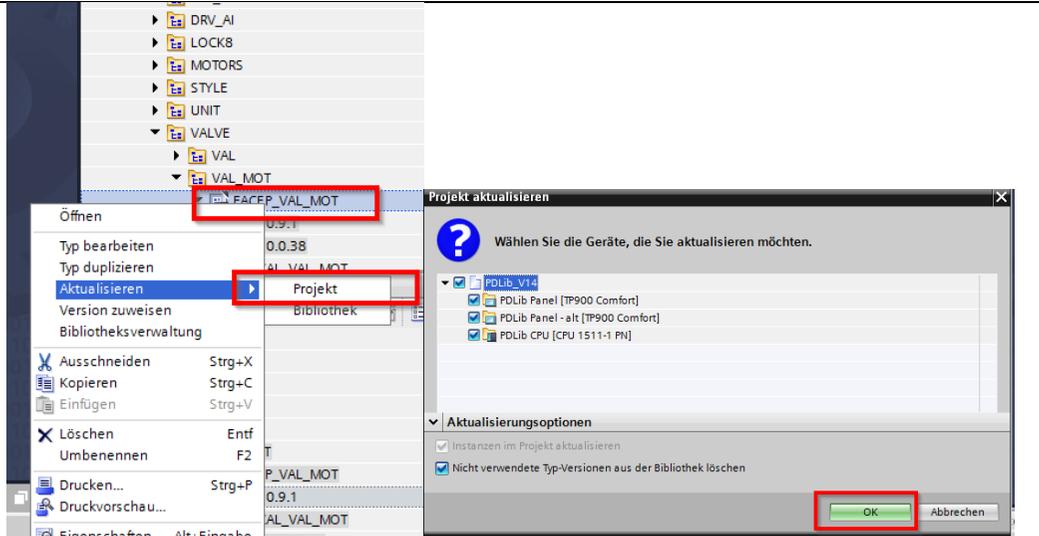
2 Das gewünschte Objekt auswählen, welches in der Projektbibliothek aktualisiert werden soll in der Globalen Bibliothek wählen.

Aktualisieren -> Bibliothek

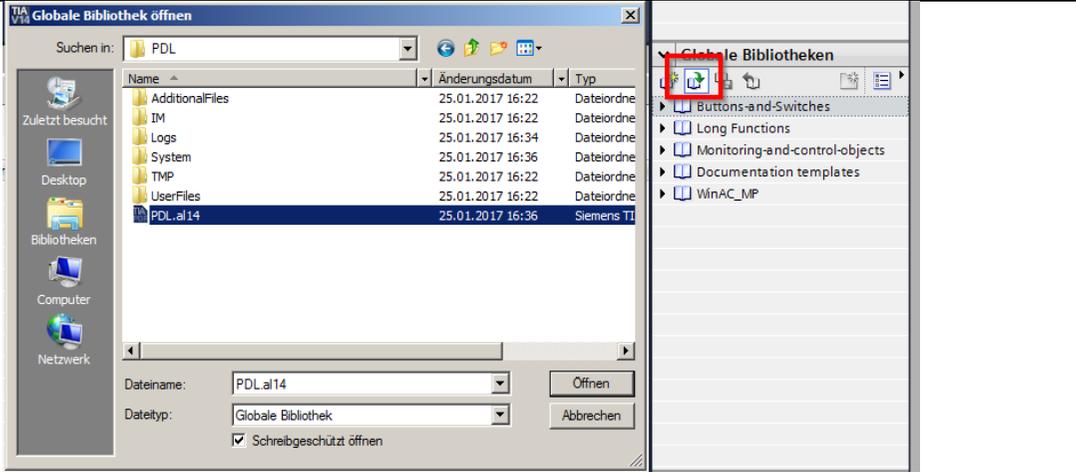
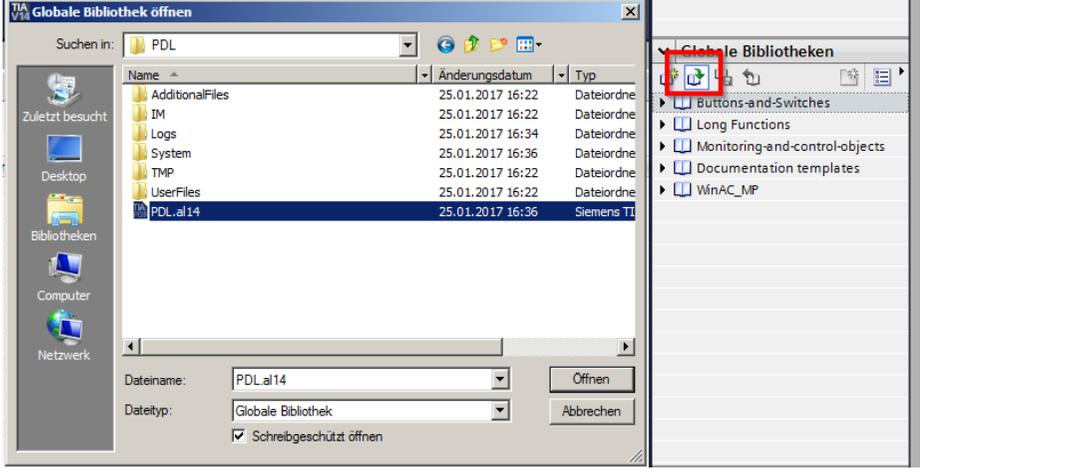
PopUp mit Okay bestätigen.

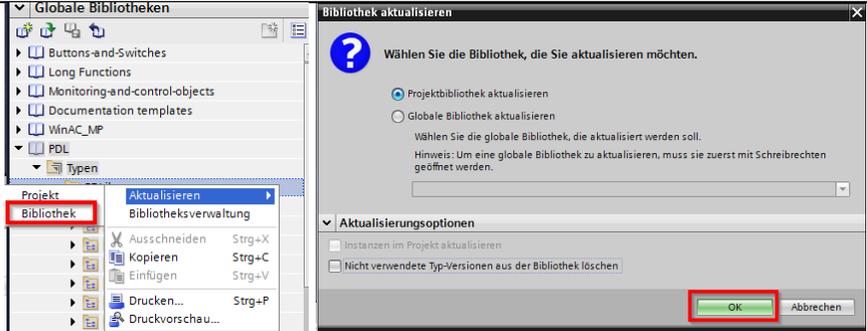
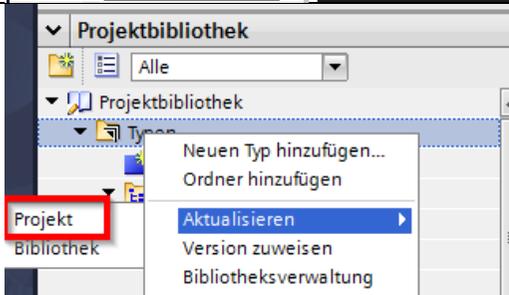
Jetzt ist das FACEP_VAL_MOT in der Projekt Bibliothek aktualisiert.

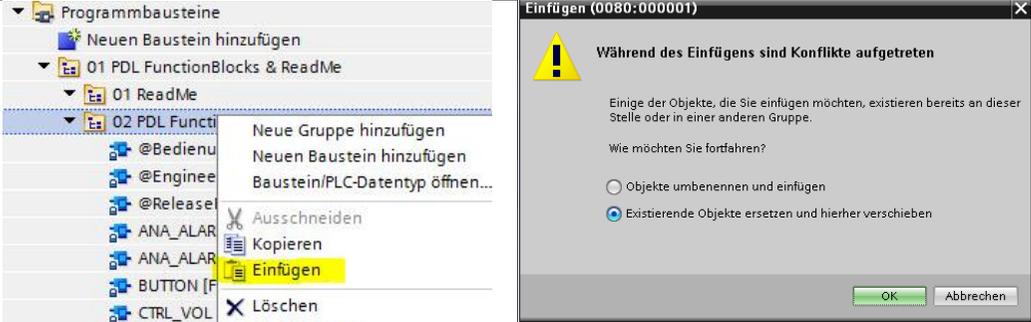


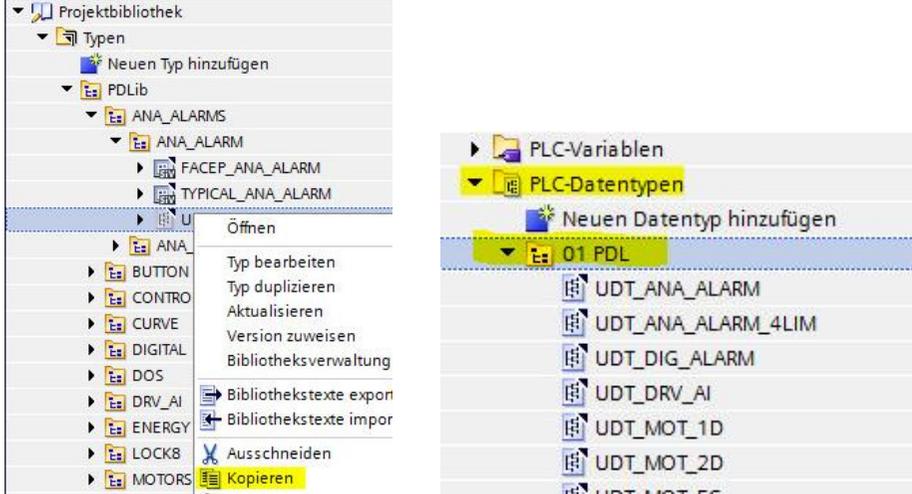
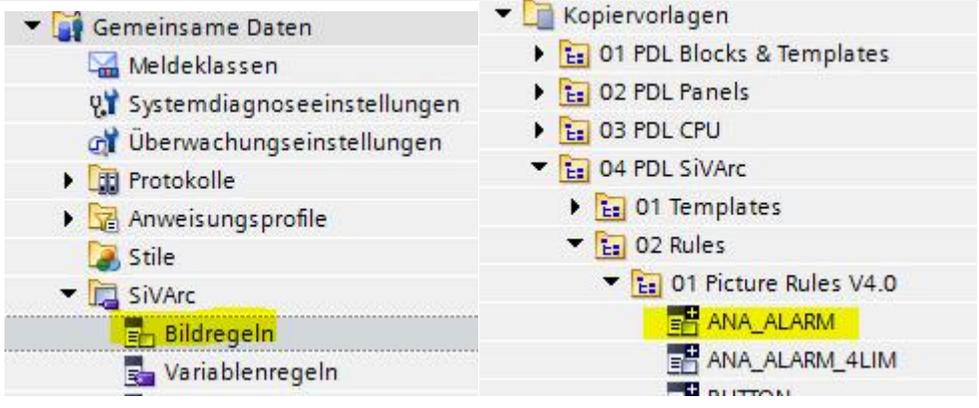
3	Jetzt muss noch das Projekt aktualisiert werden.	
---	--------------------------------------------------	------------------------------------------------------------------------------------

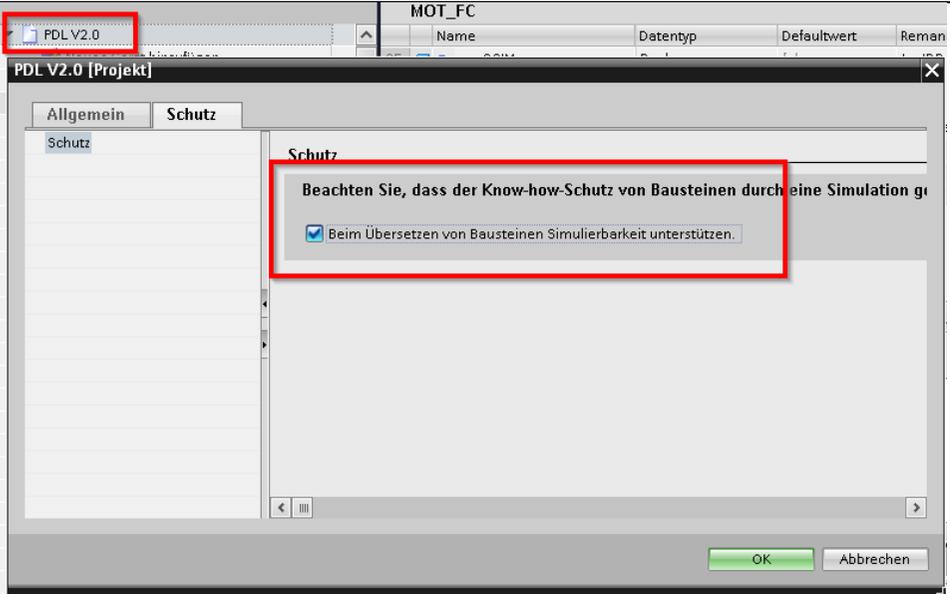
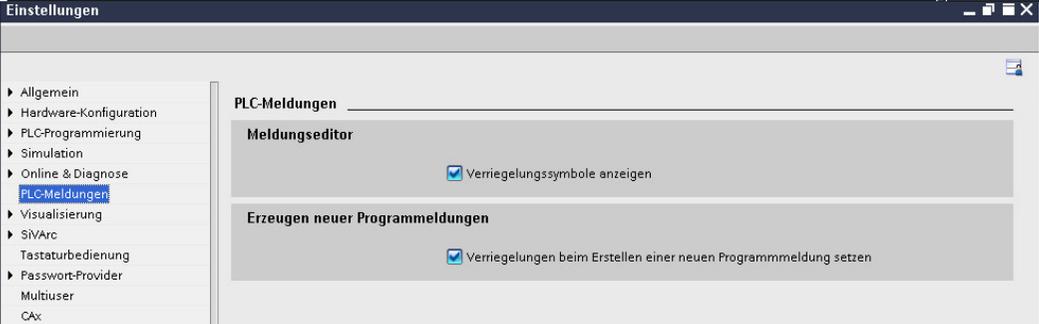
4.2 Bibliothek- / Projektupdate von ganzer Bibliothek von V1.0/V2.0/3.0 nach V4.2

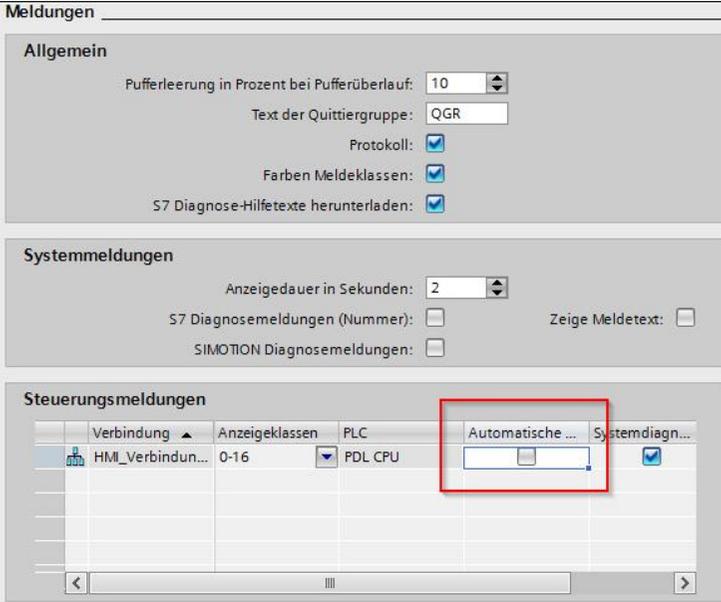
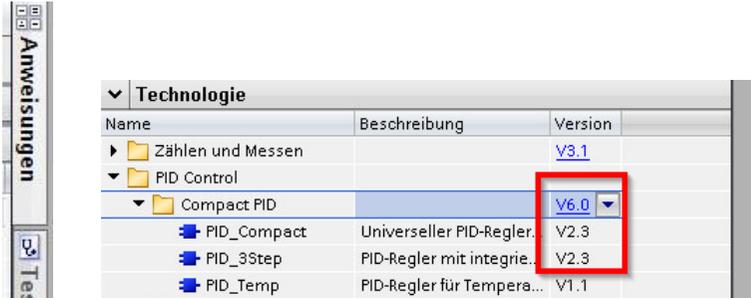
	Typ	Objekt
1.	<p>Das Projekt welches „upgedated“ werden soll öffnen.</p> <p>Die im PDL VX.X enthaltene globale Bibliothek öffnen.</p>	
2.	<p>Das Projekt welches „upgedated“ werden soll öffnen.</p> <p>In diesem Projekt die vorher angelegte Globale Bibliothek öffnen.</p>	

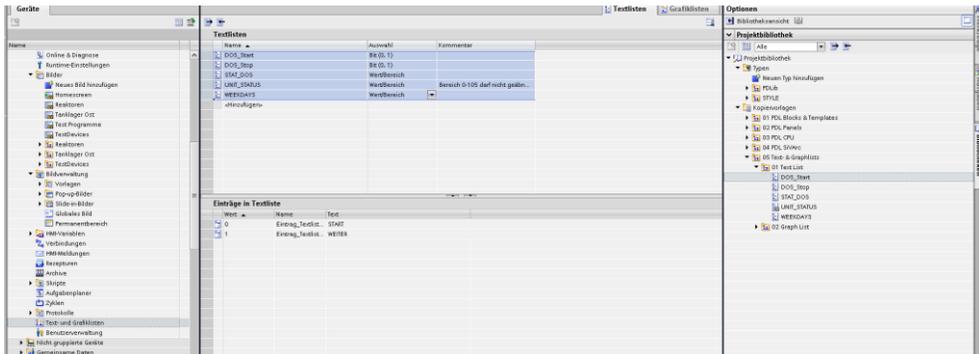
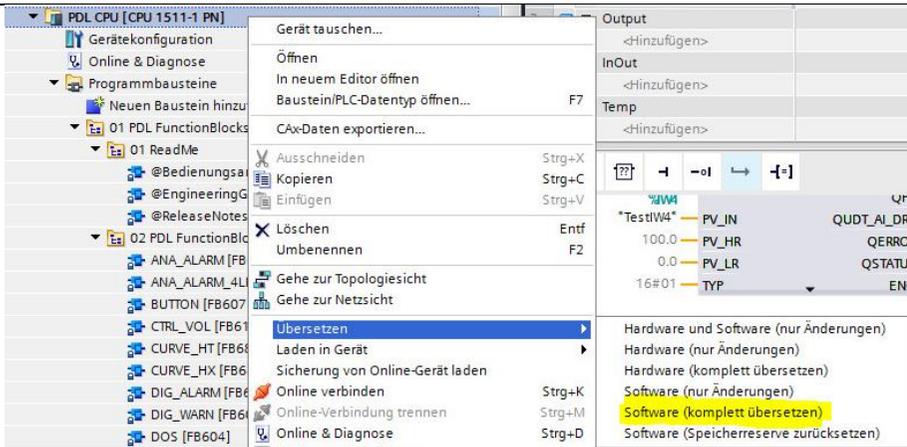
	Typ	Objekt
3.	<p>Wählen sie in der Globalen Bibliothek „Typen“ aus und Rechtsklick -> aktualisieren -> Bibliothek.</p> <p>PopUp mit OK bestätigen.</p>	
4.	<p>Jetzt muss noch das Projekt aktualisiert werden.</p>	
5.	<p>Kopiervorlagen von Globaler in Projekt Bibliothek kopieren.</p> <p>Das PopUp hochrüsten mit OK bestätigen.</p>	

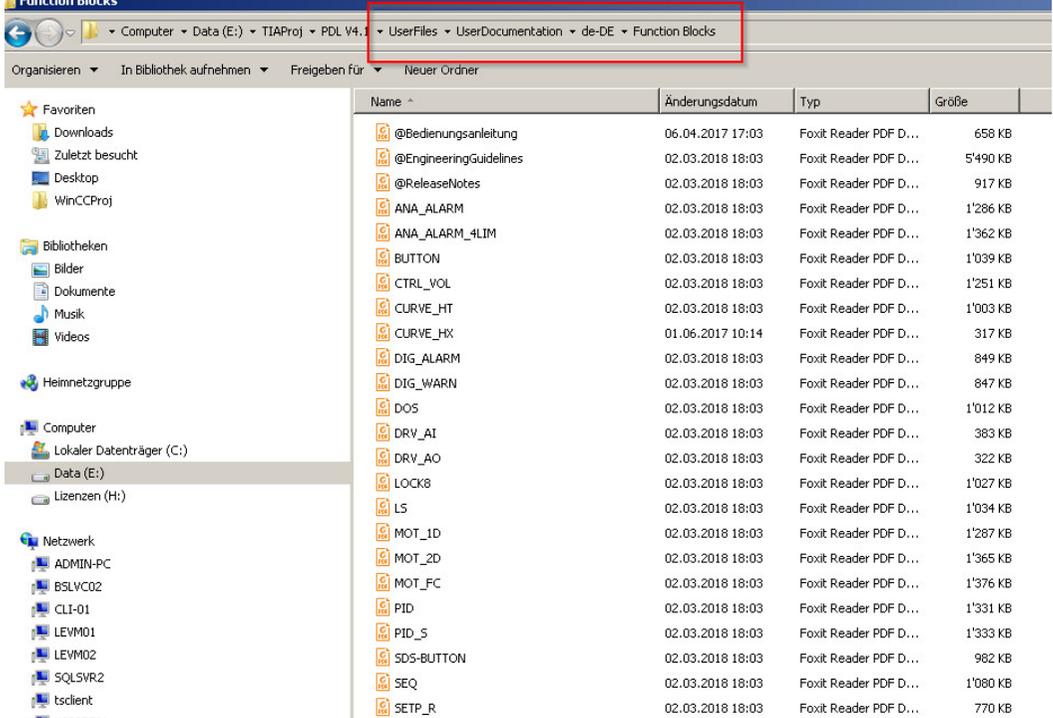
6.	Typ	Objekt
	Alle FB's markieren und kopieren.	
7.	Im PLC Projekt einfügen.	

	Typ	Objekt
8.	Alle neuen UDT Typen in PLC – Datentypen kopieren (einzeln kopieren!) oder hinüberziehen.	
9.	SiVArc Regeln von den neuen Bausteinen von der Projektbibliothek in die gemeinsamen Daten hinüberziehen.	

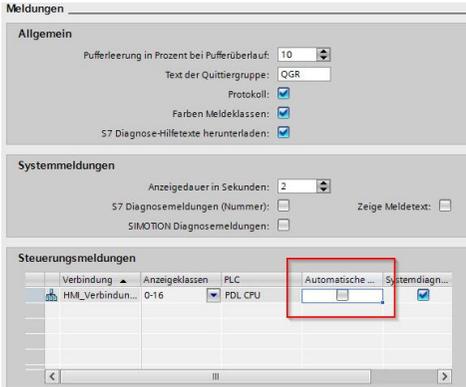
	Typ	Objekt
10.	Simulierbarkeit aktivieren (Ist unter Eigenschaften zu finden) CPU Gesamtübersetzen und Laden HMI Gesamtübersetzen und Laden	
11.	Extras; Einstellungen PLC Verriegelungen aktivieren	

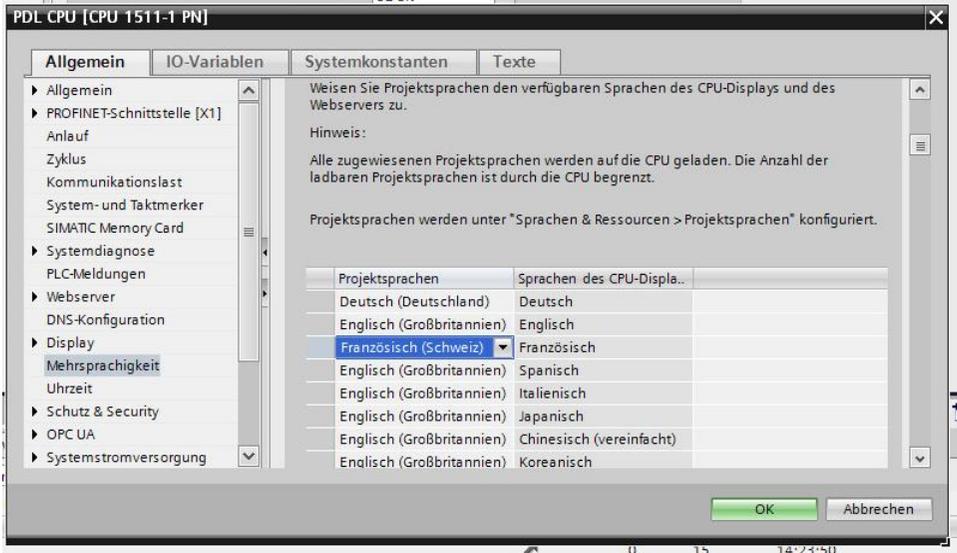
12.	Typ	Objekt
	Runtime Einstellungen von Panel	
13.	<p>Damit der PID Regler läuft, muss die Version 6.0 der PID Regler aktiviert werden. Irgendeinen Baustein öffnen, im Register Anweisungen, Technologie öffnen.</p>	

Typ	Objekt
<p>14. Alle von «05 Text und Graphiclists» alle Grafiklisten in HMI unter Text – und Grafiklisten kopieren.</p>	 <p>The screenshot shows the SIMATIC Manager interface. On the left, a tree view shows the project structure. The main window displays a table of 'Textlisten' with columns for Name, Auswahl, and Kommentar. Below it, a table 'Einträge in Textliste' shows entries for '0' and '1'. On the right, the 'Optionen' dialog is open, showing the 'Projektbibliothek' section with various library items like 'DOS_Text', 'DOS_Stop', etc.</p>
<p>15.</p> <ul style="list-style-type: none"> - Steuerung Gesamtübersetzen - HMI Gesamtübersetzen 	 <p>The screenshot shows the SIMATIC Manager interface with a context menu open over the 'Übersetzen' option. The menu includes options like 'Gerät tauschen...', 'Öffnen', 'In neuem Editor öffnen', 'Baustein/PLC-Datentyp öffnen...', 'CAx-Daten exportieren...', 'Ausschneiden', 'Kopieren', 'Einfügen', 'Löschen', 'Umbenennen', 'Gehe zur Topologiesicht', 'Gehe zur Netzsicht', 'Übersetzen', 'Laden in Gerät', 'Sicherung von Online-Gerät laden', 'Online verbinden', 'Online-Verbindung trennen', and 'Online & Diagnose'. The 'Übersetzen' option is expanded, showing sub-options: 'Hardware und Software (nur Änderungen)', 'Hardware (nur Änderungen)', 'Hardware (komplett übersetzen)', 'Software (nur Änderungen)', 'Software (komplett übersetzen)', and 'Software (Speicherreserve zurücksetzen)'. The 'Software (komplett übersetzen)' option is highlighted in yellow.</p>

	Typ	Objekt
16.	- Fehlende Bausteindokus kopieren, damit der die Hilfe per CTRL Shift geöffnet werden kann.	
17.	- Beim Device PID muss die Zeile « Format P:s99,9; » ergänzt werden, damit via SiVArc alle das Format des P Anteils bestimmt werden kann.	

Typ	Objekt
<p>18.</p> <ul style="list-style-type: none"> - Bei allen Devices muss die erste Zeile durch: Kommentar; ersetzt werden. Wenn es sich um ein mehrsprachiges Projekt handelt muss dies in jeder Sprache gemacht werden. Es empfiehlt sich dies über die Funktion: «Suchen & Ersetzen» zu machen. 	<pre> Kommentar; Unit SP:mbar; Lock_1:Lock_1; Lock_2:Lock_2; Lock_3:Lock_3; Lock_4:Lock_4; Format SP:9999; Unit Y:%; Format Y:999,9; Format P:s99,9; </pre>
<p>19.</p> <ul style="list-style-type: none"> - Jeder FB oder FC in welchem PDL Devices aufgerufen werden braucht in der ersten Kommentarzeile des Netzwerkes Picture Title: «Titel für das Bild» - Diese Änderung wurde gemacht damit Bildtitel Multilingual werden können. 	<p>Netzwerk 1: Test Devices V2</p> <p>Picture Title:Test Devices V2;</p> 

20.	Typ	Objekt
	- Diese Funktion muss deaktiviert werden, damit die Sprachumschaltung beim Program_Alarm funktioniert	

	Typ	Objekt																		
21.	- Sprachen von CPU Einstellen	 <p>The screenshot shows the 'PDL CPU [CPU 1511-1 PN]' configuration window. The 'Allgemein' tab is selected. The left sidebar shows a tree view with 'Mehrsprachigkeit' highlighted. The main content area contains the following text:</p> <p>Weisen Sie Projektsprachen den verfügbaren Sprachen des CPU-Displays und des Webservers zu.</p> <p>Hinweis: Alle zugewiesenen Projektsprachen werden auf die CPU geladen. Die Anzahl der ladbaren Projektsprachen ist durch die CPU begrenzt.</p> <p>Projektsprachen werden unter "Sprachen & Ressourcen > Projektsprachen" konfiguriert.</p> <table border="1" data-bbox="1317 564 1906 772"> <thead> <tr> <th>Projektsprachen</th> <th>Sprachen des CPU-Displa..</th> </tr> </thead> <tbody> <tr> <td>Deutsch (Deutschland)</td> <td>Deutsch</td> </tr> <tr> <td>Englisch (Großbritannien)</td> <td>Englisch</td> </tr> <tr> <td>Französisch (Schweiz)</td> <td>Französisch</td> </tr> <tr> <td>Englisch (Großbritannien)</td> <td>Spanisch</td> </tr> <tr> <td>Englisch (Großbritannien)</td> <td>Italienisch</td> </tr> <tr> <td>Englisch (Großbritannien)</td> <td>Japanisch</td> </tr> <tr> <td>Englisch (Großbritannien)</td> <td>Chinesisch (vereinfacht)</td> </tr> <tr> <td>Englisch (Großbritannien)</td> <td>Koreanisch</td> </tr> </tbody> </table> <p>Buttons: OK, Abbrechen</p>	Projektsprachen	Sprachen des CPU-Displa..	Deutsch (Deutschland)	Deutsch	Englisch (Großbritannien)	Englisch	Französisch (Schweiz)	Französisch	Englisch (Großbritannien)	Spanisch	Englisch (Großbritannien)	Italienisch	Englisch (Großbritannien)	Japanisch	Englisch (Großbritannien)	Chinesisch (vereinfacht)	Englisch (Großbritannien)	Koreanisch
Projektsprachen	Sprachen des CPU-Displa..																			
Deutsch (Deutschland)	Deutsch																			
Englisch (Großbritannien)	Englisch																			
Französisch (Schweiz)	Französisch																			
Englisch (Großbritannien)	Spanisch																			
Englisch (Großbritannien)	Italienisch																			
Englisch (Großbritannien)	Japanisch																			
Englisch (Großbritannien)	Chinesisch (vereinfacht)																			
Englisch (Großbritannien)	Koreanisch																			

5 System

5.1 Systemvoraussetzungen Hardware

Die PDL ist läuft auf allen S7 15XX er und 15XXSP für 200SP. Es werden die Panel TP700 bis TP2200 unterstützt.

Typ		Firmware Version
S7-15XX	S7 Steuerung	Ab V2.1
S7-15XX-SP	S7 Steuerung für ET200 SP	Ab V2.1
TP 700/900/ 1200 / 1500 / 1900/2200	Comfort Panel	Ab 14.0.1.0

5.2 Systemvoraussetzungen Software

Typ		Version
TIA	Totally Integrated Automation Portal	Ab V14 SP1
TIA WinCC	WinCC Comfort / Advanced	Ab V14 SP1
Step 7	Professional	Ab V14 SP1
SiVArc	Simatic Visualization Architect (Optional)	Ab V14 SP1
WinCC V7.x	WinCC Scada	Ab V7.4 Update 9

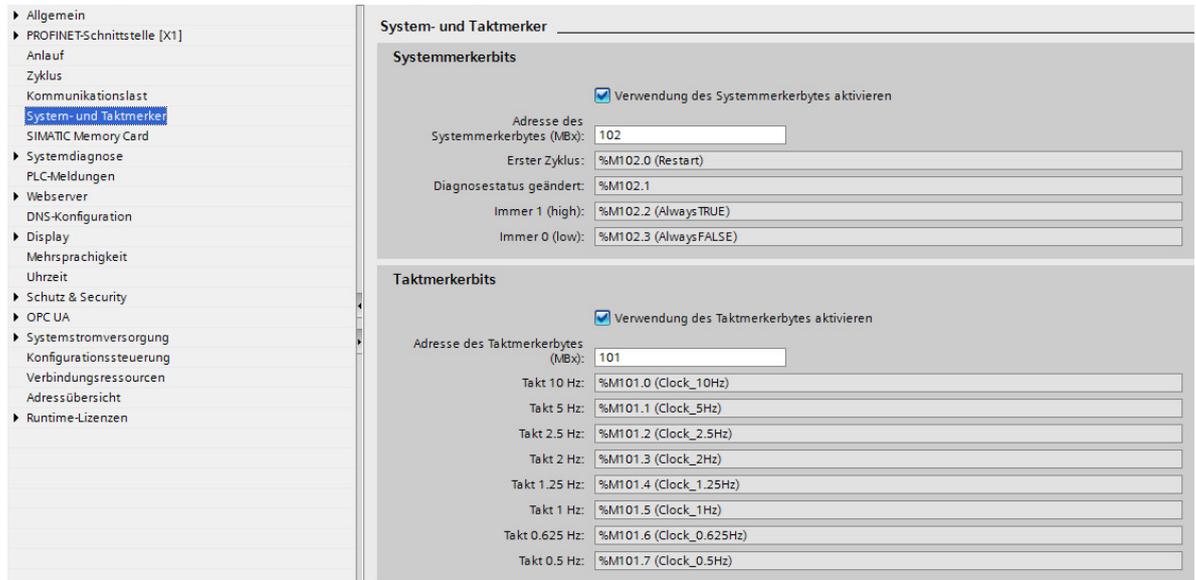
5.3 Standard

Mit dem STANDARD-Baustein werden grundlegende Variablen zur Verfügung gestellt. Dieser muss immer einmal vorhanden sein. Der Instanz-DB muss den Namen «STANDARD_DB» haben.

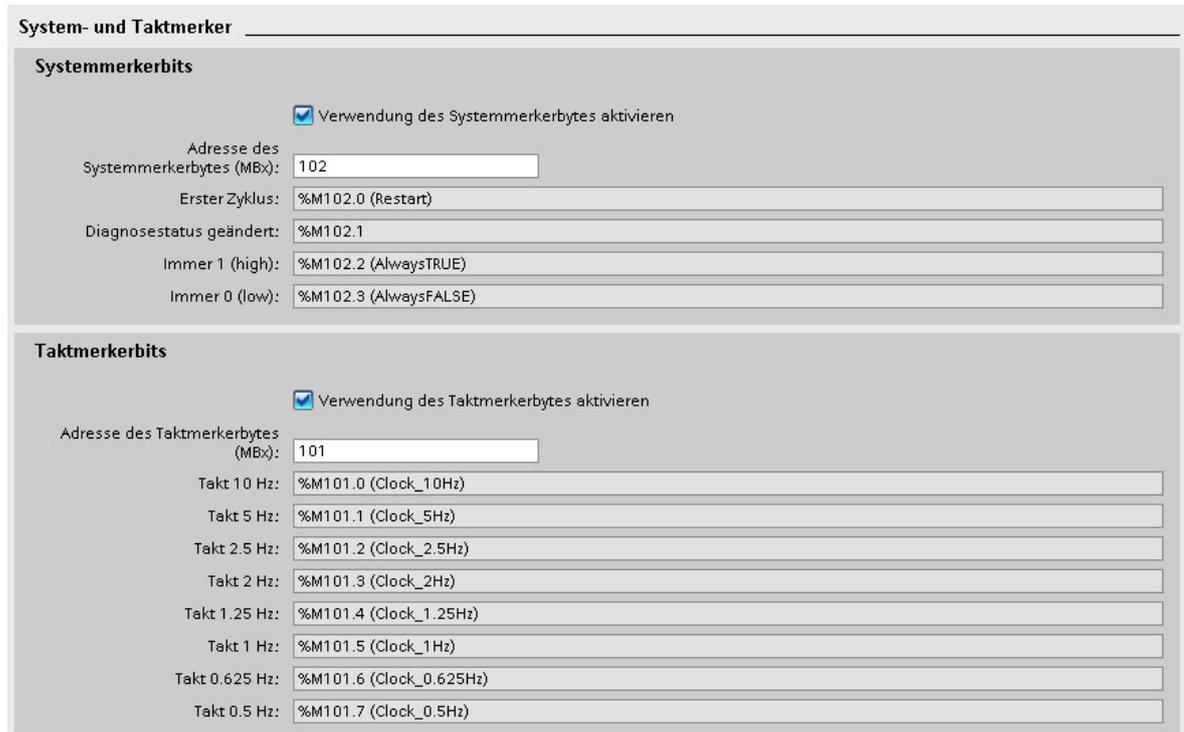
5.4 Systemmerker

In der Gerätekonfiguration der CPU sind die System- und Taktmerker einzustellen. Das Systemmerkerbyte befindet sich im MB102. Das Taktmerkerbyte befindet sich im MB101.

Im Bereich ab Merker 100.0 stehen die System Merker:

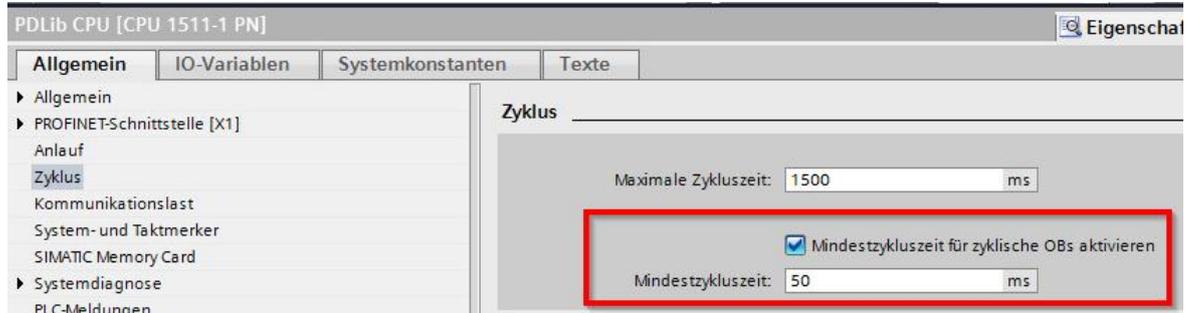


Für das Quittieren vom Panel wird der Merker M100.0 festgelegt. Dieser ist solange aktiv, wie die Quittiertaste gedrückt ist.



5.5 OB1 min. Zykluszeit für Regler

Damit der PID und PID_S korrekt funktionieren müssen diese die Mindestzykluszeit von auf 50ms gestellt werden.



5.6 Hardwareaufbau

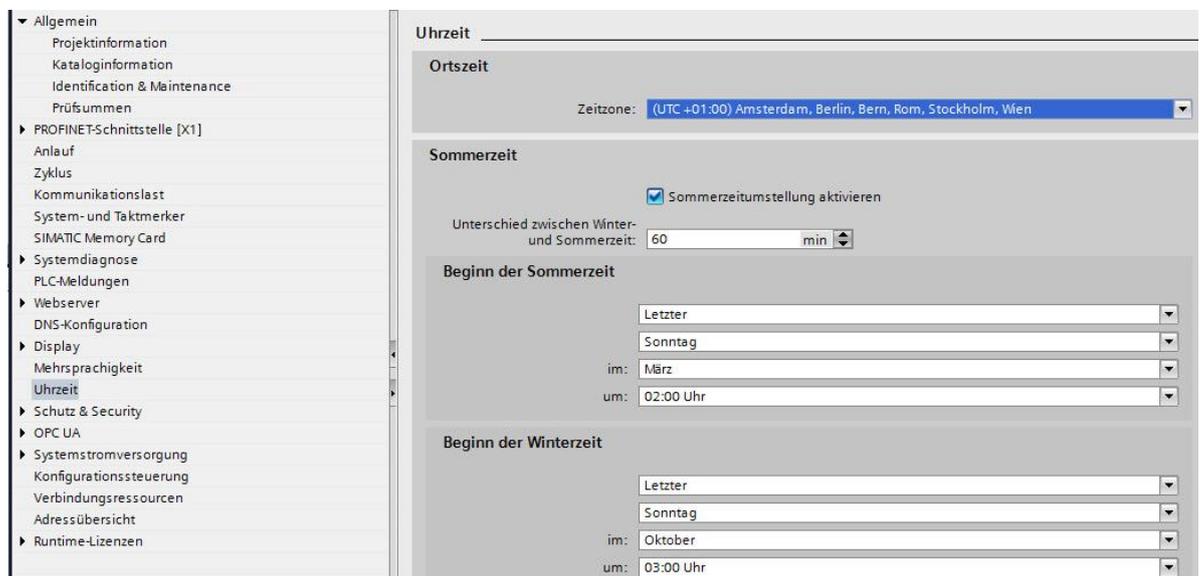
In der Gerätekonfiguration sollte auf dem Rack der CPU immer so viel Platz in Reserve sein, um weitere CP's zu stecken.

Die CPU ist nach den Vorgaben in der Systemeinstellung zu konfigurieren.

Für die Baugruppen muss zumindest der Name, das Anlagenkennzeichen und das Ortskennzeichen gepflegt werden. Für die Diagnose ist die fehlende Versorgungsspannung zu aktivieren. Weitere Einstellungen sind kundenspezifisch.

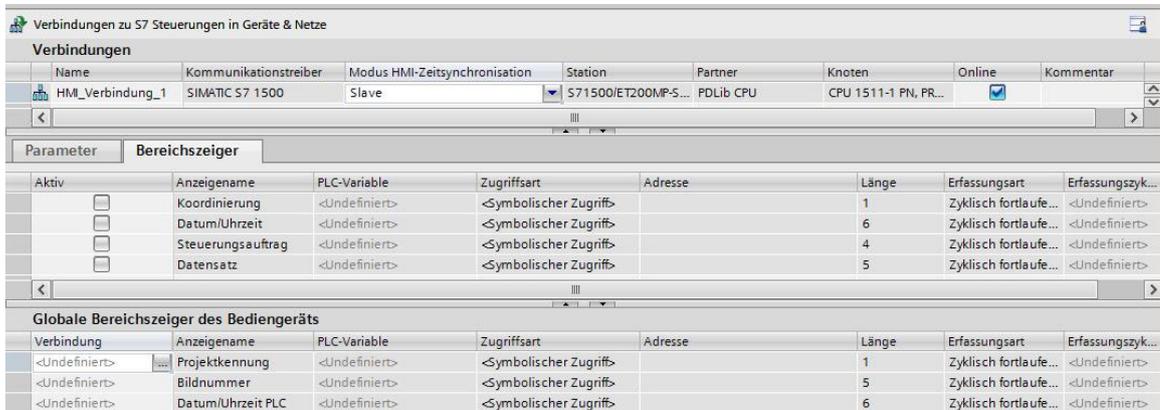
5.7 Uhrzeit / Zeitsynchronisation

Die Sommerzeit ist aktiviert.



5.8 Zeitsynchronisation

Die Zeitsynchronisation erfolgt über die CPU.



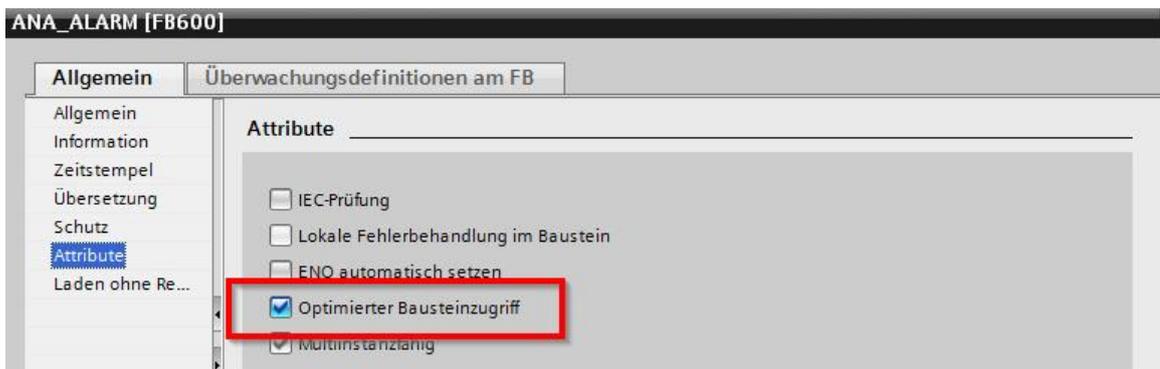
Name	Kommunikationstreiber	Modus HMI-Zeitsynchronisation	Station	Partner	Knoten	Online	Kommentar
HMI_Verbindung_1	SIMATIC S7 1500	Slave	S71500/ET200MP-S...	PDLib CPU	CPU 1511-1 PN, PR...	<input checked="" type="checkbox"/>	

Parameter	Bereichszeiger
Aktiv	
<input type="checkbox"/>	Koordinierung
<input type="checkbox"/>	Datum/Uhrzeit
<input type="checkbox"/>	Steuerungsauftrag
<input type="checkbox"/>	Datensatz

Verbindung	Anzeigename	PLC-Variablen	Zugriffsart	Adresse	Länge	Erfassungsart	Erfassungszyklus
<undefiniert>	Projektkennung	<undefiniert>	<Symbolischer Zugriff>		1	Zyklisch fortlaufe...	<undefiniert>
<undefiniert>	Bildnummer	<undefiniert>	<Symbolischer Zugriff>		5	Zyklisch fortlaufe...	<undefiniert>
<undefiniert>	Datum/Uhrzeit PLC	<undefiniert>	<Symbolischer Zugriff>		6	Zyklisch fortlaufe...	<undefiniert>

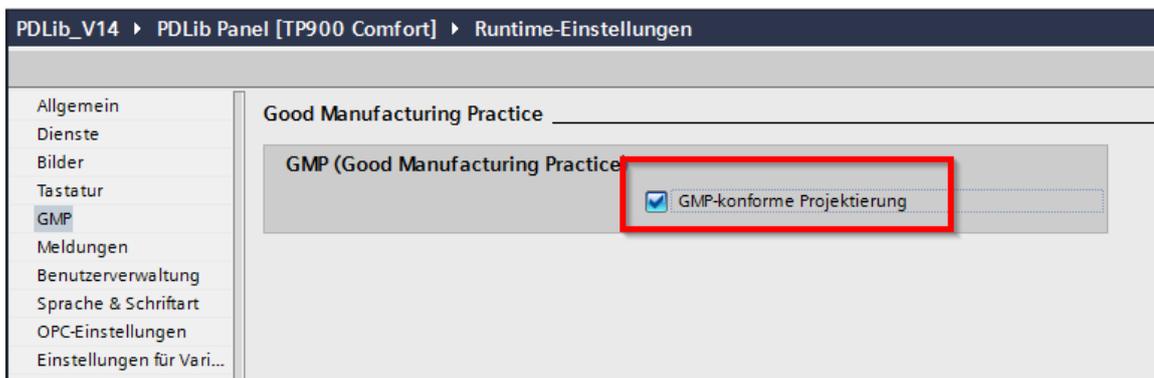
5.9 Optimierte Bausteine

Alle Bausteine wurden mit optimiertem Bausteinzugriff programmiert. Somit kann von 3th Party System nicht absolut auf die DB's zugegriffen werden. Die Kommunikation zu andern Systemen muss mit OPC UA erfolgen.

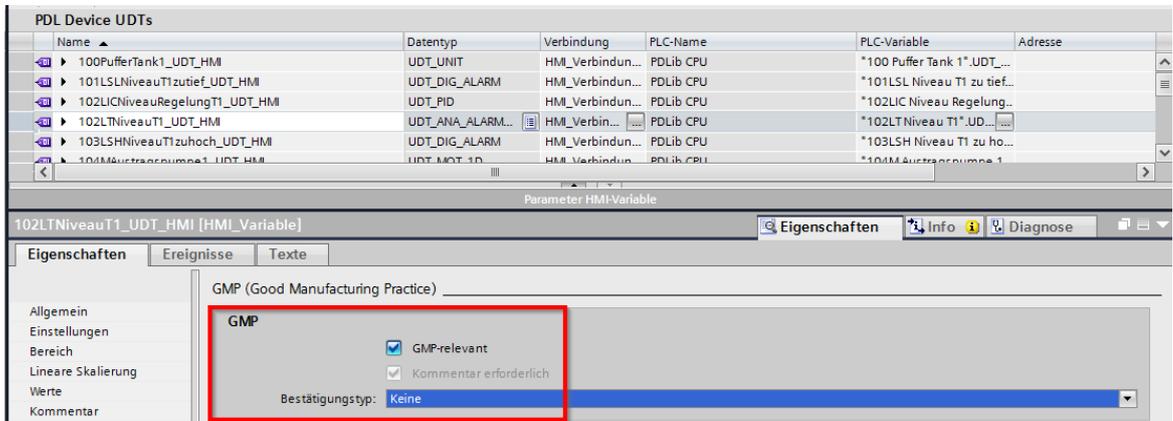


5.10 GMP Einstellungen

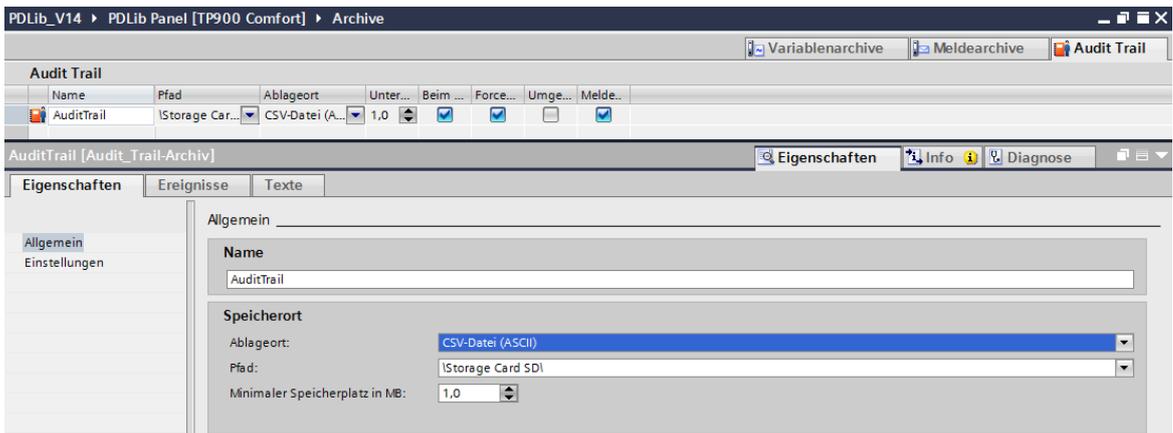
Damit Audit Trail aktiviert wird muss unter dem Panel bei den Runtime Einstellungen die Option GMP-konforme Projektierung angewählt werden.



Zudem müssen unter HMI-Variablen alle UDT's der Devices angewählt werden und die Option GMP-relevant gesetzt werden.



Anschliessend kann unter Archive/Audit das Archiv konfiguriert werden.



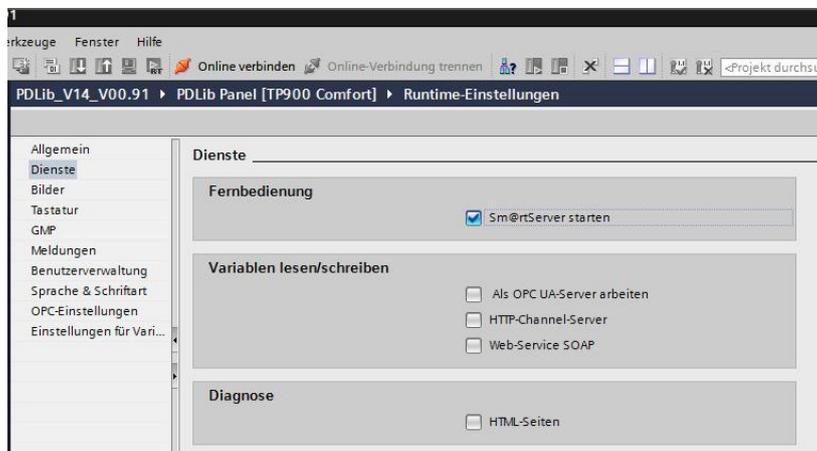
5.11 Anlaufverhalten der Devices

Der aktuelle Status des Devices ist remanent. Somit bleiben alle Stati wie Simulation Ein / Aus, Lokal / Fern usw. erhalten nach einem Neustart der CPU.

Spezielle Anlaufverhalten von Ventilen und Motoren können mit dem Anschluss "MODE" definiert werden. Genauere Infos sind in der jeweiligen Bausteinbeschreibung.

5.12 Sm@rtserver Einstellungen

Unter dem Panel Runtime Einstellungen die Fernbedienung/Sm@rtServer starten setzen. Am Panel unter Settings WinCC Internet Settings / Remote Passwort vergeben. Dies ist nur möglich, wenn Smart Server gestoppt ist. Im Panel muss unter Settings-> WinCC Internet Settings -> Remote ein Passwort vergeben werden. Dies muss in der App eingegeben werden. Die S@martServer App vom Google Playstore oder Apple Store auf Smartphone oder Tablet installieren, anschliessend die IP-Adresse angeben und starten.



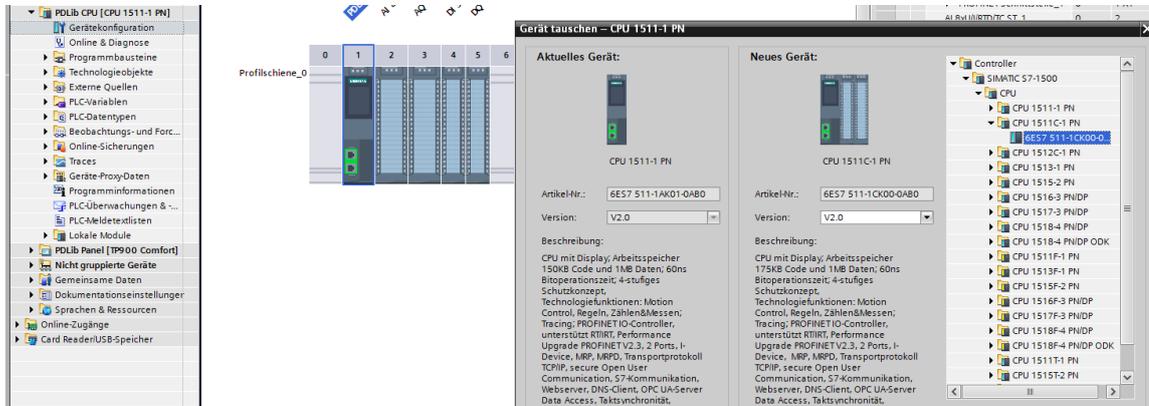
5.13 Panel Tauschen

Beim Panel unter «Gerätekonfiguration» mit Rechtsklick Objekttauschen und das TP700 bzw. TP900 auswählen. Das gleiche gilt für TP1200 nach TP1500.



5.14 CPU Tauschen

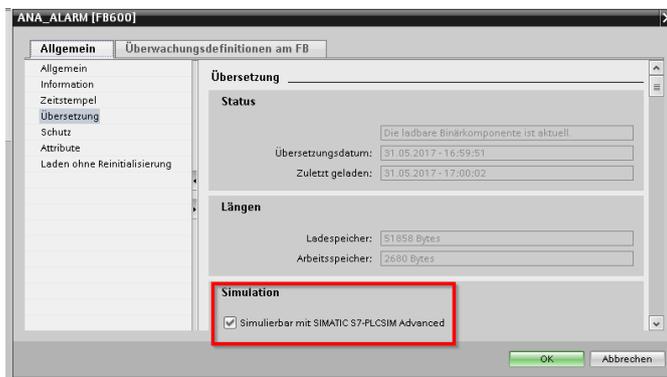
Bei der CPU unter «Gerätekonfiguration» mit Rechtsklick Objekttauschen und die gewünschte CPU.



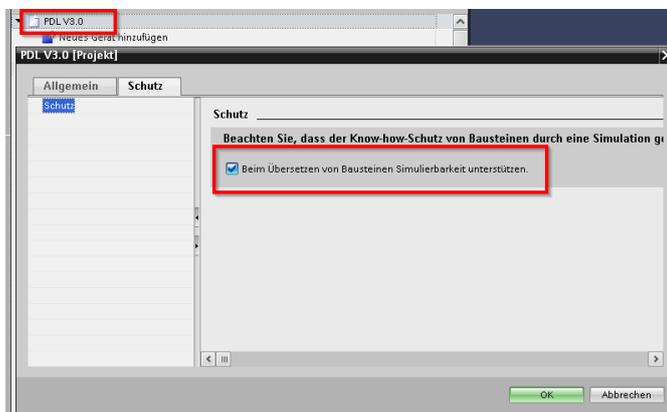
5.15 PLC Sim Advanced

Alle Bausteine wurden mit der Option "Simulierbarkeit mit PLC SIM Advanced" aktiviert. Mit PLC Sim Advanced können auch geschützte Bausteine Simuliert werden. Es ist auch möglich mehrere Instanzen von Steuerungen zu Simulieren.

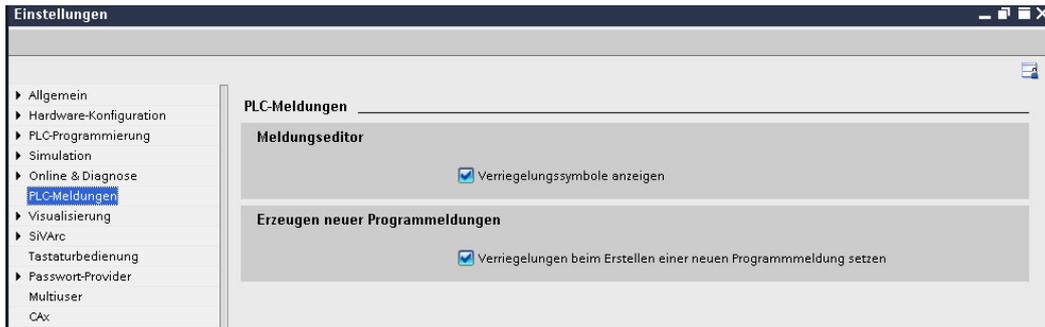
Bausteinoption:



Projekteinstellung:



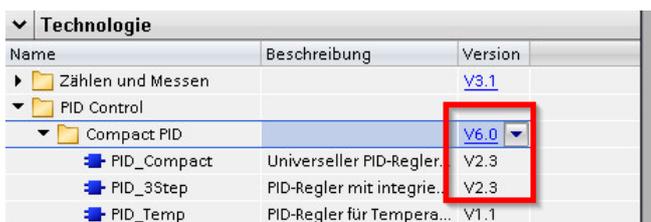
5.16 PLC Meldungen Verriegelungen aktivieren



Damit ist es möglich instanzspezifische Meldungseigenschaften zu definieren.

5.17 Technology Objekte - PID Regler Version

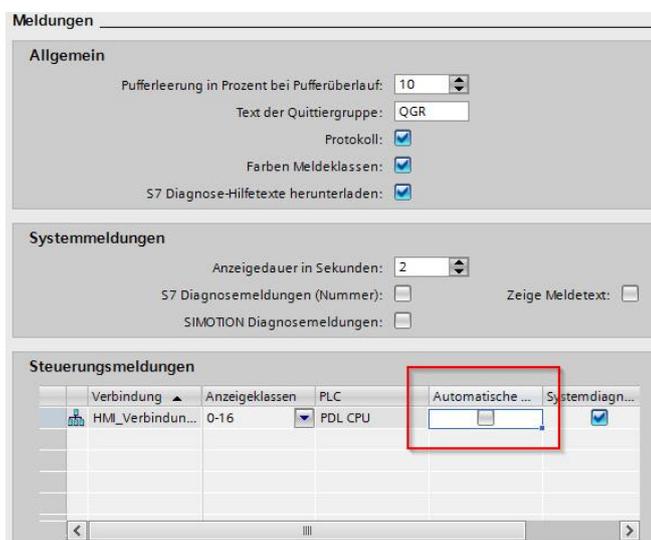
Damit der PID Regler läuft, muss die Version 6.0 der Regler Algorithmen aktiviert werden.



Nur mit der Version 2.3 läuft der PID Regler auch auf einer 1510-SP und 1512-SP

5.18 HMI Meldungen Runtime Einstellungen

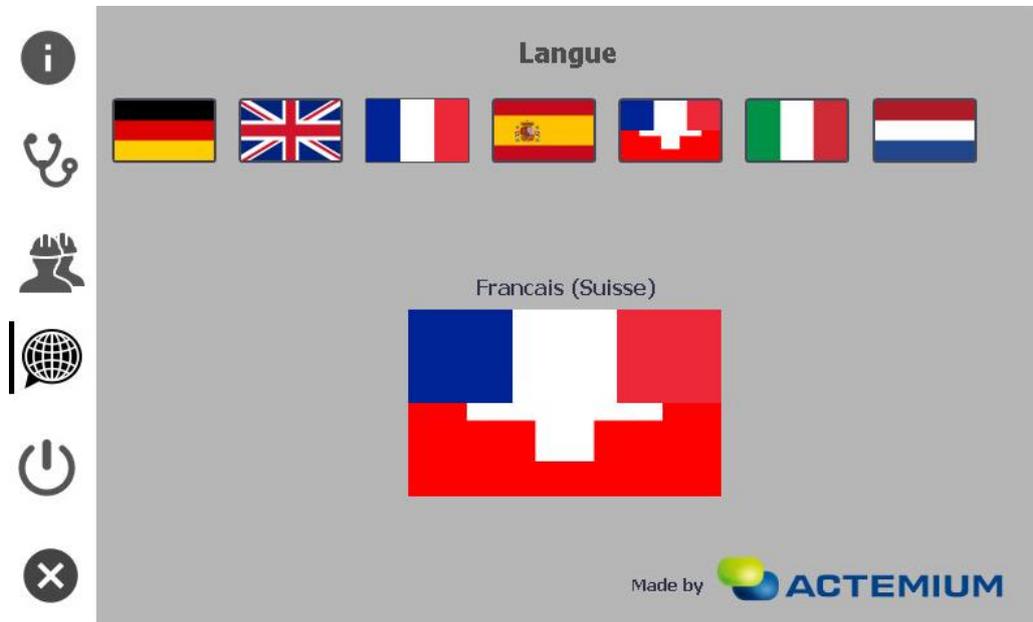
Diese Funktion muss deaktiviert werden, damit die Sprachumschaltung beim Program_Alarm funktioniert.



5.19 Sprachumschaltung

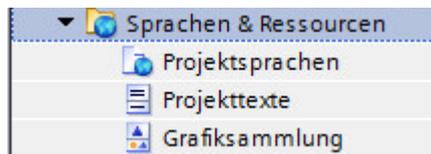
5.19.1 Sprachumschaltung auf HMI

Die Sprache kann im Systemmenu umgestellt werden.



5.19.2 Einstellungen im TIA

(1) Im Projekt:



Unter Sprachen & Ressourcen lassen sich die Projektsprachen und Projekttexte projektieren.

Allgemein

Editiersprache: Deutsch (Deutschland) Referenzsprache: Englisch (Großbritannien)

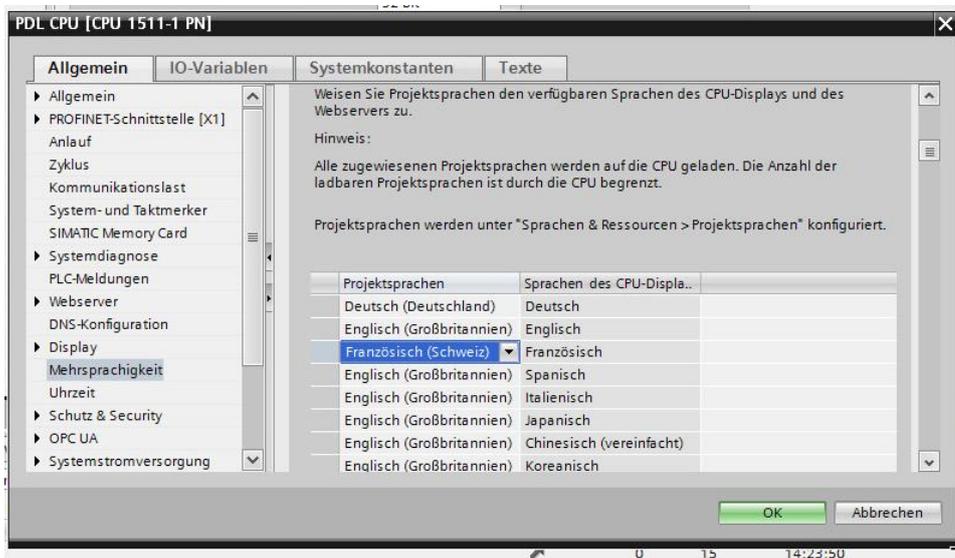
<input type="checkbox"/> Afrikaans (Südafrika) <input type="checkbox"/> Albanisch (Albanien) <input type="checkbox"/> Armenisch (Armenien) <input type="checkbox"/> Aserbaidshanisch (Kyrillisch, Aserbaidshan) <input type="checkbox"/> Aserbaidshanisch (Lateinisch, Aserbaidshan) <input type="checkbox"/> Baskisch (Baskenland) <input type="checkbox"/> Belarussisch (Belarus) <input type="checkbox"/> Bulgarisch (Bulgarien) <input type="checkbox"/> Chinesisch (Hong Kong S.A.R.) <input type="checkbox"/> Chinesisch (Macao S.A.R.) <input type="checkbox"/> Chinesisch (Singapur) <input type="checkbox"/> Chinesisch (Taiwan) <input checked="" type="checkbox"/> Chinesisch (VR China) <input type="checkbox"/> Dänisch (Dänemark) <input checked="" type="checkbox"/> Deutsch (Deutschland) <input type="checkbox"/> Deutsch (Liechtenstein) <input type="checkbox"/> Deutsch (Luxemburg) <input type="checkbox"/> Deutsch (Österreich) <input type="checkbox"/> Deutsch (Schweiz) <input type="checkbox"/> Englisch (Australien) <input type="checkbox"/> Englisch (Belize) <input checked="" type="checkbox"/> Englisch (Großbritannien) <input type="checkbox"/> Englisch (Irland) <input type="checkbox"/> Englisch (Jamaika) <input type="checkbox"/> Englisch (Kanada)	<input type="checkbox"/> Englisch (Karibik) <input type="checkbox"/> Englisch (Neuseeland) <input type="checkbox"/> Englisch (Philippinen) <input type="checkbox"/> Englisch (Simbabwe) <input type="checkbox"/> Englisch (Südafrika) <input type="checkbox"/> Englisch (Trinidad und Tobago) <input type="checkbox"/> Englisch (USA) <input type="checkbox"/> Estnisch (Estland) <input type="checkbox"/> Färingisch (Färöer) <input type="checkbox"/> Finnisch (Finnland) <input type="checkbox"/> Französisch (Belgien) <input checked="" type="checkbox"/> Französisch (Frankreich) <input type="checkbox"/> Französisch (Kanada) <input type="checkbox"/> Französisch (Luxemburg) <input type="checkbox"/> Französisch (Monaco) <input type="checkbox"/> Französisch (Schweiz) <input type="checkbox"/> Galizisch (Galizien) <input type="checkbox"/> Georgisch (Georgien) <input type="checkbox"/> Griechisch (Griechenland) <input type="checkbox"/> Hindi (Indien) <input type="checkbox"/> Indonesisch (Indonesien) <input type="checkbox"/> Isländisch (Island) <input checked="" type="checkbox"/> Italienisch (Italien) <input type="checkbox"/> Italienisch (Schweiz) <input type="checkbox"/> Japanisch (Japan)	<input type="checkbox"/> Kasachisch (Kasachstan) <input type="checkbox"/> Katalanisch (Katalonien) <input type="checkbox"/> Kirgisisch (Kirgisien) <input type="checkbox"/> Konkani (Indien) <input type="checkbox"/> Koreanisch (Korea) <input type="checkbox"/> Kroatisch (Kroatien) <input type="checkbox"/> Lettisch (Lettland) <input type="checkbox"/> Litauisch (Litauen) <input type="checkbox"/> Malaiisch (Brunei Darussalam) <input type="checkbox"/> Malaiisch (Malaysia) <input type="checkbox"/> Marathi (Indien) <input type="checkbox"/> Mazedonisch (Mazedonien, FYRM) <input type="checkbox"/> Mongolisch (Kyrillisch, Mongolei) <input type="checkbox"/> Niederländisch (Belgien) <input checked="" type="checkbox"/> Niederländisch (Niederlande) <input type="checkbox"/> Norwegisch, Bokmål (Norwegen) <input type="checkbox"/> Norwegisch, Nynorsk (Norwegen) <input type="checkbox"/> Polnisch (Polen) <input type="checkbox"/> Portugiesisch (Brasilien) <input type="checkbox"/> Portugiesisch (Portugal) <input type="checkbox"/> Rumänisch (Rumänien) <input checked="" type="checkbox"/> Russisch (Russland) <input type="checkbox"/> Sanskrit (Indien) <input type="checkbox"/> Schwedisch (Finnland) <input type="checkbox"/> Schwedisch (Schweden)
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Die Editiersprache gibt vor, in welcher Sprache im Editor projiziert wird.

Alle angewählten Sprachen werden im Projekt angelegt und auch in SiVArc generiert.

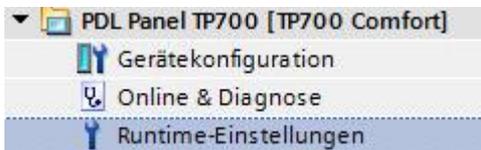
(2) In der CPU:

PDL CPU ->RMTaste-> Eigenschaften:



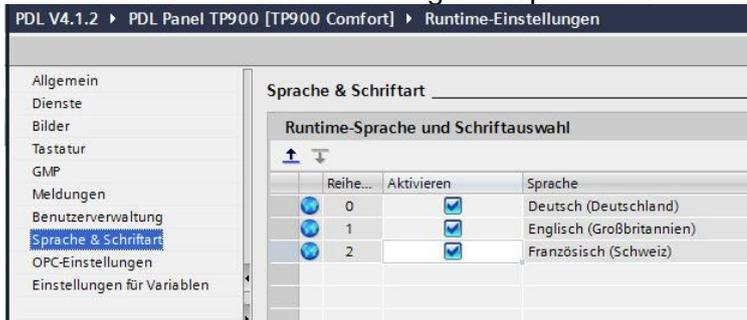
In der CPU muss unter Allgemein > Mehrsprachigkeit, die Zurodnung der Sprachen richtig projektiert sein

(3) Im Panel:



Um in der Visualisierung Projekte Mehrsprachig zu nutzen, müssen in den Runtimeeinstellungen die entsprechenden Sprachen aktiviert sein:

PDL Panel > Runtime-Einstellungen > Sprache&Schriftart:

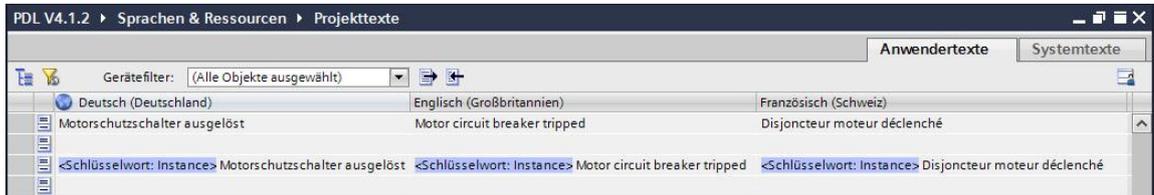


Aber Achtung: Die SiVArc Generierung funktioniert dann nur Fehlerfrei, wenn alle Relevanten Texte mehrsprachig, in allen gewählten Sprachen projektiert ist.

5.19.3 Projekttexte projektieren

Über Projekttexte können Projekttexte direkt in TIA projiziert werden:

Hier wird für jede Sprache der entsprechenden Text projiziert.



Entweder statisch, oder dynamisch über Schlüsselwörter:

Durch <Schlüsselwort: Instance> kann der Tagname (Name des Instanzdatenbausteins) dynamisch in den Projekttext projiziert werden. Es ist darauf zu achten, dass der Instanzdatenbaustein nur den TAG Namen und keine Sprachspezifische Bezeichnung enthält.

Die Projektierung der Texte direkt in TIA ist jedoch sehr einfach gehalten, daher bietet es sich an, die Projekttexte zu exportieren und in Excel zu übersetzen.

Siehe dazu folgende Punkte:

5.19.3.1 Projekttexte Exportieren

Projekttexte können in TIA V14 paarweise (z.B. De / En) exportiert werden um sie dann in Excel zu bearbeiten und dann wieder zu importieren.
z.B. wenn man als Referenzsprache Deutsch gewählt hat und alles in Englisch erweitern möchte, dann beim Exportieren Quellsprache Deutsch wählen und Zielsprache Englisch:

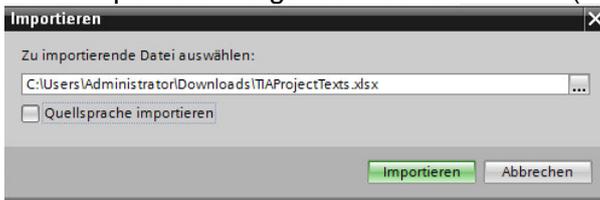


Ab TIA V15 können Projekttexte auch in mehreren Sprachen exportiert werden:



5.19.3.2 Projekttexte Importieren

Beim Importieren folgende Auswahl treffen (Quellsprache nicht anwählen):



Allgemeine Hinweise:

Projekttexte werden durch Übersetzen der Steuerung nicht überschrieben.
Projekttexte dienen bei SiVArC als Basis für Visualisierungstexte.

Bei den Projekttexten sind aber nur die Alarmtexte, Bausteinamen, Bausteinkommentare und Netzwerkkommentare projiziert.

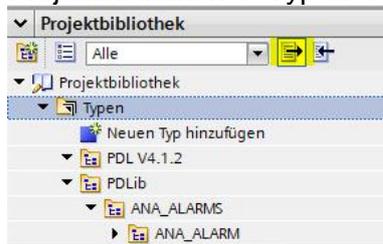
Allgemeine Texte von PopUps und HMI-Texte von Buttons sind wiederum Bibliothekstexte.

Übersicht: Projekttexte <> Bibliothekstexte

Category	Projekttexte	Bibliothekstexte
<Alarm>	Meldungen	
<BlockCommentCategory Data>	Bausteintitel, Bausteinkommentare, Bausteinanschlüsse, Netzwerktitel, Netzwerkkommentare (SiVArc),	
<HMI runtime>	HMI AllgemeineTexte, Textlisten	
<HMI comment>	HMI Kommentare	
<HMI screen>		HMI Texte, z.B. Buttons von Faceplates ...

Bibliothek Mehrsprachig :

Projektbibliothek -> Typen -> Bibliothekstexte Exportieren :



Alle Texte exportieren in allen benötigten Sprachen, dann in Excel die Spalten der Editiersprache (De) auf alle weiteren Sprachen kopieren.

Im zweiten Schritt können die Texte dann in Excel übersetzt werden.

Wichtig ist aber erstmal, dass alle Texte vorhanden sind, damit SiVARC keinen Fehler generiert.

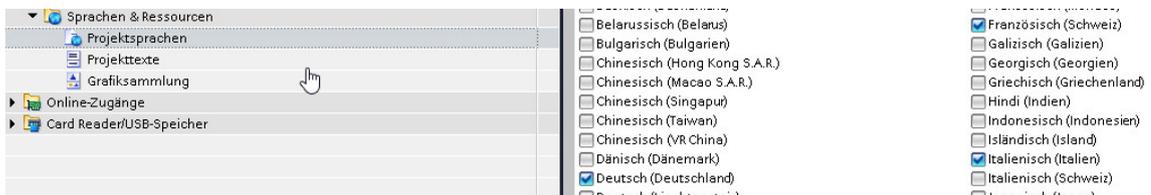
6 HMI PDL Generator SiVArc

6.1 Einleitung

Alle Devices können auch mit SiVArc generiert werden. Dafür ist eine SiVArc Installation und Lizenz notwendig.

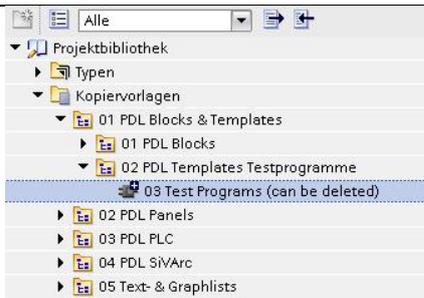
Der SiVArc generiert anhand des PLC Codes die Visualisierung inkl. Hierarchie und Devices. Ist in einem FB oder FC ein Device wie z. B. MOT_1D usw. vorhanden generiert er das Typical, PopUp und Faceplate automatisch. Die Bildbausteinschnittstelle wird mit den notwendigen Daten versorgt. Somit müssen die Devices nur noch auf dem Bild angeordnet werden. Daher ist es unerlässlich das Programm objektorientiert aufzubauen wie dies in Kapitel 3.1 beschrieben ist. Wenn im letzten Netzwerk jeweils ein UNIT Baustein verbaut wird, ist es möglich auch die Bildsprünge und Sammelhierarchie zu generieren.

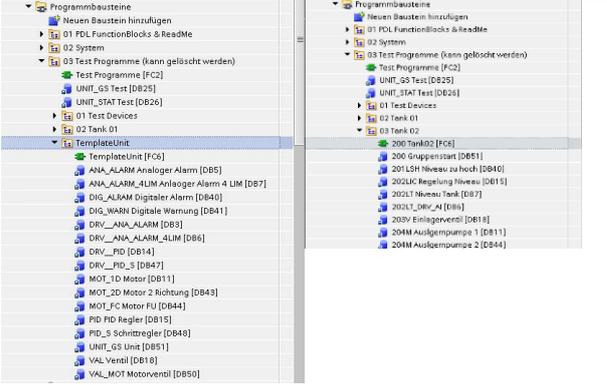
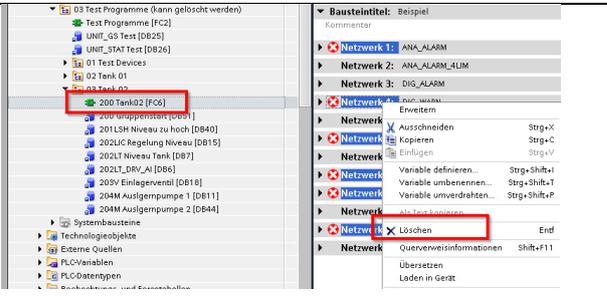
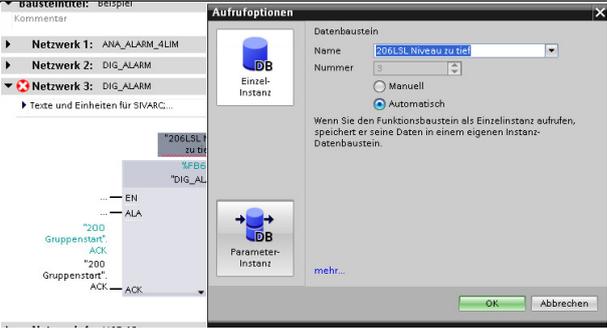
Achtung mit der Version 4.2 der PDL wurde eine Sprachumschaltung eingeführt. Daher müssen alle Netzwerke in der jeweiligen Sprache mit SiVArc Texten gepflegt werden. Wenn man dies nicht braucht kann man die Sprachen unter Sprachen und Ressourcen deaktivieren.



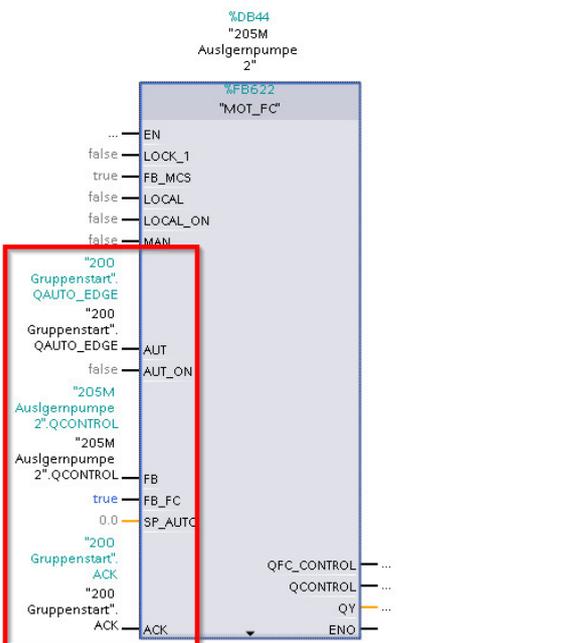
6.2 Arbeiten mit SiVArc

6.2.1 Kopiervorlage in Programm kopieren

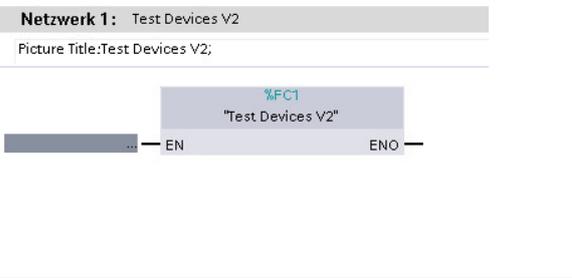
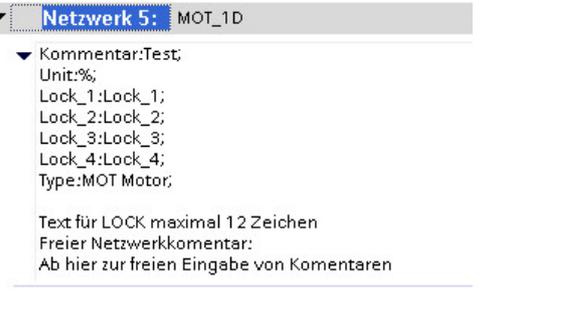
	Typ	Objekt
1	Die Projektbibliothek öffnen und unter Kopiervorlagen in das Programm ziehen.	

	Typ	Objekt
2	<p>Den Standard Namen umbenennen nach z. B Tankt 03. Zudem die jeweiligen Devices (Instanzbausteine Namen umbenennen) dabei muss folgendes Format eingehalten werden «Tagname + Leerschlag + Kommentar» z. B. «201LSH Niveau zu hoch»</p> <p>Devices die nicht gebraucht werden, können gelöscht werden.</p>	
3	<p>FC öffnen und nicht verwendete Devices löschen.</p>	
4	<p>Wenn mehr als eine Devices eines Typs im Programm enthalten sein soll wie z. B. neben einem LSH ein LSL kann das Netzwerk kopiert werden. Anschließend muss ein neuer Instanzdatenbaustein erstellwerten.</p> <p>Wenn möglich DB Nummer auf automatisch lassen!!</p> <p>Achtung: wenn es sich um ein Multilinguales Projekt handelt bitte nur den Tag als Instanzbaustein Namen verwenden.</p>	

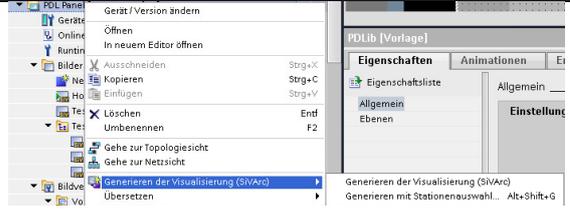
6.2.2 Automatiklogik Programmierung

	Typ	Objekt
1	<p>Jetzt kann die Automatik Logik programmiert werden. Grosser Vorteil ist, dass einige Standard Verschaltungen der Unit bereits gemacht sind, wie die Gruppenquittierung, die Devices auf Auto setzen usw.</p>	

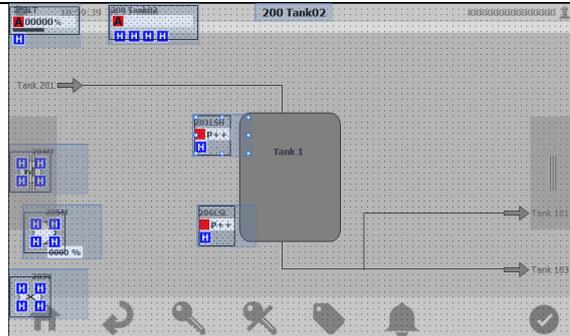
6.2.3 Texte für SiVArc eingeben

	Typ	Objekt
1	<p>Name des Bildes eingeben dort wo der FB / FC mit den PDL Instanzen aufgerufen wird!</p>	
2	<p>Jetzt können die SiVArc spezifischen Texte, Einheiten, Motor / Pumpentyp usw. eingegeben werden.</p>	

6.2.4 Übersetzten und Generieren

	Typ	Objekt
1	<ol style="list-style-type: none"> 1. Steuerung komplett übersetzten 2. Rechtsklick auf Panel und «Generieren der Visu (SiVarc)» 3. Panel komplett übersetzten 4. Panel und Steuerung Laden 	

6.2.5 Symbole anordnen

	Typ	Objekt
1	<p>Typical anordnen</p> <p>Fertig</p>	

7 HMI WinCC Classic V7.4

7.1 Einleitung

Hier wird das Vorgehen für das Einbinden von PDL-Projekten in das WinCC V7.4 beschrieben. Hierzu sind einige Regeln einzuhalten.

7.2 Anlegen des WinCC Projekts

7.2.1 Projekt entpackt

- Verzeichnis WinCCProj anlegen und freigeben
- Entpacken auf Server 01
- Öffnen und Rechnernamen umbenennen
- Serverdaten neu erzeugen
- Redundanz einstellen

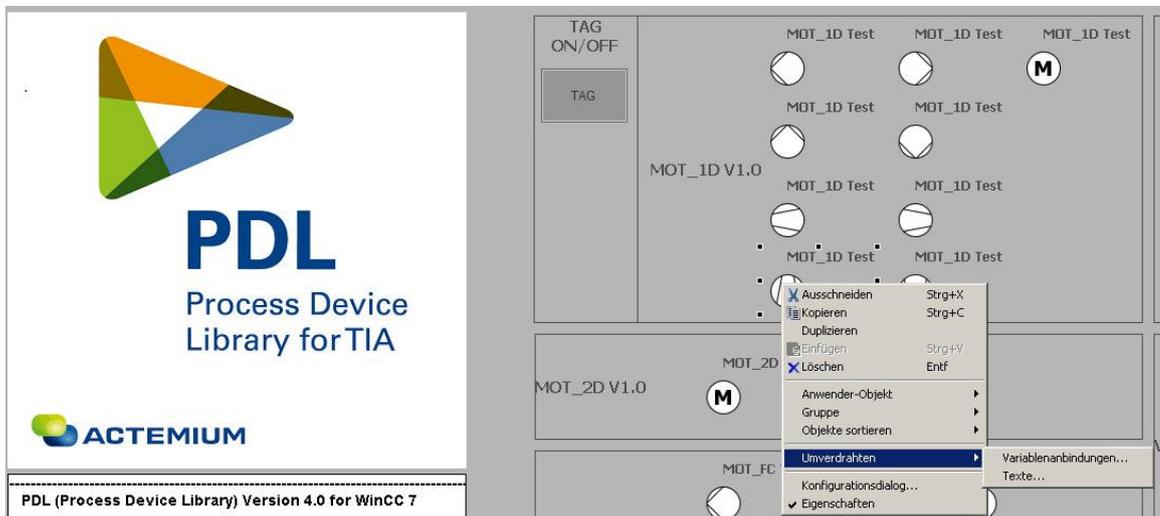
7.2.2 WinCC for PDL ins Projekt implementieren

Mit dem Projekteditor wird der Aufbau des Bildschirms definiert. Danach müssen für PDL manuelle Anpassungen durchgeführt werden.

- Zip «PDL for WinCC» dearchivieren
- Alle PDL Bilder vom GraCS Ordner in den GraCS Ordner des Projekts kopieren.
- In WinCC Explorer Text und Grafiklisten öffnen
 - Text markieren und unter «Bearbeiten» -> «Importieren» die «PDLTextList» importieren
 - Grafik markieren und unter «Bearbeiten» -> «Importieren» die «PDLGraphList» importieren
- Alle VBS Skripte in den Ordner «ScriptLib» kopieren
- Alle C Skripte in den «Library» Ordner kopieren und anschliessend im WinCC C-Editor den Header generieren.

7.2.3 Vorlagebild @PDLTypicals

Auf diesem Bild sind alle PDL Typical. Diese können einfach durch Um verdrahten Suchen und Ersetzen der «@@@» der Variablenanbindung und Texte verbunden werden.



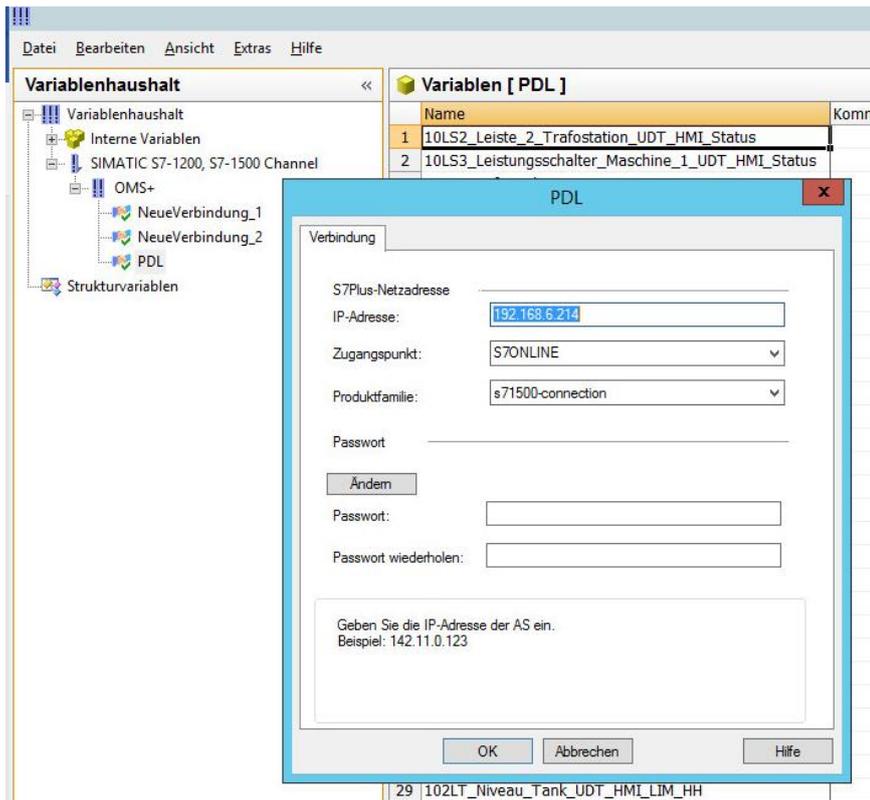
7.2.4 Benutzerverwaltung

In der PDL werden folgende Rechte verwendet. Detaillierte Informationen sind in der jeweiligen SDS des Devices.

Berechtigungen [Admin]			
	Funktion	Freigabe	Anlage1
1	Benutzerverwaltung	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
2	Freigabe für Bereich	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
3	Systemwechsel	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
4	Beobachten	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
5	Prozessbedienungen	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
6	Höherwertige Prozessbedienungen	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
7	Reportsystem	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
8	Archive bedienen	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
9	Service	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
10	Remote aktivieren	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
11	Remote projektieren	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
12	Web Zugriff - Nur beobachten	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

7.2.5 Neue Verbindung anlegen

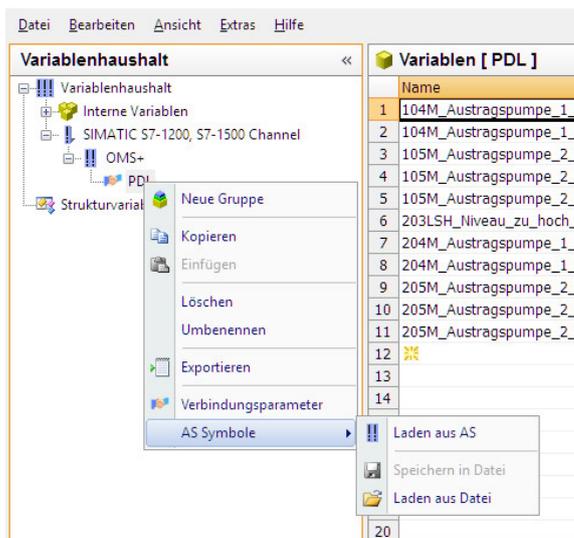
Für eine neue Verbindung gehen sie in den Variablenhaushalt auf OMS+ und fügen mit der rechten Maustaste eine neue Verbindung hinzu. Geben sie der neuen Verbindung die entsprechenden Parameter.



7.3 Anlegen eines neuen Typicals

7.3.1 Variablen einlesen

Variablenhaushalt öffnen. Die Runtime von WinCC muss laufen. Die entsprechende Verbindung zur AS anwählen und Variablen mit «Laden aus AS» einlesen.



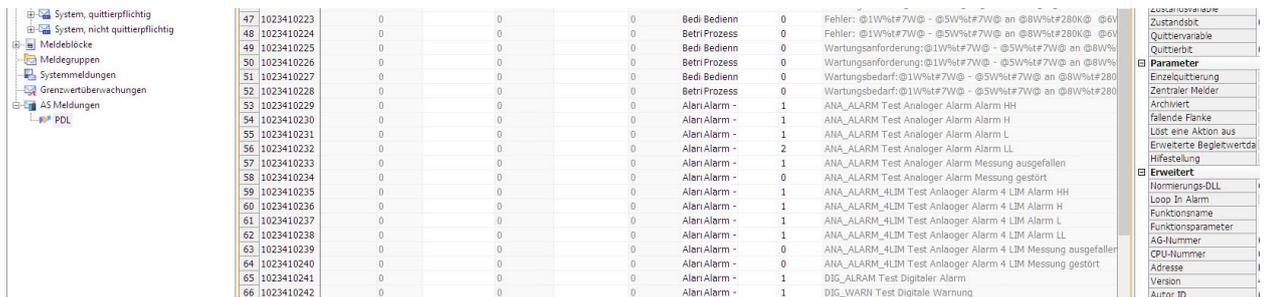
Laden aus AS (Runtime muss laufen)

403	<input type="checkbox"/>	UDT_HMI.Status	UInt	MOT_2D Test Motor 2 Richtung				
404	<input type="checkbox"/>	UDT_HMI.OP_HRS	Int	MOT_2D Test Motor 2 Richtung				
405	<input checked="" type="checkbox"/>	UDT_HMI.Status	UInt	MOT_FC Test Motor FU	MOT_FC_Test_Motor_FU_UDT_HMI_Status	Vorzeichenloser 32-Bit Wert	4	
406	<input checked="" type="checkbox"/>	UDT_HMI.OP_HRS	Int	MOT_FC Test Motor FU	MOT_FC_Test_Motor_FU_UDT_HMI.OP_HR	Vorzeichenbehafteter 32-Bit Wert	4	
407	<input checked="" type="checkbox"/>	UDT_HMI.Y	Real	MOT_FC Test Motor FU	MOT_FC_Test_Motor_FU_UDT_HMI.Y	Gleitkommazahl 32-Bit IEEE 754	4	
408	<input type="checkbox"/>	UDT_HMI.Status	UInt	BUTTON Test Schalter				
409	<input type="checkbox"/>	UDT_HMI.Status	UInt	BUTTON Test 1 aus 3				
410	<input type="checkbox"/>	UDT_HMI.P	Real	PID_S Test Schrittreger				
411	<input type="checkbox"/>	UDT_HMI.I	Real	PID_S Test Schrittreger				
412	<input type="checkbox"/>	UDT_HMI.D	Real	PID_S Test Schrittreger				

Variablen, welche importiert werden sollen auswählen.

7.3.2 Alarme einlesen

- Alarm Logging öffnen

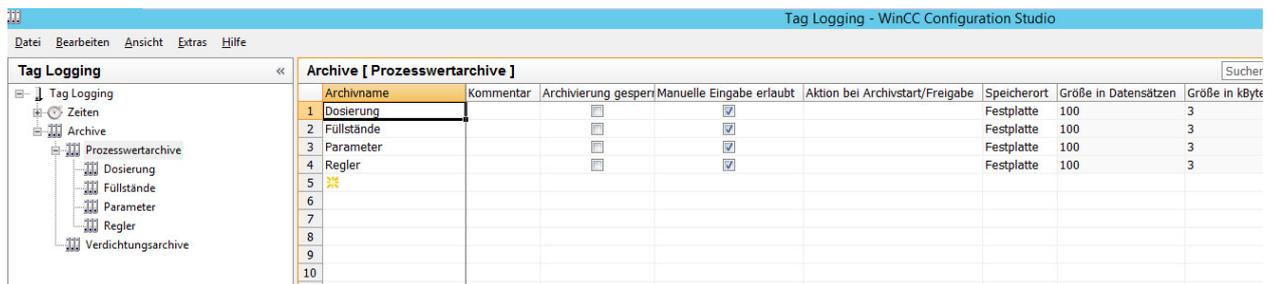


Objekt	Objektname	Objekttyp	Objektbeschreibung	Objektstatus	Objektwert	Objektbeschreibung	Objektwert	Objektbeschreibung	Objektwert
47	1023410223	0	0	0	0	Bedi Bedienn	0	Fehler: @1W%#7W@ - @5W%#7W@ an @8W%#280K@ @6W%#280K@	
48	1023410224	0	0	0	0	Betri Prozess	0	Fehler: @1W%#7W@ - @5W%#7W@ an @8W%#280K@ @6W%#280K@	
49	1023410225	0	0	0	0	Bedi Bedienn	0	Wartungsanforderung:@1W%#7W@ - @5W%#7W@ an @8W%#280K@ @6W%#280K@	
50	1023410226	0	0	0	0	Betri Prozess	0	Wartungsanforderung:@1W%#7W@ - @5W%#7W@ an @8W%#280K@ @6W%#280K@	
51	1023410227	0	0	0	0	Bedi Bedienn	0	Wartungsbedarf:@1W%#7W@ - @5W%#7W@ an @8W%#280K@ @6W%#280K@	
52	1023410228	0	0	0	0	Betri Prozess	0	Wartungsbedarf:@1W%#7W@ - @5W%#7W@ an @8W%#280K@ @6W%#280K@	
53	1023410229	0	0	0	0	Alarm Alarm	1	ANA_ALARM Test Analoger Alarm Alarm HH	
54	1023410230	0	0	0	0	Alarm Alarm	1	ANA_ALARM Test Analoger Alarm Alarm H	
55	1023410231	0	0	0	0	Alarm Alarm	1	ANA_ALARM Test Analoger Alarm Alarm L	
56	1023410232	0	0	0	0	Alarm Alarm	2	ANA_ALARM Test Analoger Alarm Alarm LL	
57	1023410233	0	0	0	0	Alarm Alarm	1	ANA_ALARM Test Analoger Alarm Messung ausgefallen	
58	1023410234	0	0	0	0	Alarm Alarm	0	ANA_ALARM Test Analoger Alarm Messung gestört	
59	1023410235	0	0	0	0	Alarm Alarm	1	ANA_ALARM_4LM Test Anlaeger Alarm 4 LIM Alarm HH	
60	1023410236	0	0	0	0	Alarm Alarm	1	ANA_ALARM_4LM Test Anlaeger Alarm 4 LIM Alarm H	
61	1023410237	0	0	0	0	Alarm Alarm	1	ANA_ALARM_4LM Test Anlaeger Alarm 4 LIM Alarm L	
62	1023410238	0	0	0	0	Alarm Alarm	1	ANA_ALARM_4LM Test Anlaeger Alarm 4 LIM Alarm LL	
63	1023410239	0	0	0	0	Alarm Alarm	0	ANA_ALARM_4LM Test Anlaeger Alarm 4 LIM Messung ausgefallen	
64	1023410240	0	0	0	0	Alarm Alarm	0	ANA_ALARM_4LM Test Anlaeger Alarm 4 LIM Messung gestört	
65	1023410241	0	0	0	0	Alarm Alarm	1	DIG_ALARM Test Digitaler Alarm	
66	1023410242	0	0	0	0	Alarm Alarm	1	DIG_WARN Test Digitale Warnung	

- Alarme, welche importiert werden sollen auswählen.
- Meldeklassen anpassen
- Falls es eine Verfahrensweisung für eine Meldung gibt, diese aktivieren.
- Alarm markieren

7.3.3 TAG-Logging

Sind Variablen vorhanden die geloggt werden sollen müssen diese im TAG-Logging aufgenommen werden.



Archivname	Kommentar	Archivierung gesperrt	Manuelle Eingabe erlaubt	Aktion bei Archivstart/Freigabe	Speicherort	Größe in Datensätzen	Größe in kByte
1	Dosierung	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		Festplatte	100	3
2	Füllstände	<input type="checkbox"/>	<input checked="" type="checkbox"/>		Festplatte	100	3
3	Parameter	<input type="checkbox"/>	<input checked="" type="checkbox"/>		Festplatte	100	3
4	Regler	<input type="checkbox"/>	<input checked="" type="checkbox"/>		Festplatte	100	3
5		<input type="checkbox"/>	<input type="checkbox"/>				
6		<input type="checkbox"/>	<input type="checkbox"/>				
7		<input type="checkbox"/>	<input type="checkbox"/>				
8		<input type="checkbox"/>	<input type="checkbox"/>				
9		<input type="checkbox"/>	<input type="checkbox"/>				
10		<input type="checkbox"/>	<input type="checkbox"/>				
...		<input type="checkbox"/>	<input type="checkbox"/>				

7.3.4 Typical anlegen

Dies ist in der jeweiligen SDS der Devices erklärt.

7.4 Projekt duplizieren

So duplizieren Sie ein redundantes Projekt im laufenden Betrieb

Wenn Sie ein redundantes Projekt bearbeiten, dann können Sie das Projekt auch im laufenden Betrieb auf dem redundanten Server aktualisieren.

Kleinere Änderungen können Sie mit der Funktion Online-Änderungsladen speichern und im laufenden Betrieb auf die Server übertragen. Beachten Sie dazu die Dokumentation zum Thema "Online-Änderungsladen".

Duplizieren mit dem Project Duplicator

Einige Projektierungen können nicht über Online-Änderungsladen gespeichert werden. In diesem Fall müssen Sie das Projekt mit dem Project Duplicator auf den redundanten Server duplizieren.

Achtung: Keine Redundanz

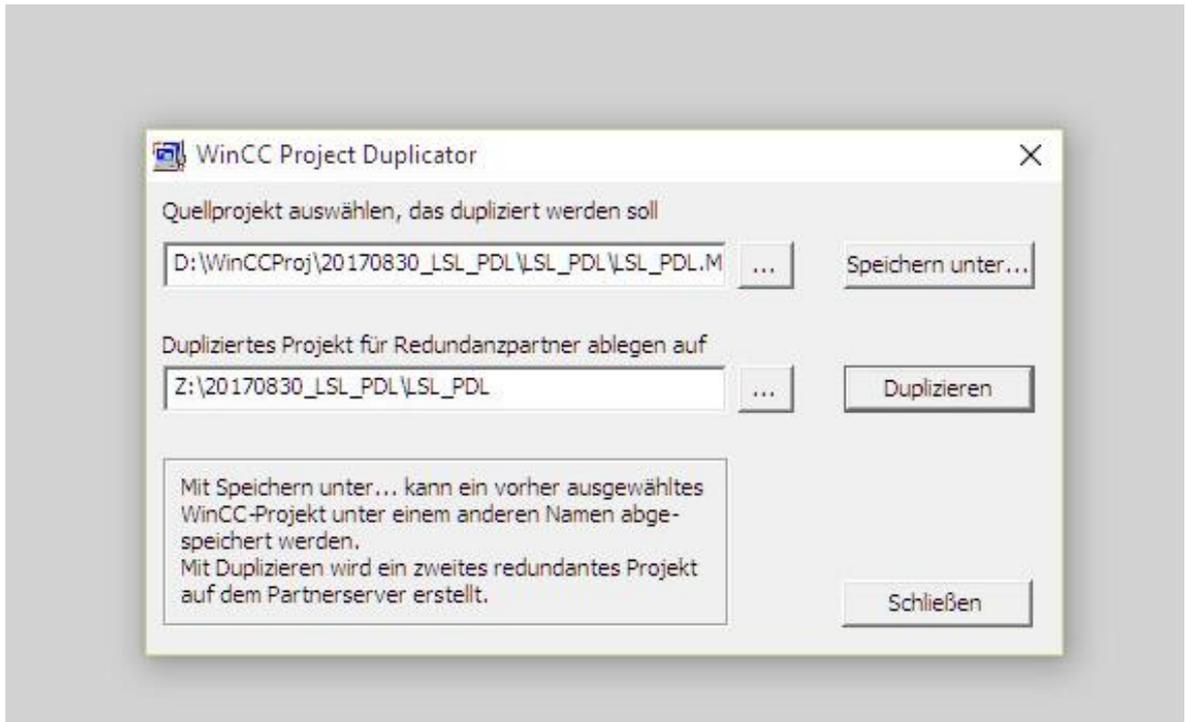
Bei Änderungen im laufenden Betrieb müssen sie einen der Partner-Server deaktivieren. In dieser Zeit steht keine Redundanz zur Verfügung.

Voraussetzungen

- Der Zielordner ist angelegt.
- Sie haben Zugriffsrechte auf den Zielordner.
- Der redundante Server, auf dem das kopierte Projekt abgelegt wird, verfügt über genügend Speicherplatz.

Vorgehensweise:

1. Beenden Sie die Runtime auf dem redundanten Server SE02 und schließen Sie das Projekt. SQL-Datenbank muss auch geschlossen sein. Ansonsten Reset WinCC ausführen. SQL Datenbank kann im SQL Manager geprüft werden.
2. Führen Sie die Projektierungsänderungen an Server SE01 in Runtime aus und speichern Sie die Änderungen.
3. Hier spätestens Runtime auf SE02 stoppen. Wenn erst hier gestoppt wird könnten an den Clients das gleiche Bild unterschiedlich angezeigt werden, da CL01 auf SE01 und CL02 auf SE02 schaut.
4. Starten Sie den Project Duplicator auf Server ES01.
5. Im ersten Feld das Projekt mit der MCP-Datei auswählen
6. Im zweiten Feld WinCCProj auf dem Server SE02 über Netzwerkverbindungen auswählen. Netzwerk -> Rechnername -> WinCCProj
7. Duplizieren ausführen und warten bis die Rückmeldung erscheint.



Projekte das erste Mal auf Server aufspielen:

1. Projektverzeichnis muss auf beiden Server angelegt und freigegeben sein.
2. Projekt auf Server SE01 entpacken und öffnen.
3. Serverpackage erzeugen und Redundanz einstellen.
4. WinCC starten und Funktionen testen
5. Projekt mit Project Duplicator auf Server SE01 duplizieren.

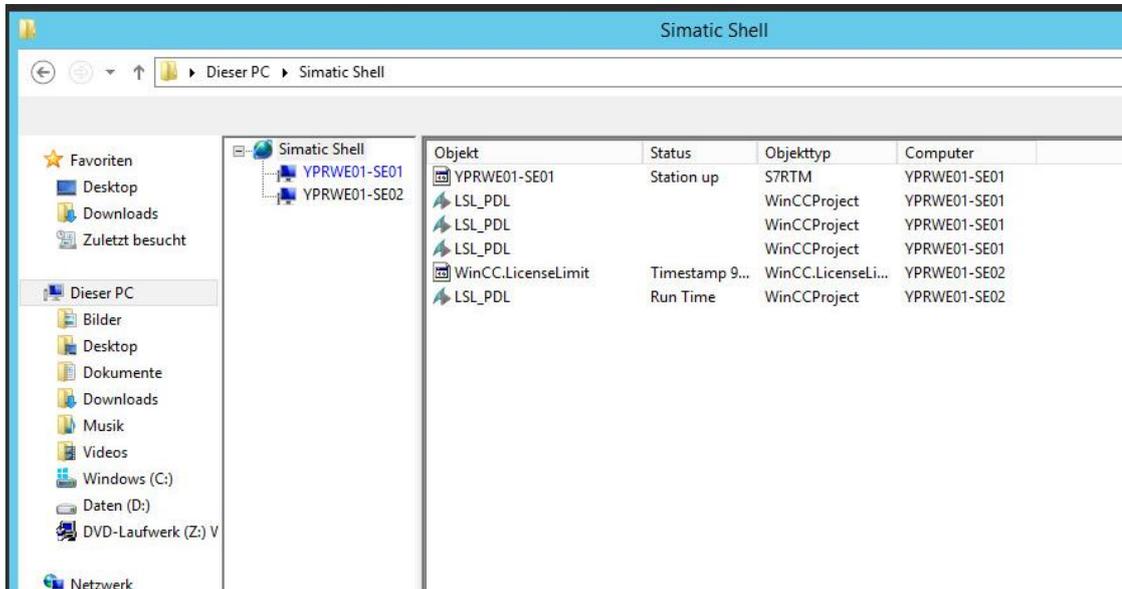
7.5 Updaten

Neue Typicals

- Vorlage öffnen, Text- und Graphikliste aus Vorlage exportieren.
- Im Windows Explorer:
 - Aus GraCS Ordner PDL kopieren
 - Alle @PG_ Bilder kopieren
 - @PDLTypcial Typicalvorlage kopieren
 - Skripte C Alle C-Skripte in Ordner Library kopieren
 - Skripte VB alle VB-Skripte in Ordner ScriptLib kopieren
- WinCCExplorer mit Projekt öffnen:
 - C-Scripte Header neu generieren
 - Exportierte Text- und Graphikliste aus Vorlage importieren
- Neu starten

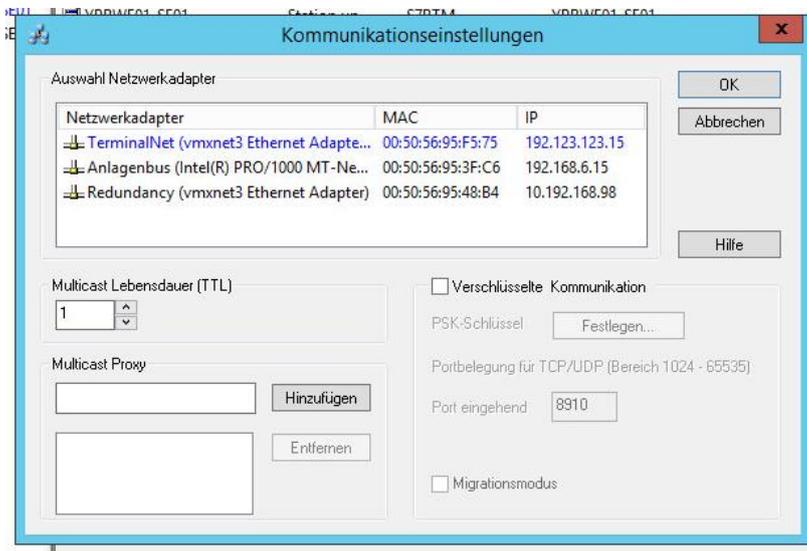
7.6 Zusatzinfos

Projekt duplizieren funktioniert gar nicht



Server müssen im Simatic Shell sichtbar sein

Simatic Shell -> rechte Mouse -> Einstellungen, ins Feld mit den Netzwerkadaptern klicken. Der Terminalbus muss blau sein. Ansonsten einstellen.



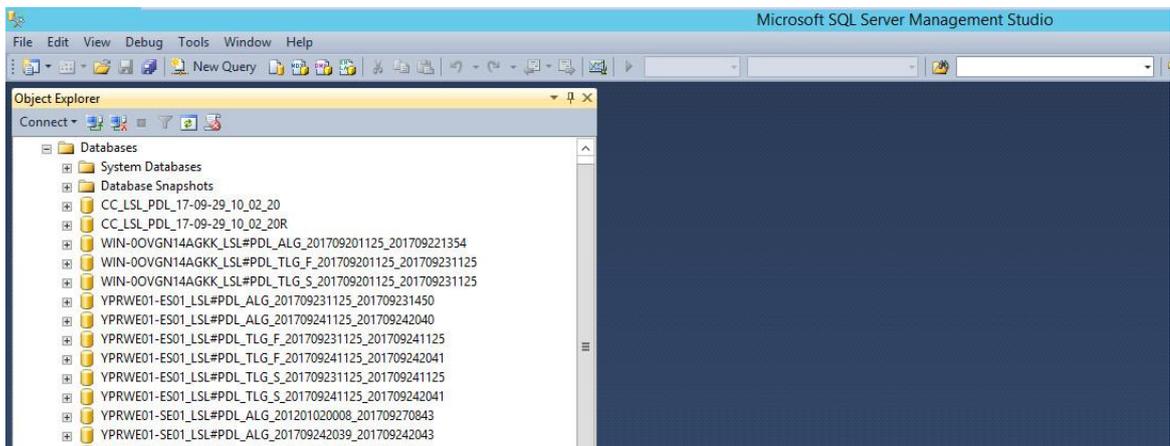
Unter Programme/Siemens/Security Manager ausführen (vorher alle Programme schliessen)

Rechner werden im Netzwerk nicht angezeigt, Netzwerkprobleme, Probleme beim Duplizieren:



- Einstellungen wiederholen.
- Balken muss durchlaufen
- Rechner neu starten

SQLManagement Server:



Wird WinCC Runtime beendet sollten sich die Datenbanken CC_»Projektname...»

Schliessen und nicht mehr sichtbar sein. Ansonsten ist das Projekt noch geöffnet und kann mit WinCC Reset beendet werden. Vorsicht es werden aber alle Siemens Applikationen beendet auch Simatic NET. Daher ist es gegebenenfalls notwendig den Rechner neu zu starten.