



Handreichung: Physical Computing – Lego Mindstorms EV3

1. Kurzvorstellung

Diese Handreichung bietet eine Vielzahl von Anregungen in Form von Aufgaben und Hinweisen zur Integration des Lego Mindstorms EV3 in ein komplexes Unterrichtsprojekt für die 8. Klasse Gymnasium. Sie zielt darauf ab, einen spielerischen Unterrichtsblock von etwa 6 Stunden mit einer komplexen Aufgabe zu gestalten. Dank der Drag-and-Drop-Programmiersoberfläche ist dieses Projekt besonders für die Unterstufe geeignet.

Die vorgeschlagenen Aufgaben bauen aufeinander auf und fördern die Kreativität der Schüler, da es verschiedene Ansätze gibt, um die gestellten Ziele zu erreichen. Darüber hinaus gibt es im Internet eine umfangreiche Community, die sowohl von Lego als auch von Nutzern gepflegt wird. Hier können weitere Funktionen des Lego Mindstorms EV3 ausprobiert werden, einschließlich Erweiterungen durch maschinelles Lernen oder die Vernetzung von Computern. Dies bietet den Schülern zusätzliche Möglichkeiten, ihre Fähigkeiten und ihr Wissen zu erweitern. Die Aufgaben sind als Partneraufgabe konzipiert. Können aber auch in Einzelarbeit oder in Gruppen bis max. 4 SuS bearbeitet werden.

Die Aufgaben gehen davon aus, dass die Lego Mindstorm Roboter bereits im Vorfeld zusammengesetzt und eingestellt worden sind (vgl. Voraussetzungen). Zur Bewertungen können Aufgaben durch einzelne Gruppen präsentiert werden. Während der Erarbeitungsphase sollte die Lehrkraft die SuS unterstützen und für Fragen bereit stehen.

2. Einordnung in die Lehrpläne

Klasse	OS	Gy	BS
7	-	-	-
8	-	LB3: Komplexaufgabe	-
9	LB2: Komplexaufgabe zur Algorithmmierung	-	-
10	-	-	-
11/12	-	-	-

3. Lernziele

Bewegungsabläufe verstehen: Die SuS erkennen und zu verstehen, dass Bewegungsabläufe aus vielen einzelnen, im Voraus programmierten Schritten bestehen, die der Roboter ausführt.

Programmblöcke verstehen: Die SuS wenden Programmblöcke an und bewerten die Vorteile von Schleifen.



Stärkung der Teamarbeit und sozialen Interaktionsfähigkeiten: Die SuS koordinieren in Teamarbeit die Aufgaben und stärken dadurch die sozialen Interaktionsfähigkeiten.

Aktoren und Sensoren erkennen: Die SuS unterscheiden Aktoren und Sensoren hinsichtlich ihrer Funktionen und Auswirkungen in einem Robotersystem.

4. Voraussetzungen

Technische Voraussetzungen:

- Kompatible Rechenmaschinen mit vorinstallierter Version von LEGO MINDSTORMS Education EV3. Diese Software läuft unter Windows 10 / 11, Mac OS 10.14 oder neuer, IOS-Tablets, Android-Tablets oder Chrome OS.
- EV3 Steine, die vollständig aufgeladen sind. Die Steine sollten eventuell umbenannt werden, um sie leichter identifizieren zu können.

Fachliche Voraussetzungen:

- Kenntnisse in Informatiksystemen (LB 2, 7. Klassenstufe).
- Verständnis für Algorithmen (LB 3, 7. Klassenstufe, LB 1, 8. Klassenstufe)

Materielle Voraussetzungen:

- Ausreichend Lego Mindstorms-Kits, inklusive Sensoren und Aktoren.
- Zusätzliche Ressourcen wie Batterien und Ladegeräte.

Optional:

- Farbblätter (z.B. Grün, Gelb, Rot) können hilfreich sein.
- DIN A3 Blätter
- Ein Lineal kann ebenfalls nützlich sein.

5. Kurzdarstellung

Hinweise:

- Programme werden nicht exakt parallel ausgeführt
- Eingegebene Werte (z.B. Entfernung & Grad entsprechen nicht der Realität!) müssen je Roboter betrachtet werden.
- Kleinere Blöcke programmieren und testen



Impulsfragen:

- Wo werden Roboter eingesetzt?
- Was macht der Roboter?
- Welche Einzelschritte sind zu erkennen?

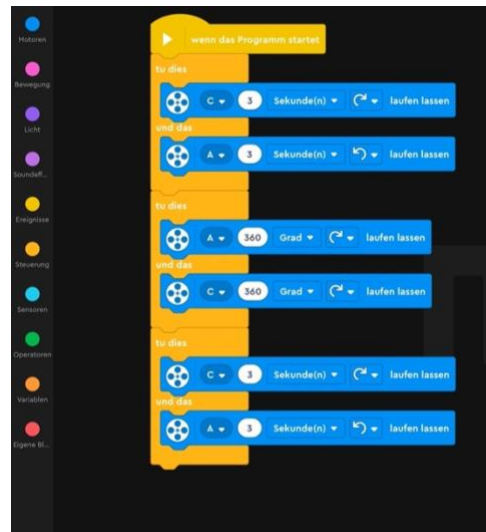
- Welche „Befehle “ erhält der Roboter?
- Was machen Sensoren? Z.B. reflektierendes Licht messen (weiß = hoher Wert, schwarz = kleiner Wert
→ Anhand des Reflektionswertes die Farbe bestimmen)
- Welche Grenzen gibt es bei der Verwendung von Sensoren?

Aufgaben:

Es gibt insgesamt drei Aufgaben, die aufeinander aufbauen und schrittweise die Funktionen des Roboters erkunden. Sowie eine Zusatzaufgabe. Im Verlauf dieser Aufgaben wird eine Ampelfunktionalität sowie ein Parcours unter Einsatz der Sensoren betrachtet. Die Musterlösung ist den Screenshots zu entnehmen. Allerdings ist zu beachten, dass verschiedene Lösungen gültig sein können.

Aufgabe 1: Einführung in die Robotik:

- 1.a. Programmiere den Roboter so, dass er geradeaus fährt. Diskutiere mit deinem Partner, welche Befehle dem Roboter gegeben werden müssen.
- 1.b. Lass den Roboter eine kurze Strecke vorwärts und dann zurück fahren (Beachte: der Roboter soll während der Fahrt nach „vorne “ schauen). Optional: Nutze ein Lineal um die Entfernung abzumessen.



Aufgabe 2: Statischer Parcours (Erkennen und Verstehen von Parametrisierung)

- 2.a. Instruiere den Roboter, genau 20 cm vorwärts zu fahren.
- 2.b. Programmiere den Roboter so, dass er sich um 90 Grad dreht.
- 2.c. Befehle dem Roboter, einmal um eine “gesperrte Zone“ (DIN A5 Größe) zu fahren. Tipp: Dafür können Schleifen verwendet werden.
- Hinweis: Einfach Blattgröße austauschen und Aufgabe wiederholen lassen. SuS mit Schleifenprogrammierung können diese Aufgabe dann wesentlich schneller lösen.



Aufgabe 3: Abstandssensor

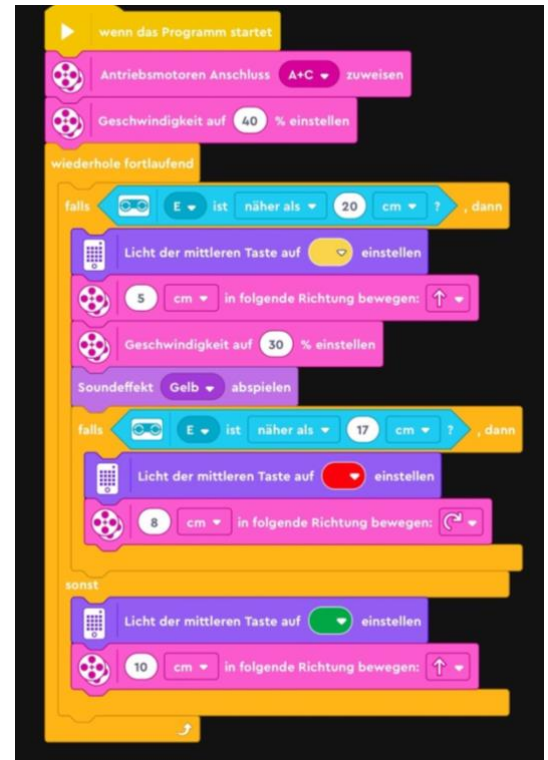
- 3.a. Programme den Roboter so, dass er kontinuierlich geradeaus fährt, bis er ein Hindernis erkennt. Nach dem Erkennen eines Hindernisses sollte der Roboter die Geschwindigkeit reduzieren und anschließend ausweichen. Nutze dafür folgende Parameter:

Geschwindigkeit normal:	40%
Hindernis erkennen & Geschwindigkeit reduzieren:	Näher 20cm & 30%
Hindernis ausweichen	Näher als 15cm

- 3.b. Verwende das Licht der mittleren Taste, um eine Ampelschaltung zu programmieren.

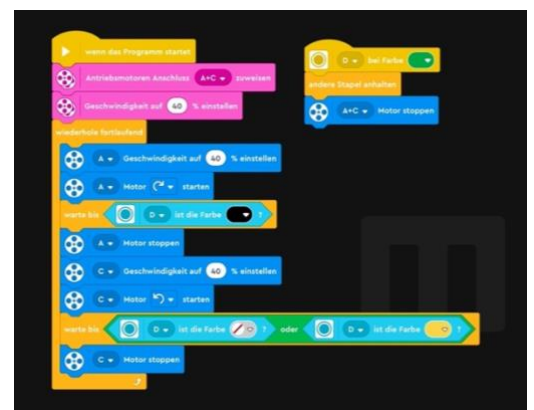
Kein Hindernis:	Grün
Hindernis kleiner 20cm	Gelb
Hindernis wird umfahren	Rot

- 3.c. Fährt der Roboter über Grüne Fläche soll er sofort stehen bleiben. Erweitere dazu deinen Code!



Aufgabe 4: Liniensensor (Für die Schnellen!)

- 4.a. Programme den Roboter so, dass er der Kante eines Blatt Papiers folgt. Nutze dafür den Liniensensor. Der Roboter soll jederzeit in der Lage sein sich entlang der Kante auszurichten.





Ampel & Parcours



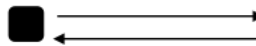
Beachte: Die Roboter sind Schuleigentum, sie műssen unbedingt sorgsam behandelt werden. Baue sie nicht um ohne dies vorher mit der Lehrkraft abzusprechen.



Bearbeite die folgenden Aufgaben mit deinem Sitznachbar. Bevor du mit der nűchsten Aufgabe beginnst, solltest du die Fűhigkeit deines Roboters kurz demonstrieren.

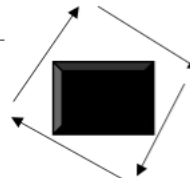
Starte die App „Lego Mindstorm“. Verbinde dich dann zunűchst in der App mit deinem Roboter. Den Namen deines Roboters erkennst du am Aufkleber. Dafűr nutzt du am rechten Bildschirmrand das „Verbinden Symbol“. Nun kannst du Befehle mittels Drag&Drop hineinziehen und durch den „Play Button“ auf deinem Roboter ausfűhren. Der Roboter kann jederzeit mit dem „Pause Button“ in der App gestoppt werden. Je Aufgabe solltest ein neues Projekt (Linke obere Bildschirmecke) starten und nach der Aufgabe benennen.

Aufgabe 1: Einfűhrung in die Robotik:



- 1.a. Programmiere den Roboter so, dass er geradeaus fűhrt. Diskutiere mit deinem Partner, welche Befehle dem Roboter gegeben werden műssen.
- 1.b. Lass den Roboter eine kurze Strecke vorwűrts und dann zurűck fahren (Beachte: der Roboter soll wűhrend der Fahrt nach „vorne“ schauen). Optional: Nutze ein Lineal um die Entfernung abzumessen.

Aufgabe 2: Statischer Parcours (Erkennen und Verstehen von Parametrisierung)



- 2.a. Instruiere den Roboter, genau 20 cm vorwűrts zu fahren.
- 2.b. Programmiere den Roboter so, dass er sich um 90 Grad dreht.
- 2.c. Befehle dem Roboter, einmal um eine „gesperrte Zone“ (DIN A5 Grűe) zu fahren. Tipp: Dafűr kűnnen Schleifen verwendet werden.

Aufgabe 3: Abstandssensor

- 3.a. Programmiere den Roboter so, dass er kontinuierlich geradeaus fűhrt, bis er ein Hindernis erkennt. Nach dem Erkennen eines Hindernisses sollte der Roboter die Geschwindigkeit reduzieren und anschlieűnd ausweichen.

Seite 1 von 4





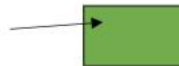
Nutze daf r folgende Parameter:

Geschwindigkeit normal:	40%
Hindernis erkennen & Geschwindigkeit reduzieren:	N�her 20cm & 30%
Hindernis ausweichen	N�her als 15cm

- 3.b. Verwende das Licht der mittleren Taste, um eine Ampelschaltung zu programmieren.

Kein Hindernis:	Gr�n
Hindernis kleiner 20cm	Gelb
Hindernis wird umfahren	Rot

- 3.c. F hrt der Roboter  ber Gr ne Fl che soll er sofort stehen bleiben. Erweitere dazu deinen Code!



Aufgabe 4: Liniensensor (F r die Schnellen!)



Abbildung 1 Quelle: <https://wshile-industrial-robots.com/ide/blog/lagerruboter>

- 4.a. Programmiere den Roboter so, dass er der Kante eines Blatt Papiers folgt. Nutze daf r den Liniensensor. Der Roboter soll jederzeit in der Lage sein sich entlang der Kante auszurichten.