

Informatik

EIN PROGRAMMIERTES SPIEL BAUEN

ANLEITUNG

- Sketch kopieren und auf den Arduino laden
- Schaltung auf der Steckplatine aufbauen
- Material zusammenstellen, Box bauen oder von Zuhause mitbringen
- Anordnung der Bauteile auf der Box bestimmen
- Plan zeichnen
- Vorbereiten der Bauteile, testen der verlöteten Bauteile
- Bauteile in die Box einbauen und am Arduino anschliessen
- Spiel spielen und Funktion überprüfen
- Eigene Töne programmieren und Spielgeschwindigkeit ändern

Programmierung

Vorbereitung: Pro Computer einmal ausführen: Installieren der Arduino Software (IDE), Download der Tone Library

Sketch: Sketch auf www.makinginderschule.ch kopieren, Sketch in der IDE öffnen



Abb. 292 | Sketch in der IDE

Am Schluss dieser Beschreibung wird gezeigt, wie eigene Töne programmiert und die Spielgeschwindigkeit abgeändert werden kann.

Durch klicken auf das Gutzeichen-Symbol  den Sketch kontrollieren. (Dabei wird der Sketch kompiliert, d. h., in die Maschinensprache übersetzt.)

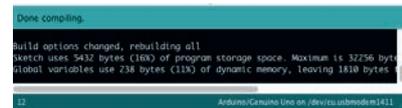


Abb. 293 | Sketch-Kontrolle

Erscheint der untere Balken grün, kann man weiterfahren, ansonsten muss der Fehler gesucht werden. (Fehler sind meist oberhalb der rot markierten Zeile zu finden.)

Häufigster Fehler: Semikolon/Strichpunkt fehlt am Ende der Zeile

Ist der Arduino mit einem USB-Kabel mit dem Computer verbunden, kann der Sketch mit dem Pfeil-Symbol  auf den Arduino kopiert werden.

Hinweis: Wenn der Computer den Arduino nicht erkennt, kann dies drei Ursachen haben:

- Kabel ist nicht richtig eingesteckt.
- Ein falsches Arduino Board wurde ausgewählt: Werkzeuge-Board-Arduino/Genuino UNO.
- Arduino wird nicht erkannt: Werkzeuge-Port - / devtl/cu.usbmodem1411 (Arduino Uno).

Schaltung

Um das Spiel zu testen und zu sehen, welche Bauteile benötigt werden, zuerst die Schaltung aufbauen (vgl. Sketch oder fotografischen Schaltplan).

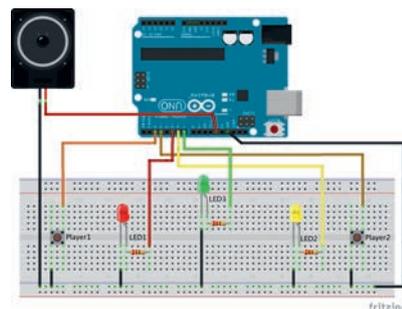


Abb. 294 | Fotografischer Schaltplan kann auf <http://www.makinginderschule.ch/programmierte-spiele.html> heruntergeladen werden.

Steckplatine: Die Steckplatine, engl. Breadboard, ist ein Kunststoffbrett, durch welches Strom fließen kann. Darauf können elektronische Bauteile direkt oder mit einem Jumperkabel elektrisch verbunden werden, ohne dass die Teile verlötet werden müssen. Die Löcher auf der Steckplatine sind in einer bestimmten Richtung miteinander verbunden (vgl. Abbildung).

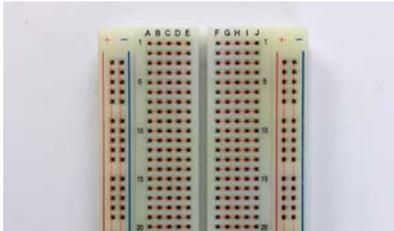


Abb. 295 | Steckplatine

Jumperkabel: Jumperkabel auch Dupont Kabel genannt, dienen zur einfachen Verbindung der Bauteile mit Steckplatine und Arduino. Jumperkabel gibt es in verschiedenen Ausführungen und Steckern:

- männlich/männlich
- weiblich/weiblich
- männlich/weiblich

Zum Zusammenstecken von ReadySteadyBaam werden männlich/männlich Jumperkabel benötigt.



Abb. 296 | weiblicher und männlicher Stecker

Material: Arduino UNO, 3 LED mit 390 Ohm Widerstand, Lautsprecher, 2 Taster, Batterieclip mit DC-Stecker, 9V Batterie, Schalter, diverse Jumperkabel, Litzen- oder Schaltdraht

Hinweis: Bauelemente sind in verschiedenen Ausführungen erhältlich, wobei sich die Verkabelung ändern kann.

Bauteile ausmessen: Welchen Durchmesser haben die Bauteile? Wie gross müssen die Löcher in der Box sein?

Box: Lehrgang Boxbau (vgl. Behältnisse mit Holzwerkstoffen XI-02) durchführen oder von Zuhause eine Box mitbringen.

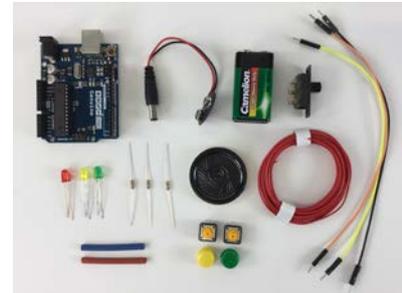


Abb. 297 | Benötigtes Material

Gestalterisches Experiment

Anordnung der Bauteile: Wie können die Bauteile auf der Box angeordnet werden, dass das Spiel logisch ist und zur Box passt?

Verschiedene Anordnungen testen und mit einem Foto oder einer Skizze festhalten.

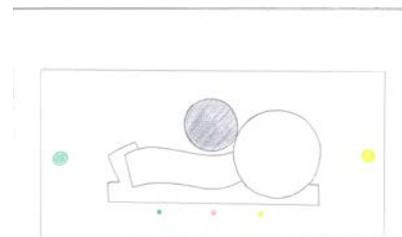


Abb. 298 | Entwurfskizze und Umsetzung

Umsetzung

In Originalgrösse einen Plan mit Massangaben zeichnen, indem die Anordnung aus dem Gestalterischen Experiment übertragen wird.

Verkabeln der Bauteile: Bei den Bauteilen für ReadySteadyBaam müssen immer zwei Kabel angelötet werden. Ein Kabel für GND (Ground, Minus-Pol) und eines für den Pin-Anschluss.

GND: Vorerst nur ein Litzen- oder Schaltdraht anlöten. Nach dem Einbau in die Box werden die Kabel zu einem Kabelbaum zusammengenommen.

Pin-Anschluss: An einem Jumperkabel einen Anschluss abschneiden und an das Bauteil löten.

Hinweis:

- Die Länge von Jumperkabel und Litzendraht ist abhängig von der Grösse der Box.
- Arbeiten in Serie. Z. B. alle LEDs verlöten.
- Vor dem Einbauen der Bauteile in die Box die verkabelten Bauteile einzeln testen. Die Jumperkabel können direkt auf den Arduino gesteckt werden, die GND-Kabel können z.B. mit einer Krokodilklemme und einem Jumperkabel eingesteckt werden.
- Mit Schrumpfschlauch über den Lötstellen werden diese gestärkt und Kurzschlüsse können vermieden werden.

Material: LED, Widerstand (Grösse berechnen), Schrumpfschlauch, Jumperkabel, Litzendraht, Lötmaterial (Lötstation, Lötzinn, Unterlage, evtl. Löthilfe)



Abb. 299 | Material zum Verlöten einer LED

Leuchtdioden (LED)

Material: LED, Widerstand, Jumperkabel, Litzendraht, Lötmaterial, Schrumpfschlauch

Anode (Plus, langes Bein): An die Anode wird der Widerstand und ein Jumperkabel angelötet.

Kathode (Minus, kurzes Bein): An die Kathode wird ein Stück Litzendraht angelötet, welches später zu einem Kabelbaum zusammengenommen wird.

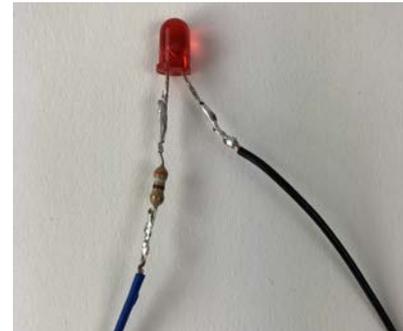


Abb. 300 | Anode und Kathode der LED. Das blaue Kabel nach dem Widerstand ist ein Jumperkabel, welches auf den programmierten Pin am Arduino angeschlossen wird. Das schwarze Kabel ist ein Stück Litzendraht, welches später zu einem Kabelbaum für GND zusammengenommen wird.

Lautsprecher

Material: Lautsprecher, Jumperkabel, Litzendraht, Lötmaterial

Auf dem + ein Jumperkabel anlöten.

Auf dem - ein Stück Litzendraht anlöten.



Abb. 301 | Rückseite des Lautsprechers

Taster

Material: Taster, Jumperkabel, Litzendraht, Lötmaterial, Schrumpfschlauch

Bei einem Taster spielt es keine Rolle, welches Bein an den Arduino angeschlossen und welches für GND genutzt wird. Üblich sind Taster mit zwei oder vier Beinen.

An ein Bein ein Jumperkabel anlöten, ans andere Bein ein Stück Litzendraht anlöten.



Abb. 302 | Beispiel eines Tasters mit vier Beinen

Kabelbaum

Nach dem Einbauen der Bauteile in der Box werden alle Litzendrähte, welche auf das GND des Arduino kommen, zu einem Kabelbaum zusammengekommen.



Abb. 303 | Kabelbaum mit Jumperkabel

Batterieclip mit Schalter

Damit das Spiel ein- und ausgeschaltet werden kann, wird in den Batterieclip ein Schalter eingebaut.

Vorüberlegung: Wo wird der Schalter eingebaut? Eventuell muss eine Kabelverlängerung eingelötet werden.

Ein Kabel in der Mitte trennen, Enden abisolieren und Schrumpfschlauch einführen.



Abb. 304 | Durchgetrennter Batterieclip mit Schrumpfschlauch und Schalter.

Dem Schalter entsprechend die abisolierten Enden anlöten.

Schrumpfschlauch positionieren und schmelzen.

Hinweis: Je nach Grösse des Schalters muss der Schalter vor dem Löten in die Box eingebaut werden.



Abb. 305 | Bei den meisten Schaltern wird ein Kabel in der Mitte und eines links oder rechts angelötet.

Endmontage

Den Plan auf die Box übertragen.

Die Löcher für die runden Bauteile können gebohrt werden. Die Löcher für die eckigen Bauteile müssen ausgesägt werden.



Abb. 306 | Löcher für Schalter, LED und Taster
Grösse der Löcher testen und danach schleifen.

Bauteile an den Jumperkabeln mit der entsprechenden Pin-Nummer anschreiben.

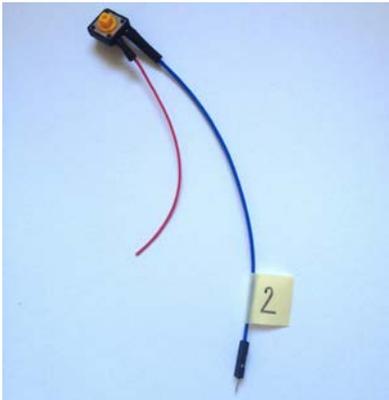


Abb. 307 | Dieser Taster wird an Pin-Nummer zwei des Arduino angeschlossen.

Bauteile in die Box stecken und befestigen.

Die Bauteile auf den entsprechenden Pin des Arduino stecken.

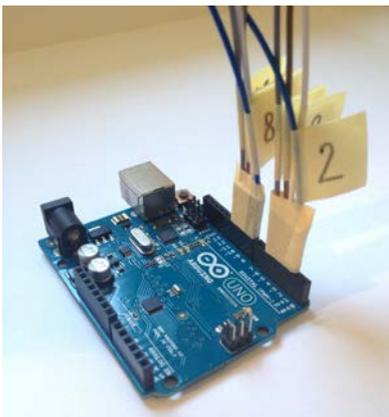


Abb. 308 | Jumperkabel auf dem Arduino einstecken.

Spiel testen: Batterie anschliessen, einschalten, testen.

Durch abändern der Töne und der Spielgeschwindigkeit erhält das Spiel eine persönliche Note.

Die Änderungen werden im Sketch ausgeführt und haben keinen Einfluss auf den Aufbau der Schaltung.

Nach dem Abändern wird der Sketch auf den Arduino kopiert und das Spiel läuft mit den neuen Eingaben.

Als Zusatz kann das Aufleuchten der LED abgeändert werden.

Hinweis: Änderungen programmieren und durch neues Hochladen gleich testen.

Töne ändern

Soll ein Ton abgespielt werden, wird dies so programmiert:

```
speaker.play (NOTE_C5, 100);
```

Beispiel Ton: NOTE_C5

Überall im Sketch wo «NOTE_xx» steht, kann ein anderer Ton eingesetzt werden. Eine Liste mit den Tönen und den dazugehörigen Frequenzen ist auf der Tone Library Download Seite ersichtlich (<https://github.com/bhagman/Tone>).

Bei ReadySteadyBaam kann die Einstiegsmelodie und der Ton beim Aufleuchten der LEDs abgeändert werden.

Spielgeschwindigkeit ändern

Zu Beginn von «void runGame()» kann die Spielgeschwindigkeit abgeändert werden.

«int bam = random(500, 3000)» beschreibt per Zufall die Zeit zwischen dem Aufleuchten der 2. und 3. LED.

«int d = 1000» beschreibt die Pause zwischen dem Aufleuchten der 1. und 2. LED, dem Anzeigen des Sieges/der Niederlage und bevor ein neues Spiel beginnt.

Die Zeiten können auch einzeln geändert werden, indem überall bei (d) deine Zahl (Millisekunden) hingeschrieben wird.

Hinweis: Die Zeit wird in Millisekunden angegeben.

SOFTWARE ARDUINO (IDE)

PROGRAMM INSTALLIEREN

Link

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

Vorgehen

Beim Herunterladen wird automatisch ein Ordner «Arduino» unter Dokumente erstellt.

Download the Arduino IDE



Abb. 309 | Betriebssystem wählen

Contribute to the Arduino Software

Consider supporting the Arduino Software by contributing to its development. (US tax payers, please note this contribution is not tax deductible). [Learn more on how your contribution will be used.](#)



Abb. 310 | Software kann gratis unter „JUST DOWNLOAD“ heruntergeladen werden.

TONE LIBRARY DOWNLOAD

PROGRAMM INSTALLIEREN

Link

<https://github.com/bhagman/Tone>

Vorgehen

- Dateien unter Dokumente-Arduino-libraries speichern
- Ordner von «Tone-master» zu «Tone» umbenennen
- Bevor die Tone-Library verwendet werden kann, muss die IDE neu gestartet werden.
- Wird die Tone-Library nicht automatisch erkannt, muss sie in der IDE eingebunden werden.



Abb. 311 | Unter „Clone or Download“ „Download ZIP“ wählen