



SIEMENS



SCE Lehrunterlagen

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

PA Modul P01-08
SIMATIC PCS 7 – Ablaufsteuerungen

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.

Zuwendungen 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.

ABLAUFSTEUERUNGEN

LERNZIEL

Die Studierenden können Ablaufsteuerungen erfolgreich mithilfe von Schrittketten realisieren. Sie verstehen die Struktur und die Wirkungsweise von Schrittketten und lernen entsprechende Entwurfsmethoden kennen. Die Kenntnisse über Betriebsarten und Schutzmaßnahmen werden für Ablaufsteuerungen erweitert. Die Studenten verstehen die Interaktion zwischen den Programmen der Basisautomatisierung und den Ablaufsteuerungen. Sie wissen, wie Ablaufsteuerungen in **PCS 7** erstellt werden können.

THEORIE IN KÜRZE

Ablaufsteuerungen ermöglichen eine zeit- oder ereignisdiskrete Abarbeitung sequenzieller und paralleler Abläufe. Sie dienen der Koordination von verschiedenen kontinuierlichen Funktionen sowie zur Steuerung von komplexen Prozessabläufen. Abhängig von definierten Zuständen oder Ereignissen werden Betriebs- und Zustandswechsel in den vorhandenen Verknüpfungssteuerungen erzeugt und so das gewünschte Ablaufverhalten realisiert. Sie werden durch eine oder mehrere **Schrittketten** (engl. **Sequential Function Charts** bezeichnet) implementiert.

Eine Schrittkette ist eine alternierende Aneinanderreihung von **Schritten**, die jeweils bestimmte Aktionen auslösen, und **Transitionen**, welche den Wechsel von einem Schritt in einen anderen veranlassen, sobald die entsprechende **Weiterschaltbedingung** erfüllt ist. Jede Schrittkette besitzt genau einen **Start-Schritt** und einen **Ende-Schritt** sowie zusätzlich beliebig viele Zwischenschritte, die jeweils durch gerichtete Kanten über zwischengeschaltete Transitionen miteinander verbunden sind. Die Graphen dürfen auch Rückkopplungen durch Schleifen innerhalb der Schrittkette erzeugen. Ebenso können sie parallele oder alternative Verzweigungen enthalten. Dabei muss jedoch beim Entwurf sichergestellt werden, dass die Kette keine unsicheren oder unerreichbaren Teile enthält.

Für den Entwurf einer Ablaufsteuerung bieten sich insbesondere die formalen Entwurfsmethoden mithilfe von **Zustandsgraphen** oder **Petrinetzen** an. Zustandsgraphen sind leicht erlernbar, ermöglichen eine automatische Fehlerdiagnose und lassen sich problemlos in viele bestehende Programmiersprachen für Ablaufsteuerungen umsetzen. Allerdings ist der Entwurf paralleler Strukturen nicht möglich, da Zustandsgraphen nur genau einen aktiven Zustand haben.

Petrinetze sind wesentlich komplexer und mathematisch anspruchsvoller. Jedoch können sämtliche Strukturen, die in Ablaufsteuerungen erlaubt sind, modelliert und umfassend analysiert werden. Damit können notwendige Eigenschaften der Steuerung formal nachgewiesen werden. Petrinetze erlauben ebenfalls eine problemlose Implementierung in Ablaufsteuerungen.

Ablaufsteuerungen parametrieren und aktivieren untergeordnete Verknüpfungssteuerungen, indem sie entsprechende globale Steuersignale setzen. Diese Steuersignale können kurzzeitig oder dauerhaft, unmittelbar oder verzögert wirken. Ablaufsteuerungen müssen ebenso wie Verknüpfungssteuerungen verschiedene Betriebsarten unterstützen, wobei insbesondere die manuelle Steuerung der Transitionen und die zeitweise oder dauerhafte Unterbrechung der Prozessabläufe möglich sein muss. Außerdem werden prozessspezifische Schutzfunktionen durch Ablaufsteuerungen realisiert.

Ablaufsteuerungen werden in **PCS 7** durch **Sequential Function Charts (SFC)** realisiert. SFC bieten eine leistungsfähige Betriebsartenverwaltung, eine hohe Steuerbarkeit durch mehrere Schaltmodi sowie eine umfangreiche Parametrierbarkeit durch verschiedene Ablaufoptionen. Die Interaktion und Verknüpfung zwischen SFC und CFC erfolgt in **PCS 7** über Prozesswerte und Steuerwerte. Das Interaktionsverhalten kann ebenfalls detailliert gesteuert werden.

THEORIE

KONTINUIERLICHE UND SEQUENTIELLE STEUERUNGEN

Im Rahmen der Basisautomatisierung werden verschiedene Verknüpfungssteuerungen entwickelt, die jeweils eine begrenzte, klar definierte Funktion realisieren. Die Funktionen verarbeiten kontinuierlich Eingangssignale und generieren entsprechende Ausgangssignale. Über verschiedene Steuersignale können die Funktionen darüber hinaus aktiviert und parametrierbar werden. Um komplexe Prozessabläufe, zum Beispiel Herstellungsvorschriften von Produkten (**Rezepte**), zu realisieren ist es notwendig die verschiedenen Funktionen zu koordinieren und zum rechten Zeitpunkt mit den richtigen Parametern zu aktivieren. Diese Aufgabe kann mit Hilfe von Ablaufsteuerungen realisiert werden.

Ablaufsteuerungen ermöglichen eine schrittweise, ereignisdiskrete Abarbeitung sequenzieller und paralleler Abläufe mit Hilfe von **Schrittketten** (auch als **Ablaufketten** bezeichnet). Sie erzeugen abhängig von definierten Zuständen oder Ereignissen Betriebs- und Zustandswechsel in den vorhandenen Verknüpfungssteuerungen und realisieren so das gewünschte Ablaufverhalten. Schrittketten werden auch als **Sequential Function Charts** bezeichnet.

AUFBAU VON SCHRITTKETTEN

Eine Schrittkeine ist eine wechselweise Abfolge von **Schritten** und **Transitionen**. Die einzelnen Schritte aktivieren jeweils bestimmte Aktionen. Transitionen steuern den Wechsel von einem Schritt in einen anderen.

Der erste Schritt einer Schrittkeine wird als **Start-Schritt** bezeichnet. Er ist der eindeutige Einstiegspunkt in die Kette und wird daher stets ausgeführt. Der letzte Schritt einer Schrittkeine wird entsprechend als **Ende-Schritt** bezeichnet. Er ist der einzige Schritt der Kette, der keine Folgetransition besitzt. Nach der Abarbeitung des Ende-Schrittes wird die Schrittkeine beendet oder die Abarbeitung beginnt von neuem. Im zweiten Fall spricht man auch von einer Kettenschleife.

Schritte und Transitionen werden durch gerichtete Graphen miteinander verbunden. Dabei kann ein Schritt mit mehreren Folgetransitionen verbunden sein, ebenso ist der umgekehrte Fall möglich. Eine Transition wird freigegeben, wenn alle vorgeschalteten Schritte aktiv sind und die Weichschaltbedingung erfüllt ist. In diesem Fall werden zunächst die unmittelbar vorangehenden Schritte deaktiviert und anschließend die unmittelbaren Folgeschritte aktiviert.

Die einfachste Form einer Schrittkeine ist die unverzweigte Kette. Auf jeden Schritt folgt in diesem Fall genau eine Transition und auf diese wiederum genau ein Folgeschritt. Damit wird ein rein sequenzieller Prozessablauf realisiert. Abbildung 1 zeigt die entsprechenden graphischen Grundelemente.

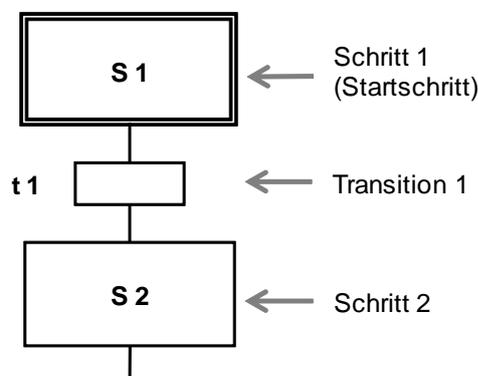


Abbildung 1: Grundelemente eines Ablauf-Funktionsplans

Schleifen innerhalb der Schrittfolge entstehen, wenn durch die Aneinanderreihung mehrerer Schritte ein zyklischer Ablauf innerhalb der Kette möglich wird. Die Kettenschleife stellt einen Sonderfall einer Schleife dar, bei dem sämtliche Schritte zyklisch durchlaufen werden.

Eine weitere Möglichkeit der Strukturierung von Schrittfolgen stellen Sprünge dar. Beim Erreichen einer Sprungmarke wird die Abarbeitung mit dem Schritt fortgesetzt, auf den die Sprungmarke zeigt. Durch Sprünge innerhalb der Schrittfolge können ebenfalls Schleifen entstehen. Da eine solche Strukturierung nur schwer nachvollziehbar ist, sollte auf Sprünge nach Möglichkeit verzichtet werden.

In vielen Fällen ist es aus Prozesssicht notwendig, zur Programmlaufzeit auf verschiedene Ereignisse unterschiedlich zu reagieren. In diesem Fall besitzt ein Schritt mehrere alternative Folgeschritte. Diese Struktur wird **Alternativverzweigung** genannt. Der Schritt ist mit jedem möglichen Folgeschritt über eine eigene Transition verbunden. Um sicherzustellen, dass zu jeder Zeit höchstens eine dieser Transitionen freigegeben wird (und die Zweige tatsächlich alternativ sind), sollten die Transitionen gegenseitig verriegelt oder aber eindeutig priorisiert werden. Ansonsten werden die Transitionen in den meisten Leitsystemen von links nach rechts ausgewertet, und die erste Transition, deren Weichschaltbedingung erfüllt ist, wird freigegeben.

Abbildung 2 zeigt den prinzipiellen Aufbau einer Alternativverzweigung mit zwei Zweigen. Sie ist durch einfassende horizontale Einfachstriche mit überstehenden Enden dargestellt. Wie zu erkennen ist, beginnen und enden Alternativverzweigungen immer mit Transitionen.

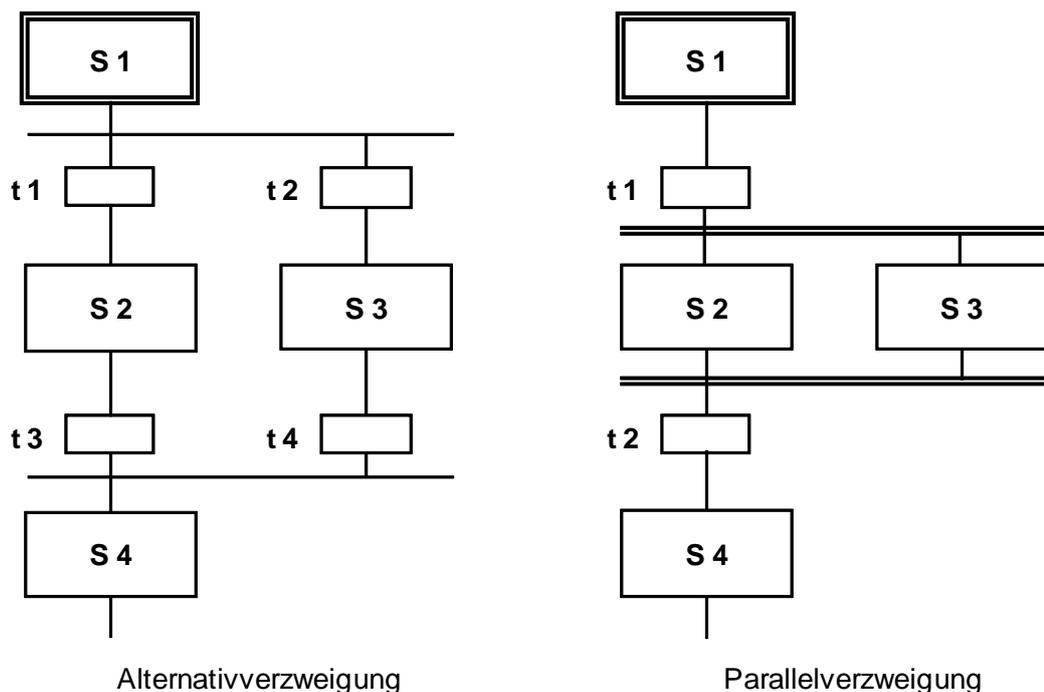


Abbildung 2: Alternative und parallele Verzweigungen in Schrittfolgen

Eine weitere häufige Anforderung besteht darin, dass nach einem Schritt mehrere Folgeschritte gleichzeitig abgearbeitet werden sollen. In diesem Fall besitzt der Ausgangsschritt genau eine Transition, die gleichzeitig mehrere Folgeschritte aktiviert. Diese Struktur wird **Parallelverzweigung** genannt. Die Folgeschritte der einzelnen Zweige werden daraufhin unabhängig voneinander abgearbeitet und anschließend wieder zusammengeführt. Sämtliche Zweige enden wiederum in einer gemeinsamen Transition. Erst wenn sämtliche Zweige vollständig abgearbeitet sind und die Weichschaltbedingung der Folgetransition erfüllt ist, kann der gemeinsame Folgeschritt ausgelöst werden.

Der Ablauf einer Parallelverzweigung mit zwei Zweigen wird ebenfalls in Abbildung 2 dargestellt. Sie sind durch einfassende horizontale Doppelstriche mit überstehenden

Enden dargestellt. Wie zu erkennen ist, beginnen und enden Parallelverzweigungen immer mit Aktionen.

Ein besonderes steuerungstechnisches Problem ist die Möglichkeit, durch die ungünstige Verwendung von Sprüngen und Verzweigungen fehlerhafte Schrittketten zu erzeugen. Dabei ist zwischen drei möglichen Fällen zu unterscheiden.

- **Unsichere Kette:** Eine Schrittfolge enthält eine Struktur, deren Erreichbarkeit durch das definierte Ablaufverhalten nicht sichergestellt ist.
- **Partielle Verklemmung:** Eine Schrittfolge enthält eine innere Schleife, die nicht mehr verlassen wird. Damit können die Schritte innerhalb dieser Schleife ausgeführt werden, nicht aber die Schritte außerhalb der Schleife. So werden Teile der Schrittfolge unerreichbar.
- **Totale Verklemmung:** Eine Schrittfolge enthält eine Struktur, für die es keine zulässige Weichschaltbedingung gibt. In diesem Fall verbleibt die Schrittfolge dauerhaft in einem Zustand und sämtliche anderen Schritte werden unerreichbar.

Solche Strukturen sind in Schrittfolgen nicht erlaubt und müssen mit entsprechenden formalen Entwurfsmethoden ausgeschlossen werden. Abbildung 3 zeigt beispielhaft zwei Schrittfolgen mit unerlaubten Strukturen.

In der linken Kette kann nicht sichergestellt werden, dass der Schritt S6 erreichbar ist, da die Alternativverzweigung nach Schritt S3 bei Freigabe der Transition t3 verhindert, dass die Parallelverzweigung in der Transition t4 wieder zusammengeführt wird. Daher ist diese Kette unsicher. Die rechte Kette hingegen wird genau einmal ausgeführt und bleibt anschließend in Schritt S4 stehen. Da der Schritt S2 in diesem Zustand nicht aktiv ist, kann die Parallelverzweigung in Transition t3 nicht mehr zusammengeführt werden. Es entsteht eine totale Verklemmung, der Schritt S5 wird unerreichbar.

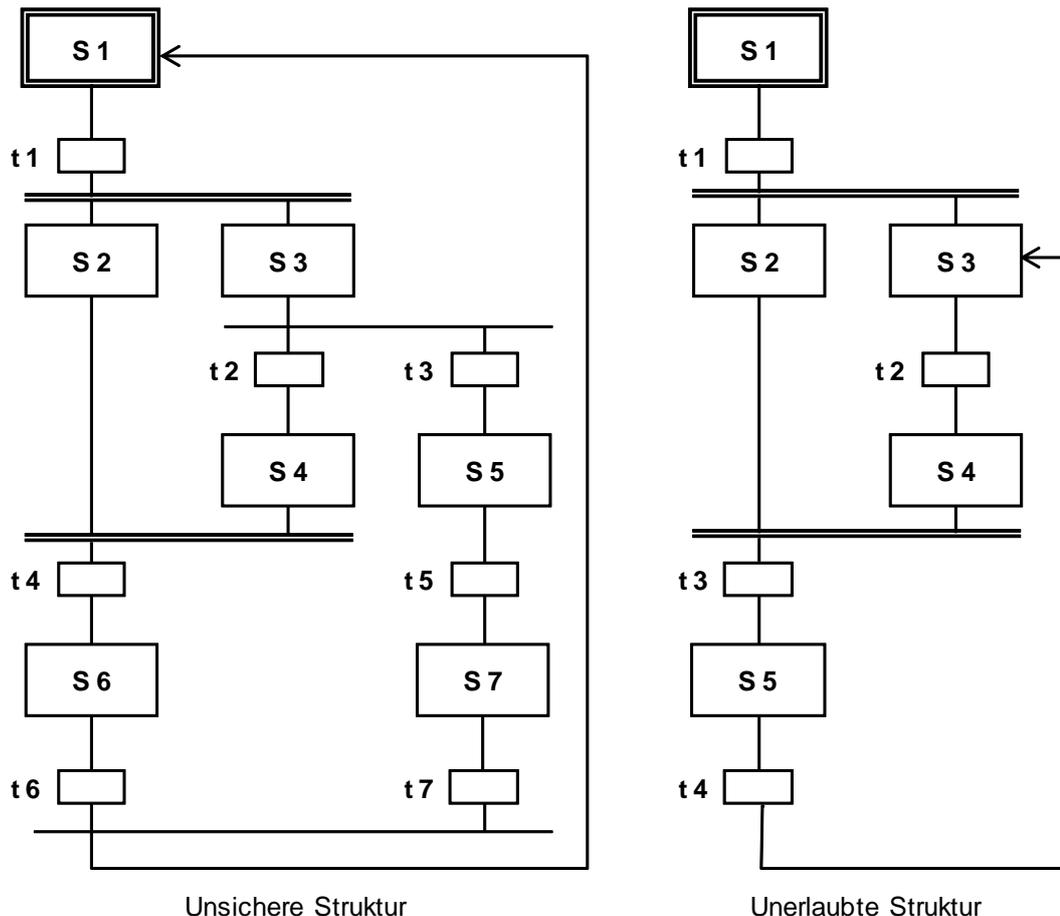


Abbildung 3: Unsichere und unerlaubte Strukturen in Schrittfolgen

ENTWURF VON ABLAUFSTEUERUNGEN

Es existiert eine Vielzahl formaler Entwurfsmethoden für Ablaufsteuerungen. In der Praxis haben sich jedoch insbesondere die Modelle des **Zustandsgraphen** sowie des **Petrinetzes** bewährt.

Ein **Zustandsgraph** ist ein zusammenhängender, gerichteter Graph. Zustände werden als Kreise dargestellt und Zustandsübergänge als Pfeile, die genau zwei Zustände miteinander verbinden. In einem Zustandsgraphen ist zu einem Zeitpunkt stets genau ein Zustand aktiv. Die Zustände können mit bestimmten Aktionen verknüpft werden. Diesen Aktionen kann ein bestimmtes Ablaufverhalten zugewiesen werden. Sie können einmalig beim Eintritt in den Zustand oder beim Austritt aus dem Zustand ausgeführt werden, oder aber zyklisch, solange der Zustand aktiv ist. Zustandsübergänge können mit Übergangsbedingungen behaftet werden.

Zustandsgraphen können hierarchisch gegliedert und miteinander verknüpft werden. Zustandsgraphen gelten als leicht erlernbar, ermöglichen eine automatische Fehlerdiagnose, zum Beispiel durch Paar-, Zeit- oder Zustandsüberwachung. Sie lassen sich problemlos in viele bestehende Programmiersprachen für Ablaufsteuerungen umsetzen.

Petrinetze eignen sich besonders zur Modellierung nebenläufiger Prozesse. Ein Petrinetz besteht aus Plätzen und Transitionen die durch gerichtete Kanten miteinander verbunden sind. Damit entsteht ebenfalls ein gerichteter Graph. Ein Platz wird als Kreis dargestellt, eine Transition als Rechteck (häufig auch reduziert zu einem Querbalken). Aktive Plätze werden durch Marken gekennzeichnet, was durch einen Punkt innerhalb des Kreises für den entsprechenden Platz dargestellt wird.

Im Unterschied zu Funktionsgraphen wird der Zustand in einem Petrinetz durch die Menge der aktiven Plätze im gesamten Netz bestimmt. Die Dynamik des Systems wird durch die Bewegung der Marken innerhalb des Netzes modelliert. Welche Bedeutung die Plätze und Transitionen für den modellierten Prozess haben (also die **Semantik** des Petrinetzes), ist grundsätzlich nicht definiert und muss je nach Anwendungsfall festgelegt werden. Petrinetze, deren Semantik festgelegt wurde, werden als **Interpretierte Petrinetze (IPN)** bezeichnet. Für den Steuerungsentwurf werden in der Regel **Steuerungstechnisch Interpretierte Petrinetze (SIPN)** verwendet.

Petrinetze können umfassend analytisch untersucht werden. Sie erlauben ebenfalls eine problemlose Umsetzung in bestehende Programmiersprachen für Ablaufsteuerungen. Es existieren zahlreiche Erweiterungen für Petrinetze, die jeweils für bestimmte Anwendungsfälle optimiert sind oder eine genauere Modellierung des Prozesses erlauben. Petrinetze können daher recht komplex werden, was sie als Entwurfsmethode entsprechend anspruchsvoll macht. Aufgrund ihrer strukturellen Ähnlichkeit zu Schrittketten und der Möglichkeit der Modellierung paralleler Abläufe bieten Petrinetze jedoch auch deutliche Vorteile.

Welche Entwurfsmethode zur Anwendung kommt, hängt letztendlich von den Anforderungen der Entwurfsaufgabe sowie von der Präferenz des Entwicklers ab. Für weiterführende Informationen sei auf die einschlägige Fachliteratur verwiesen.

INTERAKTION VON ABLAUF- UND VERKNÜPFUNGSSTEUERUNGEN

Wie bereits beschrieben können jedem Schritt in der Schrittkette bestimmte Aktionen zugeordnet werden. Diese Aktionen bestehen im Allgemeinen in der Parametrierung und Aktivierung von Verknüpfungssteuerungen. Dazu werden entsprechende Steuersignale gesetzt.

Von Schrittketten verwendete Prozess- und Steuersignale müssen global deklariert werden, damit sie den Programmen der Ablauf- und der Verknüpfungssteuerungen gleichermaßen sind. Üblicherweise werden die Signale in einer Symboltabelle zusammengefasst.

Grundsätzlich wirken Steuersignale stets so lange, wie der entsprechende Schritt aktiv ist. Zur Realisierung komplexerer Funktionsabläufe besteht aber auch die Möglichkeit die Verarbeitung eines Steuersignals selbst zu variieren (speichernd oder nicht speichernd, zeitlich verzögert oder limitiert).

Üblicherweise werden prozessspezifische Funktionen durch Ablaufsteuerungen realisiert, während Verknüpfungssteuerungen alle gerätespezifischen Funktionen implementieren.

SCHUTZFUNKTIONEN UND BETRIEBSARTEN IN ABLAUFSTEUERUNGEN

Ebenso wie bei Einzelsteuerfunktionen müssen für Ablaufsteuerungen adäquate Schutzfunktionen und Betriebsarten realisiert werden. Ablaufsteuerungen müssen auch im Fehlerfall manuell bedienbar sein. Dazu sind in der Steuerung entsprechende Betriebsarten vorzusehen.

- **Automatikbetrieb:** Die Aktion der Schrittkette wird ausgeführt, wenn die vorgeschaltete Transition freigegeben ist.
- **Handbetrieb:** Die Aktion der Schrittkette wird durch den Bediener ausgelöst, auch wenn die vorgeschaltete Transition nicht freigegeben ist.
- **Mischbetrieb:** Die Aktion der Schrittkette wird ausgeführt, wenn die vorgeschaltete Transition freigegeben ist oder der Bediener sie ausgelöst hat. Alternativ kann auch die Auslösung durch den Bediener sowie die Freigabe der vorgeschalteten Transition gefordert sein.

Mit Hilfe des Handbetriebs wird verhindert, dass eine Ablaufsteuerung in Folge eines Fehlzustands dauerhaft blockiert werden kann. Der Mischbetrieb ermöglicht eine manuelle Unterbrechung des Ablaufs zum Test- oder zur Inbetriebnahme. Die Weiterschaltbedingungen sämtlicher Transitionen der Ablaufsteuerung müssen dementsprechend erweitert werden.

Schrittketten müssen auf Störungen in den angesteuerten Geräten reagieren können. Dazu ist eine kontinuierliche Störüberwachung erforderlich. Diese erkennt und meldet Störungen in den angesteuerten Geräten. Sie ermöglicht eine automatisierte Sicherung der Anlage, indem die Schrittkette im Störfall automatisch angehalten wird. Außerdem muss eine Schrittkette bei einer Störung jederzeit durch den Bediener angehalten und abgebrochen werden können.

In beiden Fällen müssen entsprechende Schutzfunktionen aktiviert werden, um die Anlage in einen sicheren Zustand zu überführen. Im Falle einer angehaltenen Kette muss zudem eine sichere und prozesstechnisch zulässige Fortsetzung der Kette auch nach einer längeren Unterbrechung sichergestellt werden. In den Ablaufsteuerungen werden prozessspezifische Schutzfunktionen wie die sequentielle Verriegelung mehrerer Geräte im Falle eines Fehlzustandes im Prozess realisiert.

ABLAUFSTEUERUNGEN IN PCS 7

Ablaufsteuerungen werden in **PCS 7** durch **Sequential Function Charts (SFC)** realisiert. Diese enthalten die Schrittketten und definieren deren Kettentopologie, die Bedingungen der Transitionen und die Aktionen der Schritte. Die Startbedingungen und Ablaufigenschaften kann man für jede Schrittkette separat definieren und priorisieren. Außerdem ist es möglich, Vor- und Nachverarbeitungsschritte zu definieren, die einmalig vor bzw. nach der Abarbeitung der Schrittkette ausgeführt werden.

Betriebsarten und Schaltmodi

Das Verhalten einer Ablaufsteuerung in **PCS 7** ist abhängig der gewählten Betriebsart, dem festgelegten Schaltmodus, ihrem aktuellen Betriebszustand und den Ablaufoptionen. Für Ablaufsteuerungen können zwei verschiedene Betriebsarten gewählt werden.

- **Auto:** Der Ablauf wird durch das Programm gesteuert.
- **Hand:** Der Ablauf wird vom Bediener durch Befehle oder Änderung der Ablaufoptionen gesteuert.

Im Handbetrieb stehen dem Bediener die Befehle *Starten, Stoppen, Anhalten, Beenden, Abbrechen, Fortsetzen, Neustarten, Rücksetzen* und *Fehler* zur Verfügung, um die Ablaufsteuerung manuell zu bedienen. Das Verhalten einer Schrittkette beim Weiterschalten von aktiven Schritten zu den Folgeschritten kann in Abhängigkeit von der gewählten Betriebsart durch verschiedene Schaltmodi gesteuert werden.

- **Schaltmodus T:** Die Ablaufsteuerung läuft prozessgesteuert, also automatisch, ab. Bei einer freigegebenen Transition werden die Vorgängerschritte deaktiviert und die Folgeschritte aktiviert. (T = Transaktionen)
- **Schaltmodus O:** Die Ablaufsteuerung läuft bedienergesteuert, also manuell ab. Die Transition wird durch einen Bedienerbefehl freigegeben. Dazu setzt jede Folge-transition eines aktiven Schritts automatisch eine Bedienanforderung. (O = Operator)
- **Schaltmodus T oder O:** Die Ablaufsteuerung läuft prozessgesteuert oder bedienergesteuert ab. Die Transition kann entweder durch einen Bedienerbefehl oder durch eine erfüllte Weiterschaltbedingung freigegeben werden.
- **Schaltmodus T und O:** Die Ablaufsteuerung läuft prozessgesteuert und bedienergesteuert ab. Die Transition wird nur freigegeben, wenn ein Bedienerbefehl vorliegt und die Weiterschaltbedingung erfüllt ist.
- **Schaltmodus T/T und O:** In diesem Schaltmodus kann für jeden Schritt einzeln festgelegt werden, ob die Ablaufsteuerung prozessgesteuert oder bedienergesteuert erfolgt. Damit können im Testbetrieb Haltepunkte in der Ablaufsteuerung definiert werden. (T/T = Test-Transaktionen)

In der Betriebsart **Auto** können nur die Schaltmodi **T** sowie **T/T und O** gewählt werden. Der Betriebszustand der Ablaufsteuerung zeigt den aktuellen Stand im Ablauf und das resultierende Betriebsverhalten an. Eine entsprechende Betriebszustandslogik definiert die möglichen Zustände, die zulässigen Übergänge zwischen den Zuständen sowie die Übergangsbedingungen für einen Zustandswechsel. **PCS 7** definiert dabei jeweils eine eigene Betriebszustandslogik für Ablaufsteuerungen und für Schrittketten. Es besteht die Möglichkeit, Schrittketten abhängig vom Zustand der Ablaufsteuerung ablaufen zu lassen.

Ablaufoptionen

Mit Hilfe von Ablaufoptionen kann das Laufzeitverhalten einer Ablaufsteuerung gesteuert werden. Es kann unter anderem festgelegt werden, ob eine Ablaufsteuerung einmalig oder zyklisch abgearbeitet wird (Option **Zyklischer Betrieb**) oder ob die Aktionen des aktiven Schrittes tatsächlich ausgeführt werden (Option **Befehlsausgabe**). Außerdem kann eine Zeitüberwachung für die einzelnen Schritte einer Schrittkette aktiviert werden, die im Falle einer Zeitüberschreitung einen Schrittfehler meldet (Option **Zeitüberwachung**).

Interaktionsverhalten

Die Interaktion zwischen CFC und SFC erfolgt in **PCS 7** über Prozesswerte und Steuerwerte. Diese Werte werden über die gewünschten Signale, entweder aus der globalen Symboltabelle oder über die Angabe der absoluten Signaladresse, miteinander verknüpft. Eine Steuerung der Verarbeitung der Steuersignale ist über die Merkmale des SFC möglich. **PCS 7** stellt in der Bibliothek **SFC Library** bereits vorgefertigte Schrittketten für verschiedene Standardszenarien zur Verfügung. Diese Vorlagen können verwendet und an aktuelle Projekte angepasst werden.

LITERATUR

- [1] Seitz, M. (2008): Speicherprogrammierbare Steuerungen. Hanser Fachbuchverlag
- [2] Wellenreuther, G. und Zastrow, D. (2002): Automatisieren mit SPS: Theorie und Praxis. Vieweg+Teubner
- [3] Uhlig, R. (2005): SPS - Modellbasierter Steuerungsentwurf für die Praxis: Modellierungsmethoden aus der Informatik in der Automatisierungstechnik. Oldenbourg Industrieverlag
- [4] Siemens (2009): Prozessleitsystem PCS 7: SFC für SIMATIC S7.

SCHRITT-FÜR-SCHRITT-ANLEITUNG

AUFGABENSTELLUNG

Entsprechend dem Rezept aus dem Kapitel ‚Prozessbeschreibung‘ soll eine SFC-Schrittfolge angelegt und programmiert werden.

1. Zuerst sollen 350 ml aus dem Edukttank =SCE.A1.T1-B003 in den Reaktor =SCE.A1.T2-R001 und gleichzeitig 200 ml aus dem Edukttank =SCE.A1.T1-B002 in den Reaktor =SCE.A1.T2-R002 abgelassen werden.
2. Ist das Füllen von dem Reaktor =SCE.A1.T2-R001 beendet, so ist die eingefüllte Flüssigkeit bei eingeschaltetem Rührer auf 25°C zu erwärmen.
3. Ist das Füllen von dem Reaktor =SCE.A1.T2-R002 beendet, so sollen 150 ml des Edukts A aus Edukttank =SCE.A1.T1-B001 in den Reaktor =SCE.A1.T2-R002 dazu dosiert werden. Ist dies abgeschlossen, so soll 10 s später der Rührer des Reaktors =SCE.A1.T2-R002 eingeschaltet werden.
4. Hat die Temperatur der Flüssigkeit in Reaktor =SCE.A1.T2-R001 25°C erreicht, so soll das Gemisch aus dem Reaktor =SCE.A1.T2-R002 in den Reaktor =SCE.A1.T2-R001 gepumpt werden.
5. Das Gemisch im Reaktor =SCE.A1.T2-R001 soll nun auf 28°C erwärmt werden und anschließend in den Produkttank =SCE.A1.T3-B001 abgelassen werden.

LERNZIEL

In diesem Kapitel lernt der Studierende:

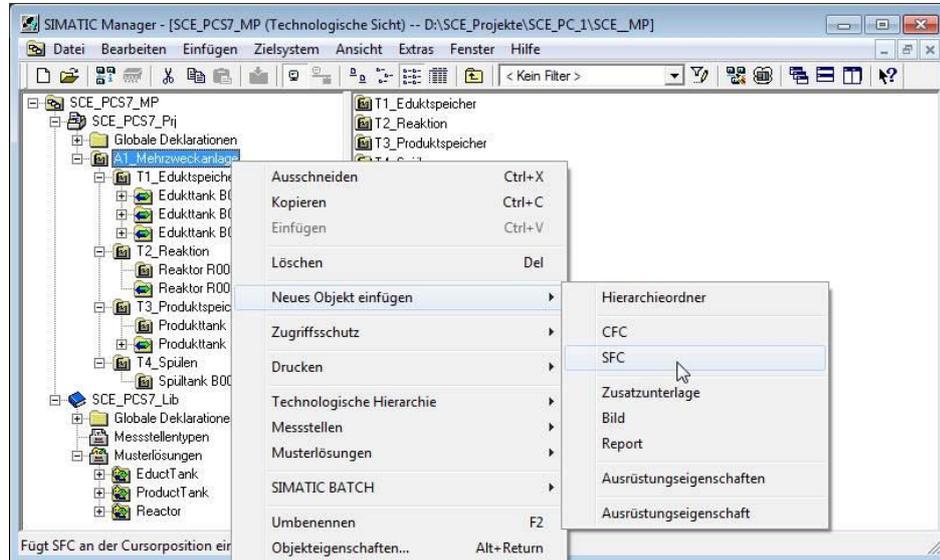
- SFC- Schrittfolgen anlegen und bearbeiten
- Verknüpfungen herstellen zwischen SFC-Schrittfolgen und CFC-Plänen
- Verknüpfungen herstellen zwischen SFC-Schrittfolgen und den Operanden aus der Symboltabelle
- Testen von Schrittfolgenprogrammen

Diese Anleitung baut auf dem Projekt ‚PCS7_SCE_0108_Ueb_R1505.zip‘ auf.

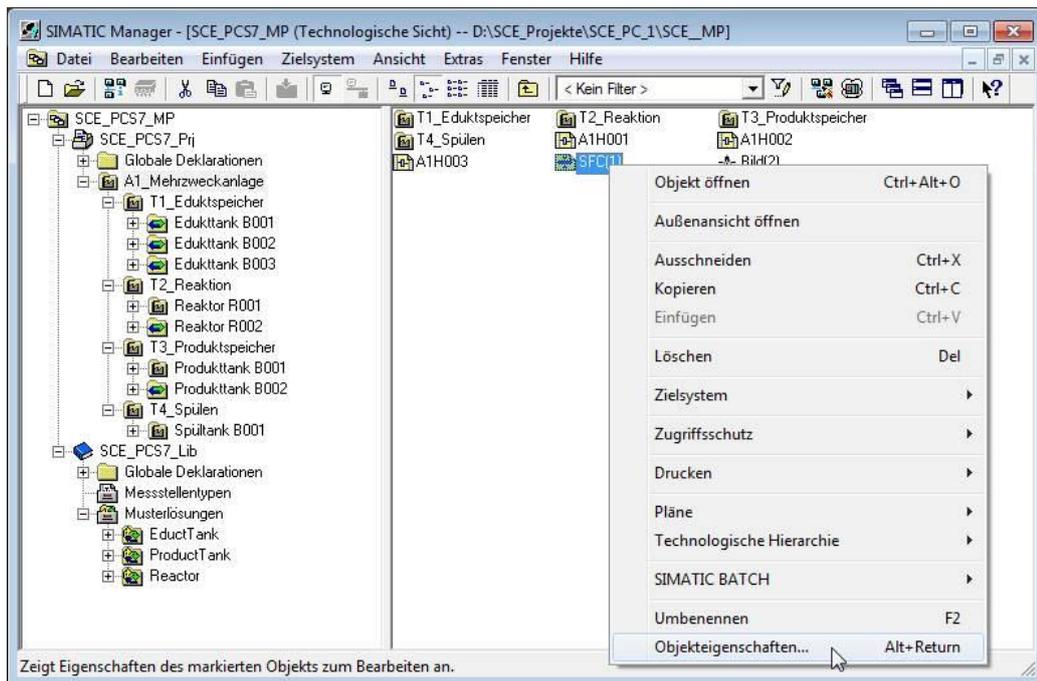
PROGRAMMIERUNG

1. Zu Beginn legen Sie in der Technologischen Sicht in dem Ordner ‚A1_Mehrzweckanlage‘ einen neuen SFC an.

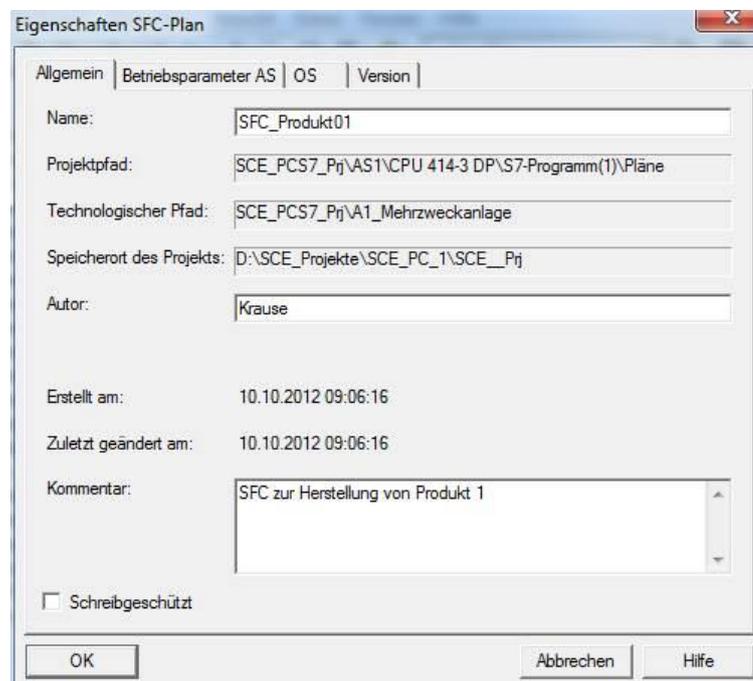
(→ A1_Mehrzweckanlage → Neues Objekt einfügen → SFC)



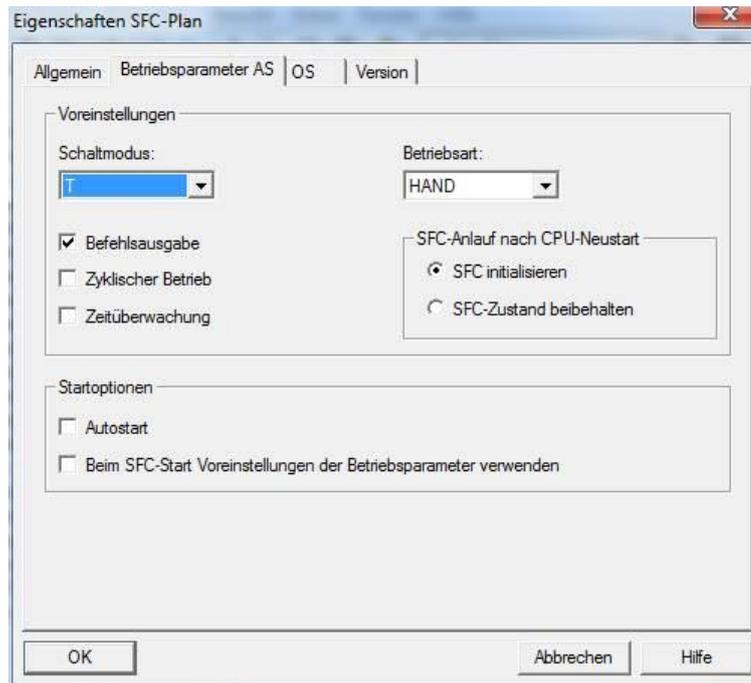
2. Nun werden die Objekteigenschaften des SFC geöffnet.
(→ SFC(1) → Objekteigenschaften)



3. Unter Allgemein wird der Name auf ‚SFC_Produkt01‘ geändert, ein Kommentar sowie der Autor eingetragen.
(→ Allgemein → SFC_Produkt01)

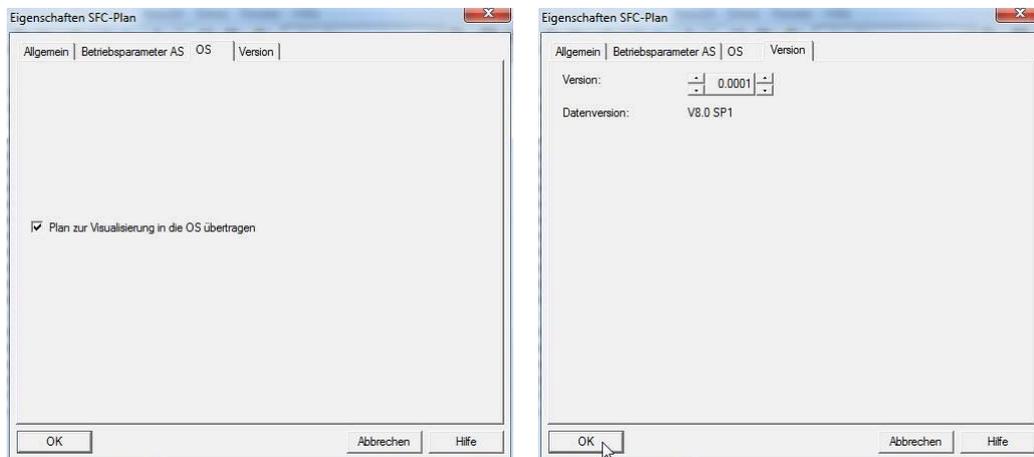


4. Die Betriebsparameter stellen Sie wie folgt ein, wobei diese auch später noch im Online-Modus geändert werden können (→ Betriebsparameter AS)

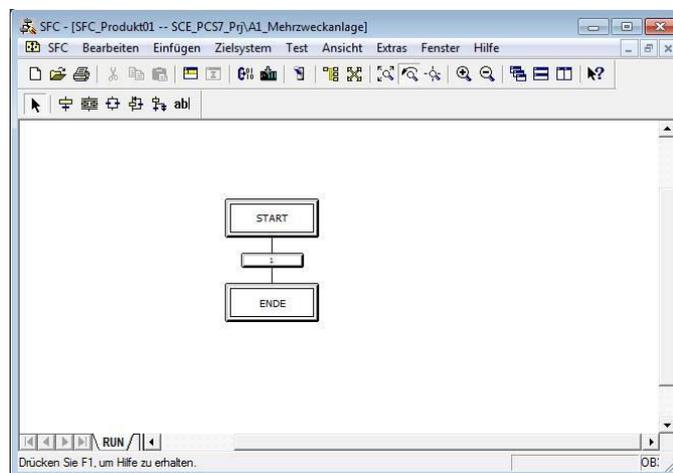
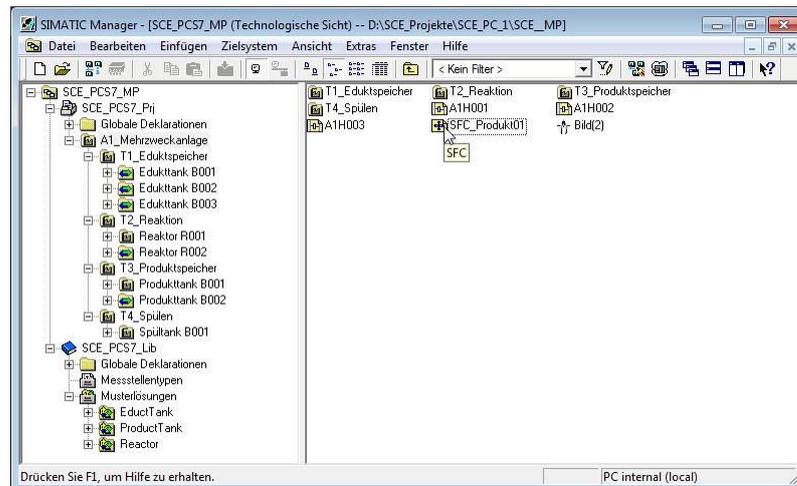


5. Bei dem Punkt OS ist es wichtig, dass der Haken gesetzt ist, damit der SFC später auch in der Visualisierung zur Verfügung steht. Bei der Anzeige der Version übernehmen Sie jetzt sämtliche Parameter mit OK.

(→ OS → Plan zur Visualisierung in die OS übertragen → Version → OK)



6. Nun wird die Schrittkette ‚SFC_Produkt01‘ im **SIMATIC Manager** mit einem Doppelklick geöffnet. (→ SFC_Produkt01)



7. Im SFC- Editor haben Sie jetzt die Möglichkeit, mit den folgenden Symbolen aus der Werkzeugleiste die Ablaufsteuerung aufzubauen.



Schaltfläche **Selektieren einschalten**



Schaltfläche **Schritt + Transition einfügen**



Schaltfläche **Parallelzweig einfügen**



Schaltfläche **Alternativzweig einfügen**



Schaltfläche **Schleife einfügen**

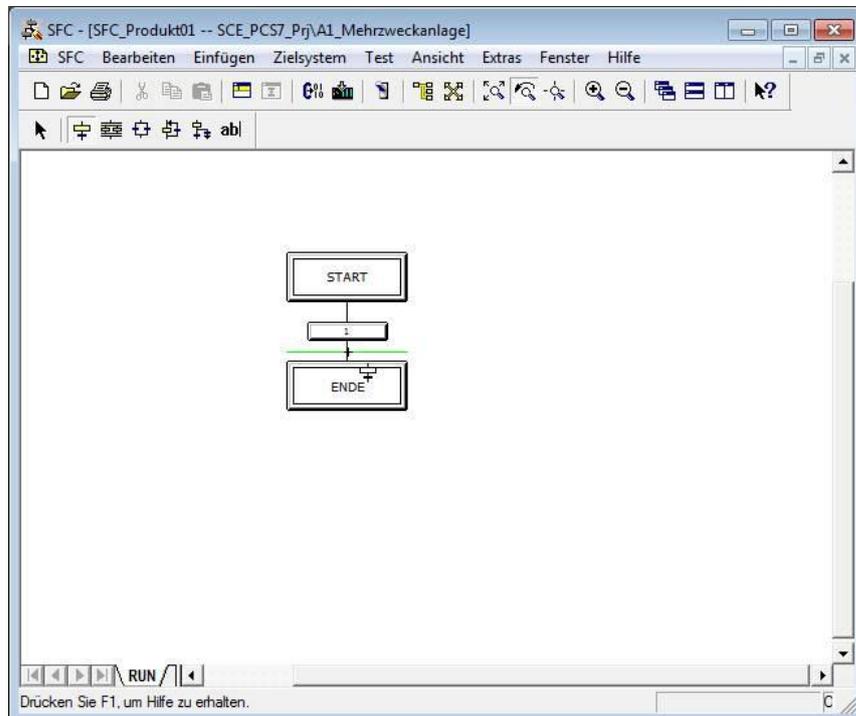
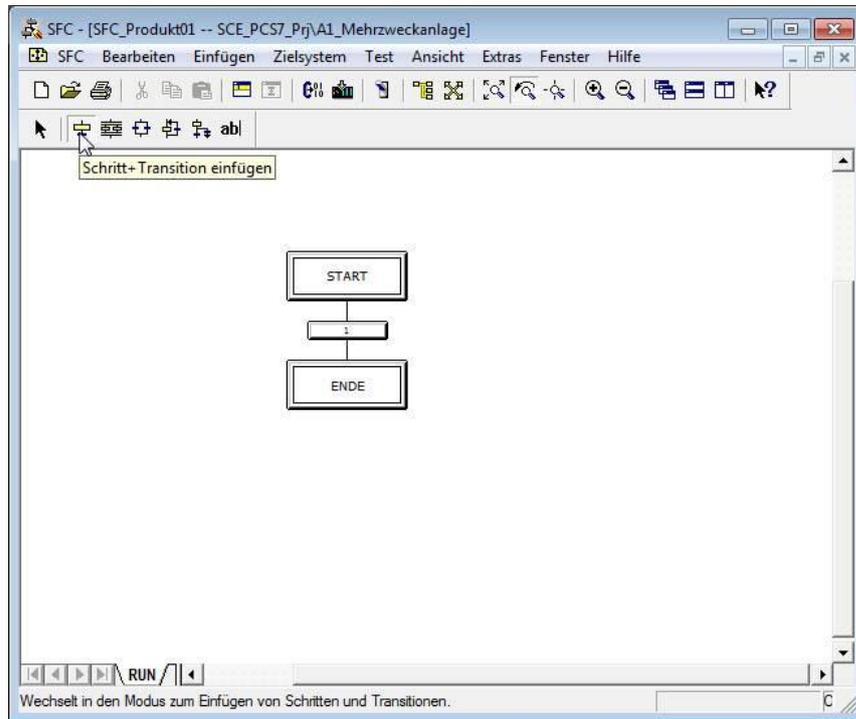


Schaltfläche **Sprung einfügen**



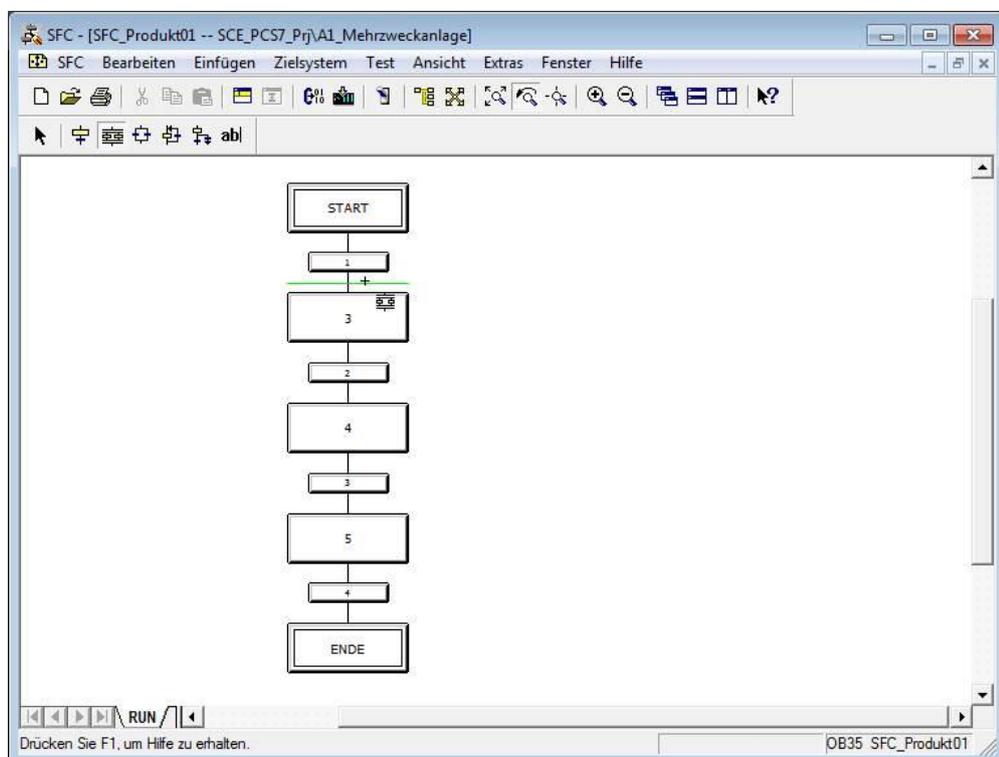
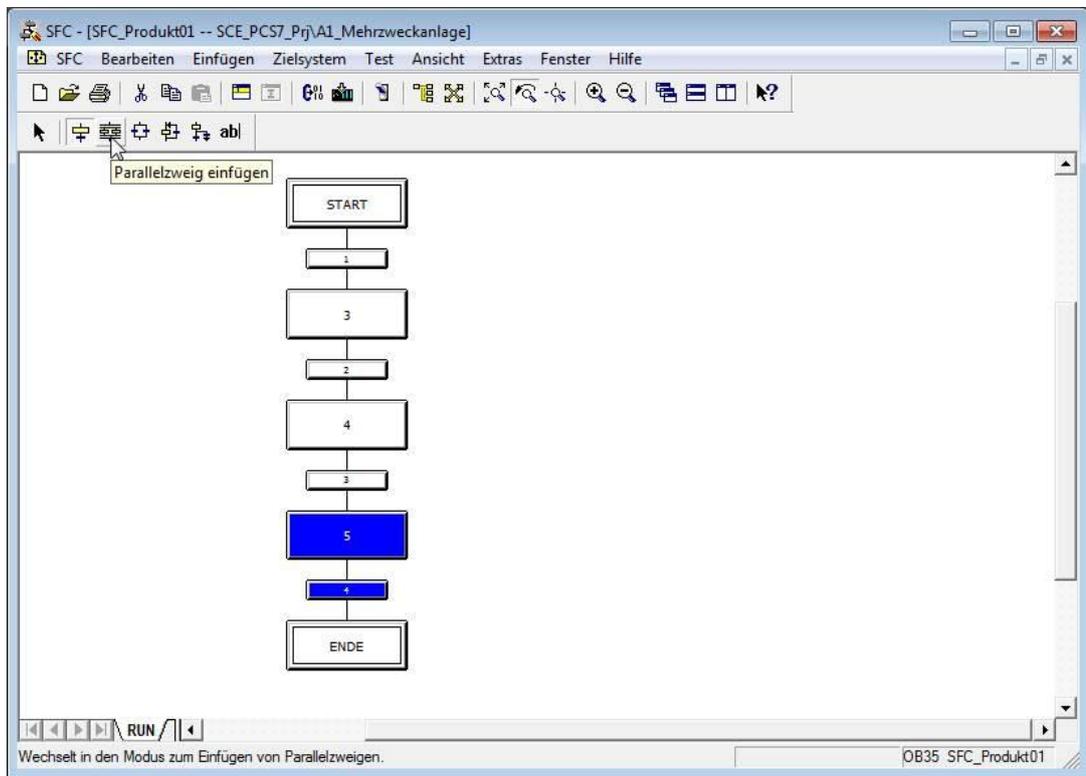
Schaltfläche **Textfeld einfügen**

8. Für die Aufgabe benötigen Sie weitere Schritte und Transitionen. Um beides einzufügen wählen Sie die Schaltfläche  und markieren danach die Stelle wo Sie diese einfügen wollen. (→ )

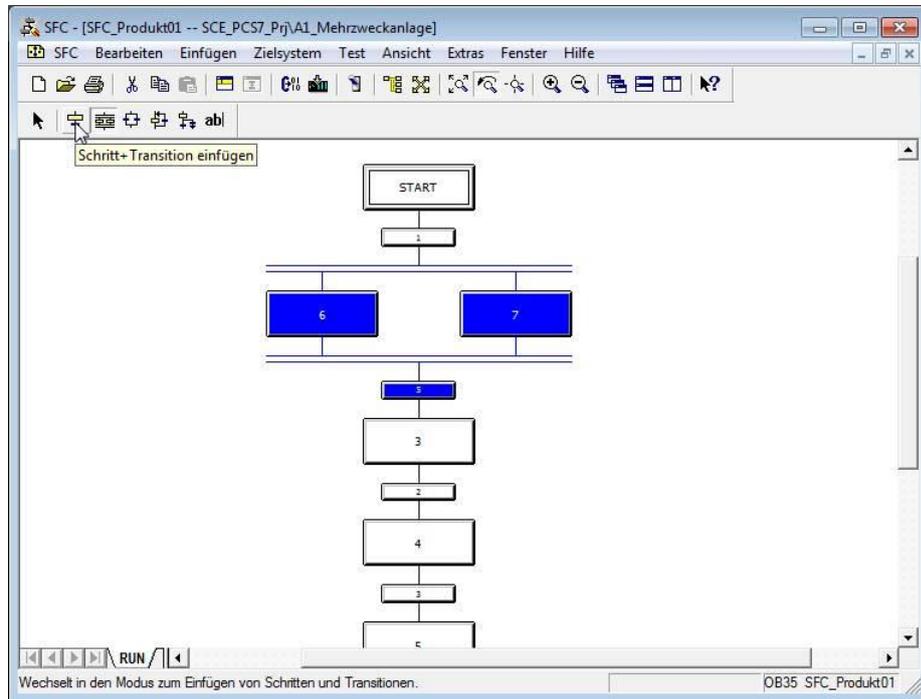


Hinweis: Die Nummerierung der Schritte und Transitionen hat für die Reihenfolge der Abarbeitung der Schrittfolge keine Bedeutung.

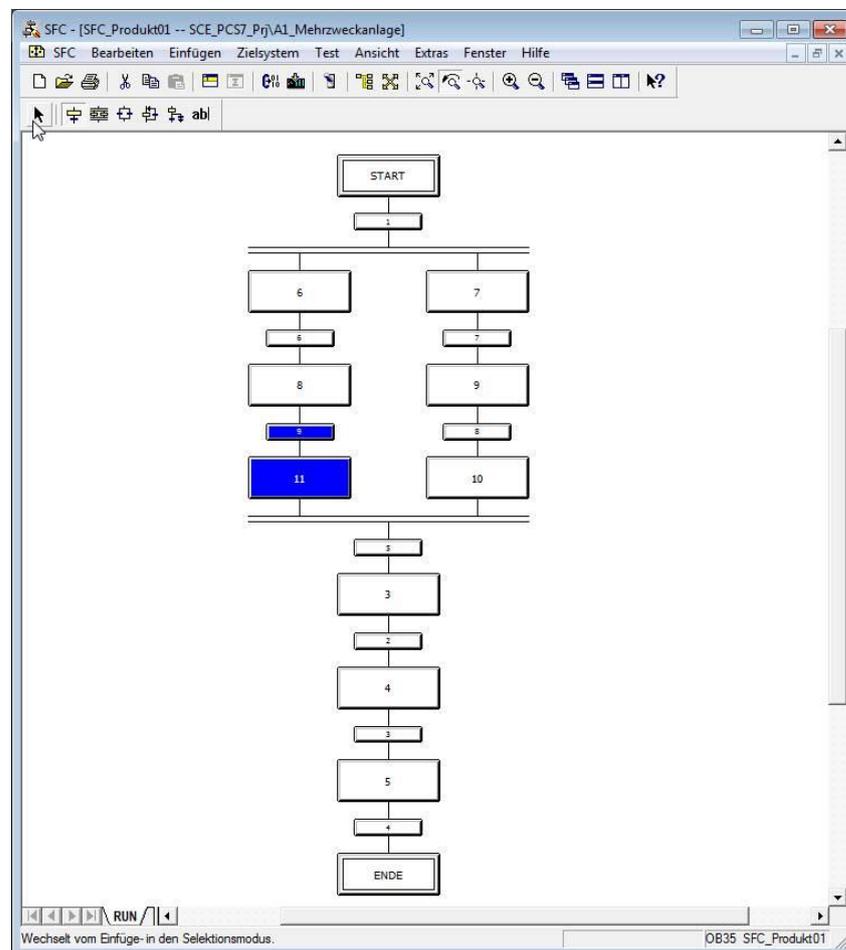
9. Nachdem auf diese Art drei Schritte und Transitionen eingefügt wurden, klicken Sie auf das Symbol  um eine parallele Verzweigung einzufügen. Markieren Sie daraufhin wieder die Stelle wo Sie diese einfügen wollen. (→ )



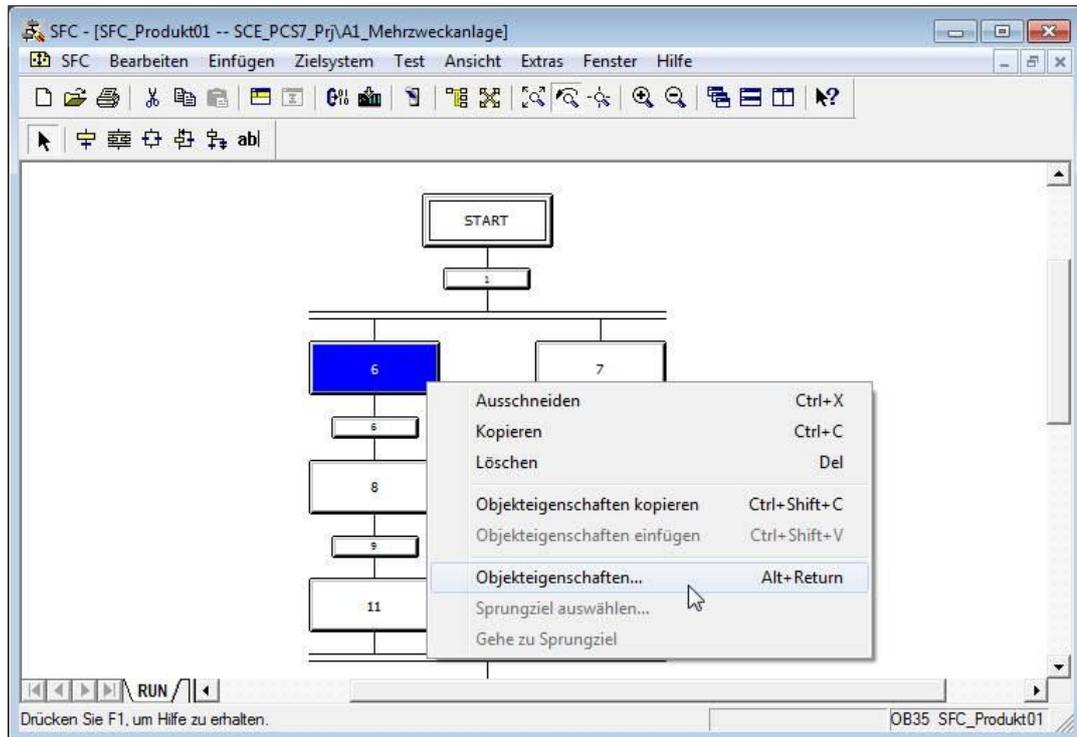
10. In die Parallelverzweigung sollen nun noch weitere Schritte und Transitionen eingefügt werden. Wechseln Sie deshalb wieder zum Symbol  und fügen Sie die weiteren Schritte und Transitionen ein. (→ )



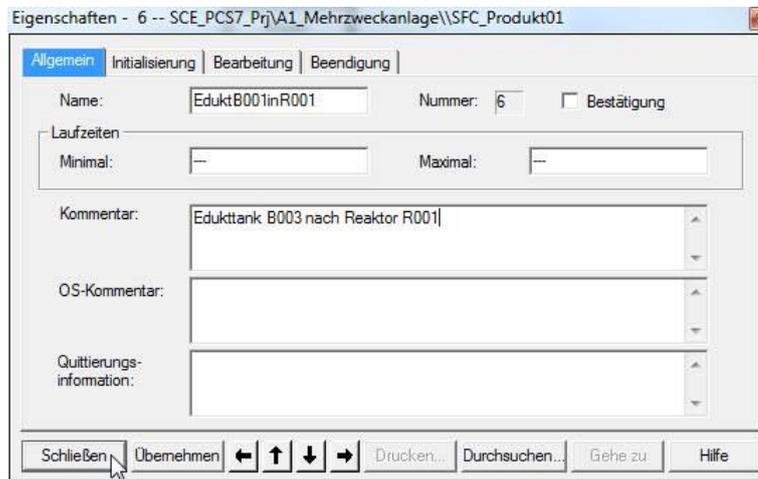
11. Klicken Sie nun auf das Symbol  um normal editieren zu können. (→ )



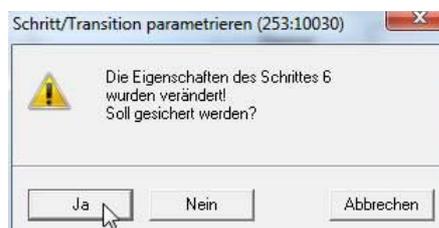
12. Nun wird zuerst gezeigt, wie die Eigenschaften eines Schrittes verändert werden können. Dazu klicken Sie mit der rechten Maustaste auf den Schritt und wählen Objekteigenschaften aus. (→ 6 → Objekteigenschaften)



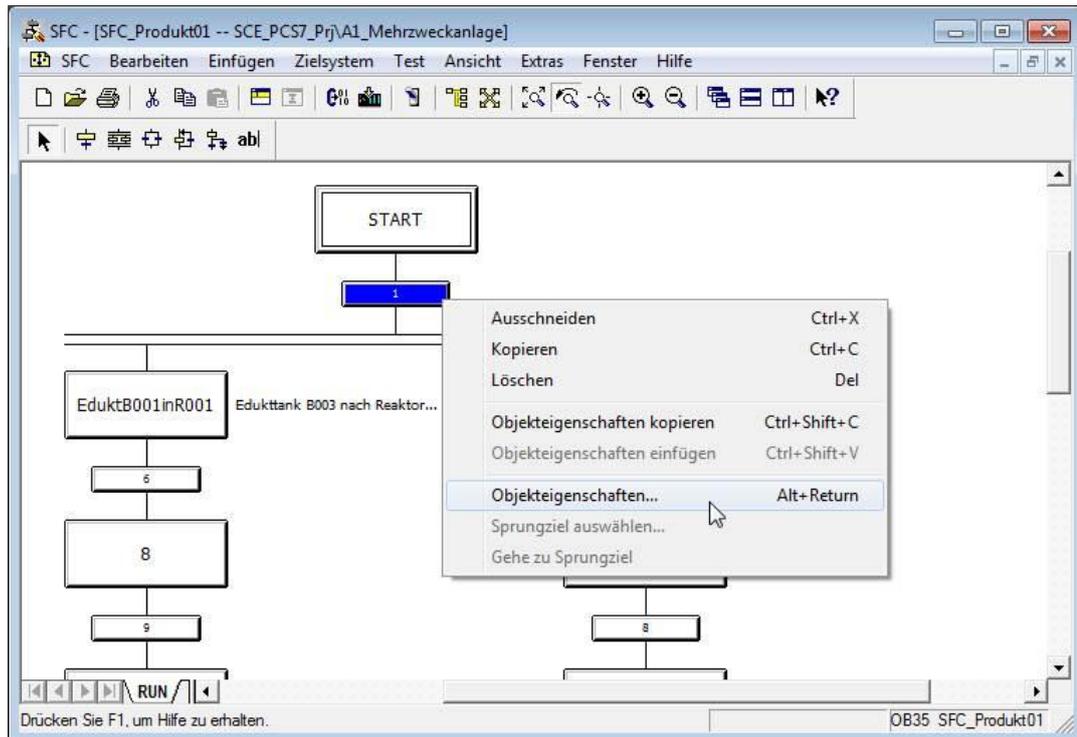
13. In den Objekteigenschaften soll zur besseren Übersichtlichkeit jedem Schritt ein Name und ein Kommentar gegeben werden. (→ EduktB003inR001 → Edukttank B003 nach Reaktor R001 → Schließen)



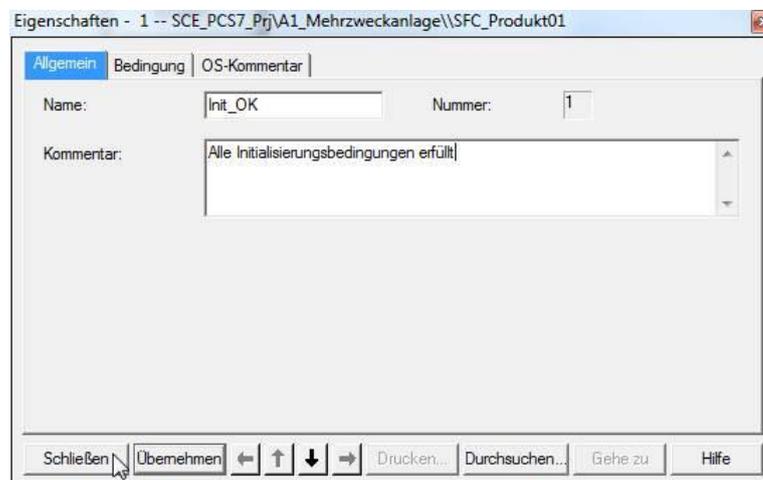
14. Die Frage ob die Änderungen gespeichert werden sollen bestätigen Sie mit „Ja“.
(→ Ja)



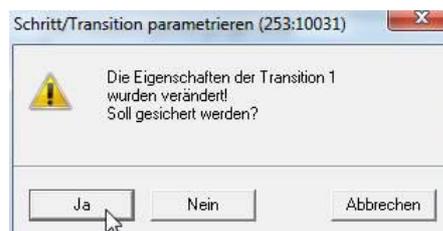
15. Wie bei den Schritten, so können auch bei den Transitionen die Eigenschaften verändert werden. Dazu klicken Sie mit der rechten Maustaste auf die Transition und wählen Objekteigenschaften aus. (→ 1 → Objekteigenschaften)



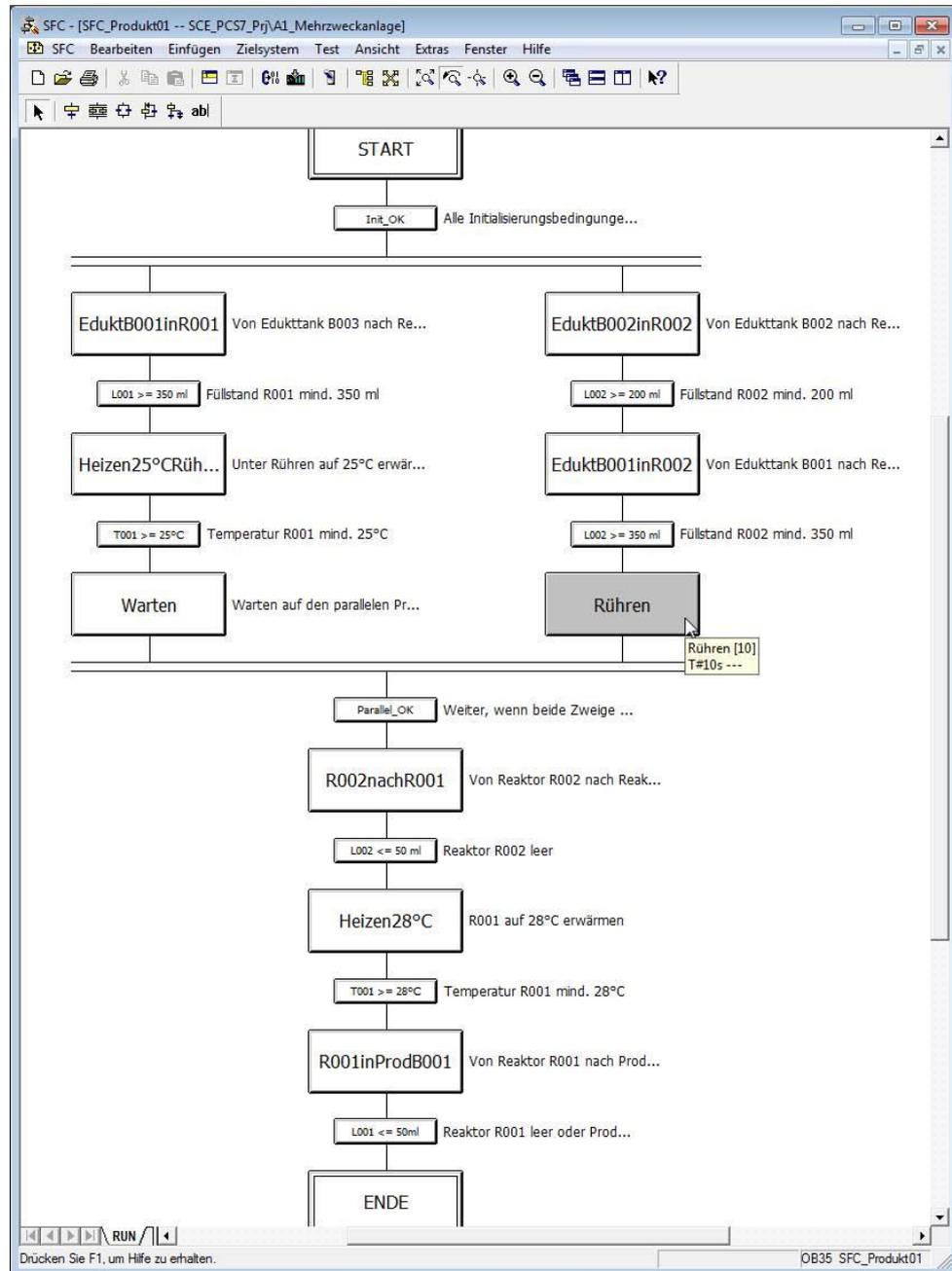
16. Auch hier werden zuerst nur der Name und der Kommentar geändert. (→ Init_OK → Alle Initialisierungsbedingungen erfüllt → Schließen)



17. Die Änderung wird wieder gesichert. (→ Ja)



18. Die vorhergehenden Schritte wiederholen Sie bis der SFC folgendermaßen aussieht. Wichtig ist es, bei dem Schritt ‚Rühren‘ auch eine minimale Laufzeit von 10 Sekunden einzutragen. (→ T#10s)

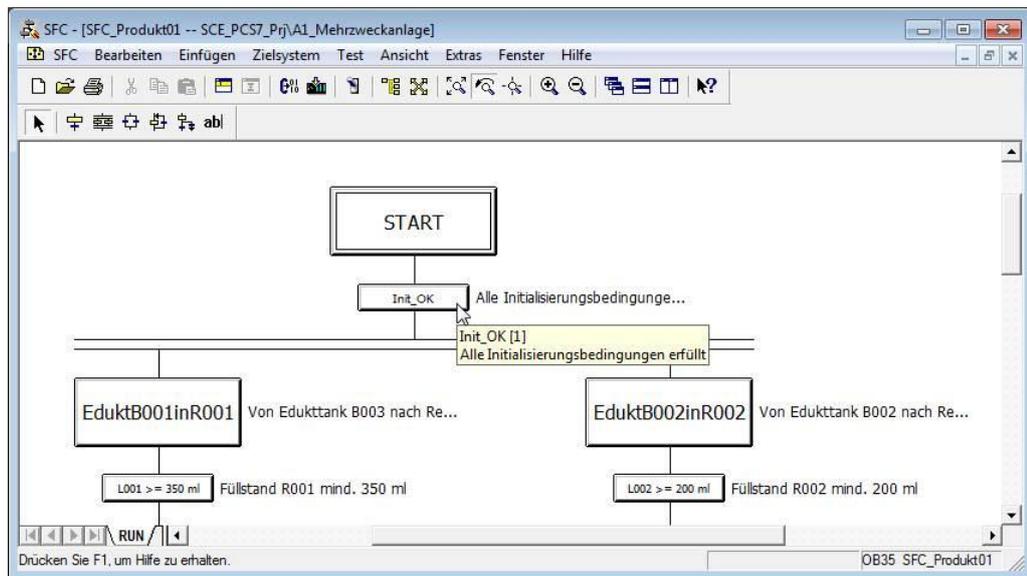


The screenshot shows the 'Eigenschaften - Rühren' dialog box for the 'Rühren' step. The 'Allgemein' tab is active, showing the following configuration:

- Name:** Rühren
- Nummer:** 10
- Bestätigung
- Laufzeiten:**
 - Minimal: T#10s
 - Maximal: ---
- Kommentar:** (Empty text area)
- OS-Kommentar:** (Empty text area)
- Quittierungs-information:** (Empty text area)

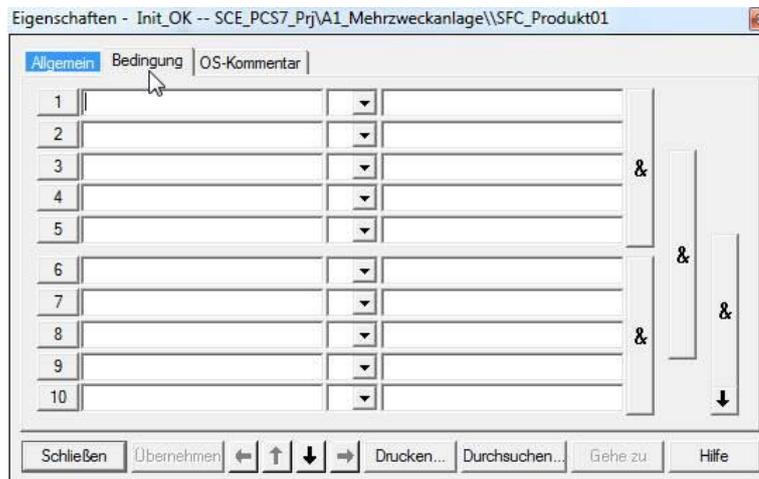
At the bottom, there are buttons: Schließen, Übernehmen, navigation arrows, Drucken..., Durchsuchen..., Gehe zu, and Hilfe.

19. Nun geht es darum die eigentliche Funktion der Schrittkette zu realisieren. In den Schritt ‚START‘ sollen keine Anweisungen kommen. Deshalb beginnen Sie mit einem Doppelklick auf die Transition ‚Init_OK‘. (→ Init_OK)

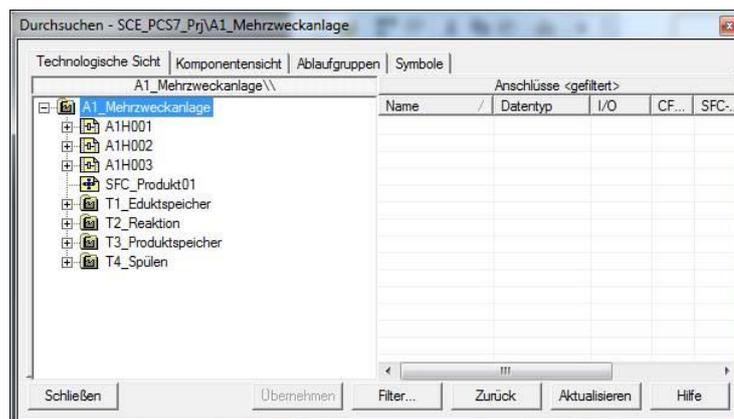


20. Wählen Sie das Register ‚Bedingung‘ aus und fügen Sie nun die Initialisierungsbedingungen hinzu, indem Sie auf ‚Durchsuchen‘ klicken.

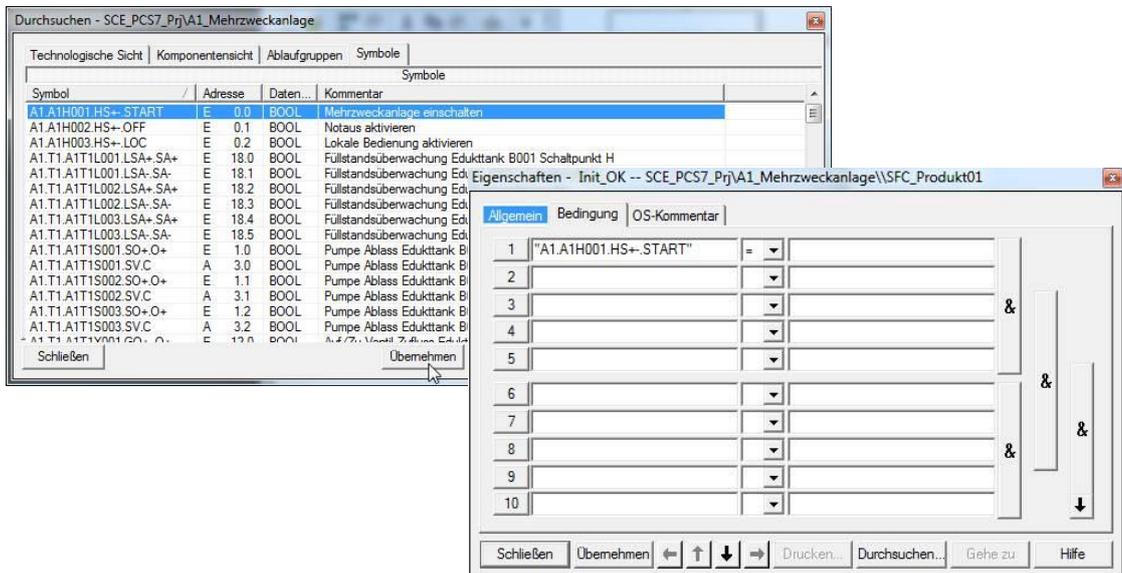
(→ Bedingung → Durchsuchen)



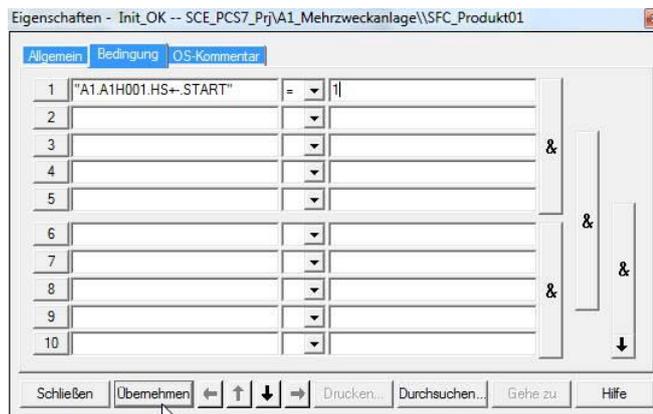
21. Es öffnet sich ein Fenster zum Hinzufügen von Anschlüssen oder Symbolen.



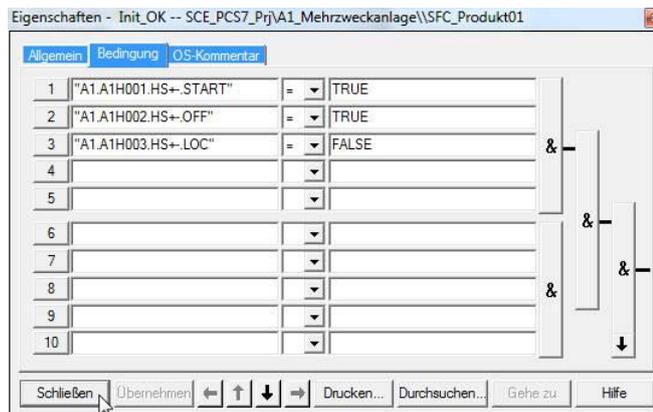
22. Wählen Sie nun das Register ‚Symbole‘ und selektieren Sie dort das Symbol des Hauptschalters ‚A1.A1H001.HS+-.START‘ und klicken Sie auf ‚Übernehmen‘. Das Symbol wird auf der linken Seite der ersten Bedingung eingetragen. (→ Symbol → A1.A1H001.HS+-.START → Übernehmen)



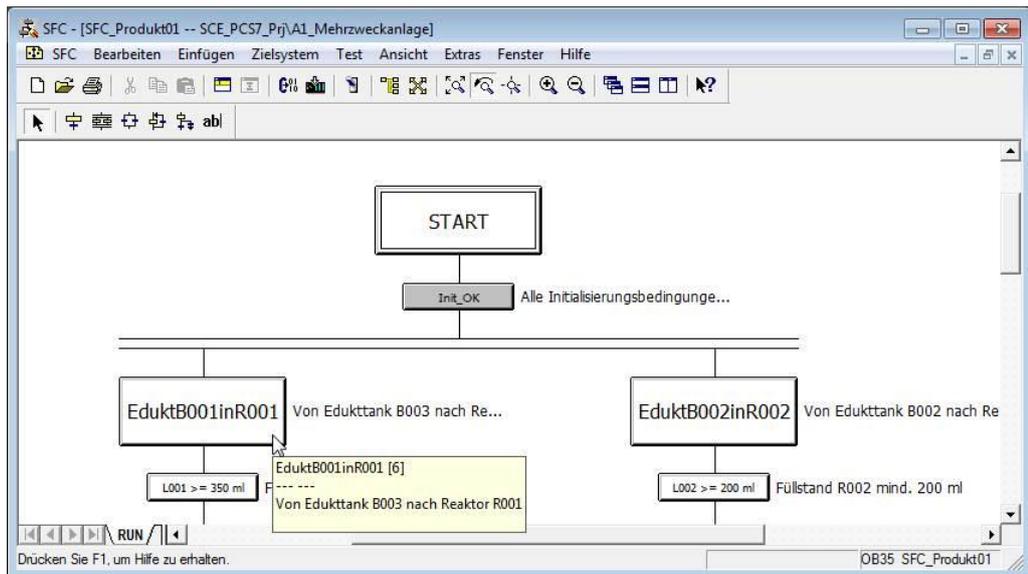
23. Tragen Sie nun auf die rechte Seite der ersten Bedingung ‚1‘ oder ‚TRUE‘ ein, damit die nächsten Schritte nur bei eingeschalteter Anlage abgearbeitet werden. Übernehmen Sie diesen Wert. (→ 1 → Übernehmen)



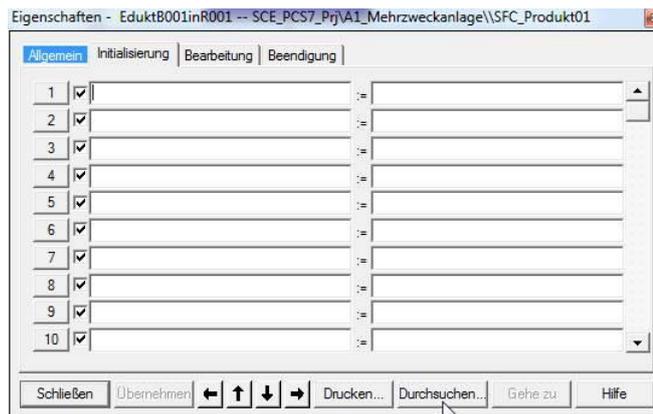
24. Fügen Sie nun noch die Bedingungen hinzu, dass das NOTAUS entriegelt und die lokale Bedienung deaktiviert ist. Schließen Sie anschließend den Dialog. (→ A1.A1H002.HS+-.OFF → 1 → A1.A1H003.HS+-.LOC → 0 → Schließen)



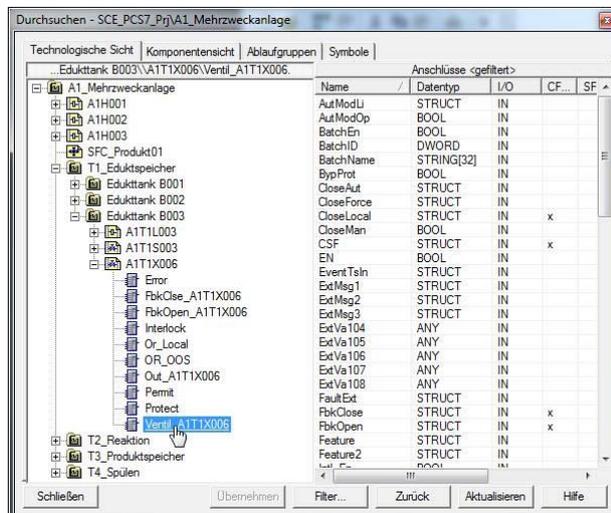
25. Öffnen Sie nun den Schritt ‚EduktB001inR001‘. (→ EduktB001inR001)



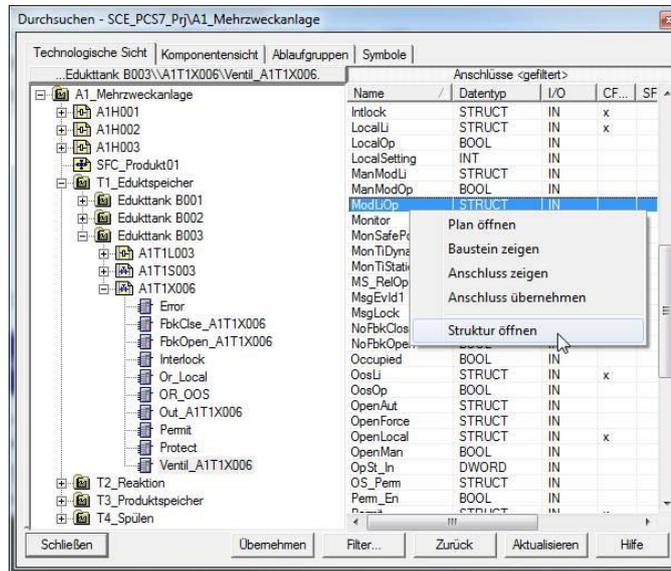
26. Wählen Sie das Register ‚Initialisierung‘ aus und klicken Sie auf ‚Durchsuchen‘. (→ Initialisierung → Durchsuchen)



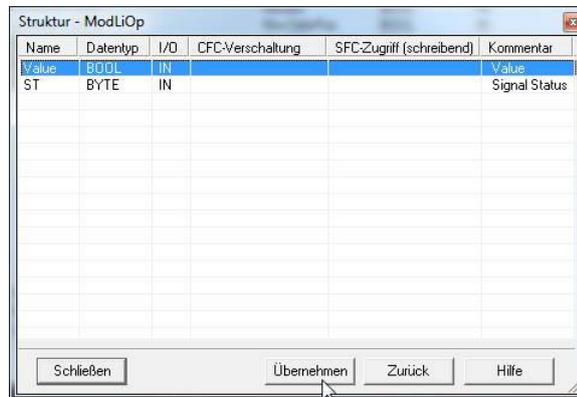
27. Danach wählen Sie im Auswahlfenster im Register ‚Technologischen Sicht‘ im CFC ‚A1T1X006‘ den Ventilbaustein ‚Ventil_A1T1X006‘ aus. (→ A1_Mehrweckanlage → T1_Eduktspeicher → Edukttank B003 → A1T1X006 → Ventil_A1T1X006)



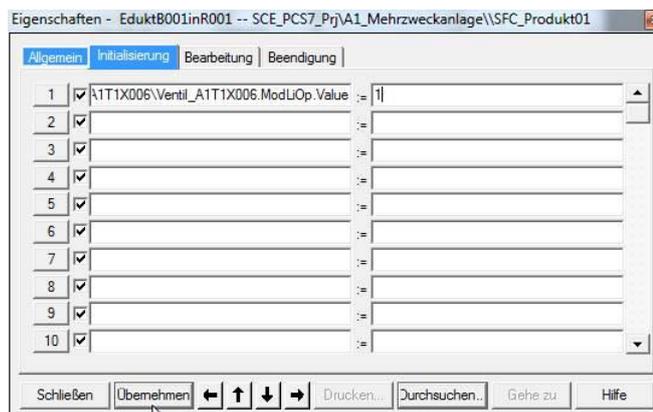
28. Setzen Sie zuerst den Anschluss ‚ModLiOp‘ auf ‚1‘, damit das Ventil nur noch über Verschaltungen oder SFC gesteuert werden kann. Da der Anschluss ‚ModLiOp‘ vom Datentyp ‚STRUCT‘ ist, müssen Sie zunächst per Rechtsklick das Kontextmenü öffnen und dort ‚Struktur öffnen‘ anklicken. (→ ModLiOp → Struktur öffnen)



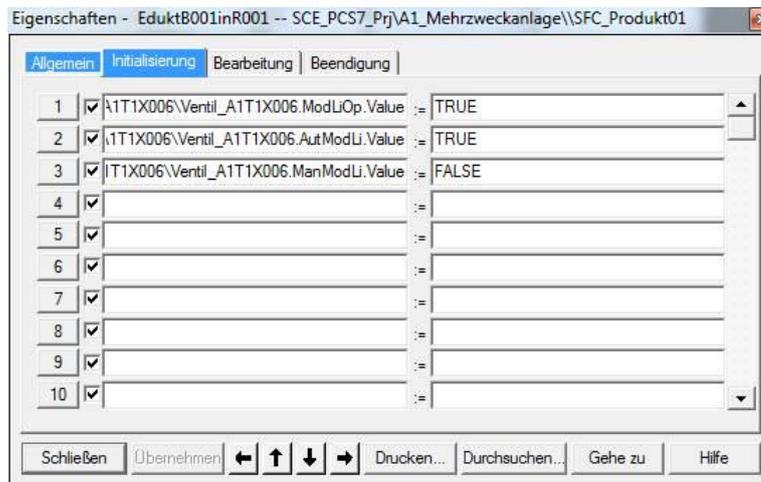
29. Der Strukturdialog öffnet sich und Sie wählen ‚Value‘ vom Datentyp BOOL aus. Mit ‚übernehmen‘ wird Ihre Auswahl auf die linke Seite der ersten Anweisung übernommen. (→ Value → Übernehmen)



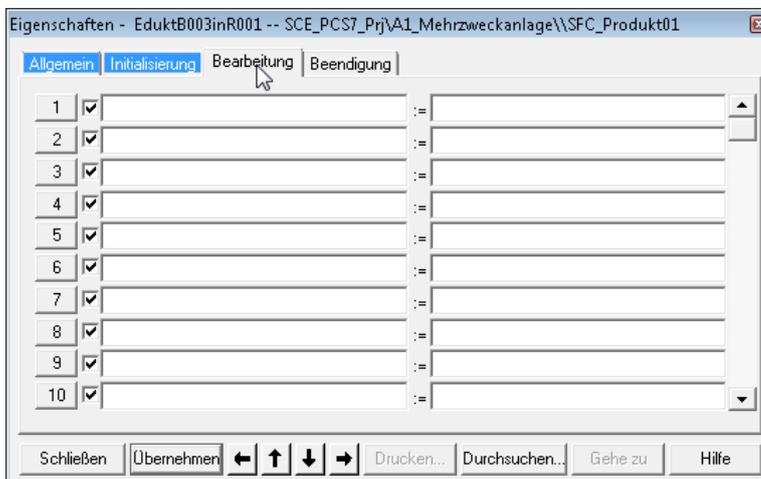
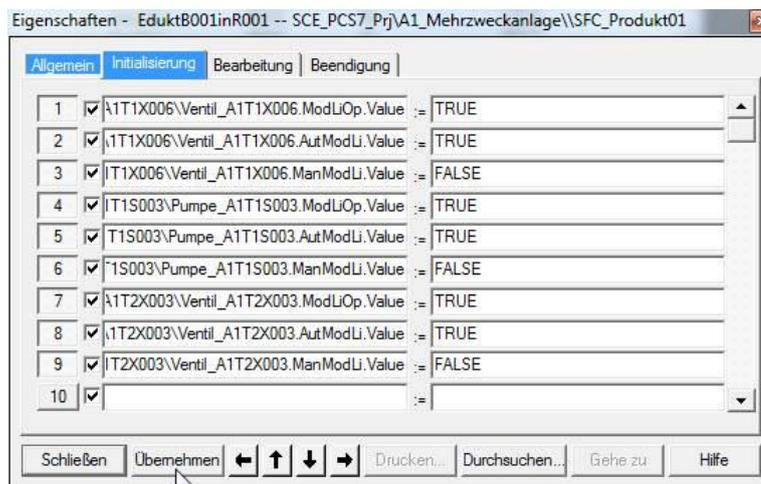
30. Auf der rechten Seite der ersten Anweisung tragen Sie nun eine ‚1‘ ein. Damit wird der Anschluss ‚ModLiOp‘ in den SFC-Modus gesetzt. Mit ‚Übernehmen‘ wird die ‚1‘ automatisch durch ‚TRUE‘ ersetzt. (→ 1 → Übernehmen)



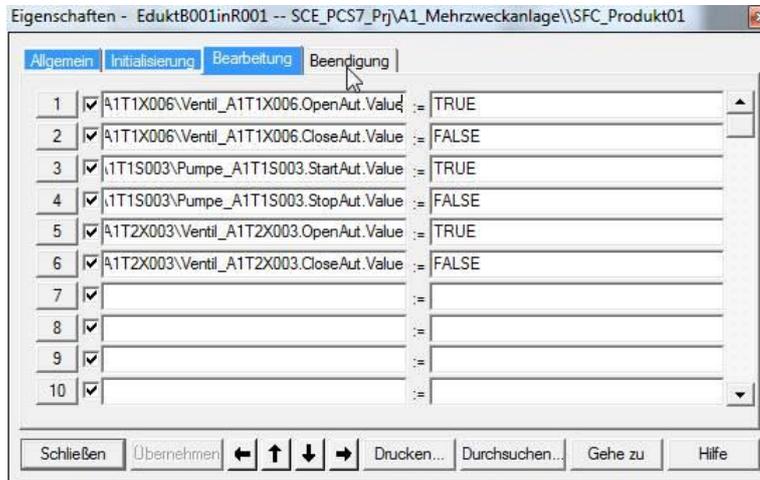
31. Nun fügen Sie die Anschlüsse ‚AutModLi‘ = ‚1‘ und ‚ManModLi‘ = ‚0‘ hinzu, damit das Ventil in den Automatikbetrieb gesetzt wird. (→ AutModLi → 1 → ManModLi → 0 → Übernehmen)



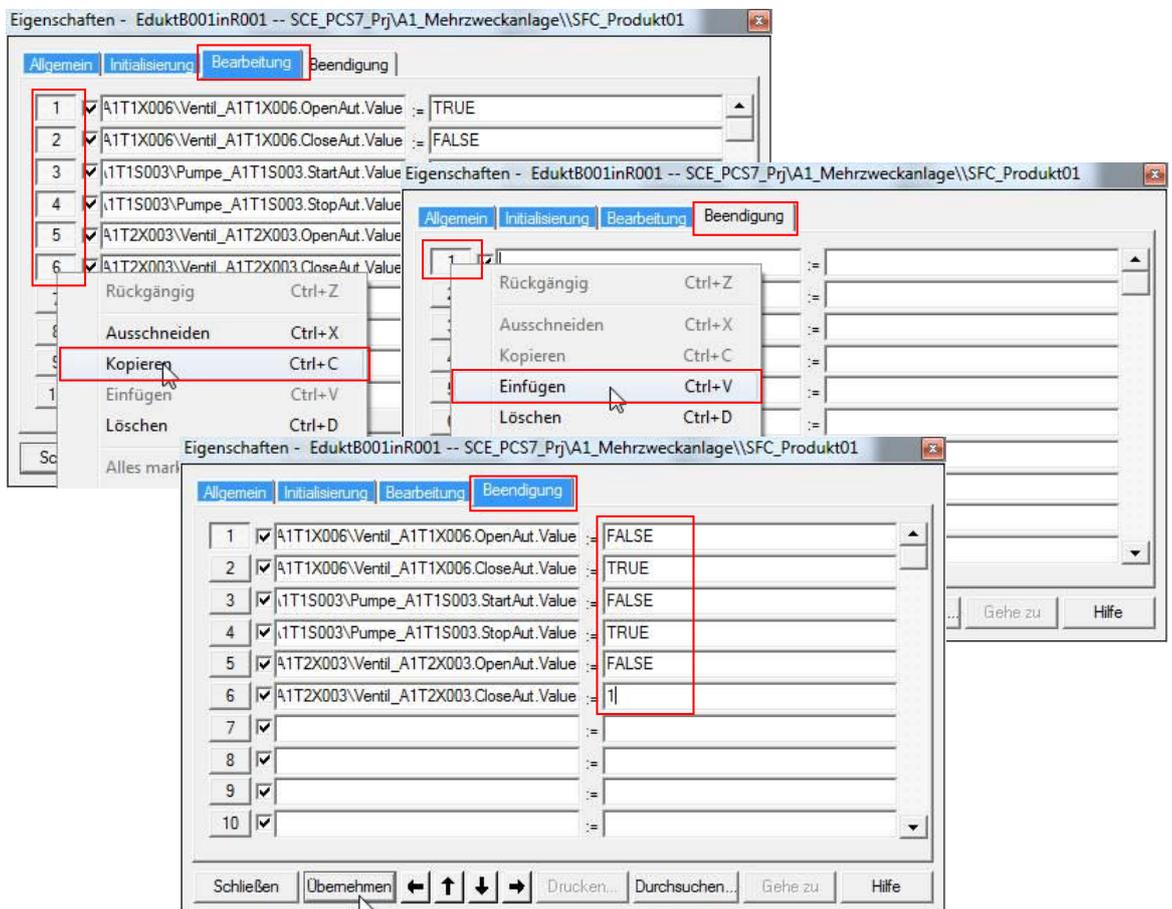
32. Dasselbe muss jetzt für die Pumpe A1T1S003 und das Ventil A1T2X003 gemacht werden, da diese ebenfalls am Befüllen von Reaktor R001 aus Edukttank B003 beteiligt sind. Anschließend wechseln Sie auf das Register ‚Bearbeitung‘. (A1T1S003 → ModLiOp.Value = 1 → AutModLi.Value = 1 → ManModLi.Value = 0 → Übernehmen → A1T2X003 → ... → Übernehmen → Bearbeitung)



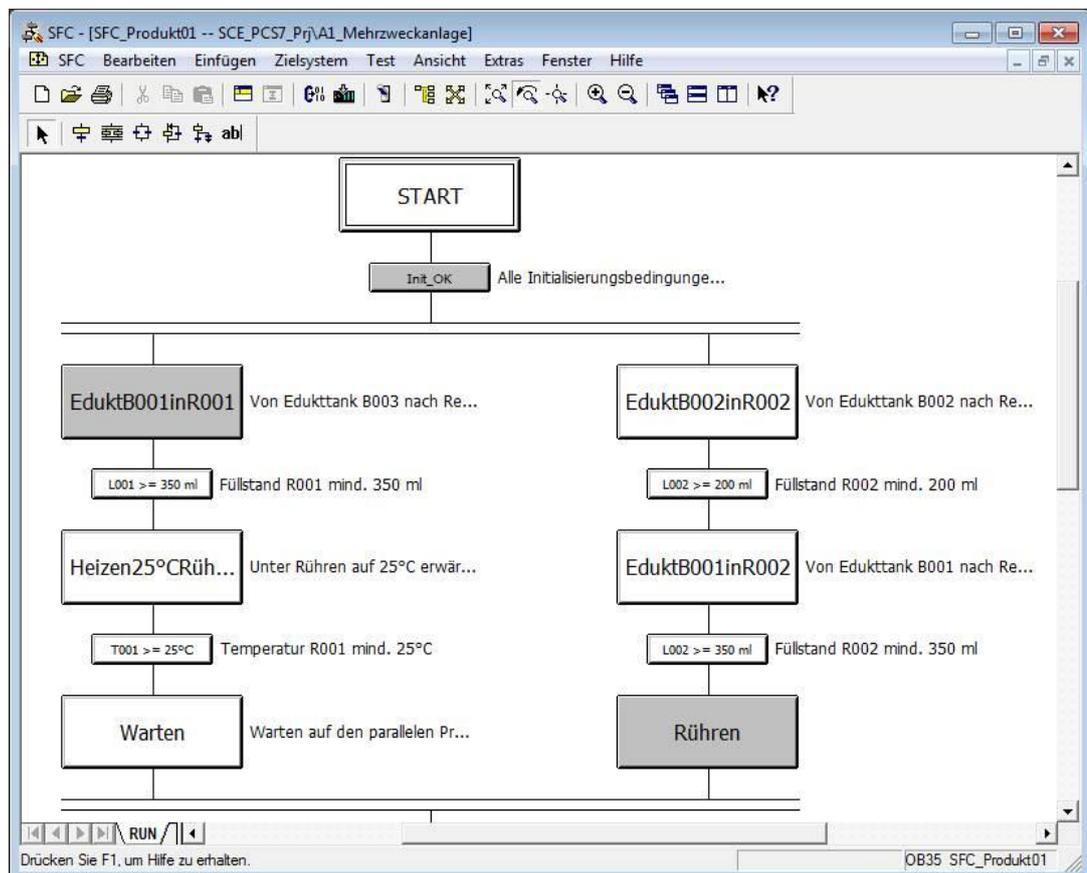
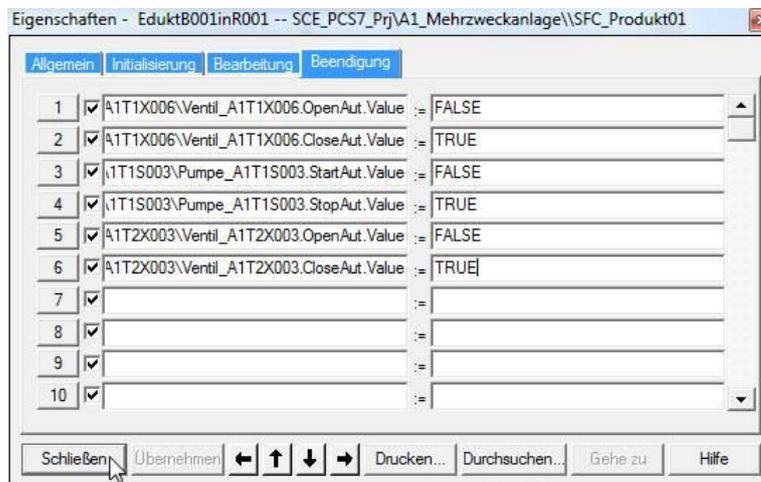
33. In ‚Bearbeitung‘ werden nun die Anweisungen zum Öffnen der Ventile und Starten der Pumpe eingetragen. Bei den Ventilen werden die Anschlüsse ‚OpenAut.Value‘ = ‚1‘ und ‚CloseAut.Value‘ = ‚0‘ gesetzt. Bei der Pumpe nutzen Sie die Anschlüsse ‚StartAut.Value‘ = ‚1‘ und ‚StopAut.Value‘ = ‚0‘. (A1T1X006 → ... → A1T1S003 → ... → A1T2X003 → ... → Übernehmen → Beendigung)



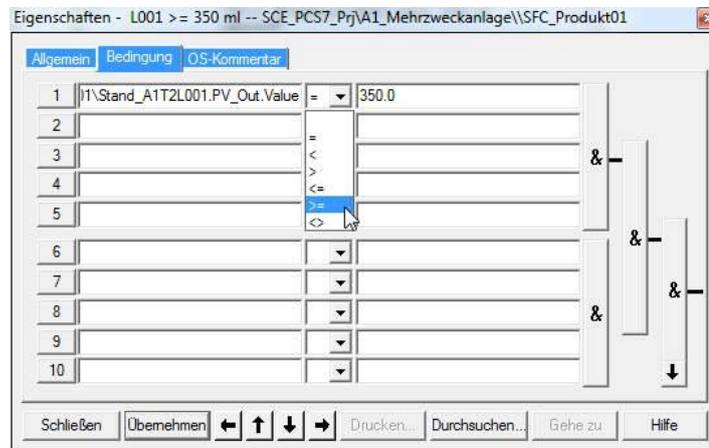
34. Nun werden die Anweisungen, die beim Beenden des Schrittes ausgeführt werden sollen in ‚Beendigung‘ eingetragen. Sie müssen hier die Ventile und die Pumpe wieder schließen. Hier könnten die Ventile und die Pumpe auch wieder in den manuellen Modus und den Operatorbetrieb zurückgesetzt werden, aber Sie heben sich das für den Schritt ‚ENDE‘ auf. Am einfachsten, Sie kopieren dabei die Anweisung von ‚Bearbeiten‘ nach ‚Beendigung‘ und invertieren anschließend nur noch die Werte (‚TRUE‘ wird ‚FALSE‘ und umgekehrt). Zum Kopieren und Einfügen müssen Sie die Zahlen vor den Anweisungen markieren und danach das Kontextmenü aufrufen.



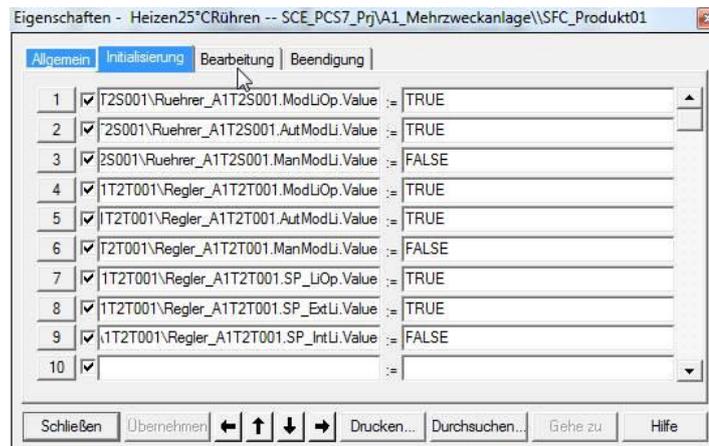
35. Schließen Sie nun den Eigenschaftendialog vom Schritt ‚EduktB003inR001‘. Im SFC-Editor sehen Sie, dass die Transition ‚Init_OK‘, die Schritte ‚EduktB003inR001‘ und ‚Rühren‘ grau hinterlegt sind, weil dort bereits Anweisungen vorhanden sind. (→ Schließen)



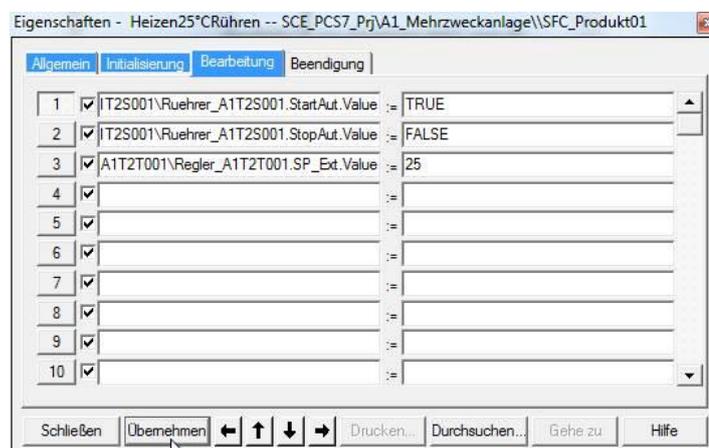
36. Nun öffnen Sie die Transition ‚L001 >= 350 ml‘. Tragen Sie die Bedingung ein, dass der Füllstand von Reaktor R001 größer bzw. gleich 350 ml ist. (→ L001 >= 350 ml → Bedingung → Durchsuchen → ...Reaktor R001\A1T2L001\Stand_A1T2L001.PV_Out → Rechtsklick → Struktur öffnen → Value → >= → 350 → Übernehmen → Schließen)



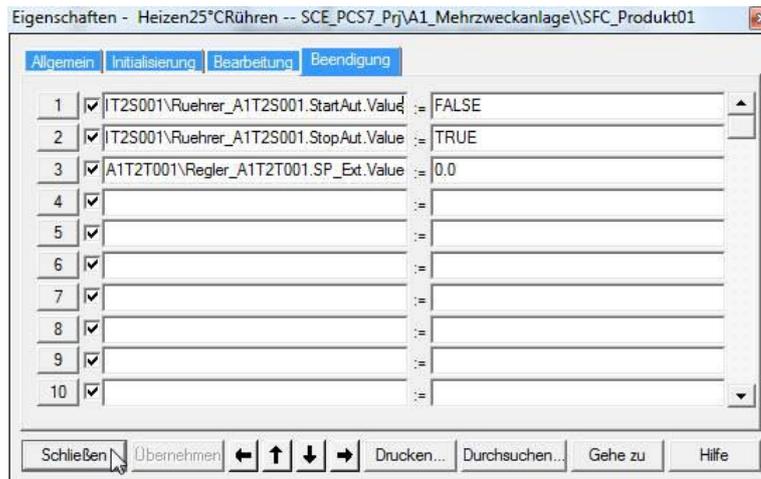
37. Im Schritt ‚Heizen25°CRühren‘ fügen Sie in der ‚Initialisierung‘ wieder die Anschlüsse ‚ModLiOp‘, ‚AutModLi‘ und ‚ManModLi‘ für den ‚Ruehrer_A1T2S001‘ und den ‚Regler_A1T2T001‘ hinzu. Für den Regler schalten Sie nun noch die Sollwertvorgabe auf SFC-Modus ‚SP_LiOp‘ = ‚1‘ und auf externe Sollwertvorgabe ‚SP_ExtLi‘ = ‚1‘ und ‚SP_IntLi‘ = ‚0‘. (→ Heizen25°CRühren → ‚Initialisierung‘ → ...)



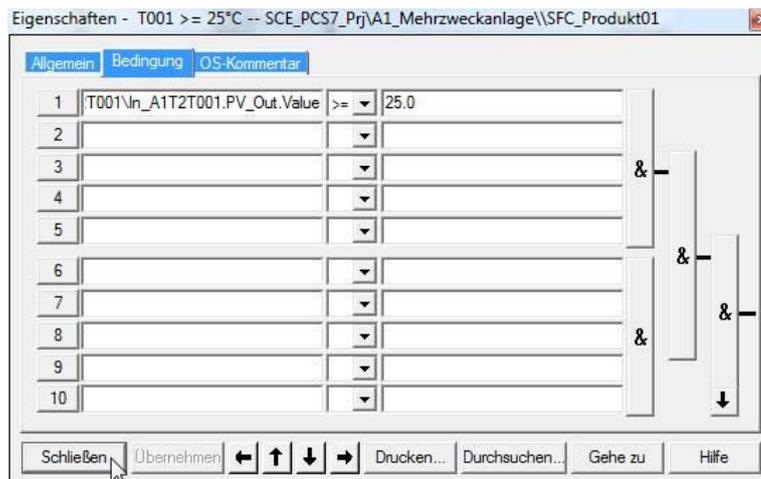
38. Wechseln Sie nun zu ‚Bearbeitung‘ und fügen Sie die abgebildeten Anschlüsse und Werte hinzu. Damit wird der Rührer gestartet und der Regler erhält den Sollwert 25°C.



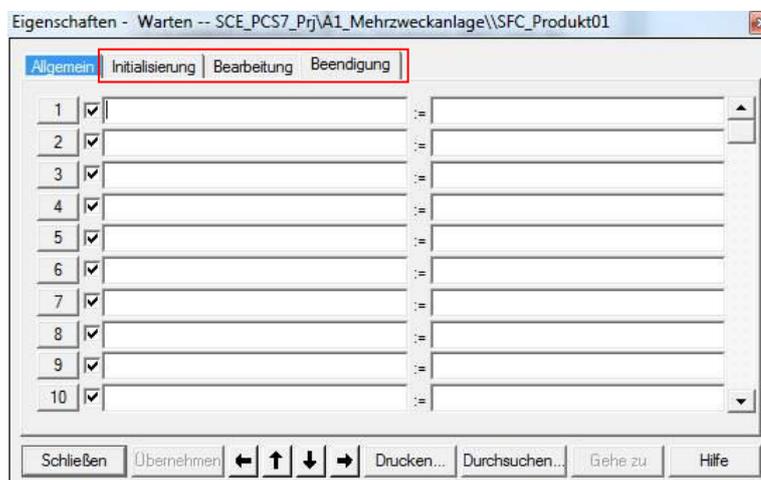
39. Unter ‚Beendigung‘ stoppen Sie den Rührer wieder und setzen den Sollwert auf 0°C. Anschließend schließen Sie den Dialog.



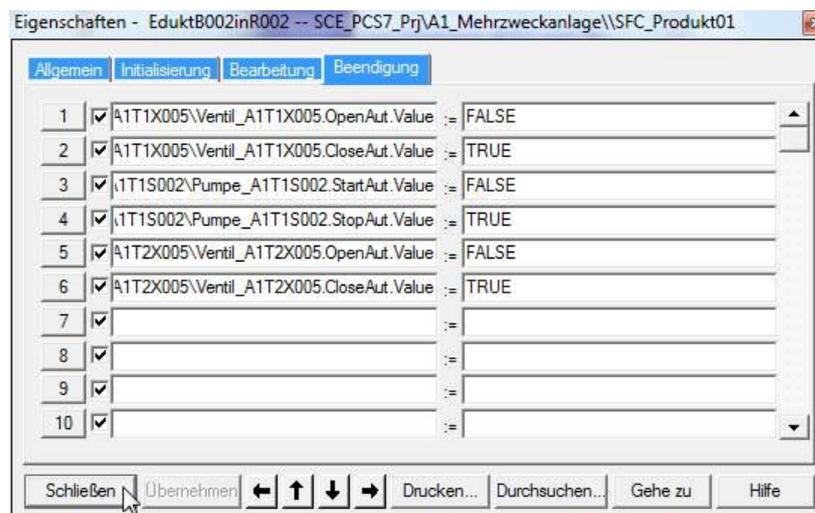
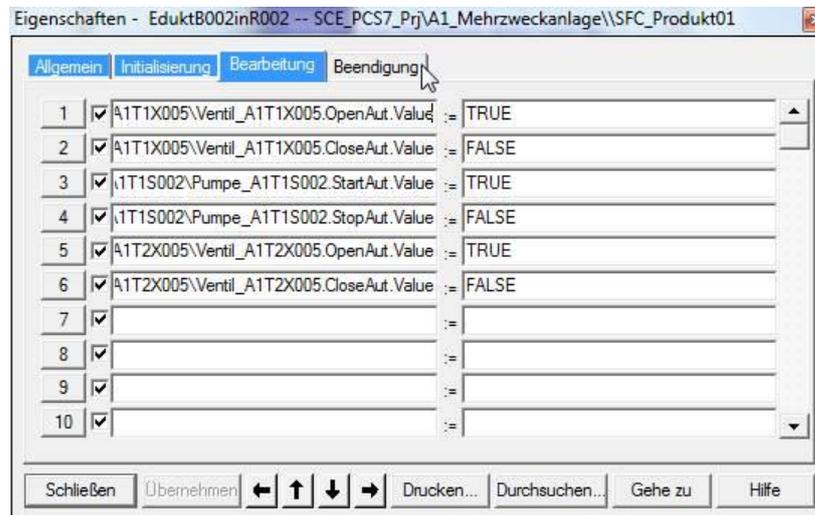
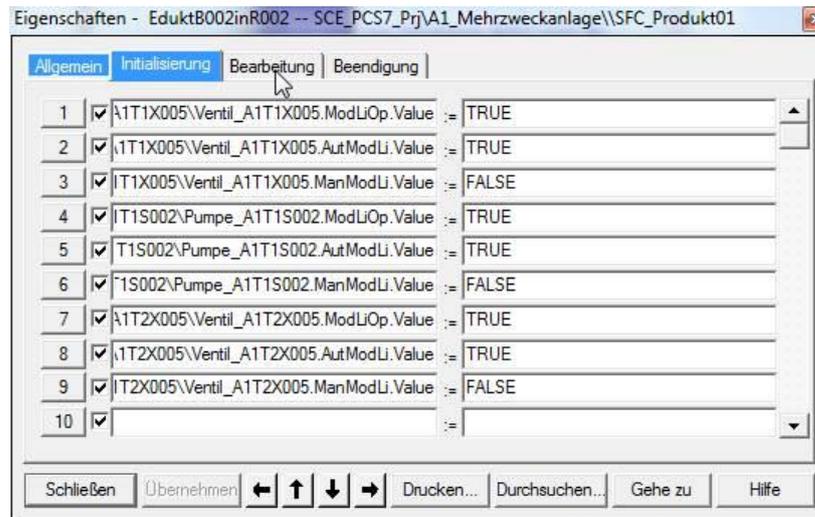
40. Nun parametrieren Sie die Transition ‚T001 >= 25°C‘. Dafür benötigen Sie die gemessene Temperatur. (→ T001 >= 25°C → Bedingung → ...\T2_Reaktion\Reaktor R001\A1T2T001\In_A1T2T001 → PV_Out → Value → Übernehmen → >= → 25.0 → Übernehmen → Schließen)



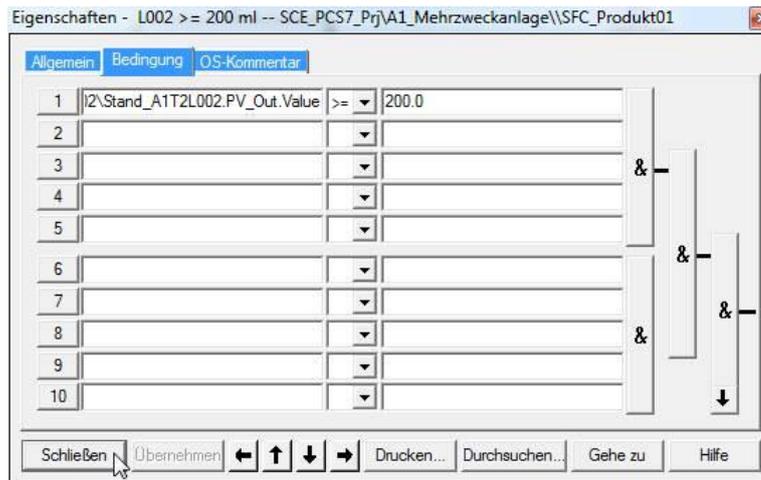
41. Im Schritt ‚Warten‘ bleiben ‚Initialisierung‘, ‚Bearbeitung‘ und ‚Beendigung‘ leer. Sie erkennen das daran, dass die Register keine Markierung aufweisen.



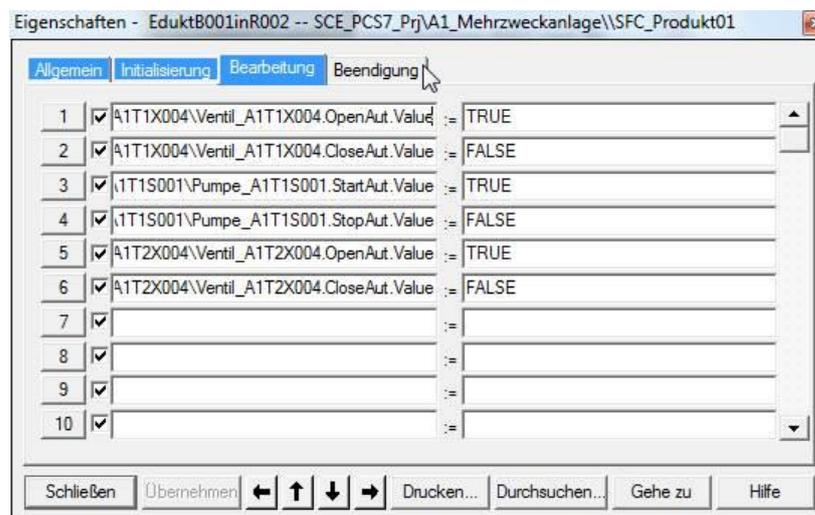
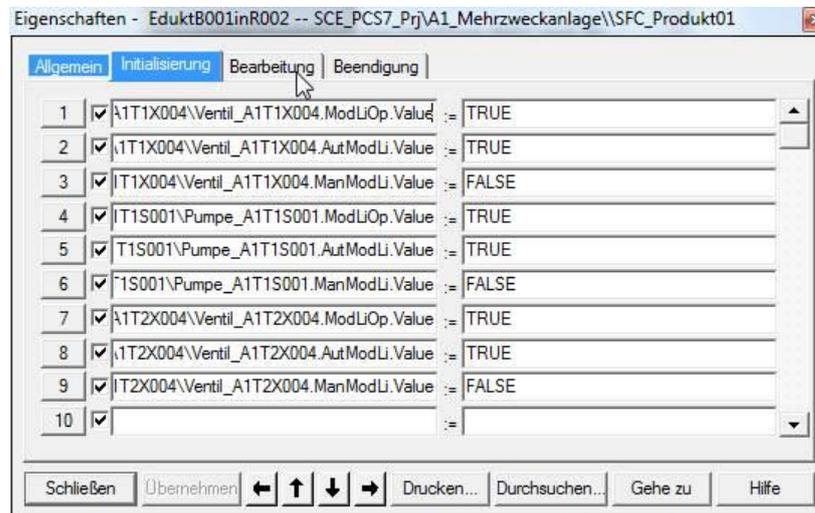
42. Füllen Sie nun den parallelen Strang aus. Beginnen Sie mit Schritt ‚EduktB002inR002‘ und nutzen Sie die folgenden Abbildungen. (→ EduktB002inR002)

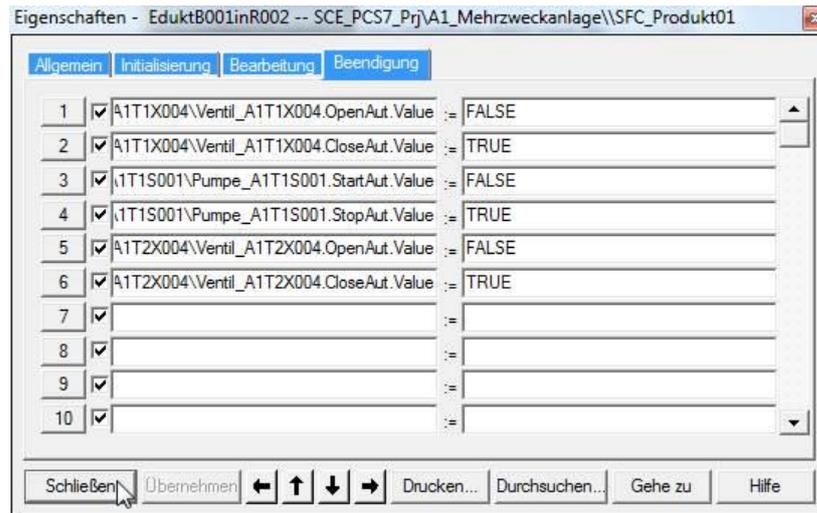


43. Die Transition ‚L002 >= 200 ml‘ sieht daraufhin wie folgt aus. (→ L002 >= 200 ml)

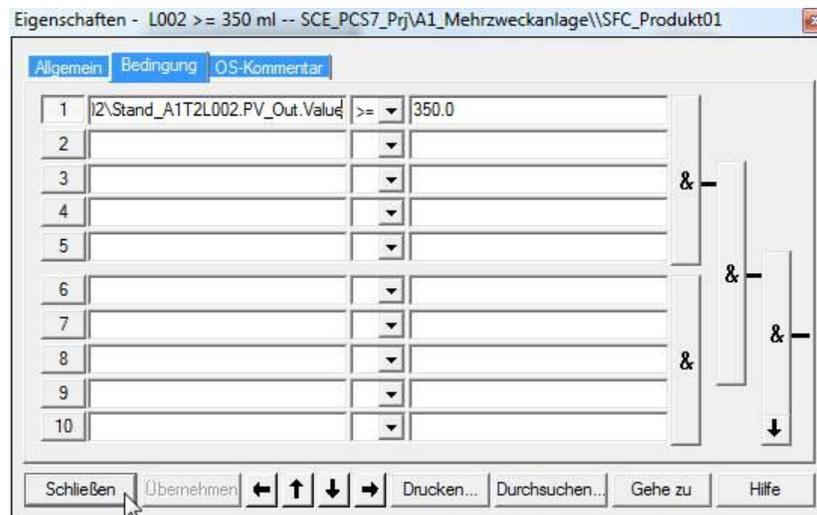


44. Im Schritt ‚EduktB001inR002‘ müssen Sie die folgenden Verschaltungen vornehmen.

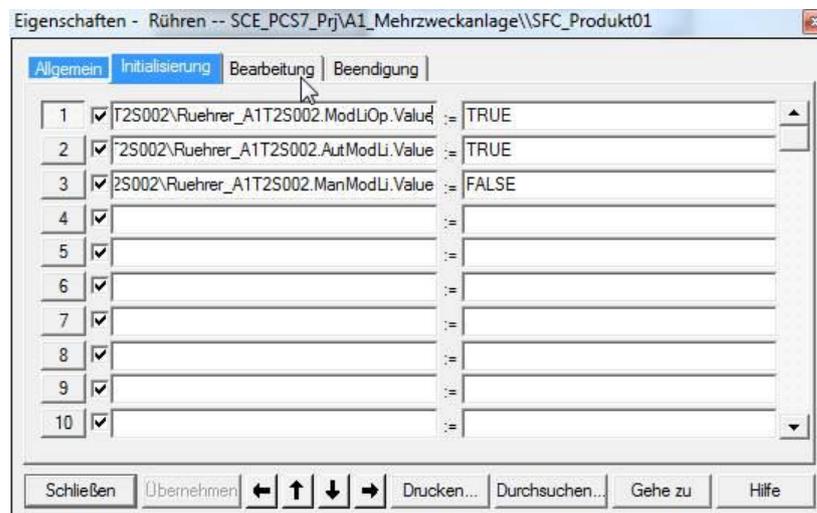


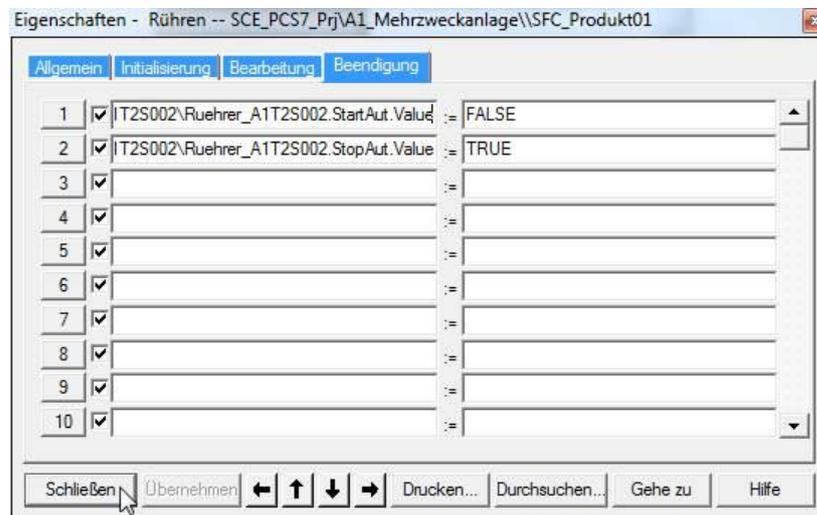
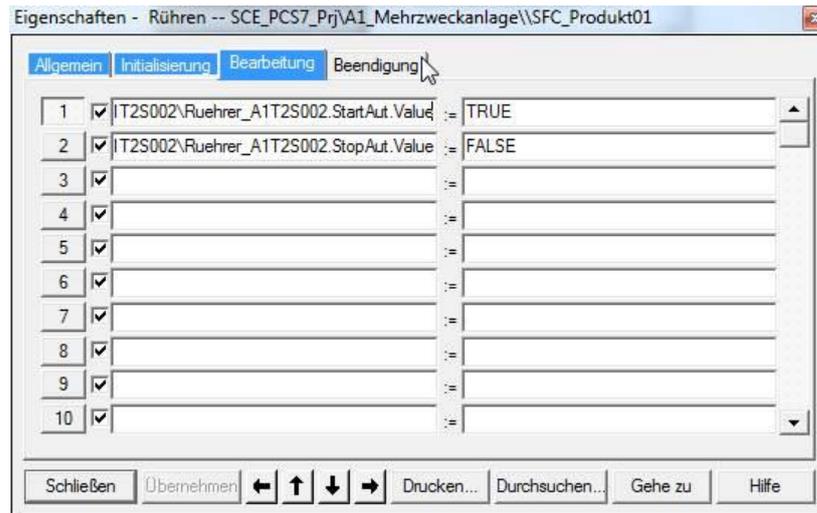


45. Die Transition ,L002 >= 350 ml' sieht nun folgendermaßen aus. (→ L002 >= 350 ml)

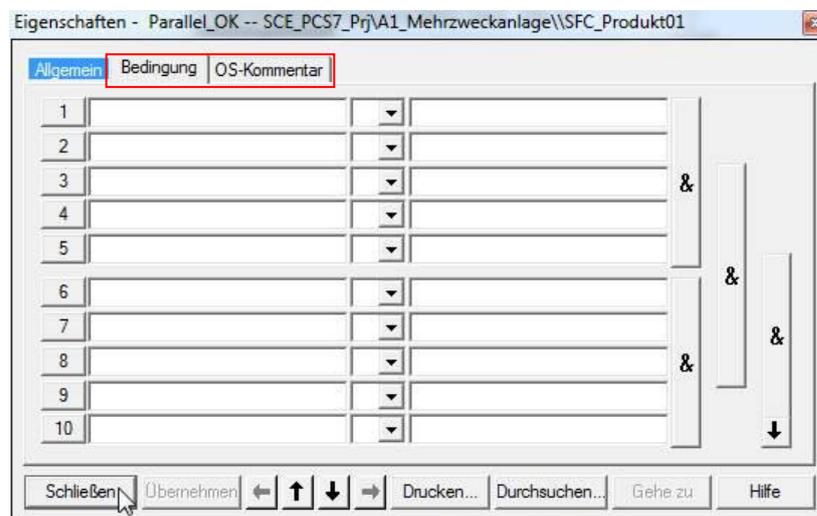


46. Der Schritt ,Rühren' hat eine minimale Laufzeit von 10 Sekunden. Das hatten Sie bereits am Anfang parametrieren. Jetzt müssen Sie den Rührer_A1T2S002 initialisieren, starten und wieder stoppen.

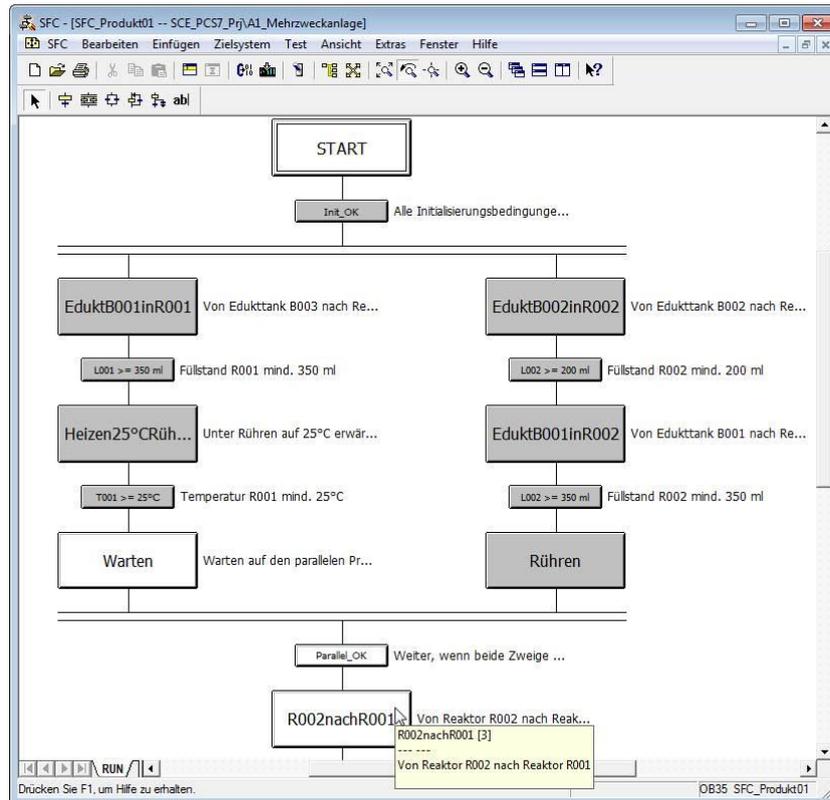




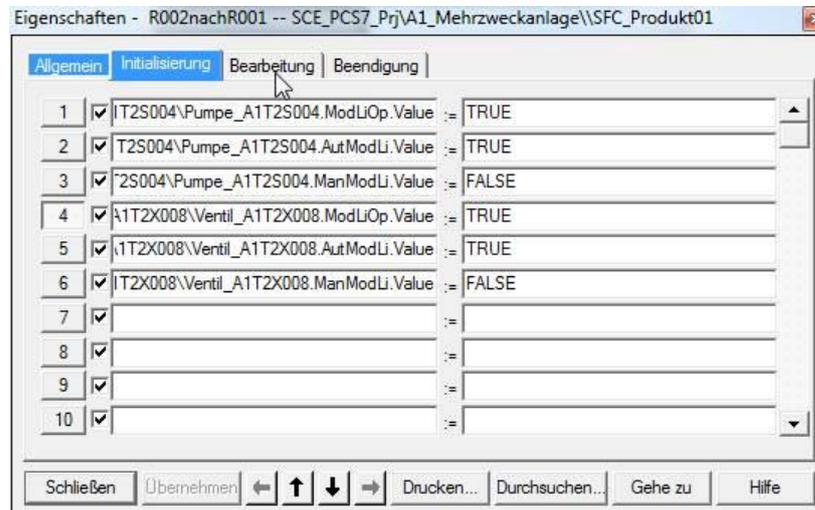
47. Jetzt haben Sie bereits den parallelen Zweig fertig parametrier. Die Transition ‚Parallel_OK‘ bleibt leer. Das bedeutet sobald die Schritte ‚Warten‘ und ‚Rühren‘ abgearbeitet sind, wird der Schritt ‚R002nachR001‘ aktiv.

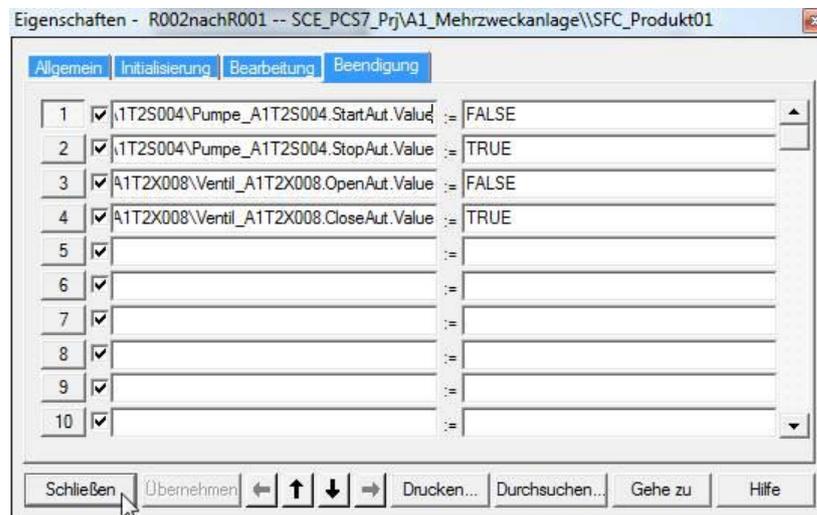
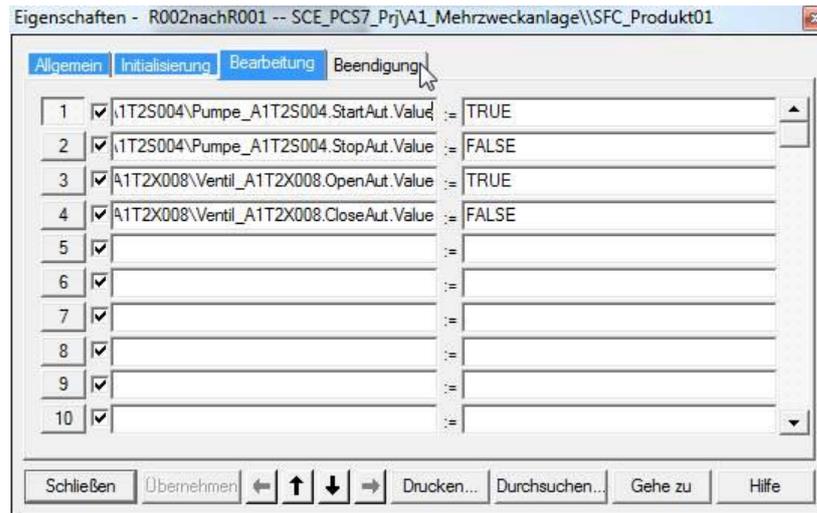


48. Die Ablaufsteuerung sieht nun wie folgt aus.

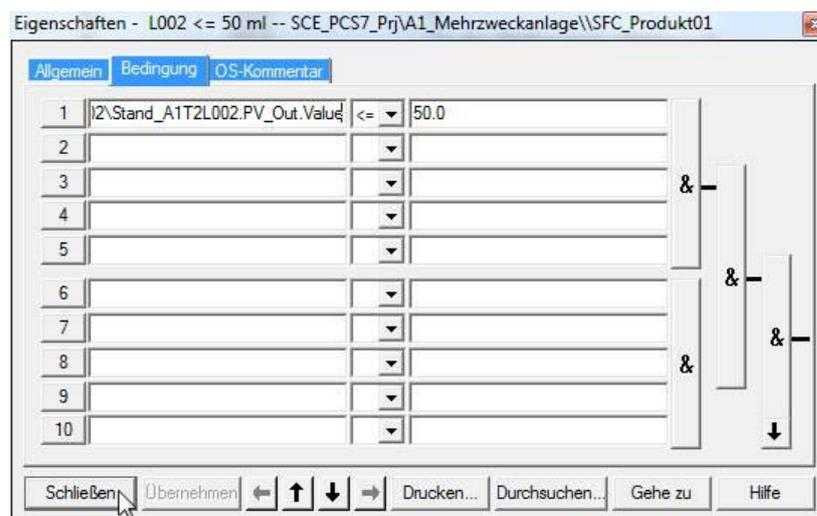


49. Nun wird der Schritt 'R002nachR001' verschaltet.

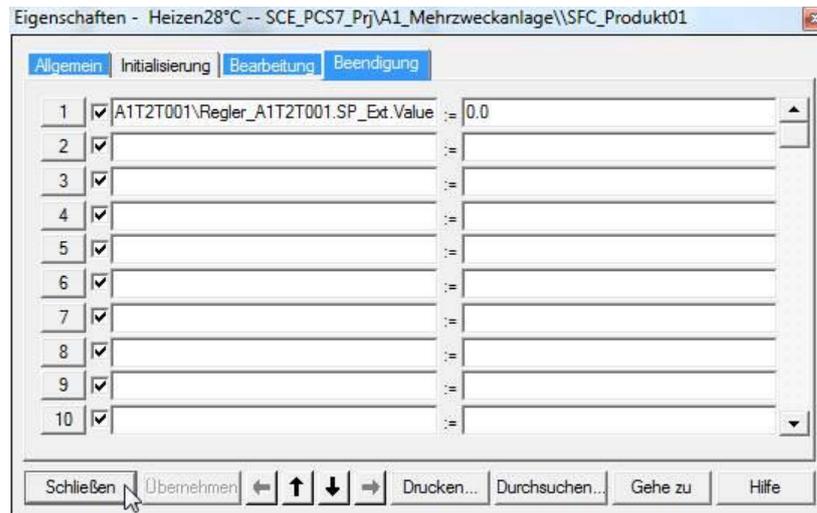
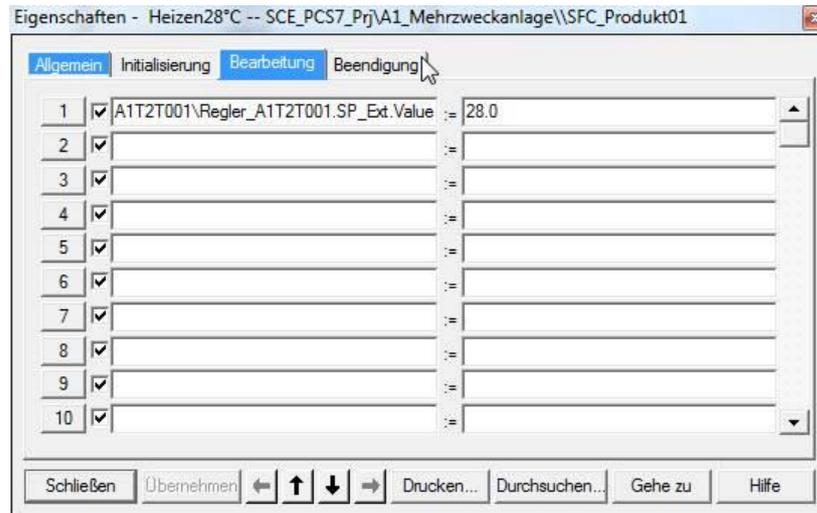




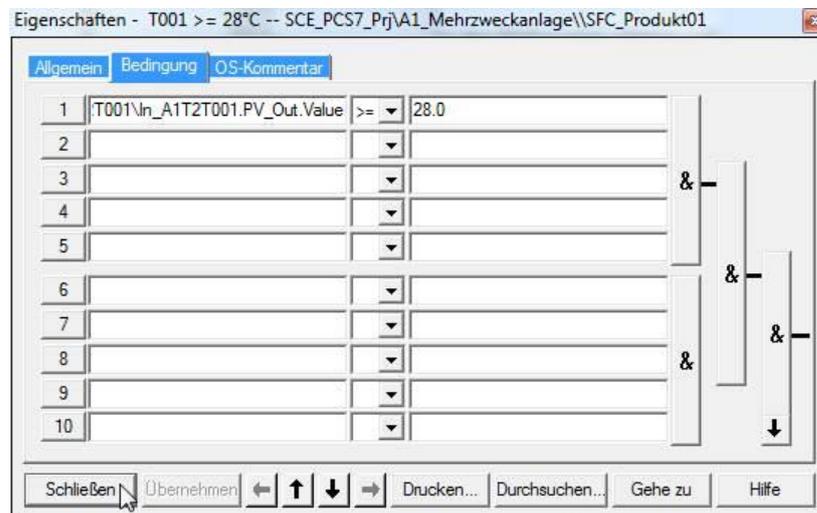
50. Die Transition ,L002 <= 50 ml' muss wie folgt verschaltet werden.



51. Im Schritt ‚Heizen28°C‘ wird wieder der Regler aktiviert. Da dieser bereits auf SFC-Modus und Automatikbetrieb steht, muss nur noch der Sollwert vorgegeben werden. Dieser muss beim Beenden wieder auf 0°C zurückgesetzt werden.



52. Die Bedingungen in der Transition ‚T001 >= 28°C‘ sieht jetzt wie folgt aus.



53. Der letzte Schritt „R001inProdB001“ des Rezepts füllt den Inhalt des Reaktors R001 in den angeschlossenen Produkttank B001. Die Verschaltungen sind nachfolgend dargestellt.

Eigenschaften - R001inProdB001 -- SCE_PCS7_Prj\A1_Mehrzweckanlage\SFC_Produkt01

Allgemein Initialisierung **Bearbeitung** Beendigung

1	<input checked="" type="checkbox"/>	\T2S003\Pumpe_A1T2S003.ModLiOp.Value	::	TRUE
2	<input checked="" type="checkbox"/>	\T2S003\Pumpe_A1T2S003.AutModLi.Value	::	TRUE
3	<input checked="" type="checkbox"/>	\T2S003\Pumpe_A1T2S003.ManModLi.Value	::	FALSE
4	<input checked="" type="checkbox"/>	\A1T3X001\Ventil_A1T3X001.ModLiOp.Value	::	TRUE
5	<input checked="" type="checkbox"/>	\A1T3X001\Ventil_A1T3X001.AutModLi.Value	::	TRUE
6	<input checked="" type="checkbox"/>	\A1T3X001\Ventil_A1T3X001.ManModLi.Value	::	FALSE
7	<input checked="" type="checkbox"/>		::	
8	<input checked="" type="checkbox"/>		::	
9	<input checked="" type="checkbox"/>		::	
10	<input checked="" type="checkbox"/>		::	

Schließen Übernehmen ⬅️ ⬆️ ⬇️ ⬇️ ➡️ Drucken... Durchsuchen... Gehe zu Hilfe

Eigenschaften - R001inProdB001 -- SCE_PCS7_Prj\A1_Mehrzweckanlage\SFC_Produkt01

Allgemein Initialisierung **Bearbeitung** Beendigung

1	<input checked="" type="checkbox"/>	\T2S003\Pumpe_A1T2S003.StartAut.Value	::	TRUE
2	<input checked="" type="checkbox"/>	\T2S003\Pumpe_A1T2S003.StopAut.Value	::	FALSE
3	<input checked="" type="checkbox"/>	\A1T3X001\Ventil_A1T3X001.OpenAut.Value	::	TRUE
4	<input checked="" type="checkbox"/>	\A1T3X001\Ventil_A1T3X001.CloseAut.Value	::	FALSE
5	<input checked="" type="checkbox"/>		::	
6	<input checked="" type="checkbox"/>		::	
7	<input checked="" type="checkbox"/>		::	
8	<input checked="" type="checkbox"/>		::	
9	<input checked="" type="checkbox"/>		::	
10	<input checked="" type="checkbox"/>		::	

Schließen Übernehmen ⬅️ ⬆️ ⬇️ ⬇️ ➡️ Drucken... Durchsuchen... Gehe zu Hilfe

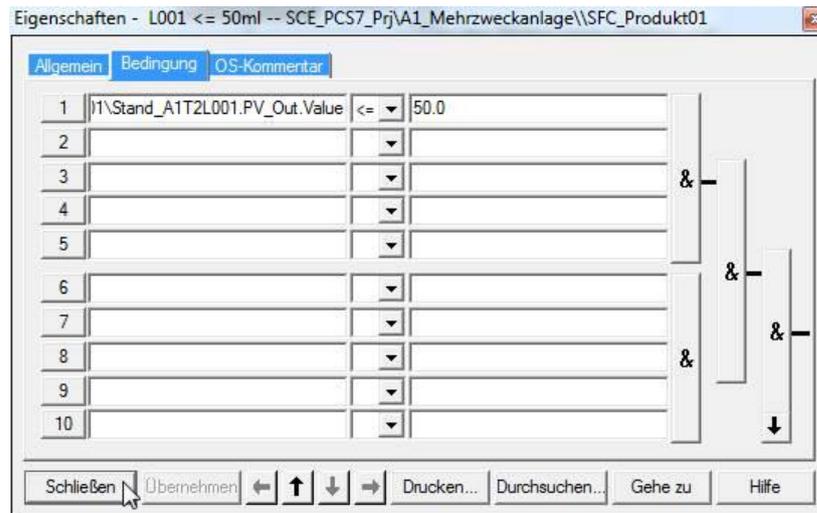
Eigenschaften - R001inProdB001 -- SCE_PCS7_Prj\A1_Mehrzweckanlage\SFC_Produkt01

Allgemein Initialisierung **Bearbeitung** Beendigung

1	<input checked="" type="checkbox"/>	\T2S003\Pumpe_A1T2S003.StartAut.Value	::	FALSE
2	<input checked="" type="checkbox"/>	\T2S003\Pumpe_A1T2S003.StopAut.Value	::	TRUE
3	<input checked="" type="checkbox"/>	\A1T3X001\Ventil_A1T3X001.OpenAut.Value	::	FALSE
4	<input checked="" type="checkbox"/>	\A1T3X001\Ventil_A1T3X001.CloseAut.Value	::	TRUE
5	<input checked="" type="checkbox"/>		::	
6	<input checked="" type="checkbox"/>		::	
7	<input checked="" type="checkbox"/>		::	
8	<input checked="" type="checkbox"/>		::	
9	<input checked="" type="checkbox"/>		::	
10	<input checked="" type="checkbox"/>		::	

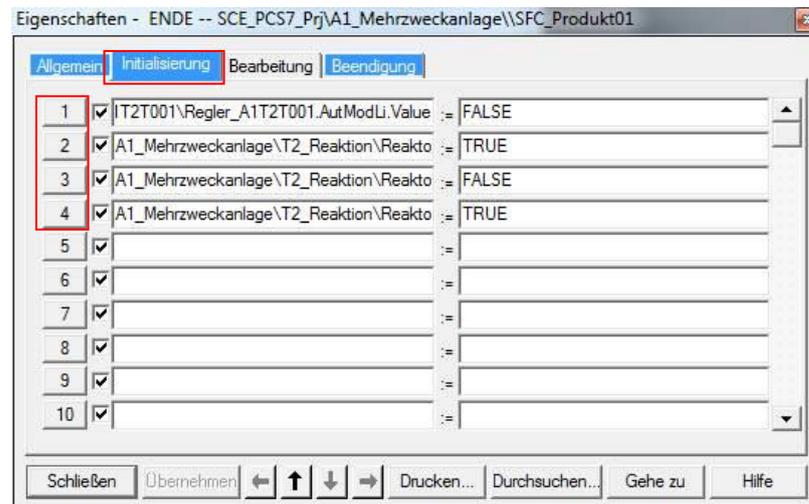
Schließen Übernehmen ⬅️ ⬆️ ⬇️ ⬇️ ➡️ Drucken... Durchsuchen... Gehe zu Hilfe

54. Die Transition ‚L001 <= 50 ml‘ ist die letzte Transition des Rezeptes. Sie kann geschaltet werden, wenn der Reaktor R001 leer (<= 50 ml) ist.



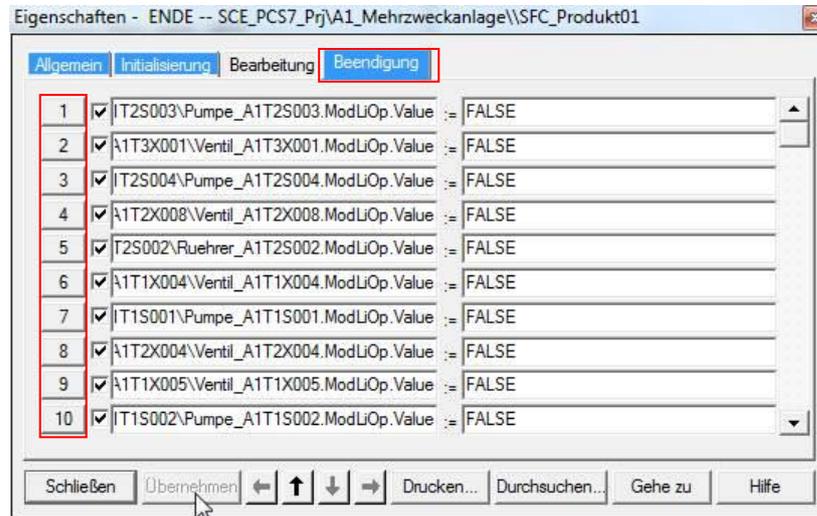
55. Im Schritt ‚ENDE‘ muss bei allen genutzten Ventile, Pumpen, Rührer und Regler der Automatikbetrieb ausgeschaltet und der Handbetrieb wieder eingeschaltet werden. Beim Regler muss zusätzlich die interne Sollwertvorgabe wieder eingestellt werden.

‚Initialisierung‘:

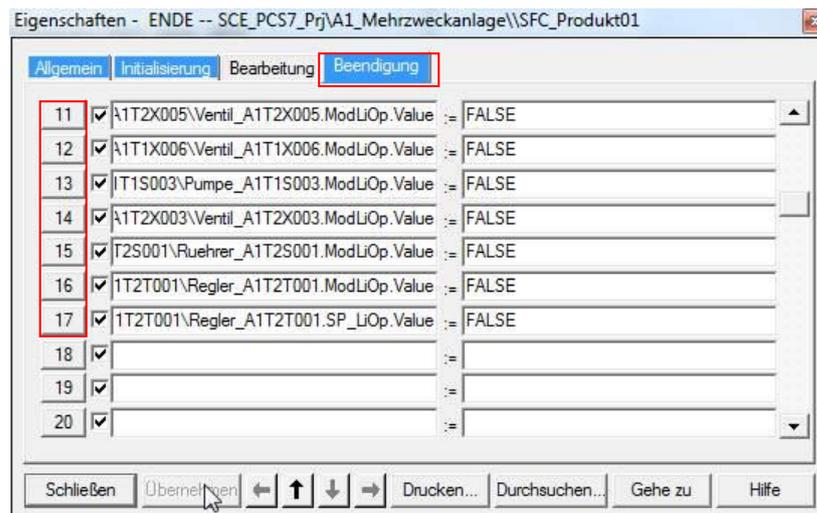


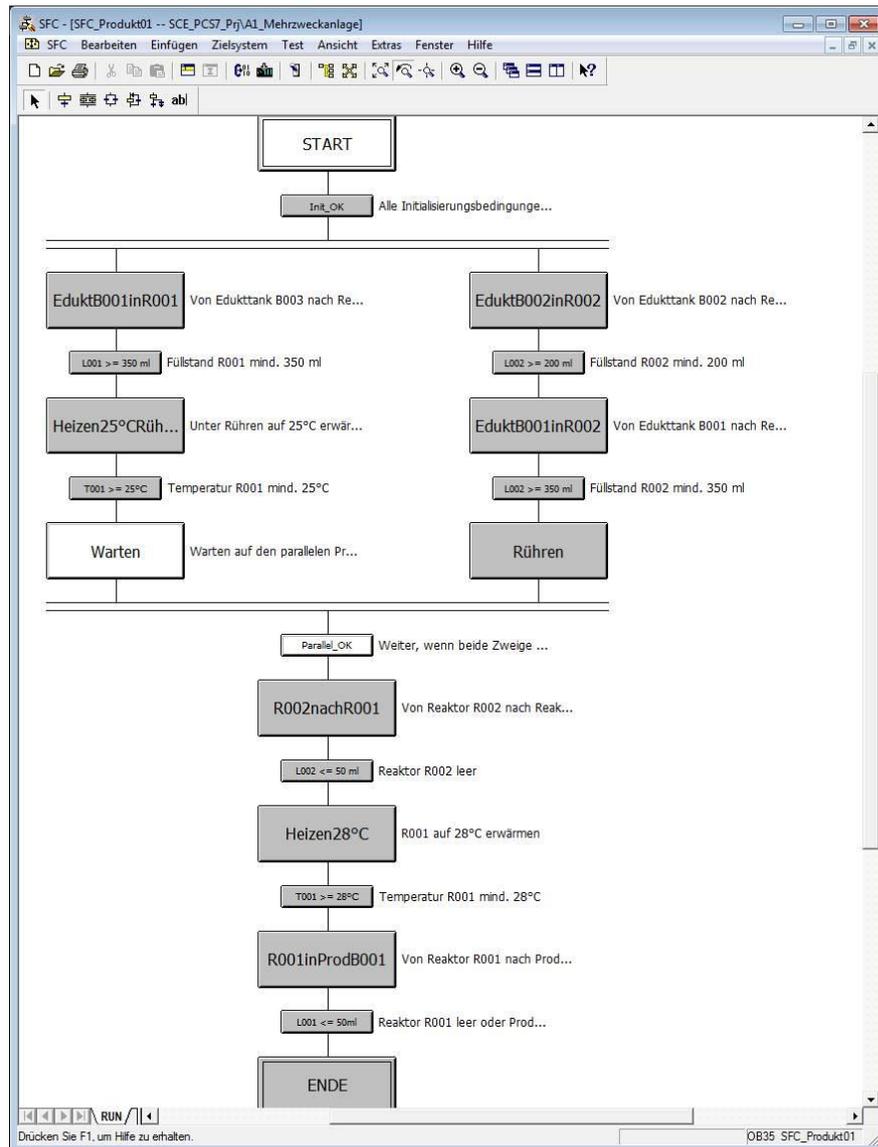
56. Anschließend werden alle genutzten Pumpen, Ventilen, Rührer und Regler auch wieder auf Operatorbetrieb („ModLiOp“ = „0“) gesetzt.

„Beendigung“ - 1:

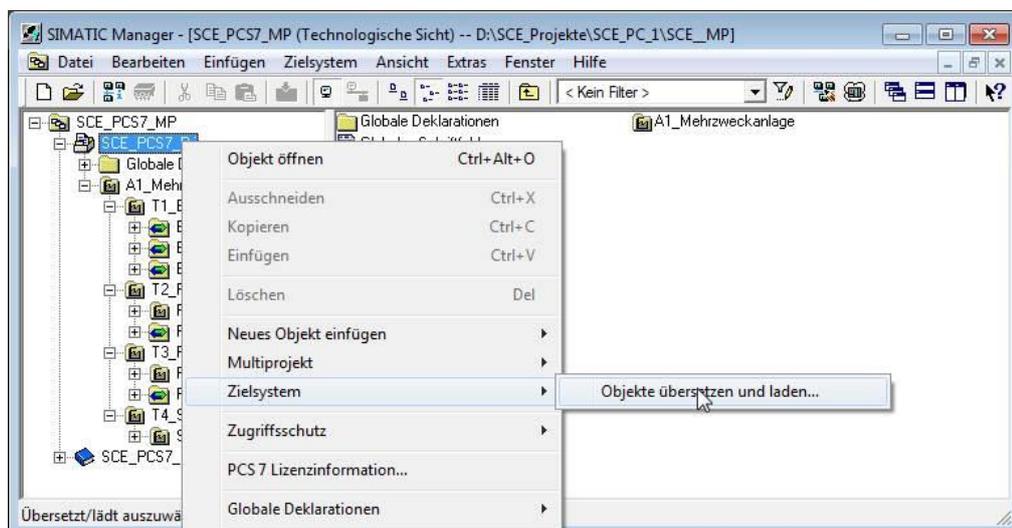


„Beendigung“ - 2:

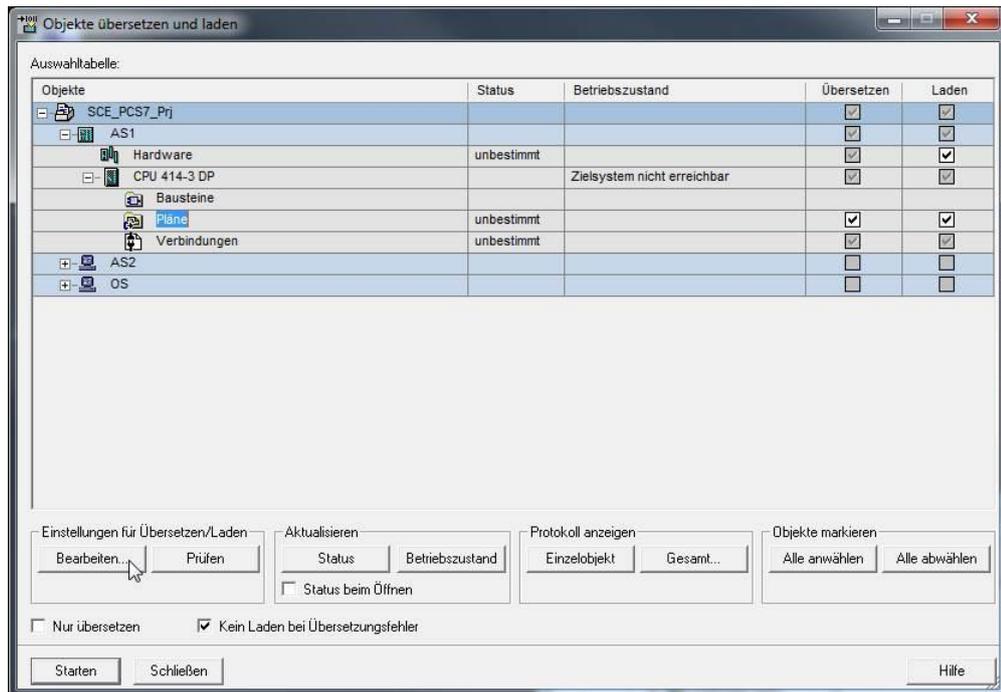




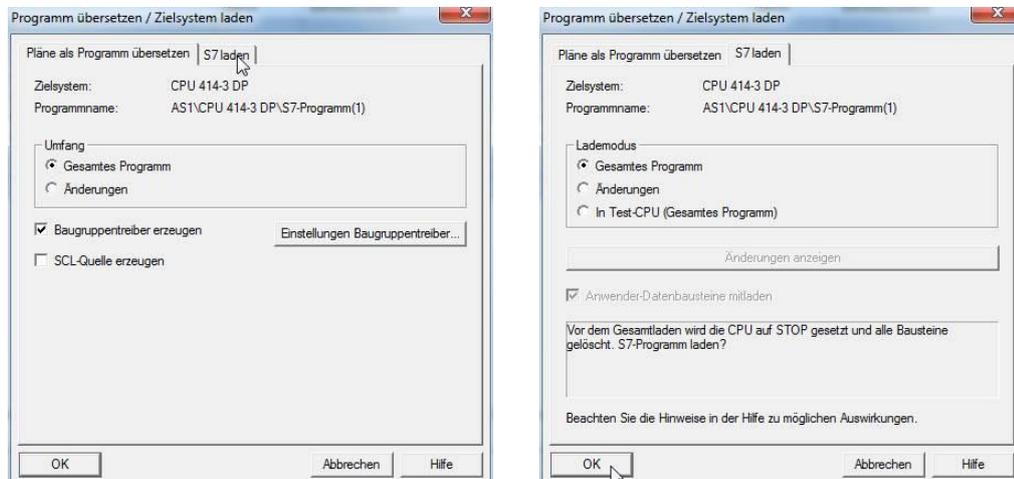
57. Nach der Fertigstellung aller Schritte und Transitionen des SFC's können Sie das Projekt, wie bereits kennengelernt, übersetzen und laden. (SCE_PCS7_Prj → Zielsystem → Objekte übersetzen und laden...)



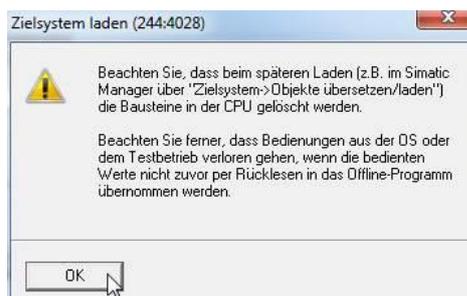
58. Bevor Sie mit dem Übersetzen und Laden beginnen, öffnen Sie die Einstellungen für das Übersetzen und Laden der Pläne. (→ Pläne → Bearbeiten)



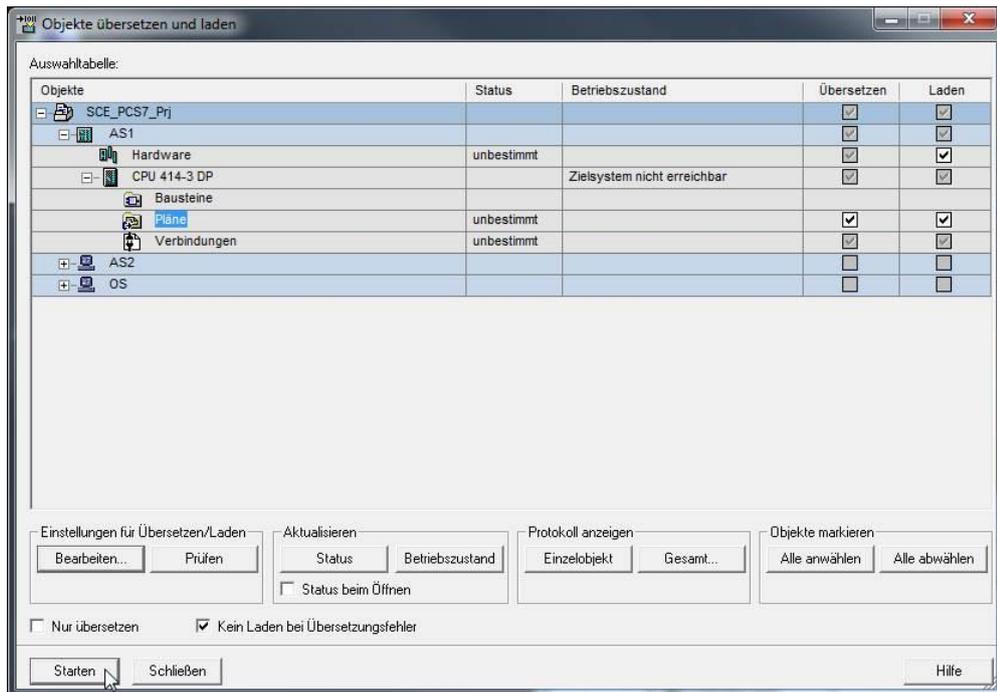
59. Hier ist es wichtig sowohl bei „Pläne als Programm übersetzen“ als auch bei „S7 laden“ beim Umfang bzw. Lademodus das gesamte Programm auszuwählen. (→ Pläne als Programm übersetzen → Gesamtes Programm → S7 laden → Gesamtes Programm → OK)



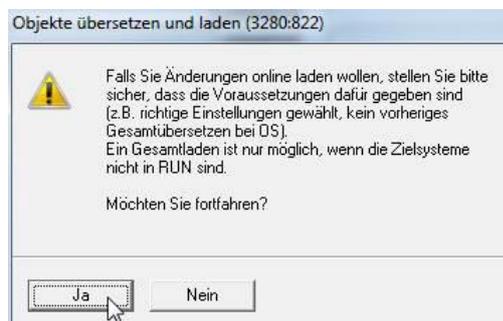
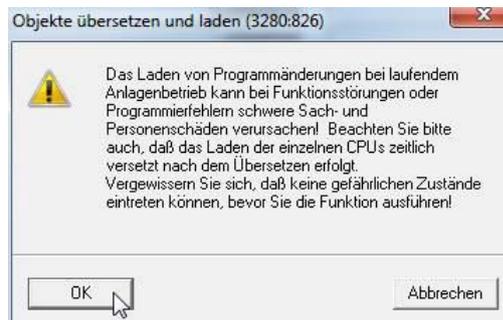
60. Der Hinweis wird mit „OK“ bestätigt. (→ OK)



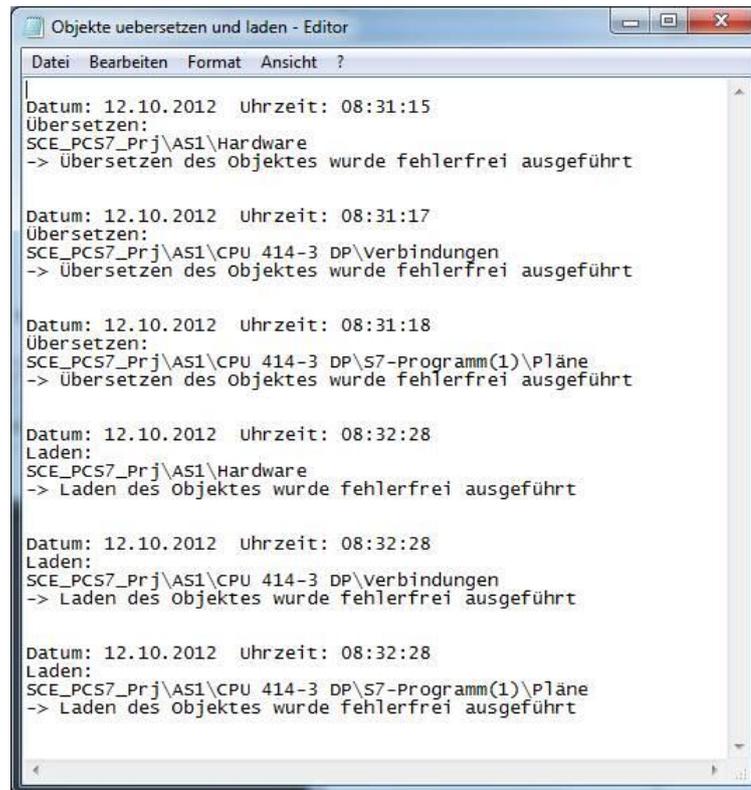
61. Nun kann mit dem Übersetzen und Laden begonnen werden. (→ Starten)



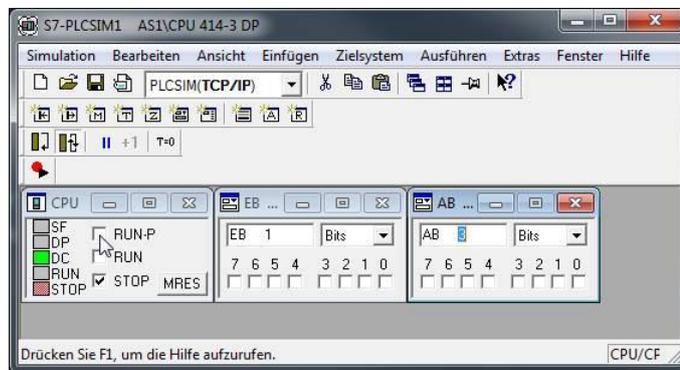
62. Alle nachfolgenden Warnhinweise werden aufmerksam gelesen und bestätigt. (→ OK → Ja)



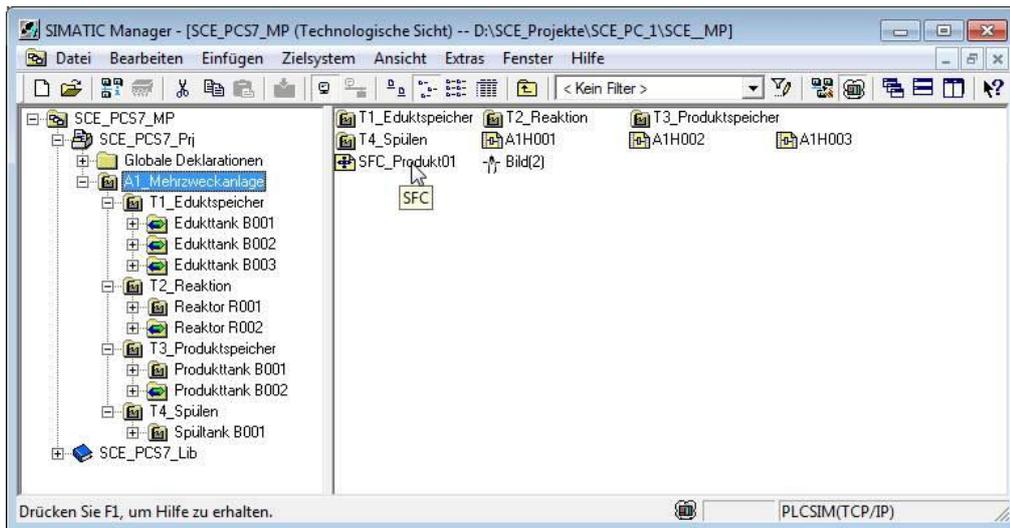
63. Im Protokoll sollten keine Fehler sondern allerhöchstens Warnung erscheinen. Details für Warnung können dem Protokoll des Einzelobjektes entnommen werden. (→ X)



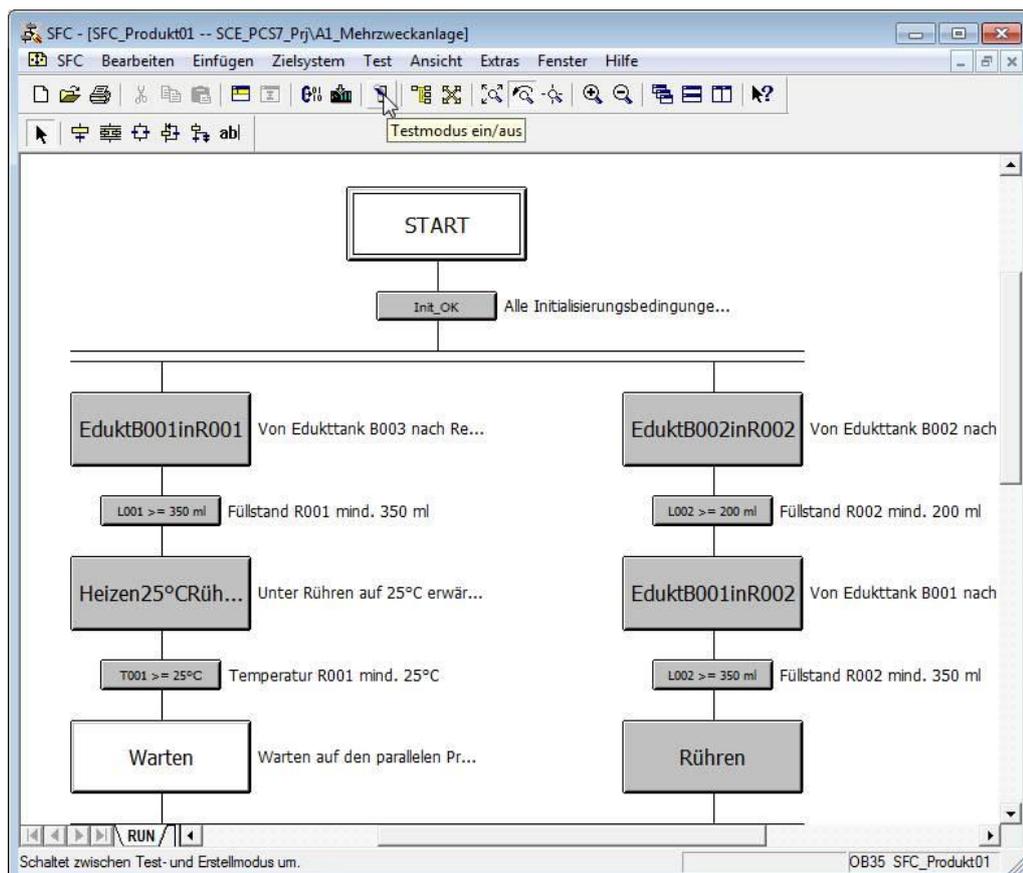
64. Nun können Sie PLCSIM in den RUN-P Modus setzen. (→ PLCSIM → RUN-P)



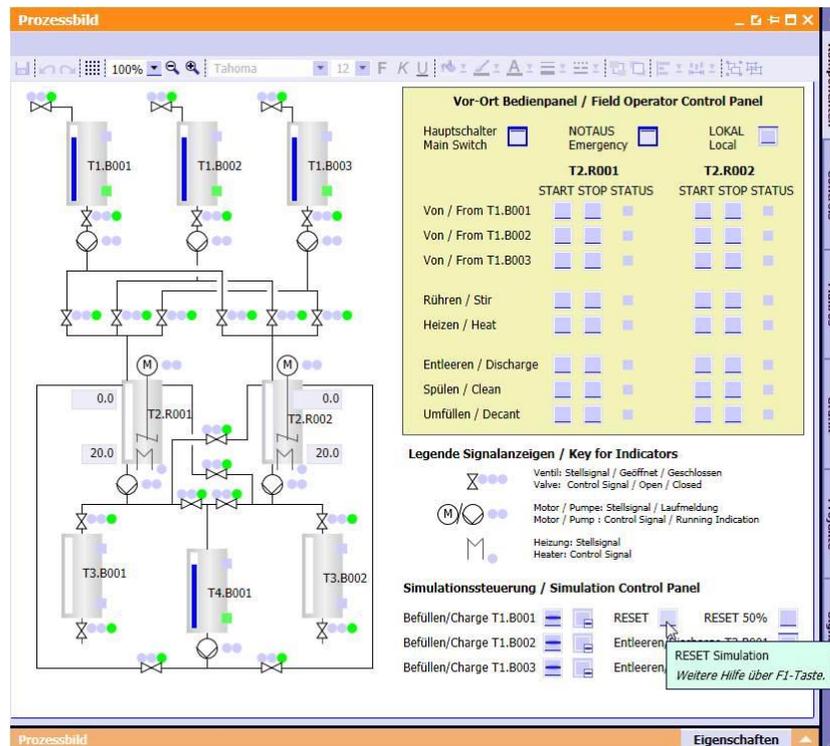
65. Sie öffnen aus der Technologischen Hierarchie heraus die Schrittkette mit einem Doppelklick. (→ SFC_Produkt01)



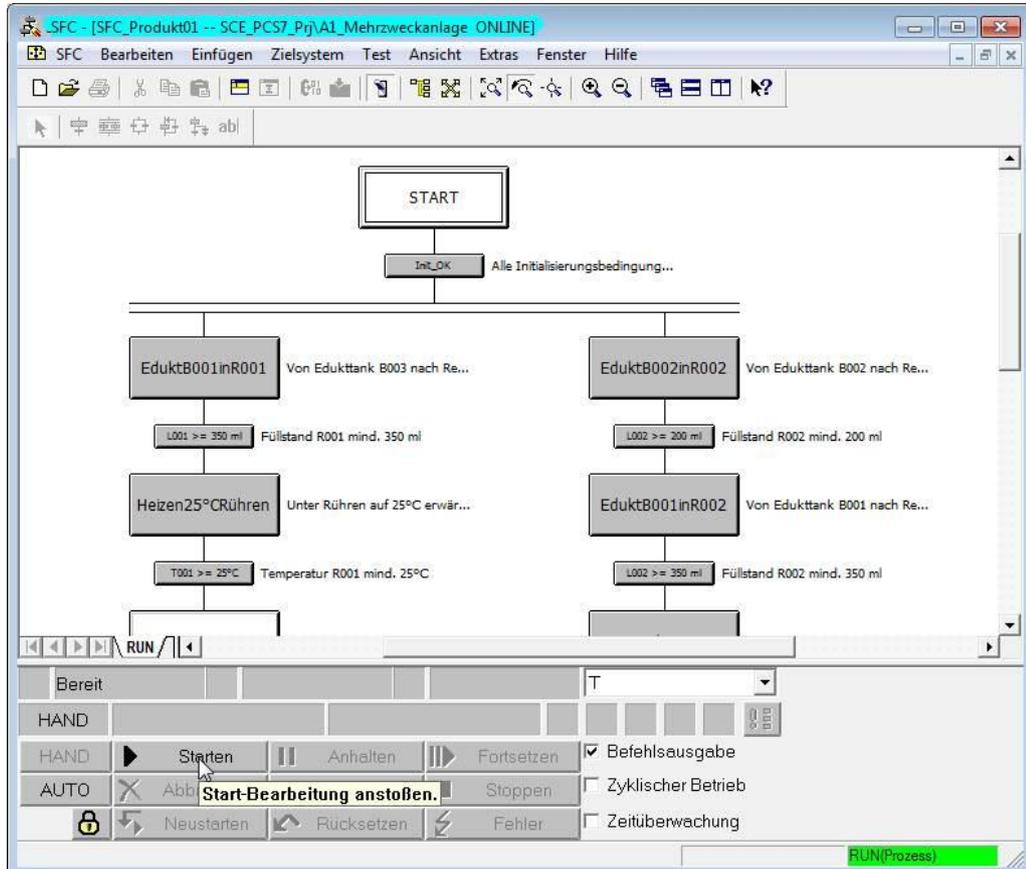
66. Um den Ablauf beobachten zu können muss der Testmodus eingeschaltet werden.
(→ Testmodus ein/aus)



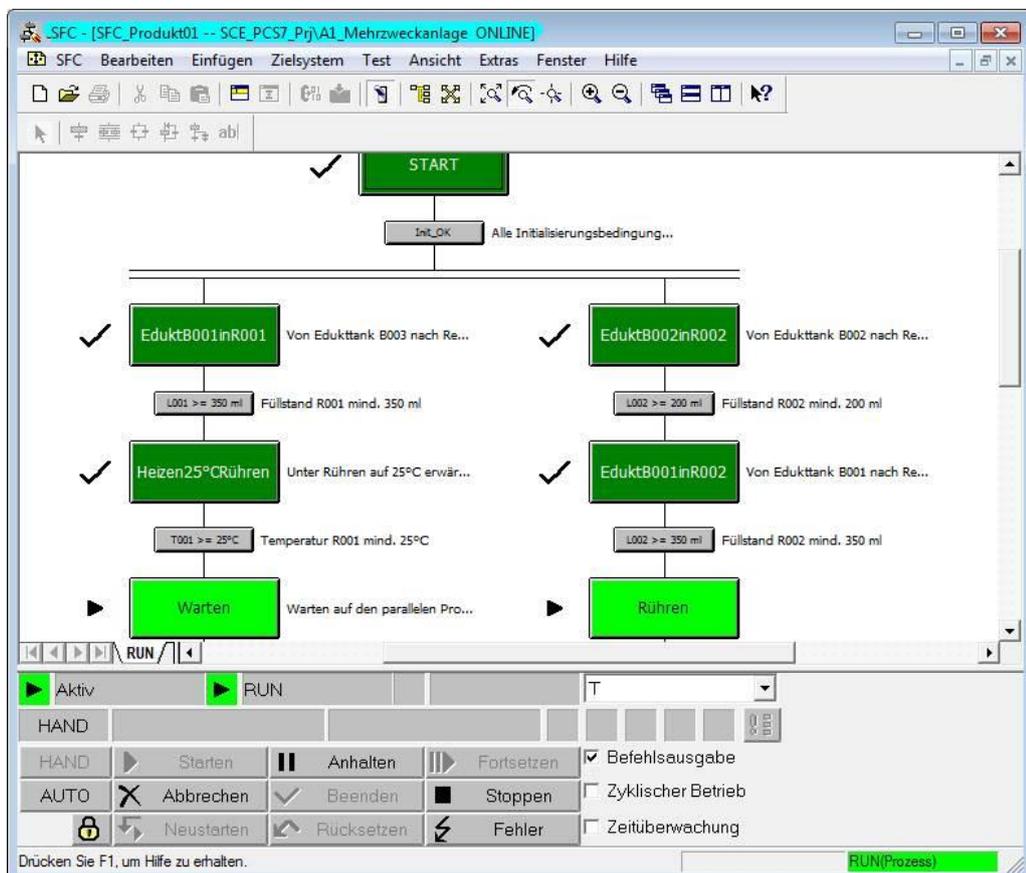
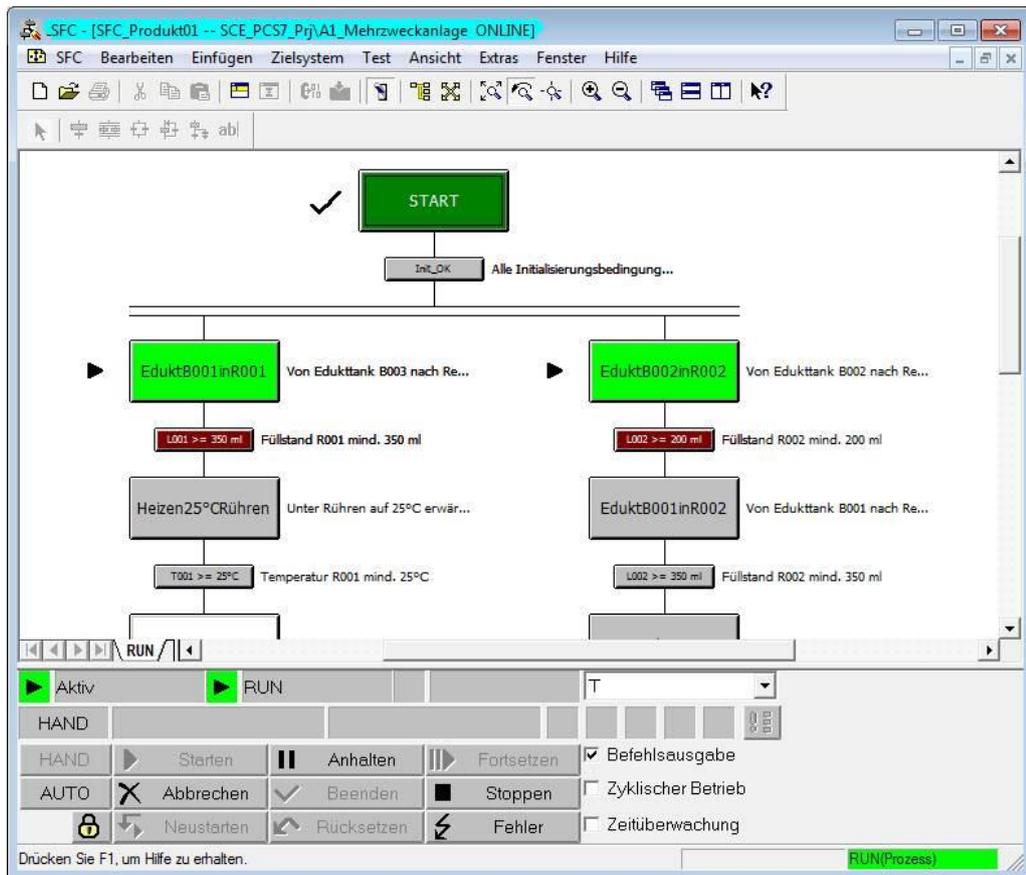
67. Die Simulation muss zurückgesetzt, der Hauptschalter und das NOTAUS aktiviert sowie die lokale Bedienung deaktiviert sein.



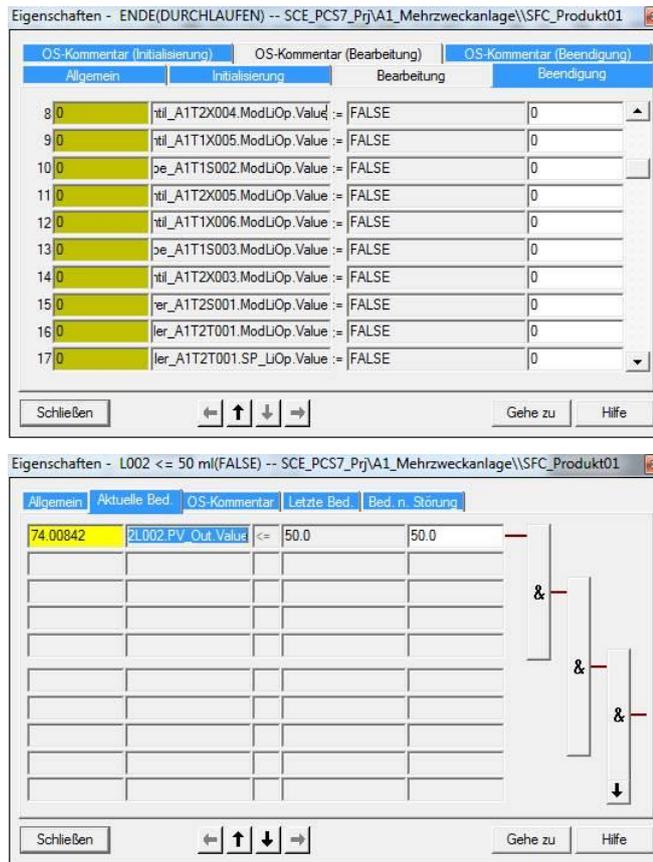
68. Nun kann der SFC gestartet werden. (→ Starten)



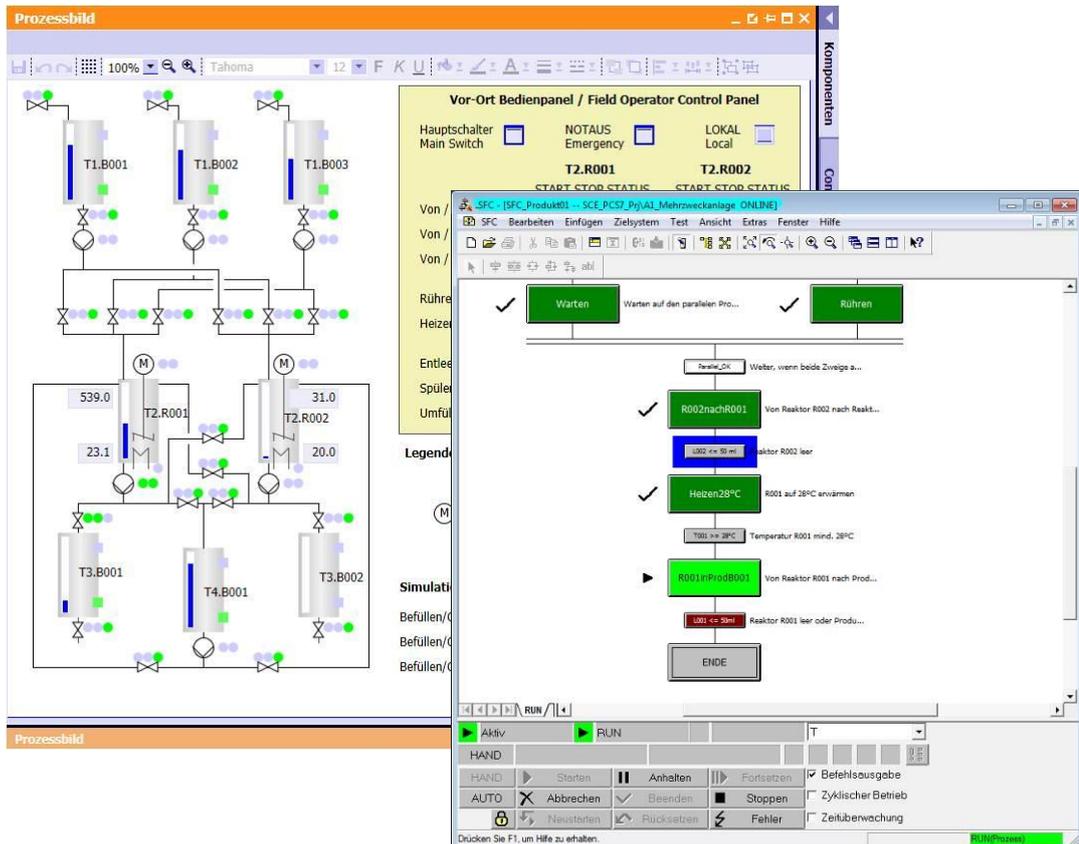
69. Der Ablauf der Schrittfolge kann nun beobachtet werden. Aktive und bereits abgearbeitete Schritte und Transitionen werden kenntlich gemacht.



70. Durch Doppelklick bzw. Öffnen von einzelnen Schritten oder Transitionen können aktuelle Bedingungen und Werte angezeigt werden.



71. Im Zustand ‚R001inProdB001‘ sehen der SFC und die Simulation wie folgt aus.



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_0108_Ueb_R1505.zip) genutzt und erweitert werden.

Diese Übung soll ein weiteres Rezept implementieren, welches die Reinigung der Reaktoren ermöglicht. Die folgende Übungsaufgabe schlägt ein mögliches Konzept vor.

ÜBUNGSAUFGABEN

1. Erstellen Sie den SFC ‚SFC_Spuelen‘ im Planordner ‚A1_Mehrzweckanlage‘, der die Reaktoren R001 und R002 mit Spülwasser reinigt. Die Reinigung soll aus folgenden Schritten bestehen:
 - Füllen der Reaktoren (bis 500 ml) mit Spülwasser
 - Rühren des Spülwassers (für 20 Sekunden) in den Reaktoren
 - Ablassen des Spülwassers in die Produkttanks.

Gestalten Sie den Spülvorgang so, dass beide Reaktoren gleichzeitig gereinigt werden!
Prüfen Sie, ob beide Reaktoren leer (< 50 ml) sind bevor der Spülvorgang beginnt.