

# Programmieren mit BBC Micro:Bit

Thema: Einführung zum BBC Micro:Bit, Schleifen und Verzweigungen

**BBC**

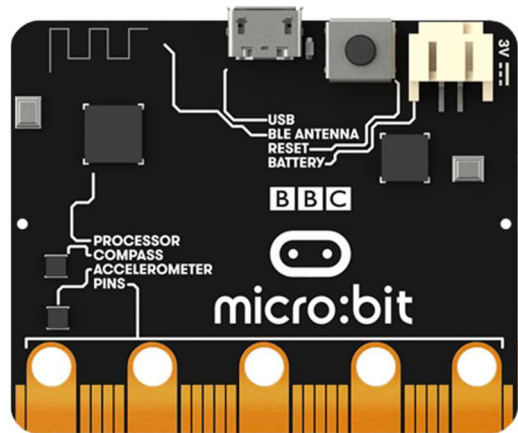
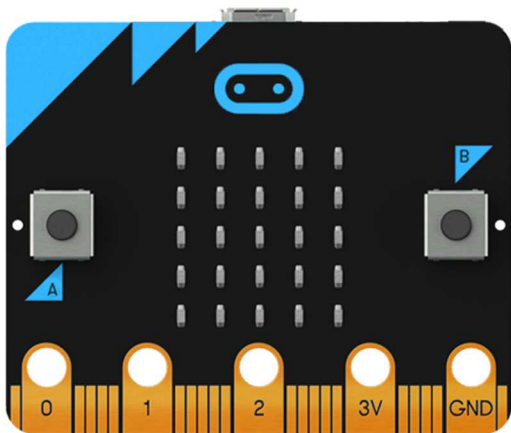


micro:bit

## **Einleitung:**

In diesem Leitprogramm lernst du die ersten Schritte zur Programmierung mit dem BBC Micro:Bit kennen. Dabei lernst du unter anderem den physischen Aufbau und die wichtigsten Bestandteile des Micro:Bits, den dazugehörigen Editor mit seinen Funktionen, die Funktion und Anwendung einer Schleife und einer Verzweigung mit bestimmten Bedingungen. Am Ende des Leitprogramms hast du letztendendes ein vollständiges Programm und du findest am Ende noch ein Lösungsbeispiel, jedoch muss deine Lösung nicht unbedingt identisch sein und kann davon abweichen. Du kannst deine Lösung aber mit dem Beispiel vergleichen und überprüfen, ob dein Ansatz in etwa richtig ist. Zum Schluss gibt es noch einen Kapiteltest mit dem du überprüfen kannst, ob du alles auch richtig verstanden hast.

## Kapitel 1: Aufbau vom BBC Micro:Bit



Der BBC Micro:Bit besteht aus vielen einzelnen kleinen Schnittstellen und Teilen. In diesem Kapitel werden wir uns jedoch nur mit den für uns relevanten Bestandteilen beschäftigen.

Ein wichtiger Bestandteil ist der obere Micro USB-Anschluss. Dieser kann mithilfe eines entsprechenden Kabels mit einem Computer verbunden werden, wodurch die programmierten Programme als Datei per Drag&Drop ähnlich wie bei einem USB Stick zum Micro:Bit übertragen werden und erst durch diese Verbindung Strom erhält.

Als nächstes fallen die 25 kleine Streifen auf, welche sich mittig befinden. Diese sind LED's, welche einzeln gesteuert und als Ausgabegerät genutzt werden können.

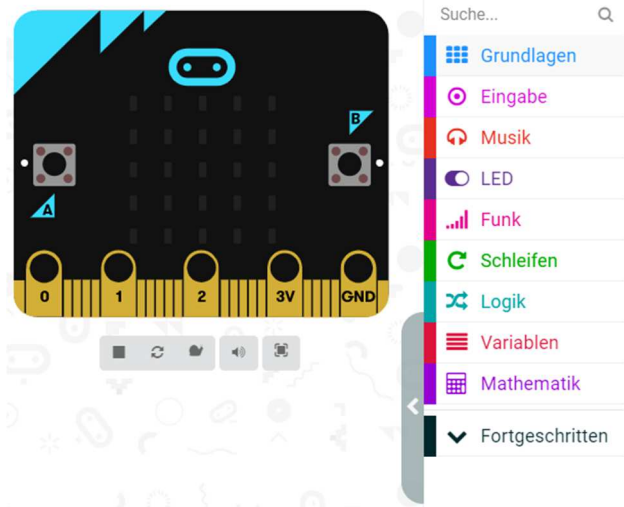
Jeweils links und rechts daneben findest du eine Markierung A und B. Diese markieren die Tasten, welche einzeln oder auch zusammen als Eingabegerät genutzt werden können.

Zuletzt ist noch die Taste wichtig, welche sich hinten neben des Micro USB-Anschlusses befindet, da dies die Reset Taste ist und durch das Drücken dieser das ganze Programm neugestartet wird.

## Kapitel 2: Funktionen vom BBC Micro:Bit

Ein weiterer Vorteil des BBC Micro:Bits ist die Programmierung, da du die Programme zwar mit JavaScript programmieren kannst, aber nicht zwangsweise musst, sondern auch den Blockeditor nutzen kannst. D.h. du musst anstelle von ganzen Blöcken Code lediglich einige Bausteine hinzufügen und sie zusammensetzen. Den Editor kannst du ganz einfach im Browser deiner Wahl unter folgendem Link aufrufen: <https://makecode.microbit.org/#editor>.

Auf der Seite kannst du dich Anmeldung ein neues Projekt erstellen und sofort anfangen. Zu Beginn sollte dir auffallen, dass du links eine Abbildung eines Micro:Bits siehst. Mit diesem kannst du dein Programm im Echtzeit testen und auf diesem simulieren. Somit musst du das Programm nicht jedes Mal erst herunterladen und auf deinen eigenen



Micro:Bit übertragen. Zusätzlich hat es den Vorteil, dass du nicht immer einen Micro:Bit brauchst und so auch bspw. zu Hause weiter an deinem Programm arbeiten kannst.

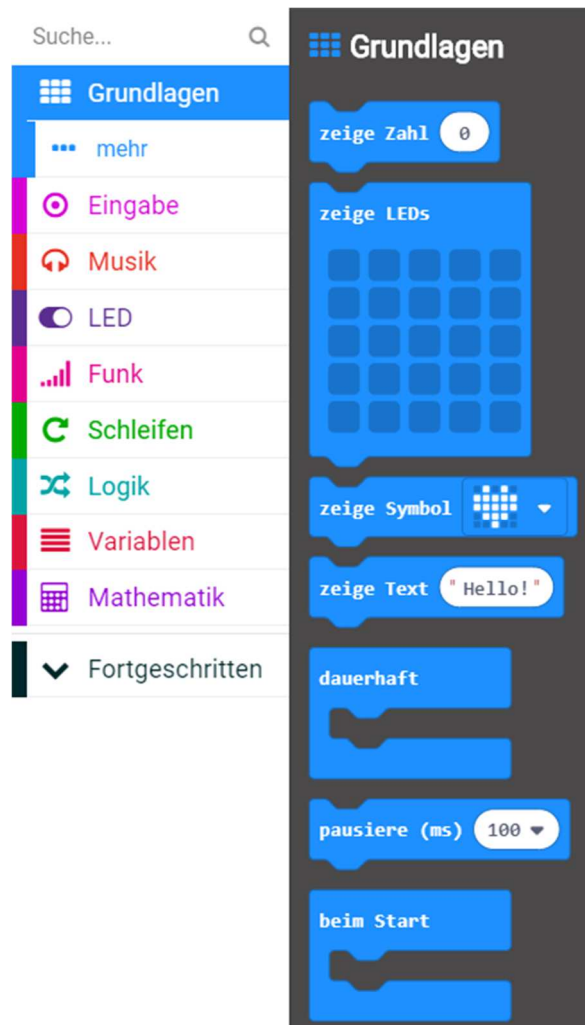
Gleich rechts daneben siehst du eine Liste mit bestimmten Kategorien. Innerhalb dieser Kategorien kannst du weitere entsprechende Blöcke finden, welche du nutzen kannst.

Für die heutige Unterrichtsstunde wirst du dich aber nur mit den Kategorien *Grundlagen* und *Eingabe* beschäftigen.

Bei der Kategorie *Grundlagen* sind folgende Blöcke relevant: Zeige Zahl, zeige LEDs, zeige Symbol, zeige Text, dauerhaft und pausiere (ms).

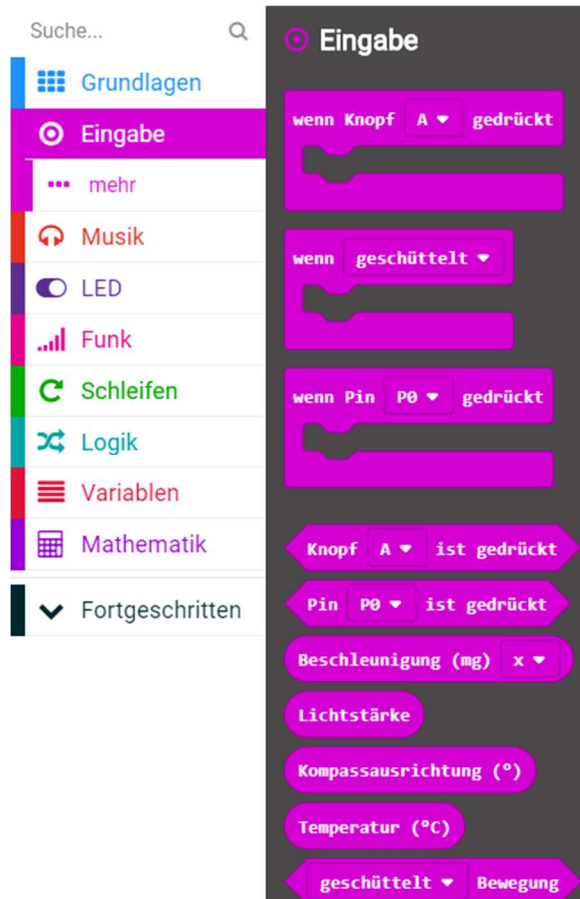
Mithilfe von zeige Zahl kannst du eine bestimmte Zahl über die LEDs anzeigen. Mit zeige LEDs kannst du selbst sagen, welche LEDs leuchten und was für ein Bild sie zeigen sollen. Über zeige Symbol kannst du ein vorgefertigtes Symbol auswählen und dir anzeigen lassen. Zeige Text funktioniert hier ähnlich wie zeige Zahl, nur kannst du hier zusätzlich auch Texte anzeigen lassen.

Dauerhaft ist hier eine unendliche Schleife, dazu mehr aber im nächsten Kapitel. Zum Schluss sorgt pausiere (ms) dafür, dass dein Skript für die jeweilige Zeit pausiert wird.



Bei der Kategorie *Eingabe* ist nur der Block wenn Knopf gedrückt wichtig, da hier als Eingabe nur die Tasten verwendet werden und keine andere Eingabequellen, wie das Schütteln oder die Pins.

Bei dem Befehl wenn Knopf gedrückt kannst du einstellen, welches Skript ausgeführt werden soll, wenn beispielsweise die Taste A gedrückt wird. Hierbei handelt es sich um eine Verzweigung, mehr dazu aber erst im übernächsten Kapitel.



### Kapitel 3: Schleifen

Um zu verstehen, was eine Schleife ist und was sie kann, stell dir folgendes vor: Du würdest einen menschlichen Körper programmieren und müsstest ihm sagen, dass er die ganze Zeit atmen soll. Dabei würdest du es so kodieren, dass der Körper den Befehl atmen immer wieder ausführen soll. Genau diese Ausführung ist eine Schleife, also eine mehrfache Ausführung eines bestimmten Befehls oder Befehlen.

Solche Schleifen können wir natürlich auch zum Beispiel bei der Programmierung mit dem BBC Micro:Bit gebrauchen. Ein Beispiel hierfür wäre, dass wir mithilfe des Programms dafür sorgen möchten, dass ein LED-Effekt angezeigt werden soll. D.h. hier soll die LED-Anzeige unterschiedliche Bilder die ganze Zeit hintereinander anzeigen.

Aufgabe 1: Versuch ein Programm zu programmieren, welches das obengenannte Beispiel veranschaulicht. Alle dazugehörigen Befehle kennst du bereits.

## **Kapitel 4: Verzweigungen**

Wie du bereits gelernt hast, sollte dein Programm Befehle durchgehend ausführen können und nicht nur einmal alles durchlaufen. Damit du nun aber auch die Tasten richtig nutzen kannst, musst du diese über Verzweigungen implementieren.

Aber kommen wir erstmal kurz zu Verzweigungen selbst. Stell dir dafür wieder folgendes vor: Du bist wieder dabei einen menschlichen Körper zu programmieren und hast ihn dazu gekriegt, dass er in einer Schleife ein- und ausatmet. Da wir aber schlaue programmieren möchten, soll der Körper die Befehle nicht einfach grundlos befolgen, sondern aus einem bestimmten Grund. Dieser ist natürlich, dass er Luft braucht. D.h. er würde nur einatmen, wenn er nicht genug Sauerstoff hat und ausatmen, wenn er die verbrauchte Luft loswerden möchte. Genau diese Unterscheidung ist eine Verzweigung. Mit anderen Worten sagst du, dass wenn der Körper Sauerstoff braucht, er einatmen soll und wenn er Luft loswerden will, er ausatmen soll.

Jetzt sollte dir auffallen, warum wir für die Nutzung der Tasten Verzweigungen brauchen. Wir sagen dem Programm, dass wenn die Taste A bspw. gedrückt wird, bestimmte Befehle befolgt werden sollen.

Aufgabe 2: Beim Drücken der A-Taste sollen 5 fertige Symbole hintereinander mit jeweils einer Sekunde Pause angezeigt werden. Die Symbole kannst du dir frei aussuchen.

Aufgabe 3: Beim Drücken der B-Taste sollen Zahlen und Texte angezeigt werden. Sei hier ruhig kreativ und lass dir etwas tolles einfallen.

Aufgabe 4: Beim gleichzeitigen Drücken der Tasten A und B soll irgendeinen LED-Effekt auftreten. Lass auch hier deiner Kreativität freien Lauf und versuche doch etwas ganz Besonderes. Du kannst hier alles benutzen!

**Lösungsbeispiel:**

The image shows a Scratch script on a grid background. It consists of four main vertical sections:

- dauerhaft (Forever Loop):** Contains two "zeige LEDs" blocks. The first block shows a 4x4 grid of 16 white squares. The second block shows a 4x4 grid of 16 dark blue squares.
- wenn Knopf A gedrückt (When Button A is Pressed):** Contains a sequence of four blocks: "zeige Symbol" (a 4x4 grid of 16 white squares), "pausiere (ms) 1000", "zeige Symbol" (a 4x4 grid of 16 dark blue squares), and "pausiere (ms) 1000".
- wenn Knopf B gedrückt (When Button B is Pressed):** Contains a single "zeige Text" block with the text "Error 404".
- wenn Knopf A+B gedrückt (When Button A+B is Pressed):** Contains a sequence of three blocks: "zeige LEDs" (4x4 grid of 16 dark blue squares), "zeige LEDs" (4x4 grid of 16 white squares), and "zeige Symbol" (a 4x4 grid of 16 dark blue squares). Below this is a "pausiere (ms) 1000" block.



## **Kapiteltest:**

Überprüfe mit diesem Test, ob du bis hierhin wirklich alles verstanden hast und es auch anwenden kannst.

Aufgabe 1: Erstelle ein neues Projekt und benenne es „Kapiteltest“.

Aufgabe 2: Füge eine Schleife hinzu, welche durchgehend die Zahl 1 anzeigt.

Aufgabe 3: Wenn die Taste A gedrückt wird, soll der Text „Error – Text fehlt“ angezeigt werden

Aufgabe 4: Wenn die Taste B gedrückt wird soll ein Herz für 2 Sekunden gezeigt werden

Aufgabe 5: Füge zur Schleife hinzu, dass nach der Zahl ein Dreieck für eine halbe Sekunde gezeigt werden soll.

Aufgabe 6: Wenn die Tasten A und B gleichzeitig gedrückt werden, soll dein Programm eine Pause von 10 Sekunden machen.

Aufgabe 7: Füge zur A Taste und zur B Taste hinzu, dass jeweils 2 eigene Symbole angezeigt werden sollen.