

Info

Ein Kreisel sensor kann die Neigung eines Objekts in verschiedenen Achsen messen. Umgangssprachlich wird ein Kreisel sensor oft als Lagesensor bezeichnet.

Hinweis

Damit das Programm mit einem Vergleichsoperator funktioniert wird der **Absolutwert** berechnet. Dieser ist sowohl von negativen wie positiven Zahlen immer positiv. Eine Abweichung egal ob -20 Grad oder +20 Grad ist somit mathematisch immer 20 Grad.

Aufgabe

Eierlaufen kennt jeder: Jetzt kannst du beweisen, ob du ein ruhiges Händchen hast um den Calliope mini sicher von A nach B zu bringen.

- Diesmal verwenden wir den Kreisel sensor des Calliope mini, der die Neigung in x- und y-Achse des Minicomputers messen kann.
- Zu Beginn ist uns der Neigungswinkel des Calliope mini unbekannt. Daher bekommt die **Variable** den Wert 0.
- Verwende einen **logischen Operator** um die Neigung des Calliope minis zu messen, die du in der **Variable** zwischengespeichert hast.
- Wähle eine Aktion aus, die ausgeführt wird, wenn die Schiefelage des Calliope mini zu groß wird. Experimentiere mit **Farben der LED** und **Tönen**.

```

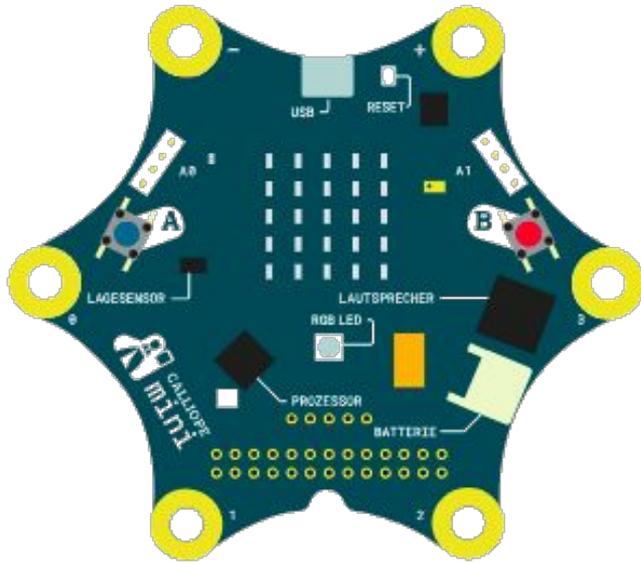
+ Start
- Variable NeigungX : Zahl ← 0
Wiederhole unendlich oft
mache
  Schreibe NeigungX gib Winkel ° Kreisel sensor x
  + - wenn
    Absolutwert NeigungX > 20
    mache Schalte LED an Port intern Farbe [rot]
  sonst wenn
    mache Schalte LED an Port intern Farbe [grün]
    
```

Den Absolutwert-Block findest du im experten Modus.

Nächste Aufgabe

Füge noch weitere Überprüfungen verschiedener Neigungswinkel, unterschiedliche LED Farben und Warntöne hinzu.

Probiere es selbst aus, bevor du dir die nächste Lernkarte holst.



Aufgabe

Schief ist nicht gleich schief! Verändere die Empfindlichkeit der Neigungsmessung.

- Verwende eine Serie von **logischen Operatoren** um verschiedene Abstufungen der Neigung des Calliope mini zu messen.

So kann ein möglicher Lösungsweg aussehen:

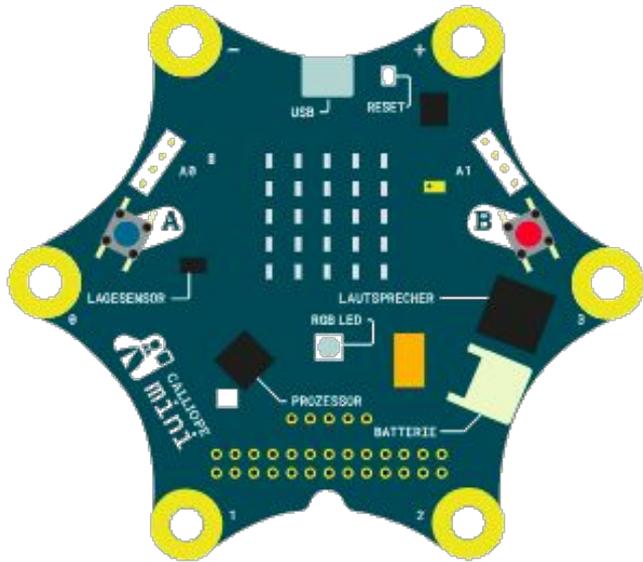
```

+ Start
- Variable NeigungX : Zahl ← 0
- Variable Fehlerpunkte : Zahl ← 0
Wiederhole unendlich oft
  mache
    Schreibe NeigungX ▾ gib Winkel ° Kreisel sensor x ▾
    + - wenn Absolutwert ▾ NeigungX ▾ > ▾ 20
      mache Schalte LED an Port intern ▾ Farbe ▾
    sonst wenn Absolutwert ▾ NeigungX ▾ > ▾ 10
      mache Schalte LED an Port intern ▾ Farbe ▾
    sonst wenn Absolutwert ▾ NeigungX ▾ > ▾ 5
      mache Schalte LED an Port intern ▾ Farbe ▾
    sonst wenn
      mache Schalte LED an Port intern ▾ Farbe ▾
  
```

Nächste Aufgabe

Die unterschiedlichen Neigungsabweichungen kannst du auch durch Warntöne in verschiedenen Tonhöhen ergänzen. Überlege dir einen Zähler, der die Neigungsabweichungen zählt.

Probiere es selbst aus, bevor du dir die nächste Lernkarte holst.



Aufgabe

Erweitere dein Programm nun um einen Zähler, der die verschiedenen Neigungsabweichungen zählt und in Punkte überführt.

- Dafür benötigst du eine zweite **Variable**. Nenne sie z.B. Fehlerpunkte.
- Auf der vorherigen Lernkarte hast du bereits verschiedene Neigungsabweichungen unterschieden. Nun kannst du diesen verschiedenen Abweichungen unterschiedliche Fehlerpunkte zuweisen. Je größer die Abweichung desto mehr Fehlerpunkte.

```

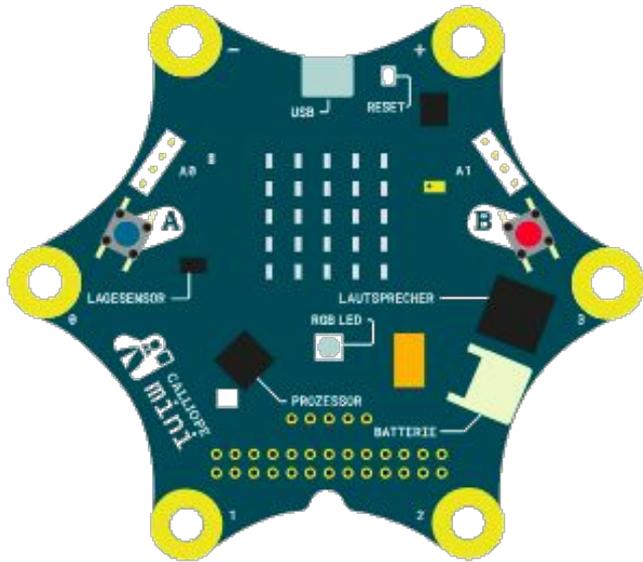
+ Start
- Variable NeigungX : Zahl ← 0
- Variable Fehlerpunkte : Zahl ← 0

Wiederhole unendlich oft
mache
  Schreibe NeigungX gib Winkel ° Kreisel sensor x
  + - wenn Absolutwert NeigungX > 20
  mache
    Schalte LED an Port intern Farbe
    Spiele Achtelnote h''
    Schreibe Fehlerpunkte Fehlerpunkte + 10
  sonst wenn Absolutwert NeigungX > 10
  mache
    Schalte LED an Port intern Farbe
    Spiele Achtelnote e'
    Schreibe Fehlerpunkte Fehlerpunkte + 5
  sonst wenn Absolutwert NeigungX > 5
  mache
    Schalte LED an Port intern Farbe
    Spiele Achtelnote e'
    Schreibe Fehlerpunkte Fehlerpunkte + 2
  
```

Nächste Aufgabe

Wie kannst du die Fehlerpunkte nun auf der LED Matrix ausgeben? Überlege dir außerdem eine Möglichkeit den Fehlerzähler auf Null zu stellen.

Probiere es selbst aus, bevor du dir die nächste Lernkarte holst.



Aufgabe

Natürlich benötigst du auch eine Möglichkeit die gesammelten Minuspunkte anzuzeigen.

- Verwende dafür eine weitere **Wenn mache** Schleife.
- Überlege dir durch welches Ereignis du die Minuspunkte auf der LED-Matrix des Calliope mini anzeigen lässt.

So kann ein möglicher Lösungsweg aussehen:

```

    Schreibe Fehlerpunkte + 2
  sonst wenn
  mache Schalte LED an Port intern Farbe [ ]
  + wenn Taste A gedrückt?
  mache Zeige Text + Erstelle Text aus " Fehlerpunkte "
  + wenn Taste B gedrückt?
  mache Schreibe Fehlerpunkte 0
  
```

Nächste Aufgabe

Bist du fertig? Dann überlege dir, wie du dein Balance Spiel noch erweitern kannst. Falls du keine eigenen Ideen hast, schau auf der Bonuskarte nach.

Schon alles erledigt?



Du hast heute deine eigene kleine Wasserwaage mit dem Calliope mini gebaut. Du hast gelernt, wie du verschiedene Sensoren sinnvoll einsetzen kannst.

Erweitere jetzt dein Balance Spiel.

Am Ende der Stunde kannst du dann das, was du gemacht hast, den anderen in deiner Klasse zeigen.

- Du könntest zum Beispiel:
 - Mehrere Achsen gleichzeitig auf ihre Neigung überprüfen. Was musst du beachten, wenn du sowohl die x-Achse als auch die y-Achse gleichzeitig in deine Programmierung mit einbeziehst?



- Fällt dir eine Möglichkeit ein die Neigung auf der LED Matrix anzuzeigen?
- Hast du noch andere Ideen? Super, dann kannst du das ausprobieren.