

Automatisierungstechnik nach internationaler Norm programmieren

Autor: Dr. Ulrich Becker
Fachzentrum Automatisierungstechnik und vernetzte Systeme im BT Rohr-Kloster
Mail: Ulrich.Becker@BTZ-Rohr.de

Folge 2: Gestaltung einer ersten Programmorganisationseinheit

Im ersten Teil stellen wir eine steuertechnische Aufgabe 1, die mit Busklemmen von WAGO und dem Programmiersystem CoDeSys (Controller Development System) gelöst werden soll. Die dafür erforderlichen Schritte und Überlegungen werden im folgenden Teil fortgeführt. Schritt für Schritt wird die Praxis einer IEC-gerechten Programmierung vorgestellt und soweit möglich mit dem Simatic Manager und Step7 verglichen.

In einem umfassenden, nach IEC 61131 gestalteten Projekt sind eine Vielzahl von Elementen im Sinne von Bausteinen zu koordinieren:

- Hardware Konfigurationen und Ressourcen
- Programm-Organisations-Einheiten
- Datentypen
- Visualisierungen

Ein Vergleich mit den Objekten eines S7-Projektes wie Stationen, Programme, MPI-, Profibus- oder Ethernet-Netze und Visualisierungsobjekte (z.B. ProTool oder WinCC) erleichtert das Verständnis. Wir werden zunächst nur diejenigen Elemente nutzen, die für die Lösung von Aufgabe 1 unverzichtbar sind. Dies sind zuerst Programm-Organisations-Einheiten als Teile eines Anwenderprogramms.

Programm-Organisations-Einheiten (POE's)

Programme werden mit POE's strukturiert, d.h. übersichtlich gestaltet. IEC 61131-3 kennt die Typen:

- Programm (PRG)
- Funktionsbaustein (FB) und
- Funktion (FUN).

Das Programmiersystem schlägt beim Editieren eines neuen Projektes zuerst das Anlegen des Bausteins PLC_PRG (PRG) vor. Dieser Baustein vom Typ Programm ist für die zyklische Programmbearbeitung grundsätzlich erforderlich. Er lässt sich mit dem Organisationsbaustein OB1 im System Step7 vergleichen.

Andere POE werden in diesem „Grundbaustein“ aufgerufen. In einer weiteren Folge werden wir klären, wann die unterschiedlichen POE-Typen einzusetzen sind und dass die Programmorganisation auch durch Task erfolgen kann.

Unser Programm für das Verteilerband soll mit weiteren Folgen wachsen. Wir strukturieren es deshalb in POE und fügen einen ersten Baustein vom Typ Funktionsbaustein mit Namen „Band_Grundfunktion“ in das Projekt ein. Diesen werden wir später im Baustein PLC_PRG aufrufen.

Gestaltung von POE's mit dem Programmiersystem CoDeSys

Die Fähigkeit, sich selbständig in neuartige Softwarewerkzeuge einzuarbeiten, ist aktueller Masstab beruflicher Qualifikation. Unterstützung findet die Fachkraft im Menu Hilfe der Software CoDeSys .

Beim Anlegen eines neuen Projektes fordert das Programmiersystem die Auswahl eines Zielsystems (Target). Durch Firmware-Updates ist sowohl die Hardware selbst als auch deren Verfügbarkeit in der Auswahlliste des Programmierertools stets auf einem aktuellen Stand zu halten. Die Vorgehensweise mit den dafür verfügbaren Softwarewerkzeugen „InstallTarget“ und „WAGO-IO-Update“ werden wir in einer späteren Folge klären.

Für den Controller unseres Trainingsrack wählen wir die Hardware 750-841. Diese Aufforderung kann ignoriert werden, wenn zunächst ein hardware-unabhängiges Programm entstehen soll. Für die spätere Kommunikation mit dem Controller müssen allerdings hardware-spezifische Einstellungen vorgenommen werden. Im Ergebnis entsteht so ein hardware-abhängiges Projekt. Beide Begriffe werden auch im Simatic-Manager streng benutzt.

Im nächsten Schritt schlägt das CoDeSys Programmiersystem das Anlegen des Bausteins PLC_PRG vor. Man kann aber unter Angabe des Bausteinnamens, des Typs und der gewünschten Sprache auch jede andere POE einfügen.

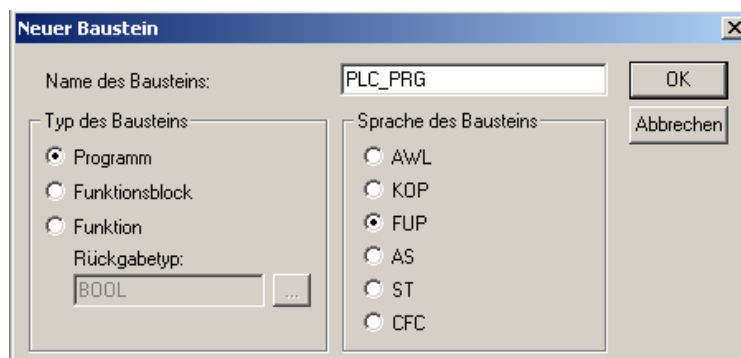


Bild 4: Anlegen von POE mit dem CoDeSys Programmsystem

Wie in **Bild 4** ersichtlich, werden 6 Sprachen zugelassen: Anweisungsliste (AWL), Kontaktplan (KOP), Funktionsbausteinsprache (FUP); Ablaufsprache (AS), Strukturierter Text (ST) und Freiprogrammierbare Funktionsbausteinsprache (CFC).

Mit Blick auf den Logikplan-Entwurf der Aufgabe 1 wählen wir die Funktionsbausteinsprache (FUP). Die Möglichkeit der grafischen Sprachen, Operatoren aus Listen auszuwählen, ist für Einsteiger von großem Vorteil. Bei den textorientierten Programmiersprachen AWL oder ST muss dagegen die Syntax aller Anweisungen selbstverantwortlich geschrieben werden. Für anspruchsvollere Aufgaben empfiehlt es sich unbedingt, sich in AWL und / oder ST einzuarbeiten! Dies ist weiteren Folgen vorbehalten.

Nun steht uns der im **Bild 5** gezeigte Bildschirm zur Bearbeitung von POE's zur Verfügung.

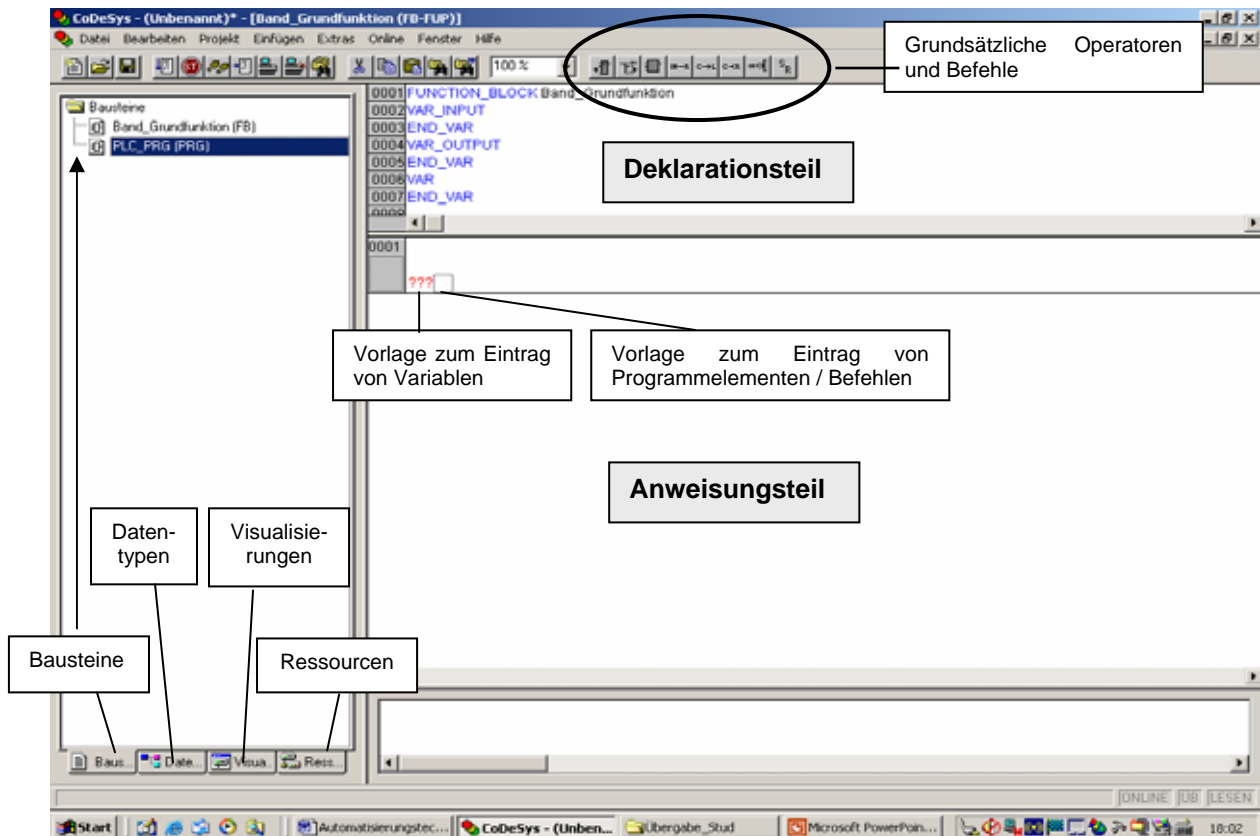


Bild 5: Der Bildschirm für die Arbeit mit dem CoDeSys Programmiersystem

Neben dem Baustein PLC_PRG (PRG) zeigt Bild 5 weiter die gewünschte POE „Band_Grundfunktion“ vom Typ FB. Diese POE wurde über die Menufunktionen > *Projekt* > *Objekt* > *Einfügen* und weiter mit den Auswahlmöglichkeiten nach Bild 4 in das Projekt eingefügt.

Als nächste Schritte sind nun im Anweisungsteil die Programm-Elemente des Logikplanes einzufügen. Dies erfolgt wie auch in den grafischen Sprachen FUP oder KOP bei Step7 in Netzwerken. Mausclick auf Elemente der Funktionsleiste oder rechter Mausclick auf markierte Blöcke erlaubt unter ->*Baustein* das Auswählen der grafischen Programmelemente.

Fügt man solche Bausteine ein, schlägt das System stets den Operator AND vor. Für andere Operatoren muss dieses Schlüsselwort mit den gewünschten Worten überschrieben werden.

IEC Operatoren

Anweisungen – auch Befehle genannt - bestehen aus Operator und Operand. Der Operator ist ein Symbol oder Schlüsselwort, welches die Aktion mit den zugehörigen Operanden bewirkt (**Tabelle 4**). Operanden werden auch als Variable bezeichnet.

Für die bitweise Signalverknüpfung von Aufgabe 1 werden zunächst Boolesche (Binäre) Operatoren benötigt. Diese sind anzuwenden auf den Datentyp BOOL.

Später werden wir die gleichen Operanden auch auf Datentypen BYTE, WORD und andere anwenden.


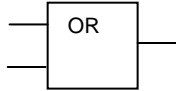

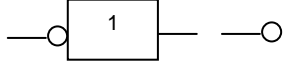

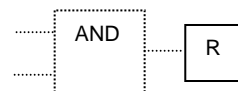
Operation		Operator	Graphische Darstellung
UND-Verknüpfung		AND (auch &)	
ODER-Verknüpfung		OR	
EXCLUSIV ODER – Verknüpfung (Antivalenz)		XOR	
NICHT-Funktion (Negation)		NOT	
SETZEN	Keine eigentlichen IEC Operatoren, in Verbindung mit einem anderen Operator und in AWL jedoch anwendbar (Siehe FB Typ RS)	S	
RUECKSETZEN		R	

Tabelle 4: IEC-Operatoren für den Datentyp BOOL

Der Ausgangswert eines Operators wird mit einer Zuweisung oder dem Setz- bzw. Ruecksetz-Befehl auf eine Variable gelegt (**Bild 6**).

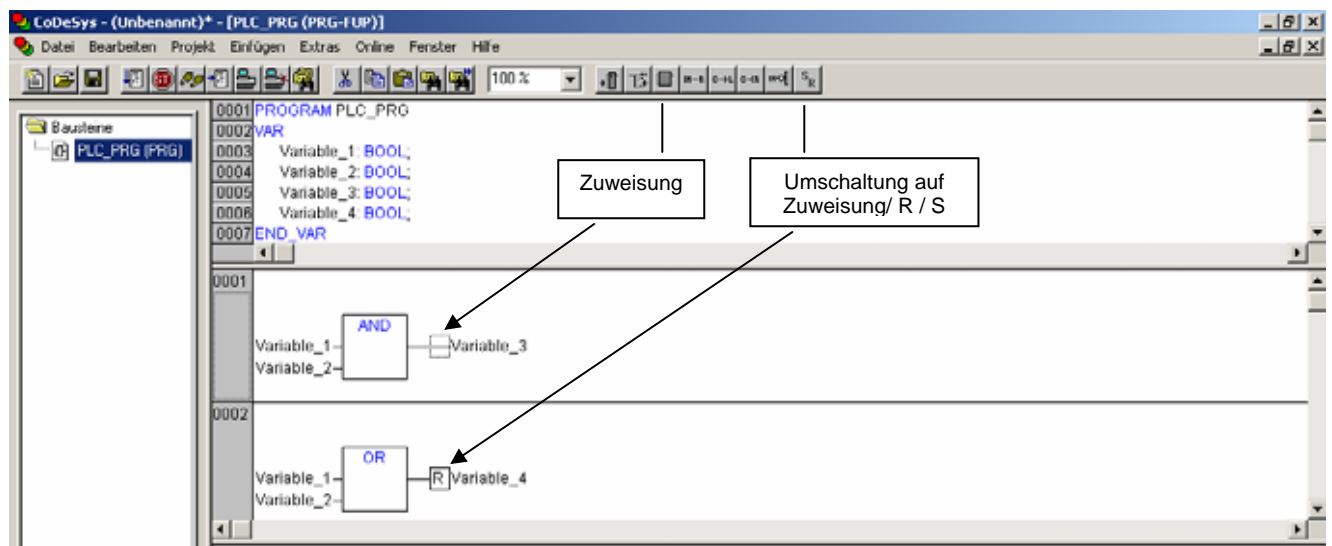


Bild 6 Einsatz der Zuweisung und des S-R-Operators im System CoDeSys

Mit dem im Teil 2 erworbenen Wissen sind wir in der Lage, den IEC-Code für Aufgabe 1 in Funktionsbausteinsprache (FUP) zu schreiben. Bei der Flankenbewertung des Lichtschranken-Signals und auch bei den gewohnten R-S-FlipFlops gibt es aber noch

Fragen. Hier müssen wir uns mit Bibliotheken, Funktionen sowie Funktionsbausteinen und deren Instanzen auseinandersetzen. Diese Betrachtungen folgen in Folge 3.

Glossar:

BOOL	Internationales Datenformat für die Datenbreite 1 Bit. Im deutschen Sprachraum spricht man eher von einem Binärwert und grenzt diese von Digitalwerten ab. Während ein Bit die Werte „0“ oder „1“ annimmt, sind die möglichen Werte des BOOL „FALSE“ und „TRUE“. Für Logik ist auch der Begriff Boolesche Algebra bekannt.
Firmware-Update	Zeitgemäße programmierbare Hardware wird durch Eintrag eines Software-Files (der „Firmware“) in ihren Eigenschaften und Schnittstellen aktualisiert. Ein Austausch der Hardware ist dann nicht notwendig. Aber auch Programmiersysteme, die eine Target-Auswahl fordern, müssen Updates erhalten, weil sonst die neuesten Geräteversionen dem System nicht bekannt sind.
Hardwareabhängiges	Programm, welches spezielle Parametrierungen der Hardware Programm nutzt, die über werksseitige Voreinstellungen hinausgehen. Beschränkt man sich auf allgemeine Voreinstellungen (Default), so entsteht ein hardware-unabhängiges Programm.
Konfiguration	ist hier zu verstehen als SPS-System
POE	Programm-Organisations-Einheit. IEC 61131-3 definiert die Typen Funktion, Funktionsbaustein und Programm. Step7 erlaubt Funktionen, Funktionsbausteine, Organisations- und Datenbausteine. Ähnlichkeiten gibt es, aber die Begriffe sind nicht identisch anzuwenden.
„Programmbaustein“	umgangssprachlich spricht man von „Bausteinen“ und meint damit die Programm-Organisations-Einheiten (POE). Weiter meint man mit Programmbausteinen aber auch Operatoren in grafischer Darstellung. Schließlich kennt Step5 Programmbausteine (PB), die in Step 7 durch Funktionen (FC) abgelöst wurden.
Ressource	ist vergleichbar mit einer Zentraleinheit (CPU). Eine Konfiguration kann so mehrere Ressourcen enthalten. Der Begriff Ressource wird aber auch für Software-Werkzeuge (Tools) benutzt.
Target	engl. Ziel. Mit Target wird eine Hardware als Zielpunkt für den Eintrag des Anwenderprogramms beschrieben.
Task	engl. in Anspruch nehmen. Aktuell wirksamer Teil des Anwenderprogramms zur Laufzeit, organisiert z.B. durch Zeitschlitze oder Prioritäten. Programmsteuerung durch Task stellt zumeist eine Alternative zur zyklischen Programmbearbeitung dar.
Visualisierung	Oberfläche auf einem PC oder TouchPanel zur Beobachtung und Steuerung von Daten. Der Begriff bezeichnet aber auch das Software-Werkzeug, mit dem diese Oberfläche gestaltet wird.
zyklische Programmbearbeitung	Überwiegende Arbeitsweise der CPU in SPS-Systemen. Befehle werden nacheinander seriell (niemals parallel!) abgearbeitet. Danach beginnt ein neuer Zyklus. Die Zeitdauer für einen Bearbeitungszyklus nennt man Zykluszeit. Diese wird zumeist überwacht (Zyklusüberwachungszeit).