

dpunkt.verlag



Komplett
in Farbe!



LEGO[®]- EV3-ROBOTER

Laurens Valk

Bauen und programmieren lernen
mit LEGO[®] MINDSTORMS[®] EV3

Übersicht

Einleitung

Teil 1 Erste Schritte

- Kapitel 1 Deinen EV3-Kasten vorbereiten
- Kapitel 2 Baue deinen ersten Roboter
- Kapitel 3 Programme erstellen und ändern
- Kapitel 4 Arbeiten mit Programmierblöcken: Aktionsblöcke
- Kapitel 5 Warten, wiederholen, Eigene Blöcke und Multitasking

Teil 2 Roboter mit Sensoren programmieren

- Kapitel 6 Wie Sensoren funktionieren
- Kapitel 7 Den Farbsensor verwenden
- Kapitel 8 Den Infrarotsensor verwenden
- Kapitel 9 Die Stein-Tasten und Motorumdrehungssensoren verwenden

Teil 3 Techniken des Roboterbaus

- Kapitel 10 Mit Balken, Achsen, Verbindern und Motoren arbeiten
- Kapitel 11 Mit Zahnrädern und Getrieben arbeiten

Teil 4 Fahrzeuge und Robotertiere

- Kapitel 12 Formel EV3: Ein Rennroboter
- Kapitel 13 ANTY: Die Roboterrameise

Teil 5 Fortgeschrittene Programme erstellen

- Kapitel 14 Datenleitungen nutzen
- Kapitel 15 Datenblöcke und Eigene Blöcke mit Datenleitungen verwenden
- Kapitel 16 Konstanten und Variablen verwenden
- Kapitel 17 Spiele auf dem EV3

Teil 6 Maschinen und menschenähnliche Roboter

- Kapitel 18 Der SNATCH3R: Ein autonomer Roboterarm

Kapitel 19	LAVA R3X: Ein Maschinenmensch, der geht und spricht
Anhang A	Fehlerbehebung für Programme, den EV3-Stein und drahtlose Verbindungen
Anhang B	On-Brick-Programme erstellen
Index	

4 Arbeiten mit Programmierblöcken: Aktionsblöcke

In [Kapitel 3](#) hast du gelernt, wie ein Programm erstellt und auf den EXPLOR3R heruntergeladen wird. In diesem Kapitel erkläre ich, wie du Programmierblöcke verwendest, um lauffähige Programme zu schreiben, mit denen sich der EXPLOR3R bewegt.

Du lernst außerdem, wie der Roboter Klänge abspielt und Text oder Bilder auf dem EV3-Display anzeigt und wie du die Farblampen auf dem EV3-Stein steuerst. Nachdem du ein wenig mit den Beispielprogrammen experimentiert hast, wirst du eigenständig einige Programmieraufgaben meistern!

Wie funktionieren Programmierblöcke?

EV3-Programme bestehen aus einer Reihe von Programmierblöcken, die den Roboter eine bestimmte Aufgabe verrichten lassen, wie sich z.B. eine Sekunde lang vorwärts zu bewegen. Programme führen die Blöcke nacheinander aus und beginnen mit dem ersten ganz links. Nachdem der erste Block abgeschlossen ist, fährt das Programm mit dem zweiten fort usw. Wenn der letzte Block abgeschlossen ist, endet das Programm.

Jeder Programmierblock hat einen oder mehrere Modi und pro Modus jeweils mehrere Einstellungen. Du änderst das Verhalten eines Blocks durch Änderung des Modus und der Einstellungen. In [Abbildung 4-1](#) sind z.B. beide Blöcke Bewegungslenkungsblöcke, jedoch in unterschiedlichen Modi. Da die Blöcke verschiedene Modi und Einstellungen haben, führt der Roboter andere Aktionen aus.

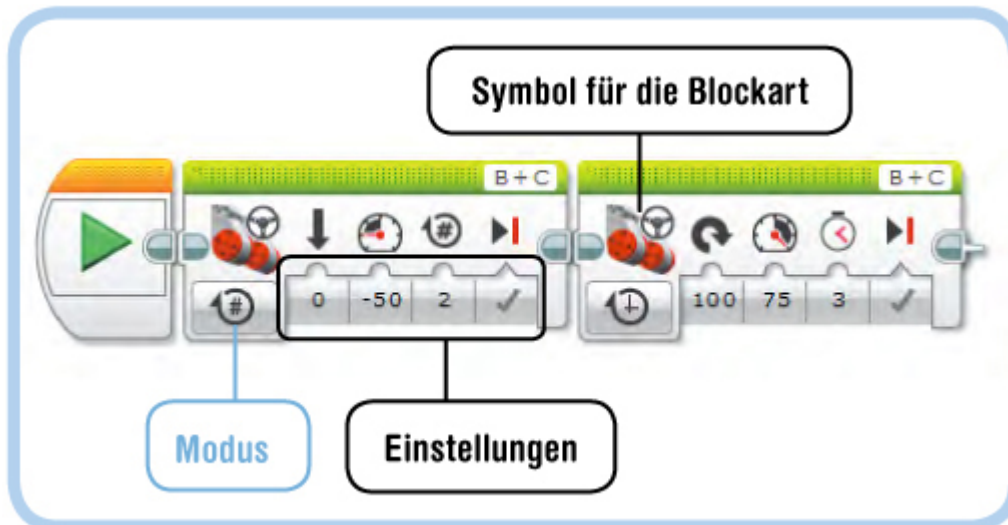


Abbildung 4-1: Um die Aktionen eines Blocks festzulegen, änderst du seinen Modus und die Einstellungen. Der erste Block lässt den Roboter z.B. rückwärts fahren und der zweite steuert ihn nach rechts. (Dieses Programm erstellst du im folgenden Abschnitt.)

Es gibt viele Arten von Programmierblöcken mit eigenen Bezeichnungen und Symbolen, sodass du sie unterscheiden kannst. Verschiedene Blöcke sind für unterschiedliche Zwecke gedacht, wobei solche mit ähnlichen Aufgaben dieselbe Farbe in der Programmierpalette haben. In diesem Kapitel stelle ich die Aktionsblöcke vor (die grünen Blöcke in der Programmierpalette).

Der Bewegungslenkungsblock

Der Bewegungslenkungsblock steuert die Bewegung der Motoren eines Roboters. Wenn du ihn in deinem Programm verwendest, kannst du den EXPLOR3R vorwärts oder rückwärts bewegen oder nach links oder rechts steuern. In [Kapitel 3](#) haben wir den Roboter mit einem Bewegungslenkungsblock nach vorn fahren lassen.

Der Bewegungslenkungsblock in Aktion

Bevor wir uns mit der Funktion des Bewegungslenkungsblocks beschäftigen, erstellen wir ein kleines Programm, um ihn in Aktion zu sehen. Dieses Programm lässt den EXPLOR3R zwei Radumdrehungen rückwärts fahren und sich dann drei Sekunden schnell nach rechts drehen. Weil es sich um zwei verschiedene Aktionen handelt, verwendest du zwei Bewegungslenkungsblöcke.

1. Erstelle ein neues Projekt namens EXPLOR3R-4. Dieses Projekt verwendest du für alle Programme in diesem Kapitel. Ändere den Namen des ersten

Programms in *Move*.

2. Wähle aus der Programmierpalette zwei Bewegungslenkungsblöcke und platziere sie im Programmierbereich, wie in [Abbildung 4-2](#) gezeigt.

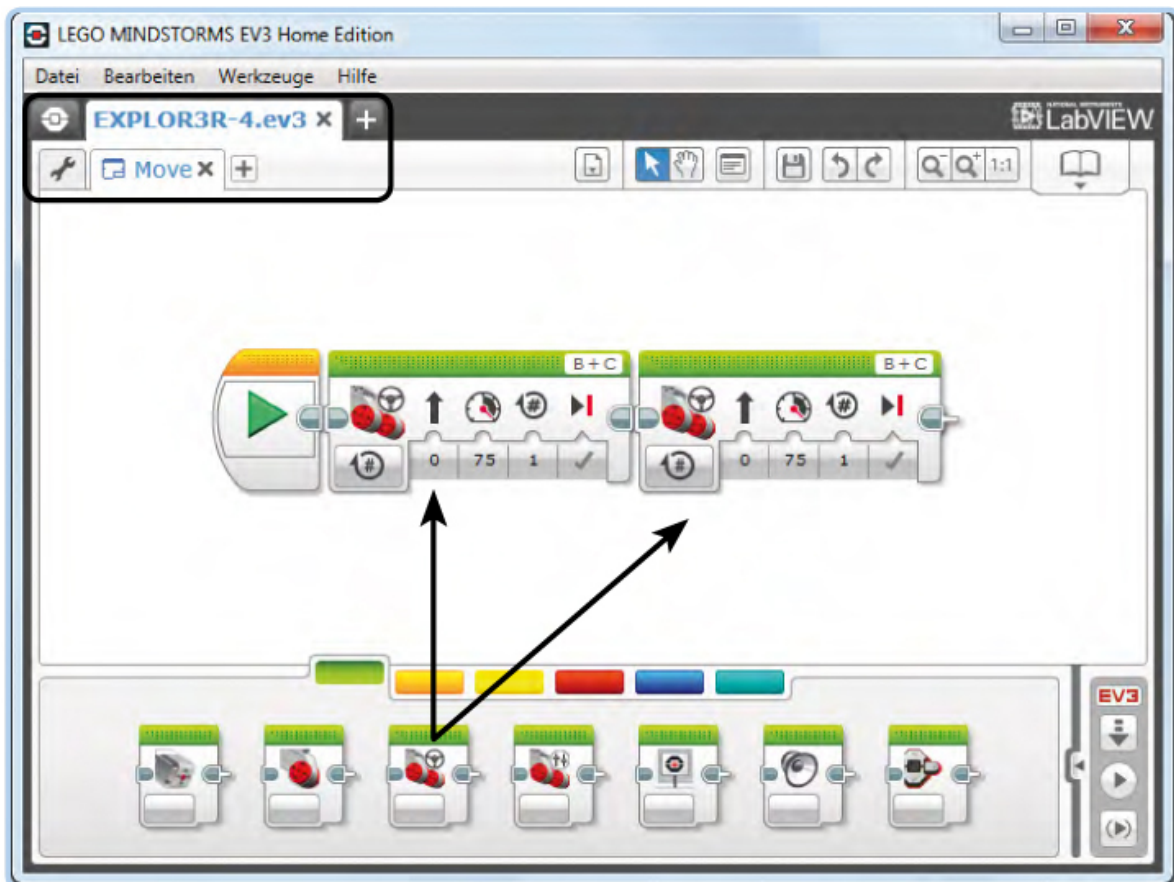


Abbildung 4.2: Erstelle das Move-Programm im EXPLOR3R-4-Projekt. Wähle einen Bewegungslenkungsblock aus der Palette und platziere ihn neben den Startblock. Platziere den zweiten Block daneben.

3. Per Voreinstellung sind die gerade von dir platzierten Blöcke so konfiguriert, dass der Roboter sich nur ein wenig bewegt. Wir wollen den Lenkungsblock jedoch so einstellen, dass er den Roboter ganze zwei Radumdrehungen rückwärts laufen lässt. Hierzu änderst du die Einstellungen des ersten Blocks, wie in [Abbildung 4-3](#) gezeigt.

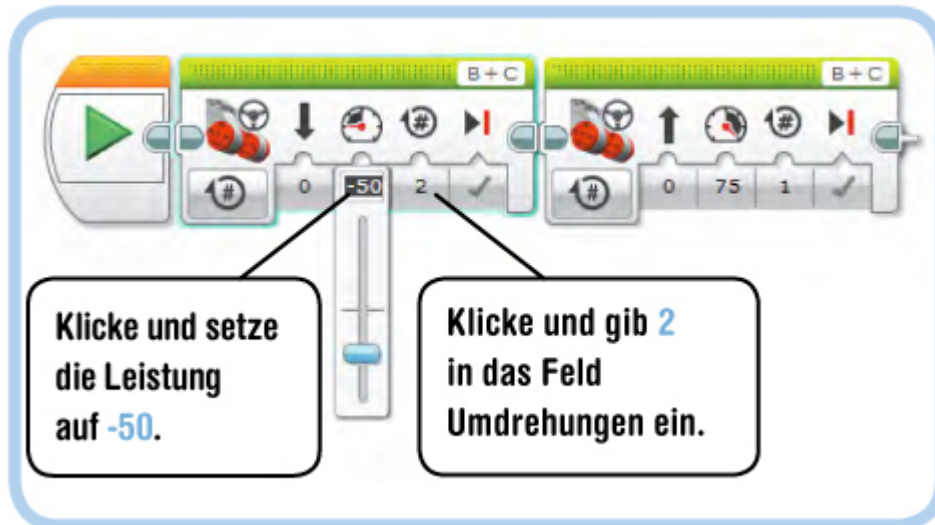


Abbildung 4.3: Konfiguriere den ersten Block, indem du die Leistung auf -50 einstellst. Dazu ziehst du den Regler nach unten oder gibst manuell -50 ein. Negative Werte lassen den Roboter rückwärts fahren. Danach gibst du in das Feld Umdrehungen 2 ein, sodass die Bewegung nach zwei Umdrehungen endet.

4. Jetzt änderst du die Einstellungen des zweiten Blocks. Dieser Block lässt den EXPLOR3R für drei Sekunden eine Drehung nach rechts machen. Zuerst veränderst du den Modus auf **An für n Sekunden**, wie in [Abbildung 4-4](#).

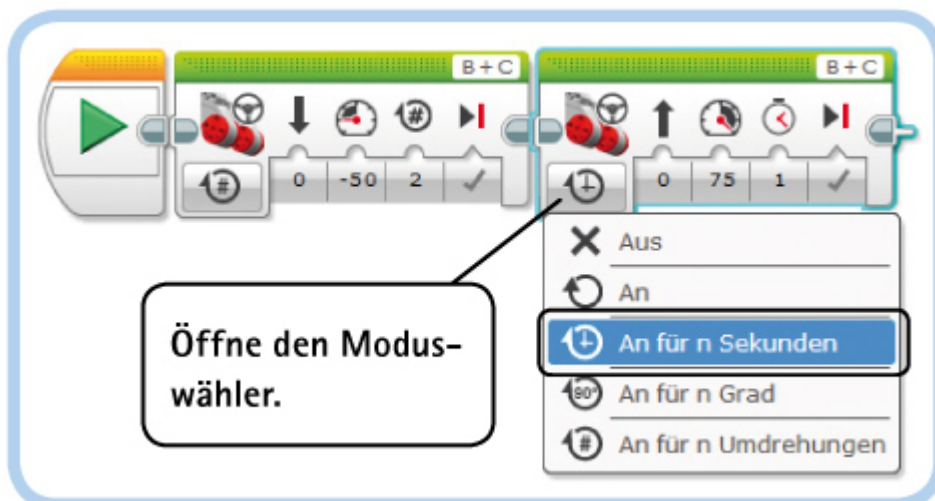


Abbildung 4.4: Klicke auf den Moduswähler des zweiten Blocks und wähle **An für n Sekunden**.

5. Damit sich der Roboter drei Sekunden lang schnell dreht, änderst du den zweiten Block, wie in [Abbildung 4-5](#) gezeigt.



Abbildung 4.5: Konfiguriere den zweiten Block, indem du den Regler ganz nach rechts ziehst und 3 in das Feld Sekunden eingibst.

- Nachdem du beide Lenkungsblöcke konfiguriert hast, kannst du das Programm auf deinen Roboter herunterladen und ausführen. Du weißt, dass es funktioniert, wenn der Roboter zwei Radumdrehungen rückwärts fährt und sich dann drei Sekunden lang schnell dreht.

HINWEIS Wenn sich dein Roboter nach links statt nach rechts dreht, hast du den Motor vielleicht an den falschen Anschluss am EV3-Stein angeschlossen. Prüfe anhand der Schritte auf [Seite 19](#) die Verkabelung deines Roboters.

Wie Modus und Einstellung funktionieren

Jetzt werfen wir einen genaueren Blick auf die Einstellungen der Blöcke, um besser zu verstehen, wie das Beispielprogramm funktioniert. Jede Aktion eines Blocks wird durch ihren Modus und die Einstellungen festgelegt. [Abbildung 4-6](#) zeigt verschiedene Wege, einen Bewegungslenkungsblock, über die beiden vorherigen Varianten hinaus zu konfigurieren.

Der Bewegungslenkungsblock hat verschiedene Modi, die im Moduswähler ausgewählt werden können und das Verhalten des Blocks geringfügig ändern. Zum Beispiel befand sich der erste Block im Modus **An für n Umdrehungen**, wodurch wir festlegen konnten, um wie viele Umdrehungen sich der Roboter bewegt, während der zweite sich im Modus **An für n Sekunden** befand, wo wir angeben konnten, für wie lange sich der Motor drehen sollte.

Die Modi im Bewegungslenkungsblock sind:

* Aus: Motor anhalten

- * An: Motor anschalten
- * An für n Sekunden: Schaltet den Motor für eine bestimmte Anzahl Sekunden ein und hält danach den Motor an.
- * An für n Grad: Dreht den Motor um eine bestimmte Anzahl Grad und hält danach den Motor an.
- * An für n Umdrehungen: Schaltet den Motor für eine bestimmte Anzahl Umdrehungen ein und hält danach den Motor an.

Du lernst die Modi An und Aus im Abschnitt »[Die An- und Aus-Modi in Bewegungslenkungsblöcken](#)« auf [Seite 45](#) kennen.

Anschlüsse

In der Anschlusseinstellung oben rechts im Block kannst du wählen, an welche Ausgabeanschlüsse des EV3 die Motoren angeschlossen sind, sodass das Programm weiß, welche Motoren geschaltet werden sollen. Die EXPLOR3R-Motoren sind mit den Anschlüssen B und C verbunden, weshalb wir in unserem Beispielprogramm die Vorgabe B + C beibehalten haben.

Lenkung

Wie du im Move-Programm gesehen hast, kannst du den Roboter auch lenken. Um die Lenkung des Roboters einzustellen, klicke auf den Regler **Lenkung** und ziehe ihn nach links (damit der Roboter nach links lenkt) oder nach rechts.

Wie dreht ein Fahrzeug ohne Lenkrad? Dieser Block steuert die Richtung des Roboters, indem beide Räder unabhängig voneinander geregelt werden. Damit der Roboter geradeaus fährt, drehen sich beide Räder mit der gleichen Geschwindigkeit in die gleiche Richtung. Um eine Kurven zu fahren, dreht sich ein Rad schneller als das andere oder die Räder drehen sich in unterschiedliche Richtungen, sodass der Roboter auf der Stelle dreht. [Abbildung 4-7](#) zeigt die verschiedenen Kombinationen bei Leistung und Lenkung für den EXPLOR3R.

Leistung

Die Leistungseinstellung regelt die Geschwindigkeit der Motoren. Null Leistung bedeutet, dass sich die Räder gar nicht drehen, während sie bei 100 die volle Drehzahl erreichen. Negative Werte, wie -100 oder -30, lassen den Roboter rückwärts fahren, wie du im Beispielprogramm gesehen hast.

Umdrehungen, Sekunden oder Grad

Abhängig vom ausgewählten Modus kannst du über die dritte Einstellung auf dem Bewegungslenkungsblock angeben, wie lange die Motoren laufen. Bei der Einstellung 3 für **An für n Sekunden** drehen sich die Motoren z.B. drei Sekunden lang.

An für n Grad ermöglicht es dir, festzulegen, um wie viel Grad sich der Motor und damit auch die Räder drehen. Eine Drehung um 360 Grad bedeutet eine volle Umdrehung der Räder, 180 nur eine halbe. Du kannst die Räder so einstellen, dass sie sich eine bestimmte Anzahl ganzer Umdrehungen drehen, wenn sich der Block im Modus **An für n Umdrehungen** befindet, wie im ersten Block des Beispielprogramms.

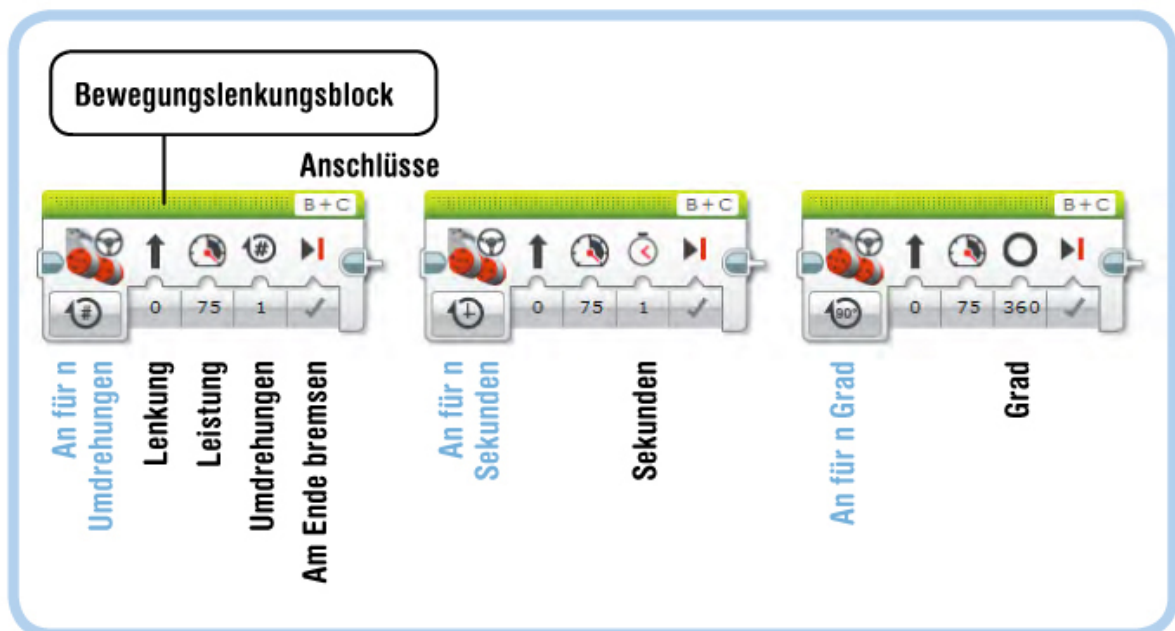


Abbildung 4-6: Modi (blau) und Einstellungen (schwarz) im Bewegungslenkungsblock. Um einen anderen Modus auszuwählen, klickst du auf dem Moduswähler und wählst einen aus dem Drop-down-Menü aus. Die meisten Einstellungen bleiben unabhängig vom Modus gleich.

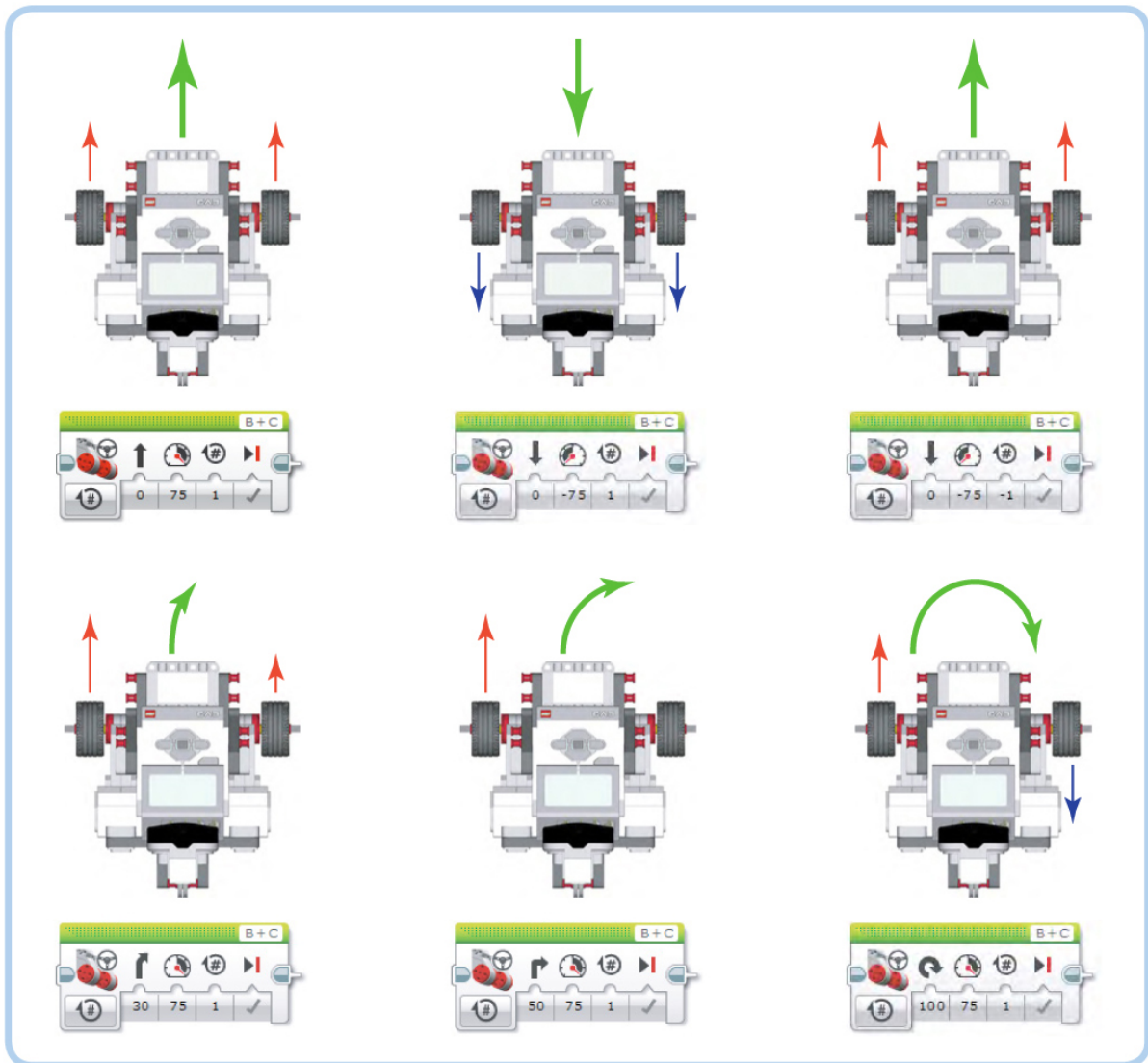


Abbildung 4-7: Damit sich der Roboter dreht, verwendest du die Einstellung Lenkung. Der Block regelt die Richtung und Geschwindigkeit beider Motoren und lässt den Roboter wenden. Die roten Pfeile zeigen an, dass ein Rad vorwärts dreht, und die blauen, dass es rückwärts läuft. Die grünen Pfeile zeigen die Richtung, in die sich der Roboter im Ergebnis dreht.

HINWEIS Die Eingabe eines negativen Leistungswerts, wie -75, und eines positiven Drehwerts, wie 1 Umdrehung oder 360 Grad, führen dazu, dass sich der Roboter rückwärts bewegt. Dies gilt auch für positive Leistung und eine negative Rotation, so wie Leistung 75 und Rotation -1. Wenn du jedoch eine negative Leistung und eine negative Rotation angibst, wie Leistung -75 und Rotation -1, fährt der Roboter vorwärts! Betrachte [Abbildung 4-7](#), wenn du unsicher bist, wie sich eine Lenkungskombination auf den Roboter auswirkt.

Am Ende bremsen

Die Option **Am Ende bremsen** regelt, wie die Motoren am Ende ihre Rotation nach Sekunden, Grad oder Umdrehungen abbremsten. Die Einstellung **Wahr** () lässt die abrupt stoppen, während **Falsch** () sie weich anhält.

Richtige Drehungen ausführen

Wenn du den Bewegungslenkungsblock verwendest, um den Roboter eine Drehung um 90 Grad machen zu lassen, könntest du denken, die Einstellung sollte 90 Grad betragen, aber das stimmt nicht. Die Grad-Einstellung gibt nur an, um wie viel Grad sich die Motoren drehen, und damit auch die Räder. Die tatsächliche Anzahl von Grad für eine 90-Grad-Drehung unterscheidet sich von Roboter zu Roboter. In *Selbst entdecken 2* findest du den richtigen Wert für deinen Roboter heraus.

SELBST ENTDECKEN 1: BESCHLEUNIGE

Schwierigkeitsgrad:  Zeit: 

Da du nun einige wichtige Eigenschaften des Bewegungslenkungsblocks kennengelernt hast, kannst du jetzt ein wenig damit experimentieren. Das Ziel dieser Aufgabe besteht darin, ein Programm zu entwickeln, das den Roboter zunächst langsam bewegen und dann zunehmend beschleunigen lässt. Zuerst ziehst du 10 Bewegungslenkungsblöcke in den Programmierbereich und konfigurierst die ersten beiden Blöcke wie in [Abbildung 4-8](#). Den dritten Block konfigurierst du genauso, stellst jedoch die Leistung der Motoren auf 30 ein. Dann erhöhst du sie mit jedem nachfolgenden Block um 10, bis die volle Leistung erreicht ist.

Die Blöcke befinden sich jetzt im Modus *An für n Sekunden*. Wenn du das Programm getestet hast, änderst du alle Blöcke in den Modus *An für n Umdrehungen*, wobei Umdrehung auf 1 gesetzt ist, und führst es erneut aus. Welches Programm läuft länger? Kannst du das unterschiedliche Verhalten erklären?



Abbildung 4-8: Die ersten Blöcke des Programms Entdeckung 1. Füge für jede Selbstentdecken-Aufgabe ein neues Programm hinzu und speichere es, wenn du fertig bist, sodass du sie später weiter verwenden kannst.

SELBST ENTDECKEN 2: EXAKTE DREHUNGEN

Schwierigkeitsgrad: Zeit:

Kannst du deinen Roboter dazu bringen, sich auf der Stelle um 90 Grad zu drehen? Erstelle ein neues Programm mit einem Bewegungslenkungsblock im Modus *An für n Grad*, wie in [Abbildung 4-9](#) gezeigt. Stelle den Lenkungsregler ganz auf rechts wie im Move-Programm. Um wie viel Grad müssen sich die Räder drehen, damit der Roboter sich genau um 90 Grad dreht?

Beginne mit der Einstellung 275 Grad. Wenn das nicht ausreicht, versuche 280, 285 usw. und führe das Programm jedes Mal aus, um zu prüfen, ob der Roboter die gewünschte Drehung macht.

Wenn du den richtigen Wert für die 90-Grad-Drehung ermittelt hast, finde heraus, welcher Wert zu einer 180-Grad-Drehung führt.



Abbildung 4-9: Das Programm für Selbst entdecken 2. Welcher Wert lässt den Roboter um 90 Grad wenden? Welcher Wert ist für 180 Grad notwendig?

SELBST ENTDECKEN 3: BEWEG UND DREH DICH

Schwierigkeitsgrad: Zeit:

Erstelle ein Programm mit drei Bewegungslenkungsblöcken, das den EXPLOR3RR dazu bringt, drei Sekunden lang mit 50% der Motorleistung vorwärtszufahren, sich dann um 180 Grad zu drehen und anschließend zu seiner Startposition zurückzukehren. Beim Konfigurieren des Blocks, der den Roboter umdrehen lässt (zweiter Block), setzt du im Feld »Grad« den Wert ein, den du in der Selbst entdecken 2 ermittelt hast.

SELBST ENTDECKEN 4: BUCHSTABIERE

Schwierigkeitsgrad: Zeit:

Entwickle ein Programm mit Bewegungslenkungsblöcken, das den EXPLOR3R den ersten Buchstaben deines Namens nachfahren lässt. Wie viele Blöcke brauchst du für deinen Anfangsbuchstaben?

HINWEIS Bei geschwungenen Kurven verwendest du den Lenkungsregler, um einzustellen, wie eng die Kurve gefahren werden soll.

Der Klangblock

Du hast gesehen, wie viel Spaß es macht, Programme zu entwickeln, die den EXPLOR3R herumfahren lassen. Noch spannender wird es, wenn du mithilfe des *Klangblocks* den Roboter dazu bringst, Klänge zu erzeugen. Dein Roboter kann zwei verschiedene Arten von Klängen abspielen: einen einfachen *Ton* (z. B. einen Piepston) oder eine *Klangdatei*, wie beispielsweise einen Applaus oder ein gesprochenes Wort (z. B. »Hallo«). Wenn du in deinen Programmen Klangblöcke einsetzt und deinen Roboter auf diese Weise zum »Sprechen« bringst, wirkt er interaktiver und lebendiger.

Die Konfiguration des Klangblocks

Die Programmierblöcke unterscheiden sich zwar durch die Handlungen, die sie auslösen, jedoch werden sie alle auf die gleiche Weise benutzt. Mit anderen Worten: Du kannst einfach einen Klangblock aus der Programmierpalette in den Programmierbereich ziehen – genau so, wie du es mit dem Bewegungsblock getan hast. Sobald sich der Block im Programmierbereich befindet, kannst du seine Einstellungen im Konfigurationsbereich ändern.

Bevor du ein Programm mit Klangblöcken entwickelst, werden wir einen kurzen Blick auf die verschiedenen Modi des Blocks werfen, die in [Abbildung 4-10](#) gezeigt werden. Die vier Modi des Klangblocks sind:

- * Datei abspielen: Spielt einen vorher aufgenommenen Klang ab, wie »Hello«.
- * Ton abspielen: Spielt einen Ton mit einer bestimmten Tonhöhe (Frequenz) für eine bestimmte Dauer ab.
- * Note abspielen: Spielt eine Klaviernote für eine bestimmte Dauer.
- * Stopp: Stoppt die laufende Klangausgabe.

Dateiname

Im Modus Datei abspielen kannst du einen Klang abspielen, indem du auf das Feld **Dateiname** klickst und im darauf folgenden Menü eine Datei auswählst. Du kannst Klänge aus verschiedenen Kategorien wählen, wie Tiere, Farben, Kommunikation und Zahlen. Du kannst auch eine eigene Klangdatei über **Werkzeuge** ▶ **Geräusch-Editor** aufnehmen. (Unter **Hilfe** ▶ **Werkzeuge** ▶ **Geräusch-Editor** findest du weitere Informationen.)

Lautstärke

Gib einen Wert zwischen 0 (Leise) und 100 (Laut) ein, um die Wiedergabelautstärke des Klangs einzustellen.

Wiedergabeart

Verwende die Einstellung Wiedergabeart, um festzulegen, was passiert, wenn der Klang abgespielt wird. Bei **Warten auf Abschluss (0)** wird das Programm so lange angehalten, bis der Klang vollständig abgespielt ist. Bei **Einmal wiederholen (1)** fährt das Programm mit dem nächsten Block fort, während der Klang abgespielt wird. Wenn du **Wiederholen (2)** wählst, wird der Klang so lange abgespielt, wie das Programm noch läuft. Für die meisten Anwendungen wirst du die Einstellung **Warten auf Abschluss** wählen.

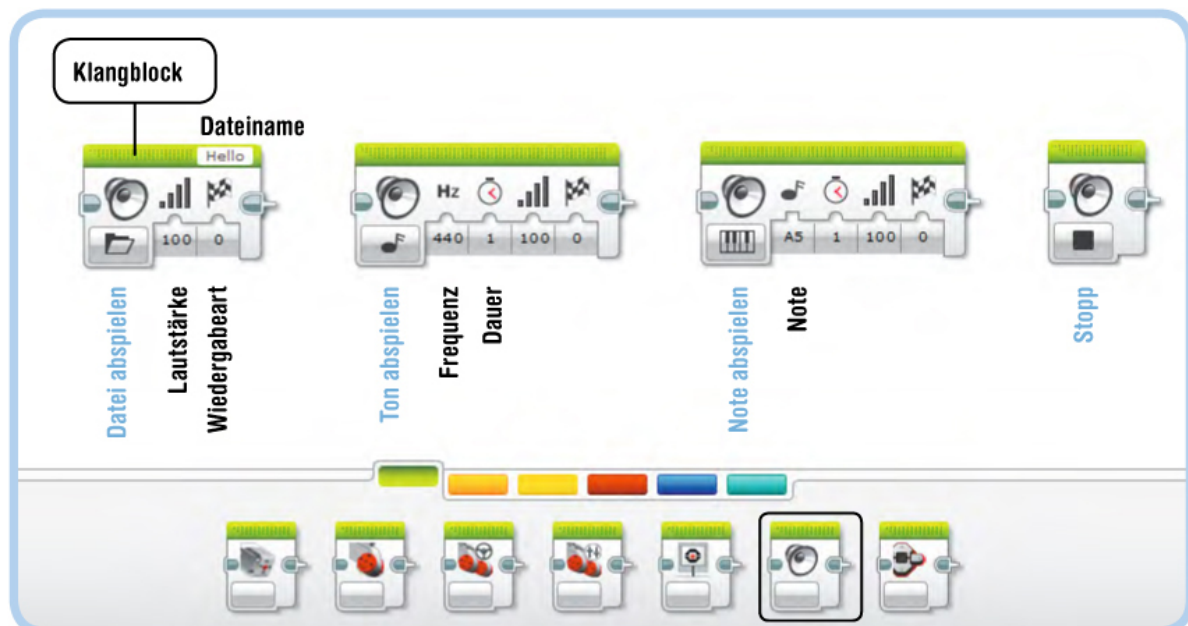


Abbildung 4-10: Die vier Modi (in Blau) des Klangblocks und ihre Einstellungen (in Schwarz)

Note oder Ton

Anhängig vom ausgewählten Modus kannst du eine Klaviernote oder eine Tonhöhe (Frequenz) in Hertz (Hz) wählen. Eine Tonhöhe von 440 Hz (die voreingestellte Frequenz) ist klar mit dem menschlichen Ohr zu hören und stellt einen prima Ton zum Testen von Programmen dar. Du kannst diesen Ton z.B. abspielen, wenn ein bestimmter Programmierblock abgeschlossen ist.

Dauer

Im Feld **Dauer** legst du fest, wie viele Sekunden die Note oder der Ton abgespielt wird.

Der Klangblock in Aktion

Lass uns jetzt ein Programm namens *SoundCheck* erstellen, das den Roboter bewegt und Klänge abspielt, sodass wir sehen können, wie der Klangblock funktioniert.

Um anzufangen, erzeugen wir ein neues Programm, wie in [Abbildung 4-11](#) gezeigt. Um dieses Programm und die weiteren Beispiele für dieses Buch zu erstellen, führst du die folgenden Schritte für jeden Block im Programm aus:

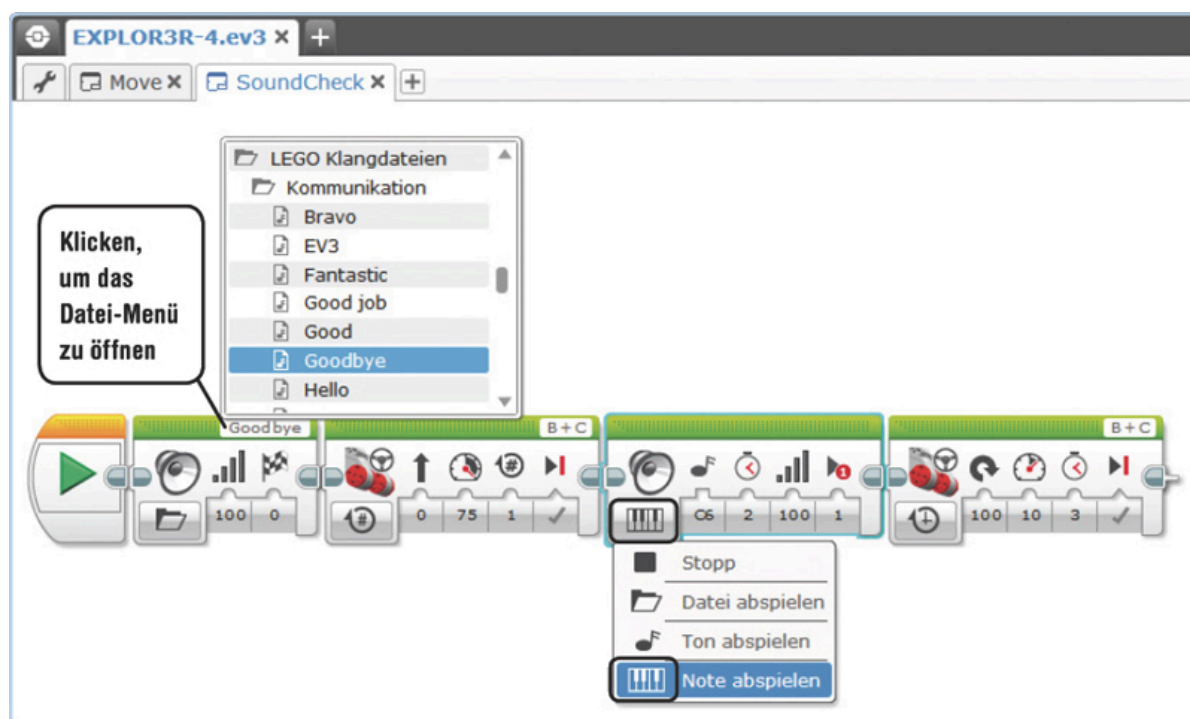


Abbildung 4-11: Das Programm **SoundCheck**. Um den Modus eines Blocks einzustellen, verwendest du den Moduswähler und wählst aus der Liste mit Symbolen dasjenige, das zur Abbildung passt.

1. Suche den Block in der Programmierpalette anhand seiner Farbe und platziere ihn im Programmierbereich. Der Klangblock z.B. ist grün und sein Symbol zeigt einen Lautsprecher.
2. Wähle den passenden Modus anhand des Symbols im Moduswähler. Das Klavier-Symbol z.B. bedeutet, dass du eine Note abspielen sollst.

3. Dann nimmst du die restlichen Einstellungen im Block vor. Zum Beispiel konfigurierst du die Werte für Lenkung und Leistung in den Bewegungslenkungsblöcken.

Wenn du dein Programm erstellt hast, lade es auf deinen Roboter herunter und führe es aus.

Wie das Programm SoundCheck funktioniert

Nachdem du das Programm ausgeführt hast, sehen wir uns an, wie es funktioniert. Der erste Klangblock lässt den EXPLOR3R »Goodbye« sagen. Die Wiedergabeart ist **Warten auf Abschluss**, sodass der Roboter das Wort abwartet, bevor er zum nächsten Block weitergeht. Nun lässt ein Bewegungslenkungsblock den Roboter eine Umdrehung vorwärts fahren und dann spielt ein weiterer Klangblock eine Note ab. Jetzt wird nicht abgewartet, bis die Note zu Ende ist, sondern der zweite Bewegungslenkungsblock lässt den Roboter drei Sekunden lang nach rechts lenken. Dann hält der Roboter an.

SELBST ENTDECKEN 5: IN WELCHE RICHTUNG GEHST DU?

Schwierigkeitsgrad:  Zeit: 

Erstelle ein Programm wie *SoundCheck*, das den Roboter die jeweils eingeschlagene Richtung ankündigen lässt, während er sich in diese Richtung bewegt. Während er sich vorwärts bewegt, sollte er »Forward« sagen, und während er sich rückwärts bewegt, sollte er »Backward« sagen. Wie konfigurierst du die Einstellung Wiedergabeart in den Klangblöcken?

SELBST ENTDECKEN 6: DJ SPIELEN

Schwierigkeitsgrad:  Zeit: 

Du kannst deine eigene Musik komponieren, indem du ein Programm mit einer Reihe von Klangblöcken entwickelst. Kannst du eine bekannte Melodie auf dem EV3 abspielen lassen oder eine eigene Melodie komponieren?

Der Anzeigeblock

Neben Bewegung und Klang kann mit einem EV3-Programm auch gesteuert werden, was auf dem LCD-Display des EV3-Steins angezeigt wird. Du kannst z.B. ein Programm entwickeln, das die Anzeige wie in [Abbildung 4-12](#) aussehen lässt. (Die LCD-Anzeige ist 178 Pixel breit und 128 Pixel hoch. Pixel sind kleine Bildpunkte.)

Es macht Spaß, mit dem Anzeigeblock herumzuxperimentieren, noch wichtiger aber ist die Möglichkeit, deine Programme besser zu testen.

Zum Beispiel kannst du einen Sensormesswert auf dem Display anzeigen, um zu prüfen, ob der Sensor richtig funktioniert, wie ich in [Teil V](#) dieses Buchs zeige.

Du kannst den *Anzeigeblock* einsetzen, um ein Bild (z. B. eine Glühbirne), einen Text (ein Wort wie z.B. »Hallo«) oder eine Zeichnung (z. B. einen ausgefüllten Kreis) auf dem Display anzuzeigen. Es ist nicht möglich, mit einem Anzeigeblock mehrere Bilder oder Textzeilen gleichzeitig abzubilden. Um die in [Abbildung 4-12](#) dargestellte Anzeige zu erstellen, musst du daher mehrere Anzeigeblocke einsetzen.

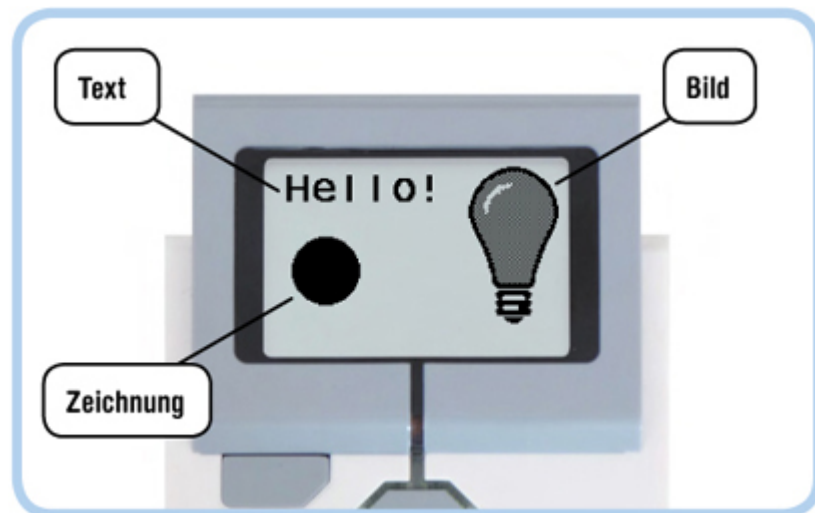


Abbildung 4-12: Mithilfe des Anzeigeblocks lassen sich Bilder, Texte und Zeichnungen auf dem Display des EV3-Steins anzeigen.

Die Konfiguration des Anzeigeblocks

Nachdem der Anzeigeblock etwas auf dem EV3-Display ausgegeben hat, geht das Programm weiter zum nächsten Block, z.B. einem Bewegungsblock. Das EV3-Display ändert sich erst, wenn durch einen weiteren Anzeigeblock etwas

anderes darauf ausgegeben wird. In diesem Beispiel würde das Bild so lange erhalten bleiben, wie sich der Roboter bewegt.

Wenn ein Programm abgeschlossen ist, springt der EV3 sofort wieder ins Menü zurück. Wenn dein letzter Programmierblock also ein Anzeigeblock ist, wirst du dessen Inhalt nicht sehen können, da sofort nach diesem Block das Programm abgeschlossen ist und das Menü erscheint. Um den Inhalt trotzdem lesen zu können, kannst du am Ende einen Block, z.B. einen Bewegungsblock, hinzufügen, damit das Programm nicht abrupt endet. [Abbildung 4-13](#) zeigt die vier Hauptmodi des Anzeigeblocks:

- * **Bild:** Zeigt auf dem Display ein von dir ausgewähltes Bild an, wie ein lächelndes Gesicht.
- * **Formen:** Zeichnet eine Linie, einen Kreis, ein Rechteck oder einen Punkt auf das Display.
- * **Text:** Gibt eine Textzeile auf das Display aus.
- * **Bildschirm zurücksetzen:** Löscht das Display und zeigt das Mindstorms-Logo an, das du normalerweise siehst, wenn du ein Programm ohne Anzeigeblöcke ausführst.

Untermodi

Einige Modi haben Untermodi. Im Formen-Modus kannst du z.B. zwischen vier Untermodi wählen: Linie, Kreis, Rechteck und Punkt, wie in [Abbildung 4-13](#) gezeigt. Der Modus Kreis gibt auf dem Display z.B. einen Kreis aus, dessen Position, Radius, Füllung und Farbe du in den Einstellungen festlegen kannst.

Dateiname

Im Bild-Modus kannst du auf Dateiname klicken und ein Bild aus Kategorien wie Augen, Ausdrücke, Objekte und LEGO auswählen.

Du kannst auch eigene Bilder erstellen oder hochladen, indem du **Werkzeuge** ▶ **Bild-Editor** aufrufst. (Mehr Informationen bekommst du unter **Hilfe** ▶ **Werkzeuge** ▶ **Bild-Editor**.)

Bildschirm löschen

Die Einstellung *Bildschirm löschen* bietet dir die Möglichkeit, das Display zu löschen, bevor etwas Neues angezeigt wird (wenn *Wahr* eingestellt wird) oder der bestehenden Darstellung etwas hinzuzufügen (bei *Falsch*). Um mehrere

Objekte auf dem Display anzuzeigen, benötigst du mehrere Anzeigeblocke. Der erste Block sollte das Display löschen, bevor etwas Neues angezeigt wird, und die anderen Blöcke sollten einfach etwas zeichnen. Um das zu erreichen, setzt du die Einstellung *Bildschirm löschen* im ersten Block auf *Wahr* und in den folgenden Blöcken auf *Falsch*. Wie das funktioniert, siehst du im Beispiel *DisplayTest*.

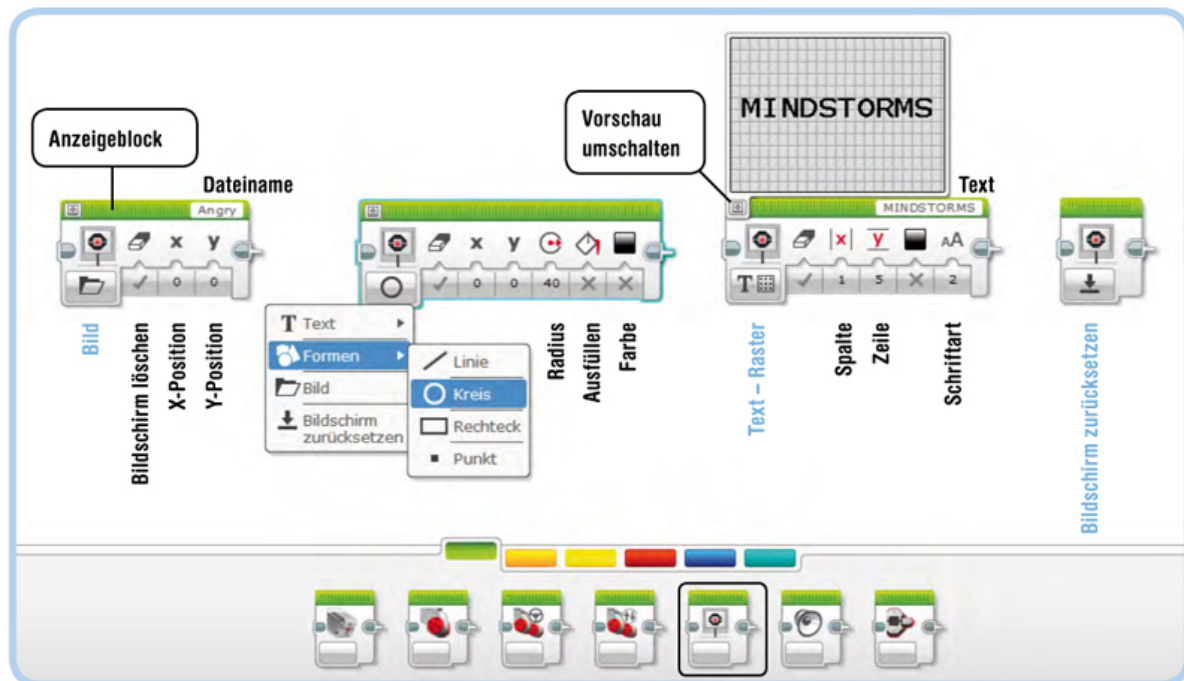


Abbildung 4-13: Die vier Modi (blau) des Anzeigeblocks und ihre Einstellungen (schwarz). Du kannst die Vorschau des EV3-Displays verwenden, um deine Einstellungen zu überprüfen. Bist du zufrieden, schließt du die Vorschau.

Radius und Füllung

Einige Einstellungen sind speziell für bestimmte Modi im Anzeigeblock gedacht. Die Einstellung *Radius* legt z.B. die Größe eines Kreises fest und *Ausfüllen* gibt an, ob du nur einen Umriss (*Falsch*) oder ein ausgefülltes Objekt (*Wahr*) zeichnest.

Farbe

Normalerweise wirst du die Farbe für deinen Text auf Schwarz einstellen, so wie bei MINDSTORMS im Feld *Text* in [Abbildung 4-13](#). Der Text kann Zahlen enthalten und du kannst die Textgröße über die *Schriftart* mit 0 (klein), 1 (fett) und 2 (groß) einstellen.

x, y, Spalte und Zeile

Wenn du ein Bild, einen Text (Modus *Text-Pixel*) oder eine Form anzeigst, kannst du die Position mit den Einstellungen X (Position relativ zum linken Displayrand) und Y (Position relativ zum unteren Rand) festlegen. Wenn du eine Vorschau des Displays verwendest (siehe [Abbildung 4-13](#)), kannst du die Werte für X und Y einfach herausfinden.

Im Modus *Text-Raster* kannst du alternativ die Spalte (0-20) und Zeile (0-11) angeben, was die Ausrichtung mehrerer Elemente erleichtert.

HINWEIS Weitere Einzelheiten über die Blöcke findest du in der Hilfe. Für den Anzeigeblock gehst du zu *Hilfe* ▶ *EV3-Hilfe einblenden* ▶ *Programmierblöcke* ▶ *Aktions-Blöcke* ▶ *Anzeige*.

Der Anzeigeblock in Aktion

Lass uns die Funktion des Anzeigeblocks testen, indem wir ein Programm schreiben, das Dinge auf dem EV3-Display ausgibt, während sich der Roboter bewegt. Erstelle ein Programm namens *DisplayTest* und platziere drei Anzeige- und zwei Bewegungslenkungsblöcke im Programmierbereich, wie in [Abbildung 4-14](#) gezeigt. Dann konfigurierst du die Blöcke wie gezeigt. Wenn du alle Blöcke fertig konfiguriert hast, lade das Programm auf deinen Roboter herunter und führe es aus.

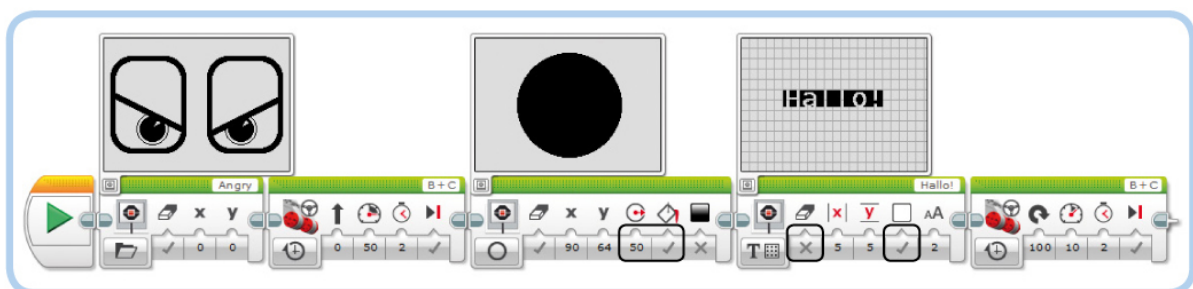


Abbildung 4-14: Das Programm **DisplayTest** bewegt deinen Roboter, während Dinge auf dem Display angezeigt werden. Die Vorschau zeige ich hier zur Vereinfachung, du kannst sie aber auch ausschalten, wenn du möchtest.

SELBST ENTDECKEN 7: UNTERTITEL

Schwierigkeitsgrad:  Zeit: 

Erstelle ein Programm, das vier Klangblöcke verwendet, um »Hello. Good morning. Goodbye!« zu sagen. Benutze die Anzeigeblocke, um

jeweils den Text, den der Roboter sagt, als Untertitel auf dem EV3-Display anzuzeigen, und lösche das Display, wenn der Roboter etwas Neues sagt. Stellst du die Anzeigeblöcke vor oder hinter die Klangblöcke?

SELBST ENTDECKEN 8: WARTEN AUF DEN EXPLOR3R

Schwierigkeitsgrad:  Zeit: 

Programmiere den EXPLOR3R so, dass er eine Acht fährt, wie in [Abbildung 4-15](#) gezeigt. Der Roboter sollte während der Bewegung Gesichter auf dem Display anzeigen. Wähle dazu verschiedene Bilder aus der Kategorie Augen.

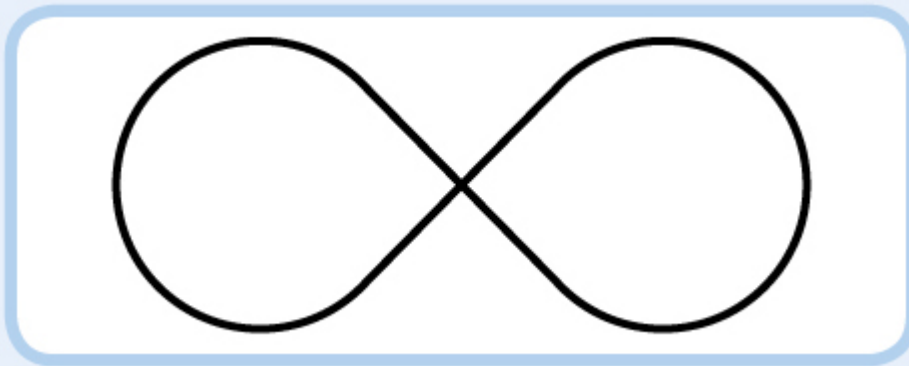


Abbildung 4-15: Die Fahrstrecke für Selbst entdecken 8. Versuche, den Roboter einen Kurs wie diesen fahren zu lassen. Allerdings muss er der Linie momentan noch nicht genau folgen. (Wie das geht, erfährst du in [Kapitel 7](#).)

Wie das DisplayTest-Programm funktioniert

Im DisplayTest-Programm geben die Anzeigeblöcke verschiedene Dinge auf dem Display aus und die Bewegungslenkungsblöcke lassen den EXPLOR3R herumfahren. Der erste Anzeigeblock löscht das Display, bevor das Bild (die bösen Augen) angezeigt wird. Dann fängt der Roboter an, sich zu bewegen, und das Bild verbleibt auf dem Display, während der Bewegungslenkungsblock ausgeführt wird. Der nächste Anzeigeblock (der ausgefüllte Kreis) löscht das Display ebenfalls, sodass das Bild mit den bösen Augen verschwindet, bevor der Kreis angezeigt wird.

Das Programm fährt dann mit einem weiteren Anzeigeblock fort, der einen weißen Text in den Kreis schreibt. Dieser Block löscht den Bildschirm vorher

nicht, sodass du Kreis und Text gleichzeitig sehen kannst. Schließlich steuert ein Bewegungsblock den Roboter nach rechts und das Programm ist beendet.

Der Stein-Statusleuchte-Block

Die Stein-Statusleuchte um die EV3-Tasten herum ist normalerweise grün. Während ein Programm ausgeführt wird, blinkt sie. Mit dem Block Stein-Statusleuchte kannst du dieses Verhalten ändern und die Leuchte steuern. Die drei Modi, gezeigt in [Abbildung 4-16](#), sind folgende:

- * **An:** Schaltet die Leuchte an und legt eine *Farbe* fest: Grün (0), Orange (1) oder Rot (2). Die Einstellung *Pulsieren* legt fest, ob die Leuchte blinkt (Wahr) oder dauernd leuchtet (Falsch).
- * **Aus:** Schaltet die Leuchte aus.
- * **Zurücksetzen:** Zeigt die grün blinkende Leuchte an, die du normalerweise bei der Programmausführung siehst.

Jetzt erstellst du das Programm *ButtonLight*, um diese Funktionen zu testen, wie in [Abbildung 4-17](#). Die Tasten sollten rot leuchten, wenn der Roboter »Red« sagt, und grün, wenn er »Green« sagt.

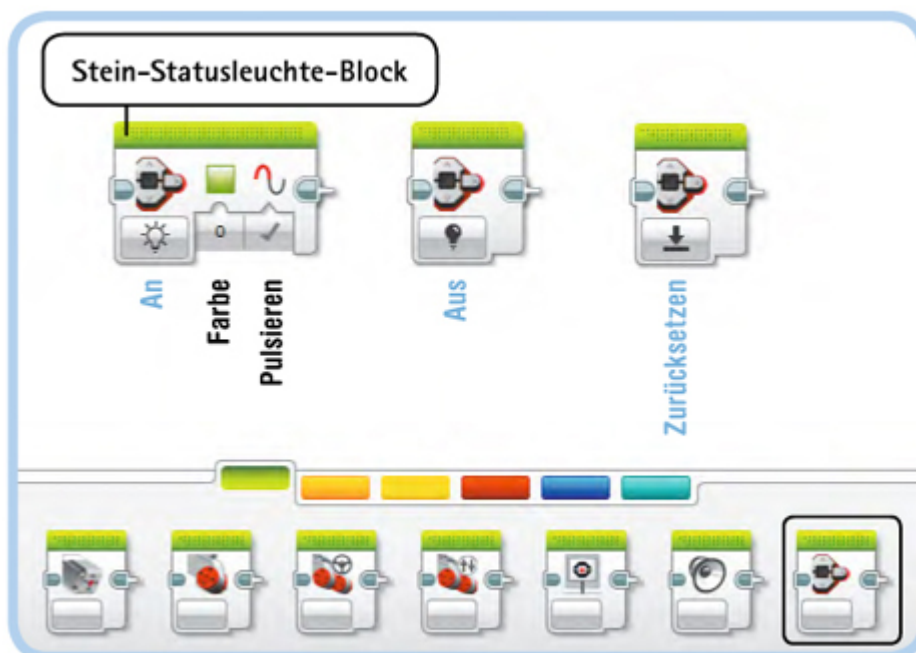


Abbildung 4-16: Die drei Modi des Stein-Statusleuchte-Blocks und ihre Einstellungen

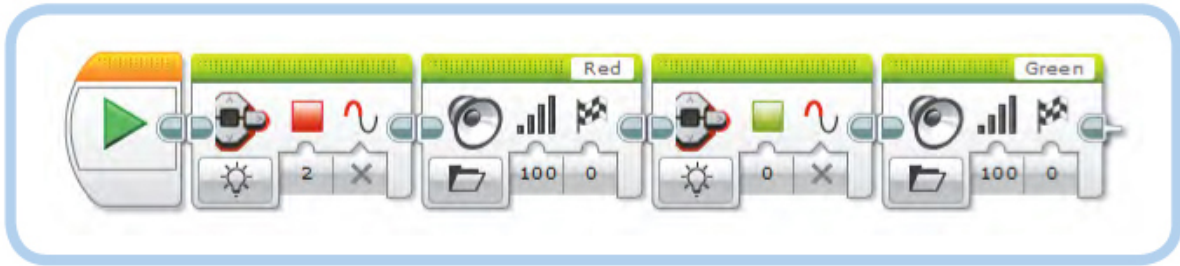


Abbildung 4-17: Das Programm **ButtonLight**

SELBST ENTDECKEN 9: AMPEL

Schwierigkeitsgrad: Zeit:

Verändere das *ButtonLight*-Programm so, dass dein Roboter zu einer Ampel wird. Lass den Roboter »Stop«, »Activate« und »Go« sagen, während er die entsprechenden Farben Rot, Orange und Grün anzeigt.

Die *An*- und *Aus*-Modi in Bewegungsblöcken

Nachdem du jetzt die Programmierung mit Aktionsblöcken kennst, bist du bereit, die Modi *An* und *Aus* im Bewegungslenkungsblock und anderen Blöcken zu erforschen. Ein Block im *An*-Modus schaltet die Motoren ein und fährt dann sofort mit dem nächsten Block fort. Während das Programm weiterläuft, drehen die Motoren ebenfalls weiter, bis es zu einem Block kommt, der sie wieder abschaltet oder etwas anderes tun lässt. Der Modus *Aus* schaltet die Motoren ab.

Um zu sehen, wozu das gut ist, erstellst du das in [Abbildung 4-18](#) gezeigte Programm *OnOff*.

Wenn du das Programm ausführst, sollte sich der Roboter bewegen und gleichzeitig anfangen, »LEGO MINDSTORMS« zu sagen. Wenn er die Wörter gesprochen hat, sollte die Bewegung aufhören und ein Klang abgespielt werden. Die Motoren halten genau in dem Moment an, in dem der Roboter die Wörter »LEGO MINDSTORMS« fertig gesprochen hat, auch wenn du vorher nicht angegeben hast, wie lange das dauern wird. So kannst du die Motoren so lange laufen lassen, bis du sie mit dem Modus *Aus* abstellst, unabhängig davon, was dazwischen passiert.

Wenn du ein Programm mit nur einem Bewegungsblock im Modus An erstellst, könntest du die Vorstellung haben, dass der Roboter nicht mehr aufhört, sich zu bewegen. Dieser Block schaltet zwar die Motoren ein, wenn das Programm jedoch alle seine Blöcke durchlaufen hat, werden am Ende auch die Motoren ausgeschaltet.



Abbildung 4-18: Das Programm OnOff. Achte genau auf die Einstellungen in den Bewegungslenkungsblöcken.

SELBST ENTDECKEN 10: RADIO IM FAHRMODUS

Schwierigkeitsgrad: Zeit:

Erweitere das Programm aus *Selbst entdecken 6* auf [Seite 42](#) und lasse den Roboter vorwärts fahren, während er den Klang abspielt. Verwende einen Bewegungslenkungsblock im Modus *An* am Programmanfang und am Ende einen weiteren im Modus *Aus*. Was geschieht, wenn du dazwischen weitere Bewegungsblöcke im Modus *An* einfügst (mit anderen Lenkbefehlen)?

Die Blöcke Hebellenkung, Großer Motor und Mittlerer Motor

Neben den Bewegungslenkungsblöcken gibt es drei weitere Blöcke, mit denen du Motoren steuern kannst. Der erste ist der Block Hebellenkung, mit dem du ein Fahrzeug mit zwei Raupen oder Rädern steuerst, wie den EXPLOR3R, ähnlich wie du es bereits getan hast. Statt jedoch Lenkung und Leistung für den gesamten Roboter einzustellen, kannst du hiermit die Leistung der Motoren einzeln steuern. Durch unterschiedliche Kombinationen der Leistung für linke und rechte Räder fährt der EXPLOR3R in unterschiedliche Richtungen (siehe [Abbildung 4-7](#)).

Das Programm *Tank* zeigt, wie der Hebellenkungsblock funktioniert (siehe [Abbildung 4-19](#)). Der Block hat beinahe dieselben Funktionen wie der Bewegungslenkungsblock, aber die Einstellungen unterscheiden sich. Wenn du Fahrzeuge dieser Art baust, verwendest du einfach den Block, der dir am besten gefällt.

Wird ein Motor programmiert, schneller zu laufen als der andere (wie im Programm *Tank*), endet der Block, sobald der schnellere Motor die angegebene Anzahl Umdrehungen oder Grad erreicht hat. (Das gilt auch für den Bewegungslenkungsblock.)

Du verwendest den Block *Großer Motor*, um einzelne große Motoren zu steuern. Das ist für Mechanismen nützlich, die nur einen Motor verwenden, wie eine Klaue, die ein Objekt greift. Die Modi und Einstellungen sind die gleichen wie im Block *Hebellenkung*, jetzt beziehen sie sich jedoch nur auf einen Motor.

Das Programm in [Abbildung 4-20](#) verwendet zwei Blöcke *Großer Motor*, um jeweils den linken (B) und den rechten (C) Motor unabhängig voneinander zu steuern.

Der Block *Mittlerer Motor* gleicht dem *Großer Motor*, nur dass er zur Steuerung des mittleren Motors dient, der im EV3-Kasten enthalten ist. (Du siehst ihn in [Kapitel 12](#) in Aktion.)

Weitere Experimente

Jetzt kennst du die Grundlagen der LEGO-Mindstorms-EV3-Programmierung. Herzlichen Glückwunsch! Du weißt jetzt, wie du Roboter so programmierst, dass sie sich bewegen, Klänge abspielen, Lampen leuchten lassen sowie Text und Bilder auf dem EV3-Display anzeigen. In [Kapitel 5](#) erfährst du mehr über den Einsatz der Programmierblöcke, z.B. wie du mit Blöcken eine Pause programmierst und eine Reihe von Blöcken wiederholst.

Bevor du weitermachst, versuche einiger der folgenden *Selbstentdecken-Aufgaben* zu lösen, um deine Kenntnisse zu vertiefen.

HINWEIS Vergiss nicht, deine Programme nach dem Lösen einer Aufgabe zu speichern. Du möchtest sie vielleicht später als Ausgangspunkt für größere Projekte verwenden.

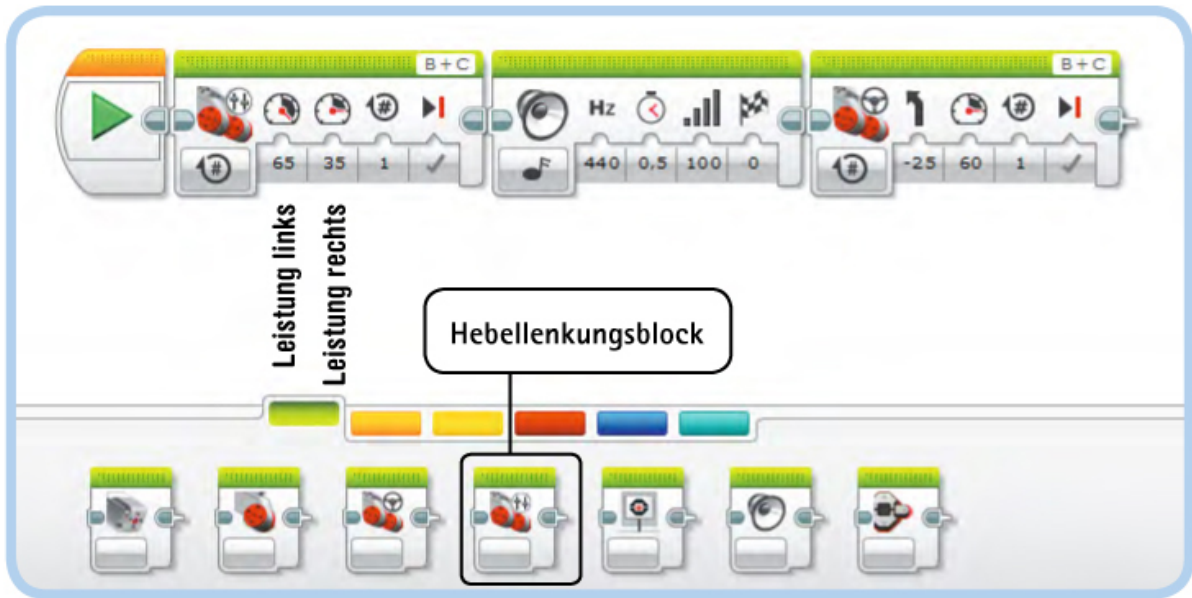


Abbildung 4-19: Das **Tank**-Programm: Der Roboter macht mit dem Hebellenkungsblock eine sanfte Drehung nach rechts. Dann hält er an, spielt einen Klang ab und bewegt sich wieder, diesmal mit einem Bewegungslenkungsblock.

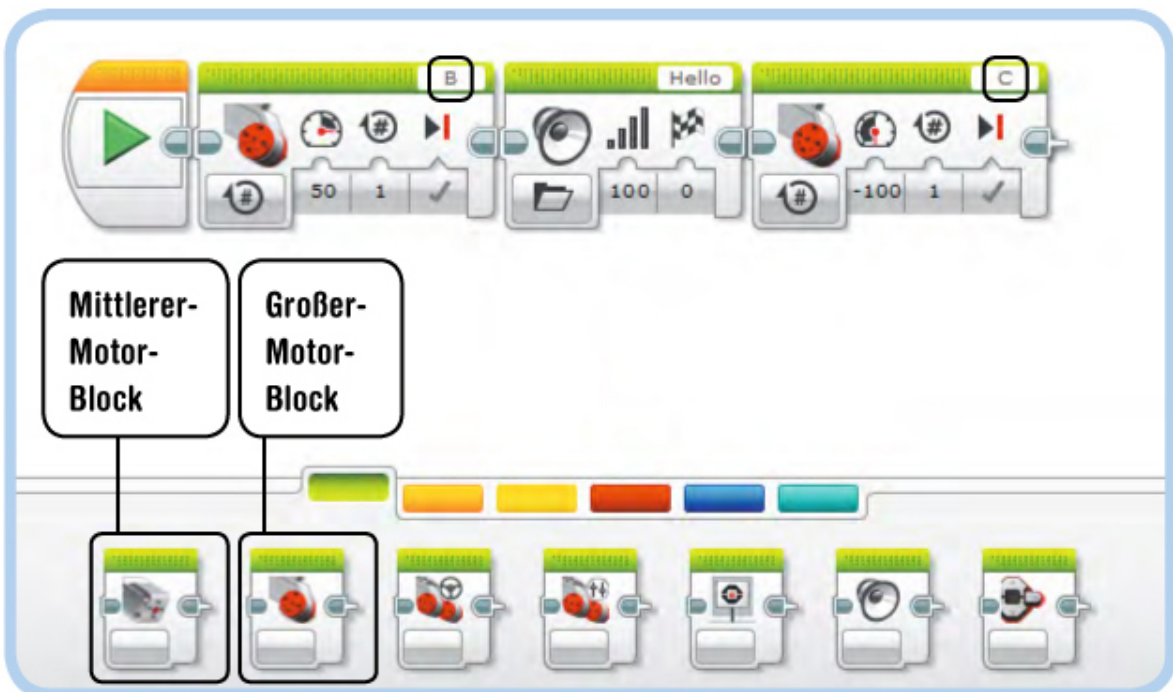


Abbildung 4-20: Das Programm **LargeMotor**: Der linke Motor dreht vorwärts, der Roboter sagt »Hello« und der rechte Motor dreht rückwärts.

SELBST ENTDECKEN 11: ZEIT, IM KREIS ZU FAHREN

Schwierigkeitsgrad: Zeit:

Kannst du den EXPLOR3R in einem Kreis mit 1 m Durchmesser fahren lassen? Dazu benötigst du nur einen einzigen Bewegungslenkungsblock. Wie konfigurierst du die Lenkung und wie lange sollten die Motoren laufen? Wie beeinflusst die Einstellung der Lenkung den Durchmesser des Kreises? Wenn du fertig bist, versuche, denselben Effekt mit dem Hebellenkungsblock zu erzielen.

SELBST ENTDECKEN 12: NAVIGATOR

Schwierigkeitsgrad:  Zeit: 

Erstelle ein Programm mit Bewegungslenkungsblöcken, die den EXPLOR3R in einem Kurs wie in [Abbildung 4-21](#) fahren lassen. Während der Bewegung sollte der Roboter Pfeile auf dem EV3-Display anzeigen, die seine Bewegungsrichtung darstellen. Am Ende sollte ein Stoppschild angezeigt werden. Zusätzlich zum Richtungspfeil sollte der Roboter auch sagen, in welche Richtung er fährt. Wie konfigurierst du die Wiedergabeart-Einstellung in den Klangblöcken?

HINWEIS Du findest alle Richtungsschilder aus [Abbildung 4-21](#) in der Bildliste unter der Einstellung Dateiname im Anzeigeblock.

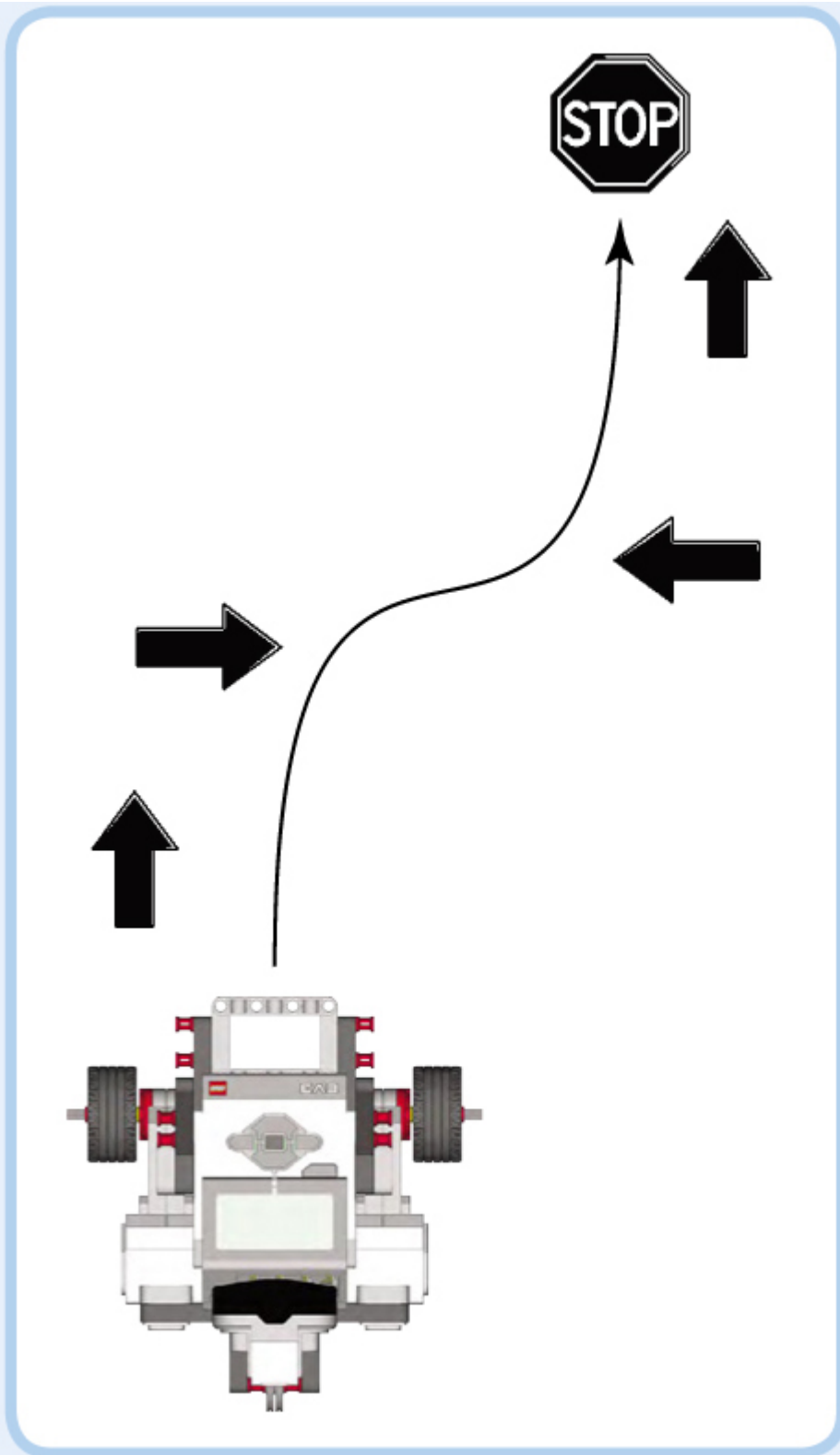


Abbildung 4-21: Der Fahrkurs und die Navigationshinweise für Selbst entdecken 12

SELBST ENTDECKEN 13: ROBOTÄNZER

Schwierigkeitsgrad:  Zeit: 

Lasse den EXPLOR3R Rhythmen und Töne abspielen (mittels Klangblöcken), während er sich in einem Zickzackkurs bewegt (mittels Bewegungslenkungsblöcken). Nach jeder Bewegung sollte der Roboter einen anderen Klang abspielen.

HINWEIS Probiere Wiederholen als Wiedergabeart-Einstellung in verschiedenen Klangblöcken aus.

SELBST KONSTRUIEREN 1: ROBOREINIGER

Bau:  Programmierung: 

Baue Lego-Teile an deinen Roboter an, sodass er ein Staubtuch vor sich auf den Boden halten kann. Dann erstelle ein Programm mit Bewegungslenkungsblöcken, das den Roboter bewegt, sodass er dein Zimmer reinigt. Statt eines Programms kannst du deinen Reinigungsroboter auch über die Fernsteuerung lenken, wie in [Kapitel 2](#) beschrieben. Saubermachen war noch nie so lustig!

SELBST KONSTRUIEREN 2: DER EXPLOR3R MACHT KUNST

Bau:  Programmierung: 

In dieser Aufgabe erweiterst du den EXPLOR3R-Roboter. Mit Lego-Teilen baust du eine Vorrichtung, über die der Roboter einen Stift halten kann. Während der Roboter über ein großes Stück Papier fährt, zeichnet er mit dem Stift Linien und Umrisse. Für den Anfang kannst du ihn die Acht zeichnen lassen, die du aus *Selbst konstruieren 8* auf [Seite 44](#) kennst.

Einen Stift fest am Roboter anzubauen ist zum Zeichnen von Beispielen sicherlich lustig, aber du bist eingeschränkt in dem, was du zeichnen kannst, da der Stift das Papier ständig berührt. Nutze den mittleren Motor aus dem EV3-Kasten, um den Stift anzuheben. Schließe diesen Motor mit einem Kabel an Ausgabeanschluss A an. Du kannst den Motor mit dem Block *Mittlerer Motor* steuern. Schaffst du es, dass der Roboter deinen Namen schreibt?

Wenn du fertig bist, mache ein Foto von deinem Design und füge es mit dem Inhalts-Editor ein, wie du in [Kapitel 3](#) gelernt hast.

Index

Zahlen und Symbole

12z-Zahnrad [265](#)

24z-Zahnrad [265](#)

36z-Zahnrad [265](#)

/ (Divisionsoperator) [332](#)

% (Modulo-Operator) [332](#)

* (Multiplikationsoperator) [332](#)

A

Achsen [105](#)

 Herausfallen verhindern [110](#)

 Kreuzlöcher [110](#)

 Länge bestimmen [9](#)

 Mit Verbindern verlängern [111](#)

 Rechtwinklig [132](#)

 Sensoren [119](#)

 Verdrehen verhindern [135](#)

Aktionsblöcke [27](#)

Aktualisieren der Firmware [25](#)

Aktualisieren-Schaltfläche [354](#)

Aktuelle Leistung [99](#), [100](#)

Am Ende bremsen [38](#), [164](#)

Änderungsmodus [67](#), [68](#), [73](#)

An für ... Umdrehungen [190](#)

Anhalten von Programmen [49](#)

Anschlussansicht [354](#)

Anschlusseinstellungen [37](#)

ANTY [171](#)

Bauen [173](#)–189

Farben in der Umgebung wahrnehmen [191](#)–193

Gehmechanismus [172](#)

Hindernissen ausweichen [190](#)–191

Motoren gegenüberliegend platzieren [190](#)

Nahrung finden [191](#)

Stückliste [173](#)

Anzeigeblock [42](#)

Anzeigefeld [42](#)

Bildschirm zurücksetzen [42](#)

Farbeinstellung [43](#)

Formen [42](#)

Fortschritt anzeigen [353](#)

Per Leitung übertragen [216](#)

Position auf dem Bildschirm [43](#)

Radius [43](#)

Textmodus [43](#)

Untermodi [42](#)–43

Anzeigemodus [33](#)

Aufkleber [5](#)

Ausführen von Programmen

Auf dem EV3-Stein [22](#)

Fehlerbehebung [352](#)–353

Manuell [28](#)

Nach dem Herunterladen [28](#)

Ausgabeanschlüsse [20](#)

Ausgabezahnrad [126](#)

Außerhalb-Modus [234](#)

Auswahl ausführen [29](#)

Auswahlwerkzeug [30](#)

B

Balken [105](#)

Anbaupunkte am großen Motor hinzufügen [135–136](#)

Erweitern [106](#)

Großer Motor [118](#)

Im rechten Winkel verbinden [111](#), [113](#)

Länge bestimmen [9](#)

Messen [106](#)

Parallel sichern [111](#)

Parallel verbinden [111](#), [112](#)

Sensoren [119](#)

Strukturen verstärken [107](#)

Winkelbalken [107–108](#)

Zahnräder flankieren [134](#)

Ballmagazin [119](#)

Ballschussgerät [119](#)

Batterien

Austauschen [21](#)

EV3, wiederaufladbar [4](#)

Fernbedienung [4](#)

Bereichsblock [234](#)

Berührungssensor [62](#)

Änderungsmodus [68](#), [73](#)

Hindernissen ausweichen [67](#)

Kombination mit anderen Sensoren [90–91](#)

- LAVA R3X [330](#)
- Messmodus [73](#)
- Motorstellung ermitteln [172](#)
- Schalterblock [69](#)
- Statusmodus [68](#)
- Stoßstange [62–65](#)
- Vergleichsmodus [68](#), [70](#), [73](#), [219](#)
- Berührungssensorblock [218](#)
- Beschleunigen eines Motors [220](#)
- Bewegungslenkungsblock [35](#)
 - Am Ende bremsen [38](#)
 - Beschleunigung [39](#)
 - Ein/Aus-Modi [45](#), [332](#)
 - Exakte Drehungen [39](#)
 - Leistungseinstellung [37](#)
 - Lenkungseinstellung [37](#)
 - Modi [37](#)
 - Verwenden [35–36](#)
- Bildschirm zurücksetzen (Anzeigeblock) [42](#)
- Blöcke (Programmierung) [35](#), [227](#) *Siehe auch die Bezeichnungen der einzelnen Blöcke*
 - Arrangieren [28](#)
 - Arten [27](#)
 - Ausgeführte Blöcke ansehen [353](#)
 - Ausgewählte Blöcke ausführen [29](#)
 - Duplizieren [31](#)
 - Eigene Blöcke *siehe* Eigene Blöcke
 - EV3-Steintasten [97–98](#)
 - Fehler beheben [351–352](#)
 - Hilfe [31–32](#)

Löschen [27](#)
Modi und Einstellungen [35, 37](#)
Multitasking
 Mehrere Startblöcke [56](#)
 Ressourcenkonflikte [57](#)
 Weiterleitungen aufteilen [56](#)
Negative Leistungswerte [38](#)
On-Brick-Programme *siehe* On-Brick-Programme
Platzierung im Programm [26](#)

Blockierte Motoren [101](#)

Bluetooth

 Dongles [357](#)
 Einstellungen [21](#)
 Programme herunterladen [356](#)
 Verbindung herstellen [357–358](#)
 WLAN [358](#)

Bonusmodelle [32](#)

D

Datei-Navigation [21, 30](#)

Datenblöcke [27, 227](#)

 Bereichsblock [234](#)
 Logische-Verknüpfungen-Block [233](#)
 Modi [233](#)
 Nicht-Modus [234](#)

Matheblock *siehe* Matheblock

Rundungsblock [235](#)

Textblock [235](#)

Vergleichsblock [232, 333](#)

Zufallsblock [231](#)

Datenleitungen [210–211](#)

- Blöcke in Schalterblöcken [222–223](#)
- Blöcke wiederholen [213](#)
- Eigene Blöcke [236](#)
- Logik [214](#)
- Logisches Array [215](#)
- Löschen [212](#)
- Matheblock [230](#)
- Mehrere Leitungen verwenden [212](#)
- Numerisch [214](#)
- Numerisches Array [215](#)
- Platzierung in Programmen [212](#)
- Schalterblöcke [221](#)
- Schleifenblock [220–221](#)
- Text [214](#)
- Typumwandlung [215–216](#)
- Warteblock [220](#)
- Wert auf EV3-Bildschirm anzeigen [216](#)
- Wertebereich [219–220](#)
- Werte in Datenleitungen einsehen [211](#)

Dauer (Klangblock) [41](#), [224](#)

Dekompilieren von Programmen [355](#)

Dekorieren von Robotern [119](#)

Demo-Programm [22](#)

Divisionsoperator (/) [332](#)

Doppelkegelzahnräder [128](#), [130–131](#)

Drahtlose Programmierung

- Bluetooth

Dongles [357](#)

Programme herunterladen [356](#)

Verbindung [357](#)–358

WLAN [358](#)

Drehen des Roboters

Exakte Drehungen [39](#)

Gegenteilige Richtung als erwartet [36](#)

Geschwungene Kurven [40](#)

Drehmodus [37](#)

Drehmoment [124](#)–125

Abwägen gegen Geschwindigkeit [127](#)

Erhöhen [125](#), [126](#)–127

Verringern [125](#)

Zahnräder [127](#)

Drehsensor [98](#)

Beine synchron halten [332](#)

Drehgeschwindigkeit [99](#)

Berechnen [100](#)

Messen [100](#)

Etch-A-Sketch-Programmkoordinaten [254](#)

Geregelte Geschwindigkeit [101](#)

Motorstellung [98](#)

Motorstellung zurücksetzen [99](#)

Radausrichtung steuern [163](#)

SNATCH3R-Greifer zurücksetzen [299](#)

Zweck [62](#)

Drehung

Geschwindigkeit [123](#)

Richtung in Getriebebezügen [126](#)

Richtung umkehren [135](#)

Dünne Elemente [114](#)

E

Eigene Blöcke [27](#)

Ausgänge [240](#), [242](#)

Bearbeiten [53](#), [239](#)

Datenleitungen [236](#)

Eingänge [236–238](#), [242](#)

Erstellen [53](#), [243](#)

Fehlerbehebung [351](#)

In mehreren Projekten verwenden [243](#)

Kopieren aus anderem Projekt [55](#)

Verwaltung im Projekt [53–54](#)

Verwenden [53–54](#)

Verwendungszwecke [243](#)

Ein/Aus-Modi (Bewegungslenkungsblock) [45](#)

Ein-/Ausschalten des EV3-Steins [21](#)

Eingabeanschlüsse [20](#)

Eingangszahnrad [126](#)

Einheitenraster [129](#)

Einstellungen (Register) [21](#)

Erweiterte Blöcke [27](#)

Erweiterter Modus des Matheblocks [229](#)

EV3-Programmiersoftware *siehe auch* Blöcke (Programmierung); Programme; Projekte

Aktualisieren [7](#)

Ausgewählte Blöcke ausführen [29](#)

Auswahlwerkzeug [30](#)

Blöcke platzieren [26](#)
Hardware-Seite [28](#)
Herunterladen [6–7](#)
Hilfe zu Blöcken [31–32](#)
Inhaltseditor [32](#)
Installieren [7](#)
Kommentarwerkzeug [30](#)
Programme auf den EV3-Stein herunterladen [25, 28](#)
Programme manuell ausführen [28](#)
Programmierbereich [27](#)
Programmierpalette [27](#)
Projekte und Programme [29](#)
Schwenkwerkzeug [30](#)
Startblock [27](#)
Starten [25](#)
Steuern von Robotern [6](#)
Symbolleiste [30](#)
Zoomwerkzeug [30](#)

EV3RSTORM [32](#)

EV3-Stein [3, 4](#) *siehe auch* On-Brick-Programme

Programme finden [30](#)

Programme herunterladen [25, 28](#)

Bluetooth [356](#)

WLAN [358](#)

Batterien [4](#)

Ein-/Ausschalten [21](#)

Fehlerbehebung *siehe* Fehlerbehebung: EV3-Stein

Firmware aktualisieren [355](#)

microSD-Karte [356](#)

- Neu starten [355](#)
- Programme ausführen [22](#)
- Speicherverwaltung [356](#)
- Statusleuchten [21](#), [44](#)
- Tasten [20–21](#)
 - Programmierung [97–98](#)

EXPLOR3R [9](#), [9](#)

- Bauen [10–18](#)
- Farbsensor anbringen [75–76](#)
- Geregelte und unregelmäßige Geschwindigkeit [101](#)
- Hindernissen ausweichen [67](#), [90–91](#)
- Kabel anschließen [19](#)
- Linien mit Farbsensor folgen [79](#)
- Linien sanft folgen [83–84](#)
- Mehrere Sensoren [90–91](#)
- Stoßdämpfer mit Berührungssensor [62–65](#)
- Stückliste [10](#)
- Teststrecke [77](#)

Externe Geräte [20](#)

F

Fahren

- Beschleunigung [39](#)
- Rückwärts [36](#), [38](#)

Farbeinstellung (Anzeigeblock) [43](#)

Farbmodus [75](#), [77](#)

Farbsensor [62](#), [75](#)

- Anschluss an EXPLOR3R [75–76](#)

- Farbmodus [75](#), [77](#)

- Farbtafel [77](#)
- Händeschütteln erkennen [345](#)
- Innerhalb einer farbigen Linie bleiben [77–78](#)
- Linien folgen [79–80](#)
- Linien sanfter folgen [83–84](#)
- Stärke des reflektierten Lichts [75](#), [81](#)
 - Linien sanfter folgen [83–84](#)
 - Schwellenwert [82](#)
- Stärke des Umgebungslichts [75](#), [85](#)
- Teststrecke [77](#)
- Umgebungsfarben erkennen
 - Abwesenheit von Farbe [191–192](#)
 - Auf Farben reagieren (ANTY) [192–193](#)

Farbsensorblock [217](#)

Farbtafel [77](#)

Fehlerbehebung

EV3-Stein

- Firmware aktualisieren [355](#)
- Hardware-Seite [354](#)
- Neustart [355](#)
- Speicherverwaltung [354–355](#)
- USB-Verbindung [355](#)
- Verbindungen [354](#)

Greifmechanismus (SNATCH3R) [301](#)

Kompilierungsfehler

- Fehlende Eigene Blöcke [351](#)
- Fehlende Variablendefinition [352](#)
- Programmierblockfehler [351–352](#)

Programme ausführen [352–353](#)

Umgekehrte Drehrichtung als erwartet [36](#)

USB-Verbindung [25](#)

Fernsteuerung [6](#)

Anwendung [22–23](#)

Batterien [4](#)

Infrarotsensor [92](#)

Programm für Formel-EV3-Rennwagen [166–167](#)

Programm für SNATCH3R [300](#)

Roboter steuern [6](#)

Fernsteuerungsmodus (Infrarotsensor) [92](#)

Firmware aktualisieren [25, 355](#)

Flankieren von Zahnrädern mit Balken [134](#)

Flexible Strukturen [114–115](#)

Formel-EV3-Rennwagenroboter [141](#)

Autonome Bewegung [168](#)

Bauen [142–162](#)

Fernsteuerprogramm [166–167](#)

Lenkung [163](#)

Links und rechts [164](#)

Testen [166](#)

Zentrieren [164–165](#)

Zurücksetzen [163](#)

Stückliste [142](#)

Formen (Anzeigeblock) [42](#)

G

Gehmechanismus von ANTY [172](#)

Geregelte Geschwindigkeit [101](#)

Gesamtübersetzungsverhältnis [126](#)

Geschwindigkeit

Anpassen über Datenleitungen [213](#)

Berechnen für Zahnräder [123](#)

Erhöhen und Verringern mit Zahnrädern [123–124](#)

Gegen Drehmoment abwägen [127](#)

Geregelt und nicht geregelt [101](#)

Verringern mit Schnecke [133](#)

Geschwungene Kurven [40](#)

Gleichheit prüfen [232](#)

Gleichung [229](#)

Gradmodus [37](#)

Greifmechanismus des SNATCH3Rs [263–265](#)

Größer-als-Vergleiche [232](#)

Großer Motor

Anbaupunkte hinzufügen [135–136](#)

Balken anschließen [118](#)

Drehgeschwindigkeit [99](#)

Drehgeschwindigkeit berechnen [100](#)

Geometrie [116](#)

Geschwindigkeitsregelung und Leistungsaufnahme [101](#)

Maximales Drehmoment [125](#)

Räder anschließen [115](#), [117](#)

Zahnräder anschließen [136](#)

Zwei Motoren über einen Rahmen verbinden [115](#)

Großer-Motor-Block [46](#)

H

Halbe Einheiten [114](#)

Zahnräder [129](#)

Händeschütteln erkennen [345](#)
Hardware-Seite [28](#), [66](#), [354](#)
Hebellenkungsblock [46](#)
Herunterladen
 Beispielprogramm [xxii](#)
 EV3-Programmiersoftware [6–7](#)
 Farbtafel [77](#)
 LEGO-Einheitenrasten [108](#)
 Teststrecke zum Verfolgen von Linien [81](#)
Herunterladen und ausführen [25](#)
Hilfe zu Programmierblöcken [31–32](#)
Hindernissen ausweichen [90–91](#), [190–191](#)
Hochladen von Dateien auf den Computer [355](#)
H-Rahmen [106](#)
Hubmechanismus des SNATCH3Rs [265](#)

I

Infrarot-Fernsteuerung *siehe* Fernsteuerung
Infrarotsensor [62](#)
 Fernsteuerungsmodus verwenden [92](#)
 Fähigkeiten [89](#)
 Fernsteuerung verwenden [92](#)
 Hindernissen ausweichen [90–91](#), [168](#)
 Nähemodus [89](#)
 Hindernissen ausweichen [90–91](#)
 Mit anderen Modi kombinieren [90–91](#)
 Sensormodi kombinieren [95](#)
 Signal-Nähe-Modus [93](#)
 Signal-Richtung-Modus [93–94](#)

Infrarotsensorblock [217](#)
Inhalts-Editor [32](#)
Inkrementieren von Variablen [249](#)
Installation der EV3-Programmiersoftware [7](#)
IR-Control-Anwendung [22](#)
IR-Fernsteuerung *siehe* Fernsteuerung

K

Kabel [20](#)
Kegelzahnräder [128](#), [130](#)–[131](#)
Keine Farbe [191](#)
Klangblock [40](#)

- Datei abspielen [40](#)
- Dauer [41](#)
- Fortschritt anzeigen [353](#)
- Lautstärke [40](#)
- Note [41](#)
- Wiedergabeart [40](#)–[41](#)

Kleiner-als-Vergleiche [232](#)
Kommentarblock [352](#)
Kommentare

- Nützlichkeit zur Fehlerbehebung [352](#)
- Werkzeug [30](#)

Kompilierungsfehler

- Fehlende Eigene Blöcke [351](#)
- Fehlende Variablendefinition [352](#)
- Programmierblockfehler [351](#)–[352](#)

Konstantenblock [245](#)

- Variablenblock im Vergleich [247](#)

Kontexthilfe [32](#)

Kopieren von Programmierblöcken [31](#)

Kreuzloch [110](#), [118](#)

Kugelzahnrad [128](#), [133](#)

L

Länge von Balken und Achsen [9](#)

Lautstärke (Klangblock) [40](#)

LAVA R3X [311](#)–[312](#)

Bauen

Beine [312](#)–[329](#)

Kopf und Arme [335](#)–[343](#)

Beine zurücksetzen [344](#)

Gehen

Beine synchronisieren [332](#)–[333](#)

Gehen bis zur Auslösung des Sensors [344](#)–[345](#)

Motor zur Ausgangsposition zurückdrehen [330](#)–[331](#)

Nach links drehen [334](#)

Testprogramm [334](#)

Zurücksetzen [330](#)

Hindernissen ausweichen [344](#)

Kopf und Arme steuern [344](#)

Reaktion auf ausgelöste Sensoren [345](#)

Stückliste [312](#)

LEGO-Einheiten [106](#)

Halbe Einheiten [114](#)

Raster [108](#)–[109](#)

LEGO MINDSTORMS EV3 Education Core [3](#)

LEGO MINDSTORMS EV3 Home Edition [3](#)

Leistung

Leistungsaufnahme und Geschwindigkeitsregelung [101](#)

Regeln der Motorgeschwindigkeit [37](#)

Lenkung

Bewegungslenkungsblock

Modi [37](#)

Verwenden [35–36](#)

Formel-EV3-Rennwagen

Links und rechts [164](#)

Zentrieren [164–165](#)

Zurücksetzen [163](#)

Lenkungseinstellung [37](#)

Lesemodus [247](#)

Logische Arrays [215](#)

Logische Datenleitungen [214](#)

Schalterblock [221](#)

Logischer Wert (Modus beim Beenden eines Schleifenblocks) [220–221](#)

Logische Variablen [246](#)

Logische-Verknüpfungen-Block [233](#), [345](#)

Modi [233](#)

Nicht-Modus [234](#)

Löschen von Datenleitungen [212](#)

M

Matheblock [228](#)

Divisionsoperator [332](#)

Erweiterter Modus [229](#)

Modulo-Operator [332](#)

Multiplikationsoperator [332](#)

M-Einheit [106](#), [114](#)

Messmodus [73](#)

 Sensorblöcke [217](#)

 Zeitgeberblock [235](#)

Messwerte in Berechnungen verwenden [228](#)

microSD-Karte

 Arbeitsspeicher hinzufügen [20](#)

 Im EV3 verwenden [356](#)

 Ordner [30](#)

Mission-Pad [5–6](#)

Mittlerer Motor

 Drehgeschwindigkeit berechnen [100](#)

 Drehmoment [125](#)

 Drehsensor [163](#)

 Geometrie [118](#)

 Zahnräder [136](#)

Mittlerer-Motor-Block [46](#)

Modi von Programmierblöcken [37](#)

Module [106](#)

Modulo-Operator (%) [332](#)

Monsterzähne [119](#)

Motoren

 Anschlusseinstellungen [37](#)

 Ausgabeanschlüsse [20](#)

 Ausgangspunkt zum Bauen [115](#)

 Bewegungslenkungsblock *siehe* Bewegungslenkungsblock

 Blockierung erkennen [101](#)

 Drehgeschwindigkeit [100](#)

 Drehsensor [62](#)

EV3 [3](#)

Geregelte und ungeregelte Geschwindigkeit [101](#)

Großer Motor *siehe* Großer Motor

Hebellenkungsblock [46](#)

Mittlerer Motor *siehe* Mittlerer Motor

Position

 Drehsensor [98](#)

 Zurücksetzen [99](#)

 Unterschiedliche Geschwindigkeiten [46](#)

Motorsteuerungsanwendung [22](#)

Motorumdrehungsblock [218](#)

Multiplikationsoperator (*) [332](#)

Multitasking

 Aufteilen von Weiterleitungen [56](#)

 Mehrere Startblöcke [56](#)

 Ressourcenkonflikte [57](#)

N

Nähemodus [89](#)

 Hindernissen ausweichen [90–91](#)

Namen von Programmen und Blöcken [352](#)

NETGEAR WNA1100 N150 (WLAN-USB-Adapter) [358](#)

Neustart des EV3-Steins [355](#)

Nicht-Modus (Logische-Verknüpfungen-Block) [233](#), [234](#)

Nocken [114](#)

Note (Klangblock) [41](#)

Numerische Datenleitung [214](#)

Numerischer Modus (Zufallsblock) [231](#)

Numerisches Arrays [215](#)

Numerische Variablen [246](#)

O

Oder-Modus (Logische-Verknüpfungen-Block) [233](#)

Offene Ansicht und Registeransicht [72](#)

On-Brick-Programme [25](#), [359](#)

- Ausführen [360](#)

- Blöcke [360](#), [361](#)

- Erstellen [359](#)

- Heruntergeladene Programme [25](#)

- Importieren [361](#), [363](#)

- Öffnen [361](#)

- Speichern [361](#)

- Verfügbare Blöcke [362](#)

O-Rahmen [106](#)

P

Parallele Balken

- Sichern [111](#)

- Verbinden [112](#)

Pins [9](#), [105](#)

Pins mit Reibung [9–10](#), [105](#)

Pins ohne Reibung [105](#)

- Flexible Strukturen [114](#)

- Pins mit Reibung [9–10](#)

Programmablaufblöcke [27](#)

Programme 3 *siehe auch* Blöcke (Programmierung); On-Brick-Programme

- Änderungen testen [353](#)

Anhalten [49](#)

Ausführen

 Auf dem EV3-Stein [22](#)

 Fehlerbehebung [352–353](#)

 Manuell [28](#)

 Nach dem Herunterladen [28](#)

Blöcke platzieren [26](#)

Dekompilieren [355](#)

Demo-Programm [22](#)

Eigene Blöcke verwenden [53–54](#)

Finden auf dem EV3-Stein [30](#)

Herunterladen auf den EV3-Stein [25, 28](#)

 Bluetooth [356](#)

 WLAN [358](#)

Projekte [29](#)

 Projekte erstellen [25](#)

 Schließen [29](#)

 Speichern [29](#)

 Umbenennen [30](#)

Programmierbereich [27](#)

Programmiersoftware *siehe* EV3-Programmiersoftware

Projekte

 Eigene Blöcke kopieren [55](#)

 Eigene Blöcke verwalten [53–54](#)

 Eigenschaften [30](#)

 Eines pro Roboter [30](#)

 Finden auf dem EV3-Stein [30](#)

 Gemeinsame Verwendung von Eigenen

 Blöcken [243](#)

Öffnen [29](#)
Programme [29](#)
Schließen [29](#)
Umbenennen [30](#)
Pulseinstellung [218](#)
Pythagoras [130](#)

R

RAC3 TRUCK [32–33](#)
Räder
 Großer Motor [117](#)
 Richtung steuern [163](#)
Radiuseinstellung (Anzeigeblock) [43](#)
Radius von Zahnrädern [129](#)
Rahmen [105](#), [106](#)
Raupenkettens [117](#)
Rechtwinklige Verbindungen
 Achen [132](#)
 Zahnräder [130–131](#)
Registeransicht [72](#)
Reibung von Zahnrädern [128](#)
Ressourcenkonflikte [57](#)
Roboter *siehe die einzelnen Roboter*
Rückwärts fahren [36](#), [38](#)
Rundungsblock [235](#)

S

Schalterblock [69](#)
 Änderungsmodus [73](#)

Blöcke hinzufügen [71](#)
Datenleitungen [221](#)
Datenleitungen innen am Block anschließen [222–223](#)
Farbsensor [77](#)
Gedrückte Taste ermitteln [98](#)
Konfigurieren [70](#)
Messen-Farbe-Modus [80](#)
Numerischer Modus [221](#)
Radausrichtung bestimmen [164](#)
Sensorwerte mit Schwellenwert vergleichen [82](#)
Signal-Richtung-Modus [94](#)
Vergleichsmodus [73](#)
Wiederholen [72](#)
Zufallsblock [231](#)

Scharniere mit Pins ohne Reibung [114](#)

Schleifenblock [50](#)

- Beenden im Modus Logischer Wert [220–221](#)
- Benennung [51](#)
- Datenleitungen [213](#), [220–221](#)
- Farbsensor [77](#)
- Größe ändern [51](#)
- Modi [50](#)
- Schalterblock [72](#)
- Schleifenindex [220](#)
- Sensoren [68–69](#)
- Unterbrechen
 - Von außen [223–224](#)
 - Von innen [223](#)
 - Warnung [224](#)

Vergleichsmodus [73](#)
Verschachteln [51–52](#)
Schleifen-Interrupt-Block [223](#)
Schließen von Programmen [29](#)
Schlupf von Zahnrädern [128](#)
Schnecke [128](#), [133–134](#), [265](#)
Schrägstrich (/) [332](#)
Schwellenwerte [82](#)
Schwenkwerkzeug [30](#)
Schwerter [119](#)
SD_Card-Ordner [356](#)
Sekundenmodus [37](#)
Sender *siehe* Fernbedienung
Sensorblöcke [27](#), [217–218](#) *siehe auch* die einzelnen Sensorblöcke
 Messmodus [217](#)
 Vergleichsmodus [218](#)
Sensoren [62](#)
 Änderungsmodus [73](#)
 Anschlusseinstellungen [67](#)
 Arten von Messungen [66](#)
 Bauen mit Sensoren [119](#)
 Berührungssensor *siehe* Berührungssensor
 Blöcke [66](#)
 Drehsensor *siehe* Drehsensor
 Eingabeanschlüsse [20](#)
 Eingebaute Sensoren [97](#)
 Farbsensor *siehe* Farbsensor
 Infrarotsensor *siehe* Infrarotsensor
 Mehrere verwenden [90–91](#)

- Messmodus [73](#)
- Schalterblöcke [69](#)
 - Blöcke hinzufügen [71](#)
 - Konfigurieren [70](#)
- Schleifenblock [68–69](#)
- Stein-Tasten [97–98](#)
- Typen [61](#)
- Vergleichsmodus [73](#)
- Warteblock [66–67](#)
- Werte einsehen [66](#)
- Service Set Identifier (SSID) [358](#)
- Servomotoren *siehe* Motoren
- Signal-Nähe-Modus [93](#)
- Signal-Richtung-Modus [93–94](#)
 - Messungen [303](#)
 - Signal-Nähe [219](#)
- SK3TCHBOT [200](#) *siehe auch* Datenleitungen
 - Bauen [200–209](#)
 - Koordinaten auf dem Bildschirm [254](#)
 - Stiftsteuerung
 - Bewegen ohne zu zeichnen [255](#)
 - Bildschirm zurücksetzen [257](#)
 - Radiergummi [255](#)
 - Stiftgröße [257](#)
 - Stückliste [200](#)
- Smartphone zur Robotersteuerung [6](#)
- SNATCH3R [263](#)
 - Bauen [266–298](#)
 - Fernsteuerungsprogramm [300](#)

- Greifmechanismus [263–265](#)
 - Fehlerbehebung [301](#)
 - Steuern [299](#)
- Hubmechanismus [265](#)
- IR-Sender suchen
 - IR-Käfer bauen [301–302](#)
 - Sender anheben und transportieren [308](#)
 - Suchalgorithmus [303–306](#)
 - Zum Sender fahren [307–308](#)
- Signal-Richtung-Messung [303](#)
- Stückliste [266–267](#)
- Sortieren von Elementen [5](#)
- Sortierkasten [5](#)
- Speicher-Browser [354](#)
- Speichern von Programmen [29](#)
- SSID (Service Set Identifier) [358](#)
- Stärke des reflektierten Lichts [75, 81](#)
 - Linien sanft folgen [83](#)
 - Schwellenwert [82](#)
- Stärke des Umgebungslichts [75, 85](#)
- Startblock [27](#)
 - Multitasking mit mehreren Startblöcken [56](#)
- Statusleuchten [21, 44](#)
- Statusmodus [68](#)
- Stein-Anwendungen [21](#)
- Stein *siehe* EV3-Stein
- Stein-Statusleuchten-Block [44, 218, 353](#)
- Steintastenblock [217](#)
- Steintastensensor [97–98](#)

Stirnzahnräder [128](#)

Stopper [105](#)

 Herausfallen von Achsen verhindern [110](#)

 Zu fest anschließen [122](#)

Stoßstange mit Berührungssensor [62–65](#)

Struktur

 Flexible Strukturen [114–115](#)

 LEGO-Einheitenraster [108–109](#)

Symbolleiste [30](#)

Synchronisieren von Beinen [332–333](#)

T

Tablet zur Robotersteuerung [6](#)

Tasten, EV3-Stein [20–21](#)

 Programmierung [97–98](#)

Technic-Produktreihe

 Elemente sortieren [5](#)

 EV3-Kasten [3](#)

Testen

 Anzeigeblock [42](#)

 Programmänderungen [353](#)

Teststrecke [77](#)

Textblock [235](#)

Textdatenleitung [214](#)

Textmodus (Anzeigeblock) [43](#)

Textvariablen [246](#)

Ton (Klangblock) [41](#)

TRACK3R [4](#)

Trennen-Schaltfläche [354](#)

U

- Umbenennen von Programmen [30](#)
- Und-Modus, Logische-Verknüpfungen-Block [233](#)
- Ungeregelte Geschwindigkeit [101](#)
- Ungeregelter-Motor-Block [101](#)
- Untermodi [42–43](#)
- USB-Verbindung
 - Externe Geräte [20](#)
 - Fehlerbehebung [25](#), [355](#)
 - Programme erstellen [25](#)
 - Programme laden [3](#)
 - Roboter programmieren [20](#)

V

- Variablen [245–246](#)
 - Definieren [246](#)
 - Durchschnitt berechnen [250](#)
 - Fehlende Definitionen [352](#)
 - Startwert festlegen [249](#)
 - Typen [246](#)
 - Werte ändern [249](#)
 - Werte erhöhen [249](#)
- Variablenblock [246–247](#)
 - Konstantenblock [247](#)
 - Variables definieren [247](#)
 - Verwendung in Programmen [247](#)
- Verbinder [105](#)
 - Achsen verlängern [111](#)

Balken im richtigen Winkel verbinden [111](#), [113](#)

Parallele Balken verbinden [111](#)

Verbindungen

Bluetooth [357–358](#)

Fehlerbehebung [354](#)

Verdrehen von Achsen [135](#)

Verfügbare Steine [354](#)

Vergleichsblock [232](#), [333](#)

Vergleichsmodus [73](#)

Änderungsmodus im Vergleich [67](#)

Berührungssensor [219](#)

Sensorblöcke [218](#)

Verringern der Geschwindigkeit [123–124](#), [126–127](#)

Verschachteln von Schleifenblöcken [51–52](#)

Verstärken von Strukturen [107](#)

W

Wahrscheinlichkeit von »wahr« [231](#)

Warteblock [49](#)

Änderungsmodus [73](#)

Berührungssensormodus [66](#)

Datenleitungen [220](#)

Farbsensor [77](#)

Sensoren [66–67](#)

Vergleichsmodus [73](#)

Weitere Roboter [32](#)

Weiterleitungen [28](#)

Aufteilen für Multitasking [56](#)

Wiederaufladbare Batterie [4](#)

Wiedergabeart (Klangblock) [40–41](#)

Winkelbalken [107–108](#)

 Zahnräder entlangführen [130](#)

WLAN (Programm herunterladen) [358](#)

X

XOR-Modus (Logische-Verknüpfungen-Block) [233](#)

Z

Zählerwert [249](#)

Zahnräder [105](#)

 12Z-Zahnrad [265](#)

 24Z-Zahnrad [265](#)

 36Z-Zahnrad [265](#)

 Ausgabegeschwindigkeit berechnen [123](#)

 Doppelkegelräder [128](#), [130](#)

 Drehmoment [124–125](#)

 Erhöhen [125](#), [126–127](#)

 Verringern [125](#)

 Einheitenraster [129](#)

 Entlang Winkelbalken anordnen [130](#)

 Flankieren mit Balken [134](#)

 Gesamtübersetzungsverhältnis [126](#)

 Geschwindigkeit erhöhen und verringern [123–124](#)

 Getriebezüge [121](#), [125–126](#)

 Halbe Einheiten [129](#)

 Im Kasten enthaltene Zahnräder [128](#)

 Kegelzahnrad [128](#), [130](#)

Kugelzahnrad [128](#), [133](#)
Motoranschluss [136](#)
Radius [129](#)
Rechtwinklige Achsen [132](#)
Rechtwinklige Verbindungen [130](#)–[131](#)
Reibung und Schlupf [128](#)
Schnecke [128](#), [133](#)–[134](#)
Stirnrad [128](#)
Übersetzungsverhältnis [123](#)
Umkehren der Drehrichtung [135](#)
Unpassende Kombinationen [130](#)
Verdrehen von Achsen verhindern [135](#)
Verhältnisrechner [129](#)
Verwendung [121](#)
Zähne [122](#)
Zeitgeberblock [218](#), [235](#)
Zoom-Werkzeug [30](#)
Zufallsblock [231](#)
Zuletzt verwendet [21](#)
Zurücksetzen-Modus (Zeitgeberblock) [235](#)
Zwischenzahnrad [126](#)