

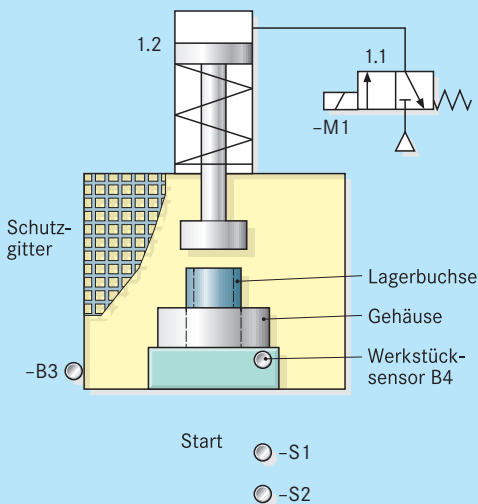
Beispiel 1: Steuerung einer pneumatischen Presse

Mit einer pneumatischen Presse werden Lagerbuchsen in Gehäuseteile eingepresst. Der Pressenzylinder 1.2 fährt aus, wenn der Handtaster S1 oder der Fußtaster S2 betätigt wird. Der Pressenzylinder darf jedoch nur dann ausfahren, wenn das Schutzgitter geschlossen ist (B3 = 1) und wenn ein Gehäusestück auf dem Pressentisch positioniert ist (B4 = 1). Ist die Startbedingung nicht mehr erfüllt, fährt der Pressenzylinder ein.

Alle Schalter (S1, S2, B3, B4) sind Schließer.

Das Programm soll in FBS und AWL erstellt werden.

Technologieschema:



Bei umfangreicheren Steuerungsaufgaben werden die Grundverknüpfungen UND, ODER, NICHT miteinander kombiniert.

Vor der Erstellung des Steuerungsprogramms wird die Zuordnungsliste erstellt. Sie zeigt die Belegung der Eingänge und der Ausgänge an der SPS.

Zuordnungsliste:

Adresse	Objektkennzeichnung	Symbolische Bezeichnung	Datentyp	Kommentar
Eingänge				
E0.1	S1	„Handtaster“	BOOL	Handtaster, Schließer
E0.2	S2	„Fußtaster“	BOOL	Fußtaster, Schließer
E0.3	B3	„Schutzgitter“	BOOL	Abfrage Schutzgitter, Schließer
E0.4	B4	„Werkstück“	BOOL	Abfrage Werkstück, Schließer
Ausgänge				
A6.1	M1	„Magnetventil“	BOOL	Zylinder fährt aus bei M1 = 1

Wahrheitstabelle:

Zeile	E0.4	E0.3	E0.2	E0.1	A6.1
0	0	0	0	0	0
1	0	0	0	1	0
2	0	0	1	0	0
3	0	0	1	1	0
4	0	1	0	0	0
5	0	1	0	1	0
6	0	1	1	0	0
7	0	1	1	1	0
8	1	0	0	0	0
9	1	0	0	1	0
10	1	0	0	1	0
11	1	0	1	1	0
12	1	1	0	0	0
13	1	1	0	1	1
14	1	1	1	0	1
15	1	1	1	1	1

Anschließend wird die Wahrheitstabelle erstellt. Zunächst werden die möglichen Kombinationen der Eingänge erfasst. Bei vier Eingängen ergeben sich $2^4 = 16$ Möglichkeiten. Diese entsprechen den Dualzahlen von 0 bis 15. Jeder Eingangskombination muss nun, entsprechend der Aufgabenstellung, das Ausgangssignal „0“ oder „1“ zugewiesen werden.

Die Presse fährt aus, wenn einer der beiden Starttaster S1 oder S2 betätigt ist und das Schutzgitter (B3) geschlossen ist und ein Gehäusestück (B4) auf dem Pressentisch liegt (Zeile 13/Zeile 14).

Die Presse fährt aus, wenn beide Starttaster S1 und S2 betätigt sind und das Schutzgitter (B3) geschlossen ist und ein Gehäusestück (B4) auf dem Pressentisch liegt (Zeile 15).

Da nur für diese Zustände die Presse ausfahren darf, wird hier ein „1“-Signal für die Ausgänge eingetragen. Die anderen Ausgänge erhalten ein „0“-Signal.

Die UND-vor-ODER-Normalform, auch disjunktive Normalform genannt, ist eine geeignete Methode, um aus der Wahrheitstabelle ein Steuerungsprogramm zu entwickeln. Diese Methode basiert auf den Zeilen, bei denen der Ausgang ein „1“-Signal hat.

! Innerhalb jeder Zeile, bei der der Ausgang „1“-Signal hat, werden die Eingänge mit UND verknüpft.

Hat ein Eingang den Signalzustand „0“, so wird er negiert in die UND-Verknüpfung aufgenommen.

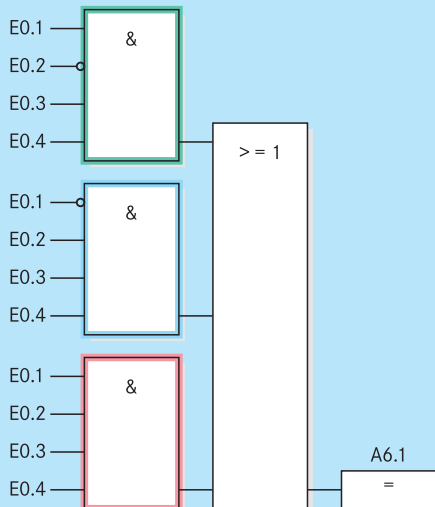
Hat ein Eingang den Signalzustand „1“, so wird er nicht negiert in die UND-Verknüpfung aufgenommen.

Das komplette Steuerungsprogramm erhält man, indem man anschließend alle UND-Verknüpfungen auf eine ODER-Verknüpfung führt.

Für die Zeile 13 bedeutet dies, dass alle vier Eingänge mit UND verknüpft werden, wobei der Eingang E0.2 negiert werden muss. In der Zeile 14 muss der Eingang E0.1 negiert werden, in Zeile 15 kommt keine Negation vor. Anschließend werden die UND-Verknüpfungen auf eine ODER-Verknüpfung geführt.

Das Programm ist in Funktionsbausteinsprache und als Anweisungsliste dargestellt. Die Funktionsbausteinsprache zeigt die Logik in übersichtlicher Form. Jede UND-Verknüpfung entspricht dabei einer Zeile der Wahrheitstabelle.

Ausführliches Programm in FBS:



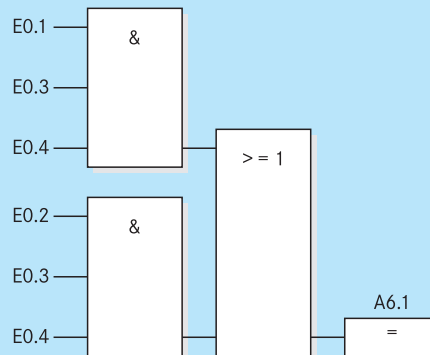
Die UND-vor-ODER-Normalform liefert häufig ein sehr umfangreiches Steuerungsprogramm. In vielen Fällen kann das Programm anschließend vereinfacht werden.

Eine erste Vereinfachung ergibt sich, indem man die UND-Verknüpfung der Zeile 15 einspart. Diese Kombination ist bereits in den Zeilen 13 und 14 enthalten, wenn man dort die negierten Eingänge E0.1 und E0.2 entfernt.

Ausführliches Programm in AWL:

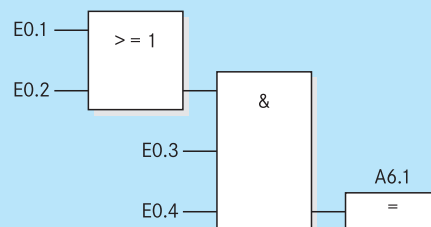
U	E	0.1
UN	E	0.2
U	E	0.3
U	E	0.4
O		
UN	E	0.1
U	E	0.2
U	E	0.3
U	E	0.4
O		
U	E	0.1
U	E	0.2
U	E	0.3
U	E	0.4
=	A	6.1

Erste Vereinfachung des Programms:



Eine weitere Vereinfachung ergibt sich durch die folgende Überlegung: Der Pressenzylinder fährt nur dann aus, wenn das Schutzgitter geschlossen ist und der Werkstücksensor ein „1“-Signal meldet. Zusätzlich muss einer der beiden Starttaster betätigt sein.

Zweite Vereinfachung des Programms:



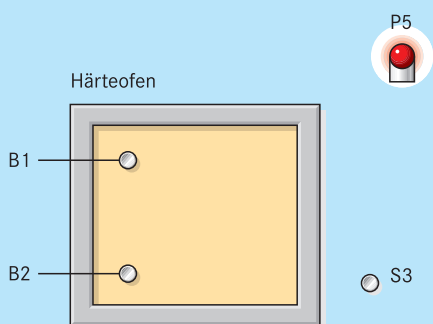
Beispiel 2: Temperaturüberwachung

In einem Härteofen wird die Temperatur aus Sicherheitsgründen durch zwei Signalglieder (B1, B2) überwacht. Ist die Temperatur zu hoch, so wird dies von den Signalgliedern gemeldet. Sobald die Temperatur überschritten wird, soll die Signallampe P5 leuchten. Um die Funktion der Signallampe P5 zu überprüfen, wird ein weiterer Taster S3 eingesetzt. Wird der Taster S3 betätigt, so leuchtet die Signallampe P5.

Die Signalglieder B1 und B2 sind Öffner und liefern bei Betätigung ein „0“-Signal. Der Taster S3 ist ein Schließer und liefert bei Betätigung ein „1“-Signal.

Das Programm soll in FBS und AWL erstellt werden.

Technologieschema:



Wahrheitstabelle:

Zeile	E1.3	E1.2	E1.1	A7.1
0	0	0	0	1
1	0	0	1	1
2	0	1	0	1
3	0	1	1	0
4	1	0	0	1
5	1	0	1	1
6	1	1	0	1
7	1	1	1	1

Die Wahrheitstabelle zeigt, dass bis auf die Zeile 3 alle Eingangskombinationen am Ausgang ein „1“-Signal führen. Erstellt man entsprechend dem Beispiel 1 die UND-vor-ODER-Normalform, so würde dies zu einem sehr umfangreichen Steuerungsprogramm führen.

Für diese Art von Wahrheitstabellen wird daher die ODER-vor-UND-Normalform, auch *konjunktive Normalform* genannt, angewandt. Diese Methode basiert auf den Zeilen, bei denen der Ausgang ein „0“-Signal hat.

Innerhalb jeder Zeile, bei der der Ausgang „0“-Signal hat, werden die Eingänge mit ODER verknüpft.



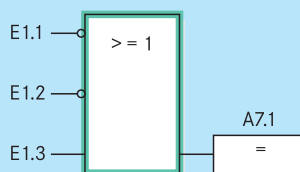
Hat ein Eingang den Signalzustand „0“, so wird er nicht negiert in die ODER-Verknüpfung aufgenommen.

Hat ein Eingang den Signalzustand „1“, so wird er negiert in die ODER-Verknüpfung aufgenommen.

Das komplette Steuerungsprogramm erhält man, indem man anschließend alle ODER-Verknüpfungen auf eine UND-Verknüpfung führt.

Da im vorliegenden Beispiel lediglich die Zeile 3 entsprechend den Vorschriften umgesetzt werden muss, entfällt die UND-Verknüpfung.

Programm in FBS:



Programm in AWL:

ON	E	1.1
ON	E	1.2
O	E	1.3
=	A	7.1

Zuordnungsliste:

Adresse	Objektkennzeichnung	Symbolische Bezeichnung	Datentyp	Kommentar
Eingänge				
E1.1	B1	„Temp-1“	BOOL	Temperatur zu hoch, Öffner
E1.2	B2	„Temp-2“	BOOL	Temperatur zu hoch, Öffner
E1.3	S3	„Funktionsprüfung“	BOOL	Funktionsprüfung von P5, Schließer
Ausgänge				
A7.1	P5	„Signallampe“	BOOL	Signallampe Temperaturüberwachung