

Workshop: LED-Würfel mit Arduino

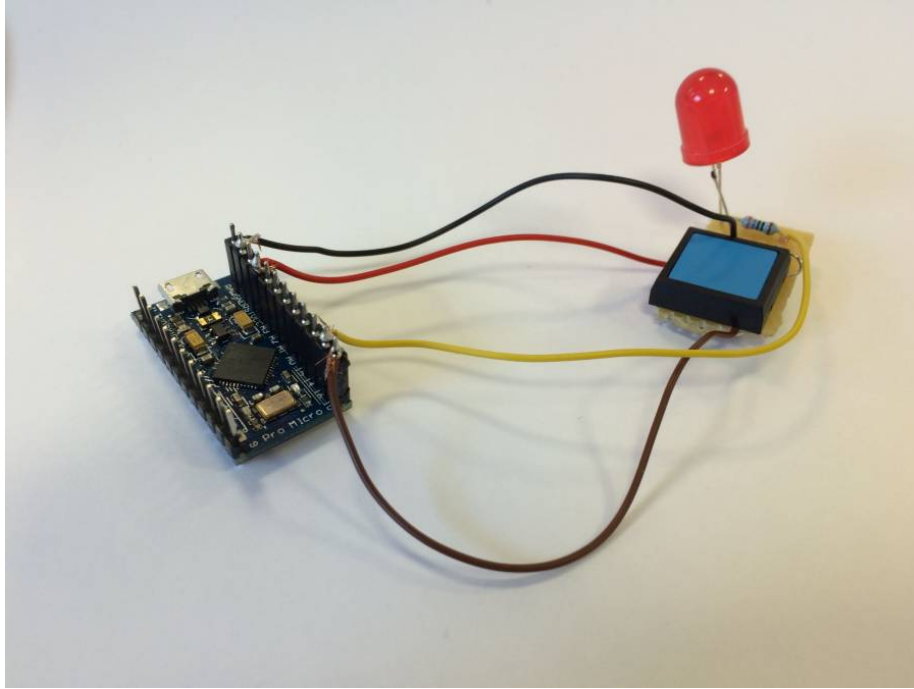


Figure 1: arduinobeispiel

Arduinos¹ sind kleine Computer in verschiedenen Größen, Formen und Farben. Sie sehen etwas anders aus als die Laptops, Smartphones, Tablets und andere Computer, die man aus dem Alltag kennt: Arduinos haben keine Tastaturen, keinen Bildschirm, kein Gehäuse und so weiter - es handelt sich also nur um den “nackten” Computer. Dafür sind sie sehr viel günstiger als “große” Computer (ein Arduino kostet nur ein paar Euro) und verbrauchen sehr, sehr wenig Strom. Anders als z.B. dein Smartphone haben Arduinos erstmal keine besonderen Funktionen, sondern sind für Bastler gemacht und sollen helfen, spannende Projekte umzusetzen. Auf dem Foto rechts siehst du zum Beispiel ein Poster, das wir etwas interaktiver gemacht haben: Mit einem Arduino und ein paar elektronischen Bauteilen haben wir einen Touch-Sensor selbst gebaut, so dass das Poster aufleuchtet, wenn man es berührt.

Der Fantasie sind kaum Grenzen gesetzt - vom eigenen Fahrradcomputer über Fernsteuerungen für die Zimmer-Beleuchtung bis hin zu Alarmanlagen fürs Sparschwein lässt sich fast alles bauen. Tipp: Einfach einmal “arduino projekte” in YouTube eingeben!

Als erstes Projekt bauen wir einen kleinen Würfel, der aus einem Arduino, einer

¹<http://www.arduino.cc/>

LED (kann leuchten), einem Schalter und einigen Widerständen (elektronische Bauteile) besteht. Auf Knopfdruck blinkt die LED zufällig 1-6 mal und kann so als Spielwürfel verwendet werden. Wie das finale Projekt aussieht, siehst du rechts.

Vorbereitung

Wie erwähnt hat so ein Arduino keine Tastatur und keinen Bildschirm. Man benötigt also einen „richtigen“ Computer um einen Arduino zu programmieren (nach der Programmierung funktioniert er dann aber selbständig). Dazu sind folgende Schritte nötig:

- Arduino-Software herunterladen² und installieren.
- Ardublock installieren. Hierbei handelt es sich um ein Programm, das mit der Arduino-Software zusammenarbeitet und es ermöglicht, einsteigerfreundlich und grafisch zu programmieren. Die Ardublock-Website³ ist etwas unübersichtlich, aber hier⁴ gibt es eine gute Video-Anleitung. Bitte verwendet diese Version von Ardublock⁵.

Damit ist euer Computer vorbereitet und im Prinzip könnt ihr nun loslegen. Wichtig: Wenn ihr auf einem unserer Workshops wart und einen Arduino habt, der so aussieht wie auf dem Bild rechts (Arduino Pro), seid ihr doch noch nicht ganz fertig. Es handelt sich hierbei nämlich um sehr, sehr günstige Arduinos, die einen besonderen Chip für die USB-Kommunikation integriert haben. Wie die Installation funktioniert, steht hier⁶. Leider gibt es die Anleitung nur auf Englisch. Wir arbeiten daran, eine Übersetzung anzubieten - wenn bis dahin Probleme auftreten, einfach bei uns nachfragen!

Elektronisches

Damit wir in unserem ersten Projekt nicht zu viel komplizierte Dinge tun müssen, nutzen wir meist ein vorbereitetes Set, das in etwa aussieht wie auf dem Foto rechts. Hierfür sollte man wissen:

Die schwarze Stecker-Leiste kann auf die “Pins” (Die Metall-Stifte) des Arduinos gesteckt werden. Wichtig: rote und schwarze Kabel müssen in Richtung des USB-Anschlusses zeigen. Die schwarzen und roten Kabel sind für Strom. Hierbei stellen wir uns Strom erst einmal wie Wasser vor: Er “fließt” von Rot (“Plus”) zu Schwarz (“Minus”) und kann auf dem Weg verschiedene Dinge “antreiben”, z.B. etwas zum leuchten bringen. Das braune Kabel ist für den Schalter. Es geht an den mit “10” beschrifteten Pin am Arduino. Diesen verwenden wir also später um zu messen, ob der Schalter gedrückt ist oder nicht.

²<https://www.arduino.cc/en/Main/Software>

³<http://blog.ardublock.com/>

⁴https://www.youtube.com/watch?v=4Yv_CPWwxaA

⁵<http://fablab-siegen.de/>

⁶<https://learn.sparkfun.com/tutorials/pro-micro--fio-v3-hookup-guide/installing-windows>

Das gelbe Kabel ist für unsere leuchtende LED und landet später an Pin “16” des Arduino. Die ist aber noch nicht angeschlossen, was wir jetzt korrigieren: Steckt nun den Widerstand (das bläuliche Teil mit den Ringen) auf eure Loch-Platine, und zwar so, dass er mit einem Bein direkt neben dem gelben Kabel steckt. Das andere Bein steckt ihr irgendwo auf die Platine. Fehlt noch die rote LED. Diese steckt ihr als nächstes auf, und zwar so, dass das lange Bein im gleichen Loch ist wie das Bein des Widerstandes, das nicht neben dem gelben Kabel steckt. Das kürzere Bein der LED kommt in das Loch direkt neben dem schwarzen Kabel (“Minus”).

Nun müssen wir noch Verbindungen herstellen: Wir löten (unter Aufsicht!):

Langes Bein der LED und Bein des Widerstandes im gleichen Loch zusammen anlöten. Anderes Bein des Widerstandes anlöten und mit gelbem Kabel verbinden. Kurzes Bein der LED anlöten und mit schwarzem Kabel verbinden. Schwarze Steckerleiste auf die Pins des Arduino stecken (rotes und schwarzes Kabel in Richtung USB-Buchse). Nun haben wir eine LED, die über das gelbe Kabel und die Stecker-Leiste an den Pin “16” des Arduino angeschlossen werden kann. Es kann Strom über das gelbe Kabel in die LED fließen, diese „antreiben“ und über das schwarze Kabel wieder “abfließen”. Der Widerstand den wir dazwischen geschaltet haben, ist notwendig, da sonst zu viel Strom durch die LED fließen würde und sie zerstören würde. Stellt es euch vor, wie einen Wasserhahn: Wenn zu viel Wasser fließt, regelt ihr per Hand am Wasserhahn nach und stellt es auf weniger Durchfluss - so etwas ähnliches macht der Widerstand!

Der Schalter hat erstmal noch nichts mit der LED zu tun. Er nutzt zwar die gleiche Abwasserleitung”, nämlich das schwarze Kabel, aber ansonsten ist er separat. Wenn wir LED und Schalter zusammenbringen möchten, dann müssen wir das nun über den Arduino selbst und seine Software tun:

Programmierung

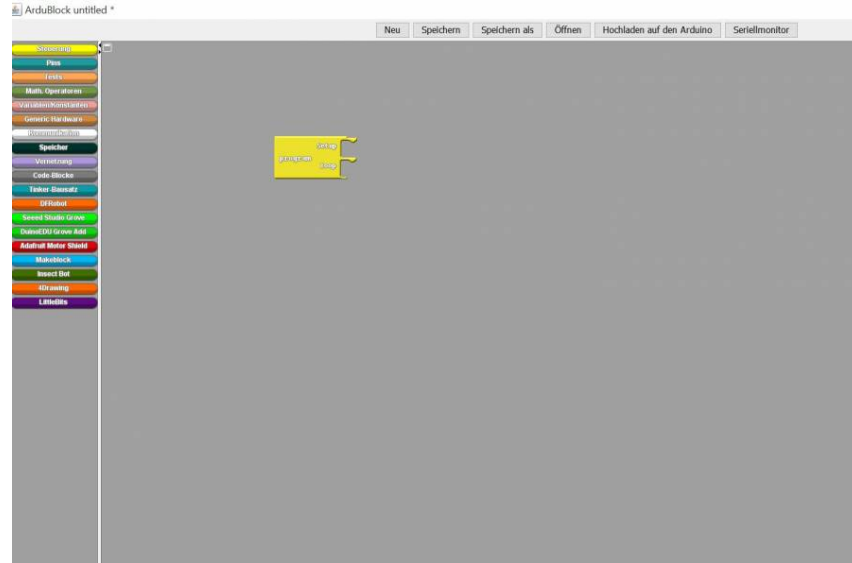
Öffnet die Arduino-Software, klickt auf „Werkzeuge“, dann auf „ArduBlock“. Ein neues Fenster öffnet sich (Achtung, beide Fenster müssen geöffnet bleiben. Bitte das Arduino-Fenster nicht schließen!). Ihr seid nun in der Ardublock-Umgebung:

Auf der linken Seite findet ihr eure Werkzeugkiste, rechts ist dann sozusagen eure Werkbank oder Arbeitsfläche. Wir gehen jetzt Schritt für Schritt vor und nähern uns unserem Würfel an:

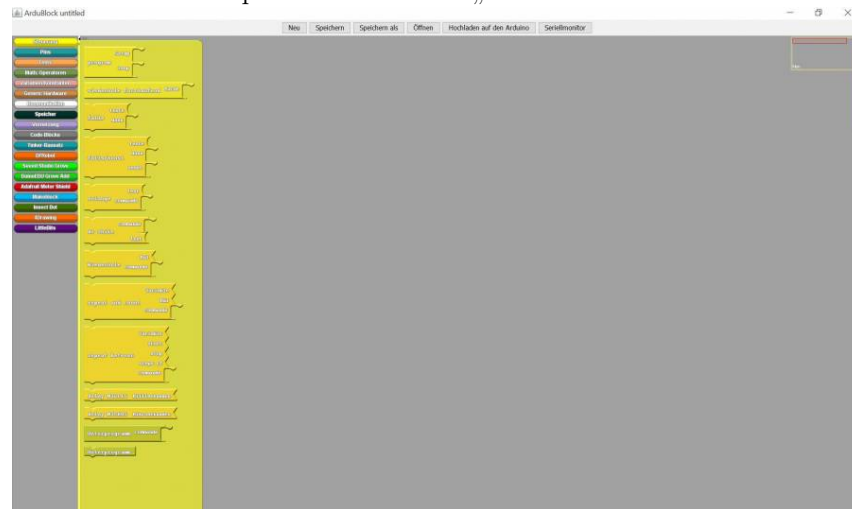
LED blinken lassen

1. Klickt auf „Steuerung“ und zieht das Werkzeug „program“ in die Arbeitsfläche rechts. Ein „program“ ist die Basis jedes Arduino-Programmes! Die „loop“, also der untere Teil des „program“ läuft später auf dem Arduino immer wieder durch - hunderte

von Malen pro Sekunde. Hier können wir also Aktionen starten:



2. Klickt auf „Pins“ in der Werkzeugkiste und zieht „Setze digital pin auf“ in die „loop“ des „program“. Hier geben wir in das rote Feld „10“ ein und lassen das untere Feld auf „HIGH“. Das klingt kompliziert, heißt aber im Prinzip nichts anderes als „lasse Strom durch Pin 10 fließen“.



3. Klickt auf „Steuerung“ und zieht „delay MILLIS Millisekunden“ ebenfalls in die „loop“. Auch das ist komplizierter ausgedrückt als nötig und heißt nichts anderes als „warte X Millisekunden“. Eine Sekunde besteht aus 1000 Millisekunden, also passiert bisher folgendes: Strom wird auf Pin 10 gegeben und anschließend wartet der Arduino eine Sekunde. Danach wird die „loop“ wieder von vorn ausgeführt.

- Um die LED blinken zu lassen, müssen wir sie wieder ausschalten. Dazu machen wir einen Rechtsklick auf das „setze digital Pin auf“, das wir erstellt haben und klicken auf „klonen“. Die nun kopieren Elemente ziehen wir in die loop unter die Wartezeit, die wir eben erstellt hatte. Nun müssen wir aus dem zweiten „HIGH“ noch ein „LOW“ machen, also den Strom auf Pin 10 ausschalten. Nun habt ihr folgendes programmiert: Eine Schleife, die immer und immer wieder durchläuft und in der Pin 10 Strom bekommt, dann wird eine Sekunde gewartet, dann wird der Strom zu Pin 10 ausgeschaltet, dann wird wieder eine Sekunde gewartet und schließlich startet die Schleife von vorn. Wie es der Zufall so will, hängt unsere LED an Pin 10, sollte also immer dann leuchten wenn Strom kommt! Auf dem Foto recht seht ihr, wie das in Ardublock aussieht.

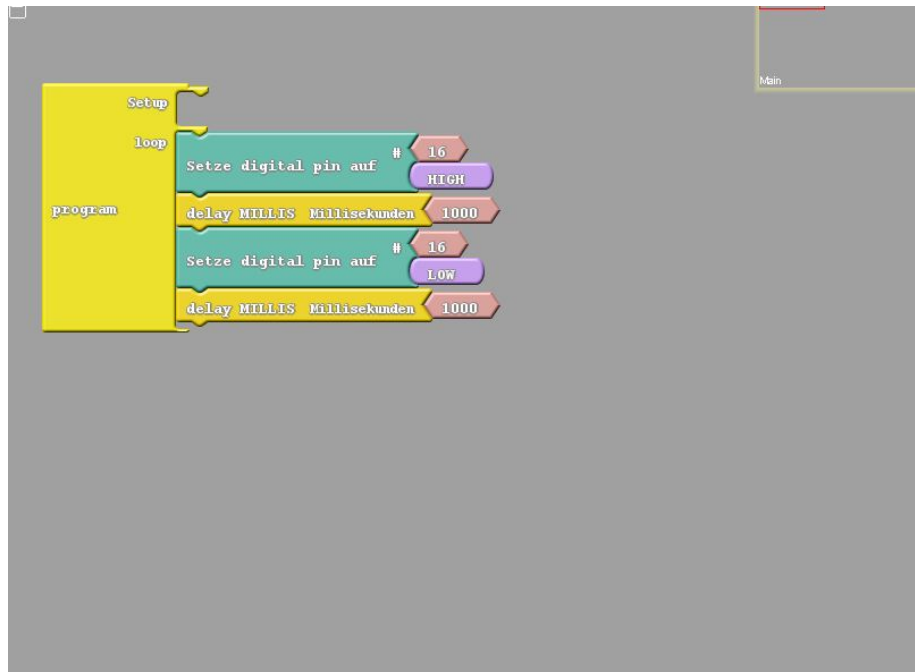


Figure 2: ardublock_blink

Das testen wir so: Wir schließen unseren Arduino per Micro-USB-Kabel an den Computer an, warten ggf. ein paar Sekunden bis der Computer ihn erkannt hat und klicken dann in Ardublock auf „Hochladen auf den Arduino“. Wenn alles geklappt hat, dauert es einige Sekunden bis euer Programm auf den Arduino geladen wurde und die LED sollte dann anfangen, zu blinken. Euer Programm ist nun auf dem Arduino und startet immer dann, wenn der Arduino Strom bekommt - ihr könnt ihn nun z.B. auch mit dem Ladegerät eures Smartphones betreiben (sofern euer Handy einen Micro-USB-Anschluss hat).

Zum Würfel ausbauen

Unsere blinkende LED ist cool, aber wir haben ja höhere Ziele - und den Schalter haben wir bisher noch gar nicht benutzt! Das ändern wir nun, diesmal allerdings selbständiger. Das Ziel ist, etwas zu bauen, das genauso aussieht wie auf dem Foto! Versucht es einfach mal und fragt uns, wenn ihr nicht weiterkommt!

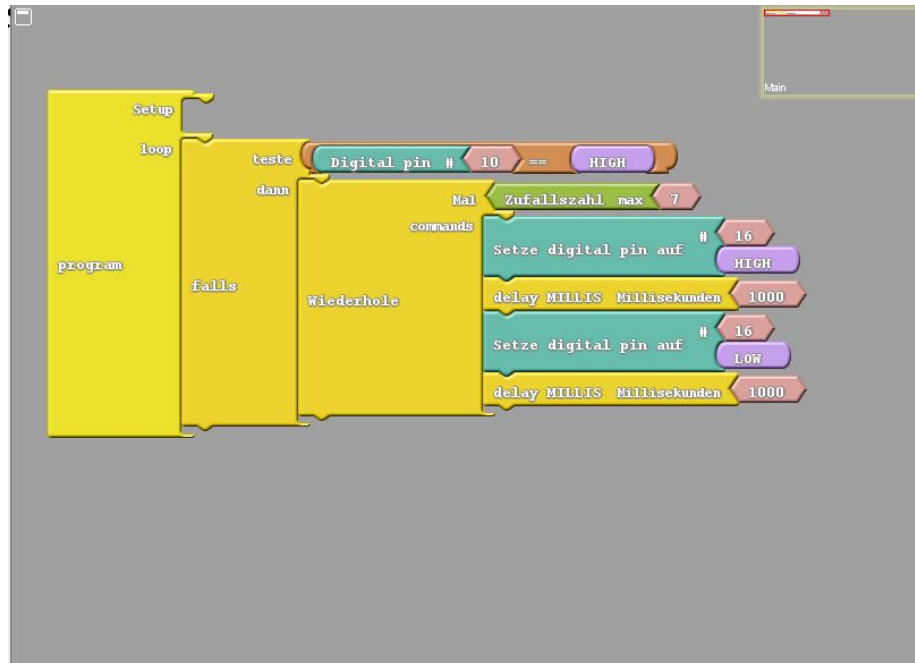


Figure 3: ardublock_wuerfel