



Elektronik und Verbesserung der
Beschäftigungsfähigkeit von
Erwachsenen mit Down-
Syndrom

2023-1-LV01-KA220-ADU-000160601



Kofinanziert von der
Europäischen Union

WP3.2

SCHULUNGS

FORMAT

Das Trainingsformat basiert auf einer experimentellen Herangehensweise an Elektronik und bietet der Zielgruppe eine anregende Lernumgebung. Die Lernenden können in ihrem eigenen Tempo lernen und sich dabei schrittweise spezifische Kompetenzen, Informationen und Fertigkeiten aneignen, je nach gewähltem Ansatz.



EFEKTAS
GROUP

NGO NEST
Berlin

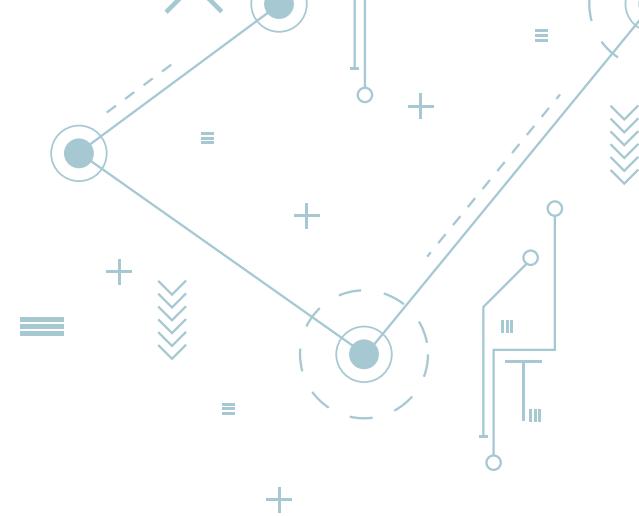
Quality Culture





Elektronik und Verbesserung der
Beschäftigungsfähigkeit von
Erwachsenen mit Down-
Syndrom

2023-1-LV01-KA220-ADU-000160601



ELEKTRONIK UND VERBESSERUNG DER BESCHÄFTIGUNGSFÄHIGKEIT VON ERWACHSENEN MIT DOWN- SYNDROM (FEAT-DS)

Gefördert von der Europäischen Union. Die geäußerten Ansichten und Meinungen sind jedoch ausschließlich die der Autoren und spiegeln nicht notwendigerweise die der Europäischen Union oder der Exekutivagentur für Bildung und Kultur (EACEA) wider. Weder die Europäische Union noch die EACEA übernehmen dafür die Verantwortung.



**Kofinanziert von der
Europäischen Union**

Zusammenfassung

Dieses Trainingsformat wurde im Rahmen von FEAT-DS – Facilitating Electronics, Accessibility and Tinkering for Down Syndrome Adults – entwickelt, einem europäischen Projekt zur Verbesserung des Zugangs zu technologischem Lernen für Erwachsene mit Down-Syndrom. Im Fokus steht die Entwicklung barrierefreier, hochwertiger Bildungsressourcen, die praktisches Experimentieren und erste Programmierübungen integrieren. Das Trainingsformat bietet einen strukturierten und flexiblen Lernpfad, der grundlegende Konzepte der Elektronik, einfache Schaltungslogik und Programmierung vermittelt. Mithilfe von Schritt-für-Schritt-Anleitungen, geführten Übungen und interaktiven Spielen auf der Webplattform des Projekts können die Lernenden praktische Kompetenzen in ihrem eigenen Tempo entwickeln. Das übergeordnete Ziel besteht darin, Werkzeuge bereitzustellen, die erfahrungsorientiertes Lernen unterstützen, die Selbstständigkeit fördern und die Teilhabe von Erwachsenen mit Down-Syndrom an Bildungs- und Beschäftigungsmöglichkeiten stärken.

Das FEAT-DS-Projekt befasst sich mit den anhaltenden Hürden, die Erwachsene mit Down-Syndrom beim Zugang zu technischer und digitaler Bildung überwinden müssen. In einer transnationalen Forschungsphase ermittelten die Partner den Bildungsbedarf der Zielgruppe und analysierten bestehende Weiterbildungsangebote in den einzelnen Teilnehmerländern. Diese Erkenntnisse bildeten die Grundlage für die Entwicklung relevanter, praxisnaher und bedarfsgerechter Schulungsmaterialien. Erwachsenenbildner und Elektronikspezialisten wirkten gemeinsam am Entwicklungsprozess mit und integrierten so pädagogische Expertise mit technischem Wissen. Diese Synergie ermöglicht die Erstellung von Materialien, die sowohl zugänglich als auch technisch fundiert sind und von Lehrkräften, die mit Erwachsenen mit Down-Syndrom arbeiten, sinnvoll eingesetzt werden können. Das hier vorgestellte Schulungsformat umfasst:

- Thematische Module, die schrittweise entwickelt wurden;
- Aktivitäten, die grundlegende Elektronik, logisches Denken und erste Programmieraufgaben kombinieren;
- Klare Anweisungen zur Unterstützung der Implementierung und Replikation;
- Direkte Links zu den interaktiven Programmierspielen, die auf der FEAT-DS-Webplattform gehostet werden.

INHALTSVERZEICHNIS

Modul 1: Praktisches Erkunden: Bauen durch Sehen und Tasten	5
Modul 2: Klare Kommunikation in der Elektronik: Die Sprache von Schaltkreisen und Code sprechen	29
Modul 3: Gemeinsam lernen: Vernetzen und Unterstützung erhalten	60
Modul 4: Elektronik in unserer Welt: Vom Hobby zu den Möglichkeiten	82
Modul 5: Individuell gestalten – Personalisierung und spielerisches Feedback	105





MODUL 1: PRAXISERLEBNIS: BAUEN DURCH SEHEN UND TASTEN



Kofinanziert von der
Europäischen Union

Überblick

Dieses Modul heißt „Praktisches Erkunden: Bauen durch Sehen und Tasten“. Hier geht es um Lernen durch praktisches Tun. Elektronik versteht man am besten, indem man die Bauteile berührt, ihre Verbindungen beobachtet und ihre Funktion erforscht. Diese praktische Lernmethode nennt man „Tüfteln“. Tüfteln bedeutet, mit Werkzeugen und Materialien zu experimentieren und ihre Funktionsweise zu verstehen. Beim Tüfteln gibt es keine Fehler, nur neue Entdeckungen!

Wir beginnen mit einigen grundlegenden elektronischen Bauteilen. Sie lernen deren Namen und Funktion kennen. Anschließend bauen Sie mit diesen Bauteilen einfache Schaltungen, sogenannte Stromkreise. Sie bringen eine Lampe zum Leuchten, einen Summer zum Klingeln und einen Motor zum Drehen. Sie werden sofort sehen, wie Ihre Handlungen etwas bewirken. Das hilft uns zu verstehen, wie alles zusammenwirkt.

Später im Modul werden wir uns auch mit einfachen Programmieraufgaben beschäftigen. Programmieren bedeutet, einem Computer Anweisungen zu geben. Wir verwenden große, farbige Blöcke auf dem Bildschirm, die einfach zu bedienen sind. Du schreibst einen einfachen Code, um eine Lampe blinken zu lassen. So siehst du, wie Anweisungen auf einem Computer Dinge in der realen Welt steuern können. Dieses Modul ist so gestaltet, dass es Spaß macht, anschaulich ist und einen idealen Einstieg in die Elektronik bietet.

Ziele

- Lerne, dass man neue Dinge am besten versteht, indem man sie selbst tut, sieht und berührt.
- Lerne die Namen und Funktionen einiger grundlegender elektronischer Bauteile kennen.
- Verstehe das Grundprinzip eines Stromkreises.
- Fühle dich selbstbewusst und erfolgreich, indem du Dinge baust, die wirklich funktionieren.
- Entdecke, wie sich elektronische Bauteile durch einfache Programmierung steuern lassen.



Erwartete Ergebnisse

Nach diesem Modul kannst du:

- eine Batterie, eine LED-Lampe und einen Schalter erkennen und benennen.
- erklären, warum ein Stromkreis einen geschlossenen Pfad benötigt.
- eine Lampe mithilfe einer Batterie und eines Schalters ein- und ausschalten.
- einen einfachen Stromkreis bauen, der einen Summer zum Tönen bringt.
- mit einfachen Codeblöcken eine echte LED-Lampe zum Blinken bringen.



1. DIE GROSSE IDEE – LERNEN MIT DEN HÄNDEN

Lernen durch Handeln

Hast du jemals gelernt, deine Schuhe zu binden? Oder wie man ein Sandwich zubereitet? Wahrscheinlich hast du es gelernt, indem du jemandem zugeschaut und es dann selbst ausprobiert hast. Du hast deine Hände und deine Augen benutzt und geübt.

Das ist der beste Weg, neue Dinge zu lernen! Wir nennen es Lernen durch Handeln oder erfahrungsorientiertes Lernen. [1]

In diesem Modul lernen wir Elektronik auf dieselbe Weise kennen. Wir lesen nicht nur darüber, sondern bauen Dinge. Wir fassen die Bauteile an und beobachten, was passiert, wenn wir sie verbinden. Das ist eine besonders effektive Lernmethode, denn man sieht die Ergebnisse der eigenen Arbeit sofort. Es macht Spaß und hilft, das Gelernte besser zu behalten.

Was ist ein Stromkreis?

Alles, was wir in diesem Modul bauen, ist ein Stromkreis. Ein Stromkreis ist einfach ein Weg, auf dem Strom fließen kann. Stellen Sie sich eine Rennstrecke vor. Die Rennwagen müssen die gesamte Strecke abfahren, um das Rennen zu beenden. Wenn die Strecke unterbrochen ist, müssen die Wagen anhalten.

Elektrizität ist wie Rennwagen. Sie braucht einen geschlossenen Weg, um von ihrem Ausgangspunkt zu gelangen, eine Aufgabe zu erfüllen (wie zum Beispiel eine Lampe zum Leuchten zu bringen) und dann wieder zu ihrem Ausgangspunkt zurückzukehren.

Ein einfacher Stromkreis besteht im Wesentlichen aus drei Elementen:

- 1. Eine Stromquelle. Hier kommt die Energie her. Für uns ist das eine Batterie.**
- 2. Ein Pfad. Entlang dieses Pfades fließt der Strom. Für uns sind das Drähte.**
- 3. Die Last. Das ist das Bauteil, das eine Funktion erfüllt. Es könnte sich um eine Lampe, einen Summer oder einen Motor handeln.**

Wenn alle diese Teile einen geschlossenen Kreis bilden, entsteht ein geschlossener Stromkreis. Und dann geschieht die Magie! Das Licht leuchtet, der Summer ertönt oder der Motor dreht sich. Gibt es eine Unterbrechung im Stromkreis, ist dieser unterbrochen und der Strom kann nicht fließen.



Erwartete Ergebnisse

Nach diesem Modul kannst du:

- eine Batterie, eine LED-Lampe und einen Schalter erkennen und benennen.
- erklären, warum ein Stromkreis einen geschlossenen Pfad benötigt.
- eine Lampe mit Batterie und Schalter ein- und ausschalten.
- einen einfachen Stromkreis bauen, der einen Summer zum Tönen bringt.
- mit einfachen Codeblöcken eine LED-Lampe zum Blinken bringen.



2. LERNEN SIE IHRE WERKZEUGE KENNEN – EIN VISUELLES LEITFADEN FÜR ELEKTRONISCHE BAUTEILE

Für unsere Übungen verwenden wir einige gängige elektronische Bauteile. Diese Bauteile sind groß und einfach zu handhaben. Lernen wir sie kennen!

Die Batterie



Abbildung 1. Batterie

Das ist eine BATTERIE.

Seine Aufgabe ist es, unseren Stromkreis mit Strom zu versorgen. Es ist wie die Nahrung, die uns die Energie zum Laufen und Spielen gibt.

Unsere Batterie hat zwei Seiten: einen Pluspol (+) und einen Minuspol (-). Das rote Kabel wird üblicherweise an den Pluspol und das schwarze Kabel an den Minuspol angeschlossen.



Die LED

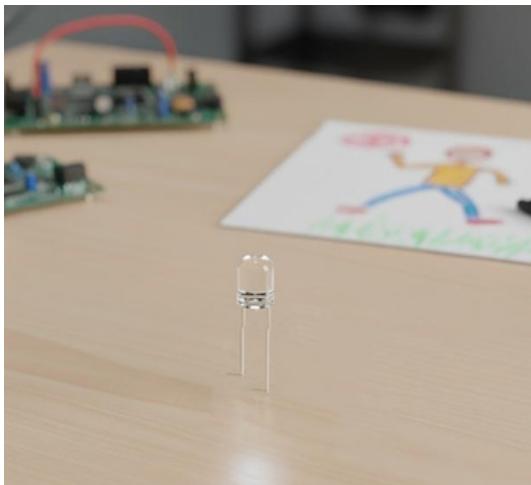


Abbildung 2. LED

Das ist eine LED. LED steht für Leuchtdiode. Ihre Aufgabe ist es, Licht zu erzeugen. Es ist eine kleine Glühbirne, die hell leuchtet, wenn Strom durch sie fließt.

LEDs haben zwei Anschlüsse. Ein Anschluss ist etwas länger als der andere. Das ist wichtig! Der längere Anschluss muss an den Pluspol (+) der Batterie angeschlossen werden, der kürzere an den Minuspol (-).

Der Schalter

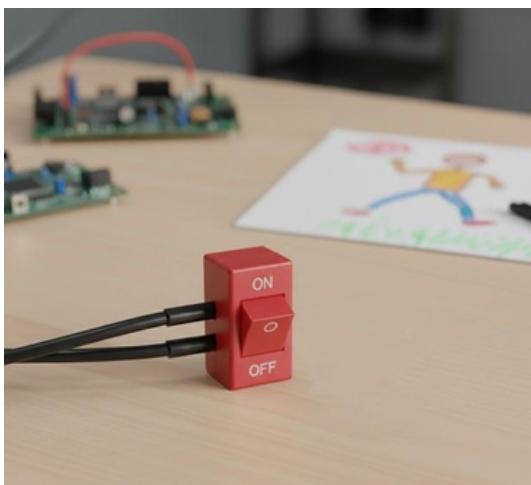


Abbildung 3. Schalter

Seine Aufgabe ist es, den Kreislauf zu öffnen und zu schließen. Es ist wie ein Tor auf der Rennstrecke.

Wenn der Schalter eingeschaltet ist, ist das Tor geschlossen und der Stromkreis unterbrochen. Strom kann fließen.

Wenn der Schalter ausgeschaltet ist, ist das Tor offen und der Stromkreis unterbrochen. Der Stromfluss wird unterbrochen.



Der Summer



Abbildung 4. Summer

Das ist ein BUZZER.

Seine Aufgabe ist es, ein Geräusch zu erzeugen. Wenn Strom durch es fließt, erzeugt es ein summendes Geräusch.

Wie die Batterie hat auch der Summer einen Pluspol (+) und einen Minuspol (-). Wir müssen die Drähte richtig anschließen, damit er funktioniert.

Der Motor



Abbildung 5. Motor

Das ist ein Motor. Seine Aufgabe ist es, sich zu drehen. Wenn Strom durch ihn fließt, dreht sich eine kleine Welle oben sehr schnell. Wir können Dinge wie einen Ventilator oder ein Rad daran befestigen.



3. SICHERHEIT UND UNTERSTÜTZUNG – WIE WIR ZUSAMMENARBEITEN

Neues zu lernen ist ein Abenteuer. Damit unser Abenteuer für alle sicher und unterhaltsam wird, werden wir gemeinsam ein paar einfache Regeln befolgen.

- **Arbeite mit einer Betreuungsperson zusammen. Deine Lehrkraft oder dein/e Betreuer/in ist da, um dich anzuleiten. Sie/Er kann dir helfen, wenn du nicht weiterkommst oder eine Frage hast.**
- **Verwende sichere Stromversorgung. Wir verwenden ausschließlich Niederspannungsbatterien (z. B. AA-Batterien). Strom aus der Steckdose wird niemals verwendet.**
- **Schalte zuerst aus. Bevor du Kabel oder Bauteile in deinem Stromkreis änderst, trenne immer zuerst die Batterie.**
- **Lass es überprüfen. Bevor du die Batterie zum ersten Mal an einen neuen Stromkreis anschließt, lass sie immer von deiner Betreuungsperson überprüfen. Dies ist die wichtigste Sicherheitsmaßnahme.**
- **Fehler sind okay! Jeder macht Fehler beim Lernen. Das gehört einfach zum Ausprobieren dazu. Wenn etwas nicht funktioniert, ist das kein Problem. Es ist eine Chance, etwas Neues zu lernen [2].**
- **Feiere deinen Erfolg! Wenn etwas funktioniert, sei stolz! Du lernst und erschaffst etwas Neues. Positive Gefühle helfen dir, noch besser zu lernen [3].**



4. LASST UNS BAUEN! – PRAKTISCHE BASTELAKTIVITÄTEN

Jetzt kommt der beste Teil: Wir bauen unsere eigenen Schaltkreise! Denk dran, es geht ums Tüfteln. Viel Spaß beim Ausprobieren und Entdecken – und seht, was ihr alles erschaffen könnt!

Aktivität 1: Bring ein Licht zum Leuchten!

In dieser Übung bauen wir unseren allerersten Stromkreis, um eine LED-Lampe einzuschalten.

Ziel dieser Aktivität:

- Um deinen ersten vollständigen Stromkreis aufzubauen.
- Um zu sehen, wie Elektrizität eine LED zum Leuchten bringt.

Was du benötigst (wir verwenden große, farbcodierte Teile, die leicht zu greifen und zu verbinden sind):

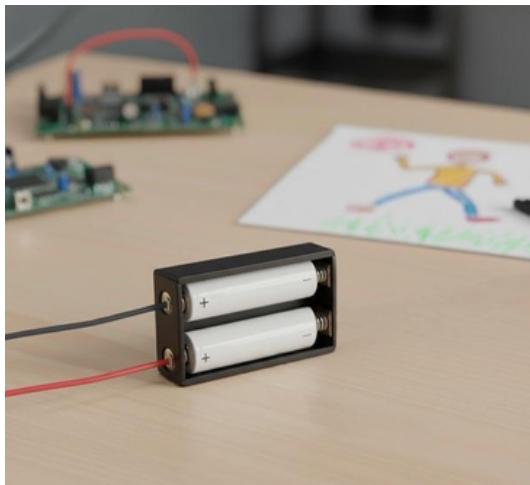


Abbildung 6.1 Batteriehalter mit Batterien

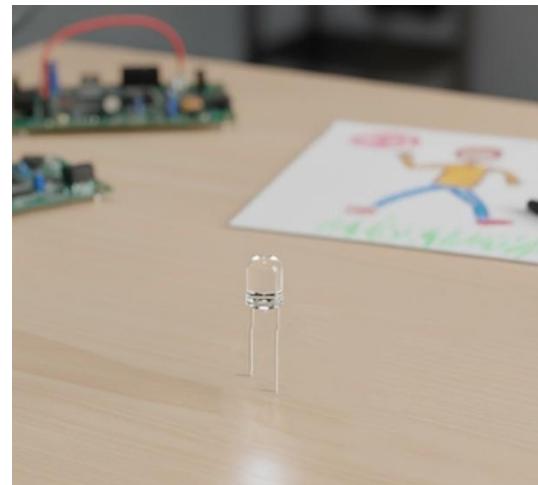


Abbildung 7. 1 LED (beliebige Farbe)



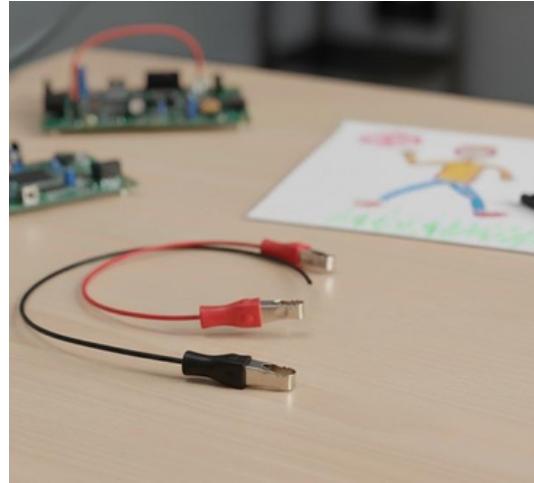


Abbildung 8.2 Drähte mit Klemmen (Krokodilklemmen)

Lasst es uns Schritt für Schritt aufbauen:

1. Hol deine Batterie.

Stelle sicher, dass sich die Batterien im Halter befinden. Du siehst ein rotes Kabel (+) und ein schwarzes Kabel (-), die herausragen.

2. Besorg dir deine LED.

Betrachte die beiden Schenkel. Finde den längeren Schenkel. Das ist der positive (+) Schenkel.

3. Schließe den ersten Draht an.

Nimm ein Kabel. Klemme ein Ende an das rote Kabel der Batterie. Klemme das andere Ende an den längeren Anschluss der LED.

4. Schließe den zweiten Draht an.

Nimm das zweite Kabel. Klemme ein Ende an das schwarze Kabel der Batterie.

5. Verlange einen Sicherheitscheck.

Bevor du mit dem letzten Schritt beginnst, bitte deine Betreuungsperson, deine Schaltung zu überprüfen, damit sie korrekt ist.

6. Schließe den Stromkreis!

Klemme nun das andere Ende des zweiten Drahtes an den kurzen Anschluss der LED.

Hinweis für Lehrkräfte: Leuchtet die LED nicht, liegt das meist an einer falschen Polung. Nutzen Sie dies als Lernmoment. Weisen Sie den Lernenden an, die Batterie abzuklemmen, die LED umzudrehen und es erneut zu versuchen. Dies verdeutlicht das Prinzip der Polarität.

Erfolg!

Deine LED sollte jetzt hell leuchten! Du hast einen kompletten Stromkreis aufgebaut. Falls sie nicht funktioniert, keine Sorge! Überprüfe einfach, ob alle Klemmen die Metallteile berühren, und versuche es erneut.

Was haben wir gelernt?

Wir haben gelernt, dass für das Aufleuchten der LED ein geschlossener Stromkreis von der Batterie durch die LED und zurück zur Batterie benötigt wird.

Fordere dich selbst heraus!

- Kann man das Licht auch ohne Kabel zum Leuchten bringen?
- Versuchen Sie es mit einer andersfarbigen LED. Funktioniert es dann genauso?

Aktivität 2: Mach ein Geräusch!

Dieses Mal bauen wir einen Stromkreis, der mit Hilfe eines Summer und eines Schalters einen Ton erzeugt.

Ziel dieser Aktivität:

- Einen Schalter zur Steuerung eines Stromkreises verwenden.
- Um einen Stromkreis zu bauen, der ein Geräusch erzeugt.

Was du benötigst (wir verwenden große, farbcodierte Teile, die leicht zu greifen und zu verbinden sind):



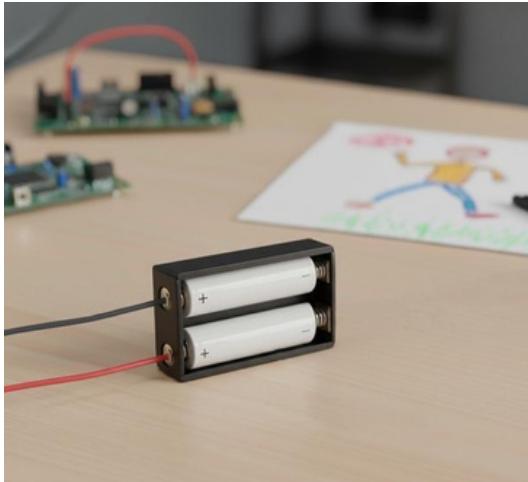


Abbildung 9.1 Batteriehalter mit Batterien

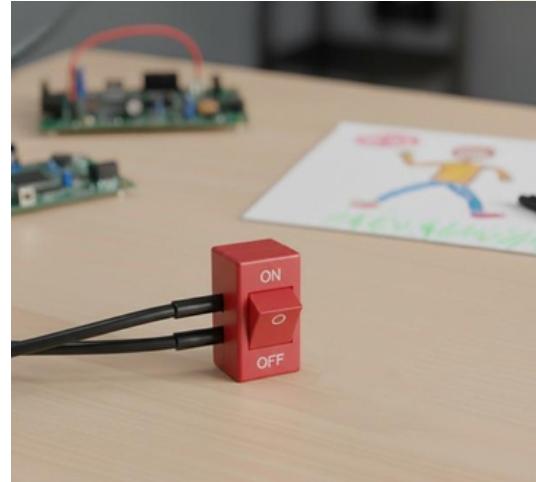


Abbildung 11.1 Schalter



Abbildung 10.1 Summer

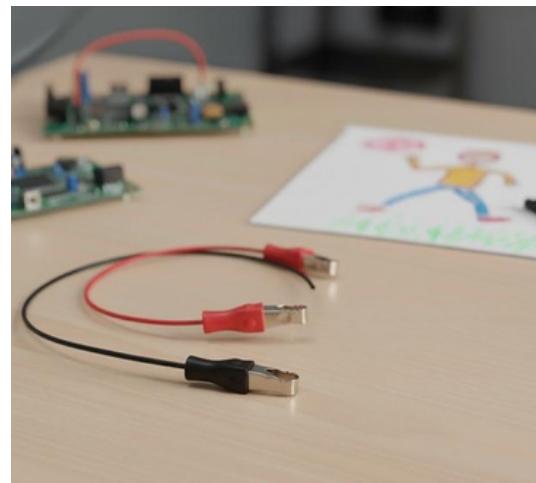


Abbildung 12. 3 Drähte mit Klemme

1. Schließe die Batterie an den Schalter an.

Nimm ein Kabel. Klemme ein Ende an das rote Kabel der Batterie. Klemme das andere Ende an einen der Metallkontakte des Schalters.

2. Schließe den Schalter an den Summer an. Nimm ein zweites Kabel. Klemme ein Ende an den anderen Metallanschluss des Schalters. Klemme das andere Ende an das rote Kabel des Summers.

3. Schließe den Summer wieder an die Batterie an.

Nimm das dritte Kabel. Klemme ein Ende an das schwarze Kabel des Summers. Klemme das andere Ende an das schwarze Kabel der Batterie.

4. Verlange einen Sicherheitscheck.

Bitte deine Betreuungsperson, sich deinen Stromkreis anzusehen, bevor du ihn einschaltest.

5. Den Schalter umlegen!

Drücke oder klappe nun den Schalter in die Position EIN.



Erfolg!

BZZZZZZ! Der Summer piept. Du hast einen Schalter benutzt, um den Stromkreis zu steuern. Wenn du den Schalter EIN schaltest, ist der Stromkreis geschlossen. Wenn du ihn AUS schaltest, hört das Piepen auf.

Was haben wir gelernt?

Wir haben gelernt, dass ein Schalter wie ein Tor funktioniert. Er kann den Stromfluss öffnen oder schließen.

Fordere dich selbst heraus!

- Kann man den Summer durch die LED aus der ersten Aktivität ersetzen?
- Funktioniert der Schalter noch zum Ein- und Ausschalten des Lichts?

Aktivität 3: Bring etwas zum Drehen!

Für unsere letzte Bastelaktivität wollen wir etwas in Bewegung setzen! Wir werden einen Motor verwenden, um einen kleinen Ventilator anzutreiben.

Ziel dieser Aktivität:

- Einen Stromkreis bauen, der etwas in Bewegung setzt.
- Um zu sehen, wie Elektrizität in Bewegung umgewandelt werden kann.

Was du benötigst (wir verwenden große, farbcodierte Teile, die leicht zu greifen und zu verbinden sind):



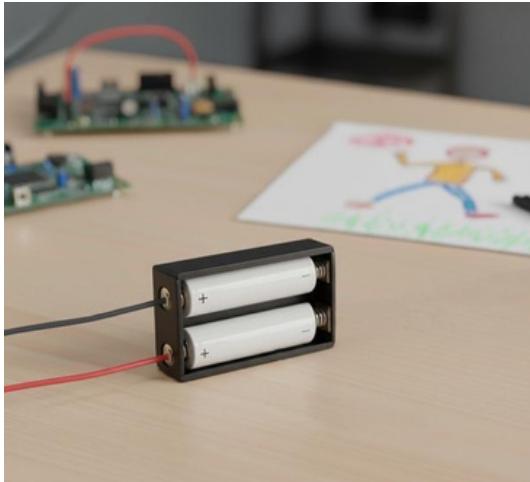


Abbildung 13.1 Batteriehalter mit Batterien



Abbildung 14.1 Motor mit angeschlossenem Lüfter

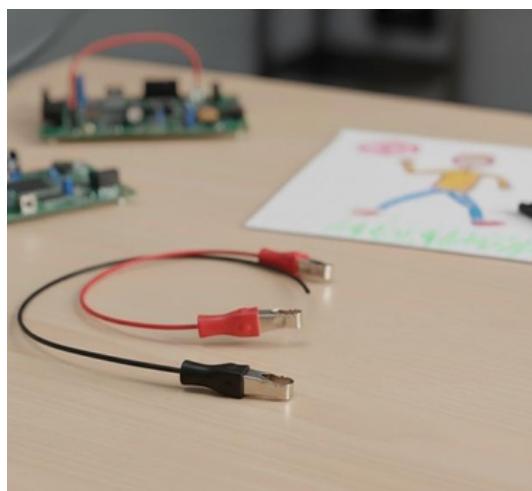


Abbildung 15. 2 Drähte mit Klemmen

Lasst es uns Schritt für Schritt aufbauen:

1. Schau dir den Motor an.

Du siehst zwei kleine Metalllaschen auf der Rückseite des Motors. Hier werden wir unsere Kabel anschließen.

2. Schließe den ersten Draht an.

Nimm ein Kabel. Klemme ein Ende an das rote Kabel der Batterie. Klemme das andere Ende an eine der Metalllaschen am Motor.

3. Schließe den zweiten Draht an.

Nimm das zweite Kabel. Klemme ein Ende an das schwarze Kabel der Batterie.

4. Schließe den zweiten Draht an.

Nimm das zweite Kabel. Klemme ein Ende an das schwarze Kabel der Batterie.

5. Verlange einen Sicherheitscheck.

Bevor du mit dem letzten Schritt beginnst, bitte deine Betreuungsperson, deine Schaltung zu überprüfen, damit sie korrekt ist.

6. Schließe den Stromkreis!

Klemme das andere Ende des zweiten Drahtes an den kurzen Anschluss der LED.

Erfolg!

Der Ventilator dreht sich! Du hast einen Stromkreis gebaut, der elektrische Energie in Bewegungsenergie umwandelt. Falls er sich nicht dreht, überprüfe den Kontakt der Klemmen und versuche es erneut.

Was haben wir gelernt?

Wir haben gelernt, dass Elektrizität mehr kann als nur Licht und Ton erzeugen. Sie kann auch Dinge bewegen!

Fordere dich selbst heraus!

Batterie abklemmen. Die roten und schwarzen Drähte an den Motoranschlüssen vertauschen.

- Was passiert mit dem Lüfter, wenn man die Batterie wieder anschließt?
- Dreht es sich in eine andere Richtung?



5. DEM COMPUTER SAGEN, WAS ER TUN SOLL! - EINE EINFACHE PROGRAMMIERÜBUNG

Du hast gelernt, wie man mit den Händen Schaltkreise baut. Jetzt lernen wir, wie man einen Schaltkreis mit einem Computer steuert. Das nennt man Programmieren.

Was ist Blockcodierung?

Wir verwenden eine spezielle Art der Programmierung, die sogenannte Blockprogrammierung. Anstatt Wörter einzugeben, verwendet man farbige Blöcke, die wie Puzzleteile aussehen. Jeder Block ist eine Anweisung für den Computer. Man setzt die Blöcke zusammen, um eine Reihe von Anweisungen zu erstellen, ein sogenanntes Programm [4].

Aufgabe 4: Programmiere ein blinkendes Licht!

In dieser Übung schreiben wir ein einfaches Programm, das eine LED-Lampe von selbst blinken lässt.

Ziel dieser Aktivität:

- Um dein erstes Computerprogramm mithilfe von Codeblöcken zu schreiben.
- Um zu sehen, wie Code ein reales elektronisches Bauteil steuern kann.

Was du benötigst:





Abbildung 16. Ein Computer mit geöffneter Codierungssoftware

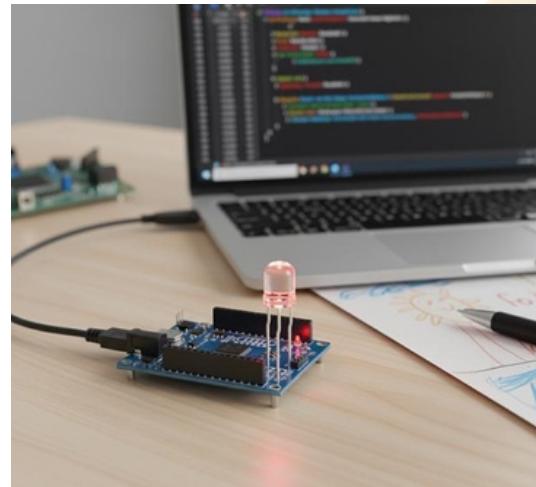


Abbildung 17. Eine an die Platine angeschlossene LED-Leuchte.



Abbildung 18. Eine spezielle Platine (wie ein Arduino oder Micro:bit), die über ein USB-Kabel mit dem Computer verbunden ist.

Ihr/e Lehrer/in wird Ihnen bei der Einrichtung helfen. Die Einrichtung sollte in etwa so aussehen:

1. Finde den Block "Licht einschalten".
 Schau in der Liste der Codeblöcke nach.
 Finde denjenigen, der das Licht einschalten soll.

2. Ziehe den Block in deinen Arbeitsbereich.
 Klicke und ziehe den Block „Licht einschalten“ in den leeren Programmablaufkasten.

3. Finde den "wait"-Block.

Suche nun einen Block, der den Computer anweist, zu warten. Wir weisen ihn an, eine Sekunde zu warten. Ziehe ihn unter den ersten Block, bis er einrastet.

4. Finde den Block "Licht ausschalten".

Suche als Nächstes den Block, der das Licht ausschaltet. Ziehe ihn unter den „Warten“-Block und lass ihn dort einrasten.

Hinweis für Lehrkräfte: Dieser Schritt verbindet die digitale mit der physischen Welt. Betonen Sie diese Verbindung, indem Sie fragen: „Seht ihr, wie die Blöcke, die ihr auf den Bildschirm legt, dem realen Licht Anweisungen geben?“

5. Füge einen weiteren "wait"-Block hinzu.

Das Licht soll für einen Moment ausgeschaltet bleiben. Füge ganz unten in deinem Programm einen weiteren „Warte 1 Sekunde“-Block ein.

6. Starte dein Programm!

Klicke auf dem Bildschirm auf die Schaltfläche „Ausführen“ oder „Starten“. Schau dir die LED-Leuchte auf der Platine an.

Erfolg!

Deine LED blinkt! Sie leuchtet eine Sekunde lang, dann ist sie eine Sekunde lang aus, immer wieder. Du hast ein Programm geschrieben! Du hast dem Computer Anweisungen gegeben, und der Computer steuert ein reales Objekt.

Was haben wir gelernt?

Wir haben gelernt, dass Code eine Reihe von Anweisungen ist. Mithilfe von Blockprogrammierung können wir elektronischen Geräten mitteilen, was sie wann tun sollen.

Fordere dich selbst heraus!

- Kannst du die Zahlen in den „Warte“-Blöcken ändern? Versuche, das Licht schneller oder langsamer blinken zu lassen.



6. SPIELEN WIR EIN SPIEL! – BESCHREIBUNG EINES PROGRAMMERSPIELS

Um das Gelernte über Schaltkreise zu üben, können Sie ein unterhaltsames Spiel auf dem Computer oder Tablet spielen. So funktioniert das Spiel.

Spieltitel: Schließe den Rennparcours ab!

Lernziel: Zu lernen, dass ein Stromkreis einen geschlossenen Pfad benötigt, um zu funktionieren.

Spielkonzept: Das Spiel zeigt auf der einen Seite des Bildschirms eine Batterie und auf der anderen eine Glühbirne. Dazwischen befinden sich Lücken im Stromkreis. Am unteren Bildschirmrand erscheinen Drahtstücke und ein Schalter. Die Aufgabe des Spielers besteht darin, die Teile vom unteren Bildschirmrand in die Lücken zu ziehen, um den Stromkreis zu schließen.

Spielanleitung:

- Schau dir den Schaltplan auf dem Bildschirm an. Du wirst sehen, wo der Stromkreis unterbrochen ist.
- Schau dir die Teile am unteren Bildschirmrand an (Drähte, Schalter).
- Zieh ein Teil mit dem Finger oder der Maus in eine Lücke.
- Lass los, damit das Teil in die Lücke fällt.
- Fahre so fort, bis alle Lücken gefüllt und der Pfad fertiggestellt ist.
- Klicke zum Schluss auf den Schalter, um ihn einzuschalten.

Feedback: Wenn du den Stromkreis erfolgreich schließt und den Schalter betätigst, erhältst du eine tolle Belohnung! Die Glühbirne auf dem Bildschirm leuchtet hellgelb auf. Ein fröhlicher Ton ertönt, wie ein „Ding!“ oder eine kurze, heitere Melodie. Noch spannender wird es, wenn eine echte Elektronikplatine an den Computer angeschlossen wird. Dann leuchtet die LED auf der Platine gleichzeitig mit der auf dem Bildschirm auf! So wird deutlich, wie das Spiel mit der realen Welt verbunden ist.



7. SCHLUSSFOLGERUNG

Herzlichen Glückwunsch! Du hast Modul 1 abgeschlossen.

Sieh dir an, was du alles geschafft hast! Du hast gelernt, was ein Stromkreis ist, indem du ihn selbst gebaut hast. Du hast eine Lampe zum Leuchten gebracht, einen Summer zum Klingeln und einen Motor zum Drehen. Du bist sogar zum Programmierer geworden und hast ein Programm geschrieben, um eine Lampe zu steuern.

Du hast den ersten großen Schritt in die faszinierende Welt der Elektronik getan. Du hast bewiesen, dass du ein Tüftler, ein Denker und ein Schöpfer bist. Du kannst sehr stolz auf dich sein.

Wir hoffen, ihr hattet viel Spaß beim Tüfteln und Entdecken. Bleibt neugierig, stellt weiterhin Fragen und baut fleißig weiter. Wir freuen uns darauf, euch im nächsten Modul wiederzusehen – für noch mehr spannende Elektronik-Erlebnisse!



8. WEITERFÜHRENDE LITERATUR

Wenn es Ihnen Spaß gemacht hat und Sie noch mehr entdecken möchten, finden Sie hier einige tolle Online-Ressourcen. Fragen Sie Ihre Lehrkraft, ob sie Ihnen dabei helfen kann.

- **Tinkercad-Schaltungen:** <https://www.tinkercad.com/circuits>

Dies ist eine kostenlose Website, auf der man am Computer Schaltkreise bauen und testen kann. Sie ist wie eine digitale Version der Aktivitäten, die wir durchgeführt haben, und ist sehr farbenfroh und benutzerfreundlich.

- **Scratch:** <https://scratch.mit.edu>

Dies ist eine sehr beliebte Website für Blockprogrammierung. Man kann Geschichten, Spiele und Animationen erstellen, indem man Blöcke zusammensetzt. Es ist eine tolle Möglichkeit, Programmierkenntnisse auf spielerische und visuelle Weise zu üben.

- **Code.org – Aktivitäten der Hour of Code:**
<https://code.org/hourofcode/overview>

Diese Website bietet viele unterhaltsame, einstündige Programmierspiele und -tutorials. Viele davon verwenden Charaktere, die du vielleicht aus Filmen und Spielen kennst. Sie eignen sich hervorragend für Anfänger.

- **YouTube-Kanal: Simple Electronics:**
<https://www.youtube.com/c/SimpleElectronics>

Dieser Kanal bietet zahlreiche Videos, die zeigen, wie man einfache und unterhaltsame Elektronikprojekte baut. Die Videos sind übersichtlich gestaltet und erklären jeden Schritt – ideal für visuelle Lerntypen.

- **Assistive Technology Internet Modules (ATIM):** <https://atinternetmodules.org>

Diese Website ist eine hervorragende Ressource für Pädagogen und Familien. Sie bietet kostenlose, flexible Lernmodule mit Videos und Fallstudien, die Nutzern helfen, mehr über Assistenz- und Unterstützungstechnologien für Menschen mit Behinderungen zu erfahren.



9. LITERATURVERZEICHNIS

- https://lvp.digitalpromiseglobal.org/content-area/adult-learnerestrategies/experiential-learning-adult-learner/summary_
- <https://www.down-syndrome.org/en-gb/library/news-update/06/1/inclusive-education-individuals-down-syndrome/>
- <https://www.positiveaction.net/blog/teaching-students-with-down-syndrome-strategies>
- <https://codakid.com/block-coding/>
- <https://www.tinkercad.com/circuits>
- <https://scratch.mit.edu>
- <https://code.org/hourofcode/overview>
- <https://www.youtube.com/c/SimpleElectronics>
- <https://atinternetmodules.org>



10. SELBSTBEWERTUNGSRASTER

Mal sehen, was du noch weißt! Wähle für jede Frage die beste Antwort aus. Die richtige Antwort ist **fett** gedruckt.

Nr.	Frage	A	B	C	D
1	Was ist die Aufgabe einer Batterie in einem Stromkreis?	Licht zu machen	Energie zu liefern	Geräusche zu machen	Sich zu drehen
2	Wie nennt man einen geschlossenen Pfad, durch den Strom fließt?	Eine Linie	Eine Box	Einen Stromkreis	Eine Straße
3	Welches Bauteil summt, wenn Strom hindurchfließt?	Eine LED	Ein Motor	Ein Schalter	Ein Summer (Buzzer)
4	Eine LED hat zwei Beinchen. Welches gehört an das positive (+) rote Kabel?	Das kurze Bein	Das lange Bein	Egal welches	Das gebogene Bein
5	Was macht ein Schalter?	Macht Dinge heller	Macht Dinge lauter	Öffnet und schließt den Stromkreis	Gibt dem Kreis mehr Power
6	Was bedeutet "Tinkering"?	Ein Buch lesen	Einem Lehrer zuhören	Durch Bauen und Erkunden lernen	Ein Video ansehen
7	Welches dieser Teile dreht sich, wenn du einen Stromkreis damit baust?	Eine Batterie	Eine LED	Ein Summer	Ein Motor
8	Was benutzt du beim Block-Coding, um Anweisungen zu geben?	Wörter tippen	Bilder zeichnen	Bunte Blöcke	Laut sprechen
9	Was ist das Wichtigste, bevor du den Schalter zum ersten Mal einschaltest?	Den Raum abdunkeln	Einem Freund Bescheid sagen	Eine Aufsichtsperson den Kreis prüfen lassen	Die Batterie schütteln
10	Was hat das Programm in unserer Coding-Aktivität mit der LED gemacht?	Blieb an	Blieb aus	Wurde heller	Blinkte an und aus



MODUL 2: KLARE KOMMUNIKATION IN DER ELEKTRONIK: DIE SPRACHE DER SCHALTUNGEN UND DES CODES SPRECHENS



Überblick

Dieses Modul widmet sich einem grundlegenden Konzept: Klare und verständliche Kommunikation ist der Schlüssel zum Erlernen von Elektronik und Programmierung. Für Erwachsene mit Down-Syndrom sind vereinfachte Anleitungen, visuelle Hilfsmittel und klar definierte Schritte nicht nur hilfreich, sondern unerlässlich für den Erwerb dieser technischen Fähigkeiten.

Studien in verschiedenen Ländern haben gezeigt, dass Herausforderungen wie das verbale Gedächtnis und das abstrakte Denken das Lernen erschweren können. Deshalb konzentriert sich unser Ansatz genau auf diese Aspekte. Mithilfe vereinfachter Sprache, unterstützt durch Piktogramme und visuelle Diagramme, helfen wir den Teilnehmenden, elektronische Anweisungen zu entschlüsseln und die grundlegende Logik der Programmierung zu verstehen.

Ziele

- Die Fähigkeit entwickeln, vereinfachte technische Anweisungen im Kontext von Elektronik und Programmierung zu interpretieren.
- Ziel ist es, die Erkennung und das Verständnis von Standard-Elektroniksymbolen, Piktogrammen und visuellen Diagrammen zu fördern.
- Ziel ist es, grundlegende Fähigkeiten im logischen Vorgehen und im schrittweisen Denken für technische Aufgaben zu entwickeln.
- Die Einführung der blockbasierten Programmierung als zugänglichen Einstieg in die Programmierung.
- Die Fähigkeit zur Selbstvertretung soll gefördert werden, um zu erkennen, wann Klärungsbedarf besteht und um effektive Fragen zu stellen.
- Ziel ist es, durch strukturierte und erfolgreiche Erfahrungen mit Elektronik und Programmierung technisches Selbstvertrauen aufzubauen.
- Die Fähigkeit entwickeln, vereinfachte technische Anweisungen im Kontext von Elektronik und Programmierung zu interpretieren.



Erwartete Ergebnisse

- Die Teilnehmer werden technische Anweisungen in vereinfachter Sprache lesen und interpretieren.
- Die Teilnehmer werden mindestens zehn gängige elektronische Bauteile und deren Schaltplansymbole erkennen und benennen können.
- Die Teilnehmer folgen mehrstufigen visuellen Anleitungen, um selbstständig einfache Schaltkreise aufzubauen.
- Die Teilnehmer werden die grundlegende Struktur der blockbasierten Programmierung verstehen und einfache Programme mit 5-10 Blöcken erstellen.
- Die Teilnehmer werden systematische Problemlösungsfähigkeiten demonstrieren, wenn Projekte nicht wie erwartet funktionieren.
- Die Teilnehmer werden ein gesteigertes Selbstvertrauen im Umgang mit technischen Aufgaben und in der Bewältigung von Herausforderungen zeigen.
- Die Teilnehmer werden effektive Kommunikationsstrategien entwickeln, um technische Fragen zu stellen und Hilfe anzufordern.
- Die Teilnehmer werden in ihrer eigenen Arbeit klare Kommunikationsprinzipien anwenden, unter anderem bei der Organisation von Materialien und der Überprüfung des Fortschritts.



1. WARUM KLARE KOMMUNIKATION WICHTIG IST

Wie barrierefreie Sprache den Weg zum technischen Lernen ebnet

Im Alltag sind wir es gewohnt, Anleitungen zu erhalten, sei es zum Befolgen eines Rezepts, zum Zusammenbauen eines Möbelstücks oder zur Bedienung eines Geräts. Damit eine Aufgabe Spaß macht, ist es wichtig, dass die Anleitungen einfach und intuitiv sind. Je komplexer, verwirrender oder unklarer sie sind, desto frustrierter und demotivierter fühlen wir uns und desto größer ist die Gefahr, dass wir das Begonnene nicht beenden können. Dies gilt umso mehr für die Programmierung und Elektronik, wo höchste Präzision erforderlich ist, denn selbst ein scheinbar kleiner Fehler kann eine Kettenreaktion auslösen und die korrekte Funktion eines Schaltkreises oder die Ausführung eines Programms verhindern.



Abbildung 1. Zwei Personen kommunizieren

Wie Studien in verschiedenen Ländern übereinstimmend belegen, ist die Fähigkeit, Anweisungen zu verstehen, für Erwachsene mit Down-Syndrom aufgrund spezifischer kognitiver Merkmale besonders wichtig. Die transnationale Analyse der Partner des FEAT-DS-Projekts in Litauen, Lettland, Italien, Deutschland und Bulgarien identifizierte ein reduziertes verbales Kurzzeitgedächtnis als häufige Schwierigkeit bei Erwachsenen mit Down-Syndrom.

Die Studie zeigte zwar Schwächen beim Lernen auf, wie beispielsweise das Verstehen langer und komplexer Sätze, hob aber auch Stärken hervor, die genutzt werden sollten: Die visuell-räumliche Wahrnehmung ist bei Menschen mit Down-Syndrom oft gut ausgeprägt. Das bedeutet, dass sie Informationen besser verstehen und behalten, wenn diese visuell statt ausschließlich verbal präsentiert werden. Die Kombination aus vereinfachter Sprache, visueller Unterstützung und taktiler Interaktion ermöglicht es Menschen mit Down-Syndrom, technische Fähigkeiten zu erwerben, die ihnen sonst verwehrt blieben.

Beispiel:

In einem herkömmlichen technischen Handbuch würde es beispielsweise heißen:
„Vor Beginn des Montagevorgangs muss sichergestellt werden, dass alle Komponenten identifiziert und systematisch nach ihrer funktionalen Kategorisierung geordnet wurden.“

Dieser Satz ist zwar formal korrekt, birgt aber viele Hürden: Er verwendet formelles und akademisches Vokabular, steht im Passiv, führt abstrakte Konzepte wie „funktionale Kategorisierung“ ein und vereint mehrere Ideen in einer komplexen Satzstruktur. Eine Person mit Down-Syndrom könnte Schwierigkeiten haben, die Kernaussage dieser Anweisung zu erfassen und sich bereits vor Beginn der Aufgabe überfordert fühlen.

Eine klare Umformulierung könnte lauten:

„Bevor du mit dem Bauen beginnst, sieh dir alle deine Teile an. Lege jedes Teil an seinen Platz auf dem Tisch. Das wird dir helfen, das zu finden, was du brauchst.“

Dieser Satz unterscheidet sich grundlegend, da er eine klare, alltagssprachliche Sprache verwendet und die Information in drei kurze, prägnante Sätze unterteilt. Darüber hinaus erklärt er nicht nur, was zu tun ist, sondern auch den Zweck dieser beiden Schritte. Dieser Ansatz gewährleistet, dass das Organisationsprinzip auch in zukünftigen Projekten Anwendung findet.



2. WARUM KLARE UND EINFACHE ANWEISUNGEN FÜR DEN ERFOLG UNERLÄSSLICH SIND

Täglich nutzen wir Anleitungen: Sie zeigen uns, wie wir kochen, reisen oder etwas Neues bauen. Stellen Sie sich Anleitungen wie eine kleine Landkarte vor: Sie führen uns Schritt für Schritt zum Ziel. Ist die Karte übersichtlich, können wir ihr problemlos folgen! Ist sie jedoch unübersichtlich oder verwirrend, verirren wir uns. Dasselbe passiert, wenn wir neue Fähigkeiten erlernen, insbesondere im Umgang mit Elektronik oder Computern. Oft gibt es viele Teile, Werkzeuge und Arbeitsschritte zu beachten. Manchmal ist das ganz schön verwirrend!

Sind die Anweisungen zu lang oder zu kompliziert, wissen wir möglicherweise nicht, wie es weitergehen soll. Das kann uns verunsichern, ermüden oder frustrieren. Sind die Anweisungen hingegen kurz, einfach und übersichtlich, ist alles leicht verständlich.

Doch was macht eine Anweisung zu einer klaren Anweisung?

Eine klare Anweisung erklärt uns nicht nur, was zu tun ist, sondern auch, warum wir es tun. Anstatt beispielsweise zu sagen: „Verbinde alle Drähte“, sollte eine klare Anweisung lauten: „Verbinde jeden Draht sorgfältig, damit der Stromkreis funktioniert.“ Wie Sie sehen, sagt uns diese Anweisung Folgendes:

1. Vorgehensweise: Alle Drähte anschließen.
2. Warum wir das tun: um den Schaltkreis zum Laufen zu bringen.

Auf diese Weise erinnern wir uns besser und fühlen uns motivierter und aufmerksamer für das, was geschieht.

Hinweis für Pädagogen:

Statt zu sagen: „Bevor du mit dem Bauen beginnst, stelle sicher, dass du alle Bauteile hast und sie in der richtigen Reihenfolge vorliegen“, können wir sagen: „Sieh dir alle deine Teile an. Lege jedes einzelne auf den Tisch. So siehst du, was du hast.“ Beide Varianten vermitteln dieselben Informationen, aber die zweite ist leichter verständlich und weniger stressig.



3. GEMEINSAM LERNEN UND DURCH VISUELLE EINBLICKE VERSTEHEN

Beim Lernen nutzen wir nicht nur unser Gehirn, sondern auch unsere Augen! Tatsächlich helfen unsere Augen dem Gehirn, Informationen schneller zu verstehen und zu behalten. Deshalb sind visuelle Hilfsmittel (wie Bilder, Symbole, Farben und Diagramme) beim Erlernen von Elektronik und Programmierung so wichtig.

In der Elektronik kann eine Abbildung zeigen, wie die Bauteile zusammenpassen. Man sieht, wo die Batterie platziert wird, wie die Drähte angeschlossen werden oder in welche Richtung ein Bauteil zeigen muss. Ein Symbol kann die Funktion jedes Teils verdeutlichen: eine kleine Glühbirne für „LED“, eine Zickzacklinie für „Widerstand“, ein Pluszeichen für „positiv“. Diese kleinen Symbole sind wie eine universelle Sprache: Man versteht sie auch ohne Worte. In der Programmierung zeigen visuelle Blöcke die Logik des Programms. Jeder Block hat eine Farbe, eine Form und eine Position in der Abfolge. Man sieht, wie Aktionen miteinander verknüpft sind: Ein Block startet eine Bewegung, ein anderer wiederholt sie, ein weiterer prüft eine Bedingung. Indem man sich die Blöcke ansieht, versteht man, wie der Computer „denkt“.

Visuelle Hilfsmittel machen das Lernen auch inklusiver!

Manchmal sind Worte schwer zu lesen oder zu behalten: Ein klares Bild oder ein farbiges Diagramm kann die gleichen Informationen leichter verständlich machen und Ihnen helfen, das Gesamtbild zu erfassen, bevor Sie beginnen, und jeden Schritt selbstbewusst auszuführen.

Bevor Sie beispielsweise mit einer Montage- oder Programmieraufgabe beginnen, können Sie sich eine Zeichnung ansehen, die alle Materialien zeigt. Anschließend können Sie jedes Teil an der gleichen Stelle auf Ihrem Tisch platzieren. Wenn Sie wissen, wie jedes Teil aussieht und wo es hingehört, können Sie viel ruhiger arbeiten und Fehler vermeiden.

Endlich macht Lernen mit Bildern Spaß! Sie bringen Farbe und Bewegung in den Prozess, und man kann die einzelnen Schritte wie eine Geschichte verfolgen, vom ersten Bild bis zum Endergebnis.

Das fesselt Ihre Aufmerksamkeit und hilft Ihnen, sich an das Gelernte zu erinnern.



AUFGABE 1: BAUE EINE EINFACHE TASCHENLAMPE

Jetzt ist es Zeit zu üben!

Sie bauen ein kleines elektronisches Gerät anhand einer bebilderten Schritt-für-Schritt-Anleitung. Ziel ist es, zu verstehen, wie klare Worte und deutliche Bilder zusammenwirken, um das Lernen zu erleichtern.

Was Sie benötigen:

- 1 kleine LED-Glühbirne
- 1 Batterie (AA oder Knopfzelle)
- 1 Batteriehalter (falls erforderlich)
- 1 kleiner Schalter (optional)
- 2 kurze Drähte
- Klebeband oder kleine Klammern
- Diese visuelle Anleitung (gedruckt ist besser)

Was Sie lernen werden:

- Üben Sie das Befolgen einfacher schriftlicher und visueller Anweisungen.
- Verstehen Sie das Prinzip eines Ablaufs: Schritt für Schritt führt eine Handlung zur nächsten.
- Erfahren Sie, wie visuelle Hilfsmittel komplexe Ideen verständlicher machen.
- Arbeiten Sie mit einem Partner zusammen, um klar zu kommunizieren und kleinere Probleme zu lösen.
- Schaffen Sie Vertrauen durch ein reales und sichtbares Ergebnis.



Schritt-für-Schritt-Anleitung:



Abbildung 2. Benötigte Materialien

1. Schauen Sie sich Ihre Materialien an.

Lege alle deine Teile auf den Tisch.

Stellen Sie sicher, dass Sie alles haben, bevor Sie beginnen.



Abbildung 3. LED

2. Suchen Sie die Batterie und die LED.

Beachten Sie, dass die LED zwei Beine hat, ein langes und ein kurzes.

Der längere Schenkel ist die positive Seite.





Abbildung 4. LED und Batterie

3. Schließen Sie die LED an die Batterie an.

Berühren Sie mit dem längeren Bein der LED den Pluspol der Batterie.

Berühre mit dem kürzeren Schenkel das negative Ende.

Sie sollten sehen, wie das Licht angeht.



Abbildung 5. Anschließen des Schalters

4. Wenn Sie einen Schalter haben:

Platzieren Sie den Schalter zwischen zwei der Anschlüsse, um den Stromkreis öffnen und schließen zu können.

Versuchen Sie, das Licht ein- und auszuschalten.



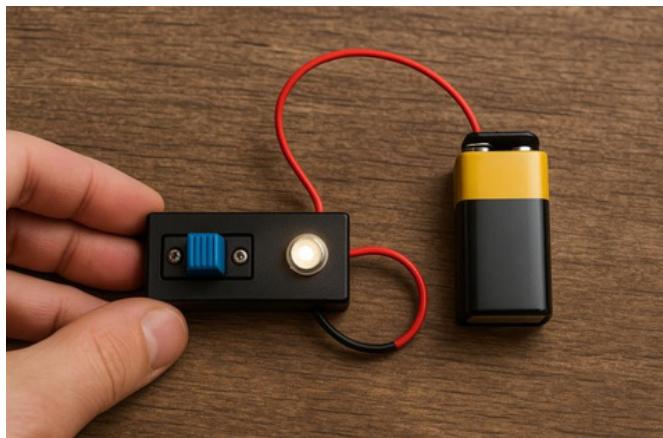


Abbildung 6. Sichern der Verbindungen

5. Sichern Sie die Verbindungen.

Wenn alles funktioniert, verwenden Sie Klebeband oder Klemmen, um die Kabel und die Batterie zusammenzuhalten.



Abbildung 7. Geschlossener Stromkreis

6. Beobachten Sie, was passiert.

Wenn der Stromkreis geschlossen ist, leuchtet die Lampe.

Wenn es geöffnet ist, schaltet sich das Licht aus.

Dies zeigt, wie sich der Strom durch die Bauteile bewegt.



4. EINFÜHRUNG IN DIE BLOCKBASIERTE PROGRAMMIERSPRACHE

Warum sollten wir die blockbasierte Programmiersprache verwenden?

Programmieren, oder Codieren, bedeutet im Grunde, einem Computer Anweisungen zu geben und jeden Schritt zu erklären. Jede Anweisung ist Teil eines Gesamtrezepts: Wir befolgen genau die Reihenfolge der Zutaten, und der Computer führt exakt das aus, was wir ihm sagen.

Aber warum haben die meisten Menschen Angst vor dem Programmieren?

In der traditionellen Programmierung werden diese Anweisungen als Text geschrieben, mit Wörtern, Zahlen und Symbolen. Doch schon ein winziger Fehler (wie ein fehlendes Komma oder ein zusätzlicher Buchstabe, genau wie eine fehlende Zutat) kann dazu führen, dass das Programm nicht mehr funktioniert. Für Anfänger kann das verwirrend und mitunter frustrierend sein.

Doch die blockbasierte Programmierung macht diesen Prozess viel einfacher und angenehmer!

Anstatt lange, komplizierte Texte zu schreiben, verwenden wir farbige Blöcke, die sich wie Puzzleteile zusammenfügen. Jeder Block steht für eine Aktion, die wir dem Computer auftragen, zum Beispiel „Figur bewegen“, „warten“ usw.

Wenn wir die Bausteine in der richtigen Reihenfolge zusammensetzen, bilden alle diese Bausteine ein vollständiges Programm, das der Computer problemlos verstehen kann und das uns das erstaunliche und interaktive Projekt zeigt, das wir geschaffen haben.

Auf diese Weise machen wir das Programmieren zu etwas Spielerischem und Intuitivem, das es den Lernenden ermöglicht, sich auf die Logik von Aktionen und Ergebnissen zu konzentrieren, anstatt sich über Rechtschreibfehler Sorgen zu machen.



Die Benutzeroberfläche von Scratch, einem der wichtigsten blockbasierten Programme



Deshalb ist die blockbasierte Programmierung für Menschen konzipiert, die am besten durch Sehen und Tun lernen. Sie ist auch deshalb besonders hilfreich für Erwachsene mit Down-Syndrom, weil:

- Es nutzt Farben und Formen, um Informationen zu strukturieren.
- Es liefert sofortiges Feedback: Man sieht sofort, ob etwas funktioniert.
- Es vermeidet verwirrende Syntaxfehler (keine Tippfehler).
- Es ermöglicht Lernenden, frei zu experimentieren, ohne Angst haben zu müssen, etwas kaputt zu machen.
- Es macht aus abstrakten Konzepten etwas Sichtbares und Konkretes.

Lasst uns die Grundlagen der blockbasierten Programmiersprache verstehen.

Bevor wir mit dem Programmieren von Spielen und all den anderen tollen Dingen beginnen, die man sich vorstellen kann, ist es wichtig zu verstehen, wie eine blockbasierte Programmierumgebung funktioniert, damit wir uns besser und ohne Angst oder Frustration mit dem Werkzeug auseinandersetzen können.

Diese Art der Programmierung ist visuell, einfach und spielerisch gestaltet. Anstatt lange Textzeilen zu schreiben, erstellen die Lernenden Programme, indem sie farbige Blöcke zusammensetzen. Jeder dieser Blöcke ist mit einer bestimmten Aktion oder Anweisung verknüpft.

Die Blöcke passen wie Puzzleteile zusammen und helfen dir, die Logik und die Reihenfolge der Operationen visuell zu erfassen. Denk also daran: Wenn die Teile nicht zusammenpassen, musst du dir eine andere Lösung überlegen!

Wenn wir die Bausteine verbinden, bauen wir im Grunde eine Geschichte, der der Computer folgen kann: Der Computer liest die Bausteine von oben nach unten, genau wie wir eine Liste von Schritten in einem Rezept lesen.

Jede blockbasierte Plattform wie Scratch, MakeCode oder Code.org ordnet Blöcke in farbcodierte Kategorien ein.

Jede Farbe steht für eine bestimmte Aktionsart, wodurch es einfacher wird, das zu finden, was wir brauchen.

Nachfolgend sind die gängigsten Kategorien und ihre Funktionen aufgeführt:



Kategorie des Blocks	Farbe des Blocks	Was der Block bewirkt
Ereignis	Gelb	Startet das Programm (z.B. „Wenn die grüne Flagge angeklickt wird“)
Bewegung	Blau	Bewegt eine Figur oder ein Objekt
Aussehen	Lila	Ändert die Farbe, zeigt eine Nachricht an oder blendet sie aus
Klang	Rosa	Spielt einen Ton oder Musik ab
Kontrolle	Orange	Wiederholt oder wartet („5 Mal wiederholen“, „2 Sekunden warten“)
Sensorik	Hellblau	Erkennt, ob ein Ereignis eintritt (z. B. „Wenn eine Taste gedrückt wird“)
Variablen	Dunkelorange	Speichert und verwendet Informationen wie Zahlen oder Punktzahlen.

Wie Programme funktionieren verstehen

Jedes Programm, das wir mit diesen Bausteinen erstellen, folgt einer klaren Abfolge: Diese Abfolge verleiht den Computern die Fähigkeit, genau das zu tun, was wir von ihnen verlangen.

Aber was ist eine Sequenz?

Eine Sequenz ist im Grunde ein Baustein, den man zu einem Programm zusammenfügt. Der Computer, der das Programm ausführt, liest die in den Bausteinen enthaltenen Anweisungen von oben nach unten, genau wie wir eine Seite lesen oder einem Rezept folgen!

Diese Sequenz ist wirklich etwas Besonderes, denn man kann die Struktur des eigenen Denkens ablesen, aber man kann auch die eigene Idee verändern, indem man einen Block nach oben oder unten verschiebt oder ihn ganz entfernt.



Und diese Veränderungen werden eine sichtbare und unmittelbare Wirkung haben! Dies wird Ihnen helfen zu verstehen, wie eine Handlung eine andere beeinflusst, und Ihnen eine wichtige Fähigkeit vermitteln: logisches Denken und das Vorhersagen von Ergebnissen.

Deshalb lieben wir diese Programmiersprache so sehr: Sie ist eine Möglichkeit, den Geist zu trainieren, Ideen zu ordnen, Muster zu erkennen und Schritt für Schritt intelligente und kreative Lösungen zu entwickeln.

Alles wertvolle Fähigkeiten für den Alltag, nicht nur für die Computerzeit!



AKTIVITÄT 2: Fang die Maus

Was Sie benötigen:

- Ein Computer
- Internetverbindung

Was Sie lernen werden:

- Verstehen Sie, dass ein Programm einem Ablauf von Anweisungen von oben nach unten folgt.
- Erkennen Sie, wie verschiedene Blöcke Aktionen und Reaktionen steuern.
- Verwenden Sie Ereignisse, Schleifen und Bedingungen, um eine einfache Sequenz zu organisieren.
- Sehen Sie, wie verbundene Anweisungen interaktives Verhalten zwischen Sprites erzeugen.
- Üben Sie Problemlösung und Anpassung durch Ausprobieren.
- Schaffen Sie Vertrauen und Neugierde für die Funktionsweise von Code durch visuelle Darstellung.

Lasst uns das Spiel programmieren!

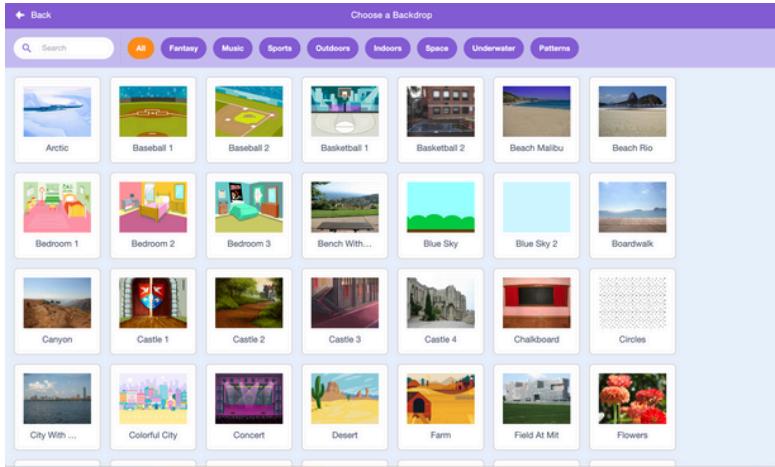


Screenshot 1

Erstelle dein Projekt:

1. Öffne Scratch und klicke auf Erstellen.
2. Löschen Sie den Standard-Sprite.
3. Füge einen neuen Sprite für die Katze hinzu.
4. Füge einen weiteren Sprite für die Maus hinzu.

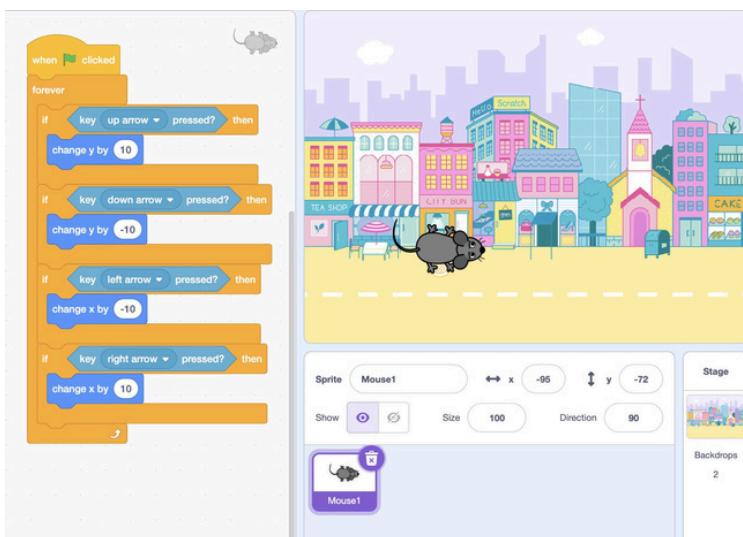




Screenshot 2

Wähle den Hintergrund:

1. Klicken Sie auf das Symbol „Hintergrund auswählen“ (unten rechts).
2. Wähle einen ansprechenden Hintergrund, an dem dein Spiel stattfinden soll.
3. Kehren Sie zum Programmierbereich zurück.



Screenshot 3

Bewegen Sie die Maus:

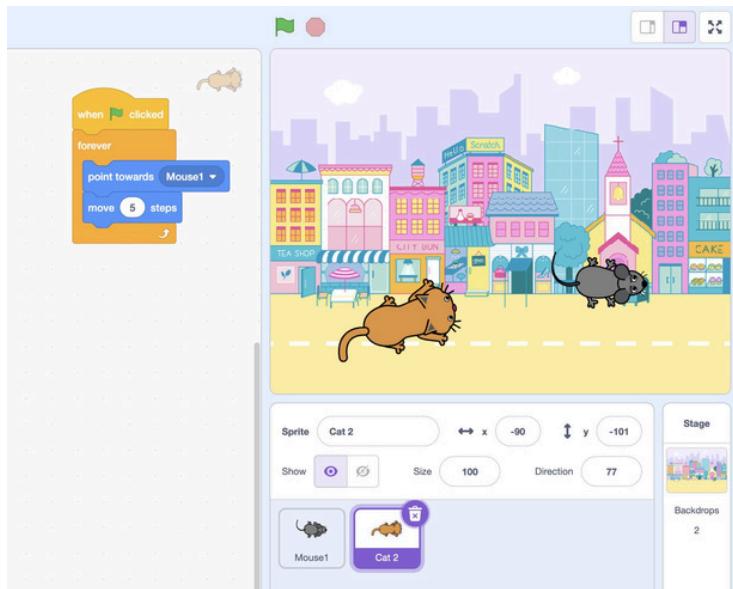
1. Klicken Sie auf das Maus-Sprite.
2. Gehen Sie zur Kategorie „Ereignisse“ und ziehen Sie den Block „Wenn die grüne Flagge angeklickt wird“ an die Wand.
3. Gehen Sie zu Control und ziehen Sie eine „Endlosschleife“.
4. Fügen Sie innerhalb der „forever“-Schleife vier „if“-Blöcke aus Control hinzu, einen für jede Pfeiltaste.



5. Wählen Sie unter „Erkennung“ die Option „Taste (Pfeil) gedrückt?“ und fügen Sie in jeden „if“-Block eine Taste ein.

6. Fügen Sie aus Motion Folgendes hinzu:

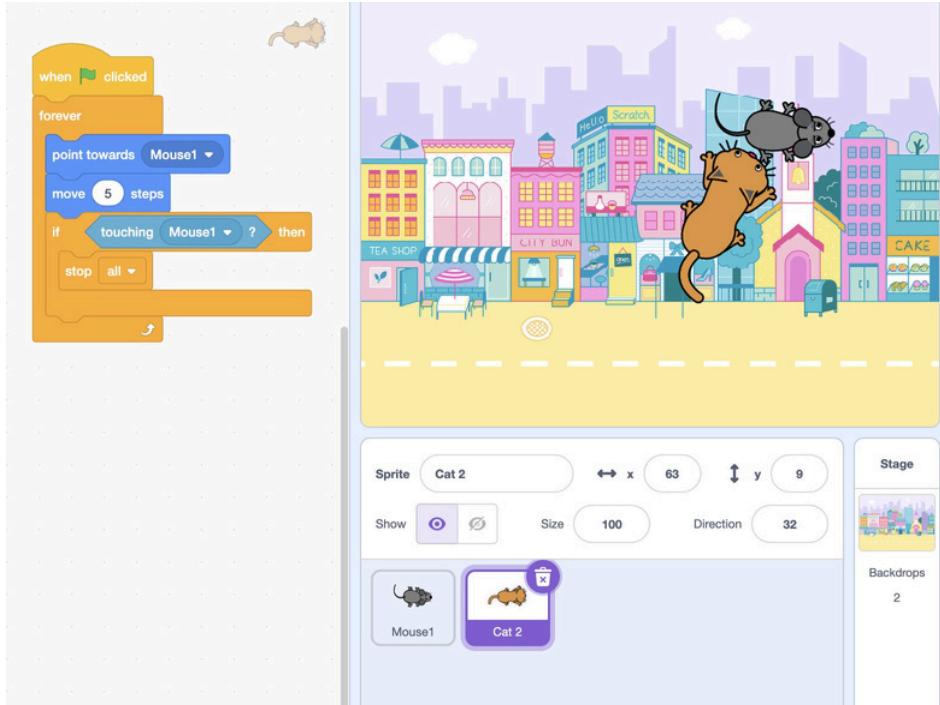
- Für den Aufwärtspfeil: „Ändere y um 10.“
- Für den Abwärtspfeil: „Ändere y um -10.“
- Für den Pfeil nach rechts: „Ändere x um 10.“
- Für den Pfeil nach links: „Ändere x um -10.“ Jetzt kann sich Ihre Maus nach oben, unten, links und rechts bewegen!



Screenshot 4

Lass die Katze die Maus jagen:

1. Klicke auf das Katzen-Sprite.
2. Fügen Sie „Wenn die grüne Flagge angeklickt wurde“ hinzu.
3. Füge „für immer“ hinzu.
4. Legen Sie die Bewegungsblöcke hinein:
 - „auf (die Maus) zeigen“
 - „5 Schritte gehen“ Nun folgt die Katze der Maus überall auf dem Bildschirm.



Screenshot 5

Das Spiel endet, wenn die Katze die Maus jagt:

1. Bleib auf dem Katzen-Sprite.
2. Füge innerhalb der „forever“-Schleife Folgendes hinzu:
 - „Wenn (Maus) berührt wird, dann“ (Steuerungsblock + Sensorblock)
 - Füge innerhalb dieses Blocks „Stopp alles“ (Kontrollblock) hinzu. Wenn die Katze nun die Maus berührt, stoppt das Spiel!

Teste das Spiel:

1. Klicken Sie auf die grüne Flagge, um zu starten.
2. Bewegen Sie die Maus mit den Pfeiltasten.
3. Schau zu, wie die Katze es jagt!
4. Falls die Katze zu schnell oder zu langsam ist, ändern Sie die Zahl in „5 Schritte machen“.
5. Versuchen Sie es erneut und haben Sie Spaß beim Experimentieren.



5. FEHLERSUCHE UND PROBLEMLÖSUNG DURCH KLARE KOMMUNIKATION

Strategien zur Erkennung und Lösung von Problemen, wenn etwas nicht funktioniert

Fehlersuche ist eine grundlegende Fähigkeit in der Elektronik und Programmierung, auch wenn sie im traditionellen Lernen oft unterschätzt wird.

Aber was genau versteht man unter Fehlersuche?

Fehlerbehebung bedeutet, die Ursache des Problems zu erkennen, zu verstehen, was es verursachen könnte, und verschiedene Lösungsansätze auszuprobieren, um es zu beheben.

Diese Fähigkeit ist genauso wichtig wie die Fähigkeit, ein Programm oder eine Schaltung korrekt zu erstellen: Tatsächlich verbringen selbst professionelle Programmierer und Ingenieure viel Zeit mit der Fehlersuche, da dies ein normaler (und sehr wichtiger) Bestandteil ihrer Arbeit ist.

Es ist kein Synonym für Versagen oder Inkompetenz, sondern ein wichtiger Bestandteil des Prozesses.

Die Fehlersuche kann sowohl eine Herausforderung als auch eine Chance darstellen: HERAUSFORDERUNG - Wenn Ihnen das Erlernen von etwas schwerer fällt, wenn es mit abstraktem Denken und Gedächtnis zu tun hat, kann sich dies auf die Fehlersuche auswirken, die die Verwaltung zahlreicher möglicher Lösungen erfordert.

CHANCE - Wenn Problemlösungsprozesse klar und speziell auf diese Schwierigkeiten zugeschnitten sind, beispielsweise mit rein visuellen Lösungen und konkreten Strategien, können Sie effektiv mit der Problemlösung beginnen.

In diesem Kapitel werden wir untersuchen, wie man Problemlösungsstrategien entwickelt und wie man mit dieser entscheidenden Phase der Programmierung umgeht.

Denken Sie daran, dass Probleme normal sind.

Eine der wichtigsten Lektionen bei der Fehlersuche ist das Verständnis, dass unerwartete Ergebnisse in der Elektronik und Programmierung dazugehören. Sie sind kein Zeichen von Versagen, mangelnder Intelligenz oder Unzulänglichkeit. Selbst die erfahrensten Profis stoßen regelmäßig auf Probleme in ihren Programmierprojekten und müssen Zeit investieren, um den Fehler zu finden und zu beheben. Der Unterschied zwischen einem Anfänger und einem fortgeschrittenen Programmierer besteht jedoch darin, dass Letzterer sich von Hindernissen nicht entmutigen lässt, da er über eine eigene Problemlösungsstrategie verfügt.



Strategien zur Fehlerbehebung

Eine effektive Fehlersuche erfordert klare Vorgehensweisen und darf nicht durch zufällige Versuche erfolgen.

Auch wenn zufällige Lösungsversuche gelegentlich zum Ziel führen, sind sie ineffizient, frustrierend und wenig lehrreich. Systematische Fehlersuche hingegen untersucht potenzielle Ursachen in logischer Reihenfolge, indem sie zunächst die wahrscheinlichsten oder am einfachsten überprüfbaren Möglichkeiten prüft und sich dann den weniger wahrscheinlichen Ursachen zuwendet.

Vorlage zum Befolgen beim Anfordern von Hilfe:

„Ich versuche [zu beschreiben, was der Schüler tun möchte].“

Jedoch [beschreiben Sie das aktuelle Ergebnis].

Ich habe bereits [Liste der überprüften oder ausprobierten Punkte] abgehakt.

Können Sie mir helfen, [Frage oder Bereich einfügen, in dem Hilfe benötigt wird] zu verstehen?

Konkretes Beispiel:

„Ich versuche, meine LED zum Blinken zu bringen. Im Moment leuchtet sie gar nicht. Ich habe bereits überprüft, ob die Batterie angeschlossen ist, der Schalter eingeschaltet ist und alle Verbindungen fest sitzen. Können Sie mir helfen herauszufinden, ob meine LED richtig angeschlossen ist?“

Diese Herangehensweise, um Hilfe zu bitten, zeigt, dass der Schüler systematisch nach Lösungen sucht, Verantwortung für die selbstständige Bewältigung von Problemen übernimmt und ein konkretes Bedürfnis hat, das er klar formuliert. Sie vereinfacht auch die Arbeit der Lehrkraft, da diese sofort versteht, worin das Problem besteht, was bereits versucht wurde und worauf sie ihre Hilfe konzentrieren kann, während der Schüler gleichzeitig aktiv in den Prozess einbezogen wird.

Das Üben dieses Kommunikationsmusters kann im Lernprozess sehr hilfreich sein!

Hinweis für Pädagogen: Pädagogen begleiten den Prozess, sind aber auch Ihre Partner in Erforschung!

Wenn Lehrkräfte zeigen, dass auch sie Fehler machen können und diese dann ruhig korrigieren, erkennen die Lernenden, dass Fehler normal und lösbar sind. Auch die Mitschüler spielen eine wichtige Rolle!

Wenn Lernende die Methoden anderer beobachten, entdecken sie unterschiedliche Denk- und Arbeitsweisen. Manchmal hilft es beiden, eine Lösung besser zu verstehen, wenn man sie einem Freund erklärt.

Gemeinsames Lernen macht das Erlebnis freundlicher, sicherer und angenehmer.

Durch die Zusammenarbeit entwickeln sich mit der Zeit Geduld, Empathie und Selbstvertretung – unerlässliche Fähigkeiten für ein unabhängiges Leben und die zukünftige Arbeit.



AKTIVITÄT 3: „Käsejagd“ zum Üben von Sequenzen, einfachen Variablen und systematischer Fehlersuche

So funktioniert das fertige Spiel:

In diesem Spiel steuerst du einen Schläger. Ein Ball bewegt sich umher und muss vom Schläger abprallen. Jedes Mal, wenn der Ball den Schläger trifft, erhöht sich deine Punktzahl. Verlässt der Ball den unteren Bildschirmrand, ist das Spiel vorbei („Game Over“). Dein Ziel ist es, vor Spielende so viele Punkte wie möglich zu erzielen.

Was Sie benötigen:

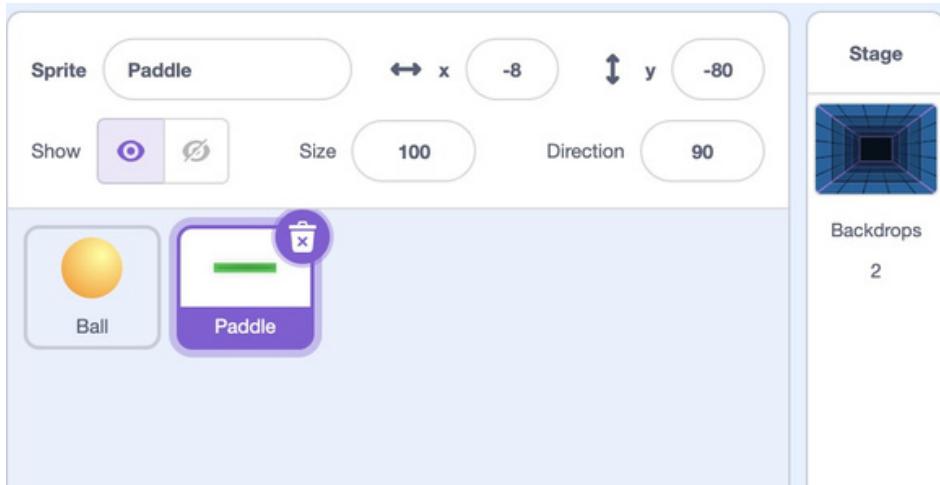
- Ein Computer
- Internetverbindung

Was Sie lernen werden:

- Den Ablauf der Anweisungen in einem Programm verstehen (Ereignis > Schleife > Bedingung > Aktion).
- Verwenden Sie Sensoren (z. B. „Berührung des Schlägers“), Variablen (z. B. „Punktzahl“) und Spielmeldungen (z. B. „Spiel vorbei“).
- Verbinde die Bewegung des Balls, die Kontrolle des Schlägers, die Abpralllogik und die Grenzen des Spielfelds.
- Wenden Sie eine einfache Methode zur Fehlerbehebung an: Symptom identifizieren > Ursache finden > Lösung anwenden.
- Entwickle logisches Denken, achte auf die Abfolge von Aktionen und gewinne Selbstvertrauen beim Testen und Optimieren von Code.

Lasst uns das Spiel programmieren!

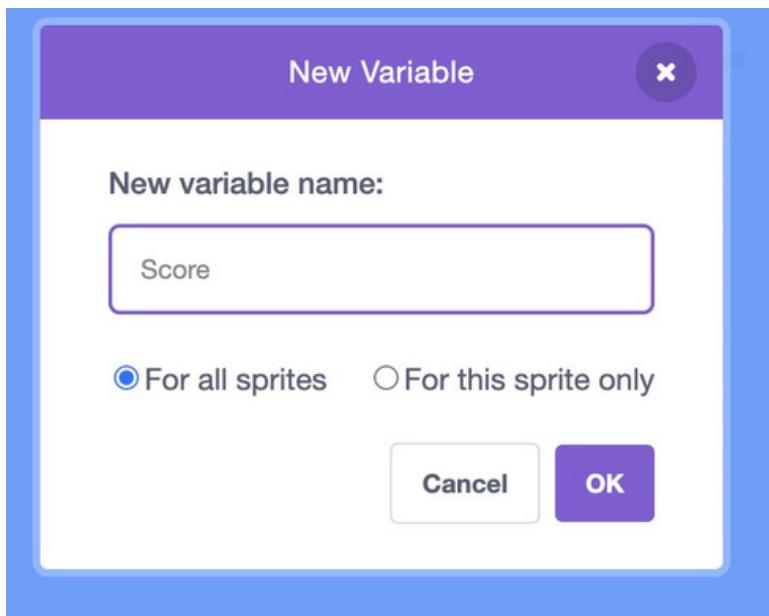




Screenshot 1

Hintergrund und Sprites

1. Öffne Scratch und erstelle ein neues Projekt.
2. Wähle einen Hintergrund für das Spielfeld.
3. Füge einen Sprite für den Ball hinzu und nenne ihn „Ball“.
4. Füge ein Sprite für den Schläger hinzu und nenne es „Schläger“.



Screenshot 2



Setze die Variable

1. Erstelle die Variable

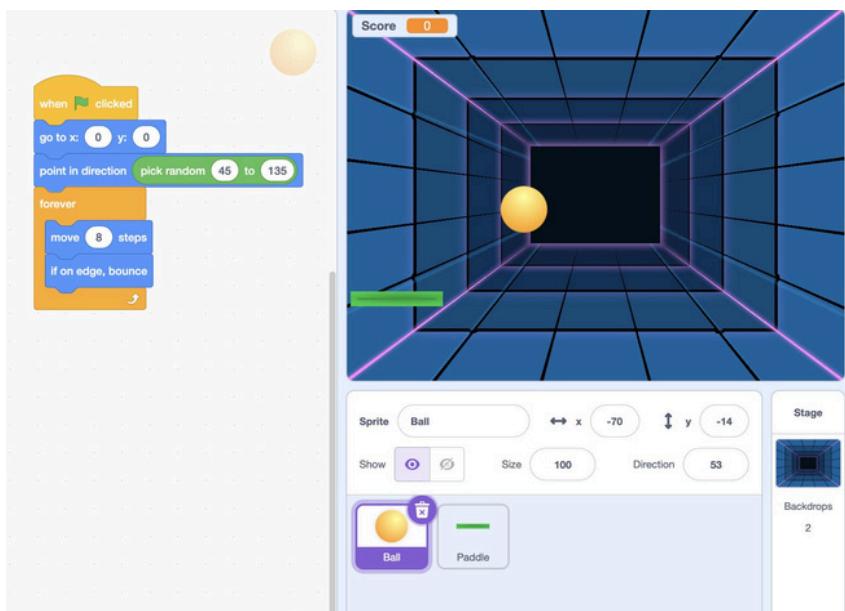
- Klicken auf den Abschnitt „Variablen“.
- auf „Variable erstellen“ klicken
- Benennen Sie die Variable in „Score“ um.
- Wählen Sie „Für alle Sprites“. Diese Variable zählt, wie oft der Ball den Schläger trifft. Wenn Sie das Spiel starten, wird der Punktestand in der oberen linken Ecke des Bildschirms angezeigt.

Code für das Paddel

Der Paddel muss sich nach links und rechts bewegen, um der Maus (oder der Tastatur) zu folgen.

Klicke auf das Paddel-Symbol und füge diese Blöcke hinzu:

1. Ziehen Sie aus dem Ereignismenü „wenn grüne Flagge angeklickt wurde“ per Drag & Drop.
2. Ziehen Sie in Motion „Gehe zu x:0 und y:-100“, um die Ausgangsposition festzulegen.
2. Ziehen Sie aus dem Control-Fenster eine Endlosschleife.
3. Ziehen Sie aus Motion „set x to“ und platzieren Sie es innerhalb der Endlosschleife.
4. Ziehen Sie unter „Sensing“ die Option „mouse x“ neben „set x to“. Nun bewegt sich der Schläger beim Starten des Spiels immer horizontal mit Ihrer Maus.



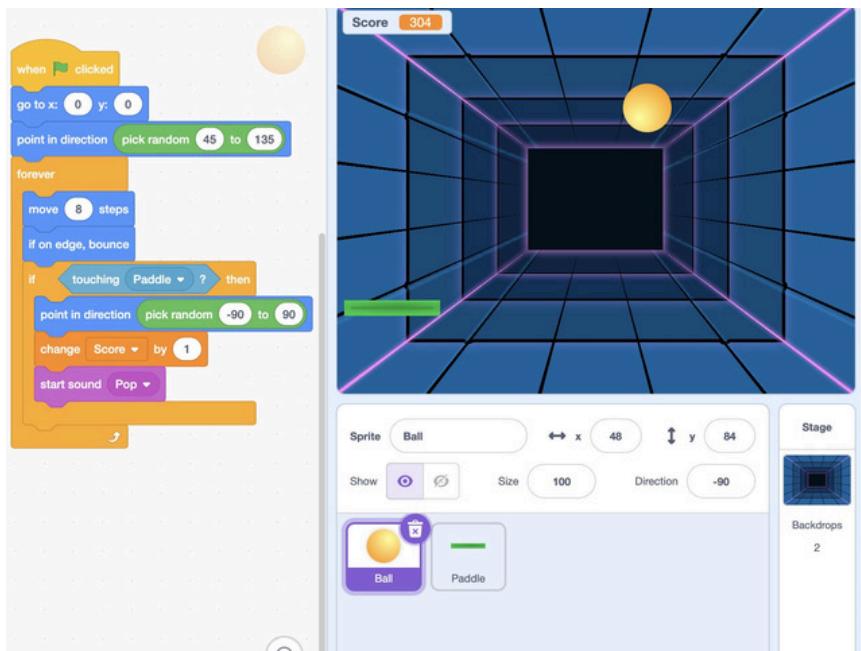
Screenshot 3



Code für den Ball

Klicke auf das Ball-Symbol und füge diese Blöcke hinzu:

1. Der Bewegungsblock „Wenn die grüne Flagge angeklickt wird“
2. „Gehe zu x: 0 y: 0“ (aus Motion): Dadurch wird der Ball in die Mitte zurückgesetzt.
3. Richtung angeben (wählen Sie zufällig 45 bis 135) (aus Bewegung und Operatoren): Dadurch startet der Ball in einer zufälligen Abwärtsrichtung.
4. Fügen Sie den Block „forever“ (aus der Steuerung) hinzu. Fügen Sie innerhalb der Endlosschleife Folgendes hinzu:
 - 8 Schritte zurücklegen (von Bewegung)
 - wenn am Rand, abprallen (von Motion) Wenn Sie jetzt auf die grüne Flagge klicken, bewegt sich der Ball kontinuierlich und prallt von den Bildschirmrändern ab.



Screenshot 4

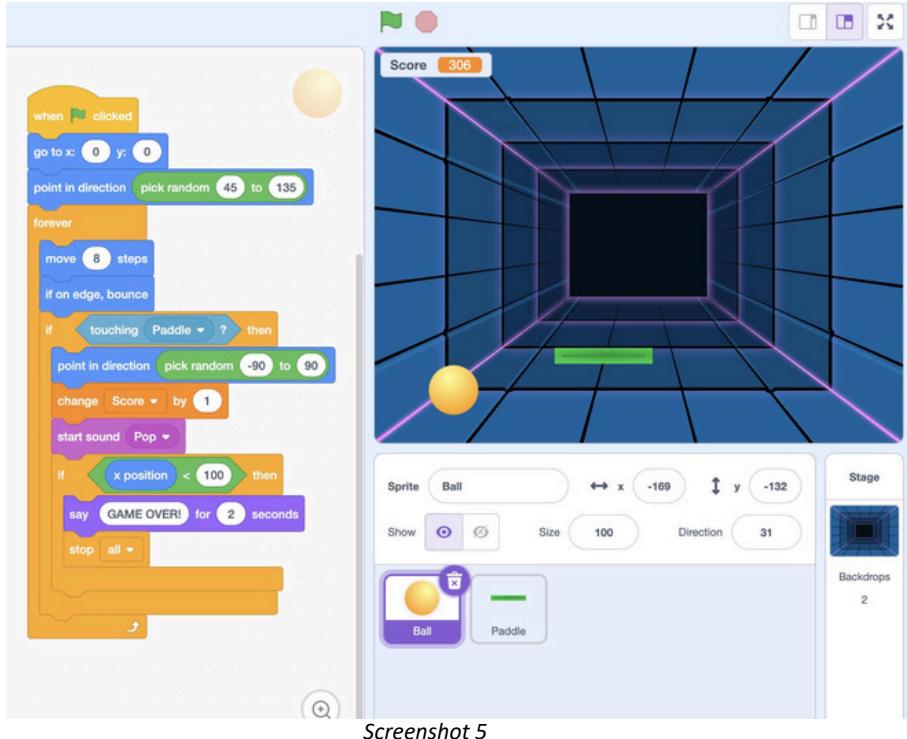
Lass den Ball auf dem Schläger abprallen und aktualisiere den Punktestand.

Bleib im Ball-Sprite.

Fügen Sie innerhalb derselben Endlosschleife unterhalb von „if on edge bounce“ diese neue Blockgruppe hinzu:

1. Wenn [Paddle] berührt wird? (von Sensing) Innerhalb dieses „if“-Blocks:
 - Richtungsanzeige aus der Bewegung > innerhalb dieses Blocks „Zufällige Werte von -90 bis 90 auswählen“ aus den Operatoren hinzufügen
 - Punktzahl um 1 ändern (aus Variablen)
 - Ton abspielen [Pop] (von Sound)





Screenshot 5

Füge „Spiel vorbei“ hinzu, wenn der Ball vom Bildschirm fällt.

Immer noch beim Ball-Sprite, innerhalb der Endlosschleife, füge unterhalb der vorherigen Blöcke eine weitere Bedingung hinzu:

1. Fügen Sie den if-Block aus den Operatoren hinzu.

-Füge innerhalb des if-Blocks "<" von den Operatoren hinzu.

- Füge innerhalb des „<“-Operators die „x-Position“ aus Motion hinzu.

2. Innerhalb dieses „if“-Blocks:

- Sag 2 Sekunden lang „Game Over“ (aus dem Looks-Stil).

- Stopp-Funktion (Steuerung): Das gesamte Spiel wird gestoppt, sobald der Ball unter den Schlägerbereich fällt.

Teste dein Spiel

1. Klicken Sie auf die grüne Flagge, um zu starten.

2. Bewegen Sie Ihre Maus nach links und rechts, um den Schläger zu steuern.

3. Beobachte, wie sich der Ball bewegt und von den Wänden und dem Schläger abprallt.

4. Prüfen Sie, ob sich die Punktzahl erhöht, wenn der Ball den Schläger berührt.

5. Lassen Sie den Ball von der unteren Kante fallen, um zu sehen, ob das Spiel richtig beendet wurde.



Checkliste zur Fehlerbehebung

Wenn Ihr Pong-Spiel nicht richtig funktioniert, befolgen Sie diese Schritte nacheinander. Überspringen Sie keine Schritte. Überprüfen Sie alles sorgfältig und testen Sie nach jedem Schritt erneut.

Order	Was zu prüfen ist	Warum
1	Prüfe, ob die grüne Flagge angeklickt wurde.	Manchmal ist das Spiel einfach nicht gestartet. Klicke die grüne Flagge erneut an.
2	Prüfe, welche Figur (Sprite) ausgewählt ist.	Stelle sicher, dass du die Blöcke der richtigen Figur hinzufügst (zum Beispiel: Ball, Schläger oder Punktestand).
3	Prüfe, ob alle Skripte korrekt starten.	Jede Figur, die etwas tun soll, sollte mit „Wenn die grüne Flagge angeklickt wird“ beginnen.
4	Prüfe, ob die Schleifen (Loops) laufen.	Überprüfe, ob die Schleifen tatsächlich aktiv sind.
5	Prüfe die Reihenfolge deiner Blöcke.	Der Computer folgt deinen Anweisungen von oben nach unten. Wenn etwas zu früh oder zu spät passiert, verschiebe die Blöcke an eine bessere Position.
6	Prüfe die Bedingungen („Falls“-Blöcke).	Stelle sicher, dass jeder „Falls wird ... berührt“-Block den richtigen Namen der Figur verwendet (zum Beispiel „Falls Schläger berührt wird“).
7	Prüfe die Positionen.	Wenn der Ball verschwindet, schau auf seine x- und y-Position. Füge einen vorübergehenden Block hinzu: „sage (verbinde y: (y-Position)) für 1 Sekunde“, um zu sehen, wo er ist.
8	Prüfe die Variablen.	Lass dir deine Variable (zum Beispiel „Punktestand“) auf dem Bildschirm anzeigen und schau, ob sie sich während des Spiels verändert.
9	Prüfe Richtung und Geschwindigkeit.	Wenn sich der Ball zu schnell oder zu langsam bewegt, ändere die Zahl im „gehe ...er-Schritt“-Block oder ändere seine Startrichtung.
10	Neustarten und erneut testen.	Klicke nach jeder Korrektur erneut auf die grüne Flagge und teste es. Beobachte, was sich ändert, und höre auf, wenn alles korrekt funktioniert.
11	Wenn nichts funktioniert, frage um Hilfe.	Erkläre genau, was du versucht hast und was passiert ist. Beispiel: „Der Ball bewegt sich, aber er prallt nicht vom Schläger ab. Ich habe den ‚Falls berührt‘-Block überprüft und er ist da.“



6. SCHLUSSFOLGERUNGEN

Sie haben dieses Modul nun abgeschlossen – eine Reise durch klare Kommunikation, visuelles Lernen und kreatives Üben. Dabei haben Sie gelernt, wie einfache Worte, aussagekräftige Bilder und schrittweise Anleitungen selbst komplexe Ideen verständlich machen. Sie haben entwickelt, programmiert und experimentiert und dabei entdeckt, dass Lernen spannend wird, wenn man die eigenen Fortschritte sehen und anfassen kann.

Erforsche weiter, baue weiter und frage dich immer wieder: „Was passiert, wenn ich das ausprobiere?“

Jeder neue Schritt ist eine Chance, etwas Erstaunliches zu lernen. Deine Reise endet hier nicht, sie geht weiter mit jedem Licht, das du einschaltest, jedem Block, den du verbindest, und jeder Idee, die du zum Leben erweckst.

Wenn es Ihnen Spaß gemacht hat, entdecken Sie die anderen Module des Lernkurses!



7. WEITERFÜHRENDE LITERATUR

Wenn es Ihnen Spaß gemacht hat und Sie noch mehr entdecken möchten, finden Sie hier einige tolle Online-Ressourcen. Fragen Sie Ihre Lehrkraft, ob sie Ihnen dabei helfen kann.

- **Scratch:** <https://scratch.mit.edu>

Auf „<https://scratch.mit.edu>“ können Sie mit Scratch Spiele, Animationen und Geschichten mithilfe visueller Blöcke erstellen. Es ist dieselbe Art des Programmierens, die wir bereits geübt haben: einfach und unterhaltsam für alle.

- **Code.org:** <https://www.code.org>

Diese Website bietet kurze und einfache Lektionen zum Thema Programmieren. Sie können Spiele spielen, Spielfiguren bewegen und Schritt für Schritt lernen, wie Computerlogik funktioniert.

- **Arduino-Schulung:** <https://education.arduino.cc>

Arduino hilft dir, Elektronik anhand von realen Projekten zu verstehen. Es eignet sich hervorragend, um zu lernen, wie Sensoren, Lampen und Motoren zusammenarbeiten.

- **MakeCode von Microsoft:** <https://makecode.microbit.org>

Auf der Website „<https://makecode.microbit.org>“ können Sie blockbasierte Programmierung ausprobieren, um ein kleines Gerät namens Micro:bit zu steuern. Sie eignet sich hervorragend, um das Gelernte über Programmierabläufe zu üben.

- **Instructables:** <https://www.instructables.com>

Auf „<https://www.instructables.com>“ finden Sie Tausende kreative DIY-Projekte, von einfachen Stromkreisen bis hin zu Bastelarbeiten aus Recyclingmaterialien. Jedes Projekt enthält Fotos und Schritt-für-Schritt-Anleitungen.

- **Exploratorium – Tüftlerstudio:** <https://www.exploratorium.edu/tinkering>

Ein unterhaltsamer und inspirierender Ort zum spielerischen Lernen. Hier finden Sie Ideen zum Bauen, Experimentieren und Gestalten von Projekten, die Kunst, Wissenschaft und Spiel verbinden.



8. SELBSTBEWERTUNGSRASTER

Mal sehen, was du noch weißt! Wähle für jede Frage die beste Antwort aus. Die richtige Antwort ist fett gedruckt.

Nr.	Frage	A	B	C	D
1	Warum ist klare Kommunikation in der Elektronik und Programmierung wichtig?	Weil Elektronik auch bei verwirrenden Anweisungen immer funktioniert	Weil kleine Fehler oder unklare Schritte verhindern können, dass Dinge funktionieren	Weil es das Projekt professioneller aussehen lässt	Weil es das Codieren schneller, aber schwieriger zu verstehen macht
2	Was ist das Hauptziel der Verwendung vereinfachter Sprache in diesem Modul?	Um komplexe Ideen leichter verständlich zu machen	Um alle Fachbegriffe zu entfernen	Um so wenig wie möglich zu schreiben	Um den Text wissenschaftlicher klingen zu lassen
3	Warum sind visuelle Hilfsmittel wie Bilder und Diagramme nützlich?	Sie machen den Text bunt und unterhaltsam	Sie helfen den Lernenden, sich besser zu erinnern und zu verstehen	Sie ersetzen Lehrer und Erklärungen	Sie lassen die Seite voll aussehen
4	Was bedeutet „blockbasierte Programmierung“?	Code mit langen Textzeilen schreiben	Code mit farbigen Blöcken bauen, die zusammenpassen	Formen zeichnen, um dem Computer zu sagen, was er tun soll	Spezielle Symbole und Zahlen tippen
5	Was solltest du tun, bevor du ein Projekt startest?	Sofort mit dem Bauen beginnen	Alle Schritte erst am Ende lesen	Deine Materialien auf dem Tisch anschauen und organisieren	Jemand anderen bitten, alles vorzubereiten



Nr.	Frage	A	B	C	D
6	Was passiert in der Aktivität „Fang die Maus“?	Die Katze jagt die Maus, bis sie sie fängt	Der Spieler zeichnet ein Labyrinth für die Maus	Die Maus sammelt Käse, um zu gewinnen	Der Spieler baut das Spiel mit echten Drähten
7	Was ist Fehlerbehebung (Troubleshooting)?	Probleme finden und beheben, wenn etwas nicht funktioniert	Schnelleren Code mit weniger Schritten schreiben	Jedes Mal ein neues Projekt starten, wenn man einen Fehler macht	Farben und Formen im Programm ändern
8	Was ist der erste Schritt bei der Fehlerbehebung eines Codierungsproblems?	Den Code löschen und von vorne beginnen	Bilder zeichnen	Bunte Blöcke	Mehr Blöcke hinzufügen, ohne zu prüfen
9	Welche Fähigkeit trainieren Lernende, wenn sie Schritt-für-Schritt-Anleitungen folgen?	Logisches Denken und Sequenzierung	Geschichten schneller lesen	Kreativere Bilder zeichnen	Musik hören beim Codieren
10	Was ist die Kernbotschaft des Modul-Abschlusses?	Hör auf zu experimentieren, sobald du die Lektion beendet hast	Weiter erkunden, bauen und durch Neugier und Praxis lernen	Nur eine Methode verwenden und sie niemals ändern	Jeden Codeblock und jede Definition auswendig lernen

MODUL 3: GEMEINSAM LERNEN: VERBINDUNGEN knüpfen & UNTERSTÜTZUNG ERHALTEN



Überblick

Dieses Modul betont die Bedeutung des Lernens als gemeinschaftliche Erfahrung im Gegensatz zu einer individuellen. Für Erwachsene mit Down-Syndrom und anderen intellektuellen Beeinträchtigungen kann der Erwerb neuer Fähigkeiten – insbesondere in technischen Bereichen wie Elektronik und Programmierung – eine Herausforderung darstellen. Forschung und Praxis zeigen jedoch, dass kollaborative Lernumgebungen nicht nur den Wissenserwerb, sondern auch Selbstvertrauen, Unabhängigkeit und soziale Integration deutlich fördern.

Das FEAT-DS-Projekt zeigte durch seine vergleichende Forschung, dass Erwachsene mit Down-Syndrom oft dann aufblühen, wenn sie auf gegenseitige Unterstützung, klare Anleitung durch Pädagogen und strukturierte Möglichkeiten zur Teamarbeit zurückgreifen können. In solchen Umgebungen wird Lernen zu einem gemeinsamen Erlebnis. Jeder Teilnehmer bringt seine Stärken ein, und die Gruppe als Ganzes profitiert von der Vielfalt an Fähigkeiten, Kreativität und Perspektiven.

Die zentralen Themen dieses Moduls – Teamarbeit, um Hilfe bitten und Fehler als Teil des Lernprozesses akzeptieren – basieren auf inklusiven Bildungsansätzen. Das Modul bietet Lehrenden eine theoretische Grundlage und Lernenden praktische Übungen. Es unterstreicht, dass technische Fähigkeiten nicht isoliert entwickelt werden können, sondern vielmehr zusammen mit sozialen Kompetenzen wie Kommunikationsfähigkeit, Kooperationsfähigkeit, Problemlösungskompetenz und Resilienz gefördert werden müssen.

Das Modul wurde mit Blick auf Barrierefreiheit entwickelt. Dazu gehören eine vereinfachte Sprache, Erklärungen anhand von Piktogrammen und visuelle Hilfsmittel. Durch die Integration von Reflexionsphasen und kooperativen Spielen erwerben die Teilnehmenden nicht nur technisches Wissen, sondern verinnerlichen auch wichtige soziale und emotionale Kompetenzen.

Dieses Modul lädt uns ein, von der individuellen Übung zur kollaborativen Teamarbeit überzugehen und zu lernen, wie wir gemeinsam kreativ sein, Probleme lösen und Erfolge feiern können!



Ziele

Das Modul zielt darauf ab, Folgendes zu erreichen:

- Die Teamfähigkeit und Kooperationsbereitschaft der Lernenden sollen durch die Bearbeitung von Programmier- und Elektronikprojekten gefördert werden.
- Fördern Sie einen positiven Umgang mit Fehlern und betonen Sie, dass Fehler natürlich, zu erwarten und für den Fortschritt unerlässlich sind.
- Die Fähigkeit der Lernenden stärken, konstruktiv Hilfe von Gleichaltrigen, Lehrenden und Ressourcen zu suchen.
- Die Bedeutung von gegenseitiger Unterstützung, Anleitung durch Lehrkräfte und Gruppenreflexion für die Schaffung einer sicheren und inklusiven Lernumgebung sollte hervorgehoben werden.
- Statten Sie Lehrende mit konkreten Werkzeugen und Strategien aus, um kollaboratives und unterstützendes Lernen zu fördern.
- Den Lernenden ein Gefühl der Selbstwirksamkeit und des Selbstvertrauens vermitteln und sie ermutigen, digitale und technische Bereiche weiterhin selbstständig zu erkunden.

Erwartete Ergebnisse

Am Ende dieses Moduls werden die Lernenden in der Lage sein:

1. Erläutern Sie den Wert von Teamarbeit in Lernkontexten und beschreiben Sie, wie die Zusammenarbeit die Problemlösung verbessert.
2. Zeigen Sie grundlegende Verhaltensweisen zur Kontaktaufnahme mit Hilfe, wie z. B. das Heben der Hand, die Verwendung von Piktogrammen oder das direkte Nachfragen bei Gleichaltrigen/Lehrkräften um Klärung.
3. Erkennen Sie Fehler als Teil des Lernprozesses an, um Ängste abzubauen und die Widerstandsfähigkeit zu stärken.
4. Beteilige dich aktiv an Gemeinschaftsprojekten und übernimm dabei verschiedene Rollen innerhalb der Gruppe.
5. Beheben Sie grundlegende Programmier- und Elektronikprobleme in Zusammenarbeit mit anderen.
6. Führen Sie eine strukturierte Reflexion über Ihre Erfahrungen durch und identifizieren Sie Herausforderungen, Erfolge und Momente, in denen Hilfe von Vorteil war.
7. Zeigen Sie gesteigertes Selbstvertrauen und Motivation bei der Verfolgung weiterer technischer und digitaler Lernmöglichkeiten.



1. DER WERT VON TEAMARBEIT BEIM LERNEN

Teamarbeit gilt weithin als Grundpfeiler inklusiver und effektiver Bildung. Für Lernende mit Down-Syndrom gehen die Vorteile der Teamarbeit über die akademischen Leistungen hinaus und umfassen auch die soziale, emotionale und persönliche Entwicklung.

Die Forschung zum kollaborativen Lernen betont, dass Lernende, wenn sie Aufgaben gemeinsam bearbeiten, Folgendes erreichen:

- Kognitive Belastung teilen: Schwierige Aufgaben werden besser zu bewältigen, wenn die Verantwortlichkeiten verteilt werden.
- Lernen Sie aus unterschiedlichen Perspektiven: Unterschiedliche Lernende nehmen unterschiedliche Details wahr, was zu kreativeren Lösungen führt.
- Kommunikationsfähigkeiten entwickeln: Anderen eine Idee zu erklären, erfordert Klarheit, Geduld und Übung.
- Soziale Kontakte knüpfen: Gemeinsames Arbeiten fördert Freundschaften und wirkt der Isolation entgegen.

Im Kontext von Programmierung und Elektronik ermöglicht Teamarbeit den Teilnehmern Folgendes:

- Weisen Sie unterschiedliche Aufgaben zu (z. B. Schaltkreise verbinden, Anweisungen lesen, Code eingeben).
- Geben Sie umgehend Feedback von Kollegen.
- Gemeinsam Erfolge feiern und so die Motivation stärken.

Pädagogen sollten betonen, dass es in einem Team keine „beste“ Rolle gibt. Jeder Beitrag ist wertvoll. Diese Botschaft reduziert den Wettbewerbsdruck und fördert die Inklusion.



Praktische Übung 1 – Partnerarbeit: Leichtberührungssensor

Zweck

Die Lernenden üben Teamarbeit, Aufgabenverteilung und den sicheren Aufbau einer kleinen Schaltung, die sofortiges Feedback (Licht/Ton) gibt. Dabei kommen der experimentelle Ansatz des Projekts, Programmieren und leicht verständliche Anleitungen zum Einsatz.

Dauer

~90–120 Minuten (Partnerarbeit + Gruppenpräsentation).

Materialien (pro Paar)

- Batteriefach (2xAA oder 3V Knopfzellenhalter)
- Ein-/Ausschalter oder Druckknopf
- 1 LED (mit Widerstand oder vorverdrahtet) oder ein kleiner Summer
- 6–8 farbcodierte Jumperkabel
- Optional: einfache Folien-, „Touchpad“ (zwei Stücke Aluminiumfolie + Draht)
- Gedruckte Schritt-karten mit großen Fotos/Piktogrammen (A5)
- Rollenkarten: Baumeister / Helfer
- Große Symbole/Piktogramme

Sicherheitsregeln (lehren & aushängen)

1. Vor dem Umstecken der Kabel die Stromzufuhr unterbrechen.

2. Die Kabel sollten ordentlich verlegt sein; lose Metallteile dürfen die Batterie nicht berühren.

3. Vor dem Werkzeugkauf fragen. Dann einschalten und testen.

(Entspricht den Projektvorgaben zu konkretem, schrittweisem Lernen und visuellen Hilfsmitteln.)

Barrierefreie Einrichtung

- Eine Idee pro Schritt; große Fotos; minimaler Text; Symbole für Strom, Kabel, Test.
- Zuerst eine kurze Demonstration, damit die Lernenden den Zusammenhang zwischen Ursache und Wirkung erkennen (LED/Summer reagiert sofort).

Minilektion (10–12 Min.)

1. Zeigen Sie einen einfachen Stromkreis aus Batterie, Schalter und LED. Einschalten: Licht. Ausschalten: Kein Licht.
2. Rollenverteilung: Baumeister (arbeitet an den Teilen), Helfer (liest Karte, überprüft Schritte). Nach der Hälfte der Zeit Rollenwechsel.
3. Fehler normalisieren: „Wenn es nicht funktioniert, prüfen wir: Stromversorgung → Verkabelung → Bauteil.“ (3-stufiges Debugging wird modulübergreifend angewendet.)



Zielsetzung (für Lernende)

- Arbeitet zu zweit in Rollen; baut eine winzige Schaltung; testet sie und erklärt ihre Funktionsweise.
- Bitten Sie um Hilfe mithilfe eines einfachen Hilfescripts; versuchen Sie die 3-stufige Fehlersuche.

Zeit & Gruppierung

- 10 Min. Demo & Einrichtung → 25–35 Min. Entwicklung & Test → 10–15 Min. Präsentation → 10 Min. Aufräumen/Reflexion.

Materialien

Wie oben aufgeführt; optionales Folien-Touchpad anstelle des Knopfes (durch Drücken zweier Folien mit einem Finger wird der Stromkreis geschlossen).

BUILDER HELPER

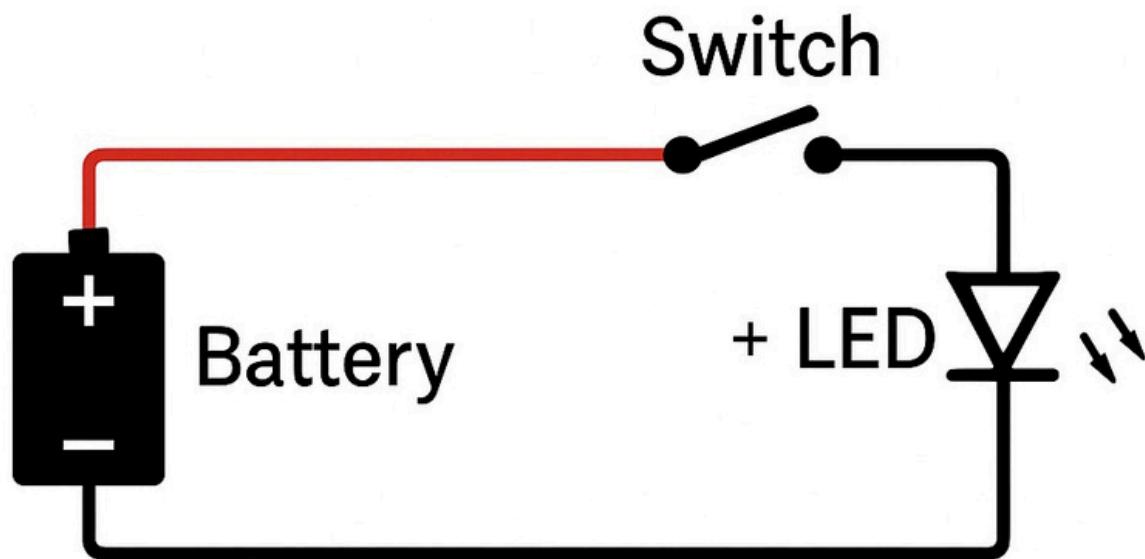


- Hands-on parts
- Say steps back
- Power OFF to rewire

- Read steps
- Check safety
- Encourage partner

*Abbildung 1a. Rollenkarten
(Baumeister / Helfer) Karten für
Paarrollen und Rollenwechsel.*

POWER OFF



*Abbildung 1b. Batterie → Schalter → LED-Verdrahtung
(farbcodiert) Karten für Paarrollen und Abwechseln.*

Schritt für Schritt (auf A5-Karten mit Fotos ausdrucken)

1. Wählt die Rollen. Der Baumeister trägt die „Baumeister“-Karte; der Helfer liest die Schrittkarte vor. (Später tauschen.)
2. Strom ausschalten. Batteriehalter an den Schalter anschließen (rot an Schaltereingang; schwarz an LED/Summer „-“).
3. Schließen Sie den Schalter an die LED „+“ (oder den Summer „+“) an.
4. Kabel prüfen: rot → Schalter → LED (+), schwarz → LED (-) → Batterie (-).
5. Strom EIN → Test. Sehen Sie Licht / hören Sie Ton?
6. Tausche die Rollen und baue alles einmal neu auf.

(Die Formulierungen sollten kurz sein, eine Idee pro Zeile; die Symbole den Verben zuordnen.)

Fehlerbehebungskarte (3-stufige Fehlersuche)

- A) Stromversorgung: Ist die Batterie angeschlossen? Ist der Schalter eingeschaltet?
- B) Verbindungen: Gibt es lose oder vertauschte Kabel? Bitte die Farbreihenfolge beachten.
- C) Komponente: Versuchen Sie es mit einer anderen LED/einem anderen Summer. (Unmittelbare sensorische Rückmeldung bestätigt den Erfolg.)

Hilfe suchendes Mikroskript (Karte)

„Ich habe ___ versucht. Das Problem ist ___. Können Sie sich ___ ansehen?“
(Verwendung vereinfachter Sprache und lokalisierter Hilfestellungen.)

Differenzierung (für heterogene Leistungsniveaus)

- **Einfacher:** Vorverdrahtetes LED-Modul verwenden; eine einseitige Anleitung mit Fotos beilegen.
- **Schwieriger:** Ersetzen Sie den Knopf durch ein berührungsempfindliches Folienpad; erklären Sie, warum durch Berührung ein Stromkreis geschlossen wird.
- **AAC unterstützt:** Piktogramme für Aktionen (verbinden / drücken / testen); Gestehinweise.

Hinweise für den Moderator (praktisch)

- Modellbauer/Helper-Gespräch: Der Helfer liest vor; der Modellbauer wiederholt und führt die Anweisungen aus. Rollentausch nach der Hälfte der Zeit.
- Loben Sie den Prozess (die einzelnen Schritte), nicht nur das Ergebnis; feiern Sie das erste erfolgreiche Licht/den ersten Ton.
- Halten Sie einen Ersatzakku/LED-Satz bereit, um Komponentenfehler schnell zu lokalisieren.

Kurzer Check (während der Aktivität)

Bitten Sie zwei Personen, auf Strom, Schalter und Verbraucher (LED/Summer) zu zeigen und in einem Satz zu beschreiben, was passiert, wenn der Schalter eingeschaltet ist. (Eine konkrete Benennung fördert das Verständnis.)

Rollenrotation

- Wechsel mitten in der Sitzung: Helfer wird zum Erbauer; Erbauer wird zum Helfer.
- Jedes Paar setzt ein Häkchen bei „Wir haben die Rollen getauscht“ auf der Schrittkarte (visuelle Verantwortlichkeit).

Unterstützung durch Gleichaltrige

- Wenn Sie 2 Minuten lang nicht weiterkommen: Versuchen Sie einen Debugging-Schritt und verwenden Sie dann das Hilfeskript mit einem Nachbarn oder einer Lehrkraft. (Stärkt die Resilienz und bietet strukturierte Unterstützung.)



Präsentation (Galerie, 10–15 Min.)

- Zwei Personen legen die Schaltkreise auf einen Tisch. Eine andere Person drückt einen Knopf/ein Touchpad, um zu sehen/hören, wie er funktioniert.
- Nutzen Sie die „Zwei Sterne & ein Wunsch“-Methode: Nennen Sie zwei positive Aspekte und eine Idee, die Sie ausprobieren möchten. Halten Sie es kurz und freundlich. (Fördert konstruktives Feedback.)

Kurze Reflexion (Exit-Ticket)

Die Lernenden vervollständigen zwei Symbole auf einem halben Blatt:

★ „Was hat bei uns funktioniert?“ (Zeichnung oder 3–4 Wörter) 🔧 „Was wir beim nächsten Mal ausprobieren sollten?“ (Zeichnung oder 3–4 Wörter)

Beobachtungscheckliste (Moderator, pro Paar ankreuzen)

- Einmal die Rolle gewechselt Beim Umstecken der Kabel die Stromzufuhr unterbrochen Funktionierenden Schaltkreis (Licht/Ton) aufgebaut Vor dem Bitten um Hilfe die 3-Schritt-Fehlerbehebung durchgeführt Das Hilfeskript mindestens einmal verwendet

Aufräumen & Aufbewahren (Routine vermitteln)

- Strom ausschalten; Batterie entfernen; Kabel entwirren und mit einem Kabelbinder bündeln; Teile in einen beschrifteten Karton zurücklegen.
- Machen Sie ein Foto vom fertigen Aufbau; notieren Sie alle defekten Teile.



Abbildung 2. Beispiel für Feedback-Tokens „Zwei Sterne & ein Wunsch“ (drei Piktogramme).

2. UM HILFE BITTEN: EINE STÄRKE, KEINE SCHWÄCHE

Eine der wichtigsten Veränderungen in der Denkweise besteht darin, Fehler nicht als Misserfolge, sondern als Chancen zu begreifen. Dieses Konzept wird in der Pädagogik auch als „Wachstumsorientierung“ bezeichnet. Für Lernende mit Down-Syndrom, die sich in akademischen oder technischen Bereichen ohnehin schon unsicher fühlen, ist diese neue Sichtweise auf Fehler besonders bestärkend.

In der Elektronik und Programmierung sind Fehler unvermeidlich:

- Ein Kabel könnte falsch angeschlossen sein.
- Möglicherweise fehlt eine Codezeile.
- Ein Sensor reagiert möglicherweise nicht wie erwartet.

Statt zu entmutigen, können diese Momente zu wertvollen Lernmöglichkeiten werden. Pädagogen sollten Folgendes betonen:

- Fehler sind normal.
- Jeder macht Fehler – selbst Experten.
- Fehler helfen uns zu erkennen, was wir anders machen müssen.
- Wir werden stärker, indem wir Fehler beheben.

Beispiel (Lehrkraft spricht zu einer Gruppe):

„Wenn das Licht nicht angeht, heißt das nicht, dass wir gescheitert sind. Es bedeutet, dass wir etwas Neues über den Schaltkreis gelernt haben. Jetzt können wir die Verbindungen gemeinsam überprüfen.“

Diese Perspektive verschiebt die Lernkultur im Klassenzimmer von der Angst vor dem Scheitern hin zur Begeisterung für das Entdecken. Eine wiederkehrende Herausforderung in der inklusiven Bildung besteht darin, dass viele Lernende mit Lernschwierigkeiten zögern, um Hilfe zu bitten. Dies kann auf frühere Stigmatisierungserfahrungen oder den Wunsch nach Unabhängigkeit zurückzuführen sein. Hilfe zu suchen ist jedoch keine Schwäche – es ist eine essenzielle Kompetenz für lebenslanges Lernen.



Pädagogen können die Inanspruchnahme von Hilfe normalisieren, indem sie:

- Explizites Lehren von Redewendungen oder Gesten, die Lernende anwenden können.
- Positive Verstärkung, wenn Lernende um Klärung bitten.
- Indem sie selbst ein Beispiel für hilfssuchendes Verhalten geben (z. B. indem eine Lehrkraft offen in einem Handbuch nachschlägt und sagt: „Auch ich brauche manchmal Hilfe“).

Hilfe kann viele Formen annehmen:

- Einen Kollegen um Hilfe bitten.
- Eine Demonstration vom Lehrer anfordern.
- Verwendung eines Piktogramms oder einer visuellen Karte zur Kennzeichnung von Verwirrung.
- Er hielt inne und fragte die Gruppe: „Kann jemand diesen Schritt noch einmal erklären?“

Entscheidend ist, dass die Lernenden verstehen, dass um Hilfe zu bitten nicht bedeutet, aufzugeben – es ist Teil des Erfolgs.

Praktische Übung 2 – Gemeinsame Fehlersuche

Erster Teil:

Zweck

Die Lernenden üben anhand einer kleinen Elektronik- und Programmieraufgabe, wie sie um Hilfe bitten und Probleme beheben können. Wir vermitteln eine klare, wiederholbare Fehlersuchroutine und unterstützen sie mit einfacher Sprache. Erfolg wird durch direktes Licht- und Tonfeedback signalisiert.

Dauer

~90–120 Minuten (Demo → gemeinsames Debuggen → Galeriefreigabe).



Materialien (pro Paar)

- Laptop/Tablet mit einem blockbasierten Codierungstool (anfängerfreundlich, große Symbole)
- USB-Mikrocontroller oder Simulator
- LED- und/oder Summer-Schaltung aus der praktischen Übung 1
- Farbcodierte Überbrückungskabel
- Gedruckte Debug-Karte & Hilfeskript (A5, Piktogramme)
- Optional: Projektor für Live-Code-Demonstration. (Große Symbole, schrittweise visuelle Darstellungen, lokalisierter, leicht lesbarer Text verwenden.)

Sicherheitsgrundregeln (wiederholen): Vor dem Bewegen von Kabeln Strom abschalten; Kabel ordentlich verlegen; Werkzeug anfordern; erst nach Überprüfung testen. (Konkrete, schrittweise Vorgehensweise.)

Minilektion (10–12 Min.)

- Zeigen Sie ein kurzes Blockprogramm, das eine LED blinken oder einen Summer piepen lassen soll.
- Nennen Sie zwei Fähigkeiten, die heute gefragt sind: Klar um Hilfe bitten + 3-Schritt-Debugging anwenden.
- Veranschaulichen Sie beides anhand eines fehlerhaften Beispiels (z. B. falsche Pin- oder Blockreihenfolge). Stellen Sie eine Verbindung zur Tüftel- und Programmiermethode des Projekts her.

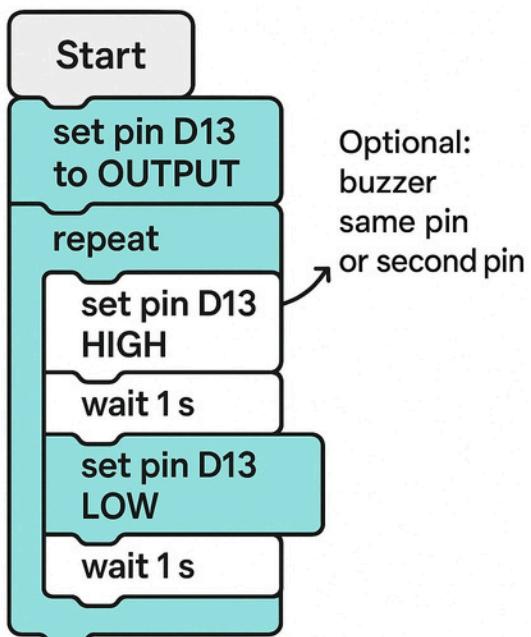


Abbildung 3. Beispiel für „Blink & Beep“-Blöcke (Start → Pin setzen → Wiederholen: Ein/Warten/Aus/Warten).



Power → Connections → Code /Pin

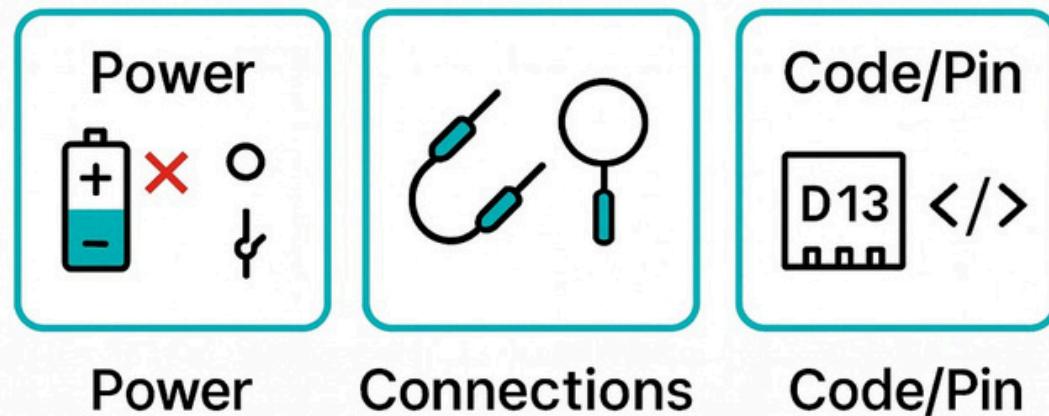


Abbildung 4. Piktogramme: Stromversorgung → Drähte → Bauteil/Pin (in dieser Reihenfolge prüfen).

```
void setup() {  
    pinMode(13, OUTPUT);  
}  
void loop() {  
    digitalWrite(13, LOW);  
    delay(1000);  
    // Optional: buzzer  
    same pin or second pin  
}
```

Abbildung 5. Piktogramme der Debug-Karte.



Zweiter Teil:

Ziele (für Lernende)

- Bitten Sie mithilfe eines einfachen Skripts um Hilfe.
- Wenden Sie die 3-Schritt-Debugging-Methode an, um Code und Schaltung zum Laufen zu bringen.
- Beschreiben Sie die Funktion jedes Hauptblocks in einem kurzen Satz.

Zeit & Gruppierung

- 10–12 Min. Demo → 30–40 Min. gemeinsames Debuggen → 10–15 Min. Präsentation in der Galerie → 10 Min. Reflexion.

Aufstellen

Jedes Zweierteam erhält ein fehlerhaftes Starterprojekt (zwei typische Fehler sind unten aufgeführt). Die Schaltung aus der praktischen Übung 1 muss anschlussfertig sein. Die Benutzeroberflächen sollten große Symbole und ein übersichtliches Layout verwenden.

Starter-Bugs (wählen Sie 1–2)

A) Falscher Pin (LED an Pin D13, aber Code setzt D12) B) Fehlender „Pin-Modus/Ausgang festlegen“-Block (oder falsche Blockreihenfolge) C) Zu kurze Wartezeit (Blinken ist für das menschliche Auge nicht sichtbar) (Verwenden Sie Fehler, die bei der Behebung sichtbare/hörbare Effekte erzeugen.)

Schritt für Schritt (auf A5 ausdrucken mit Fotos/Symbolen)

1. **Schau und sag es: Was soll das Programm tun? (Blinken/Piepen)**
2. **Schließen Sie den Stromkreis an (wie in der praktischen Übung 1). Schalten Sie die Stromzufuhr während der Verdrahtung aus.**
3. **Führe den Code aus. Was passiert? (Sag es.)**
4. **3-stufige Fehlersuche: Stromversorgung → Verbindungen → Code/Pin.**
5. **Ändere eine Sache, teste erneut.**
6. Wenn es funktioniert: Beschreiben Sie, was Sie repariert haben; machen Sie ein Foto. (Text kurz halten, eine Idee pro Zeile; mit Bildern/Piktogrammen kombinieren.)

Hilfe suchendes Mikroskript (Karten)

„Ich habe ___ versucht. Das Problem ist ___. Können Sie sich ___ ansehen?“ (Verwenden Sie vereinfachte, lokal angepasste Sprache und Symbole.)

Moderatorbewegungen (praktisch)

- Vor dem Eingreifen den Hilfetext einblenden; auf die 3-Schritte-Karte zeigen.
- Wenn die Paare nicht weiterkommen, fragen Sie: „Was habt ihr geändert? Was werdet ihr als Nächstes versuchen?“
- Feiern Sie den Prozess (Testen einer Änderung nach der anderen) und den sensorischen Erfolg (Licht/Ton).



Differenzierung

- **Einfacher:** Ein Foto der Stecknadeln bereitstellen und den richtigen Blockstapel markieren.
- **Schwieriger:** Einen zweiten Ausgang (LED + Summer) oder einen Touchpad-Trigger hinzufügen.
- **Unterstützte Kommunikationsmittel:** Piktogramme für Aktionen; Gestenanweisungen für „Testen/Ändern“.

Kurzer Check (während der Aktivität)

Bitten Sie einen Lernenden, auf den im Code und auf dem Board verwendeten Pin zu zeigen; sagen Sie: „Dieser Block schaltet die LED ein.“ (Kurze verbale + visuelle Verknüpfung.)

Teamroutinen für Hilfe und Reparatur

Eine gute Frage formulieren

- Zeigen Sie das fehlerhafte Verhalten auf → beschreiben Sie Ihre Erwartungen → zeigen Sie die Zeile/den Block auf, der Ihrer Meinung nach das Problem verursacht.
- Versuchen Sie zunächst eine Änderung, bevor Sie erneut nachfragen. (Stärkt die Widerstandsfähigkeit und sorgt für Struktur.)

Peer-Assist-Protokoll (2–3 Min.)

Der Nachbar hört zu → wiederholt das Problem → schlägt einen Debugging-Schritt vor → beobachtet den Test. Bitte leise sprechen; bei Problemen das Hilfescript verwenden. (Schafft eine kleine „Praxisgemeinschaft“.)

Präsentation (Galerie, 10–15 Min.)

- Jedes Paar führt das festgelegte Programm aus; die Partner drücken den Knopf/das Touchpad, um eine Reaktion zu sehen/hören.
- Feedback: Zwei Sterne und ein Wunsch (zwei positive Aspekte + eine Anregung). (Positive, prägnante Sprache.)

Spiegelung

★ „Was wir repariert haben“ (Zeichnung/3–4 Wörter) 🔧 „Nächstes Mal versuchen wir...“ (Zeichnung/3–4 Wörter)

Beobachtungscheckliste (Moderator)

- Hilfescript einmal verwendet
- Dreistufige Fehlersuche durchgeführt
- Korrekten Pin/Block identifiziert
- Reparatur mit Licht-/Tonfunktion überprüft



3. DIE ROLLE VON LEHRKRÄFTEN UND GLEICHALTRIGEN

Die Lehrenden in diesem Modul fungieren eher als Moderatoren denn als reine Dozenten. Zu ihren Aufgaben gehören:

- Schaffung einer sicheren und inklusiven Atmosphäre.
- Bereitstellung klarer, schrittweiser Anleitungen.
- Modellierung von Zusammenarbeit und Hilfesuche.
- Verwendung leicht zugänglicher Materialien (Piktogramme, vereinfachte Texte, Diagramme).
- Anregung zur Reflexion nach den Aktivitäten.

Der Pädagoge ist nicht der alleinige „Experte“, sondern ein Mitlernender, was zeigt, dass auch Erwachsene weiterhin lernen und sich anpassen.

Der Austausch mit Gleichaltrigen ist ebenso wichtig. Tatsächlich können Gleichaltrige Erklärungen oft in einer einfacheren und verständlicheren Sprache liefern. Sie bemerken möglicherweise auch Fehler, die Lehrkräften entgehen. Wichtig ist auch die emotionale Unterstützung durch Gleichaltrige – ein Lächeln, Ermutigung oder Applaus können das Selbstvertrauen stärker stärken als rein fachliche Anleitung.

Gleichaltrige sollten dazu ermutigt werden:

- Unterstützen Sie sich gegenseitig verbal (z. B. mit Worten wie „Gut gemacht!“ oder „Lass es uns noch einmal versuchen“).
- Die Aufgaben gerecht verteilen.
- Feiert die Erfolge der Gruppe.

Diese Dynamik verwandelt die Gruppe in eine Lerngemeinschaft, in der jeder seinen Beitrag leistet und jeder davon profitiert.



Praktische Übung 3 – Galerie zeigen und teilen

Ziele (für Lernende)

- Zeigen Sie eine funktionierende Schaltung/ein funktionierendes Programm und beschreiben Sie in einem Satz, was es bewirkt.
- Verwenden Sie ein einfaches Feedback-Skript mit einem Partnerpaar (zwei Sterne und ein Wunsch).

Zeit & Gruppierung

- 10–15 Min. Vorbereitung → 30–40 Min. Galeriegang (Paare rotieren) → 10–15 Min. Feedbackrunde → 10 Min. Reflexion.

Aufstellen

- Jedes Paar platziert seine Schaltung auf einer Anzeigematte.
- Sie erstellen einen Textstreifen: „Unser Projekt: _____. Es bewirkt: _____.“ (Leicht lesbar).

Feedback-Skripte (Karten)

- Zwei Sterne und ein Wunsch: „ Ich mag _____;  Nächstes Mal _____.“
- Mir fällt auf / Ich frage mich: „Mir fällt auf, dass _____. Ich frage mich, ob _____.“
(Verwenden, wenn Lernende eine beobachtende Sprache bevorzugen.) (Formulieren Sie kurz; verwenden Sie Symbole.)

Schritt für Schritt (A5-Karte für Lernende)

1. Probieren Sie es aus. Drücken Sie den Knopf/das Touchpad → Licht-/Tonsignal beachten.
2. Fassen Sie den Vorfall in einem Satz zusammen.
3. Platzmarker: zwei  + ein .
4. Die Gastgeber bedanken sich und notieren eine Idee auf ihrer Matte. (Das unmittelbare sensorische Feedback hält die Motivation hoch.)

Differenzierung

- Einfacher: Satzanfänge und Bildsymbole bereitstellen.
- Schwieriger: Bitten Sie die Paare, zwei Versionen (vor/nach der Korrektur) zu vergleichen und die Änderung zu erläutern.
- AAC: Symbolkarten für „like/idee/works“.

Maßnahmen des Moderators

- Kurze, prägnante Formulierung anregen: „Ein Satz.“ „Mit dem Finger zeigen, wo sich der Satz ändert.“
- Neuaustrichtung auf Kriterien: „Reagiert der Sensor wie geplant?“ „Können Sie die Lösung nennen?“



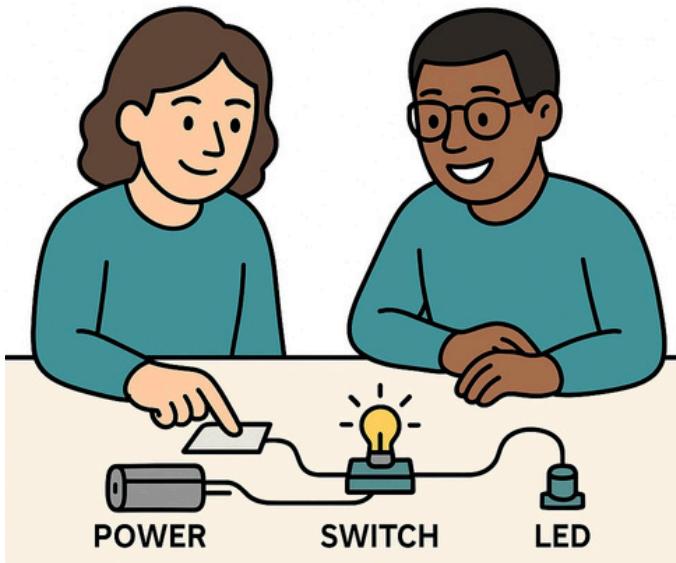


Abbildung 7. Foto von Gleichaltrigen beim Testen eines Berührungssensors; Licht schaltet sich ein.

Reflexion, Feier & Dokumentation

Reflexionskreis (8–10 Min.) Schnell im Kreis herumgehen:

★ „Was uns geholfen hat, war Folgendes: _____.“ 🔧 „Nächstes Mal versuchen wir Folgendes: _____.“ (Kurze Sätze und Piktogramme fördern die Sprachgewandtheit.)

Mini-Check (während/nach der Galerie)

Bitten Sie jeden Lernenden:

- Zeigen Sie auf den Netzschalter/Ausgang und beschreiben Sie, was passiert, wenn dieser eingeschaltet ist.
- Nennen Sie eine Änderung, die sie vorgenommen haben, um das Programm/die Verkabelung zu reparieren. (Eine konkrete Angabe erleichtert das Verständnis.)

Beobachtungscheckliste (Moderator)

- Vorgeführt wurde eine funktionierende Schaltung/ein funktionierendes Programm (Sensor reagierte)
- Zwei positive Rückmeldungen und eine Verbesserungsvorschläge wurden den Kollegen mitgeteilt
- Die Funktion des Projekts wurde in einem Satz beschrieben
- Ein Verbesserungsvorschlag wurde auf der Matte festgehalten (Verwendung für die schrittweise Bewertung während der Pilotphase und der Berichterstattung).

Feier

- Applauswelle + Gruppenfoto; optionaler Anstecker mit der Aufschrift „Wir haben unsere Arbeit geteilt!“.
- Die Sprache sollte einfach und positiv sein; Autonomie und Erfolg sollten gefördert werden.

Evaluierung & Selbsteinschätzung

Nr.	Frage	A	B	C	D
1	Wenn du Kabel austauschst, solltest du zuerst...	Werkzeuge wechseln	Den Strom AUSSCHALTEN	Einen Freund fragen	Den Knopf fester drücken
2	Was ist der erste Schritt unserer Fehlerdiagnose (Debug-Routine)?	Stromversorgung prüfen	Den Code bearbeiten	Die LED austauschen	Alle Kabel ersetzen
3	Die Rolle des „Helpers“ besteht hauptsächlich darin...	Nur Werkzeuge zu nutzen	Schritte lesen, Sicherheit prüfen, unterstützen	Sitzen und warten	Nur Fotos machen
4	Deine LED ist auf D13, aber der Code sagt D12. Beste Lösung?	Batterie tauschen	Code auf D13 ändern	LED-Farbe ändern	Den Tisch aufräumen
5	Welcher Block schaltet eine LED ein?	Warte 1 Sek.	Wiederhole fortlaufend	Setze Pin auf AN/High	Start/Wenn grüne Flagge angeklickt
6	Bevor du die Lehrkraft fragst, solltest du...	Einen Debug-Schritt versuchen	Von vorne beginnen	Zwei Dinge gleichzeitig ändern	Einpacken
7	Sicheres Aufräumen beinhaltet...	Lagerung mit angeschlossener Batterie	Strom AUS, Batterie entfernen, Kabel ordnen	Teile in einer Box mischen	Kabel lose lassen
8	„Zwei Sterne & ein Wunsch“ bedeutet...	Zwei Probleme + eine Lösung	Nur Ideen	Nur Lob	Zwei positive Punkte + eine Idee
9	Eine gute Hilfe-Anfrage beinhaltet...	„Es ist kaputt.“	„Bitte reparier es.“	Was du versucht hast, das Problem, wo man suchen muss	Schweigen und Zeigen
10	Unser Erfolgskriterium in diesem Modul ist erfüllt, wenn...	Der Code lang aussieht	Wir als Erste fertig sind	Sensor/Schaltkreis wie geplant funktionieren und wir es erklären können	Der Tisch ordentlich ist



Weiterführende Literatur

Barrierefreiheit & Leicht lesbare Sprache

- Inclusion Europe — Information für alle: Europäische Standards für leicht lesbare und verständliche Informationen.

<https://www.inclusion-europe.eu/easy-to-read-standards-guidelines> inclusion-europe.eu

- Netzwerk Leichte Sprache — Die Regeln für Leichte Sprache (Neuaufgabe 2022, PDF).

https://www.netzwerk-leichte-sprache.de/fileadmin/content/documents/regeln/Regelwerk_NLS_Neuaufgabe-2022.pdf netzwerk-leichte-sprache.de

"https://www.netzwerk-leichte-sprache.de/fileadmin/content/documents/regeln/Regelwerk_NLS_Neuaufgabe-2022.pdf

IFLA – Richtlinien für leicht verständliche Materialien (2. Aufl., PDF)

<https://www.ifla.org/files/assets/hq/publications/professional-report/120.pdf> ifla.org

Piktogramme & AAC (für Karten, Matten, Skripte)

- ARASAAC-Symbolset — durchsuchbare, kostenlose Piktogramme mit Online-Editor.

<https://beta.arasaac.org/pictograms/search?tab=1> beta.arasaac.org

Elektronik & blockbasierte Programmierung (anfängerfreundlich)

- Arduino — Blink (offizielles Tutorial; schnellstes Hardware-Feedback).

https://www.arduino.cc/en/Tutorial/Blink_Arduino

- Tinkercad Circuits — Lektion 1: Blinkende LED (keine Hardware erforderlich; sicherer Simulator für den Unterricht).

<https://www.tinkercad.com/things/bDHquWI5k2q-lesson-1-blinking-led> Tinkercad

- micro:bit — Programmieren lernen mit MakeCode (blockbasierter Editor + Gerätekopplung).

<https://microbit.org/get-started/getting-started/get-coding> microbit.org



- Microsoft MakeCode — micro:bit Erste Schritte (Browser-Editor; große Blöcke).
[https://makecode.microbit.org/tutorials/getting-started Microsoft MakeCode](https://makecode.microbit.org/tutorials/getting-started)
- Adafruit Circuit Playground Express — Quickstart (umfangreiches Licht-/Soundboard; ideal für sensorisches Feedback).
[https://learn.adafruit.com/circuit-playground-express-circuitpython-5-minute-guide/overview Adafruit Lernsystem](https://learn.adafruit.com/circuit-playground-express-circuitpython-5-minute-guide/overview)

Inklusives Design für den Unterricht (Planungsperspektive)

- CAST — UDL-Leitlinien 3.0 (Optimierung von Wahlmöglichkeiten, vielfältigen Darstellungsformen und Handlungsoptionen).
[https://udlguidelines.cast.org udlguidelines.cast.org](https://udlguidelines.cast.org)
- UDL 3.0 grafischer Organizer (barrierefreies PDF).
[https://udlguidelines.cast.org/static/udlg3-graphicorganizer-digital-nonumbers-a11y.pdf udlguidelines.cast.org](https://udlguidelines.cast.org/static/udlg3-graphicorganizer-digital-nonumbers-a11y.pdf)



Referenzen

Adafruit. (2019). Circuit Playground Express: CircuitPython – 5-Minuten-Schnellstartanleitung. <https://learn.adafruit.com/circuit-playground-express-circuitpython-5-minute-guide/overview> Adafruit Lernsystem

Arduino. (10. Februar 2024). Blinken. <https://www.arduino.cc/en/Tutorial/Blink> Arduino

CAST. (2024). Die UDL-Leitlinien 3.0. <https://udlguidelines.cast.org>

CAST. (2024). Richtlinien für universelles Design für das Lernen (Grafischer Organizer, digital, ohne Zahlen) [PDF]. <https://udlguidelines.cast.org/static/udlg3-graphicorganizer-digital-nonumbers-a11y.pdf>

Inclusion Europe. (6. Oktober 2021). Information für alle: Europäische Standards für leicht verständliche Informationen. <https://www.inclusion-europe.eu/easy-to-read-standards-guidelines/inclusion-europe.eu>

International Federation of Library Associations and Institutions (IFLA). (2010). Richtlinien für leicht verständliche Materialien [PDF].

<https://www.ifla.org/files/assets/hq/publications/professional-report/120.pdf> / ifla.org

Microsoft. (o. J.). Erste Schritte – MakeCode für micro:bit. <https://makecode.microbit.org/tutorials/getting-started> (Originalveröffentlichung 2016; Inhalte werden regelmäßig aktualisiert.) Microsoft MakeCode für micro:bit

Micro:bit Educational Foundation. (o. J.). Programmieren lernen mit MakeCode. <https://microbit.org/get-started/getting-started/get-coding> / microbit.org

Netzwerk Leichte Sprache. (2022). Die Regeln für Leichte Sprache [PDF]. https://www.netzwerk-leichte-sprache.de/fileadmin/content/documents/regeln/Regelwerk_NLS_Neuauflage-2022.pdf / netzwerk-leichte-sprache.de

Tinkercad. (o. J.). Lektion 1 – Blinkende LED.

<https://www.tinkercad.com/things/bDHquWI5k2q-lesson-1-blinking-led> / Tinkercad

Regierung von Saragossa – ARASAAC. (o. J.). ARASAAC-Piktogrammsuchmaschine. <https://beta.arasaac.org/pictograms/search?tab=1> / beta.arasaac.org



MODUL 4: ELEKTRONIK IN UNSERER WELT: VOM HOBBY ZU DEN MÖGLICHKEITEN



Überblick

Dieses Modul beleuchtet ein weiteres wichtiges Thema: „Elektronik in unserer Welt – vom Hobby zur Chance“. Es reagiert auf die „Fragmentierung von Dienstleistungen und mangelnde formale politische Koordination“. Das FEAT-DS-Projekt möchte Menschen mit Down-Syndrom dabei unterstützen, praktische Fähigkeiten im Bereich Elektronik zu erwerben. Dies stärkt nicht nur ihr Selbstwertgefühl, sondern eröffnet ihnen auch bessere Berufsperspektiven und hilft ihnen, sich in die Gesellschaft zu integrieren.

Zunächst wird der Einsatz von Elektronik in Alltagsgegenständen thematisiert. Anschließend werden Hobbys und Aktivitäten im Bereich Elektronik und Programmierung vorgestellt. Die Teilnehmenden werden außerdem gebeten, Ideen für den Einsatz von Elektronik im Alltag und in zukünftigen Berufsfeldern zu sammeln. Tüfteln und Programmieren bilden die Hauptaktivitäten des praktischen Teils. Beim Tüfteln geht es darum, ein elektronisches Gerät zu entwickeln, das sich für praktische Aufgaben wie z. B. eine Erinnerungsfunktion zum Pflanzengießen eignet. Ziel dieser Aktivität ist es, den Teilnehmenden zu verdeutlichen, dass ihre Ideen wertvoll sind. Beim Programmieren steht die Entwicklung eines elektronischen Projekts im Vordergrund, das auf den Bedürfnissen der Nutzer basiert und die praktische Relevanz ihrer Fähigkeiten demonstriert.

Ziele

- Lernen Sie, wie Elektronik in Alltagsgegenständen eingesetzt wird, und sensibilisieren Sie andere dafür.
- Entdecke Hobbys und Aktivitäten rund um Elektronik und Programmierung.
- Verstehen Sie die einfachen Möglichkeiten, wie Elektronikkenntnisse im Alltag nützlich sein können.
- Lerne durch praktisches Tüfteln, wie man ein einfaches elektronisches Gerät entwirft und baut.
- Entdecken Sie, wie man mit Programmieren interaktive elektronische Projekte erstellen kann.



Erwartete Ergebnisse

- Sie werden die Möglichkeit haben zu erforschen, wie Elektronik in Alltagsgegenständen wie Telefonen oder Fernbedienungen eingesetzt wird.
- Sie werden die Möglichkeit haben, Hobbys und Aktivitäten im Bereich Elektronik und Programmierung kennenzulernen.
- Sie werden verstehen können, wie einfache Elektronikkenntnisse im Alltag nützlich sein können.
- Sie werden in der Lage sein, ein einfaches Gerät zu bauen, das ein kleines, reales Problem löst.
- Sie werden in der Lage sein, ein elektronisches Projekt zu erstellen, das auf die Eingaben eines Benutzers reagiert.



1. ELEKTRONIK ÜBERALL – ALLTAGSGEGENSTÄNDE ENTDECKEN

Elektronik im Alltag

Ist Ihnen schon einmal aufgefallen, dass Sie den Fernseher oder das Licht im Zimmer selbstständig einschalten können? Oder haben Sie sich jemals gefragt, wie Sie gelernt haben, Ihr Handy aufzuladen? Und vor allem: Wussten Sie, dass Sie bei all diesen Aktivitäten elektronische Geräte benutzen?

In diesem Modul beschäftigen wir uns mit alltäglichen Elektronikaufgaben und lernen, die Grundlagen selbst zu bauen. Lesen macht immer Spaß, aber in diesem Modul beschränkt sich das nicht nur darauf; Sie werden auch Ihr eigenes einfaches Gerät entwickeln, das Ihnen bei der Lösung kleinerer Probleme im Alltag helfen kann.

Wo kann man Elektronikartikel finden?

Vor einigen Jahrzehnten konnte man nicht mit Sicherheit sagen, dass Elektronik allgegenwärtig ist, aber heutzutage ist sie aus unserem Alltag nicht mehr wegzudenken – sie ist überall präsent. Ein einfaches Beispiel: Wenn man einen Film sehen oder Musik hören möchte, nutzt man elektronische Geräte.

Überlegen Sie kurz, wo man Elektronikgeräte finden kann. Schauen wir uns einige Beispiele an:

- Eine Lampe, die man einschaltet, wenn man seine Wohnung betritt
- Eine Fernbedienung, mit der Sie die Klimaanlage regulieren können, wenn es Ihnen zu warm oder zu kalt ist.
- Ein Wecker, der dabei hilft, pünktlich zu sein
- Ein Spielzeugauto, mit dem man per Fernbedienung Rennen gegen Freunde fahren kann.

Alle oben genannten Objekte werden mithilfe von Elektronik und Schaltkreisen aktiviert.



2. ALLTAGSELEKTRONIK – ENTDECKEN SIE DIE ELEKTRONIKGERÄTE, DIE WIR TÄGLICH VERWENDEN

In unseren Aktivitäten werden wir zunächst Elektronikgeräte erforschen, die wir täglich verwenden, und später einfache Geräte für den täglichen Gebrauch herstellen.

Die Türklingel



Abbildung 1. Der Türklingelschalter

Dies ist die Türklingel.

Können Sie sich vorstellen, wie schwierig es wäre, jemanden mit Namen zu rufen, bis er die Tür öffnet? Türklingeln erleichtern es, jemandem mitzuteilen, dass man hinter der Tür wartet.

Wie funktioniert es?

- Wenn Sie den Knopf (Eingang) drücken, wird der Stromkreis ausgelöst.
- Ein Lautsprecher erzeugt den Ton, den wir als „Ausgangssignal“ bezeichnen.
- Die Stromversorgung erfolgt üblicherweise über eine Batterie oder einen Adapter.

Wenn Sie das nächste Mal auf den Knopf drücken, werden Sie verstehen, wie er funktioniert.



2. ALLTAGSELEKTRONIK – ENTDECKEN SIE DIE ELEKTRONIKGERÄTE, DIE WIR TÄGLICH VERWENDEN

In unseren Aktivitäten werden wir zunächst Elektronikgeräte erforschen, die wir täglich verwenden, und später einfache Geräte für den täglichen Gebrauch herstellen.

Die Fernbedienung



Abbildung 2. Die Fernbedienung

Haben Sie jemals davon geträumt, Superkräfte zu besitzen und Gegenstände zu kontrollieren? Fernbedienung verleiht diese Macht zwar nicht vollständig, aber sie sendet unsichtbare Signale an ein anderes Gerät.

Wie funktioniert es?

- Wenn man die kleinen Knöpfe drückt, wird der Stromkreis ausgelöst.
- Ein winziger Chip im Inneren wandelt das Signal in Licht einer LED (Leuchtdiode) um und schaltet das Gerät ein.
- Die Stromversorgung erfolgt durch interne Batterien.

Der Wecker



Abbildung 3. Der Wecker

Dies ist ein Wecker.

Unglaublich, oder? Dieser winzige Wecker steckt voller winziger Elektronik, die uns morgens weckt. Man könnte sich fragen, ob Wecker heutzutage überhaupt noch benutzt werden, wo doch so viele ihren Wecker auf dem Handy stellen. Es mag nicht mehr viele geben, die ihn verwenden, aber es ist trotzdem spannend zu erfahren, wie er funktioniert!

Welche Elektronik wird verwendet?

- Ein kleiner Computerchip zur Zeitmessung.
- Ein Bildschirm, auf dem die Zahlen angezeigt werden
- Ein Summer, der zum eingestellten Zeitpunkt ein Geräusch erzeugt.

Wenn Sie pünktlich sein möchten, denken Sie bitte daran, Ihren Wecker zu stellen!

3. ELEKTRONIK ALS TEIL DER HOBBYS

Wir haben bereits über den Einsatz von Elektronik im Alltag gesprochen, aber was wäre, wenn sie auch Teil unserer Hobbys sein könnte? Ja, wir können mit Elektronik und Programmierung tolle Aktivitäten gestalten! Lasst uns erkunden, welche spannenden Aktivitäten mit Elektronik möglich sind:

Malen mit Lichteffekten: Jetzt kannst du deiner Fantasie freien Lauf lassen und farbenfrohe Bilder gestalten. Füge ein Licht hinzu, das sich per Knopfdruck ein- und ausschalten lässt. Stell dir vor, du verwendest es zusammen mit einer strahlenden Sonne, um das Bild noch heller zu machen.

Einfaches Nachtlicht: Kommt es Ihnen auch manchmal so vor, als ob Ihr Zimmer abends etwas dunkel wird und Sie vor dem Einschlafen noch etwas Licht brauchen? Integrieren Sie einfach eine Lampe in Ihr Bett und bringen Sie den Knopf in der Nähe an, um sie bei Bedarf einzuschalten. Alternativ können Sie Sensoren hinzufügen, die das Licht automatisch einschalten, sobald es zu dunkel wird.

Drehspielzeug: Eine weitere tolle Aktivität zum Ausprobieren! Wähle deine Farben und dein Papiermuster und setze einen kleinen batteriebetriebenen Motor ein, damit es sich dreht. Achte darauf, dass das Papier rund ist, damit es sich besser dreht, und wähle deine Lieblingsfarben. Und dann sieh zu, wie der Strom dein Papier in etwas Lustiges verwandelt!



Abbildung 4. Der selbstgebaute Spinner

¹ Bildquelle: Frog Prince Paperie – „DIY Spinner Art STEM Project“.

Welche Rolle spielt das Programmieren bei unseren Aktivitäten? Programmieren ist einfach ein Programm, das einem Gerät ohne Worte Anweisungen gibt.

Man kann zum Beispiel eine einfache „Hallo“-Nachricht auf einem Bildschirm oder einer LED-Anzeige programmieren oder den Namen einer Person darauf schreiben und sie ihr geben. Das ist nicht nur eine unterhaltsame Aktivität, sondern auch ein kreatives Geschenk.



Abbildung 5. Die LED-Lichtplatine

Jetzt bist du an der Reihe. Überlege dir, welche anderen spannenden Aktivitäten und Hobbys man mit Elektronik und Programmierung gestalten könnte! Lass deiner Fantasie freien Lauf und schreibe deine Ideen auf!

4. PRAKTISCHE TÜFTELAKTIVITÄTEN

Jetzt kommt der beste Teil: Wir bauen unsere eigenen Schaltkreise! Denk dran, es geht ums Tüfteln. Viel Spaß beim Ausprobieren und Entdecken – und seht, was ihr alles erschaffen könnt!

Aktivität 1: Licht-Erinnerung!

In dieser Aktivität basteln wir ein kleines Licht, das Sie daran erinnern soll, bestimmte tägliche Aufgaben zu erledigen.

Ziel dieser Aktivität:

- Baue eine kleine Lampe, die mit einem Knopf ein- und ausgeschaltet werden kann.
- Verwenden Sie ein einfaches Hilfsmittel, um sich an alltägliche Aufgaben wie das Zähneputzen zu erinnern.

Was Sie benötigen:

- 1 Batteriehalter mit Batterien
- 1 LED-Leuchte (beliebige Farbe)
- 1 Ein-/Ausschalter
- 3-adrige Kabel
- 1 Stück Pappe zur Befestigung
- 2-adrige Kabel, Karton zur Montage



Abbildung 6. Batteriehalter mit Batterien



Abbildung 7. LED-Leuchten





Abbildung 8. Ein-/Ausschalter



Abbildung 9. Die Drahtkabel



Abbildung 10. Der Karton



Kofinanziert von der
Europäischen Union

Lasst es uns gemeinsam schaffen:

1. Nehmen Sie die Batterien und legen Sie sie in den Halter ein.

Achten Sie auf das rote und schwarze Kabel des Batteriehalters.

2. Eine LED hat üblicherweise zwei Anschlüsse: einen längeren Anschluss, der positiv (+) ist, und einen kürzeren Anschluss (negativer Anschluss).

3. Draht 1: Batterie (+) ☐ Ein-/Ausschalter

Jedes Kabelende hat zwei Enden. Verbinden Sie zunächst ein Ende mit dem Pluspol (rot) des Batteriehalters. Das andere Ende schließen Sie an einen der Metallkontakte hinter dem Schalter an.

4. Draht 2: Schalter ☐ LED (+)

In diesem Schritt verbinden wir den längeren Schenkel (+) der LED mit einem zweiten Draht vom anderen Schenkel des Schalters.

5. Draht 3: LED (-) ☐ Batterie (-)

Verbinden Sie nun den kurzen Anschluss der LED mit dem schwarzen (-) Draht des Batteriehalters.

6. Lasst uns sicherstellen, dass es sicher ist!

Bevor wir das Gerät einschalten, stellen wir sicher, dass alles sicher ist. Biten Sie Ihren Vorgesetzten oder die zuständigen Personen, zu überprüfen, ob alle Teile korrekt sind und nichts an den falschen Teilen befestigt ist.

7. Schalten Sie es mit dem Schalterknopf ein und später wieder aus!



Erfolg!

Kommen wir nun zum letzten Punkt: dem Karton. Du kannst ihn mit verschiedenen Aufklebern verzieren oder ein Bild davon malen, was er darstellt! Stell dir eine kleine Erinnerung an deinem Kühlschrank vor, nur eben in elektronischer Form. Solange das Licht an ist, ist die Aufgabe nicht erledigt. Jedes Mal, wenn du das Licht siehst, wirst du an deine täglichen Aufgaben erinnert, wie zum Beispiel die Pflanzen gießen oder Zähne putzen!

Was haben wir gelernt?

Wir haben gelernt, dass uns ein Stromkreis im Alltag helfen kann. Wenn die Bauteile richtig verbunden sind, leuchtet die LED auf und kann als Erinnerung an tägliche Aufgaben dienen.

Fordere dich selbst heraus!

- Können Sie erraten, welche Farbe sich besser für Erinnerungen eignet, und versuchen, LEDs in verschiedenen Farben auszuprobieren?
- Welche anderen Möglichkeiten könnten wir Ihrer Meinung nach für Erinnerungen entwickeln?

Aktivität 2: Ihr persönliches Namensschild

Im Alltag fällt es uns manchmal schwer, auf andere zuzugehen und uns vorzustellen. Was wäre, wenn wir einen intelligenten Ausweis verwenden würden, der uns unsere Identität verrät? In dieser Übung erstellen wir einen persönlichen Ausweis, der auf Knopfdruck aufleuchtet.

Ziel dieser Aktivität:

- Um einen Schalter zu verwenden, um das Etikett einzuschalten.
- Um ein Etikett mit Licht zu erstellen, damit es sichtbar wird



Was Sie benötigen:

Die meisten Teile, an die wir uns aus der ersten Aktivität erinnern, sollten wir noch einmal in Erinnerung rufen:

- 1 Batteriehalter mit Batterien
- 1 LED-Leuchte (Wählen Sie Ihre Lieblingsfarbe)
- 1 kleiner Schalter oder Druckknopf
- 3 Drähte mit Klemmen (vorzugsweise in verschiedenen Farben für eine eindeutige Anleitung)
- 1 Stück Pappe (Größe für Namensschilder anpassen)
- Stifte oder Aufkleber zum Verzieren des Kartons
- Klebstoff (um das Gerät auf dem Karton zu befestigen)



Abbildung 11. Batteriehalter mit Batterien



Abbildung 12. Karton



Abbildung 13. Ein-/Ausschalter



Kofinanziert von der
Europäischen Union



Abbildung 14. 3 Drähte mit Klemmen



15. Klebstoff



Abbildung 16. Stifte und Aufkleber zur Dekoration



Kofinanziert von der
Europäischen Union

Zeit, es Schritt für Schritt aufzubauen:

1. Fangen wir mit dem Namensschild an!

Schneide ein kleines Stück Pappe aus (dein Name sollte hineinpassen) und beschrifte es mit einem Filzstift. Für mehr Farbe kannst du Aufkleber verwenden und es nach Belieben verzieren.

2. Entscheide, auf welchen Buchstaben oder welchen Teil das Licht leuchten soll.

Jetzt gilt es, den Teil des Namensschildes auszuwählen, der leuchten soll, und ein kleines Loch zu bohren, durch das die Beinchen der LEDs geführt werden.

3. Platzieren Sie die LED

Merken Sie sich zuerst, welcher Anschluss der LED positiv und welcher negativ ist. Schieben Sie dann die LED durch das Loch und justieren Sie die Anschlüsse so, dass sie stabil stehen.

4. Verbinden Sie alle drei Drähte.

- Schließen Sie zuerst das rote Kabel vom Batteriehalter an einen Anschluss hinter dem Schalter an.
- Schließen Sie einen weiteren Draht an den längeren Schenkel der LED (den positiven) an, der durch die andere Seite des Schalters geführt wird.
- Das schwarze Kabel vom Batteriehalter sollte an das dritte Kabel des kurzen Anschlusses (Minuspol) der LED angeschlossen werden.

5. Vergewissern Sie sich, dass es sicher ist, und schalten Sie es ein!

Bitten Sie Ihren Vorgesetzten, zu überprüfen, ob alle Teile sicher und fest angebracht sind. Bitten Sie ihn außerdem um Hilfe beim Kleben oder Festkleben der Teile auf dem Etikett, damit sie nicht abfallen.

Erfolg!

Herzlichen Glückwunsch! Sie haben jetzt einen Leucht-Anstecker. Sie können ihn bei Veranstaltungen oder überall dort tragen, wo Sie das Eis brechen und sich vorstellen möchten!

Was haben wir gelernt?

Wir haben gelernt, dass wir mit Elektronik lustige Aktivitäten gestalten können!

Fordere dich selbst heraus!

- Kann man eine weitere LED hinzufügen, sodass zwei Buchstaben leuchten?
- Kann man den Namen einer anderen Person auf ein Etikett schreiben und es zum Leuchten bringen?



5. PRAKTISCHE PROGRAMMIERUNGSÜBUNG

- Computer mit MakeCode (oder ähnlicher Software)

Wie in den vorherigen Modulen wird Ihnen Ihr Dozent bei der Einrichtung behilflich sein.

Schritt-für-Schritt-Anleitung:

1. Öffnen Sie Ihr Programmierprogramm und wählen Sie ein neues Projekt.
2. Um der Lampe mitzuteilen, wann sie angehen soll, fügen Sie einen Block „wenn Taste A gedrückt wird“ hinzu.
3. Füge einen weiteren Block „LED einschalten“ hinzu.
4. Geben Sie die Zeit an, für die das Licht eingeschaltet bleiben soll.

Als Nächstes aktivieren Sie die Funktion „LED ausschalten“.

Fertig! Drücken Sie den Knopf und testen Sie Ihre Lampe.

Hinweis für Lehrkräfte: Fragen Sie Ihre Schüler, wo es nützlich sein könnte – zum Beispiel als Erinnerung daran, nach dem Schreiben eine Pause einzulegen und die Hände auszuruhen.

Erfolg!

Gut gemacht! Du hast eine Lichterinnerung programmiert, die dir bei alltäglichen Aufgaben wie „Schicke eine Nachricht an deine Eltern“ oder „Bring den Müll raus“ hilft.

Was haben wir gelernt?

Wir haben gelernt, dass Programmieren in unseren täglichen Aktivitäten Anwendung finden kann und dass man mit Programmieren kleine alltägliche Herausforderungen lösen kann.

Fordere dich selbst heraus!

- Können Sie die Wartezeit auf 10-15 Sekunden ändern?



6. ÜBERALL CLEVERE IDEEN

Jetzt sind Sie an der Reihe, Ihre Fantasie einzusetzen! Lassen Sie uns Ideen sammeln und austauschen, um zu verstehen, wie Elektronik und Programmierung mit dem Alltag zusammenhängen.

Ziel dieser Aktivität:

- Um mein Wissen aufzufrischen und weitere Ideen darüber zu entdecken, wie Programmierung und Elektronik mit unserem Alltag zusammenhängen.
- Um kreative und unterhaltsame Aktivitäten im Zusammenhang mit zukünftigen Projekten zu teilen.

Was Sie benötigen:

- Eine kleine Tafel oder ein Blatt Papier, um Ideen aufzuschreiben.
- Filzstifte oder Bleistifte
- Man kann auch Bilder aus Tagebüchern nehmen und sie auf Papier kleben.

Wo anfangen?

1. Denken Sie an elektronische Geräte, die Sie im Alltag verwenden. Das kann zu Hause oder in der Schule sein – überall.
2. Beschriften Sie die einzelnen Elektronikgeräte oder verwenden Sie Aufkleber oder Bilder davon.
3. Schreiben Sie neben jeden Eintrag dessen Namen und Funktion.
4. Überlege dir eine neue Idee für einen einfachen elektronischen Alltagshelfer und füge sie deiner „To-do-Liste“ für zukünftige Projekte hinzu.

Hinweis für Lehrkräfte: Bitten Sie Ihre Schüler, darüber zu sprechen, was Elektronik und Programmierung für sie interessant macht.



7. SCHLUSSFOLGERUNG

Gut gemacht! Modul 4: Elektronik in unserer Welt: Von Hobbys zu Möglichkeiten ist nun abgeschlossen.

Du verstehst jetzt, wie du Elektronik und Programmierung im Alltag einsetzen kannst. Erforsche, baue und entdecke, wie Technologie den Alltag einfacher und unterhaltsamer gestalten kann.

In dieser Lektion hast du deine Hände und deinen Verstand eingesetzt, um Schaltkreise zu testen, Drähte zu verbinden und sogar einfache Programme zu schreiben. Elektronik ist aber viel mehr als nur das Funktionieren von Geräten. Es kann richtig Spaß machen, Neues zu lernen und sein Können unter Beweis zu stellen.

Es gibt viele Wege, zu denken, zu bauen und Probleme zu lösen. Vom Einschalten einer Glühbirne bis hin zur Entwicklung einer eigenen genialen Idee – du kannst es auf deine Weise tun. Bleib neugierig, zeig Interesse und scheu dich nicht, Neues auszuprobieren. Jedes noch so kleine Projekt lehrt dich etwas Neues über Technologie und wie du sie optimal einsetzt.



8. WEITERFÜHRENDE LITERATUR

Wenn Ihnen dieses Modul gefallen hat und Sie mehr lernen möchten, finden Sie hier einige großartige Online-Seiten, auf denen Sie Ihr Wissen vertiefen können.

1. Science Buddies – Webseite für Elektronikprojekte für Anfänger:
<https://www.sciencebuddies.org>

Die einfachen, schrittweisen Projekte auf dieser Website zeigen, wie elektrische Schaltkreise und Bauteile funktionieren. Es gibt viele spannende und sichere Projekte, die man mit einfachen Werkzeugen durchführen kann.

2. Teach Engineering – Praktische MINT-Aktivitäten Webseite:
<https://www.teachengineering.org>

Diese Website bietet Projekte, die im Unterricht erprobt wurden und Technologie, Kreativität und Problemlösungskompetenz miteinander verbinden. Hier finden Sie Aufgaben für Anfänger, die Schaltkreise mit Erfindungen aus der realen Welt verknüpfen.

3. Kodable – Programmieren lernen durch Spiele Webseite:
<https://www.kodable.com>

Eine farbenfrohe und unterhaltsame Lernumgebung, in der Kinder mithilfe von Drag & Drop niedliche Charaktere gestalten können. Das fördert logisches Denken und vermittelt spielerisch die Grundlagen des Programmierens.

4. Instructables – Einfache Website für Elektronik-Bastelprojekte:
<https://www.instructables.com>

Eine benutzerfreundliche Website mit vielen Schritt-für-Schritt-Anleitungen für eigene kleine Technikprojekte. Im Bereich „Einfache Elektronik“ mit anschaulichen Bildern und kurzen Videos können Sie in Ihrem eigenen Tempo lernen.



9. LITERATURVERZEICHNIS

Auszug/Idee aus dem Modul	Quelle (APA 7. Auflage)	Typ
Beschreibung der Arduino-basierten Programmieraktivität	Arduino. (o. J.). *Arduino – Open-Source-Elektronikplattform*. Abgerufen von https://www.arduino.cc	Lernwerkzeug / Programmierressource
Bezugnahme auf MakeCode in der Programmieraktivität	BBC MakeCode. (o. J.). *MakeCode: Programmieren lernen mit spannenden Projekten*. Abgerufen von https://makecode.microbit.org	Bildungsplattform für Programmierung
Bildunterschrift: „DIY-Spinner-Kunstprojekt (STEM)“	Frog Prince Paperie. (o. J.). *DIY-Spinner-Kunstprojekt (STEM)*. Abgerufen von https://frogprincepaperie.com	Bildquelle
Beschreibung von Elektronikprojekten für Anfänger	Science Buddies. (o. J.). *Elektronikprojekte für Anfänger*. Abgerufen von https://www.sciencebuddies.org	Bildungswebsite
Bezug zu TeachEngineering-Klassenzimmerprojekten	TeachEngineering. (o. J.). *Praxisorientierte MINT-Aktivitäten*. Abgerufen von https://www.teachengineering.org	STEM-Bildungsressource
Erwähnung der Kodable-Plattform für Programmieren durch Spiele	Kodable. (o. J.). *Programmieren lernen durch Spiele*. Abgerufen von https://www.kodable.com	Bildungswebsite
Verweis auf Instructables für Elektronik-Bastelarbeiten	Instructables. (o. J.). *Einfache Elektronik-Bastelprojekte*. Abgerufen von https://www.instructables.com	DIY-Projektplattform



9. LITERATURVERZEICHNIS

Auszug/Idee aus dem Modul	Quelle (APA 7. Auflage)	Typ
Allgemeine Abbildungen von LEDs , Batterien und Schaltkreisen	Pixabay. (o. J.). *Kostenlose Bilder für Bildungszwecke und nichtkommerzielle Nutzung*. Abgerufen von https://pixabay.com	Bildarchiv
Illustrative Bildbeispiele für Unterrichtsmaterialien	Unsplash. (o. J.). *Kostenlose Bilder zu Bildung und Technologie*. Abgerufen von https://unsplash.com	Bildarchiv



10. SELBSTBEWERTUNGSRASTER

Mal sehen, was du noch weißt! Wähle für jede Frage die beste Antwort aus. Die richtige Antwort ist fett gedruckt.

Nr.	Frage	A	B	C	D
1	Wie nennt man das Lernen durch Bauen und Erkunden von Elektronik mit den Händen?	Videos ansehen	Über Stromkreise lesen	Lernen durch Handeln (Learning by doing)	Von anderen kopieren
2	Welcher Alltagsgegenstand nutzt Elektronik?	Ein Kissen	Eine Waschmaschine	Ein Buch	Ein Löffel
3	Welches Teil liefert den Strom für einen elektronischen Schaltkreis?	Ein Draht	Eine Batterie	Eine Glühbirne	Ein Schalter
4	Was ist die Aufgabe eines Schalters?	Das Licht heller zu machen	Einen Schaltkreis ein- oder auszuschalten	Energie zu speichern	Batterien zu verbinden
5	Welches einfache Hobby kann sowohl Elektronik als auch Coding beinhalten?	Ein blinkendes Licht-Spielzeug bauen	Mit Buntstiften zeichnen	Einen Kuchen backen	Mit Papier bauen
6	Welches Werkzeug hilft uns, Licht oder Ton mit einem Computer zu steuern?	Eine Tastatur	Ein micro:bit Board	Eine Büroklammer	Eine Glühbirne
7	Was benutzen wir beim Block-Coding, um dem Computer zu sagen, was er tun soll?	Sätze	Bunte Code-Blöcke	Lieder	Bilder



Nr.	Frage	A	B	C	D
8	Wie nennt man einen vollständigen Pfad, durch den Strom fließt?	Einen Kreis	Einen Schaltkreis	Ein Seil	Einen Drahtkasten
9	Was ist der Zweck des Codierens in Elektronikprojekten und wie unterscheidet es sich vom Tinkering?	Während Tinkering bedeutet, dem Gerät Anweisungen zu geben, bedeutet Codieren, Komponenten physisch zu verbinden.	Während Tinkering das Erstellen von Schaltkreisen mit realen Komponenten ist, ist Codieren das Schreiben von Anweisungen, die dem Gerät sagen, was es tun soll.	Das Zeichnen von Schaltkreisen auf Papier ist Teil von Codieren und Tinkering.	Tinkering und Codieren sind gleichwertig, da beide das Ein- und Ausschalten eines Geräts beinhalten.
10	Was macht eine LED in einem Schaltkreis und woran erkennt man, welche Seite positiv oder negativ ist?	Speichert Elektrizität; beide Beinchen sind gleich lang.	Gibt einen Ton ab, wenn Strom fließt; das längere Beinchen ist positiv geladen (+), das kürzere Beinchen ist negativ geladen (-).	Leuchtet, wenn Strom fließt; das längere Beinchen ist positiv geladen (+), das kürzere Beinchen ist negativ geladen (-).	Kühlt den Schaltkreis; das längere Beinchen ist negativ geladen (-).

MODUL 5: MEIN EIGENES PRODUKT – PERSONALISIERUNG & UNTERHALTSAMES FEEDBACK



Überblick

Dieses Modul unterstreicht, wie wichtig es ist, Lernen individuell und unterhaltsam zu gestalten. Für Erwachsene mit Down-Syndrom wird das Erlernen von Elektronik und Programmierung effektiver, wenn Aktivitäten Wahlmöglichkeiten, Kreativität und positives Feedback beinhalten.

Die Lernenden werden ermutigt, Projekte ihren Vorlieben entsprechend anzupassen (z. B. die Farbe der LEDs zu wählen, Geräusche hinzuzufügen, die ihnen gefallen, ihre Platinen zu gestalten). Dieses Gefühl der Eigenverantwortung steigert Motivation, Engagement und Stolz.

Gleichzeitig betont das Modul den Wert von unmittelbarem Feedback. Wenn beim Drücken eines Knopfes sofort eine LED aufleuchtet oder beim Ändern eines Codeblocks ein Ton abgespielt wird, verstehen die Lernenden den Zusammenhang zwischen Aktion und Ergebnis. Dies stärkt das Selbstvertrauen und lässt das Lernen spielerisch statt stressig erscheinen.

Schließlich beinhaltet das Modul positives Feedback von Gleichaltrigen. Die Lernenden präsentieren ihre Projekte in einer sicheren und unterstützenden Umgebung, erhalten Ermutigung von ihren Mitschülern und lernen, wie sie konstruktives Feedback in einfachen, leicht verständlichen Formaten geben können.

Vereinfachtes Sprachbeispiel (Leicht lesbar):

- „Ich kann Farben und Töne auswählen, die mir gefallen.“
- „Wenn ich einen Knopf drücke, sehe oder höre ich etwas passieren.“
- „Meine Freunde erzählen mir, was ihnen an meinem Projekt gefallen hat.“
- „Ich bin stolz, weil mein Projekt mir gehört.“

Diese Kombination aus Personalisierung, unmittelbarem Feedback und Anerkennung durch Gleichaltrige sorgt dafür, dass Elektronik und Programmierung nicht nur technische Übungen sind, sondern auch Werkzeuge für Kreativität, Selbstausdruck und Spaß.



Ziel

- Die Lernenden sollen durch die Möglichkeit, ihre Projekte zu personalisieren, gestärkt werden.
- Um zu zeigen, wie unmittelbares Feedback (Lichter, Töne, Bewegungen) das Verständnis und die Motivation fördert.
- Die Lernenden sollen dazu ermutigt werden, auf strukturierte und einfache Weise positives Feedback von Gleichaltrigen zu geben und zu erhalten.
- Elektronik und Programmierung sollen zu einem Raum für Spaß, Kreativität und Eigenverantwortung werden.
- Ziel ist es, das Selbstvertrauen und den Stolz der Lernenden auf ihre individuellen Leistungen zu stärken.

Erwartete Ergebnisse

Am Ende dieses Moduls werden die Lernenden in der Lage sein:

1. Ein Projekt so anpassen, dass es die eigenen Vorlieben widerspiegelt (z. B. Farben, Muster, Klänge, Dekorationen).
2. Den Zusammenhang zwischen Aktionen und Ergebnissen im Bereich Programmierung/Elektronik durch unmittelbares Feedback erkennen.
3. Sie sollen ihre Arbeit mit Kollegen teilen und auf positives Feedback reagieren.
4. Sie werden sich selbstsicherer und motivierter fühlen, weiterzulernen, weil die Aktivität Spaß macht und persönlich ist.



1. EIN PROJEKT AN DIE EIGENEN VORLIEBEN ANPASSEN

Eines der Hauptziele dieses Moduls ist es, den Lernenden zu zeigen, dass sich Technologie an ihre eigenen Ideen und Vorlieben anpassen lässt. Dies ist entscheidend für die Motivation: Wenn Lernende sich für ein Projekt verantwortlich fühlen, sind sie engagierter, aufmerksamer und stolzer auf das Ergebnis.

Praktische Beispiele für Personalisierung sind:

- **Auswahl der LED-Farben:** Jeder Lernende kann die Farbe wählen, die er bevorzugt (z. B. Rot für Energie, Blau für Ruhe, Grün für Natur).
- **Individuelle Mustergestaltung:** Anstatt dass alle Lichter auf die gleiche Weise blinken, können die Lernenden entscheiden, ob sie schnelles Blinken, langsames Blinken oder eine bestimmte Sequenz wünschen.
- **Verzieren:** Die Lernenden können Aufkleber, Filzstifte oder kleine Pappformen verwenden, um die Schaltbox individuell zu gestalten.
- **Klangauswahl:** Wenn das Projekt Geräusche beinhaltet, können die Lernenden eine Melodie oder ein lustiges Geräusch auswählen, das ihnen gefällt.

Pädagogische Auswirkungen: Personalisierung fördert die Selbstentfaltung, die in traditionellen Lernumgebungen oft eingeschränkt ist. Sie verbindet technisches Lernen mit der Identität und den persönlichen Präferenzen des Lernenden.

Vereinfachte Sprache:

- „Ich wähle meine Farbe.“
- „Ich mache mein Projekt zu etwas Besonderem.“
- „Es gehört mir.“



Praktische Übung 1 – Meine Licht- und Tonsignatur

Zweck

Die Lernenden übernehmen Verantwortung, indem sie Aussehen und Klang ihres Projekts selbst bestimmen. Sie wählen Farben/Muster für LEDs und einen einfachen Ton (Piepton/Summer), verbinden die Bauteile sicher und testen die Funktion für sofortiges Feedback. Klare, leicht verständliche Anleitungen und große Symbole erleichtern den Einstieg.

Dauer

~90–120 Minuten (Demo → Partnerarbeit → Präsentation).

Materialien

- Akku oder Einsteiger-Mikrocontroller (jede Platine, die eine LED blinken lassen / einen Summer piepen lassen kann)
- 1–2 LEDs (oder eine mehrfarbige LED), ggf. Widerstände, kleiner Summer oder Lautsprecher
- 1 Knopf oder Touchpad (Folie + Draht funktioniert)
- 8–10 farbcodierte Leitungen
- Laptop/Tablet mit einem blockbasierten Editor (große Symbole) oder einem sicheren Simulator
- Gedruckte Parameterkarten (Farbe, Geschwindigkeit, Muster, Ton), Schrittarten (A5, Foto zuerst)
- Rollenkarten (Bauarbeiter / Helfer)
- Große Piktogramme (Strom / Draht / Test)

Sicherheit 3 (Pfosten an der Wand)

- Zum Umstecken der Kabel muss die Stromzufuhr unterbrochen werden.
- Saubere Kabel (keine losen Metallteile, die die Batterie berühren).
- Erst fragen, dann Werkzeuge verwenden; erst dann testen.

Barrierefreiheitsanker

- Eine Idee pro Zeile. Große Fotos und Symbole. Kurze Sätze.
- Unmittelbares sensorisches Feedback (Licht/Ton) nach jedem Schritt zur Bestätigung des Erfolgs.
- Rollenwechsel zwischen Bauarbeitern und Helfern mit einem Wechsel während der Sitzung.
- Satzmuster und Piktogramme zur Auswahl: „Ich wähle Rot“, „Ich mag langsames Blinken“, „Ich wähle Piepton“.



Abbildungen (Platzhalter zum späteren Einfügen)

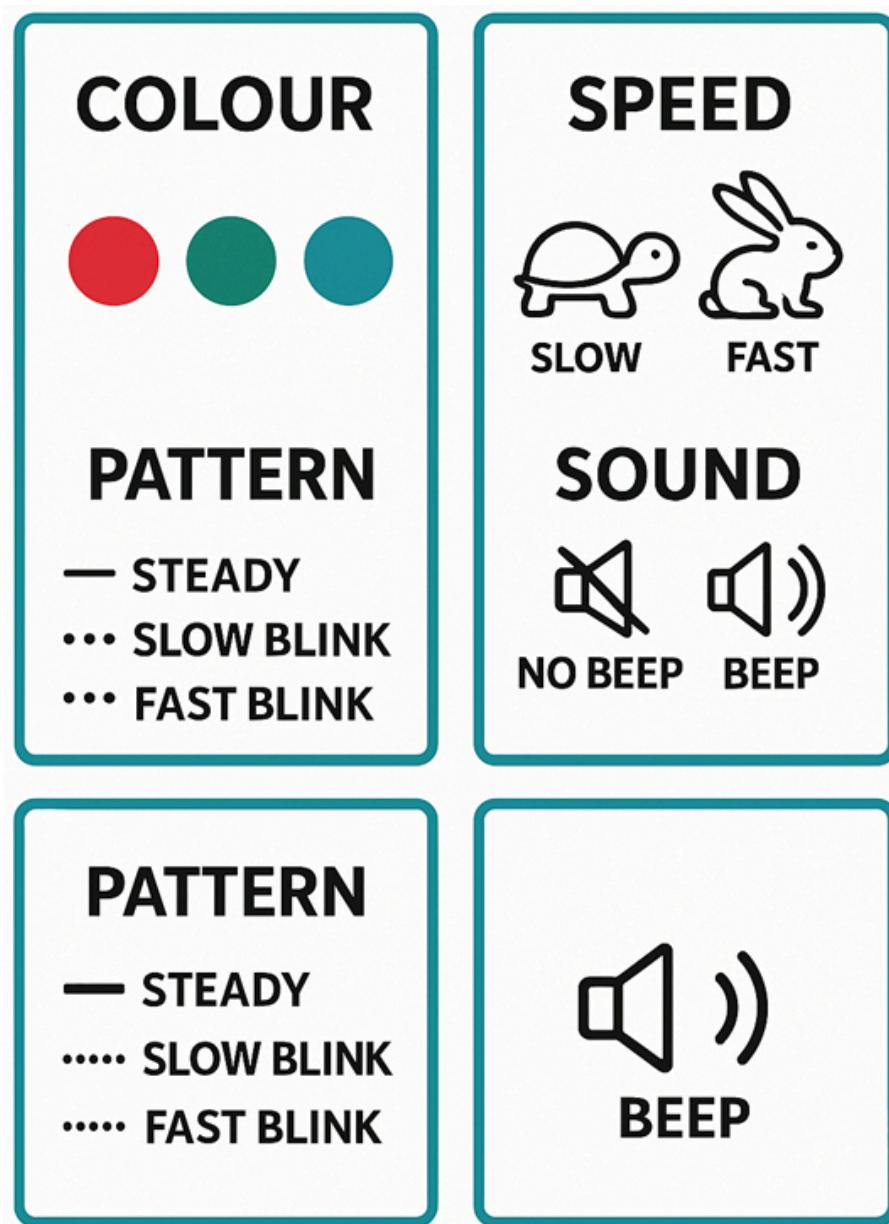


Abbildung 1. Parameterkarten (Farbe, Geschwindigkeit, Muster, Ton).

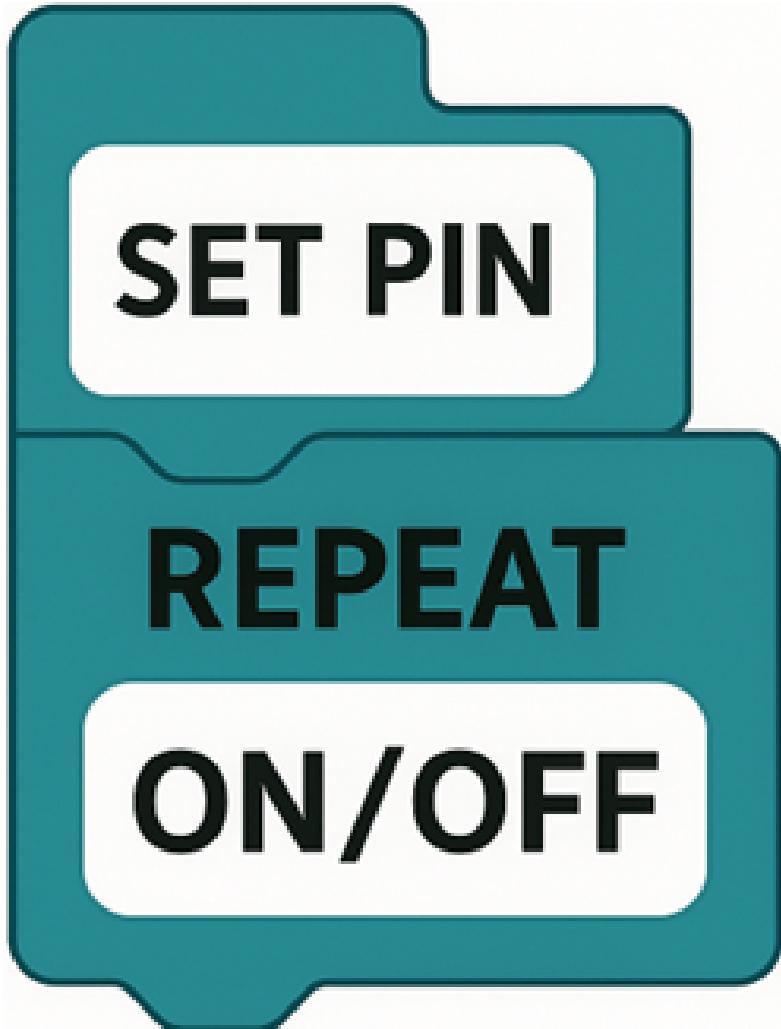


Abbildung 2. Beispiel eines Blockstapels (Set-Pin → Wiederholung Ein/Aus).

Ziele

- Wähle Look & Feel (Farbe/Muster/Sound) und sage, warum es dir gefällt.
- Verdrahten Sie die Kabel sicher und verwenden Sie die 3-Schritt-Fehlerbehebung, falls es nicht funktioniert.
- Testen und zeigen Sie die personalisierte Licht-/Tonfunktion.

Zeit & Gruppierung

- 10–12 Minuten Demo
- 30–40 Minuten Paarbildung & Test
- 10–15 Minuten zum Teilen
- 10 Minuten zum Nachdenken.

Rollen: Bauarbeiter (praktische Tätigkeiten) / Helfer (liest die Anweisungen und prüft die Sicherheit). Rollentausch nach der Hälfte der Arbeitszeit.

Schritte für Lernende

- **Wählen Sie (zeigen und sagen Sie):**
a) Farbe (rot/grün/blau), b) Muster (schnell/langsam/gleichmäßig), c) Ton (Piepton/kein Piepton).

- **Strom ausschalten. Teile anschließen:**

Batterie/Netzteil → Taste/Touchpad → LED (+) und Summer (+).

LED (–) / Summer (–) → Masse/Minuspol.

- Beim Programmieren: Öffnen Sie den Blockeditor und fügen Sie die minimalen Blöcke hinzu:

Pin setzen (LED oder Summer) → Wiederholen: Ein → Warten → Aus → Warten.

Passen Sie die Blockoptionen an Ihre Farbe/PIN und Geschwindigkeit an.

- **Einschalten → Ausführen/Testen. Sehen Sie Licht / hören Sie einen Ton?**
- **Die Rollen tauschen und eine Auswahl überarbeiten oder anpassen (z. B. schnelleres Blinzeln oder anderer Ton).**
- **Foto + Bildunterschrift (leicht lesbar): „Mein Stil: blau, langsames Blinzeln, Piepton.“**

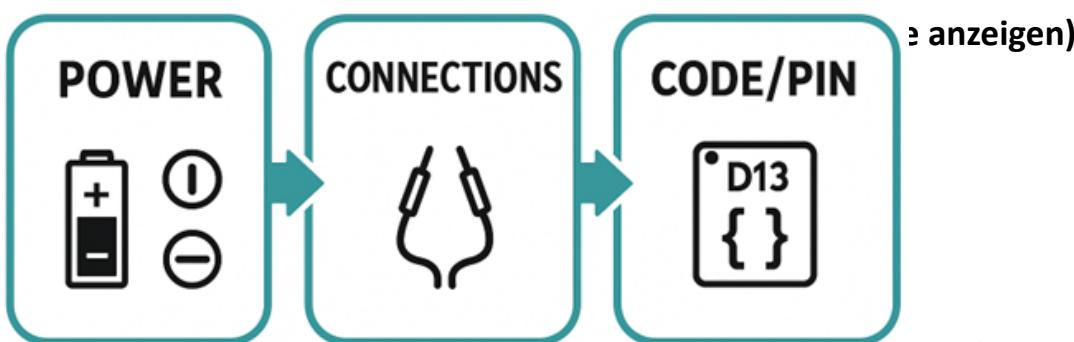


Abbildung 3. Debug-Leiste (Stromversorgung → Anschlüsse → Code/Pin).



- A) Stromversorgung: Batterie eingelegt? Schalter eingeschaltet? Platine angeschlossen?
- B) Verbindungen: Lose/vertauschte Kabel? LED richtig herum (+/-)?
- C) Code/PIN: Richtige PIN ausgewählt? Lange genug warten, um sie zu sehen/hören?

Hilfeskript

„Ich habe ___ versucht. Das Problem ist ___. Können Sie sich ___ ansehen?“

Differenzierung & AAC (Unterstützte und alternative Kommunikation)

- **Einfacher:** Eine reine Foto-Verdrahtungskarte verwenden; eine einzelne LED mit einem Geschwindigkeitsregler verwenden; kein Ton.
- **Schwieriger:** Mehrfarbige LED zulassen; zweiten Ausgang hinzufügen (LED + Summer); Lernende ein zweistufiges Muster entwerfen lassen (z. B. lang–kurz).
- **AAC:** Symbolkarten für Auswählen / Verbinden / Testen / Mögen; Farbmuster zum Zeigen.

Maßnahmen der Moderatoren

- Treffen Sie konkrete Entscheidungen: Zeigen Sie auf eine Karte, sagen Sie es aus und setzen Sie es dann um.
- Fragen Sie sich nach jeder Veränderung: „Was hat sich verändert? Was sehen/hören Sie jetzt?“
- Wenn Sie nicht weiterkommen, führen Sie die 3-Schritte-Fehlerbehebung per Geste durch, anstatt alles erneut zu erklären (tippen Sie auf das Power-Symbol → folgen Sie dem Kabel mit dem Finger → zeigen Sie auf Code/Pin).

Mini-Checks (während der Aktivität)

- „Zeig mir mit dem Finger deine Eingaben und Ausgaben.“
- „Beschreiben Sie Ihr Verhalten in einem Satz: ,Wenn ich drücke, blinkt es blau und piept.““

Checkliste für Beweise und Beobachtungen (pro Paar ankreuzen)

- Rollen übernommen und einmal gewechselt
- Strom-Aus-Regel beim Neuverkabeln befolgt
- Projekt reagierte (Licht/Ton) auf die gewählten Einstellungen
- 3-Schritt-Debugging durchgeführt, bevor um Hilfe gebeten wurde
- Foto + kurze Beschreibung des eigenen Stils bereitgestellt



Kurze Präsentation (10–15 Min.)

- Die Paare legen ihr Gerät auf den Tisch. Ein Nachbar drückt den Knopf/das Touchpad und beschreibt das Geschehene in einem Satz.
- Der Moderator sagt: „Wir haben uns für ___ entschieden, weil ___.“
- Besucher vergeben ★★ + 💡 (zwei Sterne und einen Wunsch). Gastgeber notieren eine Idee auf einem Streifen.

Reflexion (Abschlussfragebogen; leicht verständlich)

★ „Eine Sache, die ich personalisiert habe: _____.“ 🔧 „Eine Idee für das nächste Mal: _____.“

Vorbereitung auf die praktische Übung 2 (Was kommt als Nächstes?)

- Bewahren Sie Ihre Parameterkarte (Farbe/Muster/Ton) auf.
- In der nächsten praktischen Übung fügen wir eine kleine Spielregel hinzu: „Nach 3 Erfolgen → Feier (Ihre Licht- und Tonshow).“



2. DEN ZUSAMMENHANG ZWISCHEN AKTIONEN UND ERGEBNISSEN IN DER PROGRAMMIERUNG/ELEKTRONIK DURCH DIREKTES FEEDBACK ERKENNEN

Unmittelbares Feedback ist eines der effektivsten Lerninstrumente für Lernende mit Lernschwierigkeiten, da es die Distanz zwischen Handlung und Ergebnis verkürzt. Wenn etwas sofort passiert, können Lernende den Ursache-Wirkungs-Zusammenhang verstehen.

Beispiele:

- Durch Drücken eines Knopfes → LED leuchtet auf.
- Wird ein Codeblock geändert, ändert sich der Ton sofort.
- Anschluss eines Sensors → der Motor beginnt sich zu drehen.

Dieser Feedback-Kreislauf ermöglicht es den Lernenden:

- **Erfolg bestätigen:** „Ja, es funktioniert.“
- **Fehler schnell erkennen:** „Nichts ist passiert → vielleicht habe ich ein Kabel vergessen.“
- **Feiere kleine Erfolge:** Jeder Blinker oder Piepton ist ein Grund, stolz zu sein.

Pädagogische Wirkung: Feedback stärkt das Selbstvertrauen und reduziert Frustration. Lernende erleben, dass ihr Handeln zählt und dass sie die Technologie beherrschen können. Es fördert zudem die Problemlösungsfähigkeit – wenn etwas nicht funktioniert, wissen die Lernenden, dass sie die Ursache überprüfen und es erneut versuchen müssen.

Vereinfachte Sprache:

- „Ich drücke → ich sehe Licht.“
- „Ich verändere mich → ich höre Geräusche.“
- „Wenn es nicht klappt, versuche ich es erneut.“



Praktische Übung 2 – Reaktionsglühen

Zweck

Ein einfaches Aufmerksamkeitsspiel, das das Ursache-Wirkungs-Denken und das Timing fördert. Die Lernenden drücken, sobald die LED aufleuchtet; je schneller die Reaktion, desto länger die Belohnung.

Materialien

- Ihre praktische Übung 1 Gerät (Knopf oder Touchpad + LED(s) + Summer)
- Laptop/Tablet mit blockbasiertem Editor (große Symbole) oder sicherem Simulator
- Gedruckter Ablaufstreifen („Berühren → +1 → wenn 3 → Feier“) und Block-Hinweiskarte (Zähler, Wenn/Dann, Zurücksetzen)
- **Punktemarker (drei Kreise) • Schrittakten (A5, Foto zuerst)**
- Hilfeskript- und Debug-Streifenkarten

Schritte (A5-Karte)

10. Programmiere eine zufällige Wartezeit (1–3 s), dann LED EIN.
11. Der Lernende drückt, sobald die LED aufleuchtet.
12. Bei schnellem Drücken → lange Jubelfeier abspielen; andernfalls → kurze Jubelfeier abspielen.
13. Zurücksetzen und erneut versuchen. Zwei Punktzahlen (Smiley-Tokens) speichern:
😊 (schnell) | 😊 (okay).

Differenzierung

- **Einfacher:** Zeitschaltuhr entfernen; jeder Tastendruck nach dem Einschalten der LED = Jubel.
- **Schwieriger:** Bestzeit protokollieren; zwei Feierebenen zulassen.

Beobachtungscheckliste (pro Paar ankreuzen)

- Zähler funktioniert (0→1→2→3→Feier→0) Lernender hat die Regel einmal genannt Debug-Funktion vor der Bitte um Hilfe verwendet Feier sichtbar/hörbar Peer-Feedback gegeben (★★ +💡)



Teamroutinen – Mini-Galeriespiel (10–15 Min.)

- Jedes Paar verfügt über eine Station mit dem Motto „3 Berührungen = Feier“. Besucher drücken die Tasten und schauen zu.
- Besucher vergeben Sterne ★★ + 💡 und sagen: „Mir hat ___ gefallen; vielleicht nächstes Mal ___.“
- Die Gastgeber notieren jeweils eine Idee auf ihrer Matte.

Reflexion (Abschlussfragebogen; leicht verständlich)

★ „Unsere Regel lautete: _____.“ 🔧 „Nächstes Mal ändern wir sie: _____.“



3. FÜHLEN SIE SICH SELBSTBEWUSST UND MOTIVIERT, WEITERZULERNNEN, WEIL DIE AKTIVITÄT SPASS MACHT UND PERSÖNLICH IST.

Das Ziel dieses Moduls ist nicht nur die Vermittlung von Fachwissen, sondern auch ein gesteigertes Selbstvertrauen und eine erhöhte Motivation. Durch die Kombination von Personalisierung, direktem Feedback und positivem Austausch mit anderen Lernenden entdecken diese, dass Lernen Spaß machen und sinnstiftend sein kann.

Praktische Zeichen des Vertrauens:

- Die Lernenden lächeln und zeigen Stolz auf ihre Projekte.
- Die Lernenden erklären ihr Projekt anderen ohne Angst.
- Lernende melden sich freiwillig, um neue Aufgaben auszuprobieren.

Langfristige Wirkung: Lernende sind nach Projektende eher geneigt, weiterzulernen – zu Hause, mit Freunden oder in neuen Kursen. Freude weckt Neugier, und Neugier führt zu lebenslangem Lernen.

Vereinfachte Sprache:

- "Ich bin stolz."
- „Lernen macht Spaß.“
- „Ich möchte mehr lernen.“



EVALUIERUNG & SELBSTEINSCHÄTZUNG

Mal sehen, was du noch weißt! Wähle für jede Frage die beste Antwort aus. Die richtige Antwort ist fett gedruckt.

Nr.	Frage	A	B	C	D
1	Was ist der sicherste erste Schritt beim Bau deines Projekts?	Verbinde die Drähte nach Belieben.	Testen Sie vor dem Ausschalten.	Vor dem Umstecken der Kabel die Stromzufuhr unterbrechen.	Lose Kabel ignorieren.
2	Dein Projekt zu personalisieren bedeutet...	Es genau wie das deines Partners aussehen zu lassen	Farben, Muster oder Töne zu wählen, die dir gefallen	Verwendung nur einer LED	Allen Anweisungen ohne Änderungen zu folgen
3	Unmittelbares Feedback passiert, wenn...	Man einige Minuten wartet, um Ergebnisse zu sehen	Du die Rückmeldung des Lehrers brauchst, ob es funktioniert hat.	Man einen Knopf drückt und sofort etwas sieht oder hört.	Sie das Ergebnis für später notieren.
4	Was solltest du in der Aktivität „Reaction Glow“ tun?	Den Knopf jederzeit drücken	Den Knopf drücken, wenn die LED aufleuchtet.	Warten, bis der Summer zweimal piept	Die LED manuell ausschalten
5	Was ist der erste Schritt der 3-stufigen Debug-Routine?	Die Codeblöcke bearbeiten	LED-Farbe ändern	Überprüfen Sie die Stromversorgung und die Anschlüsse.	Bitten Sie Ihren Partner, es zu reparieren.
6	Was hilft dir, während des Moduls motiviert zu bleiben?	Wettbewerb, um jede Aufgabe zu gewinnen	Spaß haben und den eigenen Fortschritt sehen	Andere Projekte kopieren	So schnell wie möglich fertig werden
7	Gutes Feedback unter Gleichaltrigen sieht so aus...	„Es ist schlecht.“	„Du hast alles falsch gemacht.“	Zwei positive Punkte und eine Idee zur Verbesserung	Kein Kommentar



Nr.	Frage	A	B	C	D
8	Welcher Satz ist eine klare Bildunterschrift für dein Projekt?	„Piep.“	„Blaues langsames Blinken, weil ich beruhigende Lichter mag.“	„Ich habe etwas gemacht.“	„Mein Projekt ist gut.“
9	Wenn etwas nicht funktioniert, was solltest du zuerst tun?	Die 3-stufige Fehlerdiagnose (Debug) versuchen	Sofort die Lehrkraft rufen	Alle Teile entfernen und von vorne beginnen	Darauf warten, dass jemand anderes es repariert
10	Sich nach diesem Modul sicher zu fühlen bedeutet...	Stolz zu sein und mehr lernen zu wollen	Nie wieder Fehler zu machen	Die ganze Zeit alleine zu arbeiten	Neue Herausforderungen zu vermeiden



WEITERES MATERIAL

Leicht verständliche Standards & Stil (EN/DE)

- Inclusion Europe – Informationen für alle: Europäische Standards für leicht verständliche Materialien (mehrsprachiges Set + Gebrauchsanweisung). Ideal zum Formulieren von Bildunterschriften und Anleitungen. [Inclusion Europe](#)
- Netzwerk Leichte Sprache — Die Regeln für Leichte Sprache (Neuaufgabe 2022) (PDF). Use for German learner-facing text. [netzwerk-leichte-sprache.de](#)
- IFLA-Fachbericht 120 – Richtlinien für leicht lesbare Materialien. Klare, internationale Standards für barrierefreies Textlayout. [ifla.org](#)
- Bundesfachstelle Barrierefreiheit — overview page linking to Leichte-Sprache rules and federal guidance (DE). [Bundesfachstelle Barrierefreiheit](#)

UDL & inklusives Design

- CAST – UDL-Richtlinien 3.0 (interaktiv). Wenden Sie die Richtlinie „Mehrere Handlungs- und Ausdrucksmöglichkeiten anbieten“ auf Ihre Foto-/Bildunterschriften-/Sprachauswahl an. [udlguidelines.cast.org](#)

Tüfteln & Blockprogrammierung (Blink-/Piep-Grundlagen)

- Arduino – Blink und BlinkWithoutDelay (offizielle Dokumentation). Eine gute Referenz, wenn Sie über die Blöcke hinausgehen. [arduino.cc](#)
- micro:bit – Erste Schritte und MakeCode: Erste Schritte (Blockeditor). Ideal für Druck-Blink-Piep-Muster. [microbit.org](#)
- Tinkercad Circuits – LED-Blink-Tutorials und Lernplattform (Simulator zum Üben ohne Hardware). [Tinkercad](#)

Peer-Feedback („Zwei Sterne und ein Wunsch“)

- NWESD-Übersicht (PDF) – Unterrichtsablauf, adaptiert nach Dylan Williams formativen Assessments; ideal zur Einweisung von Hilfskräften. NWESD 189 • Cambridge University Press & Assessment – Set für formative Bewertung mit der Vorlage „Zwei Sterne und ein Wunsch“. Cambridge English • K20 Center Strategiekarte – prägnante, verständliche Erklärung. [learn.k20center.ou.edu](#)



Symbol- und Piktogrammressourcen für leicht verständliche Artefakte

- ARASAAC – große, kostenlose AAC-Piktogrammbibliothek (siehe CC-BY-NC-SA). beta.arasaac.org
- OpenSymbols – eine Sammlung von über 50.000 Symbolen unter freier Lizenz, die mehrere Symbolsätze umfasst. opensymbols.org

Widgit – weit verbreiteter kommerzieller Symbolsatz; Lizenzbestimmungen für Veröffentlichungen beachten. Widgit Software

Down-Syndrom – Bildung und Erwachsenenleben (Kontext für Kursleiter)

- Down Syndrome Education International (DSE) – evidenzbasierte Ressourcen und Empfehlungen für die Aufklärung über das Down-Syndrom. down-syndrome.org
- Europäische Down-Syndrom-Vereinigung (EDSA) – EU-weites Netzwerk, Webinare und Erfahrungsaustausch. edsa.eu
- Zentrum für Erwachsene mit Down-Syndrom – praktische Aktivitätsideen für Erwachsene (für Bereicherungstage im Rahmen des Moduls). adscresources.advocatehealth.com



LITERATURVERZEICHNIS

- Arduino. (10. Februar 2024). Blinken. <https://www.arduino.cc/en/Tutorial/Blink>
- Autodesk. (o. J.). Schaltkreise lernen. <https://www.tinkercad.com/learn/circuits>
- Cambridge University Press & Assessment. (2022). Formatives Bewertungsset (Ressource Nr. 3: Zwei Sterne und ein Wunsch) [PDF]. Cambridge English
- CAST. (2024). UDL-Leitlinien 3.0. <https://udlguidelines.cast.org>
- Down Syndrome Education International. (o. J.). Online-Ressourcen zur Aufklärung über das Down-Syndrom. <https://www.down-syndrome.org>
- Down Syndrome Education International. (o. J.). Empfehlungen für die Bildung von Lernenden mit Down-Syndrom. ds-int.org
- Europäische Down-Syndrom-Vereinigung. (o. J.). EDSA – Website und Webinare der Vereinigung. edsa.eu
- IFLA. (2010). Richtlinien für leicht verständliche Materialien (IFLA Professional Reports, Nr. 120). International Federation of Library Associations and Institutions. ifla.org
- Inclusion Europe. (6. Oktober 2021). Information für alle: Europäische Standards für leicht lesbare und verständliche Informationen. Inclusion Europe
- Inclusion Europe. (10. Dezember 2023). Anwendung der europäischen Standards (FAQ). inclusion.eu
- Micro:bit Educational Foundation. (o. J.). Erste Schritte (Einführung / Programmieren lernen). <https://microbit.org/get-started/getting-started/introduction/> und <https://microbit.org/get-started/getting-started/get-coding>
- NWESD. (2011). Zwei Sterne und ein Wunsch [PDF]. NWESD
- OpenAAC-Initiative. (o. J.). OpenSymbols. opensymbols.org
- Netzwerk Leichte Sprache e.V. (2022). Die Regeln für Leichte Sprache (Neuaufgabe) [PDF]. netzwerk-leichte-sprache.de
- Zentrum für Erwachsene mit Down-Syndrom. (Dezember 2023). Aktivitäten für zu Hause. adsresources.advocatehealth.com
- Widgit Software. (o. J.). Widgit Symbols – Lizenz- und Urheberrechtshinweise. <https://www.widgit.com/symbol-services/licensing.htm> und <https://www.widgit.com/symbol-services/copyright.htm>
- ARASAAC. (o. J.). AAC-Piktogramme (frei zugänglicher Symbolsatz). beta.arasaac.org
- (Falls Sie den MakeCode-Link direkt verwenden möchten: Microsoft. (o. J.). MakeCode für micro:bit – Erste Schritte.*)

HAFTUNGSAUSSCHLUSS

Gefördert von der Europäischen Union. Die geäußerten Ansichten und Meinungen sind jedoch ausschließlich die der Autoren und spiegeln nicht notwendigerweise die der Europäischen Union oder der Exekutivagentur für Bildung und Kultur (EACEA) wider. Weder die Europäische Union noch die EACEA übernehmen dafür die Verantwortung.

CC BY

Diese Lizenz erlaubt es Nutzern, das Material in jedem Medium und Format zu verbreiten, zu bearbeiten, anzupassen und darauf aufzubauen, solange der Urheber genannt wird. Die Lizenz gestattet die kommerzielle Nutzung. CC BY umfasst folgende Elemente:



BY: Der Urheber muss genannt werden.



Kofinanziert von der
Europäischen Union