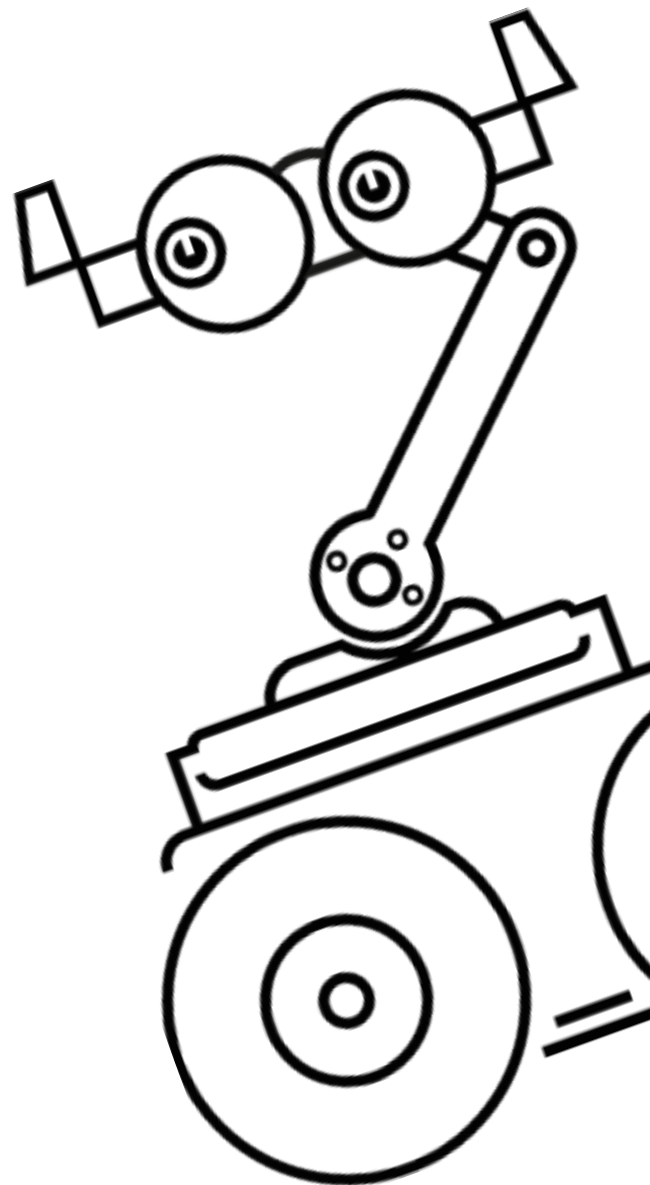


Programmieren mit Open Roberta

Unterrichtsbeispiele mit dem Lego WeDo 2.0 Set



ROBERTA
INITIATIVE

Roberta ist ein eingetragenes Warenzeichen der Fraunhofer-Gesellschaft e.V.
Roberta ist seit 2010 Mitglied der Fraunhofer Academy

Inhalt

Dieses Dokument stellt eine Unterrichtseinheit aus der Reihe [Roberta-Lernmaterialien für den Lego Education WeDo](#) dar.

Unterrichtseinheit

Einfacher Regelkreis

Die Roberta-Lernmaterialien für den Lego Education WeDo umfassen folgende Kapitel:

Kapitel 1: Einführung in Open Roberta® und das Lego® WeDo 2.0 Set

Kapitel 2: Der Übergang von Lego WeDo zu NEPO®

Weitere Unterrichtseinheiten mit Open Roberta und Lego WeDo 2.0

- Objekte orten
- Automatismen und Reflexe

Ein einfacher Regelkreis mit Lego WeDo 2.0

Kurz

Im vorliegenden Unterrichtsentwurf wird der Neigungssensor von Lego WeDo 2.0 aus der Waagerechten herausbewegt und mithilfe des zu erstellenden Programms in die Waagerechte zurückgesetzt.

Thema

Einfacher Regelkreis

Klassenstufe

3 bis 6

Fächer

Mathematik, Sachunterricht, Biologie, Naturphänomene und Technik

Zeitaufwand

ca. 90 Minuten

Material (für je 1 Kind)

- Ein Tablet mit Internetanschluss (<https://lab.open-roberta.org>) oder mit vorinstallierter App »Open Roberta Mobile« für Android und iOS
- Ein Lego Education WeDo 2.0 Set

Voraussetzungen

- Grundlegende Kenntnis der basalen Programmierblöcke von NEPO® (Open Roberta Lab)

Kompetenzen

- Die Schüler*innen beschreiben die grundsätzliche Funktionsweise eines Regelkreises in eigenen Worten oder mithilfe von Fachsprache (Istwert, Sollwert usw.).
- Die Schüler*innen setzen die grundlegenden Blöcke der Programmiersprache NEPO® zielgerichtet ein und nutzen Kontrollblöcke (Entscheidungen, Schleifen, Warten).
- Die Schüler*innen arbeiten im Team und erweitern ihre Kollaborations- und Kommunikationskompetenz.
- Die Schüler*innen entwickeln ihre arithmetischen und informatischen Kompetenzen.

Über diese Kompetenzen hinaus werden mit dem vorliegenden Projekt weitere Kompetenzen gefördert, die den jeweiligen Medienkompetenzrichtlinien der einzelnen Bundesländer zugeordnet werden können. Da diese jedoch sehr vielfältig sind (sowohl was deren Bezeichnung als auch deren Struktur angeht), werden sie an dieser Stelle nicht explizit erwähnt, sondern sind situativ von der Lehrperson auszuwählen.

1. Grundlagen

Ein wesentlicher Bestandteil technischer und damit auch informatischer Systeme sind Sensoren. Sensoren, auch als Aufnehmer oder Fühler bezeichnet, erfassen ihre Umgebung und stellen Informationen der Rechen-/Steuereinheit bereit. Typischerweise erfolgt dann eine Verarbeitung der eingeholten Umgebungswerte in Form einer Ausgabe. So gesehen sind Sensoren die Sinnesorgane der Technik. Im vorliegenden Unterrichtsentwurf wird der Neigungssensor von Lego WeDo 2.0 zur Veranschaulichung eines Regelkreises eingesetzt.

Das wohl gängigste Beispiel für einen technischen Regelkreis ist das Thermostat, präziser formuliert das eigentliche Heizkörperthermostatventil): Im Thermostat wird die aktuelle Raumtemperatur (= Istwert) gemessen und mit der eingestellten Temperatur (= Sollwert) verglichen. Ist der Ist-Wert kleiner als der Soll-Wert, wird an der Heizung ein Ventil geöffnet – es wird geheizt. Dadurch erhöht sich der Ist-Wert. Sobald der Istwert dem Soll-Wert entspricht, wird das Ventil wieder geschlossen.

Regelkreise gibt es aber auch in anderen Bereichen. In der Biologie sind z. B. die Wärmeregulation gleichwarmer Tiere oder die Populationsdichte in Räuber-Beute-Beziehungen nach demselben Schema aufgebaut, in der Ökonomie kann die Marktregulierung als Beispiel dienen.

Wenn Sie eines der angesprochenen Themen im Unterricht behandeln, können Sie Lego WeDo 2.0 anhand der Beispiele in dieser Unterrichtseinheit leicht zur Festigung, Vertiefung und Erweiterung der fachlichen Inhalte und Kompetenzen nutzen. Bezüge zum Bildungsplan finden Sie auf Seite 10. Der Übergang vom Fachinhalt zum Arbeiten mit WeDo 2.0 gestaltet sich dabei denkbar einfach – in der Regel genügt eine Formulierung im Sinne von »... und das wollen wir nun einmal selbst programmieren!«, gefolgt von der Zielsetzung als Aufgabenstellung.

2. Aufbau

Für den Bau des Lego-Modells werden benötigt:

- ein Smarthub
- ein Motor
- ein Neigungssensor
- weitere Bauteile aus dem WeDo-Kasten

Der exakte Zusammenbau des Modells kann den Schüler*innen überlassen werden. Die folgenden Abbildungen sollen lediglich als ein Beispiel dienen, wie das Objekt aussehen kann.

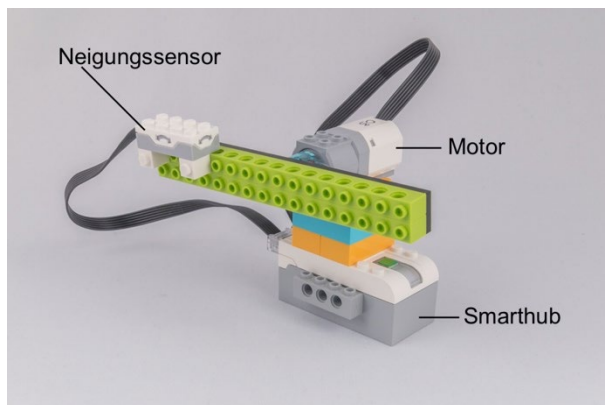


Abbildung 1: Das Lego-Modell für die vorliegende Unterrichtseinheit

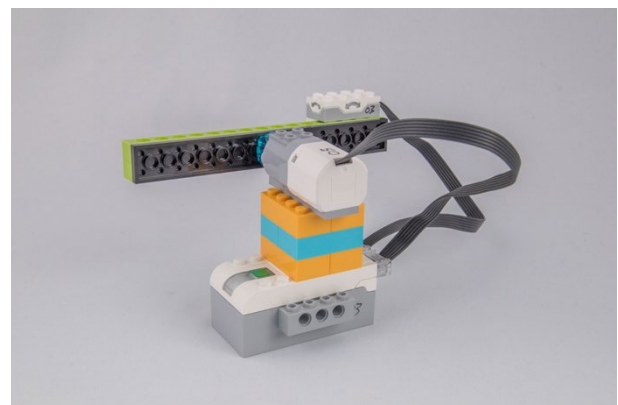


Abbildung 2: Das Lego-Modell (Rückansicht)

Tipp: Wenn Sie den Zusammenbau den Schüler*innen überlassen, geben Sie ihnen eine Zeit vor (z. B. 10 Minuten). Erfahrungsgemäß zieht sich der Zusammenbau ansonsten sehr in die Länge.

Im Open Roberta Lab bzw. in der Open Roberta Mobile App können Sie in »Roboterkonfiguration« einstellen, welches Bauteil mit welchem Anschluss verwendet werden soll. Hier ist es wichtig, dass der Neigungssensor dem Anschluss zugeordnet ist, an welchem er am Roboter tatsächlich angeschlossen ist (Reihenfolge von links an rechts. Der linke Anschluss ist der Erste). Welches Bauteil welchen Anschluss erhält, ist nicht relevant.

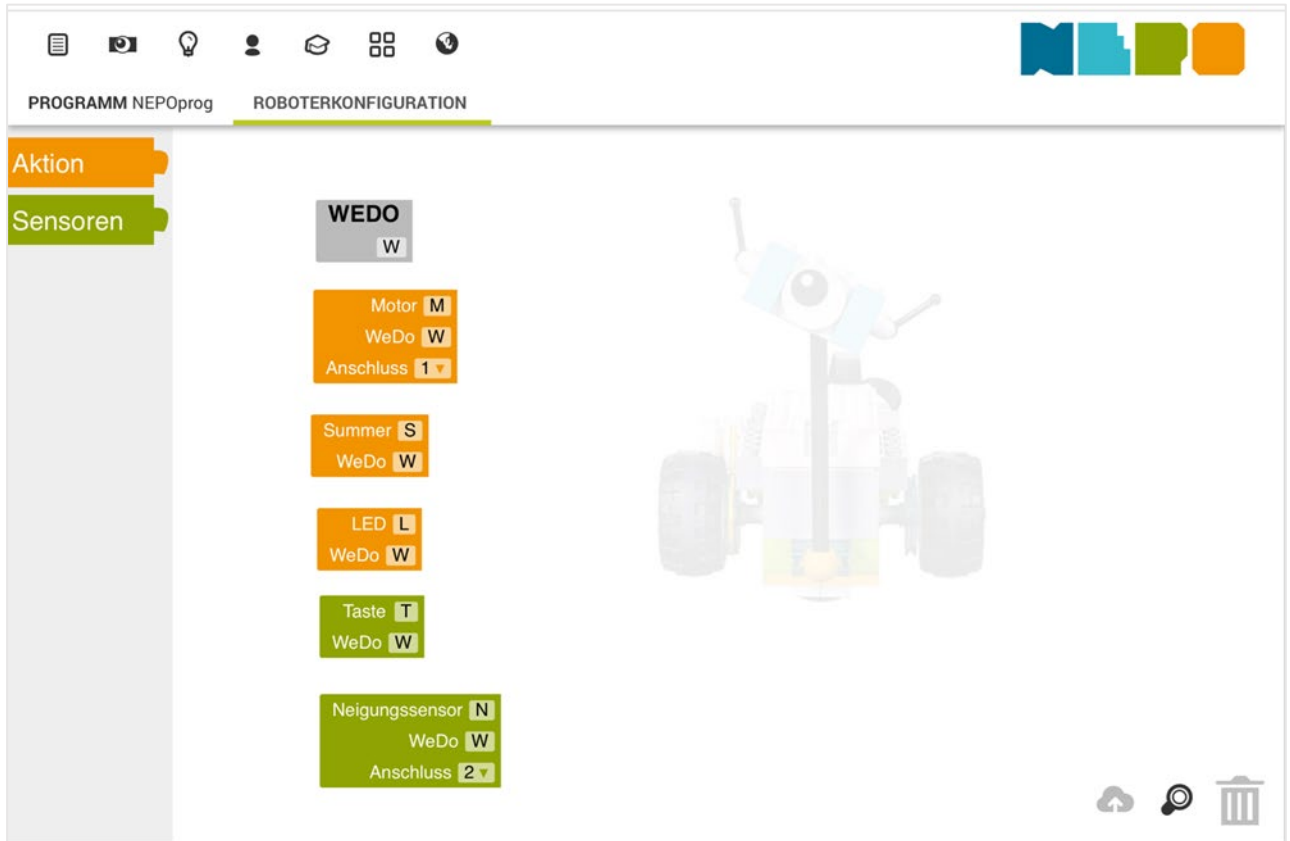


Abbildung 3: Die Roboterkonfiguration

3. Programmierung

Zielsetzung: Wenn der »grüne Balken« mit dem Neigungssensor aus der Waagerechten herausbewegt wird, soll er nach dem Starten des Programms wieder in die Waagrechte zurückgedreht werden.

Wichtig: Erst den Balken verdrehen, dann das Programm starten!

Vorab: Informationen zum Neigungssensor

Der WeDo-Neigungssensor kann sechs verschiedene Neigungszustände zurückmelden:

- aufwärts
- abwärts
- hinten
- vorne
- nicht
- irgendwie

Wenn der Sensor wie in Abbildung 1 verwendet wird, benötigt man neben »nicht« geneigt noch »aufwärts« geneigt und »abwärts« geneigt. Bei einem anderen Einbau kann es erforderlich sein, stattdessen »hinten« geneigt und »vorne« geneigt einzustellen.

Phase 1: Aufwärts? Abwärts?

In der ersten Phase klären die Schüler*innen, was mit »aufwärts« geneigt und »abwärts« geneigt gemeint ist. Dies geht mit einem »Wenn« und dem Einsatz der LED am Hub sehr einfach:

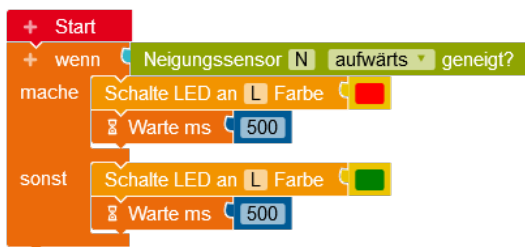


Abbildung 4: Programmcode für Phase 1

Ist das Modell wie in Abbildung 1 gebaut, bedeutet »aufwärts« geneigt, dass der Balken von links unten nach rechts oben weist.

Möglicherweise entdecken manche Schüler*innen, dass die LED bei einer waagerechten Ausrichtung des Balkens ebenfalls grün leuchtet. Diese Feststellung kann zur Erläuterung genutzt werden, dass sonst »aufwärts« – was eine grüne LED letztlich bedeutet – nicht unbedingt »abwärts« bedeutet. Auch eine waagerechte Ausrichtung (= »nicht« geneigt) ist nicht »aufwärts« geneigt.

Phase 2: Die Drehrichtung des Motors bestimmen

An dieser Stelle ist es wichtig, relativ kleine Werte für »Tempo %« und »Zeit ms« einzutragen, damit sich der Balken nicht zu weit dreht. Insbesondere der Standardwert von 1.000 ms muss unbedingt geändert werden. Die in Abbildung 5 eingestellten Werte haben sich in der Praxis als nützlich erwiesen.

Um die Drehrichtung des Motors zu bestimmen, ist der Neigungssensor nicht unbedingt erforderlich. In seiner einfachsten Form lautet der Programmcode so:



Abbildung 5: Programmcode für Phase 2

Hinweis: Bezogen auf Abbildung 1 bedeutet ein positiver Wert bei »Tempo %« eine Drehung gegen den Uhrzeigersinn.

Phase 3: Die Logik des Regelkreises

Es ist vorteilhaft, die Logik des Regelkreises vor der eigentlichen Programmierung nochmals anzusprechen. Ob die Fachbegriffe wie Regelkreis, Soll-Wert usw. dabei verwendet werden, ist von der Klassenstufe bzw. vom Kontext abhängig. Man kann auch problemlos darauf verzichten.

Bis hierher wurde Folgendes festgestellt:

1. Das Ziel ist es, dass sich der Balken am Ende des Programmdurchlaufs in der Waagerechten befindet. Dies kann mithilfe des Neigungssensors und der Einstellung »nicht« geneigt ermittelt werden.
2. Ist der Neigungssensor »aufwärts« geneigt, muss der Balken im Uhrzeigersinn gedreht werden, d. h. der Motorwert bei »Tempo %« ist negativ.
3. Ist der Neigungssensor »abwärts« geneigt, muss der Balken im Gegenuhrzeigersinn gedreht werden, d. h. der Motorwert bei »Tempo %« ist positiv.

Phase 4: Die Tücken des Programmierens

Um in Punkt 1 Sensorwerte wiederholt abzufragen, bietet sich die »Wiederhole bis«-Schleife an, die sich im Register »NEPO-Blöcke Experte« befindet. Die Punkte 2 und 3 lassen sich mit einem »wenn-mache-sonst« umsetzen.

Als »tückisch« können sich die folgenden Punkte erweisen:

- Sind die Werte bei »Tempo %« und/oder »Zeit ms« zu groß gewählt, resultiert daraus ein großer Drehwinkel. Dann kann es passieren, dass sich der Balken an der Waagerechten vorbeidreht. Mit anderen Worten: Der Balken pendelt zwischen »aufwärts« geneigt und »abwärts« geneigt hin und her, erreicht aber nie den Zustand »nicht« geneigt. Das Programm ist in einer Endlosschleife gefangen.

- Wird nach dem Motor-Befehl kein Warte-Befehl eingebaut, reagiert die Motorsteuerung ungenau. Dies kann trotz gut gewählter Parameter für „Tempo %“ und „Zeit ms“ zum selben Ergebnis führen: Der Balken dreht an der Waagrechten vorbei, pendelt hin und her und das Programm hängt in einer Endlosschleife.

Ob Sie die Schüler*innen vorab über diese „Tücken“ informieren oder es vorziehen, dass sie diese Erfahrungen selbst machen und ggf. auch selbstständig auf die Lösung kommen, ist von der methodisch-didaktischen Konzeption der Unterrichtsdurchführung abhängig.

Phase 5: Der programmierte Regelkreis

Sind alle oben genannten Phasen durchlaufen, sieht das fertige Programm wie folgt aus:

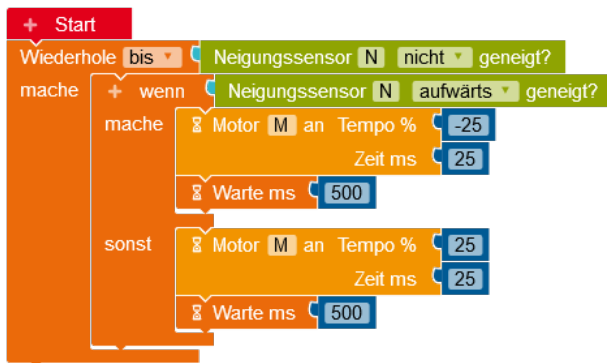


Abbildung 6: Die Programmierung für den Regelkreis

Natürlich kann im Wenn-Teil auch auf »abwärts« geneigt geprüft werden, dann muss allerdings die Drehrichtung des Motors (d. h. das Vorzeichen) im Mache- und im Sonst-Abschnitt verändert werden.

4. Bezug zum Bildungsplan

Die vorliegende Unterrichtseinheit bietet sich unmittelbar an, wenn technische oder biologische Regelkreise behandelt werden. Weitere Anknüpfungspunkte und Kompetenzen werden im Folgenden genannt.¹

Klasse 3/4

Mathematik

Beim Arbeiten im Team und beim Erarbeiten der Programme entwickeln die Schüler*innen die prozessbezogenen Kompetenzen »Kommunizieren« und »Argumentieren« sowie »Problemlösung« und »Modellieren«.

Durch die Eingabe verschiedener Werte bei der Motorsteuerung festigen die Schüler*innen ihr Vorstellungsvermögen zu Raum und Größen.

Sachunterricht

Durch den Bezug auf zugrundeliegende Phänomene wie z. B. das Thermostat bei modernen Heizungen, entwickeln die Schüler*innen die prozessbezogenen Kompetenzen »Welt erleben und wahrnehmen« und »Welt erkunden und verstehen«. Durch das Programmieren wird zudem »In der Welt handeln – Welt gestalten« angesprochen.

Weiterhin kann das Thema Wärmeregulation als Experiment im Bereich »Körper und Gesundheit« behandelt und als Basis für die Programmierung genutzt werden.

¹ Als Grundlage dienen die Bildungspläne des Landes Baden-Württemberg, siehe <http://www.bildungsplaene-bw.de>

Klasse 5/6

Biologie, Naturphänomene und Technik

Beim Arbeiten im Team und beim Erarbeiten der Programme entwickeln die Schüler*innen die prozessbezogenen Kompetenzen »Erkenntnisgewinnung«, »Kommunikation«, »Bewertung« und »Herstellung«.

Inhaltlich kreieren die Schüler*innen eigene Wege, um die ihnen gestellte (Programmier-)Aufgabe zu lösen. Anhand eines konkreten Beispiels lernen sie, dass Menschen technische Objekte anfertigen, um eine bestimmte Aufgabe zu erfüllen.

Mathematik

Beim Arbeiten im Team und beim Erarbeiten der Programme entwickeln die Schüler*innen die prozessbezogenen Kompetenzen »Argumentieren«, »Problemlösung«, »Modellieren«, »mit symbolischen, formalen und technischen Elementen (der Mathematik) umgehen« sowie »Kommunizieren«.

Inhaltlich lassen sich die folgenden Leitideen wiederfinden:

- Zahl – Variable – Operation
- Messen
- Raum und Form
- Funktionaler Zusammenhang

Kontakt

Die Roberta-Initiative im Web

roberta-home.de

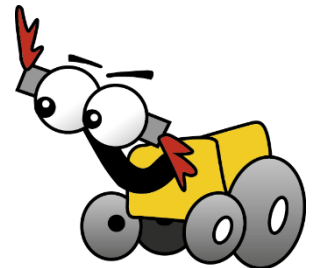
lab.open-roberta.org

FAQ rund um die Roberta-Initiative

roberta-home.de/faq

Informationen zum Datenschutz

roberta-home.de/datenschutz



Info

Dieses Material wurde zusammen mit Prof. Dr. Julia Knopf und Prof. Dr. Silke Ladel entwickelt.

Dieses Material entstand mit Unterstützung der Google Zukunftswerkstatt.

Lizenz: CC-BY-SA 4.0

Version: 1.2

Stand: Januar 2020

Warenzeichen

Roberta, Open Roberta und NEPO sind eingetragene Warenzeichen der Fraunhofer-Gesellschaft e.V.

Lego ist ein eingetragenes Markenzeichen der Lego Group.