 h_da HOCHSCHULE DARMSTADT UNIVERSITY OF APPLIED SCIENCES	Labor Automatisierungssysteme (ai/m) Versuch 2 Bausteintechnik / Förderband	Fachbereich EIT Verfasser: Prof. Dr.-Ing. S. Simons Dipl.-Ing. J. Meyer
---	---	--

Versuch 2 Bausteintechnik / Förderband

Quellen dieser Ausführungen sind Unterlagen der SIEMENS AG.


Lösen Sie die mit "Vorbereitung" gekennzeichneten Versuchsteile vor dem Versuchstermin in Papierform. Dokumentieren Sie die Lösung handschriftlich und individuell (jedes Gruppenmitglied hat eine Lösung!). Wenn möglich, testen Sie zusätzlich Ihre Lösung mit der studierenden Version des TIA Portals! Sollten Sie die Aufgaben mit dem TIA Portal gelöst und ausgedruckt haben, muss jedes Netzwerk mit einem handschriftlichen und individuellen Kommentar versehen sein!

Drucken Sie am Anfang des Labortermins für jedes Gruppenmitglied ein Deckblatt aus. Auf diesem Deckblatt wird ihre Leistung während des Laborversuches dokumentiert. Dieses Deckblatt geben Sie - mit den restlich geforderten Dokumenten - am Ende des Laborversuches ab, und ihre Leistung wird anhand des Deckblattes bewertet.

Sie sind verantwortlich, dass der Betreuer ihre fertiggestellte und geprüfte Aufgabe (Übung) auf dem Deckblatt abzeichnet.

Nur wenn es in der Aufgabenstellung (Übung) gefordert ist, drucken Sie den Programmcode aus. Sollten Fragen, Testpläne oder Kommentare zu beantworten sein, wo kein Ausdruck vonnöten ist, verwenden sie ein leeres weißes Blatt.

Versuch 2 besteht aus zwei Teilen. Es sind beide Teile vorzubereiten. Für die Durchführung am Versuchstermin ist pro Versuchsteil ein Block eingeplant.

 h_da HOCHSCHULE DARMSTADT UNIVERSITY OF APPLIED SCIENCES	Labor Automatisierungssysteme (ai/m) Versuch 2 Bausteintechnik / Förderband	Fachbereich EIT Verfasser: Prof. Dr.-Ing. S. Simons Dipl.-Ing. J. Meyer
---	--	---

1 Lernziele

Die Studierenden sollen modulare SIMATIC S7-Anwenderprogramme schreiben können, um externe Prozesse zu steuern.

Sie sollen außerdem das eingesetzte Modell kennengelernt haben, so dass Sie im weiteren Verlauf des Labors weitere Versuche damit durchführen können.

2 Einführung

Im ersten Teil des Versuchs sollen die Studierenden modulare SIMATIC S7-Programme für einfache Aufgaben erstellen, bei denen Sie die unterschiedlichen Programmbausteine kennenlernen.

Im zweiten Teil des Versuchs soll das Modell eines Förderbandes mit zwei Abschiebern kennengelernt werden. Hierzu werden kleine Teilprogramme geschrieben, um die genaue Funktionsweise des Modells zu erfahren. Bei dem Förderband handelt es sich um ein Modell der Firma Köster Systemtechnik.

Teil 1 Bausteintechnik

Vorbereitung

Lesen Sie die Kapitel 5 und 14 des Begleithefts gründlich durch.

3 Funktionsbausteine FBxx

Übung 1

Als Übung wird ein D-FlipFlop in einem Funktionsbaustein (FB) aufgebaut. Abbildung 3.1 zeigt den Logikplan, der realisiert werden soll.

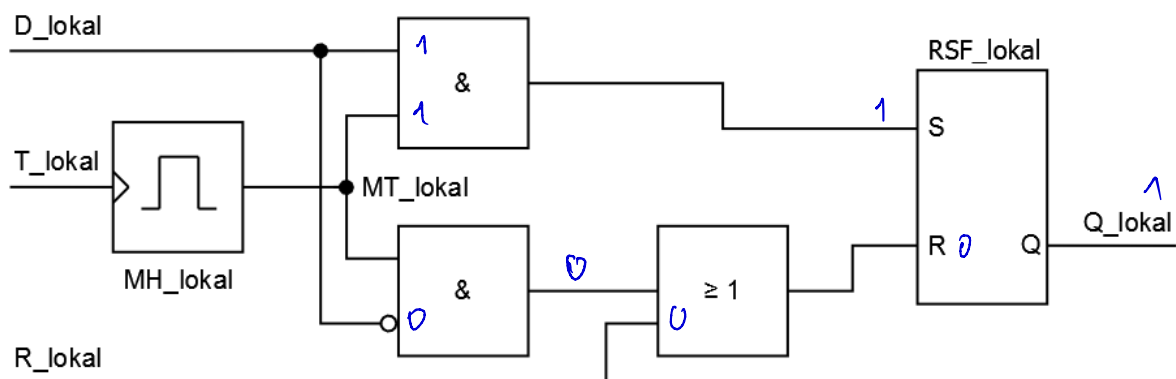


Abbildung 3.1: Logikplan D-FF, der im FB realisiert werden soll

D_lokal: Eingang D-FF
 T_lokal: Takt (positiv flankengetriggert)
 R_lokal: Reset-Eingang
 Q_lokal: Ausgang

MH_lokal: statische Zwischengröße
 MT_lokal: temporäre Zwischengröße
 RSF_lokal: FF-Zustand

Abbildung 3.2 zeigt den Deklarationsteil des Funktionsbausteins, Abbildung 3.3 den Anweisungsteil des Funktionsbausteins "D-FF".

Funktionsbaustein mit D-FF Logik:

D_FF					
	Name	Datentyp	Defaultwert	Remanenz	Kommentar
1	Input				
2	D_lokal	Bool	false	Nicht remanent	D-FF Eingang
3	T_lokal	Bool	false	Nicht remanent	Takteingang
4	R_lokal	Bool	false	Nicht remanent	Reset-Eingang
5	Output				
6	Q_lokal	Bool	false	Nicht remanent	Ausgang
7	InOut				
8	<Hinzufügen>				
9	Static				
10	RSF_lokal	Bool	false	Nicht remanent	RS-FF-Zustand
11	MH_lokal	Bool	false	Nicht remanent	Wischimpuls (Hilfsgrösse Impuls)
12	Temp				
13	MT_lokal	Bool			Wischimpuls (Hilfsgrösse)
14	Constant				

Abbildung 3.2: Zu erstellender Deklarationsteil des Funktionsbausteins für das D-FF

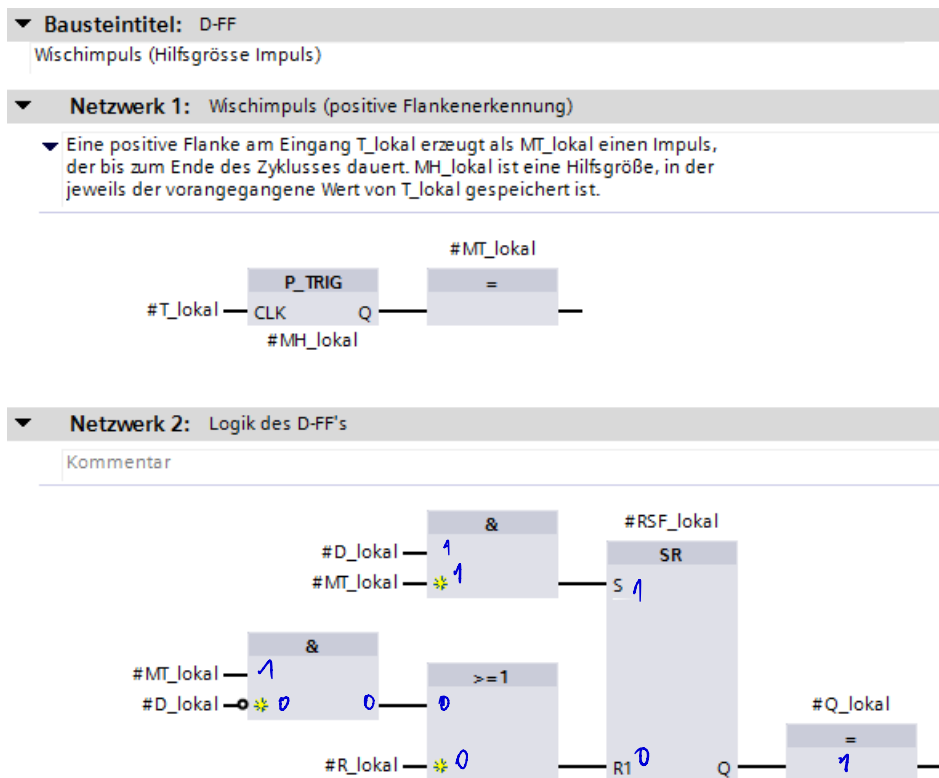


Abbildung 3.3: Zu erstellender Anweisungsteil des Funktionsbausteins für das D-FF

Die lokale Größe `MH_lokal` des positiven Wischimpulses ist der gespeicherte Wert des Eingangssignals `T_lokal` aus dem vorhergehenden SPS-Zyklus und muss deshalb statisch deklariert werden. Die CPU kann somit das aktuelle Eingangssignal `T_lokal` mit dem in `MH_lokal` gespeicherten vorangegangenen Eingangssignal vergleichen und feststellen, ob eine positive Flanke von `T_lokal` aufgetreten ist. In diesem Fall wird der Ausgang des

positiven Wischimpulses auf TRUE ansonsten auf FALSE gesetzt. Anschließend wird in jedem Fall der aktuelle Wert von `T_lokal` für den Vergleich im nächsten Zyklus in `MH_lokal` gespeichert. Die Größe `MT_lokal` mit dem Wischimpuls selber, wird nur temporär bis zum Ende des aktuellen Zyklus benötigt und zu Beginn des nächsten Zyklus jeweils neu gebildet. Auch der FF-Zustand `RSF_lokal` muss gespeichert und demnach als statisch deklariert werden.

Zusätzlich müssen die Ein- und Ausgangsgrößen deklariert werden.

In unserem Test wollen wir den Funktionsbaustein "D-FF" im OB1 in der Programmiersprache AWL aufrufen. Hierfür ziehen Sie den Funktionsbaustein D-FF per Drag-and-drop in ein AWL-Netzwerk des OB1 (Abbildung 3.4). Jedem Aufruf eines Funktionsbausteins muss in OB1 ein Instanz-Datenbaustein (DB) zugeordnet werden. Dies geschieht automatisch, wenn Sie den FB in den OB1 ziehen (Bitte keinen DB manuell anlegen!). In diesem Datenbaustein werden die Werte für die statischen Variablen (Parameter) und die Ein- und Ausgangsgrößen für den aktuellen Aufruf gespeichert. Der Datenbaustein stellt damit das Gedächtnis des Funktionsbausteins dar. Sollte der Funktionsbaustein noch ein weiteres Mal an anderer Stelle im Organisationsbaustein aufgerufen werden, müsste ihm dort ein anderer Instanz-Datenbaustein zugeordnet werden. In Funktionsbausteinen werden weitere Funktionsbausteine statt mit Instanz-Datenbausteinen mit Multiinstanz-Datenbausteinen aufgerufen. Dies wird in der Vorlesung näher erläutert.

OB1:

▼ Bausteintitel: OB1 - D-FF			
Positiv einflankengetriggertes D-FF			
▼ Netzwerk 1: Aufruf des Funktionsbausteins			
1	CALL "D_FF", "D_FF_DB"	§FB1, §DB1	
2	D_lokal := "D_global"	§E136.0	Daten-Eingang
3	T_lokal := "T_global"	§E136.1	Takt-Eingang
4	R_lokal := "R_global"	§E136.7	Reset-Eingang
5	Q_lokal := "Q_global"	§A137.0	Ausgang
6			

Abbildung 3.4: Zu erstellender Aufruf des Funktionsbausteins "D-FF" im OB1

Wird nach Einfügen des FBs in einem OBxx oder einem anderen FBxx die Schnittstellen-Deklaration geändert, stimmt die Struktur im bereits vorhandenen Datenbaustein nicht mehr mit der Struktur der Schnittstelle im FB überein. Sie müssen deshalb den Aufruf im schon vorhandenen Organisationsbaustein aktualisieren. Wenn Sie den OB1 nach einer Änderung der FB-Schnittstelle öffnen, wird der entsprechende Instanzdatenbaustein oberhalb des Funktionsbausteins rot markiert. Selektieren Sie nacheinander alle roten Aufrufe im OB1 und

wählen Sie im Kontextmenü (rechte Maustaste) den Menüpunkt » Aktualisieren«. Stimmen Sie anschließend ggf. einer Neugenerierung des Instanz-Datenbausteins zu.

Vorbereitung

Machen Sie sich in der Vorbereitung mit der Funktion des D-Flip-Flops vertraut und entwerfen Sie einen Testplan.

Test im Labor

Erstellen Sie wie, oben beschrieben, die Bausteine. Laden Sie alle Bausteine in die CPU, und testen Sie nach Testplan die Funktion des D-FlipFlops mit dem Schulungsgerät. Zeigen Sie diese funktionsfähige und vollständig bearbeitete Teilaufgabe einem Betreuer. Dieser prüft die Aufgabe nochmal mit ihnen, und weist sie gegebenenfalls auf Fehler hin die Sie dann noch verbessern. Ist die Teilaufgabe vollständig bearbeitet, zeichnet der Betreuer die Teilaufgabe auf ihrem Deckblatt ab.

4 Spezielle Organisationsbausteine

4.1 Cyclic interrupt OB30 - OB38 (Organisationsbaustein 30 bis 38 - Weckalarm)

Übung 2

zeit Zyklus

zeit Dauer.

Alle 2 sec soll der Zustand des Ausgangs A 137.0 seinen Wert ändern. Der Ausgang soll also ein oszillierendes Signal mit der Periodendauer 4 sec zeigen. Mit dem Eingang E 136.3 soll die Ausgabe des periodischen Signals freigegeben (E 136.3 = 1) oder gesperrt werden. Wenn der Ausgang gesperrt ist, soll A 137.0 den Wert 0 haben. Verwenden Sie bei dieser Übung auf keinen Fall Zeitbausteine!

Das Programm besteht aus zwei Organisationsbausteinen. Erstens dem OB1,

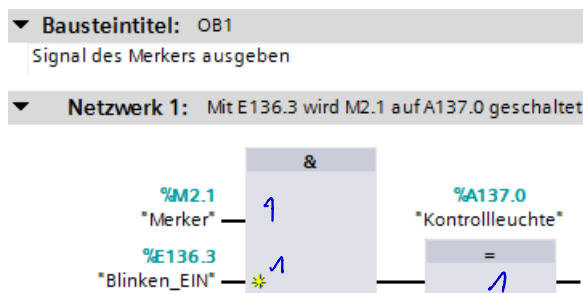


Abbildung 4.1: Zu erstellender Programmcode im OB1

und zweitens dem Weckalarm OB.

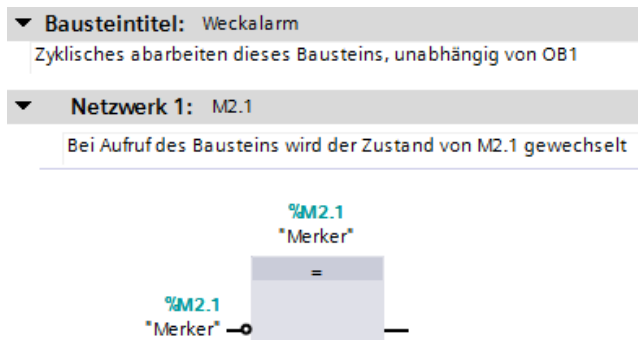


Abbildung 4.2: Zu erstellender Programmcode des Weckalarm OB

Vorbereitung

Beschreiben Sie kurz die Besonderheit der Weckalarm-Bausteine und wie und wo sie konfiguriert werden. Geben Sie einen möglichen Einsatzzweck eines solchen Bausteins an.

Test im Labor

Erstellen Sie den Baustein OB1 mit der Logik aus Abbildung 4.1. Danach erstellen Sie ein Weckalarm-Baustein nach Abbildung 4.2 mit dem entsprechenden Zeittakt aus der Übungsbeschreibung oben. Laden Sie die Bausteine in die CPU und testen Sie die Reaktion des Programms auf das Umschalten des Eingangs E 136.3.

Zeigen Sie diese funktionsfähige und vollständig bearbeitete Teilaufgabe einem Betreuer. Dieser prüft die Aufgabe nochmal mit ihnen, und weist sie gegebenenfalls auf Fehler hin die Sie dann noch verbessern. Ist die Teilaufgabe vollständig bearbeitet, zeichnet der Betreuer die Teilaufgabe auf ihrem Deckblatt ab.

Nach Beenden der Übung löschen Sie den Weckalarm Baustein.

4.2 Startup OB100 (Organisationsbaustein 100 - Warmstart)

Übung 3

Nach dem Anlaufen des Automatisierungssystems sollen mit dem OB100 alle 8 Ausgänge A 136.0 bis A 136.7 auf 1 gesetzt werden. Danach soll die Logik im OB1 bearbeitet werden.

Programm für den OB100:

Bausteintitel: OB100				
Definiertes Setzen der Ausgänge nach Neustart				
Netzwerk 1: Laden von FF in Byte 136				
1	L	16#FF	//Laden von 00FF (HEX-Code)	16#FF
2	T	"AusgangsByte_136"	//Transfer nach Ausgangsbyte 136	%AB136

Abbildung 4.3: Zu erstellenden Programmcode des OB100

Programm für den OB1:

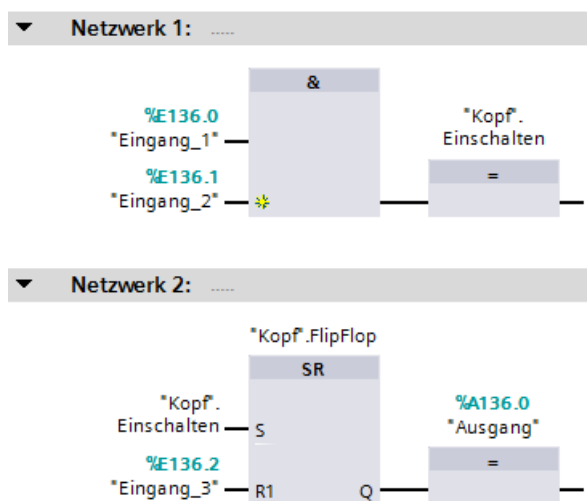


Abbildung 4.4: Zu erstellenden Programmcode des OB1

Einträge im Datenbaustein:

In dieser Übung sollen einige Variablen nicht in Merkern, sondern in einem Datenbaustein gespeichert werden (vergleiche Abbildung Abbildung 4.4). Hierzu legen Sie einen Datenbaustein "Kopf" an und erstellen dort zwei Variablen (Abbildung 4.5). Achten Sie darauf, dass Sie bei der Variable, in der der FlipFlop-Zustand gespeichert wird, den Haken bei Remanenz setzen.

Kopf								
	Name	Datentyp	Startwert	Remanenz	Erreic...	Sicht...	Einst...	Kommentar
1	Static			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
2	Einschalten	Bool	false	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	E136.0 6 & 136.1 = TRUE
3	FlipFlop	Bool	false	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	merkt sich den Zustand des FlipFlop's remanent
4	<Hinzufügen>			<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

Abbildung 4.5: Zu erstellende Einträge im Datenbaustein "Kopf"

Vorbereitung

- Beschreiben Sie kurz, was die Besonderheit des OB100 ist.
- Warum muss die Variable "FlipFlop" des Datenbausteins (Abbildung 4.5) remanent sein und die Variable "Einschalten" nicht?
- Begründen Sie kurz warum A136.0 nur kurzzeitig nach dem Einschalten der Betriebsart RUN den Wert 1 hat.
- Was müssen Sie tun, dass nach Anlauf der SPS der Ausgang A136.0 dauerhaft den Wert 1 behält?

Test im Labor

Testen Sie die Funktion der beiden Programme. Schalten Sie den Schalter E136.2 auf 1 und veranlassen Sie dann einen Warmstart der SPS. Hierfür stellen sie den Betriebsartenschalter der SPS auf STOP, warten 1 Sec und stellen ihn dann wieder auf RUN.

Zeigen Sie diese funktionsfähige und vollständig bearbeitete Teilaufgabe einem Betreuer. Dieser prüft die Aufgabe nochmal mit ihnen, und weist sie gegebenenfalls auf Fehler hin die Sie dann noch verbessern. Ist die Teilaufgabe vollständig bearbeitet, zeichnet der Betreuer die Teilaufgabe auf ihrem Deckblatt ab.

Nach Beenden der Übung löschen Sie den Baustein OB100.

Teil 2 Kennenlernen des Förderbandmodells

5 Förderbandmodell

Die Abbildung 5.1 zeigt das Förderbandmodell. Die einzelnen Komponenten des Modells sind im Dokument »Beschreibung des Förderbandmodells« beschrieben. Lesen Sie das Dokument »Beschreibung des Förderbandmodells« gründlich durch!

Das Modell ist mit zwei 37-poligen SUB D Kabeln mit dem Schulungsgerät verbunden. Die Pneumatik-Einheit ist mit der Druckluftversorgung des Labors verbunden. Sollte bei der elektrischen oder pneumatischen Verbindung etwas nicht in Ordnung sein, wenden Sie sich bitte an einen Betreuer.

Sollten z.B. die Pneumatikzylinder nicht ausfahren, überprüfen Sie jedoch zunächst erst einmal Ihr SPS-Programm. Oft liegt der Fehler an der Logik, d.h. dem SPS-Programm, aber der Entwickler schiebt es zunächst einfach mal auf das „System“, die „Hardware“ oder die „Pneumatikversorgung“. Diesen Fehler sollten Sie nicht machen.

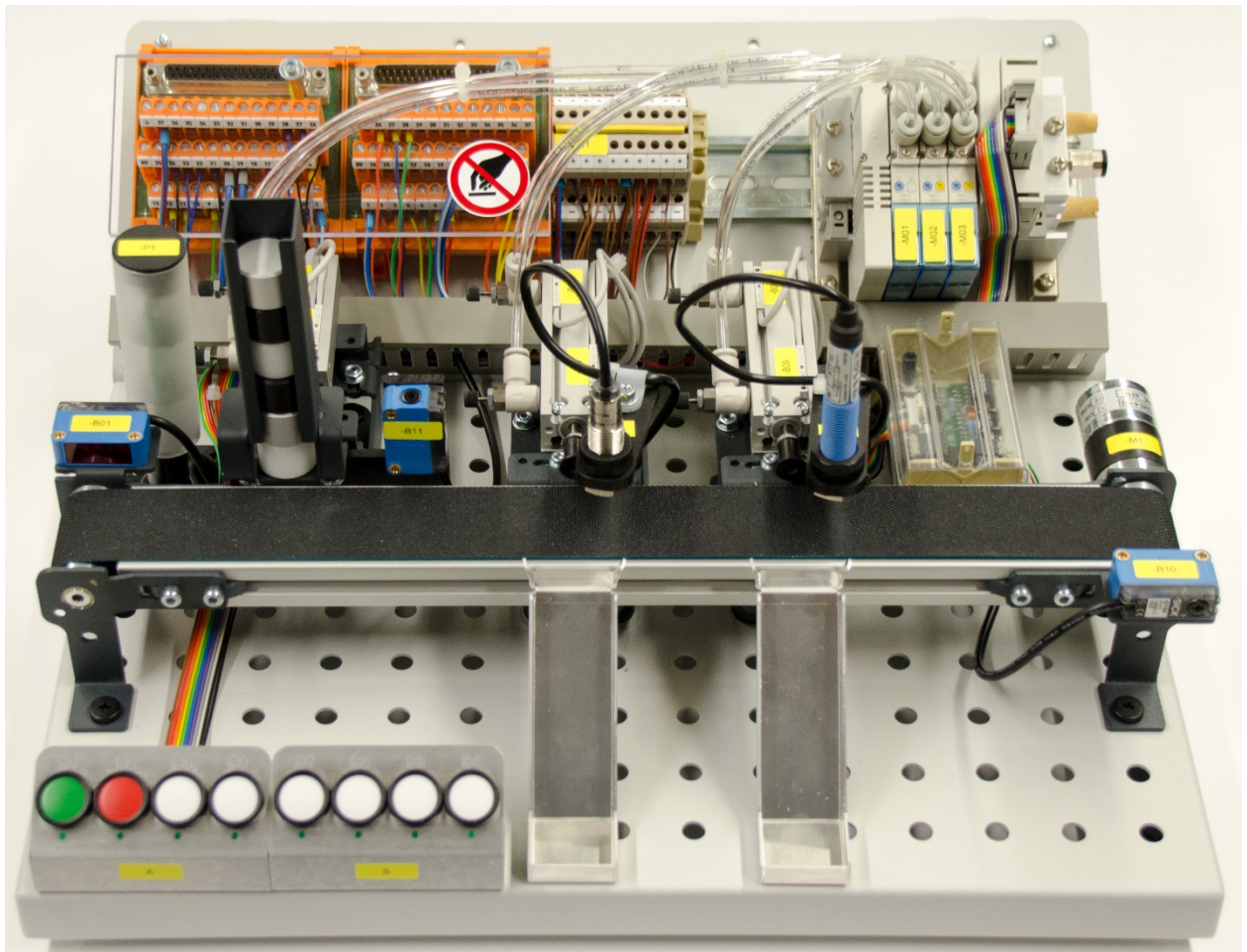


Abbildung 5.1: Förderbandmodell der Firma Köster Systemtechnik

Wichtig!

Sobald das Modell und das Schulungsgerät mit den beiden 37 pol. SUB-D Kabeln verbunden sind, ist darauf zu achten, dass die Eingangsschalter des Schulungsgerätes auf „Null“ (Mittelstellung, LED leuchtet nicht) stehen. Betätigen Sie anschließend auf keinen Fall die Eingangsschalter! Ansonsten käme es zu einem Kurzschluss, was zu einem Defekt des Modells führen würde.

6 Erstellen eine OB100 um das Modell in Ruhestellung zu versetzen

Wenn ein neues Programm in die SPS geladen wird, soll das Modell sich automatisch in seine Grundstellung begeben. Dies bedeute das die Ausgänge beim erstmaligen starten einen definierten Zustand haben. Dies realisiert man mit dem OB100. Bei dem Modell heißt das, dass die beiden Motoren aus sind und die Zylinder eingefahren werden. Die Hupe soll nicht ertönen, und nur die rote LED soll leuchten.

Vorbereitung

Tragen Sie in nachstehende Tabelle bitte die Zustände der einzelnen Ausgänge mit 0 und 1 ein.

AW136															
AB136								AB137							
.7	.6	.5	.4	.3	.2	.1	.0	.7	.6	.5	.4	.3	.2	.1	.0
Grüne LED leuchtet	Gelbe LED leuchtet	Rote LED leuchtet	Hupe-Signal ertönt	.	.	Motor Band linkslauf	Motor Band rechtslauf	.	.	Ab. kapazitiv – Zylinder fährt aus	Ab. kapazitiv – Zylinder fährt ein	Ab. Induktiv – Zylinder fährt aus	Ab. Induktiv – Zylinder fährt ein	.	Ab. Mag – Zylinder fährt aus und ein
0	0	1	0	0	0	0	0	0	0	0	1	0	1	0	0

Tabelle 6.1

In Tabelle 6.1 sind die einzelnen Bits des Ausgangswortes AW136 dargestellt. Im Versuch soll die entsprechende Hexadezimalzahl in das Ausgangswort AW136 geschrieben werden. Bitte geben Sie das Ausgangswort als Hexadezimalzahl an:

Hexadezimalzahl: _____


Erstellen Sie den OB100 in der Sprache FUP um die Hexadezimalzahl in das AW136 zu schreiben. Denken Sie daran, dass Sie niemals Zahlen direkt in Bausteinen verwenden sollten. Deklarieren Sie die Hexadezimalzahl im OB100 als Konstante.

Durchführung im Labor

Öffnen Sie ein von Ihnen bereits vorhandenes Projekt aus einer vorherigen Versuchsaufgabe. Löschen Sie eventuell vorhandene Bausteine und den Inhalt der PLC-Variablentabelle.

Erstellen Sie einen OB100 in der Sprache FUP. Geben Sie ihr zuhause erstelltes Programm ein, und laden Den OB100 auf die SPS.

Schauen Sie ob das gewünschte Verhalten am Modell eintritt.

 h_da HOCHSCHULE DARMSTADT UNIVERSITY OF APPLIED SCIENCES	Labor Automatisierungssysteme (ai/m) Versuch 2 Bausteintechnik / Förderband	Fachbereich EIT Verfasser: Prof. Dr.-Ing. S. Simons Dipl.-Ing. J. Meyer
---	--	--

Speichern Sie Ihr Projekt unter den Namen »Band_lauf« in Ihrem Verzeichnis auf dem SPS-Server.

Zeigen Sie diese funktionsfähige und vollständig bearbeitete Teilaufgabe einem Betreuer. Dieser prüft die Aufgabe nochmal mit ihnen, und weist sie gegebenenfalls auf Fehler hin die Sie dann noch verbessern. Ist die Teilaufgabe vollständig bearbeitet, zeichnet der Betreuer die Teilaufgabe auf ihrem Deckblatt ab.

7 Kennenlernen der einzelnen Funktionen des Förderbandmodells

7.1 Bandlauf nach rechts und links

In dieser Aufgabe soll sich das Band beim Drücken des Tasters »A-S3« nach links bewegen. Das Band soll sich auch weiterhin nach links bewegen, wenn der Taster »A-S3« losgelassen wird. Das Band soll erst stoppen, wenn der rote Taster »A-S2« gedrückt wird oder der linke Endtaster »B01« ein TRUE Signal liefert.

Die gleiche Funktionalität soll mit dem Taster »A-S4« für den Rechtslauf realisiert werden. Achten Sie darauf, dass der Motor nicht gleichzeitig nach rechts und links angesteuert werden kann (Verriegelung).


Eine Richtungsumkehr der Bandbewegung soll nur möglich sein, wenn zuvor der rote Taster »A-S2« gedrückt und damit das Band gestoppt wurde.

Vorbereitung

Erstellen Sie ein SPS-Programm in der Sprache FUP mit der gewünschten Funktionalität. Verwenden Sie die symbolischen Namen der E/A Adressen, aus Tabelle 2.1 und 2.2 der Modellbeschreibung.

Entwerfen Sie einen Testplan zum Testen der gewünschten Funktionalität.

Durchführung im Labor

Verwenden Sie das Projekt »Band_lauf« aus der vorherigen Versuchsaufgabe. Erstellen Sie einen OB1 in der Sprache FUP. Gehen Sie in Ihrem TIA Fenster ganz nach rechts zu den »Task Cards«, und klicken Sie auf die Card »Bibliothek«. Öffnen Sie mit dem Icon  die Globale Bibliothek »PLC_Foerderband«. Wählen Sie anschließend unter »PLC_Foerderband« die Kopiervorlage »Foerderband« aus. Ziehen Sie die Bibliothek »Foerderband« per Drag-and-drop in den Ordner PLC-Variablen im Bereich Projektnavigation auf der linken Seite im TIA Portal. Nun stehen Ihnen die symbolischen Namen, mit den richtigen E/A Adressen, aus Tabelle 2.1 und 2.2 der Modellbeschreibung zu Verfügung.

Geben Sie Ihr zu Hause erstelltes FUP-Programm in dem OB1 ein.

Laden Sie nun Ihr Projekt »Band_lauf« in die SPS des Schulungsgeräts und testen Sie die Funktionalität entsprechend Ihres Testplans aus der Vorbereitung am Modell.

Speichern Sie Ihr Projekt »Band_lauf« am Ende.

Zeigen Sie diese funktionsfähige und vollständig bearbeitete Teilaufgabe einem Betreuer. Dieser prüft die Aufgabe nochmal mit ihnen, und weist sie gegebenenfalls auf Fehler hin die Sie dann noch verbessern. Ist die Teilaufgabe vollständig bearbeitet, zeichnet der Betreuer die Teilaufgabe auf ihrem Deckblatt ab.

Als Vorbereitung für die nächste Aufgabe, löschen Sie alle Netzwerke im OB1, aber auf keinen Fall die PLC-Variablentabelle.


Speichern Sie dieses „geleerte“ Projekt unter dem Namen »Zylinder« in Ihrem Verzeichnis auf dem SPS-Server.

7.2 Kennenlernen der verschiedenen Abschiebeeinheiten (Pneumatikzylinder)

In dieser Aufgabe sollen die Abschiebeeinheiten des Förderbandmodells manuell getestet werden. Hierzu soll folgende Funktionalität erfüllt werden:

- Beim Drücken des Tasters »A-S1« soll der Zylinder der Abschiebeeinheit des Magazins ausfahren. Dazu muss das Signal (a137.0) für 1 Sekunde anliegen. Danach soll der Zylinder wieder einfahren.
- Beim Drücken des Tasters »B-S1« soll der Zylinder der Abschiebeeinheit am induktiven Sensor komplett ausgefahren werden. Beim Loslassen des Tasters »B-S1« soll der Zylinder dieser Abschiebeeinheit ausgefahren bleiben.
- Beim Drücken des Tasters »B-S2« soll der Zylinder der Abschiebeeinheit am induktiven Sensor eingefahren werden.
- Beim Drücken von »B-S3« soll der Zylinder der Abschiebeeinheit am kapazitiven Sensor komplett ausgefahren werden. Beim Loslassen des Tasters »B-S3« soll der Zylinder dieser Abschiebeeinheit ausgefahren bleiben.
- Beim Drücken von »B-S4« soll der Zylinder der Abschiebeeinheit am kapazitiven Sensor eingefahren werden.

Gleiche Logiken sollen nicht mehrfach programmiert, sondern als Funktionen oder Funktionsbausteine einmalig realisiert und mehrfach aufgerufen werden.

 h_da HOCHSCHULE DARMSTADT UNIVERSITY OF APPLIED SCIENCES	Labor Automatisierungssysteme (ai/m) Versuch 2 Bausteintechnik / Förderband	Fachbereich EIT Verfasser: Prof. Dr.-Ing. S. Simons Dipl.-Ing. J. Meyer
--	--	--

Vorbereitung

Lesen Sie die Kapitel 2.4, 2.5 und 2.6 des Dokumentes »Beschreibung des Förderband Modells« um die Funktionsweise der verschiedenen Abschiebeeinheiten zu verstehen.

Erstellen Sie ein SPS-Programm in der Sprache FUP mit der gewünschten Funktionalität. Verwenden Sie die symbolischen Namen der E/A Adressen, aus Tabelle 2.1 und 2.2 der Modellbeschreibung. Überlegen Sie beim Entwurf Ihres Programms, welche Logiken sinnvollerweise als Funktionen oder Funktionsbausteine realisiert und mehrfach aufgerufen werden. Überlegen Sie außerdem wieso Sie für die jeweilige Logik eine SPS-Funktion oder einen SPS-Funktionsbaustein verwenden wollen und schreiben Sie die Begründung auf. Entwerfen Sie einen Testplan zum Testen der gewünschten Funktionalität.

Durchführung im Labor

Geben Sie Ihr in der Vorbereitung entworfenen OB1-Programm (ggf. auch Ihr FC1- oder FB1-Programm) in Ihrem TIA-Projekt Zylinder ein. Laden Sie Ihr Projekt in die SPS des Schulungsgeräts, und testen Sie die Funktionalität am Modell entsprechend Ihres Testplans. Speichern Sie Ihr Projekt während des Eingeben gelegentlich, um bei eventuellen Programmabstürzen nicht zu viel Programmcode zu verlieren. Dies gilt für alle Durchführungen.


Zeigen Sie diese funktionsfähige und vollständig bearbeitete Teilaufgabe einem Betreuer. Dieser prüft die Aufgabe nochmal mit ihnen, und weist sie gegebenenfalls auf Fehler hin die Sie dann noch verbessern. Ist die Teilaufgabe vollständig bearbeitet, zeichnet der Betreuer die Teilaufgabe auf ihrem Deckblatt ab.

Speichern Sie Ihr TIA-Projekt Zylinder und schließen Sie es.

Bis hier **MÜSSEN** Sie den Versuch am Versuchstermin mindestens bearbeitet haben,
um ein mEt für V2 zu bekommen!

8 Programmierung der vollständigen Ansteuerung des Förderbandmodells

Entwickeln Sie aus Ihrem Projekt »Band_lauf« ein Programm, dass die volle Funktionalität des Modells realisiert. Dabei sollen Sortierzyklen nacheinander ablaufen.

 h_da HOCHSCHULE DARMSTADT UNIVERSITY OF APPLIED SCIENCES	Labor Automatisierungssysteme (ai/m) Versuch 2 Bausteintechnik / Förderband	Fachbereich EIT Verfasser: Prof. Dr.-Ing. S. Simons Dipl.-Ing. J. Meyer
--	--	---

Ein Sortierzyklus sieht wie folgt aus:

- Wenn sich ein Stein im Magazin befindet, soll Drücken von »A-S1« (grüner Taster) ein Sortierzyklus gestartet werden.
- Ein Stein soll nun aus dem Magazin auf das Band geschoben werden. Der Magazinzyylinder soll anschließend automatisch wieder eingefahren werden.
- Wenn der Magazinzyylinder eingefahren ist, soll der Bandlauf rechts starten.
- Durch Drücken von »A-S2« (roter Taster) soll das Band jederzeit zu stoppen sein. Sollte nach dem Stoppen noch ein Stein auf dem Band liegen, ist dieser von Hand zu entfernen.
- Der Stein soll, je nach Materialart, durch den induktive oder den kapazitiven Sensor erkannt und in diesem Fall soll das Band gestoppt werden.
- Nun soll der entsprechende Zylinder den Stein vom Band abschieben.
- Bitte starten Sie einen neuen Sortierzyklus erst dann, wenn der vorherige abgeschlossen ist.

Vorbereitung


Entwickeln Sie ein Programm für die SPS zur Steuerung des Förderbandmodells. Entwerfen Sie zunächst einen Logikplan. Aus dem Logikplan entwickeln Sie Ihr SPS-Programm in FUP. Entwerfen Sie für den Test in der Vorbereitung einen Testplan, in dem Sie strukturiert alle Funktionen des Automaten in aufeinanderfolgenden Schritten testen. Formulieren Sie zu jedem Testschritt das erwartete Ergebnis. Nehmen Sie außerdem auch Testschritte in den Testplan auf, die Funktionen testen, die nicht funktionieren sollen (z.B. „Starten eines neuen Sortierzyklus bevor der vorherige Zyklus beendet wurde.“).

Durchführung im Labor

Öffnen Sie Ihr Projekt »Band_lauf« und realisieren Sie Ihr zu Hause entwickeltes Programm in diesem Projekt. Denken Sie an das gelegentliche Zwischenspeichern des Projekts. Laden Sie Ihr Projekt in die SPS des Schulungsgeräts, und testen Sie die Funktionalität am Modell entsprechend Ihres Testplans.

Zeigen Sie diese funktionsfähige und vollständig bearbeitete Teilaufgabe einem Betreuer. Dieser prüft die Aufgabe nochmal mit ihnen, und weist sie gegebenenfalls auf Fehler hin die Sie dann noch verbessern. Ist die Teilaufgabe vollständig bearbeitet, zeichnet der Betreuer die Teilaufgabe auf ihrem Deckblatt ab.

Drucken Sie alles Bausteine aus, und versehen die einzelnen Netzwerke mit geeigneten Kommentare.

 h_da HOCHSCHULE DARMSTADT UNIVERSITY OF APPLIED SCIENCES	Labor Automatisierungssysteme (ai/m) Versuch 2 Bausteintechnik / Förderband	Fachbereich EIT Verfasser: Prof. Dr.-Ing. S. Simons Dipl.-Ing. J. Meyer
---	---	--

Literatur

Folien zur Vorlesung Automatisierungssysteme, lfd. Semester

Begleitheft zum Labor Automatisierungssysteme, lfd. Semester

Dokument: »Beschreibung des Förderbandmodells«

Siemens AG: Automatisierung Produkt Support: SIMATIC S7 Handbücher

<http://support.automation.siemens.com> → Produkt Support → Automatisierungstechnik → Automatisierungssysteme → Industrie-Automatisierungssysteme SIMATIC

- [SIMATIC STEP 7 Basic/Professional V15 und SIMATIC WinCC V15 Systemhandbuch](#)
- [SIMATIC WinCC WinCC Engineering V15 - Programmierreferenz](#)

Siemens AG: Siemens Automation Cooperates with Education – SCE Ausbildungsunterlagen:

<https://w3.siemens.com/mcms/sce/de/fortbildungen/ausbildungsunterlagen/Seiten/default.aspx>

DIN EN 61131-3:2014-06: Speicherprogrammierbare Steuerungen - Teil 3:

Programmiersprachen (IEC 61131-3:2013); Deutsche Fassung EN 61131-3:2013; Beuth-Verlag, Berlin, 2014

Berger, H.: Automatisieren mit SIMATIC S7-1500, Publicis Corporate Publishing, Erlangen, 2., wesentlich überarbeitete und erweiterte Auflage, Mai 2017, ISBN: 978-3-89578-451-4.

Wellenreuther, G.; Zastrow, D.: Automatisieren mit SPS, 6. Auflage, Vieweg+Teubner-Verlag, Wiesbaden, 2015, ISBN 978-3-8348-2597-1