

Erstes Programmieren

Heute programmierst Du zum ersten Mal den Attiny2313 selbst. Befolge dazu die Anweisungen ganz genau. Beachte: Auch die Reihenfolge ist wichtig!

1. Doppelblinklicht

Zwei rote Leuchtdioden sollen abwechselnd im Sekundentakt an- und ausgehen.

1.0 Vorbereitung: Kopiere die Vorlagendatei "vorlage.bas" in Dein Verzeichnis.

1.1 Starte das Programm BASCOM, öffne die Vorlage-Datei und speichere sie sofort unter dem Namen "wechselblinken.bas" in einem neuen Ordner "Wechselblinken" ab.

Ergänze die folgenden Programmzeilen und trage sie in die Rubrik "Hauptprogramm" ein:

```
do
  Portb.0 = 1
  Portb.1 = 0
  Wait 1
  Portb.0 = 0
  _____
  _____
  _____
```

1.2 Speichere das Programm ab und kompiliere es (vgl. Kurzanleitung!)

Falls BASCOM einen Fehler meldet, klicke auf die Fehlermeldung; Bascom springt dann automatisch in die Programmzeile mit dem Fehler. Verbessere den Fehler und kompiliere erneut ...
... bis das Programm fehlerfrei ist.

1.3 Stecke nun zwei LEDs in die Port-Anschlüsse PortB.0 und PortB.1 der Buchsenleiste (Die LEDs befinden sich in dem Plastik-Tütchen!). Achte dabei auf korrekte Polung! Schließe die Platine nun über das USB-Kabel an den PC an. Jetzt sollte auf der Platine eine grüne LED aufleuchten (LED 1).

1.4 Jetzt bei BASCOM die Schaltfläche zum Uploaden betätigen (vgl. Kurzanleitung!); das Upload-Programm startet. Nun die richtige COM-Nr. einstellen. Achtung: Ob Deine Wahl richtig war, entscheidet sich beim folgenden Punkt 1.5. Das Upload-Programm merkt sich immer die letzte Einstellung ; sie muss also nicht jedesmal vorgenommen werden.

1.5 Jetzt die Schaltfläche "Übertragen vorbereiten" betätigen;
danach erst den RESET-Taster betätigen.
Als Letztes die Schaltfläche "Alles übertragen" betätigen.
Nun sollte ein Fortschrittsbalken den Uploadvorgang anzeigen. Falls eine Fehlermeldung erscheint, solltest Du den Punkt 1.4 mit einer anderen COM-Nr. wiederholen.

1.6 Blinken die LEDs richtig? Prüfe ggf. nach:

- ☞ Polung
- ☞ Stecken die LEDs wirklich bei Portb.0 und PortB.1 und nicht etwa bei Portb.6 und PortB.7 oder sogar bei GND bzw. +5V?

Erstes Programmieren

- 1.7 SchlieÙe das Upload-Programm. Ziehe das USB-Kabel von der Attiny-Platine. Dadurch wird sie von der Stromzufuhr abgeschnitten; dies kannst Du auch an der grünen Kontroll-LED überprüfen. SchlieÙe nun das USB-Kabel wieder an. Hat der Mikrocontroller das Programm behalten?
- 1.8 Ändere nun das Programm so ab, dass die LEDs im 100ms-Takt blinken. Teste es aus.
- 1.9 Ändere das Programm so ab, dass die LEDs im 10ms-Takt blinken. Teste es aus. Was fällt auf?

2. Eine Ampelanlage

Nun sollst Du mit einer roten, einer gelben und einer grünen LED eine Verkehrsampel bauen und programmieren.

Beachte: Ziehe vor jedem Umbau das USB-Kabel von der Attiny-Platine!

- 2.1 Baue die rote LED bei PortB.2, die gelbe bei PortB.1 und die grüne bei PortB.0 ein.
- 2.2 Lege im Heft eine Tabelle für die Ampelphasen an:

Phase	es leuchten die LEDs	Dauer in Sekunden
1	rot	5
2	rot, gelb	1
...

- 2.3 Schreibe jetzt mithilfe der Vorlage-Datei das Programm Ampel.bas und speichere es in einem eigenen Verzeichnis ab.
- 2.4 Kompilieren, hochladen, testen... Übertrage Dein korrektes Programm in Dein Heft.

3. Ein Lauflicht (wenn noch genügend Zeit vorhanden...)

- 3.1 Stecke 8 rote LEDs in die Anschlüsse von PortB.0 bis PortB.7.
- 3.2 Schreibe ein Programm lauflicht.bas, welches die LEDs in rascher Folge (100ms-Abstand) der Reihe nach einschaltet, dann alle (fast) gleichzeitig ausschaltet und dann wieder von vorne anfängt. Teste es auch aus.

4. Abbau

- 4.1 USB-Kabel von der Attiny-Platine ziehen
- 4.2 LEDs in das Plastiktütchen legen; das Tütchen zusammenrollen, so dass nichts herausfallen kann!
- 4.3 Platine, Tütchen und USB-COM-Kabel in die Box legen!
- 4.4 Box sorgfältig verschließen und in den Karton legen!