



SIEMENS



SCE Lehrunterlagen

Siemens Automation Cooperates with Education (SCE) | 09/2015

PA Modul P01-06
SIMATIC PCS 7 –
Regelung und weitere Steuerfunktionen

Cooperates
with Education

Automation

SIEMENS

Passende SCE Trainer Pakete zu diesen Lehrunterlagen

- **SIMATIC PCS 7 Software 3er Paket**
BestellNr.: 6ES7650-0XX18-0YS5
- **SIMATIC PCS 7 Software 6er Paket**
BestellNr.: 6ES7650-0XX18-2YS5
- **SIMATIC PCS 7 Software Upgrade Pakete 3er**
BestellNr.: 6ES7650-0XX18-0YE5 (V8.0 → V8.1) bzw. 6ES7650-0XX08-0YE5 (V7.1 → V8.0)
- **SIMATIC PCS 7 Hardware Set inkl. RTX-Box**
BestellNr.: 6ES7654-0UE13-0XS0

Bitte beachten Sie, dass diese Trainer Pakete ggf. durch Nachfolge-Pakete ersetzt werden.
Eine Übersicht über die aktuell verfügbaren SCE Pakete finden Sie unter: [siemens.de/sce/tp](https://www.siemens.de/sce/tp)

Fortbildungen

Für regionale Siemens SCE Fortbildungen kontaktieren Sie Ihren regionalen SCE Kontaktpartner
[siemens.de/sce/contact](https://www.siemens.de/sce/contact)

Weiterführende Informationen zu SIMATIC PCS 7 und SIMIT

Insbesondere Getting started, Videos, Tutorials, Handbücher und Programmierleitfaden.
[siemens.de/sce/pcs7](https://www.siemens.de/sce/pcs7)

Weitere Informationen rund um SCE

[siemens.de/sce](https://www.siemens.de/sce)

Verwendungshinweis

Die SCE Lehrunterlage für die durchgängige Automatisierungslösung Totally Integrated Automation (TIA) wurde für das Programm „Siemens Automation Cooperates with Education (SCE)“ speziell zu Ausbildungszwecken für öffentliche Bildungs- und F&E-Einrichtungen erstellt. Die Siemens AG übernimmt bezüglich des Inhalts keine Gewähr.

Diese Unterlage darf nur für die Erstausbildung an Siemens Produkten/Systemen verwendet werden. D.h. sie kann ganz oder teilweise kopiert und an die Auszubildenden zur Nutzung im Rahmen deren Ausbildung ausgehändigt werden. Weitergabe sowie Vervielfältigung dieser Unterlage und Mitteilung ihres Inhalts ist innerhalb öffentlicher Aus- und Weiterbildungsstätten für Zwecke der Ausbildung gestattet.

Ausnahmen bedürfen der schriftlichen Genehmigung durch die Siemens AG. Ansprechpartner: Herr Roland Scheuerer roland.scheuerer@siemens.com.

Zu widerhandlungen verpflichten zu Schadensersatz. Alle Rechte auch der Übersetzung sind vorbehalten, insbesondere für den Fall der Patentierung oder GM-Eintragung.

Der Einsatz für Industriekunden-Kurse ist explizit nicht erlaubt. Einer kommerziellen Nutzung der Unterlagen stimmen wir nicht zu.

Wir danken der TU Dresden, besonders Prof. Dr.-Ing. Leon Urbas und Dipl.-Ing. Annett Krause, der Fa. Michael Dziallas Engineering und allen weiteren Beteiligten für die Unterstützung bei der Erstellung dieser SCE Lehrunterlage.

REGELUNG UND WEITERE STEUERFUNKTIONEN

LERNZIEL

In diesem Kapitel lernen die Studierenden wesentliche Komponenten und Anforderungen an einen Baustein zur kontinuierlichen Regelung von Prozessgrößen kennen und können eine Temperaturregelung mit den Bausteinen PIDConL und PULSEGEN anlegen und konfigurieren.

THEORIE IN KÜRZE

In der Prozessindustrie müssen bestimmte Prozessgrößen trotz Störungen auf einem bestimmten Wert gehalten (**Störverhalten**) bzw. Prozessgrößen stabil auf vorgegebene Sollwerte eingestellt (**Führungsverhalten**) werden. Dafür werden Regelkreise wie in Abbildung 1 verwendet.

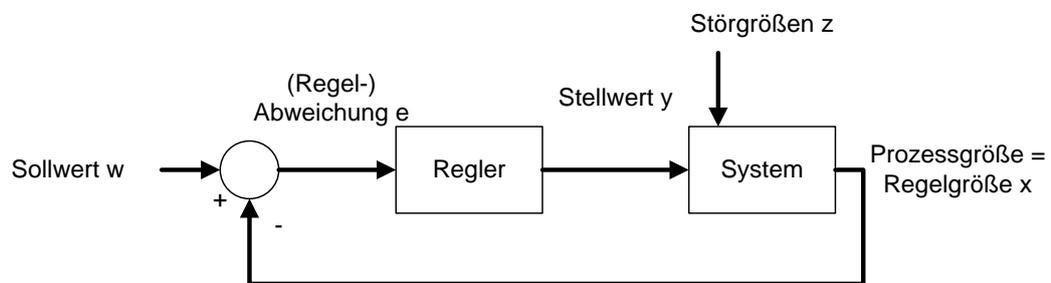


Abbildung 1: Regelkreis

Im Falle der Anlage in den Lehrunterlagen muss die Reaktortemperatur für eine spezifikationsgerechte Reaktionsführung auf einen bestimmten Wert eingestellt werden. Störgrößen sind die Umgebungstemperatur und die Einsatzstoffe mit unterschiedlichen Temperaturen. Damit die Temperatur geregelt werden kann, muss sie zunächst durch Messung bestimmt werden. Dieser Messwert, der dem **Istwert** der Prozessgröße entspricht, wird anschließend mit dem gewünschten Wert (**Sollwert**) verglichen. Der Unterschied zwischen Istwert und Sollwert wird (**Regel-)Abweichung** genannt.

Bei bekannter Regelabweichung können Gegenmaßnahmen abgeleitet werden. Bei der Temperaturregelung wird die Heizung eingeschaltet, wenn der gemessene Istwert kleiner ist als der vorgegebene Sollwert. Damit der Prozess selbsttätig reagiert, wird ein Regler benötigt. Ein Regler, der nur anhand der aktuellen Abweichung den Stellwert berechnet, wird Proportionalregler (kurz: P-Regler) genannt.

In der Praxis haben sich Regler durchgesetzt, die mit Hilfe weniger Parameter für eine große Bandbreite von Prozessen eingesetzt werden können, sogenannte **PID-Regler**.

In der **PCS 7 Advanced Process Library V80** gibt es bewährte Bausteine, die diese Funktionalität implementieren. Im Folgenden wird der Baustein PIDConL verwendet.

THEORIE

EINFÜHRUNG

Der oben erwähnte P-Regler stellt den einfachsten Regler dar. Er arbeitet nach dem Prinzip: Je größer die aktuelle Abweichung, desto größer wird der Stellwert. Sein Verhalten leitet sich also direkt aus der aktuellen Regelabweichung ab, was ihn schnell und dynamisch relativ günstig macht. Allerdings werden bestimmte Störungen nicht vollständig ausgeregelt, das heißt es gibt immer eine bleibende Regelabweichung.

Nicht jeder Prozess toleriert eine bleibende Regelabweichung, sodass weitere Maßnahmen getroffen werden müssen. Eine Möglichkeit besteht darin einen integralen Anteil zuzuschalten, wodurch der P-Regler zu einem PI-Regler wird. Die Wirkung des integralen Anteils besteht darin, dass eine anhaltende Regelabweichung aufsummiert wird. Damit wird der Stellwert trotz gleichbleibender Regelabweichung immer größer.

Treten in einem System sprungförmige Störungen auf, so kann diesen mit einem zusätzlich differenzierenden Anteil schnell entgegengesteuert werden. Der D-Anteil berechnet die Stellgröße aus der zeitlichen Ableitung der Regeldifferenz. Dieses Verhalten führt aber auch zu einer Verstärkung von stochastischen Störungen (Rauschen). Hier muss ein sinnvoller Mittelweg gefunden werden.

Eine Kombination aus P, I und D-Anteil wird PID-Regler genannt. In der Prozessindustrie werden 95 % der Anwendungen mit diesen Reglern realisiert, da der PID-Regler mit nur drei Parametern (Verstärkung, Nachstellzeit und Vorhaltezeit) eingestellt wird. Diese wenigen Parameter ermöglichen bereits eine gute Anpassung an eine Vielzahl unterschiedlicher dynamischer Prozesse.

Das Einstellen der Parameter setzt allerdings Kenntnisse über das zu regelnde System voraus. Die Kenntnisse über das System können aus Erfahrung gewonnen, experimentell bestimmt oder durch die Modellierung des Prozesses berechnet werden. Für eine große Bandbreite von Prozessen, die nicht von Totzeiten dominiert werden und in ähnlicher Art und Weise auf positive wie negative Veränderungen der Stellgrößeneingriffe reagieren, konnten verschiedene praxistaugliche Einstellregeln gefunden werden. Als Beispiele seien die Einstellregeln nach Chien, Hrones und Reswick [1], die Methode von Ziegler und Nichols [2] sowie die T-Summen-Regel [3] genannt.

Das Prozessleitsystem **PCS 7** unterstützt das Einstellen der Parameter mit Hilfe eines **PID-Tuners**.

Bei dem Reglerbaustein PIDConL heißt der Parameter für die Verstärkung GAIN, für den Integralanteil TI (Nachstellzeit) und für den Differentialanteil TD (Vorhaltezeit). Die Zeiten sind jeweils in Sekunden anzugeben. Die Eingangsgröße des Reglers sind die Regelgröße PV und der Sollwert SP, welche die Regelabweichung ER ergeben. Der Stellwert MV ist die Ausgangsgröße zur Regelstrecke, welche sich nach der folgenden Formel berechnet:

$$MV = GAIN \cdot \left(1 + \frac{1}{TI \cdot s} + \frac{TD \cdot s}{1 + \frac{TD}{DiffGain} \cdot s} \right) \cdot ER.$$

INDUSTRIETAUGLICHKEIT VON REGLERN

Damit ein Regler auch im industriellen Alltag funktioniert, müssen weitere Funktionen implementiert sein. Dazu gehören vor allem:

- Stoßfreie Umschaltung
- Anti-Reset-Windup
- Unterstützung von verschiedenen Regelstrukturen.

Die stoßfreie Umschaltung soll eine abrupte Änderung der Stellgröße bei der Umschaltung zwischen Hand- und Automatikbetrieb, zwischen interner und externer Sollwertvorgabe oder bei Parameteränderung verhindern. Eine stoßfreie Umschaltung zwischen Hand- und Automatikbetrieb wird zum Beispiel gefordert, wenn ein Prozess in der Verfahrenstechnik halbautomatisch abläuft, wenn also das Anfahren von Hand durchgeführt wird und anschließend im regulären Betrieb auf Automatik umgeschaltet wird. Im Handbetrieb wird die Stellgröße direkt vom Operator vorgegeben, während im Automatikbetrieb die Stellgröße vom Regelalgorithmus berechnet wird.

Die Funktion Anti-Reset-Windup (ARW) soll verhindern, dass sich der integrale Anteil (engl. reset) der Stellgröße immer weiter erhöht (bildlich: aufwickelt, engl. windup), weil eine Regelabweichung zum Beispiel aufgrund der Stellgrößenbeschränkung nicht ausgeregelt werden kann.

Die Unterstützung verschiedener Regelstrukturen ermöglicht die Optimierung der Regelung ohne den Regler austauschen zu müssen. Im Abschnitt ‚Erweiterte Regelstrukturen‘ werden einige dieser Regelstrukturen genauer erklärt. Mit PIDConL aus der **PCS 7 Advanced Process Library V80** lassen sich folgende Regelstrukturen realisieren:

- Festwertregelung
- Kaskadenregelung
- Verhältnisregelung
- Störgrößenaufschaltung
- Split-Range Regelung
- Smith-Prädiktor Regelung und
- Ablösende Regelung (Override).

ERWEITERTE REGELSTRUKTUREN

In verschiedenen Anwendungen reichen einschleifige Regelkreise nicht aus, sodass erweiterte Regelstrukturen einzusetzen sind, um das gewünschte Ziel zu erreichen.

Wird bei einer Prozessgröße das Führungs- und das Störverhalten nicht gleichzeitig zufriedenstellend optimiert, kann eine Stör-/Hilfsgrößenaufschaltung oder eine Kaskadenregelung eingesetzt werden.

Wenn die Störgröße gemessen wird und ihr Angriffspunkt bekannt ist, kann eine Kompensation der Störgröße auf den Reglereingang oder -ausgang aufgeschaltet werden. Mit der **Störgrößenaufschaltung** kann man die Störgröße vollständig kompensieren, sodass der Regler auf optimales Führungsverhalten eingestellt werden kann.

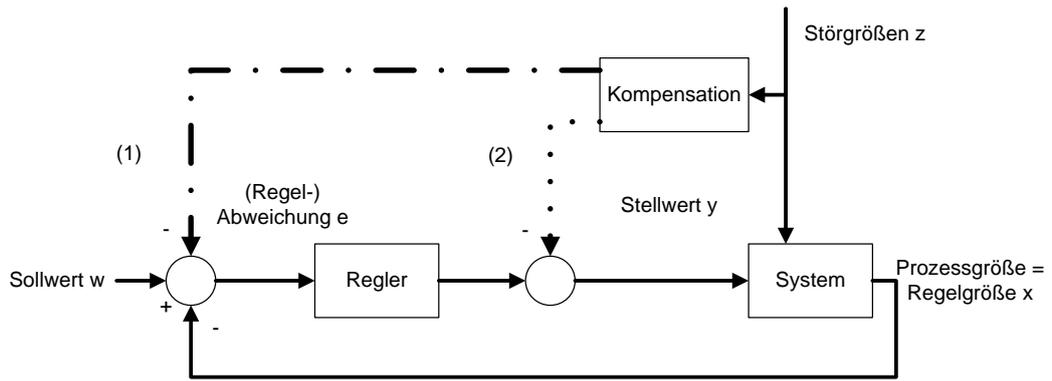


Abbildung 2: Störgrößenaufschaltung am Reglereingang (1) oder am Reglerausgang (2)

Kann die Störgröße nicht gemessen werden, dafür aber eine andere Größe im System, so wird diese Hilfsgröße mit einem Regler auf den Reglereingang geschaltet. Die **Hilfsgrößenaufschaltung** reduziert den Einfluss der Störgröße, kompensiert ihn jedoch nicht vollständig.

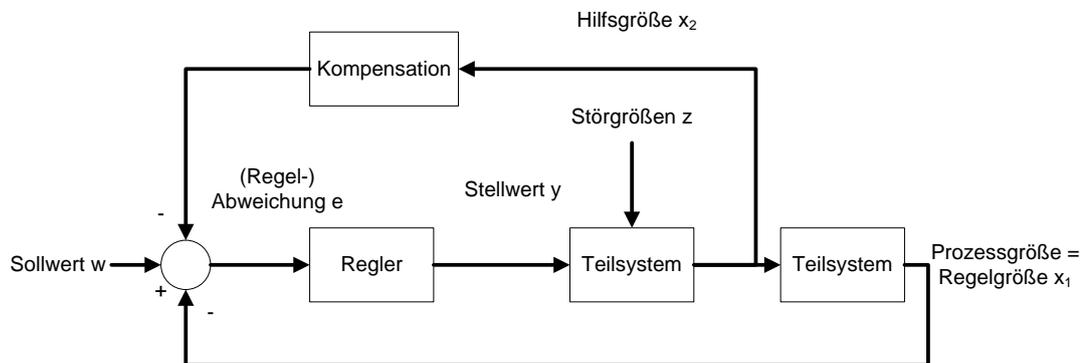


Abbildung 3: Hilfsgrößenaufschaltung

Erfolgt die Aufschaltung am Reglereingang, so sind die Kompensation und der Regler nicht unabhängig voneinander. Das bedeutet, dass bei einer Anpassung der Reglerparameter auch die Kompensation angepasst werden muss.

Reichen die Stör- und Hilfsgrößenaufschaltung nicht aus, kann der Angriffspunkt der Störgrößen nicht hinreichend genau bestimmt oder können die Teilsysteme nicht hinreichend genau modelliert werden, so wird eine zwei- oder mehrschleifige **Kaskadenregelung** eingesetzt.

Beim Entwurf der Kaskadenregelung wird davon ausgegangen, dass die unterlagerten Regelkreise (Regler 2 in Abbildung 4 – ein sogenannter Folgeregler) jeweils schneller reagieren als die überlagerten Regelkreise (Regler 1 in Abbildung 4 – ein sogenannter Führungsregler). Die Optimierung der Regelung erfolgt damit stets von innen nach außen.

Die Kaskadenregelung reduziert den Einfluss der Störgröße und macht die Regelung der Führungsgröße schneller. Damit die Kaskadenregelung eingesetzt werden kann, müssen entsprechend messbare Größen vorhanden sein.

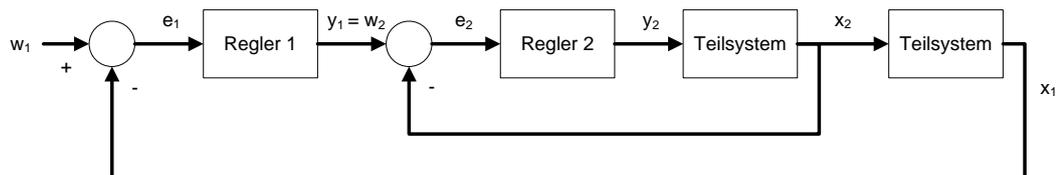


Abbildung 4: Kaskadenregelung mit zwei Schleifen

Die Verhältnisregelung wird eingesetzt, wenn die Prozessgröße in Abhängigkeit einer anderen Größe bestimmt wird, zum Beispiel die Verhältnisregelung zweier zu vermischender Flüssigkeitsströme, das heißt Regelung der Zusammensetzung der Mischung oder die Verhältnisregelung von Brenngas und Frischluft an einem Gasbrenner für optimale Verbrennung. Der Sollwert der Prozessgröße w_2 wird aus dem Verhältnis V_w und der Prozessgröße x_1 berechnet.

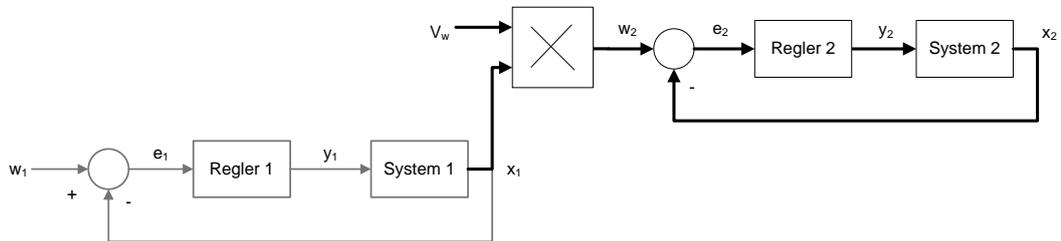


Abbildung 5: Verhältnisregelung

ANSCHALTUNG AN PROZESSE

Das kontinuierliche Ausgangssignal des Reglers wird nicht immer direkt an den Prozess ausgegeben. Besonders bei großen Kräften oder Strömen ist das nicht sinnvoll, sodass eine binäre Ansteuerung erfolgt. Dazu wird das analoge Signal über die **Impulsbreitenmodulation** (auch Pulsbreiten- oder Pulsweitenmodulation genannt) in ein binäres Signal gewandelt. In der **CFC Library** gibt es dafür den Elementarbaustein PULSEGEN [4].

Die Funktion PULSEGEN transformiert die Eingangsgröße INV (= LMN Stellwert des PID-Reglers) durch Modulation der Impulsbreite in eine Impulsfolge mit konstanter Periodendauer. Sie entspricht der Zykluszeit, mit der die Eingangsgröße aktualisiert wird und muss in PER_TM parametriert werden.

Die Dauer eines Impulses pro Periodendauer ist proportional zur Eingangsgröße. Dabei ist der durch PER_TM parametrierte Zyklus nicht identisch mit dem Bearbeitungszyklus des Funktionsbausteins PULSEGEN. Wie in Abbildung 6 dargestellt, setzt sich ein Zyklus PER_TM aus mehreren Bearbeitungszyklen des Funktionsbausteins PULSEGEN zusammen. Die Anzahl der PULSEGEN-Aufrufe pro PER_TM-Zyklus stellt dabei ein Maß für die Genauigkeit der Impulsbreitenmodulation dar.

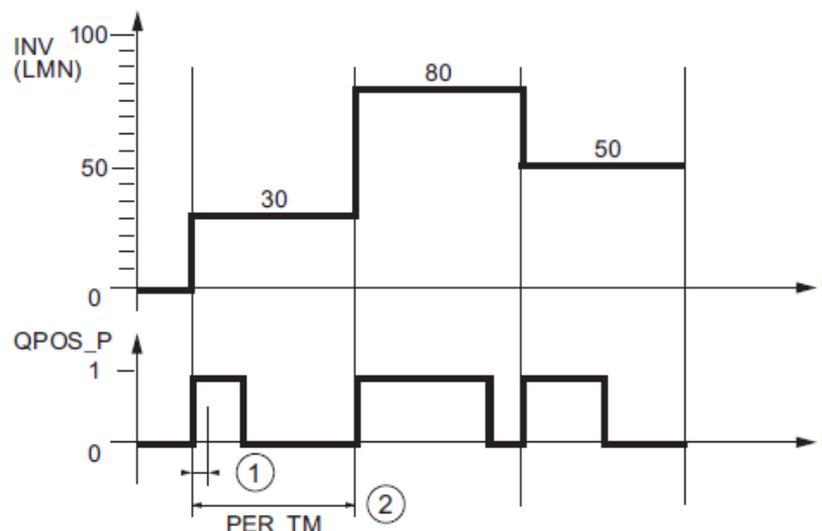


Abbildung 6: Zeitverlauf Eingang INV zu Ausgang QPOS_P von PULSEGEN [4]

Eine Eingangsgröße von 30 % bei 10 PULSEGEN-Aufrufen pro PER_TM bedeutet folgendes:

- 1 am Ausgang QPOS für die ersten drei Aufrufe des PULSEGEN (30 % von 10 Aufrufen)
- 0 am Ausgang QPOS für sieben weitere Aufrufe des PULSEGEN (70 % von 10 Aufrufen)

Die Impulsdauer wird zu Beginn jeder Periode neu berechnet. Durch ein Abtastverhältnis von 1:10 (CTRL_PID-Aufrufe zu PULSEGEN-Aufrufe) ist die Stellwertgenauigkeit in diesem Beispiel auf 10 % beschränkt. Vorgegebene Eingangswerte INV können nur im Raster von 10 % auf eine Impulslänge am Ausgang QPOS abgebildet werden. Entsprechend erhöht sich die Genauigkeit mit der Anzahl der PULSEGEN-Aufrufe pro PIDConL-Aufruf. Wenn PULSEGEN 100 mal und PIDConL nur einmal aufgerufen wird, so wird eine Auflösung von 1 % des Stellwertbereichs erreicht.



Hinweis: Die Untersetzung der Aufrufhäufigkeit müssen Sie selbst programmieren.

LITERATUR

- [1] Chien, Kun Li; Hrones, J. A.; Reswick, J. B. (1952): On the Automatic Control of Generalized Passive Systems. In: Transactions of the American Society of Mechanical Engineers, Bd. 74, Cambridge (Mass.), S. 175-185.
- [2] Ziegler, J. G. und Nichols, N. B. (1942): Optimum settings for automatic controllers. In: Trans. ASME, 64, S. 759-768.
- [3] Kuhn, U.: Eine praxisnahe Einstellregel für PID-Regler: Die T-Summen-Regel. Automatisierungstechnische Praxis, Nr. 5, 1995, S. 10-16.
- [4] SIEMENS (2009): Prozessleitsystem PCS 7: CFC – Elementarbausteine.

SCHRITT-FÜR-SCHRITT-ANLEITUNG

AUFGABENSTELLUNG

Entsprechend den Vorgaben aus dem Kapitel ‚Prozessbeschreibung‘ sollen die CFC-Pläne aus dem Kapitel Anlagensicherung durch die Temperaturregelung und der dazugehörigen Handsteuerung von Reaktor R001 ergänzt werden. Die Heizung des Reaktors wird mit Hilfe eines PID-Reglers mit nachgeschaltetem Pulsgenerator realisiert.

Die folgenden CFC-Pläne werden hier erstellt:

- A1T2H008 (Handbetätigung Heizung Reaktor R001)
- A1T2T001 (Heizung Reaktor R001)

Bei der Realisierung der Temperatursteuerung sind im CFC-Plan die folgenden Verriegelungsbedingungen zu beachten.

- Ein Aktor darf nur geschaltet werden, wenn der Hauptschalter der Anlage eingeschaltet und der NOTAUS-Schalter entriegelt ist.
- Die Temperaturen in den beiden Reaktoren dürfen 60°C nicht übersteigen.
- Die Heizungen der beiden Reaktoren dürfen nur in Betrieb genommen werden, wenn sie mit Flüssigkeit bedeckt sind (hier: minimal 200 ml im Reaktor).

LERNZIEL

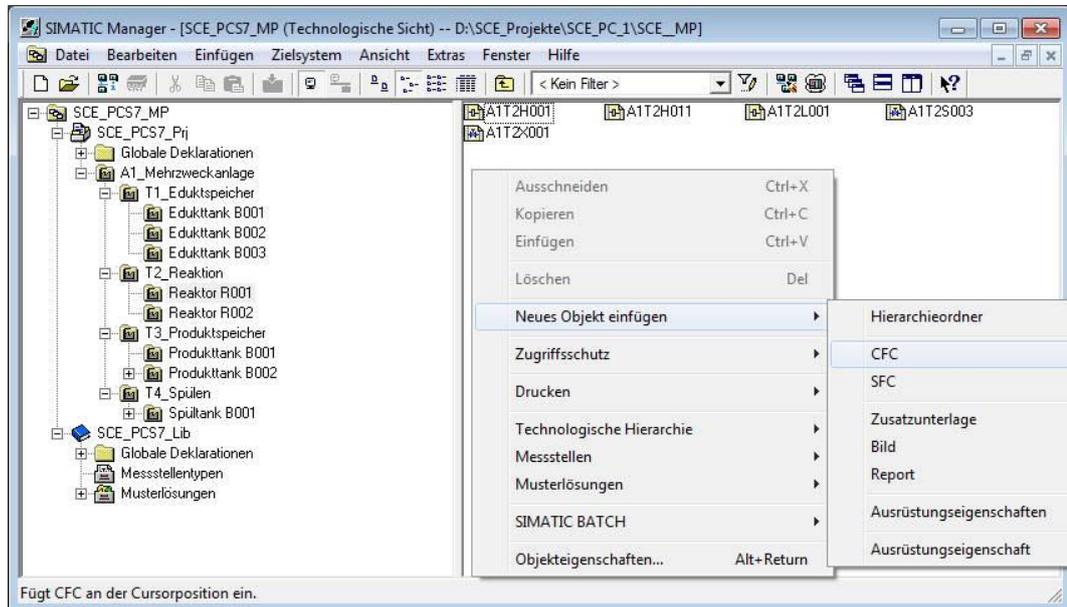
In diesem Kapitel erwirbt der Studierende:

- Kenntnisse zur Programmierung eines kontinuierlichen Reglers mit Impulsausgang und Verriegelungen

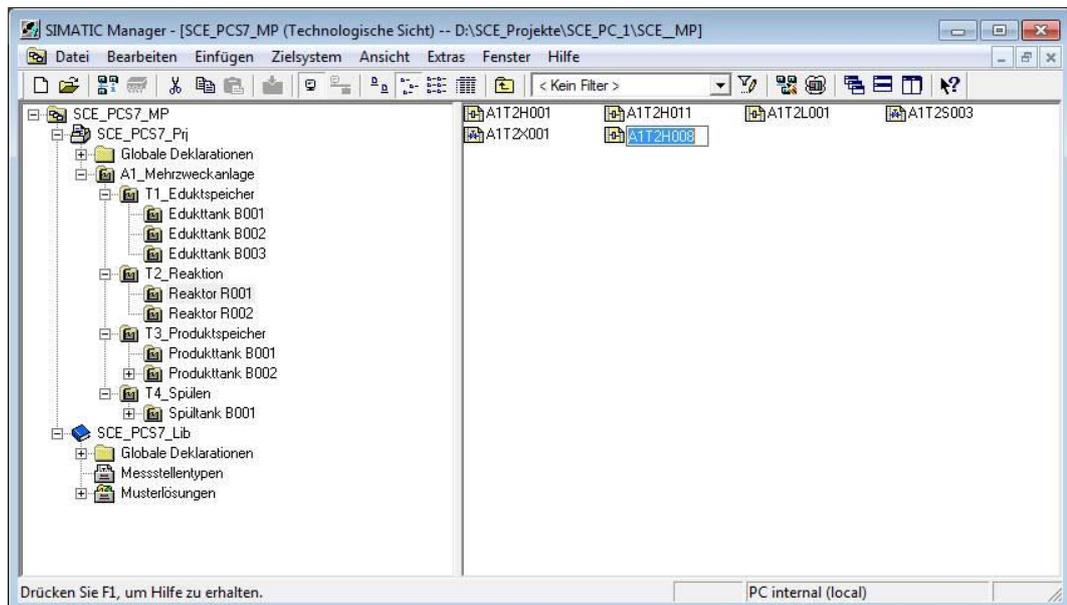
Diese Anleitung baut auf dem Projekt ‚PCS7_SCE_0106_Ueb_R1505.zip‘ auf.

PROGRAMMIERUNG

1. Zunächst fügen Sie einen neuen CFC in den Ordner Reaktor R001 ein. In diesem werden Sie zunächst die Handsteuerung für die Heizung implementieren.



2. Der neu erstellte Plan wird in A1T2H008 umbenannt.



3. Die Verschaltungen von A1T2H008 unterscheiden sich von denen aus A1T2H011 nur in den Ein- und Ausgangssignalen (Pcs7DiIn und Pcs7DiOu) und in den letzten beiden Rücksetzbedingungen (Baustein ‚Or08‘). Die Bedingungen betreffen zum einen den Mindestfüllstand von 200.0 ml und zum anderen die Maximaltemperatur von 60.0 °C.

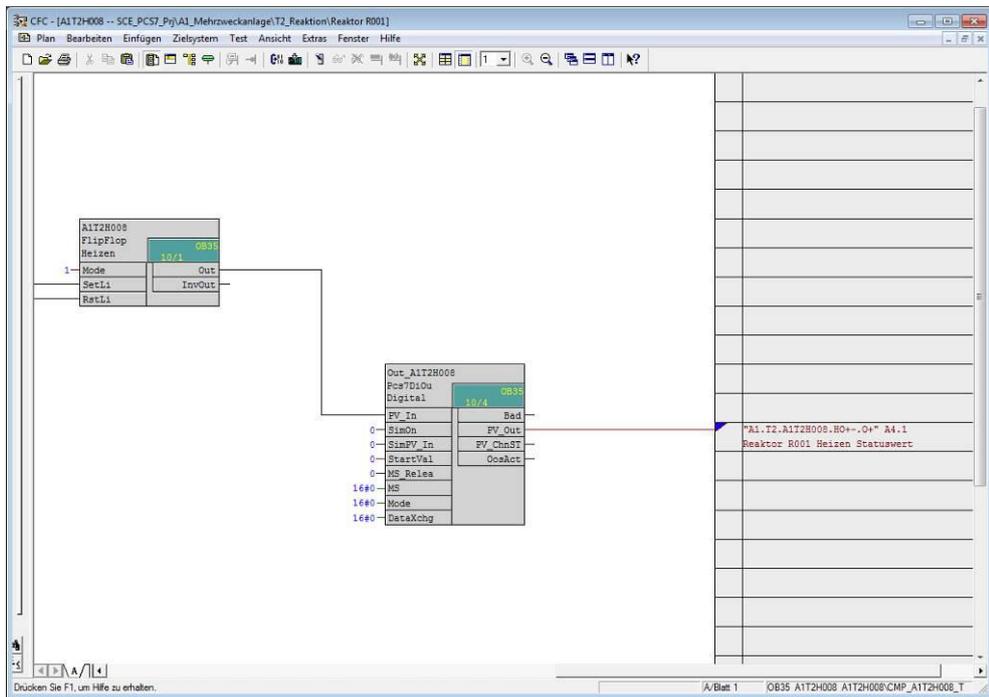
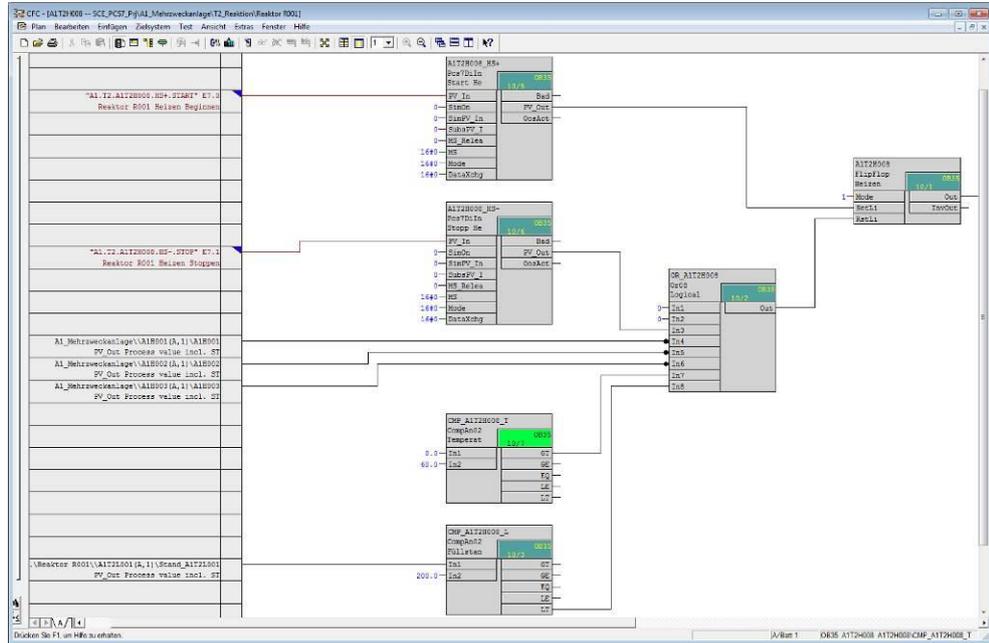


Tabelle 1: Neue Bausteine im Plan ‚A1T2H008‘

Baustein:	Katalog/Ordner:
Pcs7DiIn (2x)	Bausteine/Channel
Or08	Bausteine/LogicDi
CompAn02 (2x)	Bausteine/LogicAn
FlipFlop	Bausteine/LogicDi
Pcs7DiOu	Bausteine/Channel

Tabelle 2: Eingangverschaltungen im Plan ‚A1T2H008‘

Eingang:	Verschaltung zu:	Invertiert
Pcs7DiIn.HS+.PV_In	‚A1.T2.A1T2H008.HS+.START‘/E7.0/ Reaktor R001 Heizen Beginnen	nein
Pcs7DiIn.HS-.PV_In	‚A1.T2.A1T2H008.HS-.STOP‘ / E7.1 / Reaktor R001 Heizen Stoppen	nein
Or08.In4	A1H001(A,1) / A1H001 PV_Out Process value incl. ST	ja
Or08.In5	A1H002(A,1) / A1H002 PV_Out Process value incl. ST	ja
Or08.In6	A1H003(A,1) / A1H003 PV_Out Process value incl. ST	ja
CompAn02.T.In2	60.0	
CompAn02.L.In1	A1T2L001(A,1) / Stand_A1T2L001 PV_Out Process value incl. ST	
CompAn02.L.In2	50.0	
FlipFlop.Mode	1	

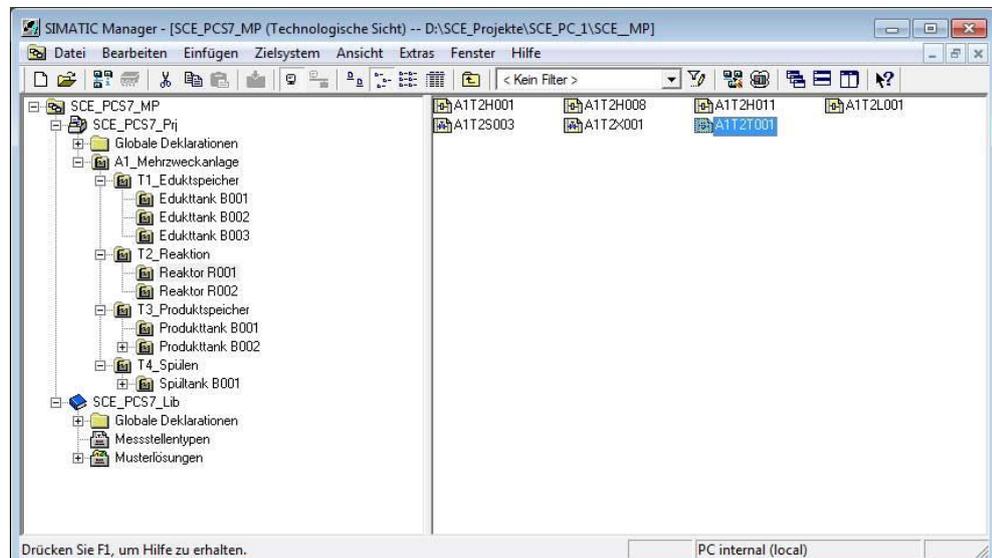
Tabelle 3: Bausteinverschaltungen im Plan ‚A1T2H008‘

Eingang:	Ausgang:	Invertiert
FlipFlop.SetLi	Pcs7DiIn.HS+.PV_Out	nein
FlipFlop.RstLi	Or08.Out	nein
Or08.In3	Pcs7DiIn.HS-.PV_Out	nein
Or08.In7	CompAn02.GT	nein
Or08.In8	CompAn02.LT	nein
Pcs7DiOu.PV_In	FlipFlop.Out	nein

Tabelle 4: Ausgangverschaltungen im Plan ‚A1T2H008‘

Ausgang:	Verschaltung zu:	Invertiert
Pcs7DiOu.PV_OUT	‚A1.T2.A1T2H008.HO+-.0+‘ / A4.1 / Reaktor R001 Heizen Statuswert	nein

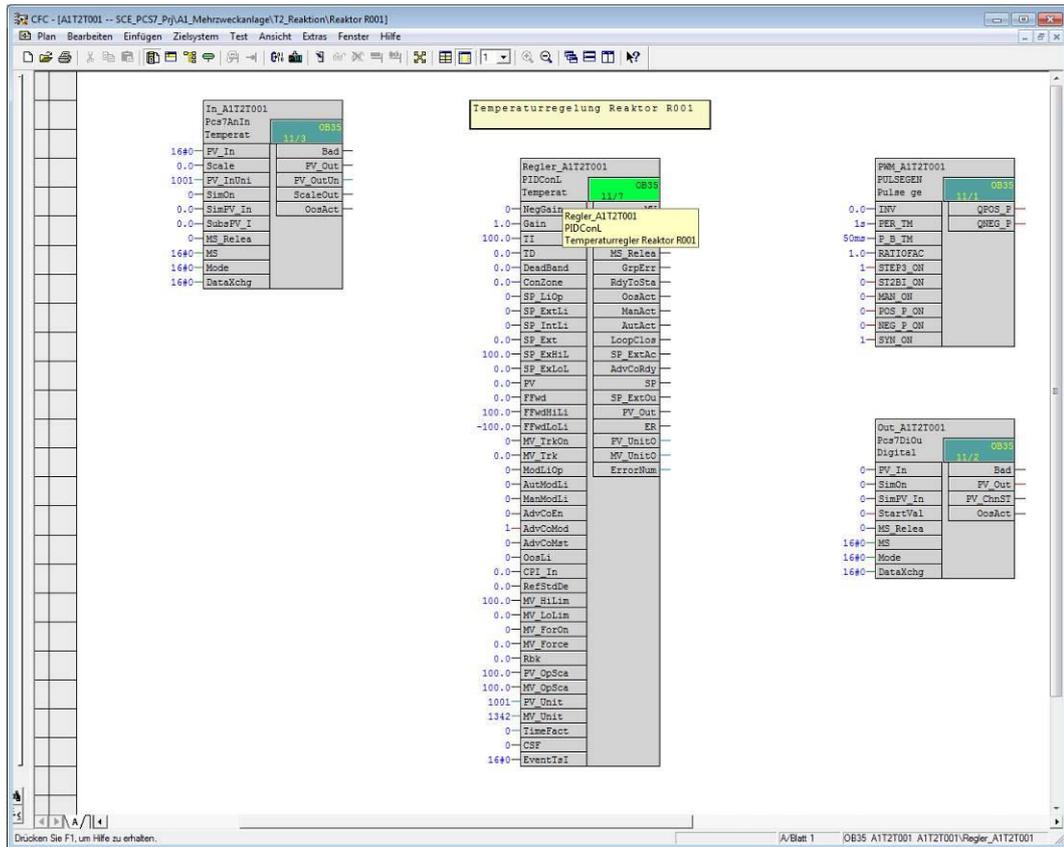
4. Nun wird ein neuer CFC mit dem Namen ‚A1T2T001‘ angelegt. In diesem implementieren Sie die eigentliche Regelung der Heizung des Reaktors R001.



5. Fügen Sie die folgenden Bausteine hinzu und benennen sie sinnvoll.

Tabelle 5: Neue Bausteine im Plan ‚A1T2T001‘

Baustein:	Katalog/Ordner:
Pcs7AnIn	Bausteine/Channel
PIDConL	Bibliotheken/PCS7 APL V80/Blocks+Templates/Blocks/Control
PULSEGEN	Bibliotheken/CFC Library/ELEM400/Blocks/CONTROL
Pcs7DiOu	Bausteine/Channel



6. Daraufhin realisieren Sie die Grundverschaltungen wie in den folgenden Tabellen dargestellt. Vergleichen Sie Ihr Ergebnis mit der Abbildung.

Tabelle 6: Eingangsverschaltungen im Plan ‚A1T2T001‘

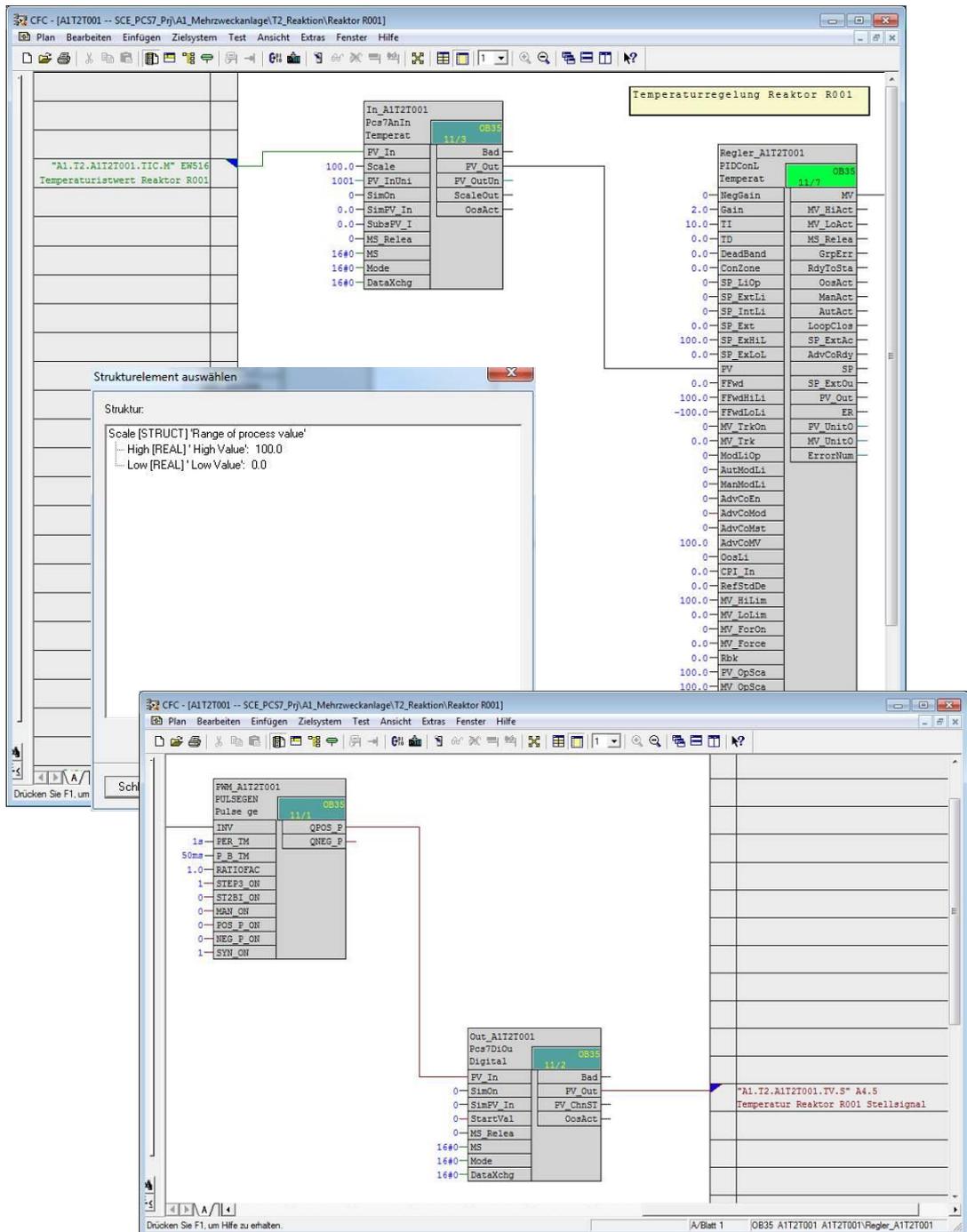
Eingang:	Verschaltung zu:	Invertiert
Pcs7AnIn.PV_In	‚A1.T2.A1T2T001.TIC.M‘ / EW516 / Temperatur-Istwert Reaktor R001	
Pcs7AnIn.Scale	High Value = 100,0, Low Value = 0.0	

Tabelle 7: Bausteinverschaltungen im Plan ‚A1T2T001‘

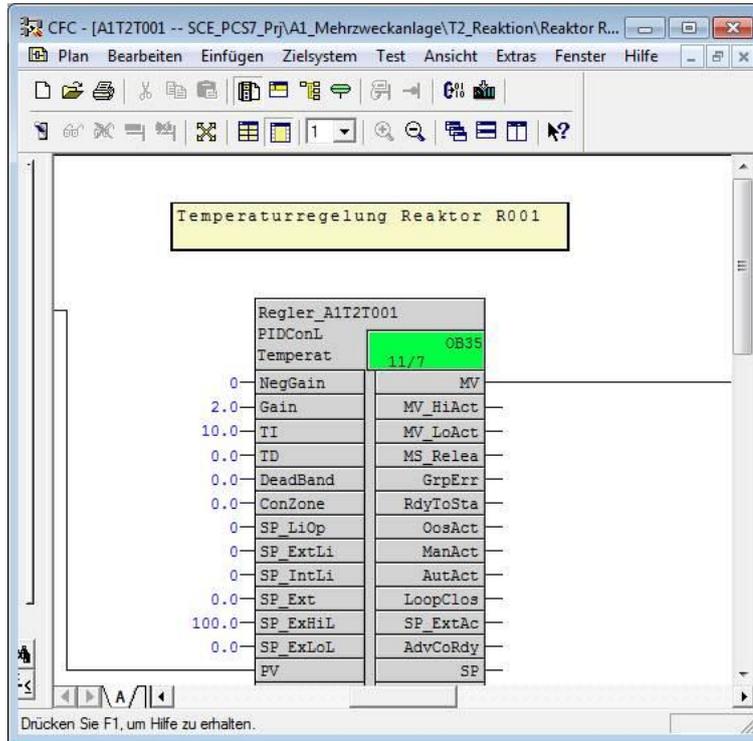
Eingang:	Ausgang:	Invertiert
PIDConL.PV	Pcs7AnIn.PV_Out	
PULSEGEN.INV	PIDConL.MV	
Pcs7DiOu.PV_In	PULSEGEN.QPOS_P	nein

Tabelle 8: Ausgangsverschaltungen im Plan ‚A1T2T001‘

Ausgang:	Verschaltung zu:	Invertiert
Pcs7DiOu.PV_OUT	‚A1.T2.A1T2T001.TV.S‘ / A4.5 / Temperatur Reaktor R001 Stellsignal	nein



7. Jetzt parametrieren Sie die Verstärkungen und die Nachstellzeit des PID-Reglers, indem Sie PIDConL.Gain = 2 und TI = 10.0 setzen.



8. Wechseln Sie nun auf Blatt 2 und legen Sie die nachfolgend dargestellten Verriegelungen an.

Tabelle 9: Neue Bausteine im Plan ‚A1T2T001/Blatt2‘

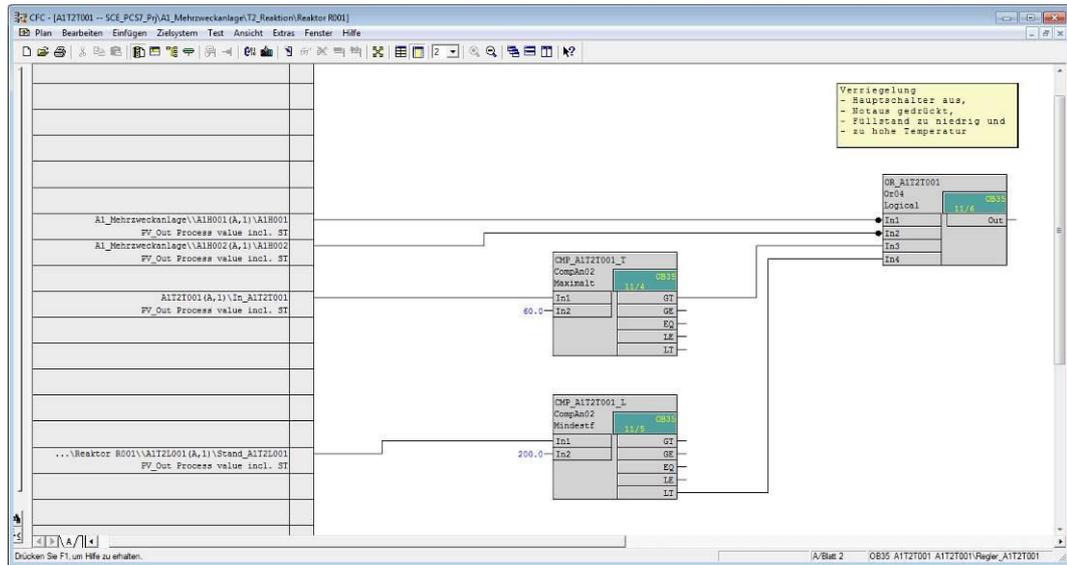
Baustein:	Katalog/Ordner:
Or04	Bausteine/LogicDi
CompAn02 (2x)	Bausteine/LogicDi

Tabelle 10: Eingangsverschaltungen im Plan ‚A1T2T001/Blatt2‘

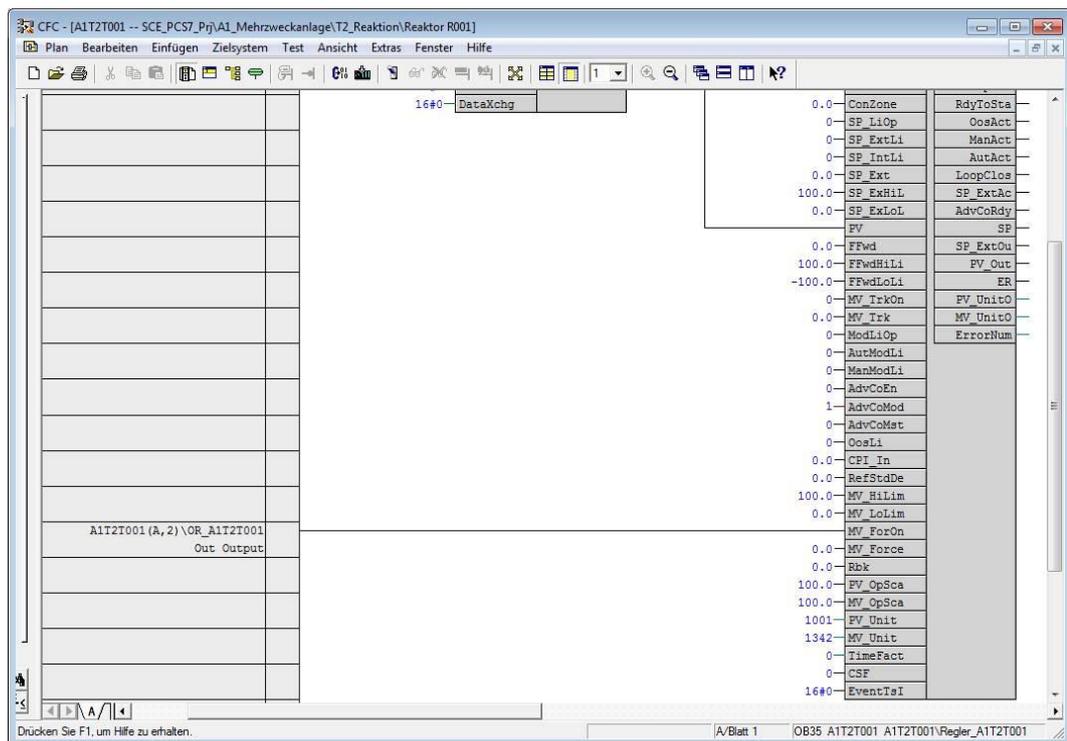
Eingang:	Verschaltung zu:	Invertiert
Or04.In1	A1H001(A,1) / A1H001 PV_Out Process value incl. ST	ja
Or04.In2	A1H002(A,1) / A1H002 PV_Out Process value incl. ST	ja
CompAn02.T.In1	A1T2T001(A,1) / In_A1T2T001 PV_Out Process value incl. ST	
CompAn02.T.In2	60.0	
CompAn02.L.In1	A1T2L001(A,1) / Stand_A1T2L001 PV_Out Process value incl. ST	
CompAn02.L.In1	200.0	

Tabelle 11: Bausteinverschaltungen im Plan ‚A1T2T001‘

Eingang:	Ausgang:	Invertiert
Or04.In3	CompAn02.T.GT	
Or04.In4	CompAn02.L.LT	



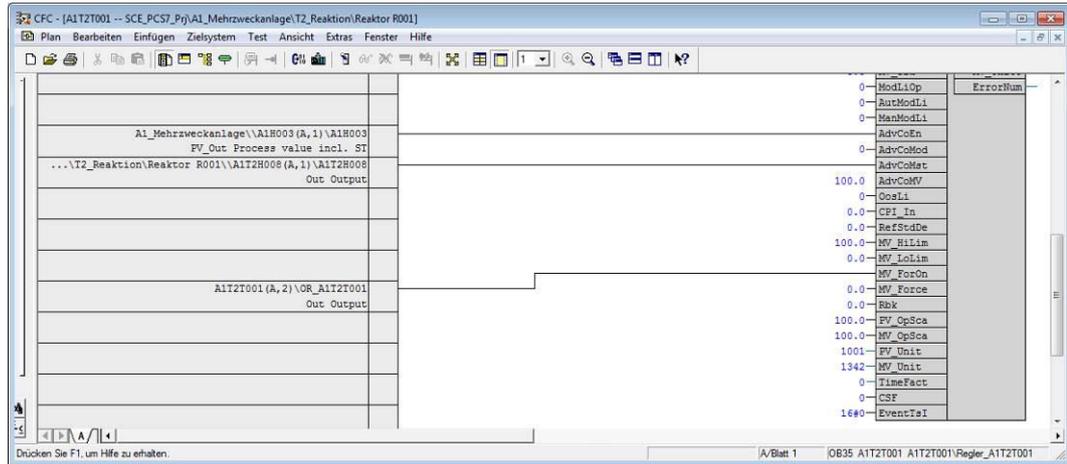
9. Darauf verschalten Sie den Ausgang ‚Out‘ des Bausteins ‚Or04‘ mit dem Eingang ‚MV_ForOn‘ des Bausteins ‚PIDConL‘ und überprüfen das ‚MV_Force‘ = 0.0 ist. Damit wird am Ausgang ‚MV‘ des PID-Reglers (Stellwert des Reglers) der Wert ‚MV_Force‘ angelegt, sobald die Verriegelungsbedingungen erfüllt sind.



10. Jetzt werden Sie die lokale Bedienung parametrieren. Dafür nutzen Sie den Programmierbetrieb. Der Programmierbetrieb wird über den Eingang ‚AdvCoEn‘ freigeschaltet und über ‚AdvCoMst‘ aktiviert. Den Parameter ‚AdvCoMod‘ setzen Sie auf ‚0‘, damit im lokalen Betrieb der Eingang ‚AdvCoMV‘ (im Standard unsichtbar) als Stellwert und nicht als Sollwert interpretiert wird. Schalten Sie ‚AdvCoMV‘ sichtbar und stellen Sie ihn auf 100 ein. Damit wird im lokalen Betrieb die Heizung ohne Regelung angesteuert.

Tabelle 12: Eingangsverschaltungen im Plan ‚A1T2T001/Blatt1‘

Eingang:	Verschaltung zu:	Invertiert
PIDConL.AdvCoEn	A1H003(A,1) / A1H003 PV_Out Process value incl. ST	nein
PIDConL.AdvCoMod	0	
PIDConL.AdvCoMst	A1T2H008(A,1) / A1T2H008 Out Output	nein
PIDConL.AdvCoMV	100.0	



11. Mit der Parametrierung von OS_Perm (Bit 0 bis 31) können die Berechtigungen des Operators eingeschränkt werden. Sie setzen Bit 4 und Bit 7 auf Null, damit der Operator den Programmierbetrieb nicht einschalten und die Handvorgabe für den Stellwert („Man“) nicht verändern kann.

#	Name	I/O	Typ	Wert	Verschaltung	Forcen a...	Forcen aktiv	Force...	SFC-Zug...	T...	Kommentar	Unsichtbar
242	OS_Perm.Bit0	IN	BOOL	1	<nicht verschaltbar>						= Operator can switch into automatic mode AutModOp	
243	OS_Perm.Bit1	IN	BOOL	1	<nicht verschaltbar>						= Operator can switch into manual mode ManModOp	
244	OS_Perm.Bit2	IN	BOOL	1	<nicht verschaltbar>						= Operator can switch into 'out of service' mode OosOp	
245	OS_Perm.Bit3	IN	BOOL	0	<nicht verschaltbar>						= Operator can switch into program mode AdvCoEn	
246	OS_Perm.Bit4	IN	BOOL	1	<nicht verschaltbar>						= Operator can switch the setpoint to external SP_ExtOp	
247	OS_Perm.Bit5	IN	BOOL	1	<nicht verschaltbar>						= Operator can switch the setpoint to internal SP_IntOp	
248	OS_Perm.Bit6	IN	BOOL	1	<nicht verschaltbar>						= Operator can change the internal setpoint SP_Int	
249	OS_Perm.Bit7	IN	BOOL	0	<nicht verschaltbar>						= Operator can change the manual parameter Man	
250	OS_Perm.Bit8	IN	BOOL	1	<nicht verschaltbar>						= Operator can change setpoint high limit SP_HiLim	
251	OS_Perm.Bit9	IN	BOOL	1	<nicht verschaltbar>						= Operator can change setpoint low limit SP_LoLim	
252	OS_Perm.Bit10	IN	BOOL	1	<nicht verschaltbar>						= Operator can change high limit for parameter Man ManHiLim	
253	OS_Perm.Bit11	IN	BOOL	1	<nicht verschaltbar>						= Operator can change low limit for parameter Man ManLoLim	
254	OS_Perm.Bit12	IN	BOOL	1	<nicht verschaltbar>						= Operator can use the setpoint's gradient limitation funct. SP_Rat...	
255	OS_Perm.Bit13	IN	BOOL	1	<nicht verschaltbar>						= Operator can change the setpoint's upward rate limit SP_UpRateLim	
256	OS_Perm.Bit14	IN	BOOL	1	<nicht verschaltbar>						= Operator can change the setpoint's downward rate limit SP_DnR...	
257	OS_Perm.Bit15	IN	BOOL	1	<nicht verschaltbar>						= Operator can change ramp mode (time or gradient value) SP_Rm...	
258	OS_Perm.Bit16	IN	BOOL	1	<nicht verschaltbar>						= Operator can change the ramp time SP_RmpTime	
259	OS_Perm.Bit17	IN	BOOL	1	<nicht verschaltbar>						= Operator can change the target setpoint in ramp function SP_Rm...	
260	OS_Perm.Bit18	IN	BOOL	1	<nicht verschaltbar>						= Operator can activate the setpoint ramp function SP_RmpOn	
261	OS_Perm.Bit19	IN	BOOL	1	<nicht verschaltbar>						= Operator can enable the PID optimization function OptmEn	
262	OS_Perm.Bit20	IN	BOOL	1	<nicht verschaltbar>						= Operator can activate the tracking of setpoint in manual mode SP...	
263	OS_Perm.Bit21	IN	BOOL	1	<nicht verschaltbar>						= Operator can activate bumpless chngover external/internal SP...	
264	OS_Perm.Bit22	IN	BOOL	1	<nicht verschaltbar>						= Operator can change the gain parameter Gan	
265	OS_Perm.Bit23	IN	BOOL	1	<nicht verschaltbar>						= Operator can change the integral time parameter TI	
266	OS_Perm.Bit24	IN	BOOL	1	<nicht verschaltbar>						= Operator can change the derivative time parameter TD	
267	OS_Perm.Bit25	IN	BOOL	1	<nicht verschaltbar>						= Operator can change the derivative gain parameter DfGain	
268	OS_Perm.Bit26	IN	BOOL	1	<nicht verschaltbar>						= Operator can change the deadband parameter DeadBnd	
269	OS_Perm.Bit27	IN	BOOL	1	<nicht verschaltbar>						= Operator can change the control zone parameter ConZone	
270	OS_Perm.Bit28	IN	BOOL	1	<nicht verschaltbar>						= Operator can change the derivative gain parameter ER_AH_DFac	
271	OS_Perm.Bit29	IN	BOOL	1	<nicht verschaltbar>						= Operator can change the derivative gain parameter ER_AL_DFac	
272	OS_Perm.Bit30	IN	BOOL	1	<nicht verschaltbar>						Reserved	
273	OS_Perm.Bit31	IN	BOOL	1	<nicht verschaltbar>						Reserved	

12. Zum Schluss können Sie noch die folgende Verschaltung vom Plan ‚A1T2H008‘ mit dem Plan ‚A1T2T001‘ durchführen.

Tabelle 13: Bausteinverschaltungen zwischen Plan ‚A1T2H008/Blatt1‘ und ‚A1T2T001/Blatt1‘

Eingang:	Ausgang:	Invertiert
CompAn02.T.In1	A1T2T001(A,1) / In_A1T2T001 PV_Out Process value incl. ST	

ÜBUNGEN

In den Übungsaufgaben soll Gelerntes aus der Theorie und der Schritt-für-Schritt-Anleitung umgesetzt werden. Hierbei soll das schon vorhandene Multiprojekt aus der Schritt-für-Schritt-Anleitung (PCS7_SCE_0106_R1505.zip) genutzt und erweitert werden.

Zur Vorbereitung des nächsten Kapitels sollen Sie die letzte fehlende Funktion des Reaktors R001 implementieren – den Rührer und die Handbetätigung des Rührers. Die Verriegelungsbedingungen lauten wie folgt:

- Ein Aktor darf nur geschaltet werden, wenn der Hauptschalter der Anlage eingeschaltet und der NOTAUS-Schalter entriegelt ist.
- Die Rührer der beiden Reaktoren sollten nur in Betrieb genommen werden, wenn sie mit Flüssigkeit in Berührung kommen (hier: minimal 300 ml im Reaktor).

Zusätzlich können Sie sich auch weiter über den PID-Regler, seine Funktionsweise und die einstellbaren Parameter informieren. Für die Funktionalität der Steuerung ist das hier aber nicht nötig.

ÜBUNGSAUFGABEN

1. Implementieren Sie den Rührer A1T2S001 im Planordner ‚Reaktor R001‘. Verwenden Sie für den Rührer denselben Messstellentyp wie für die Pumpen. Schließen Sie das Rückmeldungs- und das Stellsignal an. Benennen Sie die Bausteine sinnvoll. Fügen Sie anschließend die Verriegelungen wie oben erläutert hinzu.
2. Implementieren Sie nun die Handsteuerung A1T2H007 für den eben erstellten Rührer. Realisieren Sie hier die Verriegelungsbedingungen als Rücksetzbedingungen.
3. Informieren Sie sich über die Eingänge ‚ModLiOp‘, ‚AutModLi‘, ‚ManModLi‘ des Bausteins ‚PIDConL‘. Rufen Sie dazu mit der Funktionstaste ‚F1‘ die Hilfe zum Baustein ‚PIDConL‘ auf. Wählen Sie ‚Betriebsarten von PIDConL‘ und anschließend den Hand- oder Automatikbetrieb aus.
4. Wenn Sie mehr über die Eingänge ‚SP_LiOp‘, ‚SP_ExtLi‘, ‚SP_IntLi‘, etc. erfahren wollen, geben Sie in der Hilfe im Register ‚Suchen‘ das Wort Sollwertvorgabe ein. Unter dem vorgeschlagenen Titel ‚Sollwertvorgabe – Intern/Extern‘ erhalten Sie Informationen.
5. Wozu dienen die Parameter MV_HiLim und MV_LoLim? Suchen Sie selbständig in der Hilfe nach Informationen zu diesen Eingängen.