

AMS-Projekt Pizzabote

Philipp Hörig und Simon Pawlowski

Projektarbeit • Studiengang Informatik • Fachbereich Informatik und Medien • 18.01.2024

Aufgabenstellung

Das Ziel dieses Projektes ist die Entwicklung eines vollständig autonomen Lieferroboters, der nach den Standards des SAE-Level 5 agiert und in der Lage ist, Pizzen (repräsentiert durch Bälle) auszuliefern. Der Roboter navigiert in einem Gitternetz, in dem allerdings bestimmte Streckenabschnitte gesperrt sein können. Der Roboter muss innerhalb von zwei Minuten bis zu drei Pizzen an unterschiedliche Lieferadressen zustellen. Diese Adressen befinden sich stets am Rand der Karte. Die Herausforderung liegt darin, den Roboter so zu entwickeln und zu programmieren, dass er diese Aufgaben unter Berücksichtigung möglicher Hindernisse und Einschränkungen im Streckennetz effizient und zuverlässig erfüllt.

Streckennetz

Das Streckennetz, dargestellt als ein 7x10 Gitter, wird durch eine Zeichenkette repräsentiert, die oben links bei [0, 0] beginnt und sich bis unten rechts bei [9, 6] erstreckt (siehe Abbildung 1). Es gibt zwei mögliche Startpunkte A und B. Befahrbare Streckenabschnitte im Netz sind durch einen Punkt (.) symbolisiert, wohingegen blockierte Strecken mit einem 'x' gekennzeichnet sind. Lieferadressen werden durch ein 'F' dargestellt.

In Abbildung 2 wird die Struktur des Gitters mithilfe dieser Zeichenkette visualisiert: Die Lieferadressen (H_n) sind als umrandete Kreise abgebildet, während blockierte Strecken durch rote Quadrate hervorgehoben sind.

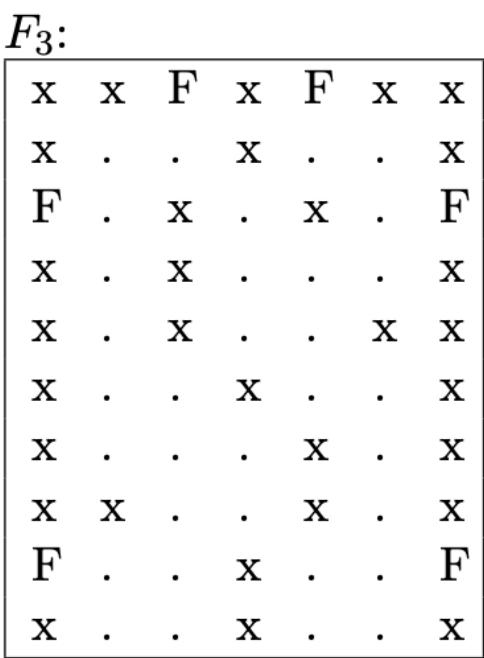


Abb. 1: Fahrauftrag

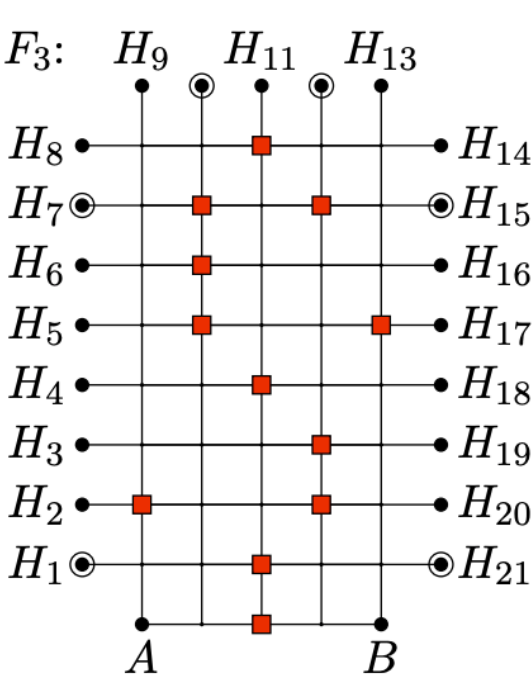


Abb. 2: Streckennetz

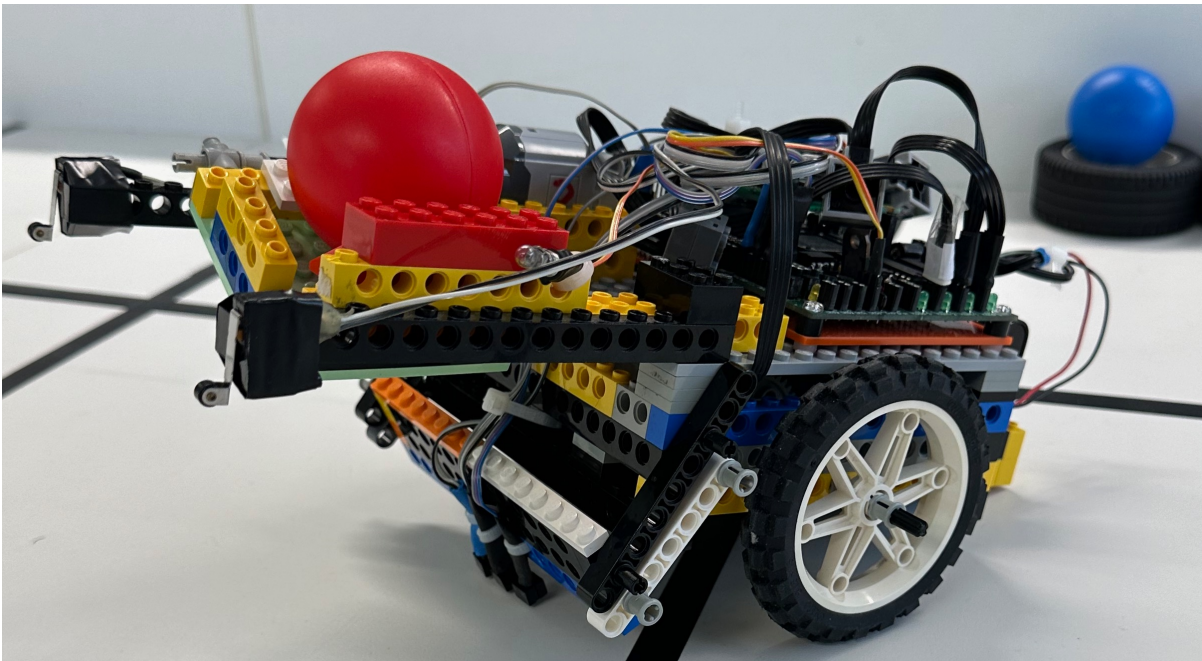
Routenplaner

Der Routenplaner, der auf der Breitensuche basiert, ist speziell für die Ausführung auf dem Mikrocontroller des Roboters konzipiert und in der Programmiersprache C geschrieben. Der Algorithmus startet mit der Erkundung des Startpunktes, während er bereits besuchte Koordinaten markiert und gleichzeitig nach benachbarten Knoten sucht, die sich genau einen Schritt weiter vom Startpunkt befinden. Dabei werden Hindernisse ausgeschlossen. Diese Schritte wiederholen sich,

bis entweder keine weiteren Nachbarn mehr zur Untersuchung zur Verfügung stehen oder das Ziel erreicht wird. Der erste gefundene Pfad in der Breitensuche muss mindestens einen der kürzesten Wege zeigen. Anschließend generiert der Algorithmus eine Zeichenkette, welche als direkte Anweisung für den Fahrtenalgorithmus des Roboters dient.

Aufbau des Roboters

Für den Bau des Roboters wurden verschiedenste Legobausteine, drei Lego M-Motoren, acht Lego-Zahnräder, ein AKSEN-Board, drei Optokoppler, zwei Taster und eine LED verwendet. Die beiden selbstgebauten Getriebe mit einer Übersetzungsrate von 1:5 verleihen dem Roboter sowohl die erforderliche Kraft als auch eine hohe Geschwindigkeit.



Fahrtenalgorithmus

Der Fahrtenalgorithmus, welcher ebenfalls in C geschrieben ist, erhält vom Routenplaner einen Fahrauftrag in Form einer Zeichenkette. Diese Zeichenkette enthält Instruktionen, wie z. B. vorwärts fahren, nach links oder rechts fahren, Pizza be- oder entladen. Um die unterschiedlichen Bewegungen korrekt auszuführen, werden im Abstand von 25ms die Werte der drei Optokoppler ausgewertet, um zu überprüfen in welcher Position sich der Roboter befindet. Dabei liefern diese bei Erkennung der schwarzen Leitlinie (die Strecke) einen Wert von >180 zurück, bei Erkennung der weißen Tischplatte <30. Basierend auf diesen Werten führt der Roboter notwendige Korrekturen durch, um stets auf der vorgegebenen Linie zu bleiben und Drehungen korrekt auszuführen.

Fazit

Die Arbeit mit Mikrocontrollern setzt eine effiziente Ressourcenverwaltung voraus, da nicht immer alle Sprachfeatures, wie z. B. die dynamische Speicherverwaltung, zur Verfügung stehen. Zudem sind alle Bauteile so präzise aufeinander abgestimmt, dass eine noch so kleine Veränderung am Roboter enorm viele Probleme verursachen kann. Aus diesen Erfahrungen haben wir gelernt, dass eine robuste Bauweise und eine gute Planung essenziell sind, damit am Ende das Zusammenspiel aller Komponenten fehlerfrei funktionieren kann.