

# Calliope-Kurs Kinder

08\_Tag5

---

Jogi Künstler, Turbine Brunnen

Herbst 2020



# **Kurs Programmieren Lernen mit Calliope Mini**

---

# Zusammenfassung

Heute wiederholen wir als erstes nochmal kurz das Gelernte vom letzten Nachmittag,

Wir vertiefen nochmal, wie das mit dem Schleifen-Programmieren funktioniert hat.

Daran anschliessend schauen wir uns nochmal an, wie man einen normalen DC-Motor ansteuert.

Um den Motor durch Kippen des Calliope ansteuern zu können, betrachten wir noch die einfachen Funktionen des Lage-Sensors an.

Dann schauen wir, wie wir Informationen zwischen zwei Calliopes via Kurzstrecken-Funk austauschen können

Und schlussendlich lernen wir noch, wie wir die Eltern oder Geschwister etwas ärgern können :-)

Wir lernen, wie man einen externen Lautsprecher an den Calliope anschliessen kann, so dass alle unsere musikalischen Experimente die ganze Familie

geschallen. ...



- 01 Auffrischen: Schleifen
- 02 Auffrischen Motor
- 03 Der Lagesensor
- 04 Funkübertragung
- 05 Externer Lautsprecher





# 08\_01\_Auffrischen

Calliope-Kurs Kinder

---

Jogi Künstner, Turbine Brunnen

Herbst 2020



## **Auffrischen / Hausaufgabe**

---

## Auffrischen Schleifen

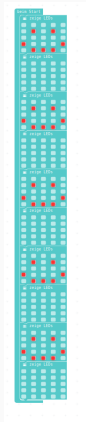
---

- Frage : Wofür braucht man Schleifen?
- Antwort 1 : Immer dann, wenn man etwas gleiches wiederholen will!
- Antwort 2 : Immer dann, wenn man etwas sehr ähnliches wiederholen will, wobei sich dabei bestimmte Dinge ändern können, die vom Schleifendurchlauf abhängen.
  - Also beim **ersten** Schleifendurchlauf wird etwas mit einer **1** gemacht
  - Beim **zweiten** Durchlauf wird etwas mit einer **2** gemacht
  - usw. usw.



# Beispiel 1 : Ohne Schleife

Beim Starten 5 mal ein Gesicht blinken lassen



**Figure 1:** Ohne Schleife

# Beispiel 1 : Mit Schleife

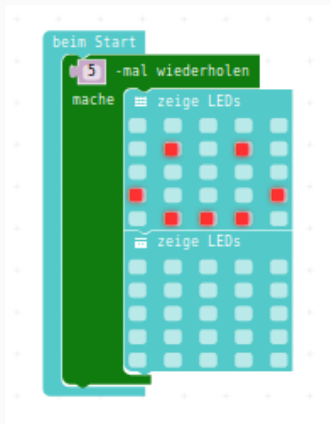
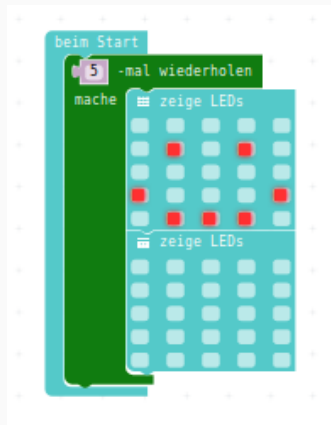
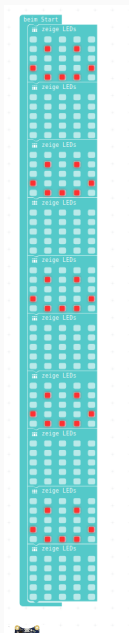


Figure 2: Mit Schleife

# Beispiel 1 : Vergleich



## Beispiel 2 : Schleife mit Zähler

Nun wollen wir innerhalb des sogenannten “Schleifenkörpers” die Anzahl der Schleifen-Durchgänge anzeigen.

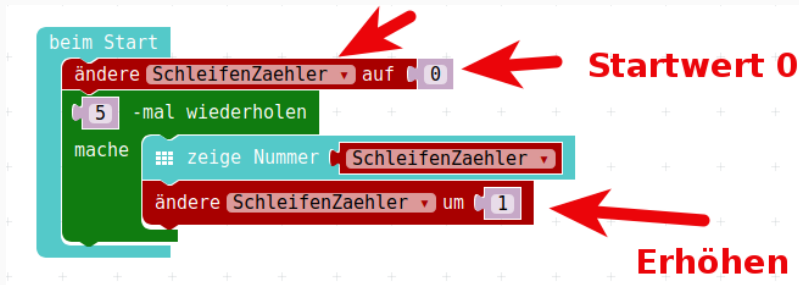
- Dazu benutzen wir die gerade vorhandene Schleife,
- legen **VOR** der Schleife eine Variable namens **SchleifenZaehler** an,
- diese belegen wir mit 0.





## Beispiel 3 : Schleife mit Zähler

Im Schleifenkörper lassen wir uns den Wert dieser Variable anzeigen (mit “Zeige Nummer”) und erhöhen anschliessend die Variable/den Zähler.



Da wir den Zähler mit 0 vorbelegen und die Schleife 5 mal läuft, bekommen wir durch dieses Programm die Zahlen 0 bis 4 angezeigt.



## Beispiel 3 : Schleife mit eingebautem Zähler

Diese Art der Schleife wird sehr oft gebraucht:  
eine Schleife, die eine bestimmte Anzahl von Durchläufen erlaubt und bei der man die Schleifendurchläufe mitzählt.

Darum gibt es dafür ein extra Programmier-Konstrukt.

Das ist die Index-For-Schleife, die wir am letzten Nachmittag schon kennengelernt haben.



# Beispiel 4 : Schleife mit eingebautem Zähler

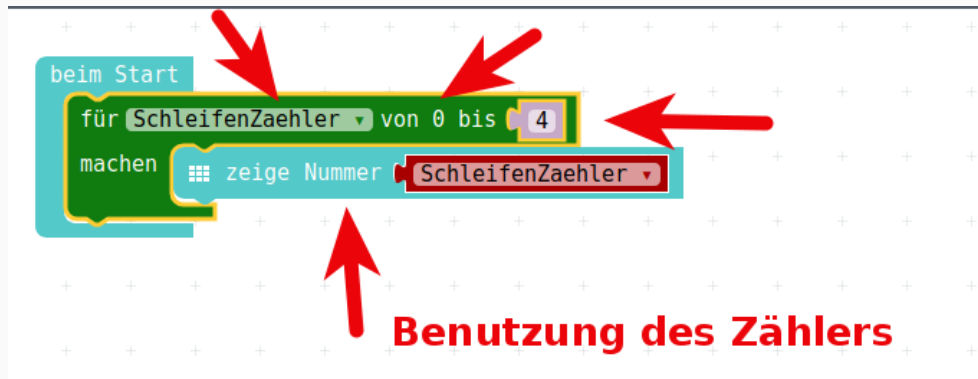
Diese finden wir ebenso im Menu Schleifen:



Figure 3: Schleifen-Menu

## Beispiel 4 : Schleife mit eingebautem Zähler

Wenn wir diese Schleife benutzen und unser Programm entsprechend umgestalten, sieht es nochmal um einiges einfacher aus:

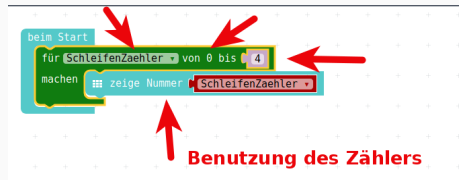
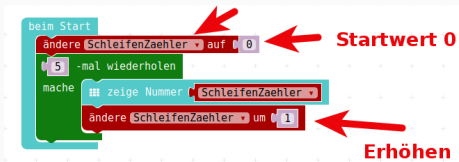


The image shows a Scratch code block structure on a grid background. It consists of a 'beim Start' block (light blue) containing a 'für SchleifenZähler von 0 bis 4' block (green) and a 'machen' block (light blue) containing a 'zeige Nummer SchleifenZähler' block (light blue). Red arrows point to the '4' in the loop block, the 'zeige Nummer' block, and the 'SchleifenZähler' dropdown menu. Below the code block, the text 'Benutzung des Zählers' is written in red.

**Benutzung des Zählers**



# Vergleich der beiden Schleifen



# Hausaufgabe 1 RGB-Led

---

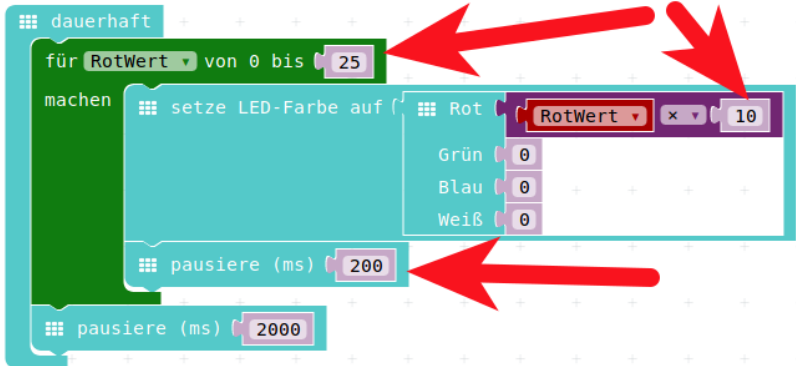
# Schleife mit Rot-Anteil

The image shows a Scratch script on a grid background. It starts with a yellow 'dauerhaft' (forever) loop block. Inside the loop, there is a green 'für RotWert von 0 bis 255' (for loop) block. Two red arrows point to the 'RotWert' variable and the '255' value. Below the 'für' block is a cyan 'setze LED-Farbe auf' (set LED color) block with a dropdown menu showing 'Rot' selected and 'RotWert' in the input field. Below that are two 'pausiere (ms)' (wait) blocks: one with '20' and another with '2000'.

- Eine Schleife um den Rot-Anteil zu verändern
- Wieviele Schleifendurchläufe sind das?
- Wie lange läuft dieses Programm einmal (inklusive pausieren!) ?



## Zweite Rot-Anteil-Schleife



- Nochmal eine Schleife um den Rot-Anteil zu verändern
- Wieviele Schleifendurchläufe sind das?
- Wie lange läuft dieses Programm einmal (inklusive pausieren!) ?
- Seht Ihr einen Unterschied?



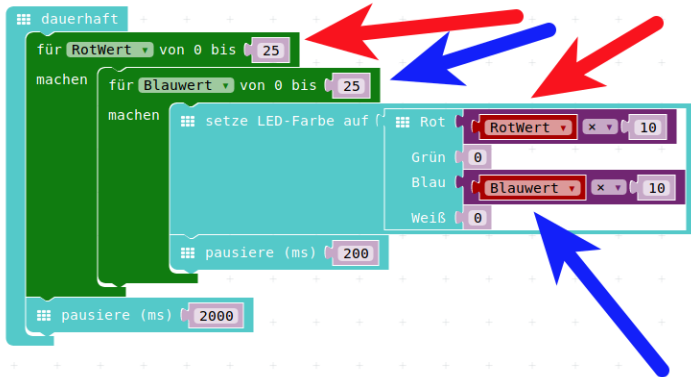


# Warum diese Aenderung

- Wir sehen kaum einen Unterschied
- Aber wir haben nur 25 anstatt 255 Schleifendurchläufe
- Nun wollen wir Schleifen “verschachteln”
- Wenn wir da auch nur 25 anstatt 255 Durchläufe machen, sehen wir etwas
- und das in einigermaßen sinnvoller Zeit. . .
- Siehe nächste Seite



# Verschachtelte Schleife



- Diesmal zwei verschachtelte Schleifen
- Die äussere ändert Rot, die innere Blau
- Wieviele Schleifendurchläufe sind das aussen?
- Wieviele Schleifendurchläufe sind das innen?
- Wie lange läuft dieses Programm einmal (inklusive pausieren!) ?



## **Hausaufgabe 2 : Display fuellen**

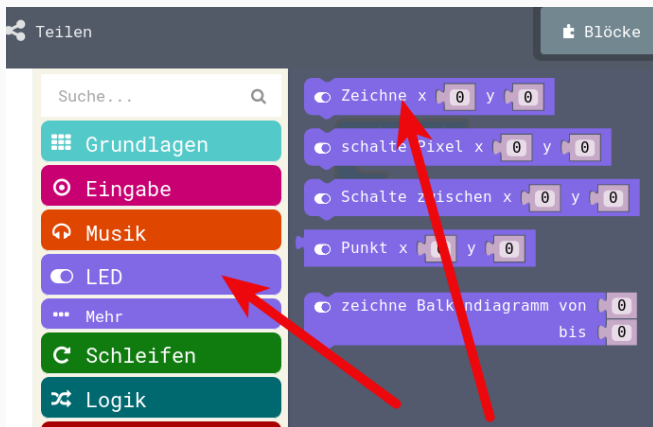
---

# Zeichen X Wert Y Wert

Dazu schauen wir uns aus dem Menu **LED** einmal einen Befehl etwas genauer an:

## Zeichne X Wert Y Wert

Damit kann man einzelne LEDs auf unserem 5 x 5 LED-Bildschirm einschalten.



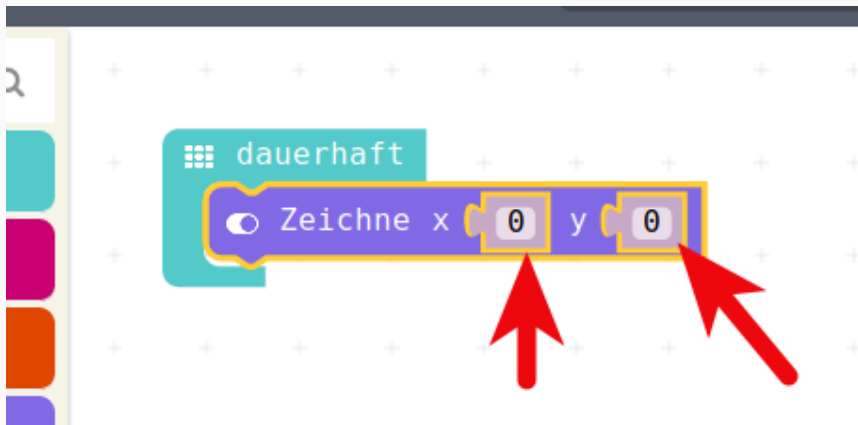
The screenshot shows the Calliope mini programming environment. On the left, a menu titled 'Teilen' (Share) contains several categories: Grundlagen, Eingabe, Musik, LED, Mehr, Schleifen, and Logik. The 'LED' category is highlighted with a red arrow. On the right, a workspace titled 'Blöcke' (Blocks) displays several LED-related blocks. The top block is 'Zeichne x y', which is highlighted with a red arrow. Below it are 'schalte Pixel x y', 'Schalte zwischen x y', 'Punkt x y', and 'zeichne Balkendiagramm von bis'.



- Das probieren wir gleich mal im Simulator aus,
- Wir klicken diesen einen Befehl in die **Dauerhaft**-Schleife ein
- Wir spielen mit den X und Y-Werten
- Wir schauen, was im Simulator passiert.



# Jede LED einschaltbar

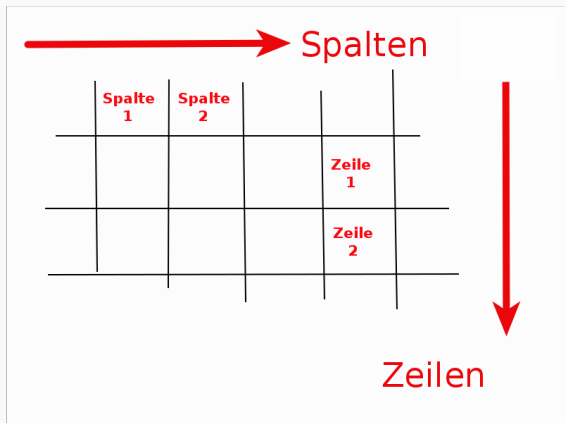


- Wir können durch Verändern der Werte für X und Y zwischen 0 und 4 jede beliebige LED auf unserem 5 x 5 - Display einschalten.



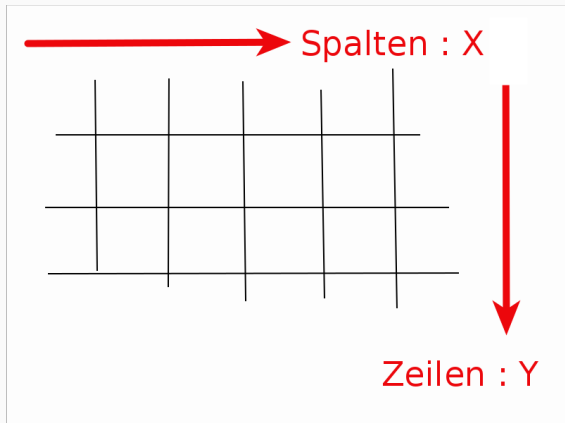
# Zeilen und Spalten

- Das kommt aus der Mathematik, man bezeichnet im Allgemeinen die Spalten-Richtung mit X
- und man bezeichnet die Zeilen mit Y



# Zeilen und Spalten

- Spalten, horizontale Richtung, Links-Rechts : **X**
- Zeilen, vertikale Richtung, Oben-Unten : **Y**



- Auch hier wieder - wie oft beim Programmieren : es geht bei 0 los !



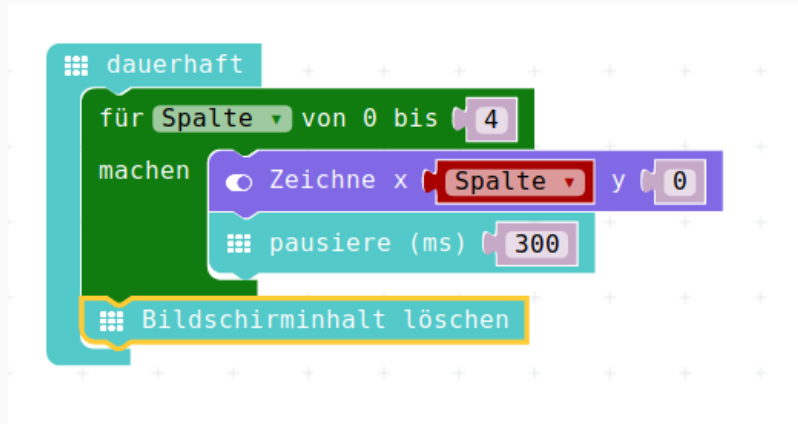


- Nun beschränken wir uns also auf eine Zeile in unserer LED-Anzeige
- Wir zeichnen einen Punkt nach dem anderen
- Dazu erhöhen wir jeweils den Spalten-Wert
- Also **X** !



# Schleife mit Variablen

- Mit unserem Schleifen-Wissen können wir das mit Schleifen programmieren
- Das Ergebnis schauen wir zuerst im Simulator an
- Wenn alles passt, können wir das auch in den Calliope programmieren.

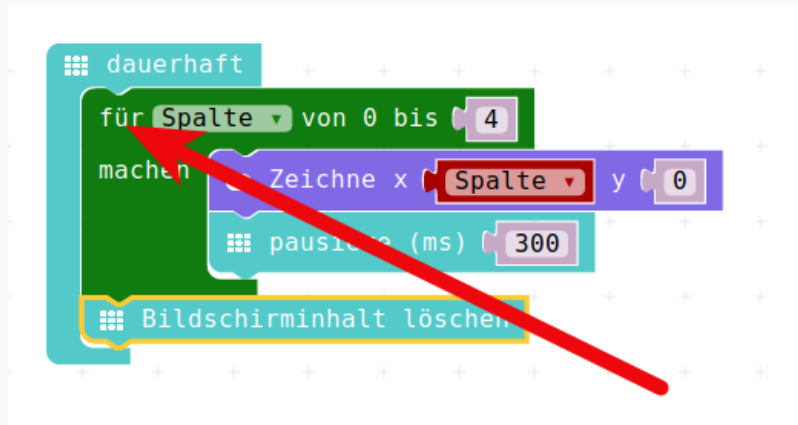


- Diese Schleife läuft nun **5 mal!**
  - Zuerst mit Variable **Spalte = 0**, dann wird die LED in Spalte (X) **0** Zeile (Y) 0 gesetzt
  - Dann mit Variable **Spalte = 1**, dann wird die LED in Spalte (X) **1** Zeile (Y) 0 gesetzt
  - Anschliessend mit Variable **Spalte = 2**, dann wird die LED in Spalte (X) **2** Zeile (Y) 0 gesetzt
  - Daraufhin mit Variable **Spalte = 3**, dann wird die LED in Spalte (X) **3** Zeile (Y) 0 gesetzt
  - Schliesslich ein letztes Mal mit Variable **Spalte = 4**, dann wird die LED in Spalte (X) **4** Zeile (Y) 0 gesetzt



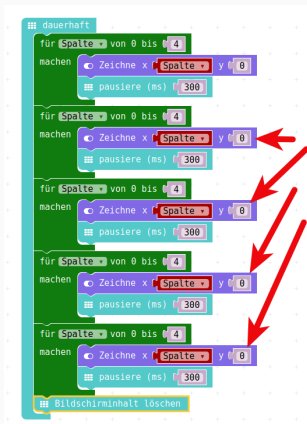
# Nächster Versuch, Ganzer Bildschirm

- Nun können wir also mit einer Schleife eine Zeile füllen, durch die Index-Schleife wird die Zeile **spaltenweise** gefüllt
- Wir wollen aber immnoch den ganzen Bildschirm einzeln mit LEDs füllen.
- Dazu kopieren wir nun die Index-Schleife 4 mal



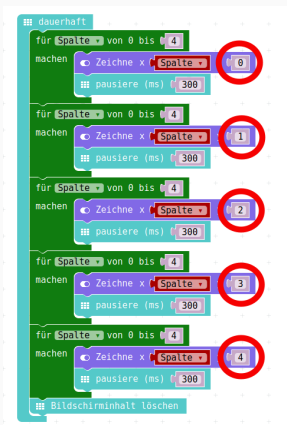
## 4 mal kopiert

- Wir hängen sie viermal untereinander
- Das Bildschirm-Löschen schieben wir dabei nach unten, das wollen wir nur einmal ganz am Schluss haben
- Nun haben wir 5 identischen Zeilenfüller, verwirklicht durch eine Schleife
- Wir müssen nun noch die Zeilen-Nummern anpassen.



# Fertiges Programm, Ganzer Bildschirm

- Wenn wir das gemacht haben, haben wir ein Programm, das den Bildschirm LED für LED füllt
- Und das ganze durch die schlaue Verwendung von Schleifen.



```
Scratch code for filling the screen with LEDs:  
- A 'forever' loop block.  
- Inside the loop, a 'for Spalte' block from 0 to 4.  
- Inside the 'for' loop, a 'make' block with 'Zeichne x Spalte' set to the loop variable.  
- Below the 'make' block, a 'wait (ms) 300' block.  
- The 'for' loop is repeated five times, with the 'Zeichne x Spalte' value set to 0, 1, 2, 3, and 4 respectively.  
- At the end of the loop, a 'Bildschirminhalt löschen' block.  
- Red circles highlight the 'Zeichne x Spalte' values in each iteration.
```



# Fuenf fast identische Codestuecke

Was wir aber immer noch haben, sind 5 fast gleich aussehende Schleifen. Und diese 5-fache Wiederholung wollen wir nun durch eine zweite, verschachtelte Schleife ersetzen.

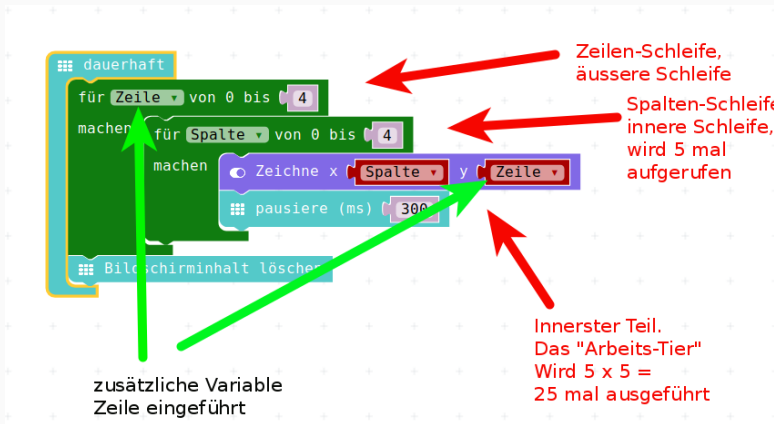
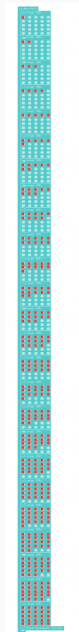


Figure 4: Zwei verschachtelte Schleifen

# “Boeser” Auftrag-Geber

Nehmt an, Ihr hättet die Schleife noch nicht kennengelernt, und hättet die Aufgabe mit einzelnen zeige LED-Befehlen programmiert:





Nun kommt der Auftrag-Geber und möchte nun doch lieber anstatt zeilenweise den Bildschirm zu füllen, diesen spaltenweise gefüllt haben.

In dem gezeigten Beispiel müsst Ihr alles neu programmieren.

23 mal andere Bildschirm-Inhalte von Hand malen.

(Der erste und der letzte Bildschirm-Inhalt passen)

Wollt Ihr es versuchen?

Ich würde lieber die Variante mit den Schleifen nehmen und dort nur die zwei Schleifen bzw die beiden Index-Variable vertauschen!



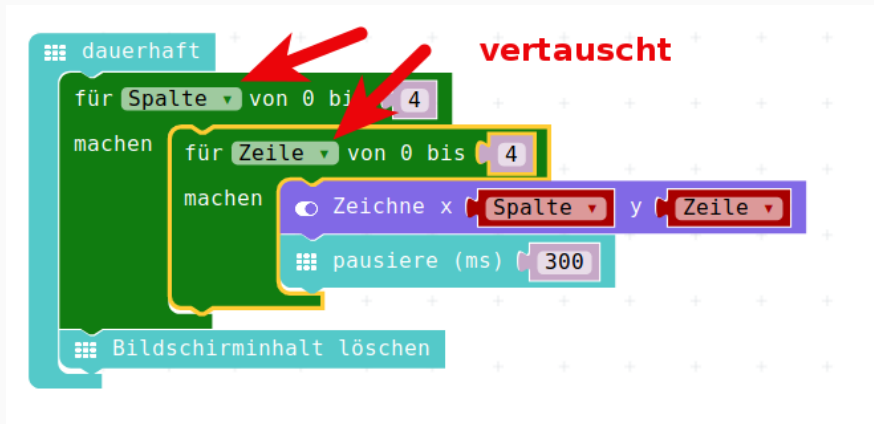


Figure 5: Zwei verschachtelte Schleifen

# 08\_02\_Motoren\_Auffrischen

Calliope-Kurs Kinder

---

Jogi Künstner, Turbine Brunnen

Herbst 2020



# DC-Motoren Auffrischen

---

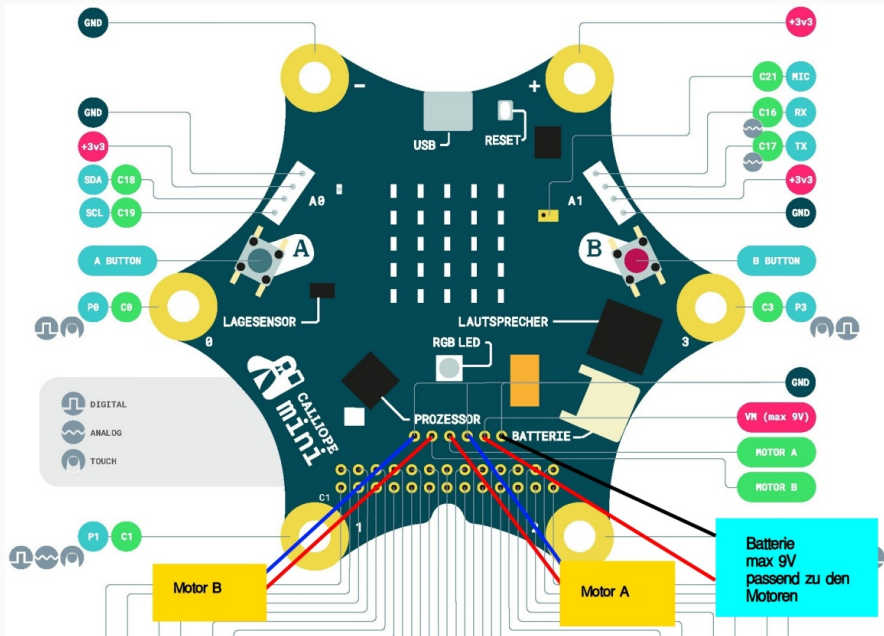
Je nach Verwendungszweck kann man an den Calliope entweder

- 1 Motor anschliessen, der kann dann vorwärts und rückwärts drehen
- 2 Motoren anschliessen, die können dann nur einzeln vorwärts drehen

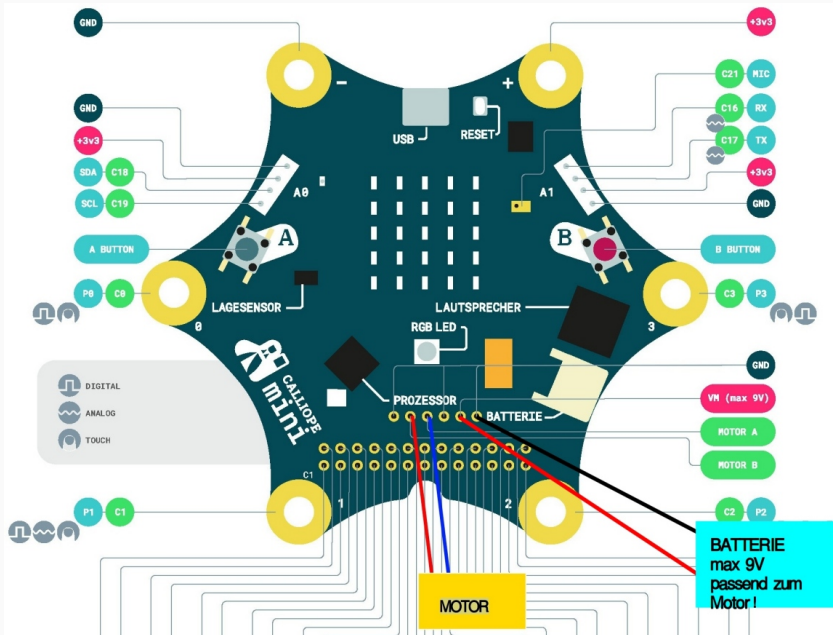
So sehen die beiden Möglichkeiten zum Anschluss von zwei oder einem Motor aus:



# Zwei Motoren



# Ein Motor



Nachdem wir nun wissen, wie wir einen einzelnen Motor an den Calliope elektrisch anschliessen, wollen wir den Ausgang für den Motor auch mit Software programmieren.

Im ersten Schritt wollen wir nur ganz einfach den Motor ein- und ausschalten können.

Dazu wollen wir mit dem linken Knopf ein und mit dem rechten Knopf ausschalten.





Die Motor-Ansteuerung findet sich im Menu Motoren:

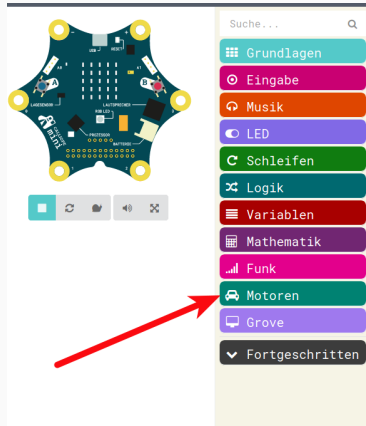


Figure 1: Menu Motor

Es gibt nicht viele Befehle zum Steuern von Motoren:

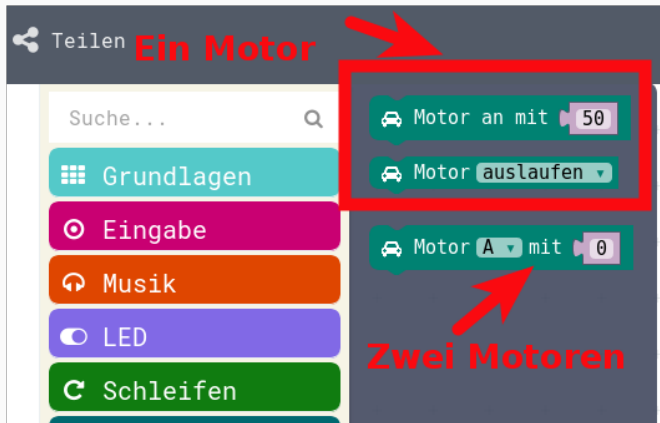


Figure 2: Motor Befehle

# Erstes Motor-Programm

Das war unser erstes Motor-Programm

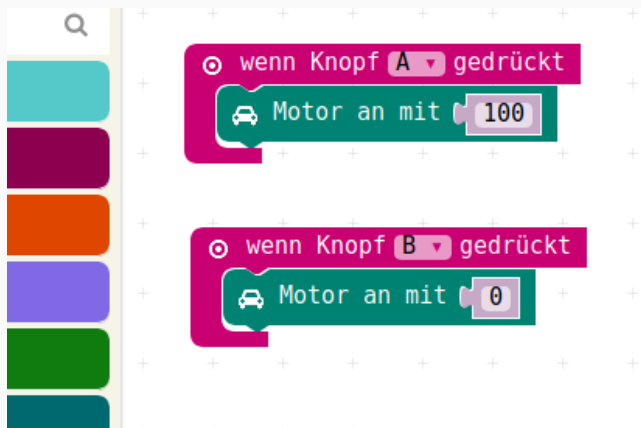


Figure 3: Motor Programm 1

Dieses Programm können wir leider im Simulator gar nicht nutzen.

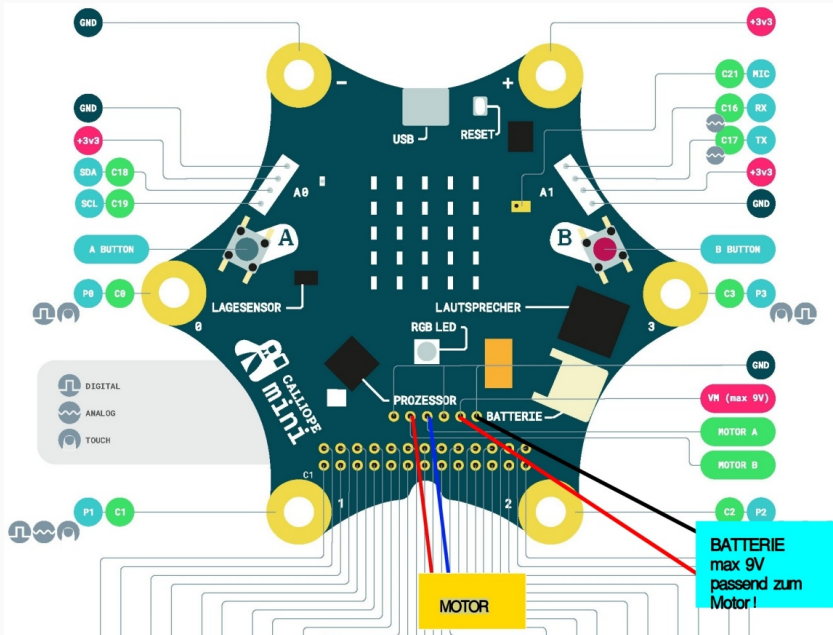
Also müssen wir das Programm auf den Calliope runterladen und dort ausprobieren.



- Der Calliope erlaubt **für den Motor** den Anschluss einer zusätzlichen Batterie.
- Deren Spannung wird **NUR** zum Ansteuern des Motors verwendet
- Die Batterie darf nicht mehr als 9V haben
- Sie muss zum Motor passen



# Zusätzliche Batterie



# 08\_03\_LageSensor

Calliope-Kurs Kinder

---

Jogi Künstner, Turbine Brunnen

Herbst 2020



# Motorsteuerung mit Lage-Sensor

---

Das funktioniert ja schon mal ganz gut.

Nun möchten wir mit diesem einfachen Motor-Steuerungs-Programm auch noch eine andere Eingangs-Möglichkeit ausprobieren:

## Den Lage-Sensor!

Der Calliope hat einen Lage-Sensor eingebaut, der in allen Raumrichtungen funktioniert.

Also :

- Oben / Unten
- Links / Rechts
- Vorne / Hinten





# Abfragen des Lagesensors

Die Abfragen, um den Lage-Sensor genau auszuwerten, sind recht kompliziert. Man muss Koordinaten-Systeme verstehen und man sollte Winkelrechnung verstehen.

Beides ist in Euerem Alter wahrscheinlich noch nicht der Fall.

Zusätzlich zu den genauen Abfrage-Möglichkeiten, die schwierig zu verwenden sind, hat der Calliope aber auch die Möglichkeit, sehr einfach den Lage-Sensor abzufragen.

Das wollen wir nun tun:

- Beim Gerade halten des Calliope soll der Motor aus sein.
- Beim Kippen nach links soll er sich nach vorne drehen
- Beim Kippen nach rechts soll er sich nach hinten drehen.



# Eingabe-Menü

Sowohl die genauen, schwierigeren Befehle als auch die Einfachen befinden sich im Menü Eingabe:

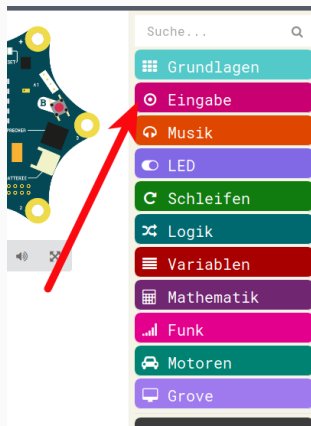


Figure 1: Menü Eingabe

# Inhalte Eingabe-Menü

Calliope mini Projekte Teilen

Suche...

- Grundlagen
- Eingabe**
- Mehr
- Musik
- LED
- Schleifen
- Logik
- Variablen
- Mathematik
- Funk
- Motoren
- Grove
- Fortgeschritten

**Einfach**

**Komplex**

- wenn Knopf A gedrückt
- wenn geschüttelt
- wenn Knopf B gedrückt
- wenn Pin P0 gedrückt
- wenn Knopf A+B gedrückt
- Knopf A ist gedrückt
- Pin P0 ist gedrückt
- Beschleunigung (mg)
- Lichtstärke
- Kompassausrichtung (°)
- Temperatur (°C)

Figure 2: Menu Eingabe Inhalt



# Wenn geschüttelt

Nun ziehen wir drei mal das **wenn geschüttelt** in unseren Arbeits-Bereich:

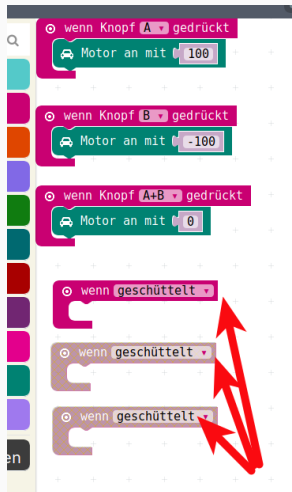


Figure 3: Dreimal Geschuettelt

Diese wandeln wir nun alle durch Druck auf das Dreieck:



**Figure 4:** Dreieck

um in drei verschiedene Reaktionen:

- “nach links neigen”
- “Display nach oben”
- “nach rechts neigen”



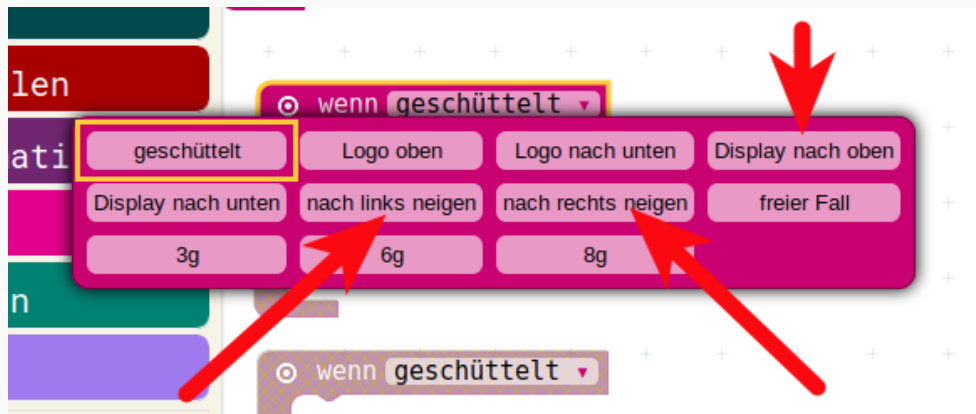


Figure 5: Lage-Sensor

Damit sieht unser Programm nun so aus:

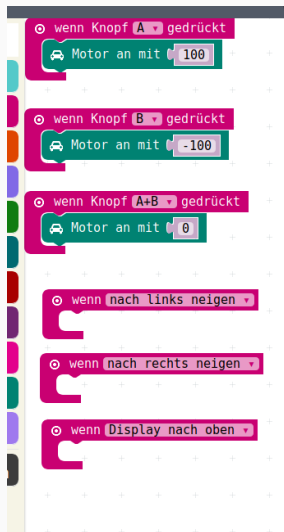
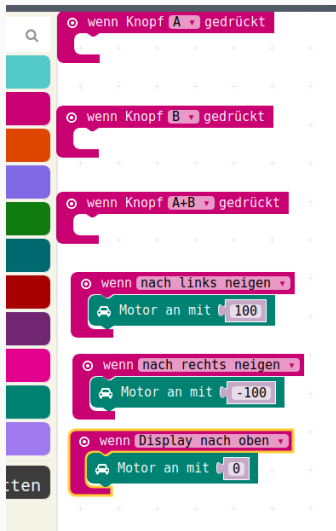


Figure 6: Lage-Sensor drin



# Fertiges Programm

und wenn wir nun die entsprechenden Befehle von oben nach unten schieben, dann können wir unseren Motor durch kippen steuern.





Dieses Programm können wir nun auch in den Calliope laden.

Achtung: Zumindest bei manchen Kombinationen von Calliope und Computer (und vermutlich angeschlossenem Computer-Ladegerät) hat der Lage-Sensor **NICHT** richtig funktioniert.

==>

Bitte steckt in diesem Fall das USB - Kabel aus und betreibt Euren Calliope nur über Batterie.



## Java-Script-Code

```
input.onGesture(Gesture.TiltLeft, () => {  
  motors.motorPower(100)  
})  
input.onGesture(Gesture.TiltRight, () => {  
  motors.motorPower(-100)  
})  
input.onGesture(Gesture.ScreenUp, () => {  
  motors.motorPower(0)  
})
```



Hex-code



# 08\_04\_Funkuebertragung

Calliope-Kurs Kinder

---

Jogi Künstner, Turbine Brunnen

Herbst 2020



# Funk - Uebertragung

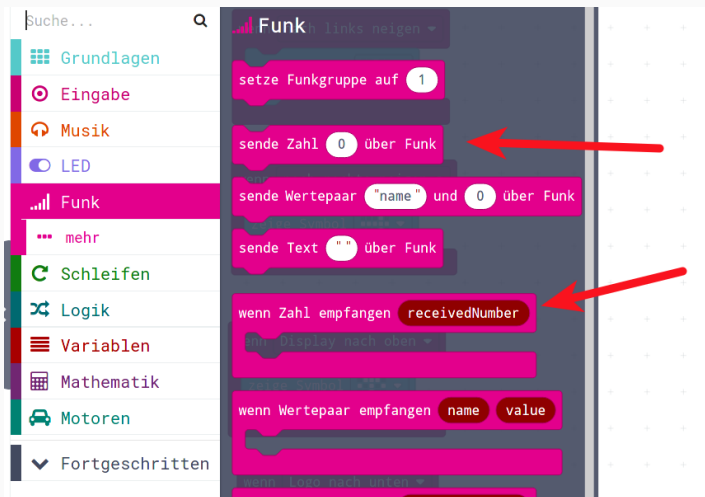
---

Im Menu Funk findet sich:

The image shows a search interface for the 'Funk' (Radio) category. On the left, a search bar contains 'Suche...' and a magnifying glass icon. Below it is a list of categories: Grundlagen, Eingabe, Musik, LED, Funk (highlighted in pink), mehr (with a red arrow pointing to it), Schleifen, Logik, Variablen, Mathematik, Motoren, and Fortgeschritten. On the right, a list of code blocks is displayed under the heading 'Funk: links neigen'. The blocks include: 'setze Funkgruppe auf 1', 'sende Zahl 0 über Funk', 'sende Wertepaar "name" und 0 über Funk', 'sende Text "" über Funk', 'wenn Zahl empfangen receivedNumber', 'Display nach oben', 'wenn Wertepaar empfangen name value', 'wenn Text empfangen receivedString', and 'wenn Text empfangen receivedString'.



Wir senden und empfangen zuerst mal Zahlen

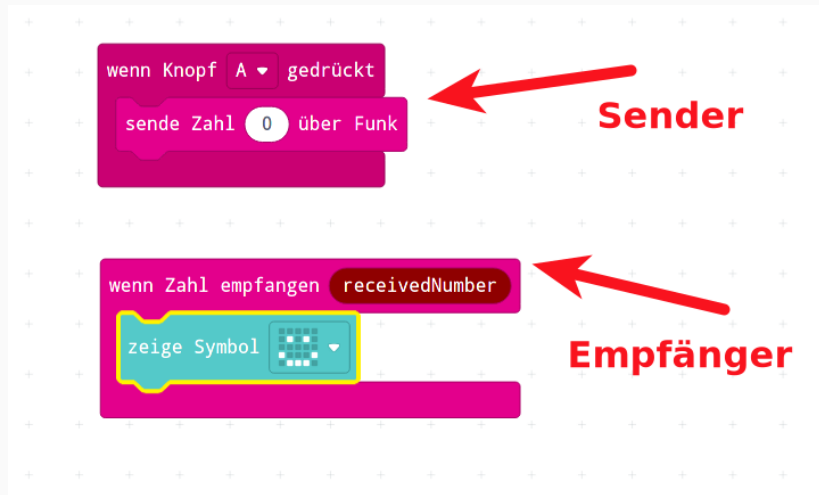


The image shows the Scratch 'Funk' (Radio) block palette on the left and a script area on the right. The palette includes categories like Grundlagen, Eingabe, Musik, LED, Funk, Schleifen, Logik, Variablen, Mathematik, Motoren, and Fortgeschritten. The script area contains several radio-related blocks:

- setze Funkgruppe auf 1
- sende Zahl 0 über Funk (highlighted with a red arrow)
- sende Wertepaar "name" und 0 über Funk
- sende Text "" über Funk
- wenn Zahl empfangen receivedNumber (highlighted with a red arrow)
- wenn Wertepaar empfangen name value



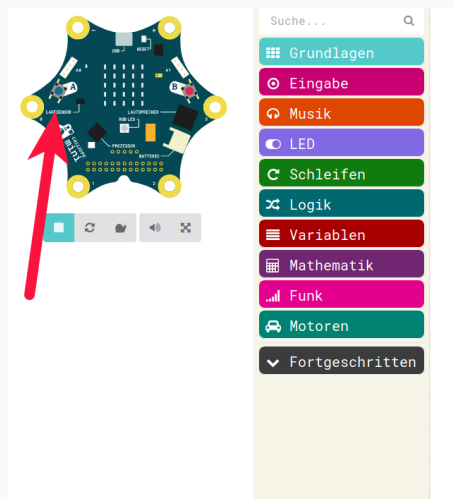
# Simpler Sender/Empfänger





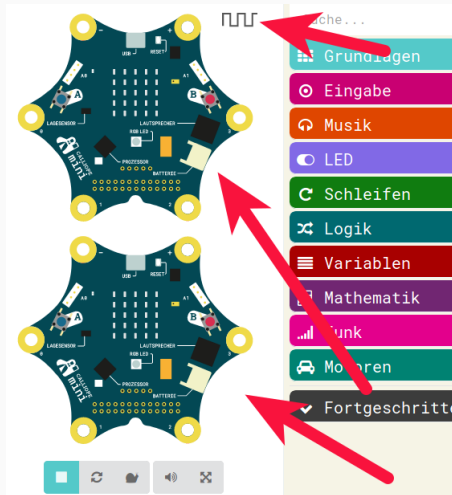
# Simulator kann auch Funk

- Sobald man den Knopf A drückt
- merkt das der Simulator und blendet einen zweiten Calliope ein



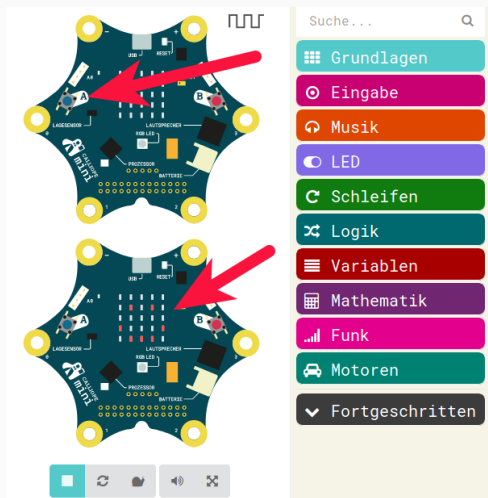
# Zwei simulierte Calliopes

- Nun sind zwei simulierte Calliopes zu sehen
- Und der obere ist am Funken



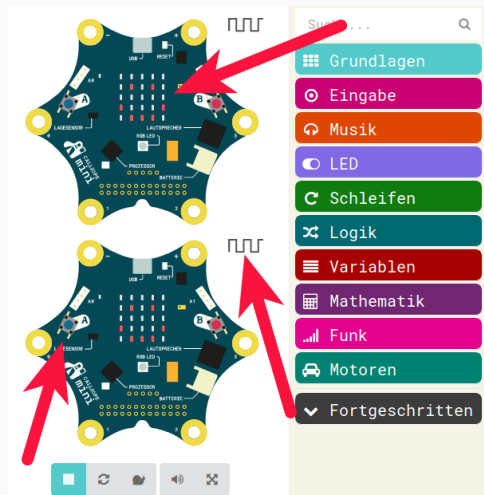
# Nochmal Knopf drücken

- Nun muss man nochmal Knopf A drücken
- damit der andere Calliope das empfangen kann



# Anderer Calliope sendet auch

- Drücken auf zweitem Calliope
- bringt den auch zum Senden
- und der erste/obere empfängt



# “Sinnvolle” Inhalte Senden

The image displays Scratch code blocks for a radio communication simulation. On the left, three pink blocks handle button presses: 'wenn Knopf A gedrückt' (send 0), 'wenn Knopf B gedrückt' (send 1), and 'wenn Knopf A+B gedrückt' (send 2). On the right, a large pink block 'wenn Zahl empfangen receivedNumber' contains a conditional logic structure: 'wenn receivedNumber = 0 dann' (show symbol), 'sonst wenn receivedNumber = 1 dann' (show symbol), and 'ansonsten' (show symbol). Blue arrows point from the '0', '1', and '2' values in the sending blocks to the corresponding values in the receiving block's conditions.



- Einigt euch auf 3 Zahlen, die Ihr senden wollt ( z.B. 0-2 )
- (am Besten verwendet Ihr das Programm aus diesem Beispiel)
- Jeder kann seinen eigenen Empfangs-Teil mit verschiedenen Symbolen machen
- Dann versucht, Euch gegenseitig was zu senden
- Klappt das ?
- ...
- ...
- ...
- ????
- Im Simulator gehts doch, oder ?



- Nun probiert mal das HEX-File von einem der beiden Computer in beide Calliopes zu programmieren.
- Jetzt gehts?
- Im Simulator gehts auch ?
- Habt Ihr nen Fehler gemacht ?
- Wer von Euch ?
- 
- Antwort: Vermutlich hat keiner von Euch einen Fehler gemacht. . .



- Wer von Euch hat Funkgeräte?
- Können die immer miteinander kommunizieren
- Können die direkt mit den Funkgeräten von Euren Freunden funken?
- Oder muss man da was einstellen, damit das funktioniert?
- **JA** : Der Sende und Empfangs-Kanal !
- Beim Calliope heisst das **“Funkgruppe”**
- und befindet sich ganz oben im **Funk** - Menu





# Setze Gruppe

The image shows a Scratch script for a radio station named "Funk". The script is written in German and consists of several blocks:

- setze Funkgruppe auf 1**: This block sets the radio group to 1. A blue arrow points to the number 1 in the block.
- sende Zahl 0 über Funk**: This block sends the number 0 over the radio.
- sende Wertepaar "name" und 0 über Funk**: This block sends a value pair with the key "name" and the value 0 over the radio.
- sende Text "" über Funk**: This block sends an empty string over the radio.
- wenn Zahl empfangen receivedNumber**: This is the start of an "if number received" block, with "receivedNumber" in the input field.

On the left side of the Scratch interface, a search bar contains the text "dlagen", "abe", "k", "eifen", "k", and "ablen".



- Wenn die Funk-Gruppe nicht von Euch im Programm-Lauf gesetzt wird
- dann setzt der Computer das im Hintergrund
- automatisch auf einen zufälligen Wert zwischen 0 - 255
- d.h. ein HEX-File enthält einen “ausgewürfelten” Kanal
- ein anderes HEX-File enthält einen anderen “ausgewürfelten” Kanal
- darum: Entweder gleiches HEX-File verwenden oder Kanal setzen
- => Setze Funkgruppe XXX beim Start



# Setze Gruppe

Nun bauen wir also das entsprechende **setze Funkgruppe** - Befehl **beim Start**

The image shows a Scratch script on a light gray grid background. On the left, a teal 'beim Start' block is connected to a pink 'setze Funkgruppe auf' block. The number '0' in the 'setze Funkgruppe auf' block is circled in blue, with a blue arrow pointing to it from the left. Below this are three pink 'wenn Knopf' blocks: 'wenn Knopf A gedrückt' with 'sende Zahl 0 über Funk', 'wenn Knopf B gedrückt' with 'sende Zahl 1 über Funk', and 'wenn Knopf A+B gedrückt'. On the right, a pink 'wenn Zahl empfangen' block is connected to a teal 'receivedNumber' block. Below it is a 'wenn receivedNumber =' block with a dropdown menu. This is followed by a 'sonst wenn receivedNumber' block with a dropdown menu, and an 'ansonsten' block with a dropdown menu. Each dropdown menu contains a 3x3 grid of dots. A plus sign icon is visible at the bottom left of the right-hand script.



# Zeichenfolgen austauschen

Ebenso kann man - anstatt einzelne Zahlen - ganze Texte von einem Calliope zum anderen schicken

Die beide dafür verantwortlichen Befehle sind

- **sende Text xxx über Funk**
- **wenn Text empfangen receivedString**

The screenshot shows the Calliope programming environment with a sidebar on the left containing category icons and labels: Musik, LED, Funk, mehr, Schleifen, Logik, Variablen, Mathematik, Motoren, and Fortgeschritten. The main workspace contains several code blocks:

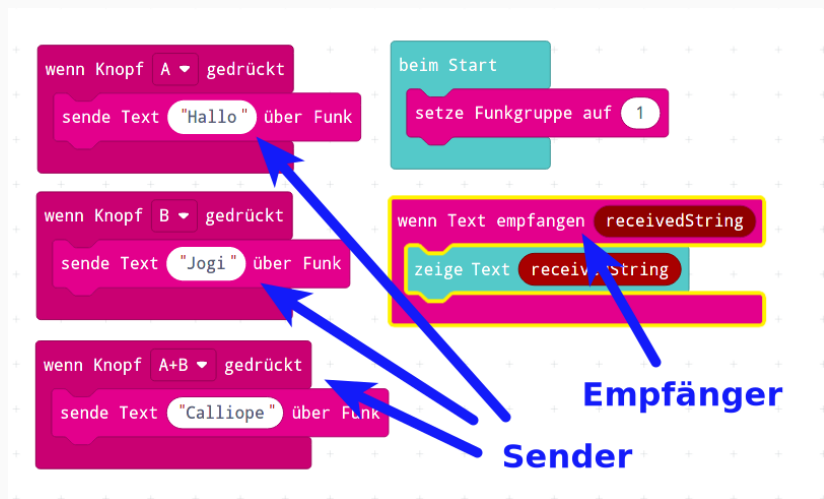
- sende Zahl 0 über Funk** (pink block)
- sende Wertepaar "name" und 0 über Funk** (pink block)
- sende Text "" über Funk** (pink block) with a blue arrow pointing to it from the right.
- wenn Zahl empfangen receivedNumber** (pink block)
- wenn Wertepaar empfangen name value** (pink block)
- wenn Text empfangen receivedString** (pink block) with a blue arrow pointing to it from the right.

The background of the workspace is a grid of small '+' symbols.



# Texte senden und empfangen

So kann man sich Text-Nachrichten hin und her senden



# Variablen und Werte schicken

Und dann gibt es noch die Möglichkeit, ganze Zeichenketten **UND** gleichzeitig Zahlenwerte zu verschicken und zu empfangen:

The image shows a Scratch script with the following blocks:

- sende Zahl 0 über Funk
- sende Wertepaar "name" und 0 über Funk
- sende Text "" über Funk
- wenn Zahl empfangen receivedNumber
- wenn Wertepaar empfangen name value

Two blue arrows point to the 'sende Wertepaar' and 'wenn Wertepaar empfangen' blocks, with the text 'Wertepaar versenden' and 'empfangen' respectively.



# Variablen und Werte schicken II

In der Praxis kann man das nutzen, um zum Beispiel verschiedene Variablen anhand der Variablen-Namen zu verschicken und gleichzeitig deren Werte zu verschicken.

Beispiel:

- Ferngesteuertes Auto
- Geschwindigkeit von 0 bis 100 (kein Rückwärts-Gang. . .)
- Richtung von -50 (nach links fahren) bis + 50 (rechts fahren) => 0 = gerade aus
- Das heisst, das ferngesteuerte Auto muss in der Lage sein, 2 Variablen zu empfangen
- Einfaches Beispiel :
  - Knopf A startet das Auto mit Geschwindigkeit 80 und Richtung 0
  - Knopf B stoppt das Auto mit Geschwindigkeit 0
- Bitte jedes Kind einen anderen Kanal verwenden, ab Kanal 2 im Uhrzeigersinn jeweils 1 mehr



# Variablen und Werte schicken III

Die Variablen, die das Auto versteht, sind **speed** und **richtung**

```
beim Start
setze Funkgruppe auf 1

wenn Knopf A gedrückt
  sende Wertepaar "speed" und 80 über Funk
  sende Wertepaar "richtung" und 0 über Funk

wenn Knopf B gedrückt
  sende Wertepaar "speed" und 0 über Funk
```





Nun wollen wir eine “richtige” Fernsteuerung für das Auto machen:

- Wenn **KEIN** Knopf gedrückt ist, dann **speed = 0**
- Wenn **Knopf A** gedrückt ist, dann **richtung = -40** und **speed = 80**
- Wenn **Knopf B** gedrückt ist, dann **richtung = 40** und **speed = 80**
- Wenn **Knopf A+B** gedrückt ist, dann **richtung = 0** und **speed = 90**

Achtung : Es gibt kein Ereignis : **Wenn Knopf losgelassen** Ihr müsst Euch anders behelfen.

- Dauerhaft 0 senden und **Wenn Knopf gedrückt** dann etwas anderes senden, oder
- Dauerhaft die Tasten abfragen in der Dauerhaft-Schleife und dort die Entscheidungen treffen



# Fernsteuerung für das Auto II

dauerhaft

wenn Knopf A ist gedrückt dann

ansonsten

wenn Knopf A gedrückt

Abfrage, mit wenn-dann

ansonsten

Möglichkeit

Ereignis

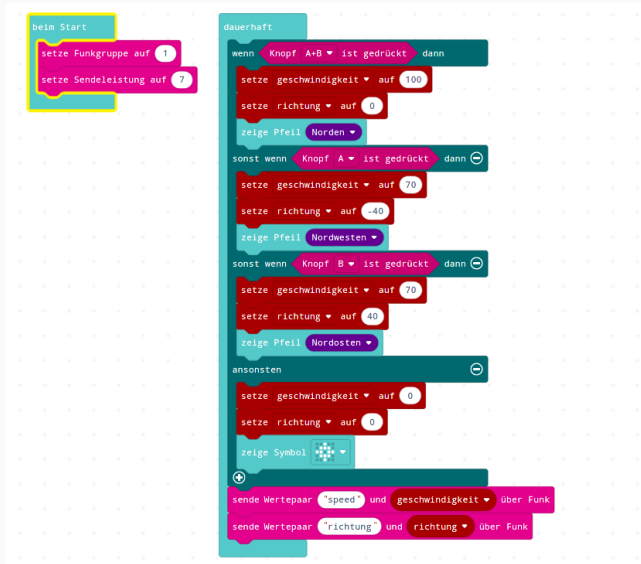


Nochmal die Aufgabe:

- Wenn **KEIN** Knopf gedrückt ist, dann **speed = 0**
- Wenn **Knopf A** gedrückt ist, dann **richtung = -40** und **speed = 80**
- Wenn **Knopf B** gedrückt ist, dann **richtung = 40** und **speed = 80**
- Wenn **Knopf A+B** gedrückt ist, dann **richtung = 0** und **speed = 90**



## Mögliche Lösung



The image shows a Scratch script for controlling a car. It is divided into two main sections: 'beim Start' (at start) and 'dauerhaft' (permanently).

**beim Start:**

- setze Funkgruppe auf 1
- setze Sendeleistung auf 7

**dauerhaft:**

- wenn Knopf A+B ist gedrückt dann
  - setze geschwindigkeit auf 100
  - setze richtung auf 0
  - zeige Pfeil Norden
- sonst wenn Knopf A ist gedrückt dann
  - setze geschwindigkeit auf 70
  - setze richtung auf -40
  - zeige Pfeil Nordwesten
- sonst wenn Knopf B ist gedrückt dann
  - setze geschwindigkeit auf 70
  - setze richtung auf 40
  - zeige Pfeil Nordosten
- ansonsten
  - setze geschwindigkeit auf 0
  - setze richtung auf 0
  - zeige Symbol

At the end of the 'dauerhaft' loop, there are two 'sende Wertepaar' blocks:

- sende Wertepaar "speed" und geschwindigkeit über Funk
- sende Wertepaar "richtung" und richtung über Funk



## Java-Script-Code

```
let richtung = 0
let geschwindigkeit = 0
radio.setGroup(1)
radio.setTransmitPower(7)
basic.forever(function () {
  if (input.buttonIsPressed(Button.AB)) {
    geschwindigkeit = 100
    richtung = 0
    basic.showArrow(ArrowNames.North)
  } else if (input.buttonIsPressed(Button.A)) {
    geschwindigkeit = 70
    richtung = -40
    basic.showArrow(ArrowNames.NorthWest)
  } else if (input.buttonIsPressed(Button.B)) {
    geschwindigkeit = 70
    richtung = 40
    basic.showArrow(ArrowNames.NorthEast)
  } else {
    geschwindigkeit = 0
  }
})
```



Hex-code



# 08\_05\_ExternerLautsprecher

Calliope-Kurs Kinder

---

Jogi Künstler, Turbine Brunnen

Herbst 2020



**Zusätzliche Batterie/Externer  
Lautsprecher**

---



Nachdem wir für den Motor die 6-polige Erweiterungs-Leiste aufgelötet haben, möchte ich noch zwei zusätzliche Möglichkeiten dieser Leiste aufzeigen. Man kann daran anschliessen:

- Zusätzliche Batterie für die Motoren
- Externer Lautsprecher, für mehr Krach



# Zusätzliche Batterie für Motor

Der Calliope wird - wie wir gelernt haben - normalerweise mit 2 Batterien a 1.5 Volt betrieben. Das macht zusammen 3 Volt.

Wenn man nun einen Motor anschliessen möchte, der mehr Spannung braucht, dann würde der sich damit kaum bewegen.

Darum haben die Erbauer des Calliope noch eine Möglichkeit vorgesehen, eine zusätzliche Batterie anzuschliessen!

Diese zusätzliche Batterie

- ist aber nur zum Betrieb des Motors/der Motoren gedacht
- sie muss zusätzlich zur normalen Batterie angeschlossen werden
- Sie muss zur Spannung des Motors / der zwei Motoren passen
- sie darf maximal 9V haben



Die zusätzliche Batterie wird an den beiden rechten Pins angeschlossen, ganz aussen Minus, daneben Plus

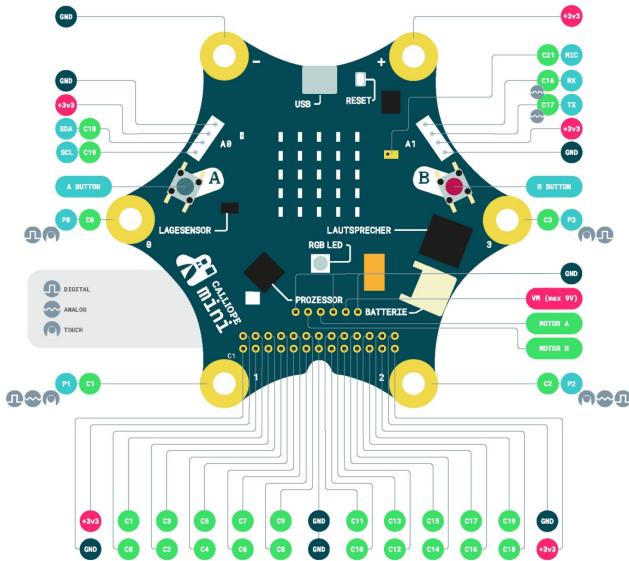
Auf dem offiziellen Bild des Calliope sieht man das ganz gut:

Quelle Bild nächste Seite :

[https://calliope-mini.github.io/assets/v10/img/Calliope\\_mini\\_1.3\\_pinout\\_fin.jpg](https://calliope-mini.github.io/assets/v10/img/Calliope_mini_1.3_pinout_fin.jpg):



# Offizielles Bild



# Detail-Ausschnitt

Hier nochmal ein Ausschnitt daraus, mit den Anschlüssen für die Batterie farblich gekennzeichnet:

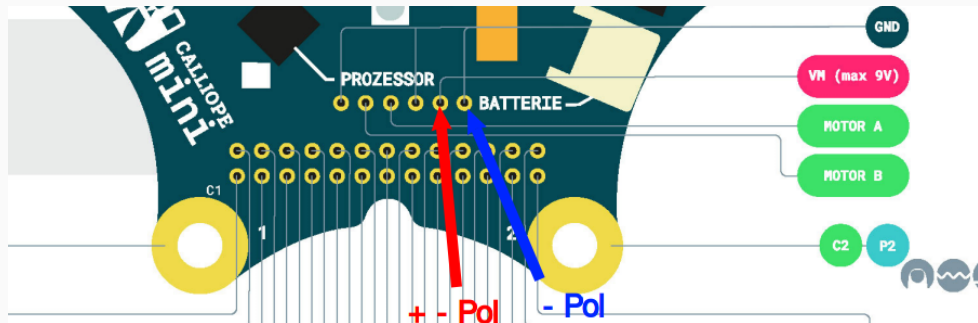
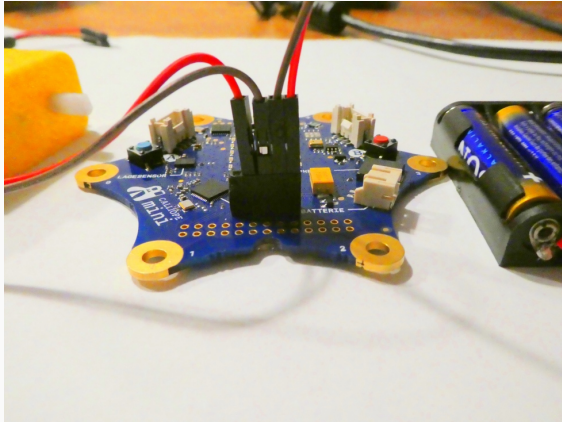


Figure 1: Ausschnitt

In Betrieb sieht das dann so aus:



**Figure 2:** Anschluss Calliope

Der eingebaute Lautsprecher am Calliope ist schon sehr klein und sehr leise.

An die Leiste kann man auch einen externen Lautsprecher anschliessen.

Dabei kann man entweder einen kleinen, sogenannten Passiv-Lautsprecher anschliessen, dieser braucht keine zusätzliche Stromversorgung.

Oder aber man kann einen sogenannten Aktiv-Lautsprecher anschliessen, das sind die Lautsprecher, wie man sie auch von Computern kennt, diese haben eine zusätzliche Strom-Versorgung.



# Ein alter Passiv-Lautsprecher



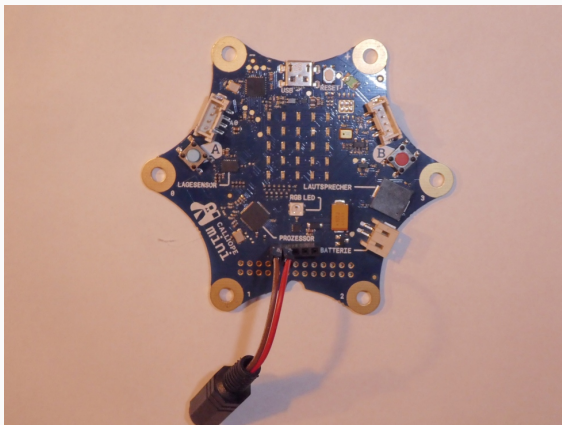
**Figure 3:** Lautsprecher





# Klinken-Anschluss

In Jedem Fall ist es sinnvoll, das über einen sogenannten Klinken-Stecker zu realisieren, der Anschluss sieht dann so aus:



**Figure 4:** Klinkenanschluss



# Klinkenstecker

Damit kann dann ein externer Lautsprecher oder Kopfhörer angeschlossen werden.

Der Klinken-Anschluss sieht so aus:



**Figure 5:** Klinkenstecker



# Auf dem Calliope-Schaltbild

Anschluss auf dem offiziellen Calliope-Schaltbild:

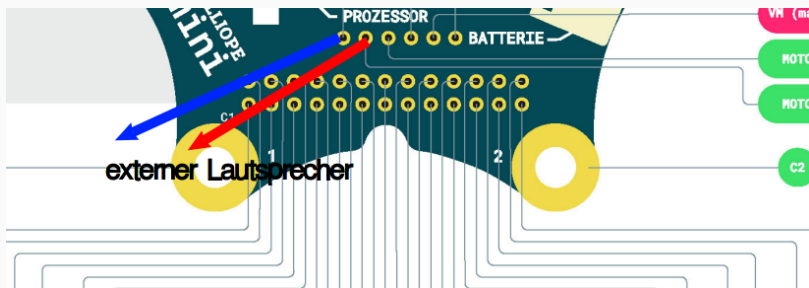


Figure 6: Lautsprecher

## ACHTUNG :

Da der Lautsprecher die gleichen Anschlüsse wie die Motoren verwendet, kann man nicht beides gleichzeitig in einem Programm machen. Man muss sich also entscheiden, ob man mit seinem Calliope Musik abspielen will, oder ob man Motoren ansteuern will.



- Zurück zu Tag 4 Hausaufgaben (Schleifen)
- Hoch zur Übersicht



Für alle Bilder auf diesen Folien/Seiten gilt:

- Autor: Jörg Künstner
- Lizenz: CC BY-SA 4.0

