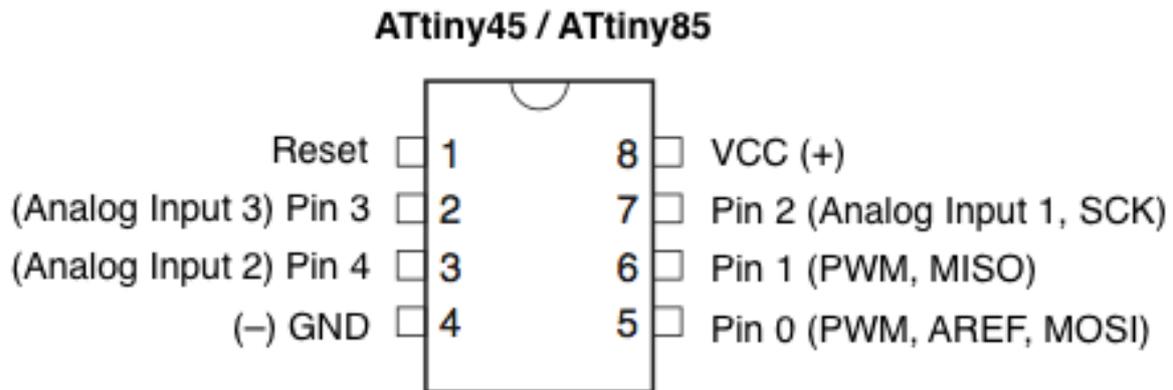


## Anleitung: ATtiny 45 mit Arduino Nano bespielen

- Arduino IDE öffnen und das Programm, was später auf dem Tiny laufen soll, schreiben.  
Dabei ist der Nano im Drop-Down ausgewählt, mit dem der Tiny später beschrieben werden soll.  
Beim Schreiben des Programms drauf achten, dass die PIN-Belegung auf den Tiny ausgelegt ist.  
Dazu am besten einen PIN-Out Schaltplan anschauen.



- Hier ein Beispiel für eine blinkende LED.  
Die LED wird am physischen Pin3 vom Tiny angeschlossen, ist aber im PIN-Out der Pin 4.

```
const int ledPin = 4;

void setup() {
  pinMode(ledPin, OUTPUT);
}

void loop() {
  digitalWrite(ledPin, HIGH);
  delay(2000);
  digitalWrite(ledPin, LOW);
  delay(2000);
}
```

- Wenn das Programm fertig ist, im oberen Menü auf **Datei** klicken.  
Im Menü selber den Punkt **Beispiele** (es öffnet sich ein weiteres Menüfeld). In diesem Menüfeld auf **ArduinoISP** mit der Maus, damit sich das nächste Feld öffnet, das auch **ArduinoISP** heißt. Da anklicken.  
Es öffnet sich ein ganz neues Fenster.
- Oben rechts in dem Fenster den Monitor einblenden (ist der Kreis mit Punkt in der Mitte). Geht man mit der Maus über das Symbol, erscheint auch die Texteinblendung **Serieller Monitor**. Da klicken, damit im unteren Bereich vom offenen Fenster ein abgetrennter Bereich erscheint. 9600Baud sollten im Drop-Down schon drinstehen.

- Ganz oben im Menü noch mal auf **Werkzeuge** klicken und prüfen, ob der Arduino Nano weiter unten im offenen Menü zu lesen ist (**Board: Arduino**)

Wenn nicht, kann er unter dem Menüpunkt Boardverwaltung ausgewählt werden. Er sollte da aber stehen, da er vor der Programmierung bereits ausgewählt wurde.

- Bevor das Programm überspielen wird, darauf achten, dass der Transistor aus dem Steckboard rausgenommen ist.
- Nun den Pfeil zum Überspielen drücken.  
Damit wird der Bootloader auf den Arduino gespielt.  
Im seriellen Monitor, der vorhin eingeblendet wurde, ist nun der ganze Ablauf zu sehen.  
Am Ende sollte hier keine Fehlermeldung zu sehen sein.  
Wenn doch, dann liegt es vielleicht an der falschen Schnittstelle – USB.
- Das ganze Fenster kann nun geschlossen werden, damit nur wieder das Fenster mit unserem Sketch zu sehen ist.
- Da der Bootloader auf dem Arduino ist, braucht er nicht auf dem Tiny überspielt werden, was Speicher spart.
- Nun im **Dropdown**, wo noch der **Arduino Nano** steht, den **Tiny** auswählen.  
Dazu auf den kleinen **Pfeil** klicken und dann auf den **Stift**, der rechts neben Nano steht, wo auch die Verbindungsschnittstelle drunter steht.

Im Fenster, was aufgeht, **45** eingeben.

Dann erscheinen ein paar Auswahlfelder, wo der Eintrag mit **25/45/85** ausgewählt wird. Rechts davon wieder USB auswählen. Bestätigen mit OK.

- Nun wieder ganz oben im Menü auf **Werkzeug** klicken. Da sollten nun die Daten vom Tiny aufgeführt sein. Da sollte auch **Clock: 1MHz internal** stehen. Wenn nicht, das Menü dazu öffnen und auswählen.
- Ein Stück weiter unten davon steht **Programmer: „Arduino as ISP“**. Wenn nicht, dann im dazugehörigen Menü auswählen.
- Bevor nun das Programm überspielt wird, muss noch der Kondensator an seinem alten Platz eingesetzt werden
- Wenn das alles eingestellt ist, dann im Programmfenster den Pfeil zum Überspielen drücken. Damit überspielt der Arduino nun das Programm auf den Tiny.

Die LED sollte nun blinken, wie es im Programm geschrieben wurde. Der Tiny ist jetzt eigenständig nutzbar.