



Roboter unterwegs

Stand: 12.07.2017

Jahrgangsstufen	Lernbereich 2: Modul 2.7.2
Fach/Fächer	Informationstechnologie
Übergreifende Bildungs- und Erziehungsziele	Technische Bildung
Zeitraumen	2 Doppelstunden
Benötigtes Material	Entwicklungsumgebung Open Roberta Lab: https://lab.open-roberta.org (kostenlos verfügbar über Webbrowser, kein Download/keine Installation erforderlich)

Kompetenzerwartungen

Die Schülerinnen und Schüler...

- beschreiben das Zusammenspiel von Bauteilen (z. B. Sensoren, Aktoren) und die Informationsverarbeitung eines Roboters bzw. eines eingebetteten Systems und dessen situationsbezogene Interaktion mit der physischen Welt.
- erzeugen Modelle, um Zustände und Verhalten eines Roboters bzw. eines eingebetteten Systems zu planen.
- statuen einen Roboter bzw. ein eingebettetes System mit den zur Lösung einer Aufgabe nötigen Bauteilen (z. B. Sensoren, Aktoren) aus.
- implementieren, testen und optimieren ein Programm, um mit dem konstruierten Roboter bzw. eingebetteten System eine Aufgabe zu lösen (z. B. Folgen einer Linie).

Aufgabe

Roberta ist ein kleiner Roboter, der mithilfe seiner Motoren und Räder fahren und dank seiner Sensoren auch seine Umwelt wahrnehmen kann.

Aufgabe 1

Programmiere Roberta so, dass sie autonom (d. h. ohne Steuerung von außen) der schwarzen Linie der Teststrecke folgt und fortlaufend ihre Runden dreht.



Aufgabe 2

Den großen blauen Stein in der „Einfachen Umgebung“ (siehe Abb. rechts) kann man verschieben. Verschiebe ihn vor dem Start so, dass er auf Robertas Weg liegt. Roberta soll, während sie der schwarzen Linie folgt, über einen geeigneten Sensor bzw. geeignete Sensoren den blauen Stein als Hindernis erkennen und darauf wie folgt reagieren:



Abb.: Open Roberta Lab: Einfache Umgebung

- stoppen,
- dann ein Audiosignal ausgeben,
- ein kleines Stück rückwärtsfahren,
- und kurz stehen bleiben.

Zum Open Roberta Lab: <https://lab.open-roberta.org>

Erste Schritte im Open Roberta Lab:

<https://mp-devel.iais.fraunhofer.de/wiki/display/ORInfo/Open+Roberta+Wiki>

Hier eine Beschreibung der exakten Roboterkonfiguration von Roberta (es wurde beim Start des Open Roberta Lab die NXT-Umgebung ausgewählt):

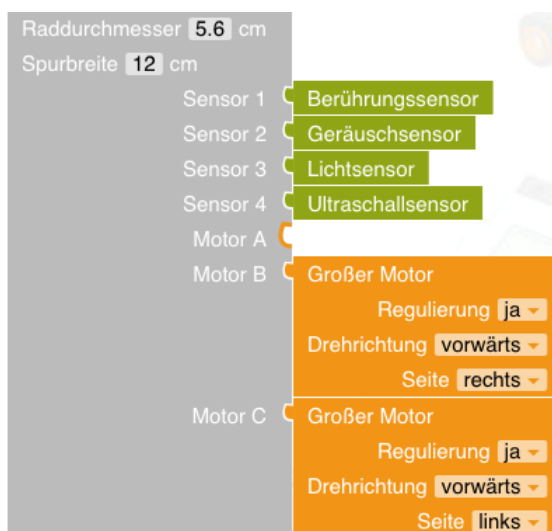


Abb.: Grundkonfiguration des Roboters (NXT-Umgebung)

Hilfestellung (gestufte Hilfestellungen im Sinne der inneren Differenzierung)

Zu Aufgabe 1

Stufe 1: Keine Hilfestellung

Die Schülerinnen und Schüler entdecken und erarbeiten sich alles Nötige selbstständig anhand des Open Roberta Wikis und der Hilfefunktion des Open Roberta Labs.

Stufe 2: Mittlere Hilfestellung

Die Schülerinnen und Schüler erhalten zusätzlich folgenden unvollständigen Programmablaufplan, den sie vervollständigen (??? ersetzen), bevor sie mit der Implementierung beginnen.

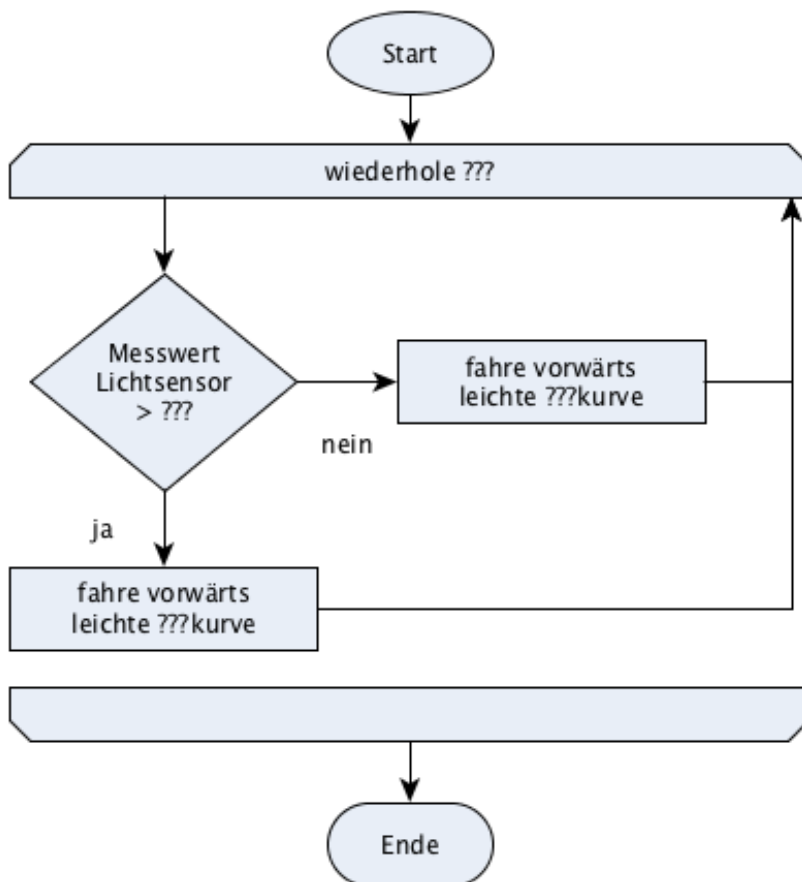


Abb.: Zu vervollständigender Programmablaufplan Aufgabe 1

Den Schwellenwert für die Abfrage des Lichtsensors kann man ermitteln, indem man den Messwert des Lichtsensors einmal bei schwarzem und einmal bei weißem Untergrund erfasst. Das kann man ebenfalls ohne eigenen Roboter in der Open Roberta Lab Simulationsumgebung machen (vgl. Abb. unten: Open Roberta Lab: Roboter-Display einblenden):

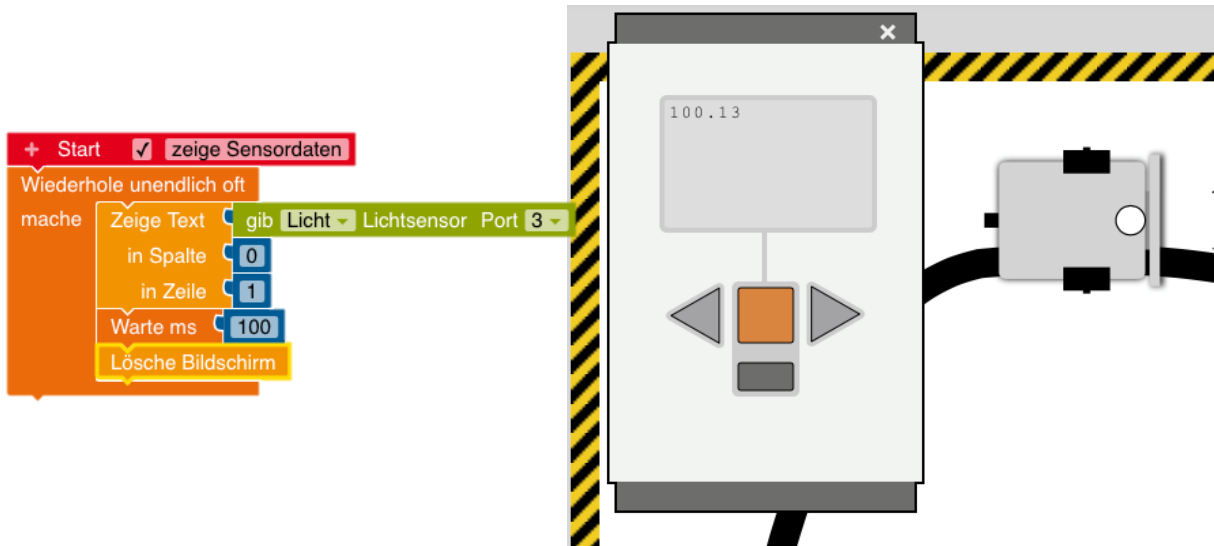


Abb.: Messwert Lichtsensor erfassen – im Display: Messwert bei weißem Untergrund

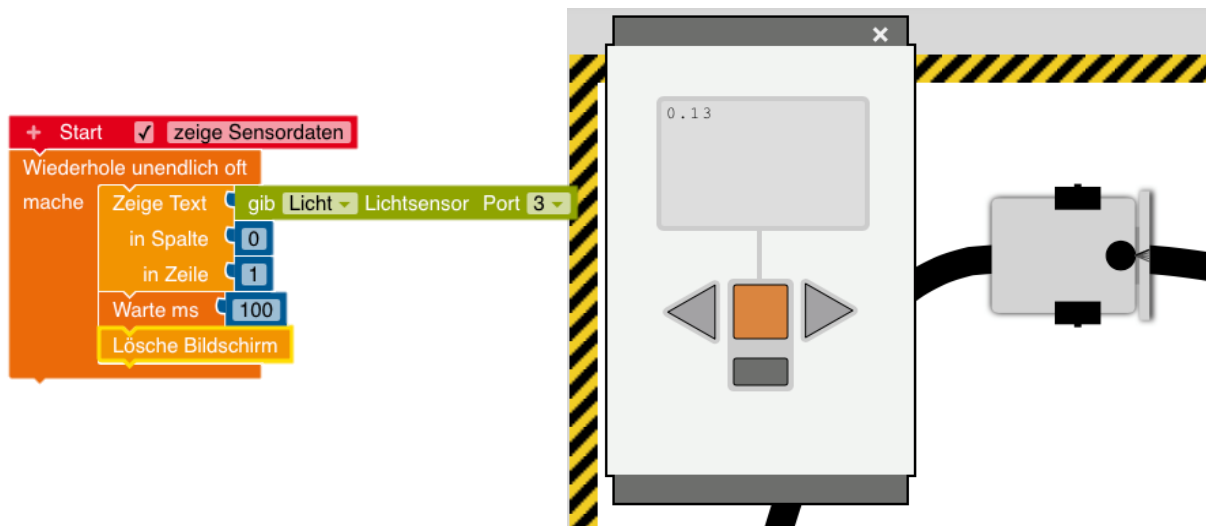


Abb.: Messwert Lichtsensor erfassen – im Display: Messwert bei schwarzem Untergrund

Bildet man den Mittelwert aus den beiden Messwerten, so kommt man auf den im Lösungsbeispiel verwendeten Schwellwert von 50.

Um das Display von Roberta in der Simulation anzuzeigen, muss man z. B. in der NXT-Umgebung im Simulationsmodus auf „NXT“ klicken:



Abb.: Open Roberta Lab: Roboter-Display einblenden

Stufe 3: Umfangreiche Hilfestellung

Zusätzlicher Tipp für Schüler:

Lass den Roboter in Abhängigkeit des gemessenen Lichtwertes eine leichte Rechts- bzw. Linkskurve fahren. Die Entscheidung, ob eine Rechts- oder Linkskurve gefahren wird, soll unendlich oft hintereinander getroffen werden.

Zu Aufgabe 2

Stufe 1: Ohne Hilfestellung

Die Schülerinnen und Schüler entdecken und erarbeiten sich alles Nötige selbstständig anhand des Open Roberta Wikis und der Hilfefunktion des Open Roberta Labs.

Stufe 2: Mit Hilfestellung

Die Schüler erhalten folgende Zusatzinformationen:

Anhand eines Ultraschallsensors kann man Entfernungen messen. Der Messwert des Sensors entspricht in etwa der Entfernung des nächstgelegenen Gegenstandes zum Ultraschallsensor in Zentimetern (vgl. Ultraschalleinsatz der Fledermaus).

Folgender Programmablaufplan wird den Schülern bereitgestellt:

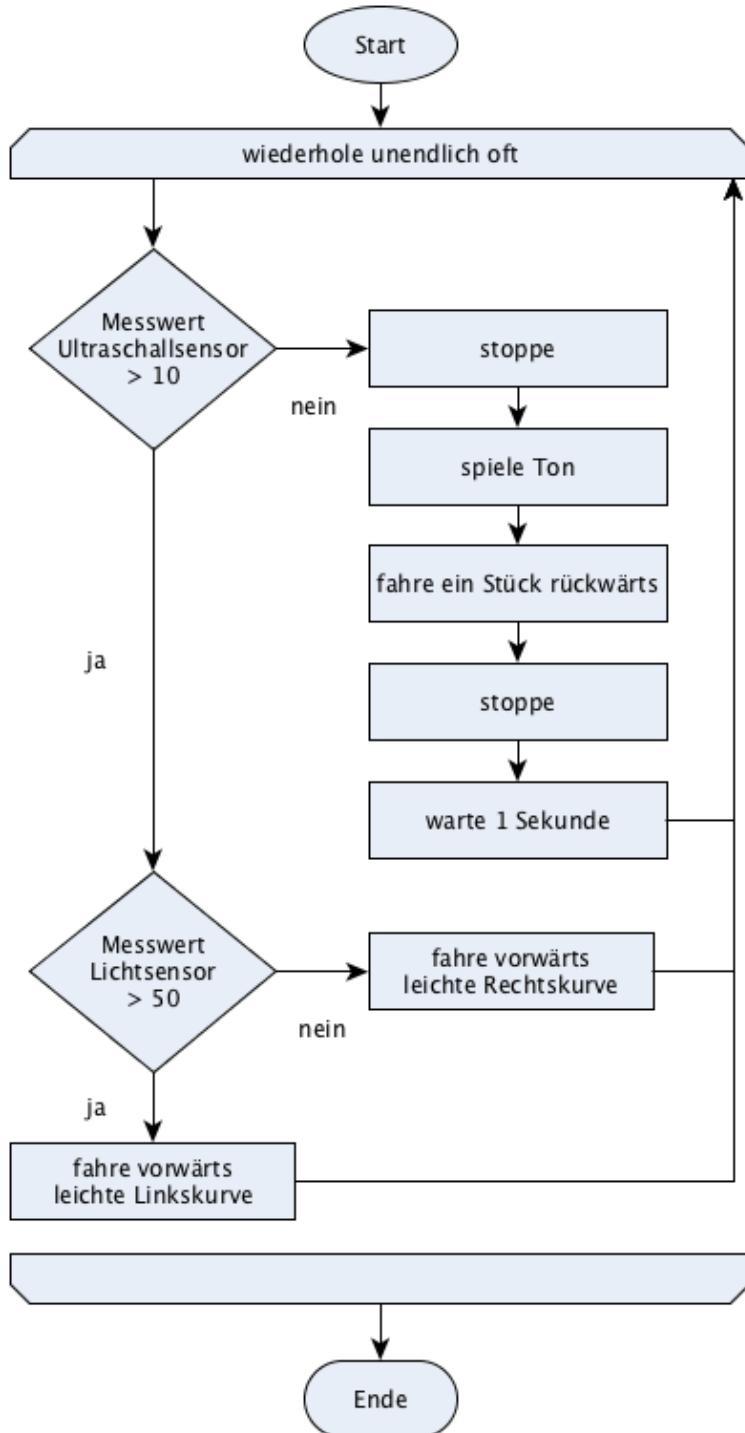


Abb.: Programmablaufplan Aufgabe 2

Hinweise zum Unterricht

Diese Aufgabe greift Kompetenzen und Inhalte aus dem Modul 2.6.1 auf:

- Algorithmus: Begriff und Beispiele
- algorithmische Grundstrukturen: Anweisung, Sequenz, Auswahl und Wiederholung
- Notationsformen, z. B. Programmablaufplan, Struktogramm, Pseudocode, Aktivitätsdiagramm
- Variablenkonzept: Bezeichner, Datentypen, Wertzuweisung

Eine Verwendung am Anfang des Moduls 2.7.2 ist möglich.

Wichtiger Hinweis: Die Aufgaben und die Lösungshinweise beziehen sich auf die NXT-Umgebung des Open Roberta Lab. Die Umgebung – z. B. NXT, EV3 u. a. ist nach Aufruf des Open Roberta Lab auswählbar.

Alle Aufgaben können direkt im Open Roberta Lab anhand des virtuellen Roboters (Simulationsmodus) umgesetzt und ausprobiert werden.

Eine Aufteilung der Gesamtaufgaben in Analyse, Entwurf, Implementierung und Test/Anpassung ist empfehlenswert.

Wählt man für die Schüler den Weg der Aufgabenbearbeitung ohne Hilfestellung, dann sollten die Schülerinnen und Schüler in der Entwurfsphase selbst einen Programmablaufplan erstellen oder eine entsprechende Modellierungstechnik anwenden.

Hinweise zu Aufgabe 1

Das Lösungsbeispiel unten ist so programmiert, dass sich der Roboter immer am rechten Rand der schwarzen Linie orientiert, egal in welche Richtung er fährt. Das ergibt sich aus dem Lösungsansatz Roberta bei weißem Untergrund (Messwert Lichtsensor > 50) eine leichte Linkskurve, bei schwarzem Untergrund (Messwert Lichtsensor ≤ 50) eine leichte Rechtskurve fahren zu lassen.

Bitte beachten Sie, dass die Startposition von Roberta in der Simulationsumgebung so gewählt werden muss, dass der Roboter die schwarze Linie aufgrund seines Wenderadius erreichen kann. Außerdem ist dabei zu beachten, dass der Winkel zur schwarzen Linie nicht zu stumpf sein darf, wenn Roberta die Linie berührt. Sonst funktioniert der im Lösungsbeispiel verwendete Algorithmus nicht.

Lösungsbeispiel Aufgabe 1

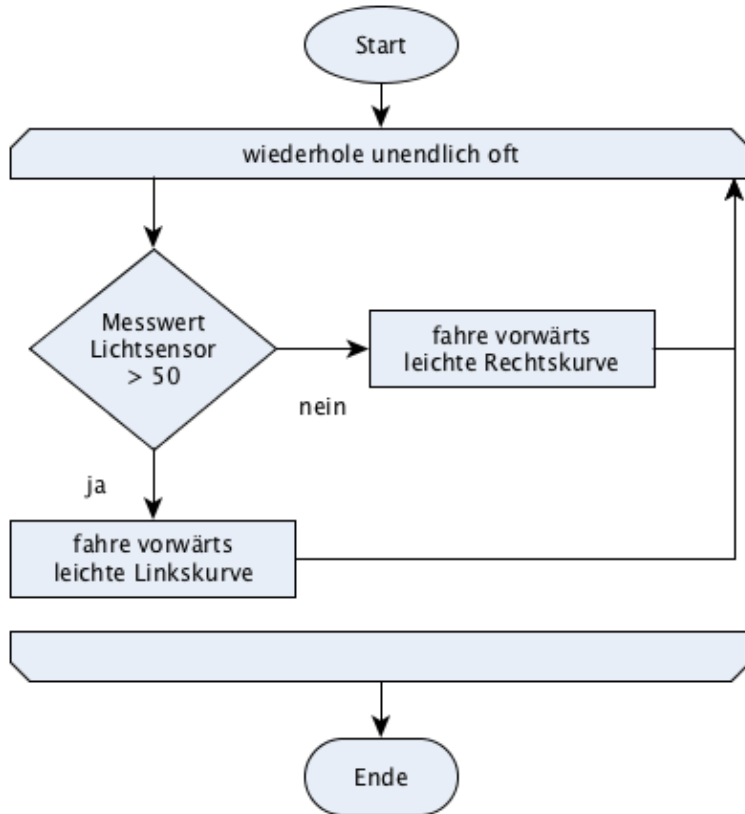


Abb.: Vollständiger Programmablaufplan zu Aufgabe 1

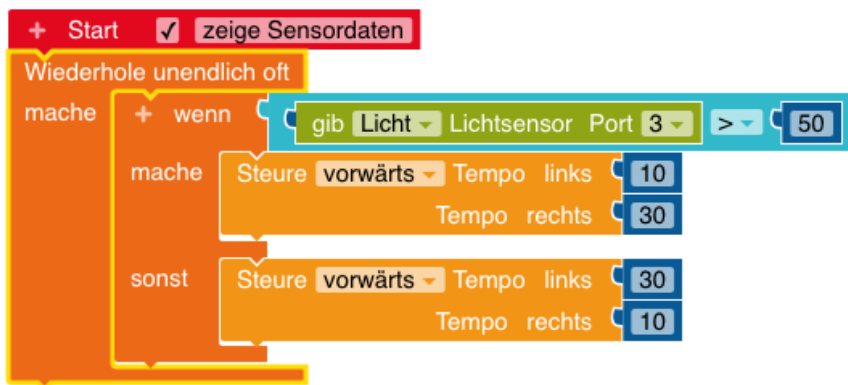


Abb.: Open Roberta Lab: Lösungsbeispiel zu Aufgabe 1

Hinweise zu Aufgabe 2

Der Programmteil mit der Abfrage des Lichtsensors entspricht der Lösung aus Aufgabe 1 und kann wiederverwendet werden.

Lösungsbeispiel Aufgabe 2

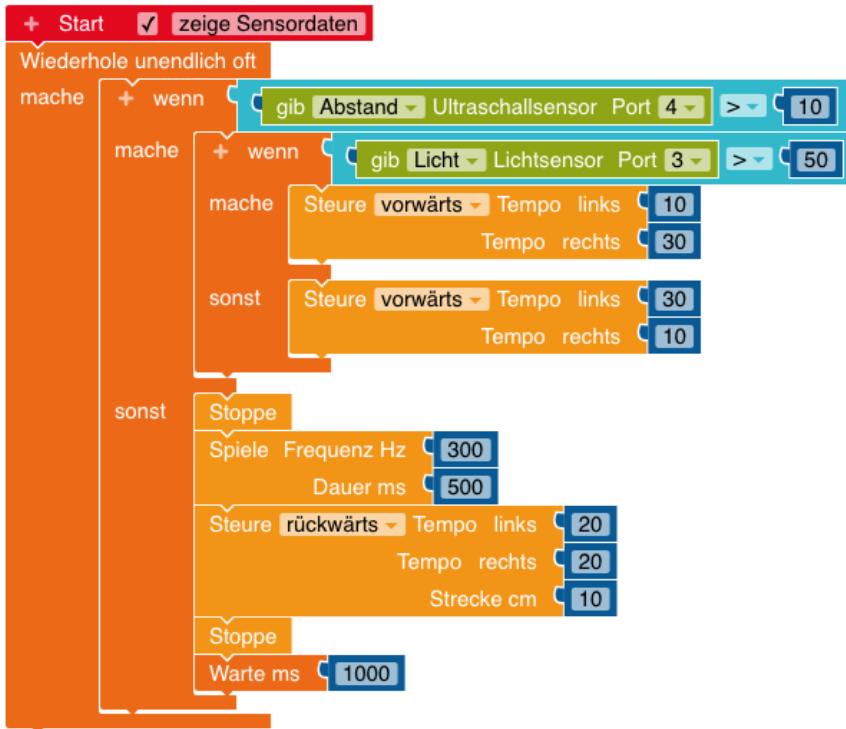


Abb.: Open Roberta Lab: Lösungsbeispiel zu Aufgabe 2

Anregung zum weiteren Lernen

Die im Lösungsbeispiel vorgeschlagene Umsetzung von Aufgabe 1 erfüllt zwar ihren Zweck, ist aber nicht besonders elegant, da der Roboter beim Fahren ständig leicht ruckelt. Das Entwickeln eines „fortgeschrittenen Linienverfolgers“ wäre eine anspruchsvolle Folgeaufgabe.

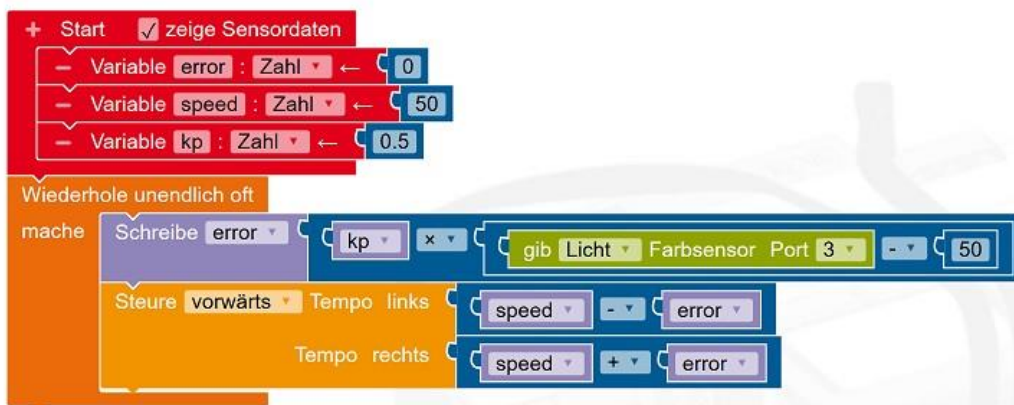


Abb.: Lösungsbeispiel für einen fortgeschrittenen Linienverfolger



Erläuterung zum Lösungsbeispiel

Der fortgeschrittene Linienvorfolger folgt der schwarzen Linie besonders elegant, da die Richtungsänderung nicht ruckartig, sondern fortlaufend und annähernd stufenlos erfolgt.

Über die Variable „error“ kann man die Schwere der Abweichung vom Normkurs erfassen (den aktuellen Wert der Variable kann man sich ggf. auf dem Roboter-Display ausgeben lassen).

Die Drehgeschwindigkeit des linken bzw. rechten Motors wird über die Variable „speed“ initial auf 50 % gesetzt und entsprechend der Abweichung vom Normkurs über den Wert der Variable „error“ fortlaufend leicht nach oben oder unten angepasst.

Die Variable „kp“ ist ein konstanter Faktor für die Berechnung der Variable „error“ über den Messwert des Lichtsensors.

Je höher der Messwert des Lichtsensors, desto höher der Wert der Variable „error“. Der aktuelle Wert von „error“ wird fortlaufend berechnet, indem man vom Messwert des Lichtsensors 50 subtrahiert und mit der Konstante „kp“ multipliziert. Dadurch wird der Wert von „error“ relativ klein und die damit verbundene Richtungsänderung sehr fein geregelt. Beim einfachen Linienvorfolger gab es im Gegensatz dazu nur die beiden Reaktionen „Leichte Linkskurve“ oder „Leichte Rechtskurve“.

Quellen und Links

Open Roberta Wiki

<https://mp-devel.iais.fraunhofer.de/wiki/display/ORInfo/Open+Roberta+Wiki>

(Stand: 12.07.2017)

Open Roberta Lab

<https://lab.open-roberta.org>

(Stand: 12.07.2017)