

Speicher Programmierbare Steuerung

mit dem AT89S8252-Mikrocontroller Board

Von Erik Martens

Bei speicherprogrammierbaren Steuerungen, kurz SPS genannt, stehen Zuverlässigkeit, Bedienungskomfort und Sicherheit an erster Stelle. Mit Hilfe eines entsprechend programmierten Mikrocontrollers lassen sich speicherprogrammierbare Steuerungen realisieren, die sogar Industriestandards erfüllen.

Mit dem AT89S8252-Mikrocontroller-Board (Elektor Dezember 2001), ein paar Standardbauteilen und einer Firmware lässt sich eine speicherprogrammierbare Steuerung aufbauen. Die Firmware, zusätzliche Software für den Mikrocontroller, steuert und überwacht selbst entwickelte SPS-Programme.

Systembeschreibung

Um einen Mikrocontroller für SPS-Zwecke zu verwenden, werden vier I/O-Anschlüsse des Mikrocontrollers benötigt (P1.0...P1.3). Der Pegel an P1.0 informiert den Mikrocontroller darüber, ob es sich bei einem Reset um einen internen oder einen Hardware-Reset handelt. Dieser Pegel wird durch das diskret mit IC2.A und IC2.B aufgebaute Flipflop bestimmt. Nach einem Hardware-Reset wird das Flipflop durch die Firmware über P1.1 gesetzt. Mit diesem Anschluss lassen sich auch die Ausgangseinheiten über IC3.D,

IC2.D und T1 deaktivieren.

Mit dem Schalter STOP/RUN/RESET kann das SPS-Programm über die Firmware gestartet und beendet werden. Außerdem kann dieser Schalter (mit R3) einen Reset des Mikrocontrollers erzwingen. Die Stellung des Schalters wird über P1.2 erkannt. Steht der Schalter in Position RUN, so kann P1.2 die Ausgangseinheiten deaktivieren. Die LEDs D7 und D8 informieren über den Status des SPS-Programms.

Fällt die Spannung am Spannungsregler unter einen Wert von etwa 8,2 V, so erfolgt über P1.3 eine entsprechende Rückmeldung an den Mikrocontroller und die Ausgangseinheiten werden deaktiviert.

Im Schaltbild sind auch zwei Eingangseinheiten zu erkennen. Die Anzahl dieser Einheiten kann je nach Anforderung über die Anschlüsse P0, P2 und P3 erweitert werden. Die Eingangseinheiten sind galvanisch

getrennt, gesichert gegen Spannungsspitzen sowie vertauschte Polarität und sind in der Lage, positive Gleichspannungen bis zu 24 V zu schalten.

Auch die Ausgangseinheiten sind galvanisch getrennt. Ihre Anzahl lässt sich je nach Anforderung ebenfalls über P0, P2 und P3 erweitern. Die Ausgangseinheiten werden erst bei angesteuertem Transistor T1 aktiviert, also wenn alle Bedingungen erfüllt sind: richtige Versorgungsspannung, P1.1 und P1.2 im Zustand logisch 1 und Schalter S1 in Stellung RUN.

Zur Programmierung des Mikrocontrollers kann auch dessen SPI-Port verwendet werden (P1.5, P1.6 und P1.7). Die Programmierung erfolgt bei aktiviertem Reset. Ein hoher RS232-Pegel sorgt dank R8...R11 sowie IC3A für einen kontinuierlichen Reset des Controllers. Die Programmierfunktionalität des SPI-Ports wird im Schaltbild nicht gezeigt.

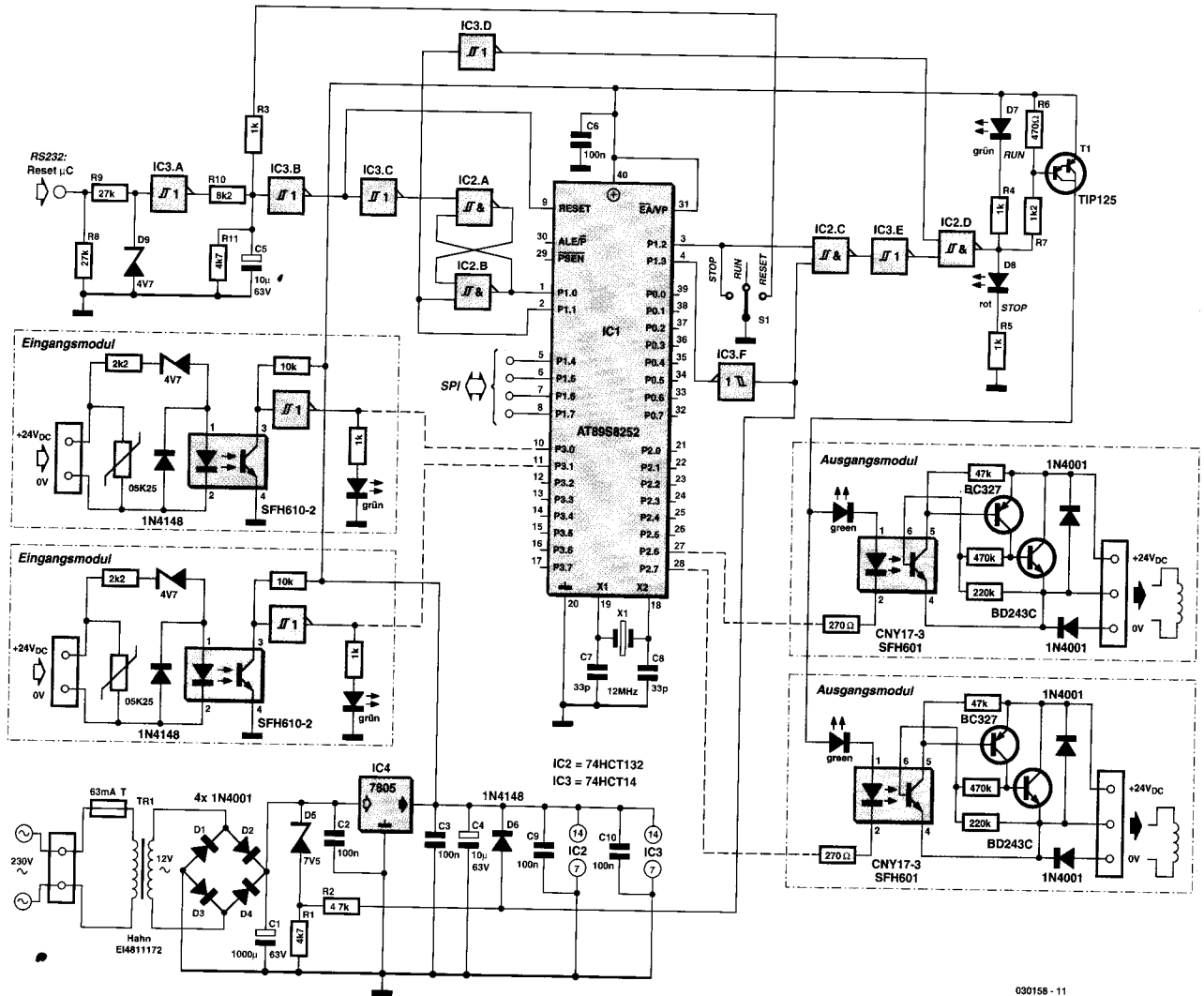


Bild 1. Schaltbild einer speicherprogrammierbaren Steuerung mit dem AT89S8252

Firmware

Der Kasten am Ende dieses Artikels zeigt ein nur etwa 100 Bytes großes einfaches Beispiel für die erforderliche Firmware, die in der Shareware-Assemblersprache TASM geschrieben wurde. Der Umgang mit dieser Sprache und die Übertragung der Software auf den Mikrocontroller wurde bereits ausführlich im Programmierkurs zum Flash-Board (Elektor 1...9/2002) beschrieben.

Die Firmware übernimmt folgende Funktionen:

Reset

Unmittelbar nach einem Reset des Mikrocontrollers muss die Firmware ausgeführt werden. Nach einem Hardware-Reset muss der erste SPS-Zyklus gestartet werden. Dazu muss sich S1 in Stellung RUN befinden und die Versorgungsspannung den geforderten Wert aufweisen. Bei einem internen Rest (nach einer Überschreitung der Zykluszeit) muss zuerst der Schalter S1

in Stellung STOP und anschließend in Stellung RUN gebracht werden, bevor der erste SPS-Zyklus ausgeführt wird.

RUN/STOP-Funktionalität

Die wichtigste Aufgabe der Firmware besteht in der Überwachung der drei Bedingungen, die zum Start eines SPS-Zyklus notwendig sind:

STOP/RUN/RESET-Schalter

Mit diesem Schalter wird das SPS-Programm gestartet (RUN) und angehalten. Wird der Schalter auf STOP gestellt, so wird der laufende Zyklus ausgeführt und kein neuer Zyklus gestartet. In dieser Stellung werden auch die Ausgangsstufen über die Hardware deaktiviert.

Überwachung der Versorgungsspannung

Vor dem Start eines SPS-Zyklus wird

durch die Firmware immer überprüft, ob die Versorgungsspannung (noch) den erforderlichen Wert besitzt. Sinkt die Spannung am Stabilisator IC4 unter 8,2 V, so wird kein neuer SPS-Zyklus gestartet, sondern die Service-Routine ERRPS aufgerufen. Diese Routine kann verwendet werden, um Variablen in einem nicht-flüchtigen Speicher (EEPROM des Mikrocontrollers) zu schreiben, bevor ein weiterer Abfall der Spannung die Funktion des Controllers vollständig unterbindet. Bei vorübergehendem Absinken der Versorgungsspannung (etwa durch eine kurzzeitige Störung, bei der kein Reset des Controllers erfolgt) wird nach dem erneuten Einsetzen der Spannung der SPS-Zyklus fortgesetzt.

SPS-Zyklusüberwachung

Der AT89S8252 verfügt über einen internen Watchdog-Timer, der zum Glück für die Überwachung der Zykluszeit des SPS-Programms sorgt. Nach jedem Start eines SPS-Zyklus

erfolgt ein Reset des Watchdog-Timers. Dauert ein SPS-Zyklus zu lange (zum Beispiel durch eine Endlosschleife im SPS-Programm), so bewirkt der interne Watchdog einen Reset des Mikrocontrollers. Im Reset-Zustand des Mikrocontrollers nimmt P1.1 den Wert logisch 1 an, was eine unmittelbare Deaktivierung der Ausgangseinheiten zur Folge hat. Die Firmware erkennt nun am Zustand von Flipflop IC2.A/B den Reset und geht in den STOP-Zustand über. Trotz der Tatsache, dass sich S1 in Stellung RUN befindet, bleibt das SPS-Programm im STOP-Zustand. Nachdem S1 in die Stellung STOP gebracht und dieser Zustand wieder aufgehoben wurde, wird verhindert, dass das SPS-Programm nach dem Überschreiten der PC-Zykluszeit erneut gestartet wird. Die SPS-Zykluszeitüberwachung lässt sich auf folgende Werte einstellen: 16, 32, 64, 128, 256, 512, 1024 und 2048 ms.

Anwendung der Mikrocontroller-SPS-Firmware

Die Firmware dient als Ausgangspunkt zum Erstellen eigener SPS-Anwendungen. Dazu müssen lediglich die Routinen MAIN, INIUC, INIPLC, ERRPS und ERRCY um die erforderlichen Funktionen erweitert werden.

MAIN: Diese Routine wird zyklisch ausgeführt. In ihr werden Eingangsdaten gelesen, logische Zusammenhänge programmiert und Ausgänge angesteuert. Das Durchlaufen dieser Routine darf nicht länger als die eingestellte Watchdog-Zeit dauern. Seien Sie innerhalb dieser Routine vorsichtig mit Warteschleifen und Rücksprüngen.

INIUC: Diese Routine wird beim Einschalten des SPS-Mikrocontrollers einmal aufgerufen. Mit ihr lassen sich zum Beispiel Variablen aus dem EEPROM zum Mikrocontroller übertragen. **Listing 1** zeigt ein Beispiel, wie man aus der Adresse 16 des EEPROMS Daten in den Akku des Controllers überträgt.

INIPLC: Diese Routine wird einmal aufgerufen, wenn S1 in Stellung RUN gebracht wird (Warmstart). In dieser Routine lassen sich zum Beispiel anwendungsabhängige Startwerte in die betreffenden Variablen kopieren oder Kommunikationslinks initialisieren. Die Zyklusüberwachung ist in dieser Routine nicht aktiviert.

Listing 1

```
ORL    WMCON,00001000B set EEMEN, uC EEPROM auswählen
MOV    DPL,#010H
MOV    DPH,#000H
MOVX   A,@DPTR          Übertrage den Inhalt von Adresse 16 in den Accu
```

Listing 2

```
MOV    DPL,#010H
MOV    DPH,#000H
ORL    WMCON,#00010000B set EEMWE, write enable uC EEPROM
M001   MOX    A,WMCON          warten, bis uC EEPROM bereit ist
JNB    ACC.1,M001
MOV    A,B
MOVX   @DPTR,A
ANL    WMCON,#11101111B reset EEMWE, write disable uC EEPROM
```

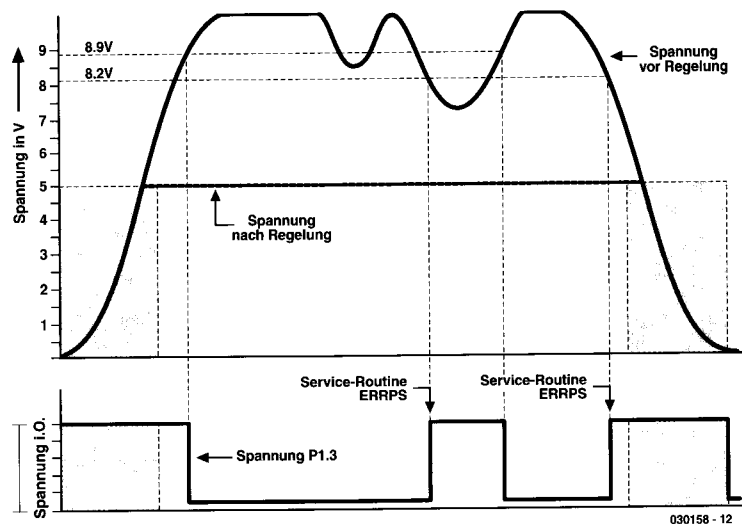


Bild 2. Überwachung der Versorgungsspannung

ERRPS: Die ERRPS-Routine wird entweder beim Ausschalten der Versorgungsspannung oder bei deren Absinken unter einen kritischen Wert aufgerufen. Mit dieser Routine lassen sich remanente Variablen in das EEPROM des Controllers übertragen. Bei remanenten Variablen handelt es sich um Variablen, deren Wert auch nach einem Ausfall der Versorgungsspannung beibehalten werden muss. **Listing 2** zeigt zum Beispiel, wie der

Inhalt des Registers B in die Adresse 16 des EEPROM kopiert wird.

ERRCY: Diese Routine wird kontinuierlich aufgerufen, wenn eine Überschreitung der Zykluszeit vorliegt und S1 sich in Stellung RUN befindet. In dieser Routine kann zum Beispiel eine LED angesteuert werden, die signalisiert, dass ein Programmierfehler erkannt wurde. In der Routine INIPLC wird die LED dann wieder ausgeschaltet.

Vorsicht

Das Netzteil dieser Schaltung ist mit einem Trafo ausgestattet, der über seine Primärseite mit dem 230-V-Lichtnetz in Verbindung steht. Aus Sicherheitsgründen müssen die entsprechenden Regeln für den Umgang mit Netzspannung daher unbedingt beachtet werden. Informationen zu diesem Thema finden Sie auf der regelmäßig erscheinenden Sicherheitsseite in Elektor. Mit dem im Schaltbild angegebenen Netztrafo kann die Einheit als Klasse-II-Gerät in einem Gehäuse untergebracht werden.

Schlussbetrachtung

Mit dem AT89S8252-Flashboard lassen sich realistische SPS-Anwendungen erstellen, die den Industriestandards im Hinblick auf Arbeitsweise, Bedie-

nung und Datenaustausch genügen. Die Anzahl der Ein- und Ausgänge kann gemäß individueller Anforderungen erweitert werden. Der Quellcode der vom Controller benötigten Firmware wird vom Autor

für Elektor-Leser kostenlos zur Verfügung gestellt und kann unter der EPS-Nr.: 030158-11 von unserer Webseite heruntergeladen werden.

(030158)

Listing 3. Beispiel für Firmware im 89S8252.

```

;FIRMWARE µC-PLC
WMCON    .equ    096H
P1.0     .equ    090H
P1.1     .equ    091H
P1.2     .equ    092H
P1.3     .equ    093H
        .org    0000H           ; Start-Adresse nach Reset
LJMP     START
        .org    0200H

;RESET µC
START    ANL     WMCON,#1111110B ; disable Watchdog
        JB      P1.0,SOFT       ; jump, Software-Reset

;HARDWARE RESET µC
HARD     CLR     P1.2           ; Ausgangssperre mit P1.2, damit mit P1.1 das Reset-Flipflop gesetzt wird
        CLR     P1.1           ; Setzt Reset-Flipflop
        JNB     P1.0,HARD      ; jump, warte bis Reset-Flipflop gesetzt ist
        SETB    P1.2           ; Ausgangsfreigabe mit P1.2
        ACALL   INIUC          ; µC-Initialisierung nach Reset
        SJMP    NCYC

;SOFTWARE RESET µC
SOFT     ACALL   ERRCY         ; STOP/RUN/RESET-Schalter auf RUN nach Zykluszeit-Überschreitung
        JB      P1.2,SOFT      ; jump, warte bis STOP/RUN/RESET-Schalter auf STOP gesetzt wird
        ACALL   INIUC          ; µC Initialisierung nach Reset
        SJMP    NCYC          ; jump, STOP/RUN/RESET-Schalter auf STOP gesetzt

;PLC CYCLUS
PLC      JNB     P1.2,NCYC     ; jump, STOP/RUN/RESET-Schalter steht auf STOP
        JB      P1.3,NCYC     ; jump, Spannung nicht ok
        SJMP    WDTR          ; jump, Bedingung zum Starten eines neuen PLC-Zyklus ist ok
NCYC     ANL     WMCON,#1111110B ; Abschalten Watchdog
        SETB    P1.1           ; Ausgangssperre mit P1.1
        JNB     P1.3,PSOK     ; jump, Spannung ist ok
        ACALL   ERRPS         ; Spannung ist nicht ok
PSNOK    JB      P1.3,PSNOK    ; jump, warte, bis Spannung ok ist
PSOK     MOV     R2,#100       ; delay: #100 x #50 x 2µs = 10.000µs = 10ms
DELO     MOV     R1,#50
DEL1     DJNZ   R1,DEL1
        DJNZ   R2,DEL2
        JNB     P1.2,NCYC     ; jump, STOP/RUN/RESET-Schalter steht auf STOP
        JB      P1.3,NCYC     ; jump, Spannung ist nicht ok
        ACALL   INIPLC        ; PLC-Initialisierung, nachdem STOP/RUN/RESET-Schalter auf RUN geschaltet wurde
        ; oder Zurückkehren der Spannung.
        CLR     P1.1           ; Ausgangsfreigabe mit P1.1, STOP/RUN/RESET-Schalter steht auf RUN und Spannung ist ok
        ORL     WMCON,#11100000B ; WMCON: ms WMCON: ms
        ; #000xxxxx 16 #100xxxxx256
        ; #001xxxxx 32 #101xxxxx512
        ; #010xxxxx 64 #110xxxxx1024
        ; #011xxxxx 128 #111xxxxx2048
WDTR     ORL     WMCON,#00000001B ; Freigabe watchdog
        ORL     WMCON,#00000010B ; Watchdog-Timer-Rese

;PLC PROGRAMMA
MAIN     ...
        LJMP    PLC           ; jump, nächster PLC-Zyklus

;INITIALISIERUNG µC einmaliger Aufruf nach Hardware-Reset
INIUC    RET

;INITIALISIERUNG PLC (STOP/RUN/RESET-Schalter auf RUN geschaltet oder Zurückkehren der Spannung)
INIPLC   RET

;POWER SUPPLY ERROR (einmaliger Aufruf nach Ausschalten der Spannung)
ERRPS    RET

;CYCLUSTIJD ERROR (STOP/RUN/RESET-Schalter auf RUN nach Zykluszeit-Überschreitung)
ERRCY    RET

```