

## **Automatisierungstechnik nach internationaler Norm programmieren (14)**

Autor: Dr. Ulrich Becker

Fachzentrum Automatisierungstechnik und vernetzte Systeme im BTZ Rohr-Kloster

Mail: Ulrich.Becker@BTZ-Rohr.de

### **Grundlagen der Ablaufsprache (AS) Teil 2**

In Folge 13 haben wir in die Ablaufsprache für Ablaufsteuerungen eingeführt. Komfortabel konnten wir mit dieser Sprache Schrittketten, Transitionen und Aktionen programmieren. Zuletzt haben wir Abläufe von Funktionen des Bandes am Trainingsrack unter Verwendung einfacher Schritte programmiert. Die gleiche Aufgabe soll nun in Folge 14 unter Verwendung von IEC-Schritten gelöst werden.

#### **Ablaufsteuerung des Verteilerbandes mit IEC - Schritten**

In Folge 13 Bild 74 wurde eine Aufgabe „Ablaufsteuerung Verteilerband“ vorgestellt und nachfolgend mit einfachen Schritten nach Bild 75 gelöst. **Bild 77** der vorliegenden Folge zeigt nun die Lösung der gleichen Aufgabe unter Verwendung von IEC-Schritten anstelle einfacher Schritte. Die Transitionen sind die gleichen wie die der Lösung mit einfachen Schritten. Dagegen wird das Programmieren der Aktionen durch Einsatz der Qualifier bedeutend einfacher. Lediglich zwei Aktionen sind zu programmieren. Zu erkennen ist auch, daß kein Timer für die Wartezeit programmieren ist. Diese Aufgabe übernimmt allein der Qualifier „D“. Da ein Rücksprung zum Initialschritt unumgänglich ist und dieser Schritt mit keinen Aktionen assoziiert werden kann, wurde ein mit der Transition TRUE direkt aktivierter Schritt „Bereitschaft“ nachgeschaltet. Dieser schaltet aufgabengemäß die LED „H1“ ein.

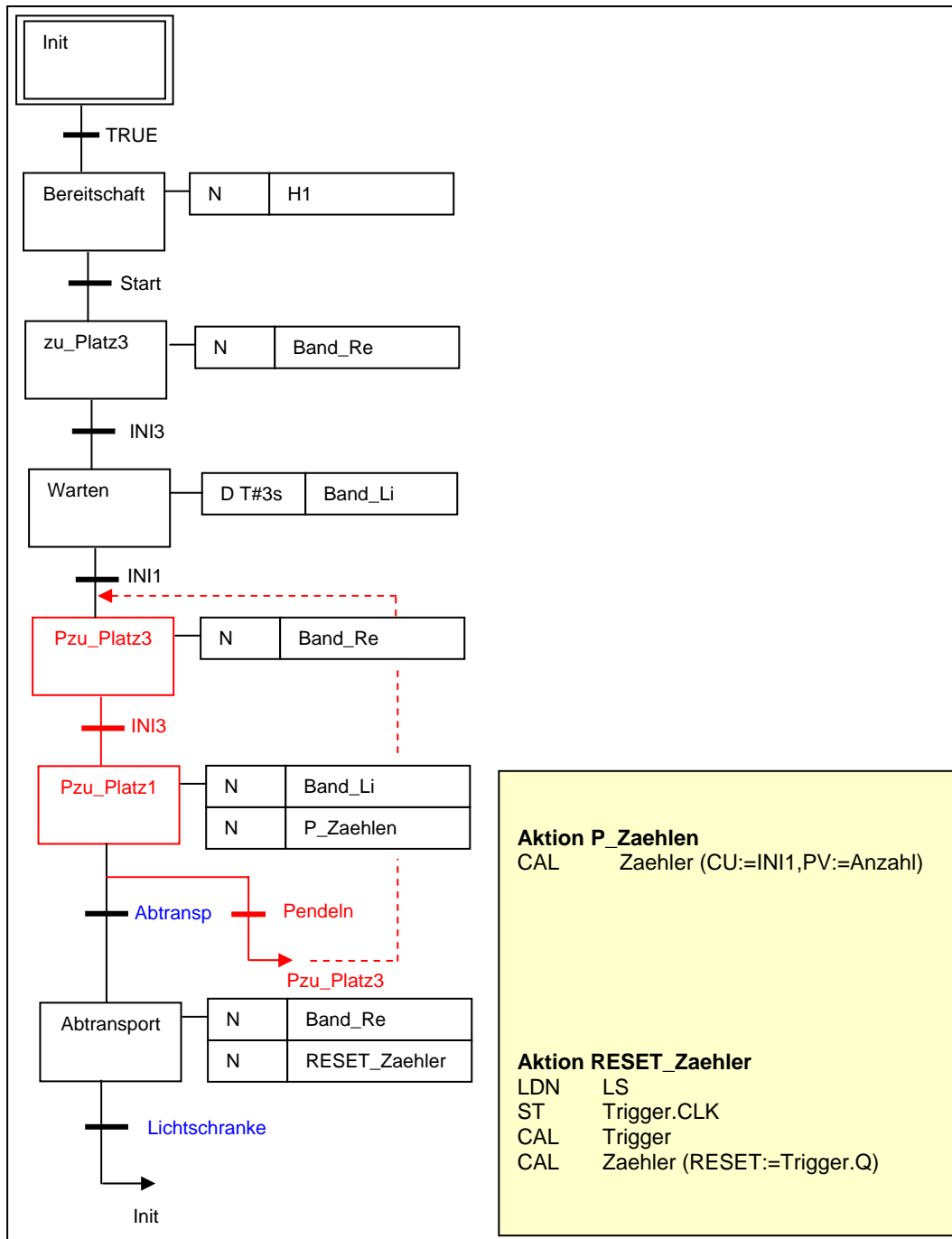


Bild 77: Ablaufsteuerung des Verteilerbandes mit IEC-Schritten und zugehörigen Transitionen

Die Ablaufsprache ist eine sehr mächtige Sprache, von der mit dieser Aufgabe nur wenige Grundlagen vermittelt werden konnten. Eine weitergehende Auseinandersetzung mit AS ist für die Lösung vieler Probleme der Automatisierungstechnik von großem Vorteil! Allein die Organisation der Einsprünge in bestimmte Schritte, ohne Endlosschleifen entstehen zu lassen, würde bei herkömmlicher Programmierung viel Überlegung erfordern! Sehr hilfreich sind auch SFC-Flags, welche durch die Bibliothek "lcsfc.lib" für die Steuerung und Überwachung einer Schrittkette bereitgestellt werden.

**Bild 78** zeigt beispielhaft die Verwendung nachfolgender ausgewählter AS\_Flags:

- SFCInit: Initialisierung der Schrittkette: Die Kette wird auf den Initialschritt zurückgesetzt, dieser jedoch nicht abgearbeitet
- SFCReset: wie SFCInit, jedoch wird der Initialschritt bearbeitet
- SFCPause: Die Abarbeitung der Kette wird unterbrochen.

Die Flags werden jeweils mit dem Wert TRUE wirksam. Bild 78 zeigt, daß solche SFC-Flags auch als Parameter vom Typ VAR\_INPUT im Ablaufbaustein zu deklarieren sind. Beim Aufruf des Ablaufbausteins in der POE PLC\_PRG sind dann konkrete Eingangssignale anzuschalten. Im Beispiel sind dies Taster von Trainingsrack und Verteilerband. Die Nutzung solcher Funktionen erfordert einige Übung, weil man sich stets Rechenschaft ablegen muß, wie nach Unterbrechung oder Abbruch der weitere Ablauf der Kette erfolgen soll. So muß im Beispiel geprüft werden, wie das Rücksetzen des Zaehlers der Pendelbewegungen bei Abbruch des regulären Verlaufs erfolgen soll. Ohne Änderung der AS würde beim Start nach Abbruch der Zaehler von falschen Werten ausgehen. An dieser Stelle bewahrheitet sich erneut, daß der ungestörte Ablauf einer Kette der einfache Fall ist, während die Gestaltung der Randbedingungen für Einstieg in und Ausstieg aus einer Schrittkette durchaus mehr Überlegungen erfordern.

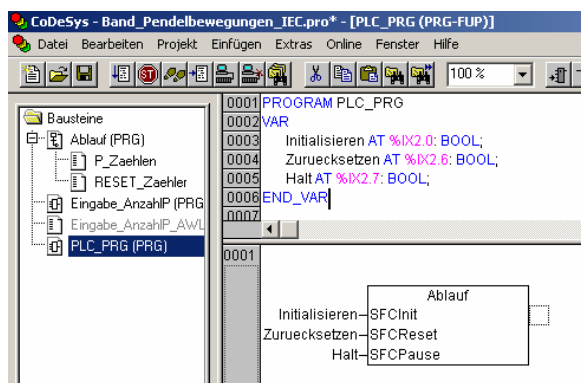


Bild 78: Prinzip der Nutzung von SFC-Flags zur Steuerung der Schrittkette

## Ablaufsprache Graph 7

Das in Step 7 Professional integrierte Tool Graph 7 ermöglicht die Anwendung der Ablaufsprache in Automatisierungssystemen Simatic S7. Viele besondere Details machen das Tool sehr komfortabel. Die Ablaufsprache Graph 7 kann nur in Step7 – Funktionsbausteinen angewendet werden, und die Daten werden im zugehörigen Instanzdatenbaustein verwaltet. Zusätzlich werden Systemfunktionen aktiviert. Beim Aufruf des Funktionsbausteins stehen dann eine Vielzahl nützlicher Parameter zur Verfügung. Diese sind mit SFC-Flags vergleichbar. Beispielsweise kann mit einem dieser Parameter die Schrittkette wie gezeigt auf den Initialschritt zurückgesetzt werden.

Graph 7 verwendet IEC-Schritte. Die Qualifier sind eher begrenzt (**Tabelle 14**), erfüllen aber typische Anforderungen. Mit dem speziellen Qualifier CAL können aktive Schritte ohne zusätzliche Anweisungen andere Bausteine aufrufen.

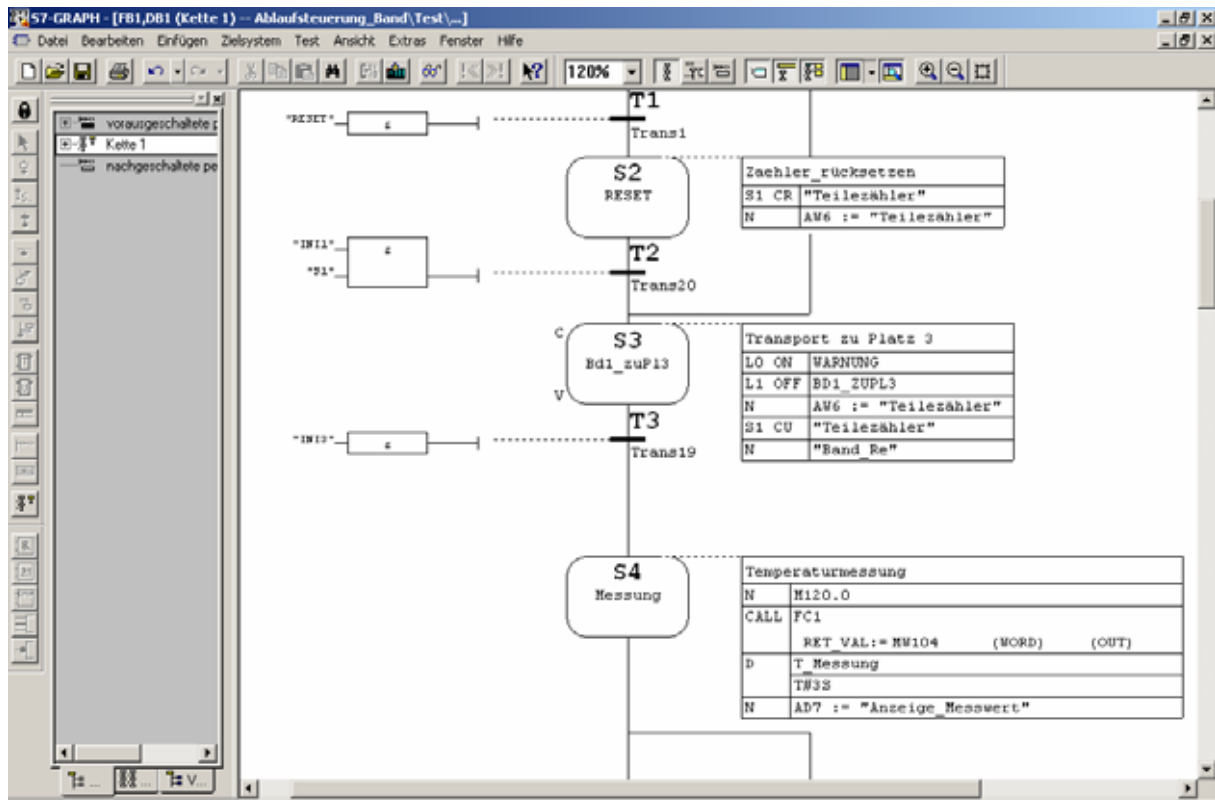
Kurzzeichen	Bedeutung	Erläuterung
N	Nicht gesetzt	Aktion wird ausgeführt, solange der Schritt aktiv ist
S	Setzen (Gespeichert)	Wird der Schritt aktiv, so wird diese Aktion gesetzt und bleibt aktiv bis zu ihrem Rücksetzen
R	Rücksetzen	Wird der Schritt aktiv, so wird diese Aktion zurückgesetzt
D	Einschaltverzögert (Delay)	Die Aktion wird eine programmierte Zeit nach Aktivierung des Schrittes ausgeführt.
L	Zeitbegrenzt (Limited)	Ist der Schritt aktiv, so wird die Aktion für eine programmierte Zeit ausgeführt
CAL	Aufruf einer POE	Solange der Schritt aktiv ist, wird eine POE ausgeführt.

Tabelle 14: Die Standardaktionen nach Graph 7

Von besonderem Wert aber sind die Möglichkeiten, Aktionen ereignisabhängig zu gestalten. Hierzu werden die Qualifier ergänzt mit den Kennzeichen „S“ (Schritt), „V“ (Supervision) und „L“ sowie „C“ (Interlock bzw. Condition).

Mit Interlock werden Schritte verriegelt. Ein programmierter Interlock kennzeichnet das Schrittsymbol mit „C“. Die Operationen der Aktionen werden ebenfalls mit „C“ ergänzt. Die Programmierung einer Supervision bewirkt eine Schrittüberwachung, beispielsweise seiner Zeitdauer. Überwachte Schritte werden mit „V“ gekennzeichnet.

**Bild 79** zeigt einen Abschnitt einer Graph 7 – Schrittkette, bei der solche ereignisabhängige Aktionen eingesetzt wurden. Schritt S3 wurde mit Interlock und Supervision ausgestattet. Weiter erscheint an Aktionen das Bestimmungszeichen „S“. Wird Schritt S3 aktiv, so bewirkt „S1“, daß ein Teilezähler um eine Einheit hochgezählt wird. Wird Schritt 2 aktiv, so wird dieser Zähler zurückgesetzt. Um die Programmierung des Zaehlers selbst braucht man sich ebensowenig zu kümmern wie etwa um die Programmierung eines Timers für die zeitliche Schrittüberwachung. Alle diese Aufgaben übernehmen die im Bild gezeigten Aktionen der IEC-Schritte.



#### Ergänzende Bestimmungszeichen in Graph 7

- S1: Schritt wird aktiviert
- S0: Schritt wird deaktiviert
- V1: Überwachungsfehler tritt auf (Störung)
- V0: Überwachungsfehler ist behoben (keine Störung)
- L0: Verriegelungsbedingung kommt
- L1: Verriegelungsbedingung geht (z.B. Störung)
- C: Verriegelungsbedingung ist erfüllt

Bild 79: Abschnitt einer Schrittkette in der Ablaufsprache Graph 7

Supervision und Interlock werden in der Darstellung „Einzelschritt“ programmiert (**Bild 80**). Graph 7 ist ein rein graphisches Tool, so daß die wenigen Programmanweisungen für Interlock und Supervision ebenfalls ausschließlich graphisch geschrieben werden.

Ebenso wie die Ablaufsprache des Systems CoDeSys enthält das Tool Graph 7 viele weiterführende Details, die das Erstellen von Programmen für Maschinen- und Anlagensteuerungen bedeutend erleichtern. Dem interessierten Leser mit Kenntnissen in Step 7 wird das Einarbeiten in Graph 7 dringend empfohlen. Dies kann bereits mit der integrierten Hilfe von Graph 7 erfolgen. Diese enthält neben dem „Handbuch Graph 7“ in der Rubrik „Erste Schritte“ eine vollständig ausgeführte Ablaufsteuerung am Beispiel einer Bohrmaschine von der Aufgabenstellung bis hin zum Lösungsprogramm.

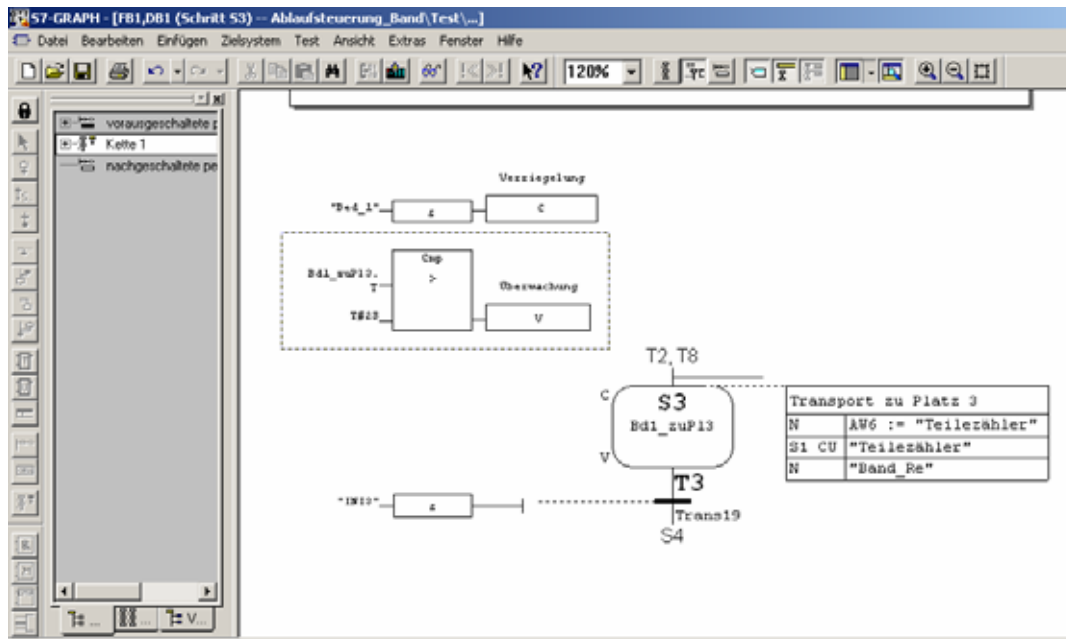


Bild 80: Programme Interlock und Supervision von Schritt S3

### Fazit:

Die Ablaufsprache (AS) ist eine der sechs nach IEC 61131-3 zugelassenen Programmiersprachen. Sie ermöglicht sehr komfortabel das Programmieren von Abläufen in Automatisierungssystemen mit streng vorgegebener Abfolge von Zuständen. Solche Aufgaben lassen sich gut mit Schrittketten beherrschen. In den Folgen 12 und 13 wurden Grundlagen der Ablaufsprache vorgestellt. Einfache Beispiele erläuterten die Elemente Schritt, Transition und Aktion sowie die Regeln für ihren Einsatz in Schrittketten. Vorrangig bestand das Ziel, Interesse für die komfortablen Programmiermöglichkeiten der Ablaufsprache zu wecken. In der nächsten Folge werden Möglichkeiten der Vernetzung von Ethernet-Controllern und Kopplern vorgestellt und aktuelle Fragen nach der Bedeutung von Ethernet und Echtzeit-Ethernet in der Automatisierungstechnik diskutiert.