



Einleitung

Du wolltest schon immer programmieren oder mit den Bestandteilen eines Computers arbeiten? Du wirst zwar nach diesem Buch keinen Computer bauen können, aber die Einleitung erklärt einiges, was du später machen kannst:

Dies erfährst du in der Einleitung:

- ☉ Was ist ein Mikrochip, ein Mikrocontroller und was ist ein Arduino?
- ☉ Wie programmiert man?
- ☉ Was brauchst du an Materialien für dieses Buch?

Du siehst, du wirst einige Grundlagen über die Elektronik kennenlernen!

Was ist ein Mikrochip?

Sicherlich hast du schon eine Vorstellung, wie ein Mikrochip aussieht. Wie eine kleine schwarze Fläche, die auf einer Platine, zum Beispiel dem Mainboard deines Computers, angebracht ist. Der Mikrochip, den du programmieren wirst, sieht etwas anders aus als ein Chip, den du kennen könntest. Er ist rechteckig, nicht quadratisch und besitzt wesentlich weniger Pins (die Metallbeine, die an der Seite herausragen). Außerdem sind diese Pins wesentlich größer, als du sie von »normalen« Mikrochips kennen könntest.

Und was ist jetzt ein Mikrocontroller?

Ein Mikrocontroller ist ein Mikrochip, der alle benötigten Komponenten (sozusagen das Zubehör) bereits in sich trägt. Er sieht genauso aus wie ein Mikrochip. Du kannst es erneut mit dem Mainboard eines Computer vergleichen! Dann wäre das Mainboard der Mikrocontroller, und der Arbeitsspeicher wäre dann ein Teil des Zubehörs. Der Arbeitsspeicher ist ebenfalls in dem Mikrocontroller integriert.



Merke dir: Wenn in diesem Buch in Zukunft über einen Mikrochip geschrieben wird, ist eigentlich der Mikrocontroller gemeint! Und zur Rechtschreibung: Man kann sowohl Microcontroller mit c als auch mit k (Mikrocontroller) schreiben.

Und was ist dann ein Arduino?

Damit du den Mikrocontroller am Anfang leichter programmieren kannst, gibt es das sogenannte Arduino-Projekt. Beim Arduino-Projekt gibt es fertige Platinen mit dem Mikrocontroller und eigener Software zum Programmieren. Die Platine wird dabei einfach als Arduino bezeichnet. In diesem Buch geht es, wie der Titel schon verrät, um die Programmierung eines Mikrochips auf der Arduino-Platine.

Das Besondere ist dabei, dass das Arduino-Projekt eine fertige Platine mit der passenden Entwicklungsumgebung für einen PC bereitstellt. Frühere Entwickler mussten sich teilweise sogar die Boards selber basteln, damit sie überhaupt lernen konnten, wie man programmiert.

Wie programmiert man?

Programmieren kann man leider nicht durch Reden, sondern nur über Texte am Computer. Diese Texte werden zudem nicht in Deutsch verfasst, sondern die wenigen Wörter sind auch auf Englisch. Das sollte dich aber nicht abhalten, programmieren zu lernen! Du lernst programmieren mit der Programmiersprache C++, einer sehr systemnahen und systematischen, aber auch komplexen Programmiersprache. Um es am Anfang nicht viel zu schwer zu machen, haben die Arduino-Macher einen einfacheren Dialekt (= eine Variation der Programmiersprache) entwickelt. C++ basiert auf wenigen Wörtern, dafür auf vielen teilweise sehr kryptisch wirkenden Zeichen. In den Klammern steht jeweils, wie man dieses Zeichen »aussprechen« würde: zum Beispiel ++ (Inkrement), % (Modulo Division).

Wenn du erst einmal weißt, was die einzelnen Symbole bedeuten, wirst du sie fehlerfrei anwenden können. Mit folgendem Quelltext zeige ich dir, was mit C++ möglich ist. Diesen Quellcode habe ich für einen selbst gebauten Terrariencomputer entwickelt, der leider noch im Anfangsstadium ist:

```
#include "dimmen.h"
#include "kern_temperatur.h"
#include "terra_temperatur_class.h"
#include "class_cool.h"
void setup() {
  pinMode(13, OUTPUT); pinMode(12, OUTPUT);
  Serial.begin(9600); ADMUX = 0xC8; delay(10);
}

void loop() {
  Cooler cooler(13); delay(100); bool hot = kern_temp(17);
  if (hot) cooler.start();
  else{
    cooler.stop();
  }
  delay(500); }
```

Zum Programmieren brauchst du allerdings noch Software für den PC, die du unter *arduino.cc* im Internet findest. Zur genauen Installation schau bitte in den Anhang oder in das nächste Kapitel.

Nun sollst du aber etwas über das Gerät lernen, das wir für das Buch benötigen, den Arduino. Der Arduino, den du vermutlich vor dir hast, ist ein Arduino Uno. Du solltest den Arduino erst einmal über USB mit dem Computer verbinden. Du kannst ihn zwar noch nicht programmieren, aber du kannst ihn schon als Stromquelle für die nächsten Schaltungen benutzen.

Materialien

Um jetzt in der Einführung weiterzumachen, brauchst du bereits einige Materialien. Dazu gehören eine LED (eine Leuchtdiode, also ein Bauteil, das Licht abstrahlt), ein schwarzes und ein graues Kabel (oft werden von Arduino-Nutzern schwarz-rote Kabel als Farbkombination benutzt, im Buch benutzen wir wegen der schwarz-weißen Schaltpläne schwarz-grau zur besseren Erkennbarkeit), der Arduino und ein sogenanntes Breadboard.

Sämtliche benötigten Materialien sind übrigens im Anhang C am Ende des Buches aufgelistet.

