

Einstieg in das algorithmisches Problemlösen mit dem Calliope

Didaktische Hinweise

Zielgruppe

Der *Calliope mini* wurde entwickelt, um Schülerinnen und Schülern bereits in der Grundschule einen Einstieg in die Algorithmik zu ermöglichen (s. [1]). In Niedersachsen gibt es jedoch noch nicht an allen Grundschulen ein solches Angebot. Die vorliegenden Materialien wurden daher für Schülerinnen und Schüler an weiterführenden Schulen entwickelt, die mithilfe des Calliope mini an das algorithmische Problemlösen herangeführt werden sollen.

Erprobt wurden die Materialien und Aufgaben im Anfangsunterricht in Jahrgang 6 und Jahrgang 9 eines Gymnasiums. Die Schülerinnen und Schüler in Jahrgang 6 arbeiten über längere Zeit motiviert und kreativ mit dem Werkzeug und den Materialien. In Jahrgang 9 eignet sich das Werkzeug eher für einen kurzen Einstieg z. B. im Rahmen des Pflichtfachs oder einzelne gezielte Projekte.

Für Lerngruppen, die bereits Vorkenntnisse im Umgang mit algorithmischen Konzepten wie Kontrollstrukturen und Variablen haben, können einzelne Aufgaben oder Projekte herausgenommen werden, um einen Einblick in das algorithmische Problemlösen mithilfe eines Sensor-Aktor-Systems zu geben.

Voraussetzungen

Es ist hilfreich, wenn die Schülerinnen und Schüler mit dem schuleigenen Dateisystem und dem Herunterladen von Dateien über den Webbrowser vertraut sind. Die entsprechenden Schritte müssen ansonsten im Unterricht erläutert und es muss entsprechend Zeit dafür eingeplant werden.

Weiter Vorkenntnisse, insbesondere im Bereich Algorithmik, sind nicht notwendig. Die verwendeten algorithmischen Konzepte werden anhand geeigneter Beispiele nach und nach eingeführt und erläutert.

Lernziele

Die Schülerinnen und Schüler lernen anhand überschaubarer Projekte, wie sie mithilfe eines Algorithmus auf die Eingabewerte der Sensoren reagieren und geeignete Ausgaben erzeugen können.

Dabei werden die algorithmischen Konzepte *Kontrollstrukturen* (Anweisungen, Sequenz, Verzweigung und Schleife), *Ereignissteuerung* und *Variablen* eingeführt. Schleifen treten dabei vor allem in Form der *dauerhaft*-Schleife auf.

Die Einheit lässt sich sowohl in das Modul *Algorithmisieren und Implementieren* aus dem Lernfeld *Algorithmisches Problemlösen* als auch in das Modul *technische Realisierung automatisierter Prozesse* aus dem Lernfeld *Automatisierte Prozesse* einordnen (vgl. [8], S. 20ff.).

Modul *Algorithmisieren und Implementieren*

Aufgeführt sind nur die für diese Einheit relevanten Kompetenzen.

Basis	Vertiefung
Die Schülerinnen und Schüler ...	
<ul style="list-style-type: none"> - beschreiben einen gegebenen Algorithmus in ihren eigenen Worten. - führen einen gegebenen Algorithmus aus. 	
<ul style="list-style-type: none"> - benennen Anweisung, Sequenz, Schleife und Verzweigung als elementare Kontrollstrukturen. - entwickeln und implementieren einen Algorithmus in einer grafischen Programmiersprache auf experimentelle Weise. 	<ul style="list-style-type: none"> - entwerfen einen Algorithmus unter zielgerichteter Verwendung der elementaren Kontrollstrukturen. - formulieren Bedingungen mit Hilfe der logischen Verknüpfungen „Und“, „Oder“ und „Nicht“. - verwenden Variablen und Wertzuweisungen in einfachen Algorithmen.
<ul style="list-style-type: none"> - überprüfen, ob eine Implementierung die Problemstellung löst. 	<ul style="list-style-type: none"> - testen systematisch, ob eine Implementierung die Problemstellung löst.
P1.1-2 P2 P3.1 P4.1 I2.1-2 I3.2	zusätzlich: P3.3 I2.3

Tabelle 1: Ausschnitt aus dem Modul *Algorithmisieren und Implementieren*, Lernfeld *Algorithmisches Problemlösen*, vgl. [8], S.20f.

Modul *technische Realisierung automatisierter Prozesse*

Basis	Vertiefung
Die Schülerinnen und Schüler ...	
<ul style="list-style-type: none"> - benennen Typen von Sensoren, Aktoren und Verarbeitungskomponenten von technischen Geräten und ordnen sie der Eingabe, Verarbeitung und Ausgabe zu. - lesen Sensoren aus und steuern Aktoren an. 	
<ul style="list-style-type: none"> - implementieren einen Algorithmus zur Steuerung einer technischen Komponente. 	entwickeln einen Algorithmus zur Steuerung eines einfachen Informatiksystems.
P1.2 P2 P3.1 P4.1 I3.1	P1.2 P2 P5.1 I2 I3.2

Tabelle 2: Ausschnitt aus dem Modul *technische Realisierung automatisierter Prozesse*, Lernfeld *Automatisierte Prozesse*, vgl. [8], S.22

Wie Tabelle 1 und Tabelle 2 zeigen, können im Rahmen dieser Einheit unterschiedliche prozessbezogene und inhaltbezogene Kompetenzen erworben werden. Wie vertieft die Kompetenzen erworben werden, hängt davon ab, wie vertieft die Arbeitsblätter bearbeitet werden. Bei den prozessbezogenen Kompetenzen kommt neben dem Bereich *P2 Implementieren*, dem Bereich *P5 Informatiksysteme als Werkzeuge nutzen* eine besondere Bedeutung zu. Wenn die Schülerinnen und Schüler geeignete Sensoren selbständig wählen, werden Hard- und Softwarewerkzeuge nicht nur bei der Problemlösung eingesetzt (P 5.1), sondern auch unter Berücksichtigung ihrer Vor- und Nachteile ausgewählt (P 5.2).

Im Bereich der inhaltsbezogenen Kompetenzen steht der Bereich *I3 Informatiksysteme*, speziell das Konstruieren und Erschließen der Funktionsweise von Informatiksystemen im Vordergrund (I 3.2 und I 3.4).

Werkzeuge

Die Hardware

Die Materialien wurden für den Calliope mini in der Version V1.3 von 2017 erstellt (s. [3]) und getestet. Die meisten Beispiele wurden inzwischen auch mit der Nachfolgeversion v2.0 (Calliope mini REV2) getestet und an einzelnen Stellen angepasst. Die Materialien können daher für beide Versionen problemlos eingesetzt werden. Die meisten Aufgaben lassen sich aber vermutlich auch mit vergleichbaren Sensor-Aktor-Systemen, wie z. B. dem micro:bit bearbeiten (s. [4]).

Der Calliope kann um zusätzliche Sensoren (z. B. einen Ultraschallsensor) und Aktoren (z. B. Motoren) erweitert werden. Die vorliegenden Materialien beschränken sich jedoch auf die Sensoren und Aktoren, die der Calliope fest verbaut mitbringt.

Die Entwicklungsumgebung Microsoft MakeCode

Es gibt verschiedene Entwicklungsumgebungen zur Programmierung des Calliope. Eine Übersicht ist auf der Webseite der Calliope gGmbH zu finden (s. [2]). An dieser Stelle erfolgt weder eine Aufzählung noch eine Bewertung der verschiedenen Entwicklungsumgebungen.

Die vorliegenden Materialien wurden für die Entwicklungsumgebung Microsoft MakeCode in der Version 4.0.29 erstellt (s. [5]). Diese Entwicklungsumgebung bietet sich aufgrund der grafischen Programmiersprache für Schülerinnen und Schüler der Sekundarstufe I ohne Vorerfahrungen an. Das Design der Blöcke wurde in der aktuellen Version an die Entwicklungsumgebung Scratch (s. [7]) angepasst, die in der Sekundarstufe I ebenfalls weit verbreitet ist.

Die Sprache des Editors kann auf Deutsch eingestellt werden. Die Blöcke werden im Hintergrund in JavaScript und Python übersetzt und es kann zwischen der grafischen und textbasierten Darstellung des Programms hin- und hergewechselt werden. Das kann für Schülerinnen und Schüler, die an der textbasierten Programmierung interessiert sind, von Vorteil sein und erleichtert die Einsicht, dass grafisches und textbasiertes Programmieren gleichwertig sind.

Die Entwicklungsumgebung Microsoft MakeCode wird im Browser gestartet. Nachdem die Seite geladen wurde, kann auch ohne Internetverbindung weitergearbeitet werden (s. [6]).

Ein neues Projekt enthält bereits die Blöcke *beim Start* und *dauerhaft*. Dies minimiert die Gefahr, dass bei Programmteilen, die in einer Endlosschleife ausgeführt werden sollen, die entsprechende Schleife vergessen wird.

Die Blöcke für die ersten Programme sind in der Kategorie *Grundlagen* zusammengefasst. Die Blöcke für die Kontrollstrukturen sind auf die Kategorien *Schleifen* und *Logik* aufgeteilt, wobei die Kategorie *Logik* sowohl die Verzweigungen als auch Blöcke mit Vergleichsoperatoren zur Konstruktion von Bedingungen beinhaltet. Variablen werden in der Kategorie für die Blöcke zum Umgang mit Variablen erzeugt.

In der Kategorie *Eingaben* stehen Blöcke zur Verfügung, die den aktuellen Wert des jeweiligen Sensors liefern. Für einige Sensoren, die Tasten A und B, die Pins und den Lagesensor, gibt es zusätzlich *wenn*-Blöcke zur ereignisgesteuerten Reaktion auf die Sensorwerte. Die ereignisgesteuerte

Reaktion ist für Anfängerinnen und Anfänger zunächst intuitiver als das explizite Auslesen der Sensorwerte und das Überprüfen einer Bedingung in einer Endlosschleife.

Durch die ereignisgesteuerte Reaktion auf Sensorwerte sowie den Einsatz mehrerer *dauerhaft*-Blöcke können parallele Programmabläufe implementiert werden. Hier ist darauf zu achten, dass diese Möglichkeit nicht zu exzessiv genutzt wird, damit die Programme überschaubar und nachvollziehbar bleiben.

Die Blöcke für die Ansteuerung der Aktoren sind in verschiedenen Kategorien zu finden. Blöcke für akustische Ausgaben über den Lautsprecher befinden sich in der Kategorie *Musik* und Blöcke für die einzelne Farb-LED in der Kategorie *Grundlagen*. In der Kategorie *Grundlagen* sind auch Blöcke zur Ansteuerung der LED-Matrix als Ganzes zu finden. Spezielle Blöcke, mit der sich einzelne LEDs der LED-Matrix ansteuern oder die Helligkeit verändern lassen, befinden sich in der Kategorie *LED*.

Für die Aufgaben werden außerdem noch die Blöcke aus der Kategorie *Mathematik*, z. B. zum Erzeugen von Zufallszahlen, und die Blöcke aus der Kategorie *Funk* für die Kommunikation der Calliopes untereinander benötigt. Diese kann als Erweiterung hinzugeladen werden.

Die Entwicklungsumgebung enthält einen Simulator, mit dem die Programme getestet werden können.

Das ausführbare Programm und der Quellcode befinden sich in derselben Datei, die für die Ausführung des Programms auf dem Calliope gespeichert wird. Für die spätere Bearbeitung sollte eine Kopie an einem anderen Ort gespeichert werden, auf den man vom Browser aus zugreifen kann. Bei der Erstellung der Materialien und der Erprobung im Unterricht wurde mit MakeCode im Browser gearbeitet und die Programme über das USB-Kabel auf den Calliope übertragen. Alternativ stehen für mobile Geräte die App „Calliope mini“ und die Übertragung per Bluetooth zur Verfügung (s. [2]). Beachten Sie ggf. die Hinweise im Bereich Hilfe, um eine Bluetooth-Verbindung zum Calliope mini herzustellen.

Didaktische Hinweise zu den Arbeitsblättern und Aufgaben

Die Arbeitsblätter bestehen im Wesentlichen aus Leitfäden, die in die Themen *Erste Schritte*, *Arbeit mit Sensoren*, *Umgang mit Variablen* und *Funk* aufgeteilt sind. In diesen Leittexten wechseln sich Erläuterungen und Aufgaben ab. Zu dem Bereich *Sensoren* gibt es zusätzlich einige Projektvorschläge.

Die Erläuterungen sind vor allem für Schülerinnen und Schüler gedacht, die selbständig weiterarbeiten oder zu Hause noch einmal etwas nachlesen möchten. Im Unterricht bietet sich die gemeinsame Erarbeitung in Unterricht an. Um den Schülerinnen und Schüler den Überblick zu erleichtern, können daher auch gezielt nur einzelne Aufgaben zur Verfügung gestellt werden.

Zu jedem Bereich sind im Folgenden einige Hintergrundinformationen zu finden.

Erste Schritte

Im Rahmen des Arbeitsblattes *01_AB_Erste Schritte* sollen sich die Schülerinnen und Schüler mit der Entwicklungsumgebung und dem Speichern der Programme vertraut machen. Die Aufgaben beschränken sich dabei zunächst auf die Verwendung der Aktoren des Calliope. Neben der *dauerhaft*-Schleife, werden unterschiedliche Anweisungs-Blöcke benötigt, die für die ersten Aufgaben in der Kategorie *Grundlagen* zu finden sind. In Aufgabe 4 kann auch die Verwendung von Schleifen der Form *wiederhole-n-mal* erprobt werden.

Einige Schülerinnen und Schüler werden die Aufgaben jedoch schnell von sich aus erweitern und beispielsweise die Ausgabe von Tönen ergänzen. Je nach Lerngruppe erübrigt sich Aufgabe 5 damit ggf.

Der Einstieg und die Verwendung der meisten Blöcke aus der Kategorie *Grundlagen* ist für die Schülerinnen und Schüler in der Regel sehr intuitiv. Lediglich die Eingabe eines Farbtons mithilfe des Rot-, Grün- und Blauwerts erfordert Hintergrundwissen zur additiven Farbmischung. Die Aufgaben können problemlos ohne diesen Block über die direkte Farbauswahl bearbeitet werden. Für interessierte Schülerinnen und Schüler enthält der Leitfaden jedoch ein Angebot sich die Farbmischung mit dem RGB-Modell zu erarbeiten. Der Zugang kann mithilfe des Calliope ganz experimentell gestaltet werden. Lässt man die Farb-LED nur schwach leuchten (RGB-Werte auf 10 setzen), so kann man diese betrachten und erkennt dabei mit bloßem Auge, dass sie eine rote, eine grüne und eine blaue LED enthält. So lassen sich beispielsweise bei gelbem Licht, die rote und die grüne LED erkennen. Diese Idee greift Aufgabe 6 auf. Testen Sie vorher, ob die Leuchtkraft der LED ausreichend schwach ist, um die Schülerinnen und Schüler die LED direkt betrachten zu lassen. Stellen Sie ggf. das fertige Programm zur Verfügung, um sicherzustellen, dass die Leuchtkraft nicht zu hoch eingestellt ist.

Optional können Materialien aus dem Paket *Darstellung und Codierung von Farben* aus dem Lernfeld *Computerkompetenz* als Exkurs oder im Anschluss bearbeitet werden, um die Zusammensetzung der Farben im RGB-Modell zu vertiefen.

Die Sensoren

Wenn den Schülerinnen und Schülern zu zweit nur ein Calliope zur Verfügung steht, ergibt sich beim ersten Arbeitsblatt *01_AB_ErsteSchritte* ggf. die Frage, welcher Name der Calliope ausgegeben soll. Darüber lässt sich die Verwendung der Tasten A und B zur Auswahl motivieren.

Am Beispiel der Tasten A und B wird die ereignisgesteuerte Reaktion auf einen Sensor mithilfe des *wenn*-Blocks eingeführt. In Version 1 des Arbeitsblattes *02_AB1_Sensoren* wird parallel auch die Überprüfung des Wertes in einer Bedingung in einer Verzweigung kombiniert mit einer *dauerhaft*-Schleife vorgestellt¹. Je nach Lerngruppe kann auf diese Variante zunächst verzichtet und erst später im Zusammenhang mit dem Helligkeitssensor eingeführt werden. Das wäre das Vorgehen in Version 2 des Arbeitsblattes.

Am Beispiel des Lichtsensors lernen die Schülerinnen und Schüler die Werte eines Sensors experimentell zu erkunden und im Kontext der Anwendung zu interpretieren. Das hier vorgestellte Vorgehen sollte von den Schülerinnen und Schülern immer wieder eingefordert werden, wenn sie sich die Verwendung weiterer Sensoren erschließen, da sie zum einen die Selbständigkeit und zum anderen das Verständnis für die Funktionsweise und Konstruktion sensorgesteuerter Systeme fördert. Da für den Lichtsensor kein Ereignis-Block zur Verfügung steht, muss hier die Formulierung einer Bedingung in einer Verzweigung in Kombination mit einer *dauerhaft*-Schleife erfolgen. Da dies nicht allen Schülerinnen und Schülern beim ersten Mal selbständig gelingen wird, steht hier für Version 1 eine Hilfekarte mit einem Grundgerüst des Programms zur Verfügung. Bei Version 2 ist das

¹ Dabei ist zu beachten, dass das Ereignis „Wenn Knopf A gedrückt“ erst ausgelöst wird, wenn die Taste wieder losgelassen wird und auch nur, wenn die Taste nur kurz gedrückt wurde. Der Block A liefert hingegen so lange den Wert wahr, wie die Taste gedrückt wird. Dieser Unterschied kann in Aufgabe 1c der Version 1 erarbeitet werden.

Grundgerüst direkt im Arbeitsblatt enthalten. Zur Sicherung und Festigung der Struktur des Grundgerüsts und Einführung erster Fachbegriffe kann das Arbeitsblatt *02_AB1a_Sicherung_Grundstruktur* verwendet werden.

Aufgabe 5 stellt eine Festigung im Umgang mit den Tasten und des Lichtsensors dar. Im Unterschied zu Aufgabe 4 reagiert der Calliope hier nicht, wenn es dunkel, sondern wenn es hell ist. Diese Aufgabe wird im Arbeitsblatt zur Verwendung von Variablen noch einmal aufgegriffen. Die Schülerinnen und Schüler sollten ihre Lösungen daher auch auf dem Rechner speichern.

Bei Aufgabe 5 können neben dem Lichtsensor, aber auch andere Sensoren verwendet werden, z. B. der Bewegungs- bzw. Lagesensor. Die Schülerinnen und Schüler haben hier oft vielfältige Ideen, erkennen aber auch das Problem, dass der Alarm nicht nur durch den „Dieb“, sondern je nach verwendetem Sensor auch durch den Besitzer oder den Transport in der Schultasche ausgelöst werden kann. Diese Diskussionen sind sehr wertvoll, da sie auch zu der Erkenntnis führen, dass Sensordaten in einem bestimmten Kontext interpretiert werden und diese Interpretation auch falsch sein kann. Mit diesem Wissen lässt sich beispielsweise verstehen, warum im Auto angezeigt wird, eine Person auf dem Beifahrersitz sei nicht angeschnallt, obwohl dort nur eine Einkaufstasche steht.

Die Aufgaben zur Einführung der Konzepte sind relativ eng geführt. Im Rahmen einer kleinen Projektphase bietet es sich daher an dieser Stelle an, die Schülerinnen und Schüler ein erstes selbst gewähltes Projekt durchführen zu lassen. Je nach Kreativität der Schülerinnen und Schüler kann die Wahl des Projektes dabei ganz frei oder anhand der beiliegenden Projektvorschläge erfolgen. Es bietet sich an, ggf. Ideen der Schülerinnen und Schüler zu sammeln, um einzugrenzen, was sich mithilfe des Calliope umsetzen lässt.

Bei einigen Projekten werden die Schülerinnen und Schüler vermutlich auf Probleme stoßen, die sich ohne die Möglichkeit sich einen Zustand zu merken, nicht lösen lassen. Damit kann dann die Einführung von Variablen motiviert werden.

Zur Verwendung der Pins gibt es ein eigenes Arbeitsblatt, da die Verwendung insbesondere zum Messen analoger Werte nicht selbsterklärend ist. Wird bei den Pins nur zwischen gedrückt und nicht gedrückt unterschieden, können sie ähnlich wie die Tasten A und B verwendet werden. Wird hingegen der analoge Wert an einem Pin gemessen, können die Leitfähigkeit differenzierter erfasst und damit andere Projekte umgesetzt werden. Dieses Arbeitsblatt kann jedoch optional eingesetzt werden. Auch eine Beschränkung auf die Blöcke in der Kategorie *Eingaben* ist möglich.

Variablen

Das Arbeitsblatt verwendet Variablen zunächst, um sich verschiedene Zustände eines Systems zu merken. Um eine Vorstellung einer Variablen zu entwickeln, wird ein Schubladenmodell verwendet. Noch anschaulicher wird es, wenn auch im Unterricht ein solches Modell zur Verfügung steht und die Vorgänge haptisch nachvollzogen werden können.

Im zweiten Teil werden Variablen dann u.a. zum Zählen verwendet.

Für Aufgabe 3 werden beide Konzepte, sowohl eine Variable als Zähler, als auch eine Variable für den Zustand des Systems benötigt. Die Herausforderung bei der Lichtschranke besteht darin, eine Person, die den Lichtstrahl unterbricht, nur einmal zu zählen. In einer ersten Annäherung kann dies mithilfe einer Wartezeit bis zum nächsten Zählen gelöst werden. In einer zuverlässigeren Variante kann in einer Variablen vermerkt werden, ob bei der aktuellen Lichtunterbrechung bereits gezählt wurde (Zustandsspeicher). Das System kann daher schrittweise optimiert werden, so dass immer weniger

Vorgaben für das Verhalten des Anwenders und der zu zählenden Personen gemacht werden müssen.

Im Aufgabe 4 kommt die Verwendung von Zufallszahlen hinzu. Wird die Zufallszahl einfach als Zahl ausgegeben, kommt diese Aufgabe auch ohne Variablen aus. Die Herausforderung besteht daher darin, unterschiedliche Würfel zu entwickeln, welche die Zahl in eine andere Darstellung übersetzen. Es werden einige Vorschläge in der Aufgabe gemacht. Die Schülerinnen und Schüler können hier aber auch noch weitere Varianten entwickeln.

Kommunikation über Funk

Die Calliopes per Funk miteinander kommunizieren zu lassen, hat die Lerngruppen, in denen die Einheit getestet wurde, besonders motiviert und inspiriert. Aufgabe 1 wurde bis hin zu einem Verabredungskommunikator, mit dessen Hilfe man Anfragen für eine Verabredung stellen und Zeit und Ort vereinbaren kann, nach eigenen Ideen der Schülerinnen und Schüler erweitert.

Aufgabe 3 ist bewusst offen formuliert. Hier können verschiedene von den Schülerinnen und Schülern bereits erprobte Komponenten kombiniert werden. Der Calliope zum Überprüfen des Lichts, kann das Licht und den Lichtschalter gleich mitsimulieren. Für den Herd kann ebenfalls ein An-/Ausschalter simuliert oder auch der Temperatursensor eingesetzt werden. Zwischen offenem und geschlossenem Fenster kann z. B. mithilfe eines Drucksensors unterschieden werden.

Außerdem müssen hier Absprachen in der Gruppe getroffen werden: Wann senden die Calliopes? In regelmäßigen Abständen, bei einer Veränderung oder auf Anfrage? Welche Nachrichten senden Sie, wie werden die Zustände „an“ und „aus“ jeweils codiert?

Das EVA-Prinzip

Zu Beginn einer Unterrichtsreihe zum Physical Computing haben die Schülerinnen und Schüler teilweise die Fehlvorstellung, dass ein Sensor nicht nur einen Wert in der Umgebung misst, sondern auch gleich für die gewünschte Reaktion sorgt. Die Kategorien Eingabe, Verarbeitung und Ausgabe werden hier gedanklich vermischt. Dies äußert sich z. B. in kreativen Wortschöpfungen, wie „Alarmsensor“.

Durch das Erstellen von Programmen für den Calliope erfahren die Schülerinnen und Schüler, dass sie die Aktionen, die der Calliope ausführen soll, selbst festlegen müssen und dass diese an Bedingungen geknüpft sein können, welche die Eingabewerte der Sensoren erfüllen müssen. Dahinter steckt das EVA-Prinzip, ein grundlegendes Prinzip der Informatik, das die Fehlvorstellung der Schülerinnen und Schüler ersetzen sollte.

Es lohnt sich daher, dieses Prinzip nach dem Abschnitt Sensoren oder am Ende der gesamten Einheit einmal explizit zu thematisieren und transparent zu machen. Alle Beispiele, welche die Schülerinnen und Schüler kennengelernt haben, können zu diesem Prinzip abstrahiert werden. Dies kann sowohl auf der Ebene des Programms als auch auf der Ebene der Hardwarebauteile des Calliope erfolgen.

Das Arbeitsblatt *05_AB_EVAPrinzip* stellt dazu einige Aufgaben zur Verfügung.

Besonders interessant ist auf der Hardwareebene die Einordnung des Funkmoduls, da es sich bei den Funkwellen sowohl um Eingaben als auch um Ausgaben handelt, je nachdem ob man den Empfänger oder den Sender betrachtet. Auch über die Pins können sowohl Eingaben gelesen als auch Ausgaben geschrieben werden. Dies ist den Schülerinnen und Schülern aber natürlich nur bewusst,

wenn die Pins im Unterricht entsprechend verwendet wurden. Aber auch die Zuordnung der anderen Bauteile gelingt in der Regel nicht allen Schülerinnen und Schülern problemlos.

Um transparent zu machen, dass der Calliope exemplarisch für ein Sensor-Aktor-System ist und sich die grundlegende Arbeitsweise nicht von komplexeren Systemen unterscheidet, bietet sich abschließend ein Vergleich mit Smartphone, Tablet und Desktop-PC an. In den Lerngruppen, in denen die Einheit getestet wurde, gelang es den Schülerinnen und Schülern sehr gut Gemeinsamkeiten herzustellen, in dem z. B. die LED-Matrix mit dem Bildschirm assoziiert oder das Funkmodul stellvertretend für die Möglichkeit, mit anderen Geräten zu kommunizieren, gesehen wurde.

Ausblick

Je jünger die Schülerinnen und Schüler sind, desto fantasievoller werden die algorithmischen Probleme, welche mit dem Calliope bearbeitet werden, in kleine Geschichten verpackt. Dies kann durch die Integration des Calliope in eine reale bzw. passend konstruierte Umgebung noch unterstützt und gefördert werden.

Sind die Schülerinnen und Schüler mit der Konstruktion und Implementierung von Sensor-Aktor-Systemen vertraut, kann der Calliope auch nur oder erneut für Projekte in anderen Modulen des Kerncurriculums (s. [8]) eingesetzt werden. Etwa zur Erprobung von Codierungen und Kommunikationsprotokollen im Modul *Datenaustausch in Netzwerken* aus dem Lernfeld „Daten und ihre Spuren“ oder im Modul *Bildbearbeitung* aus dem Lernfeld „Computerkompetenz“ zur experimentellen Erarbeitung des RGB-Modells (vgl. [8], S. 16ff.).

Literaturverzeichnis

- [1] Calliope gGmbH (2020). *Calliope. Unsere Mission*. <https://calliope.cc/idee/mission> [Datum des Zugriffs: 14.02.2020]
- [2] Calliope gGmbH (2020). *Calliope. Programmierumgebungen*. <https://calliope.cc/programmieren/editoren> [Datum des Zugriffs 24.05.2022]
- [3] Calliope.cc *Calliope mini*. <https://calliope-mini.github.io/> [Datum des Zugriffs: 14.02.2020]
- [4] Micro:bit Educational Foundation. *micro:bit* <https://microbit.org/de/> [Datum des Zugriffs: 06.02.2020]
- [5] Microsoft (2018). *Microsoft MakeCode. Version 4.0.29* <https://makecode.calliope.cc> [Datum des Zugriffs: 06.01.2023]
- [6] Microsoft (2017). *MakeCode offline verwenden*. <https://makecode.calliope.cc/offline> [Datum des Zugriffs: 14.02.2020]
- [7] MIT Lifelong-Kindergarten-Group. *Scratch*. <https://scratch.mit.edu> [Datum des Zugriffs: 16.02.2021]
- [8] Niedersächsisches Kultusministerium (2014). *Kerncurriculum für die Schulformen des Sekundarbereichs I Schuljahrgänge 5 – 10. Informatik*. Hannover: Unidruck

Lizenz

Dieses Werk ist lizenziert unter einer [Creative Commons Namensnennung - Nicht kommerziell - Keine Bearbeitungen 4.0 International Lizenz](https://creativecommons.org/licenses/by-nc-nd/4.0/).