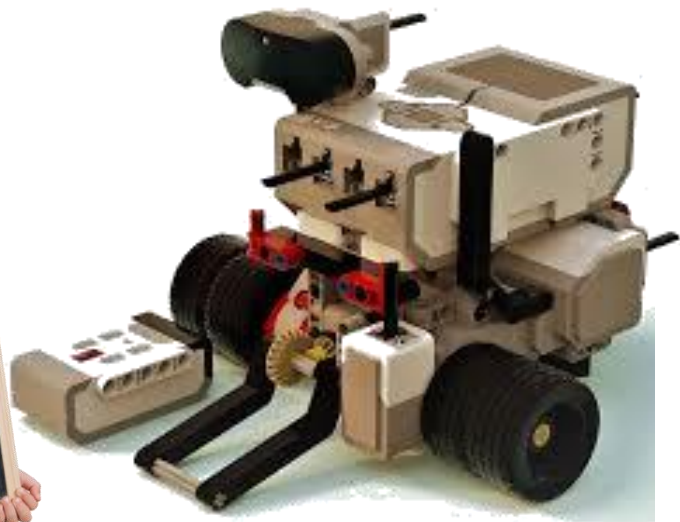


Roboterprogrammierung mit Lego Mindstorms EV3



Inhaltsverzeichnis

Inhaltsverzeichnis	1
Hardware:	2
Software:	3
Erste Aufgaben I.....	6
Erste Aufgaben II.....	10
Aufgaben – Wiederholungen / Schleife	13
Aufgaben – „Warten auf“ / Sensornutzung I	15
Aufgaben – „Warten auf“ / Sensornutzung I	16
Verwendung des Farb-/Lichtsensors	18
Sensornutzung – Kalibrieren des Lichtsensors	19
Verwendung des Gyro-/Kreiselsensors	21
Aufgaben – Sensornutzung „Warten auf“ oder Schleife	23
Aufgaben – Sensornutzung II.....	26
Schalter („Wenn dann - sonst“) am Beispiel „Linefollower“	28
Aufgaben – Linefollower	30
Aufgaben – Multitasking: Ausführen von zwei oder mehreren Programm(teil)en gleichzeitig	31
Aufgaben – Schleifen-Interrupt	33
Aufgaben – Erstellung und Verwendung von Blöcken.....	35

Hardware:

Lego Mindstorms Education Basis-Set

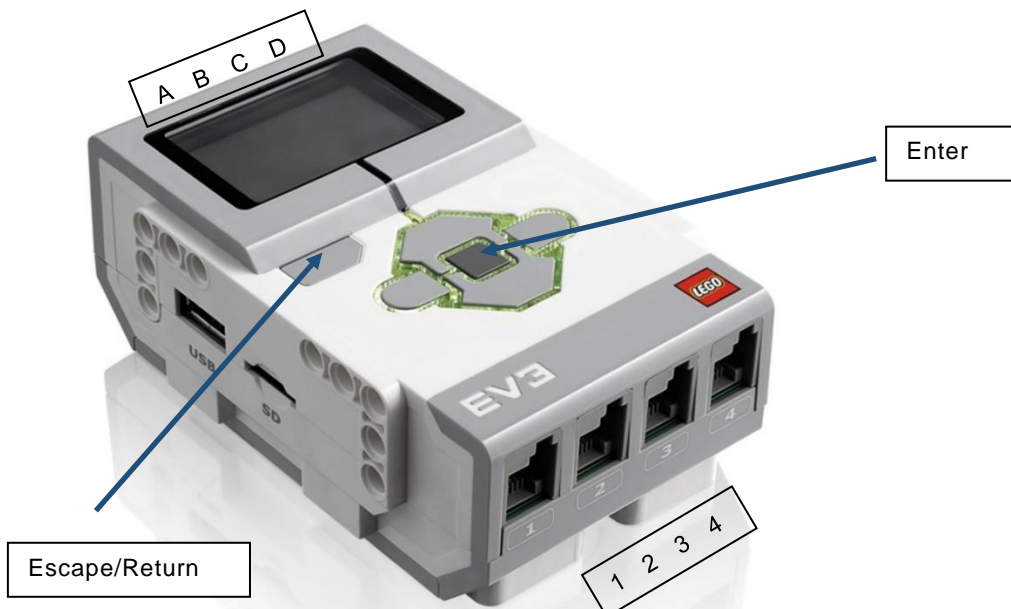


Bild: Wikimedia Commons

Vorbereitungen:

Die Schüler/innen bauen einen fahrbaren Roboter oder bekommen ihn zur Verfügung gestellt. Die Antriebsmotoren sind an den Ausgängen B und C angeschlossen, da es in der Software einen Programmblock für die Standardsteuerung gibt, der die Ausgänge B und C verwendet. Für Arbeitsmotoren stehen die Anschlüsse A und D zur Verfügung.

Für weiterführende Aufgaben werden Sensoren benötigt: Anschlüsse 1-4: Der Tastsensor ist nach vorne gerichtet. Der Farbsensor wird nach unten, knapp über dem Boden montiert. Er sollte gegen Umgebungslicht abgeschirmt werden.

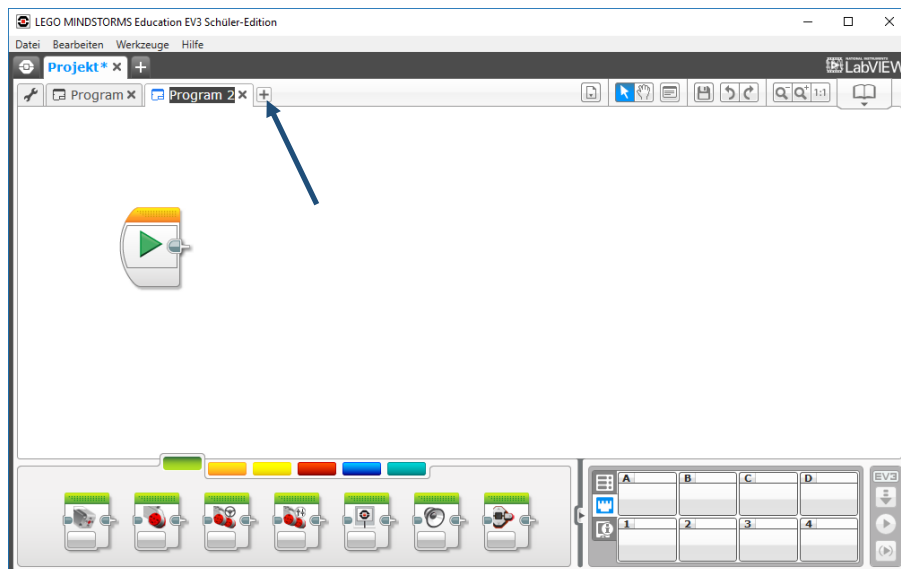
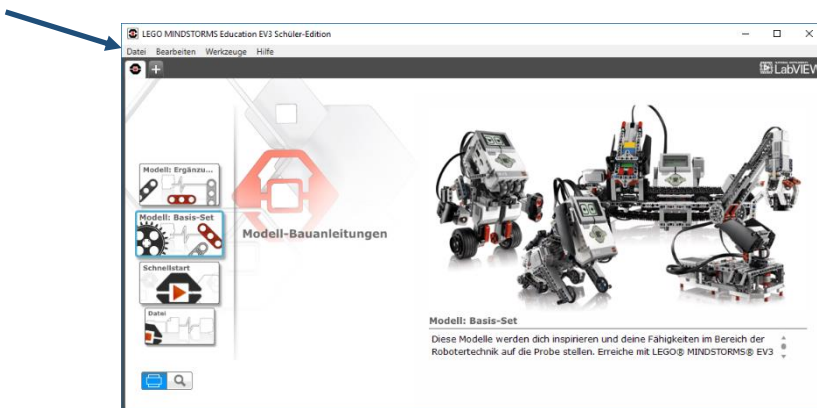
Es empfiehlt sich, an einer geraden Linie zu testen, ob der Roboter gerade fährt. Mögliche Fehlerquellen: Räder sitzen nicht fest, Reifen sind nicht ordnungsgemäß montiert, Motoren sind nicht ordnungsgemäß an den Ausgängen angesteckt.

Software:

Download unter:

<https://education.lego.com/de-de/downloads/retiredproducts/mindstorms-ev3-lab/software>

Programmstart: Datei/Neues Projekt/Programm



Mit dem „+“ können mehrere Programme in einem Projekt erstellt werden. Diese können mittels Doppelklick umbenannt werden.

Speichern: „Datei/Projekt speichern unter“ **Namen vergeben!**

Roboterprogrammierung mit Lego Mindstorms EV3

Bei der Programmierung mit der Lego Mindstorms EV3 Education-Software werden grafische Programmierblöcke aneinandergereiht und deren Parameter eingestellt.

Die tatsächliche Programmierung übernimmt die Software im Hintergrund.

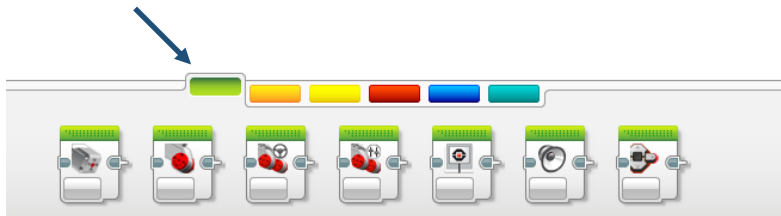
Jedes Programm beginnt mit dem **Startblock**, der beim Öffnen eines neuen Programmes automatisch erscheint.



Verschiedene **Block-Kategorien** stehen zur Verfügung:

Jeder Block hat oben einen Farbstreifen, an dem man erkennt, aus welcher Block-Kategorie er stammt.

➤ **Aktionsblöcke:**

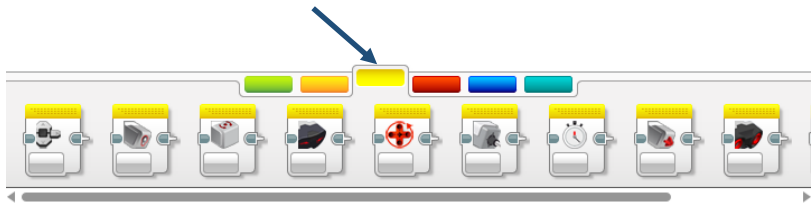


➤ **Ablaufregelung:**

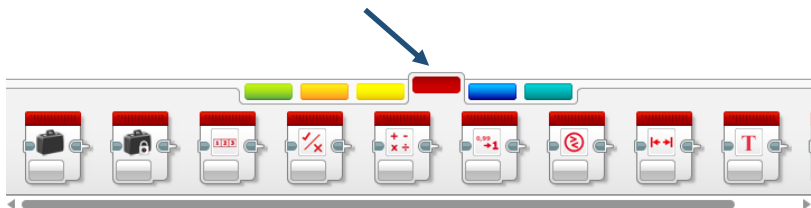


Roboterprogrammierung mit Lego Mindstorms EV3

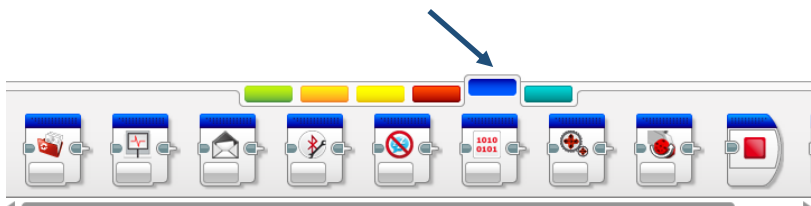
➤ Sensor-Blöcke:



➤ Blöcke für Datenoperationen:

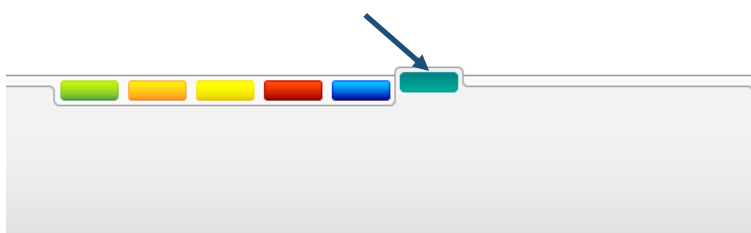


➤ Blöcke für den Erweiterungsmodus:

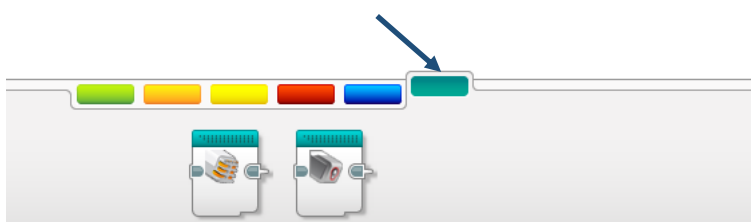


➤ Eigene Blöcke:

Diese Blockkategorie ist zunächst leer.

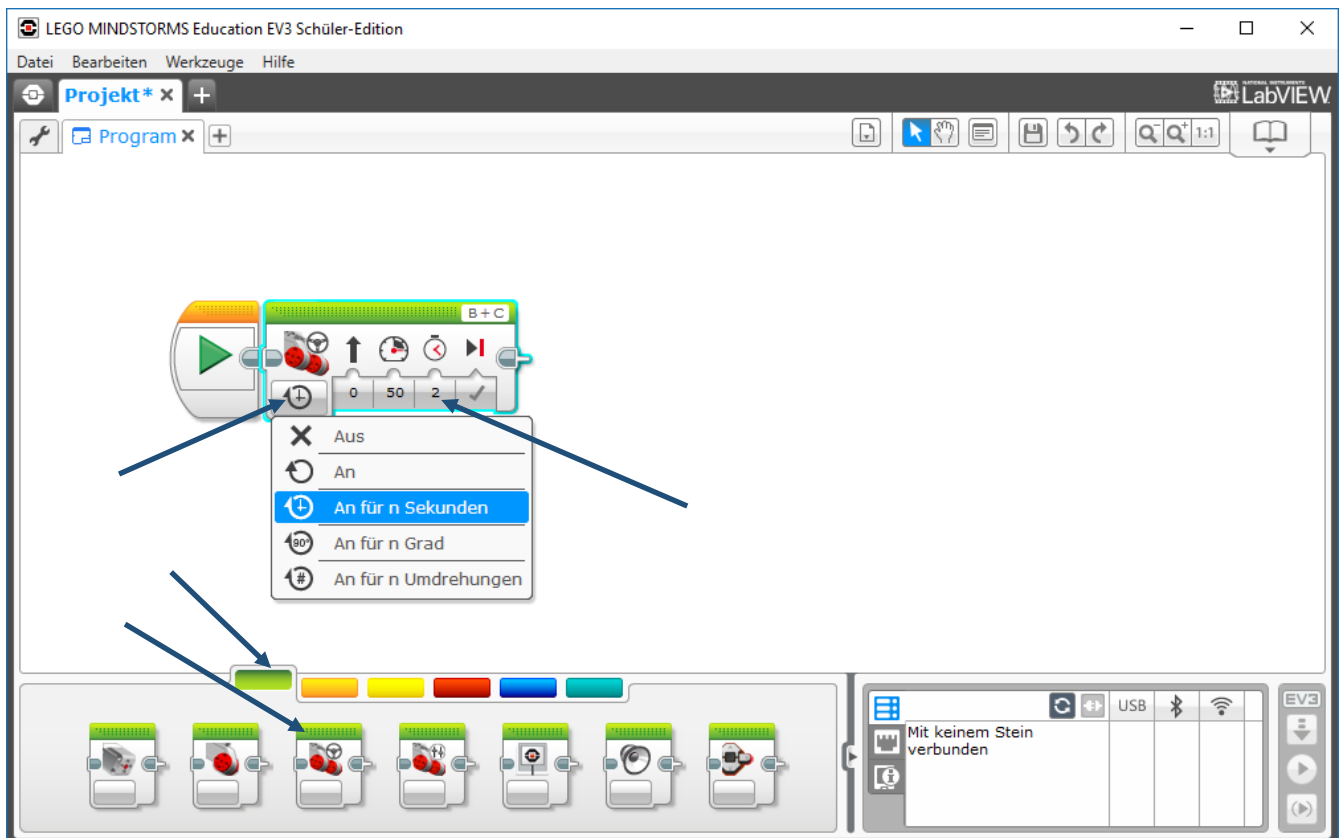


Erst wenn eigene Blöcke gespeichert wurden, werden sie hier angezeigt. s.s.



Erste Aufgaben I

Der Roboter fährt 2 Sekunden geradeaus vorwärts und bleibt dann stehen.



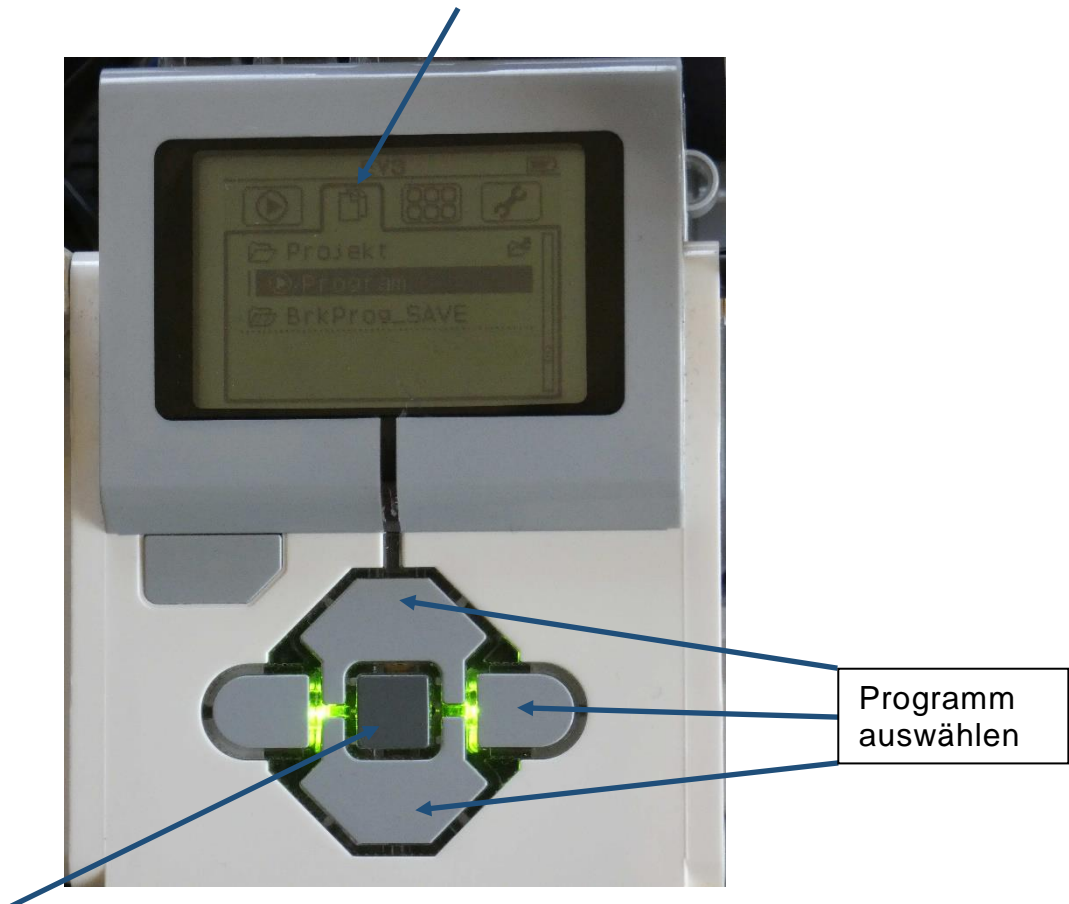
Der Block „Standard-Steuerung“ wird ausgewählt und mittels Drag-and-drop an den Startblock angedockt. Danach werden die Parameter des Blockes entsprechend der Aufgabenstellung geändert.

Zur **Übertragung des Programmes** muss der Roboter an der USB-Schnittstelle mit dem PC verbunden werden. Durch einen Klick auf den Download-Pfeil wird das Programm auf den Stein übertragen.



Roboterprogrammierung mit Lego Mindstorms EV3

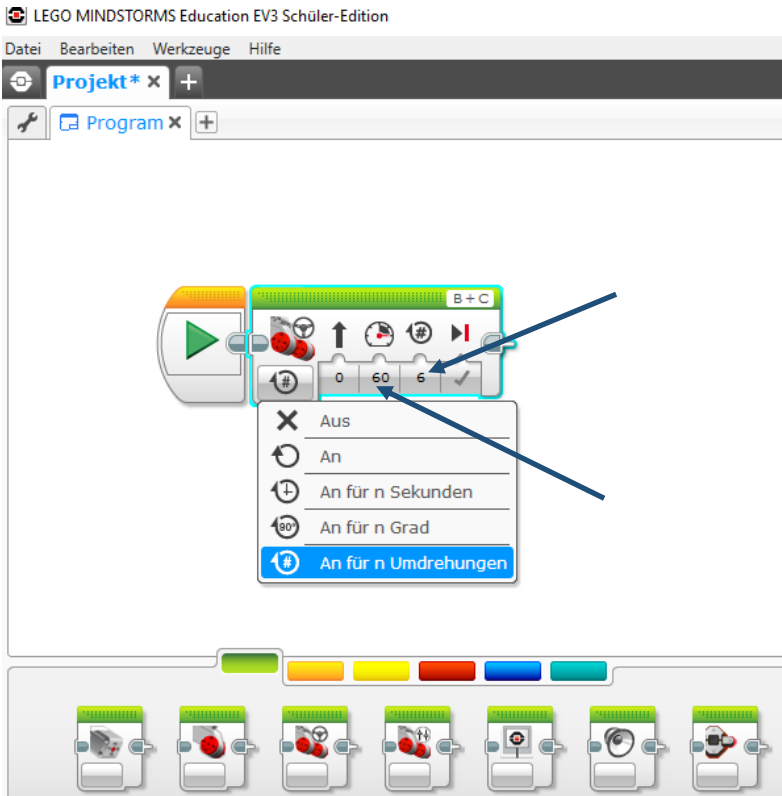
Am Roboter muss mittels Tasten das Programm ausgewählt werden: Zuerst wird das Ordnersymbol angewählt, dort das Projekt markiert und mit „Enter“ bestätigt, → die Programme des Projektes werden aufgelistet. Nun wird das gewünschte Programm markiert.



Die Enter-Taste in der Mitte startet das Programm.

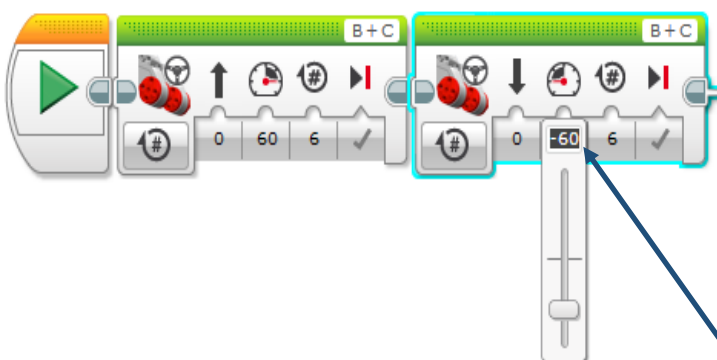
Roboterprogrammierung mit Lego Mindstorms EV3

Der Roboter fährt 6 Umdrehungen mit 60% seiner Geschwindigkeit geradeaus vorwärts und fährt dann zum Ausgangspunkt rückwärts zurück.



„Umdrehungen“ meint die Anzahl der Umdrehungen, die die Antriebsmotoren verrichten.

Die Leistung/Geschwindigkeit wird mittels Schieberegler eingestellt.

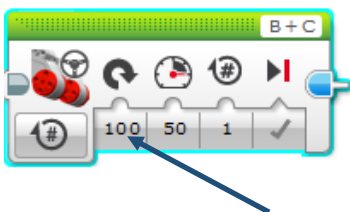
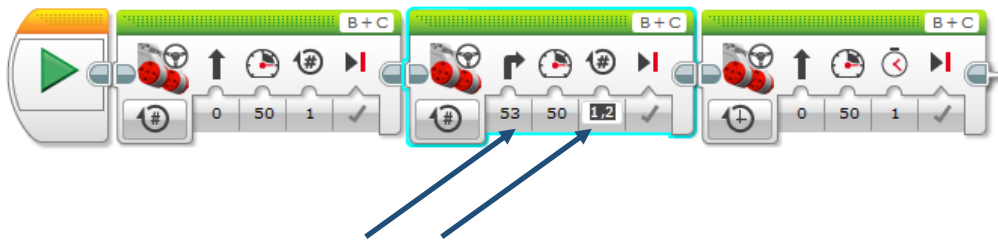


Zum Rückwärtsfahren erhält die Leistung einen negativen Wert.

Roboterprogrammierung mit Lego Mindstorms EV3

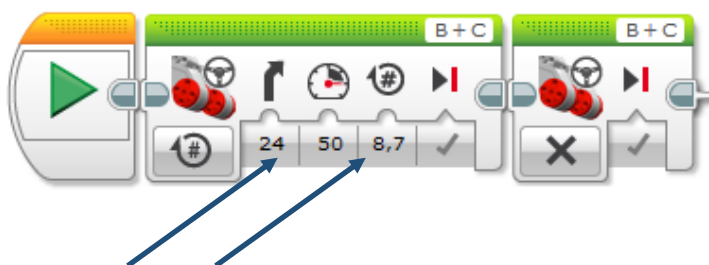
Der Roboter fährt 5 Umdrehungen geradeaus vorwärts, dreht sich um ca. 90° nach rechts und fährt dann eine weitere Sekunde geradeaus vorwärts. Der Roboter dreht im rechten Winkel.

Die **Werte für die Kurven** oder das Wenden sind von den verwendeten Rädern und vom Radstand abhängig. → Sie müssen durch Versuche ermittelt werden.



Wird der „Dreh-Wert auf +100/-100 gestellt, so dreht der Roboter am Stand.

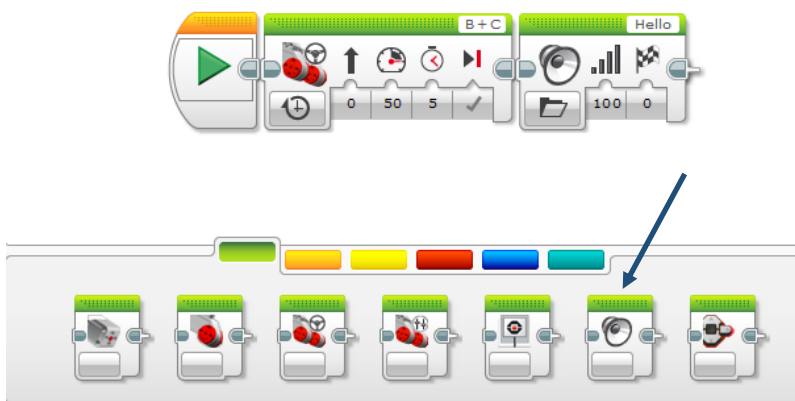
Der Roboter fährt so vorwärts, dass er einen Kreis (mit beliebigem Durchmesser) im Uhrzeigersinn beschreibt. Wenn er zum Ausgangspunkt zurückgekehrt ist, bleibt er stehen.



Erste Aufgaben II

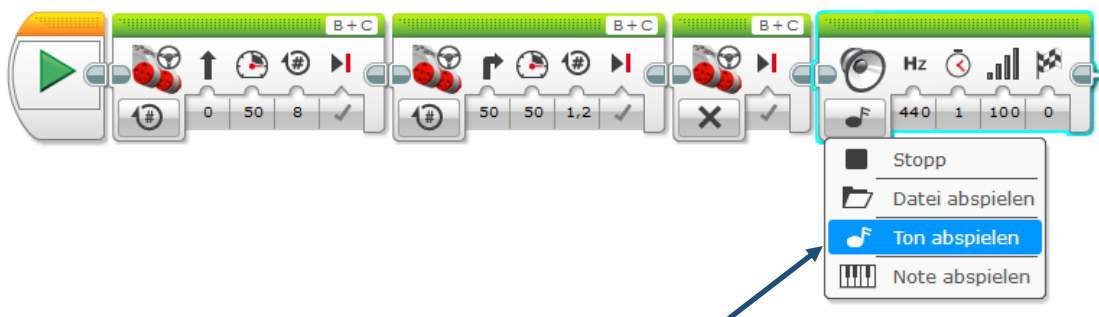
Der Roboter fährt 5 Sekunden lang geradeaus vorwärts. Dann bleibt er stehen und sagt „Hello“.

Bei den Aktions-Blöcken gibt es einen Block „Klang“. Ohne weitere Einstellungen gibt er das Wort „Hello“ aus.



Der Roboter fährt 8 Umdrehungen geradeaus vorwärts, dreht sich um 90°, bleibt dann stehen und gibt einen Ton von sich.

Öffnet man den Klang-Block, kann zwischen „Datei abspielen“, „Ton abspielen“ und „Note abspielen“ gewählt werden.

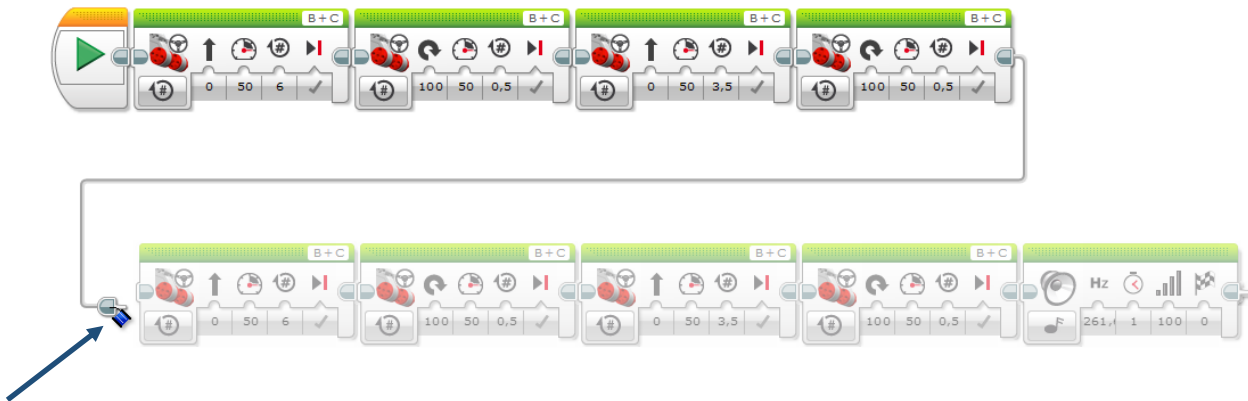


Roboterprogrammierung mit Lego Mindstorms EV3

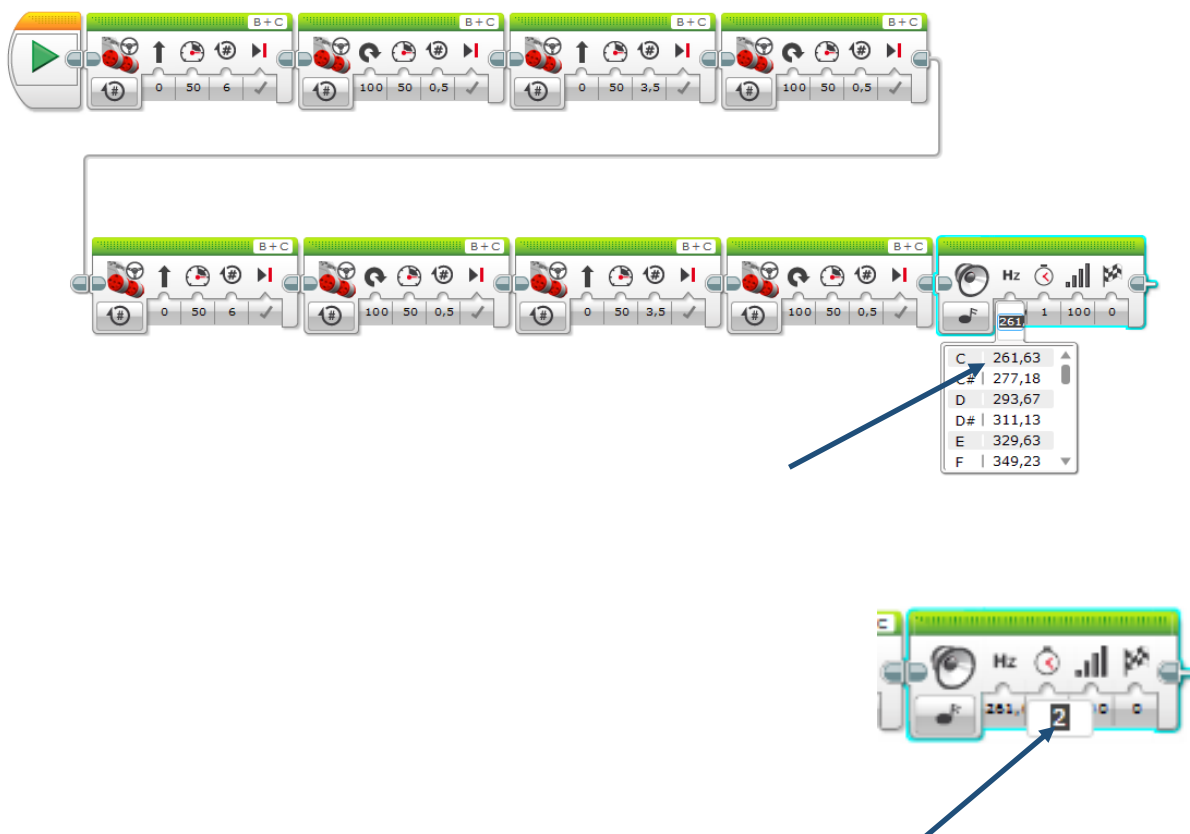
Der Roboter fährt ein Rechteck:

Länge: 6 Umdrehungen, Breite: 3,5 Umdrehungen

Wenn er fertig ist, gibt er 2 Sekunden lang den Ton „C“ von sich.



Bei langen Programmsequenzen können neue Zeilen begonnen werden:
Dann müssen die Enden händisch verbunden werden.



Roboterprogrammierung mit Lego Mindstorms EV3

Der Roboter fährt mit 40% Geschwindigkeit ein Quadrat mit einer Seitenlänge von 2,5 Umdrehungen.

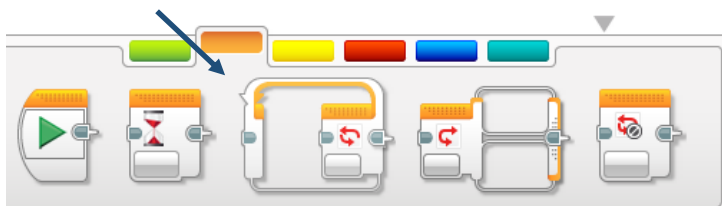
Wenn er fertig ist, spielt er eine kurze Melodie.

Kommentare helfen, sich im Programm schneller zurecht zu finden. Leider kann die Schriftgröße nicht angepasst werden.

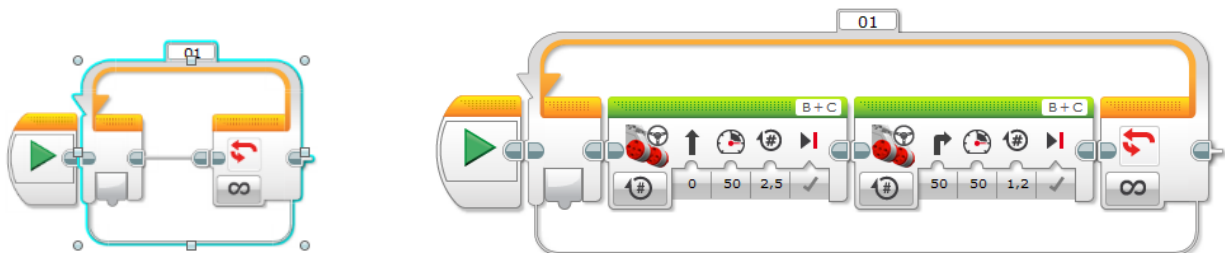
Aufgaben – Wiederholungen / Schleife

Der Roboter fährt ein Quadrat mit 2,5 Umdrehungen Seitenlänge. Benutze für die Programmierung die Schleifenfunktion!

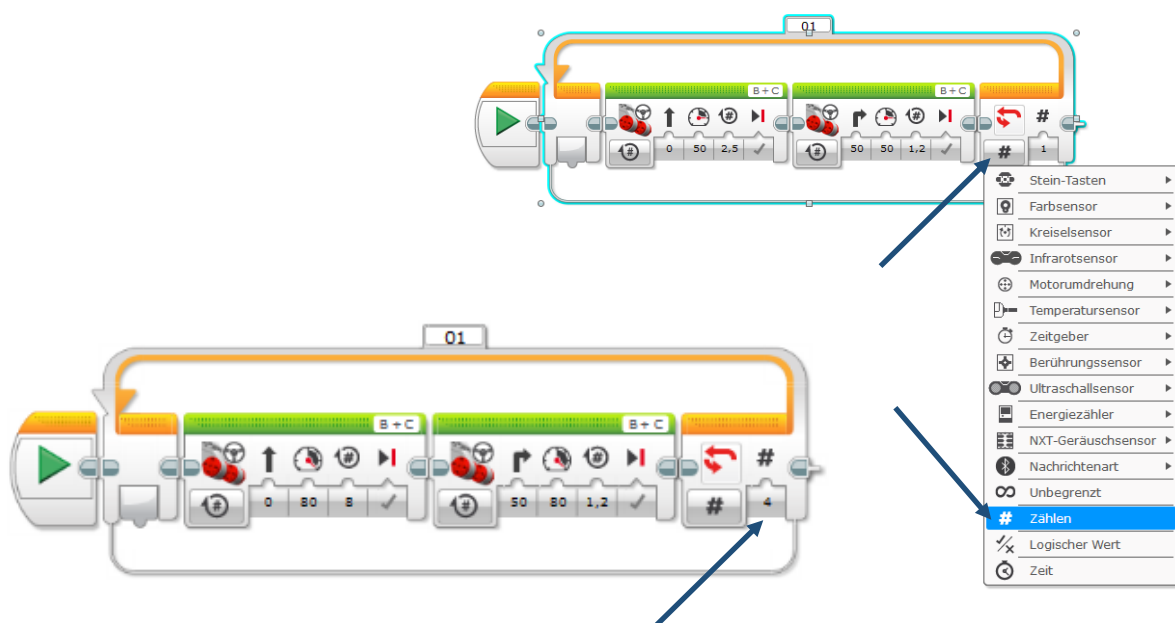
Für das Erstellen einer Schleife wird aus der Block-Kategorie „Ablauf-Regelung“ der „Schleifen-Block“ gewählt.



In die Schleife werden die Aktions-Blöcke hineingezogen.



Standardmäßig erscheint eine Endlosschleife. Bei diesem Beispiel wird für das Beenden der Schleife „Zählen“ gewählt. Danach kann die Anzahl der Wiederholungen eingestellt werden.



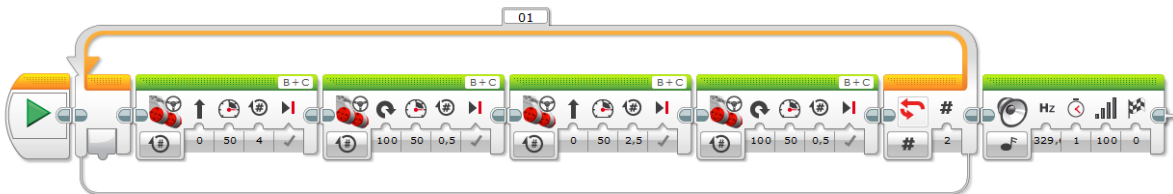
Roboterprogrammierung mit Lego Mindstorms EV3

Der Roboter soll ein Rechteck fahren:

Länge: 4 Umdrehungen, Breite: 2,5 Umdrehungen

Benutze für die Programmierung die Schleifenfunktion!

Wenn er wieder in der Ausgangsstellung angekommen ist, gibt er einen Ton von sich.

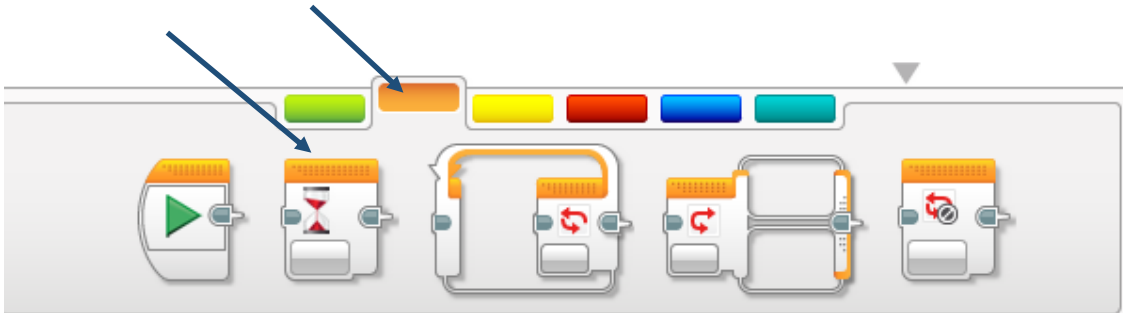


Der Roboter fährt mit 70% seiner Geschwindigkeit so vorwärts, dass er 5 Kreise gegen den Uhrzeigersinn beschreibt. Am Ende jedes Kreises soll er stehen bleiben und 2 Sekunden lang eine Note (C5) spielen.



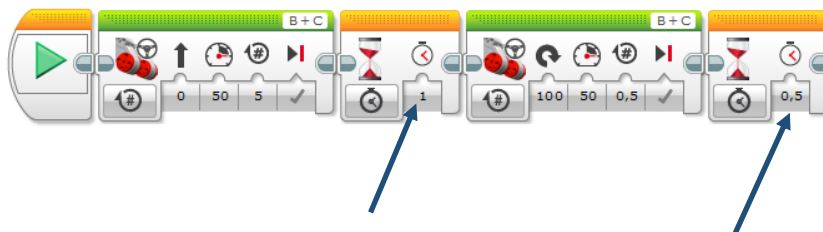
Aufgaben – „Warten auf“ / Sensornutzung I

In der Block-Kategorie „Ablauf-Regelung“ befindet sich der „Warten-Block“.



Standardmäßig ist der „Warten-Block“ auf Sekunden eingestellt.

Der Roboter fährt 5 Umdrehungen geradeaus vorwärts, wartet 1 Sekunde, dreht sich um ca. 90°, bleibt stehen und wartet 0,5 Sekunden...



*Der Roboter fährt 5 Umdrehungen geradeaus vorwärts, wartet 1 Sekunde, dreht sich um ca. 90°, bleibt stehen und wartet 0,5 Sekunden; **dann bellt er.***



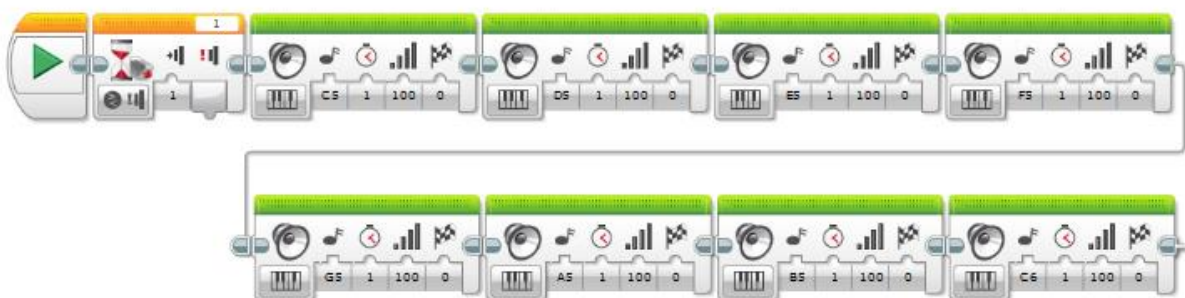
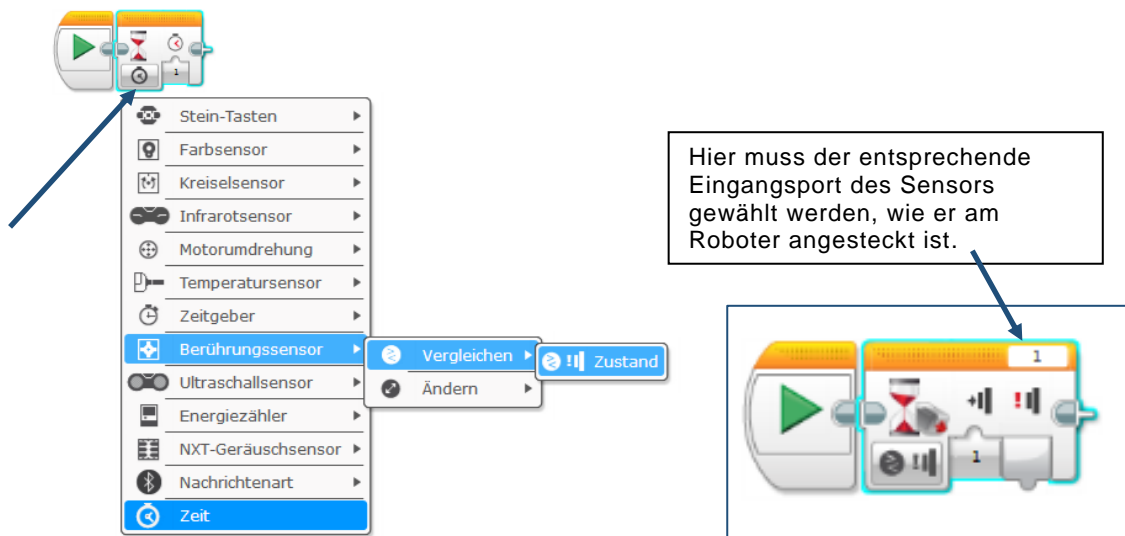
Mit Klick auf „Hello“ können verschiedenste Klänge ausgewählt werden.

Aufgaben – „Warten auf“ / Sensornutzung I

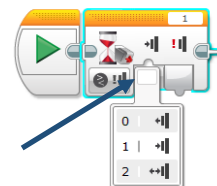
Der Warten-Block kann auf Sensor-Werte reagieren:

Der Roboter spielt nach Drücken des Tastsensors eine kurze Melodie (z.B.: Tonleiter).

Klickt man im „Warten-Block“ auf die Uhr, öffnet sich ein Drop-down-Menü: Hier kann der gewünschte Sensor ausgewählt werden, auf dessen Input der Roboter reagieren soll.



Beim Berührungssensor werden 3 Zustände unterschieden:



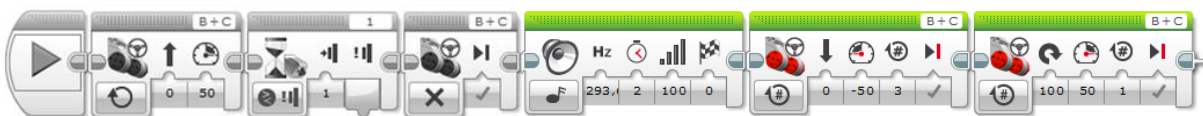
Roboterprogrammierung mit Lego Mindstorms EV3

Der Roboter fährt solange geradeaus vorwärts, bis er mit dem Tastsensor auf ein Hindernis stößt, dann bleibt er stehen ...

Dazu werden die Motoren zunächst eingeschaltet. Nach dem entsprechenden Input-Signal des Berührungssensors wieder ausgeschaltet.

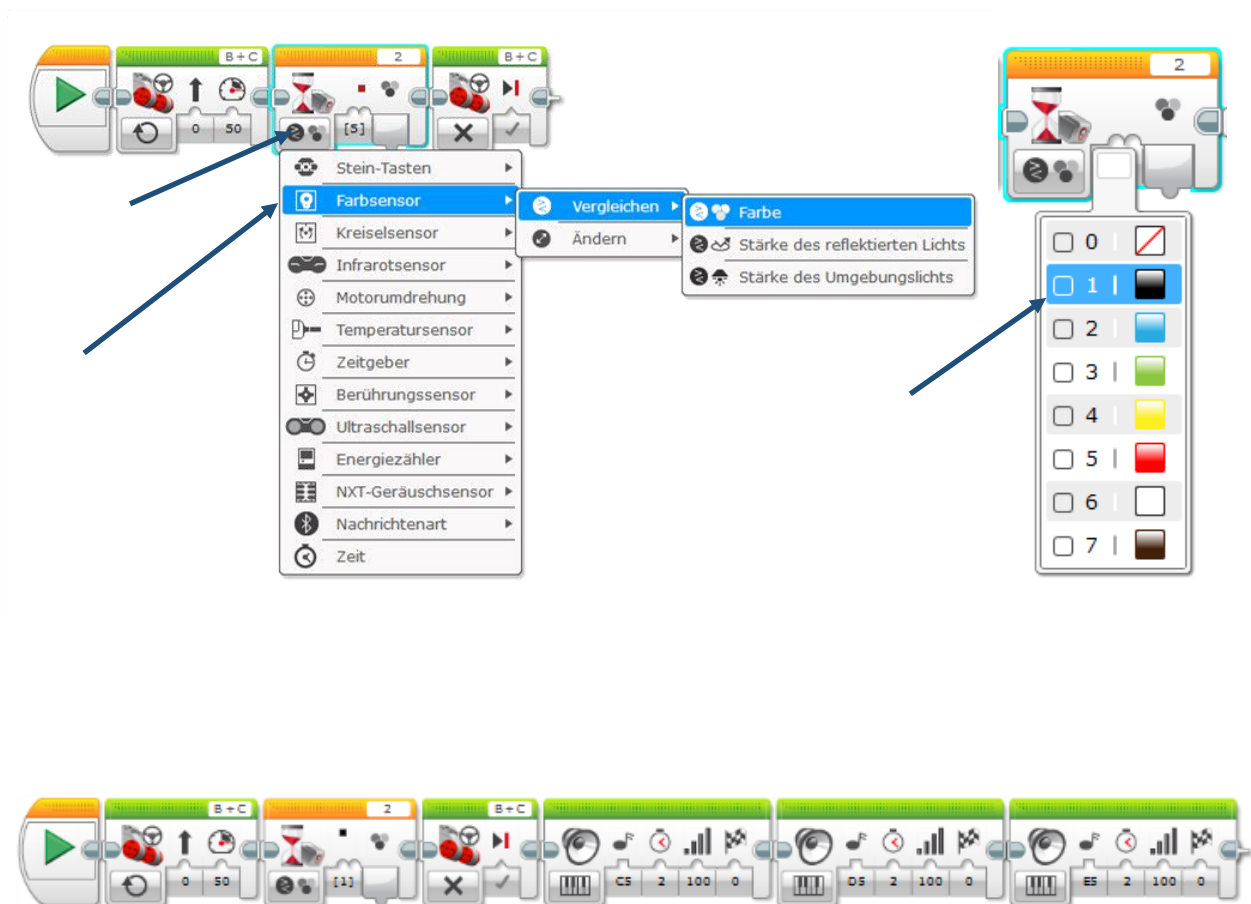


Der Roboter fährt solange geradeaus vorwärts, bis er mit dem Tastsensor auf ein Hindernis stößt, dann bleibt er stehen, gibt 2 Sekunden lang einen Warnton von sich, fährt 3 Umdrehungen rückwärts und dreht sich dann um, dass er in Richtung des Ausgangspunktes schaut.



Verwendung des Farb-/Lichtsensors

Der Roboter fährt so lange geradeaus vorwärts, bis er zu einer schwarzen Linie kommt. Dann bleibt er stehen und spielt eine kurze Melodie.



Wenn die Auswahl von „Schwarz“ nicht zum gewünschten Ziel führt, (weil die Linie zu blass ist,) können auch „Schwarz“ + „Blau“ oder „Schwarz“ + „Braun“ gewählt werden.

Jedenfalls muss das Hakerl bei „Rot“ entfernt werden.

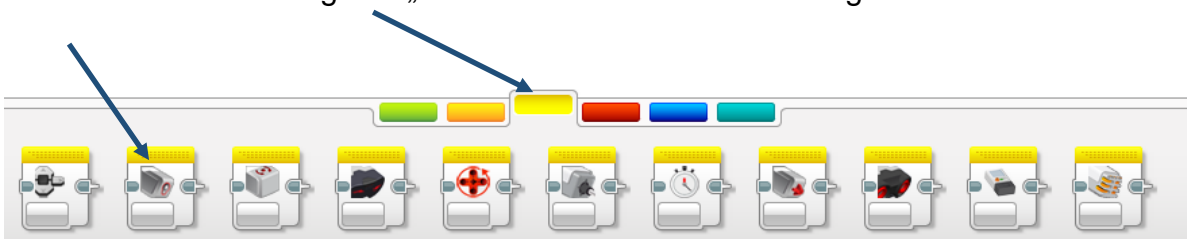
Alternativ können die vor Ort gemessenen Lichtwerte (Stärke des reflektierten Lichtes) für die Programmierung verwendet werden.

Sensornutzung – Kalibrieren des Lichtsensors

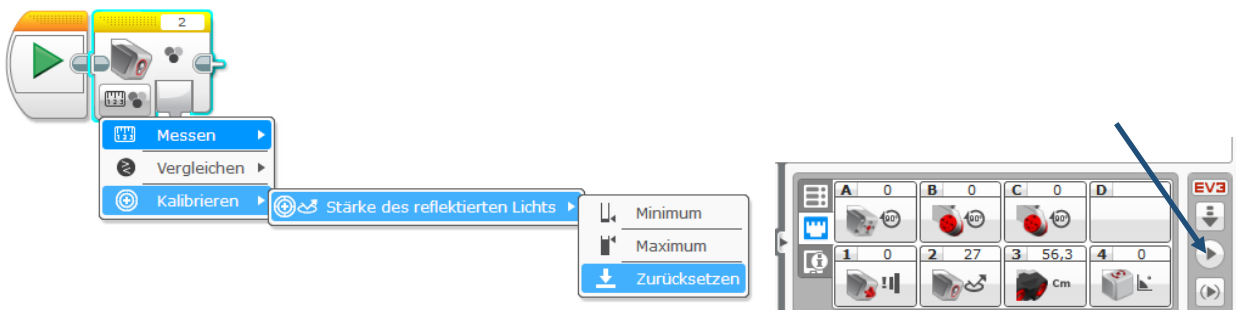
Wenn bei Verwendung von Farbsensoren nicht das gewünschte Ergebnis erzielt wird, muss der Sensor kalibriert (auf die Lichtverhältnisse des Raumes angepasst) werden. (Auch Geräuschsensoren müssen kalibriert werden.)

Dazu muss der Roboter mit dem Rechner verbunden sein:

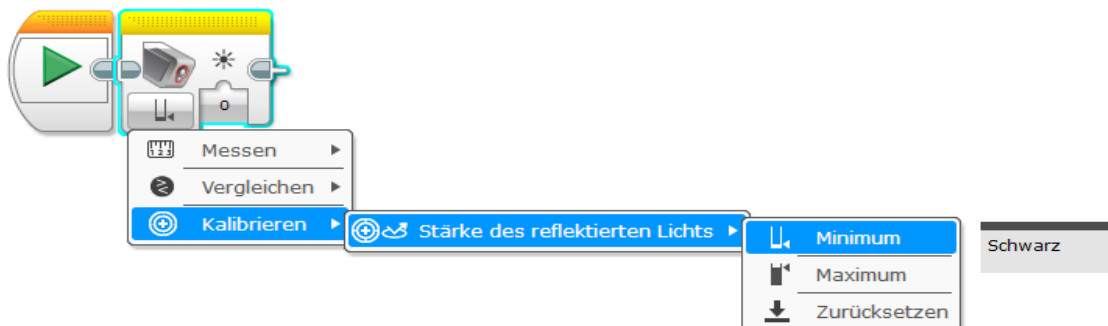
Es wird aus der Kategorie „Sensor“ der Farbsensor ausgewählt.



Zuerst wählt man „Zurücksetzen“, um gespeicherte Lichtwerte zu löschen. Durch das Klicken auf den „Ausführen-Pfeil“, wird die Aktion am Brick ausgeführt.

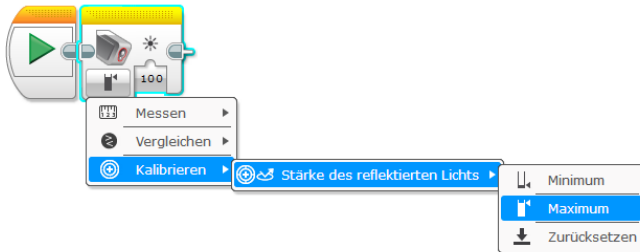


Zunächst stellt man den Farbsensor auf eine schwarze Unterlage. „Minimum“ entspricht „Schwarz“. (+ Klick auf den „Ausführen-Pfeil“, dass der Vorgang am Brick ausgeführt wird.)

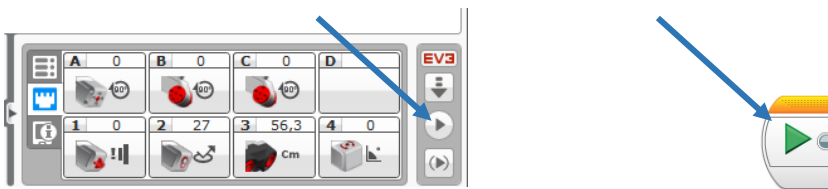


Roboterprogrammierung mit Lego Mindstorms EV3

Dann wird der Farbsensor auf eine weiße Unterlage gestellt: Maximales Licht entspricht „Weiß“.



Nicht vergessen: Erst die Programmübertragung und -ausführung führt die jeweiligen Befehle aus! (z.B.: durch Klicken auf den „Ausführen-Pfeil“ oder auf den Startbutton.)



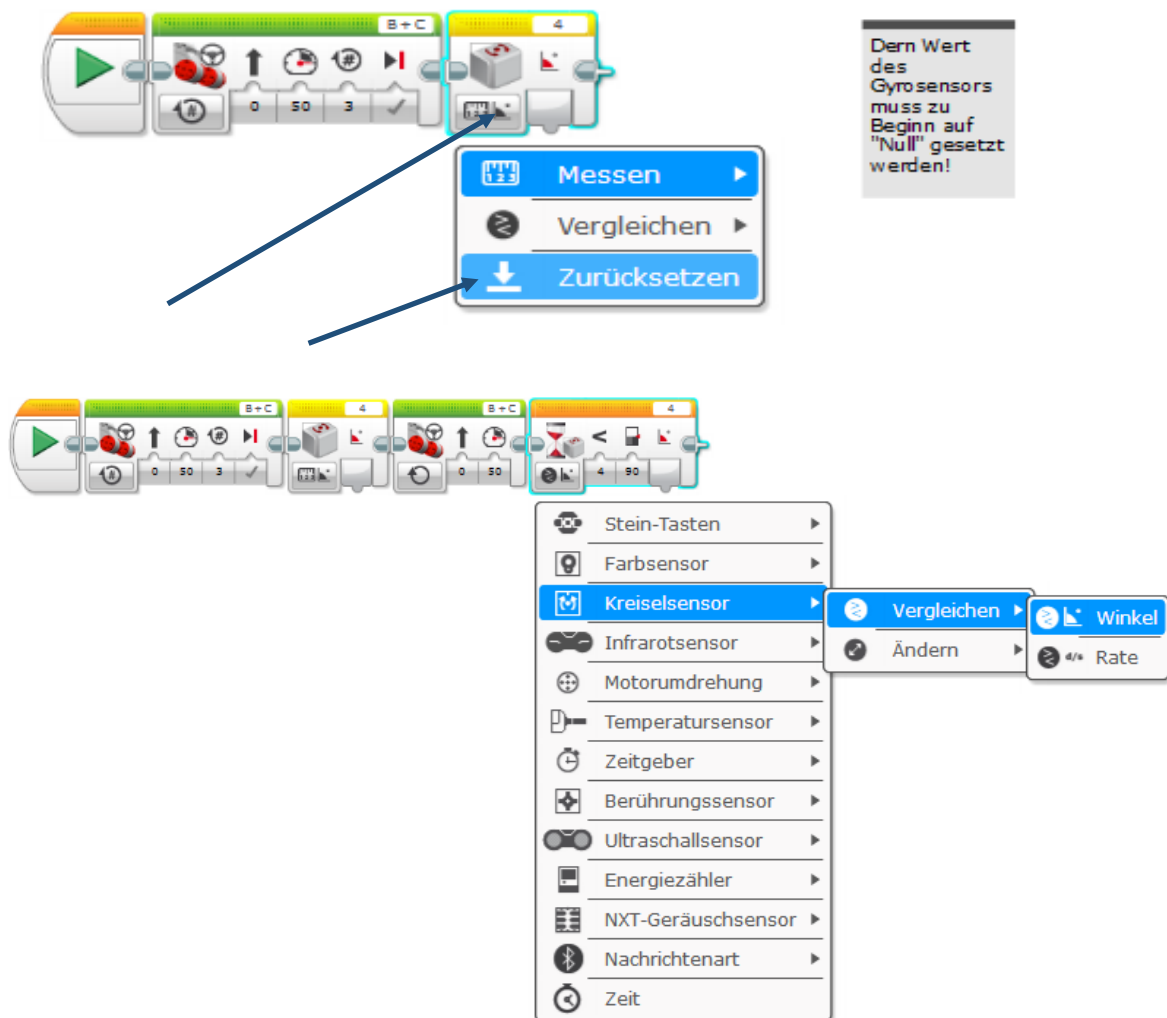
Wird ein Sensor selbst beeinflusst (zurückgesetzt, kalibriert, etc.), so wird der Sensor aus der Kategorie „Sensor-Blöcke“ ausgewählt. Wenn Sensoren für den Programmablauf verwendet werden, so geschieht dies immer beim „Warten-Block“ oder bei den Schleifeneinstellungen.

Verwendung des Gyro-/Kreiselsensors

Auch der Gyrosensor (Gyrosensor=Kreiselsensor) hat seine Tücken:

Der Roboter fährt 3 Umdrehungen geradeaus vorwärts, dann dreht er sich um 90° (Verwendung des Gyrosensors!), bleibt stehen und sagt „Hello“.

Vor Beginn der Drehung muss der Sensor auf „Null“ gestellt werden.



Roboterprogrammierung mit Lego Mindstorms EV3

Achtung! Bei Verwendung von Ultraschall- oder Infrarotsensoren, bzw. dem Gyrosensor gibt man die geforderten Distanzen/Winkel immer mit „<“, „≤“, „>“, „≥“ an! Der Wert mit der Einstellung „=“ wird in einem so kurzen Zeitraum erreicht, dass der Roboter nicht so schnell reagieren kann und somit nicht zielführend eingesetzt werden kann.

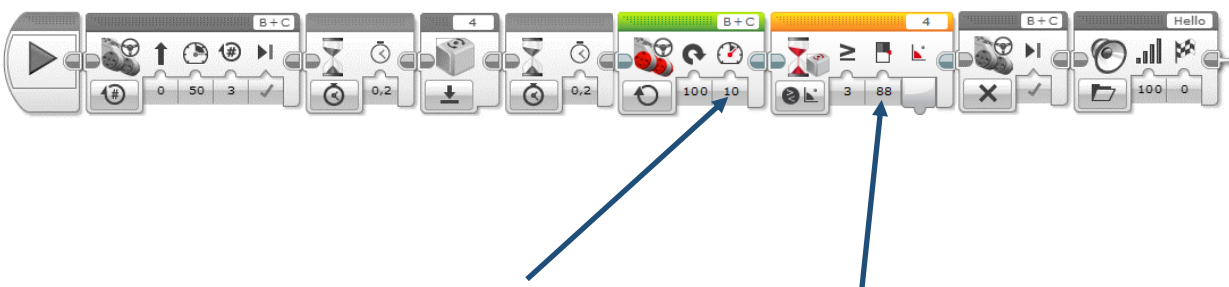


Auch dieses Programm wird noch immer fehlerhaft ausgeführt. Der Sensor braucht eine minimale Zeit, in der der Roboter in Ruhe ist, um exakte Messwerte liefern zu können. → Deshalb empfiehlt es sich, vor und nach dem Zurücksetzen des Kreiselsensors eine kurze Pause einzuprogrammieren. (Erfolgt die Drehung nicht unmittelbar nach dem Zurücksetzen des Sensors, so sollte auch vor Beginn der Drehung eine minimale Pause programmiert werden.)



Ein weiteres Problem stellt die **Trägheit des Roboters** dar, die von Tempo und Gewicht des Roboters abhängt.

→ Im Bereich der Drehung sollte das Tempo möglichst gering gehalten werden und der angestrebte Messwert minimal verkleinert werden. Der tatsächliche Wert muss im Testverfahren ermittelt werden.

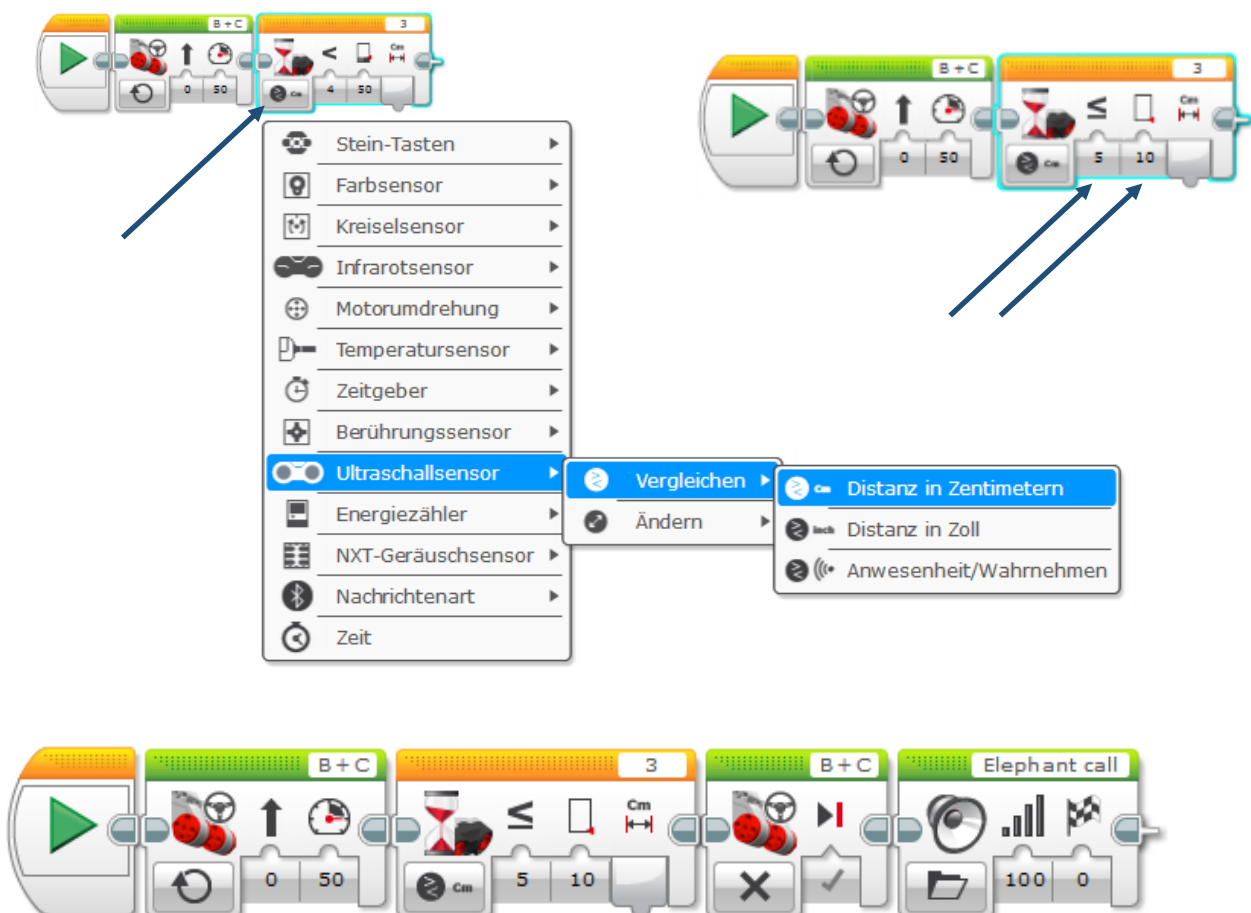


Aufgaben – Sensornutzung „Warten auf“ oder Schleife

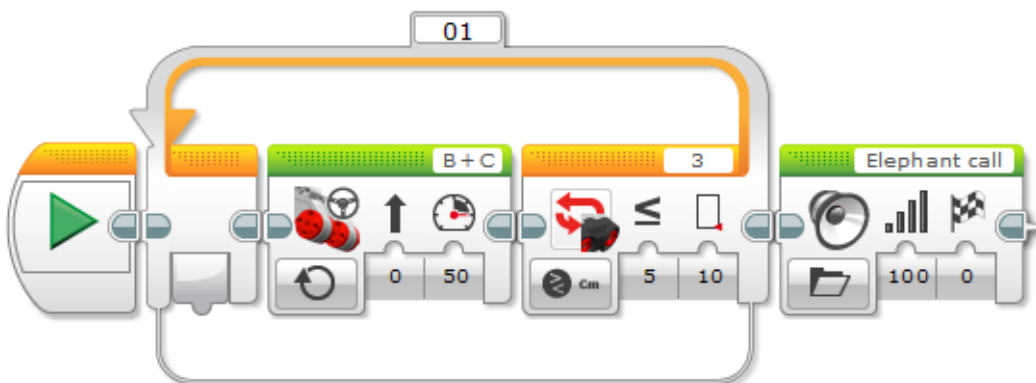
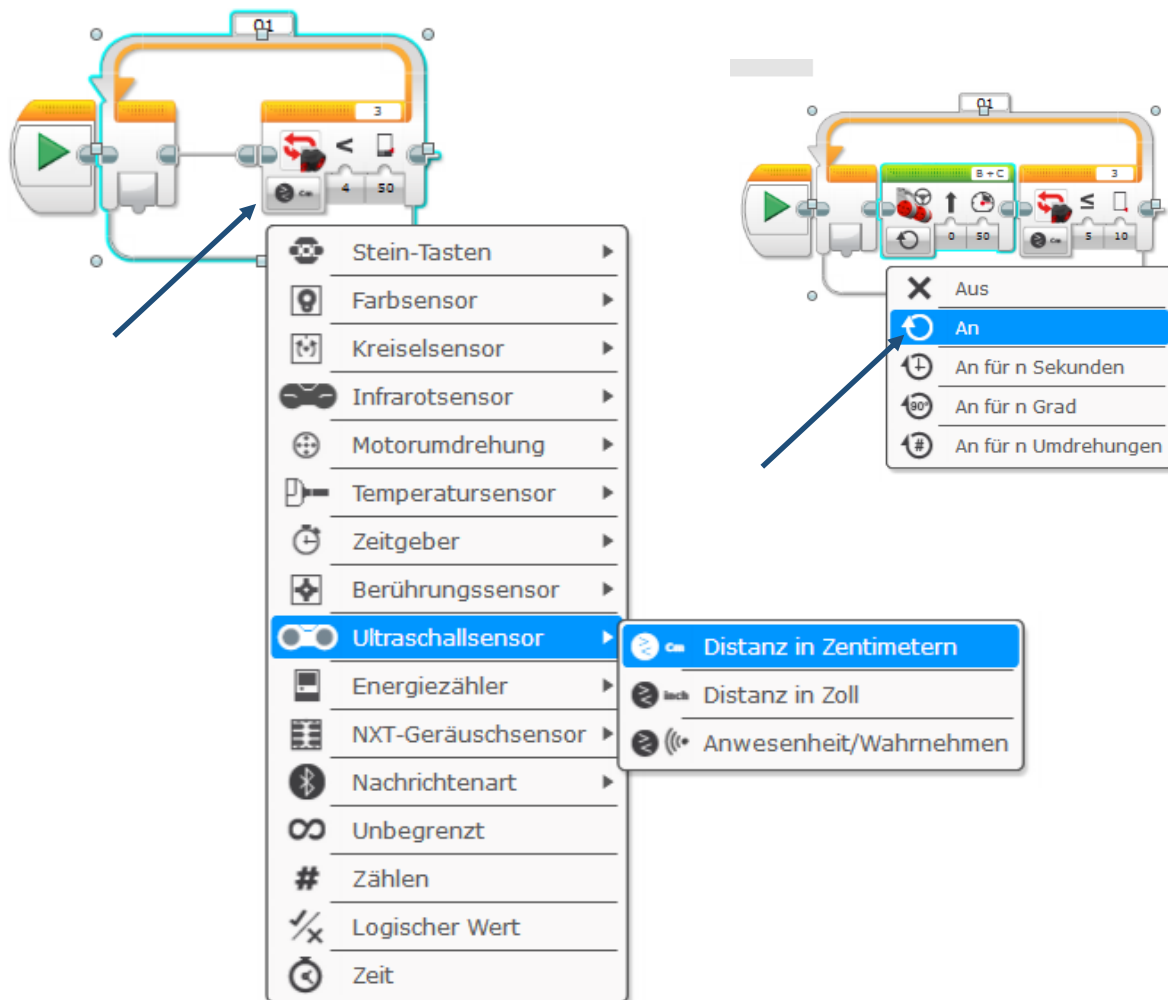
Beim Einsatz von Sensoren kann sowohl die Funktion „Warten auf“ als auch die „Schleife“ zur Programmierung gewählt werden.

Der Roboter fährt geradeaus, bis er 10 cm vor einem Hindernis ist. Dann bleibt er stehen und trompetet wie ein Elefant.

Erstelle zwei Programme für dieses Ereignis. Verwende einmal den „Warten-Block“ und einmal eine Schleife für die Programmierung.

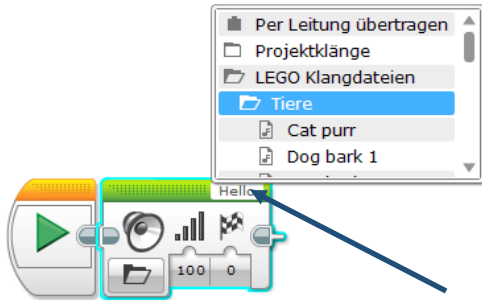


Roboterprogrammierung mit Lego Mindstorms EV3



Roboterprogrammierung mit Lego Mindstorms EV3

Immer wenn der Roboter in einer Entfernung bis 25 cm etwas wahrnimmt, bellt er, bis der Tastsensor gedrückt wird oder der „Störfaktor“ aus der Gefahrenzone verschwindet.



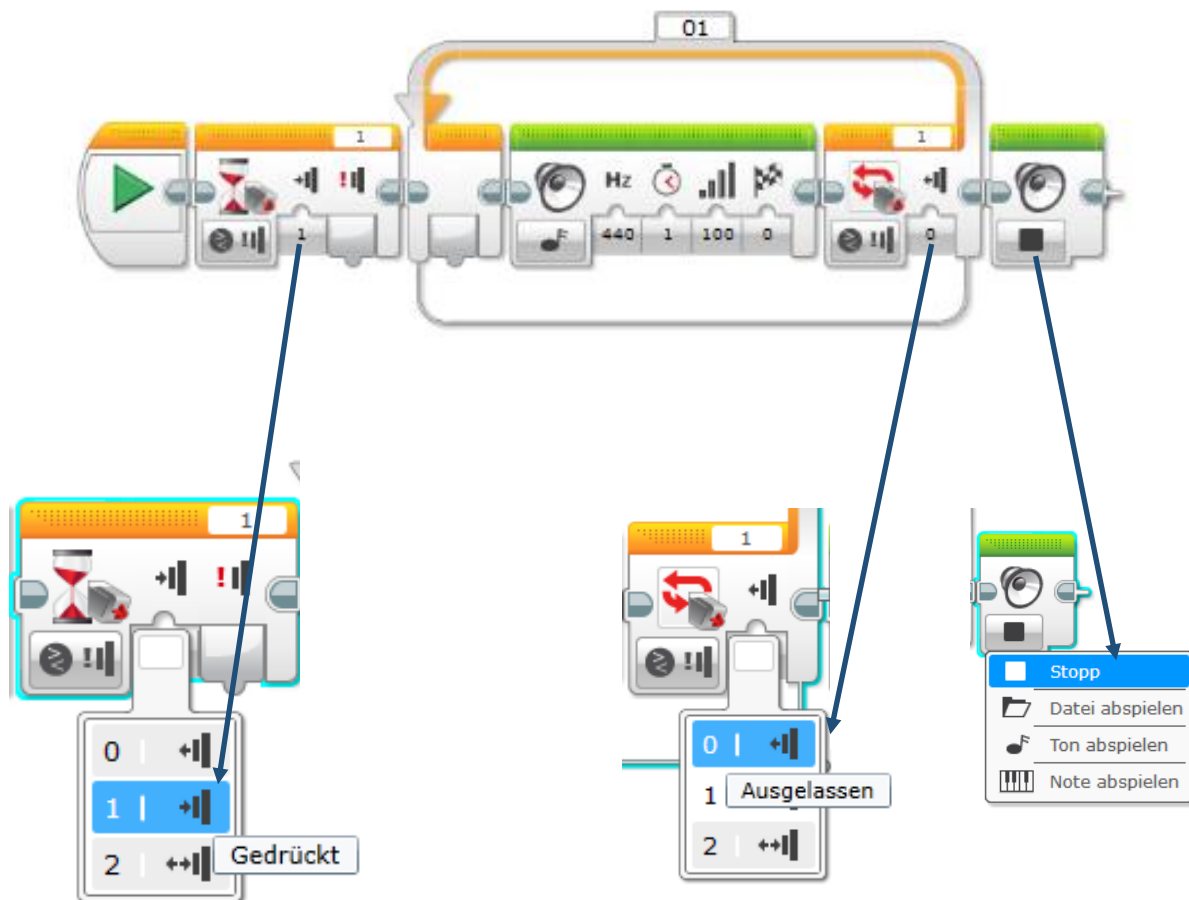
Schleifen können hier benannt werden.



Aufgaben – Sensornutzung II

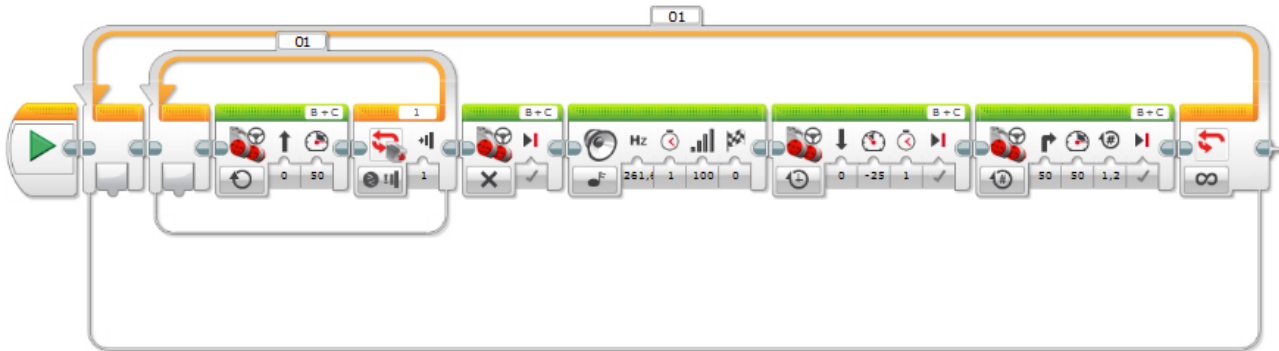
*Solange der Tastsensor gedrückt ist, gibt der Roboter einen Ton von sich.
Erstelle ein Programm für dieses Ereignis.*

Bei diesem Beispiel wird der Tastsensor als Ein/Aus-Schalter verwendet:



Roboterprogrammierung mit Lego Mindstorms EV3

Immer wenn der Roboter beim Vorwärtsfahren auf ein Hindernis stößt, bleibt er stehen, stößt einen Warnton aus, fährt langsam (25% Leistung) eine Sekunde lang zurück, dreht um 100° und fährt dann wieder vorwärts.



Variante zum Beispiel oben:

Immer wenn der Roboter beim Vorwärtsfahren auf ein Hindernis stößt, bleibt er stehen, stößt einen Warnton aus, fährt langsam (25% Leistung) eine Sekunde lang zurück, dreht um 100° und fährt dann wieder vorwärts. Diese Vorgänge wiederholt er 2 Minuten lang.



Schalter („Wenn dann - sonst“) am Beispiel „Linefollower“

Programmiere den Roboter so, dass er entlang einer schwarzen Linie fährt.

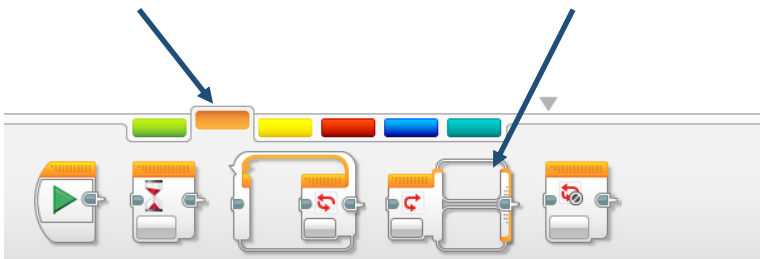
Der **Linefollower** ist DAS klassische Beispiel für eine geschickte Anwendung von Sensoren. Für die Umsetzung gibt es verschiedenste Lösungsansätze, manche sind zwar kompliziert, liefern dafür aber gute Ergebnisse. Bei (fast) allen Programmen schlängelt sich der Roboter an der Kante der Linie entlang.

Hier wird eine relativ einfache Lösung vorgeschlagen. Dabei soll der Roboter auf der linken Seite der Linie platziert werden. Immer wenn der Farbsensor auf „Weiß“ kommt, fährt der Roboter nach rechts, immer, wenn er „Schwarz“ sieht, fährt er nach links.

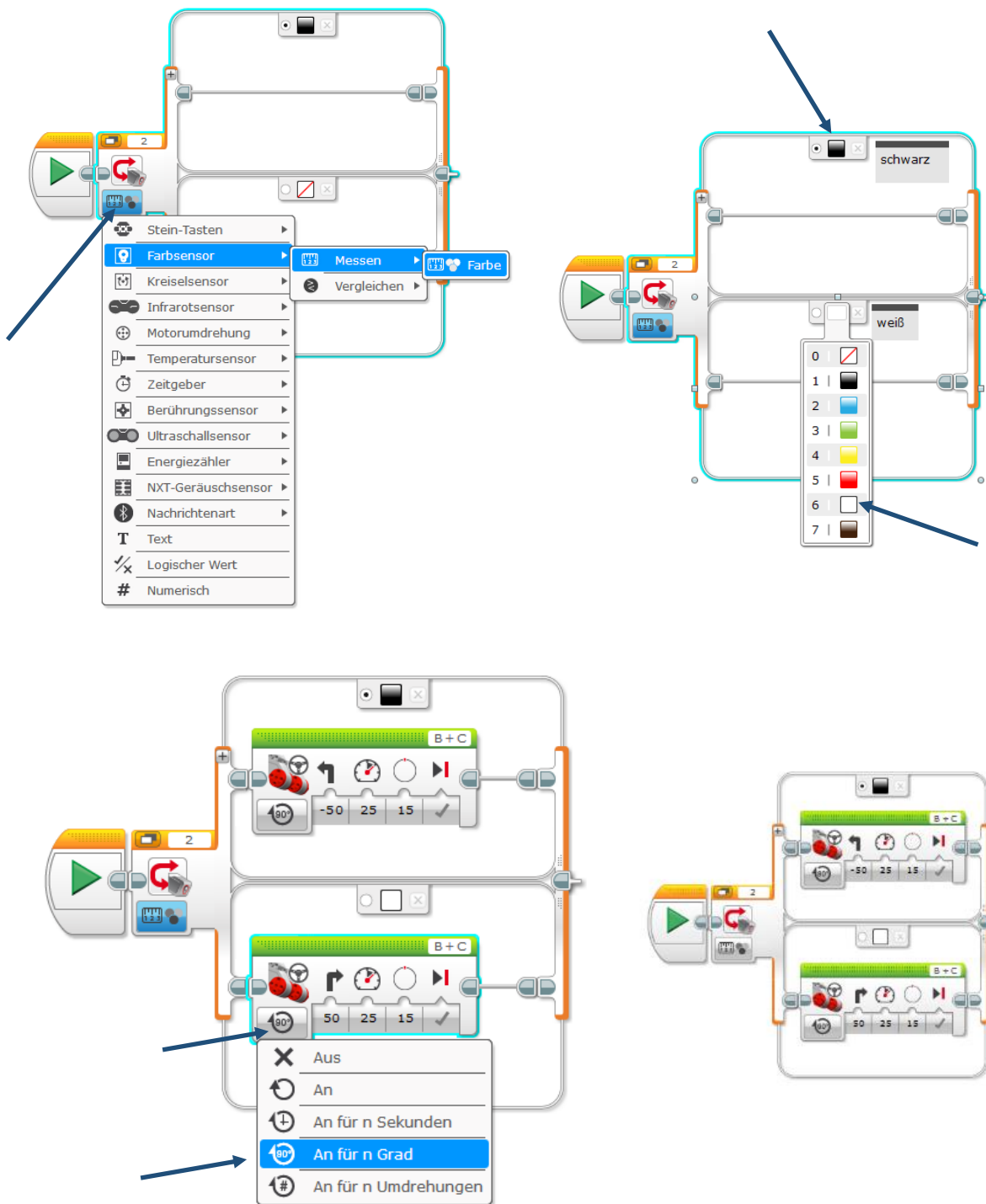
Die **Geschwindigkeit** sollte dabei **minimiert** werden, dass der Roboter ausreichend Zeit hat, zu reagieren.

Die **Wege** werden sehr **kurz** gewählt. → Dazu empfiehlt es sich, die Antriebsmotoren auf Grad statt auf ganze Umdrehungen zu programmieren.

Es wird der **Schalter** als Programmstruktur verwendet.



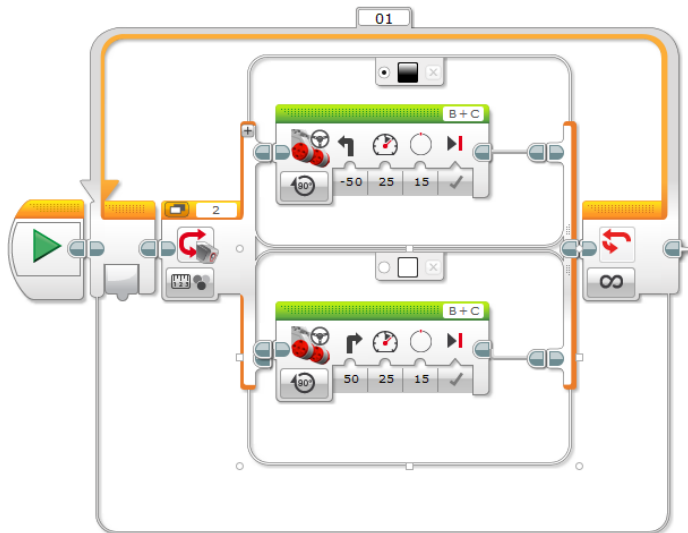
Roboterprogrammierung mit Lego Mindstorms EV3



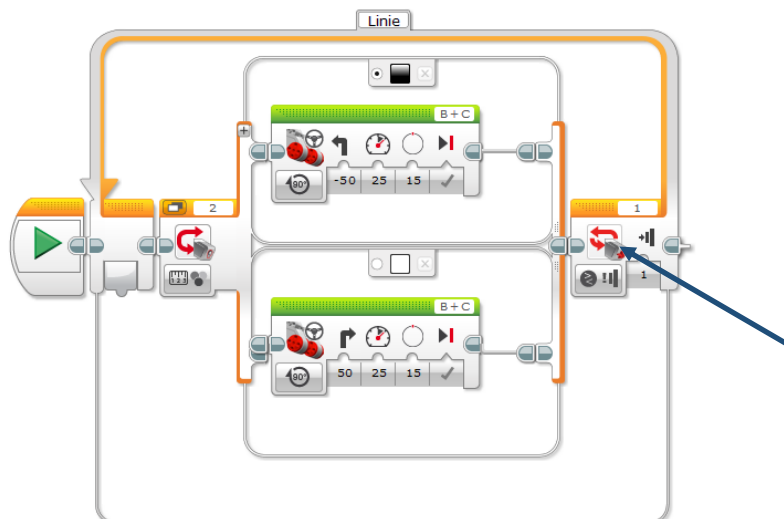
Variante: Bei einer weiteren, aber genaueren Möglichkeit zur Programmierung des Linefollowers wird der Wert des reflektierten Lichtes an der Grenze von der weißen zur schwarzen Farbe vor Ort gemessen. Dann wird statt der Farben dieser Schwellenwert für die Programmierung verwendet.

Aufgaben – Linefollower

Nachdem alle Einstellungen getroffen wurden, wird der Schalter in eine Schleife gelegt, da die Prozedur wiederholt werden soll, solange der Roboter der Linie folgt.



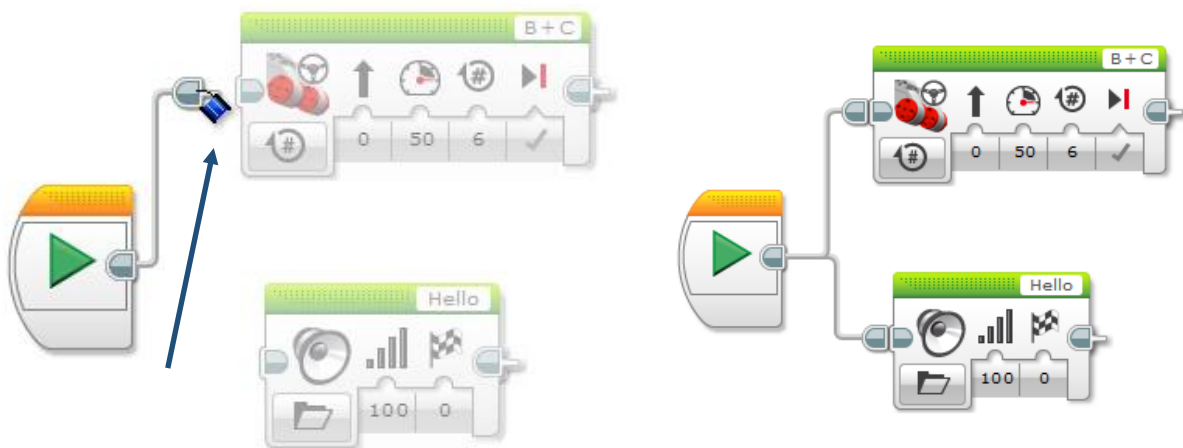
Der Roboter soll solange entlang einer Linie fahren, bis er an ein Hindernis stößt.



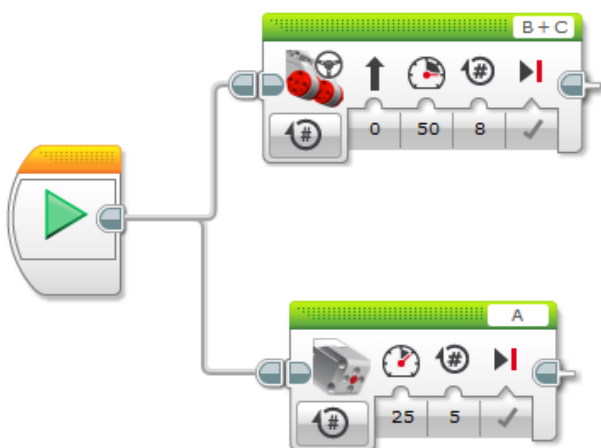
Aufgaben – Multitasking: Ausführen von zwei oder mehreren Programm(teil)en gleichzeitig

Bei der Verwendung von parallel liegenden Tasks müssen die Programmsequenzen händisch mit dem Start-Block verbunden werden.

Der Roboter fährt 6 Umdrehungen geradeaus, beim Wegfahren sagt er „Hello“.

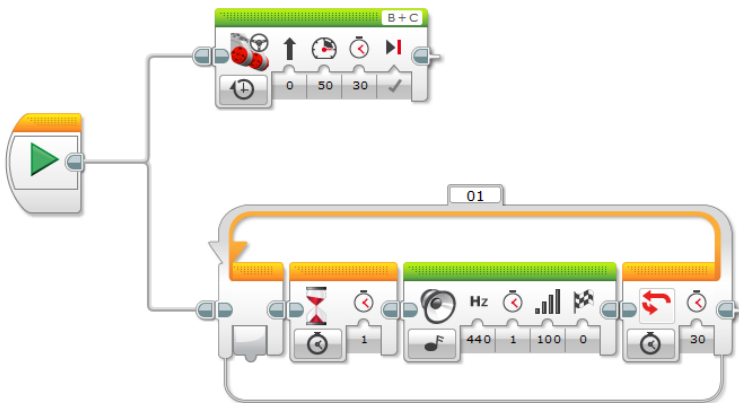


Der Roboter fährt 8 Umdrehungen lang geradeaus. Während der Fahrt dreht der Arbeitsmotor 5 Umdrehungen mit 25% seiner Geschwindigkeit.



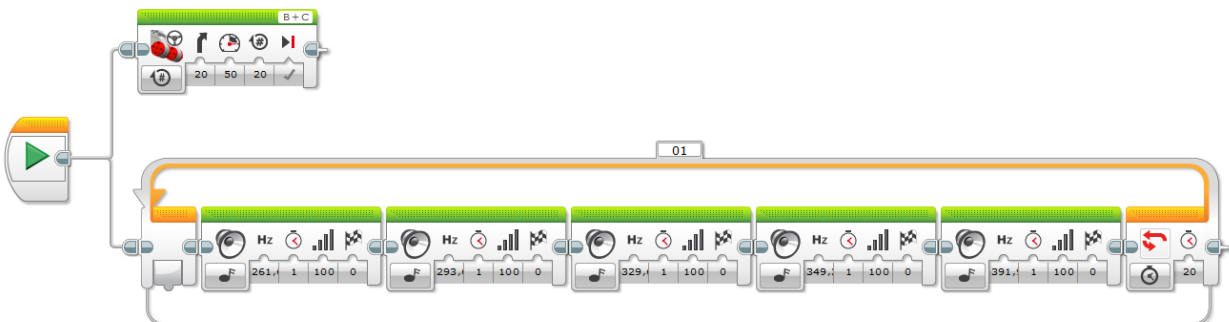
Roboterprogrammierung mit Lego Mindstorms EV3

Der Roboter fährt eine halbe Minute lang geradeaus vorwärts und gibt dabei nach jeder Sekunde einen Warnton von sich. Löse diese Aufgabe mit Hilfe von Tasks. Ein Task steuert das Vorwärtsfahren, der andere die Warntöne.



Wie man diese Aufgabe mit einer „sauberen Programmierung“ löst, findet sich im Kapitel „Schleifen-Interrupt“.

Der Roboter fährt 20 Sekunden im Kreis (beliebig groß). Während der gesamten Fahrt wiederholt er eine kurze Melodie aus 5 Tönen.

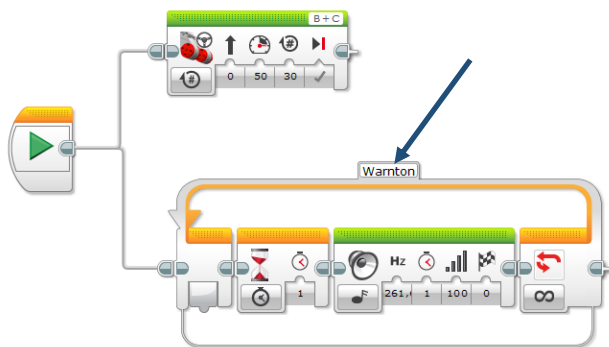


Auch diese Aufgabe sollte besser mit Schleifen-Interrupt beendet werden.

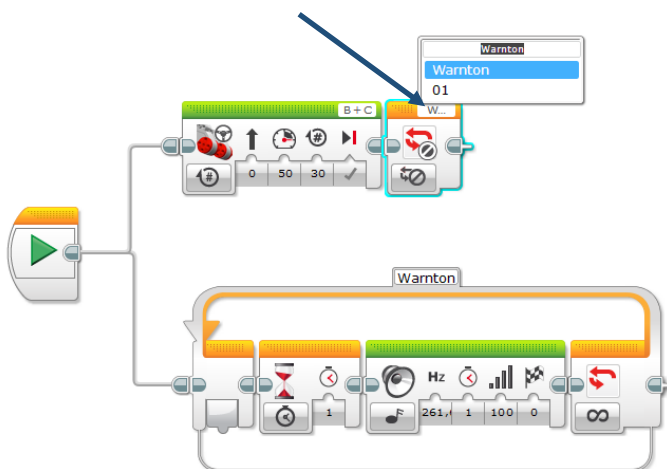
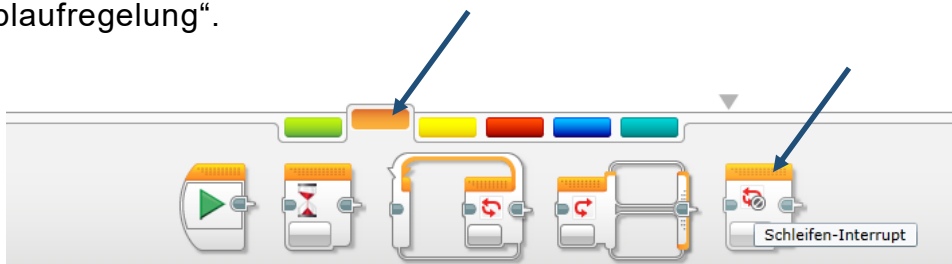
Aufgaben – Schleifen-Interrupt

Bei der Verwendung eines Schleifen-Interrupts muss die entsprechende **Schleife** (durch Doppelklick) **benannt** werden.

Der Roboter fährt eine halbe Minute lang geradeaus vorwärts und gibt dabei nach jeder Sekunde einen Warnton von sich.



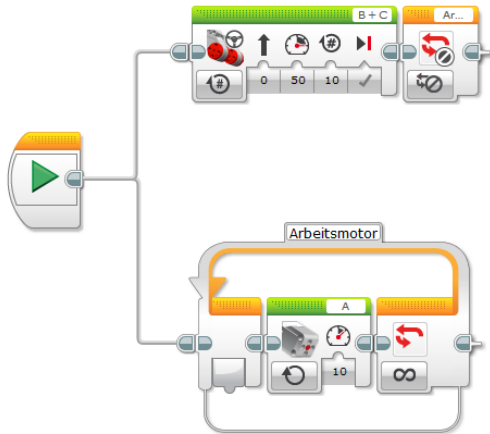
Der „Schleifen-Interrupt-Block“ findet sich in der Block-Kategorie „Ablaufregelung“.



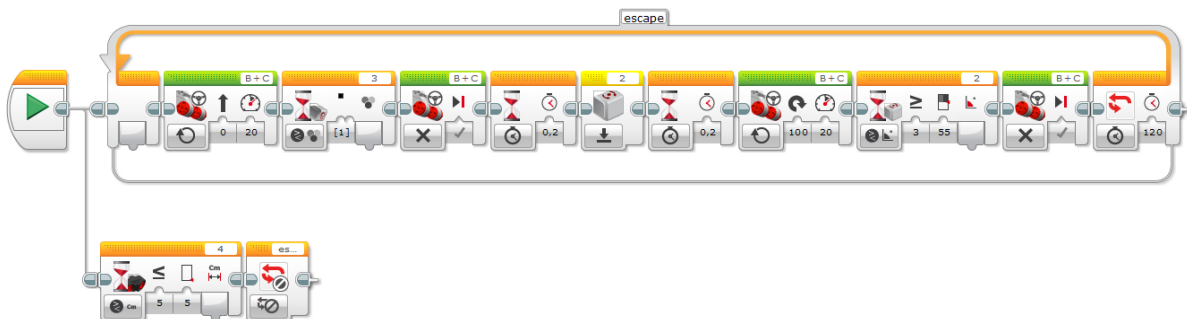
Beim Interrupt-Block wird in der Drop-down-Liste die entsprechende Schleife ausgewählt.

Roboterprogrammierung mit Lego Mindstorms EV3

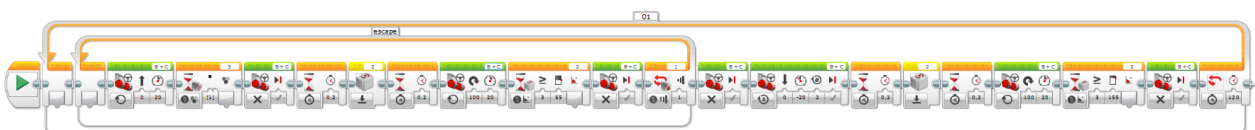
Der Roboter fährt 10 Umdrehungen lang geradeaus. Während der gesamten Fahrt dreht der Arbeitsmotor mit 10% seiner Geschwindigkeit.



Der Roboter fährt geradeaus (20% der Leistung), bis er zu einer schwarzen Linie kommt, dann dreht er sich um 55° und fährt wieder geradeaus, bis er zu einer schwarzen Linie kommt. Das Programm wird nach 3 Minuten beendet. Sollte der Roboter auf ein Hindernis treffen, bleibt er 5 cm davor stehen.



Variante: Wenn der Roboter auf ein Hindernis trifft, bleibt er stehen, fährt 2 Umdrehungen rückwärts, dreht sich um 155° und fährt dann wieder geradeaus weiter.



Aufgaben – Erstellung und Verwendung von Blöcken

Schreibe ein Programm für die Melodie von „Alle meine Entlein“.

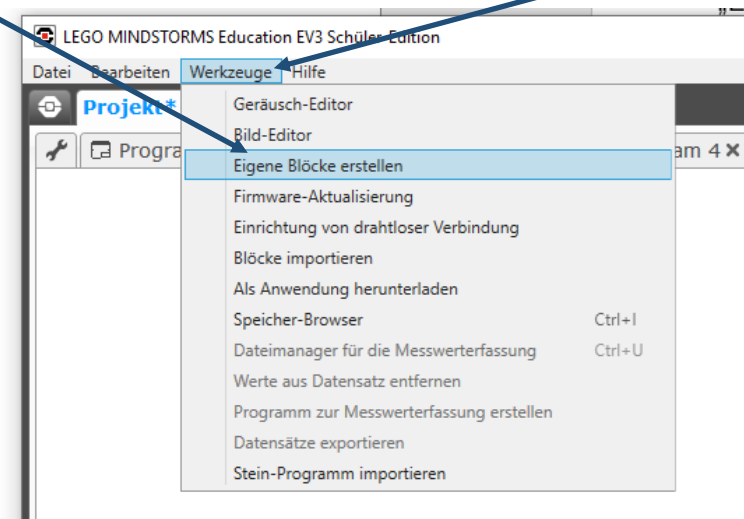
(C5/0,5“; D5/0,5“; E5/0,5“; F5/0,5“; G5/1“ G5/1“;
I:A5/0,5“; A5/0,5“; A5/0,5“; A5/0,5“; G5/1“; Pause /0;5“ :I;
F5/0,5“; F5/0,5“; F5/0,5“; F5/0,5“; E5/1“; E5/1“;
G5/0,5“; G5/0,5“; G5/0,5“; G5/0,5“; C5/1“)



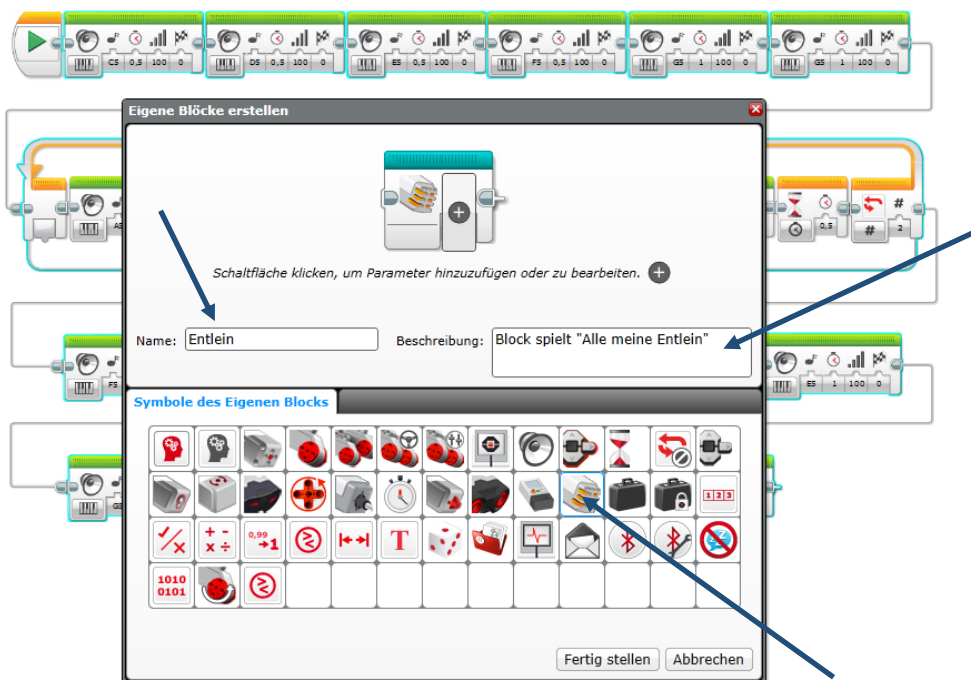
Roboterprogrammierung mit Lego Mindstorms EV3

Schreibe ein Programm für die Melodie von „Alle meine Entlein“. Erstelle aus dem Programm einen Block mit dem Titel „Entlein“.

Nach dem Erstellen der Programmprozedur wird die entsprechende Sequenz markiert (ohne Start-Block). In der Menüleiste die Kategorie „Werkzeuge“ / „Eigene Blöcke erstellen“ gewählt.

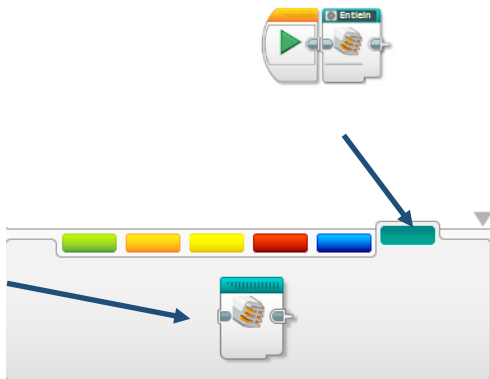


Ein neues Fenster öffnet sich. Dort werden ein passender Name sowie ein Bild vergeben. Eine Beschreibung für die Funktion des Blockes kann, muss aber nicht, hinzugefügt werden. Mit „Fertig stellen“ wird die Eingabe bestätigt.



Roboterprogrammierung mit Lego Mindstorms EV3

Der erstellte Block wird in der Kategorie „Eigene Blöcke“ gespeichert. Er steht nun allen Programmen des Projektes zur Verfügung.

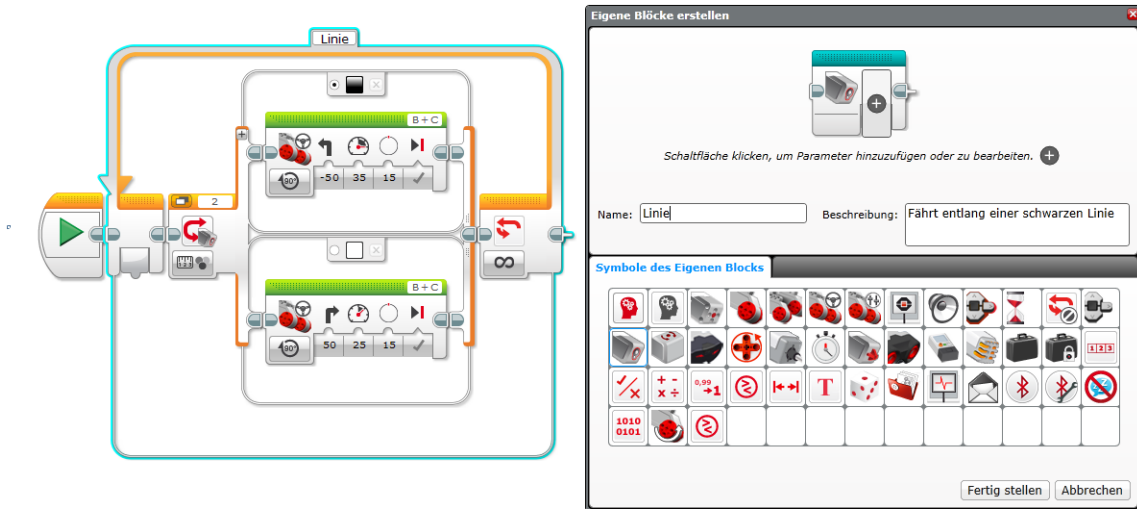


Der Roboter fährt bis 15 cm vor ein Hindernis. Dann bleibt er stehen und spielt „Alle meine Entlein“.



Roboterprogrammierung mit Lego Mindstorms EV3

Erstelle ein Programm für: „Der Roboter fährt entlang einer schwarzen Linie“. Speichere das Programm als Block mit dem Titel „Linie“.



Lass den Roboter bis zu einer schwarzen Linie fahren, dann lass ihn der Linie 20 Sekunden lang folgen. Verwende den Block „Linie“.

