TECHNISCHE UNIVERSITÄT DRESDEN

Informatische Grundkompetenzen ermöglichen mit dem Calliope mini

– Modulhandbuch –

Inhaltsverzeichnis

Kurzinformation für die Lehrkraft....2
Erläuterungen zum Modulhandbuch....3
Fachliche Analyse zu Calliope mini und MakeCode-Editor....4
Lernziele....9
Unterrichtliche Voraussetzungen und Einbettung in den Schulunterricht....11
Tabellarischer Modulablaufplan....12
Hintergründe zu den eingesetzten Methoden und Inhalten....13
Detaillierte Beschreibung des Modulablaufs....15
(1) Einführung und Anknüpfung an die Lebenswelt der Schüler*innen....15
(2) Kennenlernen der Hardware....16
(3) Einstieg in die Programmierung (Kennenlernen des Editors)....19
(4) Entwicklung erster Programme mit grundlegenden Programmierkonzepten....19
(5) Umsetzung selbstgewählter Projekte....19
(6) Schlussphase zum Transfer....20
Zur Durchführung notwendige Materialien....20
Beschreibung der Herausforderungen 1 – 4 und der einzelnen Projekte....21
Literaturverzeichnis....36

Professur für Didaktik der Informatik

EDUiNF EDUCATION IN INFORMATICS

SYLBER

GEFÖRDERT VOM Bundesministerium für Bildung und Forschung

Das vorliegende Modulhandbuch wurde im Projekt TUD-Sylber² in der Didaktik der Informatik der TU Dresden erarbeitet. Die im Modul verwendeten Arbeitsblätter sind im InfoSphere - Schülerlabor Informatik der RWTH Aachen (CC BY-SA 4.0) entstanden und wurden im Projekt TUD-Sylber² in der Didaktik der Informatik der TU Dresden weiterentwickelt.
Das diesem Material zugrundeliegende Vorhaben wird im Rahmen der gemeinsamen „Qualitätsoffensive Lehrerbildung“ von Bund und Ländern mit Mitteln des Bundesministeriums für Bildung und Forschung unter dem Förderkennzeichen 01JA1919 gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt bei den Autoren.

CC BY SA

Verfasser*innen:
Erstellung der Arbeitsblätter: Maria Theißen, Überarbeitung durch Christin Nenner
Erstellung des Modulhandbuchs: Christin Nenner, Sina Jarosch, Überarbeitung Robin Hofmann

Kurzinformation für die Lehrkraft

Titel des Schüler*innen-Moduls: Calliope mini – Programmiere deinen ersten Mini-Computer!

Schulstufe: Grundschule, Unterstufe

Optimale Jahrgangsstufe: 3. bis 6. Klasse

Themenbereiche: Hard- und Software, Sensoren und Aktoren, EVA-Prinzip (Eingabe – Verarbeitung – Ausgabe), algorithmisches Denken, Programmieren

Einbindung in den Unterricht: Das Modul bietet erste Einblicke in das algorithmische Denken und Programmieren. Die Schüler*innen befassen sich mit dem EVA-Prinzip (Eingabe – Verarbeitung – Ausgabe) und machen erste Erfahrungen in einer einfachen Programmierumgebung. Das Modul deckt einen Teil des Lernbereichs 3: Begegnung mit Automaten und Robotern der Klasse 4 im Grundschulfach Werken ab.

Vorkenntnisse: keine

Dauer: ca. 4 h

Inhaltsbeschreibung: Im Rahmen des Moduls können Schüler*innen informatische Inhalte unter Verwendung des Calliope mini selbst entdecken. Thematischer Schwerpunkt sind sowohl das Erkennen, Nachvollziehen und selbst Erarbeiten von Algorithmen als aufgeschriebene Folge klarer, kleiner Schritte, die nacheinander abzuarbeiten sind, um eine Aufgabe zu lösen, als auch das Programmieren als die technische Umsetzung dessen. Mit dem Ziel sich spielerisch mit der Programmierung auseinanderzusetzen, entwickeln die Schüler*innen angeleitet erste Programme und können diese im Anschluss in Anlehnung an ihre eigenen Interessen erweitern bzw. eigene Programme schreiben.

Das Modul ist in sechs Phasen unterteilt:
(1) Einführung und Anknüpfung an die Lebenswelt der Schüler*innen,
(2) Kennenlernen der Hardware,
(3) Einstieg in die Programmierung (Kennenlernen des Editors),
(4) Entwicklung erster Programme mit grundlegenden Programmierkonzepten,
(5) Umsetzung selbstgewählter Projekte und

CC BY SA

(6) Schlussphase zum Transfer.

Erläuterungen zum Modulhandbuch

Das Modulhandbuch soll es Lehrkräften ermöglichen, einen Einblick in die didaktischen und fachlichen Hintergründe des Moduls für Schüler*innen ***Calliope mini – Programmiere deinen ersten Mini-Computer!*** zu gewinnen. Die Lehr-Lern-Materialien zum Calliope mini-Modul umfassen Arbeitsblätter zu Herausforderungen und Projekten, die die Schüler*innen selbst (in Zweiergruppen) bearbeiten können. Sie finden das Material zum Download auf der Webseite Lehr-Lern-Material zu den EduInf-Projekten der Professur für Didaktik der Informatik der TU Dresden.
Während der Umsetzung des Moduls kommt man nicht umhin, Begriffe einzuführen. Einige Begriffe,wie Hard- und Software, Programmieren und Algorithmus, die im Modul stark thematisiert werden, werden im Modulhandbuch kurz definiert. Natürlich kann dies nicht allumfassend sein. Als Ergänzung kann bspw. das Glossar https://calliope.cc/begriffe [1] und das Glossar der Beilage Kompetenzen für informatische Bildung im Primarbereich [2] ab Seite 18 dienen.
Zur Veranschaulichung und Erläuterung der Programmierumgebung und der genutzten Programmierblöcke werden Screenshots des Calliope mini MakeCode-Editors (Stand: November 2019) verwendet. Die Screenshots wurden im Rahmen des Projekts TUD-Sylber² in der Didaktik der Informatik der TU Dresden angefertigt.

[1] Calliope gGmbH (2020): *DISPLAY, HACKEN, CODEN Erklärungen zu einigen englischen Begriffen … und Glossar*. Online verfügbar (Zugriff am 01.11.2023) unter https://calliope.cc/begriffe

[2] Best et al., Gesellschaft für Informatik e.V. (2019): *Kompetenzen für informatische Bildung im Primarbereich.* Beilage zu LOG IN, Heft Nr. 191/192. Online verfügbar (Zugriff am 01.11.2023) unter https://dl.gi.de/bitstream/handle/20.500.12116/20121/61-GI-Empfehlung_Kompetenzen_informatische_Bildung_Primarbereich.pdf?sequence=1&isAllowed=y

CC BY SA

Fachliche Analyse zu Calliope mini und MakeCode-Editor

Zum Einstieg in das Thema werden der Calliope mini, und der MakeCode-Editor, vorgestellt.

Vorstellung Calliope mini

Der Calliope mini ist der erste programmierbare Mikrocontroller, der in Deutschland nach dem Vorbild der BBC micro:bit gezielt als Werkzeug für die Bildung für die Digitale Welt in der Grundschule entwickelt wurde [3]. Programmierbar heißt, man kann für diesen Mikrocontroller Programme entwickeln, die bestimmte Funktionen / Funktionalitäten nutzen und steuern. Programme sind eindeutige, ausführbare Anweisungen um eine Aufgabe / ein Problem Schritt für Schritt zu lösen (→ Algorithmen) in einer Sprache (→ Programmiersprache), die der Computer versteht.

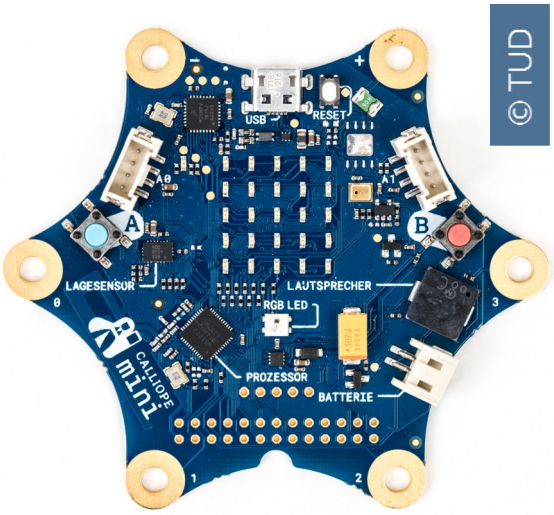


Abb. 1: Calliope mini von der Calliope gGmbH

Der Calliope mini versammelt sämtliche zum Betrieb notwendigen elektronischen Bauteile auf einer einzigen Leiterplatte. Die Bauteile sind, neben dem Prozessor, unter anderem Eingabe- und/oder Ausgabeanschlüsse wie das LED-Feld, die beiden Knöpfe »A« und »B«, die Anschlüsse P0, P1, P2 und P3, Lage- und Bewegungssensor, ein Lautsprecher, ein Helligkeitssensor und ein Funkmodul.

DETAILLIERTERE INFORMATIONEN ZUM CALLIOPE MINI

Informationen zum Einstieg finden sich unter:
https://calliope.cc/los-geht-s/einfuehrung [4]

Detaillierte Informationen zu den Bauteilen des Calliope mini finden sich unter:
https://calliope.cc/calliope-mini/uebersicht [5]

[3] Kind-Wissen-Zukunft e.V. (2019): *Calliope mini. Ein Handbuch für Grundschulen.* In enger Zusammenarbeit mit Calliope gGmbH. Online verfügbar (Zugriff am 01.11.2023) unter https://calliope.cc/media/pages/schulen/schulmaterial/335095658-1567083343/kiwiz_190110_calliope-handbuch.pdf

[4] Calliope gGmbH (2020): Calliope mini – Informationen zum Einstieg. Online verfügbar (Zugriff am 01.11.2023) unter https://calliope.cc/los-geht-s/einfuehrung

CC BY SA

Im Rahmen des Moduls werden die Bauteile mithilfe von der Vervollständigung des Legerätsels eingeführt und vorgestellt:

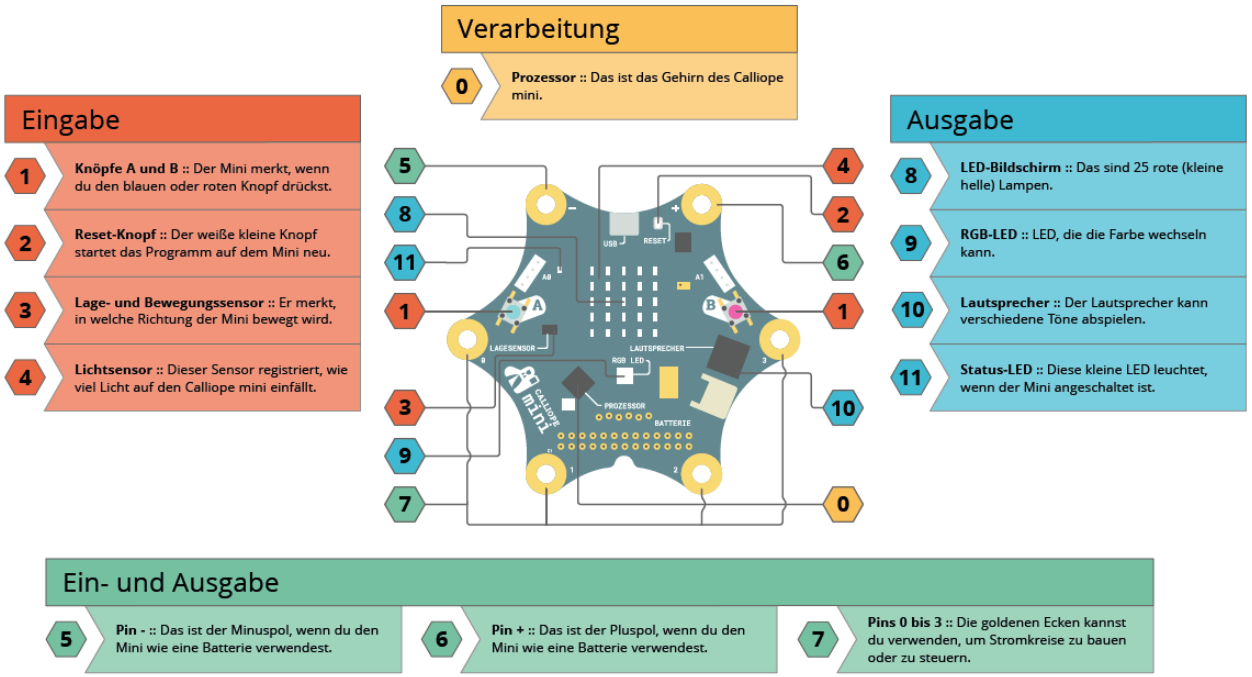


Abb. 2: Grafik Lösung des Legerätsels zum Calliope mini. Verwendetes Material vom InfoSphere - Schülerlabor Informatik der RWTH Aachen, Creative Commons Namensnennung - Weitergabe unter gleichen Bedingungen 4.0 International (CC BY-SA 4.0), weiterbearbeitet im Projekt TUD-Sylber² in der Didaktik der Informatik der TU Dresden.

MATERIAL FÜR SCHÜLER*INNEN

Auf dem **Poster „**CALLIOPE: HALLO, CALLIOPE MINI**“** [6] werden die Bauteile des Calliope mini für die Zielgruppe Schüler*innen kurz vorgestellt.

Für den Betrieb benötigt der Calliope mini Strom. Die Stromversorgung kann über die Batteriebox **oder** den Anschluss über USB an einem PC erfolgen. Der Calliope mini sollte **nicht** gleichzeitig mit Strom über die Batteriebox und den USB-Anschluss des PCs versorgt werden.

[5] Calliope gGmbH (2020): *Ein Microcontroller mit vielfältigen Möglichkeiten!*. Online verfügbar (Zugriff am 01.11.2023) unter https://calliope.cc/calliope-mini/uebersicht

[6] Calliope gGmbH (2020): *Poster CALLIOPE: HALLO, CALLIOPE MINI*. Online verfügbar (Zugriff am 01.11.2023) unter: https://calliope.cc/media/pages/los-geht-s/einfuehrung/1cb8f6d6c8-1581542438/calliopemini_poster_hardware.pdf

Vorstellung MakeCode-Editor

Zum Entwickeln der Programme verwenden die Schüler*innen im Calliope mini-Modul den MakeCode-Editor https://makecode.calliope.cc/ von Microsoft. Es folgt ein Screenshot des MakeCode-Editors mit dem Start-Programm, das ein dauerhaft blinkendes lachendes Gesicht zeigt.

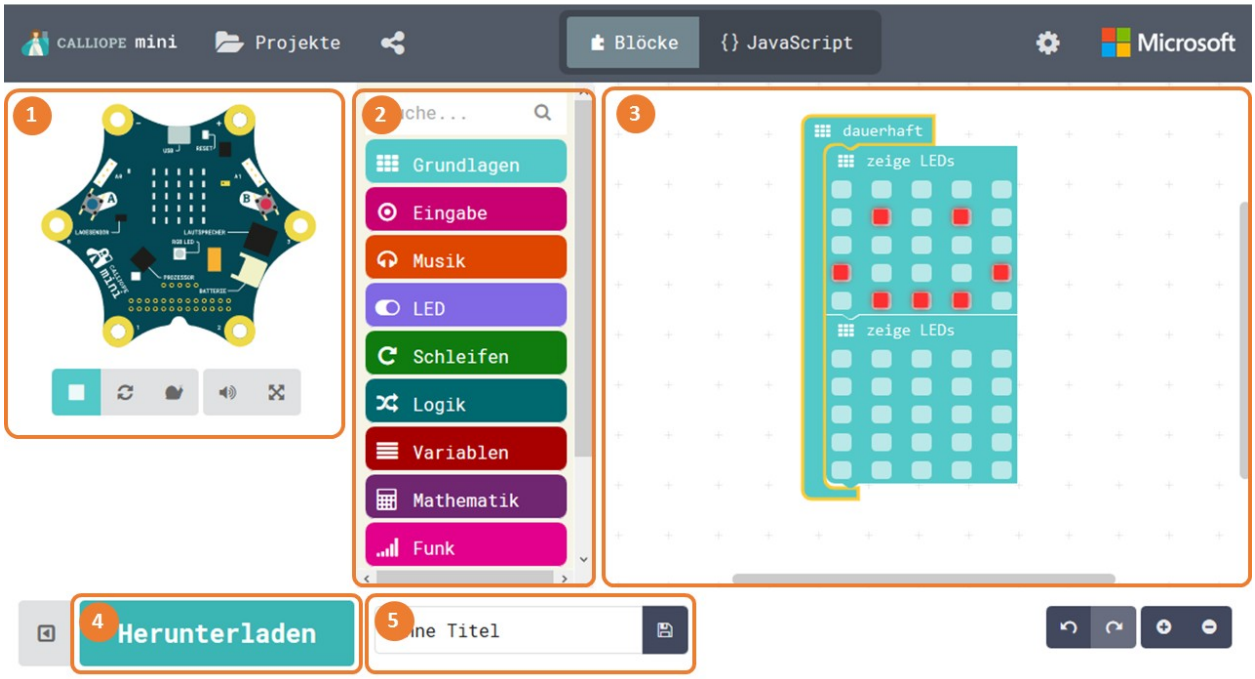


Abb. 3: Screenshot des MakeCode-Editors (https://makecode.calliope.cc/, Stand: November 2023) mit dem Start-Programm. Es sind fünf Bereiche des Editors mit Zahlen gekennzeichnet: 1) Simulator, 2) Menü mit den Programmierblöcken, 3) Programmierfläche, 4) Speicherleiste und 5) Titelleiste.

Diese Programmierumgebung ist browserbasiert, d.h. man benötigt einen Computer mit einem installierten Internet-Browser wie Microsoft Explorer, Microsoft Edge (ab Windows 10), Mozilla Firefox, Google Chrome, Opera oder Safari. Nach einer einmaligen Initialisierung kann im MakeCode-Editor auch ohne Internetanbindung programmiert werden.

Im Editor sagt man der Hardware – in diesem Fall dem Calliope mini - was sie tun soll. Der Editor ist also das Werkzeug, das man benutzt, um Programme zu schreiben. Im Editor https://makecode.calliope.cc/ entwickeln die Schüler*innen Programme mithilfe von Blöcken. Blockprogrammierung (auch: Programmierung mit graphischen oder visuellen Programmiersprachen) bedeutet, dass Blöcke per Drag & Drop aneinander gesetzt werden, anstatt Codezeilen in Textform zu schreiben. Die Blöcke werden in Form und Beschriftung hinsichtlich ihrer Funktion im Programm unterschieden. Das heißt, Anweisungen, Variablen, Kontrollstrukturen etc. werden kategorisiert und im Editor mit unterschiedlichen Farben dargestellt.

Die Blöcke sind mit Noppen-Konnektoren, die an Puzzleteile erinnern, ausgestattet. Dadurch wird neben der Ähnlichkeit zur spielerischen Tätigkeit auch das Merken von Befehlsnamen oder Code-Konventionen wie Klammern, Zeilenendzeichen etc. überflüssig.

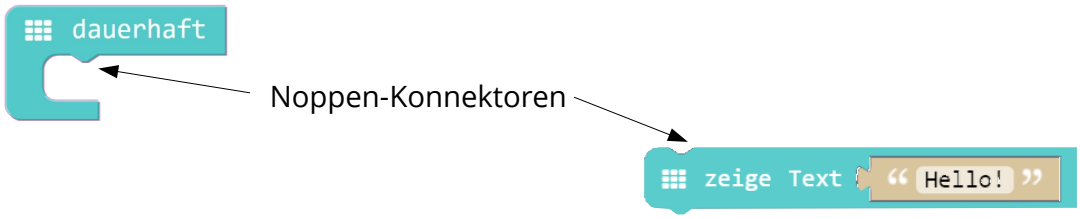


Abb. 4: Noppen-Konnektoren bei der blockbasierten Programmierung im MakeCode-Editor

Ein funktionierendes Programm erkennt man daran, dass die Blöcke in den Farben ihrer Kategorie, wie zum Beispiel mintgrün für Blöcke aus *Grundlagen*, angezeigt werden. Ein nicht-funktionierendes Programm erkennt man daran, dass die Blöcke ausgegraut sind (siehe Abb. 5).

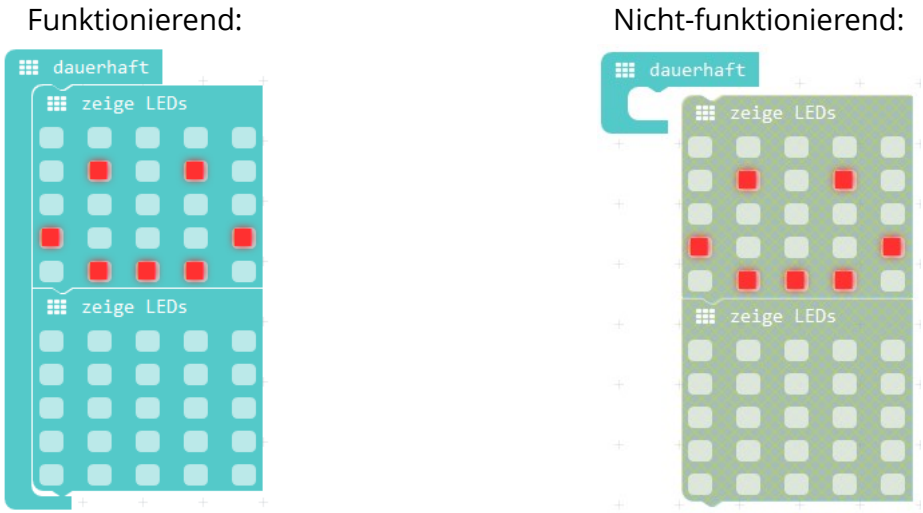
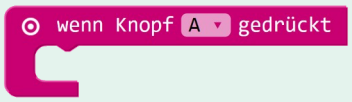
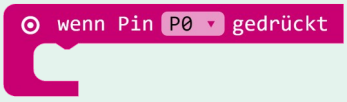


Abb. 5: Gegenüberstellung eines funktionierenden und eines nicht-funktionierenden Programms

Ein grau gefärbter Block wird vom Calliope mini nicht erkannt. Das die Programmierblöcke bzw. ein Programmteil ausgegraut sind, kann daran liegen, dass die Programmierblöcke nicht in ein Programm (Folge) eingefügt sind oder der Programmierblock bereits genauso existiert.

CC BY SA

Programmierblöcke, wie 

oder 

ermöglichen die Nutzung verschiedener Knöpfe bzw. Pins, die über den kleinen Pfeil nach unten, rechts neben dem *A* bzw. dem *P0*, festgelegt werden. Drückt man auf den Pfeil, öffnet sich eine Liste, aus der man die verfügbaren Knöpfe bzw. Pins auswählen kann.

Die Stellen der Programmierblöcke, an denen man selbst etwas ändern oder etwas aus einer Liste auswählen kann, sind heller als der Rest der Programmierblöcke dargestellt.

Um ein entwickeltes Programm auf den Calliope mini zu übertragen, muss der Calliope mini über ein Mikro-USB-Kabel an den Computer/Laptop, auf dem der Editor mit dem Programm geöffnet ist, angeschlossen werden.

ANLEITUNGEN ZUM ANSCHLIEßEN DES CALLIOPE MINI

Das **Anschließen des Calliope mini an einen Computer über ein USB-Kabel** findet sich hier kurz erläutert: https://calliope.cc/projekte/anleitung-2 [7]
Eine ausführlichere Anleitung findet sich in **Calliope mini. Ein Handbuch für Grundschulen.** auf der Seite 15, B.01-11. [3]

Um die Übertragung des Programms auf den Calliope mini zu starten, drückt man auf den großen grün-blauen Herunterladen-Knopf (siehe Abb. 3).

Beim Herunterladen ist der Speicherort zu beachten. Wo etwas gespeichert wird, kann in den Einstellungen des Browsers festgelegt werden. Ist der Calliope mini an den Computer/Laptop angeschlossen erscheint er als Wechseldatenträger MINI. Der MINI kann als fester Speicherort über die Auswahl von *Alle Dateien in folgendem Ordner abspeichern: …* festgelegt werden. Eine andere Möglichkeit ist die Variante *Jedes Mal nachfragen, wo eine Datei gespeichert werden soll* auszuwählen. In der oben erwähnten Anleitung [7] wird das Vorgehen erläutert, wenn das Programm beim Speichern direkt im Ordner Downloads abgelegt wird.

[7] Calliope gGmbH (2020): *Anleitung-2.* Online verfügbar (Zugriff am 01.11.2023) unter https://calliope.cc/projekte/anleitung-2

CC BY SA

Beim Übertragen des Programms blinkt eine kleine gelbe Leuchte, die Status-LED, auf dem Calliope mini. Daran kann man sehen, dass das Programm gerade dort gespeichert wird.

Im Editor erscheint ein Fenster. Dort wird auf *Fertig!* geklickt.

Erst wenn das Blinken ganz vorbei ist, ist das Programm vollständig auf den Calliope mini übertragen.

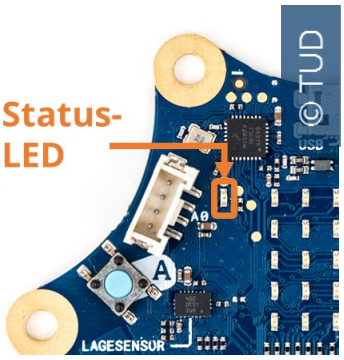


Abb. 6: Calliope mini-Ausschnitt mit Status-LED

Zu beachten ist, dass auf dem Calliope mini immer nur EIN Programm gespeichert werden kann. Das heißt, sobald ein neues Programm heruntergeladen und übertragen wird, wird das vorherige ersetzt und ist **nicht** mehr auf dem Calliope mini gespeichert.

ERKLÄRVIDEOS ZUM EDITOR UND ZUM PROGRAMMIEREN MIT BLÖCKEN

Genauere Informationen zu Aufbau und Funktionsweise des MakeCode-Editors sowie zur Programmierung mit Blöcken im MakeCode-Editor finden sich unter den folgenden Links:
Editor: https://www.youtube.com/watch?v=76ryLoWfYhs
Programmierung: https://www.youtube.com/watch?v=Bzt7cnlBZsg

Zur Umsetzung des vorgestellten Calliope mini-Moduls wird der MakeCode-Editor von Microsoft verwendet. Informationen zu weiteren Calliope mini-Editoren finden Sie hier: https://calliope.cc/los-geht-s/editor.

Lernziele

Ziel des Moduls und der entwickelten Arbeitsblätter ist es, Schüler*innen zu ermöglichen informatische Inhalte unter Verwendung des Mikrocontrollers Calliope mini selbst zu entdecken. Thematischer Schwerpunkt ist sowohl das Erkennen, Nachvollziehen und selbst Erarbeiten von Algorithmen, als auch das Programmieren als die technische Umsetzung dessen. Zur spielerischen Auseinandersetzung mit der Programmierung, entwickeln die Schüler*innen, angeleitet durch Arbeitsblätter, erste Programme und können diese im Anschluss in Anlehnung an ihre eigenen Interessen erweitern bzw. eigene Programme schreiben.

CC BY SA

- Die Schüler*innen sind nach diesem Modul in der Lage, Beispiele für Informatiksysteme in ihrer Lebenswelt zu benennen wie Ampel, Fahrkartenautomat, etc.
- Die Schüler*innen können Hardware und Software sicher unterscheiden.
- Die Schüler*innen können die Begriffe Hardware und Software mit eigenen Worten erläutern.
- Die Schüler*innen können das EVA-Prinzip mit seinen Bestandteilen *Eingabe*, *Verarbeitung* und *Ausgabe* erklären.
- Die Schüler*innen können den Begriff Programmieren mit eigenen Worten erläutern.
- Die Schüler*innen können einfache blockbasierte Programme in einer einfachen Programmierumgebung entwickeln.
- Die Schüler*innen können Programme auf externe Hardware übertragen.
- Die Schüler*innen testen entwickelte Programme auf externer Hardware.

Am Ende des Angebots verstehen die Schüler*innen die Funktionsweise von Informatiksystemen durch erstes Programmieren ein wenig mehr. Sie kennen eine Auswahl an Programmierbefehlen, den Grundaufbau von Programmen und haben einen Einblick in die Arbeitsweise von Informatiker*innen bekommen.

Anknüpfungspunkte zur Strategie der Kultusministerkonferenz „Bildung in der digitalen Welt“ [9] sind unter anderem:

- im Kompetenzbereich 5 Problemlösen und Handeln
    - *Unterpunkt 5.5: Algorithmen erkennen und formulieren* | Im beschriebenem Modul lernen die Schüler*innen, durch das Entwickeln erster Programme für den Calliope mini, Funktionsweisen und grundlegende Prinzipien der digitalen Welt kennen und verstehen. Dies ermöglicht ihnen algorithmische Strukturen in genutzten digitalen Werkzeugen zu erkennen und zu formulieren. Weiterhin planen die Schüler*innen angeleitet eine strukturierte, algorithmische Sequenz zur Lösung eines Problems.

[9] Kultusministerkonferenz (2016): *Strategie der Kultusministerkonferenz „Bildung in der digitalen Welt“*. Online verfügbar (Zugriff am 01.11.2023) unter https://www.kmk.org/fileadmin/Dateien/pdf/PresseUndAktuelles/2017/Strategie_neu_2017_datum_1.pdf

CC BY SA

Unterrichtliche Voraussetzungen und Einbettung in den Schulunterricht

Den Hauptteil des Calliope mini-Moduls macht das eigenständige Bearbeiten von Arbeitsblättern und Umsetzen der darauf beschriebenen Aufgaben in der Programmierumgebung (Editor) im individuellen Lerntempo aus. Als Grundvoraussetzung für die Bearbeitung ist die Lesefähigkeit und der kompetente Gebrauch von Maus und Tastatur, um den Drag & Drop-Editor bedienen zu können, zu sehen.

In die Grundschullehrpläne in Sachsen (Stand 2019) wurde die informatische Grundbildung mit folgendem Satz aufgenommen: „Im Rahmen einer informatischen Vorbildung eignen sich die Schüler elementare Bedienfertigkeiten im Umgang mit dem Computer oder mobilen digitalen Endgeräten an und gewinnen Einblicke in deren Funktionsweisen und nutzen diese bei der Lösung von Aufgaben.“ (Als Beispiel im Grundschullehrplan Werken, Seite VIII [9], aber auch in allen anderen Grundschullehrplänen an dieser Stelle)

Im Grundschullehrplan Werken [10] sind in der Klassenstufe 4 mit dem *Lernbereich 3: Begegnung mit Automaten und Robotern* informatische Inhalte wie das EVA-Prinzip und der Einblick in eine einfache Programmierumgebung mit Sachverhaltsanalyse, Ableitung eines Modells sowie der Programmierung eines einfachen Ablaufs inkl. Prüfen der Funktionalität verankert. Diese Themen werden im Material zum Calliope mini bearbeitet.

Weitere Anknüpfungsmöglichkeiten finden sich im Grundschullehrplan der Mathematik in Sachsen im Rahmen der Themen Variable, Würfel, Zufall und Wahrscheinlichkeit.

Zudem visiert der Sachunterricht als zentrales Bildungsziel die Unterstützung der Schüler*innen bei der sachgerechten und verstehenden Erschließung ihrer Lebenswelt an. Da die Lebenswelt der Schüler*innen stark digital geprägt ist, kommen sie mit informatischen Inhalten und Konzepten in Berührung, bei deren Verständnis der Sachunterricht sie unterstützen kann.

[10] Freistaat Sachsen, Staatsministerium für Kultus (aktualisiert 2019): *Lehrplan Grundschule Werken*. Seite 14. Online verfügbar (Zugriff am 01.11.2023) unter https://www.schulportal.sachsen.de/portal/getFile.php?lplanid=73&lplansc=5qcMpegw6pQpLsPppKnZ&aktion=lplanfile&token=44051aa1b1c0577b32b61c74f2d24d29

CC BY SA

Tabellarischer Modulablaufplan

Zeit	Phase	Inhalt	Medium	Sozialform
0:00 – 0:05	Begrüßung	Vorstellung von EduInf und der Betreuenden	Präsentation	Vortrag
0:05 – 0:20	(1) Einführung und Anschluss an die Lebensumwelt der Schüler*innen	Abfrage der Teilnehmenden zu Computer/Smartphone, Hardware, Software, Programmieren; Schritt-für-Schritt-Anleitungen im Alltag; Informatiksysteme im Alltag; Ziel und Ablauf des Moduls	Präsentation, Tafel/ Whiteboard	Plenum
0:20 – 0:40	(2) Kennenlernen der Hardware	Calliope mini begutachten und erproben; Im Legerätsel die Bauteile des Calliope mini zuordnen; Auflösung im Plenum zum EVA-Prinzip mit Eingabe, Verarbeitung und Ausgabe	Calliope mini mit Test-Programm, Legerätsel mit Schnipsel	Gruppenarbeit (je nach TN-Anzahl 3 – 6 TN/Gruppe) Plenum
0:40 – 0:50	(3) Einstieg in die Programmierung (Kennenlernen des Editors)	Vorstellung des MakeCode-Editors als Programmierumgebung: Ändern des Start-Programms im Editor, Löschen und Hinzufügen einzelner Programmierblöcke, ausgegraute Blöcke, Speichern eines Programms, Vorstellen des Simulators, Drag & Drop	Präsentation, Bildschirmvorführung (an interaktiver Tafel)	Vortrag (evtl. Video schauen durch TN)
0:50 – 1:35	(4) Entwicklung erster Programme mit grundlegenden Programmierkonzepten	Vorstellung der Herausforderungen durch Betreuende; Bearbeitung der Herausforderungen 1, 2, 3 und 4 in dieser Reihenfolge; Zusammenfassung und Visualisierung der Erfahrungen der Phase; (funktionierendes Programm, Steuerungselemente, Programm auf Calliope mini speichern, Programm testen)	Internetfähige Computer, Calliope mini mit USB-Kabel, Laminierte Programmierblöcke, Tafel/ Whiteboard	Plenum Gruppenarbeit (2 TN/Gruppe) Plenum
1:35 –	PAUSE			

CC BY SA

1:50				
1:50 – 3:40	(5) Umsetzung selbstgewählte Projekte	Vorstellung der Projekte durch Betreuende; Auswahl und Bearbeitung mind. eines der 6 mitgebrachten Projekte (ca. 50 min); Entwicklung und Umsetzung einer eigenen Projektidee (ca. 60 min)	Internetfähige Computer, Calliope mini mit USB-Kabel	Plenum Gruppenarbeit (2 TN/ Gruppe), Projektarbeit
3:40 – 3:55	(6) Schlussphase zum Transfer	Zusammenfassung zu Projektentwicklung und –umsetzung, Arbeitsweise von Programmierer*innen. Programm u. Schritt-für-Schritt-Anleitung; Feedback	Präsentation, Tafel/ Whiteboard	Plenum
3:55 – 4:00	Verabschiedung	Dank, Kontakt, Infos zu EduInf	Präsentation	Vortrag

Hintergründe zu den eingesetzten Methoden und Inhalten

Im Mittelpunkt des Calliope mini-Moduls steht die Aktivität der Schüler*innen, damit sie selbst Dinge ausprobieren und ihre eigenen Erfahrungen machen können. Einen Blick hinter die Kulissen der Geräte, die uns umgeben, zu wagen wird fokussiert. Durch das schaffen positiver Erlebnisse mit der Technik, werden Hemmungen und Ängste abgebaut sowie die Neugier geweckt. Die Lehr- und Unterstützungsfunktion übernehmen die Betreuer*innen. Wie ein Coach begleiten und unterstützen sie die Schüler*innen [11].

Die Schüler*innen arbeiten im Anschluss an die Einführungsphase in Zweiergruppen. Sie teilen sich einen Computer, einen Calliope mini und ein Set an Arbeitsblättern zu den Herausforderungen (arbeitsgleich) und den jeweiligen Projekten (arbeitsteilig).

Durch die Anleitungen auf den Arbeitsblättern zu den Herausforderungen und Projekten können die Schüler*innen in ihrem individuellen Lerntempo lernen und arbeiten. Aus den vorbereiteten Projekten können sie je nach Interesse an den Themen und/oder angezeigtem Schwierigkeitsgrad der Projekte wählen. Dadurch wird es den Schüler*innen möglich, ihren eigenen Lernprozess zu gestalten und mitzubestimmen.

Wird der Beginn des Moduls noch stark gelenkt und von einer geleiteten Sicherungsphase unterbrochen, tritt das leitende Eingreifen mit jeder Phase weiter zurück. Dies entspricht dem didaktischen Prinzip des aktiven Lernens und der abnehmenden Hilfe [12]. Zu Beginn des Moduls wird noch gemeinsam im Plenum die Lösung des Lege-Rätsels zur Hardware

[11] Schubert, Sigrid und Andreas Schwill (2011): *Didaktik der Informatik*. Heidelberg: Spektrum.
[12] Baumann, Rüdeger (1996): *Didaktik der Informatik*. Stuttgart: Klett.

CC BY SA

des Calliope mini besprochen. Die blockbasierte Programmierumgebung können die Schüler*innen dann in Zweiergruppen selbst entdecken und sich an den Herausforderungen orientieren. Um den Transfer des in den Herausforderungen erarbeiteten Wissens zu ermöglichen, wird eine Zusammenfassung mit der gesamten Gruppe angeschlossen. Bei der Bearbeitung der Projekte gehen die Zweiergruppen selbstständig vor. Betreuer*innen stehen bei Fragen und Problemen jederzeit zur Verfügung und unterstützen die Gruppen falls nötig.

> Den Schüler*innen wird nicht direkt die *richtige* Lösung serviert, sondern das eigene Denken der Schüler*innen durch Nachfragen von den Betreuenden angeregt. Oft ist das detaillierte gemeinsame Durchgehen der Arbeitsblätter hilfreich. Ziel ist es, es den Schüler*innen zu ermöglichen, selbst eine Lösungsstrategie zu entwickeln und zu erproben.

Im Anschluss an die, mithilfe der Arbeitsblätter, angeleiteten Projekte können die Schüler*innen eigene Projekte planen und umsetzen.

Übergeordnetes Ziel ist es, dass die Schüler*innen eine Folge von Anweisungen (einzelne Befehle) in einer Programmiersprache entwickeln, die der Computer ausführen kann und damit eine Aufgabe erfüllt. Die simpelste Programmierstruktur besteht aus einer Sequenz, also einer Folge von Anweisungen, die hintereinander ausgeführt werden kann. Ohne, dass die Schüler*innen es mitbekommen, erstellen sie eine Sequenz (eine lineare Abfolge), wenn sie den Schritten der Materialien folgen.

Die Arbeitsschritte, die die Schüler*innen im Rahmen des Moduls durchlaufen, sind angelehnt an das Vorgehen und die Arbeitsweise von Informatiker*innen beim Programmieren. Die Schüler*innen lesen und verstehen Handlungsvorschriften für das Arbeiten mit Informatiksystemen durch die Erarbeitung der Beispielprogramme. Sie gehen entsprechend der Aufgaben auf den Arbeitsblättern Schritt für Schritt vor und überprüfen zwischendurch die Funktion ihres Programms. Sie testen die Schwellenwerte für Sensormessungen und finden Fehler in gegebenen Werten. Sie erkennen durch Beobachtung dessen, was der Calliope mini macht, Programmfehler und Denkfehler. Sie korrigieren diese durch Veränderungen am Programm. Laut Bers sind diese Schritte besonders wichtig, da Programmieren eine Tätigkeit ist, bei der Fehler normal und erwartet sind. Sie verwenden Konzepte oder Blöcke, die sie von vorherigen Arbeitsblättern kennen, wieder und bauen darauf auf [13]. Sie bauen Teilprogramme aus Anweisungsblöcken zusammen und bilden dann aus kleinen fertigen Teilprogrammen ein Gesamtprogramm.

[13] Bers, Umaschi Marina et al. (2014): Computational thinking and tinkering: Exploration of early childhood robotics curriculum. Computers and Education 72, S. 145-157.

CC BY SA

Detaillierte Beschreibung des Modulablaufs

Das Modul ist in sechs Phasen unterteilt: (1) Einführung und Anschluss an die Lebenswelt der Schüler*innen, (2) Kennenlernen der Hardware, (3) Einstieg in die Programmierung (Kennenlernen des Editors), (4) Entwicklung erster Programme mit grundlegenden Programmierkonzepten, (5) Umsetzung selbstgewählter Projekte und (6) Schlussphase zum Transfer und für Feedback. Im Folgenden wird auf die einzelnen Phasen eingegangen.

(1) Einführung und Anknüpfung an die Lebenswelt der Schüler*innen

Um in die Thematik Informatiksysteme und Programmieren im Alltag einzuführen, werden im Modul Fragen an die Schüler*innen gestellt. Begonnen wird mit einer Frage zur Nutzung von digitalen Geräten wie Computern/Laptops/Tablets/Smartphones. Im Anschluss wird nach den Begriffen Hard- und Software gefragt. Es findet eine erste gemeinsame Annäherung (im Plenum) an die Begriffe Hardware und Software statt. Als nächstes wird erfragt, wer schon einmal etwas programmiert hat und dies als Impuls genutzt, um gemeinsam zu besprechen, was Programmieren bedeutet.

Informatiksystem	Ein Informatiksystem ist die spezifische Zusammenstellung von Hardware, Software und deren Vernetzung zur Lösung eines Anwendungsproblems. [2]
Hardware	Die Hardware ist die materielle Ausstattung des Informatiksystems. [2, 13, 14]
Software	Die Software ist die nicht-materielle Ausstattung des Informatiksystems und umfasst Programme, die bestimmen was und wie das Informatiksystem etwas ausführt. [2, 13, 14]
Programmieren	Programmieren bedeutet, dem Informatiksystem eine eindeutige und ausführbare Schritt-für-Schritt-Anleitung in einer Programmiersprache vorzugeben. [1, 2, 14]

LITERATURHINWEIS

Detaillierte Erläuterungen zu den Begriffen Hard- und Software finden sich
… im **Heft „**Smartphone und andere Computer**“ im Kapitel 1, Seite 6** [14].
Dort finden sich zudem Informationen zu Computern, ihrer Funktionsweise und ihrem Vorkommen.

[14] Bundeszentrale für politische Bildung/bpb (2019): *einfach Politik: Smartphone und andere Computer.* Online verfügbar (Zugriff am 01.11.2023) unter http://www.bpb.de/shop/lernen/einfach-politik/300205/smartphone-und-andere-computer

CC BY SA

Zur Veranschaulichung des Programmierens lässt sich einer der Betreuenden von den Schüler*innen programmieren. Daran wird deutlich, wie wichtig konkrete und endliche Anweisungen sind, damit genau das umgesetzt wird, was man möchte. Als Anknüpfung an die Lebenswelt der Schüler*innen dienen Abfolgen von Schritt-für-Schritt-Anweisungen, die die Schüler*innen aus ihrem Alltag kennen. Man erfragt, ob sie Schritt-für-Schritt-Anweisungen von zuhause bzw. aus Schule oder Hort kennen und welche das beispielsweise sind (Zähne putzen, Schuhe zubinden, Ankommen in der Schule, Hausaufgaben machen, Abmelden im Hort, Kochrezepte, etc.). Danach wird erläutert, dass Informatiksysteme solche Schritt-für-Schritt-Anleitungen in einer Sprache benötigen, die sie verstehen. Ansonsten kann die Anleitung nicht ausgeführt werden.

Algorithmus	Ein Algorithmus umfasst eindeutige, ausführbare Anweisungen um eine Aufgabe/ein Problem Schritt für Schritt zu lösen. Um Aufgaben an eine Maschine/einen Computer zu übergeben, müssen die Anweisungen in eine Programmiersprache übersetzt werden, die die Maschine/der Computer verarbeiten kann. Die in einer Programmiersprache geschriebene Folge von Handlungsanweisungen heißt Programm. [2, 14]

Anhand eines Wimmelbilds werden anschließend Informatiksysteme im Alltag der Schüler*innen, wie Ampeln, Klimaanlagen, Navigationsgeräte, Aufzügen, etc. identifiziert.

LITERATURHINWEIS

Weitere Hintergrundinformationen zum Thema Algorithmen finden sich
… im **Heft „**Informatik entdecken – mit und ohne Computer**“ ab Seite 19** [15]
… und im **Heft „**Smartphone und andere Computer**“ im Kapitel 1, Seite 14 f** [13]

(2) Kennenlernen der Hardware

Zum Kennenlernen der Hardware, in diesem Fall der Calliope mini, bekommen die Schüler*innen pro Zweiergruppe einen Calliope mini mit bereits aufgespieltem Test-Programm ausgeteilt. Sie werden nun aufgefordert den Calliope mini erst einmal durch Drücken der vorhandenen Knöpfe, Schütteln, Drehen und Betasten näher kennenzulernen. Sie dürfen den Calliope mini also haptisch erkunden. Dabei stellen sie fest, dass der Calliope mini unterschiedliche Dinge kann und auf unterschiedliche

[15] Stiftung „Haus der kleinen Forscher“ (2017): *Informatik entdecken – mit und ohne Computer.* Berlin, 1. Auflage. Seite 19. Online verfügbar (Zugriff am 01.11.2023) unter https://www.stiftung-kinder-forschen.de/fileadmin/Redaktion/1_Forschen/Themen-Broschueren/Broschuere_Informatik_2017.pdf

CC BY SA

Dinge reagiert, bspw. erscheint etwas auf dem LED-Feld, Töne bzw. Melodien erklingen, etc.

Mit dem Legerätsel übertragen die Schüler*innen das, was sie bezüglich der Funktionen des Gerätes gerade gesehen, gemacht und gehört haben, auf eine abstraktere Ebene. Es wird überlegt welche Aktion (Eingabe) – Bsp.: Drücken von Knopf A – zu welcher Reaktion (Ausgabe) – Bsp.: A erscheint auf LED-Feld – führt bzw. geführt hat. Bausteine mit ähnlichen Funktionen - Eingabe, Verarbeitung und Betrieb, Ausgabe – sind geclustert. Dadurch wird das direkte Anknüpfen an das EVA-Prinzip möglich.

MATERIAL FÜR SCHÜLER*INNEN

Auf dem Poster **CALLIOPE: EINGABE UND AUSGABE** [16] werden die Eingabe- und Ausgabemöglichkeiten des Calliope mini dargestellt.

Nach dem EVA-Prinzip funktionieren grundlegend alle Informatiksysteme, in dem sie Daten einlesen → Eingabe, berechnen → Verarbeitung und ausgeben → Ausgabe. Beispiele aus der Lebenswelt der Schüler*innen sind u. a. Ampeln, Fahrkartenautomaten und die elektronische Ausleihe von Bibliotheksbüchern.

EVA-Prinzip	EVA ist die Abkürzung für Eingabe – Verarbeitung – Ausgabe. Es beschreibt die grundlegende Funktionsweise von Informatiksystemen. Sie nehmen Eingaben an, verarbeiten diese und geben am Schluss die Ergebnisse aus. Eine Eingabe ist das, was „von außen“ als Signal an das Programm übergeben, also empfangen, und im Programm verarbeitet wird: Datenströme, Steuerung des Geräts durch Benutzer*innen oder die Messwerte von Sensoren. Eine Ausgabe ist das Aussenden und Anzeigen von Signalen oder Informationen durch die Hardware. [2]

Eingabe	Verarbeitung	Ausgabe
Knopf für (ermäßigte) Einzelfahrkarte drücken	→	Anzeigen des Fahrkartenpreises
Einwerfen der passenden Geldstücke in ausreichender Menge	→	Wenn ausreichend Geld eingeworfen wurde, Drucken des Fahrscheins

Tab. 1: Beispiel Fahrkartenautomat

[16] Calliope gGmbH (2020): *Poster CALLIOPE: EINGABE UND AUSGABE*. Online verfügbar (Zugriff am 12.03.2020) unter: https://calliope.cc/calliope-mini/uebersicht

CC BY SA

LITERATURHINWEIS

Informationen zum Thema EVA-Prinzip finden sich
... im **Dokument Kompetenzen für informatische Bildung im Primarbereich auf Seite 20** [2].

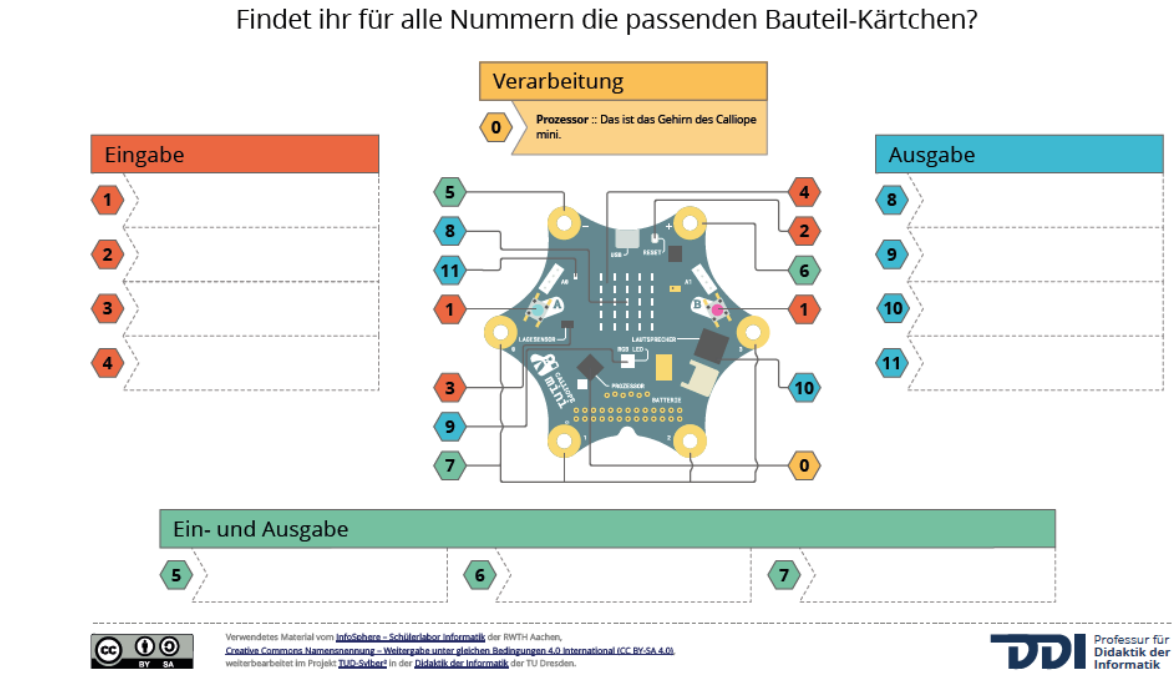


Abb. 7: Legerätsel zur Hardware Calliope mini (Lösung in Abb. 2)

In der Mitte des Legerätsels ist eine vergrößerte graphische Darstellung des Calliope mini. Pfeile nummerieren jedes zu benennende Bauteil. Auf den Bauteilkärtchen steht der Name des Bauteils. Die Beschriftungen auf dem Calliope mini selbst vereinfachen die Zuordnung der Bauteilkärtchen.

LITERATURHINWEIS

Eine detaillierte Einführung zum Calliope mini mit Erklärungen zu den einzelnen Bauteilen sowie dem Erstellen und Speichern von ersten Programmen finden sich
... im **Buch Das Calliope-Buch. Spannende Bastelprojekte mit dem Calliope-Mini-Board**. [17]

[17] N. Bergner, P. Franken, J. Kleeberger, T. Leonhardt, M. Lukas (2017): Das Calliope-Buch. Spannende Bastelprojekte mit dem Calliope-Mini-Board. Dpunkt.verlag. Online verfügbar (Zugriff am 01.11.2023) unter

(3) Einstieg in die Programmierung (Kennenlernen des Editors)

Im nächsten Schritt wird die Programmierumgebung, der Editor, vorgestellt. Ziel dieses Abschnitts ist es, dass Schüler*innen in der Lage sind, selbstständig mit dem Editor zu programmieren und die erstellten Programme auf den Calliope mini herunterzuladen.

Der Editor wird im Modul frontal an einer interaktiven Tafel durch die betreuenden Personen vorgestellt und demonstriert. Dazu wird der Editor im Browser geöffnet und die einzelnen Bereiche sowie Ändern des Start-Programms im Editor, Löschen und Hinzufügen einzelner Programmierblöcke, ausgegraute Blöcke, Speichern eines Programms und Drag & Drop erläutert.

(4) Entwicklung erster Programme mit grundlegenden Programmierkonzepten

In dieser Grundlagen-Phase bearbeiten die Schüler*innen vier aufeinander aufbauende Herausforderungen zu den grundlegenden Programmierkonzepten Sequenzen, Schleifen, Verzweigung, Ein- und Ausgabe. Es werden somit Konzepte eingeführt, die in den Projekten Anwendung finden und vertieft werden.

Ein erstes Zusammenfassen nach der Umsetzung der Herausforderungen ermöglicht das Reflektieren der durchgeführten Tätigkeiten und das Erschließen des informatischen Hintergrunds (bspw. Aufbau einer Sequenz mit der notwendigen Eingabe (wie Knopf A gedrückt, geschüttelt) und der darauffolgenden Ausgabe (die Zeichenfolge anzeigen, Herz auf LED-Feld). Zudem wird darauf eingegangen, dass informatisches Arbeiten aus Planungs- und Umsetzungsphase besteht [18].

(5) Umsetzung selbstgewählter Projekte

Im Anschluss arbeiten die Schüler*innen, angeleitet durch Arbeitsblätter, selbstständig in Zweiergruppen an Projekten. Die Arbeitsblätter umfassen detaillierte Aufgaben, ergänzt durch Hintergrundinformationen in Hinweisboxen. Betreuer*innen stehen bei Fragen und Problemen jederzeit zur Verfügung und unterstützen die Gruppen falls nötig.

Die vorbereiteten Projekte sind unterschiedlich schwierig bzw. komplex. Der Schwierigkeitsgrad ist mithilfe von Kronen auf dem Deckblatt der Projekte oben rechts (1 Krone = leicht, 2 Kronen = mittel, 3 Kronen = schwierig) gekennzeichnet.

Für einige der Projekte werden zusätzliche Materialien benötigt.

Nachdem vorgegebene Projekte durch die Schüler*innen umgesetzt wurden, können als Erweiterung ganz eigene Projektideen erarbeitet und erprobt werden. Oft ist genau dieser

https://dpunkt.de/produkt/das-calliope-buch/.

[18] Schroeder, Ulrik und Nadine Bergner (2015): *Informatik Enlightened - Informatik (neu) beleuchtet dank Physical Computing mit Arduino.* INFOS 2015, 16. GI-Fachtagung Informatik und Schule, S. 43–52.

CC BY SA

Modulteil in dem sie Eigenes ausprobieren können, das Highlight für die Schüler*innen. Daran wird für sie deutlich, was sie nun am Ende des Moduls im Vergleich zum Beginn können.

Es kann zudem sehr motivierend auf die Schüler*innen wirken, wenn ihre selbst erdachten Projekte abgefilmt und dadurch den Eltern, Geschwistern, Freunden gezeigt werden können.

(6) Schlussphase zum Transfer

In der Schlussphase werden die Geschehnisse des Moduls noch einmal kurz in Erinnerung gerufen und zusammengefasst. Dabei wird auf Informatiksysteme im Alltag, das EVA-Prinzip und das Entwickeln von Programmen eingegangen.

Abschließend evaluieren die Schüler*innen und begleitenden Lehrkräfte das durchgeführte Modul.

Zur Durchführung notwendige Materialien

- Calliope mini-Sets (enthalten sind: Calliope mini, Batteriepack mit zwei AAA-Batterien/Akkus, USB-Kabel)
- Internetfähige Geräte zum Aufrufen der browserbasierten Programmierumgebung (Editor): https://makecode.calliope.cc/
- Für das Projekt „Küchenmusik“ sind zusätzlich 5 Krokodilklemmen, und metallisches leitfähiges Besteck von Nöten.
- Für das Projekt „Nachtlichter“ braucht man neben den Calliope mini-Sets 3 selbstgebastelte Häuser, Kupferklebeband (ca. 50 – 60 cm pro Durchführung/ Gruppe), mind. 3 Leuchtdioden pro Durchführung/Gruppe. Zudem ist eine Knopfzelle (Lithium-Batterie CR2032) zum Testen der Leuchtdioden sinnvoll.

CC BY SA

Beschreibung der Herausforderungen 1 – 4 und der einzelnen Projekte

Herausforderung 1

In der Herausforderung 1 machen die Schüler*innen aus dem im Editor zum Start angezeigten blinkendem lachenden Gesicht ein blinkendes Herz.

Ziel ist es, dass die Schüler*innen ein bereits vorhandenes Programm im Editor verändern und auf den Calliope mini überführen. Das veränderte Programm wird nun auf dem Calliope mini ausgeführt.

Es müssen keine neuen Blöcke in den Programmierbereich gezogen werden, sondern nur die einzelnen, die LEDs auf dem LED-Bildschirm darstellenden, Punkte angetippt werden, um sie rot bzw. nicht rot zu schalten. Damit das Herz blinkt muss im dauerhaft-Block nach dem Herz-LED-Feld ein leeres LED-Feld sein. Durch das dauerhafte Anzeigen des Herz-LED-Felds gefolgt vom leeren LED-Feld (→ Wechsel von Herz und leerem Feld) blinkt das Herz.

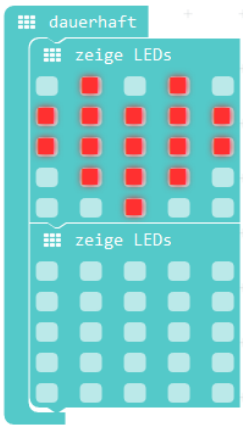
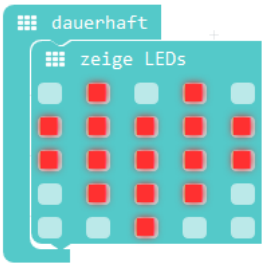
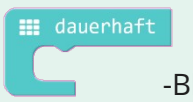
	
Wechsel von Herz und leerem Feld	Herz wird dauerhaft angezeigt, es blinkt nicht

Abb. 8: Vergleich zwischen den Programmen zur Erzeugung eines dauerhaft blinkenden und eines nicht blinkenden Herzens auf dem LED-Feld des Calliope mini.

Erläuterung auf dem Arbeitsblatt (Zielgruppe: Schüler*innen):

WAS IST EIN DAUERHAFT BLOCK?

Alles, was innerhalb des  -Blocks steht, wird vom Calliope mini die ganze Zeit, „dauerhaft“, durchgeführt.

CC BY SA

Herausforderung 2

Aufbauend auf der ersten Herausforderung wird in der Herausforderung 2 das Herz um eine selbstgewählte Botschaft erweitert.

Die Schüler*innen erweitern das bestehende Programm um einen neuen Programmierblock. Dafür suchen sie aus dem Menü mit den Programmierblöcken einen neuen Programmierblock heraus, legen diesen auf der Programmierfläche ab und fügen ihn per Drag & Drop in das bestehende Programm ein. Die Schüler*innen sehen das erste Mal einen Block auf der Programmierfläche, der noch nicht in ein Programm eingefügt wurde und deshalb ausgegraut ist. Sie lernen, solche Blöcke zu erkennen und dass ein Programmierblock in das bestehende Programm eingefügt werden muss, damit er eine Funktion erfüllt.

Erläuterung auf den Arbeitsblättern (Zielgruppe: Schüler*innen):

WAS BEDEUTET ES, WENN EIN PROGRAMMIERBLOCK GRAU IST?

Ein grau gefärbter Block, wie

wird vom Calliope mini **nicht erkannt**.

Das der Programmierblock grau ist, liegt daran, dass er **nicht in ein Programm (Folge) eingefügt** ist.

Durch das Ersetzen des vorhandenen Textes „Hello!“ im Programmierblock *Zeige Text* mit einer selbstgewählten Botschaft, stellen die Schüler*innen fest, dass in helleren Bereichen der Programmierblöcke etwas verändert werden kann.

Erläuterung auf den Arbeitsblättern (Zielgruppe: Schüler*innen):

WAS IST EINE FOLGE?

Ein Programm lässt sich ganz einfach schreiben. Dafür müsst ihr dem Computer nur die Reihenfolge vorgeben:
Aktion 1
Aktion 2
....

Der Computer macht zuerst Aktion 1, danach Aktion 2, ... Das heißt in der Fachsprache Folge oder Sequenz.

Wegen des

-Blocks fängt das Programm nach der letzten Aktion wieder bei Aktion 1 an. Man nennt das Endlosschleife.

CC BY SA

Herausforderung 3

In der Herausforderung 3 wird ein Programm entwickelt, das ausschließlich beim Start des Calliope mini angezeigt wird.

Die Schüler*innen lernen, dass es in der Programmierung verschiedene Möglichkeiten zur Steuerung von Abläufen gibt. Dazu werden Kontrollstrukturen genutzt. Es wird gezeigt, dass es neben der Kontrollstruktur *dauerhaft* noch weitere Kontrollstrukturen wie beispielsweise den Programmierblock *beim Start* gibt. Die Schüler*innen bauen ein bestehendes Programm um. Dazu dekonstruieren sie das Programm, um es anschließend nach den Vorgaben auf dem Arbeitsblatt zu rekonstruieren. Die Schüler*innen testen das Programm und vergleichen es mit dem Programm aus Herausforderung 2.

Erläuterung auf den Arbeitsblättern (Zielgruppe: Schüler*innen):

WAS IST EIN START BLOCK?

Alles, was innerhalb des

-Blocks steht, wird als erstes vom Calliope mini gemacht, danach aber nicht mehr wiederholt. Der *beim Start*-Block wird also nur ein einziges Mal (beim Start) ausgeführt.

Herausforderung 4

In der Herausforderung 4 wird ein Programm entwickelt, das ausschließlich beim Drücken des Knopfs A gestartet wird.

Die Schüler*innen lernen, dass Interaktionen zwischen Mensch und Technik ermöglicht werden können. Das heißt, dass eine Aktion des Menschen zu einer Reaktion der Technik führt. Hier wird das **EVA-Prinzip** mit **Eingabe, Verarbeitung** und **Ausgabe** angewendet. Das erstellte Programm ermöglicht dem Calliope mini eine **Eingabe** – in diesem Fall das Drücken des Knopfs – zu verarbeiten (**Verarbeitung**) und dann eine Reaktion darauf auszugeben (**Ausgabe**) – in diesem Fall das Anzeigen eines selbstgewählten Textes. Für die Verarbeitung ist der Prozessor zuständig.

Die Zusatzaufgabe für Schnelle fragt nach der Generierung verschiedener Botschaften als Reaktion auf das Drücken verschiedener Knöpfe. Ziel ist es, die Schüler*innen dazu anzuregen selbst einen Programmteil zu entwickeln. Konkret umfasst das Lösen der Aufgabe das Herauszufinden, welche Knöpfe es noch gibt, das Ausdenken weiterer Reaktionen des Calliope mini (Ausgabe) und das selbstständige Umsetzen im Editor. Die Schüler*innen beschäftigen sich hier schon gezielt mit der Situation, entwickeln ihr erstes Modell, setzen das Modell in der Programmierumgebung um und testen es anschließend.

CC BY SA

Falls das Testergebnis noch nicht dem entwickelten Modell entspricht, entwickeln die Schüler*innen es iterativ weiter. Ihre Arbeitsschritte sind angelehnt an das Vorgehen und die Arbeitsweise von Informatiker*innen beim Programmieren.

Erläuterung auf den Arbeitsblättern (Zielgruppe: Schüler*innen):

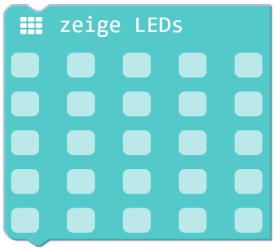
WAS IST EINE EINGABE?

Nun habt ihr gelernt, was **Eingabe** bedeutet. Ihr habt ein Programm erstellt, ***das reagiert***, wenn ihr den Knopf A ***drückt***.

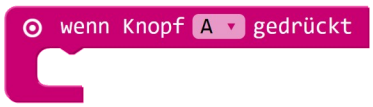
Euer Programm erkennt also, was ihr mit dem **Calliope mini** macht: Das Programm erkennt, welchen Knopf ihr drückt.

Zusammenfassung Herausforderungen 1 - 4

In den Herausforderungen 1- 4 lernen die Schüler*innen die Steuerungsblöcke *dauerhaft* und *beim Start* (informatischer Fachbegriff: Kontrollstrukturen) mit ihren Funktionalitäten kennen und nutzen. Das EVA-Prinzip wird angewendet und der Begriff Eingabe eingeführt. Es werden Ausgaben über das LED-Feld ausgeführt.

Kategorie	Block	Funktion
Grundlagen		Steuerung (Kontrollstruktur) → was innerhalb des *dauerhaft*-Blocks steht, wird endlos angezeigt
		Steuerung (Kontrollstruktur) → was innerhalb des *beim Start*-Blocks steht, wird ausschließlich beim Start angezeigt
		Ausgabe über LED-Feld
		Ausgabe über LED-Feld

CC BY SA

Eingabe		Eingabe über Knöpfe A / B / A+B

Aufbauend auf diese Grundlagen können die Schüler*innen im Anschluss Projekte auswählen und umsetzen.

Projekt Quiz-Buzzer

Das Projekt Quiz-Buzzer ist das einfachste der Projekte. Es ist daher mit nur einer Krone gekennzeichnet.

Als Einstieg in die Projekte wird in den ersten Sätzen auf den Arbeitsblättern auf das Ziel des Projekts eingegangen. Im Fall Quiz-Buzzer wird eingeführt mit: „Wer drückt zuerst? Der Calliope mini kann nach diesem Projekt erkennen, welcher der beiden Knöpfe Knopf A oder Knopf B zu erst bzw. schneller gedrückt wird."

Im Projekt Quiz-Buzzer entwickeln die Schüler*innen ein Programm, das den Calliope mini anzeigen lässt, welcher Spieler von Zweien, als erstes den Knopf - entweder A oder B - gedrückt hat. Es entsteht eine Art Buzzer, mit dem zwei Spieler gegeneinander antreten können. Beim Drücken des Knopfes A erscheint eine andere Ausgabe auf dem LED-Feld als für das Drücken des Knopfes B. Zudem kann eine dritte Ausgabe angezeigt werden, wenn die Knöpfe A und B gleichzeitig gedrückt werden.

Ziel des Projekts Quizz-Buzzer ist es, dass sich die Schüler*innen selbst Gedanken über die genaue Ausgestaltung des Programms machen. Dieser Schritt ist für die Arbeit von Softwareentwickler*innen von großer Wichtigkeit. [19] (→ Überschneidung zu Aufgabe für Schnelle in Herausforderung 4). Die Schüler*innen lernen in diesem Projekt eine weitere Möglichkeit der Eingabe (Schütteln) kennen und wenden diese an. Sie entwickeln ein Programm, das jedem Ereignis (Eingabe: Drücken Knopf A, B, A+B und Schütteln) eine Reaktion (Ausgabe: LED-Feld) zuordnet.

> Zum Spielen mit dem entstandenen Quiz-Buzzer sollte die Verbindung zwischen Calliope mini und Computer/Laptop über das Micro-USB-Kabel gelöst und das Batteriepack an den Calliope mini angeschlossen werden.

[19] Stiftung Haus der kleinen Forscher, ed. (2018): *Frühe informatische Bildung – Ziele und Gelingensbedingungen für den Elementar-und Primarbereich.* Verlag Barbara Budrich. Online verfügbar (Zugriff am 01.11.2023) unter https://www.haus-der-kleinen-forscher.de/fileadmin/Redaktion/4_Ueber_Uns/Evaluation/Wissenschaftliche_Schriftenreihe_aktualisiert/180925_E-Book_Band_9_final.pdf

CC BY SA

Aufbauend auf den Herausforderungen 1 – 4 werden im Projekt Quiz-Buzzer folgende Programmierblöcke eingeführt:

Kategorie	**Block**	**Funktion**
Eingabe		Schütteln führt zu Ausgabe
Grundlagen		Löschen des Bildschirm-inhalts (Ausgabe über LED-Feld)

Projekt Küchenmusik

Im Projekt Küchenmusik basteln die Schüler*innen aus dem Calliope mini ein Musikinstrument aus Alltagsgegenständen wie Löffeln, Gabeln, etc. Jeder Gegenstand soll einen eigenen Ton oder sogar eine eigene Melodie erzeugen. Der Calliope mini soll bemerken, dass ein Gegenstand berührt wird (Eingabe), und dann einen Ton abspielen (Ausgabe). Dies wird durch einen geschlossenen Stromkreis erreicht.

Die Schüler*innen entwickeln ein Programm, das jedem Ereignis (Eingabe: Pin gedrückt) eine Reaktion (Ausgabe: Ton) zuordnet. Sie vervollständigen ein vorgegebenes Programm durch Duplizieren und simple Veränderung von vorgegebenen Programmteilen (Sequenzen). Nach Bearbeitung dieses Projekts, können sie den Begriff *elektrischer Strom* in einfachen Worten erklären.

Im ersten Schritt verbinden die Schüler*innen die Küchenutensilien unter Verwendung von Kabeln mit Krokodilklemmen mit den Pins des Calliope mini. Die genaue Vorgehensweise ist dem Material zur Küchenmusik zu entnehmen. Im Anschluss verbinden die Schüler*innen den Calliope mini über das USB-Kabel mit dem Computer. Dann schreiben sie das Programm für ihr Instrument.

CC BY SA

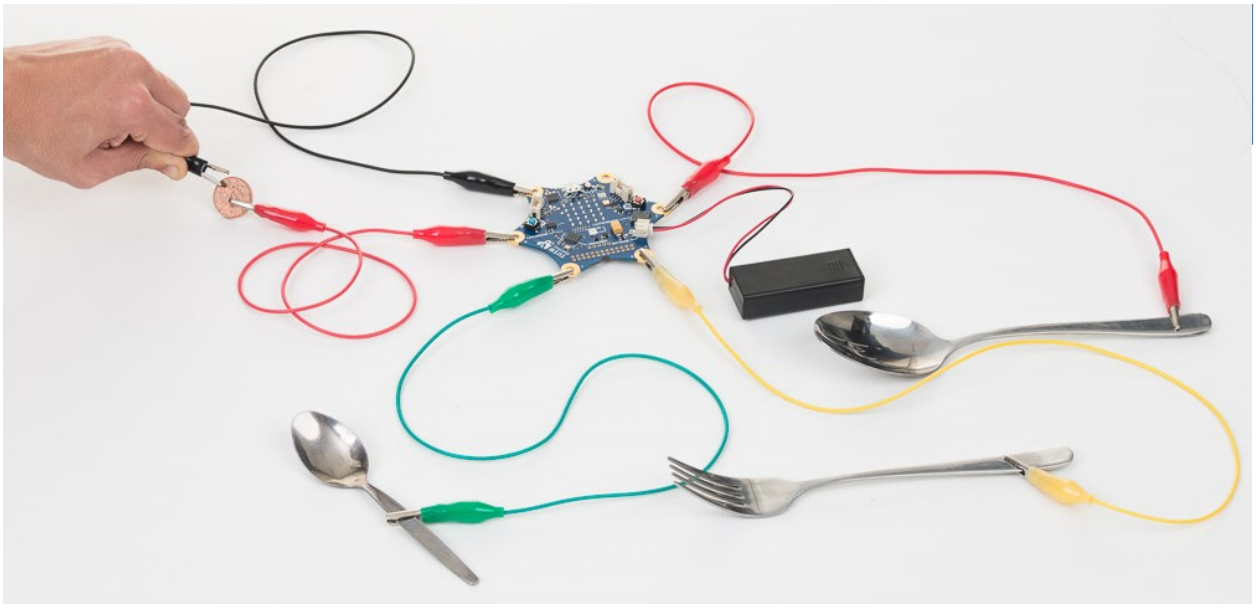
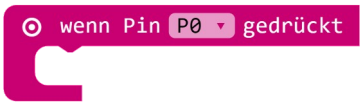
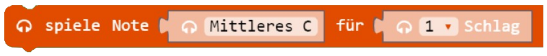


Abb. 9: Schließen des Stromkreises durch Antippen des Cents an Pin 0 mit dem Kabel am Minuspol

Die Töne erklingen **nicht** durch das bloße Drücken der Pins. Die Töne erklingen nur durch das Schließen des Stromkreises, also der Schaffung einer Verbindung (bspw. Berührung mit der Krokodilklemme) zwischen Minuspol und den jeweiligen Pins.

Aufbauend auf den Herausforderungen 1 – 4 werden Im Projekt Küchenmusik folgende Programmierblöcke eingeführt:

Kategorie	**Block**	**Funktion**
Eingabe		Schließen des Stromkreises über Pin führt zu Ausgabe
Musik		Ausgeben eines Tons über Lautsprecher (Ausgabe)

Im Ausgabe-Block *spiele Note Mittleres C für 1 Schlag* ist die Auswahl anderer Töne bzw. Teile eines Taktes oder mehrerer Takte möglich. Durch das Klicken auf Mittleres C öffnet sich eine Klaviatur auf der ein anderer Ton ausgewählt werden kann.

CC BY SA

Projekt Nachtlichter

Das Projekt Nachtlichter benötigt die meisten zusätzlichen Materialien und widmet sich stark den Themen Stromkreis und Elektrizität. Der Programmieranteil ist im Vergleich zu anderen Projekten wie *Sprungzähler* oder *Heiße Kartoffel* gering. Thematisch schließt das Projekt an Smart Home an. Das Endziel des Projekts ist es, dass kleine LEDs in Papphäusern leuchten, sobald es dunkel wird. Gemessen wird die Lichtstärke (hell bzw. dunkel)über den Helligkeitssensor des Calliope mini.

In diesem Projekt erkennen die Schüler*innen, dass die Steuerung von (programmierbarer) Technik wie Lampen abhängig von Sensordaten sein kann. Sie entwickeln ein Programm, das Sensordaten zur Steuerung nutzt und entwickeln so eine grobe Idee von der dahinterliegenden Funktionsweise. Die Schüler*innen unterscheiden den Plus- und Minuspol von Batterien und anderen Stromquellen. Sie schließen Stromkreise mit Verbrauchern (LEDs). Sie verwenden eine wenn-dann-Verzweigung (hell-dunkel) in ihrem Programm und vergleichen die Werte eines Sensors mit einem vorgegebenen Schwellenwert. Die Schüler*innen vervollständigen ein vorgegebenes Programm durch Duplizieren und simple Veränderung von vorgegebenen Programmteilen (Sequenzen). Sie testen das entwickelte und auf den Calliope mini übertragene Programm durch Veränderung der Messwerte (Abdecken des Lichtsensors).

Erläuterung auf den Arbeitsblättern (Zielgruppe: Schüler*innen):

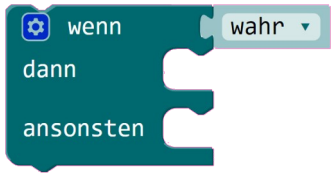
WAS IST EIN SENSOR?

Sensoren könnt ihr euch wie Sinnesorgane (Ohren, Augen etc.) vorstellen. Mit Sensoren können Geräte ihre Umwelt „fühlen“. Sie machen das zum Beispiel, indem sie messen, wie hell oder dunkel es um sie herum ist.

Im ersten Schritt testen die Schüler*innen die LED-Lampen und bilden einen minimalen Stromkreis mit einer Knopfzelle. Im nächsten Schritt erzeugen sie mithilfe des Kupferklebebands einen Stromkreis, in Verlängerung der LED-Lampen-Beinchen, mit Calliope mini im Mittelpunkt. Die Papphäuser, die über den LED-Lampen aufgestellt werden, simulieren ein kleines Dorf, in dem bei zunehmender Dunkelheit die Lichter angeschaltet werden.

CC BY SA

Aufbauend auf den Herausforderungen 1 – 4 werden im Projekt Nachtlichter folgende Programmierblöcke eingeführt:

Kategorie	Block	Funktion
Logik		Bedingung → wenn ein Wert wahr ist, dann startet der erste Block von Befehlen (dann), andernfalls startet der zweite Block an Befehlen (ansonsten); Verzweigung
		Abgleich zwischen zwei Werten
Eingabe		Nutzung des Lichtsensor-werts als Eingabe
Fortgeschritten → Pins		Nutzung der Pins zur Ausgabe

> Der digitale Wert eines Pins kann durch das Klicken auf den aktuell ausgewählten Zustand geändert werden. Es existieren die Zustände 0 = aus und 1 = an.

Projekt Sprungzähler

Das Projekt Sprungzähler beschäftigt sich, wie das Projekt Nachtlichter, mit Einsatz-möglichkeiten von Sensordaten. Es geht hier um Bewegungs-Sensoren, die man in Fitnessarmbändern und Smartphones finden kann. Ziel des Projekts ist, ein Programm zu schreiben, das die Anzahl von Sprüngen (wenn die vertikale Beschleunigung höher als die Werte für Gehen oder Laufen ist) in einer Variable speichert und ausgibt.

Bei der Bearbeitung des Projekts erkennen die Schüler*innen, dass sie selbst Sensoren bei sich tragen (Smartphone). Diese können unterschiedliche Bewegungsarten registrieren. Die Schüler*innen verwenden eine wenn-dann-Verzweigung (wenn Beschleunigung höher als vorgegeben) in ihrem Programm und vergleichen die Werte eines Sensors mit einem vorgegebenen Schwellenwert. Sie lernen das Konzept von Variablen kennen und wenden dieses an, um die Anzahl ihrer Sprünge zu zählen. Sie entwickeln ein Programm, das den Wert einer Variable ausgibt (Ausgabe auf dem LED-Feld), wenn eine Eingabe getätigt wird (Drücken eines Knopfs).

CC BY SA

Erläuterung auf den Arbeitsblättern (Zielgruppe: Schüler*innen):

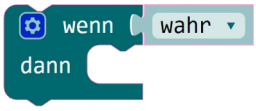
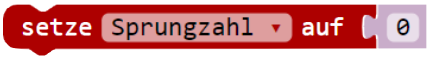
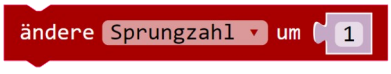
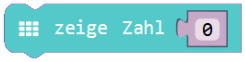
WAS IST EINE VARIABLE?

Eine Variable könnt ihr euch wie eine Kiste vorstellen, in die ihr euer Spielzeug legt. In der Computersprache nennt man das speichern.
Die Kiste bekommt einen Namen, damit man das, was man hinein legt, auch später noch wiederfindet. Bei uns ist das der Messwert mit der Anzahl der Sprünge. Der Calliope mini merkt sich diesen und findet ihn unter diesem Namen auch immer wieder.

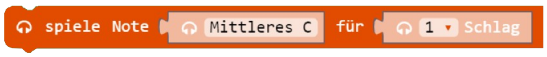
LITERATURHINWEIS

Genauere Informationen zum Thema Variablen finden sich
… im Buch **Das Calliope-Buch. Spannende Bastelprojekte mit dem Calliope-Mini-Board ab Seite 78**. [18]

Aufbauend auf den Herausforderungen 1 – 4 werden im Projekt Sprungzähler folgende Programmierblöcke eingeführt:

Kategorie	Block	Funktion
Logik		Bedingung → wenn ein Wert wahr ist, werden die Befehle im Block (dann) ausgeführt; Verzweigung
		Abgleich zwischen zwei Werten
Eingabe		Nutzung des Lagewerts als Eingabe
Variablen		Variable wird mit der Eingabe belegt
		Änderung der Variable um dem angegebenen Wert (hier: 1)
Grundlagen		Anzeigen einer Zahl auf dem LED-Feld (Ausgabe)

CC BY SA

Musik		Ausgabe eines Tons über Lautsprecher (Ausgabe)

Projekt Funk

Dieses Projekt befasst sich mit Funk. Mit Hilfe von Funk können Geräte auch ohne Kabel miteinander kommunizieren. So auch der Calliope mini. Er kann zum Beispiel Zahlen oder Texte an einen anderen Calliope mini verschicken.

Bei der Bearbeitung des Projekts lernen die Schüler*innen die Funktionsweise der Datenübertragung per Funk kennen. Sie erkennen, dass Funk in ihrer Lebenswelt an vielen Stellen zum Einsatz kommt, wie bei Walkie Talkies, Handys, Tablets oder Laptops. Nach der Bearbeitung des Projekts können die Schüler*innen die Funktionsweise von Funk mit einfachen Worten erklären und Beispiele für Funk in ihrer Lebenswelt benennen. Sie erproben, dass man (im Beispiel Calliope mini über Funkgruppen) festlegen kann und muss, welche Geräte Nachrichten miteinander austauschen können. Die Schüler*innen lernen das Prinzip und die Anwendung der Variablen als Kiste kennen: Daten können dort abgelegt und aufbewahrt werden, um sie später wieder abzurufen.

Erläuterung auf den Arbeitsblättern (Zielgruppe: Schüler*innen):

WAS IST FUNK?

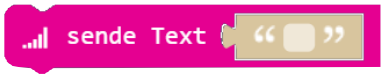
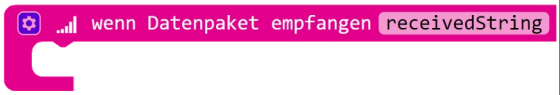
Funk bedeutet, dass Signale oder Nachrichten ohne Kabel übertragen werden können. Die Signale werden also nicht über ein Kabel, sondern unsichtbar über Funkwellen verschickt.

Funk wird an vielen verschiedenen Stellen genutzt, zum Beispiel beim Fernsehen, beim Radio, bei Walkie Talkies, Handys oder Smartphones.

Das Programm das hier entsteht, muss auf allen Calliope minis sein, an die gesendet und von denen empfangen werden soll. So können sich mehrere Calliope mini-Teams gegenseitig Nachrichten schicken.
Entscheidend ist, dass die Calliope mini-Teams, die sich Nachrichten schicken möchten, die **gleiche Funkgruppe** festlegen.

CC BY SA

Aufbauend auf den Herausforderungen 1 – 4 werden im Projekt Funk folgende Programmierblöcke eingeführt:

Kategorie	**Block**	**Funktion**
Funk		Festlegung der Funkgruppe
		Übertragung einer Zeichenfolge über Funk an jeden angeschlossenen Calliope mini in der Funkgruppe
Eingabe		Registrierung und Ausführung, wenn Datenpaket empfangen wird
Variablen		Variable, die empfangene Nachricht beinhaltet

Projekt Heiße Kartoffel

Wie durch die Kennzeichnung mit drei Kronen verdeutlicht, hat das Projekt Heiße Kartoffel einen hohen Schwierigkeitsgrad. Das im Projekt entstehende Programm ist recht umfangreich und enthält neben Variablen auch Logik-Elemente.

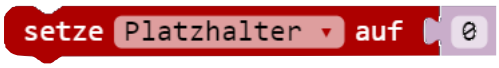
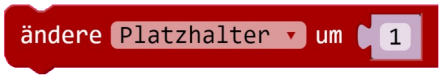
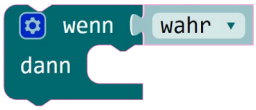
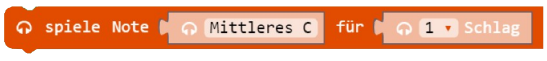
In diesem Projekt entwickeln die Schüler*innen das Spiel Heiße Kartoffel mit dem Calliope mini. Beim Spiel Heiße Kartoffel wird *die Kartoffel* von einem zum nächsten Mitspielenden weitergereicht solange die Musik läuft. Wenn die Musik gestoppt wird, hat der Mitspielende verloren, der *die Kartoffel* gerade hält. Bei der Umsetzung des Spiels mit dem Calliope mini wird der Calliope mini solange von den Mitspielenden weitergegeben, bis ein Countdown heruntergezählt ist und der Calliope mini einen Piepton von sich gibt. Dann endet das Spiel und das Kind, das den Calliope mini gerade in der Hand halt, hat verloren. In der Zusatzaufgabe schreiben die Schüler*innen das Programm so um, dass der Calliope mini, statt nach einer festen Zeit, nach einer zufälligen Zeit piepst. Das macht das Spiel noch spannender.

Die Schüler*innen lernen in diesem Projekt die Möglichkeit des Einsatzes einer zufälligen Zahl (zum Beispiel zur Generierung eines im vorgegebenem Rahmen beliebigen Countdowns) kennen und wenden diese an. Sie nutzen eine Wenn-dann-Verzweigung und verknüpfen diese mit einer Ausgabe (über den Lautsprecher). Die Schüler*innen arbeiten mit Variablen und nutzen diese zum Festlegen eines Werts (Countdown) für den Abgleich mit einer größer werdenden Variable. Sie wenden das Konzept der Iteration an, indem sie

CC BY SA

das Spiel erproben, Verbesserungswünsche benennen, überlegen, wie man diese im Programm umsetzen und anschließend selbst integrieren kann.

Aufbauend auf den Herausforderungen 1 – 4 werden im Projekt Heiße Kartoffel die folgenden Programmierblöcke eingeführt:

Kategorie	**Block**	**Funktion**
Variablen		Variable wird mit der Eingabe belegt
		Änderung der Variable um dem angegebenen Wert (hier: 1)
		Ausgeben des Werts dieser Variable
Grundlagen		Anzeigen der Zahl auf dem LED-Feld (Ausgabe)
		Pause für die angegebene Zeit
Logik		Bedingung → wenn ein Wert wahr ist, werden die Befehle im Block (dann) ausgeführt
		Abgleich zwischen zwei Werten
Mathematik		Ausgeben einer pseudozufälligen Zahl im angegebenen Bereich
Musik		Ausgabe eines Tons über Lautsprecher (Ausgabe)

CC BY SA

Projekt Rechenhelfer

Auch das Projekt Rechenhelfer ist mit drei Kronen gekennzeichnet und hat einen hohen Schwierigkeitsgrad. Das im Projekt entstehende Programm ist komplex und enthält sowohl Variablen als auch Logik-Elemente.

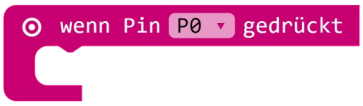
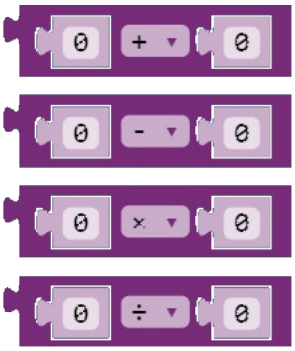
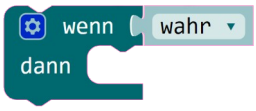
In diesem Projekt programmieren die Schüler*innen einen simplen Taschenrechner. Themen des Projekts sind die Ein- und Ausgabemöglichkeiten des Calliope mini, der Stromkreis, die Grundrechenarten (+, -, x, :) sowie das Arbeiten mit Variablen.

Die Schüler*innen lernen in diesem Projekt, dass vielen Alltagsgegenständen wie beispielsweise Taschenrechnern ein Design- und Programmierprozess zugrunde liegt. Im ersten Teil beschäftigen sie sich mit den Eingabemöglichkeiten des Calliope mini und finden einen Weg Zahlen einzugeben. Sie arbeiten mit Variablen, um die Eingabe zu speichern und anschließend mit den Zahlen rechnen zu können. Aufgrund fehlender Zahlentasten muss die Eingabeoption anders moduliert werden. Die Schüler*innen schulen dabei neben mathematischen Fähigkeiten auch Problemlösekompetenzen. Des Weiteren wird nach einer Eingabeoption für die Rechenoperationen gesucht, welche in den Pins des Calliope mini gefunden wird. Die Schüler*innen lernen den Stromkreis als Grundlage der Funktionsweise der Pins kennen.

Im zweiten Teil des Projekts programmieren die Schüler*innen die vier Grundrechenarten. Sie starten mit der Addition. Dazu lernen sie die Programmierblöcke aus dem Abschnitt *Mathematik* kennen. Die Schülerinnen und Schüler nutzen auch hier ihr mathematisches Verständnis, um die einzelnen Rechenoperationen einzupflegen. In der Subtraktion rufen sie sich in Gedächtnis, dass die Reihenfolge in der die Zahlen verrechnet werden entscheidend ist. Sie erkennen, dass in der Bildungseinrichtung vermittelte Wissen in der realen Welt praktische Anwendung findet. Zur Umsetzung der Multiplikation übertragen sie ihr Wissen aus den Programmiererfahrungen der bereits durchgeführten Addition und Subtraktion. In der Zusatzaufgabe des Projekts wird das bereits gelernte Wissen vertieft, indem eine wenn-dann-Verzweigung eingebaut wird, um die Division durch Null auszuschließen. Abschließend kann man die Schüler*innen selbst die Rechenweise des Calliope mini-Rechenhelfers erforschen lassen, indem sie mit verschiedenen Zahlen experimentieren. Dabei können die Schüler*innen das kaufmännische Runden kennenlernen. Ein weiterer Aha-Effekt ist hier die Wichtigkeit des Hinterfragens und Interpretierens von Ergebnissen, die ein Informatiksystem ausgibt. Ein entscheidender Arbeitsschritt von Programmierer*innen ist es, zu überprüfen, ob das entwickelte Programm auch das macht, was es soll und es evtl. weiterzuentwickeln bzw. zu überarbeiten bis es den Anforderungen standhält.

CC BY SA

Aufbauend auf den Herausforderungen 1 – 4 werden im Projekt Rechenhelfer folgende Programmierblöcke eingeführt:

Kategorie	**Block**	**Funktion**
Grundlagen		Anzeigen der Zahl auf dem LED-Feld (Ausgabe)
Eingabe		Schließen des Stromkreises über Pin führt zu Ausgabe
Variablen		Änderung der Variable um dem angegebenen Wert (hier: 1)
		Ausgeben des Werts dieser Variable
Mathematik		Verrechnen zweier Zahlen mit den Grundrechenarten Addition/Subtraktion/Multiplikation/Division. Anstelle der konkreten Zahlen können auch Variablen eingesetzt werden.
Logik		Bedingung → wenn ein Wert wahr ist, werden die Befehle im Block (dann) ausgeführt
		Abgleich zwischen zwei Werten

CC BY SA

Literaturverzeichnis

[1] Calliope gGmbH (2020): *DISPLAY, HACKEN, CODEN Erklärungen zu einigen englischen Begriffen … und Glossar*. Online verfügbar (Zugriff am 01.11.2023) unter https://calliope.cc/begriffe

[2] Gesellschaft für Informatik e.V. (2019): *Kompetenzen für informatische Bildung im Primarbereich.* Beilage zu LOG IN, Heft Nr. 191/192. Online verfügbar (Zugriff am 01.11.2023) unter https://dl.gi.de/bitstream/handle/20.500.12116/20121/61-GI-Empfehlung_Kompetenzen_informatische_Bildung_Primarbereich.pdf?sequence=1&isAllowed=y

[3] Kind-Wissen-Zukunft e.V. (2019): *Calliope mini. Ein Handbuch für Grundschulen.* In enger Zusammenarbeit mit Calliope gGmbH. Online verfügbar (Zugriff am 01.11.2023) unter https://calliope.cc/media/pages/schulen/schulmaterial-2020/86f4341bc8-1567083343/kiwiz_190110_calliope-handbuch.pdf

[4] Calliope gGmbH (2020): Calliope mini – Informationen zum Einstieg. Online verfügbar (Zugriff am 01.11.2023) unter https://calliope.cc/los-geht-s/einfuehrung

[5] Calliope gGmbH (2020): *Ein Microcontroller mit vielfältigen Möglichkeiten!* Diese Bauteile findest du direkt auf dem Calliope mini. Online verfügbar (Zugriff am 01.11.2023) unter https://calliope.cc/calliope-mini/uebersicht

[6] Calliope gGmbH (2020): *Poster CALLIOPE: HALLO, CALLIOPE MINI*. Online verfügbar (Zugriff am 01.11.2023) unter: https://calliope.cc/media/pages/los-geht-s/einfuehrung/1cb8f6d6c8-1581542438/calliopemini_poster_hardware.pdf

[7] Calliope gGmbH (2020): *Anleitung-2.* Online verfügbar (Zugriff am 01.11.2023) unter https://calliope.cc/projekte/anleitung-2

[8] Kultusministerkonferenz (2016): *Strategie der Kultusministerkonferenz „Bildung in der digitalen Welt“*. Online verfügbar (Zugriff am 01.11.2023) unter https://www.kmk.org/fileadmin/Dateien/pdf/PresseUndAktuelles/2017/Strategie_neu_2017_datum_1.pdf

[9] Freistaat Sachsen, Staatsministerium für Kultus (aktualisiert 2019): *Lehrplan Grundschule Werken*. Seite 14. Online verfügbar (Zugriff am 01.11.2023) unter https://www.schulportal.sachsen.de/portal/getFile.php?lplanid=73&lplansc=5qcMpegw6pQpLsPppKnZ&aktion=lplanfile&token=44051aa1b1c0577b32b61c74f2d24d29

[10] Schubert, Sigrid und Andreas Schwill (2011): *Didaktik der Informatik*. Heidelberg: Spektrum.

[11] Baumann, Rüdeger (1996): *Didaktik der Informatik*. Stuttgart: Klett.

[12] Bers, Umaschi Marina et al. (2014): *Computational thinking and tinkering: Exploration of early childhood robotics curriculum*. Computers and Education 72, S. 145-157.

CC BY SA

[13] Bundeszentrale für politische Bildung/bpb (2019): *einfach Politik: Smartphone und andere Computer.* Online verfügbar (Zugriff am 01.11.2023) unter http://www.bpb.de/shop/lernen/einfach-politik/300205/smartphone-und-andere-computer

[14] Stiftung „Haus der kleinen Forscher“ (2017): *Informatik entdecken – mit und ohne Computer.* Berlin, 1. Auflage. Seite 19. Online verfügbar (Zugriff am 01.11.2023) unter https://www.stiftung-kinder-forschen.de/fileadmin/Redaktion/1_Forschen/Themen-Broschueren/Broschuere_Informatik_2017.pdf

[15] Calliope gGmbH (2020): *Poster CALLIOPE: EINGABE UND AUSGABE*. Online verfügbar (Zugriff am 01.11.2023) unter: https://calliope.cc/calliope-mini/uebersicht

[16] N. Bergner, P. Franken, J. Kleeberger, T. Leonhardt, M. Lukas (2017): *Das Calliope-Buch. Spannende Bastelprojekte mit dem Calliope-Mini-Board*. Dpunkt.verlag. Online verfügbar (Zugriff am 01.11.2023) unter https://dpunkt.de/produkt/das-calliope-buch/.

[17] Schroeder, Ulrik und Nadine Bergner (2015): *Informatik Enlightened - Informatik (neu) beleuchtet dank Physical Computing mit Arduino.* INFOS 2015, 16. GI-Fachtagung Informatik und Schule, S. 43–52.

[18] Stiftung Haus der kleinen Forscher, ed. (2018): *Frühe informatische Bildung – Ziele und Gelingensbedingungen für den Elementar- und Primarbereich.* Verlag Barbara Budrich. Online verfügbar (Zugriff am 01.11.2023) unter https://www.haus-der-kleinen-forscher.de/fileadmin/Redaktion/4_Ueber_Uns/Evaluation/Wissenschaftliche_Schriftenreihe_aktualisiert/180925_E-Book_Band_9_final.pdf

CC BY SA