



Startvideo

Einführung

Wir wollen heute mit Calliope mini das Spiel Heißer Draht bauen.

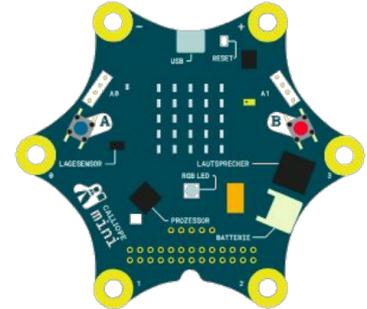
Dabei schlüpfst du in verschiedene Rollen des Berufslebens.

Als **ProgrammiererIn** programmierst du das Spiel mit einer Zeitmessung und einem Fehler-Punktstand auf dem Calliope mini.

Als **IngenieurIn** gestaltest du den physikalischen Aufbau des Heißeren Drahts.

Als **Game-DeveloperIn** überlegst du dir den Spielaufbau und die Bedienung für die Spieler.

Und natürlich wirst du als **GamerIn** das Spiel auch testen. 😊



Programm 1.0

Aufgabe 1: Erstelle ein Programm, das ein X auf der LED-Matrix des Calliope anzeigt und einen Signalton spielt, solange der Pin 1 gedrückt wird (d.h. Heißer Draht berührt wird).

Schritt 1: Öffne die Programmierumgebung. Gib hierzu lab.open-roberta.org in deinen Browser ein und wähle dann Calliope (2017) als System.

Schritt 2: Benutze, wie bei fast jedem Programm den *“Wiederhole unendlich oft mache”*-Block und erstelle ein Programm, dass ein X auf der LED-Matrix ausgibt, wenn Pin 1 gedrückt wird.

Schritt 3: Zusätzlich soll noch, solange Pin 1 gedrückt wird, ein Signalton ausgegeben werden. Hierzu kannst du den *“Wiederhole bis/solange ... mache”*-Block verwenden.

Schritt 4: Damit das X auf dem Bildschirm auch wieder verschwindet, füge nach dem *“Wiederhole solange ... mache”*-Block einen *“Lösche Bildschirm”*-Block ein.

Teste dein Programm auf dem Calliope mini.

Schritt 5: Lagere die gerade erstellte Funktionalität in eine Funktion *“heisserDraht”* aus.

Teste dein Programm erneut auf dem Calliope mini. Es sollte alles so funktionieren wie vorher.

Ergebnis



Wenn du alles richtig gemacht hast, sollte dein Ergebnis so aussehen.



Stoppuhr

Aufgabe 2: Programmiere eine Stoppuhr mit den folgenden Funktionen:

- Wenn Pin 0 gedrückt, setze die Stoppuhr zurück und starte die Stoppuhr.
- Wenn Taste A gedrückt, gib die aktuelle Zeit aus.
- Wenn Pin 2 gedrückt, "stoppe" die Zeit.
- Wenn Taste B gedrückt, gib die gestoppte Zeit aus.

Der Calliope Mini hat einen eingebauten Zeitgeber, den wir für die Stoppuhr verwenden werden.

Schritt 1: Um diesen **Zeitgeber zurückzusetzen** benutzen wir den Block **Setze Zeitgeber 1 zurück** den du unter **Sensoren** findest. Das soll passieren, wenn Pin 0 gedrückt wird. Also musst du den Block **Setze Zeitgeber 1 zurück** in einen **wenn ... mache ...** Block einbauen.

Hinweis: Sobald der Zeitgeber auf 0 zurückgesetzt ist, läuft dieser automatisch weiter. Du musst den Zeitgeber nicht extra starten.

Schritt 2: Um jetzt die **aktuelle Zeit auszugeben** kannst du den Block **gib Wert ms Zeitgeber 1** verwenden.

Um diesen Wert in Millisekunden (ms) in Sekunden (s) umzurechnen, musst du den Wert durch 1000 teilen. Hierzu findest du die nötigen Blöcke im Bereich **Mathematik**.

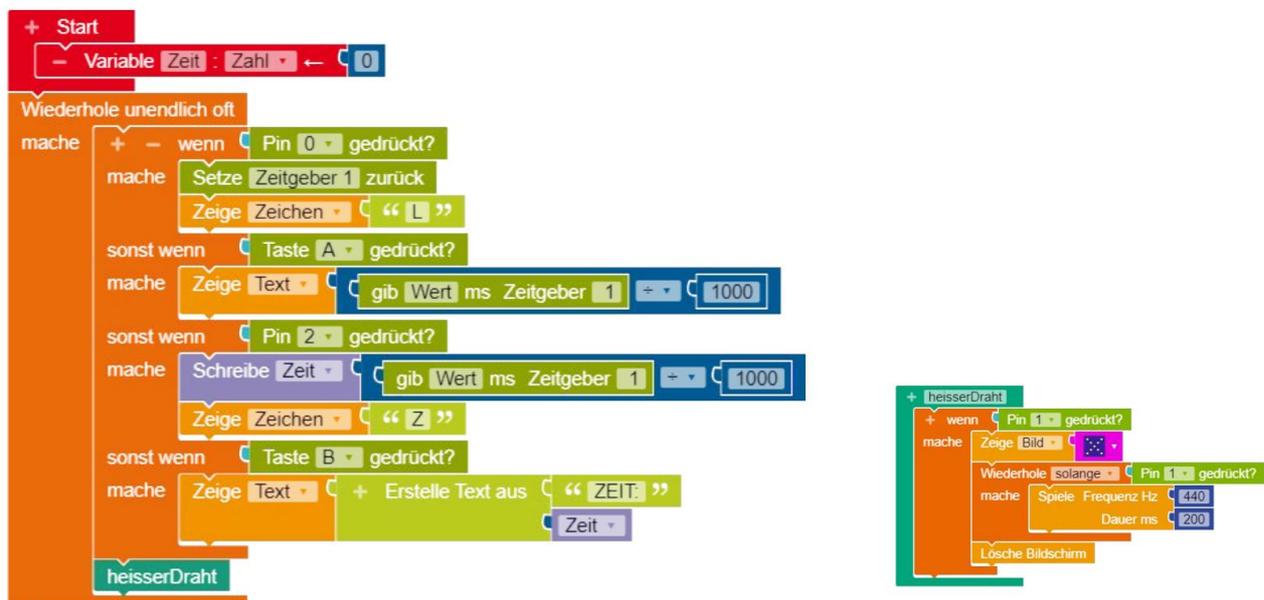
Schritt 3: Um die **Zeit zu stoppen**, müssen wir den aktuellen Wert des Zeitgebers in einer Variable speichern.

Füge deinem Programm eine Variable **"Zeit"** vom Typ **"Zahl"** hinzu indem du im **+ Start** -Block auf + drückst. Speichere in dieser Variable **"Zeit"** den Wert des Zeitgebers in Sekunden.

Schritt 4: Um die **gestoppte Zeit auszugeben**, verwenden wir die Blöcke **Zeige Text ...** und **Erstelle Text aus ...**. Die gestoppte Zeit haben wir ja in Schritt 3 in der Variable **"Zeit"** abgespeichert. Daher lassen wir uns jetzt den Wert der Variable **"Zeit"** ausgeben.

Teste dein Programm auf dem Calliope mini.

Ergebnis





Programm 2.0 - Der Spielaufbau

Aufgabe 3: Bau die Stoppuhr in den Spielaufbau des HeiÙen Drahtes ein.

Schritt 1: Erstelle eine Funktion "starteZeitmessung" und eine Funktion "stoppeZeitmessung".

Schritt 2: Erstelle eine Funktion "spiel". In dieser Funktion gestalten wir jetzt den Spielaufbau.

Hinweis: Unten findest du einen Vorschlag für den Spielaufbau mit Erklärungen.

Schritt 3: Die Funktion "spiel" soll gestartet werden, wenn Taste A gedrückt wird.

Schritt 4: Wenn Taste B gedrückt wird, soll die gestoppte Zeit ausgegeben werden.

Ergebnis

Schritt 1:

```
+ starteZeitmessung
  Setze Zeitgeber 1 zurück
  Zeige Zeichen " L "

+ stoppeZeitmessung
  Schreibe Zeit gib Wert ms Zeitgeber 1 1000
  Zeige Zeichen " Z "
```

Schritt 2:

```
+ spiel
  Zeige Zeichen " A " ← Auf-die-Plätze: Bitte begib dich mit dem Handstück zum Startpunkt des HeiÙen-Drahts.
  + Warte bis gib gedrückt Pin 0 = wahr ← Programm wartet, bis Handstück am Startpunkt ankommt.
  Zeige Zeichen " F " ← Fertig: Der Spieler berührt mit dem Handstück den Startpunkt und ist fertig für das Spiel.
  + Warte bis gib gedrückt Pin 0 = falsch ← Programm wartet, bis Handstück den Startpunkt verlässt.
  starteZeitmessung ← Los: Auf Los geht's los. Die Zeitmessung wird gestartet.
  Wiederhole bis Pin 2 gedrückt? ← Bis der Spieler am Zielpunkt ankommt, wiederhole den HeiÙen Draht.
  mache heisserDraht
  stoppeZeitmessung ← Ziel: Stoppe die Zeitmessung, da die vorherige Schleife beendet ist, hat der Spieler den Zeitpunkt erreicht.
```

Schritt 3 & 4:

```
+ Start
  - Variable Zeit : Zahl ← 0

Wiederhole unendlich oft
  mache
    + - wenn Taste A gedrückt?
      mache spiel
    sonst wenn Taste B gedrückt?
      mache Zeige Text + Erstelle Text aus " ZEIT: "
        Zeit
```

Hinweis: Ist das Spiel erst einmal über die Taste A gestartet, kann es nicht abgebrochen werden. Der Spieler muss erst das komplette Spiel durchlaufen, bevor er mit Taste B die Zeit ausgeben kann oder mit Taste A ein neues Spiel beginnt.



Programm 2.1 - Der Fehler-Punktstand

Aufgabe 4: Das Programm soll zahlen, wie haufig die Spieler wahrend eines Spiels gegen den HeiÙen Draht gekommen sind. Also wieviele Fehlerpunkte gemacht wurden. Dieser Fehler-Punktstand soll zusammen mit der Zeit ausgegeben werden.

Schritt 1: Fuge deinem Programm eine Variable *“Fehlerpunkte”* vom Typ Zahl und dem Startwert 0 hinzu.

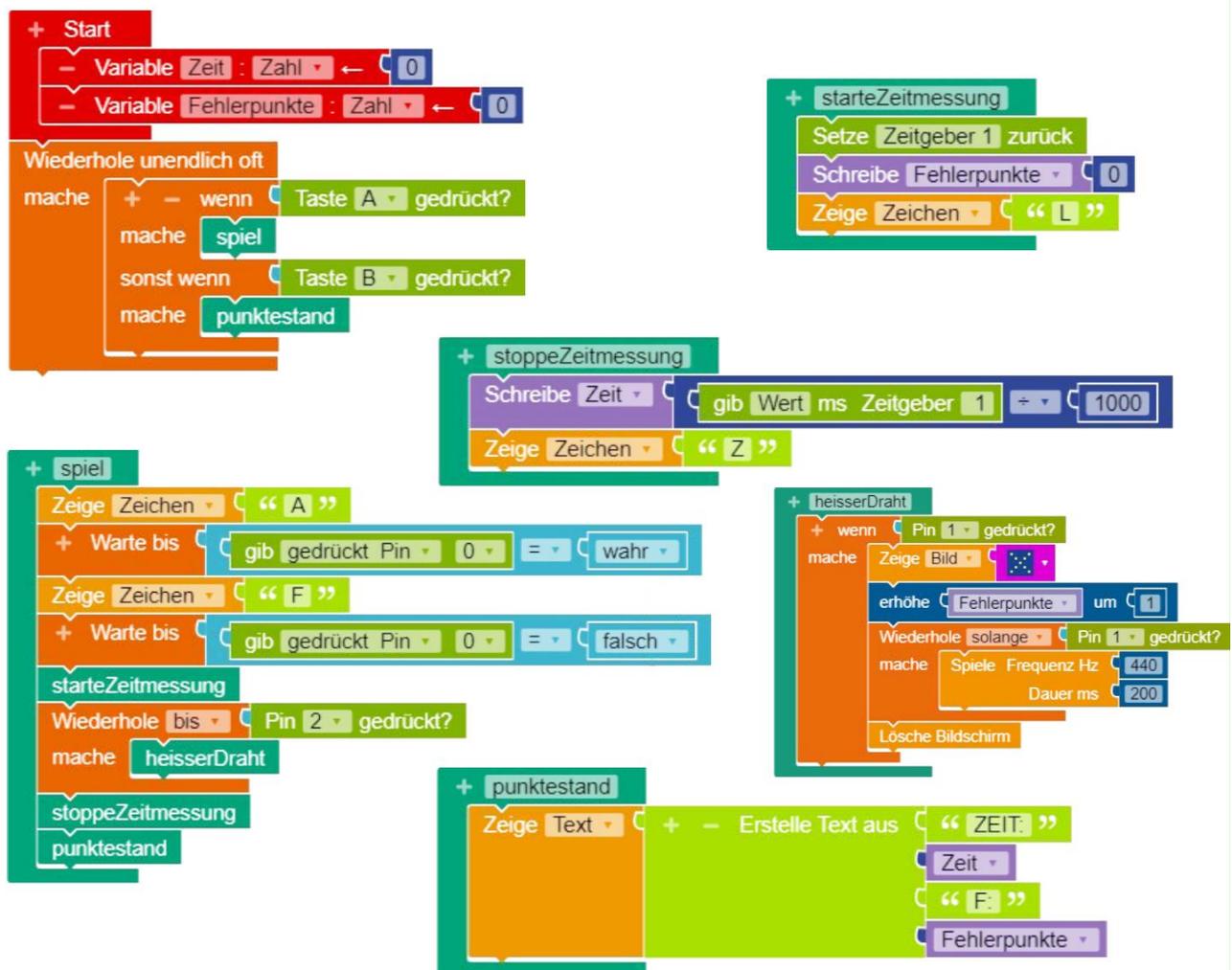
Schritt 2: Erhohe die Variable *“Fehlerpunkte”* bei jedem Beruhren des HeiÙen Drahtes (Pin 1). Hierzu kannst du den Block verwenden, den du unter *Mathematik* findest.

Schritt 3: Beginnt ein neues Spiel muss der Fehler-Punktstand auf 0 zuruckgesetzt werden. Fuge daher den Block zu deiner Funktion *“starteZeitmessung”* hinzu.

Schritt 4: Erweitere deine Ausgabe um die Textbausteine *“F:”* und den Wert von *“Fehlerpunkte”*.

Schritt 5: Lagere die Ausgabe der Zeit und der Fehlerpunkte in eine Funktion *“punktstand”* aus, die sowohl beim Drucken von Taste B als auch am Ende jedes Spiels ausgefuhrt wird.

Ergebnis





Aufbau

Super, mit der Programmierung des Heißen Drahts bist du jetzt fertig. Jetzt musst du natürlich noch den Heißen Draht basteln. Dafür brauchst du:

- Draht (z.B. Wickeldraht verzinkt)
- Klebestreifen
- Pappe (z.B. von einem Versandpaket)

Unten siehst du den Aufbau und zwei Beispiele. Du kannst kreativ sein - das kann bei dir natürlich auch etwas anders aussehen.

Aufgabe 5: Erstelle den Aufbau des Heißen Drahts.

