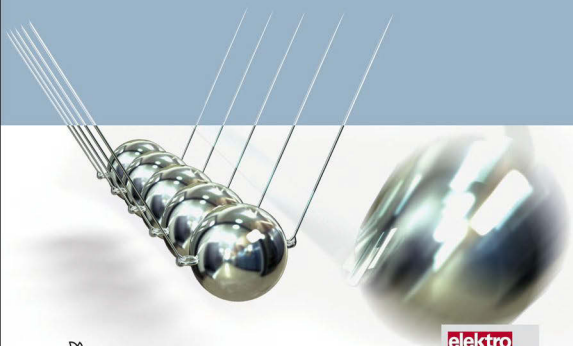


Karl-Michael Schoop

GRAF CET-Workshop interaktiv



Vogel Buchverlag

**elektro
technik**

Dieses Buch wird mit einer CD-ROM ausgeliefert, die wir Ihnen gerne kostenlos zusenden:

buch@vogel-buchverlag.de

Tel.: 0931-418-2419

Fax: 0931-418-2660

Der Onlineservice InfoClick bietet unter www.vogel-buchverlag.de nach Codeeingabe eventuell zusätzliche Informationen und Aktualisierungen. Fordern Sie für Ihr E-Book den Code unter buch@vogel-buchverlag.de an.

Dr.-Ing. Karl-Michael Schoop

GRAFCET-Workshop interaktiv

GRAFCET-Kurs mit interaktiver Lernsoftware

Vogel Buchverlag

Dr.-Ing. Karl-Michael Schoop

- 1972–1979 Studium der Mathematik
- 1980–1983 Mitarbeit bei Unternehmensberatungen:
Erstellung und Programmierung von technischen Systemen zur Mustererkennung,
Programmierung von Steuerungen für eine Müllverbrennungsanlage
- 1983–1989 Wissenschaftlicher Mitarbeiter an der Technischen Universität Hamburg-Harburg im
Arbeitsbereich Regelungstechnik und Systemdynamik
- Seit 1989 Geschäftsführer der Ingenieurbüro Dr.-Ing. Schoop GmbH
Dozent an der HAW (Hochschule für angewandte Wissenschaften Hamburg) im
Bereich Steuerungs- und Regelungstechnik
Dozent an der Hochschule Bremen für Mathematik

Weitere Informationen: www.vogel-buchverlag.de



<http://twitter.com/vogelbuchverlag>



www.facebook.com/vogel.buchverlag



www.vogel-buchverlag.de/rss/buch.rss

ISBN 978-3-8343-3330-8

1. Auflage. 2014

Alle Rechte, auch der Übersetzung, vorbehalten. Kein Teil des Werkes darf in irgendeiner Form (Druck, Fotokopie, Mikrofilm oder einem anderen Verfahren) ohne schriftliche Genehmigung des Verlages reproduziert oder unter Verwendung elektronischer Systeme verarbeitet, vervielfältigt oder verbreitet werden. Hiervon sind die in §§ 53, 54 UrhG ausdrücklich genannten Ausnahmefälle nicht berührt.

Printed in Germany

Copyright 2014 by

Vogel Business Media GmbH & Co. KG, Würzburg

Fotolia-Titelgrafik: © arsdigital - Fotolia.com

Vorwort

GRAFCET ist eine nach EN 60 848 normierte grafische technologieunabhängige Beschreibungssprache zur Darstellung von Abläufen, Steuerungen und Ablaufsteuerungen. Seit 2002 ersetzt GRAFCET den nach DIN 40 719 normierten Funktionsplan.

GRAFCET dient denen, die das Verhalten eines Systems festlegen müssen, z.B. für die Steuerungen von automatisierten Prozessen und Maschinen. Als Technologie-unabhängige Entwurfssprache ist GRAFCET auch ein Verständigungsmittel zwischen Konstrukteuren, Inbetriebnehmern und Anwendern von automatisierten Systemen.

GRAFCET ist keine Programmiersprache für Speicherprogrammierbare Steuerungen SPS, wie die Ablaufsprache (SFC – Sequential Function Chart) nach EN 61 131. GRAFCET beschreibt Steuerungssysteme und Abläufe und ist unabhängig von der eingesetzten Technologie (Hardware). Als Technologie-unabhängige, hierarchisch gegliederte Spezifikationsprache zur Planung und Beschreibung von Abläufen ist GRAFCET eine Abstraktionsebene höher als Programmiersprachen für SPS (Speicherprogrammierbare Steuerungen).

In den Berufen Elektroniker, Mechatroniker und Chemikant müssen Auszubildende seit 2008 in den PAL-Prüfungen Grundkenntnisse in GRAFCET nachweisen.

Erforderliche Grundkenntnisse

Damit Sie mit diesen Unterlagen arbeiten können, benötigen Sie allgemeine Kenntnisse im Bereich der Automatisierung.

Umfang des Tutorials

Die Beispiele und Aufgaben dieses Tutorials können nachvollzogen und bearbeitet werden mit der beigefügten Lernsoftware GRAFCET-Kursus.

In Kapitel 7 finden Sie einen Schnelleinstieg in diese Lernsoftware. Auf der beiliegenden CD-ROM sind im Unterverzeichnis «Videos» unterstützende Videos enthalten, die das Arbeiten mit der Software an einigen Beispielen demonstrieren.

Inhaltsverzeichnis

Vorwort	5
1 Einführung in GRAFCET	9
1.1 Was ist GRAFCET?	9
1.2 Warum eine neue Norm?	10
1.3 Steuerungs- und Regelungssystem	10
1.4 Aufbau von GRAFCET	11
1.5 Bildungs- und Ablaufregeln	13
2 Beschreibung der GRAFCET-Elemente mit Beispielen	15
2.1 Schritte	15
2.2 Transition und Transitionsbedingung	16
2.3 Zeitabhängige Transitionsbedingung	17
2.4 Beispiele für Transitionsbedingungen	18
2.5 Aktionen	21
2.6 Kontinuierlich wirkende Aktion	22
2.7 Kontinuierlich wirkende Aktion mit Zuweisungsbedingung	24
2.8 Kontinuierlich wirkende Aktion mit zeitabhängiger Zuweisungsbedingung	25
2.9 Gespeichert wirkende Aktion bei Aktivierung	29
2.10 Gespeichert wirkende Aktion bei Deaktivierung	31
2.11 Gespeichert wirkende Aktion bei Ereignis	33
2.12 Grafische Darstellung von Ablaufstrukturen	34
2.12.1 Ablaufkette	34
2.12.2 Geschlossene Ablaufkette	35
2.12.3 Alternative Verzweigung	37
2.12.4 Parallele (simultane) Verzweigung	40
2.13 Strukturieren von GRAFCET durch Hierarchien	44
2.14 Makro und Makroschritte	44
2.15 Einschließender Schritt mit Teil-GRAFCETs	47
2.16 Zwangssteuernde Befehle mit Teil-GRAFCETs	48
3 Beispielhafte Aufgaben und Lösungen	51
3.1 Aufgabe mit analogen Parametern	52
3.2 Aufgabe Lichterkette	53

3.3	Aufgabe Ampelsteuerung	55
3.4	Aufgaben mit Taster / Schalter	59
3.5	Wechselschaltung	62
3.6	Wendeschaltung	63
3.7	Alarmquittierung	64
4	Übersicht der Aufgaben	65
5	Einführung in die Boolesche Algebra für die Steuerungstechnik	67
5.1	Grundverknüpfungen	68
5.2	Axiome und Gesetze	70
5.3	Funktionen von zwei Variablen	71
5.4	Grundfunktionen	79
6	Die Lernsoftware GRAFCET-Kursus	81
7	Schnelleinstieg Lernsoftware GRAFCET-Kursus	85
7.1	Ausführliche Beschreibung zum Bearbeiten von Beispiel 1	85
7.2	Ausführliche Beschreibung zum Bearbeiten von Beispiel 14	95
7.3	Einstellen der kontinuierlich wirkenden Aktion mit Zuweisungsbedingung, Beispiel 17	102
7.4	Einstellen der gespeichert wirkenden Aktion, Beispiel 23, Beispiel 24	103
7.5	Arbeiten mit Makros, Beispiel 39	105

1 Einführung in GRAFCET

1.1 Was ist GRAFCET?

GRAFCET ist eine nach EN 60 848 normierte grafische technologieunabhängige Beschreibungssprache zur Darstellung von Abläufen, Steuerungen und Ablaufsteuerungen.

Seit 2002 ist GRAFCET der Nachfolger des nach DIN 40 719 normierten Funktionsplans.

Mit GRAFCET steht erstmals eine europaweit gültige Darstellungsform für Steuerungsabläufe zur Verfügung.

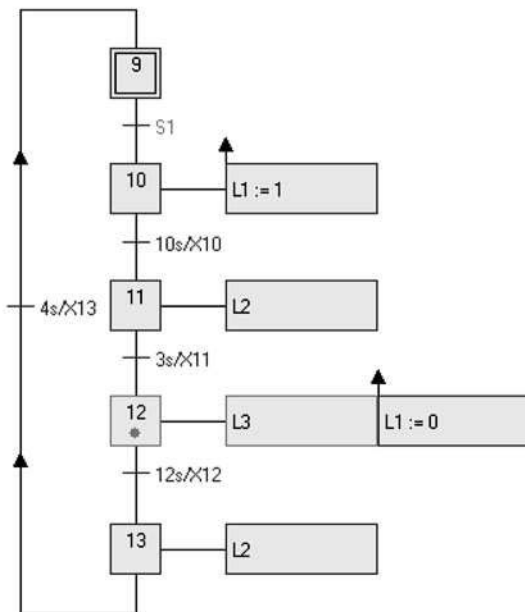


Bild 1.1 Muster eines GRAFCET-Plans

GRAFCET kommt aus dem Französischen und ist eine Abkürzung von: *Grappe Fonctionnel de Commande Etape Transition*. In der deutschen Übersetzung der Norm wird GRAFCET bezeichnet als: Spezifikationssprache für Funktionspläne der Ablaufsteuerung.

GRAFCET soll denen dienen, die das Verhalten eines Systems festlegen müssen, z.B. für die Steuerungen von automatischen Prozessen und Maschinen. Als technologieunabhängige Entwurfssprache ist GRAFCET auch ein Verständigungsmittel zwischen Konstrukteuren, Inbetriebnehmern und Anwendern von automatisierten Systemen.

GRAFCET ist keine Programmiersprache für speicherprogrammierbare Steuerungen wie die Ablaufsprache (SFC – *Sequential Function Chart*) nach EN 61 131. GRAFCET beschreibt Steuerungssysteme und Abläufe und ist unabhängig von der eingesetzten Technologie (Hardware). Als technologieunabhängige, hierarchisch gegliederte Spezifikationssprache zur Planung und Beschreibung von Abläufen ist GRAFCET eine Abstraktionsebene höher als Programmiersprachen für speicherprogrammierbare Steuerungen.

1.2 Warum eine neue Norm?

- Unklare und widersprüchliche Stellen wurden eliminiert.
- Fehlende bzw. neue Inhalte wurden erfasst und genormt.
- Eine Hierarchieebene wurde eingeführt.
- Die Beschreibungen der Aktionen und Abläufe wurden vereinfacht.
- GRAFCET ist eine europaweite Norm und ist deshalb für den internationalen Markt von großem Vorteil.

Mit GRAFCET können Ablaufsteuerungen in strukturierter Form erstellt werden. Die Unterteilung von GRAFCET-Plänen in Teil-GRAFCETs und deren Einsatz in Makros, Zwangssteuerungen und einschließenden Schritten ermöglicht eine übersichtliche Strukturierung auch von komplexen Abläufen (Ablaufsteuerungen).

1.3 Steuerungs- und Regelungssystem

Ein Steuerungs- und Regelungssystem besitzt analoge und binäre Prozesseingänge und Prozessausgänge. Diese werden vom Prozess, einer Maschine oder einer Anlage eingelesen bzw. an diese ausgegeben.

- Beispiele für binäre Eingänge:
Schalter, Taster, Rückmeldungen, Endschalter, Positionsgeber;
- Beispiele für binäre Ausgänge:
Ventile, Antriebe, Motoren, Pumpen;
- Beispiele für analoge Eingänge (Sensoren):
Temperatur (PT100), Druck, Position, Durchfluss, Füllstand;
- Beispiele für analoge Ausgänge (Stellsignale):
FU, verstellbare Ventile, drehzahlverstellbare Motoren, drehzahlverstellbare Pumpen.

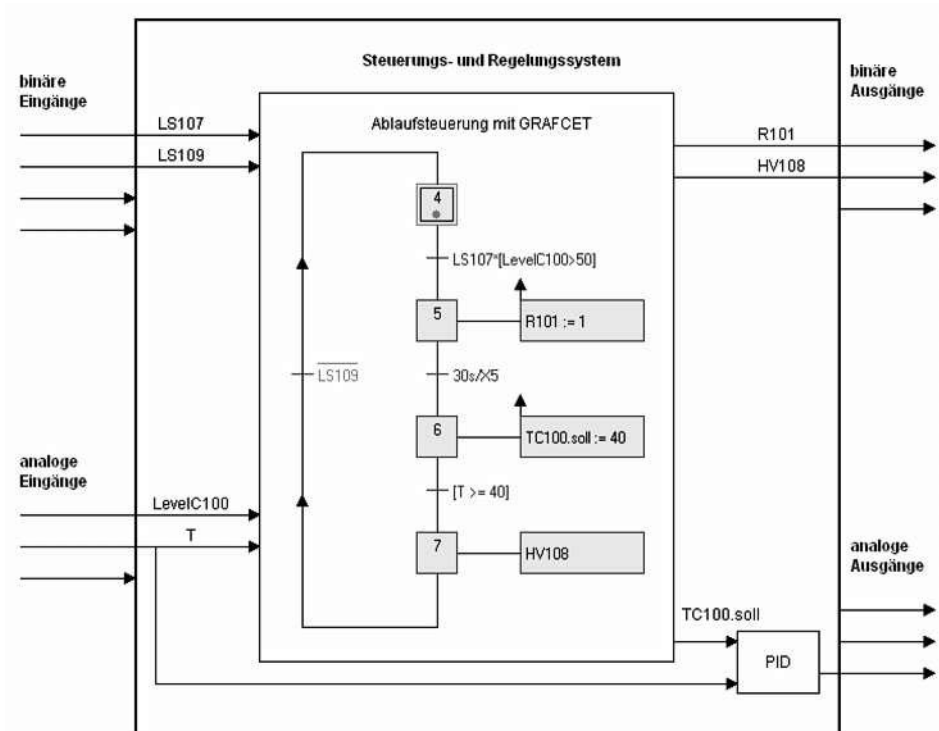


Bild 1.2 Steuerungs- und Regelungssystem mit GRAFCET als Ablaufteil

Abhängig vom Zustand der Ein- und Ausgänge und dem Zustand der Steuerung werden Steuerungsschritte bestimmt und ausgeführt. Bild 1.2 beschreibt die Beziehungen zwischen den Ein- und Ausgängen mit GRAFCET und einem PID-Regler.

1.4 Aufbau von GRAFCET

GRAFCET, als grafische Beschreibungssprache von Abläufen und (Ablauf-) Steuerungen, basiert auf Schritten, Transitionen (Weiterschaltbedingungen), Wirkverbindungen und Aktionen. In GRAFCET können mehrere Schritte gleichzeitig aktiv sein. Die Fortschrittsbedingungen von einer Gruppe von Schritten zur nächsten werden bestimmt durch Wirkverbindungen und Transitionen (Bild 1.3).

GRAFCET unterteilt sich in den Ablauf der Steuerungen, bestimmt durch die Schritte und Transitionen, und die auszuführenden Befehle (Aktionen).

GRAFCET unterscheidet:

- den Strukturteil und
- den Wirkungsteil.

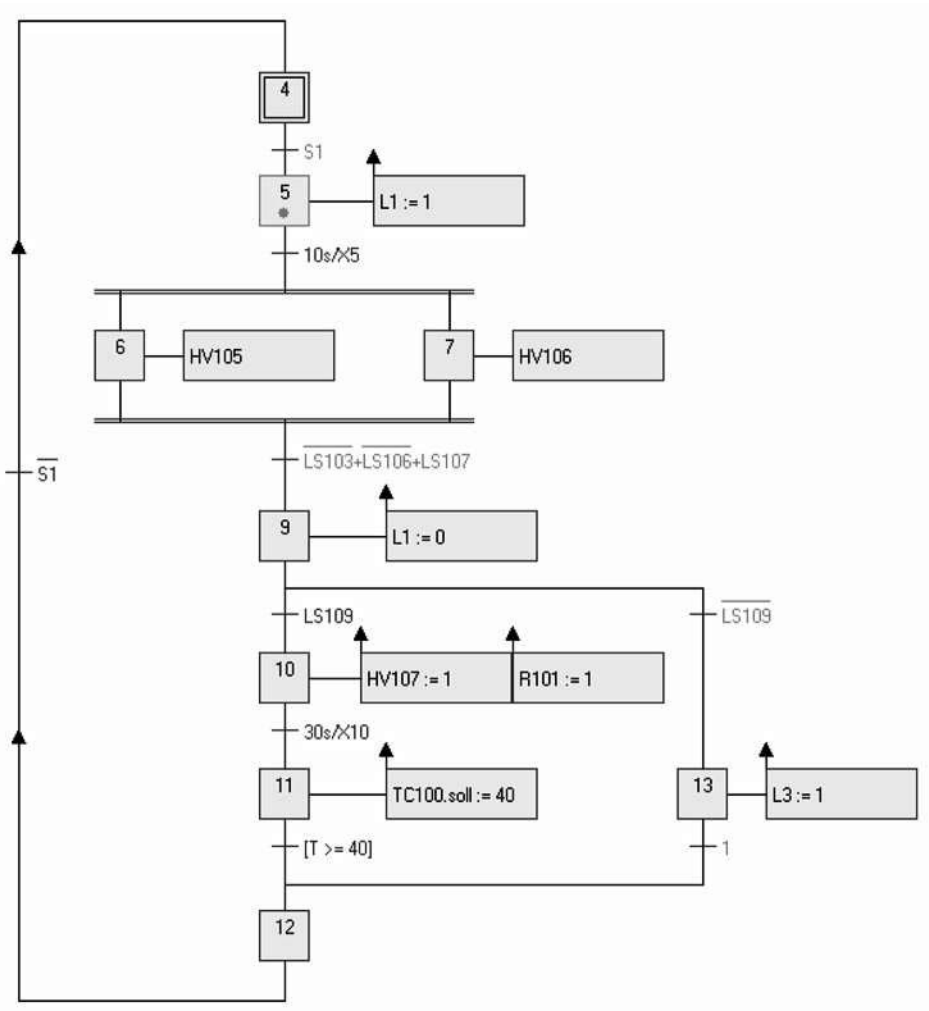


Bild 1.3 GRAFCET-Plan mit alternativer und simultaner Verzweigung

Der **Strukturteil** beschreibt den zeitlichen Ablauf der Steuerung durch Schritte, Transitionen und Wirkverbindungen.

Der **Wirkungsteil** besteht aus den Aktionen, die anzeigen, was mit den Signalen (Ausgangssignale, interne Signale) geschehen soll und den Transitionsbedingungen, die abhängig vom Zustand der Ein- und Ausgangssignale und der internen Signale den Zustand **True** (1) oder **False** (0) annehmen.

1.5 Bildungs- und Ablaufregeln

- ❑ Jeder GRAFCET besteht aus Schritten, Transitionen, Wirkverbindungen und Aktionen.
- ❑ Schritte und Transitionen müssen sich abwechseln. Eine Wirkverbindung darf nur einen Schritt mit einer Transition bzw. eine Transition mit einem Schritt verbinden.
- ❑ Jedem Schritt können eine oder mehrere Aktionen zugeordnet werden. Sie werden ausgeführt, wenn der Schritt aktiv ist.
- ❑ Einem Schritt muss keine Aktion zugeordnet werden. Zur Strukturierung und zum besseren Verständnis kann es sinnvoll sein, Schritte ohne Aktionen einzusetzen.
- ❑ Abläufe können sich verzweigen und wieder zusammengeführt werden. Man unterscheidet simultanen (parallelen) und alternativen Ablauf.
- ❑ Ein Schritt wird aktiv, wenn der vorherige Schritt aktiv ist und die zugehörige Transition erfüllt wird.
- ❑ Wird ein Schritt aktiv, so wird der vorherige Schritt deaktiviert.
- ❑ Nach Start des GRAFCETs werden die als Startschritte (doppelter Rahmen) gekennzeichneten Schritte aktiv.

2 Beschreibung der GRAFCET-Elemente mit Beispielen

Im Folgenden werden die genormten GRAFCET-Elemente beschrieben, und ihre Funktionsweisen werden anhand von Beispielen dargestellt. Die Beispiele können mit der mitgelieferten Lernsoftware GRAFCET-Kursus nachvollzogen werden.

2.1 Schritte

Ein GRAFCET ist unterteilt in Schritte. Ein Schritt ist aktiv (Zustand 1) oder inaktiv (Zustand 0). Der Zustand eines Schrittes kann durch die Schrittvariable X^* abgefragt werden (* ist die Schrittnummer).

GRAFCET unterscheidet Schritte und Anfangsschritte, die durch einen doppelten Rahmen gekennzeichnet werden.

Anfangsschritte werden beim Start einer GRAFCET-Seite aktiv gesetzt.

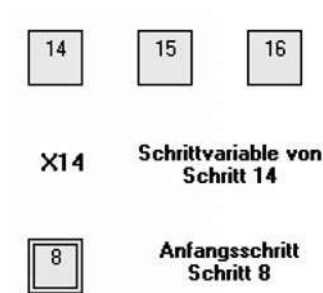


Bild 2.1 Schritte in GRAFCET

2.2 Transition und Transitionsbedingung

Eine Transition bestimmt den Übergang zwischen zwei Schritten.

Die Transitionsbedingung gibt die Weichschaltbedingung zwischen zwei Schritten an. Die Transitionsbedingung ist ein boolescher Ausdruck (Term) und kann **True** (1, wahr) oder **False** (0, falsch) sein.

Beispiel 1:

Der erste Schritt ist ein Anfangsschritt. Der zweite Schritt wird angenommen, wenn der Taster S4 gedrückt wird.

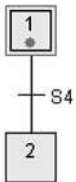


Bild 2.2 Beispiel 1



HINWEIS

Eine ausführliche Beschreibung zum Bearbeiten und zum Überprüfen des Ablaufs von Beispiel 1 mit der Lernsoftware GRAFCET-Kursus finden Sie in Abschnitt 7.1.

Beispiel 2:

In dem Beispiel bedeutet der Term $S1 \cdot S2$:

Signal S1 und Signal S2 müssen **True** (1) sein, damit der Gesamtausdruck **True** (1) ist und die Transition erfüllt wird. In GRAFCET wird die UND-Verknüpfung mit * und die ODER-Verknüpfung mit + angegeben.

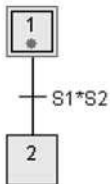


Bild 2.3 Beispiel 2



HINWEIS

Wenn Sie dieses Beispiel mit der Lernsoftware nachvollziehen wollen, können Sie den Plan von Beispiel 1 nutzen und im GRAFCET-Editor durch Doppelklick auf die Transition die neue Bedingung durch Eingabe von «S1*S2» einstellen. Das Bearbeiten und Überprüfen des Ablaufs dieses Beispiels entspricht der Beschreibung für das Beispiel 1 aus Abschnitt 7.1.

2.3 Zeitabhängige Transitionsbedingung

Soll der Übergang zwischen zwei Schritten nach einer bestimmten Zeit erfolgen, so wird hierfür die zeitabhängige Transitionsbedingung benutzt. Die zeitabhängige Transitionsbedingung hat die Form: Zeit/ Schrittvariable.

Beispiel 3:

15s/X2 bedeutet, dass der Schritt nach der Transitionsbedingung angenommen wird, wenn der vorherige Schritt (Nr. 2) 15 Sekunden aktiv war.

In dem Beispiel 3 wurde der GRAFCET-Plan (Bild 2.4) um einen dritten Schritt erweitert, der angenommen wird, wenn der zweite Schritt 15 Sekunden aktiv war.

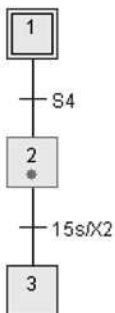


Bild 2.4 Beispiel 3



HINWEIS

Zum Bearbeiten dieses Beispiels mit der Lernsoftware orientieren Sie sich bitte an der Beschreibung für das Beispiel 1 aus Abschnitt 7.1.

2.4 Beispiele für Transitionsbedingungen

Im Folgenden werden Beispiele mit unterschiedlichen Transitionsbedingungen dargestellt. Bei der Bearbeitung der folgenden Beispiele mit der Lernsoftware GRAFCET-Kursus orientieren Sie sich bitte an der ausführlichen Beschreibung in Abschnitt 7.1 für das Beispiel 1.

Beispiel 4:

Die Transitionsbedingung ist erfüllt, wenn für die binären Signale S1, S2 und S3 gilt: S1 und (nicht S2 oder S3).

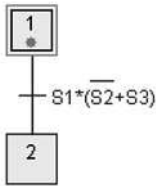


Bild 2.5 Beispiel 4



HINWEIS

Die Transition aus Beispiel 4 geben Sie in der Lernsoftware über die Tastatur im Dialog für die Transition folgendermaßen ein: $S1*(\overline{S2+S3})$.

Beispiel 5:

Die Transitionsbedingung ist erfüllt, wenn für das binäre Signal S4 gilt:

Das Signal S4 wechselt vom Zustand 0 auf den Zustand 1 (Flanke für S4 geht von 0 auf 1).

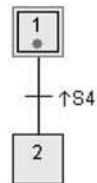


Bild 2.6 Beispiel 5



HINWEIS

Die Transition aus Beispiel 5 geben Sie in der Lernsoftware über die Tastatur im Dialog für die Transition folgendermaßen ein: $\wedge S4$.

Beispiel 6:

Die Transitionsbedingung ist erfüllt, wenn für die binären Signale S3 und S4 gilt:
 S3 oder (Flanke für S4 geht von 0 auf 1).

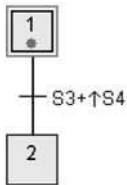


Bild 2.7 Beispiel 6

Beispiel 7:

Die Transitionsbedingung ist erfüllt, wenn für die binären Signale S3 und S4 gilt:
 Fallende Flanke von S3 (Flanke für S3 geht von 1 auf 0) und S4.

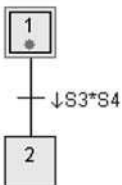


Bild 2.8 Beispiel 7

**HINWEIS**

Die Transition aus Beispiel 7 geben Sie in der Lernsoftware über die Tastatur im Dialog für die Transition folgendermaßen ein: $\downarrow S3 * S4$.

Beispiel 8:

Die Transitionsbedingung ist erfüllt, wenn für das binäre Signale S1 gilt:

Wenn S1 auf 1 geht, ist 3 s später die Transitionsbedingung erfüllt.

Wenn S1 wieder auf 0 geht, bleibt die Bedingung noch 7 s aktiv.

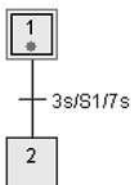


Bild 2.9 Beispiel 8



ACHTUNG

Theoretisches Beispiel: Kann in der Praxis zu Problemen führen, weil u. a. der Zustand des Schrittes 1 nicht berücksichtigt wird.

Beispiel 9:

Die Transitionsbedingung ist erfüllt, wenn gilt:

Schritt 1 war 4 s aktiv (X1 ist die Schrittvariable, die den Zustand von Schritt 1 angibt).

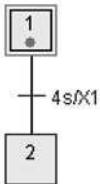


Bild 2.10 Beispiel 9

Beispiel 10:

Die Transitionsbedingung ist erfüllt, wenn für das analoge Signal R1 gilt:

R1 gleich 40

(Die analoge Bedingung bei der Transition muss in eckigen Klammern [] stehen.)

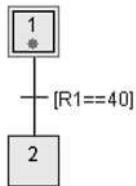


Bild 2.11 Beispiel 10

Beispiel 11:

Die Transitionsbedingung ist erfüllt, wenn für das binäre Signal S1 und für das analoge Signal R1 gilt: (R1 größer 45) und S1

(Die analoge Bedingung bei der Transition muss in eckigen Klammern [] stehen.)

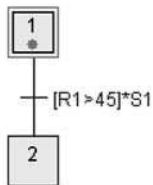


Bild 2.12 Beispiel 11

Beispiel 12:

Die Transitionsbedingung ist erfüllt, wenn für das binäre Signal S2 und für das analoge Signal R1 gilt:

S2 oder (R1 kleiner gleich 50)

(Die analoge Bedingung bei der Transition muss in eckigen Klammern [] stehen.)

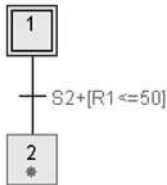


Bild 2.13 Beispiel 12

Beispiel 13:

Die Transitionsbedingung ist erfüllt, wenn für die analogen Signale R1 und R2 gilt:

R1 größer R2

(Die analoge Bedingung bei der Transition muss in eckigen Klammern [] stehen.)

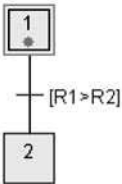


Bild 2.14 Beispiel 13

2.5 Aktionen

Jedem Schritt können eine oder mehrere Aktionen (Befehle) zugeordnet werden. Die Aktionen werden ausgeführt, wenn der Schritt aktiv ist.

Man unterscheidet zwei Arten von Aktionen:

- kontinuierlich wirkende Aktion,
- gespeichert wirkende Aktion.

Ihr Verhalten wird durch entsprechende Zusätze bestimmt.

Wie in Bild 2.15 dargestellt, kann die Anbindung mehrerer Aktionen an einen Schritt auf unterschiedliche Weise realisiert werden. Die oberen beiden Darstellungen werden im Normalfall benutzt.

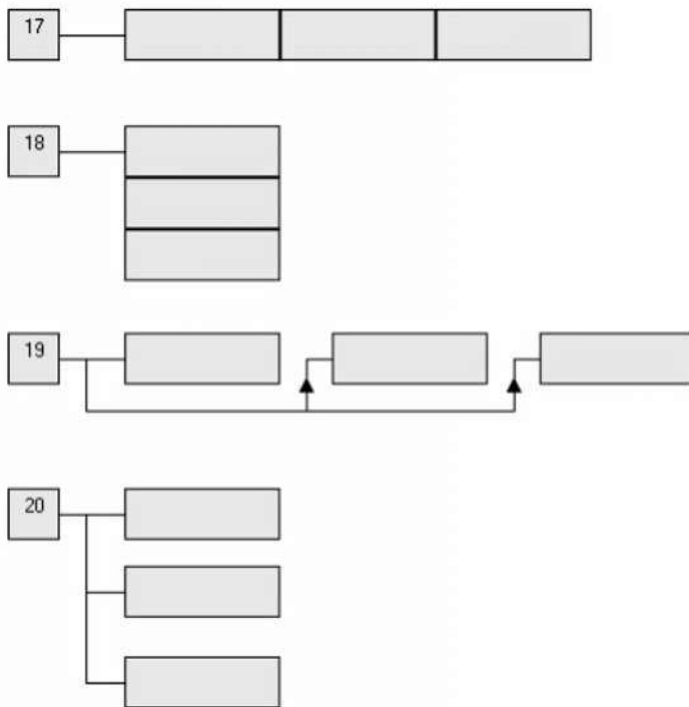


Bild 2.15 Schritte mit zugeordneten Aktionen

2.6 Kontinuierlich wirkende Aktion

Bei einer kontinuierlich wirkenden Aktion erhält das zugewiesene Signal den Wert 1 (True), wenn der zugehörige Schritt aktiv ist.

Ist der Schritt nicht mehr aktiv, erhält das Signal den Wert 0 (False).



ACHTUNG

Die kontinuierlich wirkende Aktion und die gespeichert wirkende Aktion dürfen nicht kombiniert für das gleiche Signal eingesetzt werden. Der Zustand des Signals wird nur von der kontinuierlich wirkenden Aktion bestimmt.

Bei der kontinuierlich wirkenden Aktion erhält das Signal nur dann den Wert 1, wenn irgendwo im GRAFCET ein Schritt aktiv ist, der mit einer kontinuierlich wirkenden Aktion für dieses Signal verknüpft ist. Wenn kein zugehöriger Schritt aktiv ist, erhält das Signal den Wert 0 (False). Dies gilt auch, wenn versucht wird, dieses Signal über eine gespeichert wirkende Aktion zu setzen.

Beispiel 14:

Die Lampe P1 soll durch den Schalter S1 eingeschaltet werden, 15 s leuchten und dann wieder ausgehen.

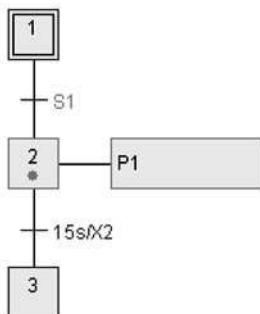


Bild 2.16 Beispiel 14



HINWEIS

Eine Beschreibung zum Bearbeiten dieses Beispiels mit der Lernsoftware finden Sie in Abschnitt 7.2.

Beispiel 15:

Erweitern Sie den GRAFCET-Plan so, dass mit der Lampe P1 auch die Lampen P2 und P3 entsprechend an- und wieder ausgeschaltet werden.

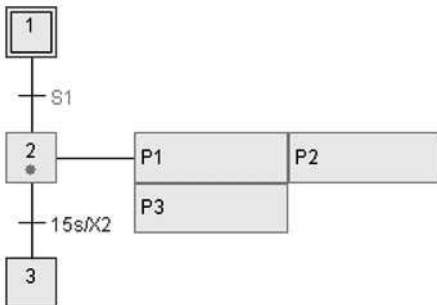


Bild 2.17 Beispiel 15

2.7 Kontinuierlich wirkende Aktion mit Zuweisungsbedingung

Bei einer kontinuierlich wirkenden Aktion mit Zuweisungsbedingung wird das zugehörige Signal nur dann auf 1 gesetzt, wenn der mit der Aktion verknüpfte Schritt aktiv und gleichzeitig die Zuweisungsbedingung erfüllt ist, also den Wert 1 (True) hat.

Beispiel 16:

Die Leuchtdiode P4 soll über den Schalter S1 und die Leuchtdiode P5 über die Schalter S2 oder S3 an- und ausgeschaltet werden.

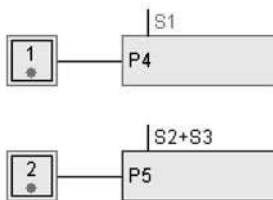


Bild 2.18 Beispiel 16

Da die kontinuierlichen Aktionen mit einem Anfangsschritt verbunden sind und der Anfangsschritt bei Ausführung des GRAFCET-Plans sofort aktiv ist, sind P4 und P5 nur noch abhängig von S1, S2, S3.

Beispiel 17:

Erweitern Sie den Plan aus Beispiel 14 so, dass die Lampe P1 nur angeht, wenn zusätzlich der Schalter P2 gedrückt wird.

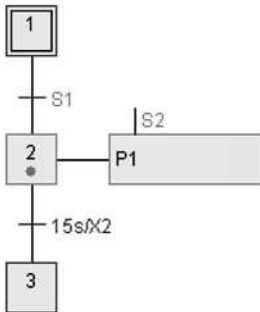


Bild 2.19 Beispiel 17

**HINWEIS**

Eine Beschreibung zum Einstellen der «kontinuierlich wirkenden Aktion mit Zuweisungsbedingung» in der Lernsoftware finden Sie in Abschnitt 7.3.

2.8 Kontinuierlich wirkende Aktion mit zeitabhängiger Zuweisungsbedingung

Bei der kontinuierlich wirkenden Aktion mit zeitabhängiger Zuweisungsbedingung wird die Zeit vor der Bedingung (Signal oder Schrittzustand) als Einschaltverzögerung und die Zeit nach der Bedingung als Ausschaltverzögerung genommen.

Beispiel 18:

Wenn Schritt 1 aktiv ist, soll 3 s nachdem der Schalter S1 betätigt wurde die Lampe P1 anfangen zu leuchten. Geht S1 wieder auf 0, soll P1 noch 5 s weiter leuchten und dann ausgehen.

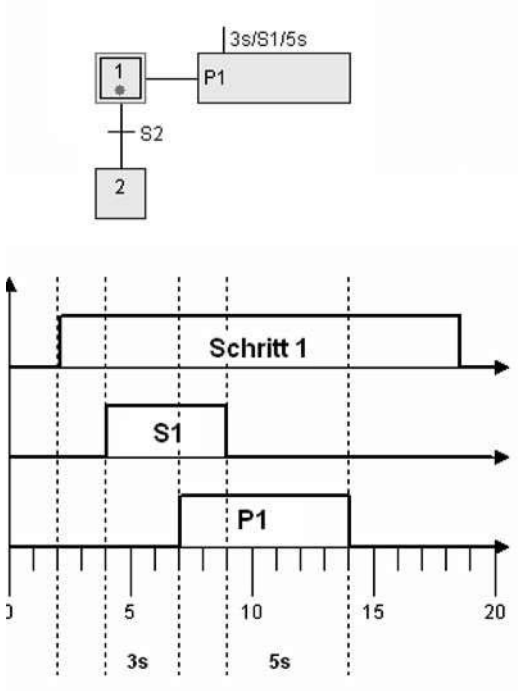


Bild 2.20 Beispiel 18

Im Beispiel 18 wird 3 s nachdem das Signal S1 auf 1 gegangen ist das Signal P1 auf 1 gesetzt (Einschaltverzögerung).

Nachdem S1 wieder ausgeschaltet wurde, wirkt die Ausschaltverzögerung und schaltet nach 5 s das Signal P1 auf 0. Schritt 1 muss natürlich während dieser Zeit aktiv sein.

Wird der Schritt 1 vorher durch Drücken von S2 verlassen, geht P1 sofort auf 0 bzw. wird gar nicht erst gesetzt.

Mit den kontinuierlich wirkenden Aktionen mit zeitabhängiger Zuweisungsbedingung sind auch zeitverzögernde Aktionen und zeitbegrenzte Aktionen möglich.

Beispiel 19:

Darstellung der zeitverzögernden kontinuierlich wirkenden Aktion

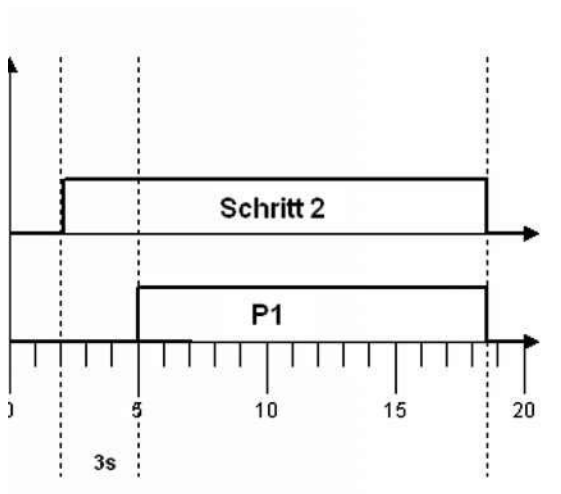
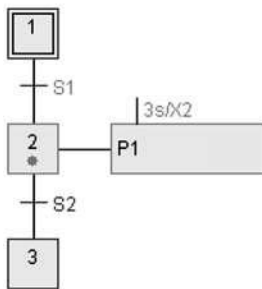


Bild 2.21 Beispiel 19

Die mit Schritt 2 verknüpfte kontinuierlich wirkende Aktion ist eine zeitverzögernde Aktion.

Die Lampe P1 wird 3 s nachdem der Schritt 2 aktiv wurde eingeschaltet. Sollte der Schritt 2 weniger als 3 s aktiv sein, geht die Lampe P1 gar nicht erst an.

Beispiel 20:

Darstellung der zeitbegrenzten kontinuierlich wirkenden Aktionen

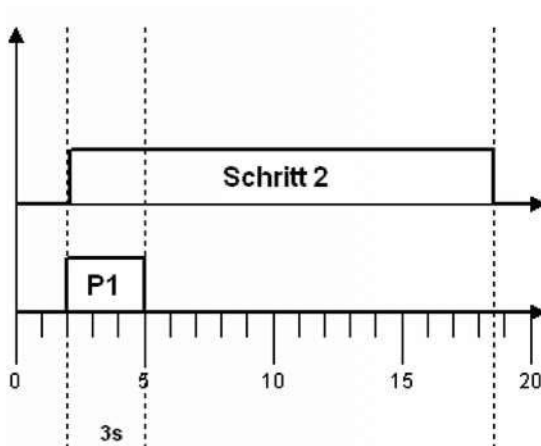
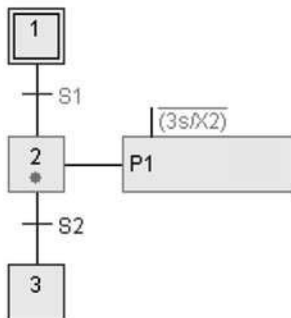


Bild 2.22 Beispiel 20

Die mit Schritt 2 verknüpfte Aktion ist eine zeitbegrenzte Aktion.

Wird Schritt 2 aktiv, so wird die Lampe P1 für 3 s eingeschaltet und dann ausgeschaltet. Sollte der Schritt 2 weniger als 3 s aktiv sein, wird die Lampe P1 sofort ausgeschaltet.



HINWEIS

In der Lernsoftware geben Sie die obige Zuweisungsbedingung für die zeitbegrenzte kontinuierlich wirkende Aktion folgendermaßen ein:
 $!(3s/X2)$.

Beispiel 21:

Die Leuchtdiode P8 soll 5 s nachdem der Schalter S1 betätigt wurde anfangen zu leuchten. Wenn S1 ausgeschaltet wird, soll P8 ausgehen.

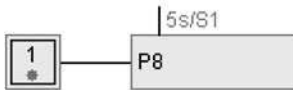


Bild 2.23 Beispiel 21

Da die kontinuierlich wirkende Aktion mit einem Anfangsschritt verbunden ist, ist das Schalten der Leuchtdiode P8 nur abhängig von S1.

Beispiel 22: Erstellen eines GRAFCET-Plans mit folgenden Eigenschaften:

3 s nach dem Start des GRAFCET-Plans soll die Leuchtdiode P8 anfangen zu leuchten. Wird der Schalter S1 gedrückt, soll P8 nach Ablauf von 5 s ausgehen.

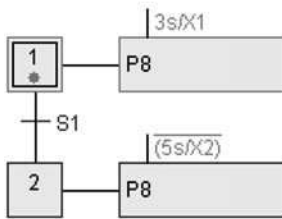


Bild 2.24 Beispiel 22

2.9 Gespeichert wirkende Aktion bei Aktivierung

Bei der gespeichert wirkenden Aktion bei Aktivierung wird das zugewiesene Signal auf den angegebenen Wert gesetzt, wenn der Schritt aktiv wird, d.h. wenn die Flanke für den Schritt von 0 auf 1 geht. Das Signal behält seinen Wert, auch wenn der zugehörige Schritt nicht mehr aktiv ist. Durch die gespeichert wirkende Aktion können binäre Signale auf Zustände (0 oder 1) und analoge Signale auf Werte gesetzt werden.



ACHTUNG

Die kontinuierlich wirkende Aktion und die gespeichert wirkende Aktion dürfen nicht kombiniert für das gleiche Signal eingesetzt werden. Der Zustand des Signals wird nur von der kontinuierlich wirkenden Aktion bestimmt.

Bei der kontinuierlich wirkenden Aktion erhält das Signal nur dann den Wert 1, wenn irgendwo im GRAFCET ein Schritt aktiv ist, der mit der kontinuierlich wirkenden Aktion für dieses Signal verknüpft ist. Wenn kein zugehöriger Schritt aktiv ist, erhält das Signal den Wert 0 (False). Dies gilt auch, wenn versucht wird, dieses Signal über eine gespeichert wirkende Aktion zu setzen.

Beispiel 23:

Erstellen eines GRAFCET-Plans mit fünf Schritten, bei dem im zweiten Schritt die Lampe P1 mit der gespeichert wirkenden Aktion bei Aktivierung angeschaltet wird und im vierten Schritt die Lampe mit der gespeichert wirkenden Aktion bei Aktivierung wieder ausgeschaltet wird. Die Schritte sollen jeweils 5 s aktiv sein.

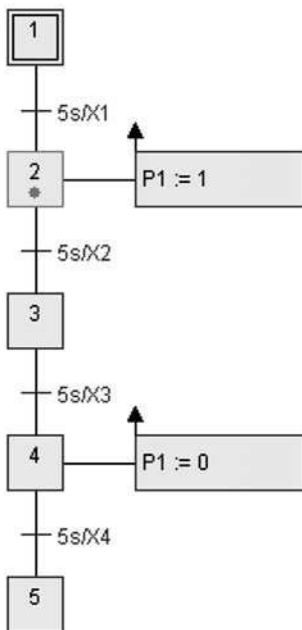


Bild 2.25 Beispiel 23