

## Was kann der micro:bit

## Wie programmiere ich den micro:bit?

Wir programmieren den micro:bit mit der «Entwicklungsumgebung» MakeCode. Du kannst den micro:bit auf 4 unterschiedliche Arten programmieren.

1. Laptop mit dem Programm «makecode-mircobit.exe» auf dem Stick
2. Laptop über den Browser mit: <https://makecode.microbit.org/>
3. iPad mit der App: «micro:bit»
4. Android Tablet mit der App «micro:bit»

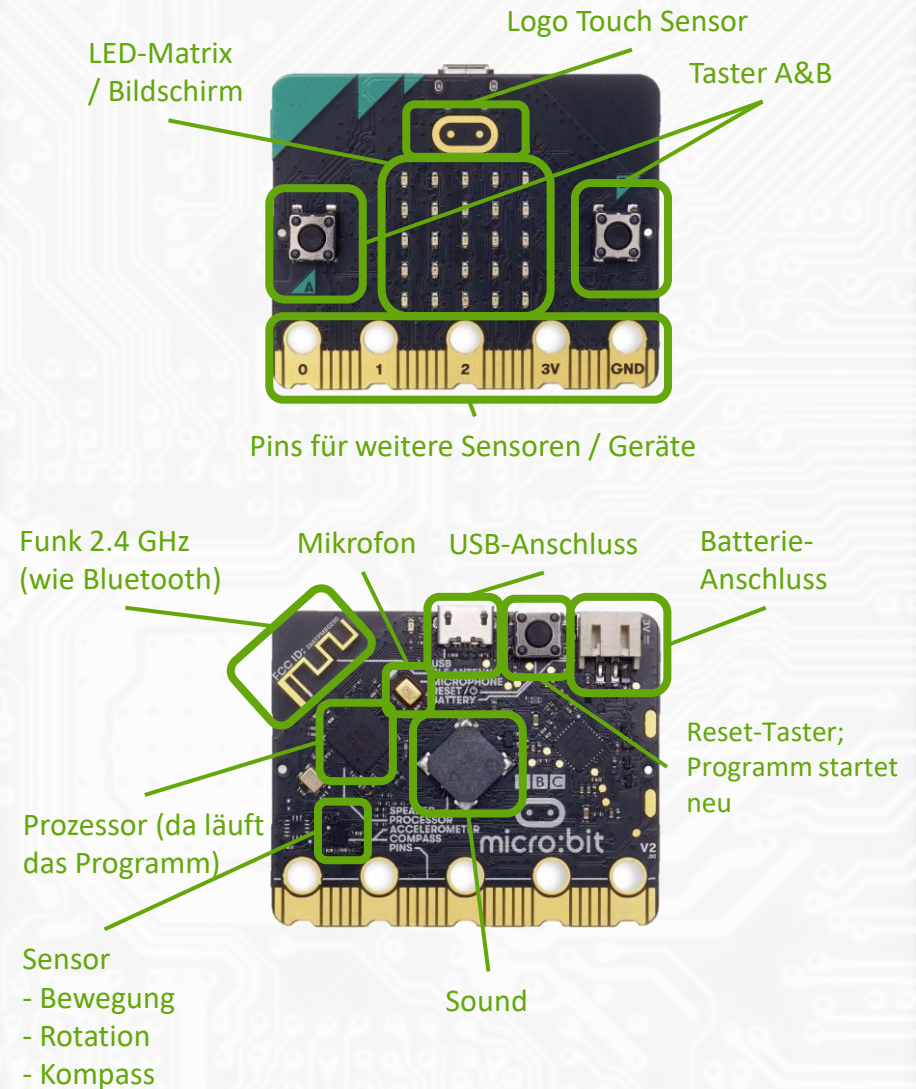
## Hinweise

Das Programmieren ist bei allen 4 Möglichkeiten identisch. Du kannst folgende 3 Programmiersprachen auswählen:

1. **Blöcke** (ist am einfachsten, der ganze Kurs verwendet Blöcke)
2. Textbasiert (Schwieriger, wird nur empfohlen, wenn du bereits viele Programmiererfahrungen hast)
  1. JavaScript
  2. Python



Du kannst auch während dem Programmieren die Sprache umstellen und du siehst, wie deine programmierten Blöcke in Python oder JavaScript aussehen.



## Hallo Welt

Meistens startet man beim Programmieren mit dem Programm „Hallo Welt“. Dabei soll der micro:bit auf dem Bildschirm den Text „Hallo Welt“ anzeigen.

Überlege dir, wie das „Hallo Welt“ einmal beim Starten des Programms erscheint.

Das „Hallo Welt“ Programm wird seit 1978 von Programmier:innen bei fast jedem Programm verwendet. Damit wird getestet, ob der Computer oder der Mikrokontroller programmierbar ist.

### Hinweise

- Wenn du das Programm neu starten willst, drücke den Reset-Taster auf der Rückseite des micro:bits.
- Wenn du das Programm änderst, musst du es jedes Mal neu auf den micro:bit herunterladen
- Wenn du zur nächsten Aufgabe gehst, kannst du den Code der letzten Aufgabe aus der Schleife ziehen. Sonst arbeitet der micro:bit immer den alten Code zuerst ab, bevor er zum neuen kommt.

### Material

- micro:bit
- PC oder iPad

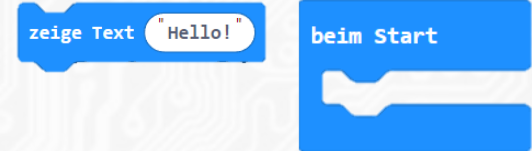


### Tipps

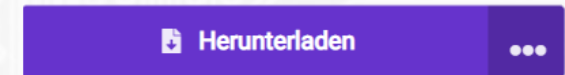
Verwendete Befehlsgruppen

 Grundlagen

Verwendete Blöcke



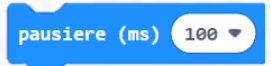
Programm auf micro:bit laden



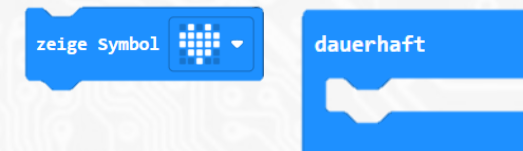
Abspeichern:



### Zusatzaufgaben

- Programmiere den micro:bit so, dass beim Start ein Symbol angezeigt wird und später dauerhaft ein Text.
- Erzähle mit Symbolen auf dem Display eine Geschichte. Du kannst die Symbole auch selbst erstellen.
- Experimentiere mit  zwischen den Symbolen

Verwendete Blöcke:



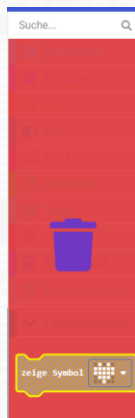
## Startblöcke

Befehlssequenzen können durch verschiedene Blöcke gestartet werden. In der Challenge 1 hast du bereits den Block «beim Start» kennengelernt. Die Befehle in diesem Block werden nur einmal ausgeführt, danach ist das Programm beendet. Alle Befehle, die du im Block «dauerhaft» einsetzt, werden unendlich oft wiederholt, solange der micro:bit mit Strom versorgt wird.

Erweitere dein Hallo Welt Programm, sodass abwechselungsweise ein Text geschrieben und danach ein Symbol für eine Sekunde angezeigt wird. Diese zwei Schritte sollen dauerhaft wiederholt werden.

## Hinweise

- Wenn du das Programm änderst, musst du es jedes Mal neu auf den micro:bit herunterladen.
- Wenn du zur nächsten Aufgabe gehst, kannst du den Code der letzten Aufgabe löschen oder aus der Schleife ziehen. Verschiebe deine Blöcke dafür in den Bereich der Befehlsgruppen.

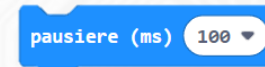
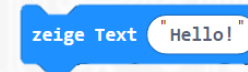


## Tipps

Verwendete Befehlsgruppen

Grundlagen

Verwendete Blöcke



## Material

- micro:bit
- PC oder iPad



## Taster A und B benutzen

Mit den Tastern A und B kannst du den micro:bit steuern. Aber du musst den micro:bit zuerst so programmieren, dass er auf die Taster reagieren kann.

Versuche, zum Beispiel, dass beim Drücken vom Taster A ein Smiley erscheint.



### Hinweise

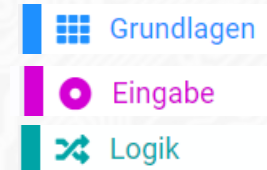
- Teste dein Programm auf dem Simulator, bevor du es auf das Board herunterlädst.
- Klicke das «+» Zeichen innerhalb des «Wenn-Dann»-Blocks, um weitere Bedingungen zu prüfen.
- Wenn man den Taster A drückt, erscheint der Smiley auf dem Display. Dieses Symbol bleibt auch bestehen, wenn der Taster A wieder losgelassen wird. Weshalb?

### Material

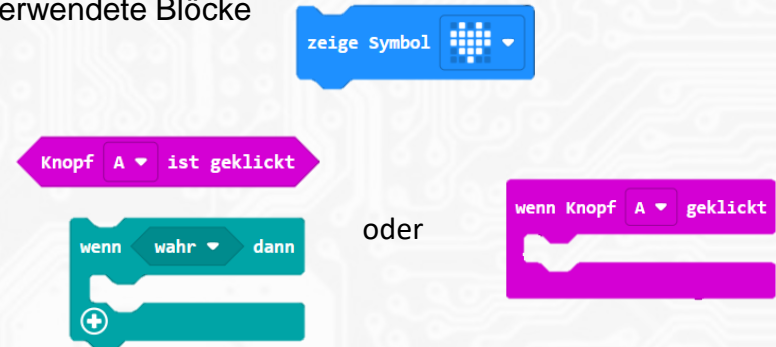
- micro:bit
- PC oder iPad

### Tipps

Verwendete Befehlsgruppen



Verwendete Blöcke



### Zusatzaufgaben

- Erweitere das Programm so, dass beim Drücken des Tasters B ein anderes Symbol angezeigt wird
- Verwende zusätzlich den Touch-Sensor „Logo“ um ein eigenes Symbol anzuzeigen

Logo ist gedrückt

## Befehle wiederholen

Befehle wiederholen sich beim Programmieren oft. Da Programmierer faul sind, möchten sie so wenig Blöcke wie möglich einsetzen. Dafür gibt es Schleifen. Eine Schleife wiederholt eine Reihe von Blöcken, bis eine Bedingung zum Abbrechen erfüllt ist. Mit der Variablen „Index“ weisst du, beim wievielten Durchgang die Schleife ist.

Dein neues Programm soll von 0 bis 9 zählen und die Zahlen auf dem LED-Display anzeigen.



### Hinweise

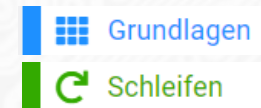
- Teste dein Programm auf dem Simulator, bevor du es auf das Board herunterlädst.
- Du kannst die Anzeigedauer der Zahlen mit dem „pausiere (ms)“ Block variieren

### Material

- micro:bit
- PC oder iPad

### Tipps

Verwendete Befehlsgruppen



Verwendete Blöcke



### Zusatzaufgaben

- Beim Drücken des Tasters A soll in 2er-Schritten gezählt werden.
- Beim Drücken des Tasters B soll der Zähler auf 0 zurückgesetzt werden.

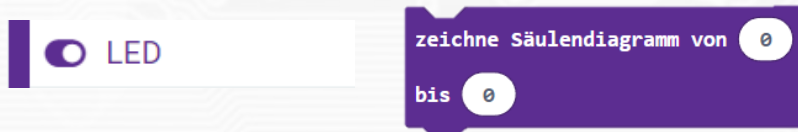
## Wie laut ist das?

Findest du das Mikrofon auf dem micro:bit? Das Mikrofon ist einer von vielen Sensoren auf dem micro:bit. Programmiere mit dem micro:bit eine Lautstärkenwarnung. Zeige dazu auf dem Bildschirm ein Symbol an, wenn die Lautstärke höher als 100 ist.

Schreie den micro:bit an, bis du das Symbol siehst. Wenn du nicht mehr schreist, sollte kein Symbol mehr angezeigt werden.

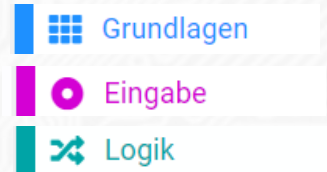
### Zusatzaufgaben

- Wie laut kannst du Schreien? Spiele mit dem Vergleichswert der Lautstärke!
- Versuche die aktuelle Lautstärke auf dem Bildschirm anzuzeigen. Nutze dazu den folgenden Block:



### Tipps

Verwendete Befehlsgruppen



Verwendete Blöcke



### Material

- micro:bit
- PC oder iPad



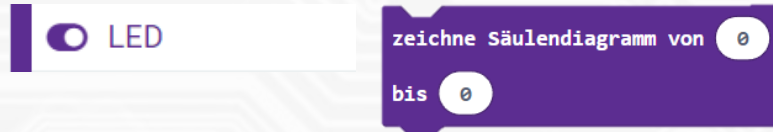
## Wie Hell ist es?

Dein micro:bit kann die Helligkeit messen. Die Helligkeit wird mit einer Zahl von 0 bis 255 angegeben, wobei 0 der dunkelste und 255 der hellste Wert ist.

Lasse deinen micro:bit einen Warnton abspielen, wenn der Helligkeitswert grösser als 150 wird.

### Zusatzaufgaben

- Versuche die aktuelle Helligkeit auf dem Bildschirm anzuzeigen. Nutze dazu den folgenden Block:

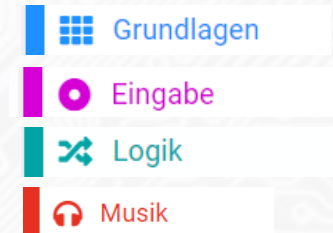


### Material

- micro:bit
- PC oder iPad

### Tipps

Verwendete Befehlsgruppen



Verwendete Blöcke



## Würfel

Du kannst den micro:bit auch als Würfel verwenden! Zeige auf dem Display nach dem Schütteln die gewürfelte Zufallszahl zwischen 1 und 6 an!

Erweitere die Anzeige so, dass das Resultat wie bei einem normalen Würfel mit 7 Lichtpunkten angezeigt wird.



### Zusatzaufgaben

- Zeige während dem Würfeln (Schütteln) ein anderes Symbol (Zum Beispiel ein Smiley oder auch abwechselnd Nummern)
- Zinke den Würfel, sodass der Würfel immer eine 6 anzeigt. Damit der gezinkte Würfel nicht auffällt, soll er nur eine 6 würfeln, wenn du gleichzeitig zum Schütteln das Logo berührst.

### Erklärung Variable

Eine Variable ist eine gespeicherte Zahl, welche einen Namen hat. Die Zahl kann man verändern, auf einen Wert setzen und / oder auch damit rechnen.

### Tipps

Verwendete Befehlsgruppen



Grundlagen

Eingabe

Mathematik

Verwendete Blöcke



### Material

- micro:bit
- PC oder iPad

### Resultate

Würfle 20 bis 50-mal und schaue, wie oft welche Nummer gewürfelt wurde. Ist dein Würfel fair?

1:\_\_\_\_\_ 2:\_\_\_\_\_ 3:\_\_\_\_\_ 4:\_\_\_\_\_ 5:\_\_\_\_\_ 6:\_\_\_\_\_



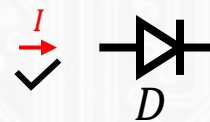
## Die Diode

Die Diode ist wie eine Einbahnstrasse für Strom in einem elektronischen Gerät. Wenn Strom versucht, in die falsche Richtung zu fließen (was manchmal passieren kann), blockiert die Diode den Strom. Aber wenn der Strom in die richtige Richtung fließt, lässt die Diode ihn durch.

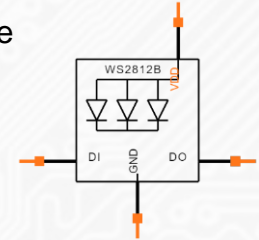
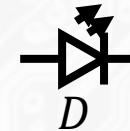
Durch die Diode wird hier sichergestellt, dass der micro:bit keinen Strom an den LED-Ring geben kann. Denn die LEDs vom LED-Ring benötigen mehr Strom als der micro:bit liefern kann.

## Schema - Symbole

Diode



Leuchtdiode



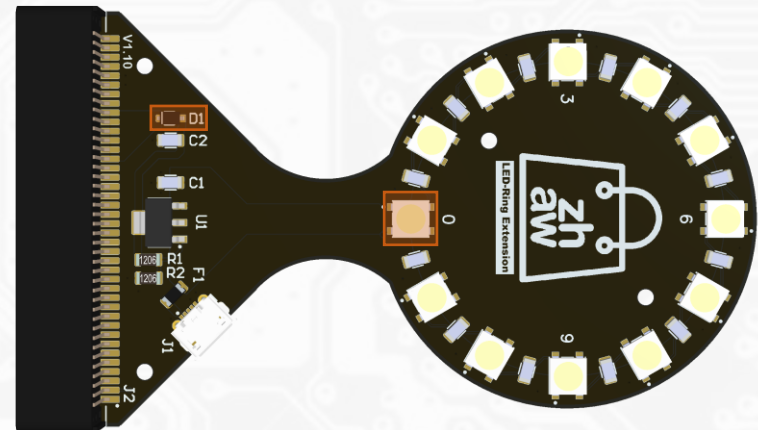
## Die Leuchtdiode (LED)




Eine LED ist eine lichtemittierende Diode. Das bedeutet, sie erzeugt Licht, wenn Strom durchfließt. Wie bei der Diode kann der Strom auch bei der LED nur in eine Richtung fließen. Die Leuchtfarbe einer LED wird vom Material der LED bestimmt. Deshalb gibt es LEDs in verschiedenen Farben.


LEDs werden heutzutage auch häufig als Beleuchtung eingesetzt, da sie sehr wenig Energie für das entstehende Licht benötigen.

## Bestückungsplan



## Material

- Leiterplatte LED-Ring
- 1x Schottky Diode (D1) 

**Hinweis:** Die Diode ist polarisiert. Schaue dass der Strich auf der Diode beim Punkt auf der Leiterplatte ist. 

## Der LED-Ring

Der LED-Ring ist neben micro:bit das Wichtigste des Lightbags. Er besteht aus 12 LED-Elementen, die alle miteinander in einer Kette verbunden sind. Jedes dieser Elemente besteht aus drei verschiedenfarbigen LEDs: Rot, Grün und Blau. Das erste LED-Element verbindet sich mit dem Pin 0 auf dem micro:bit und leitet alle Anweisungen, die du gleich programmieren wirst an die nachfolgenden LEDs weiter.

Findest du die beiden Verbindungsstellen 0 auf dem micro:bit und dem LED-Ring Stecker? Verbinde die beiden Teile miteinander.



### LED-Ring Speisung

Der LED-Ring verbraucht mehr Strom, als der micro:bit über den Verbindungsstecker zur Verfügung stellen kann. Deshalb musst du den Ring immer direkt mit einer Stromquelle verbinden. Das kann dein Computer, ein Handyladegerät oder ein Powerbank sein.

Der LED-Ring stellt dem micro:bit genug Strom zur Verfügung, damit ein bereits installiertes Programm ausgeführt werden kann. Wenn du ein neues Programm auf deinen micro:bit laden möchtest, musst du weiterhin den USB-Stecker am micro:bit verwenden.

Verstehst du hier den Anwendungsfall der Diode?

## Additive Farben

Du kannst bestimmt die Farbmischung mit Wasserfarben. Mit Rot Gelb und Blau kannst du jede mögliche Farbe bei den Wasserfarben mischen.

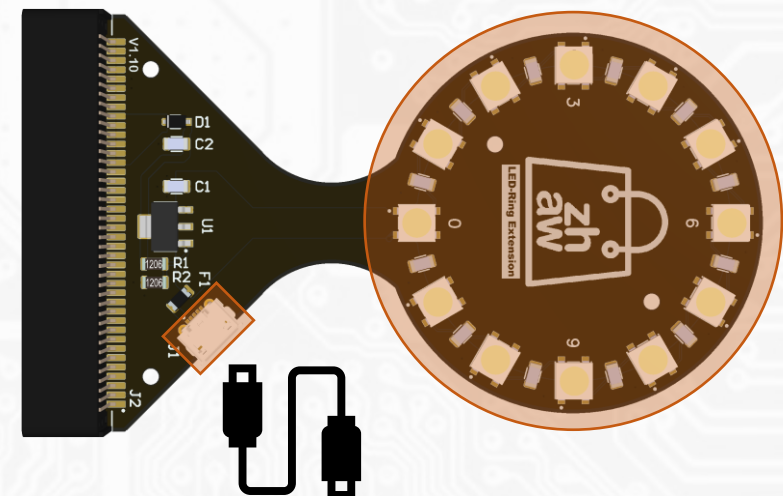


Bei farbigem Licht funktioniert die Farbmischung leicht anders. Mit den Lichtfarben Rot, Grün und Blau kannst du jede mögliche Farbe mit Licht mischen.

Siehst du den Unterschied der beiden Farbkreise?



## Bestückungsplan



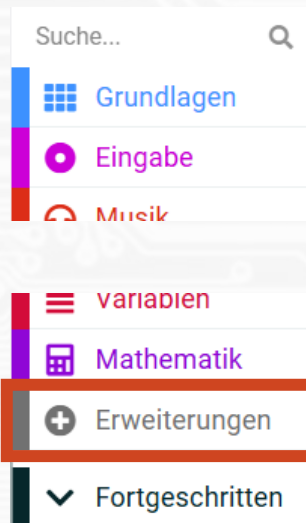
2.

## Erweiterung

Der Lightbag benötigt neue Funktionen zum Ansteuern der LEDs. Diese werden in sogenannten Erweiterungen definiert. Zum Hinzufügen der LED-Erweiterung musst du folgende Schritte machen.

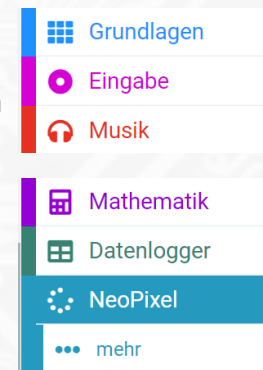
1.

- Klicke in der Befehlsgruppen Übersicht auf Erweiterungen

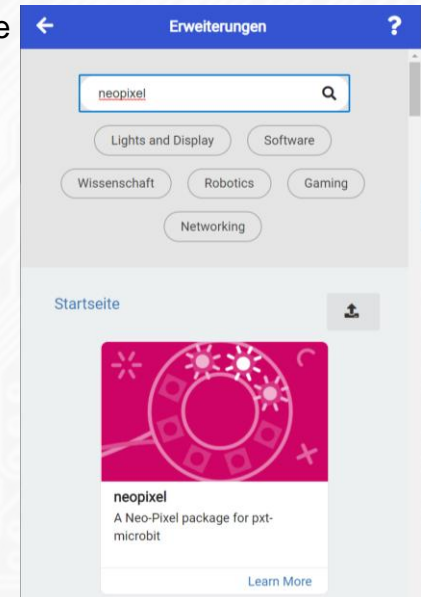


3.

Alle Funktionalitäten der Neopixel sind für dich in dieser Erweiterung zusammengestellt. Diese Erweiterungen erscheinen nun als neue Befehlsgruppe «Neopixel» Weitere Funktionen findest du gleich darunter in der Gruppe «mehr».



Suche nach Neopixel und füge die Erweiterung dem Projekt hinzu





## LED-Ring RGB Farben mischen

Dein LED-Ring hat 12 LEDs. Bringe nun vier von diesen LEDs zum Leuchten. Arbeite mit der vorhin eingerichteten Neopixel Erweiterung und nutze das Element aus den Tipps.

rot 255 grün 255 blau 255

LED an der Stelle 0: Rot  
LED an der Stelle 3: Orange  
LED an der Stelle 6: Blau  
LED an der Stelle 9: Grün

### Hinweise

Die Werte (Farben) der LED sind immer auf dem micro:bit gespeichert. Er schickt diese aber nur an die LEDs wenn du den Block

strip anzeigen verwendest.

Den Block

setze strip auf ... und Modus RGB (GRB Format)

brauchst du nur 1x im «beim Start» Block.

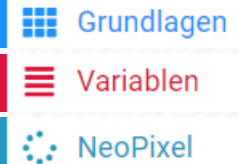
### Material

- micro:bit
- LED-Ring
- PC oder iPad

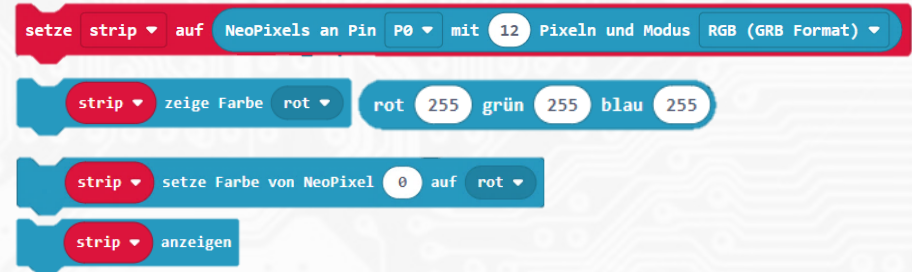


### Tipps

Verwendete Befehlsgruppen



Verwendete Blöcke



### Zusatzaufgabe

Definiere vier Bereiche (range) mit je 3 LEDs auf dem Ring und lasse diese in unterschiedlichen Farben leuchten. Kannst du auch unterschiedliche Helligkeiten einstellen?



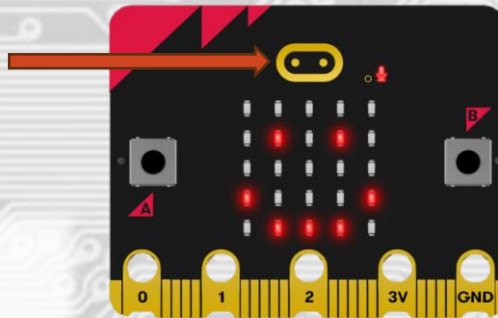
## Regenbogen Ein- und Ausschalten

Lasse auf deinem LED-Ring ein Regenbogen leuchten. Programme deinen micro:bit so, dass du den Regenbogen mit dem Berühren des Logos aus- und wieder einschalten kannst.

### Hinweise

Nutze eine Variable, um den Zustand des LED-Ringes zu speichern. Mit einem Befehl aus der Logik Gruppe entscheidest du danach, ob du den Ring ein- oder ausschaltest. Auch nach dem Abschalten, braucht es den Anzeigen Block.

### Touch Sensor

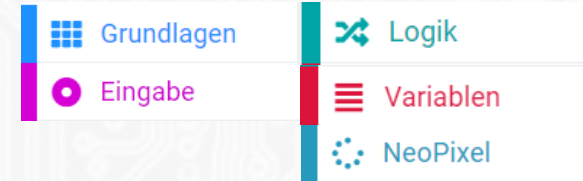


### Material

- micro:bit
- LED-Ring
- PC oder iPad

### Tipps

Verwendete Befehlsgruppen



Verwendete Blöcke



### Zusatzaufgaben

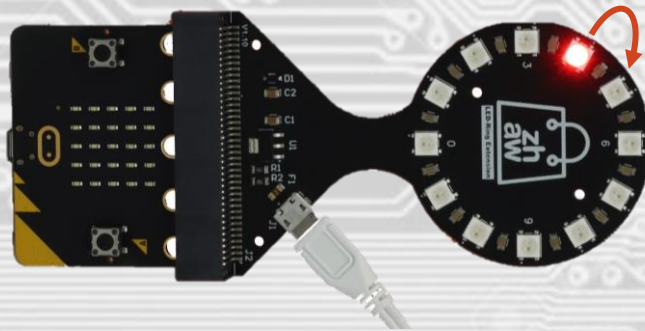
- Ändere die Helligkeit des Lichts. Klicke auf A und der Regenbogen wird heller. Klicke auf B und er wird dunkler.
- Was passiert, wenn die Helligkeit  $>255$  ist oder kleiner 0?  
Programmiere, dass die Helligkeit nicht  $< 0$  oder  $> 255$  werden kann.

## Lauflicht

Auf deinem LED-Ring soll immer nur eine LED leuchten. Immer nach einer Sekunde springt das Licht zur benachbarten LED und bewegt sich so endlos im Kreis.

### Hinweise

- Nutze eine Schleife mit einer Zählvariable (Index).



### Material

- micro:bit
- LED-Ring
- PC oder iPad

### Tipps

Verwendete Befehlsgruppen



Verwendete Blöcke



### Zusatzaufgaben

- Ändere bei Logoklick die Richtung.
- Programmiere das Lauflicht so, dass es nach jedem Schritt eine andere Farbe hat.
- Programmiere das Licht so, dass es langsam ausschaltet und währenddessen das nächste Licht langsam einschaltet. Tipp: Du brauchst dazu Variablen und Logikbauteile.
- Lasse immer zwei benachbarte LEDs gemeinsam im Kreis laufen. Wie löst du den Übergang von der LED 11 zur LED 0?



## Lautstärke anzeigen

Mit dem Mikrofon auf dem micro:bit kannst du die Lautstärke von deiner Umgebung messen. Zeige auf dem LED-Ring an, wie laut es ist. Je lauter, desto mehr LEDs sollen leuchten.

Wiederhole die Messung dauerhaft und aktualisiere deine LEDs.



### Hinweise

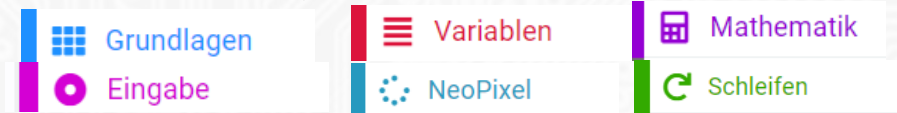
- Die gemessene Lautstärke auf dem micro:bit ist eine Zahl zwischen 0 und 255, wobei 0 leise und 255 laut ist.
- Deine 256 Lautstärkewerte musst du auf 12 LEDs verteilen. Wie viele Werte wählst du pro LED? Probiere verschiedene Werte aus.

### Material

- micro:bit
- LED-Ring
- PC oder iPad

### Tipps

#### Verwendete Befehlsgruppen



#### Verwendete Blöcke



### Zusatzaufgaben

- Zeige den Lautstärkewert zusätzlich auf deinem micro:bit als Zahl an.
- Spiele einen Warnton ab, wenn deine Umgebung zu laut wird.

## Fang das Licht

In der Challenge 4 hast du ein Lauflicht programmiert. Programmiere nun ein kleines Spiel: Ein zweites Licht wird über die Knöpfe A und B gesteuert. Wenn du auf B klickst bewegt sich dein Licht im Uhrzeigersinn, mit A auf die andere Seite. Wenn dein Licht vom Lauflicht eingeholt wurde ist das Spiel verloren. Überlege dir, wie du schwierigere Levels von diesem Spiel programmieren kannst.

Starte das Spiel beim Drücken von Taster A&B gleichzeitig.

### Zusatzaufgaben

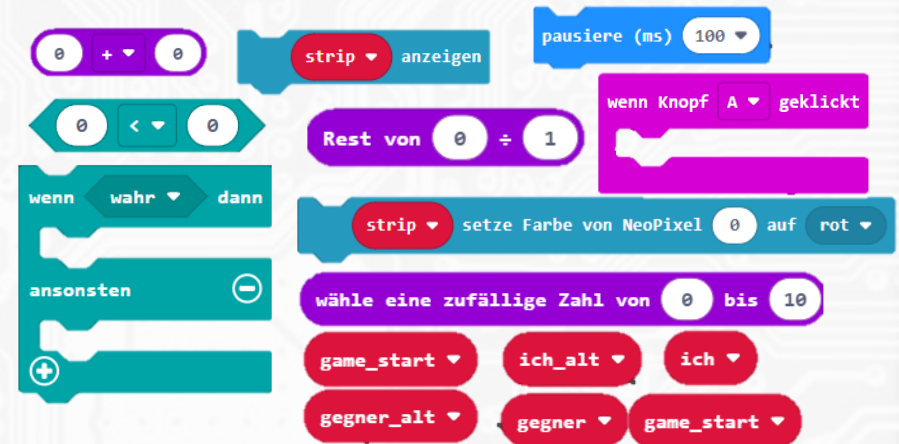
- Mache, dass du bei Taster A+B immer wieder neu starten kannst.
- Zähle, wie viele Schritte du zum Fangen des Lichtes benötigst und zeige es danach auf dem Display an.
- Ändere die Geschwindigkeit des Pixels, dass du einfangen musst zufällig. Dazu kannst du zum Beispiel die Pause im Bereich von 200...1000 ms verändern.
- Ändere die Richtung des Pixels, dass du fangen musst zufällig.
- Beweg den Gegner automatisch weg von der Bewegung des Spielers.
- Beweg den Gegner schneller, wenn der Spieler in der Nähe ist.
- Passe das Spiel so an, dass du gegen einen Freund spielen kannst. Dazu könnt ihr die Position des Gegners über Funk übertragen.

### Tipps

Verwendete Befehlsgruppen



Verwendete Blöcke



### Material

- micro:bit
- LED-Ring
- PC oder iPad

## Kompass

Hast du dich auch schon gefragt, ob es einen digitalen Kompass gibt? Auf dem micro:bit ist ein Kompasssensor verbaut. Zeige mit dem LED-Ring die Kompassnadel an! Der Norden soll wie beim richtigen Kompass Rot und der Süden Weiss angezeigt werden.

### Hinweise

Nutze eine Variable, um den Zustand des LED-Ringes zu speichern. Mit einem Befehl aus der Logik Gruppe entscheidest du danach, ob du den Ring ein- oder ausschaltest. Auch nach dem Abschalten, braucht es den Anzeigen Block.

### Tipps

abrunden ▾

Kompassausrichtung (°)

/ ▾

30

11

- ▾

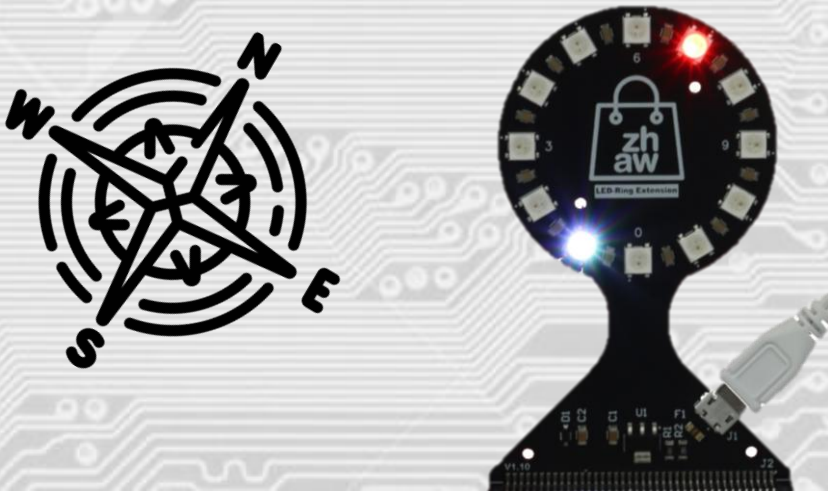
Grad ▾

### Material

- micro:bit
- LED-Ring
- PC oder iPad
- Magnet

### Zusatzaufgaben

- Was passiert, wenn einen Magneten um deinen micro:bit bewegst?
- Zeige zusätzlich auf der LED-Matrix die aktuelle Ausrichtung deines LED-Rings an. Schreibe dazu die Buchstaben N, O, S oder W auf die LEDs.





## Uhr



Programmiere eine Uhr, die mit einer Farbe die Stunde und mit einer anderen Farbe die Minute auf dem LED Ring anzeigt. Mit dem Knopf A soll die Stunde und mit dem Knopf B die Minute verändert werden können.

Überlege dir Antworten zu folgenden Fragen bevor du programmierst:

Nach wie vielen Minuten muss der Minutenzeiger eine LED weiterspringen?

Wann muss der Stundenzeiger eine LED weiterspringen?

Wie sollen deine LEDs leuchten, wenn beide Zeiger auf der gleichen LED sind?

## Zusatzaufgaben

- Erweitere deine Uhr um eine Timer Funktion. Nutze den Touch Sensor, um in den Timer Modus zu wechseln und den Timer zu starten.

## Material

- micro:bit
- LED-Ring
- PC oder iPad

## Tipps

wenn Knopf A  geklickt

pausiere (ms) 100



## Zusammenbau

## Mechanischer Zusammenbau «Bag»

### 1. LED-Ring festschrauben

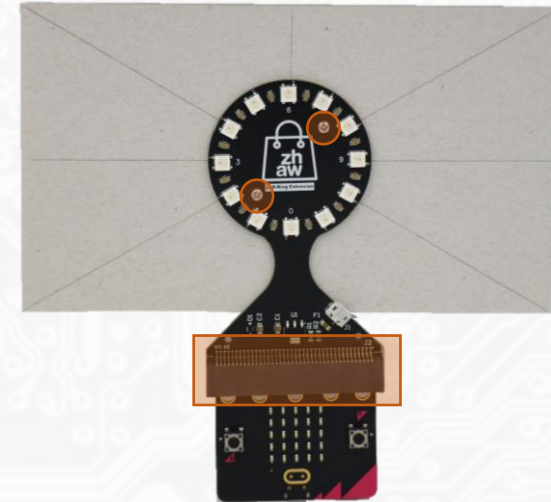
Schraube die Leiterplatte mit zwei Schrauben auf den Kartonboden.

### 2. Bag

Lege deinen Aufbau auf dem Kartonboden in den Papiersack. Der Sack hat bereits ein Loch, durch das du die LED-Ring Leiterplatte schieben kannst. Zum Schluss kannst du noch das USB-Kabel mit dem Schalter mit der Leiterplatte verbinden.

### 3. micro:bit verbinden

Stecke deinen LED-Ring auf die Stecker an der Leiterplatte. Achte darauf, dass die LEDs vom Ring und die LEDs vom micro:bit in die gleiche Richtung schauen. Deine neue Nachttischlampe ist fertig, du kannst sie mit jedem USB-Ladegerät, einem Powerbank oder mit einem Computer verbinden. Mit der Challenge 10 kannst du dir ein Programm für die Nachttischlampe auf dem micro:bit installieren.



### Material

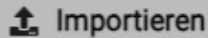
- LED-Ring Leiterplatte
- micro:bit
- Kartonboden
- 2x M2 Schraube (Länge 6 mm) mit Muttern
- Papiersack
- USB-Kabel mit Schalter

## Nachttischlampe

## Nachttischlampe installieren

Mit dieser Challenge installierst du deine Nachttischlampe.

1. Öffne die Makecode Startseite:  
<https://makecode.microbit.org>
2. Klicke rechts auf den Knopf «Importieren»
3. Wähle im neuen Menü «Importiere URL...»
4. Tippe folgende URL in das Feld ein:  
[https://github.com/InES-HPMM/zhaw\\_lightbag](https://github.com/InES-HPMM/zhaw_lightbag)
5. Klicke auf «Los geht's!»
6. Nun kannst du das Programm wie gewohnt auf deinen micro:bit herunterladen.
7. Was passiert, wenn du die Knöpfe A, B oder den Touch Sensor betätigst?

A grey button with a white upward arrow icon and the text "Importieren".

### Zusatzaufgaben

Schau dir das Programm an. Verändere und erweitere es damit du deine eigene einzigartige Nachttischlampe erhältst. Sieh dir auch die Challenge 11 an für weitere Ideen.



### Material

- micro:bit
- Lightbag zusammengebaut
- PC oder iPad



## Nachttischlampe erweitern

Erweitere deine Nachttischlampe mit eigenen Funktionen. Wenn dir keine Funktion einfällt kannst du auch eine der folgenden umsetzen.

Schalte deine Lampe mit einem lauten Klatschen Ein oder Aus. Miss dazu die Lautstärke mit dem Mikrofon auf dem micro:bit.

Schaue dir die Funktion «wechsel\_farbig» an und programmiere dein eigenes Licht.

Programmiere einen Einschlaf-Timer. Deine Lampe soll sich nach einer gewissen Zeit automatisch abschalten. Hast du eine Idee, wie du mit den Tasten A und B die Zeit einstellen kannst?

### Material

- micro:bit
- Lightbag zusammengebaut
- PC oder iPad

### Tipps

Frage bei den Workshopleitern nach Unterstützung für deine Idee.

### LED-Matrix

Hat dir der Workshop gefallen? Dann schau dir unbedingt unseren LED-Matrix Workshop an.



Alle Infos findest du unter:

[www.zhaw.ch/ines/led-matrix](http://www.zhaw.ch/ines/led-matrix)

