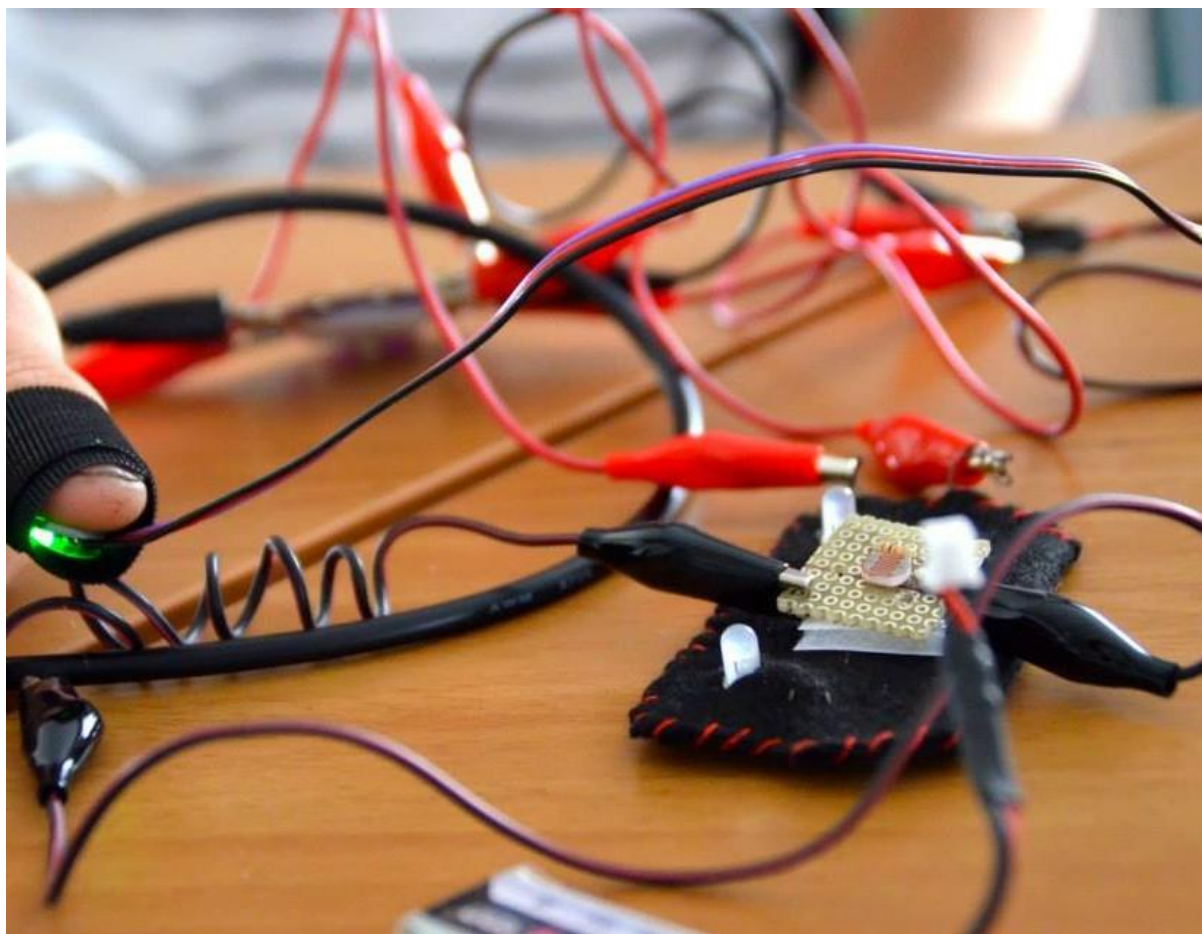


Internet der Dinge im Bildungsbereich

Die Aufgabe dieses Handbuchs ist es, verschiedene Methoden, Lernszenarien und Bildungsaktivitäten zu Codierung und IoT in der Bildung vorzustellen.



Um erfolgreich zu sein, müssen sie lernen, innovative Lösungen für die unerwarteten Probleme zu entwickeln, die zweifellos in ihrem Leben auftreten werden. Ihr Erfolg und ihre Zufriedenheit basieren auf ihrer Fähigkeit, kreativ zu denken und zu handeln. Wissen allein reicht nicht aus: Sie müssen lernen, ihr Wissen kreativ einzusetzen.

- Mitchel Resnick, MIT Media Lab

Eine Einführung in das Internet der Dinge und Wearables (tragbare Geräte) in der Bildung

In diesem Handbuch erfahren Sie, wie Sie mit einem einfachen Online-Editor einfache Sensoren und Ausgänge programmieren und mit ihnen interagieren. Das Internet der Dinge ist ein wachsendes Marktfeld, von Thermostaten bis hin zu Smartwatches.

Im Bildungsbereich wird es wichtig sein, all diese Technologien einzuführen, da es für Lernende eine Aktivität mit konkreter Anwendung ist und man die Theorie der 4P anwenden kann.

Project (Projekt), Peer (Gleichgesinnte), Play (Spiel) und Passion (Leidenschaft) – eine Lernmethode, die von Mitchel Resnick vom MIT Lifelong Learning Lab (MediaLab) geprägt wurde und gut zu den IoT-Aktivitäten im Bildungsbereich passt. Darüber hinaus bieten wir in diesem Handbuch die Möglichkeit, ein Programm mit einigen Sensoren zu erstellen, die auf einer anderen kommerziellen Plattform vorhanden sind und alle mit dem Online-Editor Scratch 3 verknüpft sind. In den Lernszenarien verwenden wir auch verschiedene Programme wie Snap for Arduino oder Makecode. Alle verwendeten Softwareprodukte sind kostenlos und kompatibel mit den wichtigsten Robotikplattformen wie Lego, Microbit, Arduino und Raspberry PI.

Einführung in Arduino und ähnliche Shields.

Wir präsentieren nur das Arduino-Shield und die compatible Plattform, da dies Open Source ist. In EU-Projekten betrachten wir die Verwendung billiger und Open Source-Plattformen wie Arduino und Elegoo als ethische Verantwortung.

Diese Shields sind vollständig kompatibel mit einem Poletora OS-Sensor und Aktuatoren.

Arduino ist die beliebteste Open-Source-Plattform für die Elektronik, die dank des niedrigen Preises und der einfachen Verwendung von Prototyping und Programmierung die Welt der Bildung verändert.

Ein typisches Shield besteht aus einem 8-Bit-Mikrocontroller mit verschiedenen Chips

aus der Mega AVR-Familie. Für jede Abschirmung gibt es einen digitalen Eingang, einen digitalen Ausgang und einen analogen Ein- und Ausgang.

Kostenlose Online-Software

In diesem Projekt verwenden wir nur kostenlose Software, um unsere Objekte zu programmieren. Das beliebteste Programm ist Scratch, das aktuell in der Version Scratch3 erhältlich ist.

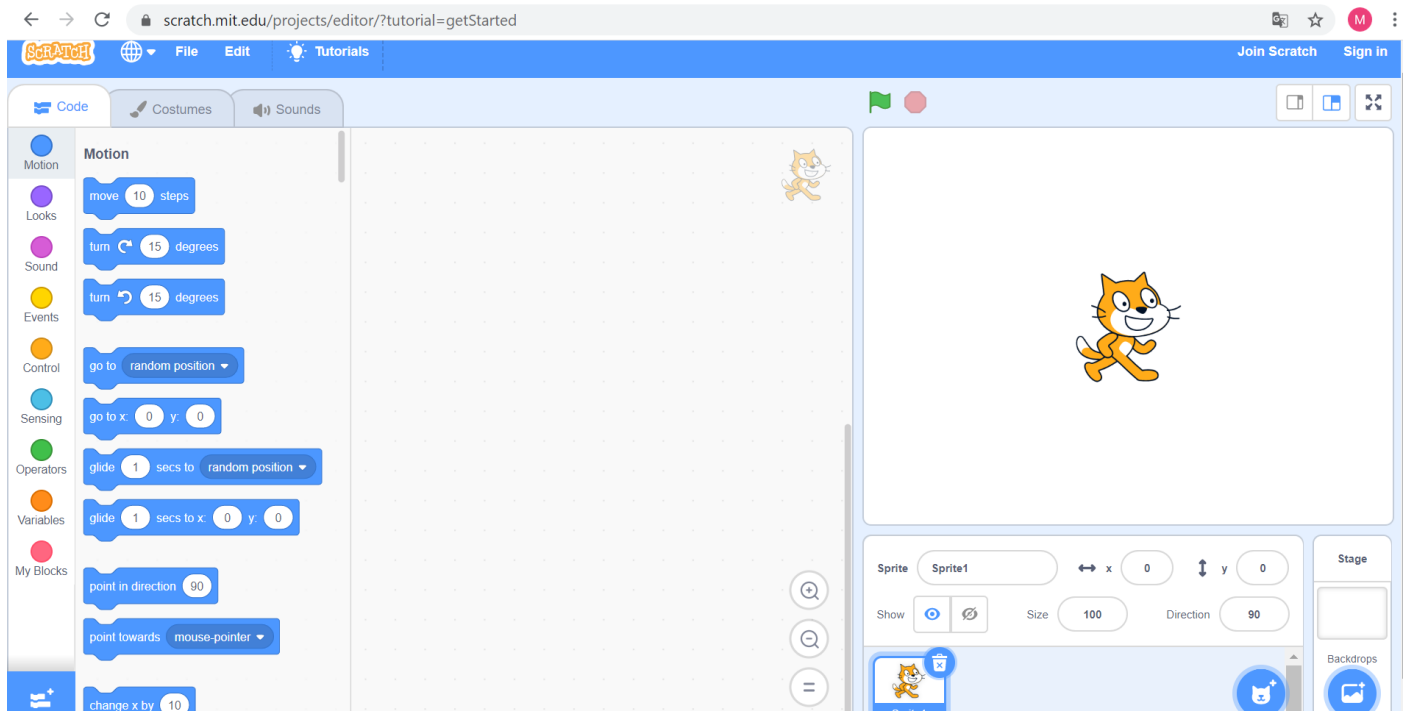
Scratch

Scratch ist ein visueller Programmiereditor, der vom MIT Media Lab entwickelt wurde. Scratch wurde 2006 geboren und wird heute in den meisten Schulen der Welt eingesetzt. Es ist in mehr als 70 Sprachen verfügbar.

Scratch 2 ist auch offline erhältlich. Hier kann es heruntergeladen werden: <https://scratch.mit.edu/download>. Scratch 2 ist nicht für Tablets erhältlich.

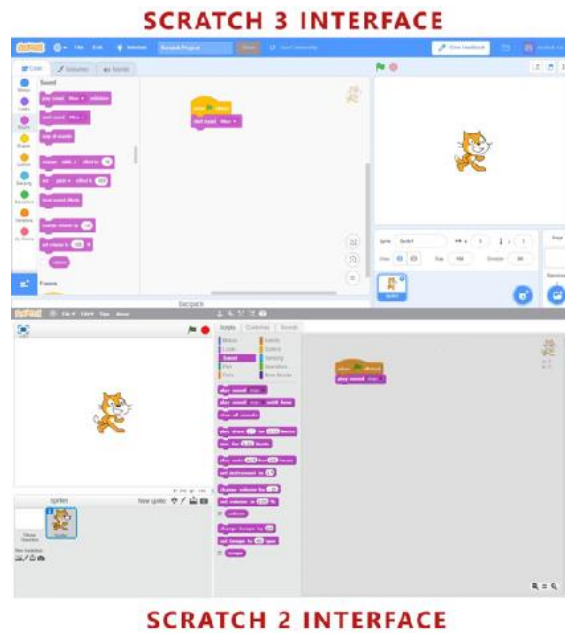
Hier die wichtigsten Unterschiede zwischen Scratch 2 und 3: Beide Scratch-Versionen laufen online über Internet-Browser über den PC. Scratch 3 ist in HTML5, sodass es auch auf Android-Tablets oder iPads läuft, aber es hat im Gegensatz zu Scratch 2 (Flash) keine Offline-Version. Beide Programme können mit externen Geräten interagieren, jedoch nur auf Computern, da ein kleines Linkerprogramm installiert werden muss. Scratch 2 kann Lego WeDo 1 und 2 und Picoboard nativ steuern.

Scratch 3 kann Lego WeDo 2 und Lego Mindstorm EV3, Microbit, nativ steuern. Es hat auch zusätzliche Funktionen wie Text-to-Speech in einer anderen Sprache und Übersetzer. Es ist möglich, neue Erweiterungen zu erstellen, damit in Zukunft mehr zusätzliche Funktionen verfügbar sind. Es gibt einige andere zusätzliche Funktionen, die den beiden Plattformen gemeinsam sind, wie Zeichenstift, Musikinstrumente und Videoerfassung.



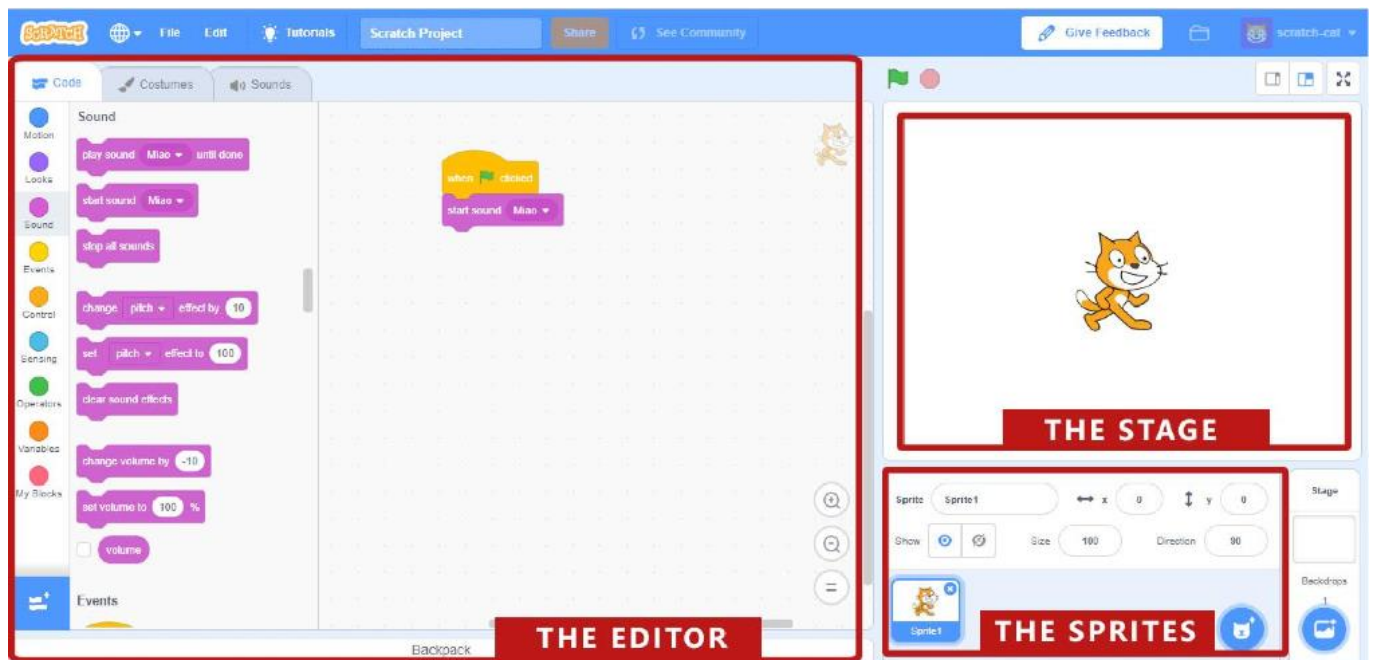
Hauptunterschiede zwischen Scratch 2 und Scratch 3		
	Scratch 2	Scratch 3
Offline-Version (Windows, OSX)		
Online-Version	(nur auf Computer)	(auf Computer und Tablet)
Externe Geräte		
Lego WeDo 1	X	
Lego WeDo 2	X	X
Picoboard	X	
Lego Mindstorm EV3	X	X
Microbit		X
Extra-Funktionalitäten		
Text-to-speech in verschiedenen Sprachen	X	X
Graphikstift	X	X
Musikinstrumente		X
Videoerfassung		X
Textübersetzung		X

In beiden Versionen können Scratch-Projekte in der Scratch-Community gespeichert und freigegeben werden. Die Dateien sind mit beiden Versionen kompatibel. Seit Januar 2019 ist Scratch 3 die einzige Online-Version und Scratch 2 die einzige Offline-Version.



Die Oberfläche

So sieht die neue Oberfläche von Scratch 3 aus.



Sie ist in drei Hauptbereiche unterteilt:

Stage (=Bühne) ist der wichtigste Bereich, wo das Programm 'zum Leben erweckt wird'.

The sprite's area (=Bereich des Objektes), in diesen Bereichen befinden sich alle Komponenten/Objekte (sprite), die Teil der Bühne sind.

The editor area (=der Editierbereich), umfasst die drei verschiedenen Editorentypen

1. **Der Kodiereditor (The code editor):** enthält die Liste aller verfügbaren Blocks und alle Blocks, die verwendet werden, um das Verhalten des Charakters zu beschreiben. Es ist wichtig zu verstehen, dass jedes Sprite seine eigenen Blöcke hat. Wenn wir also verschiedene Sprites im Sprite-Bereich auswählen, ändern sich die Blöcke im Skriptbereich.
2. **Der Kostümeditor (The costumes editor):** erlaubt das Zeichnen und Ändern des Aspekts der Sprites. Jedes Sprite hat seine eigenen Kostüme.
3. **Der Ton-Editor (The sound editor):** ermöglicht das Aufnehmen und Bearbeiten von Sounds zur Verwendung im Programm.

Snap 4 Arduino

Snap 4 Arduino ist eine Modifikation des Blockprogramms Snap!, das von der University of California in Berkeley erstellt wurde. Dank Snap (das ständig weiterentwickelt wird) können wir problemlos alle Arduino-Boards programmieren.



Laut der offiziellen Webseite, sind die Funktionen von Snap4Arduino folgende:

- Blockbasierte, dynamische, Live-, gleichzeitige, parallele Programmierung
- Fast alle Arduino-Boards werden unterstützt.
- Verwendet Standard Firmata Firmware
- Automatisch konfigurierbare Pinbelegung und Hardware-Abstraktionen und high level Hardware-Abstraktionen
- Man kann mit verschiedenen Boards gleichzeitig interagieren.
- Desktop-Versionen für die drei Hauptbetriebssysteme
- Online-Version, die über ein Chrome / Chromium-Plugin eine Verbindung zu Arduino-Boards herstellen kann
- Kostenlose Software, lizenziert unter der Affero GPLv3
- Transpilation einfacher Skripte in Arduino-Skizzen
- HTTP-Protokoll zur Fernsteuerung und zum Live-Streaming des Snap! Bühne
- Befehlszeilenversion für eingebettete GNU / Linux

Installation der Software

Installation von Scratch 3: <https://scratch.mit.edu>

Installation von Snap4Arduino: <http://snap4arduino.rocks>

Focus auf Scratch 3

Scratch wurde speziell für Kinder zwischen 8 und 16 Jahren entwickelt, ist jedoch ein sehr nützliches Werkzeug für jeden Programmieranfänger.

Scratch wird in mehr als 150 verschiedenen Ländern verwendet und ist in mehr als 40 Sprachen verfügbar. Scratch wird als Einführungssprache verwendet, da die Erstellung interessanter Programme relativ einfach ist und die erlernten Fähigkeiten auf andere Programmiersprachen wie Python und Java angewendet werden können.

Category	Notes	Category	Notes
Motion	Moves sprites, changes angles and changes X and Y values.	Sensing	Sprites can interact with the surroundings the user has created
Looks	Controls the visuals of the sprite; attach speech or thought bubble, change of background, enlarge or shrink, transparency, shade	Operators	Mathematical operators, random number generator, and-or statement that compares sprite positions
Sound	Plays audio files and effects. Programmable sequences are now available as an extension category named "Music".	Variables	Variable and List usage and assignment
Events	Contains event handlers placed on the top of each group of blocks	My Blocks	Custom procedures (blocks).
Control	Conditional if-else statement, "forever", "repeat", and "stop", etc.		

Noch dazu gibt es für Scratch folgende Erweiterungen:

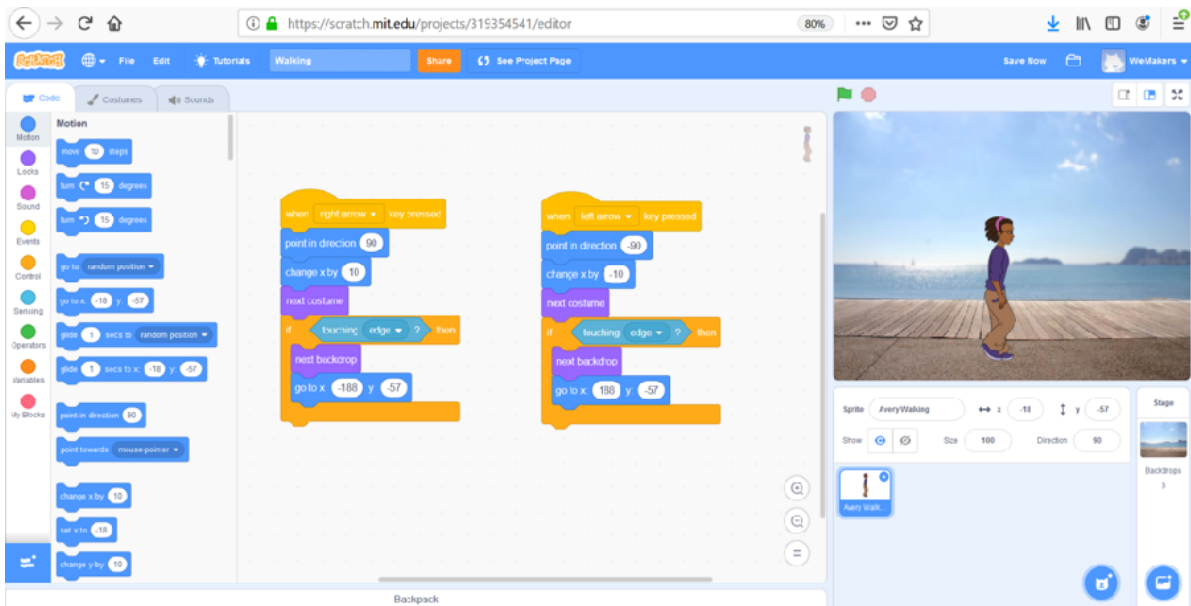
- Musik
- Stift (pen)
- Videoerfassung
- Text To Speech
- Übersetzung
- Makey Makey
- micro:bit
- LEGO MIDSTORMS EV3
- LEGO BOOST
- LEGO Education WeDo 2.0
- Go Direct Force & Acceleration

Applikation 1: eine Figur, die geht (linke/rechte Pfeile)

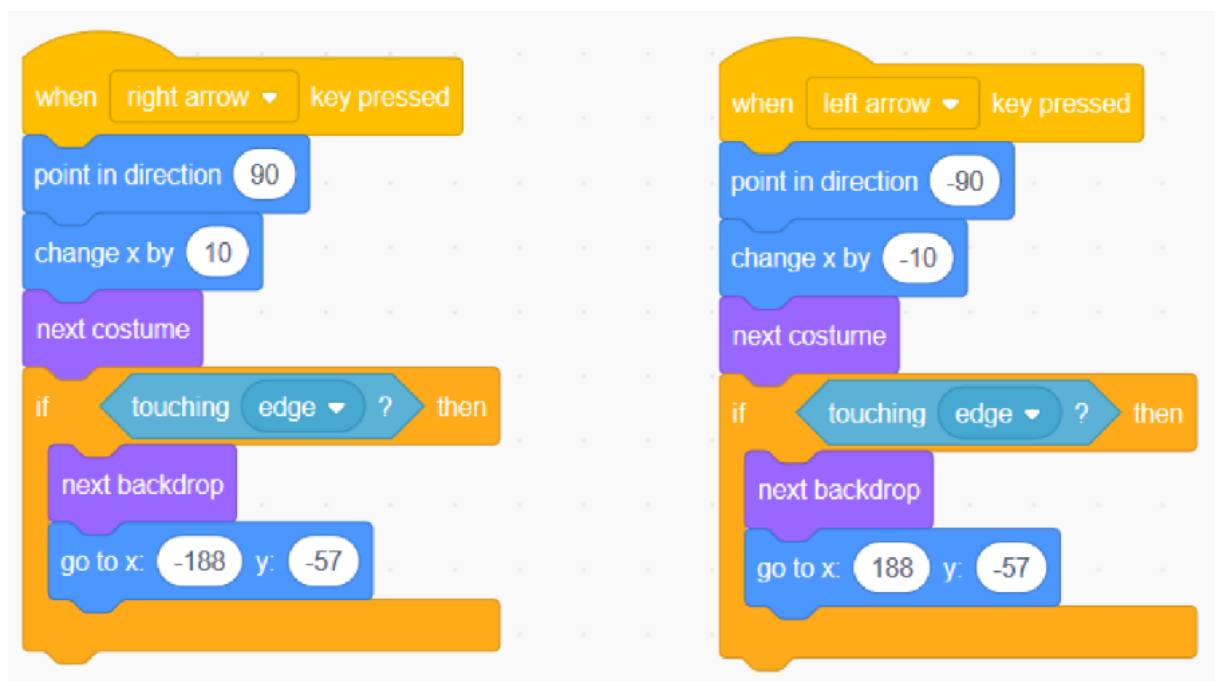
Die folgende Anwendung hat einen Charakter (Sprite), der durch linke und rechte Pfeile gesteuert wird und in drei oder mehr Hintergründen (Hintergründen) läuft.

Schritt-für-Schritt-Programmierung:

1. Wähle ein Sprite (am besten eins mit einem "Geh-"/Flug- oder Schwimm-Kostüm).
2. Wähle drei oder mehr Hintergründe.



3. Schreibe die im nächsten Bild dargestellten Skripte.
4. Teste das Ergebnis mit dem Pfeil nach rechts und links.



Vorschlag für eine Herausforderung (Challenge)

Challenge 1: Lass den Charakter springen und vorrücken, wenn du den Aufwärtspfeil drückst.

Challenge 2: Entwirf ein Spiel:

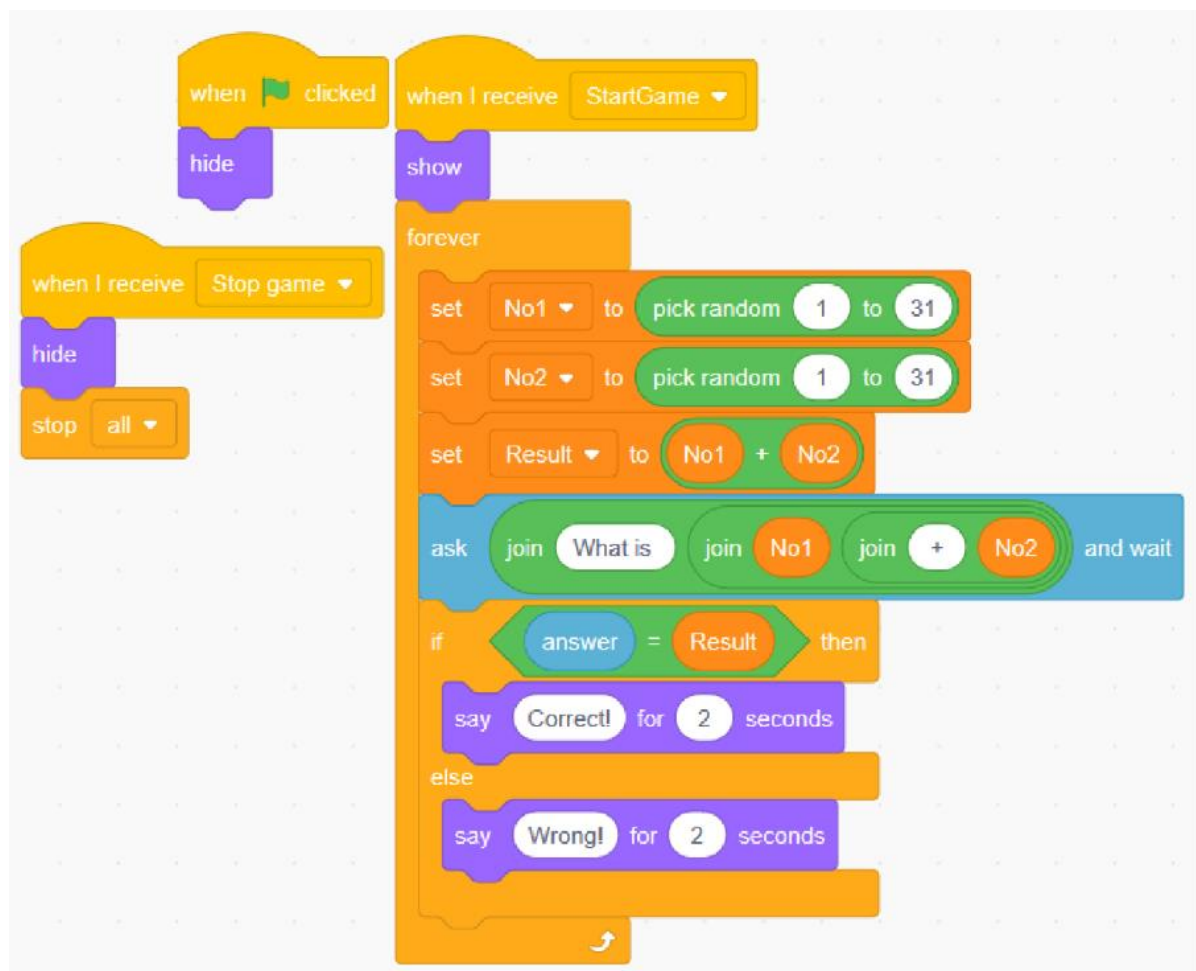
1. Ein kontrollierbares Objekt und ein Frosch, die zufällig voranschreiten (nur x-Werte verändern)
2. Das Objekt darf den Frosch nicht berühren (es soll über den Frosch springen)
3. Es hat drei Leben.
4. Wenn es den Frosch berührt, verliert es ein Leben.
5. Das Spiel endet, wenn kein Leben mehr da ist.

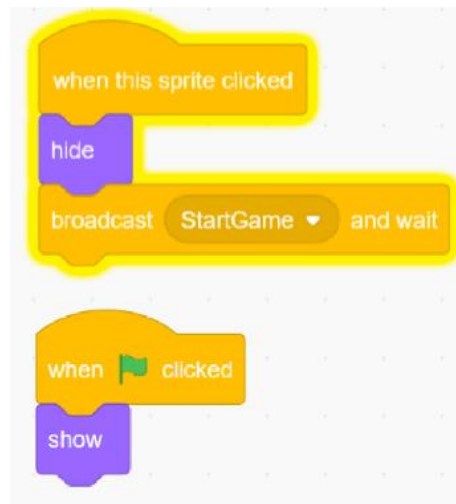
Du kannst den entsprechenden Code hinzufügen, damit sich der Charakter 2 Sekunden lang in einen Frosch oder ein anderes Tier ändert, wenn er den Frosch berührt.

Challenge 3: Das Spiel generiert zufällige Ergänzungen und zeigt zu jeder Antwort Richtig oder Falsch an. Es hat drei Hintergründe und drei Sprites (2 Schaltflächen und ein Objekt, das die Frage stellt und die Antwort gibt).



Die Skripte für jedes Sprite und für die Bühne sind:





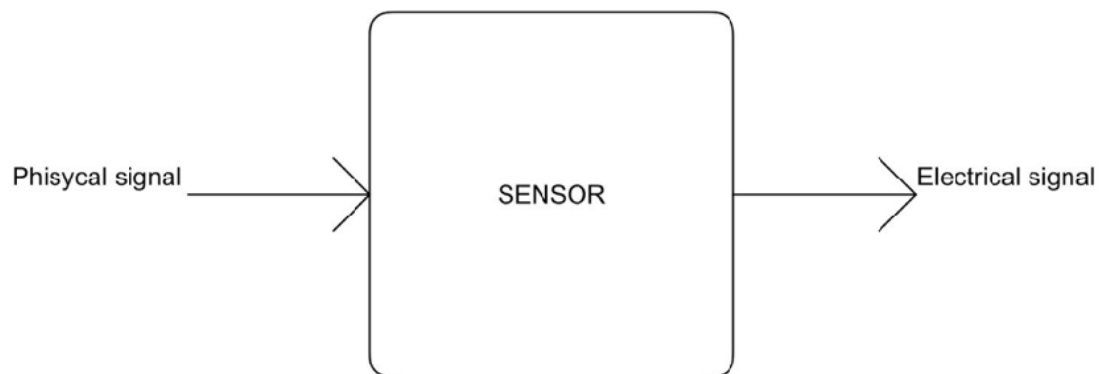
Challenge 4: Füge zwei Variablen hinzu, um die Gesamtzahl der Fragen und die Gesamtzahl der richtigen Antworten zu zählen. Zeige die beiden Werte am Ende an.



Challenge 5: Entwirf dein eigenes Quiz mit 2-3 Fragen.

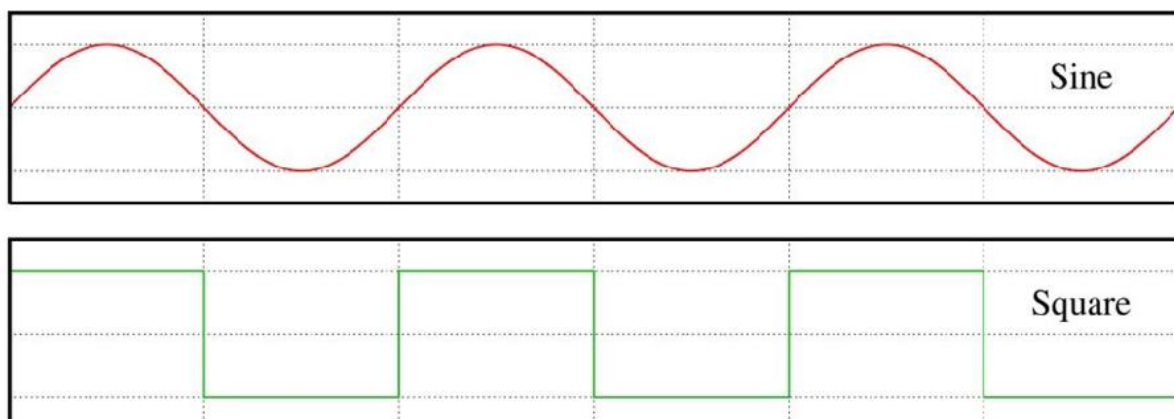
Sensoren: ein kurzer Überblick

Ein Sensor ist ein Gerät, das in der Elektronik verwendet wird, um Änderungen eines physikalischen Parameters in der Umgebung zu erfassen und diese in ein elektrisches Signal kodierten Informationen an andere Geräte zu senden, um sie zu manipulieren und zu analysieren.



Das elektrische Signal kann analog oder digital sein. Analog bedeutet, dass das Signal mit der Kontinuität variiert und alle Werte zwischen einem Minimum und einem Maximum liegen. Digital bedeutet, dass das Signal nur eine begrenzte Anzahl von Werten annehmen kann, typischerweise zwei: niedriger und hoher Pegel.

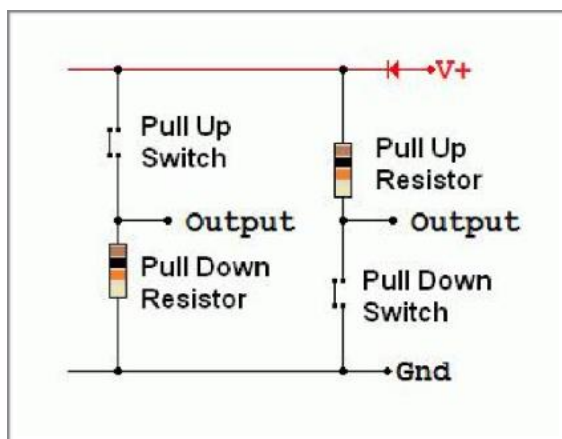
In der folgenden Abbildung sehen Sie zwei Beispiele für Signale: eine Sinuswelle und eine Rechteckwelle. Der erste nimmt alle Werte von der unteren bis zur oberen Grenze an, also ist es ein analoges Signal. Der zweite nimmt nur zwei Werte an und wechselt von einem niedrigen in einen hohen Zustand und umgekehrt. Es handelt sich also um ein digitales Signal.



Ein Beispiel für ein analoges Signal ist die Temperatur: Sie kann jeden Wert vom absoluten Nullpunkt bis unendlich annehmen. Das typische Beispiel für ein digitales Signal ist der Zustand einer Taste: Sie kann nur gedrückt oder gedrückt werden. Der typische analoge Sensor ist der variable Widerstand (zum Beispiel das Potentiometer zum Erhöhen und Verringern der Lautstärke unserer Stereoverstärker), und der digitale ist die Taste, die wir im nächsten Abschnitt sehen werden.

Berührungssensor

Der einfachste Sensor, den wir konstruieren können, ist der Berührungssensor. Es gibt verschiedene Arten von Berührungssensoren, aber die einfachste, die wir verwenden können, ist die Taste und der Schalter.



Ein Knopf ist nur ein elektromechanisches Gerät, das einen schließen oder öffnen kann Stromkreis. Wenn der Stromkreis geschlossen ist, kann der elektrische Strom fließen, andernfalls kann der Strom beim Öffnen nicht fließen.

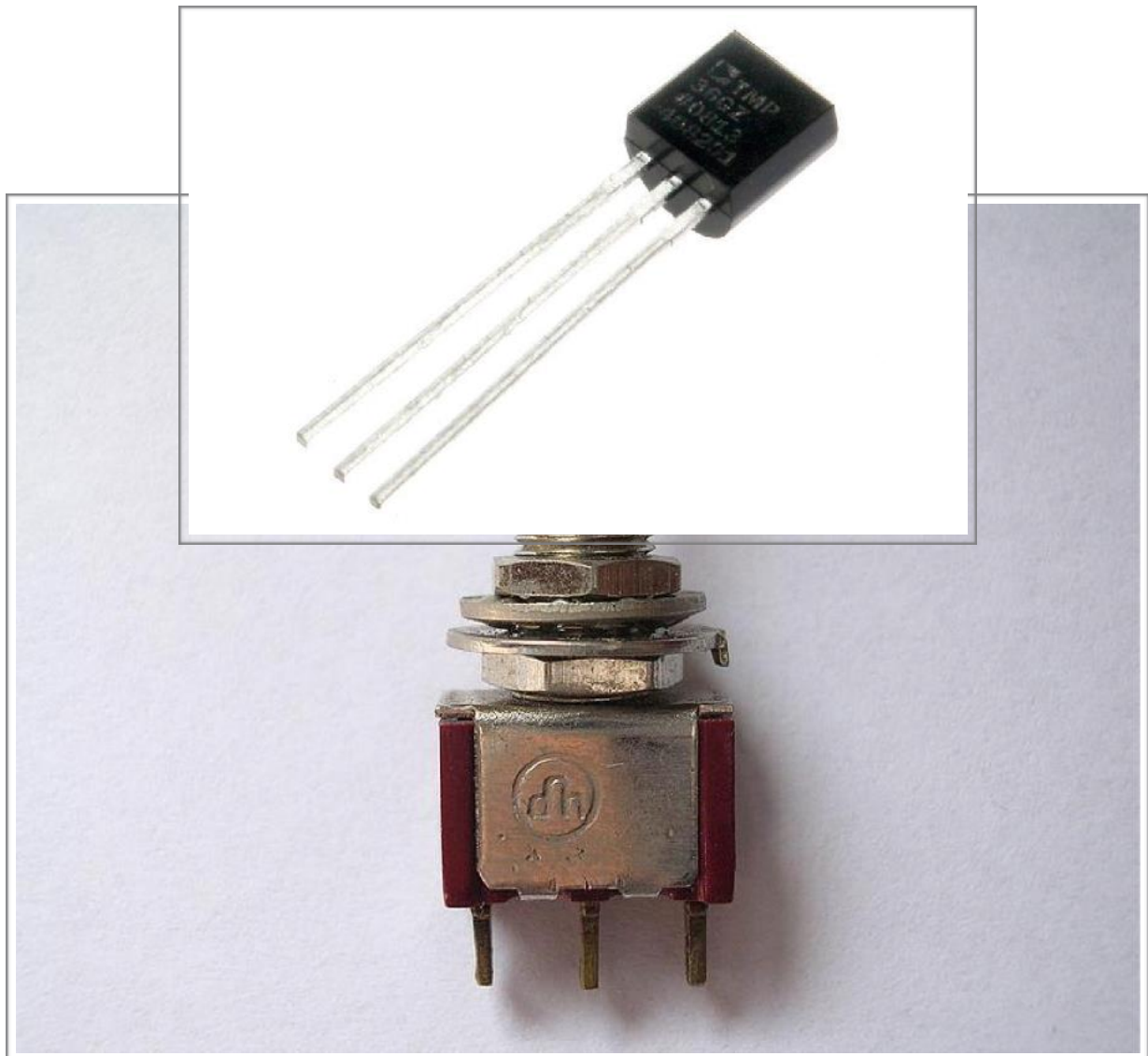
Wenn Sie eine Taste in eine geeignete Schaltung mit einem Widerstand einbauen, erhalten Sie eine Schaltung, die nur zwei niedrige oder hohe Spannungswerte liefern kann, die dem Zustand der Taste entsprechen. Sie sehen das folgende Schema für die Verbindungen. Sie können den Widerstand und den Schalter auf zwei Arten platzieren. Das



Erstens: Der „Pulldown-Widerstand“ bietet einen niedrigen Wert am Ausgang, wenn der Schalter geöffnet ist, und einen hohen Wert, wenn er geschlossen ist. Die Version „Pull-up-Widerstand“ bietet das entgegengesetzte Verhalten. Zum Beispiel verwenden wir Knöpfe und Unterbrecher in unserem Haus, um das Licht ein- und auszuschalten, oder im Aufzug, um die richtige Etage auszuwählen.

Unter diesem Link finden Sie weitere Informationen zu Buttons und einer Open Source-Programmierskarte: <https://www.arduino.cc/en/Tutorial/Button>

Es gibt andere Berührungssensoren, und die beliebtesten sind die kapazitiven Sensoren. Diese Technologie wird in unseren Touchscreen-Geräten verwendet. Sie messen die Änderung der Kapazität eines Kondensators aufgrund menschlicher Anwesenheit. Am Ende haben wir ein Gerät, das wie die Taste einen zweistufigen Ausgang bietet.

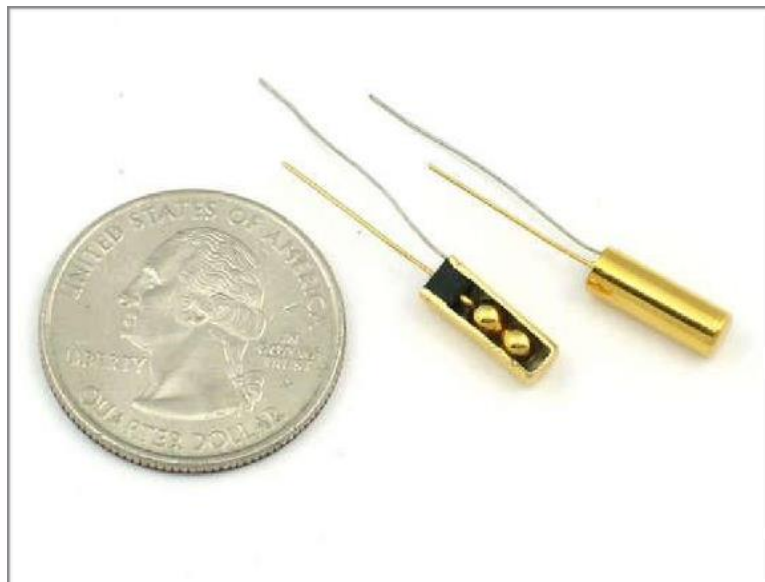


Neigungssensor

Der Neigungssensor wird verwendet, um zu erkennen, ob ein Objekt in eine oder mehrere Richtungen geneigt ist. Der einfachste Neigungssensor ist der Quecksilberschalter: Er besteht aus einer Vakuumlampe mit zwei Kontakten und etwas Quecksilber, die sich frei bewegen kann. Wenn die Glühbirne gekippt wird, bewegt sich das Quecksilber und wenn es die Kontakte erreicht, schließt es den Stromkreis. Das Quecksilber wird oft durch eine Metallkugel ersetzt. Mit mehr Kontakten können mehrachsige Neigungssensoren erstellt werden.

Ein Neigungssensor wird zum Beispiel in einem lichtemittierenden Jojo verwendet, das beim Spielen blinkt, oder in einem einfachen Pitch-Roll-Sensor in einigen Fahrzeugen.

Unter diesem Link finden Sie ein Beispiel mit einem einachsigen Neigungssensor und einer programmierbaren Karte: <https://learn.adafruit.com/tilt-sensor/using-a-tilt-sensor>

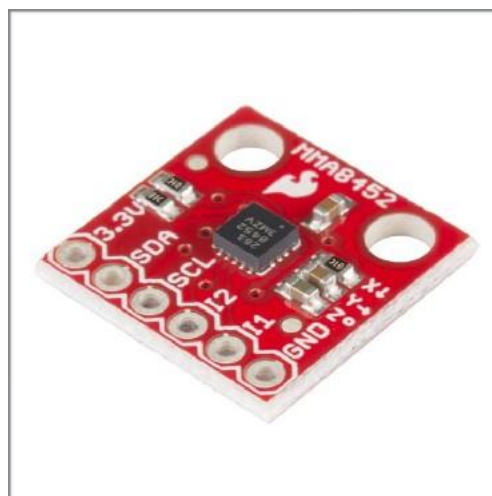


Beschleunigungsmesser (accelerometer)

Der Beschleunigungsmesser ist ein Gerät zur Messung der richtigen Beschleunigung. Es funktioniert aufgrund der physikalischen Trägheit einer Masse. Konzeptionell besteht ein Beschleunigungsmesser aus einer Masse, die mit einer gedämpften Feder verbunden ist. Wenn der Beschleunigungsmesser einer Beschleunigung ausgesetzt wird, bewegt sich die Masse in Bezug auf den Behälter proportional zum Beschleunigungsmodul.

Durch Messen der Verschiebung wird die Beschleunigung angegeben, der die Geräte ausgesetzt sind. Die Beschleunigung wird dann in gewisser Weise in ein elektrisches Signal codiert. Beschleunigungsmesser werden beispielsweise in Navigationssystemen verwendet, um die relative Position in Bezug auf einen Nullpunkt zu ermitteln, in unserem Smartphone als Eingabegeräte, um den Bildschirm zu drehen, um ein Spiel zu spielen, als Schrittzähler usw. oder in einigen Computern mit magnetischer Festplatte als Fallmelder, um Festplattenprobleme zu vermeiden und den Lesekopf in eine sichere Position zu bringen.

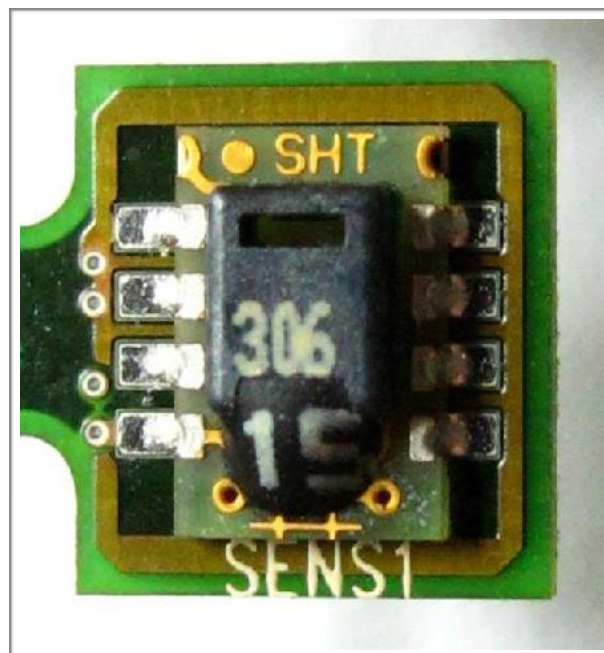
Unter diesem Link finden Sie ein Beispiel für die Verwendung eines analogen 3-Achsen-Beschleunigungsmessers mit einer programmierbaren Karte, der drei zu den Beschleunigungen proportionale Spannungen liefert:
<https://www.arduino.cc/en/Tutorial/ADXL3xx>



Feuchtigkeitssensor

Feuchtigkeitssensoren messen die Luftfeuchtigkeit in einem Medium wie der Luft oder dem Mutterboden. Sie messen meistens die Variation einer Kapazität und / oder eines Widerstands aufgrund der Variation der Wassermenge im Medium. Feuchtigkeitssensoren werden zum Beispiel in der Meteorologie, auch in unserer meteorologischen tragbaren Station, und in automatischen Bewässerungssystemen verwendet, die das Wasser nur öffnen können, wenn der Boden trocken ist.

Unter diesem Link finden Sie ein Beispiel mit einer Open Source-Karte und einem sehr verbreiteten Feuchtigkeitssensor (der auch die Temperatur messen kann).: https://www.tutorialspoint.com/arduino/arduino_humidity_sensor.htm



Temperatursensor

Ein Temperatursensor wandelt die Temperatur in ein elektrisches Signal um. Dies ist je nach Sensortyp aufgrund unterschiedlicher physikalischer Phänomene möglich. Zum Beispiel bietet der Thermistor eine Änderung seines Widerstands entsprechend

einer Änderung der Temperatur, und das Thermoelement bietet für den thermoelektrischen Effekt eine Spannung, die proportional zum Temperaturunterschied ist.

Temperatursensoren werden in der Meteorologie, im medizinischen Thermometer, in Computer-CPUs usw. verwendet.

Hier sehen Sie ein Beispiel für eine Temperaturmessung mit einer programmierbaren Karte: <https://learn.adafruit.com/tmp36-temperature-sensor/using-a-temp-sensor>

Output: kurzer Überblick

Eine Ausgabe (output) ist ein Element, das auf unterschiedliche Weise auf die reale Welt einwirken kann.

LED

Wenn Sie LEDs in Ihrer Klasse einführen möchten, können Sie je nach Schultyp viele verschiedene Methoden anwenden. In diesem Handbuch zeigen wir Ihnen allgemein alle elektronischen Komponenten, die Sie benötigen, damit Lehrer aus verschiedenen Disziplinen sie problemlos verwenden können

Bevor wir eine Komponente definieren, müssen wir die Erstellung der Komponente an das Leben ihres Erfinders binden. Der Erfinder der LED ist Nicholas Holonyak Jr., ein Amerikaner mit russischer Herkunft, der bereits 1962 die erste LED entwickelt hat. Sie konnte nur rotes Licht emittieren. Heutzutage ist es möglich, LEDs mit sehr unterschiedlichen Farben zu kaufen. Dank der blauen LEDs von Asaki, Amano und Nakamura war diese Forschung so wichtig, dass sie einen Nobelpreis erhielten.



Eine LED ist eine scheinbar sehr einfache Ausgabe, mit der Licht unterschiedlicher Farben zuverlässig emittiert werden kann. Sie hält lange und ist kostengünstig. Heutzutage werden LEDs für viele Anwendungen verwendet, nicht nur für gewerbliche Zwecke, sondern auch für Haushaltszwecke.

Das Funktionsprinzip basiert auf Elektronen, die einige Lücken in einem Halbleitermaterial einnehmen und Energie oder die Photonen freisetzen. Die unterschiedlichen Frequenzen, auch unterschiedliche Farben, hängen von den Materialien ab. Die LED benötigt konstanten Strom, um mit Strom versorgt zu werden. Dies ist der Grund, warum sie immer mit einem Widerstand verbunden ist, der es ermöglicht, den elektrischen Fluss zu steuern, der die Diode erreicht.

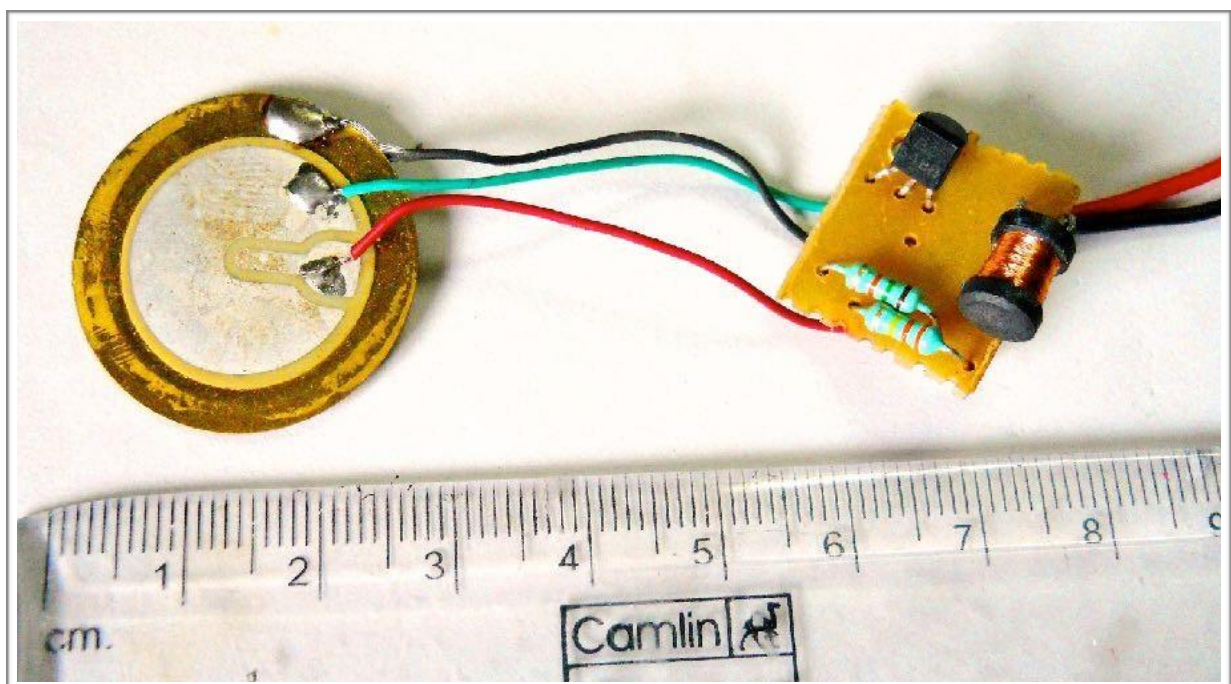
Mehr Übungen:

Hier ist eine Liste einiger zusätzlicher Übungen (die in diesem Handbuch nicht behandelt werden), die Ihren Schülern helfen, ihr Wissen zu üben und zu verbessern. Es ist auch nützlich, die Bewertung des Lehrers zu vereinfachen:

- Ändern Sie die Blinkzeit der LED von 1 auf 3 Sekunden.
- Ändern Sie den digitalen Pin (nicht 0 und 1!), An dem die LED angeschlossen ist, und ändern Sie das Programm entsprechend.
- Ändern Sie die Blinkzeit folgendermaßen: LED leuchtet 1,5 Sekunden lang - LED leuchtet 2/3 Sekunden lang.

Buzzer

Das erste ist, dass Sie Arduino verwenden können, um Sounds mit unterschiedlichen Tönen und Längen auszugeben. Daher können Sie über Musik und die Lieblingslieder und -musik der Schüler sprechen. Darüber hinaus ist es möglich, einen Teil aus S. Spielbergs "Enge Begegnungen der dritten Art" zu sehen, in dem Menschen mithilfe von 5 Noten mit Außerirdischen kommunizieren (die Interpretation des Wissenschaftlers, der diese Szene startet, ist François Truffaut). Natürlich können Sie die Schüler dazu bringen, ihre eigene Musik zu machen.

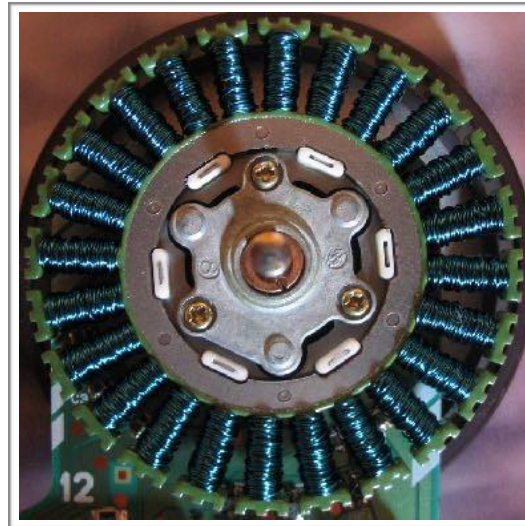


Auf diese Weise können Sie die Schüler einbeziehen, um die Codierung auf kreative Weise zu verwenden. Nachdem Sie die Einführung zum Ausgeben von Noten mit dem Summer vorgenommen haben, können Sie die Software so programmieren, dass Sie lernen, wie Sie berühmte Songs spielen.

Motoren

Motoren ermöglichen es, den Roboter zum ersten Mal zu bewegen, die erste Bewegung des Roboters wird sehr einfach und albern sein, der Roboter wird unendlich sorglos über Hindernisse und die Betriebszeit vorwärts gehen. In der Arduino-Welt Die H-Brücke ist ein Transistor, mit dem wir zwei Gleichstrommotoren gleichzeitig steuern

können. Ohne die H-Brücke wäre es nicht möglich, den Roboter in Bewegung zu setzen. Die H-Brücke ist in der Tat eine Batterie, die die Energieversorgung der Roboter verwaltet. Motoren, die im beliebtesten Arduino-Kit enthalten sind, sind Gleichstrommotoren ohne Sensor. Deshalb ist es nicht möglich, die Motordrehung zu steuern, sondern nur den an sie gelieferten Strom.



„Jeder Hersteller von Videospielen weiß etwas, was die Hersteller von Lehrplänen nicht zu verstehen scheinen. Sie werden nie sehen, dass ein Videospiel, für das geworben wird, einfach ist. Kinder, die die Schule nicht mögen, werden Ihnen sagen, dass es nicht so ist, weil es zu schwer ist. Es ist, weil es langweilig ist “

Seymour Papert

Design-Aufgabe

In diesem Kapitel stellen wir einige geführte Designherausforderungen mit unterschiedlichen Schwierigkeitsgraden und unterschiedlicher zu verwendender Hardware und Software vor.

Jede Design-Herausforderung wird als Herausforderung angesehen, die mit unseren Klassenzimmern mithilfe einiger Lehrmethoden gelöst werden muss:

- *4P*: project, passion, peers & *play* (Projekt, Leidenschaft, Gleichaltrige und Spiel)
- PBL: *Project Based Learning* (projektbasiertes Lernen)
- *Flipped Classroom* (kurz: Lernen zu Hause, Anwenden in der Schule)

Hier einige kurze Definitionen der einzelnen Methoden, die wir auf unsere pädagogischen Herausforderungen anwenden werden:

4P: Project, Passion, Peer & Play

Mitchel Resnick stellte mit dem Buch *Kindergarten: LifeLong* das Konzept des kreativen Lernens vor. Ein kreativer Lernender (Student) muss das Problem lösen und sich dabei konzentrieren. In der Schule können wir Probleme natürlich nicht drastisch lösen, aber wir können einen oder mehrere Prozesse lernen, die das Lernen ermöglichen.

Projekte (projects):

Ein Projekt ist ein Prozess, der eine Reihe von Problemen lösen kann, aus denen sich das Problem zusammensetzt, das wir lösen möchten. Wir brauchen kein Konzertprojekt, wir können auch die Theorie verwenden, um die Welt zu verstehen und einen Vorschlag zur Verbesserung eines oder mehrerer Probleme zu finden.

Leidenschaft (passion):

Es steht für Engagement. Wir können mit den positiven Gefühlen unserer Schüler arbeiten, wenn wir Technologie einsetzen.

Gleichaltrige (peers):

Zusammenarbeit ist von grundlegender Bedeutung für den Lernprozess. Sie lehrt Gleichaltrige, tauscht Informationen aus und hört

unseren Kollegen zu. Mit Klassenkameraden können wir unser Wissen und unser Bewusstsein für unsere Freunde und das Publikum verbessern. Dank dieses Teils können wir Empathie entwickeln.

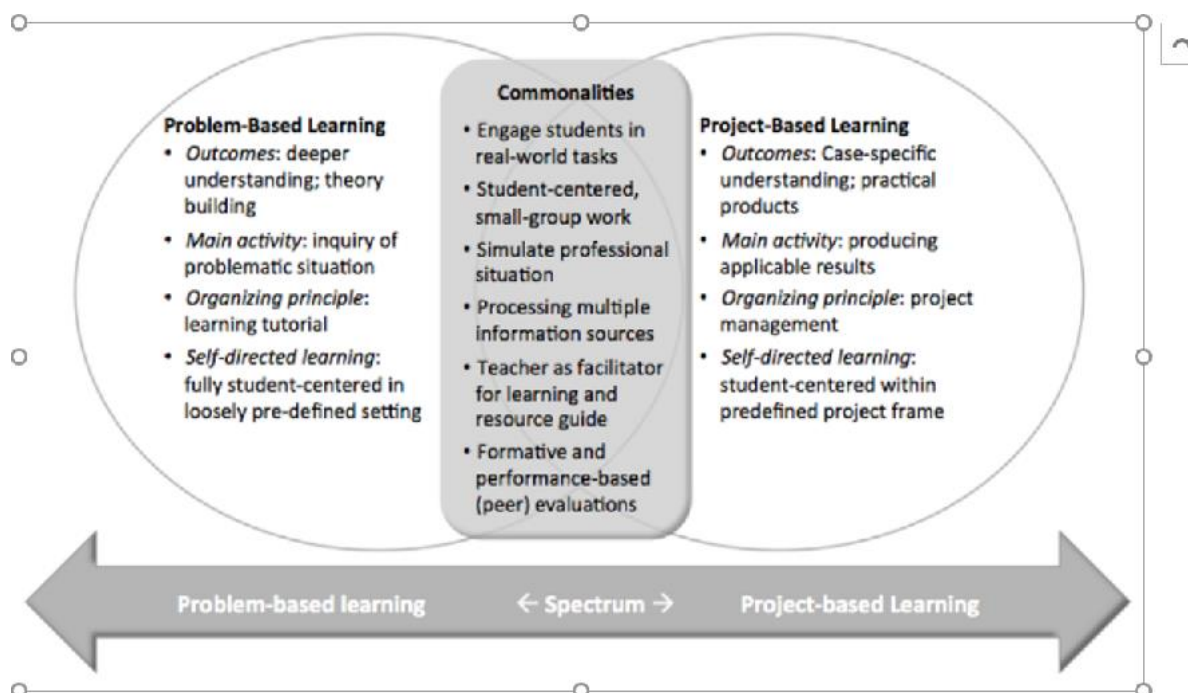
Spiel (play):

Wir brauchen Spaß während unseres Lernprozesses. Eine einfache Umgebung und glückliche Gesichter ermöglichen es den Menschen, besser und schneller zu lernen.

Projektbasiertes Lernen

Um kreative Lernende zu ermutigen, ist PBL eine Methodik, die auf dem Studium eines Projekts basiert, das alle Disziplinen einbeziehen kann. Dies bedeutet, dass der Schüler ein reales Szenario plant, in dem Ideen mit der Lösung realer Probleme verbunden werden können. Dank der PBL-Methodik sind die Schüler engagierter und können die Zusammenarbeit und Soft Skills verbessern, die Kreativität fördern und das Lernen zum Spaß machen!

In dieser Wikipedia-Darstellung wird projektbasiertes Lernen deutlich:



Flipped Classroom (Umgekehrtes Klassenzimmer)

Diese Methode basiert auf Blended-Learning-Methoden und prognostiziert eine Veränderung des Klassenzimmers, in dem die Schüler verschiedene Fächer studieren und ihrem Klassenkameraden beibringen müssen, was sie lernen sollen. In diesem Prozess kann der Lehrer seine Rolle ändern und eine Art Coach, Planer der Lernenden werden und sie dazu bringen, mit ihrem Arbeitsteam zu entdecken und neue Dinge zu präsentieren.

Design Challenge 1: Zauberstab

Robotics kit: Lego Wedo 2 or Arduino or Microbit (in this case is developed for Lego Wedo 2)

Software: Scratch 3.0 and Lego Digital Designer Age: from 10 to 15 yo

Vorbereitung

Wir müssen einen Zauberstab mit Lego WeDo 2 Teilen zusammenbauen, das ist sehr einfach und erfordert keine Handbücher. Aus diesem Grund kann der Unterricht aus pädagogischer Sicht ausgehend vom Projekt des Zauberstabs sowohl auf Blättern als auch mit der kostenlosen Software Lego Digital Designer organisiert werden.

In dieser Phase müssen die Schüler:

1. Den Zauberstab auf Papier oder dem Computer entwerfen (via LDD)
2. Den Zauberstab mit Lego WeDo mit dem Neigungssensor entwerfen

Ziel der Übung ist es, die Aufmerksamkeit und Beteiligung der Schüler durch gemeinsame Erzähltechniken (Geschichten über Zauberer) zu wecken. Dank der Verwendung des Stabes ist es einfach, die Verwendung des Neigungssensors einzuführen und die Verwendung des "if" bei Scratch zu verstehen.

Storytelling/Geschichten erzählen

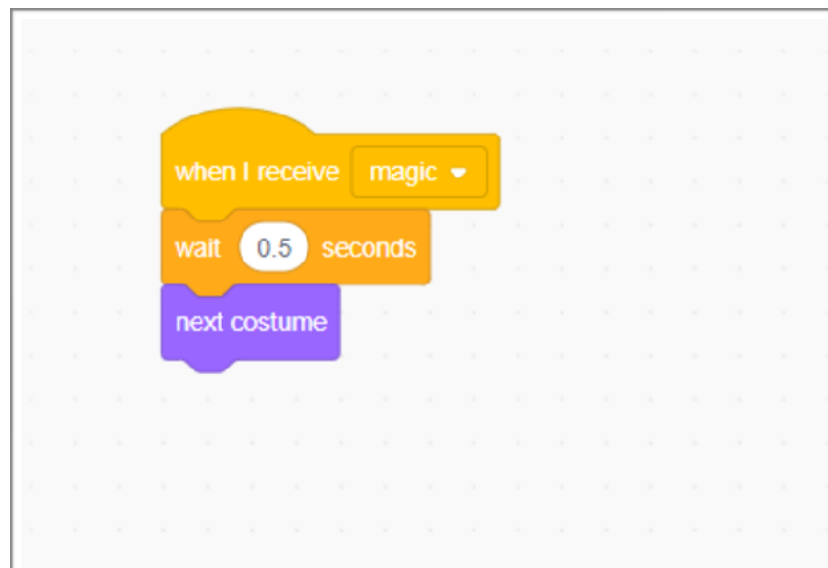
Die Übung bietet sich an, um viel am narrativen Teil zu arbeiten. Kinder können Teil einer Geschichte werden, in der es Protagonisten-Magier gibt. Sie können Zauberhüte vorbereiten und sie während des Unterrichts für Kinder aufsetzen. Sie können an Märchen arbeiten, die Magiern gewidmet sind, indem Sie die Konstruktion

und die nächsten Programmierzauber als Phase der gesamten Erzählung verwenden.

Programm 1

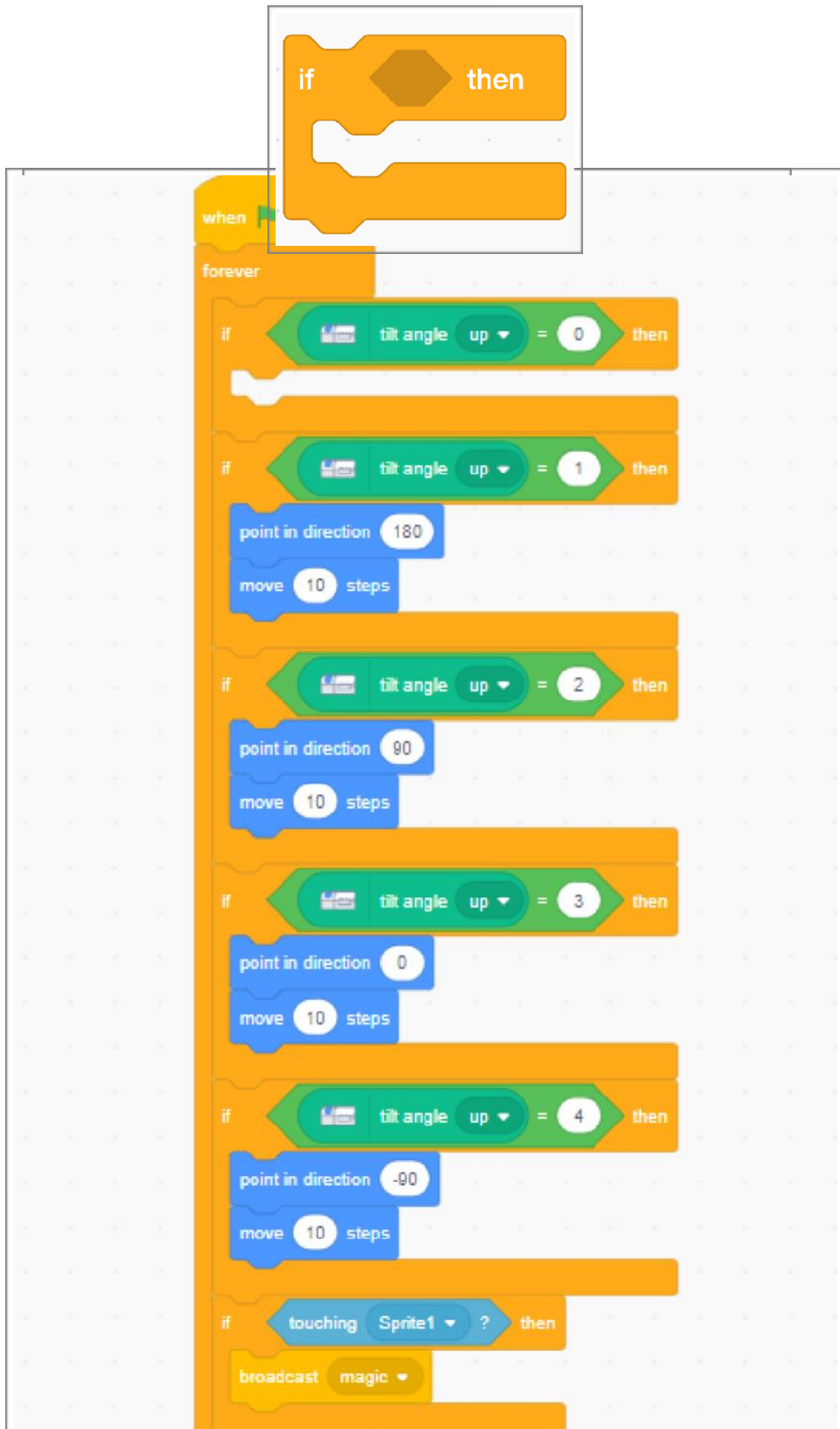
Das erste Programm wird nur verwendet, um den Zauberstab zu verwalten und das Kostüm eines in Scratch eingefügten Sprites zu ändern, wenn der Zauberstab bewegt wird.

Wenn Sie beispielsweise die Position des Stabes von vertikal nach horizontal ändern, ändert sich die Form des Sprites. Für die Ausführung dieses Programms sind keine besonderen Voraussetzungen für die Programmiersprache erforderlich. In dieser einfachen Übung können die Schüler in der Informatik 2 wichtige Blöcke entdecken:



Mit diesem Block können Sie eine Schleifenwiederholung bis zur Endlosschleife erstellen. In diesem Fall bedeutet dies, dass wir jedes Mal kontrollieren, wo sich die Position des Zauberstabs befindet.

Der zweite wichtige Block ist IF-THEN, dank dessen wir die Position des Sensors vergleichen können und wenn der Winkel größer ist, können wir das Bild auf dem Bildschirm ändern.

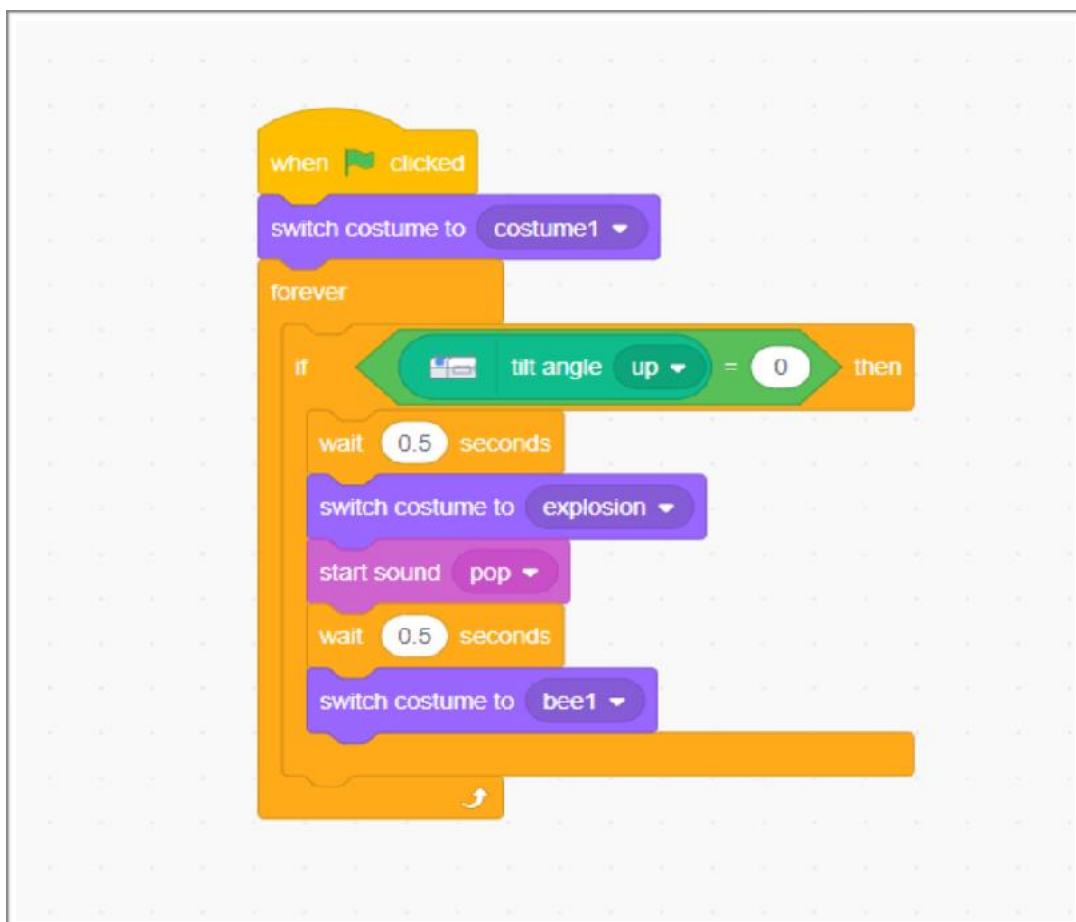


Dank dieses einfachen Programms können die Schüler das Potenzial des Befehls "IF" verstehen und ein einfaches Flussdiagramm erstellen, das ein oder enthält

Weitere Optionen basierend auf dem TILT-Sensorstatus des Lego WeDo.

Programm 2

Das zweite Programm beinhaltet keine Erweiterung des Lernhorizonts, sondern nur die Festigung der neu erlernten Konzepte. Es ist vorgesehen, dass zum Zeitpunkt der Ausführung der Magie ein Klang und ein grafischer Effekt vorhanden sind, der die Magie ankündigt. Aus didaktischer Sicht gibt es keine neuen Konzepte, sodass Lehrer diese Übung verwenden können, um zu verstehen, ob die Schüler das Potenzial der Software verstanden haben.



Darüber hinaus ermöglicht dieser Teil des Programms eine starke Personalisierung durch den Schüler (nicht nur den Klang, sondern auch das Design des magischen Effekts). Dort ermöglicht die Personalisierung eine stärkere Einbeziehung der Schüler und ein größeres und unmittelbareres Lernen.

Programm 3

Bei den ersten beiden Übungen können Sie zahlreiche Variationen des Themas vornehmen, die für sie gleichermaßen schwierig sind. Um das Programmierniveau zu erhöhen, können Sie auf diese Weise die Anforderung an den Schüler ändern: "Erstellen Sie einen Zauberstab, der im Video den Stern steuert. Alles, was vom Zauberstab berührt wird, wird verwandelt."

Diese Anfrage beinhaltet die Erstellung neuer Algorithmen:

1. Bewegung eines Sprites gemäß der Position des TILT-Sensors (4 identifizierte Zustände
2. Verknüpfen eines Programmierflusses mit einem anderen (in einem anderen Sprite vorhanden) mit dem Befehl "Broadcast".

Design Challenge 2: Fisch-Spiel mit Neigungssensor

Ziel ist es, das gesamte bisher erworbene Wissen in einem komplexen Programm einzusetzen, das die Interaktion vieler Sprites beinhaltet. Die verwendete Variable (bereits in Scratch vorhanden) heißt "Dimension".

Die Struktur der Speisekammer unterscheidet sich geringfügig von den vorherigen, um die Lehrer in Scratchs Programmierung zu begleiten: Es werden keine Übungen absolviert, sondern nur eine Schritt für Schritt beschrieben.

Zweck

Das Ziel ist es, ein Spiel zu erstellen, in dem ein Raubfisch (durch den Neigungssensor gesteuert), die nur die kleinsten Fische fressen kann. Bei jedem gefressenen Fisch nimmt der Raubfisch an Größe zu. Der Feind ist ein viel größerer Fisch, ein Super-Raubfisch, welcher vermieden werden muss bis der Raubfisch selber so groß ist, dass kein Risiko mehr besteht. Erst dann ist das Spiel beendet.

Storytelling

Um dieses Videospiel zu erstellen, müssen Sie einige Elemente einführen, die sich auf die Handlung einer Geschichte beziehen. Wir müssen die Figur des Protagonisten, den Zweck des Spiels (andere Fische essen) und die Figur des Antagonisten, des Feindes, vorstellen, der das Erreichen des Ziels behindert.

Dies kann ein erster Schritt zur tatsächlichen Einführung einer Geschichte sein, in der alle Akteure klar definiert sind: Protagonist, Helfer des Protagonisten, Antagonist, Helfer des Antagonisten, magisches Objekt, Ziel zu erreichen.

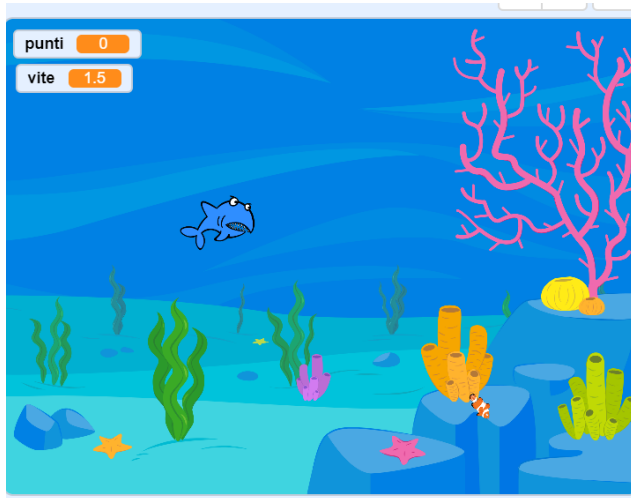
Das Videospiel

Um das Videospiel realisieren zu können, wird die Programmierung jedes Schritt-für-Schritt-Elements und ein Bild des fertigen Spiels präsentiert.

Hintergrund (Bühne)

Wie in der Speisekammer 1 können Sie einen Hintergrund importieren und mit dem Zeichnungseditor bearbeiten oder, falls vorhanden, einen neuen erstellen, wie dies für Cloud-Sprites getan wurde. In dieser Übung ist es nicht erforderlich, keine Hintergrundaktion zuzuordnen, aber es ist möglich, die Abwechslung verschiedener

Hintergründe vorherzusagen. In diesem Fall müssen sie zu geeigneten Zeiten angezeigt werden, um das Erscheinungsbild zu planen.



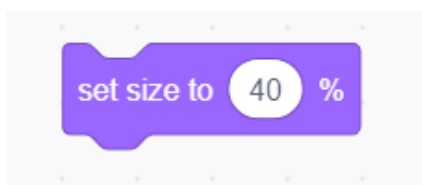
Sprite Protagonist - Bewegung

Um das Sprite des Protagonistenfischs zu steuern, fügen Sie einfach das Skript in das Hauptskript ein, das im Übungsprogramm 3 erstellt wurde, oder ein anderer Skripttyp kann sein:

Sprite Protagonist - Variable Initialisierung "Dimension"

Dieses Konzept liegt jedem Code zugrunde, der die Initialisierung der Variablen des Programm. In diesem Fall wird die Variable, die wir verwenden werden, bereits von Scratch definiert (wir müssen sie nicht selbst erstellen) und heißt Dimension. Die Initialisierung erfolgt mit dem Befehl:

"Set size to %."



Sprite ‚kleiner Fisch‘ – Bewegung und Interaktion

Im Videospiel gibt es viele kleine Fische, die alle dasselbe Skript verwenden. Erstellen Sie einfach einen einzelnen Sprite "kleinen Fisch" und duplizieren Sie ihn dann so oft wie nötig. Alle Sprites im Videospiel müssen über dieselbe Schaltfläche aktiviert

werden, in diesem Fall über das "Leerzeichen" (dies sind Aktionen, die "gleichzeitig" aktiviert werden).

Wenn Sie ein Videospiel programmieren, müssen Sie denken, dass der Status "show" oder "hide" nicht initialisiert wurde, insbesondere für den Fall, dass Sie während des Spiels auch den Befehl "Hide" verwenden möchten.

Zu Beginn müssen wir das Sprite anzeigen und seine anfängliche Größe auf 20% einstellen.

Anschließend programmieren wir die Bewegung und Interaktion. Die Bewegung des Fisches erfolgt entlang einer Linie, Sie müssen jedoch vorsichtshalber den Befehl "Bounces, wenn Sie die Kante berühren" eingeben, um die Kontinuität der Bewegung zu gewährleisten. Wenn sich das Sprite zum Zeitpunkt des Abpralls auf den Kopf stellt, wählen Sie die Schaltfläche "Nur rechts abbiegen" und links "(wie im Bild gezeigt, ist das Schaltflächenquadrat in der Mitte mit horizontalen Pfeilen).

Sprite – Protagonist (Hauptfisch)

Um richtig zu programmieren, müssen Sie berücksichtigen, dass zu dem Zeitpunkt, an dem das Sprite des Protagonisten von einem Fisch berührt wird, die Bewertung vorgenommen werden muss, wenn der Fisch größer oder kleiner ist. Wenn der Protagonist auf einen kleineren Fisch trifft (<30 in unseren Einstellungen ...), der ja mit dem Befehl "Wenn ich ... Größe xx erhalte" realisiert, reicht es aus, dem Programm vorzuschlagen, die Größe des Protagonisten um einen zu erhöhen Prozentsatz unserer Wahl (im Beispiel 10).

Sprite - Großer Fisch

Im Gegensatz zu kleinen Fischen lässt es die großen Fische zufällig an einer Position erscheinen (x, y), verwaltet durch die Kombination von zufälligen Zahlen, die mit dem entsprechenden Skript (wie gezeigt) eingefügt werden. Zufällig ist auch die Bewegungsrichtung des Fisches. Die anfängliche Größe des großen Fisches beträgt 70%. Um eine abwechslungsreiche Bewegung im gesamten verfügbaren Raum zu gewährleisten, können Sie den Fisch so einstellen, dass er zu jeder Kante zurückspringt und eine zufällige Geschwindigkeit

aufweist (die Anzahl der Schritte wird zufällig von 1 bis 15 generiert). Bei kleinen Fischen muss der Protagonist eine nominierte Nachricht als Größe senden (in diesem Fall 70).

Sprite Protagonist – Nachricht vom großen Fisch

Wenn der Protagonist die Nachricht vom großen Fisch erhält, muss er verstehen, dass er ‚gefressen‘ wird oder dem Gegner gegenüber treten muss. Diese Bedingung ist erfüllt, wenn seine Größe mindestens 60% beträgt. Wenn die Bedingung wahr ist, endet das Videospiel mit einer extremen Vergrößerung des Protagonisten. Falls der Protagonist kleiner als der Super-Raubfisch ist, endet das Spiel mit einer schriftlichen Nachricht, repräsentiert durch ein Sprite, das aktiviert wird, wenn es die ‚game over‘-Nachricht erhält.

Sprite Game Over - Script

Game Over ist ein vom Benutzer entwickeltes Sprite. Das Wort Game Over wird nur aktiviert, wenn es die Game Over-Nachricht empfängt, während es sich zu Beginn des Programms versteckt.

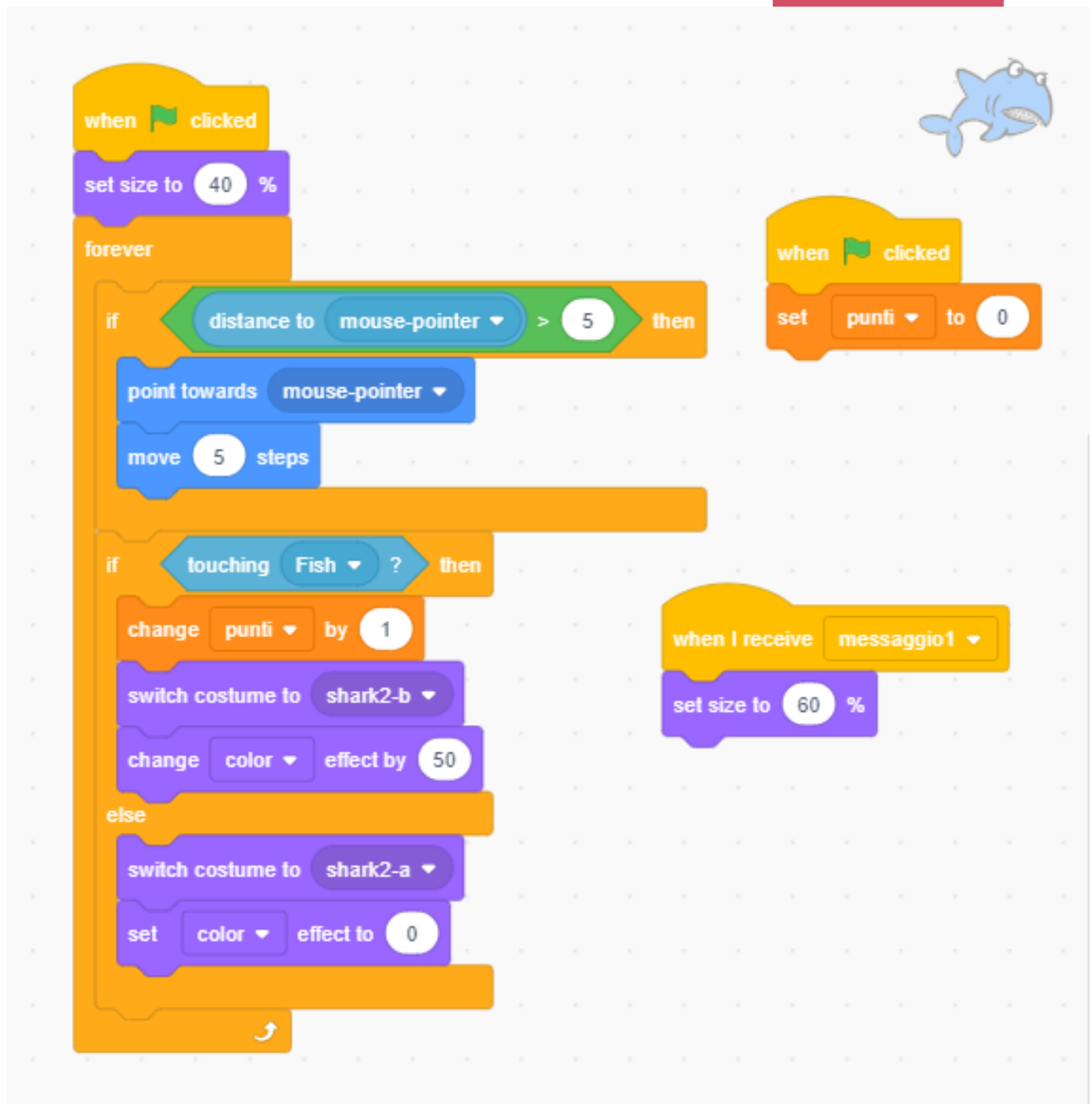
Sprite Big Fish – Victory/Sieg

Der Protagonist - im Falle eines Sieges - sendet die Nachricht "gewonnen" an alle Sprites. Wenn die Nachricht das Sprite des Großen Fisches erreicht, muss dieses verschwinden. Falls gewünscht, kann ein Sprite "Du hast gewonnen" hinzugefügt werden. Im vorgeschlagenen Programm wird der Raubfisch riesig und schwimmt weiter.

Finale (recap)

Die Umsetzung dieser Software ist komplex. Um den Messwert nicht zu beeinträchtigen, haben wir keine didaktischen Notationen wie in den vorherigen Handouts in den Text eingefügt. Der Lehrer, der ein Labor leitet, das der Realisierung dieses Spiels gewidmet ist, muss einige Schlüsselkonzepte klären.

Ein Beispiel für ein vereinfachtes Skript könnte sein:



Die Variable "Dimension"

Scratch ermöglicht das Erstellen neuer Variablen, einige sind jedoch bereits eingefügt und definiert. "Dimension" ist das klassische Beispiel für eine lokale Variable, die im Gegensatz zu anderen Variablen, die von mehr Sprites verwendet werden können, nur für das Sprite gelten kann, für das sie verwendet wird.

Die Verwendung dieser Variablen erlaubt es beispielsweise nicht, sie in den

Skripten anderer Sprites abzurufen.

Dank der "Dimension" kann das Konzept der Variablen erklärt werden. In diesem Spiel ist es klar, dass die Änderung der variablen Dimension den Erfolg der nicht oder nicht verringert

Spiel. Sehr oft, wenn die Variablen in der Schule eingeführt werden, riskieren wir posthume Beispiele, die ad hoc erstellt wurden und nicht sehr funktionale oder zu komplexe Beispiele sind. Die Verwendung in diesem Videospiel bietet jedoch einen weichen Ansatz für die Fragenvariablen.

Zufällige Zahlen

Um Fische zu erzeugen, die sich mit zufälliger Geschwindigkeit bewegen, müssen Sie den Scratch-Zufallszahlengenerator verwenden. Auf diese Weise können die Schüler verstehen, wie diese "Generatoren" in der Computerwelt verwendet werden.

Senden - empfangen

In diesem Programm ist die Kommunikation zwischen Sprite unerlässlich. Dies ermöglicht es uns, parallele und Programmierabläufe einzuführen, um komplexe vorbereitende Flussdiagramme zu erstellen.

Design Challenge 3: Oracle

Software: Scratch 3

Diese Herausforderung ist eine Verbindung zur Welt der Sprachassistenten, die ein wichtiges Thema auf dem Markt ist und bei der alle wichtigen Unternehmen ihre Sprachassistenten produzieren.

Zu Beginn des Unterrichts können wir unsere Schüler inspirieren, beginnend mit einem Science-Fiction-Film, 2001 Space Odyssey, und danach mit einem Überblick über einige wichtige Sprachassistenten. In Kubricks Film 2001 Space Odyssey können wir HAL2001 entdecken, einen Assistenten, der mit Astronauten interagieren und das Raumschiff steuern kann. Danach können wir auch die Welt von Star Trek zitieren, in der alle Besatzungsmitglieder mit dem „Schiff“ sprechen und um Hilfe, Vorschläge und Service bitten können.

Wir können uns mit der realen Welt verbinden und alle (von fast allen) realen Produkten auf dem Markt präsentieren. Hier einige Beispiele:

Alexa (Amazon)



Google Home (Google)



Siri



Alle diese Sprachassistenten können Ihre Fragen beantworten, indem sie Informationen im Internet überprüfen und die wichtigsten Apps verwenden, um zu Online-Musikplattformen, Cloud-Diensten und Online-Märkten zu gelangen.

Aus ethischer Sicht kann man mit den Studenten die Fallstudie diskutieren, die sich auf die Verwendung der Stimme eines bekannten Sängers bezieht (wie es im schwarzen Spiegel geschieht). Die Verwendung der Stimme eines berühmten Sängers führt zu einigen Fällen des Missbrauchs dieses Sprachassistenten, der die wirklichen Freunde der Teenager ersetzt.

Die Einführung eines Sprachassistenten kann auch zur Einführung von KI-Konzepten verwendet werden.

Sprachassistent und Internet der Dinge

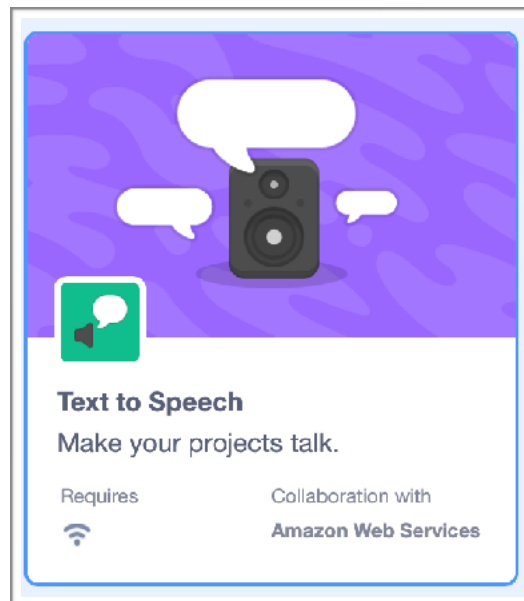
Wir stellen die Verwendung von Sprachassistenten allein vor, aber in naher Zukunft könnte all dieser Assistent die Hauptschnittstelle für alles sein, was in unserem täglichen Leben mit dem Internet verbunden ist.

Nach dieser Einführung erfahren Sie, wie Sie mit Scratch 3 einen Sprachassistenten programmieren:

In Scratch ist es nicht möglich, die Spracherkennung zu verwenden, aber wir können ein Programm erstellen, in dem die Interaktion mit der Stimme vom Computer und dem Text des menschlichen Benutzers erfolgt.

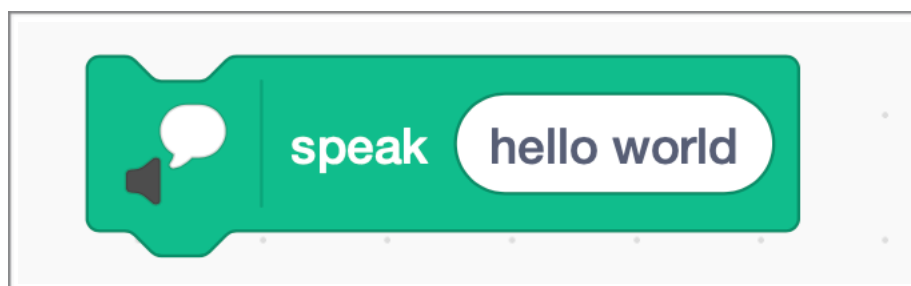
Um einen Sprachassistenten, eine Art Orakel, zu programmieren, können wir den Befehl „Text to Speech“ verwenden.

Um diese Funktion, Scratch, als Erweiterung zu verwenden, müssen wir auf "Weitere Erweiterungen" klicken und können dieses Bild auswählen:



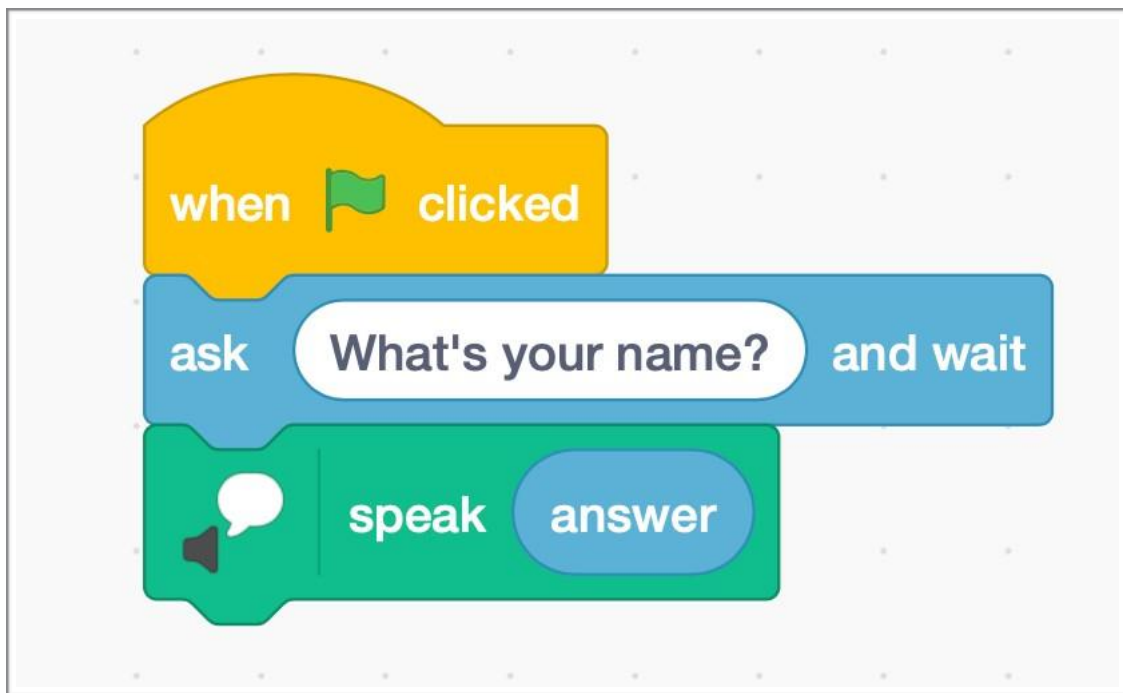
Was ist Text zu Sprache (text to speech)?

Diese Blöcke ermöglichen es dem Computer, zu lesen, was wir in einem Textformat schreiben, so dass die Interaktion interessant sein wird.



Wie wir diese Blöcke verwenden

Dank dieses Programms können wir den Namen fragen und per Tastatur schreiben. Mit Text zum Sprechen kann das Programm den Namen lesen. Wir können das Programm verbessern, indem wir einige Dinge fragen und unterschiedliche Antworten erhalten.



Micro:bit



Scratch mit micro:bit

Micro: bit ist ein Mikrocomputer im Taschenformat, mit dem Kinder lernen, mit Technologie zu programmieren und zu erstellen. Es verfügt über viele Funktionen, darunter eine LED-Anzeige, Tasten und einen Bewegungssensor. Es kann damit verbunden und mit Scratch, MakeCode und Python programmiert werden.

Das BBC micro: bit enthält einen Beschleunigungsmesser, der erkennen kann, ob es geschüttelt wird oder in welche Richtung das micro: bit gehalten wird. Beschleunigungsmesser erkennen nicht nur, ob sie auf dem Kopf stehen, sondern können auch einige der auf sie einwirkenden Kräfte erkennen. Das micro: bit verfügt über einen eingebauten Temperatursensor, der die aktuelle Temperatur des Geräts in Grad und Celsius erfassen kann. Die vollständige Liste der Funktionen finden Sie in der folgenden Tabelle:

Feature	Description
2 buttons	Programmable action push buttons
25 LED lights	Can be individually programmed to show shapes, text or numbers
USB connector	Connect to a computer for power or to load programs onto the micro:bit
Accelerometer	Senses if the micro:bit is being moved, tilted, shaken, or in free-fall, and at what acceleration
Compass	Detects which direction the micro:bit is facing
Processor	Where the program is executed
Radio	Communicate with other micro:bits for multiplayer games
Bluetooth antenna	Wirelessly sends and receives signals to Bluetooth enabled PCs, Smartphones, or Tablets
Reset button	Restarts the micro:bit
Battery socket	Power the micro:bit using batteries
Temperature sensor	Detects the current temperature of the micro:bit in degrees Celsius
Light sensor	The LEDs on the micro:bit can also act as a light sensor to detect ambient light
Edge Connector	25 external connectors, called Pins, on the edge of the micro:bit allow you to connect to other electronics hardware, including LEDs, motors, and other sensors. These can behave as inputs or outputs.

Derzeit verfügt Scratch 3.0 nicht über Codeblöcke zur Verwendung von Kompass, Thermometer oder Lichtsensor. Außerdem werden nur 3 der 25 Pins verwendet.

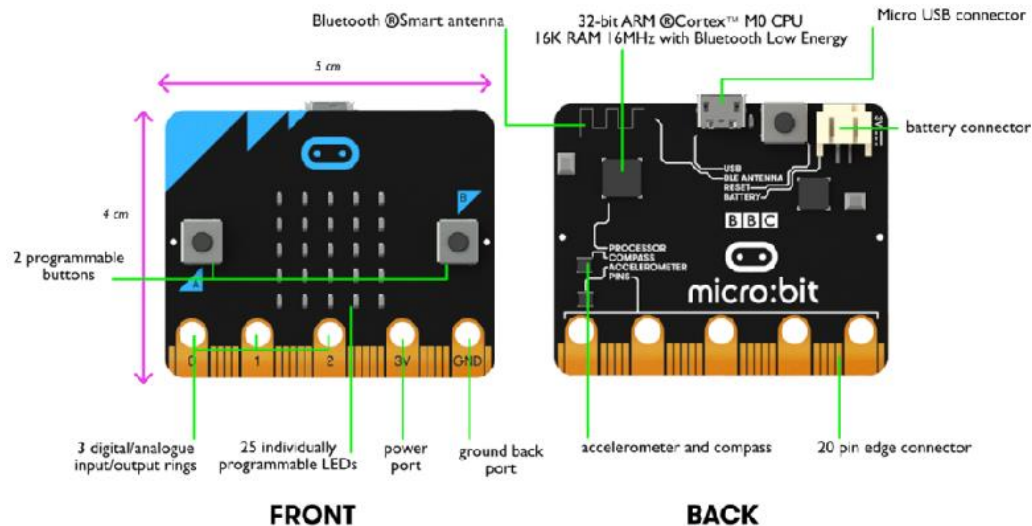


Image source: <https://www.pakronics.com.au/pages/microbit-in-australia>

Verwendung

Micro: Bit kann sowohl auf dem Desktop (Macs, PCs, Chromebooks, Linux, einschließlich Raspberry Pi) als auch auf Mobilgeräten programmiert werden. Um micro: bit mit Scratch verwenden zu können, müssen die folgenden Schritte ausgeführt werden (für Windows).

Schritt 1: Installieren Sie Scratch Link aus dem Microsoft Store oder verwenden Sie die Links von der Seite <https://scratch.mit.edu/microbit>

Schritt 2: Starten Sie Scratch Link und stellen Sie sicher, dass es ausgeführt wird. Es sollte in Ihrer Symbolleiste angezeigt werden.

Schritt 3: Ihre Sie das Mikro: Bit mit einem USB-Kabel und Ihrem Computer an. Das micro: bit wird auf dem Computer als Laufwerk mit dem Namen 'MICROBIT' gehört.

Schritt 4: Laden Sie die Scratch micro: bit HEX-Datei von <https://scratch.mit.edu/microbit> herunter

Schritt 5: Entpacken Sie das Archiv und ziehen Sie die HEX-Datei per Drag & Drop auf Ihr micro: bit

Schritt 6: Versorgen Sie Ihr micro: bit mit USB oder einem Akku.

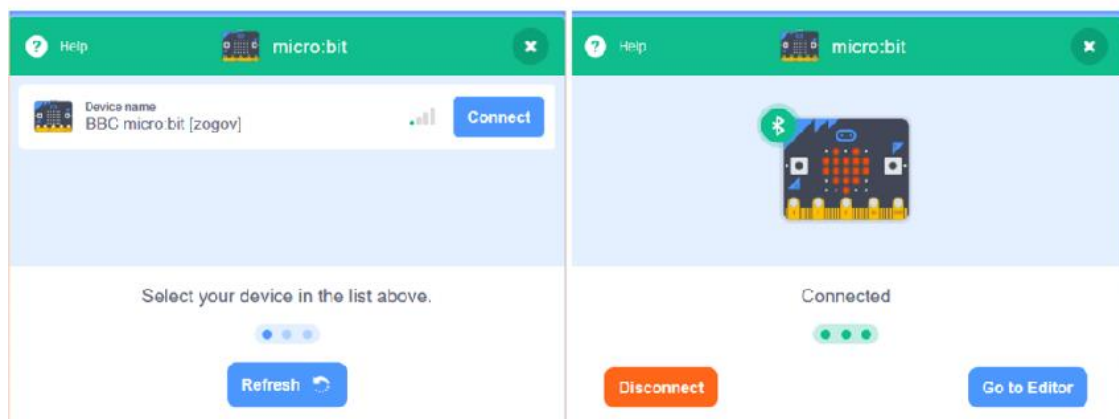
Schritt 7: Verwenden Sie den Scratch Editor

Schritt 8: Fügen Sie die micro: bit-

Erweiterung hinzu

Schritt 9: Drücken Sie Verbinden und

gehen Sie zum Editor



Design Challenge

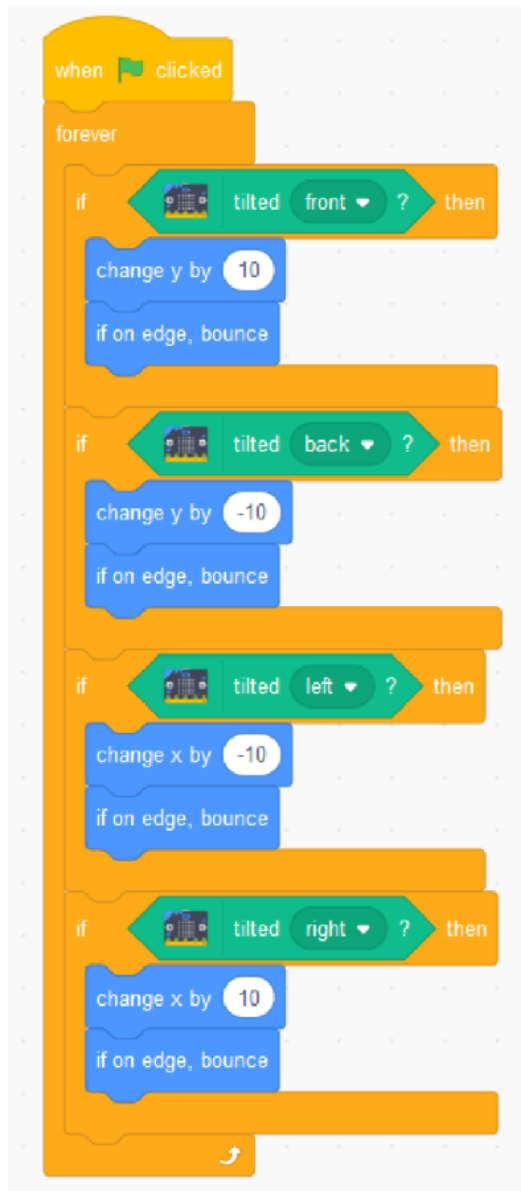
Design Challenge 1: Herzschlag

Die Anwendung (von <https://scratch.mit.edu/microbit>) wird über die beiden Schaltflächen von micro: bit gesteuert. Abhängig von der gedrückten Taste werden die beiden Herzen auf dem micro:bit angezeigt (eines kleiner und eines größer) und gleichzeitig ändert sich in Scratch die Größe eines Herzens und es werden zwei verschiedene Töne gespielt.



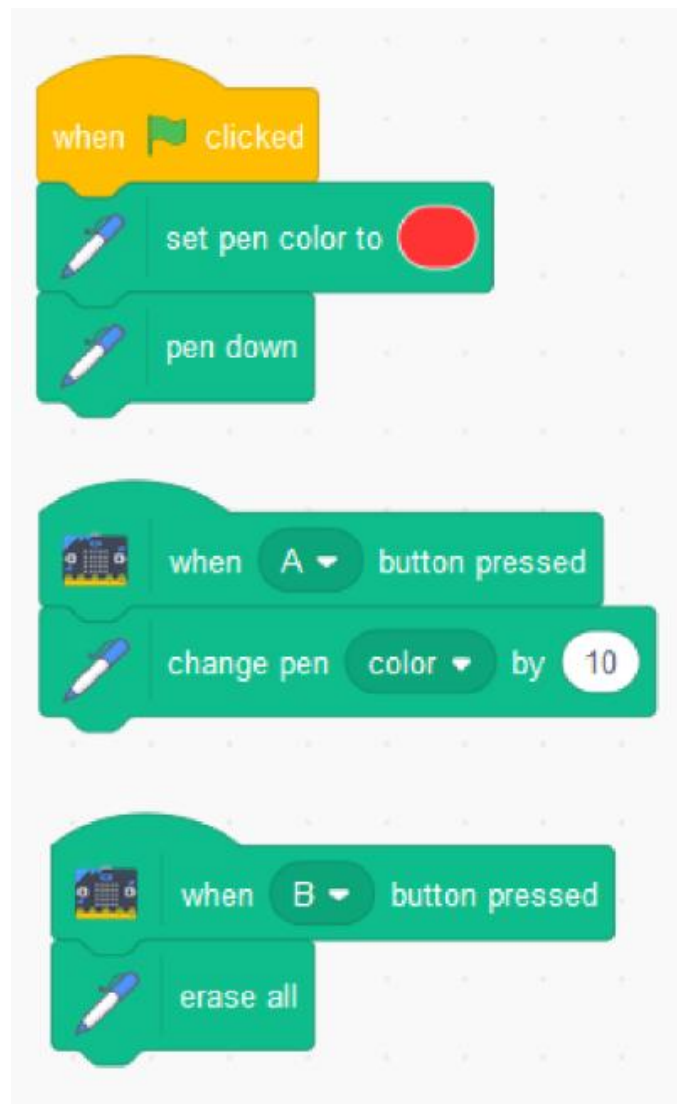
Design Challenge 2: Ball

Durch Ändern der Neigung von micro: bit kann Ball auf dem Bildschirm bewegt werden.

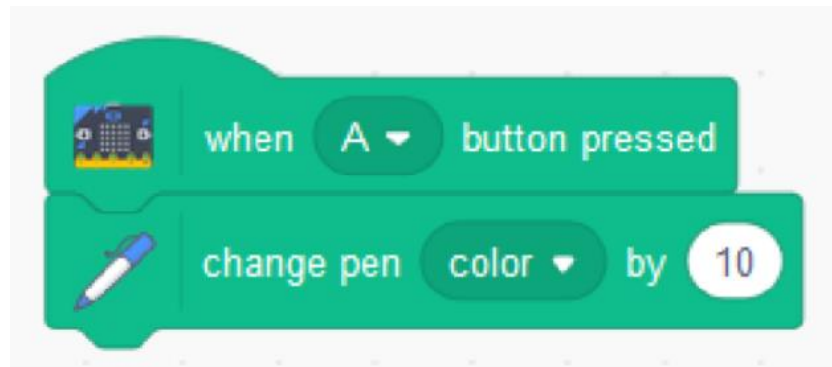


Design Challenge 3: Stift

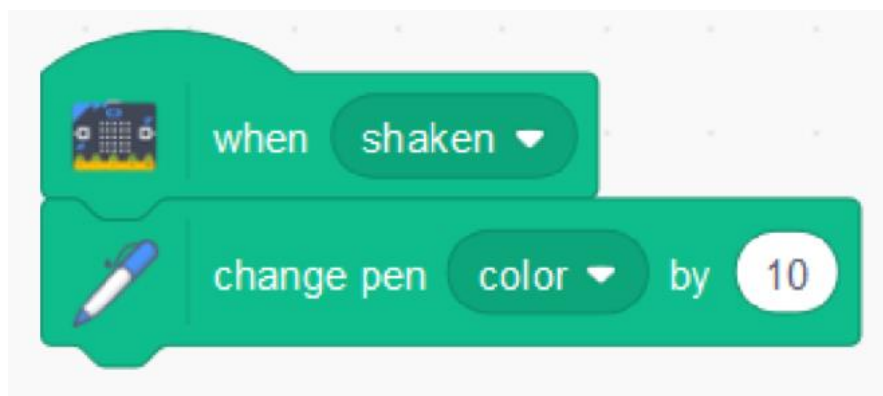
Fügen Sie die Stifterweiterung zu Scratch hinzu. Fügen Sie der vorherigen Anwendung die nächsten Skripte hinzu. Testen Sie das Ergebnis.



Wenn Sie Folgendes



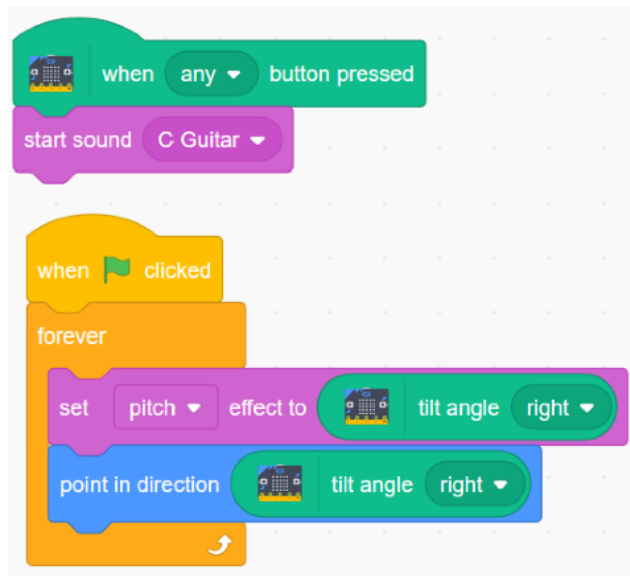
durch Folgendes



ersetzen, ändert sich die Farbe, wenn der micro:bit geschüttelt wird.

Design Challenge 4: Gitarre spielen

Die Anwendung gibt einen Ton wieder, wenn eine micro:bit-Taste mit einem Tonhöheneffekt gedrückt wird, der von der Neigung des micro:bit abhängt.

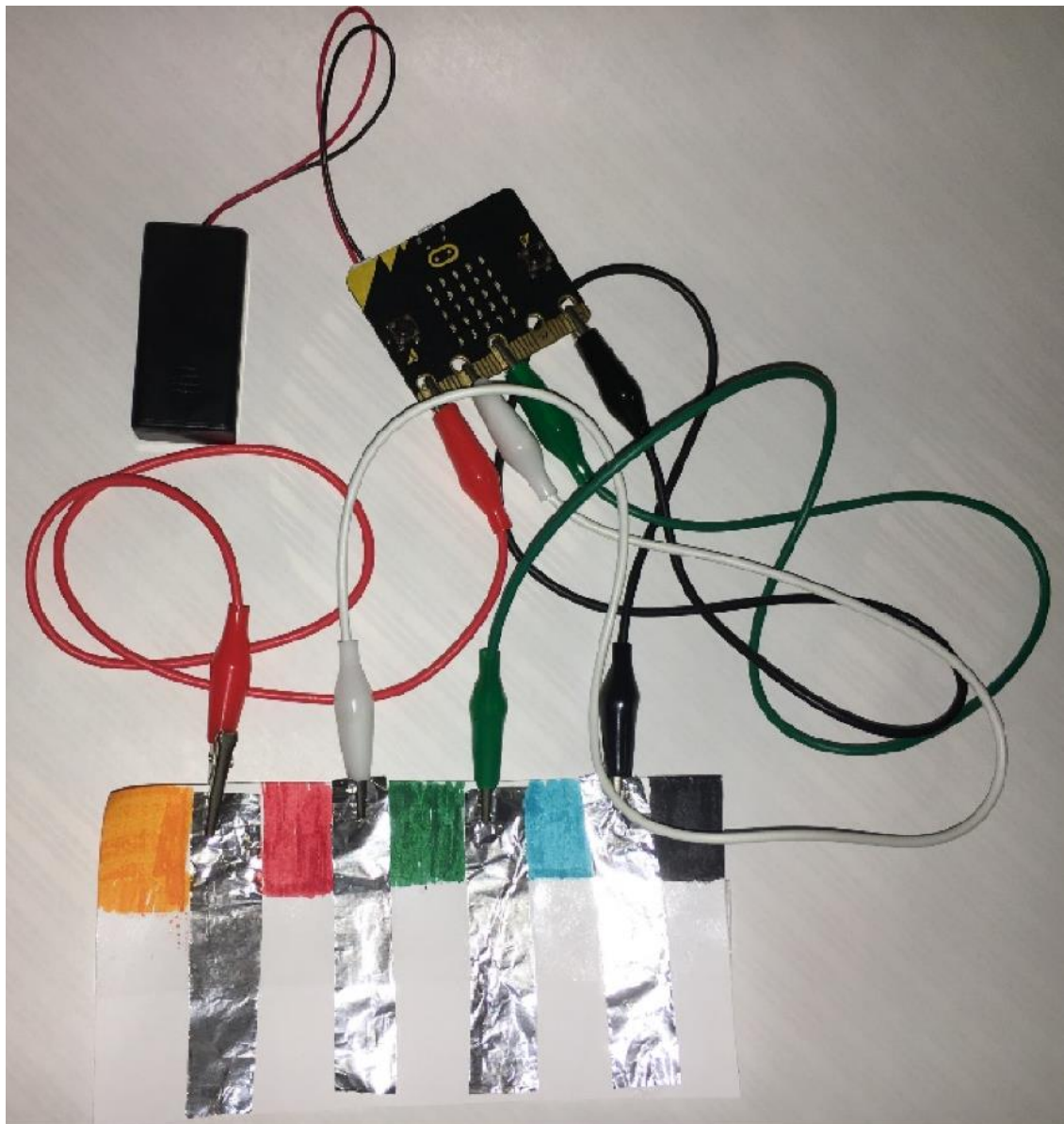


Design Challenge 5: ein selbstgebasteltes Klavier

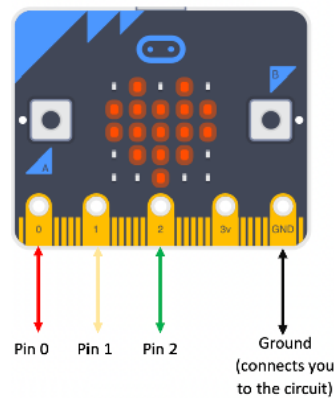
Um die folgende Anwendung zu bauen (inspiriert von [https:// microbit.org/scratch/](https://microbit.org/scratch/)) braucht man den micro:bit-Chip, vier Kabel mit

In order to build the next application (inspired by an example from [https:// microbit.org/scratch/](https://microbit.org/scratch/)) you will need the micro:bit chip, 4 cables with Krokodilklemmen, Alufolie, eine Schere, Kleber und Papier.

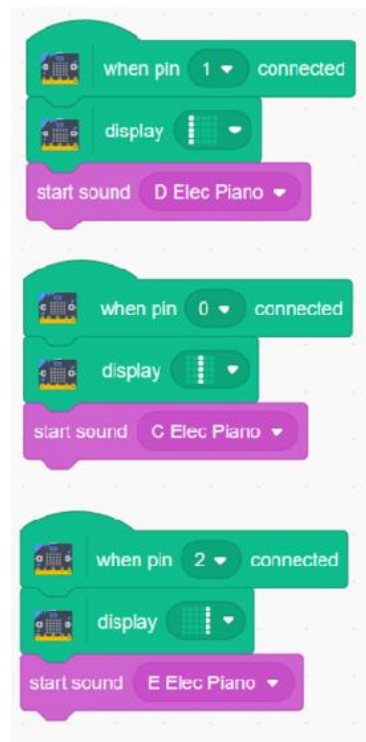
Bauen Sie ein Klavier wie auf dem nächsten Bild. Verbinden Sie jeden Stift von 0 bis 2 und die Masse mit einem Alufolienband.



Das Schema dieses Klaviers ist im folgenden Bild dargestellt:



Schreiben Sie die Skripte:



Testen Sie die Anwendung durch Berühren des Klaviers (Alufolienbänder) (ein Finger sollte das mit Masse verbundene Alufolienband berühren, um den Stromkreis zu schließen).

Weitere Challenges

1. Bauen Sie ein "Schere-Stein-Papier"-Spiel. Die Anwendung soll per Zufall einen Stein, Papier oder eine Schere anzeigen, wenn der micro:bit geschüttelt wird.
2. Erstellen Sie Ihr eigenes tragbares Gerät mit einer Schnittstelle in Scratch. Entwerfen Sie eine Hülle / Unterstützung für Ihr Gerät und drucken Sie sie mit dem 3D-Drucker.

Quellen:

<https://microbit.org/scratch/>

<https://llk.github.io/microbit-extension/iste18/>

<https://make.techwillsaveus.com/>

<http://libraryadventuring.blogspot.com/2018/10/coding-and-making-with-bbc-microbit.html>

<http://blog.sparkfuneducation.com/five-wearable-projects-with-microbit>

<https://scratch.mit.edu/discuss/youtube/44Xo76Bbqil/>

Verweise

<https://microbit.org/scratch/>

https://diyodemag.com/education/kids_coding_scratch_30_meets_micro_bit

MakeCode für micro:bit

Microsoft MakeCode ist eine webbasierte Umgebung zum Erlernen des Codierens mit physischen Computergeräten wie dem micro:bit. MakeCode ist kostenlos und funktioniert auf allen Plattformen und Browsern.

MakeCode ist als Online-Editor erhältlich für:

- micro:bit - <https://makecode.microbit.org/>
- Circuit Playground Express - <https://makecode.adafruit.com/>
- Minecraft - <https://minecraft.makecode.com/>
- LEGO® MINDSTORMS® Education EV3 - <https://makecode.mindstorms.com/>
- Arcade - <https://arcade.makecode.com/>
- Chibi Chip - <https://makecode.chibitronics.com/> and as application for:
- micro:bit
- Adafruit
- Cue by Wonder Workshop



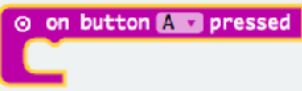



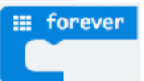

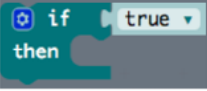




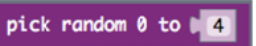
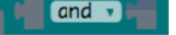





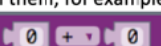
Die MakeCode for micro:bit-Anwendung bietet gegenüber dem Online-Editor einige zusätzliche Funktionen. Mit der Desktop-Anwendung kann das micro: bit direkt über USB programmiert werden, ohne dass die Datei per Drag & Drop auf das micro: bit-Laufwerk gezogen und serielle Daten direkt vom micro: bit gelesen werden müssen.

Scratch vs MakeCode für micro:bit

Micro:bit Support

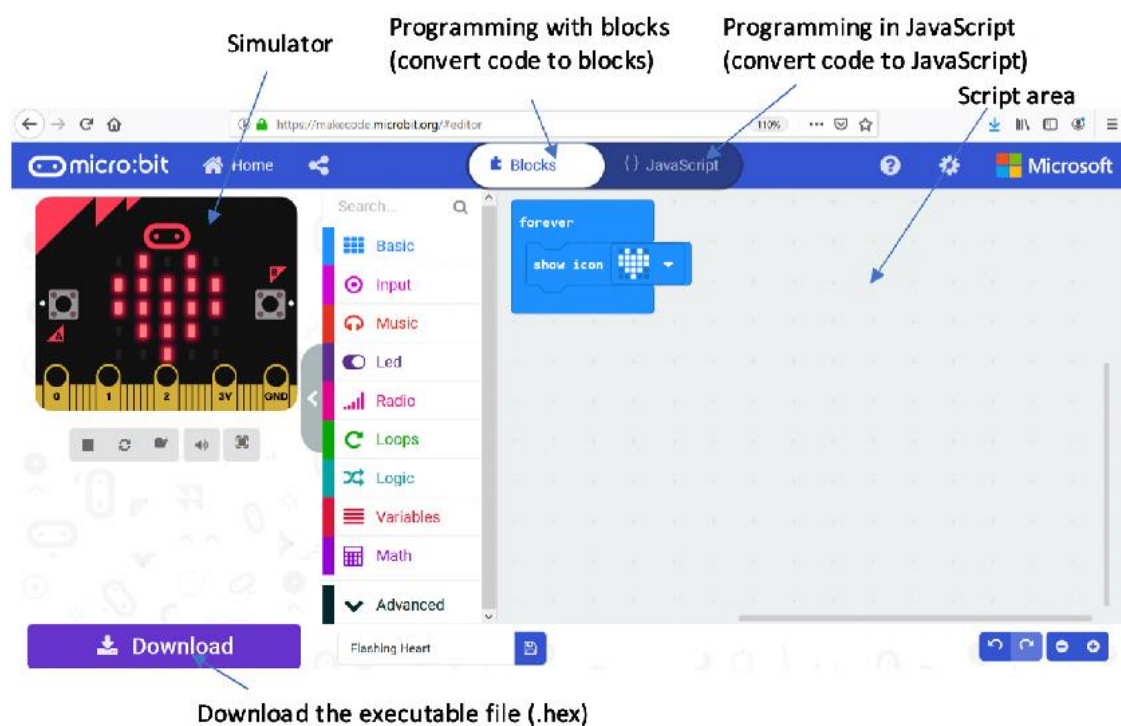
<https://support.microbit.org/support/solutions/articles/19000080171-moving-between-scratch-and-makecode>

Es wird eine unten aufgeführte Vergleichstabelle zwischen Scratch und Makecode für micro: bit veröffentlicht, um die subtilen Unterschiede zwischen mehreren wichtigen Blöcken hervorzuheben. Die Tabelle kann für den Benutzer eines der beiden Online-Editoren nützlich sein, die sich leidenschaftlich für micro: Bit, um den anderen zu verwenden.

Scratch	Makecode
Events wait for a user action, like clicking the green flag in scratch or pressing a button on the keyboard  	Input waits for a user input like pressing the A button or shaking the micro:bit 
Control is about the flow of your program tasks. In scratch you can add a forever , repeat or if block to an event to trigger it.   	Makecode breaks controls into programming concepts. Forever is it's own loop (it's triggered as soon as the micro:bit is powered on).  Repeats are found in the Loops menu of Javascript Blocks  And if blocks are found in the Logic menu 
Operators let you do arithmetic and make comparisons   	Arithmetic and random number pickers can be found in Math , whereas a comparison between something and something else is found in Logic   
Data lets you define variables that might change within your program. Here we have made a variable called item 	We can define these in the Variables menu of makecode 
Looks let you display actions on the screen, which can be done by adding a say block. The word in the box is called a string 	To show a word on the micro:bit display we can use show string , found in the Basic menu 
Blocks that let you edit them have white backgrounds that you can type in for example operators have circular input areas 	Blocks that you can edit usually look like jigsaw pieces and may already have an example in them, for example Math sums have a '0' in the block 

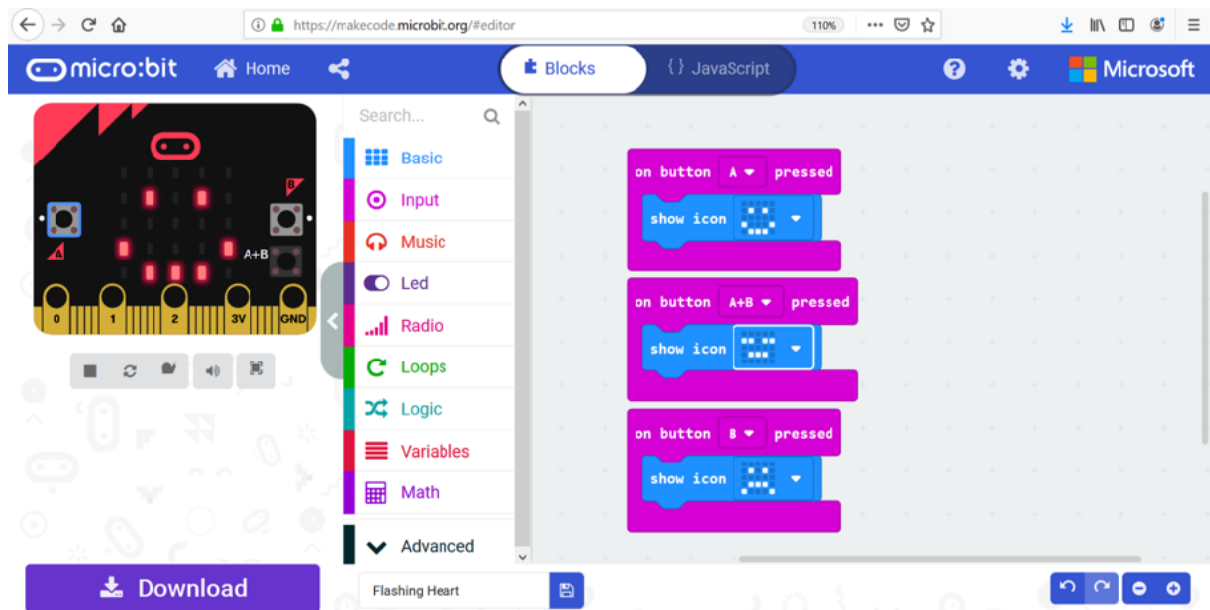
Ein großer Vorteil von Scratch im Vergleich zu Online-MakeCode besteht darin, dass OTA-Programmierung (Over-the-Air) über Bluetooth-Kommunikation möglich ist. Der große Nachteil ist, dass Scratch einen grundlegenden Satz von Blöcken für micro: bit (nur 10) bereitstellt. Für diesen Moment ist der umfangreichere Satz von micro: bit-Blöcken in MakeCode vorzuziehen.

Oberfläche



Design Challenge 1: Gesichter

Diese Anwendung zeigt ein Smiley-Gesicht, wenn die Taste A gedrückt wird, ein trauriges Gesicht, wenn die Taste B gedrückt wird, und ein schlafendes Gesicht, wenn die Tasten A und B gedrückt werden.



Im Simulationsbereich befinden sich die Tasten A, B und A + B, mit denen simuliert werden kann, was passiert, wenn die Micro: Bit-Tasten gedrückt werden.

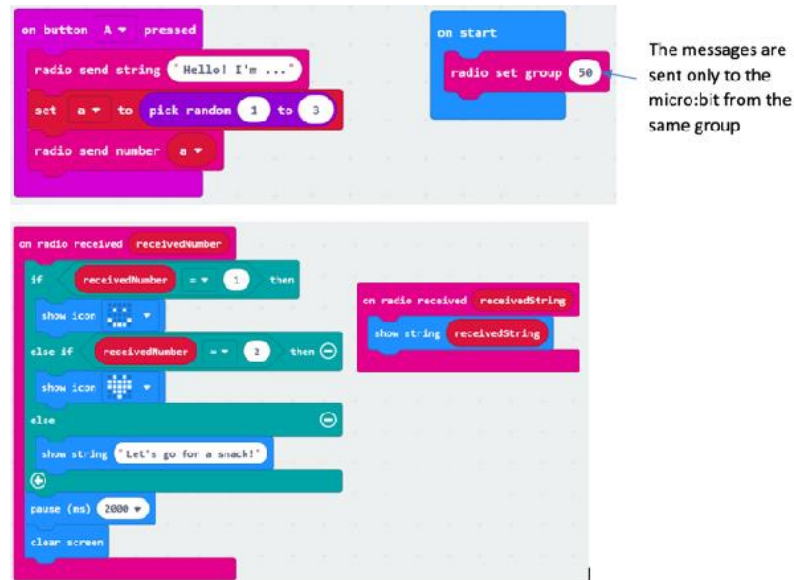
Um die Anwendung auf einen micro: bit-Chip zu stellen, müssen Sie das Skript herunterladen (als Hex-Datei). Schließen Sie das micro: bit mit einem USB-Kabel an Ihren Computer an. Suchen Sie die heruntergeladene .hex-Datei und ziehen Sie sie auf das MICROBIT-Laufwerk.

Diese Anwendung wurde von [https://makecode.microbit.org/ projects/](https://makecode.microbit.org/projects/) inspiriert. Probieren Sie auch andere Tutorien.

Design Challenge 2: Chat

Bei der nächsten Anwendung können zwei oder mehr Micro-Bit-Chips über Funk miteinander kommunizieren. Die Variablen `receiveString` und `receiveNumber` werden

aus dem Empfangsblock on radio gezogen. Die Anwendung muss auf jedes micro:bit hochgeladen werden.



Modusbeispiele mit Funkverbindung finden Sie unter:

[https:// makecode.microbit.org/projects/radio-games](https://makecode.microbit.org/projects/radio-games)

und

[https://www.instructables.com/ id/Radio-Signals-on-Microbit/](https://www.instructables.com/id/Radio-Signals-on-Microbit/).

Design Challenge 3: Musik und Licht

Die nächste Anwendung ermöglicht das Musizieren durch Variieren der Lichtintensität auf den Micro-Bit-Lichtsensoren.

Benötigte Materialien:

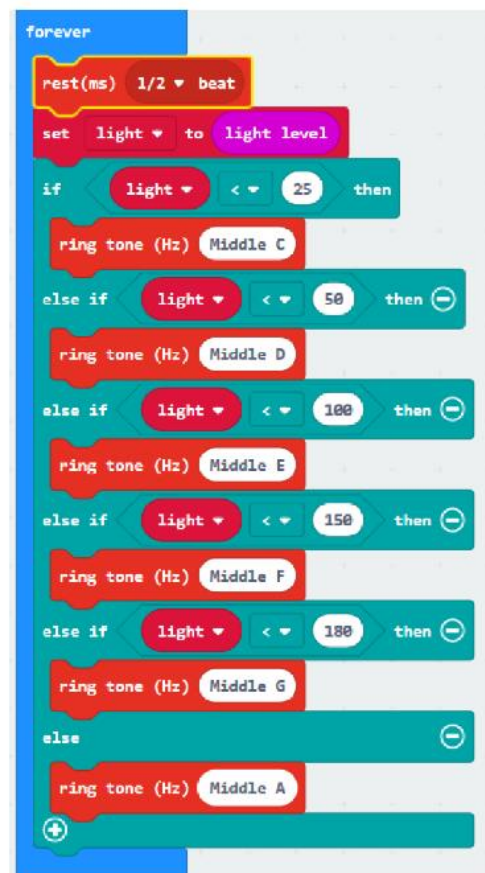
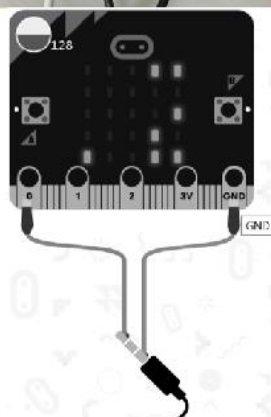
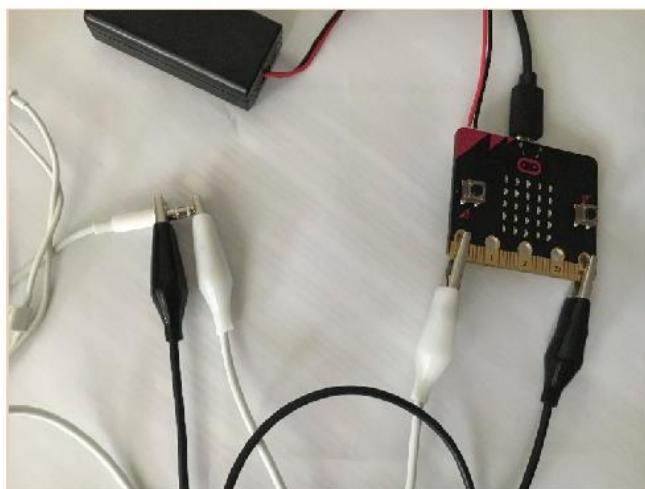
1x Micro:bit

1x Kopfhörer-Set

2 Kabel mit Krokodilklammern

1x USB-Kabel

Zu Beginn müssen Sie die Kopfhörer an das Mikro anschließen: Bit wie in den nächsten Bildern:



Definieren Sie eine neue Variable namens *light* und schreiben Sie den Code. Kopieren Sie die

.hex-Datei auf das micro: bit-Laufwerk. Die Lichtstärke liegt zwischen 0 (dunkel) und 255 (hell). Das Licht wird mithilfe verschiedener LEDs vom micro: bit-Bildschirm gemessen.

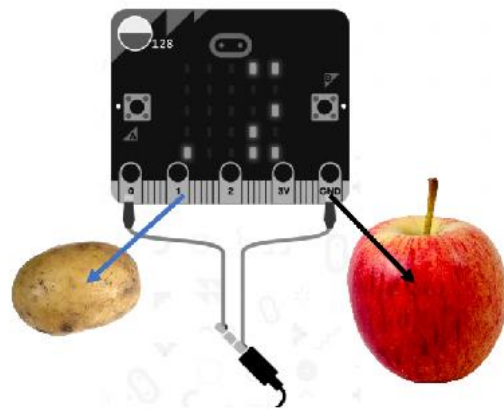
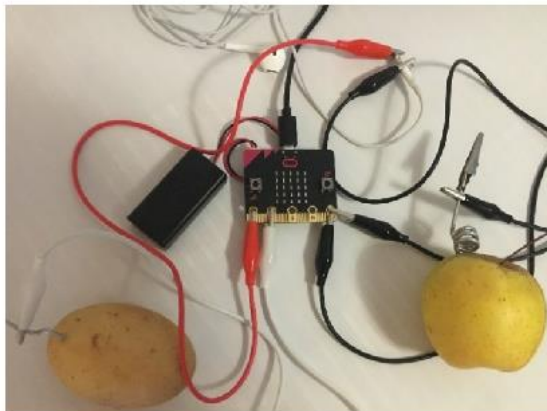
Design Challenge 4: Musik aund... Obst und Gemüse

Dies ist eine unterhaltsame Anwendung, bei der Obst und Gemüse verwendet werden, um den Stromkreis zu schließen und Musik zu machen.

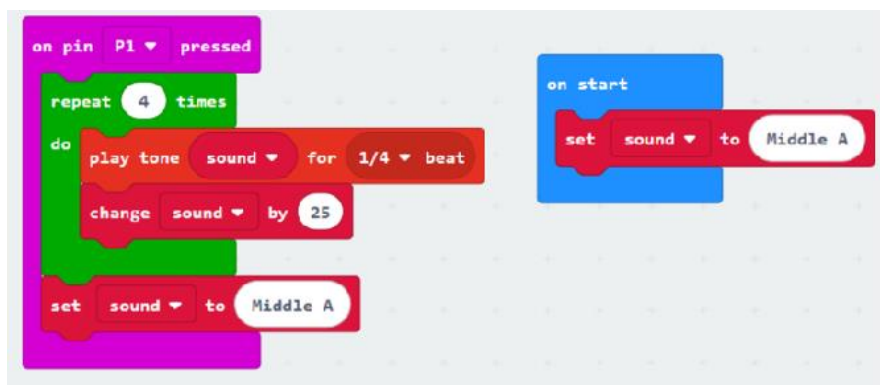
Benötigte Materialien:

- 1x Micro:bit
- 1x Kopfhörer
- 4x Kabel mit Krokodilklemmen
- 1x USB-Kabel
- 2x Früchte oder Gemüse (Kartoffeln, Banane, Orangen, Äpfel usw.)

Stellen Sie folgende Verbindungen her:



Erstellen Sie eine Variable mit dem Namen `sound`, um eine Musiknote zu speichern. Schreiben Sie den nächsten Code.



Speichern Sie die .hex-Datei und kopieren Sie sie auf das micro: bit-Laufwerk. Erstellen Sie Musik, indem Sie das Obst / Gemüse mit dem Boden verbunden halten (in unserem Beispiel der Apfel) und das andere Obst / Gemüse berühren (verbunden mit Pin 1).

Sie können ein anderes Obst / Gemüse an Pin 2 anschließen, um andere Sounds zu erzeugen. In diesem Fall können Sie den Code für Pin 1 duplizieren, Pin 2 auswählen und beispielsweise den Wert 25 mit -25 ändern.

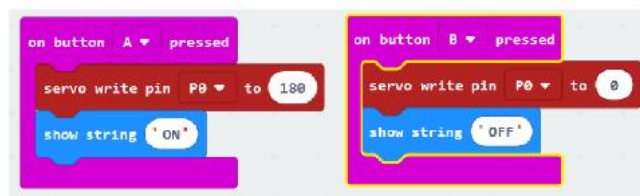
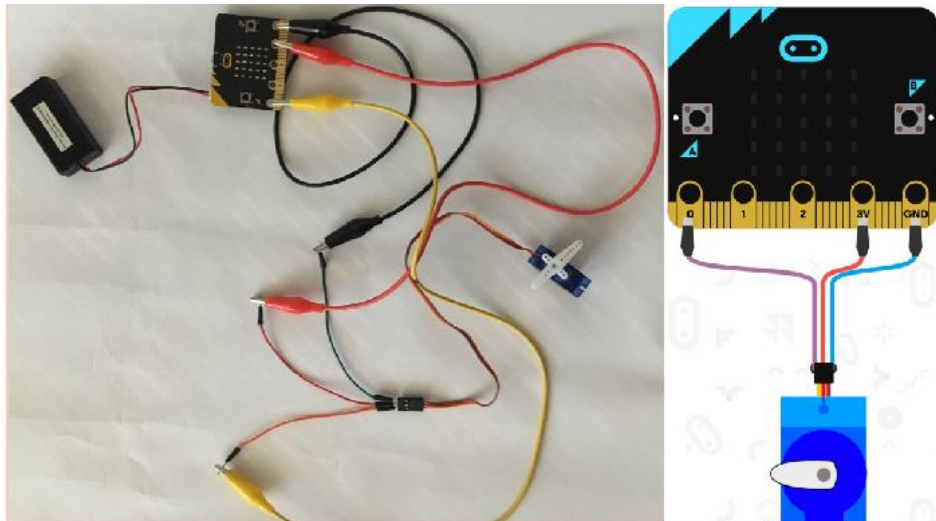
Design Challenge 5: Servomotor

Diese Anwendung zeigt, wie ein Servomotor an ein micro: bit angeschlossen wird. Benötigte Materialien:

- 1x micro:bit
- 3x Kabel mit Krokodilklemmen
- 1x USB-Kabel
- 3x Stecker-zu-Stecker Kabel
- 1x Servomotor TowerPro SG90

Der Servomotor muss wie folgt angeschlossen werden:

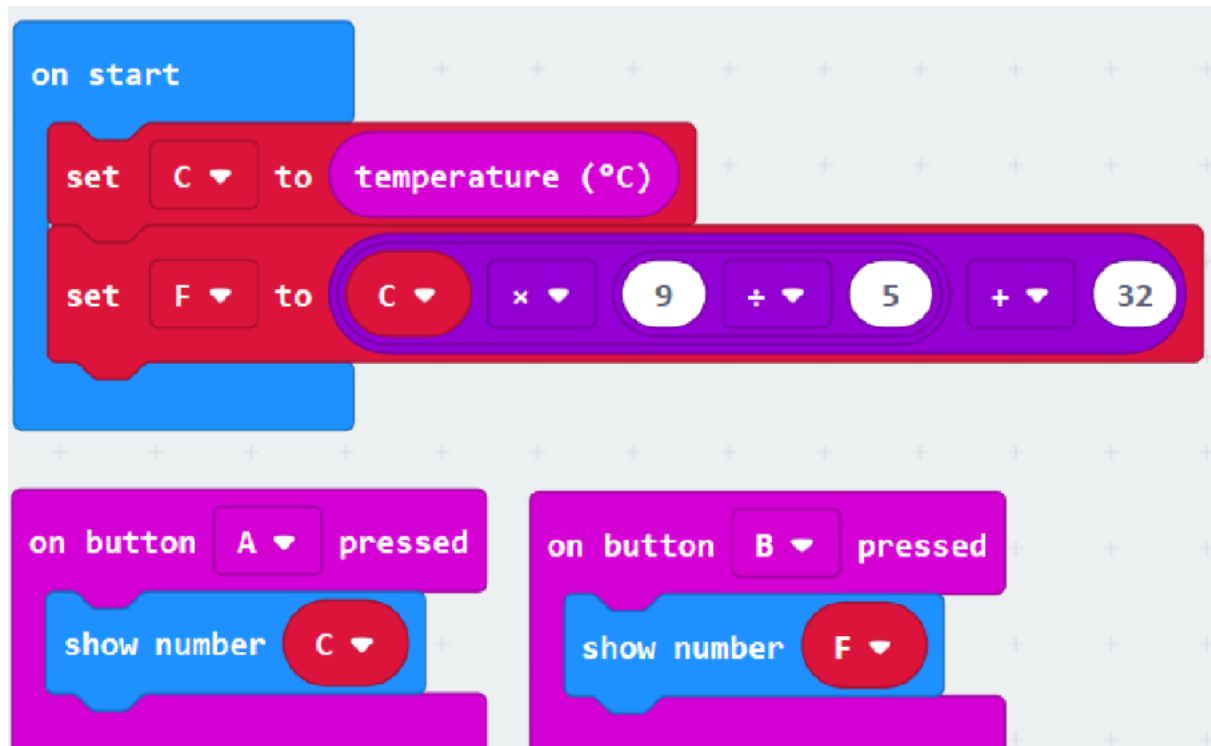
micro:bit	Servomotor
GND	brauner Draht
3V	roter Draht
P0	oranger Draht



Design Challenge 6: Thermometer

This application uses the built-in temperature sensor to display the temperature in Celsius degrees, when button A is pressed and in Fahrenheit degrees when button B is pressed.

Diese Anwendung verwendet den eingebauten Temperatursensor, um die Temperatur in Grad Celsius (beim Drücken von Taste A) und in Fahrenheit (beim Drücken von Taste B) anzuzeigen.



Um einen Wert zu erhalten, der dem realen näher kommt, sollten Sie den Wert von micro: bit mit einem Wert von einem realen Thermometer vergleichen. Dann kann das Programm geändert werden, indem der Unterschied von der Zahl subtrahiert wird, die das micro: bit anzeigt.

Design Challenge 7: Compass

Die nächste Anwendung zeigt an, in welche Himmelsrichtung das micro: bit zeigt. Nach dem Kopieren der Hex-Datei auf das micro: bit-Laufwerk fordert der Chip die Kalibrierung an. Dazu müssen Sie das micro: bit in alle Richtungen neigen, bis alle LEDs leuchten. Sie werden wissen, dass die Kalibrierung erfolgreich war, wenn ein glückliches Gesicht angezeigt wird.



Mehr Challenges?

1. Schrittzähler bauen
2. Eine eigene App für das eigene Unterrichtsfach bauen.

Quellen:

<https://makecode.microbit.org/>

<https://makecode.com/labs>

<https://makecode.microbit.org/projects/>

<https://www.itpro.co.uk/desktop-hardware/26289/13-top-bbc-micro-bit-projects>

<https://www.101computing.net/category/bbc-microbit/>

<https://support.microbit.org/support/solutions/articles/19000080171-moving-between-scratch-and-makecode>

References

<https://makecode.microbit.org/projects/>

<https://support.microbit.org/support/solutions/articles/19000080171-moving-between-scratch-and-makecode>

"Egal, ob Sie die Geheimnisse des Universums aufdecken oder nur eine Karriere im 21. Jahrhundert anstreben möchten, grundlegende Computerprogrammierung ist eine wesentliche Fähigkeit zum Lernen."

- *Stephen Hawking, theoretischer Physiker und Kosmologe*