

Speicherprogrammierbare Steuerungen

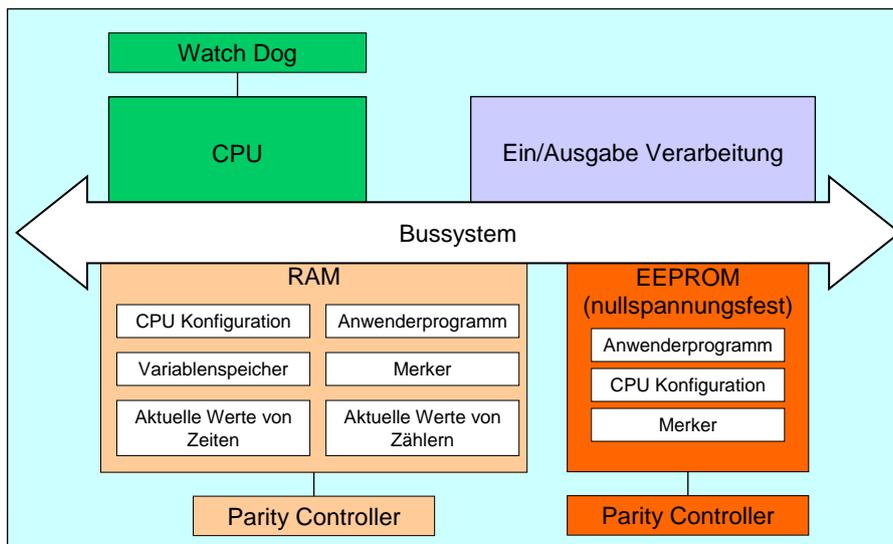
Einführung
Aufbau
KOP, FUP, AWL
Adressierung
Operationssatz
Beispiele

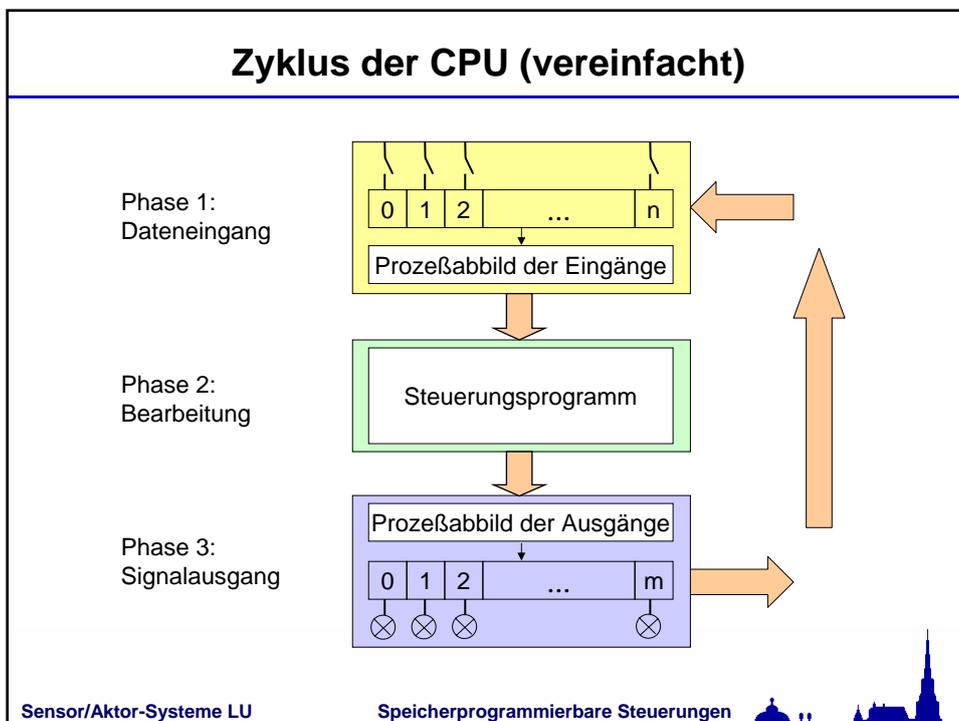
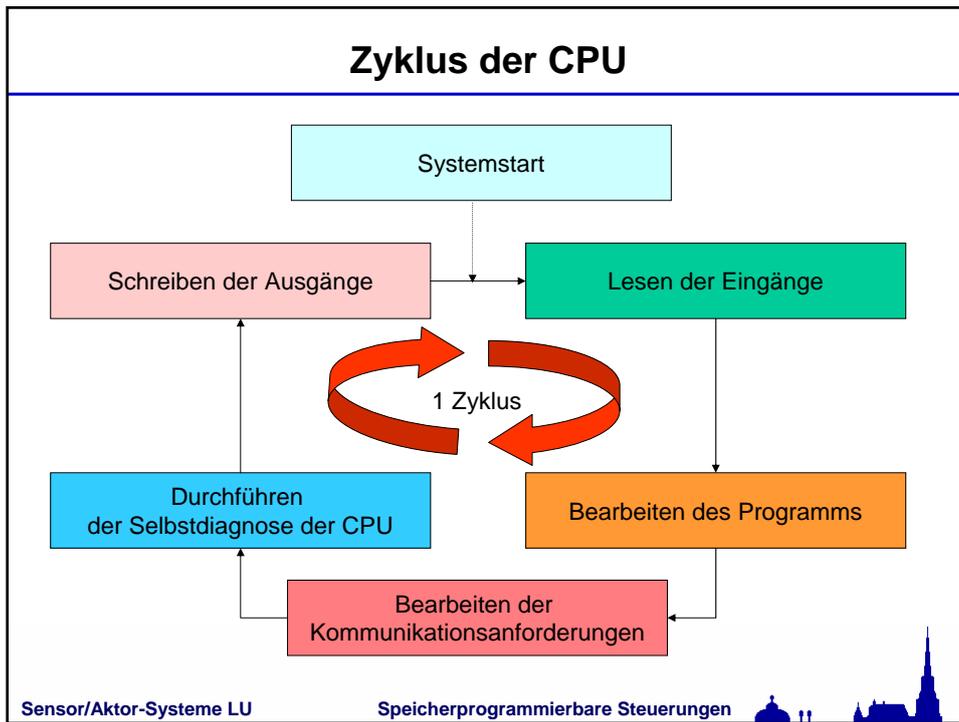


Wolfgang Kastner, EMail: k@auto.tuwien.ac.at
Institut für Rechnergestützte Automation, TU Wien



SPS Aufbau





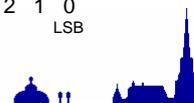
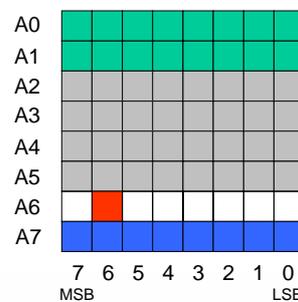
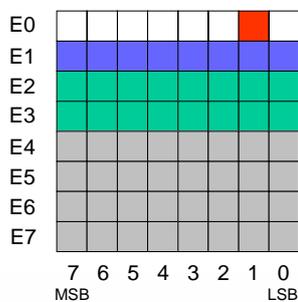
Direkte Adressierung

- **Bit-, Byte-, Wort- und Doppelwort-weiser Zugriff auf**
 - Prozeßabbild der Eingänge (E)
 - Prozeßabbild der Ausgänge (A)
 - Variablenspeicher (V)
 - Merker (M)
 - Ablaufsteuerungsrelais (S)
 - Sondermerker (SM)
- **Wortweiser Zugriff auf**
 - Analogeingänge (AE)
 - Analogausgänge (AA)
- **Zugriff auf**
 - Zeitbits und Zeitwerte (T)
 - Zählbits und Zählwerte (Z)

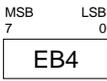
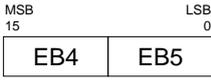
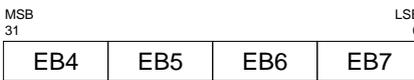


Direkte Adressierung

Adressierung	Format (E, A, V, M, S, SM)	Bsp.
Bit	F[Adr. des Byte].[Adr. des Bit]	E0.1, A6.6
Byte	FB[Anfangsadresse des Byte]	EB1, AB7
Wort	FW[Anfangsadresse des Byte]	EW2, AW0
Doppelwort	FD[Anfangsadresse des Byte]	ED4, AD2



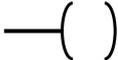
Direkte Adressierung (Big Endian)

- EB4**

 - Adresse des Byte
 - Zugriff im Byteformat
 - Bereichskennung
- EW4**

 - Adresse des ersten Byte
 - Zugriff im Wortformat
 - Bereichskennung
- ED4**

 - Adresse des ersten Byte
 - Zugriff im Doppelwortformat
 - Bereichskennung



SPS Programmiersprachen - KOP

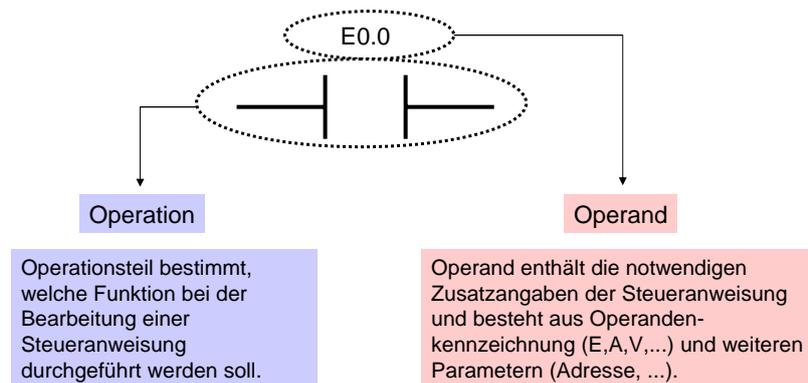
Ein Kontaktplan (KOP) ist eine **graphische Programmiersprache**, die elektrischen Schaltplänen ähnelt. Beim Entwurf eines KOPs arbeitet man mit graphischen Komponenten, die sogenannten **Netzwerken** zugeordnet werden und darin die gewünschte Funktionalität realisieren. Die folgenden Elemente werden zum Erstellen eines Programms verwendet:

- Kontakte** stellen einen Schalter dar, durch den Strom fließen kann. Strom fließt nur dann durch einen Schließerkontakt, wenn der Kontakt geschlossen ist (logischer Wert 1). Strom fließt durch einen Öffnerkontakt oder einen negierten Kontakt (NOT), wenn der Kontakt geöffnet ist (logischer Wert 0).
 
- Spulen** stellen ein Relais bzw. einen Ausgang dar, der durch Signalfuß aktiviert wird.
 
- Boxen** stellen eine Funktion dar (z.B. eine Zeit, einen Zähler oder eine arithmetische Operation), die ausgeführt wird, wenn der Signalfuß die Box erreicht.
 



Steueranweisung

Die Steueranweisung ist die kleinste Einheit eines SPS-Anwenderprogrammes. Eine Anweisung besteht aus einem Operationsteil und einem Operanden.

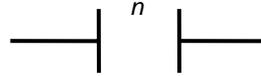


Schließer und Öffner

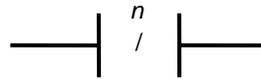
Prozeß			Signal am Eingang der SPS	Abfrage auf ...	
Der Geber ist ein ...	Der Geber ist ...	Spannung am Eingang ist ...		Signalzustand 1 mit dem Symbol 	
			Abfrageergebnis		
Schließer 	betätigt 	vorhanden	 1	"Ja" 1	"Nein" 0
	nicht betätigt 	nicht vorhanden	 0	"Nein" 0	"Ja" 1
Öffner 	betätigt 	nicht vorhanden	 0	"Nein" 0	"Ja" 1
	nicht betätigt 	vorhanden	 1	"Ja" 1	"Nein" 0

Operationssatz (Bitverknüpfung)

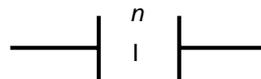
- **Schließerkontakt** ist geschlossen (ein), wenn der Bitwert an Adresse n gleich 1 ist.



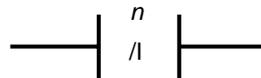
- **Öffnerkontakt** ist geschlossen (ein), wenn der Bitwert an Adresse n gleich 0 ist.



- **Direkter Schließer** ist geschlossen, wenn der Bitwert des angegebenen Eingangs n gleich 1 ist.

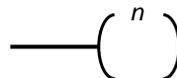


- **Direkter Öffner** ist geschlossen, wenn der Bitwert des angegebenen Eingangs n gleich 0 ist.

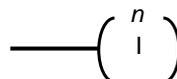


Operationssatz (Bitverknüpfung)

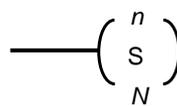
- Bei **Zuweisen** wird der angegebene Bit n entsprechend dem Zustand des Signalfusses gesetzt.



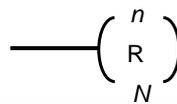
- Bei **Bitwert direkt zuweisen** wird der angegebene physikalische Ausgang n entsprechend dem Zustand des Signalfusses direkt gesetzt.



- Bei **Setzen** wird eine Anzahl N an Ausgängen beginnend mit jenem Wert, der vom Parameter n angegeben wird, gesetzt (eingeschaltet).

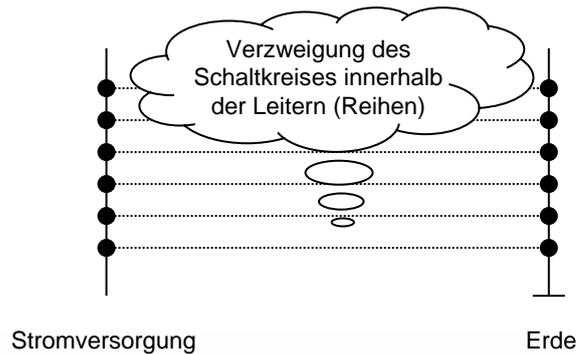


- Bei **Rücksetzen** wird eine Anzahl N beginnend mit jenem Wert, der vom Parameter n angegeben wird, rückgesetzt (ausgeschaltet).



Kontaktplan

Ein Netzwerk setzt sich aus Kontakten, Spulen und Boxen zusammen und bildet einen geschlossenen Kreis. Der Strom fließt von der **linken Stromschiene** (die im KOP-Editor durch eine senkrechte Linie am linken Rand des Fensters dargestellt wird) durch die **geschlossenen Kontakte** und **aktiviert Spulen** oder Boxen. Der KOP wird immer von links oben nach rechts unten abgearbeitet.



Sensor/Aktor-Systeme LU

Speicherprogrammierbare Steuerungen



Kontaktplan

Regeln zum Anordnen von Kontakten:

- Ein Netzwerk muß mit einem Kontakt beginnen.
- Ein Netzwerk darf nicht mit einem Kontakt beendet werden.

Regeln zum Anordnen von Spulen:

- Ein Netzwerk darf nicht mit einer Spule beginnen.
- Spulen dienen zum Beenden von Netzwerken.
- Ein Netzwerk kann mehrere Spulen enthalten, solange die Spulen in parallelen Verzweigungen des Netzwerks angeordnet sind (keine Reihenschaltung).

Regeln zum Anordnen von Boxen:

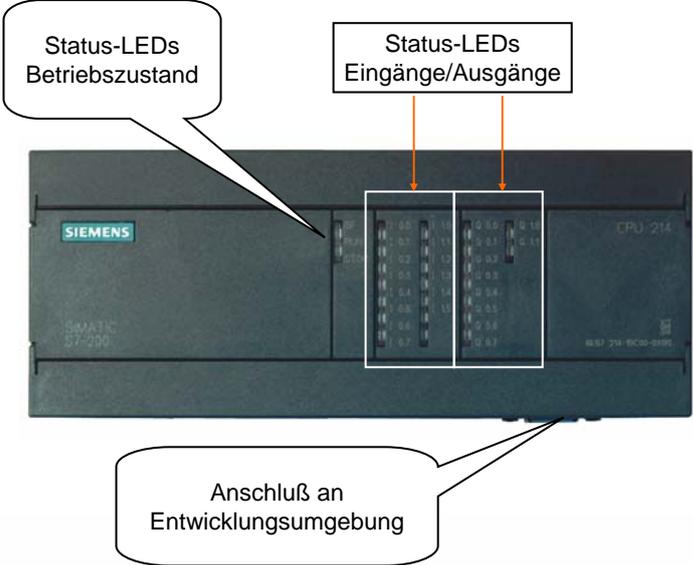
- Verfügt eine Box über einen Freigabeausgang ENO, geht der Signalfuß nach der Box weiter.
- Hat eine Box keinen Freigabeausgang ENO, darf keine weitere Operation nach der Box angeordnet werden.

Sensor/Aktor-Systeme LU

Speicherprogrammierbare Steuerungen



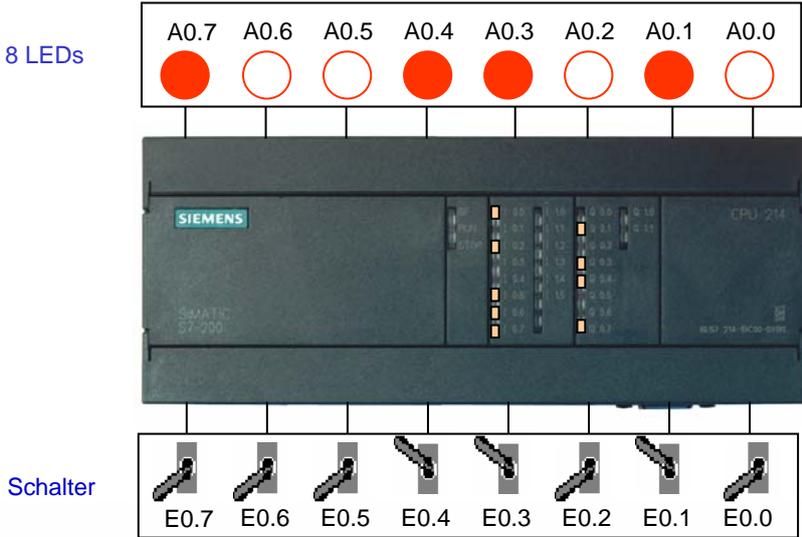
SPS S7-200/CPU 214



Sensor/Aktor-Systeme LU

Speicherprogrammierbare Steuerungen

Anschlußplan SPS-Bedienpanel



Sensor/Aktor-Systeme LU

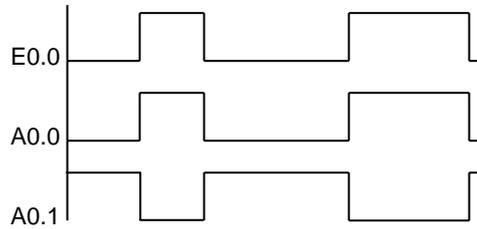
Speicherprogrammierbare Steuerungen

Ein erstes Beispiel

Aufgabe:

Wenn Schalter 0 an,
dann LED 0 an.

Wenn Schalter 0 aus,
dann LED 1 an.

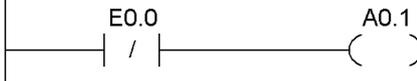


Lösung:

Netzwerk 1



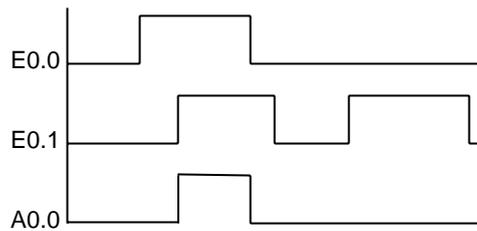
Netzwerk 2



Serienschaltung (UND)

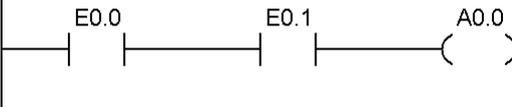
Aufgabe:

Wenn Schalter 0 und
Schalter 1 an, dann
LED 0 an.



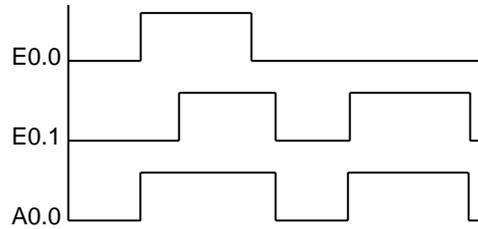
Lösung:

Netzwerk 1

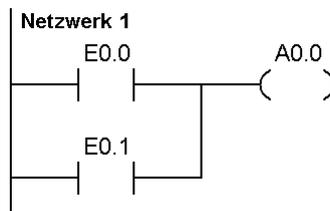


Parallelschaltung (Oder)

Aufgabe:
Wenn Schalter 0 oder Schalter 1 an, dann LED 0 an.

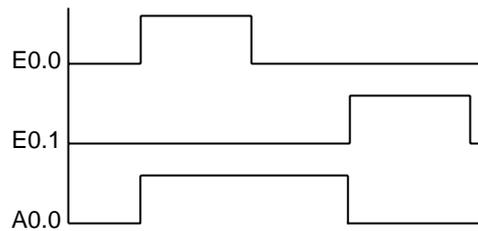


Lösung:

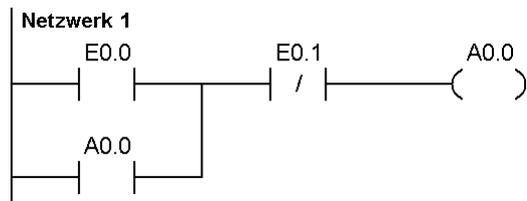


Selbsthaltung

Aufgabe:
Bei Schalter 0 an, soll LED 0 eingeschaltet werden und eingeschaltet bleiben, solange Schalter 1 aus ist.



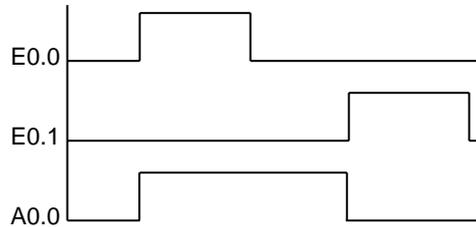
Lösung:



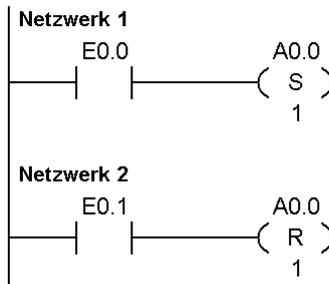
Selbsthaltung mit Setzen und Rücksetzen

Aufgabe:

Bei Schalter 0 an, soll LED 0 eingeschaltet werden und eingeschaltet bleiben, solange Schalter 1 aus ist.



Lösung:

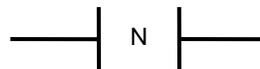


Operationssatz (Bitverknüpfung)

- Der Kontakt **Steigende Flanke** läßt den Signalfluß bei jeder steigenden Flanke genau einen Zyklus lang zu.



- Der Kontakt **Fallende Flanke** läßt den Signalfluß bei jeder fallenden Flanke genau einen Zyklus lang zu.



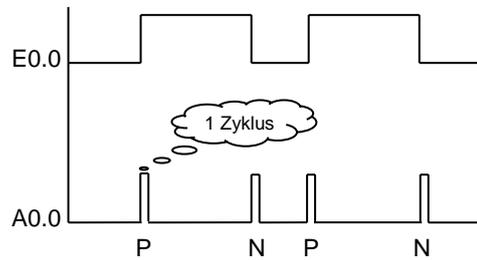
- Der Kontakt **NOT** ändert den Zustand des Signalflusses.



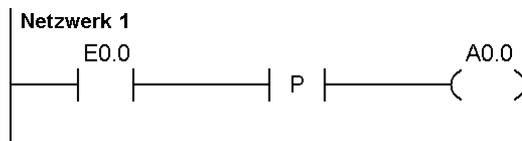
Impulsschaltung

Aufgabe:

Wenn Schalter 7 an ist, soll für einen Zyklus LED 7 eingeschaltet werden.



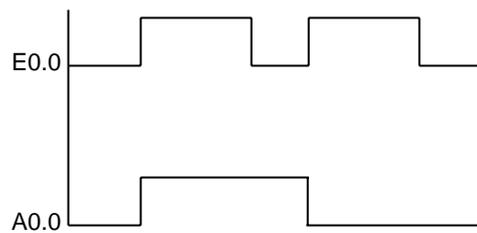
Lösung:



Umkehrschaltung

Aufgabe:

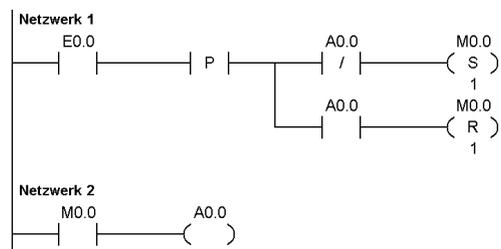
Bei jeder Betätigung von Schalter 0 soll LED 0 ihren Zustand umkehren.



Lösung:

Achtung:

Der Inhalt von Merkern steht sofort (im gleichen Zyklus) für nachfolgende Verknüpfungen bereit!



Sondermerker

Sondermerker bieten verschiedene Status und Steuerinformation. Auf Sondermerker kann Bit-, Byte, Wort- und Doppelwort-weise zugegriffen werden.

Der Sondermerker SMB0 enthält 8 Statusbits, die am Ende eines jeden Zyklus aktualisiert werden. Für die LU wichtig sind:

- SM0.0: Dieses Bit ist immer eingeschaltet
- SM0.1: Dieses Bit ist nur im ersten Zyklus eingeschaltet und sollte bei Initialisierungen verwendet werden.
- SM0.4: Dieses Bit sorgt für einen Takt, der 30 Sekunden lang eingeschaltet ist und 30 Sekunden lang ausgeschaltet ist, bei einer Zykluszeit von 1 Minute.
- SM0.5: Dieses Bit sorgt für einen Takt, der 0.5 Sekunden lang eingeschaltet ist und 0.5 Sekunden lang ausgeschaltet ist, bei einer Zykluszeit von 1 Sekunde.



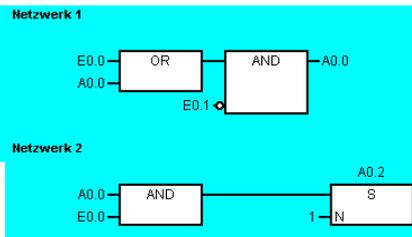
Weitere SPS Programmiersprachen

Funktionsplan (FUP)

- „Datenfluß“
- logische Verknüpfungen durch Funktionsbausteine

```
NETZWERK 1
LD    E0.0
O     A0.0
UN   E0.1
=     A0.0

NETZWERK 2
LD    A0.0
U     E0.0
S     A0.2,1
```



Anweisungsliste (AWL)

- Assembler-ähnlicher Dialekt
- Stackorientierte Abarbeitung



Direkte Adressierung

- Zeit (T)
 - Auflösungen von 1ms, 10ms und 100ms
 - Timer besteht aus
 - aktuellem 16-Bit Wert
 - Zeitbit, das eingeschaltet ist, wenn der aktuelle Wert der Zeit größer oder gleich einem voreingestellten Wert ist.
 - Zugriff: T[Nummer der Zeit]
- Zähler (Z)
 - Vorwärts-Zähler, Vorwärts- und Rückwärts-Zähler
 - Zähler besteht aus
 - aktuellem 16-Bit Wert
 - Zählerbit, das eingeschaltet ist, wenn der aktuelle Wert des Zählers größer oder gleich einem voreingestellten Wert ist.
 - Zugriff: Z[Nummer des Zählers]



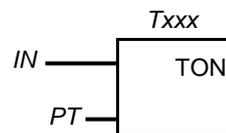
Operationssatz (Zeiten)

- Die Operation **Zeit als Einschaltverzögerung starten (TON)** zählt die Zeit, wenn der Freigabeeingang *IN* eingeschaltet ist.

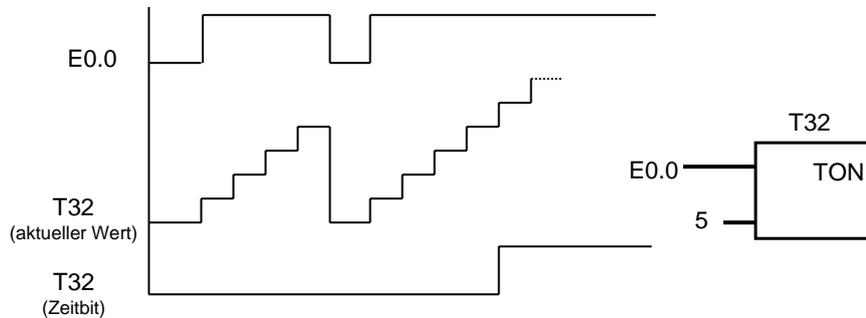
Ist der aktuelle Wert *Txxx* größer als oder gleich dem voreingestellten Zeitwert *PT*, dann wird das Zeitbit eingeschaltet.

Der aktuelle Wert der Zeit wird **gelöscht**, wenn der Freigabeeingang ausgeschaltet wird.

Die Zeit zählt weiter, nachdem die Voreinstellung erreicht ist, und sie stoppt beim Höchstwert von 32767.



Operationssatz (Zeiten)

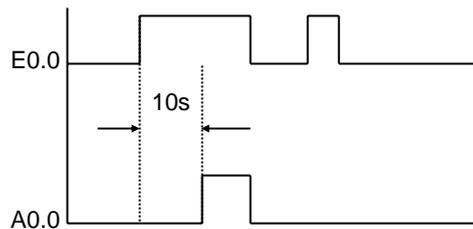


Auflösung	Höchstwert	Nummer der Zeit
1ms	32,767s	T32, T96
10ms	327,67s	T33-T36, T97-T100
100ms	3276,7s	T37-T63, T101-T255

Einschaltverzögerung

Aufgabe:

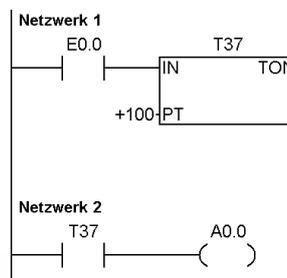
Falls Schalter 0 an ist, soll 10 Sekunden später LED 0 eingeschaltet werden (sofern Schalter 0 dann noch an ist).



Lösung:

Achtung:

Zeiten dürfen in einem SPS-Programm nicht doppelt genutzt werden!

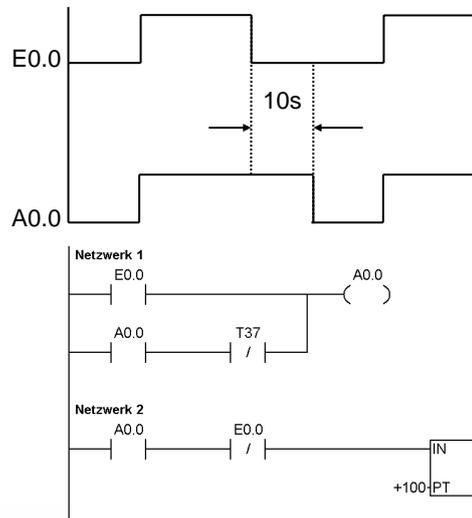


Ausschaltverzögerung

Aufgabe:

Falls Schalter 0 an ist, soll LED 0 leuchten. Sobald Schalter 0 aus ist, soll LED 0 weitere 10 Sekunden leuchten.

Lösung:



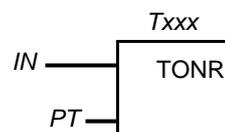
Operationssatz (Zeiten)

- Die Operation **Zeit als speichernde Einschaltverzögerung starten (TONR)** zählt die Zeit, wenn der Freigabeeingang *IN* eingeschaltet ist.

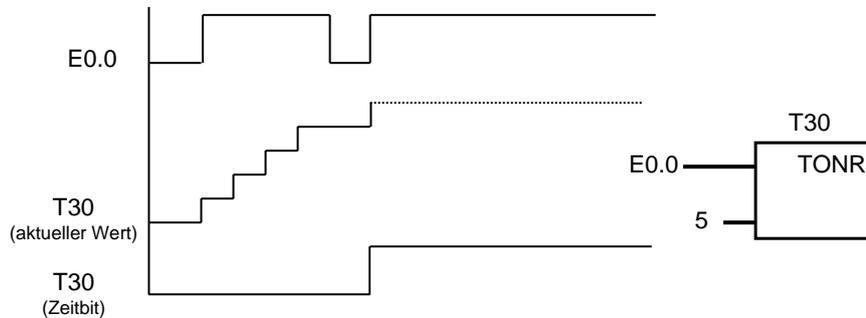
Ist der aktuelle Wert *Txxx* größer als oder gleich dem voreingestellten Zeitwert *PT*, dann wird das Zeitbit eingeschaltet.

Der aktuelle Wert der Zeit wird **gespeichert**, wenn der Freigabeeingang ausgeschaltet wird.

Die Zeit zählt weiter, nachdem die Voreinstellung erreicht ist, und sie stoppt beim Höchstwert von 32767.



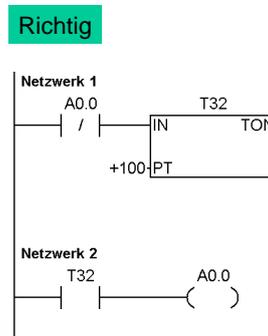
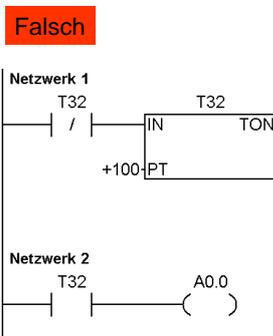
Operationssatz (Zeiten)



Auflösung	Höchstwert	Nummer der Zeit
1ms	32,767s	T0, T64
10ms	327,67s	T1-T4, T65-T68
100ms	3276,7s	T5-T31, T69-T95

Aktualisierung von Zeiten

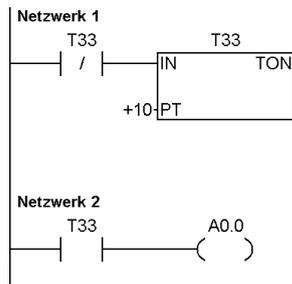
- Auflösung 1ms:
 - Aktualisierung durch Systemprogramm (unabhängig vom Zyklus des Automatisierungssystems und des Anwenderprogramms)



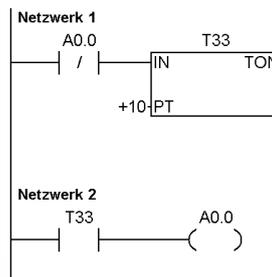
Aktualisierung von Zeiten

- Auflösung 10ms:
 - Aktualisierung automatisch zu Beginn eines jeden Zyklus

Falsch



Richtig



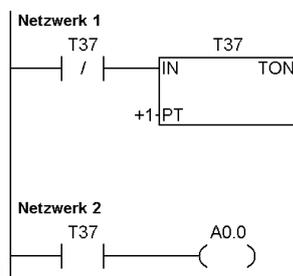
Sensor/Aktor-Systeme LU

Speicherprogrammierbare Steuerungen

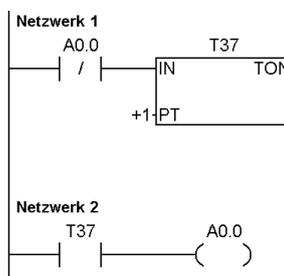
Aktualisierung von Zeiten

- Auflösung 100ms:
 - Aktualisierung nur bei Ausführung einer Zeitoperation

Richtig



Besser



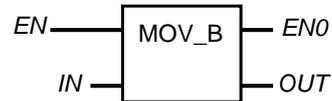
Sensor/Aktor-Systeme LU

Speicherprogrammierbare Steuerungen

Operationssatz

- **Übertragung:**

- Byte, Wort, Doppelwort übertragen

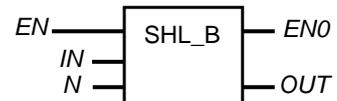


- Anzahl (N) an Bytes, Wörtern, Doppelwörtern übertragen



- **Schieben/Rotieren**

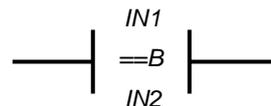
- Byte, Wort, Doppelwort um Schiebezahl N rechts oder links schieben (SM1.0 beachten!)
- Byte, Wort, Doppelwort um Schiebezahl N rechts oder links rotieren (SM1.0 beachten!)



Operationssatz

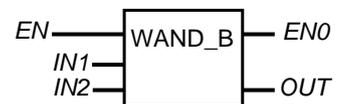
- **Vergleichen:**

- Byte, Wort, Doppelwort (und Realzahlen) vergleichen (==, <>, <, >, <=, >=)



- **Verknüpfen**

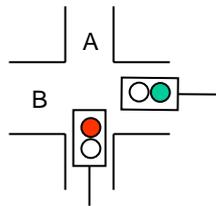
- Bytes, Wörter, Doppelwörter durch UND, ODER, EXKLUSIV ODER verknüpfen oder Einerkomplement erzeugen.



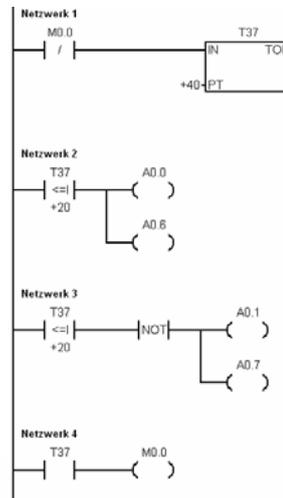
Fußgeher-Ampelsteuerung

Aufgabe:

Für (schnelle) Fußgeher soll eine Ampelsteuerung implementiert werden. LED0 (Rot) und LED1 (Grün) gelten für Straße A, LED7 (ROT) und LED6 gelten für Straße B. Umschaltdauer: 2 Sekunden!

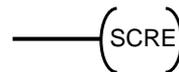
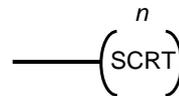
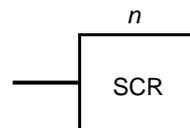


Lösung:



Operationssatz (Programmsteuerung)

- Die Operation **Ablaufsteuerungsrelais (SCR)** kennzeichnet den Beginn eines SCR-Segments. Ist das Ablaufsteuerungsrelais n eingeschaltet, wird der Signalfuß zum SCR-Segment freigegeben.
- Die Operation **Flanke Ablaufsteuerungsrelais (SCRT)** kennzeichnet das nächste Ablaufsteuerungsrelais n , das gesetzt werden soll. Fließt Strom zur Spule, dann wird das spezifizierte Bit n eingeschaltet und das Bit, das dieses SCR-Segment freigegeben hat, wird ausgeschaltet.
- Das SCR-Segment muß mit der Operation **Ende Ablaufsteuerungsrelais (SCRE)** beendet werden.

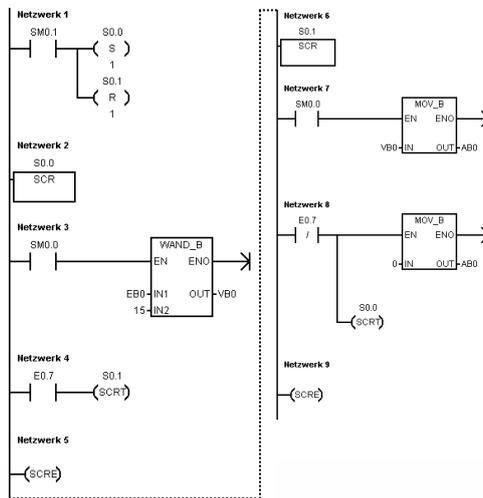


Code-Eingabe

Aufgabe:

Auf Schalter 0-3 soll beliebig lange ein Code eingegeben werden. Sobald Schalter 7 eingeschaltet wird, soll der Code auf den LEDs 0-3 angezeigt werden. Falls Schalter 7 ausgeschaltet wird, soll der Code gelöscht werden.

Lösung:



Weiterführende Information



- **Betreute Übungszeit Tutoren**
- S7 Microwin Online-Hilfe
- S7-200 CPU 214 Handbuch (Kap. 10)
- Information im Netz (<http://www.tilab.tuwien.ac.at>)
- Forum (TUWIS)