Versuch: Positionierung mit SPS

Versuchsvorbereitung
Von Klaus Habermann, Florian Knorn, Sebastian Taege (STK01)

1. Zwei mögliche Schaltungsvarianten für die Not-Endschalter sind in Bild 1 und Bild 2 dargestellt: im ersten Fall sind zwei Schalter in Reihe geschaltet, die im normalen Betriebszustand geschlossen sind. Im zweiten Fall sind 2 Schalter parallel geschaltet, die für ordnungsgemäßen Betrieb geöffnet sein müssen.
   Tritt ein „Notfall“ auf, so soll im ersten Fall ein bestehendes Signal unterbrochen (ein Schalter geöffnet werden), was der Steuerung den „Notfall“ signalisiert. Im zweiten Fall solle hingegen einer der Schalter geschlossen werden.
   Vorteil der ersten Variante ist jedoch, dass auch ein Ausfall des Sicherungssystems an sich bemerkt und von der Steuerung entsprechend reagiert werden kann (vgl. „Live Zero“-Konzept). Dies ist bei der zweiten Realisierung nicht möglich. Deshalb ist die erste Schaltungsvariante die sicherere Lösung.

Bild 1

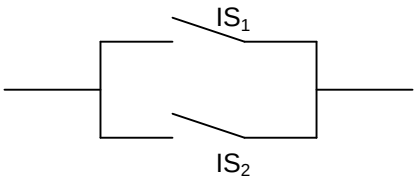

Bild 2

2. Klassifizierung der Steuerung nach
   a. Technische Funktionsrealisierung: Arbeit mit speziellem Wortprozessor, der auf STEP 5 zugeschnitten ist.
   b. Zeitliche Abarbeitung: synchrone Abarbeitung der Steuerungsfunktionen.
   c. Arbeitsweise: Ablaufsteuerung, d.h. prozessabhängiges, bedingtes Weiterschalten
   d. Weiterschaltbedingung: Transitionen werden von den induktiven Sensoren bzw. nach Erledigung bestimmter Aufgaben / Aufträge aktiviert.

3. Eine Skizze der verschiedenen Positionen ist im Bild 3 dargestellt.
   a. Referenzpunktfahrt: Rückwärtsfahrt (d.h. in Richtung Schrittmotor SM) bis Endlagenschalter E_- erreicht ist. Anschließende Vorwärtsfahrt bis Endlage E_+ und Rückkehr zum Referenzpunkt RP.
   b. Nachdem der Wagen den Referenzpunkt erreicht hat soll er, laut Aufgabenstellung (Punkt 4. in [1]), vom Silo C = Referenzpunkt zum Silo A fahren, dort 5s warten. Anschließend soll er zum Silo B fahren, ebenfalls 5s warten und schließlich zum Punkt C zurückkehren und nochmals 5s warten. Die Strecke C–A soll dabei mit erhöhter Geschwindigkeit zurückgelegt werden, da der Wagen hier noch nicht beladen ist.

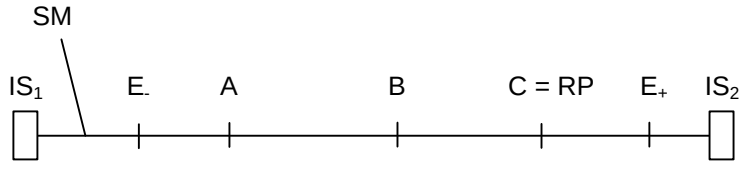

Bild 3

4. Die Simatic S5-100U ist ein modulares Automatisierungsgerät. Prinzipieller besteht eine minimale, modulare SPS aus folgenden Bausteinen:
    a. Netzteil 24 V, Funktionskleinspannung
    b. U mit Mikroprozessor, RAM (Arbeitsspeicher für Programm und Variable), ROM (Festwertspeicher für Betriebssystem, heute Flash–ROM), Bussystem für SPS, Schnittstelle für Programmierung und Vernetzung
    c. Eingangsbaugruppe (digital 0/24V, analog 0…10V, analog 2…10V (draht-bruchsicher), analog 0…20mA, analog 4…20mA (drahtbruchsicher)
    d. Ausgangsbaugruppe (digital 0/24V auch 0/230V, analog 0…10V)
    e. Programm ist durch Batteriepufferung nicht flüchtig

    Die S5-100U besteht speziell aus:
    a. Einer Zentralbaugruppe
    b. den Busmodulen und den Peripheriebaugruppen, die auf die Busmodule gesteckt werden
    c. folgende P.-Baugruppen stehen zur Auswahl:
        i. Digitalein- und Ausgabebaugruppen
        ii. Analogein- und Ausgabebaugruppen
        iii. Ex-Baugruppen zum Anschluss von Sensoren und Aktoren aus explosionsgefährdeten Bereichen,
        iv. Spezialbaugruppen für ext. Zeitfunktionen,
        v. intelligente Baugruppen für schnell verarbeitende Prozesse usw.

    Programmiert wird in AWL (Anweisungsliste), KOP (Kontaktplan) und FUP (Funktionsplan). KOP und FUP sind grafische Darstellungen von bestimmten AWL-Sequenzen. Es lassen sich also alle FUP- und KOP-Symbole in AWL darstellen, aber nicht jede AWL in FUP oder KOP. Auf dem PG (Programmiergerät) wird nur AWL gespeichert. Die FUP- oder KOP-Darstellung wird bei der Bearbeitung jedesmal neu aus der AWL erzeugt.

    Im AG (der SPS) wir daber nur ein Maschinencode namens MC7 gespeichert, der aus der AWL generiert wird. Auf dem PG liegen also die Quelldateien (AWL-Programm, Symboltabellen, Kommentare); in die SPS wird aber nur der lauffähige Maschinencode übertragen.
    Das heißt, wenn man aus der SPS ein Programm lädt, hat man keine Symboltabelle und keine Kommentare zur Verfügung, weil diese nur in der ursprünglichen Quelle gespeichert sind.

    Eine wichtige Rolle bei der Programmierung einer SPS S5-100U spielen die Ausgänge, Eingänge und Merker:
    a. Ausgänge können gesetzt und abgefragt werden
    b. Eingänge können nur gesetzt werden
    c. Merker sind Ausgänge ohne (echten) Ausgang; sie können gesetzt und abgefragt werden („Merker sind Hilfsschütze der SPS“); sie sind Zwischenspeicher für Verknüpfungs- und andere Zwischenergebnisse; ansprechbar über eine absolute Adresse, deren erstes Zeichen ein M ist

    In der SPS sind Organisationsbausteine die Schnittstelle zwischen Betriebssystem und Programm. Ein Funktionsbaustein verhält sich wie eine Funktion mit „Gedächtnis“. Dieses Gedächtnis liegt in einem Datenbaustein, der dem Aufruf des Funktionsbausteins zugeordnet ist. Man bezeichnet ihn als Instanzdatenbaustein. Es können bei verschiedenen Aufrufen des gleichen Funktionsbausteins unterschiedliche (Instanz-)Datenbausteine angegeben werden, die zwar die gleiche Datenstruktur besitzen müssen, aber unterschiedliche Werte enthalten können. Auf diese Weise ist es z.B. möglich, verschiedenen Motoren mit einem Funktionsbaustein zu steuern. Für jeden Motor wird dann ein Datenbaustein angelegt, in dem die entsprechenden Parameter gespeichert werden.

    Die Abarbeitung des Programms in einer speicherprogrammierbaren Steuerung erfolgt seriell, die Steuerbefehle des Programms werden nacheinander abgearbeitet. Das ist langsamer als die parallele Verarbeitung mit Schützen oder Elektronikkomponenten (VPS). Bei einer großen Anzahl von Eingangsvariablen und vielen Programmschritten ist mit einer unter Umständen erheblichen Reaktionszeit der SPS in Bezug auf eine Änderung der Eingangsvariablen zu rechnen. Im Gegensatz zur VPS muss bei Änderungen der Steuerungsaufgabe lediglich das Programm umgeschrieben werden, also ohne mechanischen Eingriff. Eine Programmänderung ist damit einfach durchzuführen.

5. Mit den in 6. bestimmten wurde der FB 57 ausgefüllt. Das Schema war immer das gleiche: zunächst soll der gewünschte Hexadezimalwert (Hexkonstante KH) per `L KH xxxx`$_H$ in den AKKU1 geladen werden, anschließend soll der Wert dann in ein bestimmtes Ausgangswort (AW) transferiert werden: `T AW xx`.

6. Bestimmen der Konfigurierungs- und Positionierungsdaten
   a. Tabelle 5 wurde mit folgenden Überlegungen / Berechnungen ausgefüllt:
      i. Nach [1], S. 7 beträgt der Multiplikator für die SS-Frequenz gerade 2
      ii. KB1 und KB0 müssen auf 0 gesetzt sein, damit das Konfigurierungstelegramm übernommen werden kann
      iii. Umstellen der Gleichung für die Frequenzzunahme (S. 3 in [1]) liefert $ZI = \frac{BW \cdot R}{4 \cdot 0{,}032 \cdot a}$. Mit BW = 40 (resp. 20 Hz), R = 1,0 und a = 5 Hz/ms folgen die Ergebnisse.
      iv. Um eine Verfahrgeschwindigkeit von 200 (resp. 100) mm/s zu erreichen, müssen dem Motor f_A = 200 / 0.405 ˜ 493,83 (resp. 246,91) Impulse pro Sekunde zugeführt werden. Umstellen der Gleichung für die Ausgangsfrequenz (S. 5 in [1]) liefert $G = \frac{f_A}{BW \cdot R}$. Mit BW = 40 (resp. 20) Hz und R = 1,0 folgen die Ergebnisse.
      v. Die Strecke C–A entspricht einer Rückwärtsfahrt, die Strecken A–B und B–C hingegen einer Vorwärtsfahrt. Deshalb müssen die Betriebsarten-Bits gemäß Tabelle 4 aus S. 5 in [1] auf (1,0) bzw. (0,1) gesetzt werden.
      vi. Braucht der Schrittmotor 200 Impulse pro Umdrehung, und entspricht eine Umdrehung einer Schlittenwegstrecke von 81 mm, so benötigt man z.B. für 600,21 mm gerade 1482 Impulse (einfacher Dreisatz). Die anderen Werte werden analog berechnet.
   b. Die Konfigurierungs- und Positionierungstelegramme (Tabellen 4.3 und 4.6) wurden mit den in a. bestimmten Daten ausgefüllt und die Binärwerte in Hexadezimalwerte umgerechnet.