

# KiTec Digital

## Kinder entdecken Technik

Handbuch für Lehrkräfte



Ergänzungs-  
modul für KiTec  
mit Elementen  
aus IT2School

Eine Entwicklung in Kooperation von



# Inhaltsverzeichnis

<b>1 Einführung</b> .....	<b>5</b>
<b>2 KiTec Digital im Kontext Schule</b> .....	<b>7</b>
2.1 Ziele .....	7
2.2 Grundprinzipien .....	8
<b>3 Bestandteile von KiTec Digital</b> .....	<b>11</b>
3.1 Werkzeug .....	11
3.2 Mechanische Bauteile .....	12
3.3 Elektrische Bauteile .....	13
3.4 Mikrocontroller .....	15
<b>4 Umsetzung</b> .....	<b>16</b>
4.1 Episoden der KiTec-Geschichte in der Fortsetzung .....	16
4.2 Grundlegendes zum Einsatz der Bauteile .....	17
4.3 Darstellung und Funktion der Bauteile .....	18
4.4 Kartonplatine .....	27
4.5 Mikrocontroller .....	30
4.6 Werkzeuge .....	32



<b>5 Bauaufträge</b> .....	<b>34</b>
5.1 Bauauftrag: Eine Beleuchtung anbringen. ....	34
5.2 Bauauftrag Motor mit einer Richtung .....	43
5.3 Bauauftrag Motor mit zwei Richtungen. ....	52
<b>6 Weitere Ideen für Bauaufträge</b> .....	<b>65</b>
6.1 Ampel .....	65
6.2 Aufzug. ....	66
6.3 Feuerwehrauto .....	67
6.4 Schranke .....	67
6.5 Zugbrücke. ....	68
6.6 Kran .....	68
6.7 Karussell. ....	69
<b>7 Fachinformationen</b> .....	<b>70</b>
7.1 Elektrotechnik. ....	70
7.2 Aufbau des Mikrocontroller. ....	78
7.3 Einrichtung der Programmierumgebung. ....	82
7.4 Programmieren. ....	85
7.5 Fehleranalyse. ....	90
<b>8 Impressum</b> .....	<b>93</b>



# 1. Einführung

**W**as vor 10 Jahren die Nachbarin neugierig über den Zaun hat schauen lassen, weil da ein grünes rundes Etwas wie von Zauberhand bewegt über den Rasen fährt, ist heute fast schon Standardausstattung eines jeden Gartenbesitzers mit Grünfläche: Der Mähroboter. Und die Robotik hat in den privaten Haushalt Einzug gefunden: Saugroboter, Fensterputzroboter, Kühl-

schränke, die melden, was eingekauft werden muss, Geräte, die über Sprachsteuerung in Betrieb genommen werden, Jalousien, die nach einem Zeitprogramm hinauf- und hinabgehen und vieles mehr. Das SMART-Home ist bereits Gegenwart. Und bis die ersten selbstfahrenden Autos für den allgemeinen Straßenverkehr zugelassen werden, brauchen wir wohl keine 20 Jahre mehr zu warten.

Die Kinder von heute sind die Forscherinnen und Ingenieure von morgen. Bereits heute sind sie Nutzerinnen und Nutzer von automatisierter Technik. Da liegt es nahe, ihnen die Chance zu geben, hinter die Kulisse der Bits und Bytes zu schauen. Denn wenn sie die Verbindung zwischen Mechanik, Elektrik und Fernsteuerung über Programme nachvollziehen können, ist ihr Entwicklungsdrang kaum noch zu bremsen.

Bleiben wir beim Beispiel mähen:

Da gibt es ein Werkzeug, das Gras schneidet: die Sense oder die Sichel. In einer nächsten Stufe baut man eine Mechanik ein, um nicht nur eine Schneidklinge zu betätigen, sondern mehrere gleichzeitig. Diese Klingen auf einer Walze angeordnet, die sich dreht, bildet nun den Handrasenmäher. Damit es für den schiebenden Menschen nicht mehr so anstrengend ist, kommt nun die Elektrik hinzu: Sie übernimmt den Antrieb mittels Motor.

Lange Zeit blieb die technische Entwicklung auf dieser Stufe: Antriebe wurden verfeinert, Motorleistungen erhöht, verschiedene Arbeitsvorgänge in Maschinen kombiniert.

Der Mensch blieb der Steuernde: er entscheidet, wann die Maschine in Betrieb kommt, wie lange sie läuft, wobei sie genau eingesetzt wird, wie viel Leistung sie erbringen soll.

Seit geraumer Zeit nun ist die Technik auf dem Level der Digitalisierung angelangt: Computer sind nun maßgeblich in die Steuerung der Vorgänge integriert. Für den Einsatz unseres Rasenmähers heißt das: Er „weiß“ nun von selbst, wann er loslegt, wie lange er mähen soll oder welche Höhe das Gras haben soll. Und damit er nicht vergeblich versucht, einen Apfelbaum umzumähen, sondern sich mit dem Rasen zufriedengibt, wird er mit Sensoren ausgestattet, die ihm den Weg zeigen. Der Mensch bleibt der Steuernde, denn er bestimmt, an welchen Stellen der Computer welche Verarbeitungs- und „Denk“-Leistung übernimmt.

Mit KiTec lernen die Kinder bereits den Umgang mit wesentlichen Werkzeugen, sie setzen sich mit den Grundlagen der Mechanik auseinander: Kräfte und Kraftübertragungen, Getriebe und Antriebe und erste Schritte in der Elektrik mit dem einfachen Stromkreis. KiTec Digital knüpft direkt dort an<sup>1</sup>. Die Kinder vollziehen den Brückenschlag hin zur digitalen Steuerung von einfachen Bauwerken oder Fahrzeugen. Es geht um das optimale Zusammenspiel zwischen dem, was im Computer-Programm beschrieben ist, und dem, was das Gerät letztendlich macht: sich bewegen, etwas transportieren, eine Tätigkeit ausführen. Kurz gesagt: Software und Hardware müssen gut miteinander kommunizieren. Mit KiTec Digital machen Kinder die Erfahrung, dass sie auch in der Welt der Digitalisierung und Automatisierung von Vorgängen die Gestaltenden, die Entscheidenden, die Steuernden bleiben.

---

1 KiTec Digital schlägt die Brücke zwischen KiTec und dem Projekt IT2School. Deswegen ist das Feld der IT hier auch nur anskizziert.

# 2 KiTec Digital im Kontext Schule

**K**iTec Digital ist ein praxisorientiertes Projekt, mit dem das Zusammenspiel von Computer und mechanischen Erzeugnissen selbst erprobt und verstanden werden soll. So lernen die Kinder und Jugendlichen die

Grundlagen der Elektrotechnik – hier speziell der einfache Stromkreis – als verbindendes Element zwischen Hard- und Software kennen und machen damit ein eigenes analoges Produkt mit Hilfe eines Mikrocontroller „lebendig“ und „selbstständig“.

## 2.1 Ziele

Der Perspektivrahmen Sachunterricht beschreibt im Hinblick auf das, was die Kinder und Jugendlichen an Kompetenzen in der Technik erwerben sollen, zwei Ebenen: auf der Prozessebene die Denk-, Arbeits- und Handlungsweisen und auf der fachlich-inhaltlichen Ebene die technikbezogenen Kompetenzen.

Die prozeduralen Kompetenzen sind im KiTec-Handbuch<sup>2</sup> S. 8-10 gut beschrieben. Sie gelten im Grunde für alle Lernprozesse, wenn Kinder sich im Miteinander mit Themen auseinandersetzen.

Die technikbezogenen Kompetenzen konkretisieren sich bei KiTec Digital wie folgt:

### Die Schüler\*innen...

<b>Technik konstruieren und herstellen</b>	<ul style="list-style-type: none"><li>✓ ...bauen elektronische Bauteile in Schaltungen lagerichtig ein</li><li>✓ ...stellen elektrische Verbindungen her, indem sie sie mit Aderendhülsen verpressen</li><li>✓ ...stellen Werkzeuge bedarfsgerecht ein, z.B. beim Justieren der Abisolierzange</li><li>✓ ...isolieren Litzen mit Hilfe einer Abisolierzange ab</li><li>✓ ...schneiden Drähte und Litzen mit Hilfe des Seitenschneiders ab</li><li>✓ ...schließen eine elektrische Schaltung an einen Mikrocontroller an und kombinieren die Bauteile auf Basis der Kartonplatine miteinander</li></ul>
--	--

<p><b>Technik und Arbeit erkunden und analysieren</b></p>	<ul style="list-style-type: none"> <li>✓ ...erkennen elektronische Bauteile: Widerstand, LED, Transistor</li> <li>✓ ...unterscheiden die Bauteile aufgrund ihrer Kennzeichnung und spezifischen Merkmale (z.B. Farbringe der Widerstände)</li> </ul>
<p><b>Technik nutzen und gestalten</b></p>	<ul style="list-style-type: none"> <li>✓ ...erstellen Programme, mit denen einfache Bauteile angesteuert werden</li> </ul>
<p><b>Technik bewerten</b></p>	<ul style="list-style-type: none"> <li>✓ ...testen elektrische Schaltungen</li> <li>✓ ...nehmen eine systematische Fehlersuche vor</li> </ul>
<p><b>Technik kommunizieren</b></p>	<ul style="list-style-type: none"> <li>✓ ...lesen Schaltpläne auf der Kartonplatine</li> <li>✓ ...üben sich darin, mithilfe der Schaltpläne und der Kartonplatinen ihre Arbeitsvorgänge und Vorrichtungen abstrakt darzustellen</li> </ul>

## 2.2 Grundprinzipien

Die oben beschriebenen Kompetenzen dienen der Orientierung. Sie sind nicht als konkret zu erreichende Ziele zu verstehen. Schwerpunkt bei KiTec insgesamt und damit auch bei KiTec Digital bleibt die motivationale Ebene und nicht das Erlernen bestimmter überprüfbarer Kompetenzen.

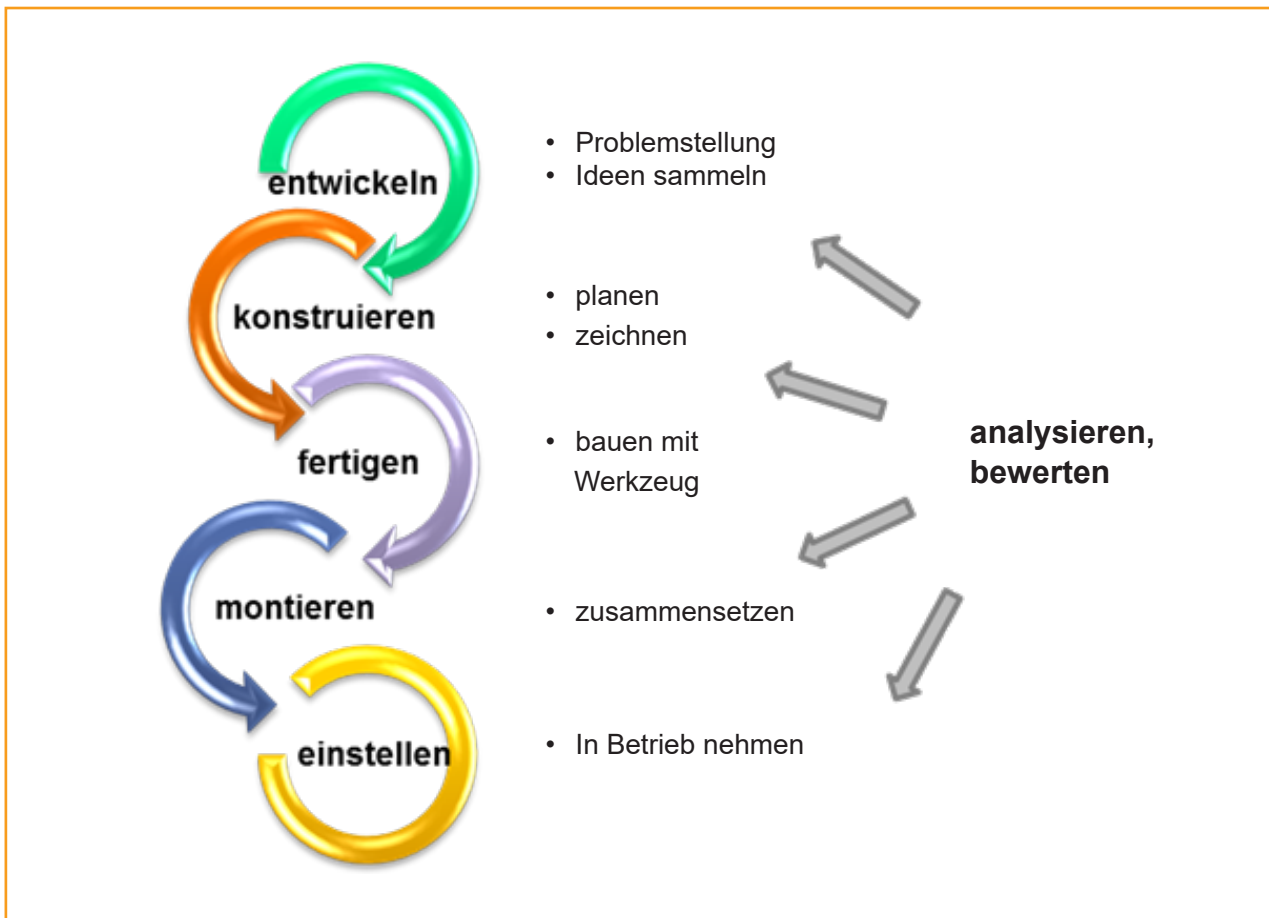
Die zugrunde liegenden lernwissenschaftlichen Erkenntnisse sind im KiTec-Handbuch und im KiTec für die Kita-Handbuch ausführlich beschrieben. Und so orientieren sich auch bei KiTecDigital die Prozesse an der Didaktischen Acht. Das Handeln, die eigenen Erfahrungen sind die Türöffner fürs Nachdenken, Reflektieren und Abstrahieren.

Die Abläufe orientieren sich auch bei KiTec Digital am Prinzip der vollständigen Handlung. Die Kinder erstellen Produkte, die über die Mechanik hinaus nun elektrotechnische und informationstechnische Eigenschaften haben. Gerade diese zusätzlichen Eigenschaften erfordern nun eine stärkere Vorstrukturierung. Damit die Kinder und Jugendlichen auch in diesen Bauprojekten im Trial-and-Error-Modus lernen können, müssen die Arbeitsschritte in einer Weise visualisiert und angeleitet sein, die ihnen mögliche Fehler aufzeigen können.

Wir greifen bei KiTec Digital deshalb auf vorstrukturierte Informationen und Arbeitshilfen zurück. Die herausfordernden Aufgaben können dadurch von den Kindern und Jugendlichen selbst bewältigt werden. Sie stellen eigenständig funktionsfähige Schaltungen bzw. Produkte her. Das Bauen nach Schaltplänen und konkreten kleinschrittigen Handlungsanweisungen ist übrigens auch in der Berufswelt gängige Praxis. Insofern erleben die Kinder und Jugendlichen, dass auch die Welt der Elektro- und Informationstechnik davon lebt, dass Arbeitsschritte vereinfacht und für die Übertragbarkeit abstrahiert werden.



Die Phasen der vollständigen Handlung hier im Einzelnen:



## 1. Informieren und Entwickeln

Die Idee für ein Bauprojekt kann durch Beobachtung der Umgebung entstehen. Eine weitere Möglichkeit besteht, wie bei KiTec, durch das Vorlesen einer kurzen Geschichte. Für KiTec Digital gibt es dafür die KiTec Episoden in der Fortsetzung im Kapitel 4.1. Die Kinder entwickeln Ideen und beschaffen sich selbstständig die notwendigen Informationen. In Kitec Digital bestimmen sie, welche Art von Produkt gebaut wird und welche Funktionen das zu mechanisierende und durch den Computer gesteuerte Bauwerk haben soll. So entsteht ein konkreter Arbeitsauftrag mit einem Ziel.

Wenn die Aufgabenstellung geklärt ist, wird das zur Verfügung stehende Material erkundet. Dies kann z.B. ein Memory zum Kennenlernen der Bauteile sein. Eine kleine Werkzeugkunde kann helfen, mit den Zangen gut umzugehen. Auch das korrekte Abisolieren einer Leitung muss geübt werden. Diese Vorbereitungen bilden die Basis für die nun folgende Realisierung des Bauprojekts.

## 2. Konstruieren, Planen und Entscheiden

Die Schüler\*innen erstellen Skizzen, nutzen Schaltpläne und suchen sich die entsprechenden Materialien heraus. Dabei sind auch Überlegungen anzustellen, wie die elektrische Schaltung anzubringen ist und welche Teilschaltungen erforderlich sind. Als Auswahl stehen drei Grundschaltungen

zur Verfügung, die beliebig kombiniert werden können. Auch das ist in der Elektrotechnik und im Maschinenbau üblich: Auch Profis greifen auf Sammlungen elektrotechnischer Schaltungen zurück, wenn sie auf Probleme stoßen.

### **3. Fertigen und Montieren**

Die Schüler\*innen setzen ihre Planung um. Sie ordnen die Bauteile entsprechend des Schaltplans an und verbinden sie mit Hilfe der speziellen Zangen. Damit entsteht eine Schaltung und eine Steuerung, mit der die Mechanik angetrieben oder Licht erzeugt wird.

### **4. In Betrieb nehmen, Einstellen und Überprüfen**

Die Schaltung wird nun in Betrieb genommen. Die Schüler\*innen überprüfen alle Funktionen. Dabei ist eine schrittweise Vorgehensweise empfehlenswert, um mögliche Fehlerquellen besser zu lokalisieren.<sup>3</sup>

Das Reflektieren, Analysieren und Bewerten spielen in jeder Bauphase eine Rolle. Dabei helfen die Hinweise in den jeweiligen Bauaufträgen. Von besonderer Bedeutung ist dabei, dass die Schaltpläne immer wieder mit der eigens aufgebauten Schaltung verglichen werden.

---

<sup>3</sup> Zur Programmierung und Fehleranalyse siehe Kapitel 7

# 3 Bestandteile von KiTec Digital

**K**iTec Digital besteht aus diesem Handbuch und haptischen Materialien für die Bauaufträge. Das KiTec Digital Handbuch baut auf dem KiTec-Handbuch und dem KiTec für die Kita-Handbuch auf. Die Bauaufträge

in Kapitel 5 können als Arbeitsblätter an die Schüler\*innen gegeben werden. Die Kapitel 4.3 bis 4.7 können die Kinder bei den Bauaufträgen unterstützen und als Hilfefkarten dienen. Alle weiteren Kapitel sind Hintergrundinformationen für Lehrer\*innen.

Die haptischen Materialien für die KiTec Digital-Bauaufträge sind in einem Klassensatz enthalten. Der Klassensatz KiTec Digital besteht aus Werkzeugen, elektrischen Bauteilen und mechanischen Bauteilen. In einem Klassensatz ist genug Material enthalten, sodass die Schüler\*innen in 5er Gruppen die Bauaufträge gemeinsam bearbeiten können.

## 3.1 Werkzeug

Bild	Werkzeug	Stk. / Gruppe	Stk. / Klasseneinheit
	Abisolierzange	1	5
	Seitenschneider, klein	1	5
	Spitzzange, klein	1	5

## 3.2 Mechanische Bauteile

Bild	Material	Stk. / Gruppe	Stk. / Klasseneinheit
	Reduzierhülse, 4 auf 3 mm	6	30
	Reduzierhülse, 4 auf 2 mm	12	60
	Elektromotor RE 140	2	10
	Schweißstab, d = 3 mm ca. 50 cm	1	5
	Zahnrad M1, 10 Zähne	1	5
	Zahnrad M1, 20 Zähne	1	5
	Schnecke M1 für Achse 2 mm	1	5
	Kabelbinder 200 mm lang, 3,6 mm breit	3	15
	Getriebeblock Eigenanfertigung über 3D-Druck	1	5

### 3.3 Elektrische Bauteile

Diese Tabelle dient der ersten Übersicht zu den bei KiTec Digital verwendeten Bauteilen. Wie sie richtig eingesetzt werden und worauf es bei ihnen ankommt, folgt in Kapitel 4.

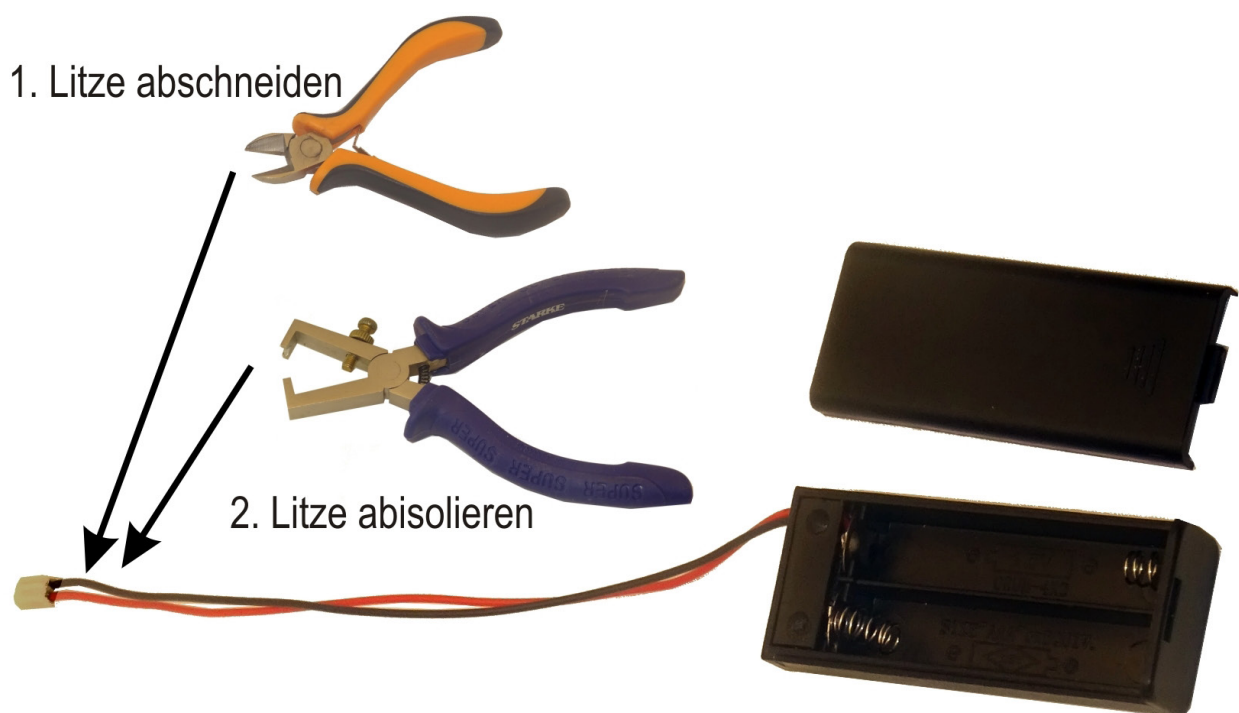
Bild	Material	Stk. / Gruppe	Stk. / Klasseneinheit
	Aderendhülsen, klein	20	100
	Aderendhülsen, groß	20	100
	pnp-Transistor BC 327	2	10
	npn-Transistor BC 337	3	15
	Widerstand 33 Ohm	5	25
	Widerstand 1,8 kOhm	5	25
	LED, rot, 5 mm	2	10
	LED, gelb, 5 mm	2	10

	LED, grün, 5 mm	2	10
	LED, blau, 5 mm	2	10
	Litze, schwarz, 0,14 mm <sup>2</sup>	1 m	10 m
	Litze, rot 0,14 mm <sup>2</sup>	1 m	10 m
	Litze, gelb 0,14 mm <sup>2</sup>	1 m	10 m
	Litze, blau 0,14 mm <sup>2</sup>	1 m	10 m
	Litze, grün 0,14 mm <sup>2</sup>	1 m	10 m

## 3.4 Mikrocontroller

Sets mit Mikrocontroller bestehen aus dem Mikrocontroller selbst (Micro:Bit oder Calliope), einem Batteriekasten und einem USB-Kabel. Diese sind nicht in KiTec Digital vorhanden und müssen extra bestellt werden. Für einen Klassensatz werden 5 Sets Mikrocontroller, entweder Micro:Bit oder Calliope, benötigt. Beide können bei KiTec Digital eingesetzt werden.

Der vorhandene Batteriekasten muss noch für die Arbeit mit Kitec Digital vorbereitet werden:



# 4. Umsetzung

**G**eschichten als Möglichkeit, sich in eine Situation hineinzusetzen und daraus den Grund für Bauprojekte zu entwickeln, hat sich bereits im KiTec-Handbuch bewährt. Daher bieten wir hier weitere Episoden an, die die Kinder, die auf der Insel gestran-

det sind und sich ihr Leben dort einrichten müssen, erleben. Daraus können direkt die Bauaufträge abgeleitet werden. In den Episoden sind jeweils zwei Teilaufträge enthalten: Zunächst stellen die Kinder ein elektrisches Gerät her, und dann wird es mit einer automatischen Steuerung versehen.

## 4.1 Episoden der KiTec-Geschichte in der Fortsetzung

### Episode 11 [Leuchtturm]

Der Leuchtturm ist gebaut und hat sogar eine Beleuchtung. Wenn das Licht aber permanent an ist, fällt es irgendwann nicht mehr auf. Ein Blinklicht ist die Lösung! Dazu bauen die Kinder einen entsprechenden Taster in ihren Stromkreis für die Lampe ein. Die Kinder beschließen, dass immer jemand im Turm sitzt und den Taster zum Blinken betätigt – Tag und Nacht!

Das geht eine ganze Weile gut, weil sich die Kinder beim Blinken abwechseln. Mit der Zeit macht das aber ganz schön müde. Tagaus tagein auf einen Taster zu drücken, kann nicht wirklich die Lösung sein. Das muss doch irgendwie automatisch gehen...

### Episode 12 [Ventilator]

In der Hütte ist es super heiß. Die ganze Zeit mit einem Blatt in der Hand herumwedeln, verschafft zwar Kühlung, aber da wird ja die Hand ganz lahm. Da muss ein Ventilator her – gesagt, getan – er läuft nun dauernd. Na ja, zumindest, bis die Batterie leer ist... Das bringt die Kinder ins Nachdenken: Der Ventilator muss ja eigentlich nur laufen, wenn sie sich in der Hütte aufhalten und mal gemütlich chillen wollen. Ließe sich das irgendwie einstellen? Aber nicht nur mit einem einfachen Schalter ... Kann man da was programmieren, dass die Hütte schon ein bisschen kühler wird, bevor die Kinder von ihren Ausflügen zurückkommen?

### Episode 13 [Fahrzeug]

Das Ziehen der Karren ist schon eine mühselige Arbeit, da wäre Erleichterung angebracht. Am besten, man baut einen Motor an das Rad! Aber das Fahrzeug soll auch vorwärts und rückwärts fahren können. Also brauchen wir etwas, mit dem wir dem Motor sagen können, in welche Richtung er sich drehen soll. Vielleicht geht das ja mit einer Art Schalter?

### Episode 14 [Fahrzeug mit Blinken]

Einmal hatten die Kinder Riesenglück. Bei einem großen Lagerfeuer, das sie zur Feier des Leuchtturms veranstaltet hatten, fing ein Schuppen plötzlich Feuer. Das Feuerwehrauto war schnell zur



Stelle. Allerdings hätte es da beinahe einen Zusammenstoß mit einem anderen Fahrzeug gegeben, weil der Fahrer die Feuerwehr schlicht übersehen hatte! Die Kinder beschließen, dass das nicht noch einmal passieren darf. Das Feuerwehrauto braucht unbedingt ein Blinklicht!

### **Episode 15 [Brückenampel]**

Die Brücke war eine gute Idee, um die Schlucht zu überqueren. Aber aus Materialgründen haben die Kinder sie damals so schmal gebaut, dass keine zwei Fahrzeuge nebeneinander passen. Und jedes Mal gibt es Streit bei der Überquerung, wenn von beiden Seiten ein Fahrzeug kommt und diese sich dann in der Mitte treffen. Keiner will zurück! Kann man das nicht mit Hilfe einer Ampel lösen? Wie muss ich das programmieren, damit auch wirklich nur einer auf die Brücke fährt?

### **Episode 16 [Karussell]**

Technik muss auch für den Spaß da sein! Und weil ja viele Geräte schon automatisch gesteuert werden, haben die Kinder auch Zeit fürs Vergnügen. Sie bauen sich ein Karussell. Doch wer treibt es an? Der Wind? Na, der hat nicht genügend Kraft. Mit einer Kurbel? Ja klar! Aber wer kurbelt? Schließlich wollen doch alle auf dem Karussell sitzen. Dafür muss es doch eine Lösung geben ...

### **Episode 17 [Kran]**

Schon beim Brücken- und Turmbau hatten die Kinder gemerkt, dass es unglaublich mühsam ist, Baumaterial hin- und her zu transportieren; erst recht, wenn in die Höhe gebaut wird! Klar, dass da ein Kran die bessere Lösung ist! Und wie sie auf anderen Baustellen schon gesehen haben, kann man einen Kran auch mit einer Fernbedienung steuern. Das müssten sie doch auch schaffen, oder? Sie fangen an zu tüfteln und machen sich ans Werk.

## **4.2 Grundlegendes zum Einsatz der Bauteile**

Damit sich ein Ventilator drehen kann oder ein Fahrzeug über einen Motor angetrieben wird, braucht es eine elektrische Schaltung. Die verwendeten Bauteile erfüllen nur ihre Funktion, wenn das richtige ausgewählt und korrekt angeschlossen wird. Dies gilt insbesondere für die Bauteile LED und Transistor.

Der Strom wird durch jede Leitung gleich gut geleitet, egal welche Farbe die Isolierung der Leitung hat. Aber bei den vielen Leitungen kann schon mal der Überblick verloren gehen, und deshalb hat jede Farbe eine Bedeutung. Die Zuordnung zu rot und schwarz entspricht den allgemeinen Regeln der Technik. Die anderen Farben haben wir spezifisch für KiTec Digital festgelegt. Die Steuerung der Schaltung erfolgt später über ein kleines Computerprogramm, das selbst programmiert wird. Hier kommt der Mikrocontroller zum Einsatz. Näheres findet sich in Kapitel 7.3 und 7.4.

Übrigens: Auch in der Alltagswelt der Elektrotechnik gibt es feste Vorschriften und Farbzusordnungen. Der Schutzleiter z.B. hat immer die Farbe grün-gelb. Das schützt vor Fehlan schlüssen und damit vor Unfällen.

Falls Sie das Thema Strom noch nicht in den Unterricht eingeführt haben, empfehlen wir Kapitel 7, Elektrotechnik, im KiTec-Handbuch.

Gerade das Lesen der abstrakten Schaltpläne wird die besondere Herausforderung sein. Besprechen Sie mit den Kindern und Jugendlichen selbst, was ihnen hilft, die Verbindung zwischen den Bauteilen, deren spezifischen Einsatz und der grafischen Darstellung auf den Schaltplänen zu erkennen.

Wenn Sie mit Ihren Schüler\*innen spielerisch und ohne digitale Geräte das Thema Programmieren einführen wollen, empfiehlt sich das Modul 8: Calliope, Arbeitsmaterial B8.1 von IT2School, welches Sie sich hier kostenlos herunterladen können:

<https://www.wissensfabrik.de/downloadmaterial-it2school/#modul-b8-calliope>

### 4.3 Darstellung und Funktion der Bauteile

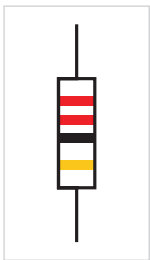
An dieser Stelle beschreiben wir die Bauteile, deren Funktion in den Bauprojekten relevant ist. Es bietet sich an, die jeweilige Grafik als „Merkkarte“ für die Schüler\*innen griffbereit zu halten. Das unterstützt sie beim Lesen der Schaltpläne, in der alle elektrischen Verbindungen zu sehen sind. Auf der Kartonplatine ist der Schaltplan aufgezeichnet. So vereinfachen wir den Schritt, die elektrischen Bauteile auf die Kartonplatine zu montieren.

## Widerstand

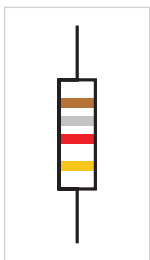
Ein Widerstand wird eingebaut, wenn man möchte, dass nicht zu viel Strom durch die Leitung fließt.

Es gibt Widerstände mit unterschiedlichen Werten. Dieser kann aber auf die Widerstände nicht gedruckt werden, deshalb erhalten alle Widerstände verschiedene Farbringe, an denen der Wert abgelesen wird. Im Material von KiTec Digital gibt es zwei unterschiedliche Widerstände.

Die Merkmale sind deshalb zum allgemeinen Schaltzeichen zusätzlich auf der Kartonplatine eingezeichnet.



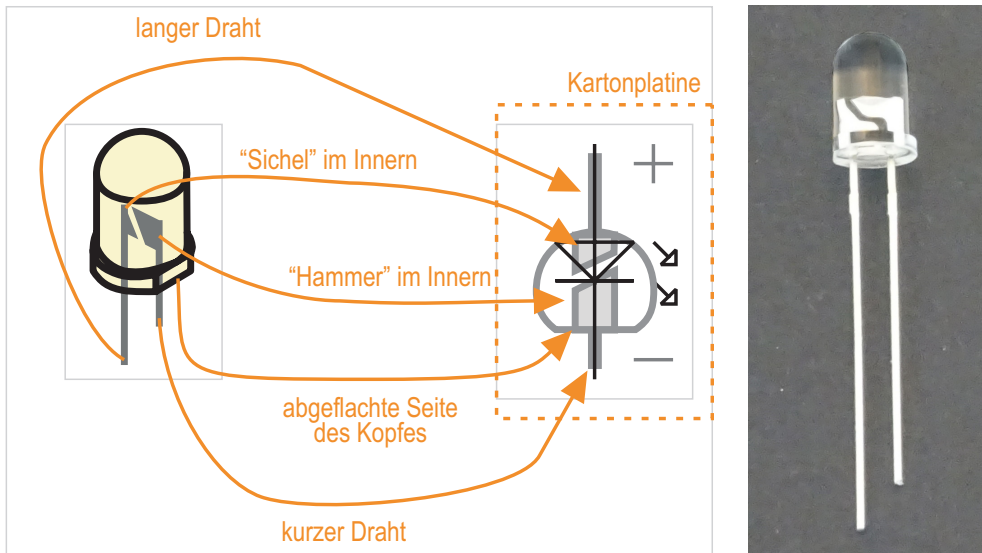
33  $\Omega$  ( $\Omega$  = Ohm)



1,8 k $\Omega$  (k $\Omega$  = kiloOhm)

## LED

Die LED (Leuchtdiode, engl. **light-emitting-diode**), als Leuchtmittel im Elektronikbereich schon lange etabliert, ist ein Bauelement, welches auf der Kartonplatine lagerichtig eingebaut werden muss, damit es funktioniert. Lagerichtig heißt: es ist entscheidend, an welche Stelle das „lange Beinchen“ und das „kurze Beinchen“ angebracht oder eingesteckt werden. Deshalb sind die dafür notwendigen Merkmale zusätzlich auf der Kartonplatine eingezeichnet. Die langen und kurzen „Beinchen“ heißen Anschlussdraht.



Im Kopf der LED, dem Leuchtkörper, befinden sich Silberplättchen, die mit den Anschlussdrähten („Beinchen“) verbunden sind: Der lange Anschlussdraht hat einen kleinen Silberabschnitt, der kurze Anschlussdraht hat einen breiteren Abschnitt. Der lange Anschlussdraht, also die Seite mit dem kürzeren Silberabschnitt wird mit dem Pluspol verbunden. Da man die Anschlussdrähte beim Verbinden später in ihrer jeweiligen Länge kaum noch unterscheiden kann, ist der LED-Körper an einer Seite abgeflacht. Das ist die Seite mit dem breiteren Silberabschnitt, bzw. dem kurzen Anschlussdraht. Hier wird der Minuspol angeschlossen.

Übrigens: die Pole werden auch als **Anode** = Pluspol, und **Kathode** = Minuspol bezeichnet

## Transistor

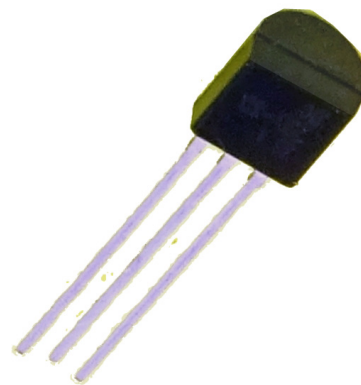
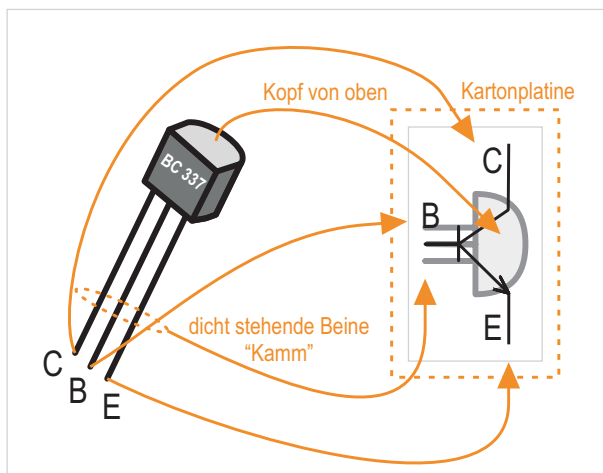
Der Transistor ist ein elektronisch betätigter Taster<sup>4</sup>. Zur Unterscheidung von Transistortypen wird die Bezeichnung auf die abgeflachte Seite des Gehäuses gedruckt. Das ist aber nur unter einer Lupe lesbar. Bei KiTec Digital arbeiten wir mit zwei Transistoren, die zur besseren Unterscheidung unterschiedlich geformte „Beinchen“ haben. Auch hier sprechen wir von Anschlussdraht.

- npn-Transistor mit der Bezeichnung BC 337: Er hat dicht stehende „Beine“ und wird hier in KiTec Digital auch als „Kamm“-Transistor bezeichnet.
- pnp-Transistor mit der Bezeichnung BC 327: Er hat auseinanderstehende „Beine“ und wird hier in KiTec Digital auch als „Gabel“-Transistor bezeichnet.

Die einzelnen Anschlussdrähte haben unterschiedliche Funktionen. Deswegen werden sie extra bezeichnet mit E, B und C<sup>5</sup>. Für den Einbau ist es wichtig, welcher Anschlussdraht an welcher Stelle verbunden wird.

Damit man die Anschlussdrähte richtig zuordnen kann, ist der Transistorkopf an einer Stelle abgeflacht. Der Transistor wird von oben betrachtet, damit man die Anschlussdrähte E, B und C eindeutig zuordnen kann.

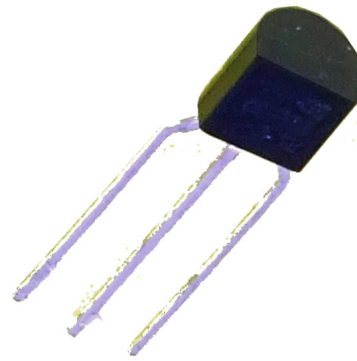
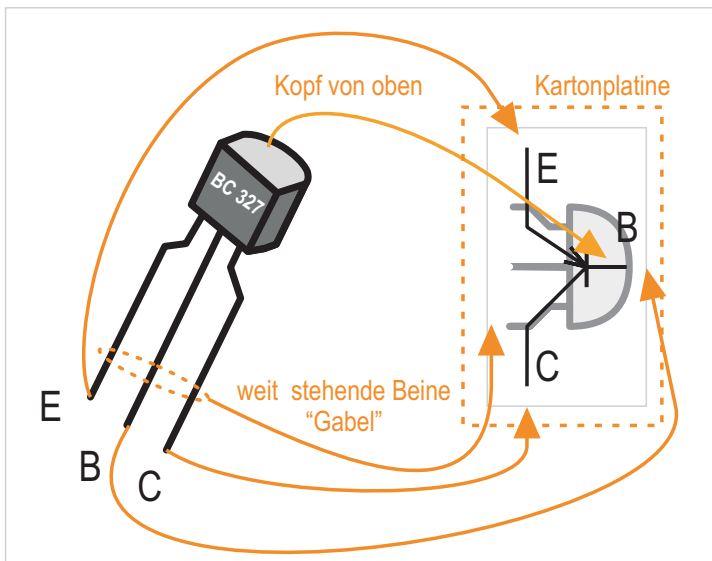
Diese Merkmale sind auch auf der Kartonplatine eingezeichnet.



*npn-Transistor BC 337 („Kamm“-Transistor)*

4 Wichtig zum Verständnis: Wenn ein Transistor geschlossen wird, dann schließt er einen Stromkreis und es fließt Strom. Wenn ein Transistor offen ist, dann ist der Stromkreis offen und es fließt kein Strom.

5 Die genaue Erläuterung siehe Kapitel 7.1



pnp-Transistor BC 327 („Gabel“-Transistor)

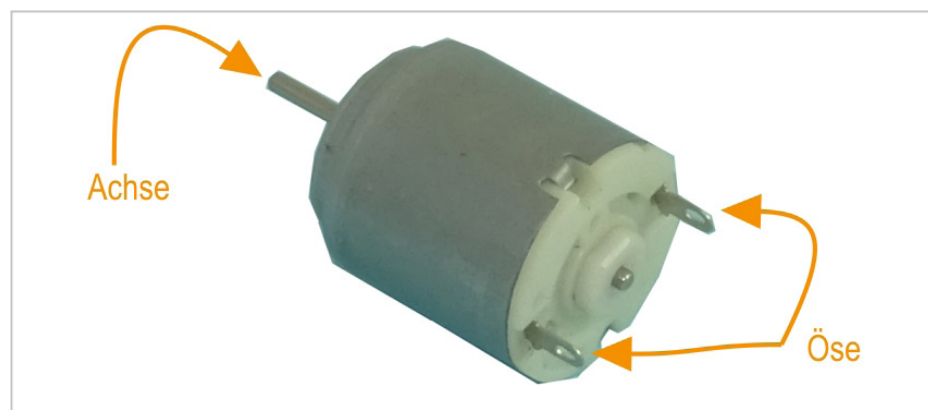
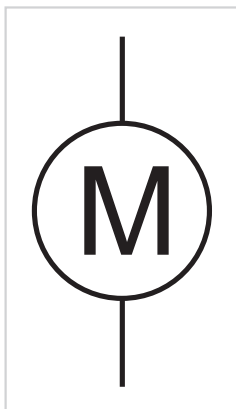
Der npn-Transistor BC 337 zeigt folgendes Verhalten: Wenn die Basis (B) **einen** Strom erhält bzw. an „+“ liegt, wird die Strecke C-E geschlossen und Strom fließt von C nach E.

Ein Klingeltaster funktioniert vergleichbar: Sobald er betätigt wird, wird der Stromkreis geschlossen und die Klingel surrt.

Der pnp-Transistor BC 327 zeigt das entgegengesetzte Verhalten: Wenn die Basis (B) **keinen** Strom erhält bzw. an „-“ liegt, ist die Strecke E-C geschlossen und Strom fließt von E nach C. Die Funktionsweise dieses Transistors ist vergleichbar mit einer elektrischen Sicherung. Solange sie nicht betätigt wird, ist der Stromkreis geschlossen und es fließt ein Strom. Wenn die Sicherung betätigt wird, dann wird der Stromkreis unterbrochen und es fließt kein Strom mehr.

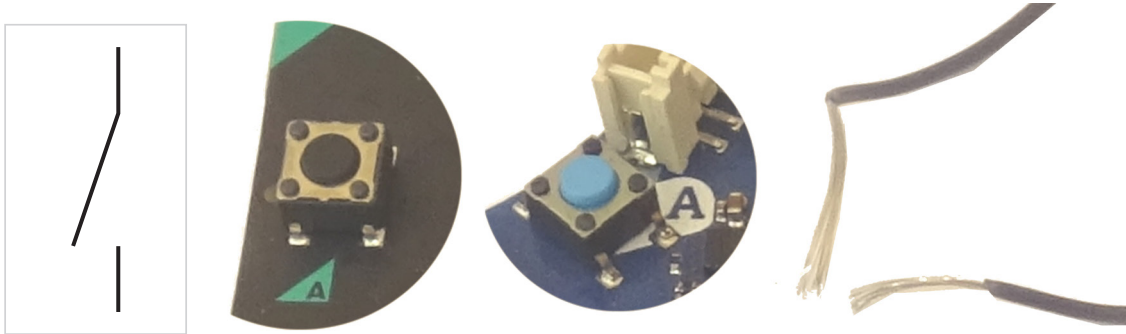
## Motor

Der Motor hat eine Achse, die sich rechts oder links herum drehen kann. Die Drehrichtung hängt davon ab, an welchen Ösen die Litze vom Pluspol oder Minuspol angeschlossen wird. Die Drehrichtung wird durch Ausprobieren festgestellt.



## Taster

Ein Taster schließt einen Stromkreis nur so lange, wie er betätigt wird. Im Alltag findet man ihn als Klingeltaster. Ein Schalter schließt einen Stromkreis dauerhaft wie z.B. ein Lichtschalter. Erst bei einer erneuten Betätigung wird der Stromkreis wieder geöffnet. Bei KiTec Digital gibt es Taster auf dem Mikrocontroller und auch abisolierte Enden von Litzen, die durch Zusammenhalten wie ein Taster funktionieren.



## Batterie

Eine oder mehrere Batterien versorgen einen Stromkreis mit elektrischer Energie und wird auch als Gleichspannungsquelle bezeichnet. Jede Gleichspannungsquelle hat einen Pluspol und einen Minuspol. Bei der Batterie ist der Pluspol an einer Erhebung zu erkennen, während die flache Seite den Minuspol darstellt.



Für den Pluspol gibt es die Zeichen „+“ oder „3V“ bei den Mikrocontrollern. Für den Minuspol die Zeichen „-“, Masse oder GND.

## Litze

Litzen sind dünne flexible Kabel, die den Strom transportieren und werden im Schaltplan als Striche dargestellt. Sie haben eine Isolierhülle in verschiedenen Farben. Die Farben helfen bei der Verkabelung, die richtigen Anschlüsse herzustellen.

Im Allgemeinen gilt:

- die rote Litze wird angeschlossen am Pluspol, „+“ oder „3V“
- die schwarze Litze wird angeschlossen am Minuspol, „-“ oder „GND“

Die Anschlüsse am Mikrocontroller heißen **Pin**. In KiTec Digital haben wir folgende Zuordnung:


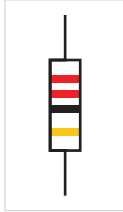
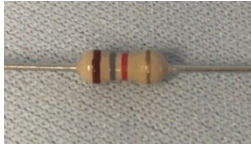
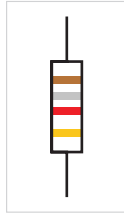

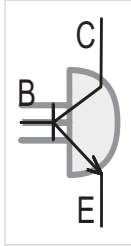

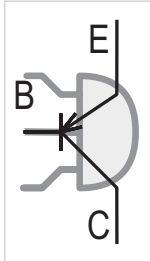

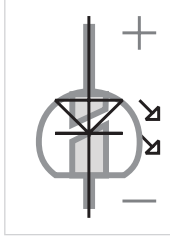

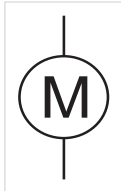
- die gelbe Litze wird an Pin 0 angeschlossen
- die blaue Litze wird an Pin 1 angeschlossen
- die grüne Litze wird an Pin 2 angeschlossen

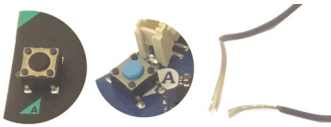


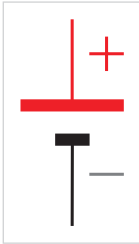
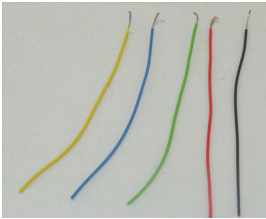



## Kurzübersicht mit Darstellung im Schaltplan

Beim Verbinden der einzelnen Bauteile kommt es wie gesagt darauf an, dass sie auf eine bestimmte Weise, also „richtig“ eingebaut werden. Dabei hilft ein Schaltplan. Man könnte nun auf den Schaltplänen jeweils die Bezeichnung der Bauteile schreiben oder ein Foto des Bauteils abbilden. Dazu reicht der Platz nicht. Es bedarf einer abstrakten Darstellung mit festgelegten Symbolen, die für alle Schaltpläne in diesem Handbuch gleich sind.



Bauteil	Realität	Symbol im Schaltplan
Widerstand orange-orange-schwarz und gold 33 $\Omega$		
Widerstand braun-grau-rot und gold 1,8 k $\Omega$		
NPN-Transistor (BC 337) „Kamm“-Transistor		
PNP-Transistor (BC 327) „Gabel“-Transistor		
LED		
Motor		

<p>Taster</p>		
<p>Batterie</p>		
<p>Litze Schwarz = „-“ / „GND“ / Masse Rot = „+“ / „3V“ grün = Pin 2 blau = Pin 1 gelb = Pin 0</p>		

Dieser Schritt hin zur Abstraktion ist ein mühsamer Lernweg, der Übung erfordert. Es ist daher durchaus legitim, wenn die Schüler\*innen beim Lesen und Verstehen der Schaltpläne die Übersetzungshilfe in Form dieser Tabelle griffbereit haben. Die Symbole dieser Bauteile finden sich auch auf den Kartonplatinen (im folgenden Kapitel).

## 4.4 Kartonplatine

Elektronische Bauteile werden immer auf einem Trägerelement montiert. Dieses Trägerelement heißt Leiterplatte oder Platine. In der Technik werden Platten aus Epoxid-Glasfaser-Laminat mit einer Dicke von 0,5 mm bis 2 mm verwendet, die ein- oder beidseitig mit einer hauchdünnen Kupferschicht versehen sind. Diese Kupferschicht kann bearbeitet werden und die Bauteile können dann auf die Platte gelötet werden.

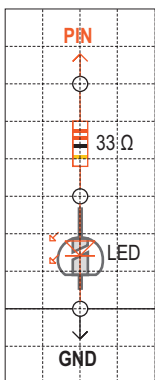
**In KiTecDigital bestehen die Platinen aus Karton.** Die Bauteile werden auf der Oberseite angebracht, an den aufgezeichneten Kreisen durch den Karton gestochen und auf der Rückseite miteinander verbunden. Damit diese Verbindung hält, kommen Aderendhülsen zum Einsatz. Die Anschlussdrähte und Litzen werden eingesteckt. Durch Verpressen (Zusammendrücken mittels Spitzzange) sind sie dann elektrisch leitend verbunden. Platinen haben auf der oberen Seite jeweils Bezeichnungen, damit die Montage problemlos funktioniert. Dies ist auch so bei KiTec Digital: auf der Kartonplatine ist der Schaltplan aufgezeichnet.

Für die KiTec Digital-Bauprojekte werden drei Grundschaltungen benötigt. Unten zeigen wir die Schaltpläne der drei Grundschaltungen. Nachdem viel Erfahrung mit den Schaltplänen gesammelt wurde, können sie auch miteinander kombiniert werden. Die genaue Anleitung für den Bauauftrag erfolgt in Kapitel 5. Wie der Anschluss an den Mikrocontroller erfolgt (Micro:Bit oder Calliope) wird ebenso in Kapitel 5 erläutert.

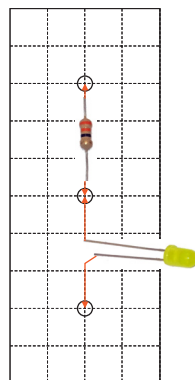
### 1. Beleuchtung:

#### Schaltung mit einem Verbraucher mit kleiner Leistung

Da der Mikrocontroller nur geringe Leistungen direkt zur Verfügung stellt, können damit nur Verbraucher (z.B. LED) mit geringem Strom angeschlossen werden. Exemplarisch wird dafür eine Beleuchtung verwendet, die mit einer LED realisiert wird.



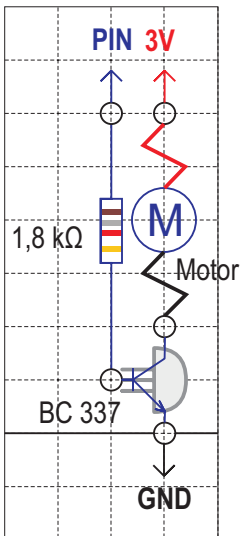
Platine von oben



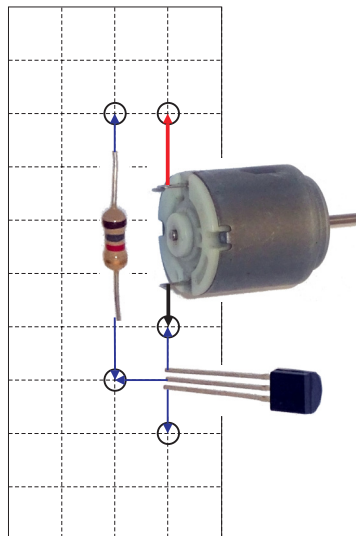
Platine von oben mit Bauteilen

## 2. Einsatz eines Motors mit einer Drehrichtung: Schaltung mit einem Verbraucher mit großer Leistung

Verbraucher (z.B. Motor) mit größeren Leistungen benötigen einen größeren Strom und werden über eine externe Spannungsquelle versorgt. Der Mikrocontroller betätigt dadurch nur einen Taster, der mit Hilfe eines Transistors realisiert wird. Anstelle des Motors können auch andere Verbraucher mit großer Leistung eingesetzt werden.



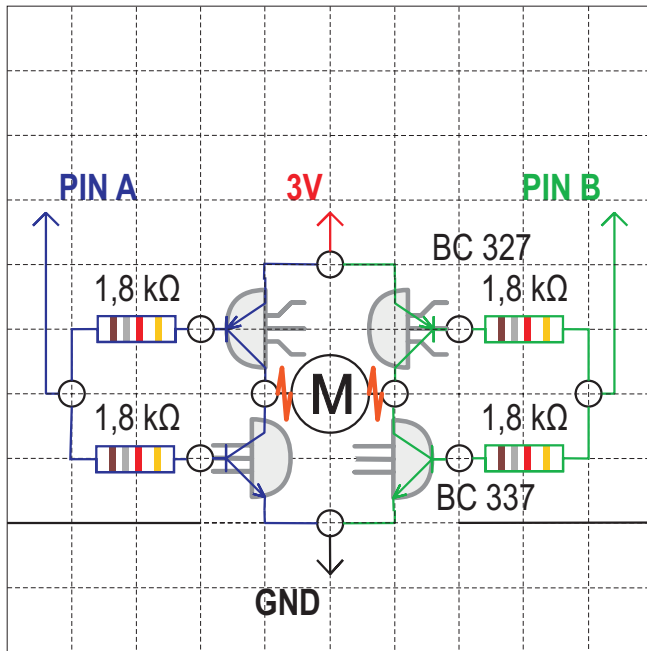
Platine von oben



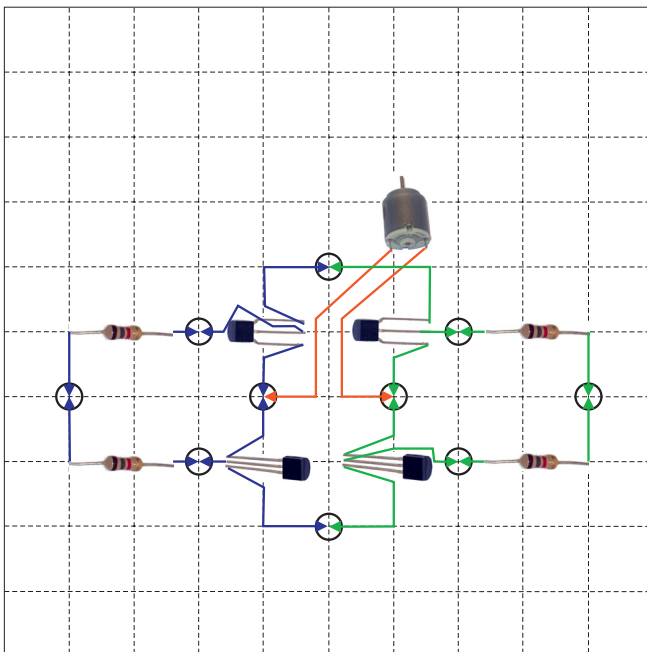
Platine von oben mit Bauteilen

## 3. Einsatz eines Motors in zwei Richtungen: Schaltung mit einem Verbraucher, bei dem der Strom in verschiedenen Richtungen gesteuert werden muss

Die Drehrichtung eines Motors hängt von der Stromrichtung ab. Der Mikrocontroller kann nun durch unterschiedliche Ansteuerung von PIN A und PIN B ermöglichen, dass der Strom in unterschiedlichen Richtungen durch den Motor fließt. Anstelle des Motors können auch andere Verbraucher mit großer und kleiner Leistung eingesetzt werden.



Platine von oben



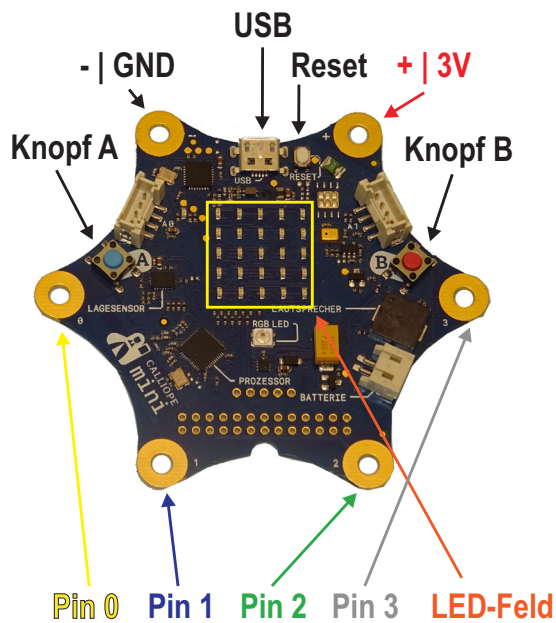
Platine von oben mit Bauteilen

Damit die Schüler\*innen sich mit dieser Darstellungsweise vertraut machen, empfehlen wir, dass sie zunächst die einzelnen Bauteile entsprechend des Schaltplans auf der Kartonplatine auf dem Tisch platzieren und dabei die einzelnen Verbindungen überprüfen: Ist alles verstanden? Was gehört wohin? Was ist mit welchem Teil verbunden?

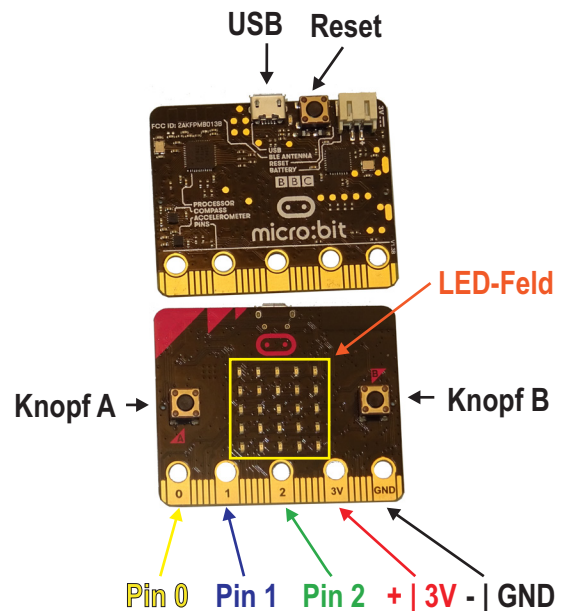
## 4.5 Mikrocontroller

### Aufbau und Anschlüsse

Bei KiTec Digital kommen entweder Calliope oder Micro:Bit zum Einsatz:



Calliope



Micro:Bit

Aufbau und Funktion sind bei den beiden Mikrocontrollern fast identisch. Die Bezeichnungen an den Anschlüssen variieren leicht und der Calliope hat im Gegensatz zum Micro:Bit berührungsempfindliche Pins. Berührungsempfindlich heißt: bereits das Berühren des Pins löst ein Signal im Calliope aus.

#### USB Anschluss des USB-Kabels von PC zum Mikrocontroller

**Wichtig für Calliope: entweder USB-Anschluss oder Batteriekasten anschließen, nie beide Energiequellen gleichzeitig. Bei Micro:Bit ist das nicht relevant.**

Reset Taster zum Rücksetzen des Programms

+ | 3V Anschluss des Pluspols des Batteriekasten mit 2 Microbatterien (AAA)  
*rot als Leitungsfarbe zum Anschließen des Batteriekastens*

- | GND Anschluss des Minuspols (Masse) des Batteriekastens mit 2 Microbatterien (AAA)  
*schwarz als Leitungsfarbe zum Anschließen des Batteriekastens*

- PIN 0 Anschluss  
*gelb als Leitungsfarbe in Kitec Digital festgelegt*
- PIN 1 Anschluss  
*blau als Leitungsfarbe in Kitec Digital festgelegt*
- PIN 2 Anschluss  
*grün als Leitungsfarbe in Kitec Digital festgelegt*
- PIN 3 weitere Anschlussmöglichkeit (nur beim Calliope vorhanden)  
*bei Bauaufträgen in KiTec Digital nicht genutzt*
- Knopf A Taster zur Eingabe
- Knopf B Taster zur Eingabe
- LED-Feld Ausgabe von Zahlen, Zeichen und Symbolen



*Hinweis: Der Mikrocontroller wird erst programmiert, wenn die komplette Schaltung hergestellt ist und funktioniert. Mehr dazu in Kapitel 7.*

## 4.6 Werkzeuge

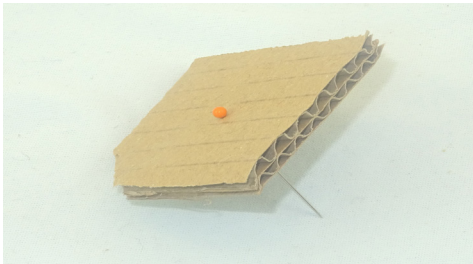
Zum Verarbeiten der elektrischen Bauteile werden folgende Werkzeuge benötigt:

 A pair of wire stripping pliers with blue handles and a gold-colored metal head. The head has a curved, notched design for stripping insulation from wires.	<p><b>Abisolierzange</b> dient zum Abisolieren von Litzen</p>
 A pair of side cutters with orange and black handles and a gold-colored metal head. The head has a curved, pointed design for cutting wires and strips.	<p><b>Seitenschneider</b> dient zum Abschneiden der Drähte und Litzen</p>
 A pair of needle-nose pliers with orange and black handles and a gold-colored metal head. The head has long, pointed jaws for gripping and crimping.	<p><b>Spitzzange</b> dient zum Verpressen der Aderendhülsen</p>

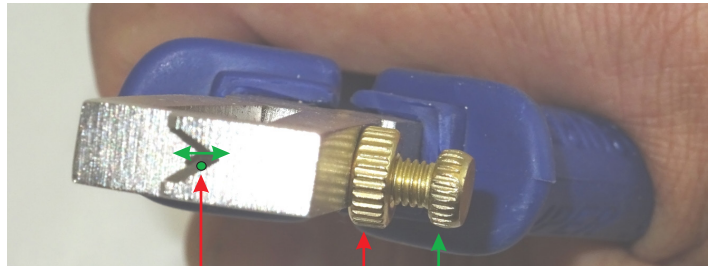


## Litzen abisolieren

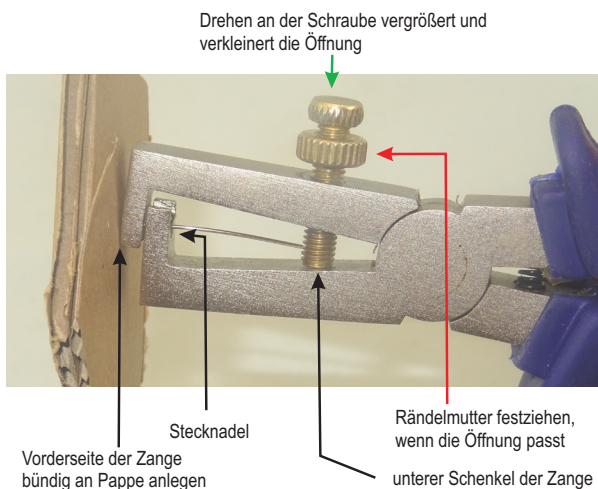
Für das Abisolieren der Litzen wird die Abisolierzange verwendet. Zum erstmaligen Gebrauch muss sie eingestellt werden. Einstellen heißt: mit der Rändelschraube (siehe Abbildung) wird der Abstand zwischen den beiden Zangenschenkeln eingestellt, so dass zwar die Isolierung abgeschnitten wird, der Draht oder die Litze aber unbeschädigt bleibt. Eine Stecknadel hat den passenden Durchmesser. Daher eignet sie sich gut dazu:



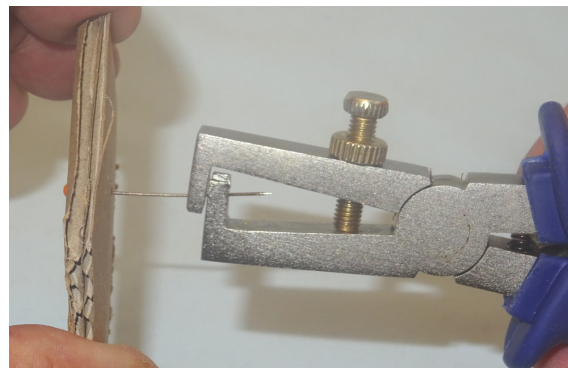
Stecknadel durch ein Stück Wellpappe stechen.



Öffnung für das Abisolieren  
Rändelmutter festziehen, wenn die Öffnung passt  
Drehen an der Schraube vergrößert und verkleinert die Öffnung



Drehen an der Schraube vergrößert und verkleinert die Öffnung  
Stecknadel  
Vorderseite der Zange bündig an Pappe anlegen  
Rändelmutter festziehen, wenn die Öffnung passt  
unterer Schenkel der Zange



Stecknadel in die Öffnung einlegen und Zange mäßig zusammendrücken. Schraube bis an den unteren Schenkel der Zange drehen.

Rändelmutter festziehen. Danach muss sich die Zange noch leicht von der Stecknadel abziehen lassen. Anschließend sollten noch ein paar Probestücke der Litzen abisoliert werden, um die Einstellung zu überprüfen.

Idee: Damit die Kinder die Funktion der Abisolierzange auf einer körperlichen Ebene nachvollziehen können, hilft es ihnen ggf. wenn sie das Abisolieren auch mit einer Kombizange oder Seitenschneider probieren und dann vergleichen. Das, was die voreingestellte Abisolierzange leistet, nämlich den Abstand einzuhalten, müssen die Kinder bei der Kombizange oder Seitenschneider mit ihrem Fingerspitzengefühl und dem behutsamen Krafteinsatz steuern.

# 5. Bauaufträge

**D**ie Bauaufträge beziehen sich auf die elektrische Schaltung und Steuerung. Das Bauwerk selbst, das über diese Schaltung betrieben wird, müssen die Kinder bereits gefertigt haben. Bei den KiTec Digital-Baufträgen kann es hilfreich sein, auf

ausgewählte KiTec-Materialien zurückzugreifen. Der Vorstecher aus dem KiTec-Material eignet sich beispielsweise, um die Kartonplatte für den Bau der Schaltung vorzubereiten. Beim Bauauftrag 5.3 Motor mit zwei Richtungen kann ein Hammer und die Bügelsäge hilfreich sein.

Der Ablauf innerhalb der Bauaufträge folgt nun der Logik der vollständigen Handlung in den entsprechenden Phasen (siehe Kapitel 2.2).

Der Bauauftrag zur Beleuchtung wird sehr detailliert beschrieben, weil die Schüler\*innen hier den grundlegenden Umgang mit den einzelnen Bauteilen üben. Sie bekommen einen Verkabelungsplan sowie eine Kartonplatte und bauen die Bauteile entsprechend dieser Vorlagen zusammen. Zudem gehen die Kinder hier so systematisch vor, wie sie das in den anderen Bauaufträgen auch tun werden. Daher ist es angebracht, dass die Schüler\*innen zunächst mit diesem Bauauftrag beginnen, bevor sie sich an die etwas komplexeren Verbindungen in den anderen Bauaufträgen machen.

## 5.1 Bauauftrag: Eine Beleuchtung anbringen

### Informieren

Leuchten können zwei Funktionen erfüllen: Sie beleuchten einen Gegenstand oder sie vermitteln eine Information dadurch, dass sie blinken. Eine einfache Vermittlung von Informationen ist z.B. durch einen Leuchtturm oder eine Baustellenbake möglich. Beide signalisieren durch regelmäßiges Blinken: Achtung Gefahr! Das ist die Information.

Der Auftrag lautet daher: Versee die Baustellenbake oder ein anderes Bauwerk deiner Wahl mit einem Blinklicht. Dafür brauchst du eine LED.

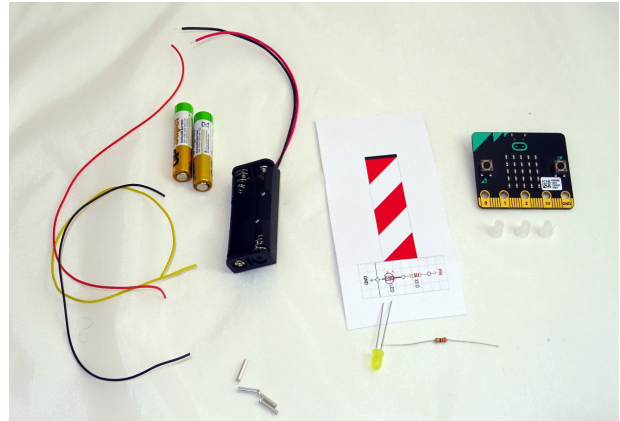
### Planen und Entscheiden

Für den Bau des Blinklichts brauchst du verschiedene Materialien. Suche dir diese zusammen und lege sie geordnet auf deinen Tisch.

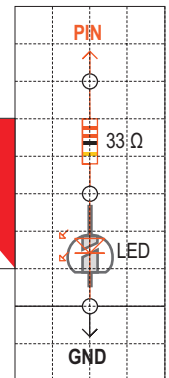
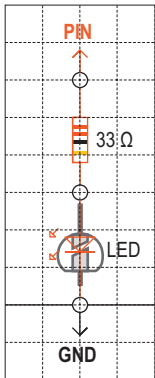
Hierfür verwendest du den Schaltplan „Beleuchtung“ (siehe Kapitel 4.4), für die du einen Widerstand von  $33 \Omega$  (Farbringe orange-orangenschwarz und gold) sowie eine LED in der Farbe gelb, grün, rot oder blau benötigst.

## Material:

- 1 Widerstand  $33\ \Omega$ , Farbringe orange-orange-schwarz und gold
- 1 LED (Farbe egal)
- 1 Kartonplatine (ausgedruckt als Vorlage)
- 1 Batteriekasten mit 2 Batterien
- je 1 Litze von ca. 15 cm Länge in den Farben rot, schwarz und gelb
- 3 Aderendhülsen
- 3 Reduzierhülsen (4 auf 2 mm)

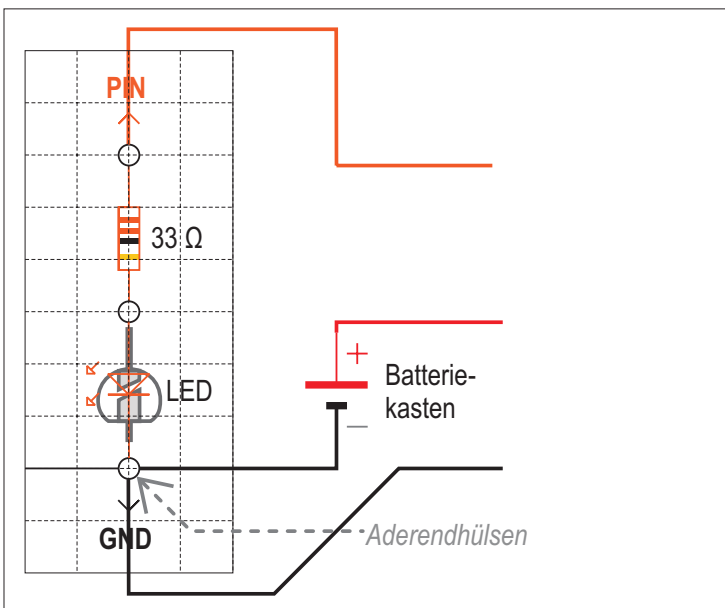


## Kartonplatine



## Verkabelungsplan für den Aufbau

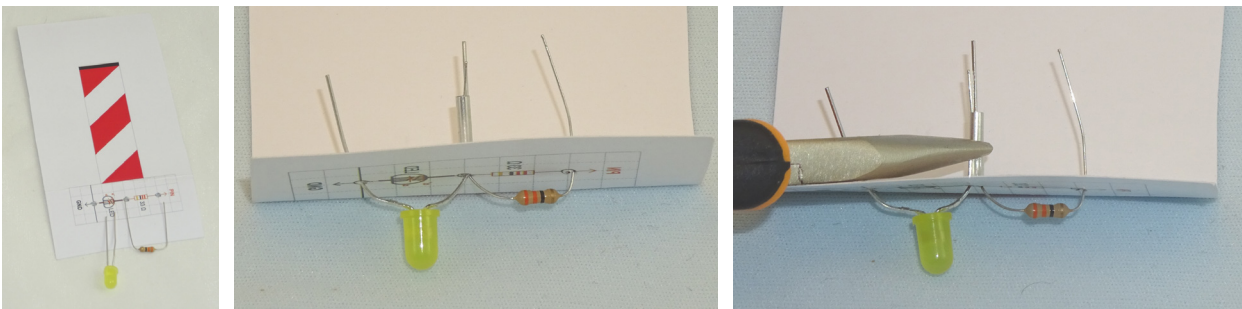
Damit die LED an der Baustellenbake leuchtet, musst du die verschiedenen Bauteile miteinander verbinden. Der Verkabelungsplan ist deine Anleitung, die du dir immer wieder ansehen solltest.



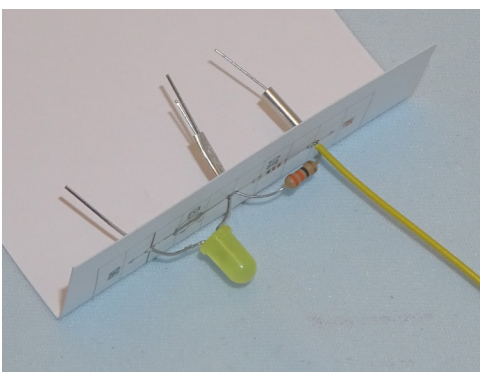
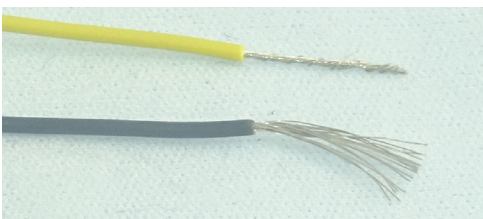
## Fertigen und Montieren

Um die LED zum Leuchten zu bringen, musst du die verschiedenen Bauteile miteinander verbinden. So stellst du die elektrische Schaltung her:

- Steche die Kartonplatine an den Kreisen vor. Dafür kannst du einen Vorstecher oder eine Prickelnadel benutzen.
- Biege die Bauteile so, dass du sie gut in die Kartonplatine hineinstecken kannst. Achte darauf, dass du die LED lagerichtig einbaust. Dabei helfen dir diese Tipps: Der lange Anschlussdraht muss dort hineingesteckt werden, wo auf der Kartonplatine die LED eine runde Seite und einen langen dicken Strich hat. Der kurze Anschlussdraht kommt an die Stelle, wo die gerade Kante und der kurze dicke Strich ist.
- Stecke eine Aderendhülse auf die beiden mittleren Drähte und verpresse sie mit der Spitzzange.

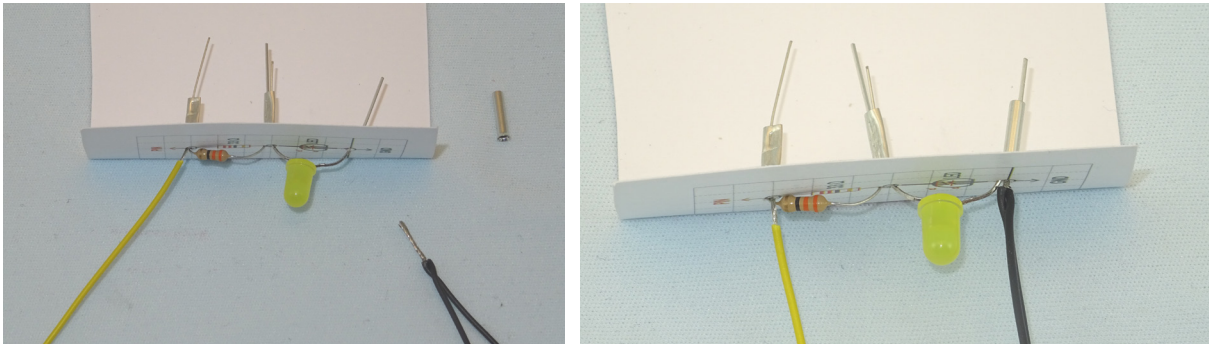
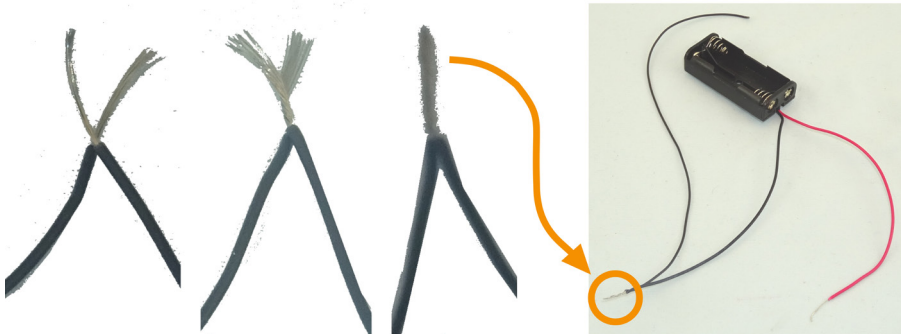


- Nimm dir die rote, schwarze und gelbe Litze und isoliere sie ab.
- Nach dem Abisolieren der Litzen, sieht die Litze so aus wie die schwarze auf dem Foto unten. Als nächstes solltest du die Enden so verzwirbeln, wie unten auf dem Foto bei der gelben Leitung. So passen die Leitungen besser in die Aderendhülsen.



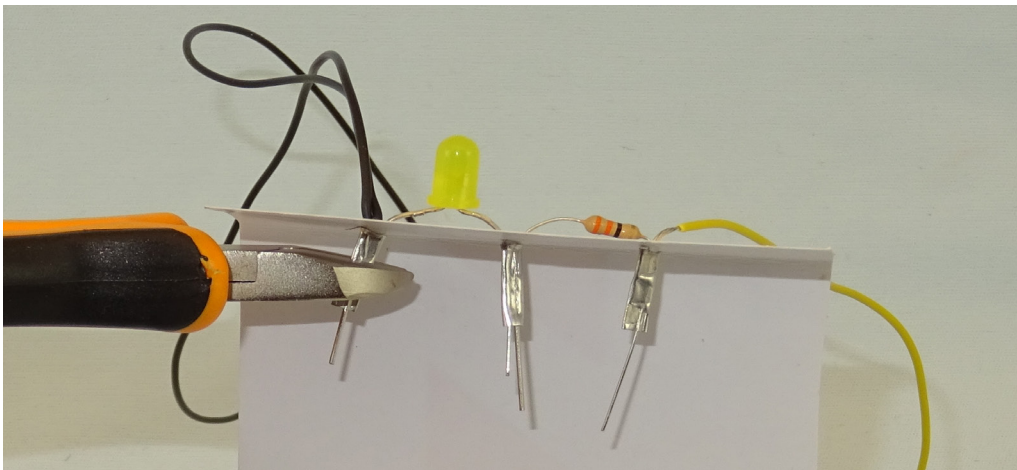
Die Litzen werden nun gemäß Verkabelungsplan mit den Anschlussdrähten der Bauteile verbunden. Achte darauf, dass in der Aderendhülse nur die blanke Litze (ohne Isolierung) ist.

Mehrere Litzen kannst du ebenfalls miteinander verzwirbeln, wenn sie in eine Aderendhülse gesteckt werden sollen.



Wenn du alle Bauteile in die Kartonplatte gesteckt hast und die Anschlussdrähte mit einer Aderendhülse verpresst sind, kannst du die Schaltung fertig stellen. Im Moment schauen aus den Aderendhülsen noch lange Drähte hervor. Sie können sich berühren und den elektrischen Stromkreis stören. Deshalb ist es wichtig, sie mit dem Seitenschneider abzuwickeln und zwar direkt am Ende der Aderendhülse.

**Sicherheitshinweis: Beim Abwickeln mit dem Seitenschneider können die Drähte von der Zange wegfliegen. Halte den Seitenschneider deshalb immer so, dass der Drahtrest von dir weg in Richtung Fußboden fliegt. So wie auf dem Bild unten gezeigt. Dann kann nichts „ins Auge“ gehen.**

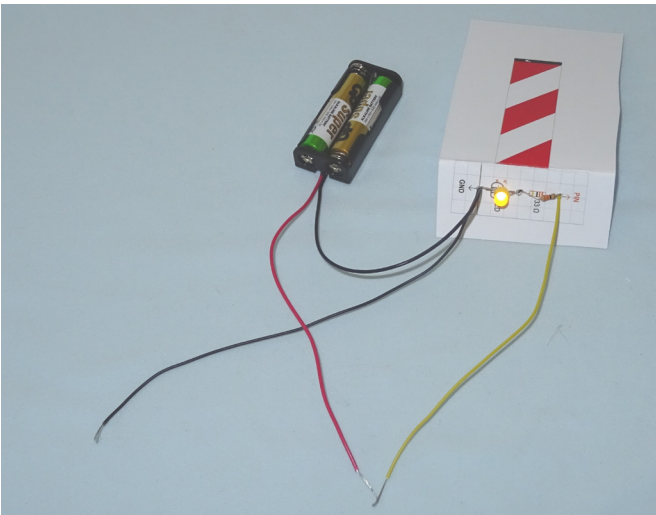


## In Betrieb nehmen, Einstellen und Überprüfen

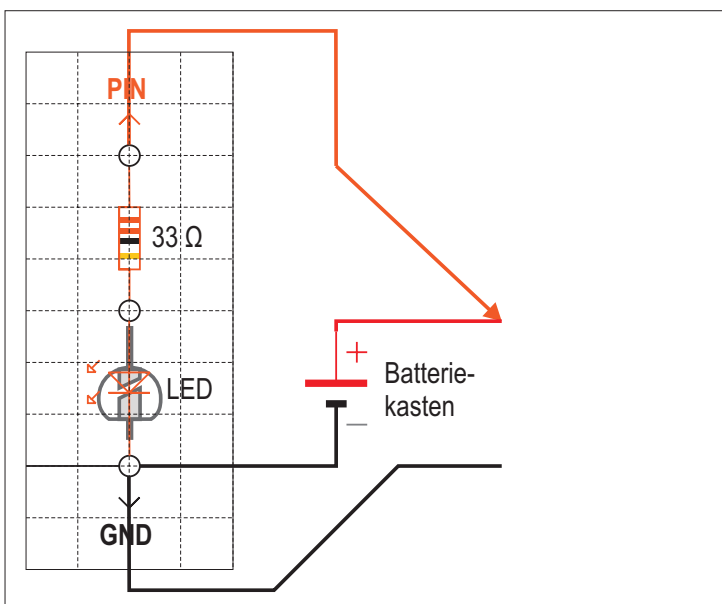
Nachdem das Produkt nun fertig ist, muss es in Betrieb genommen werden:

### → Teste dafür die elektrische Schaltung

Funktionsprüfung: Halte die gelbe Leitung an die rote Leitung (+-Pol des Batteriekasten). Die Schaltung funktioniert, wenn die LED leuchtet



### Verkabelungsplan für den Test



Wenn die LED nicht leuchtet, dann überprüfe folgendes:

- Hast du die Batterien im Batteriekasten richtig eingelegt? (+ und – Pol an der richtigen Stelle?)
- Prüfe, ob die LED richtig eingebaut ist. Schau nochmal auf den Schaltplan: Zeigt die flache Seite in Richtung GND?
- Prüfe alle Verbindungsstellen, z.B. indem du mit der Spitzzange die Aderendhülsen nachdrückst.

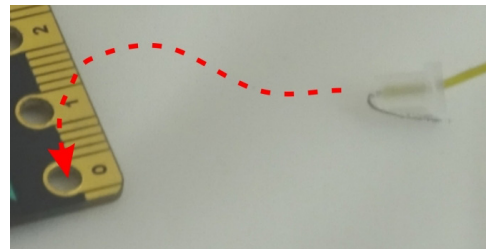
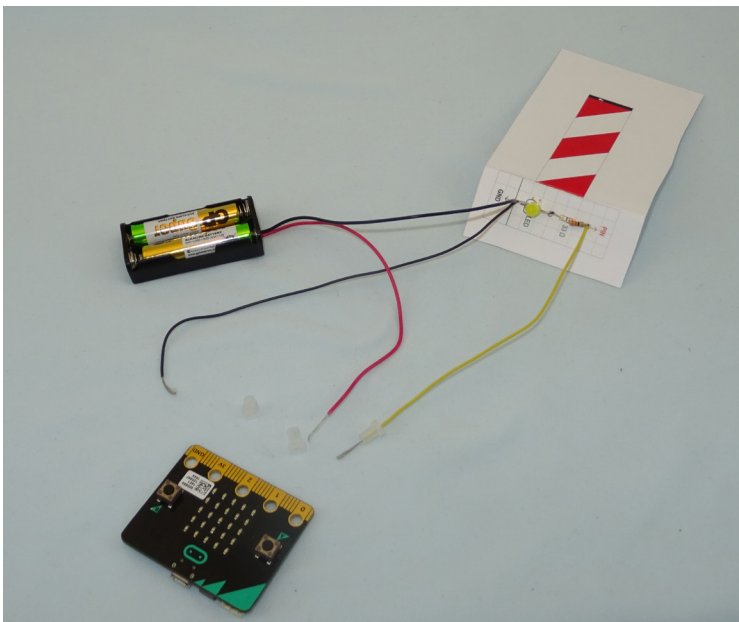
### → Anschluss an den Mikrocontroller

Wenn dieser Test erfolgreich war, dann kannst du die Schaltung mit dem Mikrocontroller verbinden. Die gelbe Litze schließt du an Pin 0 an. Dazu fädelst du die blanke Litze (ohne Isolierung) durch Reduzierhülsen und knickst sie danach noch um. Anschließend drückst du die Reduzierhülsen mit Litze passend in die Löcher des Mikrocontrollers.

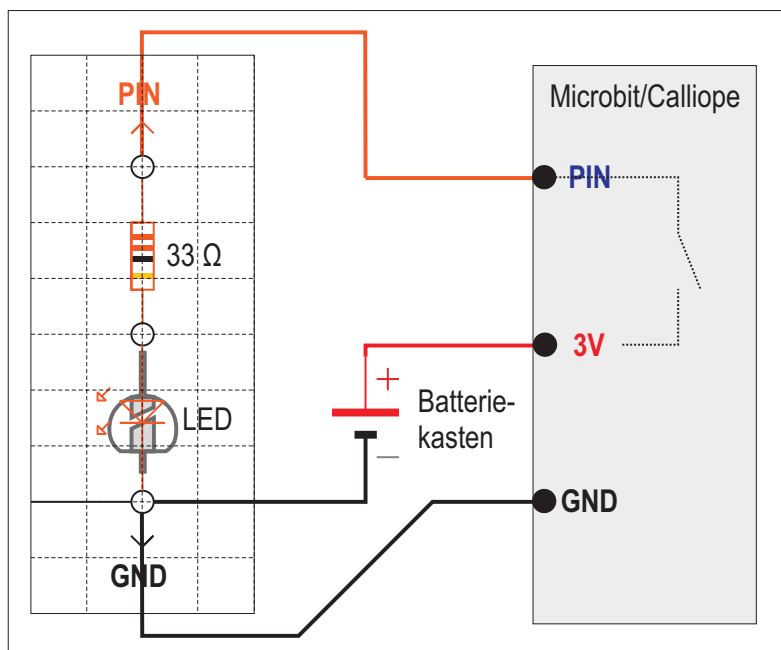
Schwarz an GND

Rot an 3V

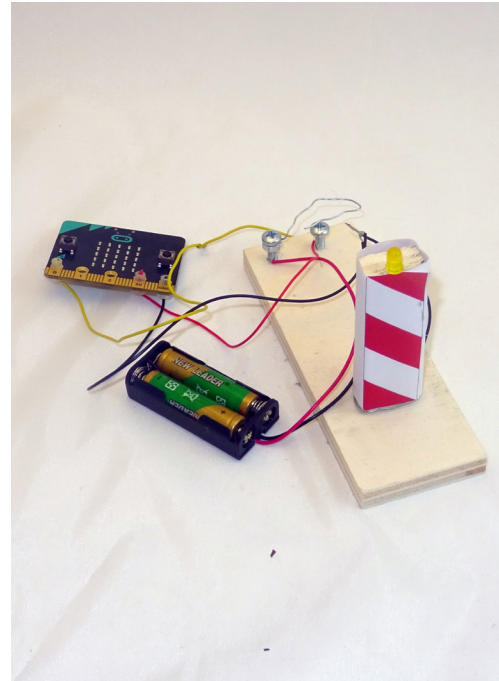
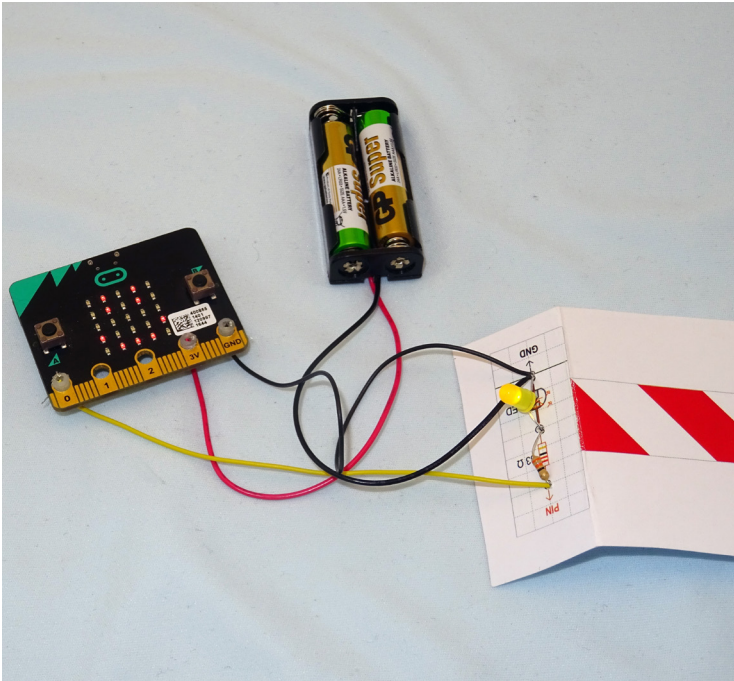
Gelb an Pin 0



### Verkabelungsplan Mikrocontroller



→ Programmierung des Mikrocontroller



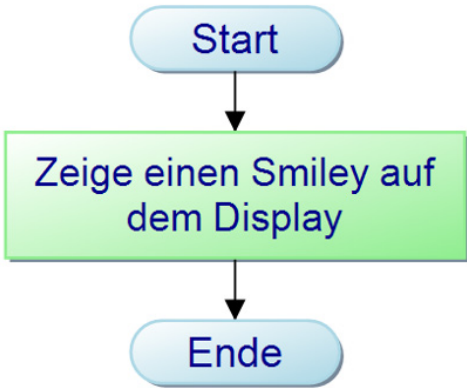

Schließe den Mikrocontroller (Calliope oder Micro:Bit) über ein USB-Kabel an den Computer und starte das Programm microBlocks<sup>7</sup>, Vorgehensweise unter Kapitel 7.3.

<sup>7</sup> Hinweis für Lehrer\*innen: Vorgehensweise unter Kapitel 7.3 und 7.4.



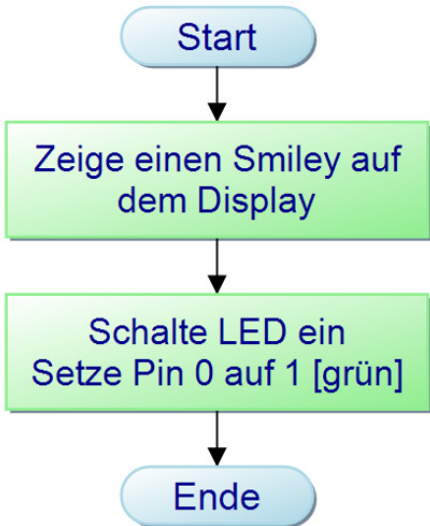



Zuerst überprüfst du mit dem ersten Programm, ob die Verbindung zwischen Computer und Mikrocontroller (Calliope oder Micro:Bit) funktioniert. Dafür erstellst du das folgende kleine Programm:

Programmablaufplan	Programm-Skript in MicroBlocks
 <pre> graph TD     Start([Start]) --&gt; Display[Zeige einen Smiley auf dem Display]     Display --&gt; Ende([Ende])         </pre>	

Wenn das Smiley auf dem Display (LED-Feld) des Mikrocontrollers erscheint, ist alles richtig.

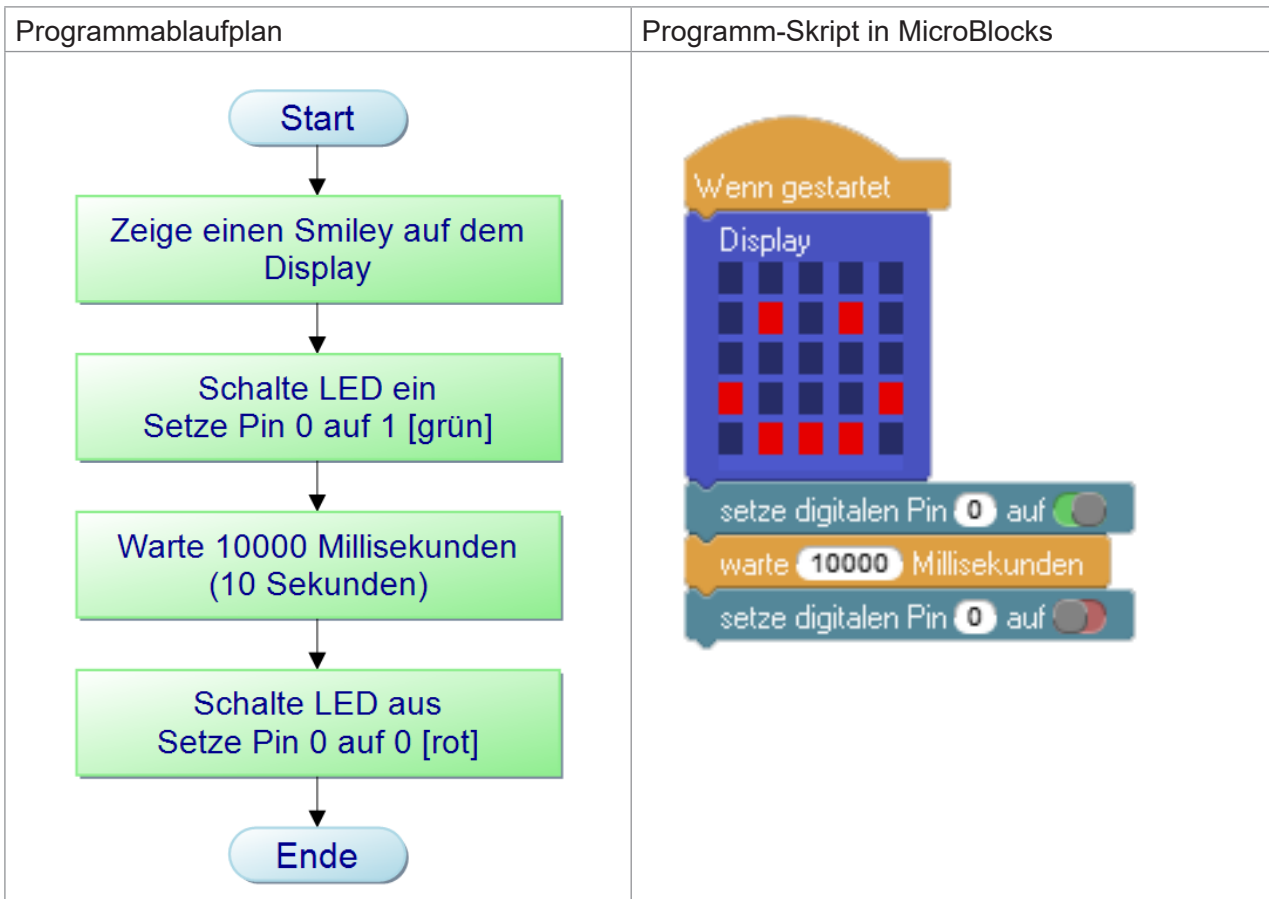
Als nächsten Schritt schaltest du die LED über den Mikrocontroller an. Dazu muss der Pin 0 angesteuert werden. Dafür verwendest du folgendes:

Programmablaufplan	Programm-Skript in MicroBlocks
 <pre> graph TD     Start([Start]) --&gt; Display[Zeige einen Smiley auf dem Display]     Display --&gt; LED[Schalte LED ein Setze Pin 0 auf 1 [grün]]     LED --&gt; Ende([Ende])         </pre>	

Wenn das Smiley auf dem Display (LED-Feld) des Mikrocontrollers erscheint und die LED leuchtet, hast du alles richtig gemacht.

Wenn die LED nicht leuchtet, kontrolliere, ob du den richtigen Pin 0 ausgewählt hast und er auf grün steht.

Die LED leuchtet nun unendlich lange, aber sie soll nach einer Zeit von 10 Sekunden (= 10 000 Millisekunden) wieder dunkel werden. Dazu musst du das Programm mit weiteren Befehlen ergänzen.



Wenn die LED 10-Sekunden leuchtet, hast du alles richtig gemacht. Super!

Wenn die LED nicht aus geht, kontrolliere,

- a) ob du den richtigen Pin 0 ausgewählt hast
- b) ob der Pin 0 auf rot steht.
- c) ob du die richtige Wartezeit angegeben hast

## 5.2 Bauauftrag Motor mit einer Richtung

### Informieren

Motoren erzeugen mechanische Bewegungen. Im einfachsten Fall kann eine Drehbewegung in einer Richtung realisiert werden z.B. bei Karussells, Rührgeräten und Ventilatoren. Eine einfache Umsetzung ist ein Ventilator, bei denen die Geschwindigkeit verändert werden kann.

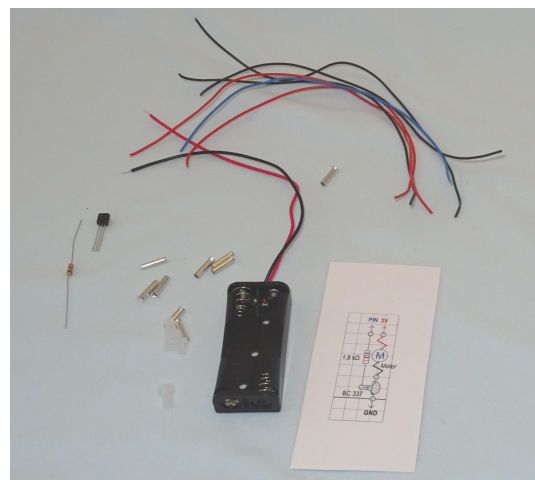
Der Auftrag lautet: Sorge dafür, dass sich die Rotoren deines Ventilators elektrisch drehen. Dafür brauchst du einen Motor.

### Planen und Entscheiden

Für den Betrieb des Ventilators brauchst du verschiedene Materialien. Suche dir diese zusammen und lege sie geordnet auf deinen Tisch<sup>8</sup>.

#### Material elektrisch:

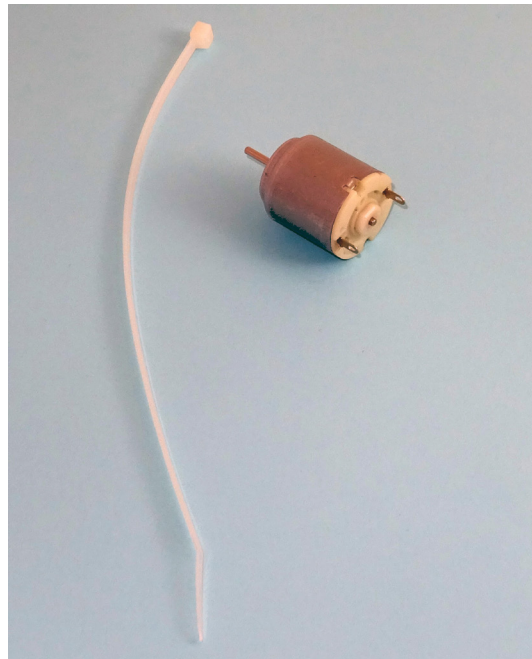
- 1 Widerstand 1,8 k $\Omega$ ,  
(Farbringe braun-grau-rot und gold)
- 1 npn-Transistor BC 337  
(eng stehende Drähte)
- 1 Kartonplatine (ausgedruckt als Vorlage)
- 1 Batteriekasten mit 2 Batterien
- je 2 Litzen von ca. 15 cm Länge in den  
Farben rot und schwarz
- 1 Litze von ca. 15 cm Länge in der  
Farbe blau
- 5 Aderendhülsen
- 3 Reduzierhülsen (4 auf 2 mm)
- 1 Mikrocontroller (Micro:Bit/Calliope)



<sup>8</sup> Beim Ventilator kommt die Grundschialtung „Motor in einer Richtung“ zur Anwendung (siehe Kapitel 4.4).

### Zusätzliches Material mechanisch:

- 1 Motor
- 1 Kabelbinder



### Fertigen und Montieren

So könnte dein Ventilator aussehen.

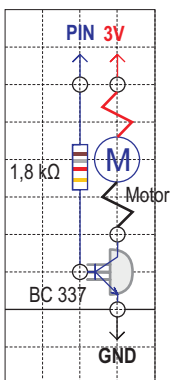


Wie bei der Baustellenbake, musst du auch hier eine Schaltung einbauen, damit der Ventilator über den Mikrocontroller gesteuert werden kann. Nur brauchst du hier andere Bauteile und einen anderen Schaltplan. Als grundlegende Arbeitsschritte machst du folgendes:

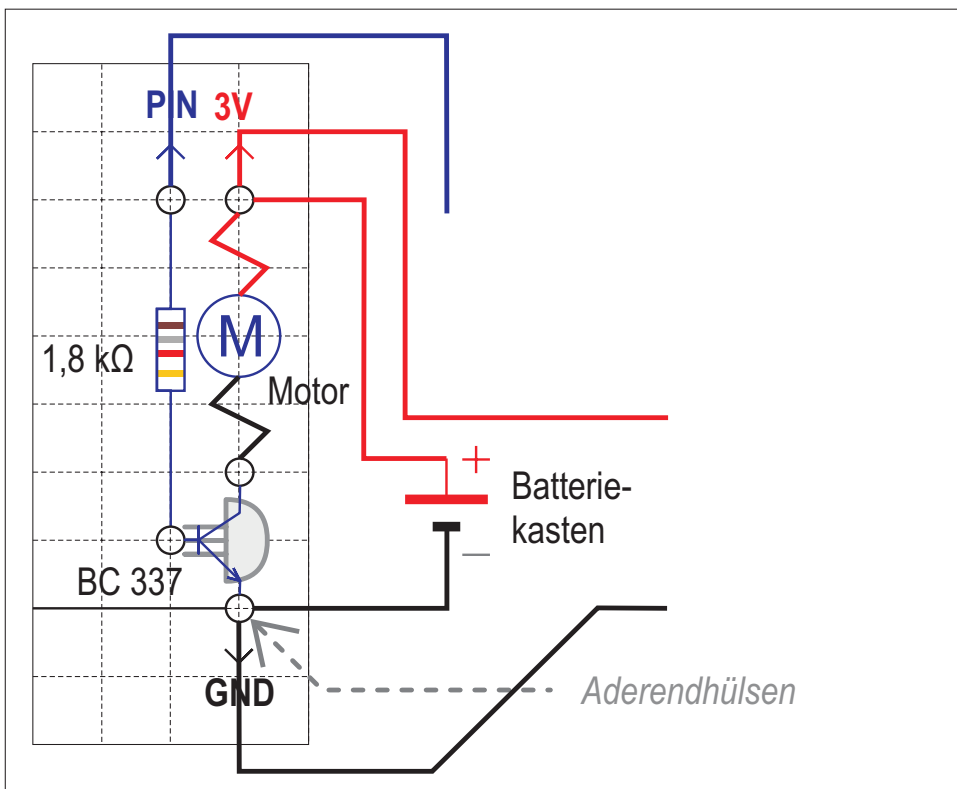
- Auf der Kartonplatine stichst du die Löcher vor.
- Du isolierst die benötigten Leitungen ab.
- Du biegst die Bauteile (npn-Transistor und Widerstand) und steckst diese in die Kartonplatine.
- Auf der Rückseite der Kartonplatine verbindest du alle Bauteile wie auf dem Verkabelungsplan gezeigt und verpresst die Drähte und Litzen mit einer Aderendhülse.

In den nachfolgenden Abbildungen findest du noch Hilfe, wie die Arbeitsschritte im Einzelnen aussehen.

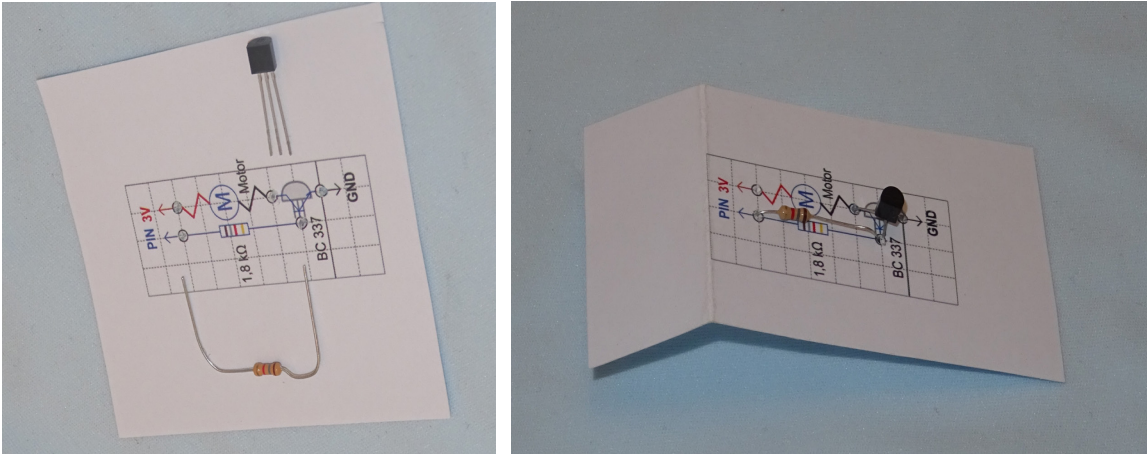
### Kartonplatine



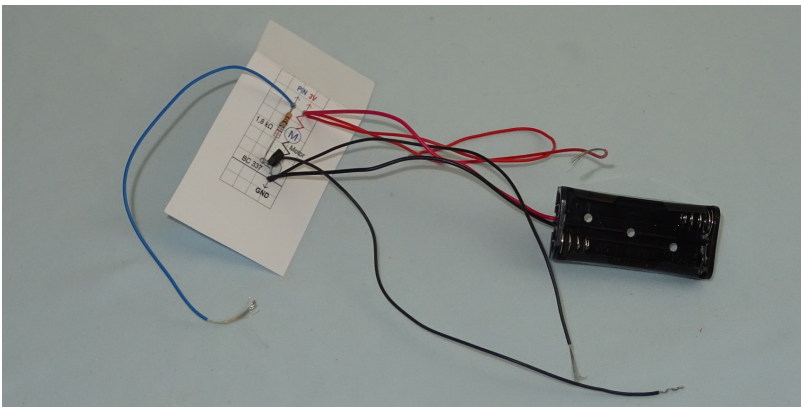
### Verkabelungsplan für den Aufbau



Und so gehst du genau vor

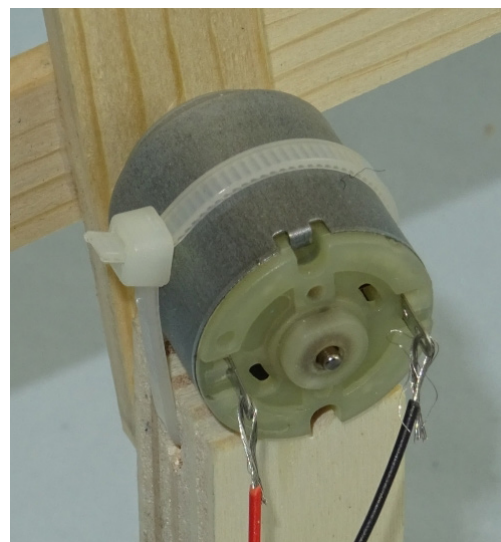
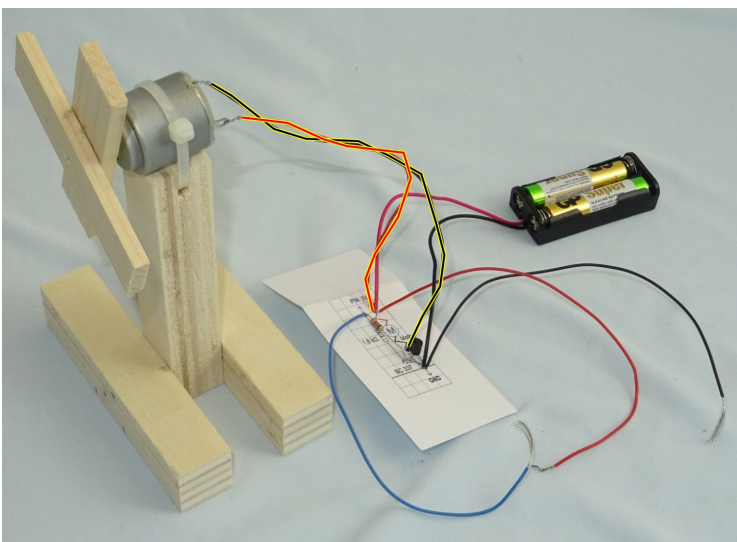


Widerstand und Transistor auf Kartonplatine einsetzen, die Drähte durch die Löcher stecken und an der Unterseite mit einer Aderendhülse verpressen.



Batteriekasten anschließen.

Rote Litze (gelb hinterlegt) von 3V und schwarze Litze (gelb hinterlegt) vom Transistor an den Motor anschließen. Die Litzen fädelt du durch die Ösen des Motors und verzwirbelst sie.



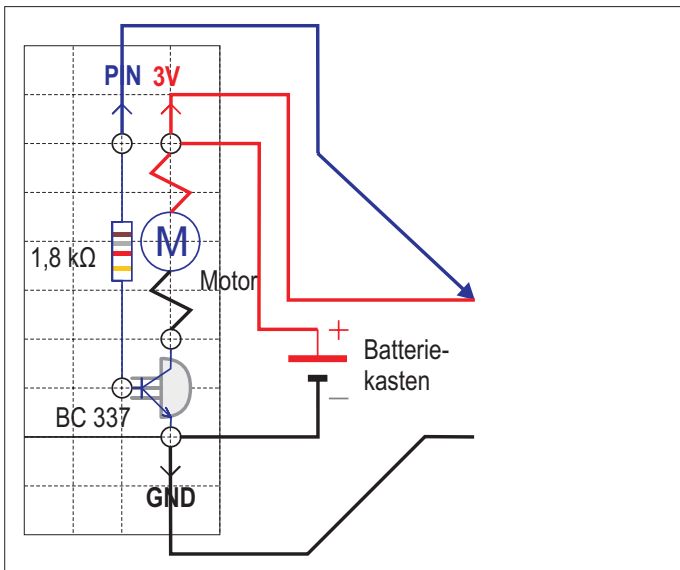
## In Betrieb nehmen, Einstellen und Überprüfen

Nachdem das Produkt nun fertig ist, muss es in Betrieb genommen werden:

### → Testen der elektrischen Schaltung

Funktionsprüfung: du hältst die blaue Litze an die rote Litze. (Pluspol des Batteriekasten). Wenn du alles richtig gemacht hast, läuft der Motor.

### Verkabelungsplan für den Test



Wenn der Motor nicht läuft, dann überprüfe folgendes:

1. Hast du die Batterien im Batterie-kasten richtig eingelegt? (Plus- und Minuspol an der richtigen Stelle?)
2. a) Hast du den richtigen Typ des Transistors BC 337 („Kamm“-Transistor, eng stehende Anschlussdrähte) verbaut?  
b) Hast du den Transistor lagerichtig eingebaut? Schau nach, ob der Kopf des Transistors von oben wie auf dem Verkabelungsplan aussieht und ob du die Anschlussdrähte in die richtigen Löcher gesteckt hast.
3. Prüfe alle Verbindungsstellen, z.B. indem du mit der Spitzzange die Aderendhülsen noch einmal zusammenpresst

## Anschluss an den Mikrocontroller

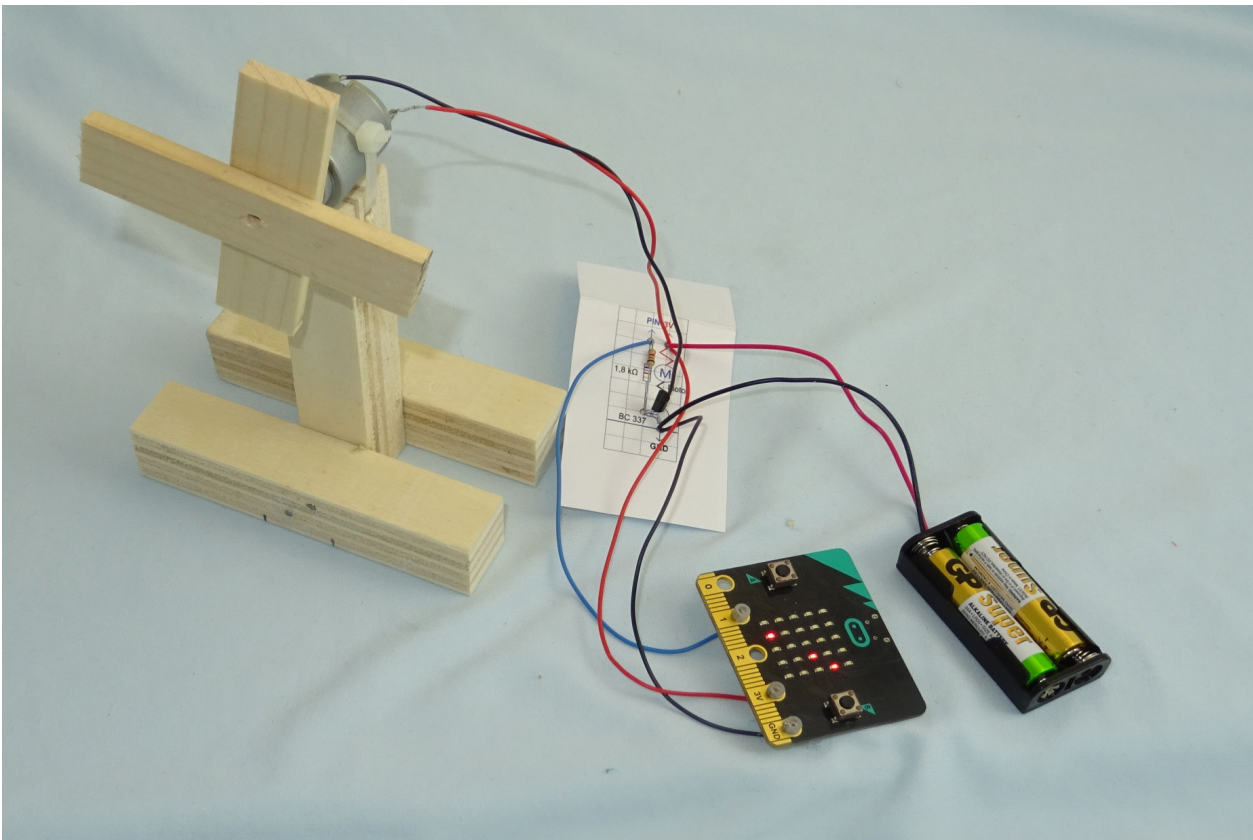
Wenn dieser Test erfolgreich war, dann kannst du die Schaltung mit dem Mikrocontroller verbinden. Die blaue Leitung schließt du an Pin 1 an<sup>9</sup>.

Dazu fädelst du die Litze durch Reduzierhülsen und knickst sie danach noch um. Anschließend drückst du die Reduzierhülsen mit Litze passend in die Löcher des Mikrocontrollers.

Schwarz an GND

Rot an 3V

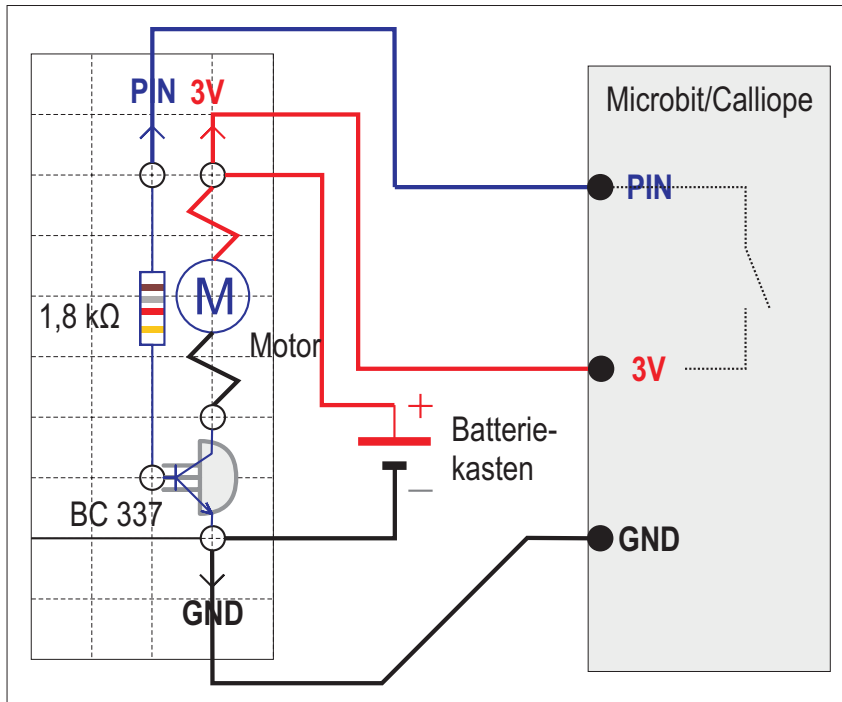
Blau an Pin 1



<sup>9</sup> Wie in Kapitel 4.2 beschrieben, haben wir uns auf diese Zuordnung bei KiTec Digital festgelegt. Es ist keine allgemeingültige Zuordnung, sondern dient der leichteren Verständigung und hilft bei der Fehlersuche.



## Verkabelungsplan Mikrocontroller



## Programmierung des Mikrocontrollers

Schließe den Mikrocontroller (Calliope oder Mico:Bit) über ein USB-Kabel an den Computer und starte das Programm microBlocks. Vorgehensweise unter Kapitel 4.3<sup>10</sup>.

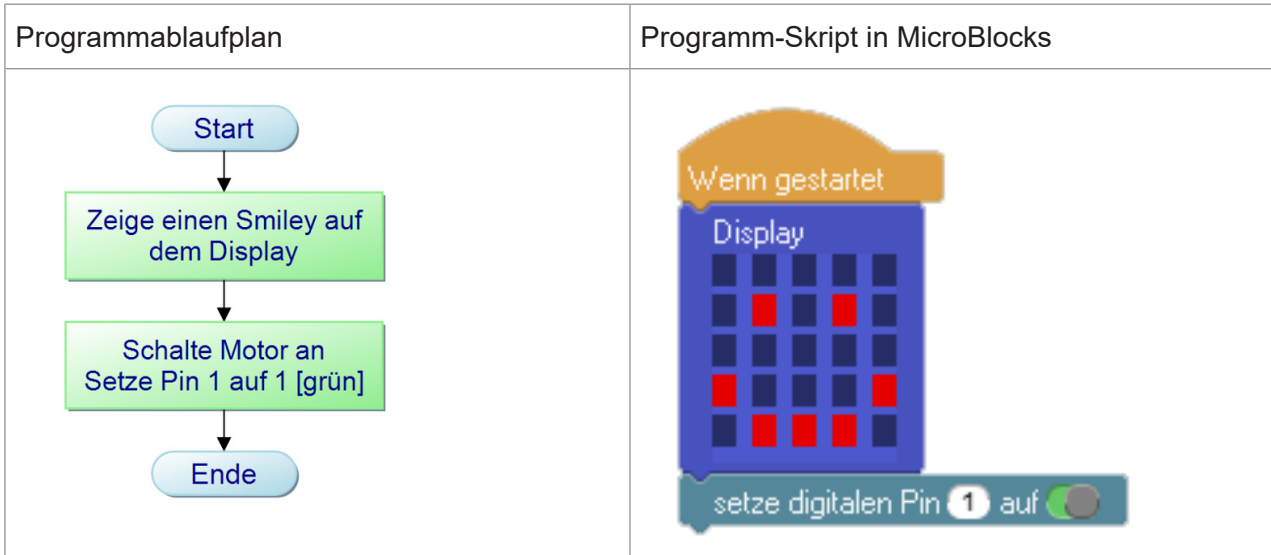
Zuerst überprüfst du mit dem ersten Programm, ob die Verbindung zwischen Computer und Mikrocontroller (Micro:Bit oder Calliope) funktioniert. Dafür erstellst du das folgende kleine Programm:

Programmablaufplan	Programm-Skript in MicroBlocks
<pre> graph TD     Start([Start]) --&gt; Display[Zeige einen Smiley auf dem Display]     Display --&gt; Ende([Ende])         </pre>	

<sup>10</sup> Hinweis für Lehrer\*innen: Vorgehensweise unter Kapitel 7.3 und 7.4.

Wenn das Smiley auf dem Display (LED-Feld) des Mikrocontrollers erscheint, ist alles richtig.

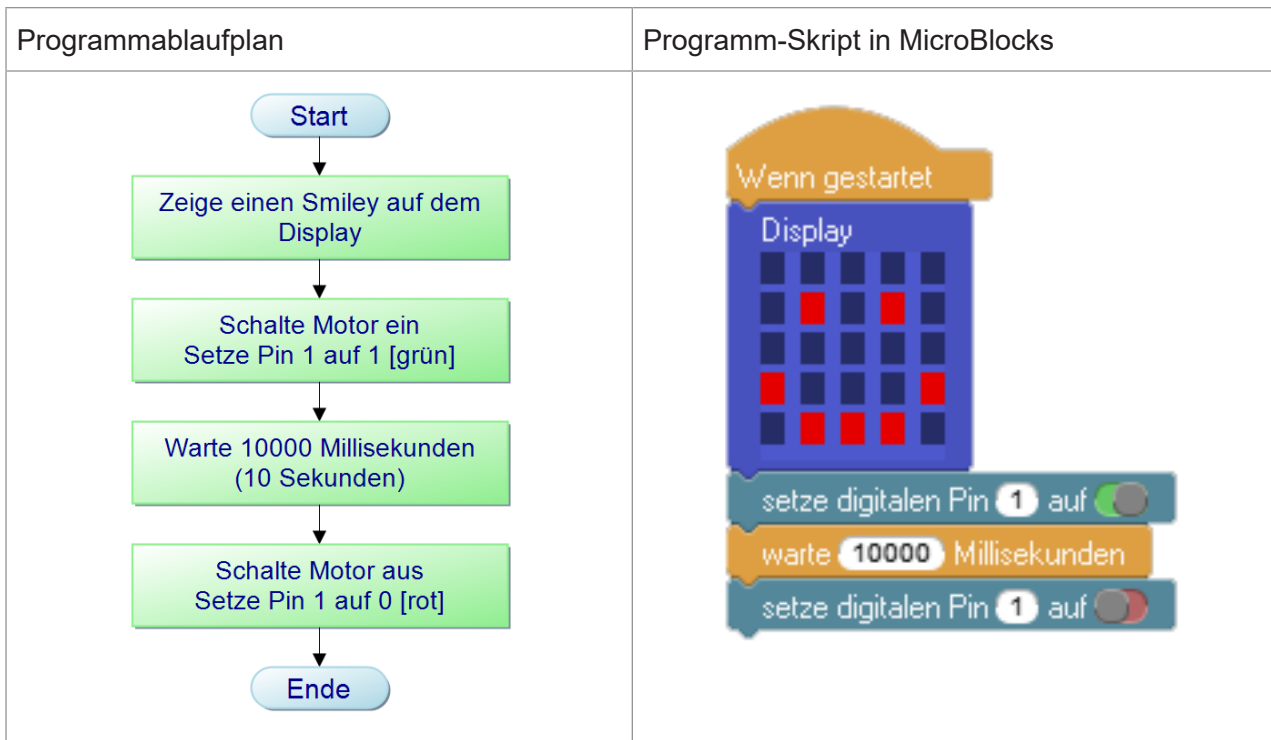
Als nächsten Schritt schaltest du den Motor über den Mikrocontroller an. Dazu muss der Pin 1 an-gesteuert werden. Dafür verwendest Du folgendes:



Wenn das Smiley auf dem Display (LED-Feld) des Mikrocontrollers erscheint und der Motor sich dreht, hast du alles richtig gemacht.

Wenn der Motor sich nicht dreht, kontrolliere, ob du den richtigen Pin 1 ausgewählt hast und er auf grün steht.

Der Motor dreht sich nun unendlich lange, aber er soll nach einer Zeit von 10 Sekunden (= 10 000 Millisekunden) wieder ausgeschaltet werden. Dazu musst du das Programm mit weiteren Befehlen ergänzen.



Wenn sich der Motor nach 10 Sekunden wieder ausschaltet, hast du alles richtig gemacht. Super!

Wenn der Motor sich nicht abschaltet, kontrolliere,

- a) ob du den richtigen Pin 1 ausgewählt hast.
- b) ob der Pin 1 auf rot steht.
- c) ob du die richtige Wartezeit angegeben hast.

## 5.3 Bauauftrag Motor mit zwei Richtungen

### Informieren

Motoren erzeugen mechanische Bewegungen. In vielen technischen Systemen ist es dabei erforderlich, in beiden Richtungen zu arbeiten wie z.B. bei Aufzügen und Fahrzeugen.

Der Auftrag lautet: Erstelle eine Schaltung, mit der das Fahrzeug vorwärts und rückwärts fahren kann. Dafür wird ein Motor benötigt.

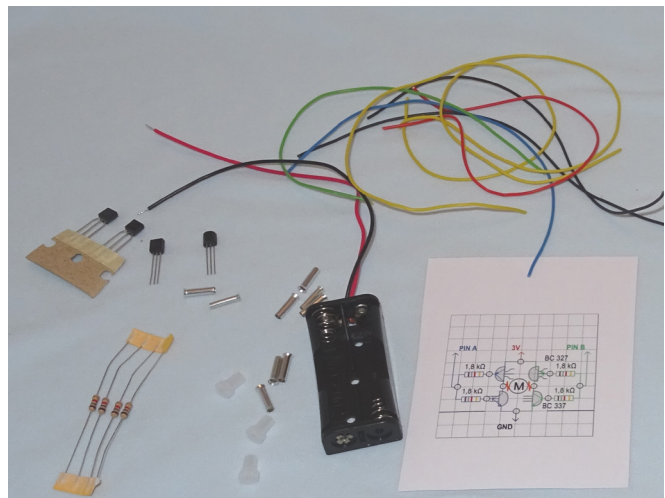
### Planen und Entscheiden

Beim Fahrzeug kommt die Grundschaltung „Motor mit zwei Richtungen“ zur Anwendung. Die Welle eines Elektromotors läuft relativ schnell. Für ein Fahrzeug braucht man aber eine langsame Bewegung. Die Umwandlung von einer schnellen in eine langsame Bewegung erfolgt durch ein Getriebe, wofür eine Schnecke und ein Zahnrad benötigt wird.

Für den Betrieb des Fahrzeuges brauchst du verschiedene Materialien. Suche dir diese zusammen und lege sie geordnet auf deinen Tisch.

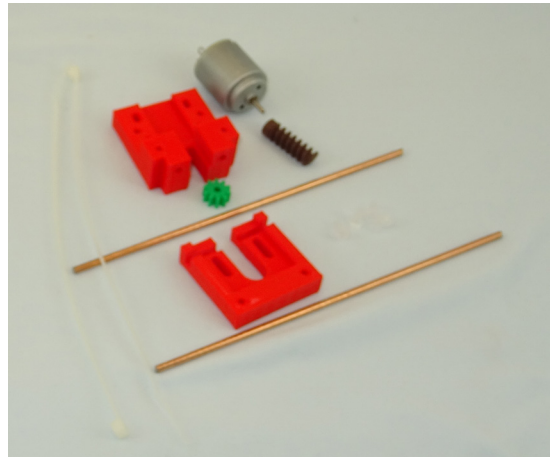
### Material, elektrisch:

- 4 Widerstände 1,8 k $\Omega$   
(Farbringe braun-grau-rot und gold)
- 2 npn-Transistor BC 337  
(eng stehende Drähte)
- 2 pnp-Transistor BC 327  
(weit stehende Drähte)
- 1 Kartonplatine (ausgedruckt als Vorlage)
- 1 Batteriekasten mit 2 Batterien
- je 1 Litze von ca. 15 cm Länge in den  
Farben rot, schwarz, blau und grün
- 2 Litzen von ca. 20 cm Länge in der  
Farbe gelb
  
- 10 Aderendhülsen
- 4 Reduzierhülsen (4 auf 2 mm)
- 1 Mikrocontroller (Micro:Bit/Calliope)



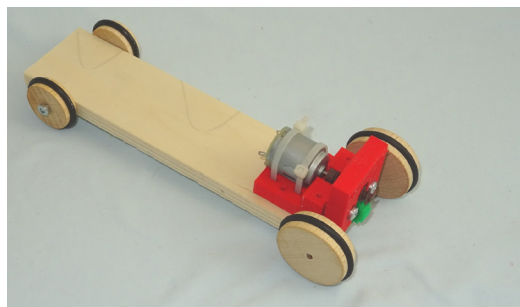
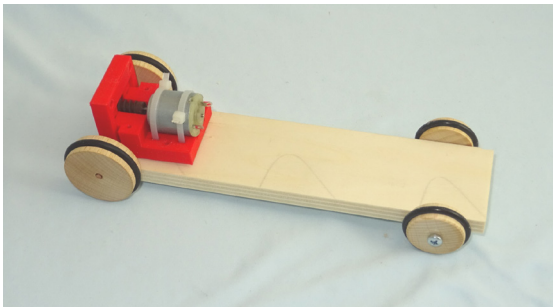
### Zusätzliches Material mechanisch:

- Schweißstab 3 mm, die Anzahl musst du selber festlegen und mit einer Bügelsäge zurecht sägen.
- 2 Reduzierhülsen (4 auf 3 mm)
- 1 Motor
- 2 Kabelbinder
- 1 Getriebeblock
- 1 Schnecke
- 1 Zahnrad mit 20 Zähnen (wahlweise auch 10 Zähnen)

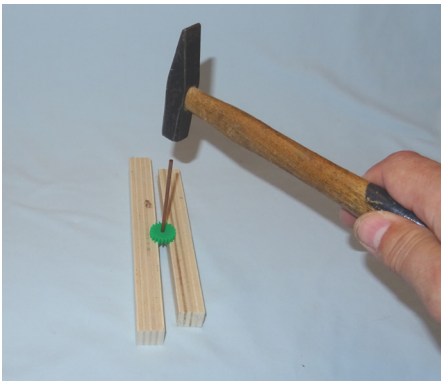


### Fertigen und Montieren

So könnte dein Fahrzeug aussehen.



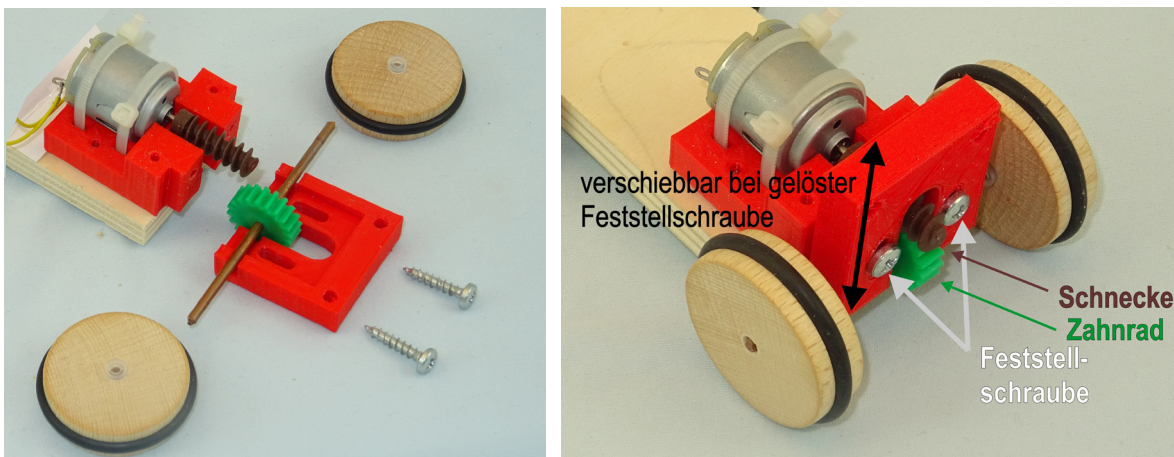
Hinweise zum Zusammensetzen des Getriebes:



Kürze zuerst den Schweißstab so, dass er zu deinem Bauwerk passt. Hierfür kannst du die Bügelsäge aus KiTec verwenden.

Presse den Schweißstab dann in das Loch des Zahnrades. Du kannst dir diese Arbeit erleichtern, indem du den Stab mit einem Hammer einschlägst. Aber Vorsicht: Du benötigst zwei Holzleisten, auf die du das Zahnrad legst. Der Stab kann nun durch das Loch geschlagen werden, sodass er anderen Seite auch übersteht.

Die Holzleisten sind hier als Ersatz für die Schraubstockbacken zu sehen.



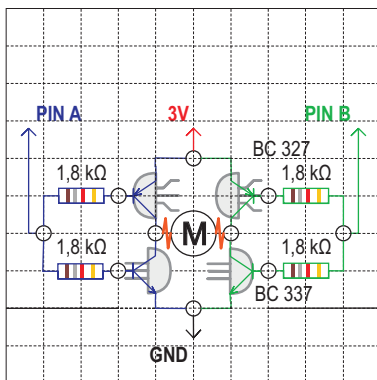
Der Motor wird durch Kabelbinder am Getriebeblock befestigt. Die Schnecke schiebst du auf die Motorachse. Bei den Rädern kommen nun die Reduzierhülsen (die Nabe) zum Einsatz: Du steckst sie in die vorgebohrten Löcher der Räder. Sie reduzieren (= verringern) den Durchmesser des Lochs, so dass das Rad fest auf der Achse sitzt.

Die Feststellschrauben werden zuerst so angezogen, dass sich der hintere Block noch verschieben lässt. Die Einstellung muss so sein, dass zwischen Schnecke und Zahnrad noch leichtes Spiel ist (etwas locker). Erst dann werden die Feststellschrauben fest angezogen. Durch Anlegen der Batteriespannung an beide Ösen des Motors muss sich dieser leicht drehen können.

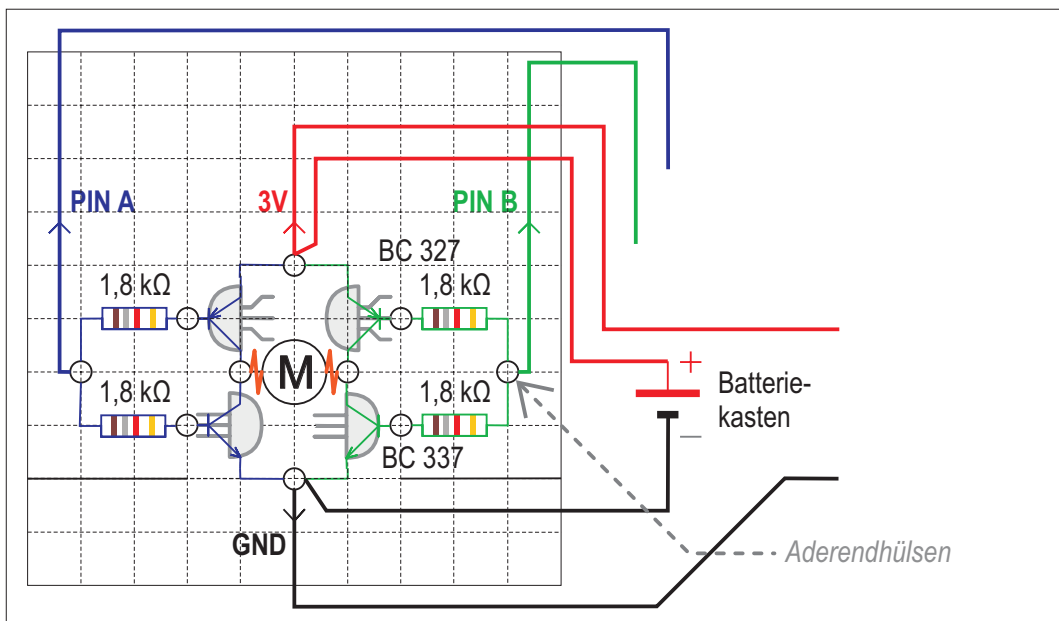
Für die Herstellung der Schaltung musst du Dich an den Aufbau auf der Kartonplatine halten. Als Arbeitsschritte erfolgen:

- Auf der Kartonplatine stichst du die Löcher vor
- Du isolierst die benötigten Leitungen ab
- Du biegst die Bauteile und steckst diese in die Kartonplatine.
- Auf der Rückseite der Kartonplatine verbindest du alle Bauteile gemäß Verkabelungsplan.

### Kartonplatine

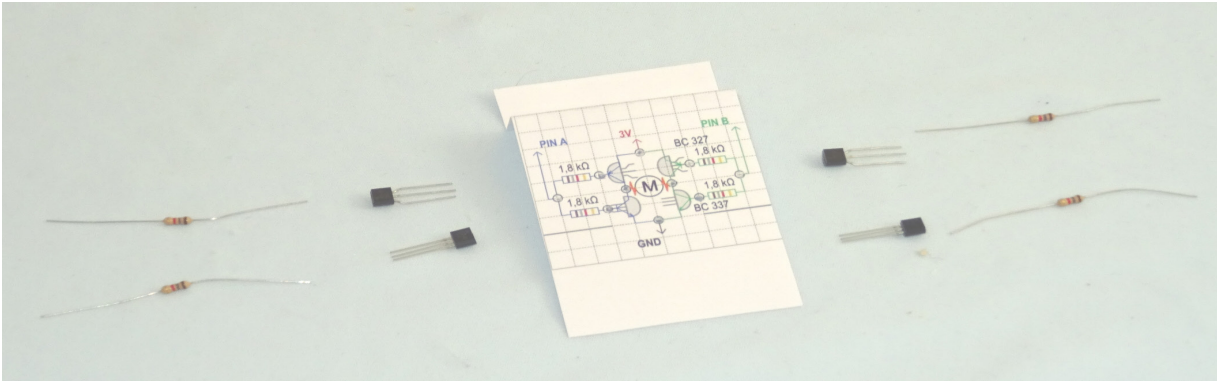


### Verkabelungsplan für den Aufbau

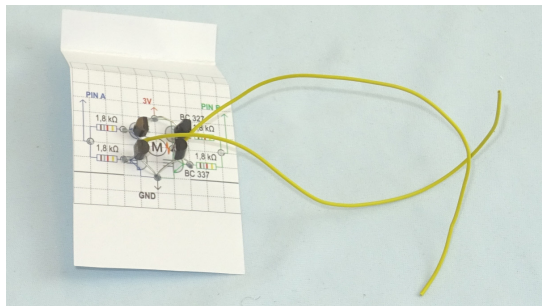
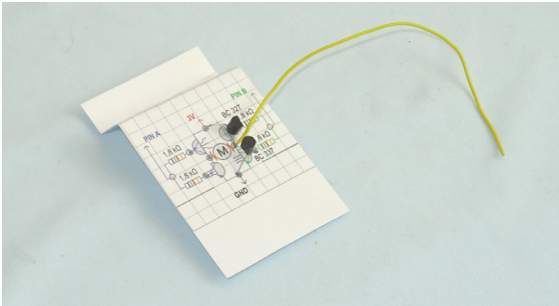


Für den Anschluss an den Mikrocontroller werden nur die Litzen in den Farben rot, schwarz, grün und blau benötigt. Die Farbe gelb (auf der Kartonplatine zur besseren Sichtbarkeit in orange eingezeichnet) ist übrig und wird deshalb für den Anschluss an den Motor verwendet.

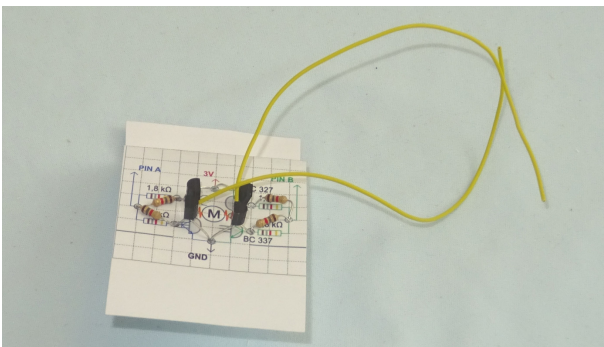
## Und so gehst du Schritt für Schritt vor



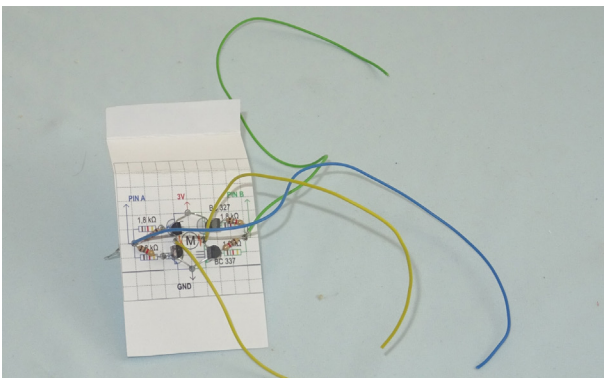
Bereitlegen der Transistoren und Widerstände. Die Anschlussdrähte der Transistoren erst unmittelbar vor dem Einbau zurechtbiegen, da sie sonst leicht zu verwechseln sind.



Transistoren einstecken, die gelbe Litze zwischen beiden Transistoren berücksichtigen und jeweils mit einer Aderendhülse verpressen.

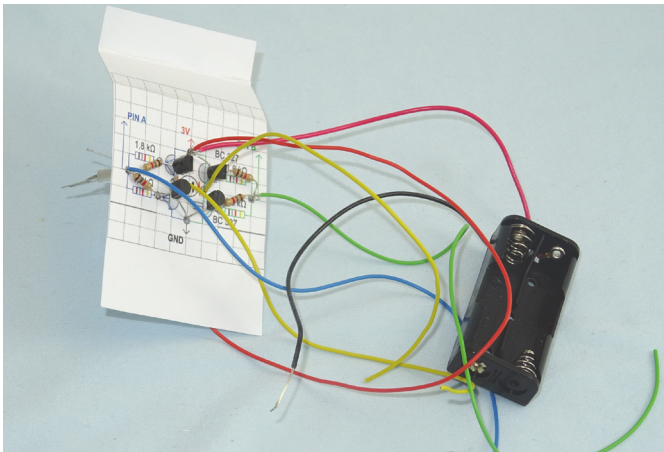


Widerstände einstecken und mit den Transistoren verbinden.

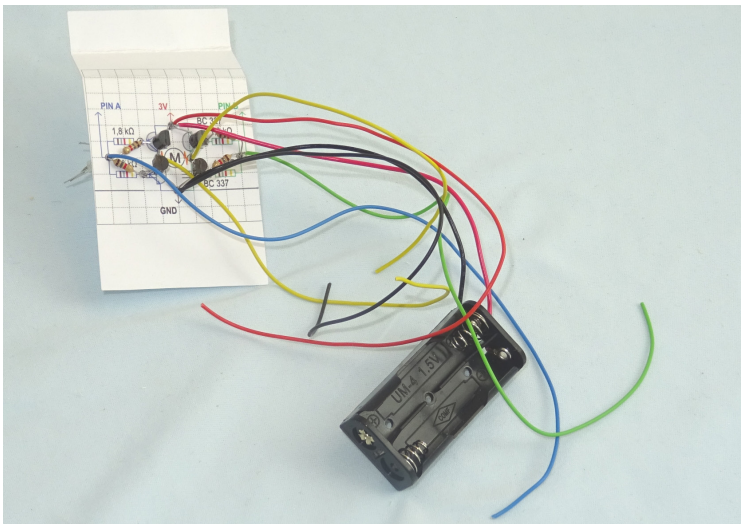


Grüne und blaue Litze jeweils an den Widerständen einstecken und verpressen.

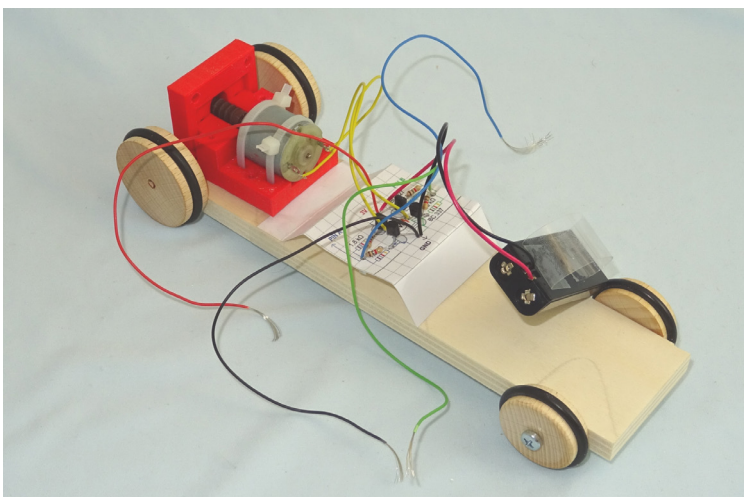




Rote Litze des Batteriekastens mit einer separaten roten Litze (für den Mikrocontroller) verzwirbeln, in Loch bei 3V stecken und mit den Anschlussdrähten des Transistors verbinden, die durch das Loch bei 3V gesteckt wurden.



Schwarze Litze des Batteriekastens mit einer separaten schwarzen Litze (für den Mikrocontroller) verzwirbeln, in Loch bei GND stecken und mit den Anschlussdrähten des Transistors verbinden, die durch das Loch bei GND gesteckt wurden.



Gelbe Litze mit dem Motor verbinden (durch die Öse stecken und verzwirbeln).  
Kartonplatine und Batteriekasten mit Klebeband am Fahrgestell befestigen.

## **In Betrieb nehmen, Einstellen und Überprüfen**

Nachdem das Produkt nun fertig ist, muss es in Betrieb genommen werden:

### **→ Testen der elektrischen Schaltung**

Funktionsprüfung:

Der Motor läuft nur dann, wenn du folgendes machst:

Der Motor läuft rechtsherum, wenn du

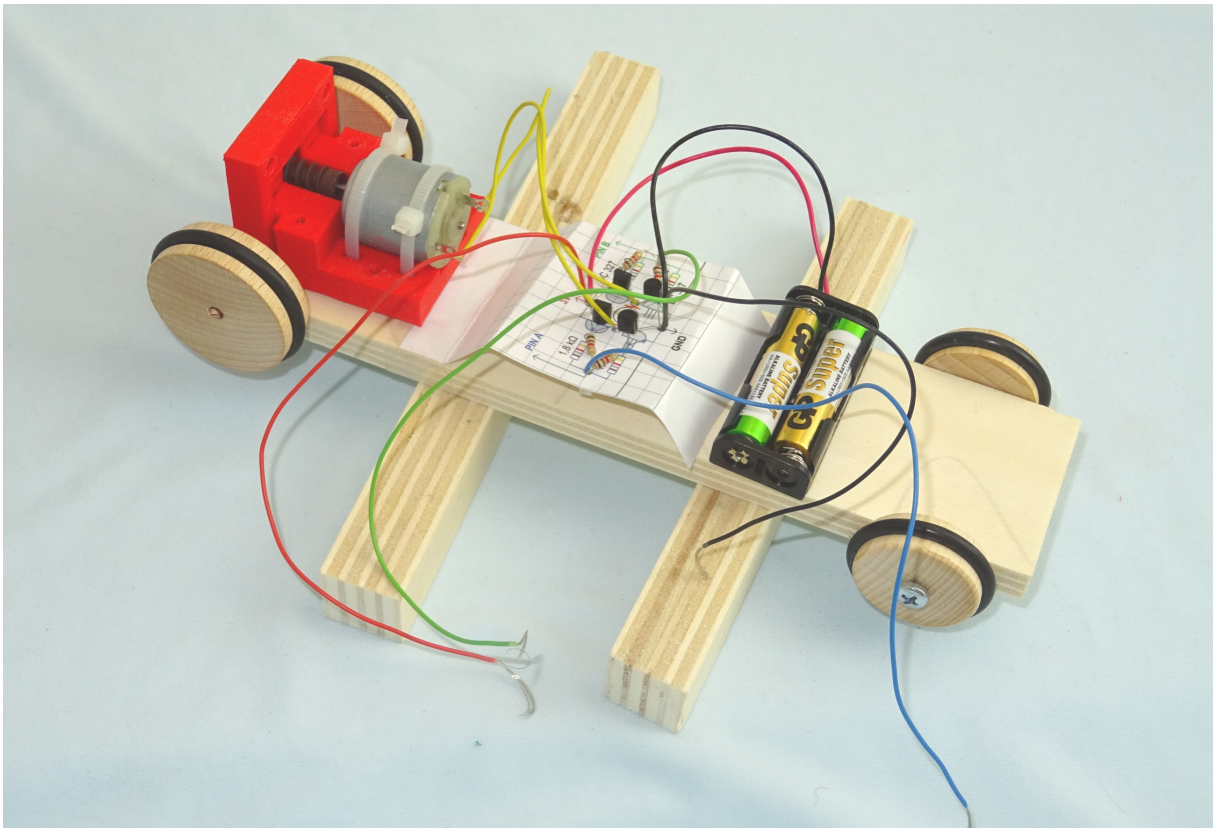
- die blaue Litze an die schwarze Litze hältst  
und
- die grüne Litze an die rote Litze hältst

Der Motor läuft linksherum, wenn du

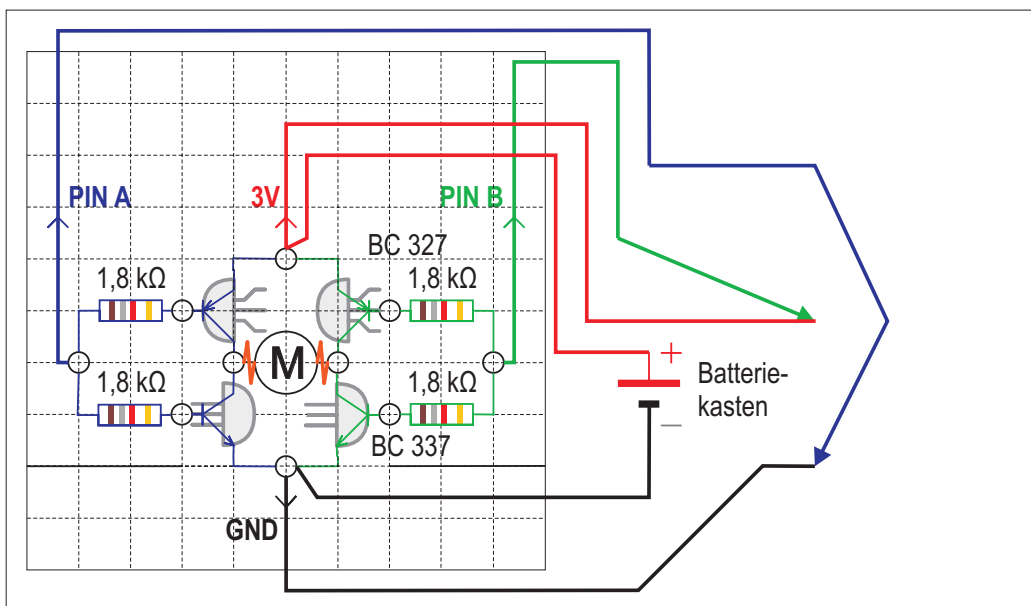
- die blaue Litze an die rote Litze hältst  
und
- die grüne Litze an die schwarze Litze hältst

Bei anderen Farbkombinationen darf sich der Motor nicht drehen.

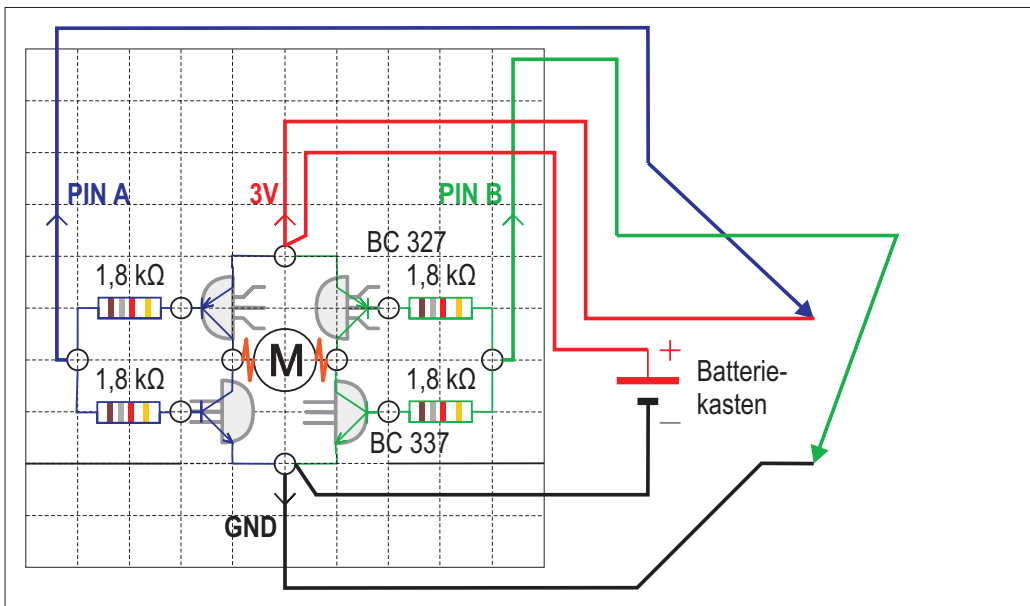
Durch Aufbocken – zwei Holzleisten unterlegen – kann das Hinterrad sich frei bewegen. Dies wird folgendermaßen getestet:



### Verkabelungsplan für den Test Rechtslauf



## Verkabelungsplan für den Test Linkslauf



Wenn der Motor nicht läuft, dann überprüfe folgendes:

1. Hast du die Batterien im Batterie-kasten richtig eingelegt?  
(Plus- und Minuspol an der richtigen Stelle?)
2. a) Hast du die richtigen Typen des Transistors BC 337 bzw. BC 327 („Kamm“-Transistor, bzw. „Gabel“-Transistor) an den richtigen Stellen verbaut?  
b) Hast du die Transistoren lagerichtig eingebaut? Schau nach, ob der Kopf des Transistors von oben wie auf dem Verkabelungsplan aussieht.  
c) Hast du die Anschlussdrähte in die richtigen Löcher gesteckt?
3. Prüfe alle Verbindungsstellen, z.B. indem du mit der Spitzzange die Aderendhülsen noch einmal zusammenpresst.

### Anschluss an den Mikrocontroller

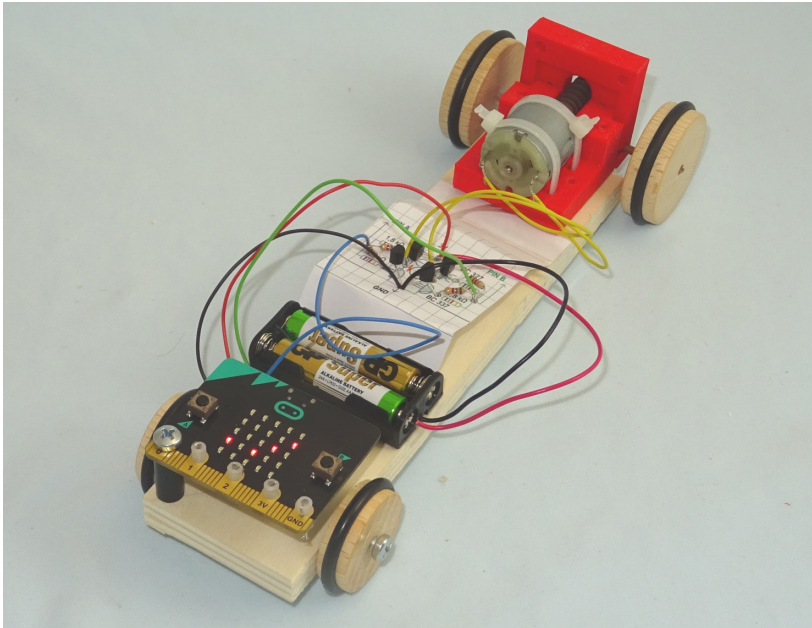
Wenn dieser Test erfolgreich war, dann kannst du die Schaltung mit dem Mikrocontroller verbinden. Die blaue Litze wird mit Pin 1 und die grüne Litze wird mit Pin 2 verbunden. Dazu fädelt du die Litze durch Reduzierhülsen und knickst sie danach noch um. Anschließend drückst du die Reduzierhülsen mit Litze passend in die Löcher des Mikrocomputers.

Schwarz an GND

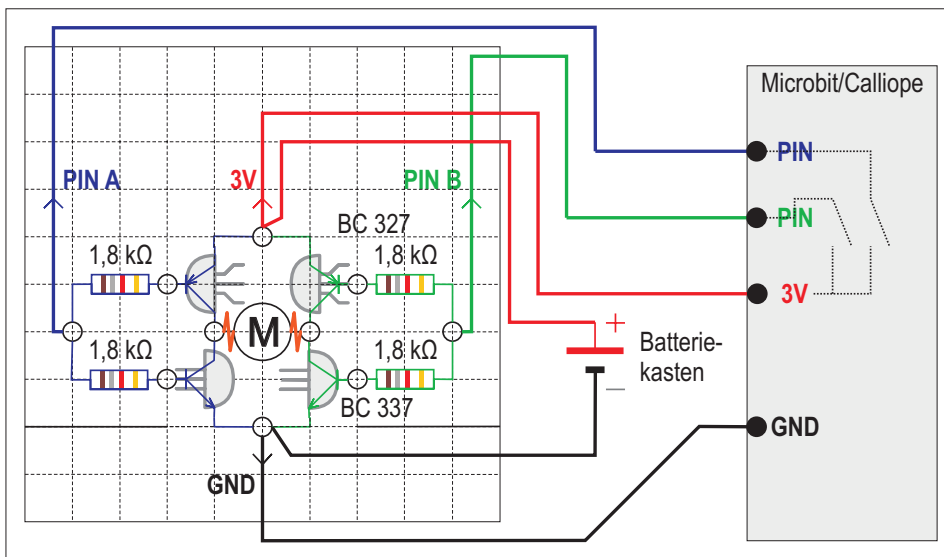
Rot an 3V

Blau an Pin 1

Grün an Pin 2



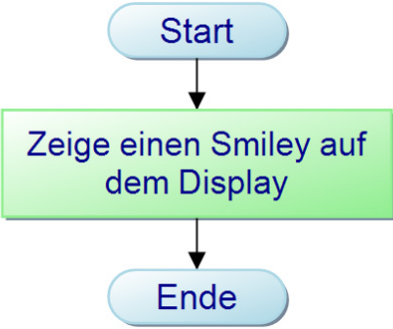

### Verkabelungsplan für den Einplatinencomputer



## Programmierung des Mikrocontroller

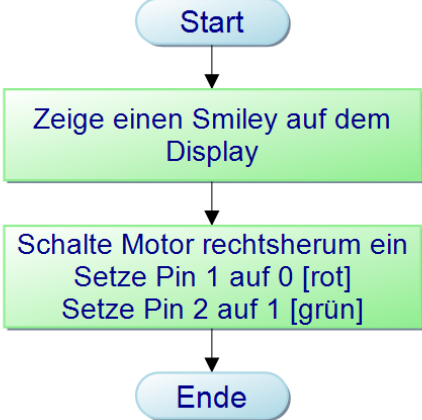

Schließe den Mikrocontroller (Micro:Bit oder Calliope) über ein USB-Kabel an den Computer und starte das Programm microBlocks. Vorgehensweise unter Kapitel 7.3<sup>11</sup>.

Zuerst überprüfst du mit dem ersten Programm, ob die Verbindung zwischen Computer und Mikrocontroller (Micro:Bit oder Calliope) funktioniert. Dafür erstellst du das folgende kleine Programm:

Programmablaufplan	Programm-Skript in MicroBlocks
 <pre> graph TD     Start([Start]) --&gt; Display[Zeige einen Smiley auf dem Display]     Display --&gt; Ende([Ende])         </pre>	

Wenn das Smiley auf dem Display (LED-Feld) des Mikrocontrollers erscheint, ist alles richtig.

Als nächsten Schritt schaltest du den Motor über den Mikrocontroller an. Dazu muss der Pin 1 und Pin 2 angesteuert werden. Dafür verwendest du folgendes:

Programmablaufplan	Programm-Skript in MicroBlocks
 <pre> graph TD     Start([Start]) --&gt; Display[Zeige einen Smiley auf dem Display]     Display --&gt; Motor[Schalte Motor rechtsherum ein Setze Pin 1 auf 0 [rot] Setze Pin 2 auf 1 [grün]]     Motor --&gt; Ende([Ende])         </pre>	

<sup>11</sup> Hinweis für Lehrer\*innen: Vorgehensweise unter Kapitel 7.3 und 7.4.

Wenn das Smiley auf dem Display (LED-Feld) des Mikrocontrollers erscheint und der Motor sich rechts herum dreht, hast du alles richtig gemacht.

Wenn der Motor sich nicht dreht, kontrolliere,

- a) ob du Pin 1 auf rot gesetzt hast
- b) ob du Pin 2 auf grün gesetzt hast

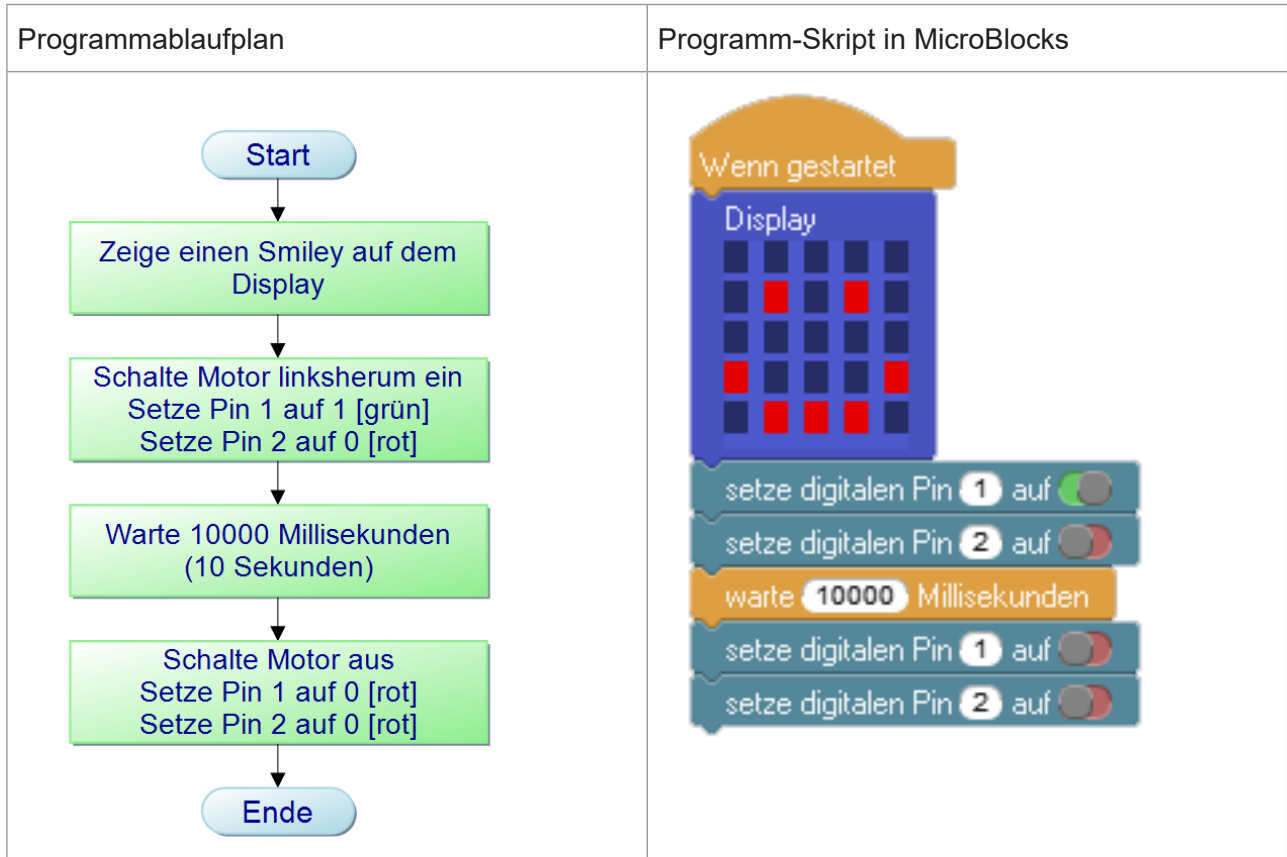
Der Motor dreht sich nun unendlich lange, aber er soll nach einer Zeit von 10 Sekunden (= 10 000 Millisekunden) wieder ausgeschaltet werden. Dazu musst du das Programm mit weiteren Befehlen ergänzen.

Programmablaufplan	Programm-Skript in MicroBlocks
<pre> graph TD     Start([Start]) --&gt; Display[Zeige einen Smiley auf dem Display]     Display --&gt; Motor[Schalte Motor rechtsherum ein Setze Pin 1 auf 0 [rot] Setze Pin 2 auf 1 [grün]]     Motor --&gt; Wait[Warte 10000 Millisekunden (10 Sekunden)]     Wait --&gt; Stop[Schalte Motor aus Setze Pin 1 auf 0 [rot] Setze Pin 2 auf 0 [rot]]     Stop --&gt; End([Ende])         </pre>	<pre> graph TD     Start[Wenn gestartet] --&gt; Display[Display]     Display --&gt; Pin1[setze digitalen Pin 1 auf ]     Pin1 --&gt; Pin2[setze digitalen Pin 2 auf ]     Pin2 --&gt; Wait[warte 10000 Millisekunden]     Wait --&gt; Pin1Off[setze digitalen Pin 1 auf ]     Pin1Off --&gt; Pin2Off[setze digitalen Pin 2 auf ]         </pre>

Wenn der Motor sich nicht abschaltet, kontrolliere,

- a) ob der Pin 1 auf rot gesetzt ist.
- b) ob der Pin 2 auf rot gesetzt ist.
- c) ob du die richtige Wartezeit angegeben hast.

Du hast jetzt ein Programm eingesetzt, mit dem der Motor für eine Zeit von 10 Sekunden rechts herum läuft. Damit der Motor links herum läuft, müssen Pin 1 und Pin 2 beim Einschalten anders gesetzt werden: Pin 1 auf 1 (grün) und Pin 2 auf 0 (rot)



Wenn der Motor sich nicht abschaltet, kontrolliere,

- ob der Pin 1 auf rot gesetzt ist.
- ob der Pin 2 auf rot gesetzt ist.
- ob du die richtige Wartezeit angegeben hast.



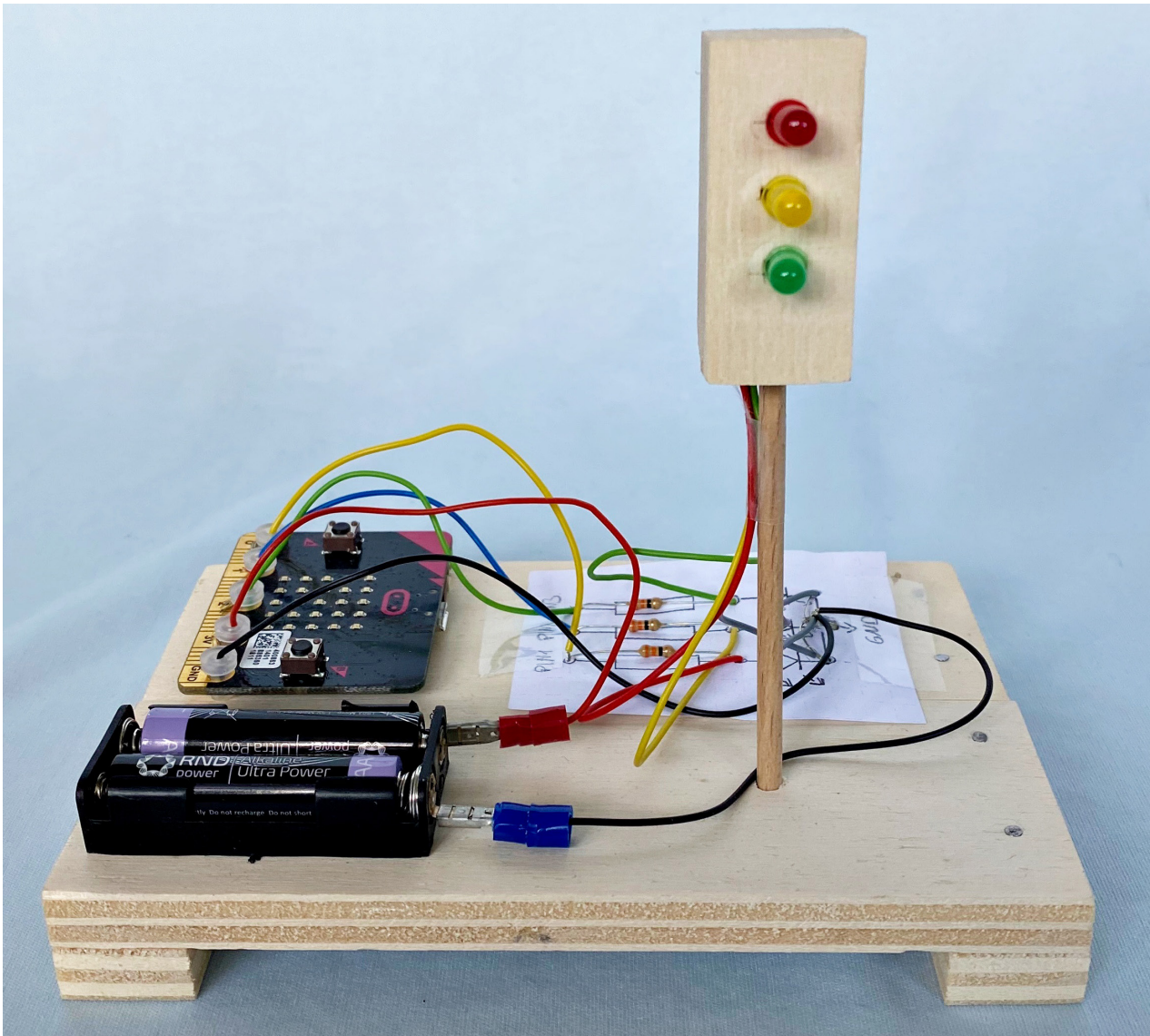


# 6. Weitere Ideen für Bauaufträge

**N**achfolgende Abbildungen zeigen mögliche Objekte, die sich mit den Grundsaltungen betreiben lassen. Hierfür gibt es

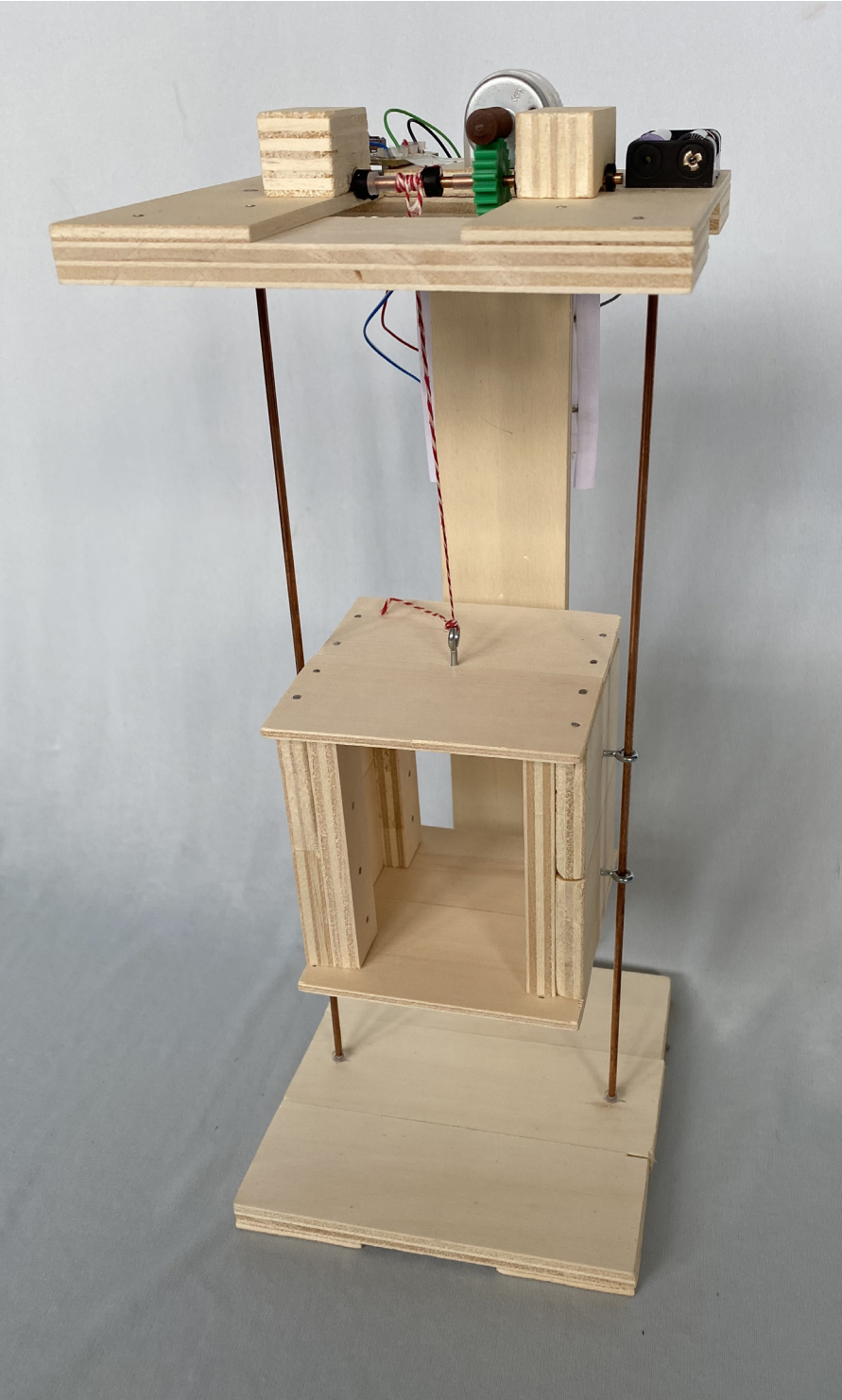
keine differenzierten Verkabelungspläne und Anleitungen. Sie dienen als Anregung.<sup>9</sup>

## 6.1 Ampel

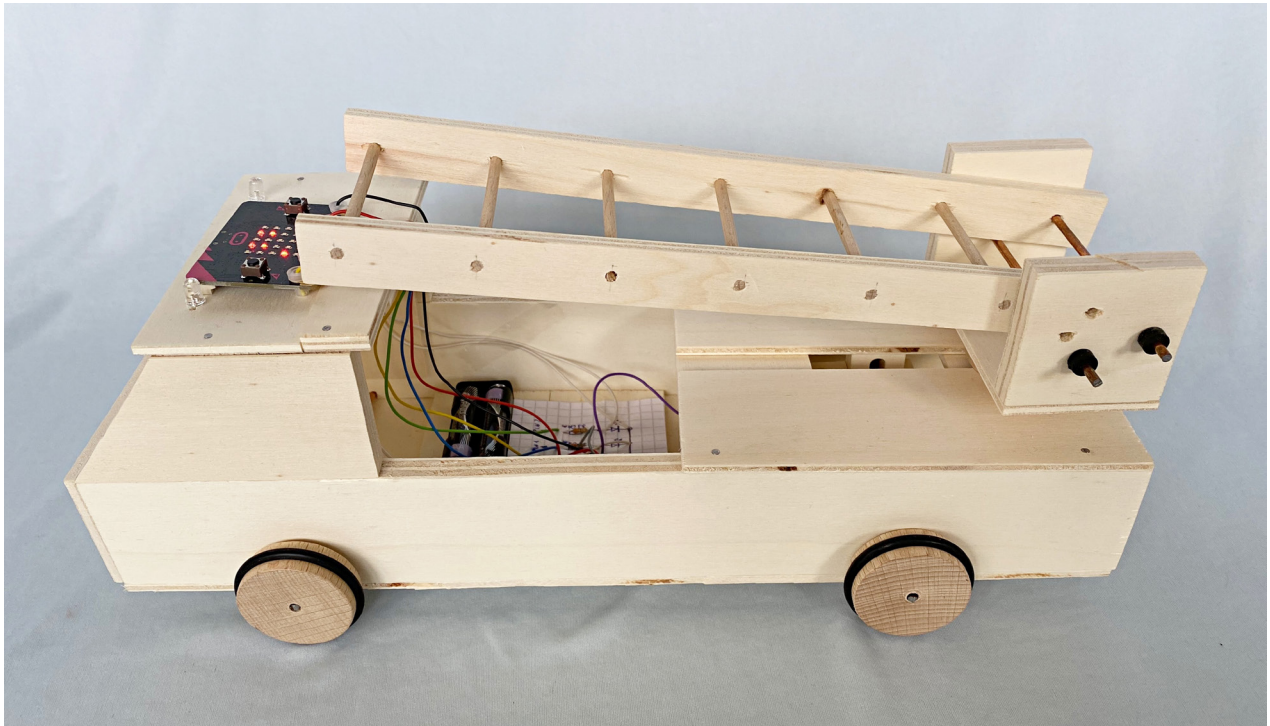


<sup>9</sup> Die Steuerung der vielfältigen Funktionen, die bei diesen Objekten möglich sind, erfordern eine vertiefte Auseinandersetzung mit dem Programmieren. Ansätze dafür finden sich in IT2School.

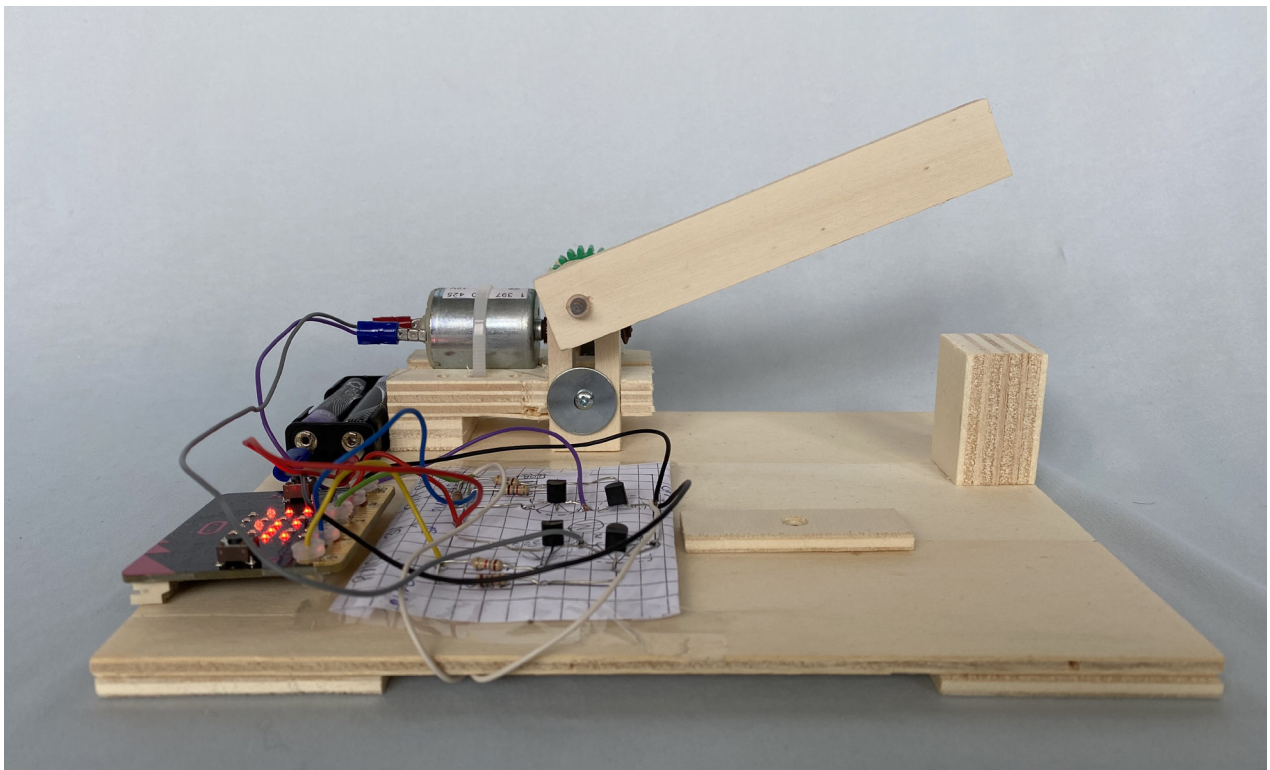
# 6.2 Aufzug



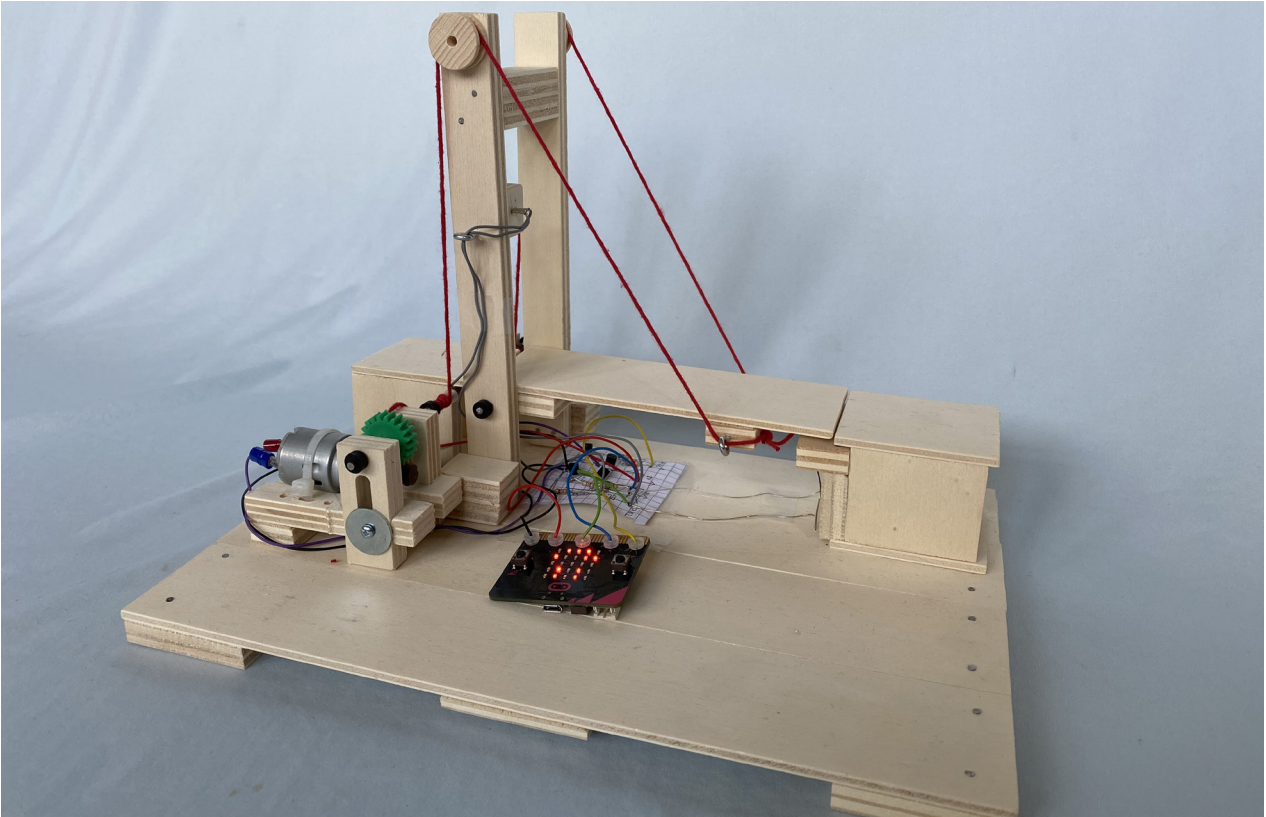
## 6.3 Feuerwehrauto



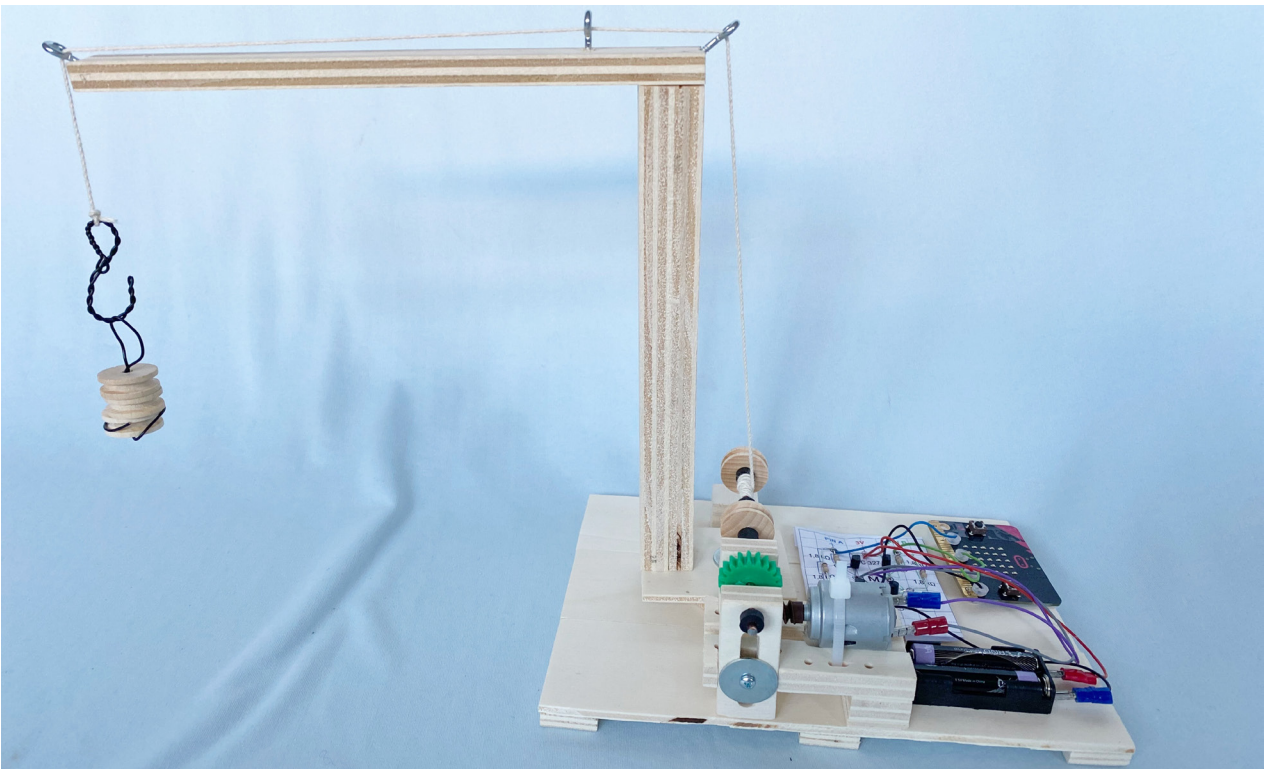
## 6.4 Schranke



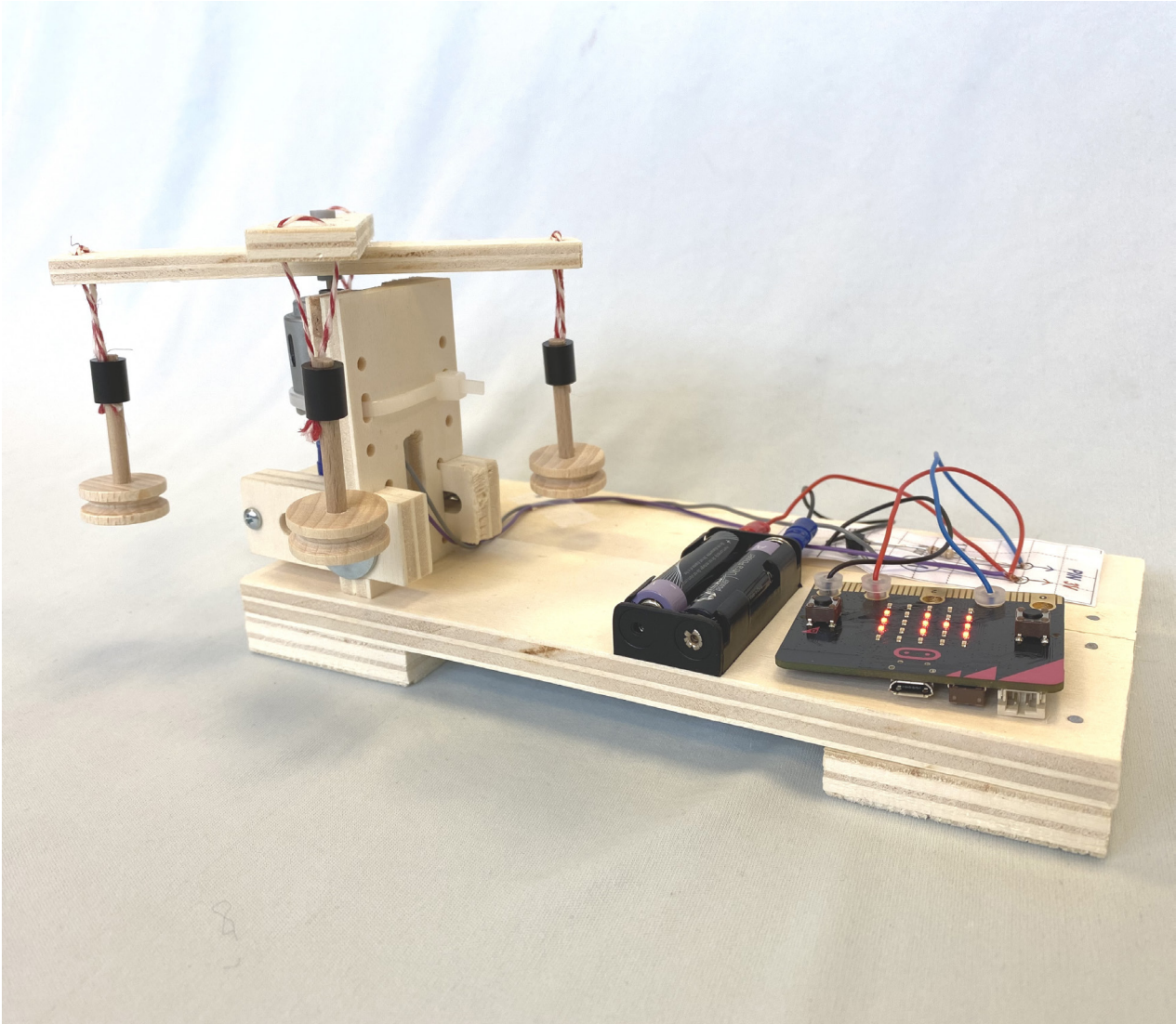
## 6.5 Zugbrücke



## 6.6 Kran



# 6.7 Karussell



# 7. Fachinformationen

In diesem Kapitel gehen wir auf folgende Themen genauer ein:

- Funktionsweisen der Grundschaltungen (Elektrotechnik)
- Einrichten der Verbindung von Computer und Mikrocontroller (Programmierungsumgebung)
- Programmieren (Mikrocontroller)
- Fehleranalyse

Je nach Kenntnisstand und Vorerfahrungen der Schüler\*innen können einzelne Abschnitte auch im Unterricht bearbeitet werden.

## 7.1 Elektrotechnik

### Stromkreis

Das Grundprinzip der Schaltungen zur Verbindung von Mikrocontroller und einem Aktor/Sensor (siehe Kapitel 7.4) ist der einfache Stromkreis. Werden mehrere Aktoren/Sensoren angeschlossen, dann entspricht das einer Parallelschaltung.

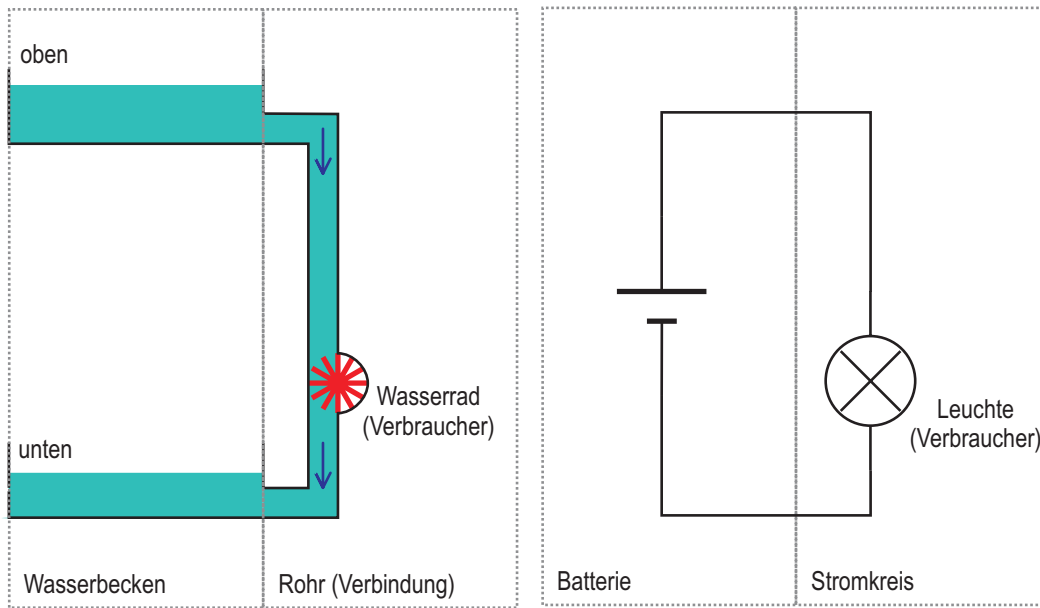
Damit ein Bauteil seine Funktion ausführen kann (z.B. leuchten), muss es einen geschlossenen Stromkreis von + (3V) nach – (GND) geben.

Aktoren können wenig oder viel Strom benötigen. Deshalb muss der Stromkreis entsprechend angepasst werden.

Zur allgemeinen Erklärung werden wir nachfolgend wie in KiTec das Wasser-Strom-Modell als Analogie heranziehen. Weitere Hinweise und Hintergrundwissen zum Stromkreis finden Sie auch im KiTec-Handbuch auf Seite 151ff.

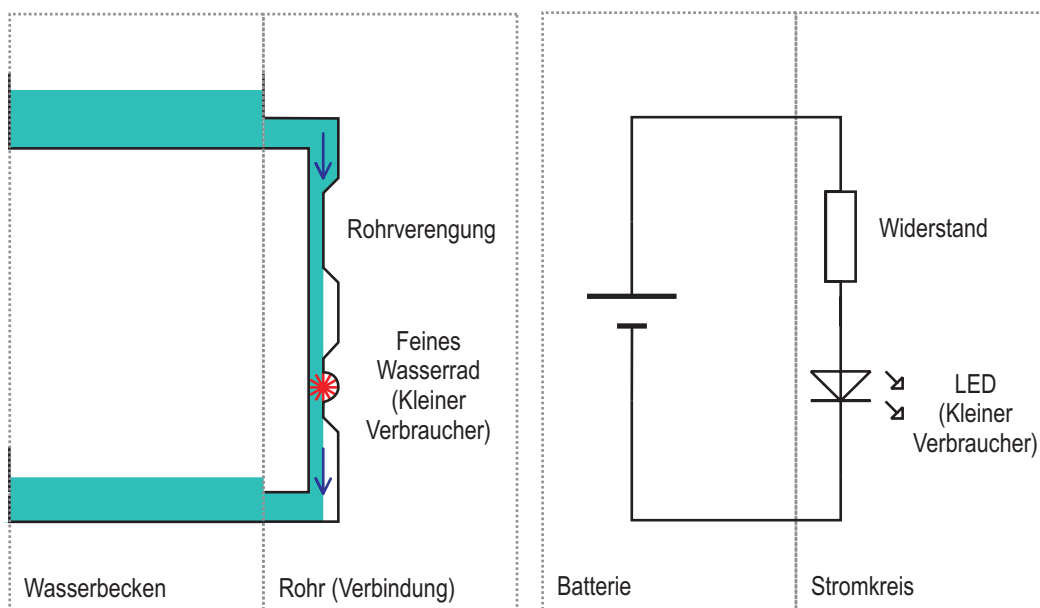
### Einfacher Stromkreis

Wenn wir einen Verbraucher betreiben wollen, muss das Wasser vom Oberbecken und das Unterbecken fließen können. Der Wasserfluss treibt das Wasserrad an. Die Spannung der Batterie entspricht dabei dem Abstand von Ober- und Unterbecken und der Strom die Wassermenge pro Zeit, die durch das Rohr strömt. Der Durchmesser des Rohres und das Wasserrad begrenzen den Strom und wirken deshalb wie ein Widerstand.



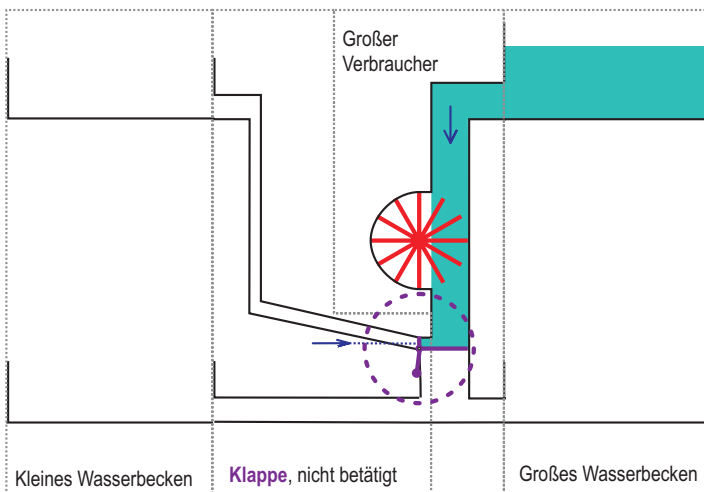
### Einfacher Stromkreis mit kleinen Verbrauchern

Kleine Verbraucher, z.B. ein Wasserrad mit sehr feinen Schaufeln, würden kaputt gehen, wenn die komplette Wassermenge durchläuft. Zum Schutz des kleinen Verbrauchers müssen wir also den Wasserstrom begrenzen, indem wir eine Rohrverengung davorsetzen. Im elektrischen Stromkreis gibt es auch kleine Verbraucher, in diesem Fall die LED. Auch diese gehen kaputt, wenn der Strom zu hoch ist (in der Analogie zum Wasser: wenn zu viel Wasser auf einmal auf die kleinen Schaufeln trifft). Das Pendant zur Rohrverengung ist im elektrischen Stromkreis der Widerstand.



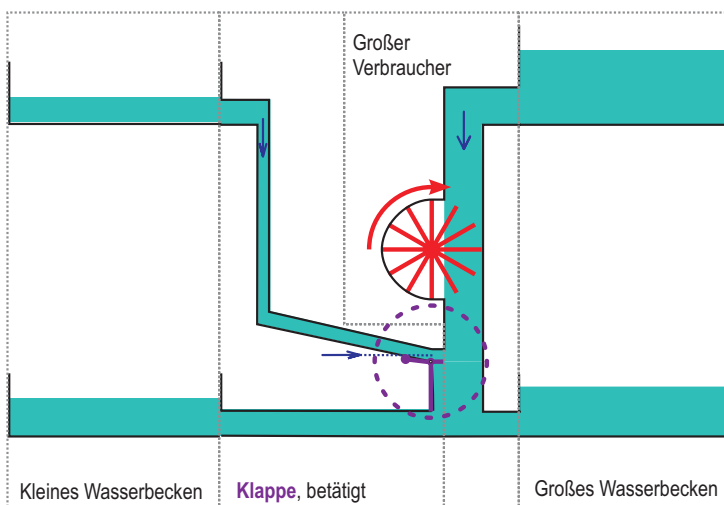
## Stromkreis mit großen Verbrauchern

Es gibt jedoch auch Verbraucher, z.B. ein Wasserrad mit sehr großen Schaufeln, die sehr viel Wasser benötigen. Dafür reicht das normale Wasserbecken nicht aus, sondern es muss noch ein weiteres leistungsfähigeres Wasserbecken bereitgestellt werden. Damit aber trotzdem mit einer kleinen Wassermenge eine größere Wirkung erzeugt werden kann, wird eine Klappe eingebaut. Die Klappe wird durch eine kleine Wassermenge aus dem kleinen Wasserbecken bewegt (Steuerstrom) und macht damit den Weg für die große Wassermenge aus dem großen Wasserbecken frei (Arbeitsstrom) – es ist also eine durch Wasser gesteuerte Klappe. Wenn kein Wasser vom kleinen Wasserbecken kommt, dann ist der Durchlass für das Wasser aus dem großen Becken gesperrt und das Wasserrad dreht sich nicht (oberes Bild, Klappe nicht betätigt). Wenn eine kleiner Menge Wasser fließt (unteres Bild, Klappe betätigt), dann ist der Durchlass für das Wasser aus dem großen Becken frei und das Wasserrad dreht sich, es kann arbeiten.



*Klappe nicht betätigt:*

→ *Das große Wasserrad dreht sich nicht*



*Klappe betätigt:*

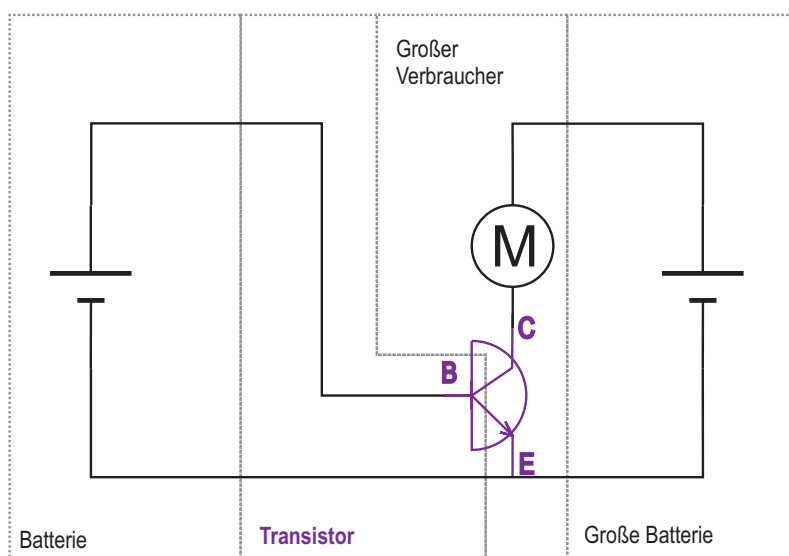
→ *Das große Wasserrad dreht sich*



Übertragen auf den Stromkreis heißt das:

Es gibt zwei Stromkreise: der eine steuert den Taster (Klappe beim Wasser), der andere liefert Strom für den Verbraucher (Wasserrad).

In der Elektrotechnik übernimmt der Transistor die Funktion des Tasters. Die drei Anschlussdrähte des Transistors E, B und C haben dabei unterschiedliche Aufgaben: Die Basis (B) ist der Eingang für den kleinen Strom (der in diesem Fall vom Mikrocontroller kommt). Dieser macht den Weg frei für den großen Strom, der vom Collector (C) zum Emitter (E) fließt. Der Stromkreis für den kleinen Strom heißt Steuerstromkreis und der Stromkreis mit dem großen Verbraucher (z.B. einem Motor) heißt Arbeitsstromkreis.



*Steuerstromkreis*

*Arbeitsstromkreis*

### Einfache Transistorschaltung

Mikrocontroller können selbst nur kleine Ströme bereit stellen. Was heißt das für unsere Situation, wenn auch größere Verbraucher gesteuert werden sollen? Die Lösung dafür bieten uns Transistorschaltungen.

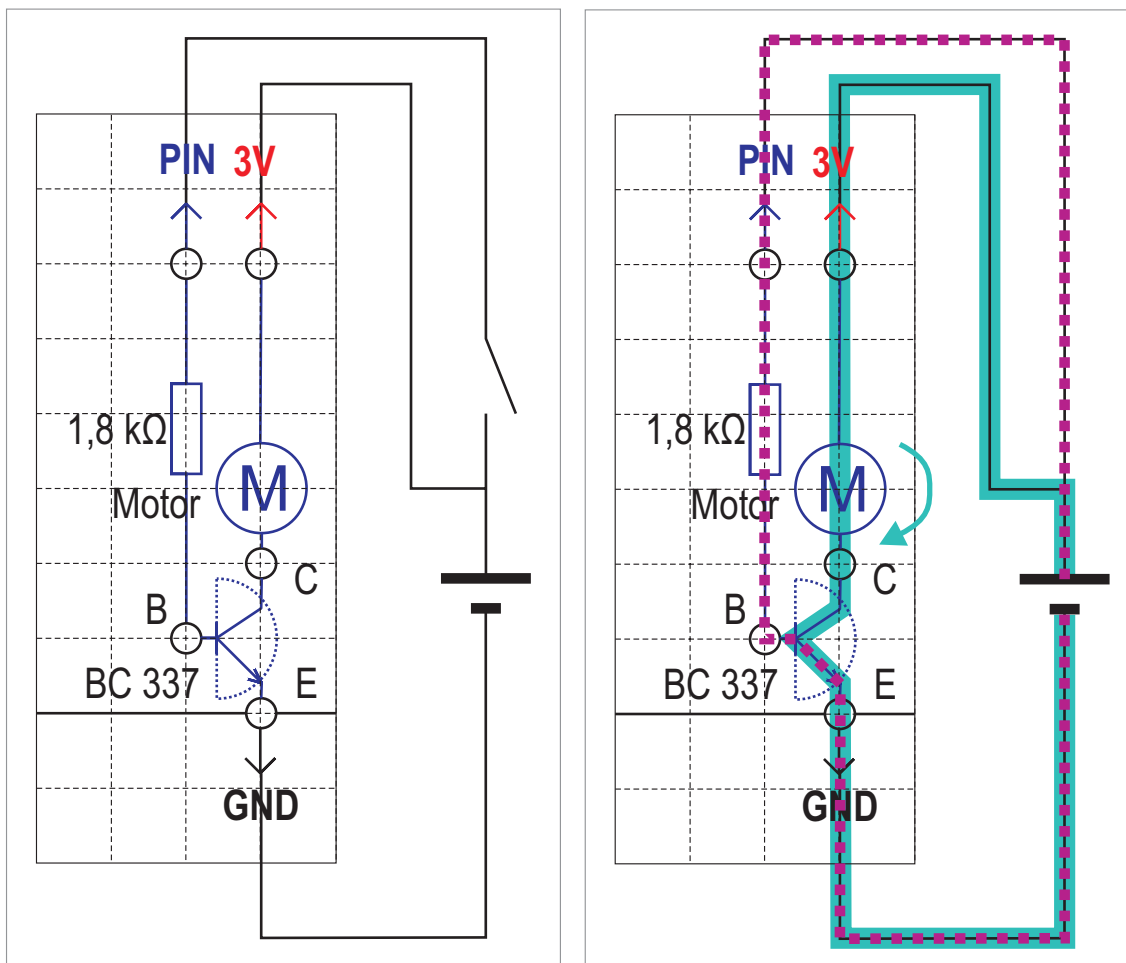
Ein zusätzlicher mechanischer Taster betätigt den Steuerstromkreis. Diese Rolle übernimmt später der Mikrocontroller.

Ist der mechanische Taster offen, dann erhält die Basis (B) des npn-Transistors BC337 keinen Strom (linke Abbildung).

Ist der mechanische Taster geschlossen, dann erhält die Basis (B) des Transistors Strom (rechte Abbildung). Damit wird nun der Transistor geschlossen und ermöglicht einen Stromdurchfluss von C nach E. Der Stromkreis mit dem Motor ist geschlossen (hellblauer Stromkreis). Während über den Steuerstromkreis (magenta, gestrichelt) die Information vermittelt wird, ist der Arbeitsstromkreis (hellblau, durchgezogen) für die Bereitstellung der Energie zuständig.

Die Übernahme des mechanischen Tasters durch den Mikrocontroller bedeutet: bei offenem Taster wird am Pin GND (keine Spannung) und bei geschlossenem Taster wird am Pin 3V (Spannung) bereit gestellt.




Alle Verbraucher (z.B. Glühlampen, Elektromotoren...), die einen höheren Strom als 15 mA benötigen, müssen auf diese Weise angeschlossen werden.



## Transistorschaltung als H-Brücke

Soll sich ein Motor in beide Richtungen drehen, werden zwei unterschiedliche Transistortypen benötigt: pnp-Transistor BC 327 (T1 und T2, siehe Abbildung) und npn-Transistor BC 337 (T3 und T4, siehe Abbildung). Der Strom kann dadurch in unterschiedliche Richtungen fließen. Damit wird beeinflusst, ob der Motor sich rechts oder links herum dreht. Diese Schaltung (Anordnung dieser Transistoren) nennt sich auch H-Brücke. Wird diese Schaltung auf das Auto mit Getriebe montiert, kann das Auto sowohl vorwärts als auch rückwärts fahren.

Zur Steuerung der Drehrichtung des Motors werden zwei Pins benötigt. Während im Handbetrieb die Pins per Hand geschaltet werden (Verbindung an GND / - oder 3V / +), übernimmt das der Mikrocontroller durch ein Programm. Je nachdem, wo die Pins angeschlossen werden, entsteht ein Rechtslauf, ein Linkslauf oder der Motor steht still:

Verhalten des Motors (Arbeitsstromkreis  )	PIN A (Steuerstromkreis  )		PIN B (Steuerstromkreis  )	
	per Hand	Mikro- controller	per Hand	Mikro- controller
Steht still	an GND / -	0	an GND / -	0
Rechtslauf	an GND / -	0	an 3V / +	1
Linkslauf	an 3V / +	1	an GND / -	0
Steht still	an 3V / +	1	an 3V / +	1

## Steuerung des Motor-Rechtslaufes

Bei der Steuerung des Motor-Rechtslaufes ist PIN A an GND / - und PIN B an 3V / + angeschlossen. Der Steuerstromkreis von PIN A (magenta, gepunktet) schließt den Transistor T1, während der Steuerstromkreis von PIN B (magenta, gestrichelt) den Transistor T3 schließt. Der Arbeitsstrom (hellblau, durchgezogen) fließt von 3V / + über den Transistor T1, durch den Motor von links nach rechts und anschließend über den Transistor T3 nach GND / -.

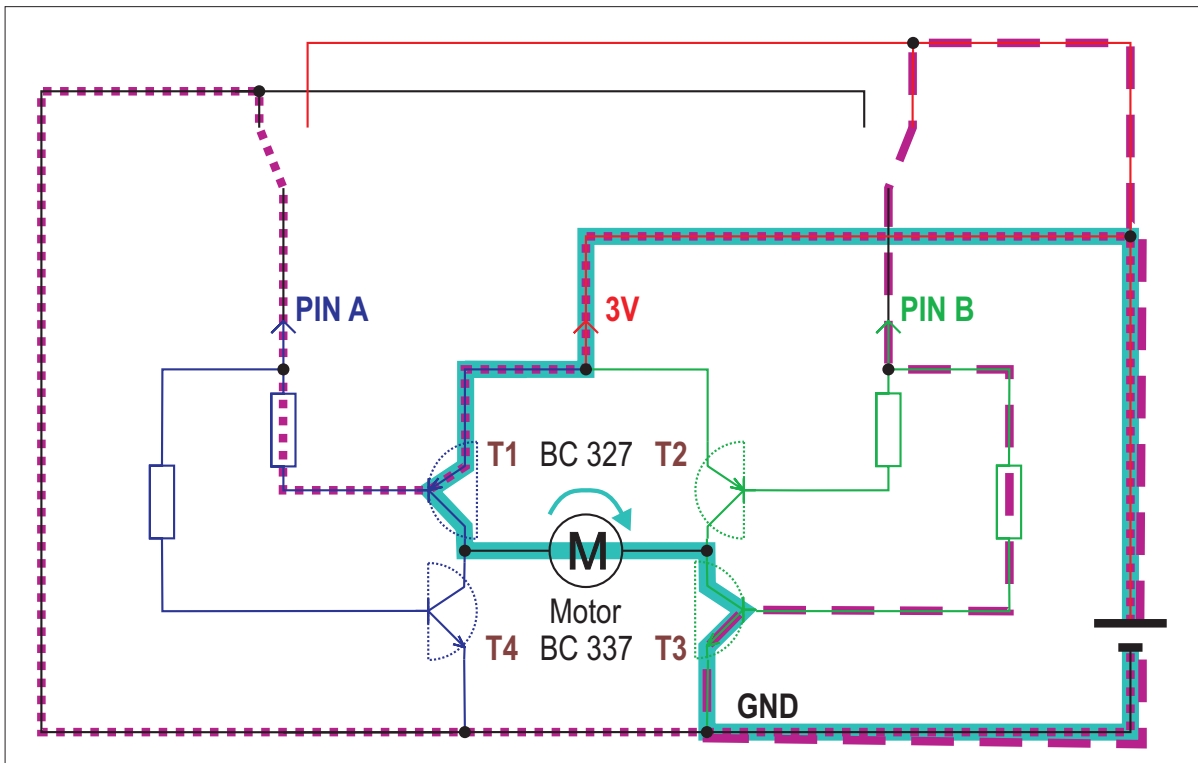


Abbildung Motor Rechtslauf

## Steuerung des Motor-Linkslaufes

Bei der Steuerung des Motor-Linkslaufes ist PIN A an 3V / + und PIN B an GND / - angeschlossen. Der Steuerstromkreis von PIN A (magenta, gepunktet) schließt den Transistor T4, während der Steuerstromkreis von PIN B (magenta, gestrichelt) den Transistor T2 schließt. Der Arbeitsstrom (hellblau, durchgezogen) fließt von 3V / + über den Transistor T2, durch den Motor von rechts nach links und anschließend über den Transistor T4 nach GND / -.

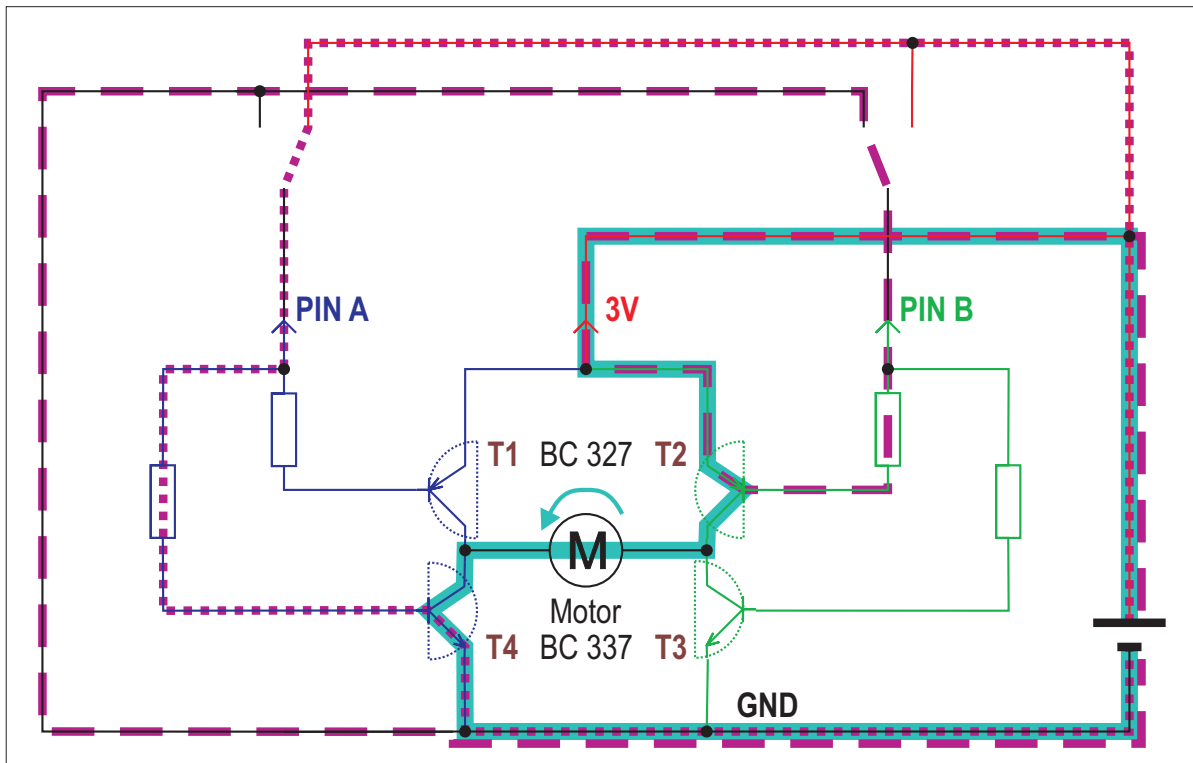


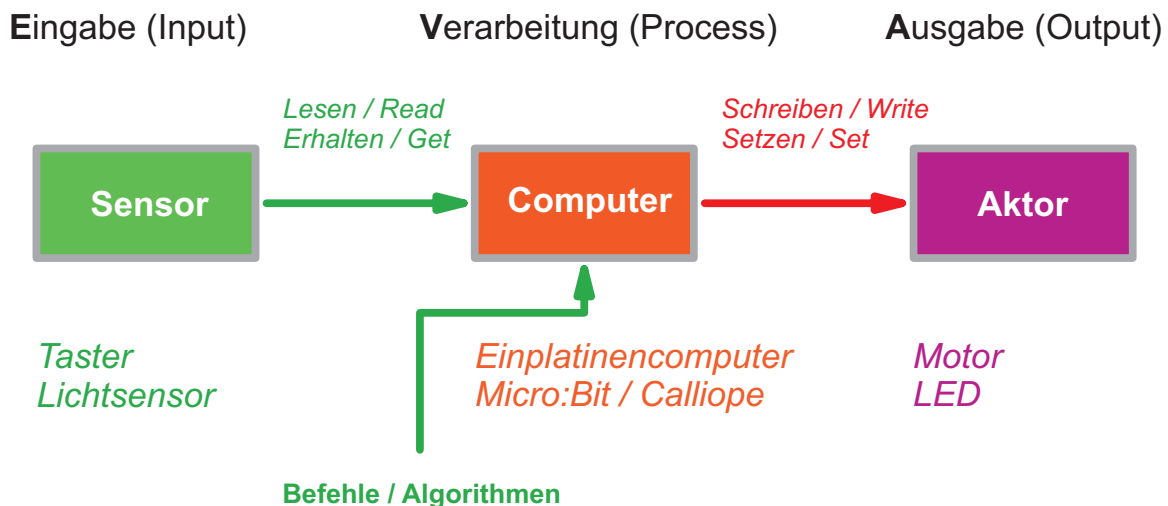
Abbildung Motor Linkslauf

## Motor im Stillstand

Sind jedoch beide Pins jeweils nur an einem Pol angeschlossen, also entweder an 3 V / + oder an GND / -, gibt es keinen geschlossenen Stromkreis durch den Motor. Der Motor dreht sich nicht.

## 7.2 Aufbau des Mikrocontrollers

Die Mikrocontroller arbeiten nach einer bestimmten Struktur: Es beginnt mit einer Eingabe, diese wird verarbeitet und es erfolgt eine Ausgabe in Form einer Aktion.



### Eingabe

Der Input eines Computers passiert über Sensoren. Dazu gehören die auf dem Mikrocontroller bereits eingebauten Taster A und B sowie die Pins. Intern verfügt der Mikrocontroller bereits über Sensoren für die Lage, Beschleunigung, Lichtstärke und Temperatur. Weitere Sensoren wären möglich, die sich dann an den Pins anschließen lassen. In KiTec Digital beschränken sich die Bauaufträge auf die Betätigung der Taster A oder B.

### Verarbeitung

Das wesentlichste Element der Verarbeitung ist der Algorithmus. Der Algorithmus entspricht einer eindeutigen Handlungsvorschrift zur Lösung eines Problems. Das Ziel ist einen Handlungsablauf in einen computergerechten Algorithmus umzuwandeln. Dieser muss extrem kleinschrittig und eindeutig sein, da der Computer nur in der Darstellung in Form von binären Zuständen - „an“ oder „aus“ oder „ja“ oder „nein“ - arbeiten kann. Je eindeutiger sie formuliert werden, desto leichter sind sie im Programm umsetzbar. Dazu ein einfaches Beispiel:

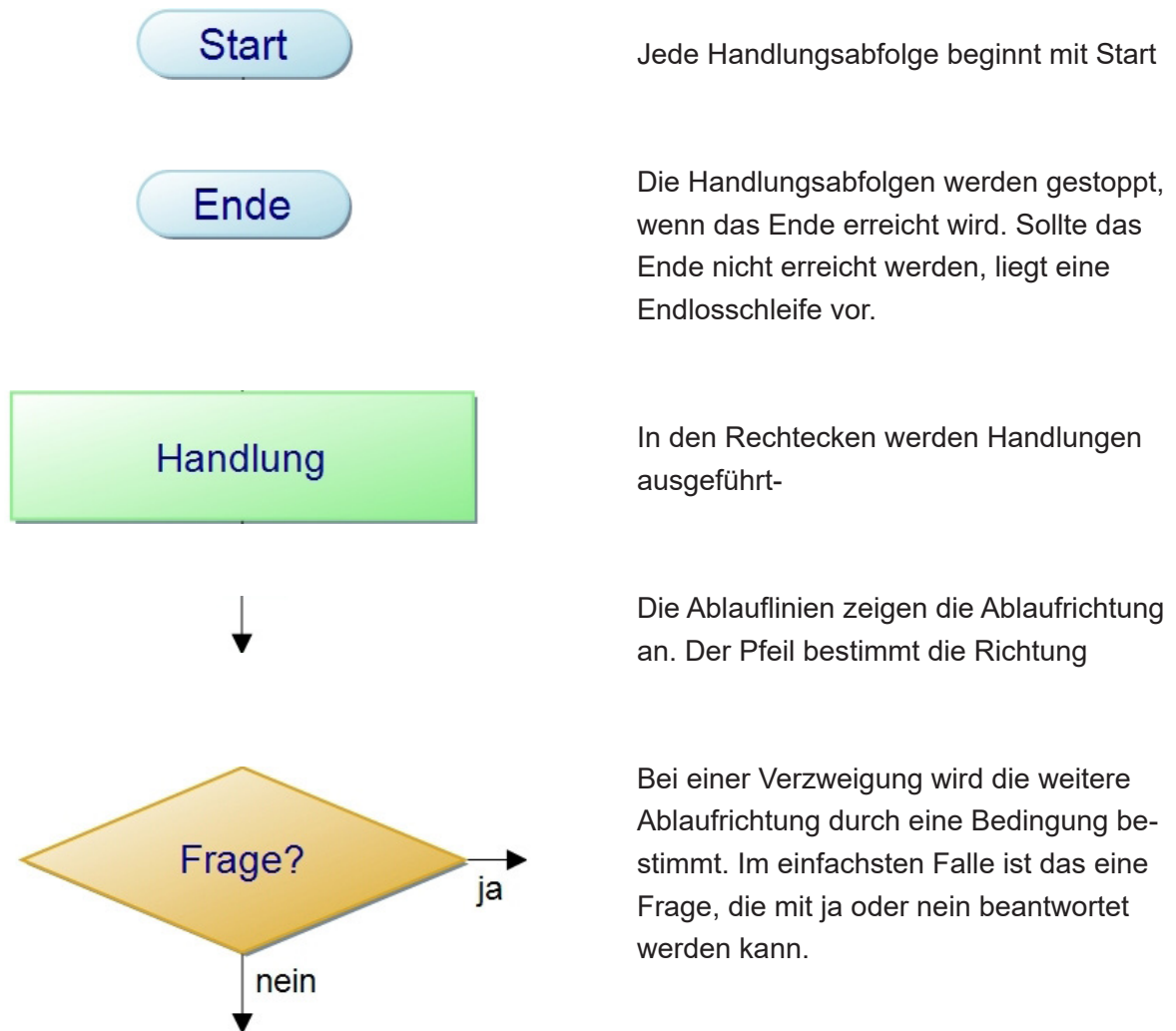
Mit KiTec wird ein Leuchtturm gebaut, auf dessen Spitze sich eine Leuchte befindet. Diese soll insgesamt 3x aufleuchten und dann wieder dunkel sein.

Die Handlung ist von Hand relativ schnell durchgeführt, es wird 3x ein Taster gedrückt und dann der Vorgang beendet.

Für den Computer muss die Handlung aber sehr viel kleinschrittiger und genauer beschrieben werden.

Solche Handlungsabfolgen können durch eine Aufzählung dargestellt werden. Grafisch einfacher und leichter nachzuvollziehen sind jedoch Programmablaufpläne – vor allem dann, wenn auch Entscheidungen getroffen werden müssen (Schleifen). In anderen Bereichen spricht man von Flußdiagramm.

Programmablaufpläne enthalten im Wesentlichen folgende Elemente:



Jede Handlungsabfolge beginnt mit Start


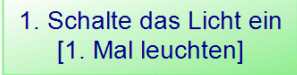
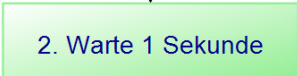
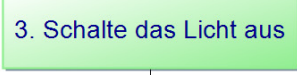
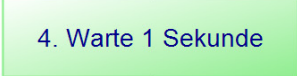
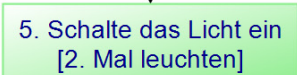
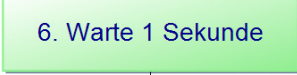
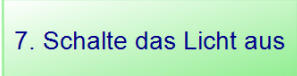

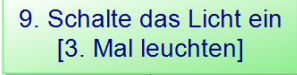

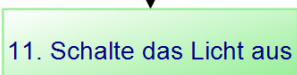

Die Handlungsabfolgen werden gestoppt, wenn das Ende erreicht wird. Sollte das Ende nicht erreicht werden, liegt eine Endlosschleife vor.

In den Rechtecken werden Handlungen ausgeführt-

Die Ablauflinien zeigen die Ablaufrichtung an. Der Pfeil bestimmt die Richtung

Bei einer Verzweigung wird die weitere Ablaufrichtung durch eine Bedingung bestimmt. Im einfachsten Falle ist das eine Frage, die mit ja oder nein beantwortet werden kann.

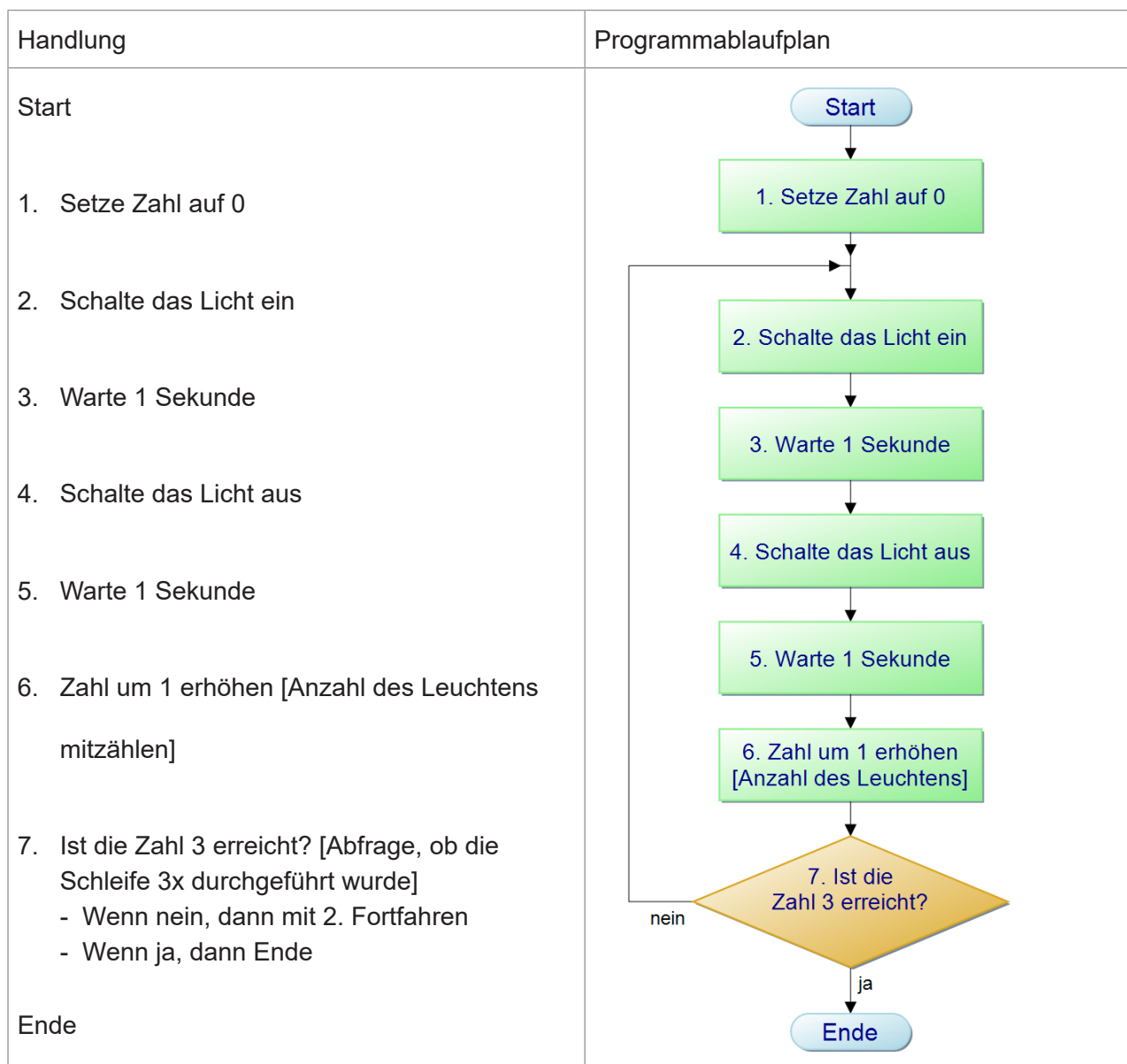
## Programmablauf für das Beispiel Leuchte auf Leuchtturm steuern

Handlung	Programmablaufplan
Start	
1. Schalte das Licht ein [1. Mal leuchten]	
2. Warte 1 Sekunde	
3. Schalte das Licht aus	
4. Warte 1 Sekunde	
5. Schalte das Licht ein [2. Mal leuchten]	
6. Warte 1 Sekunde	
7. Schalte das Licht aus	
8. Warte 1 Sekunde	
9. Schalte das Licht ein [3. Mal leuchten]	
10. Warte 1 Sekunde	
11. Schalte das Licht aus	
Ende	



Dies führt zu einem funktionsfähigen Programm, aber es fällt auf, dass die Schritte 1 bis 4 sich insgesamt 3x wiederholen. Es ist also naheliegend, dies zu vereinfachen. Um aber die Gewissheit zu haben, dass es nicht unendlich oft wiederholt wird, muss mitgezählt werden (siehe 6. In Abbildung).

Die Anzahl der Schleifen wird in der Variablen Zahl gespeichert und übernimmt die Funktion eines Zählers.



### Ausgabe

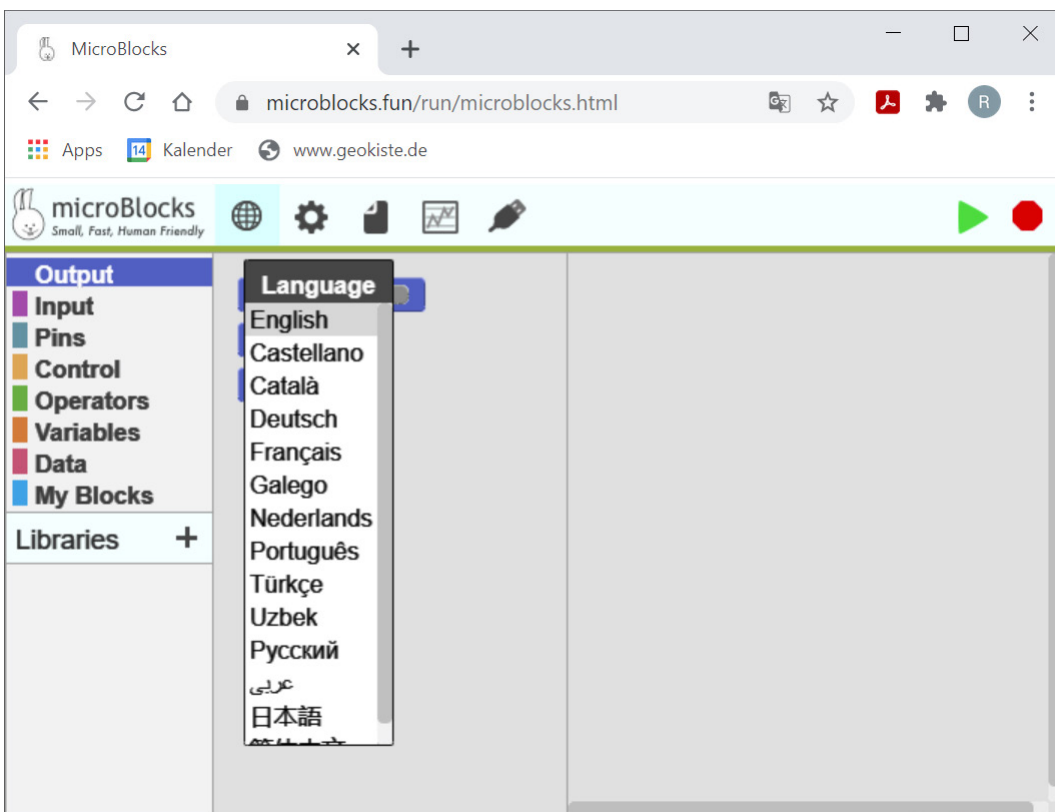
Der Output des Computer erfolgt über Aktoren. Das LED-Feld und der Lautsprecher (nur bei Micro:Bit V2 oder Calliope) sind bereits auf im Mikrocontroller eingebaut. Die ebenso auf dem Mikrocontroller befindlichen Pins lassen sich als Ausgänge schalten, an denen sich Aktoren anschließen lassen. Bei Kitec Digital sind es LED und Motoren.

## 7.3 Einrichtung der Programmierumgebung

Für das Erstellen und Übertragen des Programmes verwenden wir bei KiTec Digital die Programmierumgebung MicroBlocks. Sie funktioniert ähnlich wie der Calliope mini Editor, Open Roberta Lab oder Snap!<sup>10</sup>

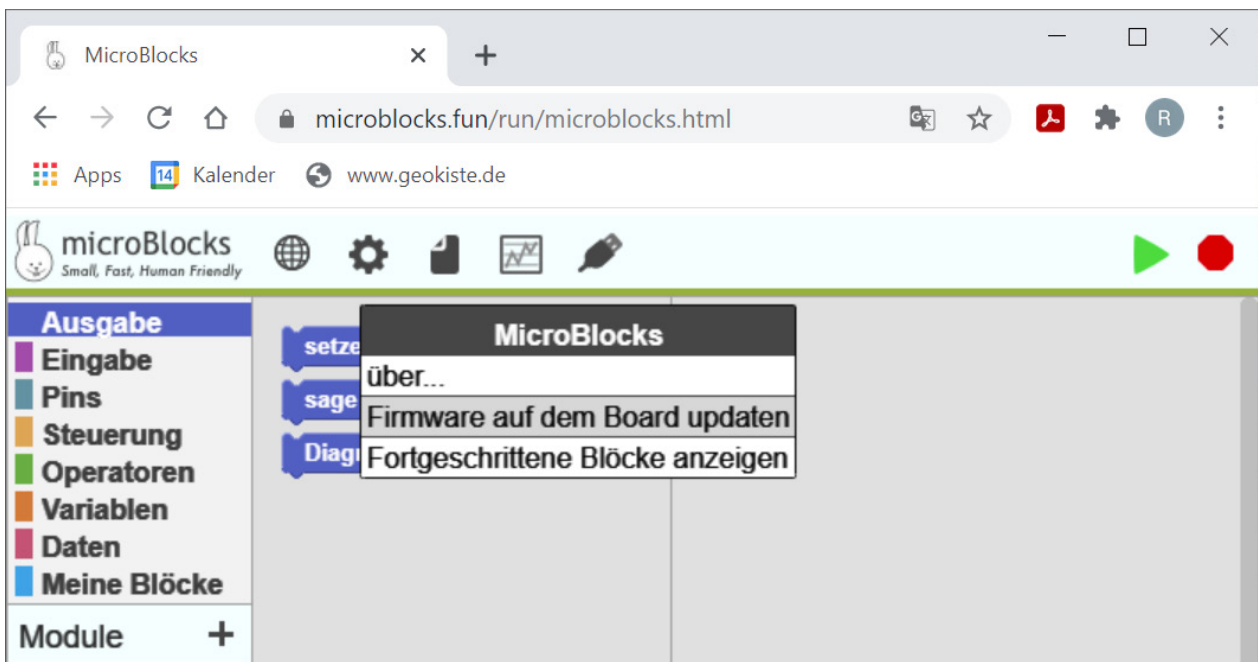
### Ablauf:

1. Anschluss des Mikrocontroller durch USB-Kabel an den Rechner
2. Aufruf der Programmierumgebung MicroBlocks:  
<https://microblocks.fun/run/microblocks.html> im Browser Chrome oder Edge oder als Desktopversion herunterladen: <https://microblocks.fun/download>
3. Einstellung der Sprache deutsch

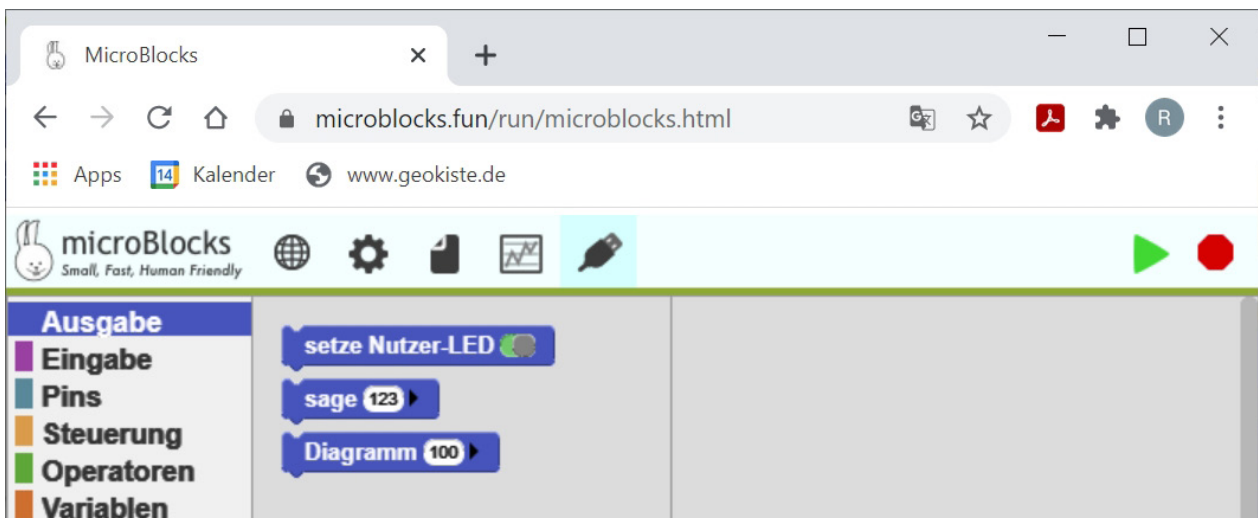


4. Firmware updaten und entsprechenden Mikrocontroller auswählen; im Explorer das Laufwerk „Micro:Bit / Micro:Bit V2 / Calliope“ auswählen und speichern drücken.

<sup>10</sup> MicroBlocks überträgt automatisch die kompilierte Datei an den Mikrocontroller. Bei anderen Programmen muss die Datei vom Programmierer explizit vom Laptop auf den Mikrocontroller übertragen werden.



## 5. Mikrocontroller verbinden



Die Verbindung ist erfolgreich, wenn der USB-Stecker mit einem grünen Kreis hinterlegt ist.



6. Programmieren mit MicroBlocks: In der Modul-Leiste befinden sich verschiedene Komponenten, durch Anklicken werden die Befehle im Block sichtbar. Der gewünschte Befehl wird per drag & drop in das Programm-Skript übertragen.



**Modul-Leiste      Block                                  Programm-Skript**

Durch Klicken auf den grünen Pfeil rechts oben in der Leiste (→ Programm startet)

Durch Klicken auf den roten Punkt rechts oben in der Leiste (→ Programm stoppt)

### Module aus der Modul-Leiste

Modul	Funktionen
Pins	Pin 0, 1 oder 2 schalten („setze digitalen Pin .. auf grün (=ein) / rot (=aus)“ („setze Pin .. auf 1023“ [Werte von 0 bis 1023 möglich])
Steuerung	Programm starten („Wenn Gestartet“) Schleifen mit festgelegter Wiederholungszahl („Wiederhole“) Dauerschleife („Fortlaufend“) Pausen („Warte ... Millisekunden“ [1000 Millisekunden = 1 Sekunde])
Operatoren	Rechnen (+, - x [mal], [geteilt]/ Vergleichen (< [kleiner], = [gleich] > [größer])
Module +	Weitere Module holen z.B: LED Anzeige
LED Anzeige	LEDs auf dem LED-Feld anzeigen (Display) alle LEDs löschen (lösche Display)
Scrollen	Längeren Text in Laufschrift anzeigen („Scrolle Text“)

Unter <https://microblocks.fun/learn> gibt es weitere Dokumente, mit denen man sich in Microblocks einarbeiten kann.

Weitere Editoren für den Calliope sind im Modul B8 Calliope von IT2School beschrieben: <https://www.wissensfabrik.de/downloadmaterial-it2school/#modul-b8-calliope>

## 7.4 Programmieren

Die Schaltungen sind auch ohne Mikrocontroller funktionsfähig. Durch den Anschluss des Mikrocontrollers können jedoch die Aktoren (LED, Motoren) viel flexibler und variantenreicher als per Hand angesteuert werden.

Dafür erstellen wir eine Handlungsanweisung, die wir als Programmablaufplan aufschreiben. Die Übersetzung in ein Programm-Skript führen wir dann wieder in MicroBlocks durch.

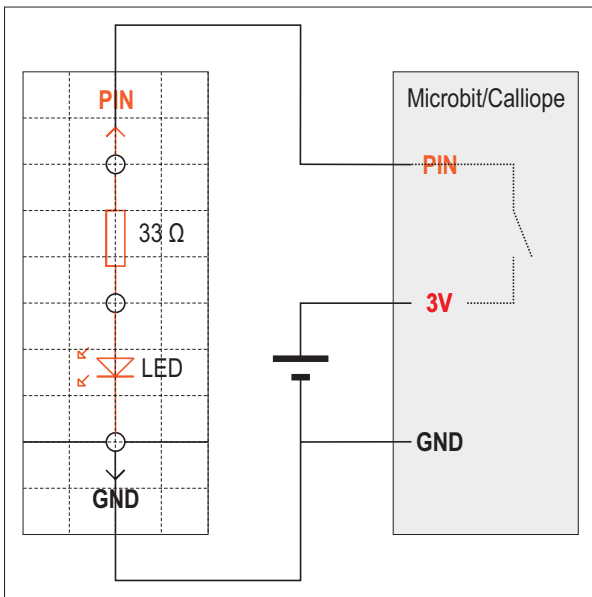
### Schalten eines einfachen Aktors

Der Aktor (z.B. eine LED) wird über einen Pin (z.B. Pin 0) am Mikrocontroller (Micro:Bit<sup>11</sup> oder Calliope<sup>12</sup>) angeschlossen. Im Programm müssen wir dann auch den gleichen Pin (hier Pin 0) aktivieren.

In den nachfolgenden Programmbeispielen wird gezeigt, wie der angeschlossene Pin gedimmt werden kann. Damit lässt sich die Helligkeit der LED verringern oder die Geschwindigkeit, mit der sich der Motor dreht. Die Werte können zwischen 0 und 1023 liegen.

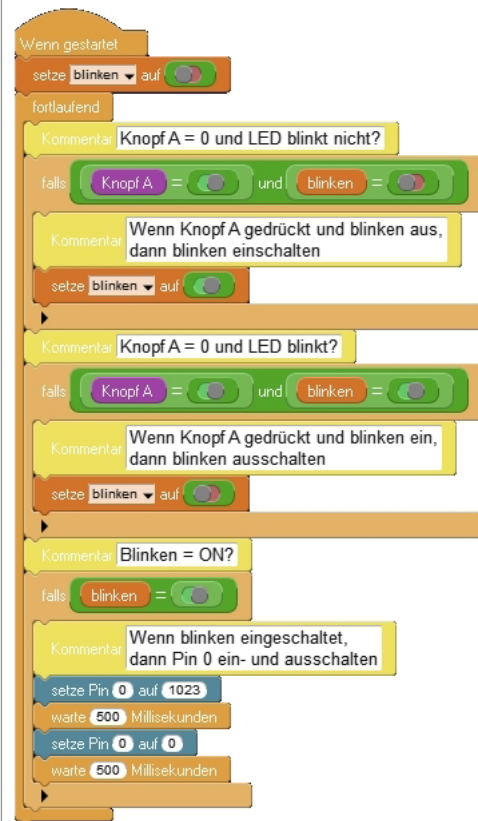
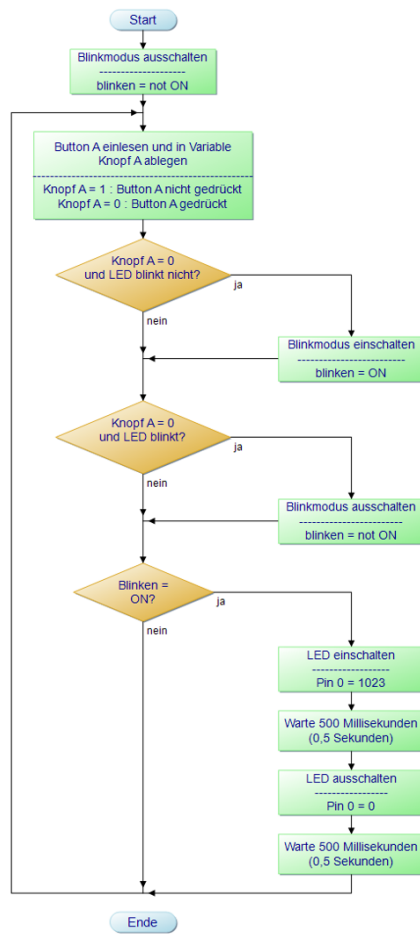
11 Wenn der Micro:Bit über ein USB-Kabel am Computer angeschlossen wird, dann kann der Anschluss der Batterie entfallen (Kabel an 3V und GND).

12 Wenn der Calliope über ein USB-Kabel am Computer angeschlossen ist, dann sollte der Anschluss der Batterie entfallen (Kabel an 3V und GND).



Funktion	Programmablaufplan	Programm-Skript in Microblocks
<p>LED mit maximaler Helligkeit für eine Zeitdauer von 2000 Millisekunden (= 2 Sekunden) einschalten</p>	<pre> graph TD     Start([Start]) --&gt; A[LED einschalten Pin 0 = 1023]     A --&gt; B[Warte 2000 Millisekunden (2 Sekunden)]     B --&gt; C[LED ausschalten Pin 0 = 0]     C --&gt; Ende([Ende])         </pre>	<pre> graph TD     Start([Wenn gestartet]) --&gt; A[setze Pin 0 auf 1023]     A --&gt; B[warte 2000 Millisekunden]     B --&gt; C[setze Pin 0 auf 0]         </pre>
<p>LED mit verminderter Helligkeit für eine Zeitdauer von 2000 Millisekunden (= 2 Sekunden) einschalten</p>	<pre> graph TD     Start([Start]) --&gt; A[LED einschalten Pin 0 = 600]     A --&gt; B[Warte 2000 Millisekunden (2 Sekunden)]     B --&gt; C[LED ausschalten Pin 0 = 0]     C --&gt; Ende([Ende])         </pre>	<pre> graph TD     Start([Wenn gestartet]) --&gt; A[setze Pin 0 auf 600]     A --&gt; B[warte 2000 Millisekunden]     B --&gt; C[setze Pin 0 auf 0]         </pre>

LED blinken  
jeweils durch  
Drücken des  
Knopfes A ein-  
und ausschalten



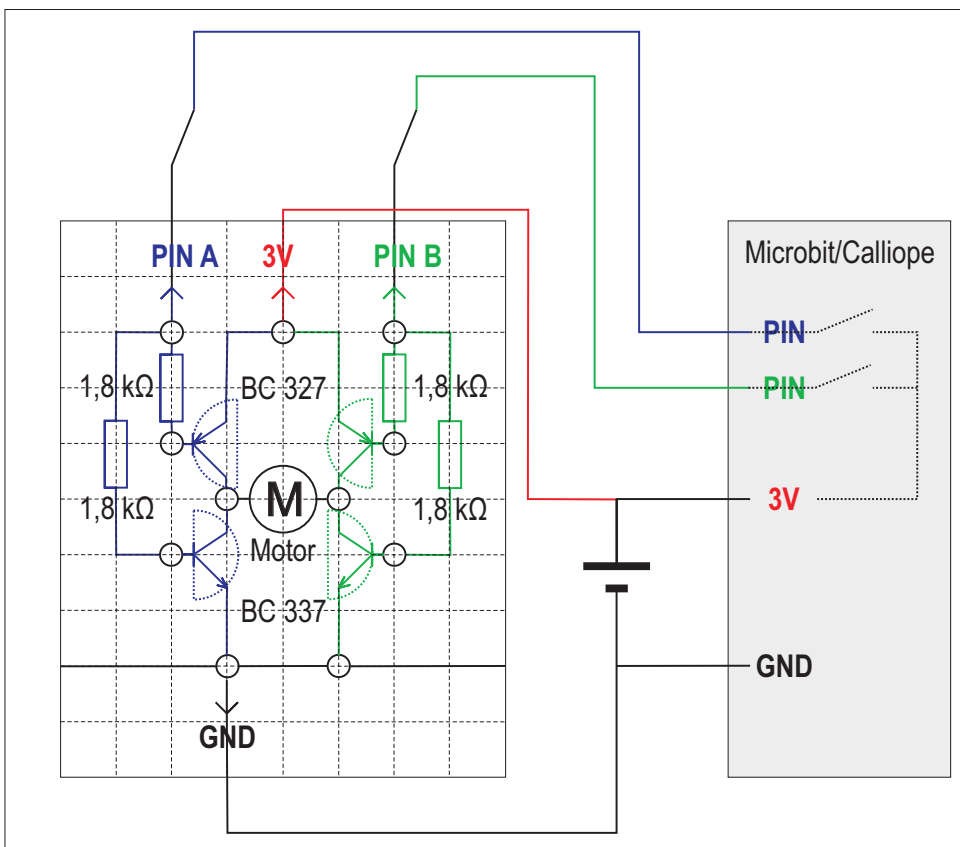
Die gelben Blöcke sind Kommentare ohne Funktion und dienen der Erläuterung.

## Schalten eines doppelt angeschlossenen Aktors

Bei einem Motor werden zwei Pins benötigt, wenn der Motor in verschiedene Richtungen laufen soll. Dabei haben die Anschlüsse folgende Bedeutungen:

Verhalten des Motors	<b>PIN A</b> (Pin 1)	<b>PIN B</b> (Pin 2)
Steht still	0	0
Rechtslauf	0	1
Linkslauf	1	0
Steht still	1	1

In nachfolgenden Beispielen wird die blaue Leitung mit Pin A und die grüne Leitung mit Pin B gleichgesetzt.





Funktion	Programmablaufplan	Programm-Skript in Microblocks
<p>Mit Knopf A wird der Motor für 7 Sekunden mit 50% des Maximalwerts nach vorne und mit Knopf B für 7 Sekunden mit 50% des Maximalwertes nach rückwärts bewegt.</p>	<pre> graph TD     Start([Start]) --&gt; DisplayHI[Display "H I"]     DisplayHI --&gt; MotorAusstellen["Motor ausstellen Pin 1 = 0 Pin 2 = 0"]     MotorAusstellen --&gt; KnopfA{Knopf A gedrückt?}     KnopfA -- ja --&gt; DisplayLoeschenOb["Display löschen Display "Pfeil oben""]     DisplayLoeschenOb --&gt; MotorVorwaerts["Motor vorwärts Pin 1 = 500"]     MotorVorwaerts --&gt; Warte7000["Warte 7000 Millisekunden"]     Warte7000 --&gt; MotorAus["Motor ausschalten Pin 1 = 0"]     MotorAus --&gt; DisplayLoeschenOb     KnopfA -- nein --&gt; KnopfB{Knopf B gedrückt?}     KnopfB -- ja --&gt; DisplayLoeschenUnten["Display löschen Display "Pfeil unten""]     DisplayLoeschenUnten --&gt; MotorRueckwaerts["Motor rückwärts Pin 2 = 500"]     MotorRueckwaerts --&gt; Warte7000     Warte7000 --&gt; MotorAus     MotorAus --&gt; DisplayLoeschenUnten     KnopfB -- nein --&gt; Ende([Ende])   </pre>	<pre> // Wenn gestartet Display setze Pin 1 auf 0 setze Pin 2 auf 0  // fortlaufend // Kommentar: Wenn Knopf A gedrückt, dann ... falls Knopf A = gedrückt   lösche Display   Display   setze Pin 1 auf 500   warte 7000 Millisekunden   setze Pin 1 auf 0   lösche Display // Kommentar: Wenn Knopf B gedrückt, dann ... falls Knopf B = gedrückt   lösche Display   Display   setze Pin 2 auf 500   warte 7000 Millisekunden   setze Pin 2 auf 0   lösche Display   </pre> <p>Die gelben Blöcke sind Kommentare ohne Funktion und dienen der Erläuterung.</p>

## 7.5 Fehleranalyse

Trotz guter Planung und Konstruktion kann es bei der Inbetriebnahme vorkommen, dass etwas nicht so funktioniert wie gedacht. Die Ursache für mögliche Fehler kann selbst bei kleinen Vorhaben wie in den Bauaufträgen KiTec Digital in unterschiedlichen Teilsystemen liegen: es können die mechanischen Komponenten sein, die nicht funktionieren, die elektrische Verbindung zwischen Bauwerk und Mikrocontroller kann fehlerhaft sein oder aber der Algorithmus im Mikrocontroller selbst hat Fehler. Bevor der Mikrocontroller angeschlossen wird, sollte die Schaltung einwandfrei funktionieren. Hier lohnt sich ein systematisches Vorgehen:

### 1. Spannungsquelle überprüfen

Sind die Batterien richtig in das Batteriefach eingelegt (Knopf an Knopf und Feder an flache gerade Seite)?

Sind die Batterien geladen?

### 2. Kontakte mechanisch überprüfen

Gibt es Wackelkontakte? Sind alle Aderendhülsen ordentlich verpresst?

### 3. Leitungsverbindungen prüfen

Entsprechen alle Verbindungen zu den Bauteilen, zwischen Mikrocontroller und Schaltung sowie von und zur Batterie dem Verkabelungsplan?

### 4. Bauteile überprüfen

Sind die richtigen Widerstandswerte (siehe Farbringe) verwendet worden?

Ist die LED lagerichtig eingebaut (erkennbar an „Hammer“ und „Sichel“, abgeflachte Seite)?

Ist der Transistor lagerichtig eingebaut (erkennbar an der abgeflachten Seite)?

Wenn die Schaltung selbst in Ordnung ist, kann der Mikrocontroller angeschlossen werden. Sollte der Betrieb des Gerätes immer noch nicht funktionieren, dann ist mit hoher Wahrscheinlichkeit ein Fehler im Programm.

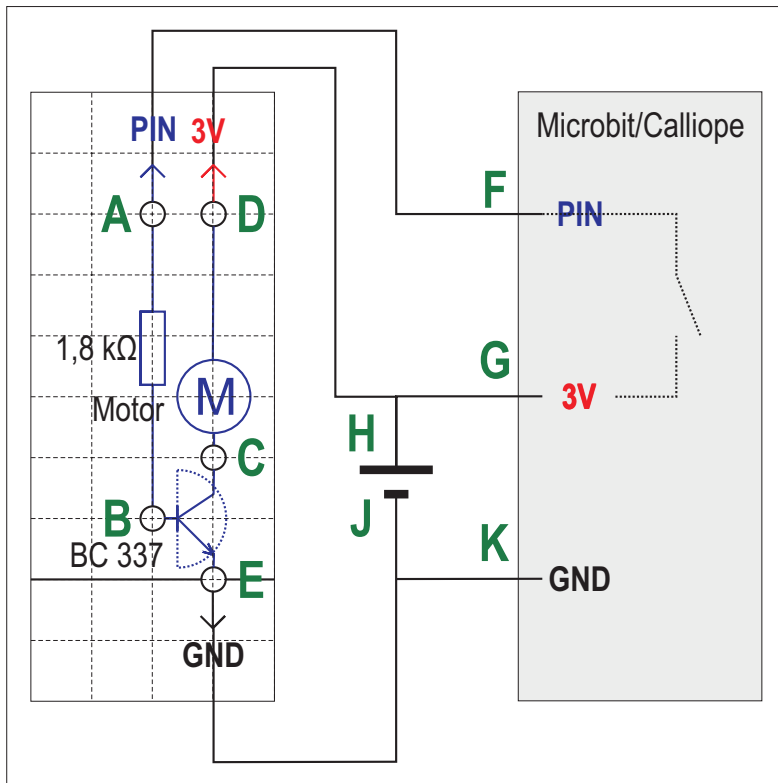
Sollte der Verdacht bestehen, dass der Mikrocontroller selbst defekt ist, dann kann man ihn durch einen anderen austauschen und dasselbe Programm aufspielen.

Unter <https://www.wissensfabrik.de/kitec-digital-faq/> haben wir die häufigsten Fehler und ihre Lösungsmöglichkeiten aufgelistet.

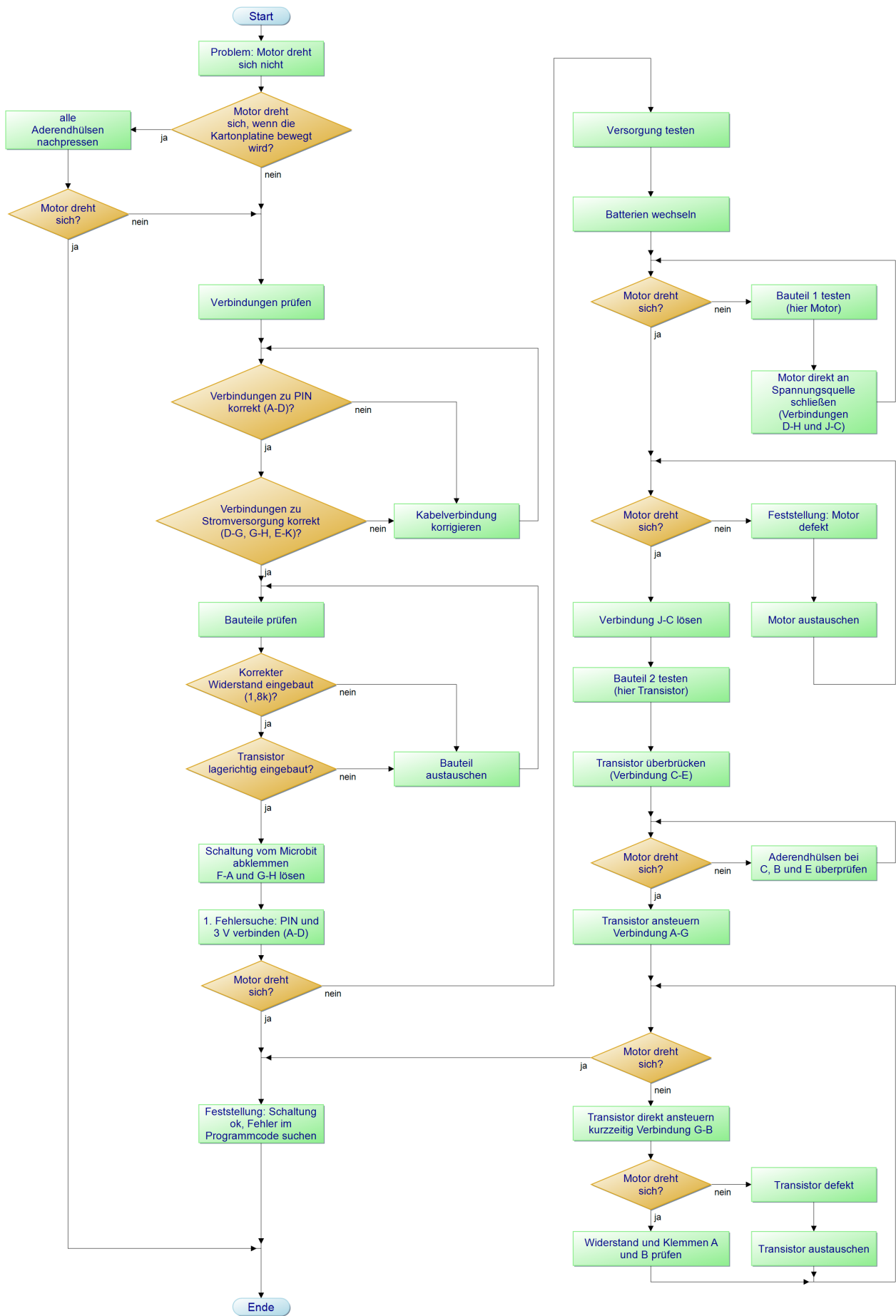


### Beispielhafte Vorgehensweise bei einer Fehlersuche

Die nachfolgende Schaltung besteht aus einem Motor, der über einen Transistor ein- und ausgeschaltet werden soll und durch einen Mikrocontroller (Calliope oder Micro:Bit) angesteuert wird. Beschreibung des Fehlers: Der Motor dreht sich nicht.



Im Verkabelungsplan sind zusätzlich jeweils in grün die Messpunkte A bis K eingetragen. Diese Messpunkte sind dazu da, die Verbindung konkret zu prüfen und einzelne Bauteile auf ihre Funktion zu testen. Auch die Fehlersuche ist ein Algorithmus. Visualisiert darstellen kann man sie in einem Programmablaufplan. Hieraus wird deutlich, dass man abschnittsweise vorgehen kann. Moderne Fehlerdiagnosesysteme arbeiten computerbasiert, die Programme haben im Grunde die gleiche Logik und Systematik wie unten im Programmablaufplan dargestellt.



# Impressum

## Herausgeber

**Wissensfabrik – Unternehmen für Deutschland e.V.**

Ruthenstraße 23  
67063 Ludwigshafen  
www.wissensfabrik.de

Katharina Missling  
Projektleitung  
E-Mail: Katharina.missling@wissensfabrik.de

Dr. Axel Jentzsch  
Leiter Bildung  
E-Mail: Axel.jentzsch@wissensfabrik.de

## Erstellung

**IngenieurDidaktik**

Technische Universität Dortmund  
Roland Hirsch  
Otto-Hahn-Str. 6  
44227 Dortmund  
www.id.mb.tu-dortmund.de  
E-Mail: roland.hirsch@tu-dortmund.de

Für ihre Unterstützung bei der Finalisierung des Handbuchs danken wir sehr herzlich:  
Petra Evanschitzky  
Diplom Sozialpädagogin, Diplom Sozialwirtin,  
Organisationsberaterin  
Mozartstr. 46  
70180 Stuttgart

Für inhaltliche Anregungen danken wir dem Arbeitskreis Bildung der Wissensfabrik – Unternehmen für Deutschland e.V. und im speziellen der Projektgruppe. Außerdem danken wir allen Lehrkräften und Unternehmensvertretern, die aktiv an der Pilotphase beteiligt waren oder uns Feedback gegeben haben:

Ralf Berger, Leibniz Gymnasium, Neustadt  
Sebastian Kirch, Werner Richard Grundschule, Herdecke  
Bianca Rüter, HARTING Stiftung & Co. KG  
Josef Baader, Boehringer Ingelheim  
Josephine Starzmann, Trumpf GmbH + Co. KG  
Olivera Haas, Storopack  
René Mühlroth, Teachcom Edutainment

## Bildquellen

Titellustration: Mathias Weber, S2/3, 68161 Mannheim  
Abbildungen und Fotos von Roland Hirsch, TU Dortmund, IngenieurDidaktik

## Gestaltung und Umsetzung

active-screen, Mannheim

## Urheberrechte



Dieses Werk bzw. Inhalt steht unter einer Creative Commons Lizenz (Namensnennung, Nicht kommerziell, Weitergabe unter gleichen Bedingungen). Alle Teile dieses Werkes sind vom Herausgeber und von der für die Erstellung verantwortlichen Redaktion sorgfältig erwogen und geprüft worden. Eine Haftung des Herausgebers bzw. der für die Redaktion verantwortlichen Institutionen für etwaige Personen-, Sach- oder Vermögensschäden, die sich aus dem Gebrauch dieses Werkes ergeben oder ergeben könnten, ist ausgeschlossen.