

# PIC-Grundlagen, Teil 5

*In diesem Teil der Artikelserie wird eine kleine Testschaltung vorgestellt, die den Test selbsterstellter Programme ermöglicht.*

## Allgemeines

Nachdem in den letzten Teilen der Artikelserie die Grundlagen zum Erstellen von eigenen Programmen, bis hin zur Programmierung des Controllers mit dem PIC-START-PLUS-Programmiergerät vermittelt wurden, soll nun eine kleine Testplatine vorgestellt werden, mit der die eigenen Programme ausprobierbar sind.

Zwar können die einzelnen Programmabläufe bereits komfortabel mit dem Softwaresimulator auf dem PC durchgespielt werden, wenn es jedoch zum Beispiel um die Überprüfung von zeitlichen Abläufen geht, sind die Programmierung eines Controllers und die Überprüfung im Zielsystem unbedingt erforderlich.

Ebenso wünscht sich der Einsteiger, der seine ersten Erfahrungen im Programmie-

ren von PICs macht, nach der grauen Theorie eine kleine Schaltung, die das eigene erstellte Programm ausführt. So ist auch „nur“ eine blinkende Leuchtdiode für den Programmierer ein Zeichen dafür, daß er die Vorgehensweise und Funktionen verstanden hat und gibt so neuen Aufwind und Mut für weitere Projekte.

Zwar ist es auch möglich, eine kleine Schaltung auf einem Steckbrett oder auf einer Lochrasterplatine aufzubauen, in der dann der PIC überprüft werden kann. Unterläuft einem dabei jedoch ein Fehler, so arbeitet der Controller nicht, und es ist schlecht festzustellen, ob nun ein Fehler in der Testschaltung vorliegt, oder ob bereits ein Fehler im Programm oder bei der Programmierung des PICs aufgetreten ist.

Dieser Unsicherheitsfaktor kann nun mit der Testplatine ausgeschlossen werden. Die Platine ist mit zwei IC-Sockeln bestückt, in

die ein PIC 16C55, 16C57, 16C54 oder 16C56 einsetzbar ist. Über 8 LEDs sind die Ports RB0 bis RB7 für Ausgaben nutzbar. An RA0 bis RA3 ist ein 4fach DIP-Schalter angeschlossen, mit dem Eingaben erfolgen können. Die PICs 16C55 und 16C57 verfügen zusätzlich über die Ports RC0 bis RC7, die auf eine Stiftleiste geführt sind und nach eigenen Vorstellungen beschaltet werden können.

## Schaltung

Die Schaltung der PIC-Testplatine ist in Abbildung 23 dargestellt. Die Spannungsversorgung erfolgt auf einfache Weise über ein Steckernetzgerät, das eine Gleichspannung im Bereich von 8 V bis 15 V liefern muß und an die Klinkenbuchse BU 1 angeschlossen wird. Dabei muß der Pluspol mit dem vorderen Kontakt des Steckers ver-

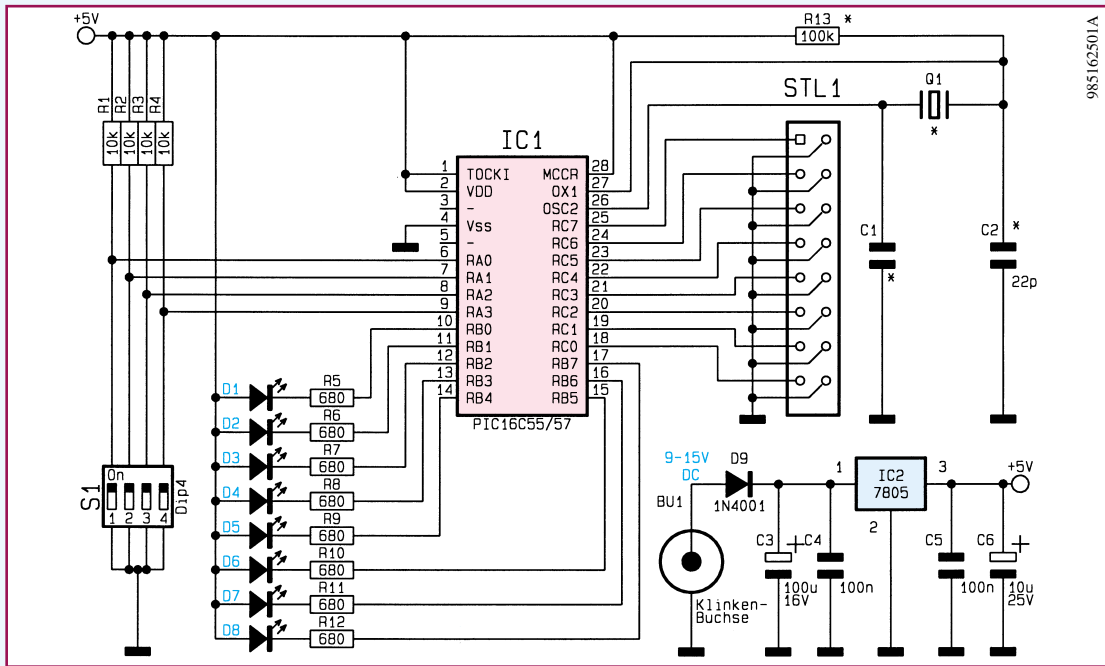


Bild 23: Schaltbild der PIC-Testplatine

bunden sein. Um bei einer versehentlichen Verpolung die Schaltung vor Zerstörung zu schützen, ist die Diode D 9 vom Typ 1N4001 vorgesehen.

Die Stabilisierung der 5V-Betriebsspannung erfolgt über den Spannungsregler IC 2 vom Typ 7805, wobei die Kondensatoren C 3 bis C 6 zur Pufferung und Unterdrückung von Schwingneigungen dienen.

Um auf der Testplatine die 18poligen PICs 16C54 und 16C56, sowie die 28poligen Typen 16C55 und 16C57 betreiben zu können, sind zwei IC-Sockel vorgesehen, die den jeweiligen Typ aufnehmen. Dabei sind die gleichnamigen Signale der beiden Sockel miteinander verbunden.

Die positive Spannungsversorgung erhält der PIC IC 1 über den Pin 2 und die Masse ist mit Pin 4 verbunden. Der Reset-Eingang (Pin 28) des Controllers ist mit der positiven Spannungsversorgung verbunden, so daß der Reset von der internen Reset-Schaltung des PICs generiert wird.

Der Oszillator, der die Taktfrequenz des PICs bestimmt, ist mit den Bauteilen R 13,

Q 1 und den Kondensatoren C 1 und C 2 aufgebaut, wobei verschiedene Oszillatorvarianten realisierbar sind. Im einfachsten Fall wird ein RC-Oszillator aufgebaut, wobei die Bauteile Q 1 und C 1 entfallen. Bei einer Kombination aus einem 100k-Widerstand (R 13) und einem 22pF-Kondensator (C 2) ergibt sich für den Oszillator eine Frequenz von ca. 160 kHz.

Um einen Quarzoszillator zu realisieren, sind die Kondensatoren C 1 und C 2 sowie der Quarz Q 1 zu bestücken, wobei der Widerstand R 13 dann entfällt.

Für den Test sind die Ports RA 0 bis RA 3 des IC 1 mit dem DIP-Schalter S 1 beschaltet, über den die Pins einzeln auf Masse-Potential gelegt werden können. Im geöffneten Zustand führen die Pins über die 10kΩ-Pull-Up-Widerstände R 1 bis R 4 High-Potential.

Für Ausgaben können die Ports RB0 bis RB7 genutzt werden, die jeweils über einen Vorwiderstand R 5 bis R 12 mit einer LED D 1 bis D 8 verbunden sind. Die

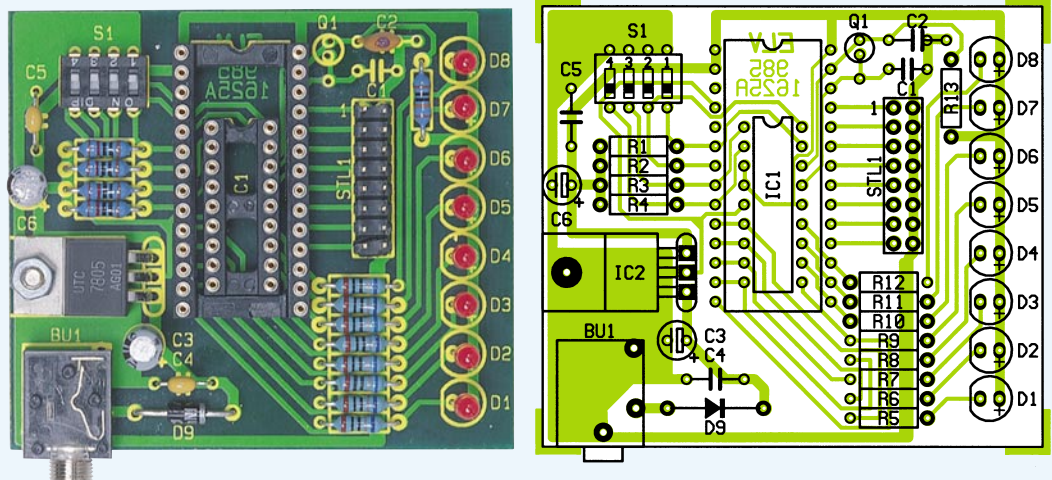
Anoden der LEDs sind mit der 5V-Versorgungsspannung verbunden, so daß die Portpins auf Low-Potential geschaltet werden müssen, um eine LED zu aktivieren.

Die 28poligen PICs 16C55 und 16C57 verfügen zusätzlich über die Portpins RC0 bis RC7, die auf die 2reihige Stiftleiste STL 1 geführt sind. Diese Pins können sowohl als Ein- und Ausgänge genutzt werden und erlauben den Anschluß von externen Komponenten.

### Nachbau

Der Nachbau der PIC-Testplatine gestaltet sich sehr einfach, da nur wenige Bauteile auf der einseitigen, 66 x 58 mm messenden Leiterplatte zu bestücken sind. Die Bestückung erfolgt in gewohnter Weise anhand der Stückliste und des Bestückungsdruckes, wobei zuerst die niedrigen, gefolgt von den höheren Bauteilen bestückt und von der Unterseite verlötet werden. Danach sind die überstehenden Drahtenden mit einem Seitenschneider abzuschnei-

Ansicht der fertig bestückten PIC-Testplatine mit zugehörigem Bestückungsplan



den, ohne dabei die Lötstellen zu beschädigen.

Für das IC 1 sind zwei IC-Sockel zu bestücken, zur späteren Aufnahme des zu testenden PIC. Dabei muß der mittlere Steg der 28poligen IC-Fassung mit einem Seitenschneider entfernt werden, da der 18polige IC-Sockel innerhalb der nun freien Fläche des 28poligen Sockels gesetzt wird.

Die Anschlußdrähte des Spannungsreglers IC 2 sind um 90° nach hinten abzuwickeln, und das Gehäuse ist vor dem Verlöten mit einer M3x 6mm-Zylinderkopfschraube und einer M3-Mutter auf der Leiterplatte zu verschrauben.

Bei der Bestückung der Oszillatorschaltung ist darauf zu achten, das bei einem RC-Oszillator nur der Widerstand R 13 und der Kondensator C 2 zu bestücken sind und die Bauteile Q 1, und C 1 entfallen. Soll ein Quarzoszillator Einsatz finden, so entfällt R 13 und C 1, C 2 und Q 1 sind zu bestücken.

Damit ist der Aufbau abgeschlossen, und der erste Test eines PIC-Controllers kann folgen.

## Testprogramm

Ein einfaches Testprogramm für den PIC 16C54 soll das erste Programm sein, das mit Hilfe der Testplatine überprüft wird. Es handelt sich hierbei um ein einfaches Lauflicht, bei dem immer ein Port (RB0 bis RB7) auf Low-Pegel liegt, wodurch eine LED leuchtet. Nach einer Wartezeit, die

### Stückliste: PIC-Testplatine

#### Widerstände:

680Ω ..... R5-R12  
10kΩ ..... R1-R4  
100kΩ ..... R13

#### Kondensatoren:

22pF/ker ..... C2  
100nF/ker ..... C4, C5  
10µF/25V ..... C6  
100µF/16V ..... C3

#### Halbleiter:

7805 ..... IC2  
1N4001 ..... D9  
LED, 3mm, rot ..... D1-D8

#### Sonstiges:

Klinkenbuchse, 3,5mm,  
mono, print ..... BU1  
Mini-DIP-Schalter, 4polig,  
liegend ..... S1  
Stiftleiste, 2 x 8polig ..... STL1  
1 Präzisions-IC-Fassung, 18polig  
1 Präzisions-IC-Fassung, 28polig  
1 Zylinderkopfschraube, M3 x 6 mm  
1 Mutter, M3

```

;*****
; PIC-Grundlagen-Demoprogramm Nr. 2   V1.0
; Datum      : 18.08.98
; Dateiname  : C54_LAUF.ASM
; Controller : PIC16C54A
;*****

LIST p=16C54A ; PIC16C54A Controller

#include <p16c5x.inc>

UNI1    equ    8           ; Adresse des UNI1-Registers
UNI2    equ    9           ; Adresse des UNI2-Registers

;*****
;                               Programmablauf
;*****
org      1FFh              ; Speicherende (Startposition nach RESET)
goto    start              ; Sprung zum Programmanfang

start    org      000h      ; Beginn des Hauptprogramms
         clrw                    ; W-Register löschen
         option                ; OPTION-Register löschen
         tris    PORTB         ; alle Pins von PORT B als Ausgänge
         movlw   0FEh          ; (=1111110b) in W-Register laden
         movwf  PORTB          ; W-Register nach Port B kopieren

loop     ; Anfang der Programmschleife
         ; Wartezeit bis zum Weiterschalten der LEDs
         movf   PORTA,0        ; PORTA nach W kopieren
         movwf  UNI1           ; W —> UNI1
         swapf  UNI1,1         ; High- und Low-Nibble tauschen
         movlw  0F0h           ; Maske (11110000)
         andwf  UNI1,1         ; Low-Nibble von UNI1 löschen
         incf   UNI1,1         ; UNI1 = UNI1 + 1

warte1   movlw   0FFh          ;
         movwf  UNI2           ; UNI2 = FFh

warte2   decfsz  UNI2,1        ; UNI2 -1 bis 0 dann springen
         goto   warte2
         decfsz  UNI1,1        ; UNI1 -1 bis 0 dann springen
         goto   warte1

         ; nächste LED
         bsf    STATUS,0       ; C-Flag setzen
         rlf    PORTB,1        ; PORTB eine Stelle nach links schieben
         btfsc  STATUS,0       ; C-Flag gelöscht —> Befehl überspringen
         goto   loop
         ; Durchlauf beendet —> wieder bei D0 beginnen
         movlw  0FEh          ; (=1111110b) in W-Register laden
         movwf  PORTB          ; W-Register nach Port B kopieren
         goto   loop           ; Sprung zum Schleifenanfang

END      ; Programmende

```

durch die Stellung des Schalters S 1 (an den Ports RA0 bis RA3) bestimmt ist, wechselt die Ausgabe, wobei die LED erlischt und die nächste aufleuchtet. Nach einem Durchlauf soll das Lauflicht wieder bei der ersten LED beginnen.

Das Erstellen eines Projektes und die Bedienung des Microchip-Entwicklungssystem wurden im vorherigen Artikel be-

reits ausführlich beschrieben. Da der abgebildete Quellcode ausreichend dokumentiert ist, bedarf dieser keiner weiteren Erklärungen.

Im nächsten Teil der Artikelserie stellen wir den PIC 18F84 vor, der über einen Flash-Programmspeicher verfügt, dessen Programmspeicher ohne Einsatz eines UV-Löschgerätes wieder überschreibbar ist. **ELV**