# Aufgabe 8: Spurfolger

Die (mechanische) Konstruktion des Spurfolgers baut auf dem Grundmodell des Buggy aus Aufgabe 6 auf. Er wird im Verlauf der Aufgabe zusätzlich mit einem Ultra­schallsensor zur Hinderniserkennung, zwei Frontstrahlern und für eine analoge Regelung mit der Kamera ausgestattet.

## Thema

Digitale Steuerung (Endliche Automaten) und analoge Regelung eines Fahrzeugs (Roboters) entlang einer Spur.

## Lernziel

* Funktionsweise digitaler Sensoren
* Einfache Zweipunktregelung unter Verwendung digitaler Sensoren
* Ergänzung einer Hinderniserkennung (Ultraschall-Entfernungsmessung)
* Ergänzung einer Steuerung durch Farbflächen
* Analoge Regelung unter Verwendung einer Kamera mit Bildauswertung
* P- und PD-Regler

## Zeitaufwand

Die Konstruktion des Spurfolgers nach Anleitung sollte den Schülerinnen und Schülern innerhalb von max. 45 Minuten gelingen; Schülerinnen und Schüler mit fischertechnik-Erfahrung benötigen weniger Zeit. Sofern der Buggy aus Aufgabe 6 bereits zusam­mengebaut ist, erfordert der Anbau und Anschluss der Sensoren lediglich wenige Minuten.

Für die Entwicklung des Steuerungsprogramms zur Lösung der ersten beiden Programmieraufgaben (digitaler Spurfolger) benötigen Schülerinnen und Schüler maximal 90 Minuten. Für die Hindernisumfahrung sind etwa weitere 45 Minuten zu veranschlagen, für die Farbflächensteuerung 45-90 Minuten.

Die Experimentieraufgaben (analoger P- und PD-Regler) sind aufwändiger. Sofern Grundkenntnisse über Regler (P-, D-Glied) vorausgesetzt werden können, liegt der Aufwand für die Programmierung und vor allem die Einstellung der Regler bei 90-180 Minuten.

## 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, SEK 2 | INF-3.5 Algorithmisches Problemlösen, S. 25; INF-3.9 Physical Computing (Wahlthemenfeld), S. 27; SEK2 11-13 Informatik-4.4 Sprachen und Automaten, S. 22 |
| BB | SEK 1 | INF-3.5 Algorithmisches Problemlösen, S. 25; INF-3.9 Physical Computing (Wahlthemenfeld), S. 27; SEK2 11-13 Informatik-4.4 Sprachen und Automaten, S. 22 |
| 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 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; SEK 2 KC-INF LF1 Automaten und Sprachen, S.19 |
| 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 1, 2 | 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 1, 2 | 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 1 | 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

# Aufgabe 8: Spurfolger

## 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 geschlossener Kreislinie und Farbflächen
* Hindernis (Karton, Dose, …)

## Weiterführende Informationen

[1] Wikipedia: [*Endlicher Automat (Zustandsautomat)*](https://de.wikipedia.org/wiki/Endlicher_Automat)

[2] Ferdinand Wagner, Ruedi Schmuki, Thomas Wagner, Peter Wolstenholme: [*Modeling Software with Finite State Machines. A Practical Approach*](http://is.ifmo.ru/download/modelingsoftwarewithfinitestatemachinesapracticalapproach.pdf). Auerbach Publications, 2006.

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

[4] Wikipedia: [*Regelungstechnik*](https://de.wikipedia.org/wiki/Regelungstechnik).

[5] Wikipedia: [*Regler*](https://de.wikipedia.org/wiki/Regler).

[6] RN-Wissen: [*Regelungstechnik*](http://rn-wissen.de/wiki/index.php/Regelungstechnik1).

[7] Tim Wescott: [*PID without a PhD*](https://www.magentacloud.de/lnk/oyB5kKLU). Embedded Systems Programming, 10/2000, S. 86-108.