

Um ein elektronisches Twister-Spiel zu bauen, benutzen die Kursteilnehmer*innen Variablen, Wenn-Dann-Schleifen und Zufallsfunktionen. Dafür füllen Sie Lückentexte aus.

Schlagworte: Calliope Mini; blockbasierte Programmiersprache; Variablen definieren; Wenn-Dann-Schleife; Zufallsfunktion; Array



E-Book Einzelbeitrag
von: Kai Feldheim

Twister mit Calliope

Neue Blockprogrammiersprache lernen

aus: Technik spielend (kennen)lernen (9783763972647)

Erscheinungsjahr: 2023

Seiten: 205 - 214

DOI: 10.3278/172647w018

Dieses Werk ist unter folgender Lizenz veröffentlicht: Creative Commons Namensnennung-Share Alike 4.0 International

TWISTER MIT CALLIOPE

NEUE BLOCKPROGRAMMIERSPRACHE LERNEN



Anzahl Teilnehmer*innen:

max. 5 + 2 Leiter*innen

Dauer: ca. 60–90 Min.

Autor: Kai Feldheim

Material: 1 x Calliope-mini-Controller

1 iPad pro Kind

mit der App „Calliope“

Blatt Papier + Stift

original Twister-Spiel



ZIEL DES WORKSHOPS

Ziel des Workshops ist es, ein Programm zu schreiben für den Calliope mini, damit er danach als Controller verwendet werden kann, der die klassische Twister-Dreh-scheibe ersetzt. Diese zeigt an, welches Körperteil auf welchem Farbfeld platziert werden muss.

Was die Kinder lernen?

- grundsätzliche Funktionsweise von Blockprogrammiersprachen
- einen Befehl formulieren
- Sprache in einen Code übersetzen bzw. einen bestehenden Code über Sprache interpretieren
- Coding-Begriffe: Wenn-Dann-Funktion, Variable, Zufallsgenerator, Array

Bei diesem Workshop kommen bereits gesammelte Erfahrungen mit Blockprogrammiersprachen zur Anwendung, d. h., der Workshop eignet sich nicht unbedingt zur Einführung in das Programmieren. Der schon fertige Code wird in diesem Workshop so weit verfremdet, abgeändert und in Form eines Lückentextes präsentiert, den die Kursteilnehmer*innen vervollständigen und so abändern müssen, dass das Programm funktioniert und es den Befehl ausführt, den mensch haben möchte.

VORBEREITUNG

1. Installation der App „Calliope“ auf den iPads
2. Calliope-mini-Controller mit korrektem Programm bereithalten

Da sich beim Übertragen des Programms über USB oder Bluetooth oftmals technische Fehler einschleichen und es zu Verzögerungen kommt, sollte in je-

dem Fall immer ein Calliope-mini-Controller bereitliegen, auf dem das Programm schon installiert und der Calliope also zum Spielen schon bereit ist.

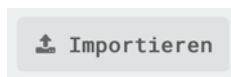
3. Code importieren

Folgender Code ist entweder per Link oder QR-Code in der App zu öffnen.

Die Vorgangsweise ist je nachdem etwas komplizierter, weil sowohl über den Link als auch über den QR-Code das Programm im Browser und nicht in der App geöffnet wird. Das heißt, es muss zunächst der Link kopiert werden und dann in der App über die Importieren-Funktion eingefügt werden. Das ist dann der Fall, wenn diese Anleitung nicht auf den iPads vorhanden ist und der Link also nicht einfach hieraus kopiert werden kann.

https://makecode.calliope.cc/_aicFpTDMHU4D

⇒ entweder Link kopieren und in der App unter
Code erstellen → MakeCode → Importieren



Importierte URL ...

Link einsetzen

oder



⇒ QR-Code scannen

Code bearbeiten

Code teilen → Projekt veröffentlichen → Link kopieren

App „Calliope“ öffnen

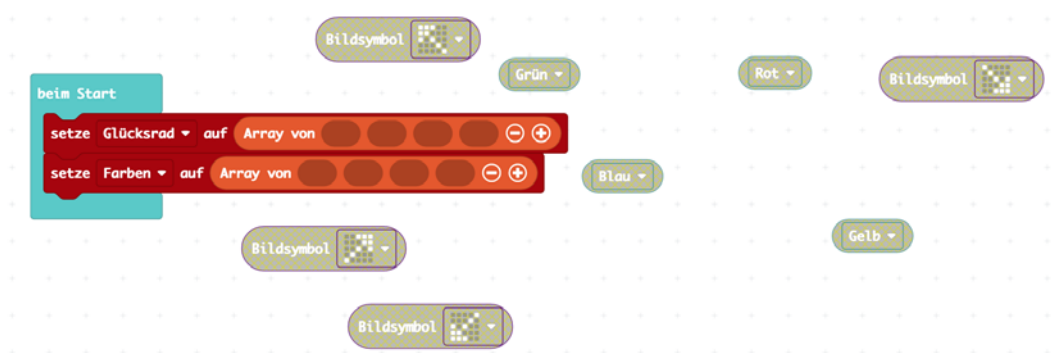
Code erstellen → Makecode → Importieren → Link einsetzen

4. Plakat des fertigen und funktionierenden Codes auf bspw. dem Whiteboard befestigen

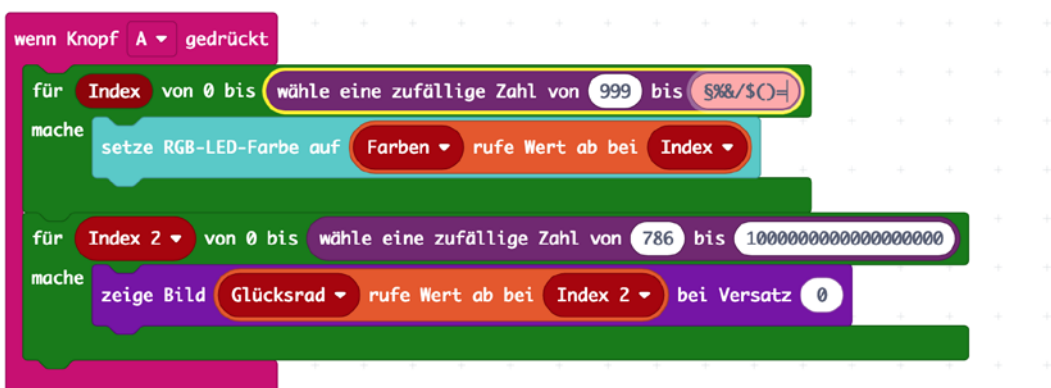
Didaktisch erlernen die Kursteilnehmer*innen das Schreiben des Programms über eine Art Laufdiktat. Dabei müssen sie den abgeänderten Code in Form eines verfremdeten Lückentextes vervollständigen und richtigstellen, indem sie eine Vorlage in Form eines Plakates kopieren.

5. Umwandeln des Codes in einen Lückentext

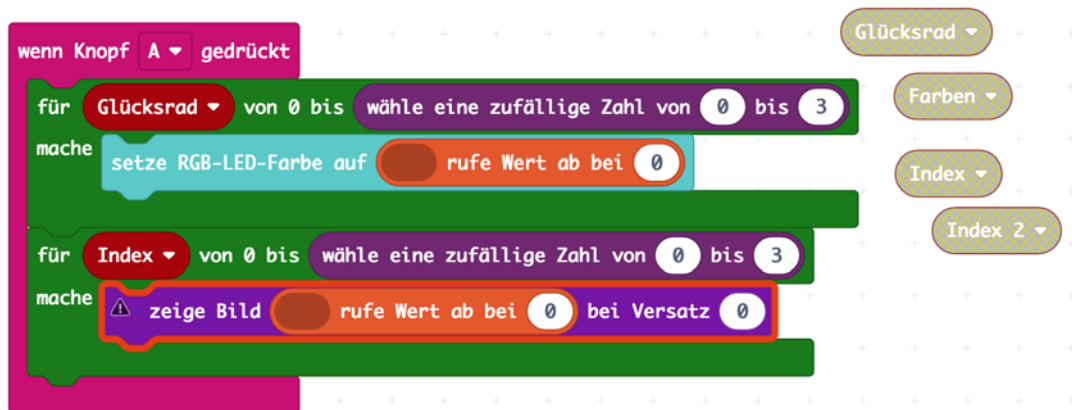
Achtung! Der Code in der App auf JEDEM iPad sollte so zu einem Lückentext verfremdet werden, dass der Schwierigkeitsgrad den Erfahrungsleveln und Kenntnissen der Kursteilnehmer*innen mit Blockprogrammiersprachen entspricht, beispielsweise ...



Beispiel 1: Hier müssen die Kursteilnehmer*innen die Pfeilsymbole in die richtigen Lücken im Array des Glücksrades und in die Farben im Array der Farben einsetzen.



Beispiel 2: Hier müssen die Kursteilnehmer*innen bei „wähle eine zufällige Zahl von ... bis ...“ die richtigen Zahlen 0 bis 3 einsetzen.



Beispiel 3: Hier müssen die richtigen Variablen an der richtigen Stelle eingesetzt werden.

6. Twister-Spiel mit Matte und Drehschreibe zur Demonstration bereithalten

NÜTZLICHES WISSEN

1. DER CALLIOPE MINI

Der Calliope mini ist ein sogenannter Mikrocontroller. Mikrocontroller imitieren große Computer mit verschiedenen Funktionen, die in der digitalen Welt überall eingesetzt werden und deren sogenannte Hardware – das heißt, das Sicht- und Anfassbare, also was mensch sieht und anfassen kann – über die sogenannte Software – also die Programme, die dafür geschrieben werden – Handlungsanweisungen bekommen und gesteuert werden können. Äußerlich hat der Calliope mit einem Computer ebenso viel gemein, weil er aussieht wie eine ganz übliche sogenannte Platine, die Steuerungszentrale von Computern. Nur ist diese Platine des Calliope mini, wie der Name schon verrät, sehr viel kleiner. Der Calliope-mini-Controller ist die sogenannte „Hardware“ – im Vergleich zur Software –, also das Haptische, was angefasst, gefühlt und gesehen werden kann.

Der Calliope mini verfügt über einen A- und einen B-Knopf zur Dateneingabe und über ein 5-x-5-LED-Feld in quadratischer Form, das heißt, es gibt insgesamt 25 herkömmliche LED-Leuchten, sowie über eine sogenannte RGB-LED-Leuchte, welche die Standardfarben im RGB-Spektrum anzeigen kann. Die weiteren Funktionen des Calliope sind für diesen Workshop nicht relevant und können aber in weiterführenden Workshops erläutert werden.

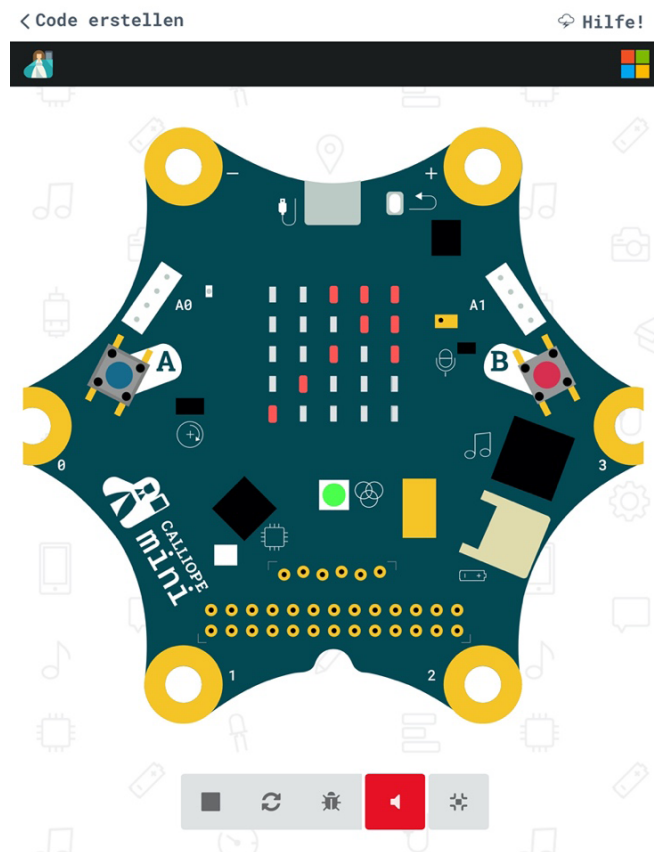
2. DIE APP „CALLIOPE“

In der App kann sowohl der Code für das Programm geschrieben und bearbeitet werden als auch der Calliope mini per Bluetooth mit dem iPad zur Übertragung des Programms gekoppelt werden. Die App ist im Vergleich zur Hardware die sogenannte „Software“, das Programm – etwas, was nicht angefasst und nicht wirklich gesehen werden kann, weil es digital ist –, mit welchem wir wiederum die Software bzw. ein Programm für den Controller schreiben, welches später dann darauf installiert und ausgeführt werden soll. Beim Programmieren wiederum kann menschliche Sprache in der Befehlsform also ganz logisch in eine Befehlsform in Programmiersprachen übersetzt werden, die einer ganzen einfachen Logik folgen.

Das Koppeln funktioniert über den Reiter „Verbinden“. Drücke am Calliope mini A+B und Reset, lass nach einer Sekunde los. Wenn das Balkenmuster auf der 5-x-5-LED-Anzeige am Calliope erscheint, kannst du A+B loslassen. Übertrage dann das Muster der LEDs des Calliope mini in die App. Oftmals gibt es beim Koppeln und insbesondere später beim Übertragen des Programms auf den mini technische Probleme, weshalb, wie gesagt, ein Calliope-mini-Controller mit dem installierten Programm, welches am Computer über USB schon übertragen wurde, bereitliegen sollte, damit das Spiel auch auf jeden Fall funktioniert und gespielt werden kann.

Im Reiter „Code erstellen → MakeCode: Editor öffnen“ sollten alle Programme zu finden sein. Wie du den Code importierst, wurde unter Schritt II Vorbereitung 3. Code importieren bereits beschrieben.

In der unteren rechten Ecke befindet sich eine digitale Simulation des Calliope mini, mit der der Code auf Richtigkeit überprüft werden kann. Durch Drücken der Vollbildtaste rechts oben wird die Simulation groß angezeigt und der Code kann überprüft werden, indem auf

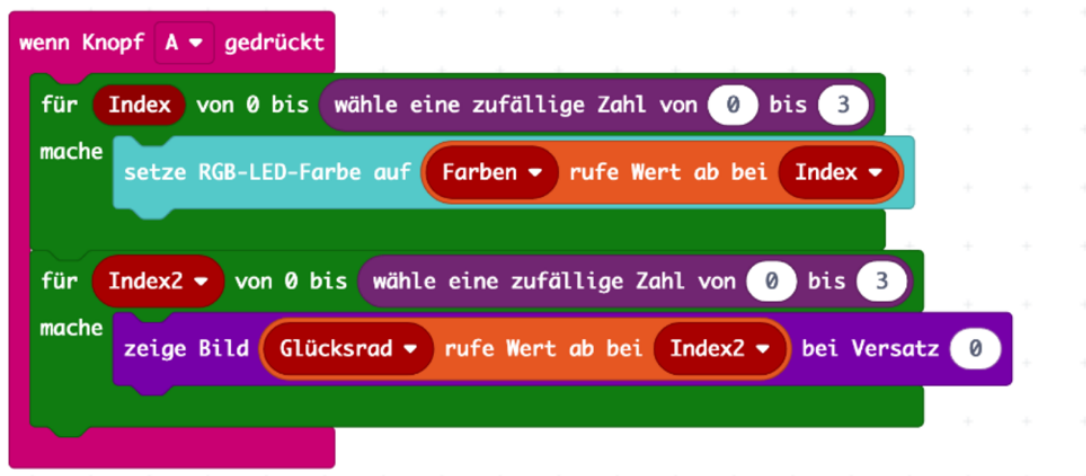


den A-Knopf per Touch gedrückt wird. Werden nun die Befehle wie gewünscht ausgeführt, dann ist der Code richtig. Mit jedem Drücken des A-Knopfes sollte sich eine zufällige Kombination aus Pfeilrichtung auf dem 5-x-5-LED-Feld und Farbe auf der RGB-LED ergeben.

3. DIE WENN-DANN-SCHLEIFE (IF-FUNKTION)

Wenn-Dann-Schleifen gehören zu den üblichsten und grundlegendsten und wichtigsten Funktionen beim Programmieren. Mit ihnen werden einem Computer oder Controller Handlungsanweisungen gegeben in der Form „Wenn das und das passiert, dann mach das und das ...“.

In unserem Programm verwenden wir eine solche „if-Funktion“ bzw. Schleife in der Form „Wenn Knopf A gedrückt wird, dann ...“. In der Blockprogrammiersprache MakeCode für den Calliope sowie in allen vergleichbaren Blockprogrammiersprachen ist sehr gut ersichtlich, warum es sich um eine sogenannte Schleife handelt, denn alles, was sich in dieser Schleife befindet, trifft auf diese Funktion zu, wie in der folgenden Abbildung zu sehen.



Die „Wenn-Dann-Schleife“ (hier in pink) wird Schleife genannt, weil alles wiederholt wird, was sich innerhalb dieser Schleife bzw. der pinken Klammer befindet, solange die Funktion „Wenn“ („Wenn Knopf A gedrückt“) erfüllt ist, also immer, wenn Knopf A von Neuem gedrückt wird.

4. VARIABLEN

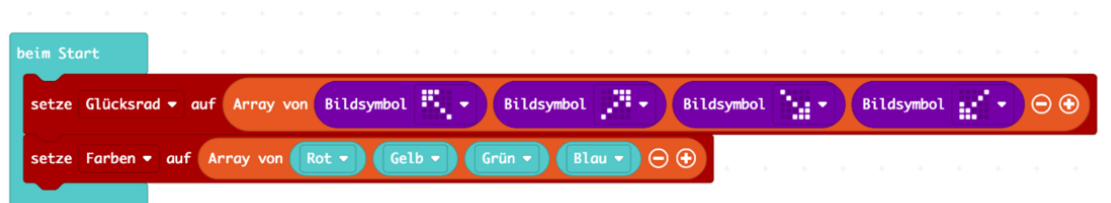
Beim Programmieren ist eine Variable ein leerer Behälter für eine Größe, die im Verlauf des Programms immer wieder zum Einsatz kommt und im Gegensatz zur

sogenannten Konstante immer wieder verändert werden kann. Für gewöhnlich erhalten Variablen im Programm einen Namen, sodass immer wieder ersichtlich ist, was gemeint ist. Dieser Name kann frei gewählt werden. In unserem Programm verwenden wir verschiedene Variablen, die im Programm immer wieder von vorne einen neuen Wert erhalten – einen Wert in Form einer Zahl, die für eine bestimmte Pfeilrichtung und also ein Körperteil steht, und einen Wert in Form einer Farbe, die auf der RGB angezeigt wird.

Da sich die Wertermittlung der Variablen innerhalb der „Wenn Knopf A gedrückt“-Schleife befindet, wird für die Variable bei jedem erneuten Drücken des A-Knopfes ein neuer Wert ermittelt.

5. ZUFALLSFUNKTION UND ARRAY

Am Anfang, beim Starten des Programms, werden die Variablen über einen sogenannten „Array“, zu Deutsch: eine Reihe von Werten, genauer gesagt, insgesamt vier Werten für jede Variable, definiert. Damit sagt das Programm dem Calliope zunächst, dass jede Variable einen aus einer Möglichkeit von vier Werten erhalten kann.



Damit in der Zufallsfunktion beim Ausführen des Programms auch eine Farbe bzw. eine Pfeilrichtung angezeigt werden kann, müssen in dem Programm numerische Werte, also Zahlenwerte, in eine Bedeutung übersetzt werden. Jede Zahl von 0 bis 3 (= 4 Werte) bekommt eine Bedeutung. Die Zahl 0 wäre beispielsweise die Pfeilrichtung oben links, die Zahl 1 die Pfeilrichtung oben rechts usw. Mit den Farben verhält es sich genauso. Die Zahl 0 bekommt eine Farbe zugeordnet, genauso wie die Zahlen 1, 2 und 3. Die Zufallsfunktion ist nun über zwei weitere Variablen, den Index für die Farbe und den Index 2 für die Pfeilrichtung, definiert. Was das Programm nun sagt, ist: Wenn Knopf A gedrückt wird, dann wird für die Indexe eine zufällige Zahl gesucht und wiederum jeder Zahl ist, wie zuvor beschrieben, ein Wert zugeordnet, der dann ausgegeben wird über die Anzeige auf dem 5-x-5-LED-Feld bzw. der RGB-LED:



KURSABLAUF

1. THEORIE IN FORM EINES QUIZ

- Was ist ein Computer, eine Platine, ein Mikro-Controller?
- Was ist Hardware?

Lass die Kursteilnehmer*innen den Calliope mini einmal in die Hand nehmen, er kann reihum gegeben werden oder jede*r Teilnehmer*in erhält einen eigenen. Dabei können Fragen gestellt werden.

- Habt ihr schon einmal Twister gespielt?
- Was muss beim Twister-Spielen mit der „Twister-Uhr“ herausgefunden werden?
- Worum geht es beim Twister eigentlich?
- Was ist eine Programmiersprache, was sind Befehle ...?
- Was sind Variablen, was ist ein Zufall ...?
- Wie viele unbekannte Größen gibt es beim Twister?

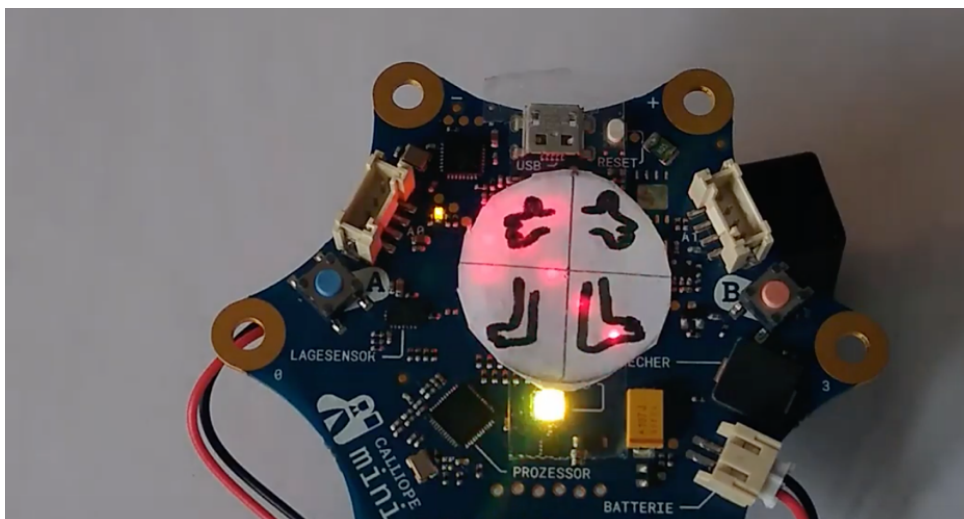
*Im Idealfall: Zeig den Kursteilnehmer*innen die „Twister-Uhr“ vom Originalspiel und lass sie herausfinden, was wir eigentlich mit dem Calliope später simulieren wollen. Tu so, als seist du der Computer, und fordere die Kursteilnehmer*innen auf, dir zu sagen, was du machen sollst. So kannst du die Brücke dazu schlagen, dass die Kursteilnehmer*innen lernen, dass eine Programmiersprache bzw. ein Co-der oder ein Programm in der ganz einfachen Befehlsform funktionieren, dass also menschliche Sprache nach einer gewissen Logik in eine Programmiersprache übersetzt werden kann.*

2. PROGRAMM SCHREIBEN

- Kursteilnehmer*innen erhalten pro Person ein vorbereitetes iPad
- öffnen der App „Calliope“ → Coder erstellen → MakeCode-Editor öffnen
- entsprechenden abgeänderten Quellcode-Lückentext öffnen
- Laufdiktat: Kursteilnehmer*innen müssen iPad am Platz lassen und immer wieder zwischen ihrem Platz und dem Plakat als Vorlage am Whiteboard hin- und herlaufen, um den richtigen Code Stück für Stück im Gedächtnis behalten, um ihn am eigenen iPad vervollständigen zu können.
- wenn Kursteilnehmer*innen fertig sind, sollen sie aufzeigen und kriegen den Simulator gezeigt, wo sie überprüfen können, ob der Code richtig ist
- Hilfestellung geben, die Kursteilnehmer*innen sich gegenseitig Hilfe geben lassen, bis alle fertig sind

3. LETZTE VORBEREITUNGEN ...

- einen Kreis in der Größe einer 2-€-Münze auf ein dünnes Blatt Papier zeichnen, ausschneiden und in vier gleich große Viertel, welche jeweils für ein Körperteil stehen, durch zwei gerade Linien teilen
- in jedem Viertel wird vermerkt, welches Körperteil gemeint ist, entweder durch Abkürzungen wie RF (rechter Fuß), LH (linke Hand) oder durch Zeichnungen
- Kreis sowie angeschlossenes Batteriekit mit Klemme mit mitgeliefertem Gummi-band am Calliope befestigen (Kreis auf 5-x-5-LED-Feld, Batterie auf der Rückseite)



4. SPIELEN ...

- Auswahl einer*s Spielleiter*in, die*der den Calliope bedient, während die anderen das Spiel auf der Twister-Matte spielen.
- Entweder, es wird so gespielt, dass jede*r Kursteilnehmer*in, die*der hinfällt oder einen falschen Befehl ausführt, die*den Spielleiter*in ablöst,
- oder, es wird nach Punkten und einem Zeitrahmen gespielt.