# Aufgabe 6: Buggy mit Hinderniserkennung

Der kleine dreirädrige Buggy bewegt sich eigenständig im Raum, kann Hindernisse und Begrenzungslinien erkennen und weicht diesen aus.

## Thema

Synchrone Steuerung zweier Motoren, Erkennung von Hindernissen und Navigation.

## Lernziel

* Verständnis der synchronen Motorsteuerung
* Hindernis- und Begrenzungserkennung durch Sensoren
* Navigation über Encoderimpulse

## Zeitaufwand

Die Konstruktion des Buggys nach Anleitung sollte den Schülerinnen und Schülern innerhalb von 30-45 Minuten gelingen. Das Grundmodell wird auch in den beiden folgenden Aufgaben 7 und 8 verwendet.

Für die Entwicklung des Steuerungsprogramms zur Lösung der Programmieraufgaben benötigen Schülerinnen und Schüler abhängig von ihren Vorerfahrungen in der Programmierung bis zu 90 Minuten. Eine Erläuterung der Funktionsweise von Ultraschall-Sensoren (Schallgeschwindigkeit in unterschiedlichen Ausbreitungs­medien) wird empfohlen.

Die erste und zweite Experimentieraufgabe sind mit den Vorkenntnissen aus den bisherigen Aufgaben ebenfalls leicht zu lösen (45-90 Minuten).

Etwas komplexer ist die Lösung der dritten Experimentieraufgabe, in der der Buggy mit einer einfachen Zielnavigation ausgestattet wird. Diese Aufgabe wird für erfahrenere oder ältere Schülerinnen und Schüler (Sekundarstufe II) empfohlen.

## Bezug Curriculum

|  |  |  |
| --- | --- | --- |
| Land | Stufe/Fächer | Bezüge |
| BW | SEK 1 | IMP 8-3.1.1.2 Algorithmen (1), S. 28ff; INFWF 8-3.1.2 Algorithmen (1), S. 15; INFWF 9-3.2.2 Algorithmen (2), S. 21; INFWF 10-3.3.2 Algorithmen (2), S. 28 |
| BY | SEK 1 | RS- IT 2.7 Logik und Robotik, S.699; GYM 9/10 LPLUS INF - Modellieren, Implementieren, Anwenden, Softwareprojekte |
| BE | SEK 1 | INF-3.5 Algorithmisches Problemlösen, S. 25; INF-3.9 Physical Computing (Wahlthemenfeld), S. 27 |
| BB | SEK 1 | INF-3.5 Algorithmisches Problemlösen, S. 25; INF-3.9 Physical Computing (Wahlthemenfeld), S. 27 |
| HB | SEK 2 | GYM OS INF-Algorithmen und Datenstrukturen, S. 6; GYM OS INF-Imperative Programmierung, S. 7; GYM OS INF-Sprachen und Automaten, S. 7 |
| HH | SEK 1 | GYM 7/8 INF-M1 Informatik strukturieren und präsentieren, S. 18; GYM 9 INF-M2 Prozesse analysieren und modellieren, S. 19; SEK1 GYM 10 INF-M3 Daten und Prozesse, S. 20 |
| HE | SEK 2 | GYM KCGO INF-E1/E2 Einführung in die Informatik, S. 26; GYM KCGO INF-Q1 Algorithmik und objektorientierte Modellierung, S. 26; GYM KCGO INF-Q3 Konzepte und Anwendungen der theoretischen Informatik, S. 26 |
| MV | SEK 1 | GYM 5 INF-3 Programmieren? Kinderleicht!, S.16; GYM 6 INF-3 Entscheidungen treffen und Spiele gestalten, S.19 GYM 7 INF-3 Spiele entwickeln, S.22; GYM 8 INF-3 Sensorgesteuerte Anwendungen entwickeln, S.25; GYM 9 INF-3 Problemlösen durch Programmieren, S.32 |
| NI | SEK 1 | KC-INF LF Algorithmisches Problemlösen; S.19; KC-INF LF Automatisierte Prozesse; S.22; SEK 2 KC-INF LF1 Algorithmen und Datenstrukturen, S.14; SEK 2 KC-INF LF1 Informationen und Daten, S.16 |
| NW | SEK 1, 2 | RS 9/10 WPF TECHNIK 2.3 Inhaltsfeld 7: Kommunikations- und Digitaltechnik S.23; 5/6 KLP INF - Algorithmen, S. 17, 18; 5/6 KLP INF - Automaten und künstliche Intelligenz, S. 18; SEK 2 KLP GOS INF - 2 Algorithmen, S. 21 ff; KLP GOS INF - 3 Formale Sprachen und Automaten, S. 22 |
| RP | SEK 1 | IGS/GYM INF-2.2 Algorithmisches Problemlösen, S. 20 |
| SL | SEK 1, 2 | GYM 9 INF - Algorithmik, S. 3; GYM 9 INF - Imperative Programmierung, S. 3; GYM INF GOS-Strukturiertes Programmieren, S.4ff. |
| SN | SEK I, II | GYM 7 INF WB 3: Computer verwenden, S. 8; SEK1 GYM 7/8 INF-Ziele: Modellieren von Zuständen und Abläufen, S. 6, S.10; GYM 11/12 INF LB 4: Algorithmen und Programme, S. 27; |
| ST | SEK I, II | GS 9/10 Technik-3.4 Kompetenzschwerpunkte - Technische Prozesse steuern, S. 20; GYM 9 INF 3.2 Algorithmen interpretieren und entwickeln, S.15; GYM 11/12 INF 3.4 Kurs 3 Software-Engineering und Projektarbeit, S. 23 |
| SH | SEK I | INF KB3 Algorithmen, INF PB1 Modellieren und Strukturieren, S. 12; INF PB2 Implementieren, Programmieren, Realisieren, S. 13 |
| TH | SEK 1 | GS 9/10 Technik-2.2.3 Untersuchen und Handhaben von Steuerungen und Reglungen (Modellbau), S. 12; GYM 10 INF - 5 Modellierung und Problemlösen; GYM 10 INF - 2.3 Algorithmen, S. 14 ff.; GYM 10 INF 2.5.1 Technische Informatik, S. 18ff. |

Anlagen

# Buggy mit Hinderniserkennung

## Erforderliches Material

* PC für Programmentwicklung, lokal oder über Web-Schnittstelle.
* USB-Kabel oder BLE- bzw. WLAN-Verbindung für die Übertragung des Programms auf den TXT4.0.
* Parcours-Bogen mit schwarzer, 2 cm breiter Kreislinie
* Hindernis (Karton, Dose, …)

## Weiterführende Informationen

[1] Online-Diagrammeditor zur Erstellung von Zustandsübergangsdiagrammen (Format drawio): <https://www.diagrammeditor.de/>