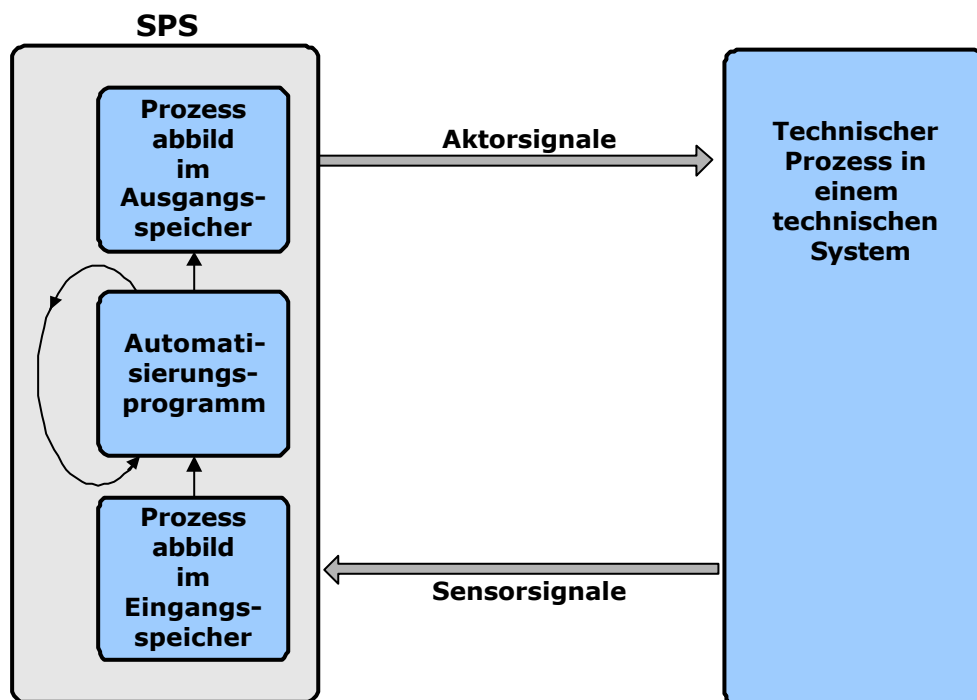


SPS-Programmierung

Arbeitsweise einer SPS

Eine SPS arbeitet in einem zyklischen Betrieb. Dabei werden zu Beginn jedes Zyklus die aktuellen Eingangsdaten eingelesen und im Speicher abgelegt. Man bezeichnet diesen Vorgang als Erstellung eines Prozessabbilds. Im Anschluss daran erfolgt die eigentliche Programmbearbeitung, d.h. die Berechnung der neuen Ausgangswerte auf Basis des abgespeicherten Prozessabbilds. Änderungen im technischen Prozess während dieser Rechenphase werden von der Steuerung (noch) nicht erkannt und auch in der Berechnung nicht berücksichtigt. Nach Beendigung der Berechnungen werden die sich im Ausgangsspeicher befindlichen Ausgangswerte an den technischen Prozess ausgegeben und der Zyklus beginnt von Neuem. Die Zykluszeit ist dabei nicht konstant sondern hängt von Programm und Ausführungspfad ab.



Bezeichner

Entsprechend der Arbeitsweise einer SPS werden 3 Arten von Größen unterschieden: Eingänge, Ausgänge und interne Größen. Bei den internen Größen handelt es sich um Elemente wie RS-FF, Zeitglieder, Merke u.a.m. Die Bezeichner für die einzelnen Elemente sind z.T. hersteller- und sprachabhängig, so dass hier zu Unterschieden in der Praxis kommen kann. Eine allgemeine Form lautet:

I<Nr>	Eingang (Input)
Q<Nr>	Ausgang (Output)
...	interne Größen

Dabei ist <Nr> eine ganze positive Zahl ist (z.B. I1, Q5....)

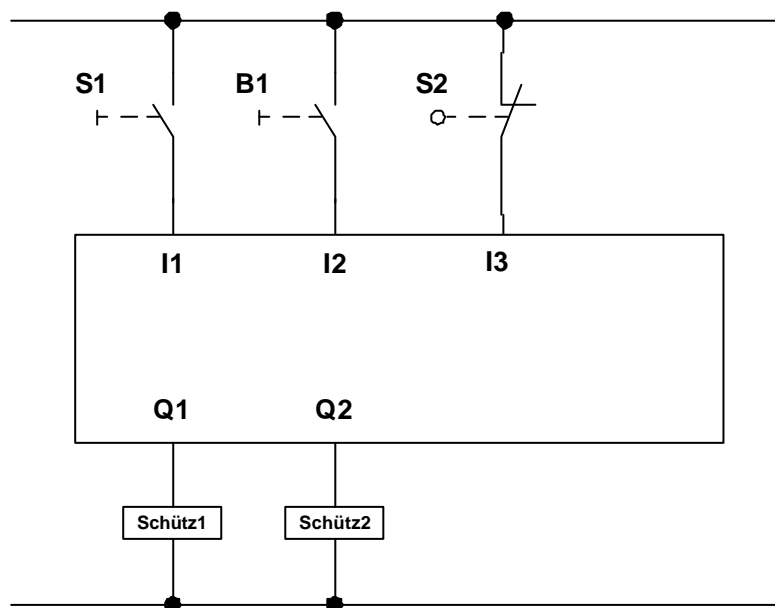
Die Firma Siemens verwendet bei ihrer weit verbreiteten Simatic Steuerung eine etwas andere Bezeichnung. Dabei wird die Nummer aus zwei Zahlen zusammengesetzt. Diese Kombination ergibt sich aus den verwendeten I/O-Modulen.

E<Nr>.<Nr>	Eingang
A<Nr>.<Nr>	Ausgang
....	interne Größen.

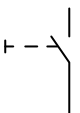
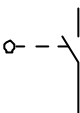
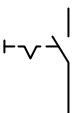
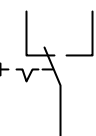
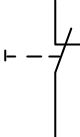
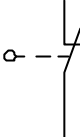


Bsp: E0.0, A4.1 u.a.m.

Beschaltung einer SPS:

Um die interne Berechnung korrekt durchführen zu können, muss der Beschaltungsplan der SPS vorliegen. Dieser gibt Aufschluss darüber, welche reale Prozessgröße auf welchen Eingang aufgeschaltet ist. Die nachfolgende Abbildung zeigt ein Beispiel dafür.



wobei die angeschlossenen Schalter wie folgt ausgeführt sein können:

Allgemein	mit Rückstellkraft		ohne Rückstellkraft: Stellschalter	
	Taster	Endschalter	Ein/Aus- schalter	Wechsel- schalter
Arbeitskontakt (Schließer)				
Ruhekontakt (Öffner)				

Programmiersprachen der SPS

Für die Programmierung von Speicherprogrammierbaren Steuerungen existiert eine Unzahl von Sprachen, da jeder Hersteller z.T. eigene Sprachen für seine Steuerungen entwirft. Es gibt allerdings 4 (größtenteils) standardisierte Sprachen:

- Strukturierter Text
- Anweisungsliste
- Kontaktplan
- Funktionsbausteinsprache

Bis auf die zuerst angeführte Sprache "Strukturierter Text" orientieren sich alle anderen an der Schaltungslogik. Dies spiegelt sich auch in der Art der Programmierung wieder.

Strukturierter Text

Bei strukturierten Text handelt es sich um eine Pascal-ähnliche Sprache die für Anwender mit Programmiererfahrung gedacht ist. Sie bietet bekannte Strukturen wie Schleifenkonstrukte, bedingte Verzweigungen usw. Sie soll an dieser Stelle nicht weiter behandelt werden.

Anweisungsliste (AWL)

Die Anweisungsliste entspricht einer textuellen Umsetzung von Logikverknüpfungen, wie UND, ODER, NICHT u.a.m. Dabei wird jede Anweisungszeile nach dem gleichen Schema aufgebaut:

Art der Verknüpfung	Bezeichner
U	E0.0

Beispiel:

Es soll ein Ausgangssignal in Abhängigkeit von 2 Eingangssignalen geschaltet werden (Ausgang = Eingang1 und Eingang2). Die zugehörige AWL ergibt sich damit zu:

U	E0.1
U	E0.2
=	A0.1

Die nachfolgende Tabelle gibt einen Auszug aus den AWL-Sprachelementen der Siemens Simatic SPS wieder ohne einen Anspruch auf Vollständigkeit zu erheben.

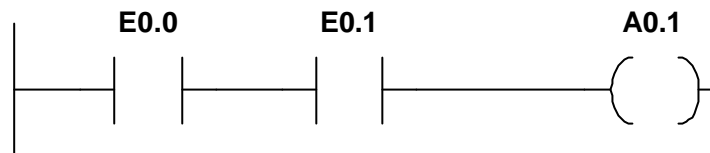
Mnemonik	Beschreibung
=	Zuweisung
)	Verzweigung schließen
AUF	Aufschlage Datenbaustein
BEA	Bausteinende absolut
BEB	Bausteinende bedingt
CALL	Bausteinaufruf
FN	Flanke negativ
FP	Flanke positiv
L	Laden
LOOP	Programmschleife
NOT	Negiere VKE
O	ODER
O(ODER mit Verzweigung
ON	ODER NICHT
ON(ODER NICHT mit Verzweigung
R	Rücksetze
S	Setze
SA	Zeit als Ausschaltverzögerung
SE	Zeit als Einschaltverzögerung
SI	Zeit als Impuls
SET	Setze VKE (=1)
SPA	Springe absolut
SPB	Springe, wenn VKE = 1
SPBN	Springe, wenn VKE = 0
SPN	Springe, wenn Ergebnis \neq 0
SS	Zeit als speichernde Einschaltverzögerung
SV	Zeit als verlängerter Impuls starten
T	Transferieren
U	UND
U(UND mit Verzweigung
UN	UND NICHT
UN(UND NICHT mit Verzweigung
X	EXCLUSIV ODER
X(EXCLUSIV ODER mit Verzweigung
XN	EXCLUSIV ODER NICHT
XN(EXCLUSIV ODER NICHT mit Verzweigung
ZR	Zählen rückwärts
ZV	Zählen vorwärts

Kontaktplan (KOP)

Der Kontaktplan ist von seiner Beschaffenheit stark an Schaltpläne angelehnt. Dabei werden UND-Verknüpfungen als Reihenschaltung und ODER-Verknüpfungen als Parallelschaltung realisiert. Dabei werden 3 Arten von Elementen unterschieden: Öffner, Schließer und Verknüpfungsergebnis. Dabei können alle vorhandenen Größen, d.h. auch interne Größen oder Ausgänge ein Verknüpfungselement darstellen und damit Schalterfunktion übernehmen.



Das oben bereits angesprochene Beispiel hätte also folgende Form:

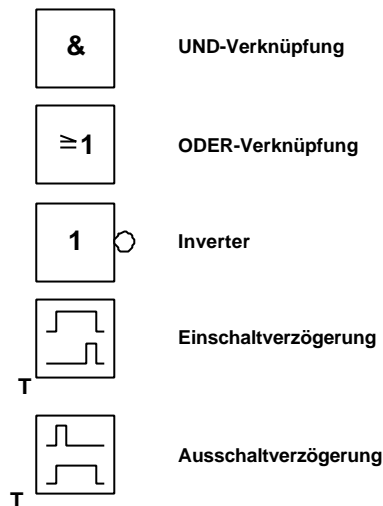


Die nachfolgende Tabelle gibt einen Auszug der KOP-Sprachelemente der Siemens Simatic SPS wieder.

Mnemonik	Beschreibung
-(ZR)	Abwärtszählen
-(ZV)	Aufwärtszählen
-(R)	Ausgang rücksetzen
-(S)	Ausgang setzen
-(AUF)	Datenbaustein öffnen
-(CALL)	FC / SFC aufrufen ohne Parameter
-(P)-	Flanke 0 -> 1 abfragen
-(N)-	Flanke 1 -> 0 abfragen
-(#)-	Konnektor
- / -	Öffnerkontakt
-()	Relaisspule / Ausgang
- -	Schließerkontakt
-(POS)	Signalflanke 0 -> 1 abfragen
-(NEG)	Signalflanke 1 -> 0 abfragen
-(JMPN)	Springe wenn 0
-(JMP)	Springe wenn 1
- NOT -	Verknüpfungsergebnis invertieren
-(SZ)	Zähleranfangswert setzen
-(SA)-	Zeit als Ausschaltverzögerung starten (SA)
-(SE)-	Zeit als Einschaltverzögerung starten (SE)
-(SI)	Zeit als Impuls starten
-(SS)	Zeit als speichernde Einschaltverzögerung starten
-(SV)	Zeit als verlängerter Impuls starten

Funktionsbausteinsprache (FBS)

Die Funktionsbausteinsprache ist etwas stärker an Logikglieder (Bausteine) angelehnt, die miteinander verknüpft werden. Die nachfolgende Liste gibt einen Überblick über standardmäßig vorhandene Bausteine ohne dabei einen Anspruch auf Vollständigkeit zu erheben.



Damit hätte das Beispiel die folgende Form:

