

Teil 1

Lektion

2

**Erster
Blick**

Arduino

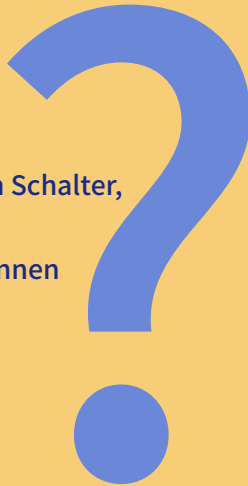
1. Was ist ein Arduino?

Arduino basiert auf der Idee, benutzerfreundliche Hardware und Software zu implementieren und ist eine Open-Source-Elektronikplattform.

Arduinos können Eingangssignale lesen:

- ☐ Beschleunigung und Lenkung,
- ☐ ein gedrückter Finger auf einem Schalter,
- ☐ eine Twitter-Nachricht senden,
- ☐ Licht auf einem Sensor zu erkennen

.....



und verwandelt es in ein Ausgangssignal:

- ☐ um einen Motor zu aktivieren
- ☐ eine LED einzuschalten
- ☐ etwas online veröffentlichen.....

Sie können Ihrem Board mitteilen, was zu tun ist, indem Sie eine Reihe von Anweisungen an den Mikrocontroller auf dem Board senden. Dazu verwenden Sie die Programmiersprache Arduino (basierend auf Wiring) und die Arduino Software (IDE) basierend auf Processing.

Eine weltweite Community von Machern - Studenten, Hobbyisten, Künstler, Programmierer und Profis - hat sich um diese Open-Source-Plattform versammelt. Ihre Beiträge haben zu einer unglaublichen Menge an zugänglichem Wissen geführt, das sowohl Anfängern als auch Experten eine große Hilfe sein kann.

Möchten Sie einen Roboter bauen, der laufen, tanzen und von Ihnen gesteuert werden kann?

Möchten Sie ein intelligentes Armband herstellen, das Ihre Herzfrequenz, Schrittnummer und die Uhrzeit, das Datum und den Song jederzeit wiedergibt?

Denken Sie, Sie können Ihre Einrichtung zu Hause auch aus der Firma heraus steuern?

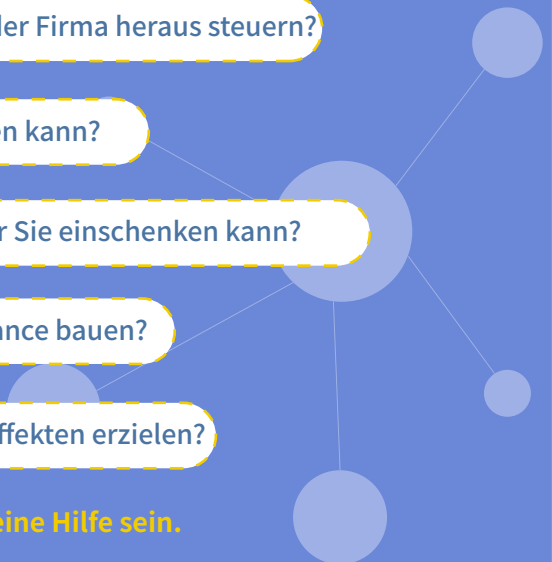
Möchten Sie ein Roboter bauen, der für Sie den Boden kehren kann?

Möchten Sie ein Gerät herstellen, das automatisch Kaffee für Sie einschenken kann?

Möchten Sie einen Handschuh wie den aus der Avenger Alliance bauen?

Möchten Sie alle Arten von coolen und romantischen Lichteffekten erzielen?

Bei all diesen Aufgabenstellungen kann Ihnen der Arduino eine Hilfe sein.



2. die Historie des Arduinos

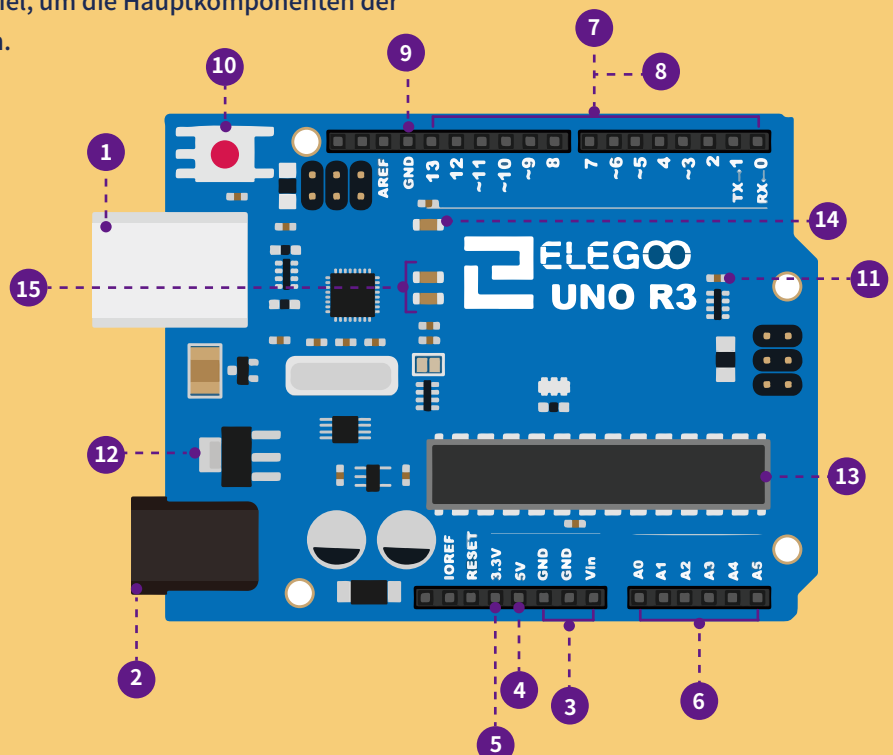
- Als Open-Source-Projekt, das von Massimo Banzi, David Cuartielles, Tom Igoe, Gianluca Martino und David Mellis gegründet wurde, war Arduino das Gehirn von Tausenden von Projekten, von Alltagsgegenständen bis hin zu komplexen wissenschaftlichen Instrumenten.
- Arduino wurde ursprünglich am Interaction Design Institute Ivrea in Norditalien entwickelt. Es stammt von Wiring, einer Plattform, die Hernando Barragan als Masterarbeit bei Interaction-Ivrea gebaut hat. Hernando wurde von Massimo und Casey Reas beraten. Wiring und Arduino bauen auf früheren Arbeiten von Massimo und Casey auf - Massimos Programm2003-Prototyping-Plattform für Elektronik und der Processing-Plattform von Casey und Ben Fry. Frühere Versionen von Wiring und Arduino stützten sich auch auf die avr-lib-Bibliotheken von Pascal Stang.
- Die ersten Arduino-Boards wurden von Massimo Banzi und David Cuartielles entworfen. David Mellis entwickelte die erste Arduino-Software, die auf Wiring basiert, mit vielen Beiträgen von Nicholas Zambetti. Tom Igoe von ITP in New York war ein Early Adopter und Berater des Arduino-Projekts. Gianluca Martino half bei der Herstellung und beim Hardware-Design. Das Arduino-Projekt wird jetzt von einem internationalen Unternehmen mit Büros und Mitarbeitern auf der ganzen Welt unterstützt.

Arduino hat seinen Namen von einer Bar in Ivrea, die nach einem frühen König von Italien aus Ivrea benannt wurde.

3. Eine Einführung zum Board

Wir verwenden Elegoo Uno als Beispiel, um die Hauptkomponenten der Arduino-Platine zu veranschaulichen.

- ① the USB connection
- ② the 6V~12V barrel jack
- ③ GND
- ④ 5V pin supplies
- ⑤ 3.3V pin supplies
- ⑥ analog pin
- ⑦ digital pin
- ⑧ (pwm~)
- ⑨ GND
- ⑩ Reset Button
- ⑪ Power LED Indicator
- ⑫ Voltage Regulator
- ⑬ Main IC
- ⑭ Pin D13 indicator LED
- ⑮ Serial communication indicator LEDs



Stromversorgung (USB / Barrel Jack)

- Jedes Arduino-Board benötigt eine Möglichkeit, an eine Stromquelle angeschlossen zu werden. Der Arduino UNO kann über ein USB-Kabel von Ihrem Computer oder ein Netz-Adapter mit Strom versorgt werden, das in einer Buchse angeschlossen ist. In der Abbildung oben ist die USB-Verbindung mit (1) und die Laufbuchse mit (2) gekennzeichnet.
- Über die USB-Verbindung können Sie auch Code auf Ihr Arduino-Board laden. Weitere Informationen zum Programmieren mit Arduino finden Sie in unserem Tutorial zum Installieren und Programmieren von Arduino.

HINWEIS: Verwenden Sie KEIN Netzteil mit mehr als 20 Volt, da Ihr Arduino diese Spannung nicht verträgt (und dadurch zerstört wird). Die empfohlene Spannung für die meisten Arduino-Modelle liegt zwischen 6 und 12 Volt.

Zu 3.) bis 9.) PINs (5 V, 3,3 V, GND, Analog, Digital, PWM, AREF)

- Die Stifte an Ihrem Arduino sind die Stellen, an denen Sie Drähte anschließen, um eine Schaltung aufzubauen (wahrscheinlich in Verbindung mit einem Steckbrett und etwas Draht. Sie haben normalerweise schwarze Kunststoff-Header, mit denen Sie einfach einen Draht direkt in das Board einstecken können. Der Arduino hat verschiedene Arten von Stiften, von denen jeder auf der Platine beschriftet ist und für verschiedene Funktionen verwendet wird.

GND (3): Abkürzung für „Ground“. Auf dem Arduino befinden sich mehrere GND-Pins, von denen jeder zur Erdung Ihres Stromkreises verwendet werden kann.

5V (4) & 3.3V (5): Wie Sie sich vorstellen können, liefert der 5V-PIN 5 Volt und der 3,3V-PIN 3,3 Volt. Die meisten einfachen Komponenten, die mit dem Arduino verwendet werden, werden problemlos mit 5 oder 3,3 Volt betrieben.

Analog (6): Der Bereich der PINs unter dem Etikett „Analog In“ (A0 bis A5 auf der UNO) sind Analog In-PINs. Diese PINs können das Signal von einem analogen Sensor (wie einem Temperatursensor) lesen und in einen digitalen Wert umwandeln.

Digital (7): Gegenüber den analogen Pins befinden sich die digitalen Pins (0 bis 13 bei der UNO). Diese Pins können sowohl für den digitalen Eingang (z. B. feststellen, ob eine Taste gedrückt wird) als auch für den digitalen Ausgang (z. B. dem Ansteuern einer LED) verwendet werden.

PWM (8): Möglicherweise haben Sie die Tilde (~) neben einigen der digitalen Pins (3, 5, 6, 9, 10 und 11 auf der UNO) bemerkt. Diese PINs fungieren als normale digitale PINs, können jedoch auch für die sogenannte Pulsweitenmodulation (PWM) verwendet werden. Wir haben ein Tutorial zu PWM, aber stellen Sie sich diese PINs vorerst so vor, dass sie den analogen Ausgang simulieren können (wie das Ein- und Ausblenden einer LED).

AREF (9): Steht für eine analoge Referenz. Er wird manchmal verwendet, um eine externe Referenzspannung (zwischen 0 und 5 Volt) als Obergrenze für die analogen Eingangs-PINs festzulegen.

Wenn Sie im folgenden Kurs über den Code verwirrt sind, können Sie die STRG-Taste gedrückt halten und auf die folgende Website klicken, um eine Hilfestellung zum Befehl/Syntax zu erhalten.

<https://www.arduino.cc/reference/en/#functions>