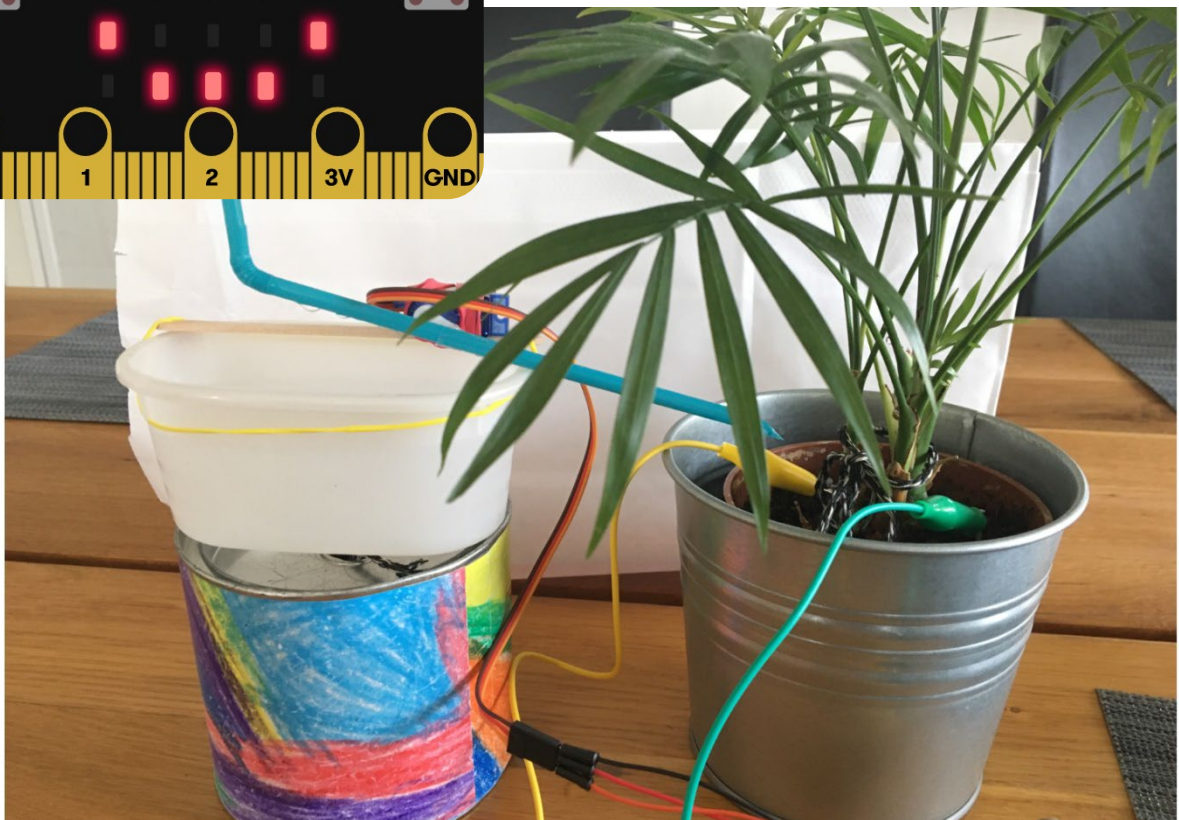
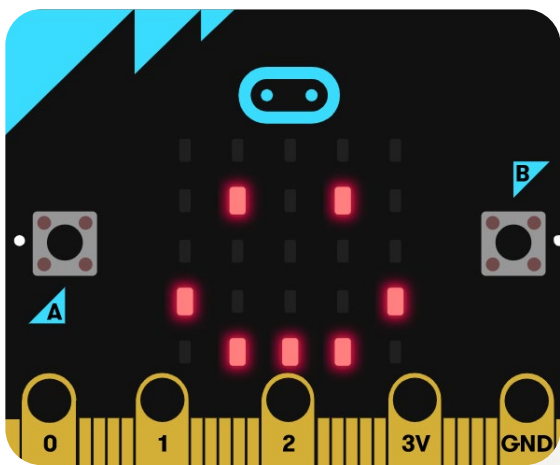


Micro:bit Einführung

Ein Minicomputer für viele Experimente



Grundlagen, Tutorials und kurze Projekte für microbit.

Diversity &
Nachwuchsförderung

Diese Einführung und Exploration kombiniert Materialien aus verschiedenen Open Education Projekten.

Einige Einführungsaufgaben stammen aus dem iMake-IT-Projekt der Pädagogischen Hochschule Schwyz (PHSZ) und aus dem OpenSource-Material unter <https://makecode.micro-bit.org/>.

Der Workshop wird im Rahmen der Diversity and MINT-Förderinitiative der ZHAW School of Engineering konzipiert und organisiert.

Weitere Informationen über den PHSZ Projekt unter:

phsz-facile.ch/imake-it

imake-it.ch

Und Kontakt mit der School of Engineering Diversity unter:

dqtm@zhaw.ch

diversity.engineering@zhaw.ch

Dieses Einführungs-Tutorial ist als OER veröffentlicht unter:

<https://code4you.ch/> >> Aufgaben >> code+Motion

Impressum

Version 1.0 (Mai 2022)

Autoren: Marina de Queiroz Tavares, Seraina Betschart, Jason Curtins

dqtm@zhaw.ch

imake-it@phsz.ch

Bilder, Grafiken, Screenshots: PHSZ, ZHAW SoE

Icons: S.20 Misc Dice by glitch (openclipart.org), S.23 Compass Rose by Firkin (openclipart.org), S.27 Blue Robot by Scout (openclipart.org), S.30 Piano Keyboard by GDJ (openclipart.org), S.33 Rock-Paper-Scissors by uoresch (openclipart.org)



Namensnennung
Weitergabe unter gleichen Bedingungen

Inhalt

Grundlagen

Nummer	Challenge	Inhalt	Seite
1	Hallo Welt! 🌍 & Schlagendes Herz ❤️	Wir lernen das LED-Display des micro:bits kennen und benutzen die Blöcke: "beim Start" und "dauerhaft".	5
2	Knöpfe A und B benutzen	Wir brauchen die zwei eingebauten Knöpfe A & B von dem micro:bit. Zudem arbeiten wir mit «Wenn, dann»-Blöcken.	7
3	Wie laut ist das? Messung Visualisieren	Wir kennen zwei Möglichkeiten eine Messung zu visualisieren. Mit Grenzwerten und mit der Konsole.	9
4	Schrittzähler 	Wir verwenden den Bewegungssensor, um die Schritte zu erkennen und eine Variable, um die Schritte zu zählen.	10
5	Funk mal was! 💬📢	Wir lassen zwei micro:bits über Funk miteinander sprechen.	11

Exploration

Nummer	Challenge	Inhalt	Seite
6	Wie hell ist das? Schuhbladen-Alarm	Baue einen Alarm ein, der erkennt, wenn jemand deine geheime Schublade öffnet.	12
7	Wie schräg ist das? Kipp-Alarm	Untersucht das Programm eines Alarms, der singt, wenn das micro:bit kippt.	13
8	Leitend oder nicht leitend? Bodenfeuchtigkeit	Erkennt, ob der Boden trocken oder feucht ist.	14
9	Pflanztopfgiesser	Misst den Widerstand und steuert einen Servomotor, der kleine Mengen Wasser ausgiesst.	17

1) Hallo Welt! 🌍 & Schlagendes Herz ❤️

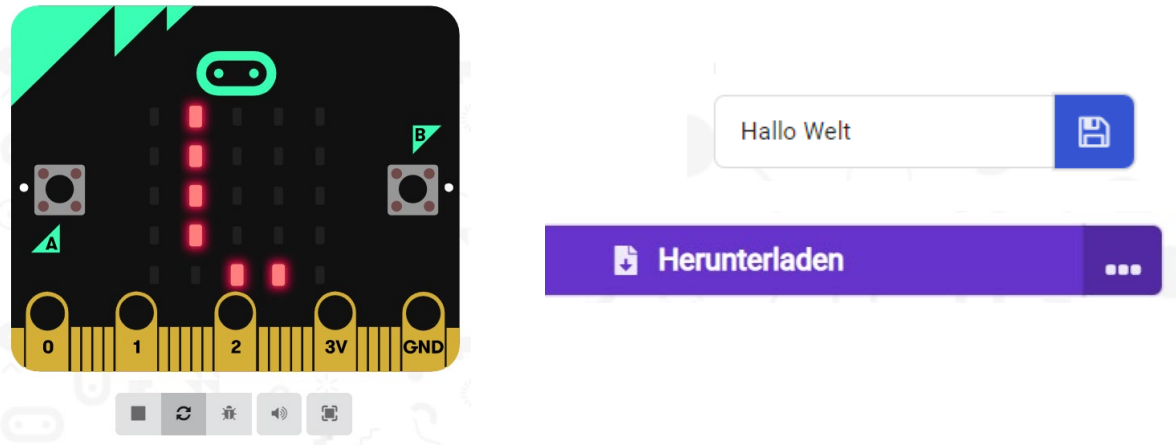
Baue den untenstehenden Code nach. Im blauen Kasten «verwendete Befehlsgruppen» siehst du, wo du die Befehle dafür findest.

VERWENDETE BEFEHLSGRUPPEN

Grundlagen



Überprüfe die Funktion im Simulator und speichere das Programm. Lade dann das Programm auf das micro:bit Board herunter.



Wenn du alles richtig gemacht hast, läuft der Text einmal auf dem micro:bit Display.

Hinweis:

Wenn du das Programm neu starten willst, drücke auf die Reset-Taste auf der Rückseite des micro:bits.

Challenge 1a

Verändere das Programm so, dass ein anderer Text auf dem LED-Display angezeigt wird.

Hinweis:

Um ein neues oder verändertes Programm auf dem micro:bit zu testen, muss es jedes Mal von Neuem auf den micro:bit heruntergeladen werden. Dabei wird das alte Programm überschrieben.

Challenge 1b

Passe das Programm an, damit der Text unendlich lange über das Display läuft. Im blauen Kasten siehst du, wo du die Befehle dafür findest.

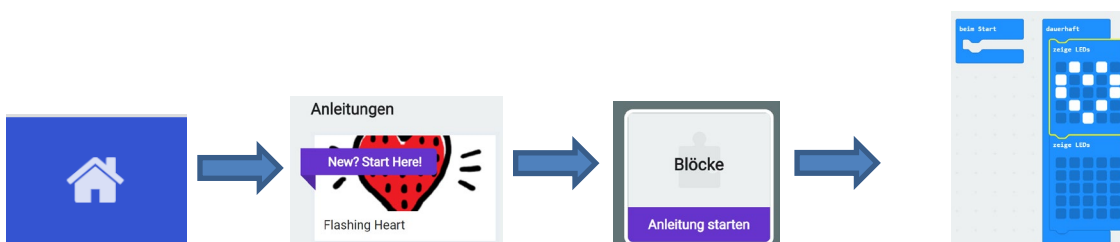
VERWENDETE BEFEHLSGRUPPEN

Grundlagen

Challenge 1c Schlagendes Herz

Wir machen jetzt ein neues Programm, ähnlich wie bei Herausforderung 1b, aber statt eines Textes zeigen wir ein Herz, das erscheint und verschwindet.

Wenn du einen Tipp brauchst, schau dir das Tutorial auf der Homepage von makecode.microbit.org an.



Hinweis:

Um nicht mehr benötigte Programmierenteile zu löschen, ziehe sie mit der Maus nach links, bis ein Mülleimersymbol erscheint.



2) Knöpfe A und B benutzen

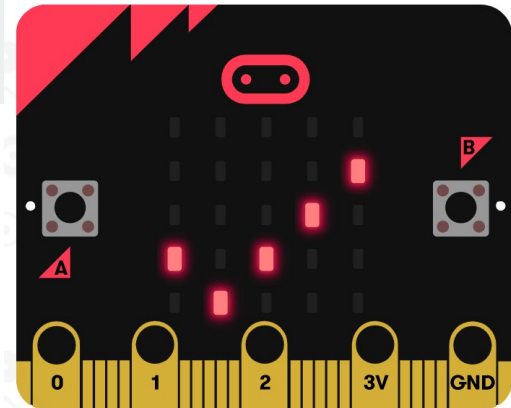
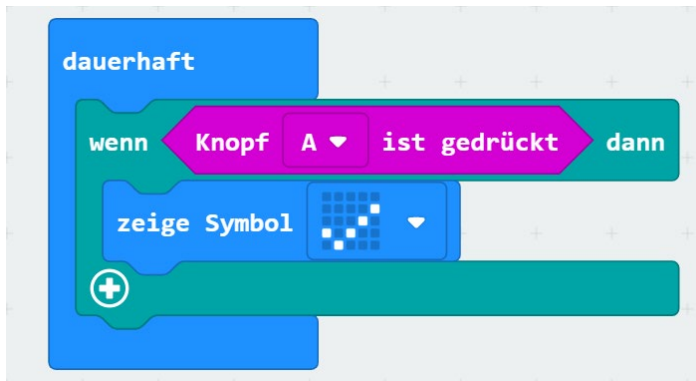
Baue den untenstehenden Code nach.

VERWENDETE BEFEHLSGRUPPEN

Grundlagen

Logik

Eingabe



Wenn man Knopf A drückt, erscheint ein Häkchen-Symbol auf dem Display.

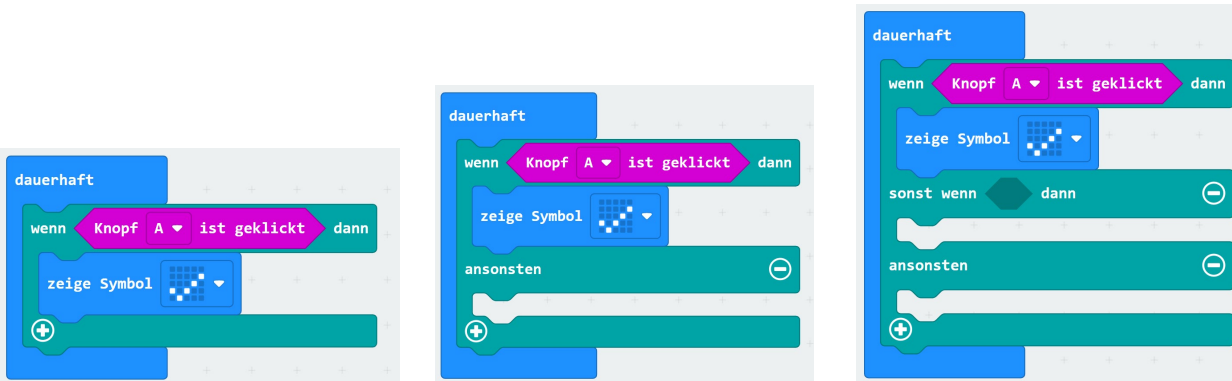
Tipp: Teste dein Programm auf dem Simulator, bevor du es auf das Board herunterlässt.

Challenge 2a

Erweitere das Programm, sodass auch für Knopf B ein Symbol gezeigt wird.

Tipp:

Drücke das «+» Zeichen innerhalb des "wenn dann"-Blocks, um weitere Bedingungen zu prüfen.



Challenge 2b

Gestalte ein eigenes Symbol für Knopf A.

.....
 VERWENDETE BEFEHLSGRUPPEN

Grundlagen

Challenge 2c

Verwende eine Bewegung als Input, anstelle von Knopf A und B.

.....
 VERWENDETE BEFEHLSGRUPPEN

Grundlagen Eingabe

3) Wie laut ist das? Messung Visualisieren

Baue den untenstehenden Code nach. Lade dein Programm herunter und teste es.

VERWENDETE BEFEHLSGRUPPEN

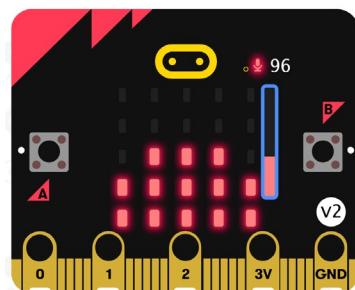
Grundlagen Logik Eingabe LED

```
dauerhaft
wenn Lautstärke > 100 dann
  zeige Symbol
ansonsten
  Bildschirminhalt löschen
```

Was macht dein Programm?

Jetzt wollen wir das gemessene Volumen im Detail verfolgen. Versuche eine grafische Darstellung mit dem Befehl "Balkendiagramm zeichnen", wie unten gezeichnet.

```
dauerhaft
zeichne Säulendiagramm von Lautstärke
bis 200
```



4) Schrittzähler



Nächstes wollen wir den Bewegungssensor benutzen, um Schritte zu erkennen, und eine Variable, um die Schritte zu zählen.

Baue den untenstehenden Code nach.

VERWENDETE BEFEHLSGRUPPEN

Grundlagen

Eingabe

Variablen

LED

mehr

```

beim Start
  setze step auf 0

dauerhaft
  zeige Zahl step

wenn geschüttelt
  ändere step um 1
  Halte Animation an
    
```



Lade den Code herunter und befestige einen Batterieblock an deinem micro:bit.

Du bist bereit, dein Programm zu testen! Stecke den micro:bit in deine Socke oder befestige ihn mit einem Band um deinen Knöchel und laufe herum!

5) Funk mal was!


Baue die untenstehenden Codes nach. Für den Test brauchst du zwei micro:bits, die sich gegenseitig anfunken! Arbeite mit einer Kollegin oder einer Kollegen zusammen.

.....
VERWENDETE BEFEHLSGRUPPEN

 Grundlagen

 Funk

 Eingabe



```

beim Start
  setze Funkgruppe auf 234
  wenn Knopf A geklickt
    sende Text "Hoi" über Funk
  wenn Text empfangen receivedString
    zeige Text receivedString
  
```

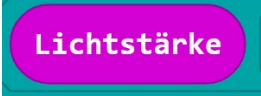

Was macht dein Programm?

Vereinbart eine Nummer zwischen 0 und 255 für eure Funkgruppe.
Trage diese Nummer vorne auf den Taffeln.

Hinweis:

Wenn mehrere Sender und Empfänger gleichzeitig Nachrichten senden und empfangen, ist es besser, für jedes Sender-Empfänger-Paar eine andere Funkgruppennummer zu wählen. Ganzzahlige Werte zwischen 0-255 sind möglich.

6) Wie hell ist das? Schuhbladen-Alarm

Mini-Projekt Aufgabe	Baue einen Alarm ein, der erkennt, wenn jemand deine geheime Schublade öffnet.
Hinweise	Die benötigten Code-Elemente sind unterhalb der Aufgabe gegeben.
	Man kann mit einer Messung der Lichtstärke arbeiten. Diese Messung ist verfügbar als Block in der Befehlsgruppe Eingabe.
	Schau nach, wie du Aufgabe 3) aufgebaut hast. Viele Ideen können hier hilfreich sein.
	Der Alarm kann einen Ton oder ein Lied abspielen. Schau dir die Blöcke in der Befehlsgruppe Musik an.

Code-Elemente

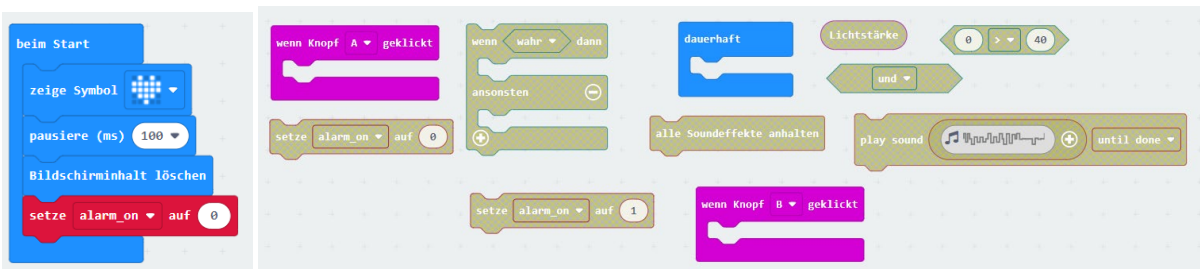


Extra

Ein/Aus-schalten

Baue eine Variable in dein Programm ein, mit der du den Alarm ein- und ausschalten kannst.

Code-Elemente



7) Wie schräg ist das? Kipp-Alarm

Untersucht das Programm eines Alarms, der singt, wenn das micro:bit kippt.

Hinweis	tilt = kippen enable = aktivieren
----------------	--------------------------------------

The code is written in Scratch and consists of the following blocks:

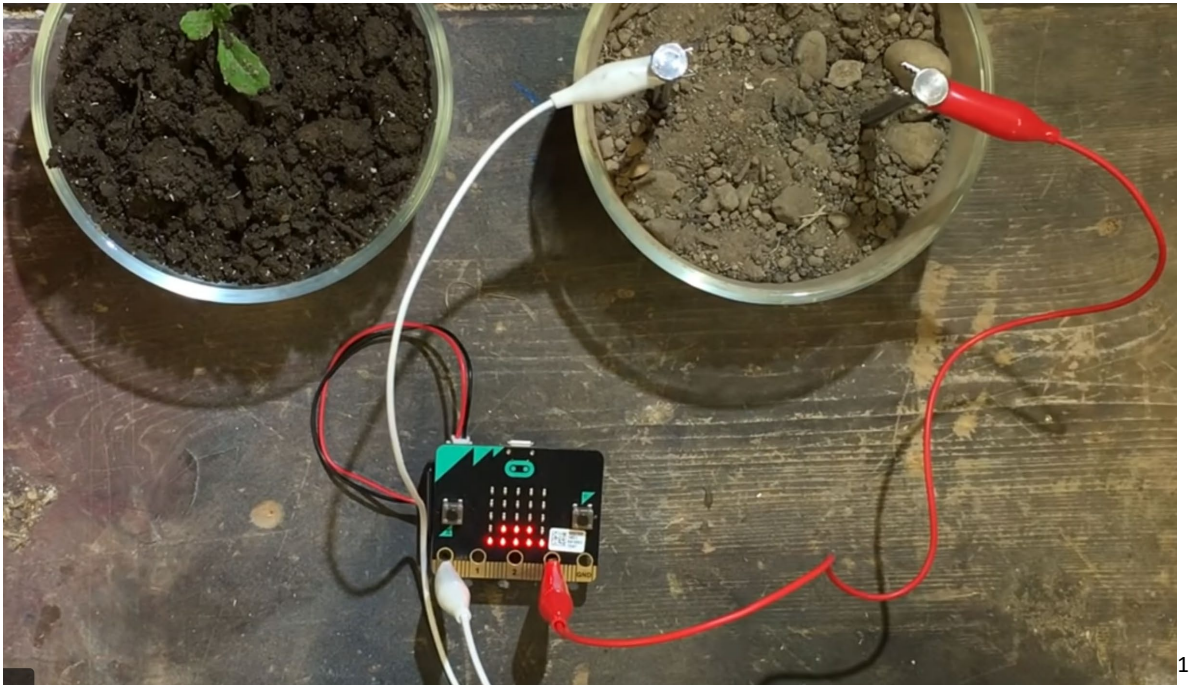
- beim Start (blue blocks):**
 - zeige Symbol (grid icon)
 - pausiere (ms) 100
 - zeige Pfeil (Osten)
 - pausiere (ms) 100
 - zeige Pfeil (Westen)
 - setze enable auf 0
- wenn Knopf A geklickt (purple block):**
 - setze enable auf 1
- wenn Knopf B geklickt (purple block):**
 - setze enable auf 0
- dauerhaft (blue loop block):**
 - zeige LEDs (5x5 grid)
 - wenn enable dann:
 - setze tilt auf Rotation (°) rollen
 - wenn tilt > 0 dann:
 - Note (Hz) tilt
 - ansonsten:
 - Note (Hz) 180 - tilt
 - ansonsten:
 - alle Soundeffekte anhalten

Wie funktioniert dieses Programm? Lade es herunter und teste es. Nimm einige Änderungen vor und besprich deine Ideen mit einer unseren Workshop-Betreuende.

Kannst du einige Ideen des Kipp-Alarmes nutzen, um deinen Schuhbladen-Alarm zu verbessern?

8) Leitend oder nicht leitend? Bodenfeuchtigkeit

Erkennt, ob der Boden trocken oder feucht ist.



Hier ist, was du tun musst, um deinen Bodenfeuchtesensor zu bauen:

- Verbinde einen Nagel mit einer Krokodilklemme mit dem 3V-Pin und stecke ihn in den Boden.
- Verbinde den anderen Nagel mit einer Krokodilklemme mit dem P0-Pin und stecke ihn in den Boden.

Das war's!

Erstelle nun den Code zum Messen und Anzeigen des Analogwerts an Pin P0.



¹ Für mehr Details und Erweiterungen schau unten: <https://makecode.microbit.org/projects/soil-moisture>

VERWENDETE BEFEHLSGRUPPEN

Fortgeschritten Pins

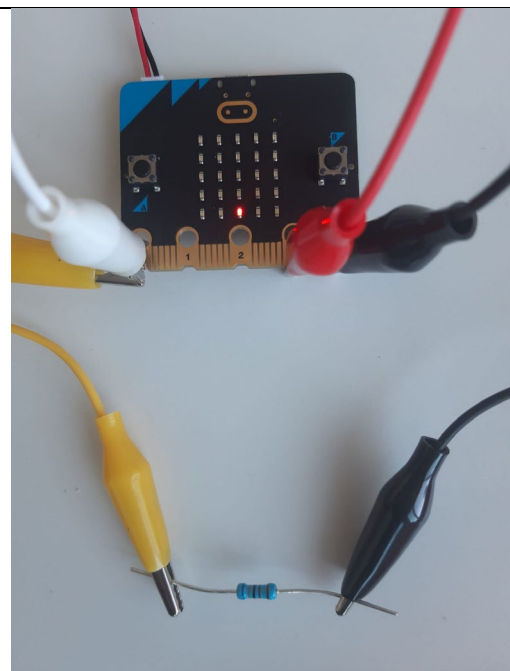
Grundlagen Eingabe Variablen LED Logik

Hinweise

Wenn der analoge Messwert zu hoch ist (selbst für den trockenen Boden), schliesse einen zusätzlichen Widerstand zwischen P0 und Masse an (zum Beispiel 6,8kOhm).

Du kannst dein Programm auch mit trockenen und nassen Wattestücken testen.

Die nassen Wattestücke solltest du mit Wasser + Salz und etwas Zitronensaft vorbereiten.



FRAGEN

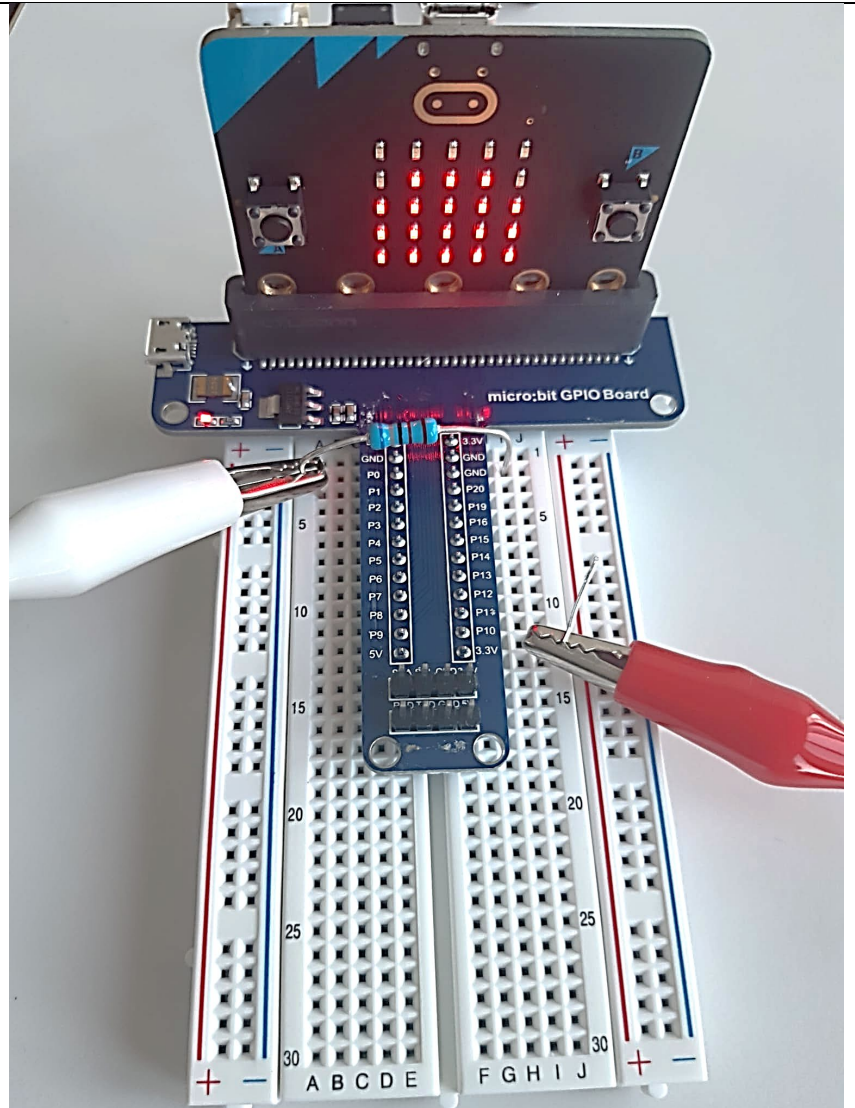
- Notiere die Werte, die du für den trockenen und den nassen Boden gemessen hast.
- Was ist der minimale und maximale Wert, den wir hier messen können?

c) Bevor wir weitermachen, baue eine stabilere Struktur mit einem lötfreien Breadboard und einer micro:bit-Adapterplatine. In der vorderen Tabelle findest du ein Beispiel, um zu sehen, wie es im Detail gemacht wird.

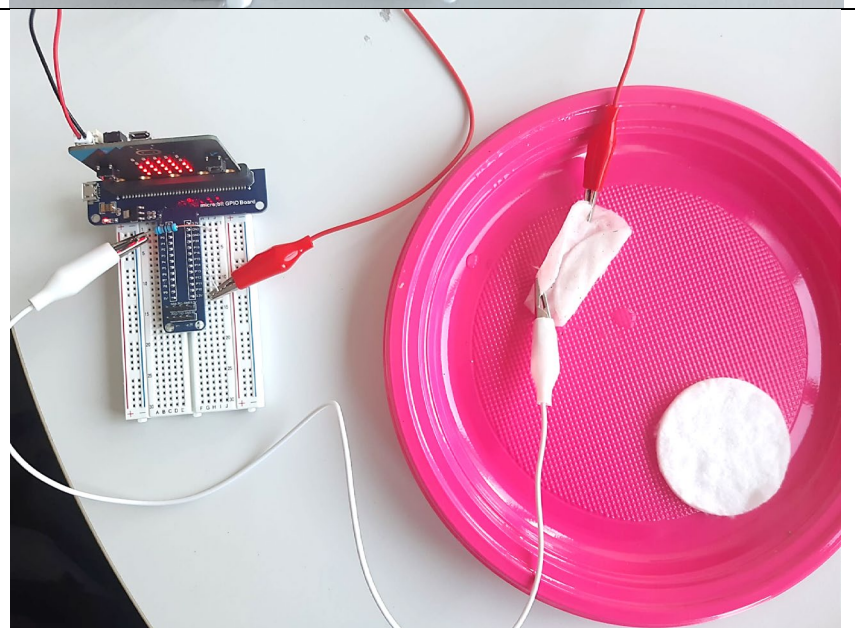
Der Widerstand liegt zwischen P0 und GND.

Schließe das erste Krokodilkabel an P0 an (benutze das Widerstandsbein).

Das 2. Krokodilkabel wird an 3V3 angeschlossen (verwende ein Metallstück).



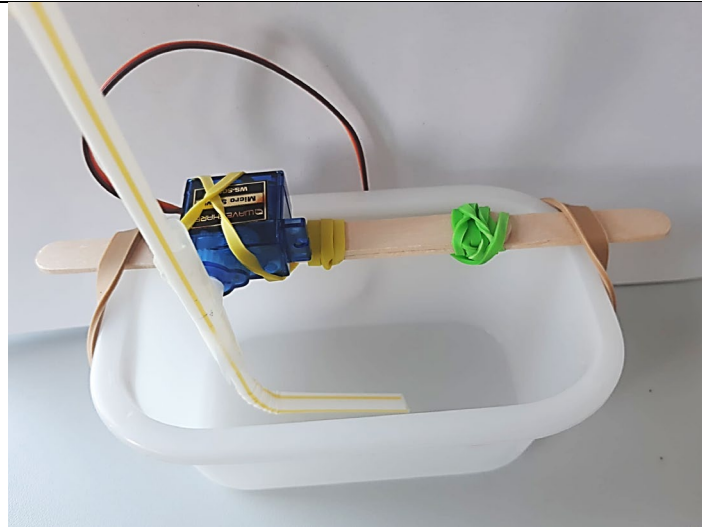
d) Kontrolliere erneut die Funktion mit den trockenen und feuchten Wattestücken.



9) Pflanztopfgiesser

Teil-1 Aufbau und Servomotor steuern

Folge die Erklärungen, um den Pflanztopfgiesser zu bauen, so dass am Ende es so aussieht.



Erstelle nun Teil 1 des Programms und versuche, es auszuführen.

Wichtig!

Nimm den Holzstab heraus, um die Position des Strohhalms anzupassen.

```

beim Start
  zeige Symbol [Matrix]
  setze Winkel von Servo an P2 auf 0°
  pausiere (ms) 2000
  Bildschirminhalt löschen

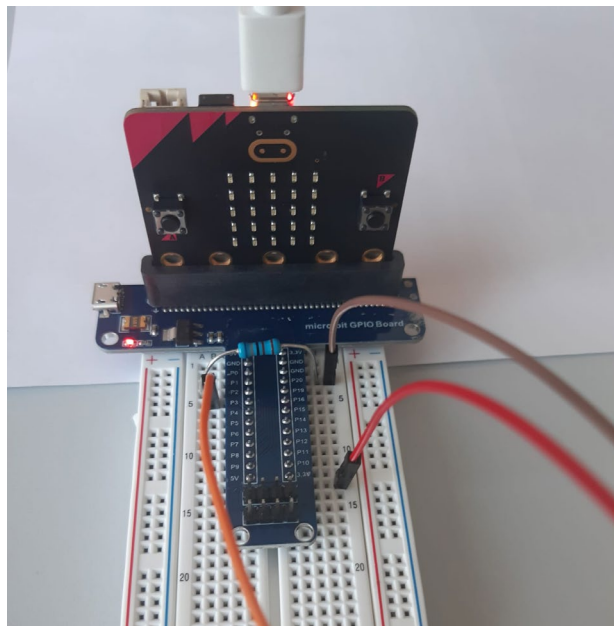
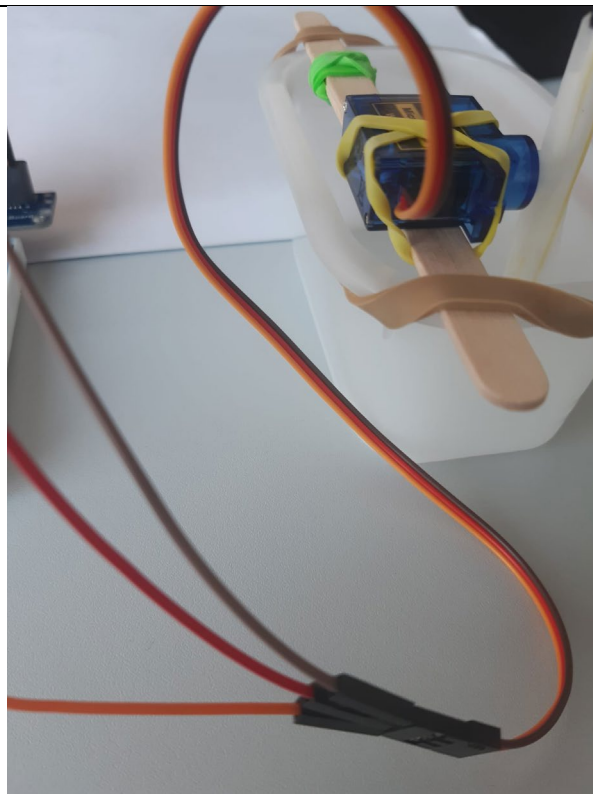
dauerhaft
  wenn Knopf B ist geklickt dann
    zeige Symbol [Matrix]
    setze Winkel von Servo an P2 auf 90°
    pausiere (ms) 3000
    setze Winkel von Servo an P2 auf 0°
    pausiere (ms) 2000
    Bildschirminhalt löschen
  
```

Schließe die Pins des Servomotors wie angegeben an:

Braun: GND

Rot: 3V3

Orange: P2



Teil-2 Kombinieren mit den Bodenfeuchtesensor

Der letzte Schritt besteht darin, das Programm so zu erweitern, dass die Pflanzenbewässerung nur aktiviert wird, wenn der Boden trocken ist. Überprüfe, welche Teile von Aufgabe 8 du brauchst, um das umzusetzen, und erweitere dein Programm.

Wenn du das zu schwierig findest, frag eine Workshop-Betreuende um Hilfe.

Mögliche Lösung für Challenge:

6) Wie hell ist das? Schuhbladen-Alarm

The image shows a Scratch script for a "Schuhbladen-Alarm" (shoe blade alarm) system. The script is organized into three main sections: "beim Start" (when started), "wenn Knopf A geklickt" (when button A is clicked), and "wenn Knopf B geklickt" (when button B is clicked). The "beim Start" section includes a "zeige Symbol" block with a grid icon, a "pausiere (ms)" block set to 200, a "Bildschirminhalt löschen" block, and a "setze alarm_on auf 0" block. The "wenn Knopf A geklickt" section includes a "setze alarm_on auf 1" block. The "wenn Knopf B geklickt" section includes a "setze alarm_on auf 0" block. The "dauerhaft" (forever) loop section contains a "wenn alarm_on und Lichtstärke > 100 dann" block. The "dann" block includes a "spiele Ton" block with a musical note icon and a "bis zum Fertigstellen" dropdown. The "ansonsten" block includes an "alle Soundeffekte anhalten" block.

```
beim Start
  zeige Symbol [Grid Icon]
  pausiere (ms) 200
  Bildschirminhalt löschen
  setze alarm_on auf 0

wenn Knopf A geklickt
  setze alarm_on auf 1

wenn Knopf B geklickt
  setze alarm_on auf 0

dauerhaft
  wenn alarm_on und Lichtstärke > 100 dann
    spiele Ton [Musical Note] bis zum Fertigstellen
  ansonsten
    alle Soundeffekte anhalten
```

Mögliche Lösung für Challenge:

9) Pflanztopfgiesser

Teil-2

```
beim Start
  zeige Symbol
  setze Winkel von Servo an P2 auf 0 °
  pausiere (ms) 2000
  Bildschirminhalt löschen

dauerhaft
  setze Messwert auf analoge Werte von Pin P0
  wenn Messwert < 500 dann
    zeige Symbol
    setze Winkel von Servo an P2 auf 90 °
    pausiere (ms) 3000
    setze Winkel von Servo an P2 auf 0 °
    pausiere (ms) 2000
    Bildschirminhalt löschen
  wenn Knopf A ist geklickt dann
    zeige Zahl Messwert
```