

Ablaufsprache (AS) Quelle : Onlinehilfe von PC WorX der Firma Phoenix Contact

Die IEC 61131-3 definiert die Syntax von fünf Programmiersprachen und beschreibt deren Darstellung und Sprachelemente. Wir unterscheiden zwei textuelle und drei graphische Sprachen. Die **Ablaufsprache** gehört zur Kategorie der **graphischen Sprachen**.

Einführung in die Ablaufsprache

Programmcode in der graphischen Sprache AS besteht aus **Schritten** und **Transitionen**, die durch direkte Verbindungen miteinander verbunden sind. Einem Schritt können ein oder mehrere **Aktionsblöcke** zugewiesen werden.

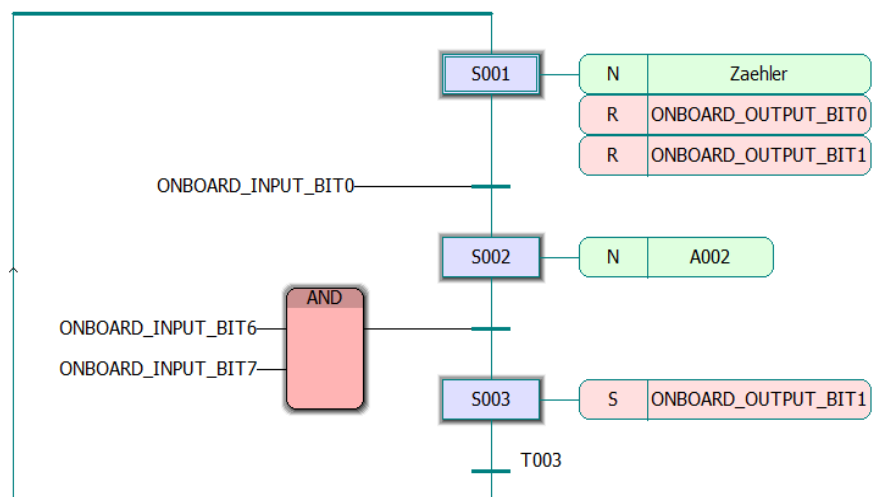
Ein **Aktionsblock** besteht aus einer Aktion und dem dazugehörigen **Qualifier**. Qualifier für Aktionen geben an, welchen Einfluss ein zugehöriger Schritt auf eine Aktion hat. Während der Schritt aktiv ist, wird die dazugehörige Aktion gemäß ihrem Qualifier ausgeführt. Eine Aktion kann entweder eine boolesche Variable oder ein in AWL/ST/KOP/FBS programmiertes Code-Arbeitsblatt (eine sogenannte 'Lupe') sein.

Eine **Transition** stellt die Bedingung dar, unter der die Abarbeitung des Codes mit dem nächsten Schritt fortgesetzt wird. Wenn eine Transition TRUE wird, wird der vorangehende Schritt noch einmal ausgeführt und dann wird der nachfolgende Schritt aktiv. Die Transition kann entweder eine einzelne boolesche Variable sein oder ein direkt verbundener boolescher Ausdruck in FBS oder KOP. Der auszuführende Code einer Transition kann auch in einem zusätzlichen Code-Arbeitsblatt editiert werden, das als Lupe bezeichnet wird.

Alle verbundenen Objekte zusammen bilden ein **AS-Netzwerk**. Ein AS-Netzwerk muss immer einen Anfangsschritt haben, der beim Aufrufen der AS-POE als erster Schritt ausgeführt wird.

AS-Netzwerke bestehen in erster Linie aus Schritten und Transitionen. Zu einer Transition gehört immer ein Schritt. Sie können keine Strukturen erstellen, in denen zwei Schritte oder zwei Transitionen direkt aufeinander folgen. Die Verbindung zwischen einem Schritt und einer Transition wird als vertikale Linie dargestellt.

In das AS-Netzwerk können simultane (gleichzeitig ausgeführte) oder alternative (alternativ ausgeführte) Zweige eingefügt werden.



Schritt in AS

Schritte sind grundlegende Elemente eines AS-Netzwerks. Einem Schritt können ein oder mehrere Aktionsblöcke zugewiesen werden.

Der Zustand eines Schritts ist entweder **aktiv** oder **inaktiv**. Aktiv bedeutet, dass der Schritt in diesem Moment von der SPS ausgeführt wird. Die dem Schritt zugewiesene Aktion wird ausgeführt. Wenn ein Schritt inaktiv wird, wird die Aktion noch einmal ausgeführt. Falls einem Schritt kein Aktionsblock zugeordnet ist, so wird dieser erst dann ausgeführt, wenn eine nachfolgende Bedingung TRUE wird.

Der Zustand des Schritts wird durch die Variable 'schrittname.x' dargestellt. 'schrittname.x' ist TRUE, wenn der Schritt aktiv ist. Diese Variable wird auch als Schrittmerker bezeichnet.

HINWEIS Die Variable 'schrittname.x' wird intern vom System erzeugt und muss nicht in der Variablendeklaration der POE deklariert werden.

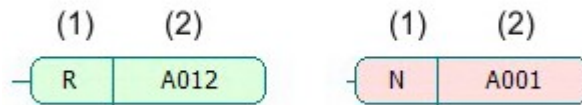
Schritte sind mit ihren vorangehenden oder nachfolgenden Transitionen über direkte Verbindungen verknüpft, die durch Linien dargestellt werden. Der Name des Schritts wird in dem Rechteck dargestellt, welches den Schritt im graphischen Editor symbolisiert.

Sie können mehrere **Schritttypen** verwenden:

- Der **Anfangsschritt** ist der erste Schritt, der beim Aufrufen der POE oder beim Starten der Konfiguration aktiviert wird. In jedem AS-Netzwerk kann nur ein Anfangsschritt verwendet werden. Anfangsschritte werden graphisch durch ein doppelt umrahmtes Rechteck dargestellt.
- **Normale Schritte** sind Schritte, die vom Programm in der im Netzwerk festgelegten Reihenfolge ausgeführt werden. Sie werden als Rechtecke dargestellt.
- **Endschritte** sind Schritte, auf die keine Transition folgt. Sie können verwendet werden, um das AS-Netzwerk zu beenden. Diese Art von AS-Netzwerken wird häufig eingesetzt, um Anwendungen zu initialisieren.
- **Sprünge** sind Schritte, die Nachfolgeelemente in anderen Arbeitsblättern derselben POE oder an einer anderen Stelle im selben Arbeitsblatt besitzen. Der Name des Sprungs gibt dabei an, wohin gesprungen wird.

Aktionsblock, Aktion und Qualifier für Aktionen in AS

Aktionsblöcke sind grundlegende Elemente in AS-Netzwerken. Ein Aktionsblock besteht aus einer Aktion (Abb. 2) und einem Qualifier (Abb.1) Bestimmungszeichen. Ein Aktionsblock ist immer einem Schritt zugewiesen.



HINWEIS : Aktionsblöcke, die einen eigenen Code besitzen, d. h. als Lupe definiert sind, werden im Editor grün dargestellt. Aktionen, die als Variablen definiert sind, erscheinen rot (siehe Bild oben).

Sie können eine Aktion mit Hilfe der Qualifier 'S' und 'R' mehreren Schritten zuweisen. Dies kann z. B. zum Setzen und Rücksetzen einer booleschen Variablen verwendet werden. Aneinandergereihte Aktionsblöcke sind mehrere Aktionsblöcke, die zu einem Schritt gehören.

Aktionen sind Teil eines Aktionsblocks. Zusammen mit dem Qualifier bildet die Aktion einen Aktionsblock in SFC. Eine Aktion kann wie folgt realisiert sein:

- als einzelne boolesche Variable oder
- als Lupe, d.h. mit eigenem Code in einer der verfügbaren Programmiersprachen.

Qualifier für Aktionen geben an, welchen Einfluss der zugehörige Schritt auf die zugewiesene Aktion hat. Die folgende Tabelle enthält die Liste der möglichen Aktions-Qualifier.

Aktions-Qualifier	Beschreibung	Funktion
N	nicht gespeichert	Der Code der Aktion wird ausgeführt oder die boolesche Variable wird gesetzt, solange der Schritt aktiv ist.
R	vorrangiges Rücksetzen	Der Code der Aktion wird nicht länger ausgeführt oder die boolesche Variable wird rückgesetzt. Die Aktion muss gesetzt werden, bevor der Qualifier 'S' verwendet wird.
S	setzen (gespeichert)	Der Code der Aktion wird ausgeführt oder die boolesche Variable wird gesetzt. Dieser Zustand wird gespeichert, sobald der Schritt aktiv wird. Er kann nur explizit rückgesetzt werden, indem dieselbe Aktion mit dem Qualifier 'R' einem anderen Schritt zugewiesen wird.
L	zeitbegrenzt	Der Code der Aktion wird ausgeführt oder die boolesche Variable wird gesetzt, solange der Schritt aktiv ist, jedoch maximal für das festgelegte Zeitintervall.

Aktions- Qualifier	Beschreibung	Funktion
D	zeitverzögert	Der Code der Aktion wird ausgeführt oder die boolesche Variable wird gesetzt, nachdem die festgelegte Zeitverzögerung abgelaufen ist. Die Aktion bleibt aktiv, solange der Schritt aktiv ist. Wenn der Schritt für einen kürzeren Zeitraum aktiv ist als die festgelegte Zeitverzögerung, wird die Aktion nicht aktiv.
P	Impuls (Flanke)	Sobald der Schritt aktiv ist, wird für einen Arbeitszyklus der Code der Aktion ausgeführt oder die boolesche Variable gesetzt.
SD	gespeichert und zeitverzögert	Der Code der Aktion wird ausgeführt oder die boolesche Variable gespeichert und gesetzt, wenn die festgelegte Zeitverzögerung nach Aktivierung des Schritts abgelaufen ist, auch wenn der Schritt inaktiv wird. Die Aktion bleibt aktiv, bis sie rückgesetzt wird. Wenn der Schritt für einen kürzeren Zeitraum aktiv ist als die festgelegte Zeitverzögerung, wird die Aktion trotzdem aktiv.
DS	verzögert und gespeichert	Der Code der Aktion wird ausgeführt oder die boolesche Variable wird gesetzt, wenn die festgelegte Zeitverzögerung nach Aktivierung des Schrittes abgelaufen ist. Die Aktion bleibt aktiv, bis sie rückgesetzt wird. Wenn der Schritt für einen kürzeren Zeitraum aktiv ist als die festgelegte Zeitverzögerung, wird die Aktion nicht aktiv.
SL	gespeichert und zeitbegrenzt	Der Code der Aktion wird für ein festgelegtes Zeitintervall ausgeführt oder die boolesche Variable gesetzt und gespeichert, sobald der Schritt aktiv ist. Wenn der Schritt kürzere Zeit aktiv ist, als das Zeitintervall andauert, ist die Aktion trotzdem für die ganze Dauer des Intervalls aktiv. Wenn die Aktion während des Zeitintervalls rückgesetzt wird, wird die Aktion inaktiv, sobald sie rückgesetzt wird.

HINWEIS Informationen zu **Eigenschaften von Aktionen** entnehmen Sie bitte der Onlinehilfe von PC WorX → Stichwort 'Eigenschaften: Aktion'

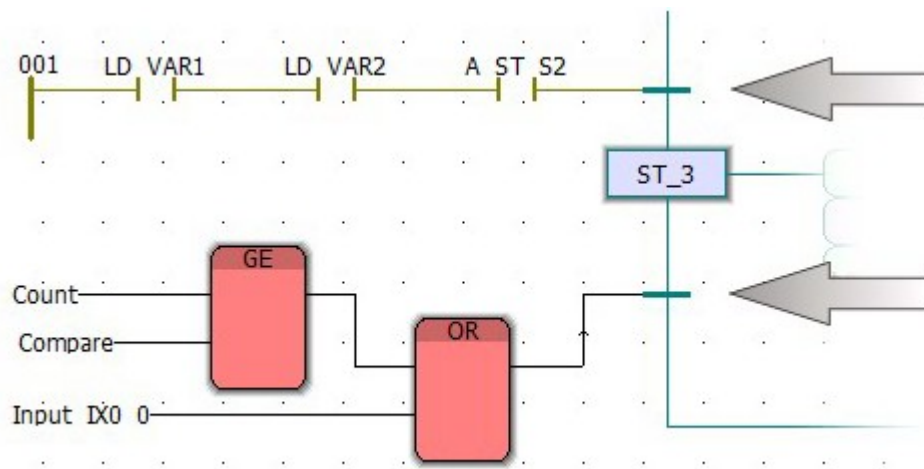
HINWEIS Informationen zu **Aktionstypen** entnehmen Sie bitte der Onlinehilfe von PC WorX → Stichwort 'Aktionstypen: Bearbeiten von Lupen und Variablen'

Transition in AS

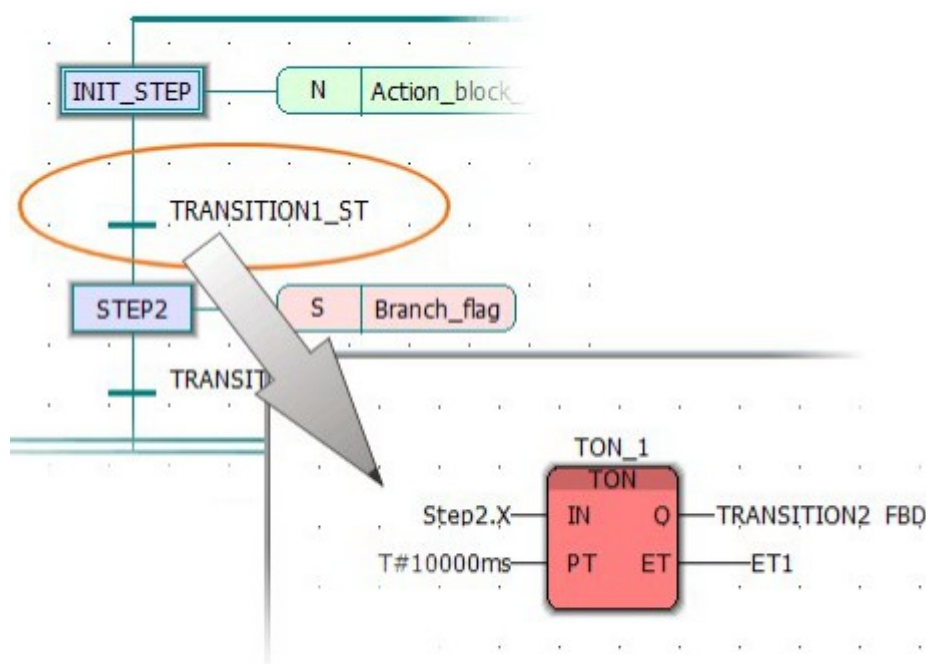
Transitionen sind grundlegende Elemente eines AS-Netzwerks. Eine Transition charakterisiert die Bedingung, wann die SPS-Abarbeitung von einem Schritt zum nächsten übergeht. Zu einer Transition gehört immer eine **Transitionsbedingung**. Diese kann eine boolesche Variable oder ein Code in AWL, ST, KOP oder FBS sein.

Transitionen können wie folgt realisiert sein:

- Als **direkte Verbindung**, d.h. mit angeschlossenen Variablen oder KOP-/FBS-Objekten aber ohne eigenes Code-Arbeitsblatt, in dem die Transitionsbedingung programmiert ist. Direkte Verbindungen haben keinen Transitionsnamen. Das folgende Beispiel zeigt zwei direkte Verbindungen.



- Als **Lupe**, d.h. mit eigenem Code in einer der verfügbaren Programmiersprachen.



Transitionen werden graphisch als horizontale Linie dargestellt.

HINWEIS Informationen zu **Transitionstypen** entnehmen Sie bitte der Onlinehilfe von PC WorX → Stichwort 'Transitionstypen: Bearbeiten von Lupen und direkten Verbindungen'