

Modul 1

Was ist Programmieren?

Lehrerhandreichung zur Bremer
Explorationsstudie 'Calliope mini'

LEHRERHANDREICHUNG ZUR BREMER EXPLO- RATIONSSTUDIE *CALLIOPE MINI*

MODUL 1

WAS IST PROGRAMMIEREN?

Autor*innen:

Iris Bockermann, Simon Engelbertz, Saskia Illginnis, Antje Moebus, Lydia Murmann, David A. Reid, Heidi Schelhowe

Deckblattgestaltung:

Claire Granereau

In Zusammenarbeit mit dem Landesinstitut für Schule (LIS) Bremen

2. Version, Oktober 2018



Ermöglicht durch die Deutsche Telekom Stiftung



DANKSAGUNG

An dieser Stelle möchten wir uns beim Landesinstitut für Schule Bremen für die gute Kooperation bedanken. Vor allem gilt unser Dank auch den beteiligten Lehrkräften für ihr großes Engagement, für die Durchführung der Unterrichtseinheiten an den Schulen und für die Unterstützung bei der wissenschaftlichen Begleitung. Die Offenheit der Schüler*innen wie auch deren Eltern hat uns die Untersuchung ermöglicht und bildete für uns die Basis.

INHALTSVERZEICHNIS

I. Abbildungsverzeichnis	5
II. Abkürzungen.....	5
III. Vorwort/Vorbemerkung.....	6
IV. Über diese Handreichung	8
1. Über sowohl informatische Bildung als auch Digitalisierung im Primarbereich	9
2. Der Calliope mini – erste Schritte	11
2.1 Einleitung/Hintergrundgeschichte	11
2.2 Aufbau und Funktionsweise des Calliope Mini	11
2.2.1 Voraussetzungen.....	11
2.2.2 Die Bauteile	12
2.2.3 Die LED-Anzeige (Aktoren, Ausgabe)	13
2.2.4 Die RGB-LED (Aktoren, Ausgabe)	13
2.2.5 Knopf A und Knopf B (Eingabe).....	13
2.2.6 Der RESET-Knopf (Eingabe).....	13
2.2.7 Die Sensoren (Eingabe)	14
2.2.8 Die Ecken/Pins (Eingabe).....	14
2.2.9 Der USB-Anschluss.....	14
2.2.10 Bluetooth.....	15
2.2.11 Der Lautsprecher (Aktoren, Ausgabe).....	15
2.2.12 Der Prozessor (Verarbeitung)	15
2.2.13 Die Grove-Connectoren.....	15
2.2.14 Der Motoranschluss	15
2.3 Die Inbetriebnahme des Calliope mini.....	15
2.3.1 Vorbereitungen	16

2.3.2	Die Editoren/Programmierungsumgebungen	16
2.3.3	Die Programmierung des Calliope mini	17
2.3.4	Die Versorgung des Calliope mini mit Strom	17
3.	Die Unterrichtseinheiten	19
3.1	„Was ist Programmieren?“ (Teil 1 bis Teil 3).....	24
3.1.1	Kurze Beschreibung der Einheit	24
3.1.2	Empfehlungen/Tipps/Praxisbeispiel(e).....	26
4.	Teil 1: Was ist Programmieren? Algorithmus, Programm, Programmieren	28
4.1	Zeitlicher und inhaltlicher Aufbau	29
4.2	Arbeitsblätter – Teil 1.....	31
5.	Teil 2: Was ist Programmieren? Mikrocontroller, E-V-A-Prinzip.....	40
5.1	zeitlicher und inhaltlicher Aufbau	40
5.2	Arbeitsblätter –Teil 2.....	42
6.	Teil 3: Was ist Programmieren? Speichern, Ordner, Datei	47
6.1	zeitlicher und inhaltlicher Aufbau	47
6.2	Arbeitsblätter –Teil 3.....	49
7.	Weiterführende Literatur	68
7.1	Internetseiten.....	68
7.2	Bücher	70

I. ABBILDUNGSVERZEICHNIS

Abbildung 1: Übersicht Calliope-Arbeitsblatt (Lösung)	12
Abbildung 2: Übersicht über die Unterrichtseinheiten	19
Abbildung 3: laminierte Befehlsblöcke	23
Abbildung 4: zwei exemplarische Abfolgen von Befehlsblöcke.....	23
Abbildung 5: Einblick in den Unterricht zum Thema Mikrocontroller.....	26
Abbildung 6: Einblick in den Unterricht zum Thema Algorithmus - Tafelbild	27
Abbildung 7: Einblick aus dem Unterricht zum Thema Algorithmus - exemplarisches ausgefülltes Arbeitsblatt.....	27

II. ABKÜRZUNGEN

Die Unterrichtsmaterialien bestehen pro Unterrichtseinheit jeweils aus einer Übersicht, dem Unterrichtsverlauf und den Arbeitsblättern. Folgende Abkürzungen sind aufgeführt:

AN = Anleitung

ÜB = Übungsblatt

ÜA = Übungsaufgabe

SP = Spiel

VÜ = Vorüberlegung

WB = Wissensbit

III. VORWORT/VORBEMERKUNG

Der Mikrocontroller Calliope mini wurde der breiten Öffentlichkeit auf dem IT-Gipfel 2016 in Saarbrücken vorgestellt. Er wurde für Grundschulkinder und deren Hände entwickelt und soll einen möglichst unkomplizierten Zugang zu ersten eigenen Programmiererfahrungen bieten. Das Besondere: Die Kinder programmieren nicht etwa Effekte, die sie ausschließlich an einem Computerbildschirm beobachten oder hören können. Sie benötigen auch keinen Bausatz, bei dem sie zunächst allerlei Einzelteile zusammenstecken müssen. Stattdessen programmieren sie ein kleines Gerät, den Mikrocontroller Calliope mini, der Sensoren hat, mit denen er einiges erfassen kann, und Aktoren, mit denen er einiges bewirken kann. Der Mikrocontroller ist zudem klein, kann herumgetragen und in größere Objekte eingebaut werden.

Mit der Entwicklung des Calliope mini war die Absicht verbunden, diesen Mikrocontroller bundesweit allen Grundschulen zur Verfügung zu stellen und Kindern flächendeckend die Möglichkeit zu bieten, schon sehr früh – in der Grundschule – zu erfahren, was durch eigenes Programmieren bewirkt werden kann.

Vorschläge für solche überschaubaren Programmiervorhaben lagen vor, allerdings fehlten Erfahrungen damit, wie Grundschullehrer*innen diese – für sie neuen – Inhalte im regulären Schulalltag aufgreifen und einbinden können. Was für ein Vorhaben, mit mehr als zwanzig Kindern gemeinsam zu programmieren! Zumal die Schüler*innen und Schüler recht unterschiedliche Erfahrungen mit digitalen Geräten haben, so dass sie sehr heterogene Lernvoraussetzungen mitbringen, und die wenigen Grundschullehrer*innen über eigene Programmiererfahrung verfügen.

Die Deutsche Telekom Stiftung hat daher im Rahmen ihres Projektes „Digitales Lernen Grundschule“ eine einjährige Explorationsstudie gefördert, die im Schuljahr 2017/2018 in Bremen umgesetzt und vom Zentrum für Medien am Landesinstitut für Schule (LIS) unterstützt wurde.

Drei Lehrer*innen – und ihre Schüler*innen – an drei sehr verschiedenen Schulen haben sich auf den Weg gemacht, Unterricht erprobt und über ihn reflektiert. Die Konzeption, Weiterentwicklung und Dokumentation der Unterrichtssequenzen, die Begleitung und Durchführung von Lehrer*innen- und Schüler*innen-Auswertungsgesprächen sowie die Datenauswertung erfolgten durch die Universität Bremen.

Im Rahmen des Projektes kooperierten Fachdidaktiker*innen für Mathematik (David A. Reid) und Interdisziplinäre Sachbildung/Sachunterricht (Lydia Murmann) mit der Arbeitsgruppe Digitale Medien in der Bildung (Heidi Schelhowe und Iris Bockermann). Die im Kontext dieser Studie entwickelten, erprobten und überarbeiteten Unterrichtseinheiten sind in der vorliegenden Handreichung zusammengestellt.

Besonderer Dank gebührt Antje Moebus, Saskia Illginnis und Simon Engelbertz, die mit großem Engagement sowohl die konkrete Umsetzung der Studie als auch die Erarbeitung dieser Handreichung unternommen haben.

Bremen, Oktober 2018

Lydia Murmann

IV. ÜBER DIESE HANDREICHUNG

Was ist ein Mikrocontroller? Was hat er mit der Technik in meinen alltäglich genutzten Gegenständen zu tun? Diesen Fragen gehen wir in dieser Handreichung unter anderem nach. Sie soll eine praktische Hilfestellung für den Einstieg in die informatische Grundbildung im Primarbereich liefern und damit einhergehend das Verständnis der Schüler*innen in Bezug auf ihre digitale Lebenswelt fördern.

Neben einer theoretischen Rahmung werden die ersten Schritte der Inbetriebnahme des Mikrocontrollers Calliope mini im Modul 1 gezeigt sowie praktische Arbeitsmaterialien für den Unterricht bereitgestellt. So kann der Mikrocontroller mit neuen grundschuldidaktischen Unterrichtskonzepten im Unterricht eingesetzt werden.

Diese Konzepte wurden bereits in Grundschulen getestet und modifiziert, so dass neben der Beschreibung der verschiedenen Unterrichtseinheiten auch Praxisbeispiele und Tipps bereitgestellt werden. Die Arbeitsmaterialien lassen sich beispielhaft in die Fächer ‚Sachunterricht‘ oder ‚Mathematik‘ bzw. in die fächerübergreifende Medienbildung einbetten, können darüber hinaus aber auch flexibel in anderen Grundschulfächern, wie zum Beispiel Kunst oder Sport, eingesetzt werden.

1. ÜBER SOWOHL INFORMATISCHE BILDUNG ALS AUCH DIGITALISIERUNG IM PRIMARBEREICH

Teilhabe an digitaler Kultur ist an Voraussetzungen geknüpft – zum Beispiel an die Verfügbarkeit der Technologie und der Geräte, aber auch an die Fähigkeit, sie nutzen und einschätzen zu können. Die sogenannten Digital Natives, die mit digitaler Alltagstechnik aufwachsen, erschließen sich oft leicht und informell einige Nutzungskompetenzen. Ein Grundverständnis der grundlegenden Funktionsweisen kann durch reine Nutzung und durch beiläufiges Lernen kaum gefördert werden.

Darauf zielt die Förderung des algorithmischen Denkens und des forschenden Lernens sowie die Heranführung an grundlegende informatische Konzepte bereits im Primarbereich ab. Diverse nationale und internationale Studien bestätigen, dass Schüler*innen im Primarbereich bereits grundlegende informatische Konzepte nachvollziehen und anwenden können.

Bei der Vermittlung informatischer Grundkonzepte geht es aber um viel mehr als nur um kognitive Leistungen, nämlich darum,

- einen Lebensweltbezug herzustellen und die Begegnung mit Informatikphänomenen aus der digitalen Lebenswelt der Schüler*innen auch im Schulkontext zu ermöglichen.
- die Schüler*innen für Informatikphänomene und elementare informatische Methoden und Werkzeuge zu begeistern und zu sensibilisieren, bevor sie jegliches Interesse daran und Selbstvertrauen verlieren, was häufig in der Übergangsphase beim Wechsel auf die weiterführenden Schulen passiert.

Dies bringt auch das folgende Zitat aus der Broschüre „Informatik entdecken – mit und ohne Computer“ (2018) der Stiftung Haus der kleinen Forscher zum Ausdruck:

Die Beherrschung elementarer informatischer Methoden und Werkzeuge ist damit auf dem besten Weg, neben Schreiben, Lesen und Rechnen zur

vierten Kulturtechnik zu werden. Damit wird informatische Bildung zu einer gesellschaftlichen Aufgabe und sollte zukünftig ein fester Bestandteil einer grundlegenden Allgemeinbildung sein.

Wesentliche Grundlage digitaler Geräte ist ihre Programmiertheit. Was bedeutet es, dass ein Gerät programmiert ist? Dies erfährt man am ehesten dadurch, dass man selbst Einblicke ins Programmieren gewinnt.

Hierzu wurden Konzepte für die Grundschule erstellt. Erste Schritte in der Programmierung können handlungsorientiert und begreifbar mit einem Mikrocontroller erprobt werden. Mittlerweile sind verschiedene Mikrocontroller auf dem Markt, die für verschiedene Alters- und Schulstufen geeignet sind.

Der Calliope mini ist ein Mikrocontroller, der für den Einsatz im Grundschulkontext entwickelt wurde. Dieser Mikrocontroller ist klein, handlich, ziemlich robust und lässt sich vielfältig einsetzen. Er unterscheidet sich optisch deutlich von den digitalen Geräten, die Kindern sonst begegnen, denn er sieht er aus, als sei er irgendwo ausgebaut worden. Es handelt sich um eine Platine mit diversen Anschlüssen, Knöpfen und Anzeigemöglichkeiten, die weit entfernt von einem Touchscreen sind, und er kann mit Hilfe eines weiteren Computers programmiert werden.

2. DER CALLIOPE MINI – ERSTE SCHRITTE

2.1 EINLEITUNG/HINTERGRUNDGESCHICHTE

Mit dem Calliope mini können viele kreative Ideen umgesetzt werden, zum Beispiel kann ein eigener Würfel programmiert oder es können Nachrichten verschickt werden. Mit wenigen Klicks können eigene Programme erstellt und auf den Calliope mini übertragen werden.

Der Calliope mini ist eine Platine mit vielen verschiedenen Komponenten und Funktionen. Mikrocontroller sind in vielen technischen Geräten, wie zum Beispiel in Fernsehern, Fitnessarmbändern, Autos, Ampeln etc., verbaut.

Vorreiter des Calliope mini war der BBC micro:bit, welcher 2012 von dem englischen Radio- und Fernsehsender BBC entwickelt wurde. Mit dem micro:bit werden in Großbritannien mit 11- bis 12-jährigen Kindern eigene Projekte umgesetzt.

Der Calliope mini wurde ebenfalls für Kinder als Hauptzielgruppe entwickelt. Die Editoren, mit denen der Calliope mini programmiert werden kann, sind kostenlos.

2.2 AUFBAU UND FUNKTIONSWEISE DES CALLIOPE MINI

2.2.1 VORAUSSETZUNGEN

Für die Programmierung des Mikrocontrollers Calliope mini müssen einige Voraussetzungen erfüllt sein. So werden ein Computer, ein Laptop oder ein Tablet benötigt sowie einen Zugang zum Internet.

2.2.2 DIE BAUTEILE

Der Calliope mini enthält mehrere Bauteile. Es gibt 25 rote LEDs und eine RGB-LED, wobei alle LEDs leuchten können. Ebenso gibt es zwei programmierbare Knöpfe (Knopf A und Knopf B) und einen kombinierten Lagesensor mit Bewegungssensor und Kompass. Um mit anderen Geräten kommunizieren zu können, existiert ein Bluetooth-Modul. Das Board des Calliope mini lässt sich mit dem Computer/Laptop oder auch mit einer App über das Tablet programmieren. Um die Programme von dem Computer oder Laptop zu übertragen, wird ein USB-Kabel (im Karton mit beigelegt) benötigt. Kabellos funktioniert es per Tablet und App. Im Folgenden werden die einzelnen Bauteile näher beschrieben:

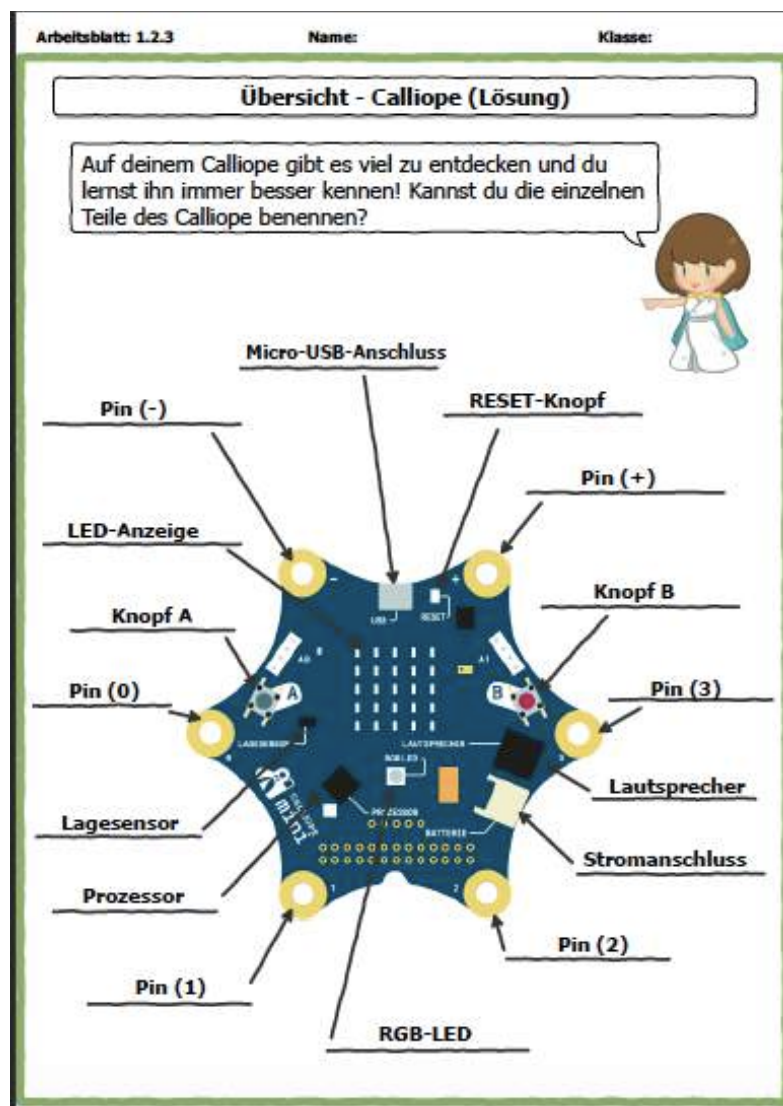


Abbildung 1: Übersicht Calliope-Arbeitsblatt (Lösung)

2.2.3 DIE LED-ANZEIGE (AKTOREN, AUSGABE)

In der Mitte des Calliope mini sind 25 LEDs (lichtaussendende Dioden) in einem 5x5-Raster angebracht, die rot aufleuchten. Die LEDs sind in einem Viereck angeordnet und können einzeln angesteuert werden. Es lassen sich aber auch vorprogrammierte Symbole, wie zum Beispiel ein Herz oder eine Giraffe, auf dem LED-Feld anzeigen.

2.2.4 DIE RGB-LED (AKTOREN, AUSGABE)

Unter dem LED-Feld liegt eine einzelne RGB-LED, welche die Darstellung weiterer Farben ermöglicht. Sie ist größer als die anderen LEDs und rund. Die Abkürzung RGB steht für Rot, Grün und Blau. Diese drei Farben sind die Grundfarben und lassen sich zu anderen Farben mischen. Die RGB-LED lässt sich beispielsweise mit dem Befehl „Setze LED-Farbe auf Rot“ programmieren.

2.2.5 KNOPF A UND KNOPF B (EINGABE)

Mit dem blauen Knopf links (Knopf A) und dem roten Knopf rechts (Knopf B) können Eingaben vorgenommen werden. Sobald ein Knopf gedrückt wurde, wird ein Schaltkreis geschlossen. Der Calliope mini kann so programmiert werden, dass er etwas tut, wie zum Beispiel, dass er einen Ton abspielt, wenn Knopf A oder Knopf B oder beide Knöpfe gemeinsam gedrückt werden. Der Befehl dafür findet sich im Editor unter „Eingabe“ und heißt zum Beispiel „Wenn Knopf A gedrückt“.

2.2.6 DER RESET-KNOPF (EINGABE)

Oben in der Mitte ist ein kleiner weißer Knopf, der mit „RESET“ beschriftet ist. Mit diesem Knopf kann der Calliope mini auf das aktuelle Programm zurückgesetzt werden bzw. neu gestartet werden.

2.2.7 DIE SENSOREN (EINGABE)

Es gibt auf dem Calliope mini außerdem Sensoren, die zum Beispiel die Temperatur, die Helligkeit, die Himmelsrichtung, die Bewegung oder seine Lage messen. Eingesetzt wurden im Unterricht folgende Sensoren:

- Der Lichtsensor: Dieser Sensor ist hinter dem LED-Feld versteckt und misst die Lichtstärke (Helligkeit). Er wird zum Programmieren der Alarmanlage benötigt.
- Der Lagesensor: Über den Lagesensor merkt der Calliope mini, in welcher Position er gehalten wird.
- Der Bewegungssensor: Hier wird gemessen, ob der Calliope zum Beispiel durch Schütteln bewegt wird.
- Der Kompass: Über den Kompass, der unter Knopf A liegt, kann der Calliope mini herausfinden, in welcher Richtung Norden ist.

2.2.8 DIE ECKEN/PINS (EINGABE)

Es gibt an dem Calliope mini sechs runde goldene Ecken/Pins, die jeweils anders beschriftet sind. Es gibt Ecken mit einem Pluszeichen (+) und Ecken mit einem Minuszeichen (-) sowie Ecken mit Ziffern (0, 1, 2 und 3).

An diese können weitere Sensoren und Aktoren, wie zum Beispiel Temperaturfühler oder ein Feuchtigkeitsmesser, angeschlossen werden. Außerdem sind die Pins 1 und 3 berührungsempfindlich.

2.2.9 DER USB-ANSCHLUSS

Über den Micro-USB-Anschluss lässt sich der Calliope mini mit dem Computer oder Laptop verbinden. So können die Programme, die zuvor im Editor erstellt wurden, auf den Calliope mini übertragen werden. Über dieses Kabel wird der Calliope mini auch mit Strom versorgt.

2.2.10 BLUETOOTH

Mittels Bluetooth können funkfähige Geräte kabellos miteinander verbunden werden. Somit kann sowohl eine kabellose Kommunikation zwischen den einzelnen Calliope-mini-Geräten stattfinden als auch eine Datenübertragung zwischen dem Tablet und dem Calliope mini. Hinweis: Für Bluetooth gibt es kein beschriftetes Bauteil auf dem Calliope mini.

2.2.11 DER LAUTSPRECHER (AKTOREN, AUSGABE)

Rechts auf dem Calliope mini ist ein viereckiger, schwarzer Kasten, welcher der integrierte Lautsprecher ist. Über diesen lassen sich Töne und Melodien abspielen.

2.2.12 DER PROZESSOR (VERARBEITUNG)

In der unteren linken Ecke befindet sich der Prozessor, welcher alle Funktionen des Calliope mini miteinander verbindet.

2.2.13 DIE GROVE-CONNECTOREN

Es können Erweiterungsmodule, die zum Beispiel Licht oder Farbe messen, über die weißen Verbindungsstecker angeschlossen werden. Diese wurden in den Unterrichtsmaterialien jedoch nicht weiter berücksichtigt.

2.2.14 DER MOTORANSCHLUSS

An den Calliope mini können zwei Motoren an den Motoranschluss angeschlossen werden. Dieser wird benötigt, um ein selbstfahrendes Auto oder einen Roboter zu steuern.

2.3 DIE INBETRIEBNAHME DES CALLIOPE MINI

Nachdem alle wichtigen Bauteile des Calliope mini beschrieben wurden, geht es nun darum, das Board selbst zu testen und zu programmieren sowie eigene Projekte damit umzusetzen.

2.3.1 VORBEREITUNGEN

Um den Calliope mini selber programmieren zu können, werden (noch) ein internetfähiges Gerät (Computer/Laptop/Tablet) und ein USB-Kabel benötigt. Letzteres, um den Calliope mini mit dem Computer/Laptop zu verbinden. Um zu den Editoren zu gelangen, muss der Internetbrowser geöffnet werden. Auf der Internetseite <https://www.calliope.cc> befinden sich unter dem Navigationspunkt „Los geht’s“ die Links zu den „Editoren“, welche im Folgenden beschrieben werden.

2.3.2 DIE EDITOREN/PROGRAMMIERUMGEBUNGEN

Unter <https://www.calliope.cc/los-geht-s/editor> werden drei mögliche Editoren für das Programmieren des Calliope mini und der dazugehörigen Verknüpfung dargestellt. Für die im Anhang bereitgestellten Unterrichtsmaterialien wurde mit dem MakeCode-Editor gearbeitet.

2.3.2.1 DER CALLIOPE MINI EDITOR



Mit dem *Calliope mini Editor* lässt es sich am einfachsten programmieren, da dieser nur einen kleinen Umfang an Programmierbefehlen aufweist. Somit lassen sich schnell Symbole und Texte programmieren und auf dem

Calliope mini sichern.

<http://miniedit.calliope.cc>

2.3.2.2 DER MAKECODE-EDITOR



Mit dem *MakeCode-Editor* lassen sich gegenüber dem *Calliope mini Editor* vielfältige Programme erfinden. Diese können

entweder durch das Anordnen von farbigen Programmier-Blöcken oder mit *JavaScript*TM geschrieben werden. <https://www.makecode.calliope.cc>

2.3.2.3 DIE PLATTFORM OPEN ROBERTA LAB[®]



Mit der grafischen Programmiersprache *NEPO*[®] intuitiv kann auf dem Open Roberta Lab[®] programmiert werden. Diese Open-Source-Plattform beruht auf der langjährigen Erfahrung der Fraunhofer-Initiative „Roberta[®] – Lernen mit Robotern“.

<https://lab.open-roberta.org>

2.3.3 DIE PROGRAMMIERUNG DES CALLIOPE MINI

Der Calliope mini kann nun mit Hilfe der angezeigten Bausteine und dem beigefügten Material programmiert werden. Sobald ein Programmcode für ein Projekt fertig ist, kann dieser mit einem Klick auf „Herunterladen“ auf dem Computer/Laptop oder direkt auf den Calliope mini gespeichert werden. Da der Calliope mini wie auch ein USB-Stick angeschlossen wird, ist er bei Macintosh[®]-Rechnern auffindbar unter „Gehe zu“, „Computer“, „Geräte“, „MINI“ oder über den Finder als „MINI“ und am PC unter „Arbeitsplatz als Wechseldatenträger“. Wenn eine Datei auf den Calliope mini übertragen wird, blinkt die gelbe Status-LED auf.

2.3.4 DIE VERSORGUNG DES CALLIOPE MINI MIT STROM

Es gibt zwei Möglichkeiten, den Calliope mini mit Strom zu versorgen:

1. Der mitgelieferte Batterieblock wird mit dem kleinen, weißen Plastikende (JST-Stecker) an den befestigten Batterieanschluss des Calliope mini gesteckt. Der Batterieblock hat einen On-/Off-Knopf so dass dieser bei mehrfacher Benutzung eingesteckt bleiben kann.

Hinweis: Vorsicht beim Ein- und Ausstecken des Batterieblocks, damit die Kabel nicht kaputtgehen.

2. Der Calliope mini wird ebenfalls mit Strom versorgt, wenn die Verbindung zum Computer/Laptop mittels Micro-USB-Kabel besteht. Diese wird auch benötigt, um Programme auf den Calliope zu sichern. Dafür wird das kleine Ende in den USB-Anschluss des Calliope mini gesteckt und das größere Ende an den Computer/Laptop angeschlossen.

3. DIE UNTERRICHTSEINHEITEN

Es wurden für die Arbeit mit dem Calliope mini in der Grundschule exemplarisch sechs Unterrichtseinheiten ausgearbeitet. Wir empfehlen, die Unterrichtseinheit „Was ist Programmieren?“ (Modul 1) als Grundlage für weitere Unterrichtseinheiten zu nehmen.



Abbildung 2: Übersicht über die Unterrichtseinheiten

Im Anschluss können je nach Interesse oder Unterrichtsfach die anderen Unterrichtseinheiten (Module 2-6) behandelt werden. In den Unterrichtsverläufen, die es für jede Unterrichtseinheit gibt, steht jeweils, in welche Fächer sich diese einordnen lassen und wie viel Zeit einzuplanen ist. Dabei können die Unterrichtseinheiten in ihrer Tiefe und im zeitlichen Aufwand variiert werden.

Im Folgenden werden die sechs Unterrichtseinheiten, die für die Explorationsstudie entwickelt wurden, vorgestellt. Da die fächerübergreifende Medienbildung nicht als eigenes Fach ausgewiesen ist, wurden die Unterrichtseinheiten nach der inhaltlichen bzw. thematischen Ausrichtung zusätzlich entweder dem Sach- oder Mathematikunterricht zugeordnet.

1. Was ist Programmieren?

Diese Einheit besteht aus insgesamt drei in sich abgeschlossenen Teilen. Ziel dieser Einheit ist es, die Schüler*innen an die grundlegenden Begriffe und Konzepte von

Informatiksystemen heranzuführen und sie mit den ersten Schritten des Programmierens vertraut zu machen. Die Einheit wurde für den Sachunterricht und die fächerübergreifende Medienbildung entwickelt. Unterrichtszeit: ca. drei Stunden.

1.1 Algorithmus, Programm und Programmieren

Im ersten Teil werden die Schüler*innen mit informatischen Begriffen, wie ‚Algorithmus‘, ‚Programm‘ und ‚Programmieren‘, vertraut gemacht. Den Begriff des Algorithmus lernen sie am Beispiel des Fieberthermometers kennen. Im Anschluss können sich die Schüler*innen Gedanken darüber machen, welche Algorithmen sie im Alltag kennen und wie zum Beispiel eine Smartwatch funktioniert.

1.2 Mikrocontroller

Im zweiten Teil lernen die Schüler*innen den Mikrocontroller kennen. Ziel ist es, die Komponenten und Funktionen des Mikrocontrollers kennenzulernen. Dies umfasst auch, dass sie den Mikrocontroller an den Computer anschließen und von ihm trennen sowie die Energieversorgung mittels Batterie herstellen können.

1.3 Datei, Ordner und Speichern

In diesem dritten Teil wird geübt, mit dem Computer/Laptop, der Tastatur, der Maus und dem Internetbrowser umzugehen. Ziel ist es vor allem, den Weg des Speicherns, also die Sicherung der Daten, kennenzulernen. Diese Unterrichtseinheit wurde im Laufe des Projektes erstellt und daher nicht in allen Lerngruppen durchgeführt. Sie wird der Vollständigkeit halber hier mit dargestellt.

2. Roboter ‚Rumpelstilzchen‘

In dieser Einheit lernen die Schüler*innen, anhand einer kleinen Geschichte Programmieraufgaben selbstständig zu lösen. Es wurden fünf Programmieraufgaben gestellt. Dabei können die Schüler*innen ihr bisheriges Wissen zur Bedienung des Editors und Mikrocontrollers Calliope mini vertiefen. Diese Einheit wurde für die fächerübergreifende Medienbildung und den Sachunterricht entwickelt. Unterrichtszeit: ca. eine Stunde. Sie wurde im Laufe des Projektes erstellt und daher nicht in allen Lerngruppen durchgeführt. Sie wird der Vollständigkeit halber hier mit dargestellt.

3. Hüpf-/Klickzähler (Thema Variablen)

Thematisch geht es beim Hüpf- oder Klickzähler um Variablen und darum, wie diese programmiert werden können. Diese Einheit wurde für das Fach ‚Mathematik‘ und die fächerübergreifende Medienbildung entwickelt. Unterrichtszeit: ca. eine bis zwei Stunden.

4. Nachrichten senden (Thema ‚Funk‘)

In dieser Einheit findet ein spielerischer und kreativer Einstieg in das Thema „Funk“ statt. Die Schüler*innen sollen in Grundzügen verstehen, wie Nachrichten kabellos gesendet und empfangen werden können. Diese Einheit wurde für das Fach ‚Sachunterricht‘ und die fächerübergreifende Medienbildung entwickelt. Unterrichtszeit: ca. zwei Stunden.

5. Zufallsgenerator-Würfel (Thema „Wahrscheinlichkeiten und Zufall“)

In dieser Einheit lernen die Schüler*innen, einen digitalen Würfel zu programmieren und zu benutzen. Dabei wird auf das Konzept von Zufallszahlen eingegangen und auf die Programmierung des digitalen Würfels übertragen. Die Einheit wurde für das Fach ‚Mathematik‘ und die fächerübergreifende Medienbildung entwickelt. Unterrichtszeit: ca. zwei Stunden.

6. Alarmanlage (Thema „Sensoren und Aktoren“)

In dieser Einheit wird eine Alarmanlage programmiert. Sie lässt sich in das Fach ‚Sachunterricht‘ und die fächerübergreifende Medienbildung einbetten und dauert ca. zwei Stunden. Ziel ist es, dass die Schüler*innen das Konzept von Sensoren und Aktoren kennenlernen und es beim Programmieren des Mikrocontrollers anwenden können. Diese Einheit wurde für das Fach ‚Sachunterricht‘ und die fächerübergreifende Medienbildung entwickelt. Unterrichtszeit: ca. zwei Stunden.

Aus unseren praktischen Erfahrungen können wir dazu raten, die Unterrichtsmaterialien farbig auszudrucken, da Schüler*innen diese besser verstehen. Zudem empfehlen wir, die einzelnen Befehlsblöcke des MakeCode-Editors farbig auszudrucken und gegebenenfalls zu laminieren. Mit diesen Befehlsblöcken können die Schüler*innen puzzeln, um den richtigen Code zu finden.



Abbildung 3: laminierte Befehlsblöcke

Besonderen Wert haben wir zudem auf den spielerischen Charakter gelegt. Das heißt, dass in jeder Unterrichtseinheit ein spielerisches Element verankert ist. Jede Unterrichtseinheit hat eine Übersicht mit dem Fachbezug, dem Ziel und der Dauer und beinhaltet Arbeitsblätter (beigefügt) sowie einen möglichen Unterrichtsverlauf.



Abbildung 4: zwei exemplarische Abfolgen von Befehlsblöcke

3.1 „WAS IST PROGRAMMIEREN?“ (TEIL 1 BIS TEIL 3)

3.1.1 KURZE BESCHREIBUNG DER EINHEIT

Diese Einheit besteht aus insgesamt drei in sich abgeschlossenen Teilen, welche ca. drei Unterrichtsstunden dauern.

Sie kann in die Unterrichtsfächer ‚Sachunterricht‘ oder die fächerübergreifende Medienbildung eingebettet werden.

Ziel dieser Einheit ist es, ein grundlegendes Programmierverständnis zu entwickeln. Die Schüler*innen kennen nach dieser Einheit erste Programmierbefehle und können diese benennen und identifizieren.

In **Teil 1** werden die Schüler*innen mit informatischen Begriffen, wie ‚Algorithmus‘, ‚Programm‘ und ‚Programmieren‘, vertraut gemacht.

Den Begriff ‚Algorithmus‘ lernen sie am Beispiel des Fieberthermometers kennen (Arbeitsblatt 1.1.1). Im Anschluss daran können sich die Schüler*innen Gedanken darüber machen, welche Algorithmen sie im Alltag kennen bzw. wo Algorithmen eine Rolle spielen könnten (siehe Arbeitsblatt 1.1.2).

Sie lernen außerdem ‚Calli‘ kennen. Calli ist die Figur auf den Arbeitsblättern.

In **Teil 1** werden weder Computer/Laptops noch der ‚Calliope mini‘ benötigt. Um erste ‚Befehle‘ geben zu können, spielen die Schüler*innen das Roboter-Computer-Spiel. Dabei lotsen sie entweder ihren/ihre Lehrer*in oder Mitschüler*in durch den Klassenraum (siehe Arbeitsblatt 1.1.4).

Im Anschluss können die Befehle, die zuvor eher mündlich stattfanden, mit dem Arbeitsblatt ‚Calli-Labyrinth‘ verschriftlicht werden. Hier können die Schüler*innen im Rahmen einer Geschichte Calli sicher vom Start ins Ziel bringen, ohne dass sie in das Feuer fällt (siehe Arbeitsblatt 1.1.5).

In **Teil 2** lernen die Schüler*innen zum ersten Mal den Mikrocontroller kennen (siehe Arbeitsblatt 1.2.1). Ziel ist es, zunächst die Komponenten und Funktionen

des Mikrocontrollers vorzustellen. Dazu gehört auch, dass die Schüler*innen den Mikrocontroller an den Computer anschließen und trennen sowie die Energieversorgung mittels Batterie herstellen können (siehe Arbeitsblatt 1.2.2).

Der Calliope mini wird zum ersten Mal gestartet und die vorprogrammierten Spiele gezeigt. Bei diesem Vorgang wird das EVA-Prinzip (Eingabe – Verarbeitung – Ausgabe) von der Lehrkraft erläutert und mit der Aktion auf dem Calliope in Verbindung gebracht (siehe Arbeitsblatt 1.2.4). Im Anschluss daran lernen die Schüler*innen zum ersten Mal den Editor kennen (in diesem Fall den MakeCode-Editor). Dabei ist das Ziel, sich mit der Bedienung des Computers/Laptops, der Tastatur, der Maus und des Internetbrowsers vertraut zu machen. Sie können am Editor erste kleine Programme erstellen. Als Unterstützung hierfür können ‚Callis Aufträge‘ dienen.

Hinweis: Die Programme werden noch nicht auf den Calliope gespielt, sondern nur im Simulator angesehen. Die Schüler*innen sollten die Möglichkeit haben, die vorprogrammierten Spiele zu Hause zu testen.

In **Teil 3** wird weiter geübt, mit dem Computer/Laptop, der Tastatur, der Maus und dem Internetbrowser umzugehen. Ziel ist es vor allem, den Weg des Speicherns, also die Sicherung der Daten, kennenzulernen. Da die Schüler*innen dies immer wieder brauchen, sollten sie sich dafür genügend Zeit nehmen und ausreichend lange üben.

Callis Aufträge können auch in dieser Einheit als Programmierbeispiele genutzt werden. Die Schüler*innen lernen zudem kennen, was Speichern ist. Dies wird mit Hilfe der Analogie „Arbeitsblatt = Datei, Mappe = Ordner, Schulrucksack = Datenspeicher“ sinnbildlich unterstützt.

Wichtig ist, zu üben, dass sie ihren Dateien und Ordnern passende Namen geben (siehe Arbeitsblatt 1.3.1), um diese auch wiederfinden zu können.

Um den Weg des Speicherns jederzeit wieder nachvollziehen zu können, gibt es eine Schritt-für-Schritt-Anleitung (siehe Arbeitsblatt 1.3.2).

3.1.2 EMPFEHLUNGEN/TIPPS/PRAXISBEISPIEL(E)

Aus der Praxis zeigt sich, dass die Schüler*innen beim Roboter-Computer-Spiel besonders viel Spaß hatten, gerade wenn die Lehrkraft durch den Raum geschickt wird. Ein großes Interesse wird auch bei der Begutachtung der verschiedenen Mikrocontroller gezeigt, da diese für den Großteil der Schüler*innen ganz neu sind.



Abbildung 5: Einblick in den Unterricht zum Thema Mikrocontroller

Auch die Arbeit mit Computern/Laptops stellt eine interessante Arbeit dar, wenn diese noch nicht oft in der Grundschule genutzt worden sind. Hier bietet es sich an, dass die Schüler*innen vorab gemeinsam mit der Lehrkraft Regeln für den Umgang mit Computern/Laptops erstellen. Regeln könnten zum Beispiel sein:

- Die Hände waschen, bevor mit dem Computer/Laptop gearbeitet wird.
- Das Gerät erst aufklappen, sobald dies gesagt wird.
- Die Tische freiräumen, so dass nur die Mappe und der Laptop auf dem Tisch stehen.
- Vereinbaren, wer den Laptop holt/anmacht und zurückbringt/ausmacht.
- Vereinbaren, dass die jeweiligen Nutzer*innen sich mit der Bedienung von Maus und Tastatur abwechseln.

Für die meisten Schüler*innen sind Begriffe, wie zum Beispiel ‚Algorithmus‘, neu und abstrakt. Daher bietet es sich an, diese Begriffe mit Gegenständen aus ihrem Alltag zu verknüpfen. Viele Schüler*innen haben Schwierigkeiten, diese Begriffe verbal wiederzugeben. Trotzdem können sie so auf andere Dinge, die dazu thematisch besprochen wurden, zurückgreifen.

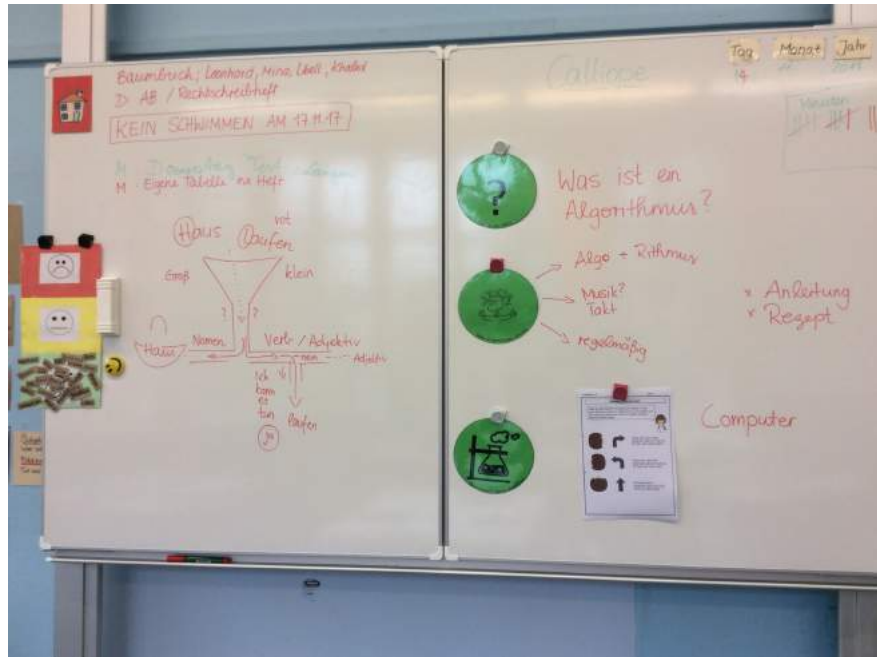


Abbildung 6: Einblick in den Unterricht zum Thema Algorithmus - Tafelbild

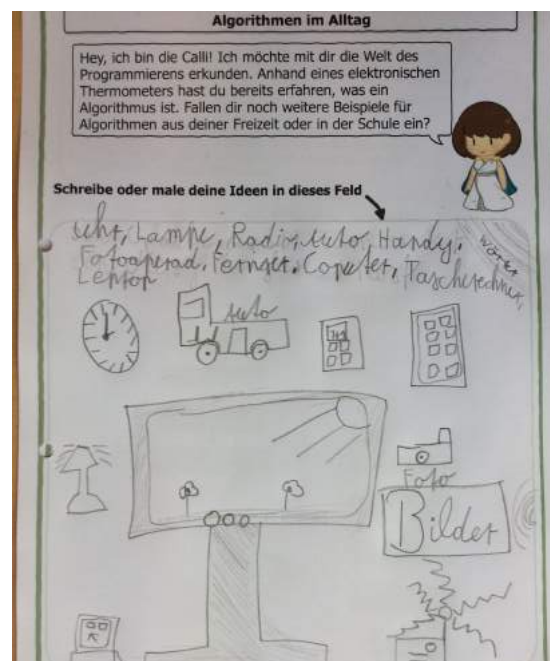








Abbildung 7: Einblick aus dem Unterricht zum Thema Algorithmus - exemplarisches ausgefülltes Arbeitsblatt

4. TEIL 1: WAS IST PROGRAMMIEREN? ALGORITHMUS, PROGRAMM, PROGRAMMIEREN

Inhalt	Eine Einführung / Grundlagen in das Programmieren (3 Teile)		
Fach/Fächer	Fächerübergreifende Medienbildung Sachunterricht		
Lernziele	 Die Schüler*innen entwickeln ein grundlegendes Programmierverständnis.  Sie entwerfen, implementieren und testen eigene Programme.  Sie nehmen den Mikrocontroller in seiner technischen und funktionalen Einbindung als solchen wahr.  Sie kommen mit der Bedienung des Computers, der Tastatur, der Maus und des Internetbrowsers zurecht.  Sie verstehen erste Programmierbefehle ("Computer Science Unplugged") und können sie benennen und identifizieren (informatische Kontrollstrukturen).  Sie können die technische Einbindung, das heißt den Grund und den Zeitpunkt des Anschlusses an einen Computer/der Trennung von einem Computer und gegebenenfalls die eigene Energieversorgung, überblicken.		
Zeit	Eine Stunde pro Unterrichtseinheit		
Klassenstufe	Ab der 3. Klasse		
Voraussetzungen	Internetzugang, Computer/ Laptops und Calliope-Mikrocontroller		
Material	Titel	Konzepte	Arbeitsmaterial
	Was ist Programmieren? (Teil 1)	Algorithmus, Programm, Programmieren	Unterrichtsverlauf: Was ist Programmieren, Teil 1 1.1.1 Algorithmus (WB) 1.1.2 Algorithmen im Alltag (VÜ) 1.1.3 Programm u. Programmieren (WB) 1.1.4 Roboter-Computerspiel (SP) 1.1.5 Calli-Labyrinth (ÜA) 1.1.6 Calli-Labyrinth-Lösung (ÜA)
	Was ist Programmieren? (Teil 2)	Mikrocontroller, E-V-A-Prinzip	Unterrichtsverlauf: Was ist Programmieren, Teil 2 1.2.1 Übersicht über Mikrocontroller (WB)

			1.2.2 Übersicht - Calliope (ÜA) 1.2.3 Übersicht - Calliope-Lösung (WB) 1.2.4 Callis Was-ist-was: Mikrocontroller (WB)
	Was ist Programmieren? (Teil 3)	Speichern, Ordner, Datei	Unterrichtsverlauf: Was ist Programmieren, Teil 3 1.3.1 Speichern/Ordner/Datei (WB) 1.3.2 Programme speichern, Teil 1 (AN) 1.3.3 Programme speichern, Teil 2 (AN) 1.3.4 Callis-Aufträge (ÜA)

4.1 ZEITLICHER UND INHALTLICHER AUFBAU

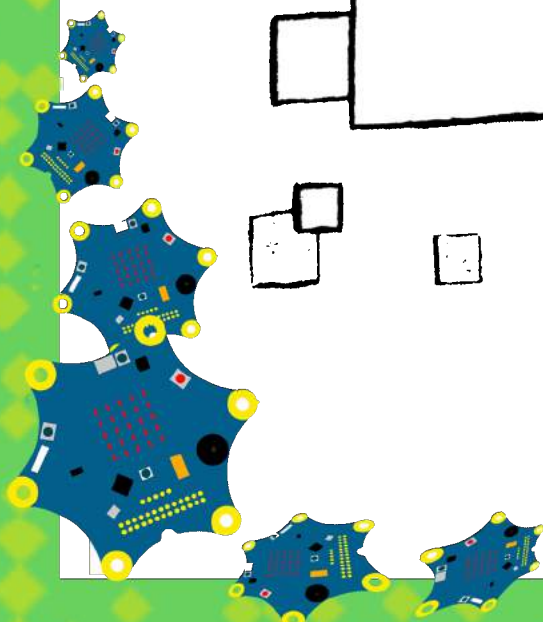
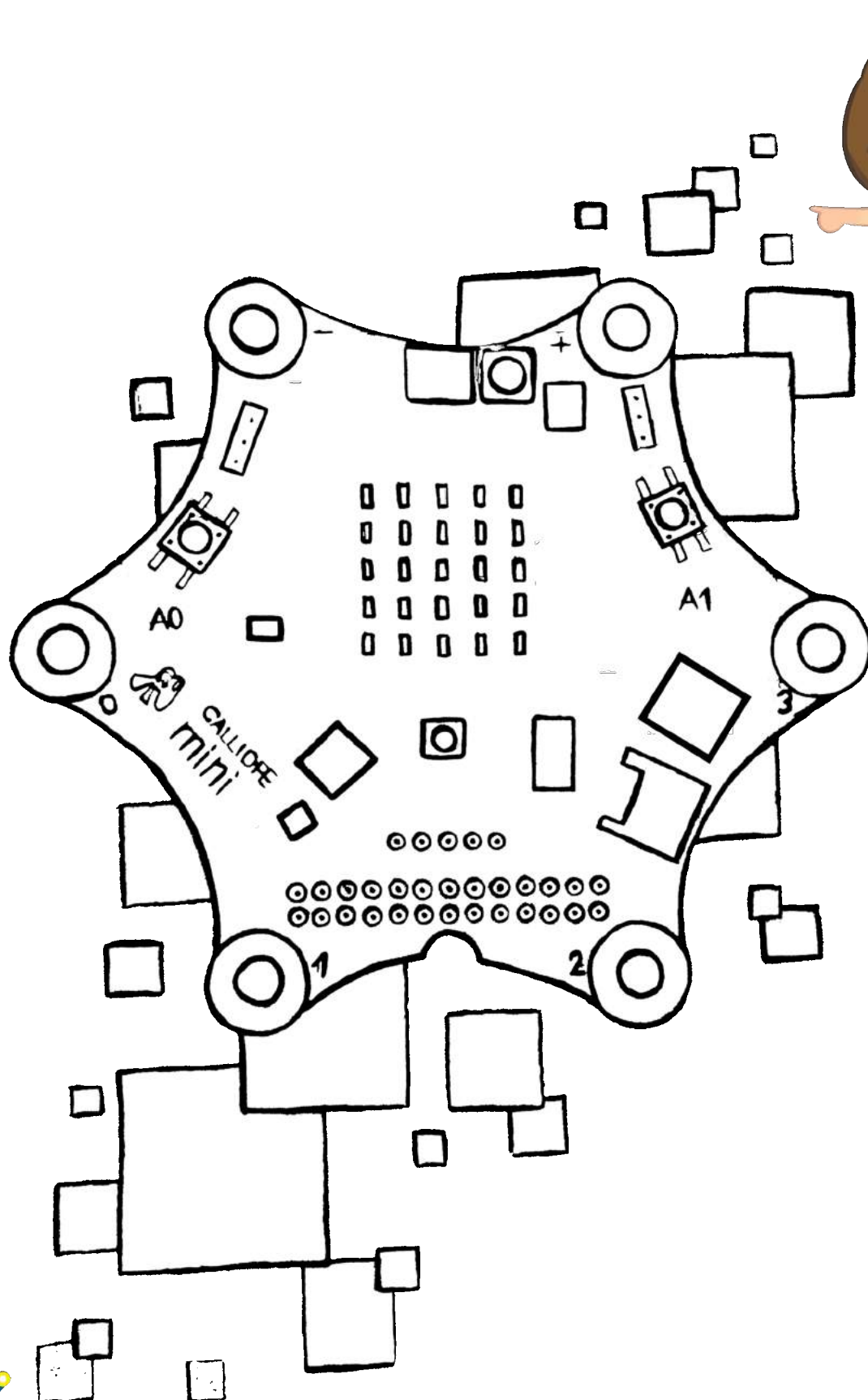
Zeit	Sozialform	Inhalt	Material	Bemerkung
5 Min.	Plenum	Die Lehrerin/der Lehrer führt in den Calliope-Unterricht ein.		
20 Min.	Plenum	Die Schüler*innen lernen den Begriff 'Algorithmus' kennen. Dabei wird das Ablaufdiagramm am Beispiel des Fieberthermometers an die Tafel geschrieben werden.	1.1.1 Callis Was-ist-Was: Algorithmus (WB)	
10 Min.	Einzelarbeit Plenum	Die Schüler*innen sollen mit dem ersten Blatt Calli kennenlernen und erste Gedanken zum Thema „Algorithmus im Alltag“ aufschreiben/aufmalen. Danach können die Schüler*innen im Plenum erläutern, was ihnen dazu einfiel.	1.1.2 Algorithmen im Alltag (VÜ)	Ein Beispiel könnte exemplarisch mit einem Ablaufdiagramm besprochen werden.
25 Min.	Gruppenarbeit Spiel	1. Der/die Lehrer*in oder ein/eine Schüler*in ist ein Roboter. Die anderen Schüler*innen sollen dem Roboter den Weg von A nach B erklären, ohne dass er gegen einen Tisch läuft. Dabei muss ihnen verdeutlicht werden, dass der Roboter eine Maschine ist und die Befehle, die sie ihm geben, eine Programmiersprache sind. Die tatsächliche Programmiersprache lernen sie später kennen. 2. Die Projektmappen werden (gegebenenfalls im Ziel) ausgegeben.	1.1.3 Callis Was-ist-Was: Programmieren (WB) 1.1.4 Roboter-Computer-Spiel (SP) Ziel-Blatt Projektmappen	Dies wird im Zusammenhang mit dem Spiel erklärt.
15 Min.	Einzelarbeit Gruppenarbeit	1. Die Schüler*innen sollen den gekennzeichneten Weg auf dem Arbeitsblatt ansehen und aufschreiben, wie Calli gehen müsste. 2. Der/die Tischnachbar*in soll dann	1.1.5 Calli-Labyrinth (mögliche Lösung) (ÜA) 1.1.6 Calli-Labyrinth (Lösung)	

		das Arbeitsblatt betrachten und die Schrittfolge kontrollieren.		
10 Min.	Einzelarbeit Hausaufgabe	<ul style="list-style-type: none"> • Die Schüler*innen heften ihre Arbeitsergebnisse in ihre Calliope-Mappe ein. • Die Eltern/die Geschwister programmieren. • Die Ergebnisse werden präsentiert und die Unterrichtseinheit wird resümiert sowie geschlossen. 		

4.2 ARBEITSBLÄTTER – TEIL 1

Klasse : _____

Name: _____



Callis Was-Ist-Was: Algorithmus

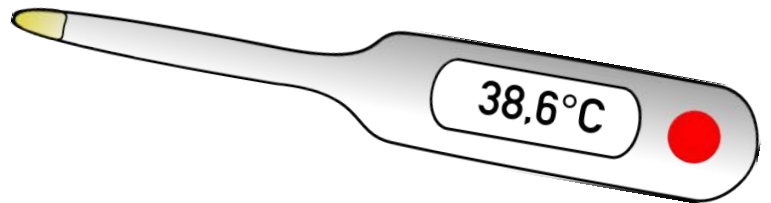
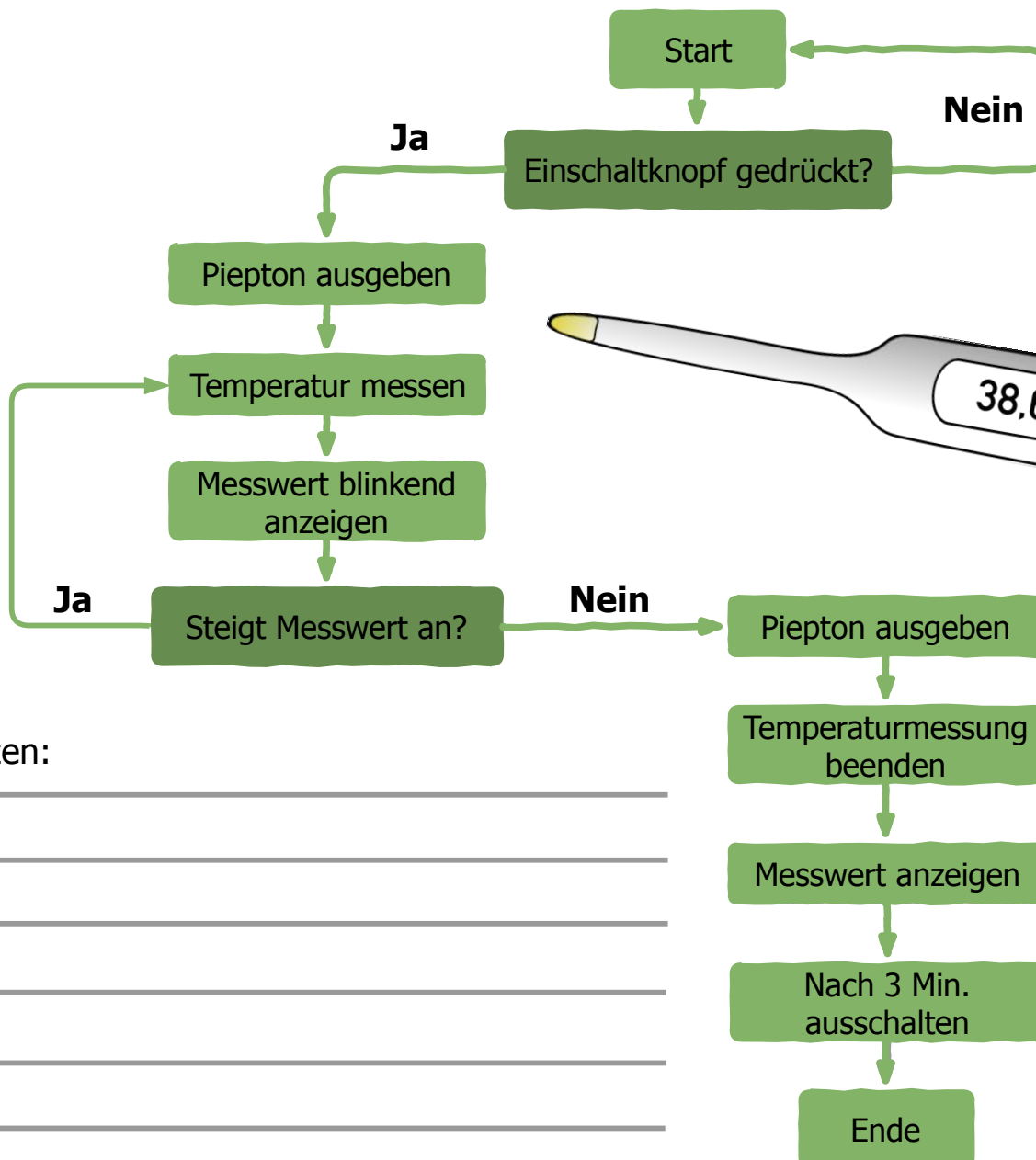
Was ist eigentlich ein Algorithmus?

Algorithmus

Ein Algorithmus ist eine Reihe von präzisen Anweisungen, um eine oder mehrere Aufgaben zu erledigen, zum Beispiel einen Fahrradschlauch zu wechseln oder dein Lieblingsbutterbrot zu schmieren.



Ein Algorithmus kann mit Hilfe eines Ablaufdiagramms dargestellt werden. Hier siehst du das Ablaufdiagramm für den Algorithmus der Funktionsweise eines elektronischen Fieberthermometers.



Notizen:

Algorithmen im Alltag

Hey, ich bin Calli! Ich möchte mit dir die Welt des Programmierens erkunden. Anhand eines elektronischen Thermometers hast du bereits erfahren, was ein Algorithmus ist. Fallen dir noch weitere Beispiele für Algorithmen aus deiner Freizeit oder in der Schule ein?



Schreibe oder male deine Ideen in dieses Feld,



zum Beispiel die Funktionsweise einer elektronischen Waage oder eines Fahrstuhls, der Ablauf beim Schuhebinden ...

Callis Was-Ist-Was: Programm und Programmieren

Was ist eigentlich ein Programm? Und was ist Programmieren?

Programm

Ein Programm ist ein Algorithmus, der von einer Maschine (zum Beispiel von einem Computer oder einem Mikrocontroller) verstanden und ausgeführt werden kann. Dafür braucht man Programmiersprachen. Bekannte Programmiersprachen sind: C++, Java, Python, JavaScript, PHP... usw.



Programmieren

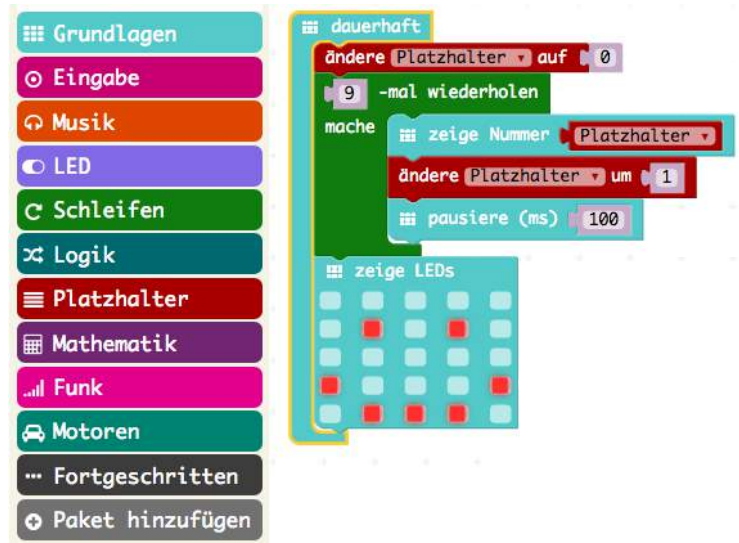
Programmieren nennt man den Prozess, ein Programm zu erstellen. Zum Programmieren braucht man einen Editor, wie zum Beispiel den PXT-Editor.

```

1 let Platzhalter = 0
2 basic.forever(() => {
3   Platzhalter = 0
4   for (let i = 0; i < 9; i++) {
5     basic.showNumber(Platzhalter)
6     Platzhalter += 1
7     basic.pause(100)
8   }
9   basic.showLeds(`
10     . . . . .
11     . # . # .
12     . . . . .
13     # . . . #
14     . # # # .
15     `)
16 })
17

```

Ein Programm in der Programmiersprache JavaScript



So sieht der PXT-Editor aus, mit dem du den Calliope Mikrocontroller programmierst.

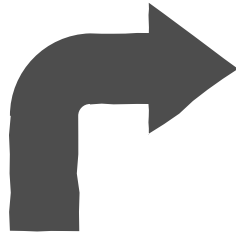
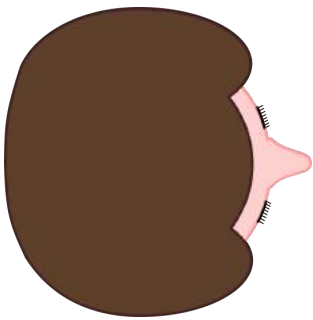
Hast du vielleicht noch Fragen?



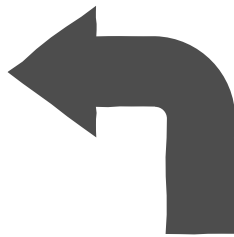
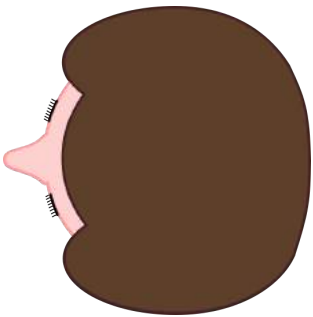
-
-
-
-

Roboter-Computer-Spiel

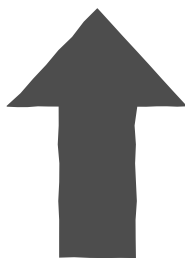
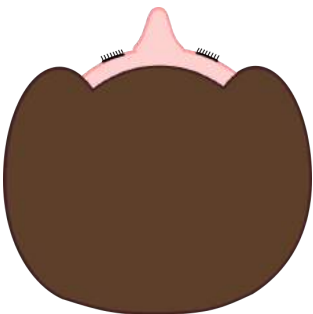
Lasst uns das Roboter-Computer-Spiel spielen! Lenkt euren Roboter (den/die Lehrer*in oder einen/eine Mitschüler*in) vom Start zum Ziel, ohne an Tische oder Stühle zu stoßen. Benutzt folgende Befehle dafür:



"Drehe dich nach rechts"
bedeutet, dass sich Calli um
90 Grad nach rechts dreht.



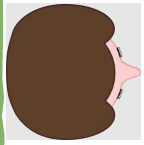
"Drehe dich nach links"
bedeutet, dass sich Calli um
90 Grad nach links dreht.



"Mache einen Schritt
geradeaus" bedeutet, dass
Calli einen Schritt nach vorne
macht.

Calli-Labyrinth

Schreibe die Befehle für mich auf, damit ich durch das Feuer zum Ziel komme. Ich soll immer nach vorne gehen. Manchmal muss ich mich zuerst drehen, um geradeaus gehen zu können. Also immer der Nase nach!



Start



Befehle und Abkürzungen

- Schritt geradeaus (SG) ↑
- Drehe links (DL) ↶
- Drehe rechts (DR) ↷

Tipp:

Manchmal braucht ihr auch 2 Befehle in einer Zeile.

Ziel

Start

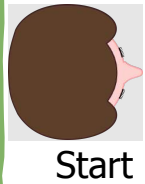
1. _____
2. _____
3. _____
4. _____
5. _____
6. _____
7. _____
8. _____

9.

10. _____
11. _____
12. _____
13. _____
14. _____
15. _____
- Ziel _____

Calli-Labyrinth (eine mögliche Lösung)

Schreibe die Befehle für mich auf, damit ich durch das Feuer zum Ziel komme. Ich soll immer nach vorne gehen. Manchmal muss ich mich zuerst drehen, um geradeaus gehen zu können. Also immer der Nase nach!



Befehle und Abkürzungen

- Schritt geradeaus (SG) ↑
- Drehe links (DL) ↶
- Drehe rechts (DR) ↷

Tipp:

- Manchmal braucht ihr auch 2 Befehle in einer Zeile.

Ziel

Start: SG



1. SG



2. SG



3. DR + SG



4. SG



5. SG



6. DL + SG



7. SG



8. DR + SG



9. SG



10. SG



11. DL + SG



12. SG



13. SG



14. DR + SG



15. DL + SG



Ziel erreicht! Super!



5. TEIL 2: WAS IST PROGRAMMIEREN? MIKRO-CONTROLLER, E-V-A-PRINZIP

5.1 ZEITLICHER UND INHALTLICHER AUFBAU

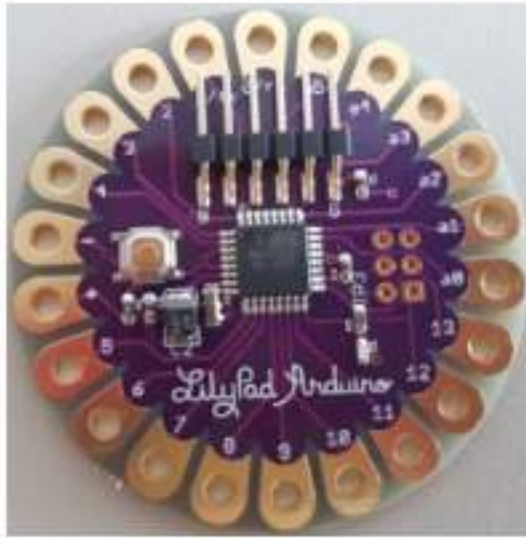
Zeit	Sozialform	Inhalt	Material	Bemerkung
5 Min	Plenum	1. Der/die Lehrer*in wiederholt Teil 1 von Modul 1 kurz. 2. Falls nötig, werden die restlichen Inhalte aus Teil 1, die zeitlich bisher nicht bearbeitet werden konnten, behandelt.		
5 Min	Plenum (optional)	Die Schüler*innen erhalten einen Überblick über Mikrocontroller, indem der/die Lehrer*in ihnen verschiedene Mikrocontroller zeigt und erläutert.	1.2.1 Übersicht über Mikrocontroller (WB)	Die Mikrocontroller werden mitgebracht.
10 Min	Plenum	Wenn dies vorher noch nicht besprochen wurde oder auch zur Wiederholung werden diese Fragen beantwortet: <ul style="list-style-type: none"> • Was ist ein Computer? • Was ist Programmieren? • Was kann programmiert werden? 	Die Ideen werden auf einem Wandplakat gesammelt/an die Tafel geschrieben.	Sie können auch als Wortspeicher für weitere Unterrichtseinheiten verwendet werden.
15 Min	Plenum Einzelarbeit	<ul style="list-style-type: none"> • Die Schüler*innen erhalten die nummerierten Calliope-Geräte. Sie machen sich mit dem Karton, dem USB-Anschluss und der Batterie vertraut. • Der/die Lehrer*in erläutert ihnen die Warnhinweise. • Als Nächstes schließen die Schüler*innen die Batterie an den Computer an und ... • Die Lehrkraft kann die Schüler*innen fragen, ... und ob sie so etwas schon einmal woanders gesehen haben (Assoziation). 	Der Calliope wird erstmalig ausgeteilt. 1.2.2 Übersicht-Calliope (ÜA) 1.2.3 Übersicht über den Calliope (Lösung) (ÜA)	Das Arbeitsblatt mit der Übersicht wird verteilt und kann im Laufe des Projektes mit den neuen Funktionen ergänzt werden.
10 Min	Plenum	EVA-Prinzip: Die Lehrkraft startet den Calliope und zeigt die vorprogrammierten Spiele. Dabei wird der Vorgang des EVA-Prinzips (Eingabe - Verarbeitung - Ausgabe) erläutert und mit dem in Verbindung gebracht, was gerade auf dem Calliope passiert. Hierfür wird die Tafel genutzt.	Tafel	Das EVA-Prinzip wird hier niederschwellig erläutert, so dass später immer darauf Bezug genommen werden kann.

		<ul style="list-style-type: none"> • Eingabe (entspricht dem Drücken eines Knopfes auf dem Calliope) • Verarbeitung (geschieht durch den Prozessor im Calliope) • Ausgabe (erfolgt über LEDs, die leuchten, o. Ä.) 	1.2.4 Callis Was-ist-Was (WB)	
5 Min	Einzelarbeit	Nun probieren die Schüler*innen kurz die Programme selbst aus. Danach werden die Calliope wieder ausgemacht und zurück in den Karton gelegt.		Die Calliope-Geräte werden mit nach Hause genommen, so dass die Schüler*innen die Spiele länger ausprobieren können.
40 Min	Einzelarbeit	<ul style="list-style-type: none"> • Der Editor wird mittels Beamer vorgestellt. • Die Schüler*innen sollen die Laufschrift, das LED-Feld und die LED-Anzeige selbst ausprobieren, indem sie Code eingeben (Reproduktion/reine Umsetzung). • Sie arbeiten nur am Simulator (noch keine Übertragung), da sie die vorprogrammierten Spiele noch eine Woche testen sollen. 	Beamer/ Laptop/ Computer Übungsheft: Callis Aufträge	Die Schüler*innen werden in der nächsten Einheit lernen, ihre Dateien zu sichern und auf den Calliope zu spielen.
10 Min	Einzelarbeit Plenum	Die Schüler*innen heften ihre Arbeitsergebnisse in ihre Calliope-Mappe ein. Präsentation, Resümé der UE & Abschluss		

5.2 ARBEITSBLÄTTER –TEIL 2

Übersicht Mikrocontroller

Schau mal, was es alles für verschiedene Mikrocontroller gibt!



Lilypad



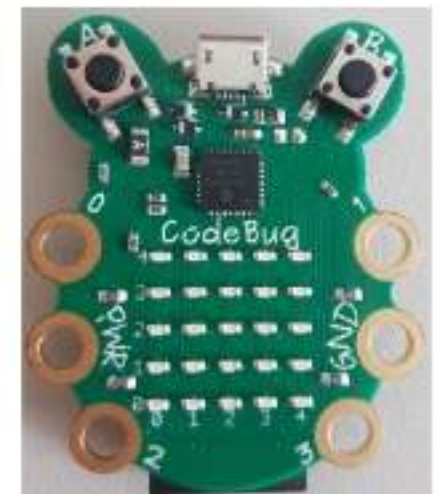
Calliope mini



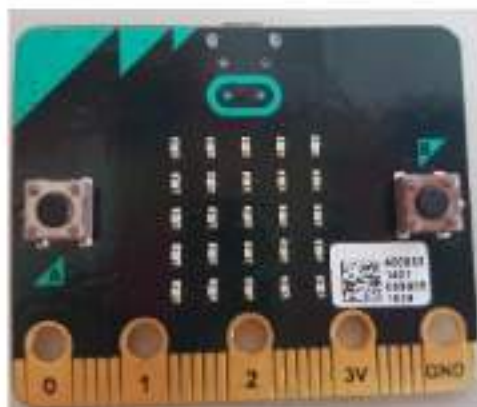
Prog BOB!



Arduino UNO



CodeBug



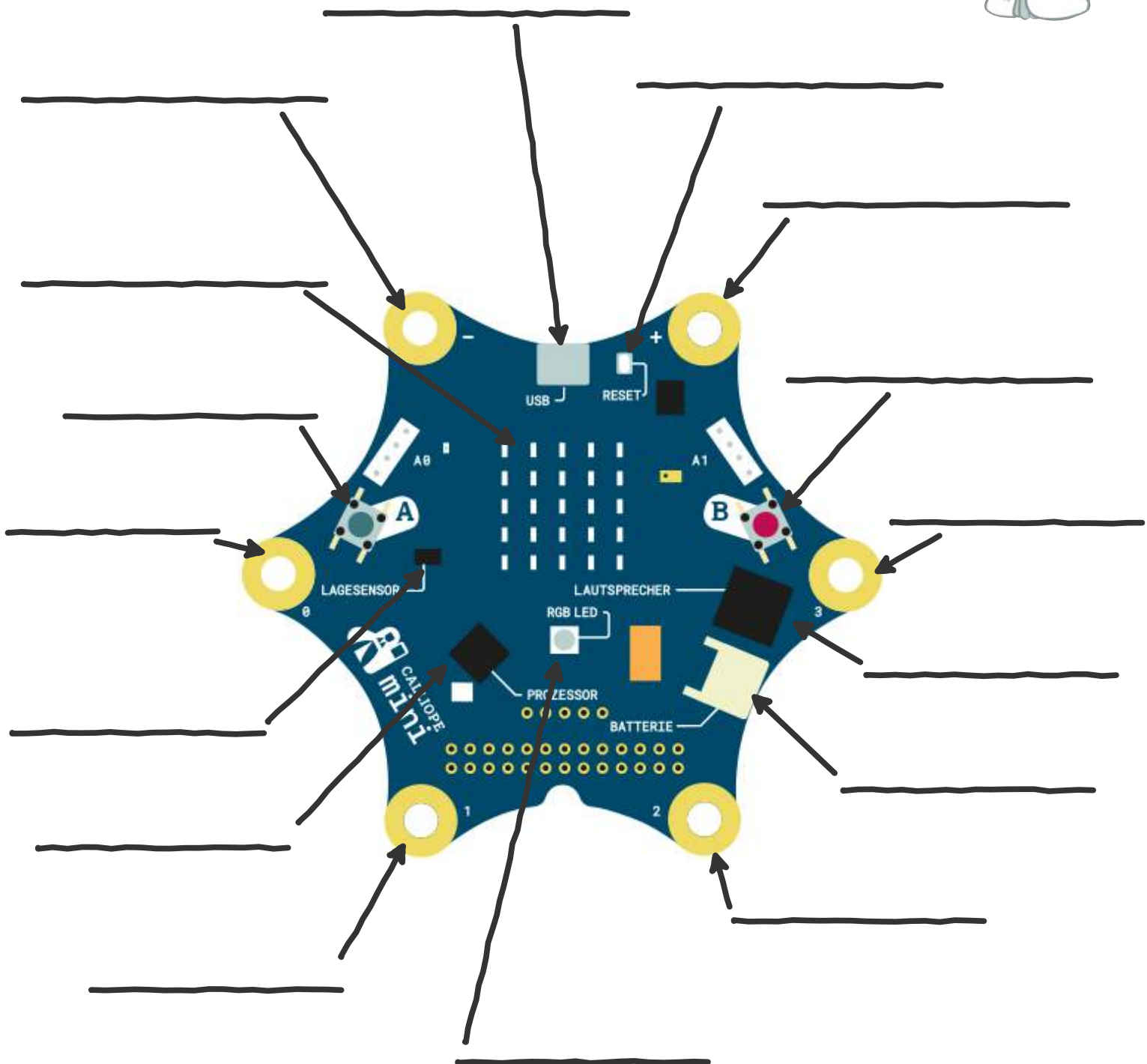
micro:bit



Raspberry Pi

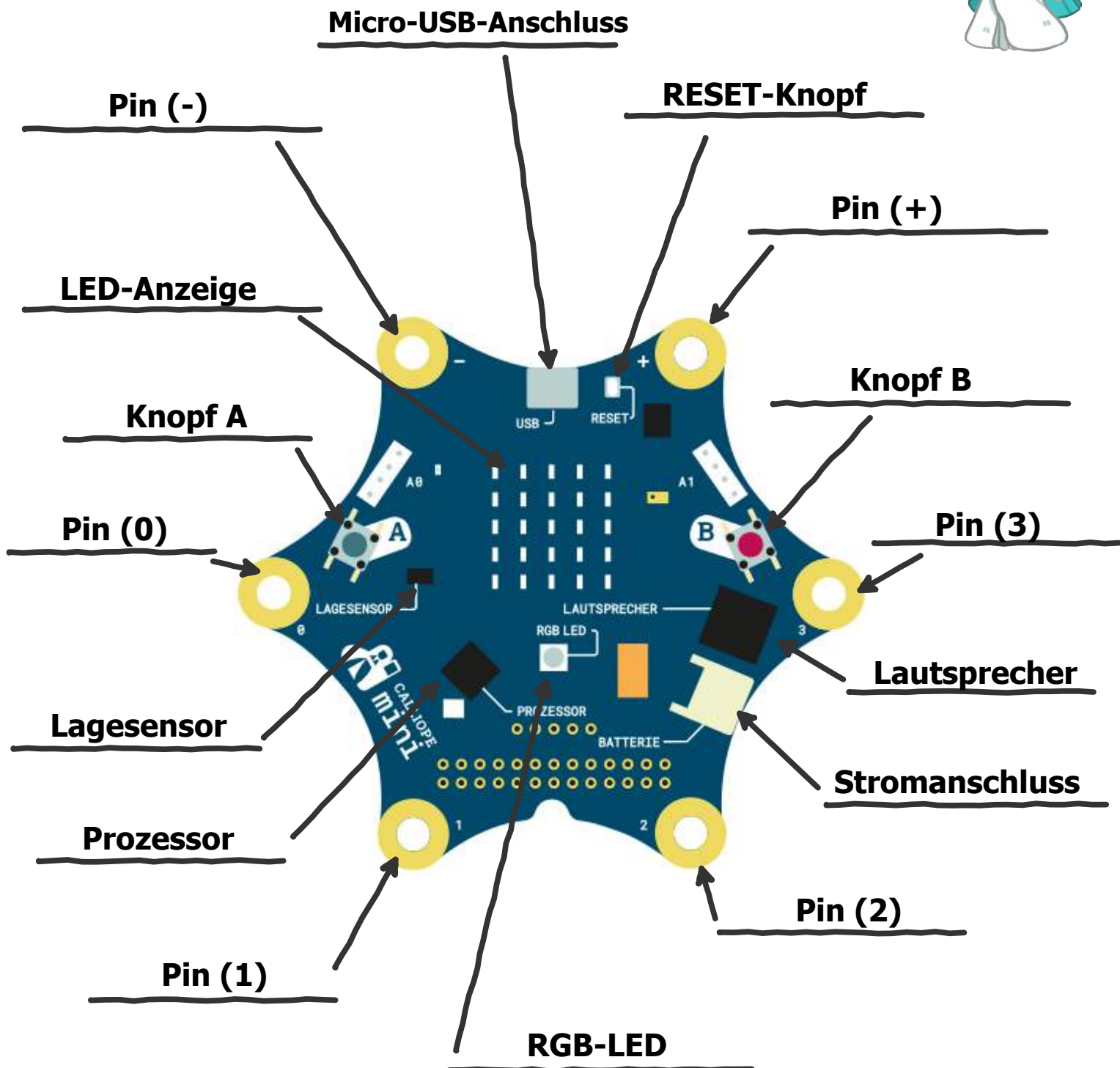
Übersicht - Calliope

Auf deinem Calliope gibt es viel zu entdecken und du lernst ihn immer besser kennen! Kannst du die einzelnen Teile des Calliope benennen?



Übersicht - Calliope (Lösung)

Auf deinem Calliope gibt es viel zu entdecken und du lernst ihn immer besser kennen! Kannst du die einzelnen Teile des Calliope benennen?



Callis Was-Ist-Was: Mikrocontroller

Was ist eigentlich ein Mikrocontroller?

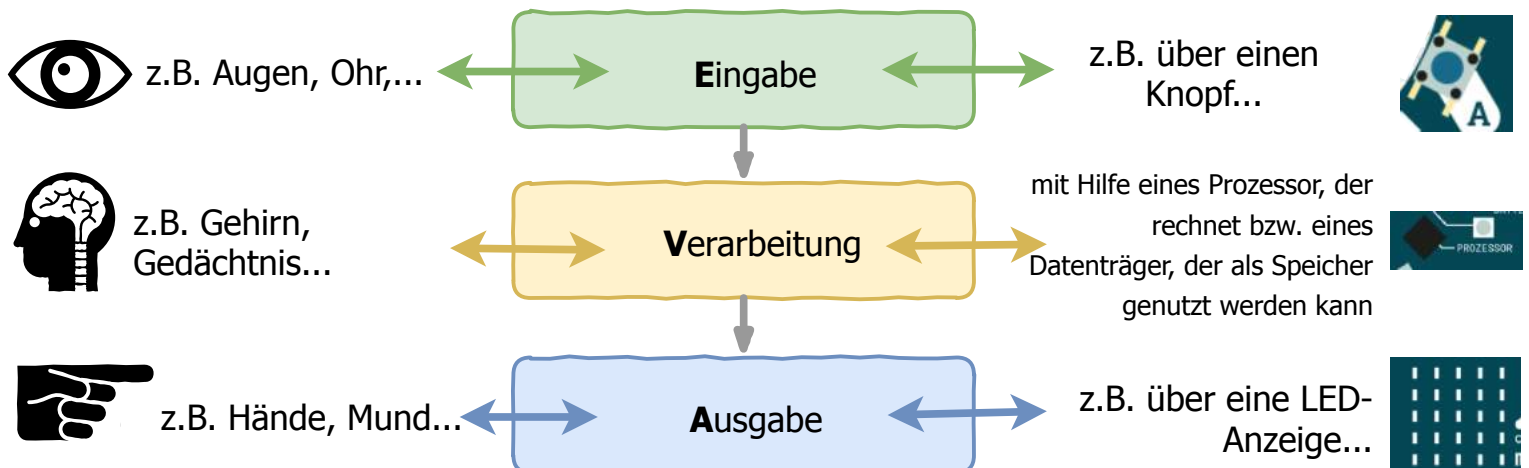


Mikrocontroller

Mikrocontroller sind "**Mini-Computer**", die heutzutage in vielen Geräten eingebaut sind, zum Beispiel in elektronischen Fieberthermometern oder in Alarmanlagen.

Mikrocontroller funktionieren nach dem **EVA-Prinzip**. Es werden Informationen eingegeben, diese werden dann verarbeitet und danach wieder ausgegeben. Dieses Prinzip lässt sich auch auf den Menschen übertragen.

Mensch ↔ **EVA-Prinzip** ↔ **Mikrocontroller**



Kennst du noch weitere Gegenstände, in denen Mikrocontroller sind?



6. TEIL 3: WAS IST PROGRAMMIEREN? SPEICHERN, ORDNER, DATEI

6.1 ZEITLICHER UND INHALTLICHER AUFBAU

Zeit	Sozialform	Inhalt	Material	Bemerkung
10 Min	Plenum	<ul style="list-style-type: none"> Es wird eine Reflexionsrunde durchgeführt und die letzte Stunde wird wiederholt. Die Schüler*innen werden gefragt, ob sie den Calliope dabei haben und zu Hause mit ihm gearbeitet haben. 		
20 Min	Plenum Sitzkreis	<p>In dieser Stunde wird mit dem Editor geübt und vor allem gelernt, wie die eigenen Programme gespeichert werden:</p> <ul style="list-style-type: none"> Die Lehrkraft kann zum Beispiel sagen: "Wenn ihr ein neues Programm auf dem Calliope speichert, ist das alte Programm verloren. Was können wir machen, damit wir alle Programme behalten können?" Die Lehrkraft zeigt den Kindern am Beamer, wie sie ihren Programmcode abspeichern und herunterladen (Programmtitel eingeben und herunterladen) und dann das Programm abspeichern (zum Beispiel auf dem 'Marktplatz' im Klassenordner). Zum Schluss sollen sie den Code in ihren Ordnern finden und auf den Calliope spielen. 	<p>1.3.1 Callis-Was- Ist-Was: Speichern (AN)</p> <p>1.3.2 Programme speichern (Teil 1) (AN)</p> <p>1.3.3 Programme speichern (Teil 2) (AN)</p>	Die Schüler*innen sollen lernen, ihre Projekte passend zu benennen, abzuspeichern sowie diese wiederzufinden.
5 Min	Plenum	Die Schüler*innen zeigen, wie das Programm direkt auf den Calliope mini gespeichert werden kann.	Beamer	
30 Min	Partnerarbeit	Die Schüler*innen können nun das Übungsheft „Callis Aufträge“ weiter bearbeiten bzw. die Programmcodes jetzt passend gekennzeichnet sichern und auf den Calliope speichern.	Callis- Aufträge	

10 Min	Plenum Einzelarbeit	<p>In der Abschlussrunde wird das Gelernte zusammengefasst:</p> <ul style="list-style-type: none"> • Was könnt ihr jetzt schon mit dem Calliope? • Was ist das wichtigste beim Speichern? • Was sind Ordner und Dateien? • Welche Programmierbefehle habt ihr heute genutzt? <p>Die Schüler*innen heften ihre Arbeitsergebnisse in ihre Calliope-Mappe ein.</p>	<p>Einzelne Befehlsblöcke werden ausgedruckt, damit die Schüler*innen, damit die Schüler*innen sie beim Erklären nutzen können.</p> <p>Mappe</p>	
--------	------------------------	---	--	--

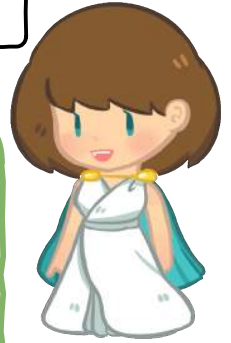
6.2 ARBEITSBLÄTTER –TEIL 3

Callis Was-Ist-Was: Speichern

Was ist eigentlich Speichern?

Speichern

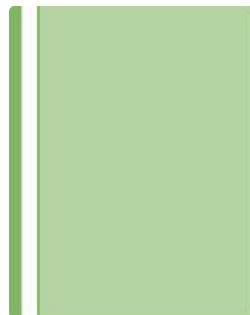
Bei Computern bedeutet Speichern, Daten in einen Datenspeicher, zum Beispiel auf eine Festplatte oder eine DVD, zu schreiben. Speichern kannst du dir ungefähr so vorstellen wie das Abheften deines Arbeitsblattes in deine Mappe und das Packen deiner Mappe in die Schultasche. Siehe dir dazu das Schaubild unten an.



! Beim Speichern ist es sehr wichtig, dass du deinen Dateien und Ordnern passende Namen gibst, damit du sie besser wiederfindest. !



Arbeitsblatt



Mappe



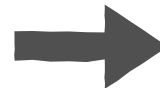
Schulrucksack



Datei



Ordner

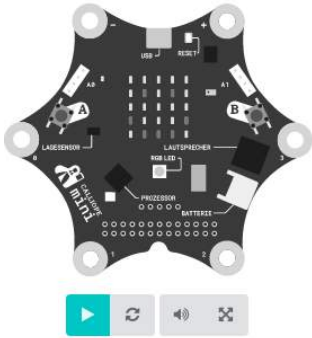
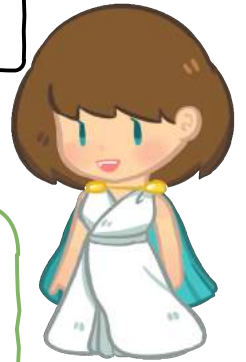


Datenspeicher

Notizen:

Programme speichern (Teil 1)

Ich zeige dir jetzt, wie du dein Programm abspeicherst!

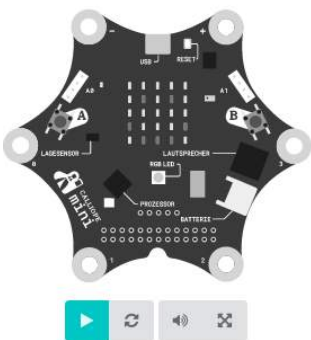


Gib deinem Programm einen passenden Titel. Wenn du zum Beispiel eine Alarmanlage programmierst, könntest du dein Programm "Alarmanlage" nennen.

Notizen:

Herunterladen

Programmtitel



Klicke links auf "Herunterladen", um dein Programm herunterzuladen.

Notizen:

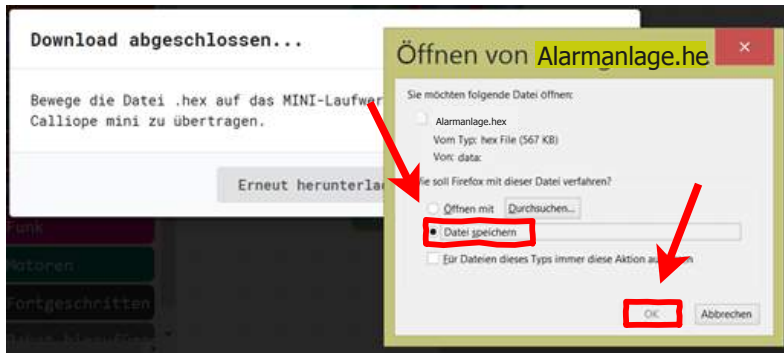
Herunterladen

Alarmanlage



Weiter auf der nächsten Seite

Programme speichern (Teil 2)



Klicke auf "Datei speichern" und abschließend auf "ok".

Notizen:



Wähle jetzt den Ort aus, wohin du dein Programm speichern möchtest. Suche dafür den entsprechenden Klassenordner (nach Absprache mit der Lehrkraft). Kontrolliere hier noch einmal deinen Programmtitel. Nicht wundern: Vor deinen Programmtitel wird automatisch das Wort "mini" gesetzt.

Notizen:



Wenn der Download abgeschlossen ist, klickst du nur noch auf "Fertig"!


Notizen:

Callis Aufträge

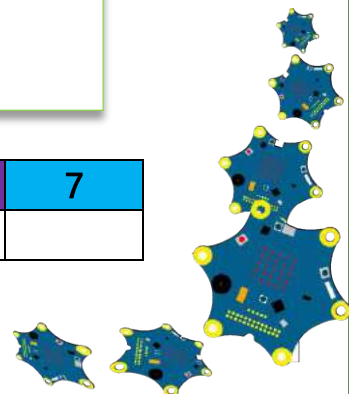
Probiere doch mal diese Beispiele im Editor einzugeben. Beginne dabei immer mit „Start“. Diese Anleitung hilft dir dabei. Hake deine fertigen Aufgaben ab. Viel Erfolg!





Start

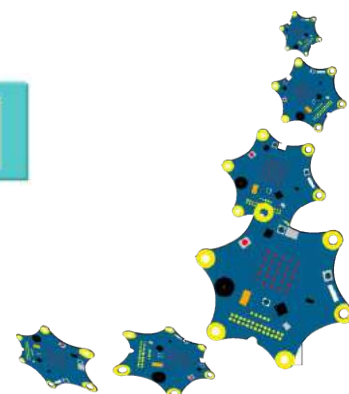
- Öffne den Internetbrowser und gib **pxt.calliope.cc** ein.
- Gehe auf  Projekte und starte ein neues Projekt.
- Lösche die angezeigten Blöcke bevor du beginnst. Dabei ziehst du sie nach links auf die bunten „Befehlsblöcke“, bis ein Papierkorb erscheint.

Auftrag:	1	2	3	4	5	6	7
Fertig?							



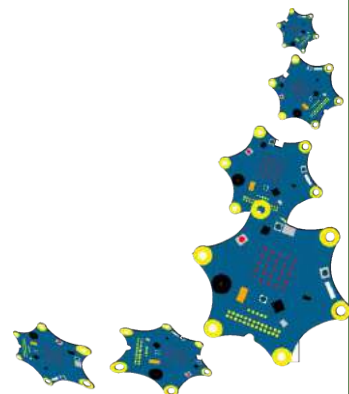
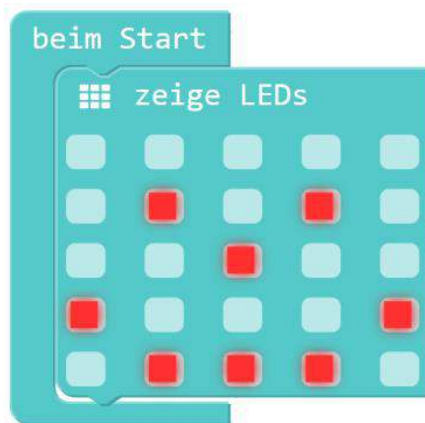
Callis Auftrag 1

- Ziehe aus  Grundlagen den Block „**beim Start**“ auf die Programmierfläche.
- Füge aus  Grundlagen den Block „**zeige Zeichenfolge** „**hello!**“ hinzu.
- Setze den Block unter „**beim Start**“, bis er einrastet.
- Tippe in das Feld deinen eigenen Namen ein.
- Starte die Simulation, indem du links auf das türkise Kästchen klickst.



Callis Auftrag 2

- 1. Ziehe aus **Grundlagen** den Block „**beim Start**“ auf die Programmierfläche.
- 2. Füge aus **Grundlagen** den Block „**zeige LEDs**“ hinzu.
- 3. Setze den Block unter „**beim Start**“, bis er einrastet. Versuche nun den Smiley nachzumachen.
- 4. Starte die Simulation, indem du links auf das türkise Kästchen klickst.



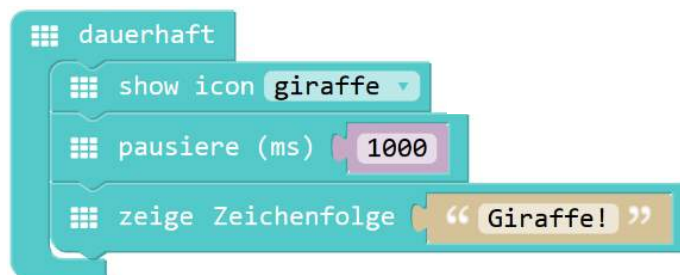
Callis Auftrag 3

- 1. Ziehe aus **Eingabe** den Block „**wenn Knopf A gedrückt**“ auf die Programmierfläche.
- 2. Füge aus **Grundlagen** den Block „**setze LED-Farbe auf Rot**“ hinzu und verändere die Farbe.
- 3. Starte die Simulation, indem du links auf das türkise Kästchen klickst. Danach musst du noch den „**Knopf A**“ auf dem Calliope drücken.



Callis Auftrag 4

- 1. Ziehe aus **Grundlagen** „**dauerhaft**“ auf die Programmierfläche.
- 2. Füge aus **Grundlagen** den Block „**show icon**“ hinzu.
- 3. Setze den Block unter „**dauerhaft**“, bis er einrastet.
- 4. Wähle nun das Wort „**giraffe**“ aus.
- 5. Füge aus **Grundlagen** den Block „**pausiere (ms)**“ hinzu und verändere die Millisekunden (ms) auf **1000**.
- 6. Füge aus **Grundlagen** den Block „**zeige Zeichenfolge**“ hinzu und ändere das Wort in „**Giraffe!**“
- 7. Starte die Simulation, indem du links auf das türkise Kästchen klickst.



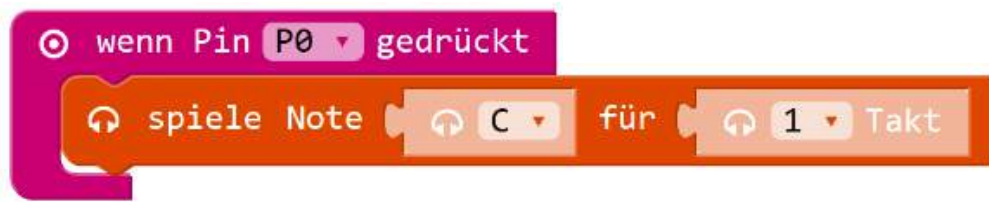
Callis Auftrag 5

- 1. Ziehe aus **Eingabe** den Block „**wenn Knopf A gedrückt**“ auf die Programmierfläche.
- 2. Füge aus **Grundlagen** den Block „**show icon**“ hinzu und wähle „**yes**“ (ja) aus.
- 3. Ziehe aus **Eingabe** den Block „**wenn Knopf B gedrückt**“ auf die Programmierfläche.
- 4. Füge aus **Grundlagen** den Block „**show icon**“ hinzu und wähle „**no**“ (nein) aus.
- 5. Starte die Simulation, indem du links auf das türkise Kästchen klickst. Danach musst du noch den „**Knopf A**“ oder/ und den „**Knopf B**“ auf dem Calliope drücken.



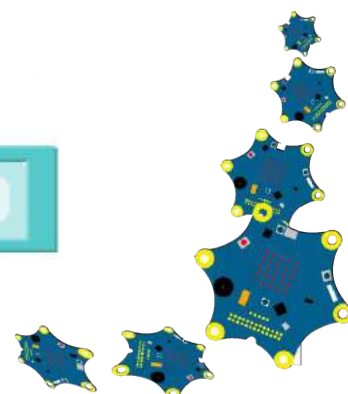
Callis Auftrag 6

- 1. Ziehe aus **Eingabe** den Block „**wenn P0 gedrückt**“ auf die Programmierfläche.
- 2. Füge aus **Musik** den Block „**spiele Note C für 1 Takt**“ hinzu.
- 3. Starte die Simulation, indem du links auf das türkise Kästchen klickst. Danach musst du noch den „**Pin 0**“ drücken.
- 4. Hinweis: Wenn du einen Pin programmierst, kannst du dir das Ergebnis nicht im Simulator anzeigen lassen.



Callis Auftrag 7

- 1. Ziehe aus **Eingabe** den Block „**wenn geschüttelt**“ auf die Programmierfläche.
- 2. Füge aus **Grundlagen** den Block „**setze LED-Farbe auf**“ hinzu und wähle die Farbe „**Gelb**“ aus.
- 3. Starte die Simulation, indem du links auf das türkise Kästchen klickst. Danach musst du den Calliope noch **schütteln**.



zeige Nummer Zufallszahl ▼

zeige Nummer Sprünge ▼

zeige Symbol



dauerhaft

setze LED-Farbe auf Rot ▼

zeige Zeichenfolge receivedString ▼

setze LED-Farbe auf Grün ▼

pausiere (ms) 100

zeige Nummer

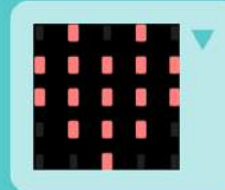


Bildschirminhalt löschen

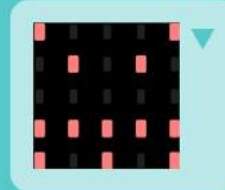
zeige LEDs



zeige Symbol



zeige Symbol



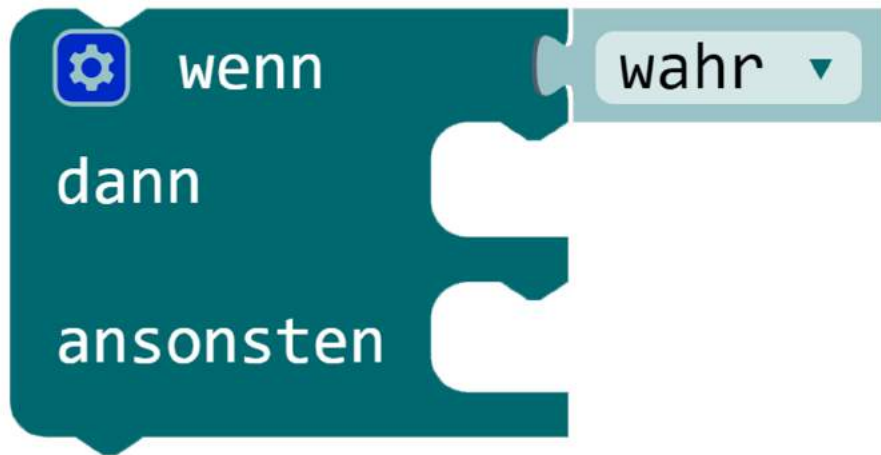
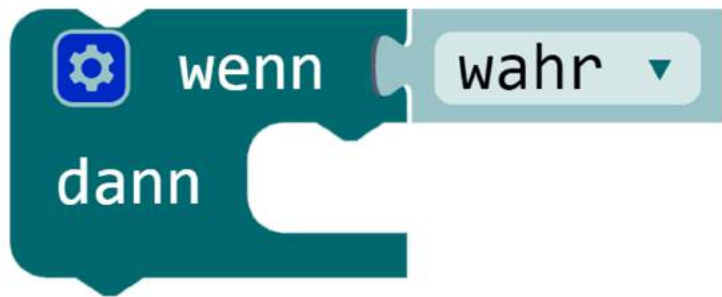
beim Start

zeige Symbol



zeige Zeichenfolge







sende Zahl



setze Gruppe



schicke Wert



=



sende Zeichenfolge



wenn Datenpaket empfangen

receivedNumber



wenn Datenpaket empfangen

name

value



wenn Datenpaket empfangen

receivedString



sende Zeichenfolge



 Geschwindigkeit (bpm)

 Mittleres C

 1 ▼ Takt

 ändere die Geschwindigkeit (bpm)  20

 ändere Geschwindigkeit auf (bpm)  120

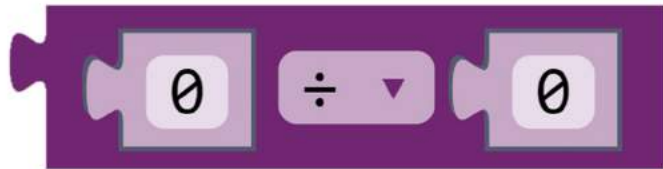
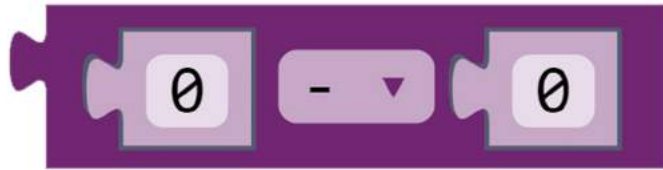
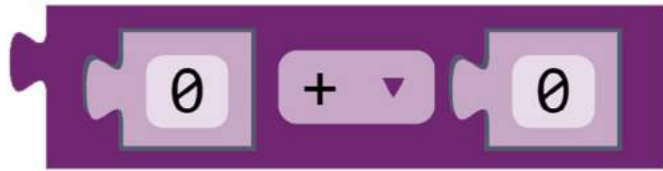
🎧 Musik bei Note aus Melodie gespielt ▼

🎧 spiele Note 🎧 Mittleres C für 🎧 1 ▼ Takt

🎧 Beginne Melodie 🎧 Dadadum ▼ Wiederhole einmal ▼

🎧 Klingelton (Hz) 🎧 Mittleres C

🎧 pausiere (ms) 🎧 1 ▼ Takt



Zeichne x 0 y 0 Helligkeit 255

Helligkeit

Setze Helligkeit auf 255

Halte Animation an

Halte Animation an

LED aktivieren falsch ▼

LED aktivieren falsch ▼

 Zeichne x  y 

 schalte Pixel x  y 

 schalte um x  y 

⦿ wenn Knopf A ▼ gedrückt

⦿ wenn geschüttelt ▼

⦿ wenn Knopf B ▼ gedrückt

⦿ Lichtstärke

ändere Platzhalter auf 0

ändere Platzhalter um 1

Platzhalter

item

name

receivedNumber

ändere Zufallszahl auf 0

receivedString

Sprünge

Zufallszahl

Neue Variable anlegen

ändere Zufallszahl um 1

7. WEITERFÜHRENDE LITERATUR

Es haben bereits mehrere Initiativen und Autor*innen Ideen zu Calliope-mini-Projekten entwickelt, die teilweise in die hier dargestellten Unterrichtsvorschläge eingeflossen sind. In den folgenden Quellen lassen sich weitere Anregungen sammeln.

7.1 INTERNETSEITEN

Die Calliope gGmbH mit Sitz in Berlin wurde mit dem Zweck gegründet zur Verbreitung des Calliope mini beizutragen. Sie setzt sich dafür ein, dass Kinder ab der 3. Klasse unentgeltlich mit Konzepten, Materialien und Technologie für digitale Bildung ausgestattet werden und veröffentlicht alle ihre Materialien unter offener Lizenz: <https://www.calliope.cc/>.

Im Saarland bestand früh der politische Wille, den Calliope mini für alle Grundschulen verfügbar zu machen. Mehrere Institutionen und Einzelpersonen, u.a. die Universität des Saarlandes (Fakultät für Mathematik und Informatik) unterstützen Schulen dabei. Über den Blog <http://calliope-saarland.de> sind Lernkarten und Lehrermaterialien verfügbar, sowie Ankündigungen zu bundesweiten Aktivitäten rund um den Calliope mini.

Die AG Didaktik der Informatik der Bergischen Universität Wuppertal <https://ddi.uni-wuppertal.de/> hat zum Calliope mini unter anderem eine Lehrerhandreichung veröffentlicht:

https://ddi.uni-wuppertal.de/Calliope_Handreichung.pdf

Die Stiftung „Haus der kleinen Forscher“ ist eine gemeinnützige Stiftung zur Förderung von MINT-Projekten unter der Schirmherrschaft des Bundesministeriums für Bildung und Forschung. Sie pflegen einen eigenen Bereich zu Informatik für Kinder:

<https://www.haus-der-kleinen-forscher.de/de/praxisanregungen/experimentethemen/informatik/>.

IT2School ist ein Projekt des „Wissensfabrik – Unternehmen für Deutschland e.V.“. Dieser wurde von verschiedenen Unternehmen, darunter BASF, Bosch, Fischer und Trumpf, gegründet, hat seinen Sitz in Ludwigshafen und kooperiert u.a. mit Universitäten. Der Verein OFFIS e.V. hat mit der Universität Oldenburg Lernmaterialien zum Calliope mini entwickelt, die als Modul 8B über it2school abgerufen werden können:

https://portal.wissensfabrik.de/portal/fep/de/dt.jsp?setCursor=1_554644

Appcamps ist eine gemeinnützige Unternehmensgesellschaft mit Sitz in Hamburg, die von diversen Stiftungen unterstützt wird und kostenlose Materialien zum Calliope mini bereitstellt: <https://appcamps.de/>

Der Cornelsen-Verlag ist ein ehemaliger Schulbuchverlag, mittlerweile eine Verlagsgruppe, die u.a. digitale Materialien vertreibt. Zum Calliope mini stellt Cornelsen Lernmaterialien bereit und vertreibt auch den Calliope mini selbst: <https://www.cornelsen.de/calliope/>

GitHub ist ein us-amerikanischer Onlinedienst, der 2018 von Microsoft übernommen wurde. Er stellt auf seinen Servern eine Dateiverwaltung und Speicherplatz für digitale Projekte zur Verfügung, unter anderem für Calliope mini-Projekte: <https://github.com/calliope-mini/>

Hackster ist eine us-amerikanische Hardware-Entwicklerplattform und -community. Einige Mitglieder beschäftigen sich mit dem Calliope mini:
<https://www.hackster.io/calliope-mini>

7.2 BÜCHER

Bergner, Nadine; Franken, Patrick; Kleeberger, Julia; Leonhardt, Thiemo, Lukas, Mario (2017): Das Calliope-Buch. Spannende Bastelprojekte mit dem Calliope-Mini-Board. DPunkt Verlag GmbH

Bergner, Nadine; Leonhardt, Thiemo (2017): Programmieren mit dem Calliope mini für Dummies Junior. Kleiner Computer mit ganz viel Power! Verlag Wiley-VCH

Immler, Christian (2017): Der kleine Hacker: Programmieren lernen mit dem Calliope mini. Franzis Verlag GmbH