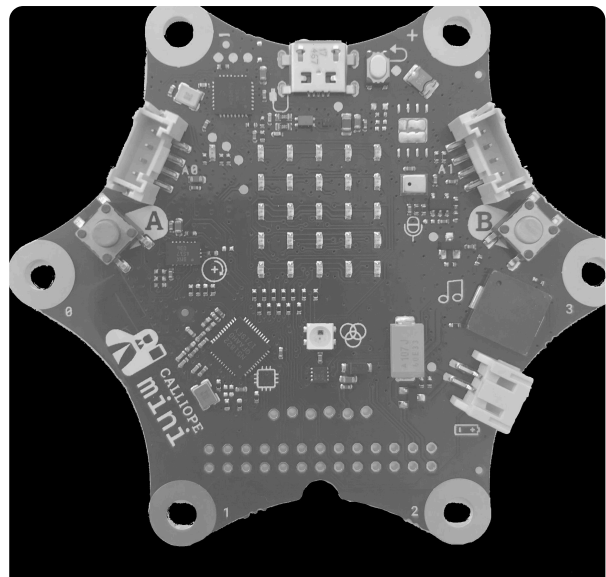


Theorie - was du können solltest

- Elemente auf der Platine kennen
- Fachbegriffe erklären können / Beispiele nennen, z.B. Sensoren, Wiederhole-Schleife, Variablen, Wenn-Sonst-Entscheidung, Logik-Bausteine
- Programme kommentieren (Übung auf Seite 2)

Du kannst alle Programme aus dem Unterricht einsehen unter <https://teach.appcamps.de/karten>

mit den bekannten Kennwörtern aus dem Lehrergang



Praxis - eine Übung

① Programm „Pusten“

- Wenn du auf die Platine pustest, erscheint ein Herz. Probiere Geräusch > 50. (Eventuell musst du den Wert ein wenig anpassen.)
- Nach der Anzeige des Herzens wird nach 500 ms der Bildschirm gelöscht.
- Es soll gezählt werden, wie viele Herzen erschienen sind.
- Die Anzahl der erschienenen Herzen wird angezeigt, wenn man Pin 1 drückt (an der Platine: Pin 1 gleichzeitig mit dem minus-Pin anfassen).
- Wenn die Taste A gedrückt wird, wird der Zähler wieder auf 0 gesetzt. Es erscheint ein Bild und die LED geht in Rot an und wieder aus. Ein Ton wird abgespielt.
- Mögliche Erweiterung: Wenn der Zähler bei 10 ist, passiert etwas - denk dir etwas aus.
- Exportiere das Programm. Suche es im download-Ordner und nenne es „vorname_pusten“

② Speichern und abgeben

- Das Programm heißt vorname_pusten
- Erstelle auf p: einen Ordner vorname_uebung
- Kopiere das fertige Programm in deinen Ordner vorname_uebung
- Kopiere einen gesamten Ordner und füge ihn ein unter Uebergabe

Kommentiere das Programm. Nenne die Funktion der einzelnen Bausteine.

The code starts with a 'Start' block, followed by two variable initialization blocks: 'Zufall' (random) set to 0 and 'Spielstand' (game state) set to 0. A 'Wiederhole unendlich oft' (repeat forever) loop begins. Inside the loop, a 'wenn' (if) block checks 'gib geschüttelt' (shake) 'Lage' (position). This is followed by a 'Zeige Zeichen' (show character) block for 'Spielstand', a 'Warte ms' (wait) block for 500ms, and a 'Zeige Bild' (show image) block displaying a 5x5 grid. The grid is mostly empty, with '#' symbols in the top row and middle column. After another 500ms wait, a 'Schreibe Zufall' (write random) block sets 'Zufall' to a random integer between 0 and 1. A 'wenn' block then checks if 'Zufall' is 0. If true, it shows the grid with '#' symbols in the top row and middle column. If false, it shows the grid with '#' symbols in the top row and middle column, and also in the second column. The loop ends with 'pruefeLinks' and 'pruefeRechts' blocks.

The 'pruefeLinks' block starts with a 'wenn' (if) block checking 'Taste A gedrückt?' (key A pressed) and 'nicht Taste B gedrückt?' (not key B pressed). Inside, a 'wenn' block checks if 'Zufall' is 0. If true, it 'Schreibe Spielstand' (write game state) to 'Spielstand + 1', plays a sound 'Spiele' at 440 Hz for 500ms, and shows the grid with '#' symbols in the top row and middle column. If false, it shows the grid with '#' symbols in the top row and middle column. The 'pruefeRechts' block starts with a 'wenn' (if) block checking 'Taste B gedrückt?' (key B pressed) and 'nicht Taste A gedrückt?' (not key A pressed). Inside, a 'wenn' block checks if 'Zufall' is 1. If true, it 'Schreibe Spielstand' (write game state) to 'Spielstand + 1', plays a sound 'Spiele' at 440 Hz for 500ms, and shows the grid with '#' symbols in the top row and middle column. If false, it shows the grid with '#' symbols in the top row and middle column.