

# Aufgaben

## A Bearbeiten Sie zwei der folgenden fünf Aufgaben:

### 1. Drehzahlmesser

Eine Fotodiode (z. B. an Port D.6) misst die Anzahl der Verdunklungen für ein Zeitintervall von 10 Sekunden. Diese Anzahl wird anschließend mit 6 multipliziert und über Leuchtdioden an Port B oder über die serielle Schnittstelle an einem Terminal ausgegeben. Benutzen Sie den Timer/Counter0.

### 2. Frequenzgenerator

Programmieren Sie einen Frequenzgenerator mit dem Timer/Counter0. Er soll eine Frequenz von 880 Hz. bzw. 12 kHz erzeugen. Überlegen Sie sich für beide Fälle einen geeigneten Prescale-Wert und berechnen Sie daraus den passenden Preset-Wert. Überprüfen Sie Ihr Ergebnis mit einem Frequenzmessgerät (Moderator ggf. danach fragen!).

Sollten Sie Probleme bei der Berechnung des Preset-Wertes haben, benutzen Sie einfach die EXCEL-Tabelle ...

### 3. Dimmer (fest eingestellt)

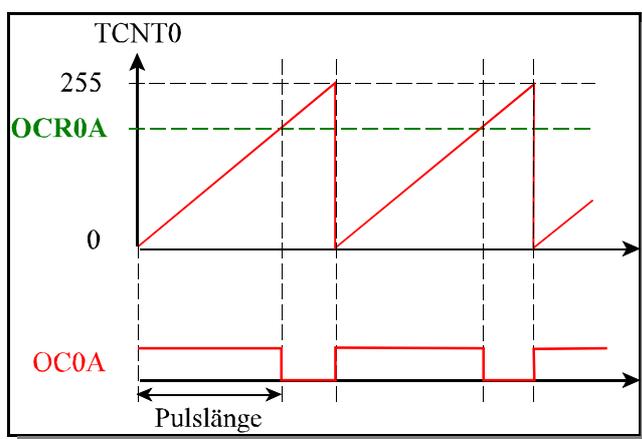


Abbildung 1

Schließen Sie zunächst eine LED am PortB.2 an. Laden Sie dann das Programm "dimmer0.bas" aus dem Verzeichnis PWM. Ergänzen Sie nun die fehlenden Stellen des Programms so, dass

- die Pulse am Ausgang OC0A wie in Abb. 1 (und nicht umgekehrt) gebildet werden,
- der Prescale 8 beträgt,
- die Pulslänge 70 Timer-Takte beträgt.

Berechnen Sie auch, wie lang (in Millisekunden) die An- und Auszeit der LED jeweils sind.

Testen Sie auch andere Pulslängen.

#### 4. Dimmer (per Tastendruck)

Über die Taster T0 und T1 soll die PWM-Pulsweite an OC0A vergrößert bzw. verkleinert werden. Benutzen Sie als Prescale den Wert 8. Die Veränderung des OCR0A-Wertes durch die Tasten kann im Rahmen einer Do-Loop-Schleife im Hauptprogramm oder durch Interrupts geschehen. Ggf. kann auf das (unvollständige) Programm “dimmer0.bas” im Verzeichnis PWM zurückgegriffen werden.

#### 5. Dimmen (automatisch)

Zunächst soll die LED an PortB.2 hell leuchten. Nach (einmaligem) Betätigen der Taste T0 soll sie langsam erlöschen. Das Programm soll Fast-PWM benutzen. Ggf. Kann auf das (unvollständige) Programm “dimmer0.bas” im Verzeichnis PWM zurückgegriffen werden.

...

## B Bearbeiten Sie die folgenden Aufgaben in der angegebenen Reihenfolge

#### 6. D/A-Wandler mit PWM

Bauen Sie den Tiefpassfilter (Abb. 2) auf Ihrer Platine auf. Benutzen Sie  $R = 1,5 \text{ k}\Omega$  und  $C = 10 \text{ }\mu\text{F}$  (Polung beachten: Minus an Masse!).

Öffnen Sie die Datei “DA-Wandlung mit FastPWM.bas”, welche PWM-Signale am Pin B2 ausgibt; die Pulsweite lässt sich mit dem Terminal über die COM-Schnittstelle einstellen.

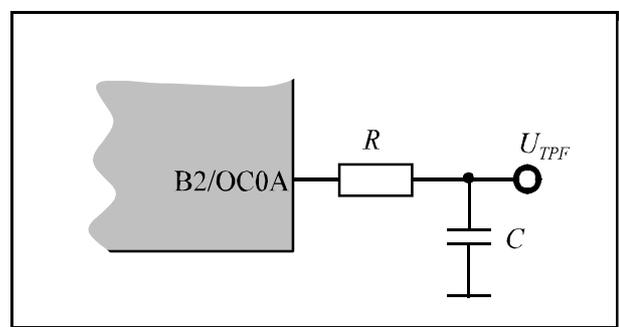


Abbildung 2

Überprüfen Sie, ob die Ausgangsspannung  $U_{TPF}$  am Tiefpassfilter und die Pulslänge proportional zueinander sind. Legen Sie dazu eine Messtabelle an und stellen Sie diese mithilfe eines Tabellenkalkulationsprogramms grafisch dar.

## 7. Spannungen vergleichen

Der Attiny soll die Spannung  $U_{TPF}$  am Ausgang des Tiefpassfilters (vgl. Aufgabe 6) mit der Spannung am Drehpoti ( $U_{var}$  auf dem Lageplan) vergleichen: Wenn  $U_{TPF}$  größer als  $U_{var}$  ist, dann soll eine LED an PortB.7 leuchten, sonst nicht.

Zu Beginn soll das Programm vom Terminal eine Zahl zwischen 0 und 255 entgegennehmen und am Ausgang  $U_{TPF}$  eine entsprechende Gleichspannung erzeugen.

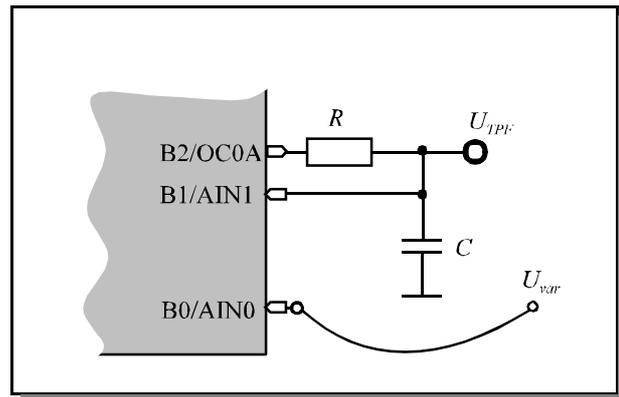


Abbildung 3

Die Spannung  $U_{var}$  wird mit dem Drehpoti eingestellt (zwischen 0 und 5 Volt).

Um die beiden Spannungen zu vergleichen, verbinden wir den Anschluss  $U_{var}$  mit B0 und den Anschluss  $U_{TPF}$  mit B1 (vgl. Abb. 3). Diese beiden Pins dienen dem Attiny nämlich auch als Analogeingänge AIN1 und AIN0. Die beiden Spannungswerte werden vom Attiny-**Komparator** fortwährend verglichen: Wenn  $AIN0 > AIN1$ , dann wird das Bit ACO im Register ACSR auf 1 gesetzt, sonst auf 0; dies geschieht fortwährend, ohne dass eine weitere Programmierung dafür erforderlich wäre. (Vgl. auch den Informationskasten auf der nächsten Seite!)

Um die Aufgabe zu lösen, muss jetzt nur dieses ACO-Bit fortwährend auf PortB.7 ausgegeben werden.

Nicht vergessen: Datenrichtungsbits für PortB.0 und PortB.1 auf 0 setzen!!!

### **Und so testen Sie Ihr Programm:**

Drehen Sie zunächst das Poti ganz nach links. Starten Sie dann das Attiny-Programm und geben sie über das Terminal eine Zahl ein (z. B. 120, das entspricht ungefähr 2,4 V am Ausgang des Tiefpassfilters).

Drehen Sie nun das Poti *langsam* nach rechts, bis die LED an Port B.7 aufleuchtet. Messen Sie nun die Spannung  $U_{var}$  am Poti und vergleichen Sie diesen Wert mit der Spannung  $U_{TPF}$ .

### 8a. Spannungen messen mit dem Komparator

(alternativ können Sie auch direkt Aufgabe 8b bearbeiten)

Das Programm aus Aufgabe 7 kann benutzt werden, um eine unbekannte Spannung  $U_{var}$  am Poti in einem try-and-error-Verfahren zu messen. Überlegen Sie sich, wie man dazu am besten vorgeht. Testen Sie Ihr Verfahren aus.

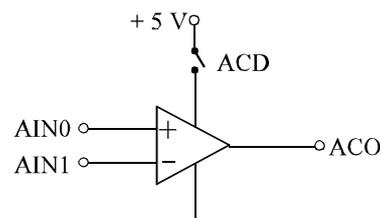
### 8b. A/D-Wandler

Der Attiny soll nun die Spannung  $U_{var}$  am Poti “messen” und über das Terminal ausgeben. Das Messergebnis soll als Zahl zwischen 0 und 255 gemäß der Kalibrierkurve aus Aufgabe 5 erfolgen: Die Spannung von 2,5 V soll also z. B. als Zahl 127 dargestellt werden.

### Informationen zum Komparator

Der Komparator besteht im Wesentlichen aus einem Operationsverstärker. Dieser kann zum Zwecke des Energiesparens mit der Anweisung  $ACSR.ACD = 0$  ausgeschaltet werden. Standardmäßig ist er aber eingeschaltet.

Ein vereinfachtes Schaltbild ist rechts dargestellt. Ausgelassen sind hier die Interruptmöglichkeiten des Komparators. Der Ausgang ACO des Komparators wirkt unmittelbar auf das Bit  $ACSR.ACO$ . Dieses Bit ist gleich 1, wenn  $AIN0 > AIN1$ , und sonst 0.



Bit	7	6	5	4	3	2	1	0	ACSR
	ACD	ACBG	ACO	ACI	ACIE	ACIC	ACIS1	ACIS0	
Read/Write	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	N/A	0	0	0	0	0	