

Realisierung eines Vier-Gewinnt Roboter

Mithilfe von Lego Spike und Mirco Python

Studienarbeit T3_3100

Studiengang Elektrotechnik

Studienrichtung Automation

Duale Hochschule Baden-Württemberg Ravensburg, Campus Friedrichshafen

von

Patrik Peters / Simon Gschell

| | |
|------------------------|------------------------------|
| Abgabedatum: | 6. November 2024 |
| Bearbeitungszeitraum: | 10.10.2024 - 13.06.2025 |
| Matrikelnummer: | 187 /0815 |
| Kurs: | TEA22 |
| Betreuerin / Betreuer: | Prof. Dr. ing Thorsten Kever |

Erklärung

gemäß Ziffer 1.1.14 der Anlage 1 zu §§ 3, 4 und 5 der Studien- und Prüfungsordnung für die Bachelorstudiengänge im Studienbereich Technik der Dualen Hochschule Baden-Württemberg vom 29.09.2017 in der Fassung vom 24.07.2023.

Ich versichere hiermit, dass ich meine Studienarbeit T3_3100 mit dem Thema:

Realisierung eines Vier-Gewinnt Roboters

selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe. Ich versichere zudem, dass die eingereichte elektronische Fassung mit der gedruckten Fassung übereinstimmt.

Musterstadt, den 6. November 2024

Patrik Peters / Simon Gschell

Kurzfassung

Problemstellung

Ziel der Arbeit

Vorgehen und angewandte Methoden

Konkrete Ergebnisse der Arbeit, am besten mit quantitativen Angaben

In der vorliegenden Studienarbeit wird die Entwicklung eines Roboters für das Spiel „Vier gewinnt“ unter Verwendung des LEGO Spike Prime Systems behandelt. Ziel des Projekts ist es, einen Roboter zu entwerfen, der in der Lage ist, autonom die Rolle eines menschlichen Gegners im Spiel „Vier gewinnt“ zu übernehmen. Der Roboter soll nicht nur die Spielzüge des menschlichen Mitspielers erkennen und darauf reagieren, sondern auch selbstständig Spielsteine in das Spielfeld einwerfen und sicherstellen, dass das Spielfeld für den nächsten Zug bereit ist. Dies erfordert eine präzise Steuerung des Roboters, insbesondere beim Platzieren der Spielsteine, sowie die Fähigkeit, das Spielfeld zu überwachen, um die Position der bereits gesetzten Steine zu erkennen. Das Projekt umfasst verschiedene technische und organisatorische Aspekte. Dazu gehört die mechanische Konstruktion des Roboters, einschließlich des Mechanismus zum Einwerfen der Spielsteine und die Überwachung des Spielfelds, sowie die Entwicklung der Software, die die Spielzüge und das Verhalten des Roboters steuert. Ein weiterer wichtiger Punkt ist die Sensorik: Der Roboter muss in der Lage sein, die aktuelle Spielsituation durch Farbsensoren zu erfassen, um zu wissen, welche Felder im Spielfeld bereits besetzt sind und wo er seinen nächsten Spielstein platzieren kann. Zum Abschluss des Projekts wird ein kleines Turnier organisiert, bei dem die von verschie-

denen Teams entwickelten Roboterlösungen gegeneinander antreten. Dies bietet die Möglichkeit, den Roboter unter realen Bedingungen zu testen und seine Fähigkeiten im direkten Vergleich mit den Lösungen der anderen Kommilitonen zu messen. Um zusätzliche Motivation zu schaffen, wird die beste Lösung am Ende prämiert, was den Wettbewerbsgedanken fördert und den Anreiz erhöht eine effektive Lösungen zu entwickeln.

Abstract

English translation of the „Kurzfassung“.

Inhaltsverzeichnis

| | | |
|----------|--|-----------|
| 1 | Einleitung | 1 |
| 2 | Grundlagen | 3 |
| 2.1 | Spielregeln | 3 |
| 2.1.1 | Vorschriften: | 3 |
| 2.2 | Geschichte | 3 |
| 2.3 | Spieltheorie | 3 |
| 2.4 | Spike | 3 |
| 2.5 | MicroPython | 3 |
| 3 | Vorgehen | 5 |
| 3.1 | Vorgehensweise | 5 |
| 3.2 | Anforderung | 6 |
| 3.3 | Konzept | 6 |
| 3.4 | Nutzwertanalyse | 6 |
| 3.5 | Zeitplan | 6 |
| 4 | Umsetzung und Ergebnisse | 7 |
| 4.1 | Voraussetzungen | 7 |
| 4.2 | Aufbau | 8 |
| 4.3 | Software | 8 |
| 4.3.1 | Herausforderung der Programmierung | 8 |
| 5 | Zusammenfassung | 9 |
| | Abbildungsverzeichnis | 11 |

| | |
|--|-----------|
| Tabellenverzeichnis | 13 |
| A Ergänzungen | 15 |
| A.1 Details zu bestimmten theoretischen Grundlagen | 15 |
| A.2 Weitere Details, welche im Hauptteil den Lesefluss behindern | 15 |
| B Details zu Laboraufbauten und Messergebnissen | 17 |
| B.1 Versuchsanordnung | 17 |
| B.2 Liste der verwendeten Messgeräte | 17 |
| B.3 Übersicht der Messergebnisse | 17 |
| B.4 Schaltplan und Bild der Prototypenplatine | 17 |
| C Zusatzinformationen zu verwendeter Software | 19 |
| C.1 Struktogramm des Programmentwurfs | 19 |
| C.2 Wichtige Teile des Quellcodes | 19 |
| D Datenblätter | 21 |
| D.1 9.Oktober.2024 | 27 |
| Sachwortverzeichnis | 30 |

1 Einleitung

2 Grundlagen

2.1 Spielregeln

2.1.1 Vorschriften:

- max. Zeitbegrenzung eines Zuges: -> scannen des Spielfeldes und Rechenzeit müssen begrenzt sein

2.2 Geschichte

2.3 Spieltheorie

2.4 Spike

2.5 MicroPython

3 Vorgehen

3.1 Vorgehensweise

Zur Realisierung lässt sich dieses Projekt in drei wesentliche Aspekte aufteilen.

- Entwicklung eines mechanischen Konzepts: Es soll eine funktionale Vorrichtung entworfen werden, der in der Lage ist, Spielsteine präzise in den Spielständer einzuführen und diese anschließend für den nächsten Spielzug freizugeben.
- Erfassung der Ist-Situation: Durch den Einsatz geeigneter Sensorik soll der Roboter die aktuelle Position der gelben und roten Chips im Spielstand erfassen und diese Informationen an den Mikrocontroller weiterzugeben.
- Erstellung eines effizienten Algorithmus: Ein in microPython programmierter Algorithmus muss entwickelt werden, der innerhalb der begrenzten Rechenkapazitäten des verwendeten Controllers effizient arbeitet und die nötigen Steuerbefehle für die Roboteraktionen bereitstellt. Für die beste Strategie soll dabei mathematische Spieltheorie analysiert werden und diese in das System eingebunden werden.

3.2 Anforderung

3.3 Konzept

3.4 Nutzwertanalyse

3.5 Zeitplan

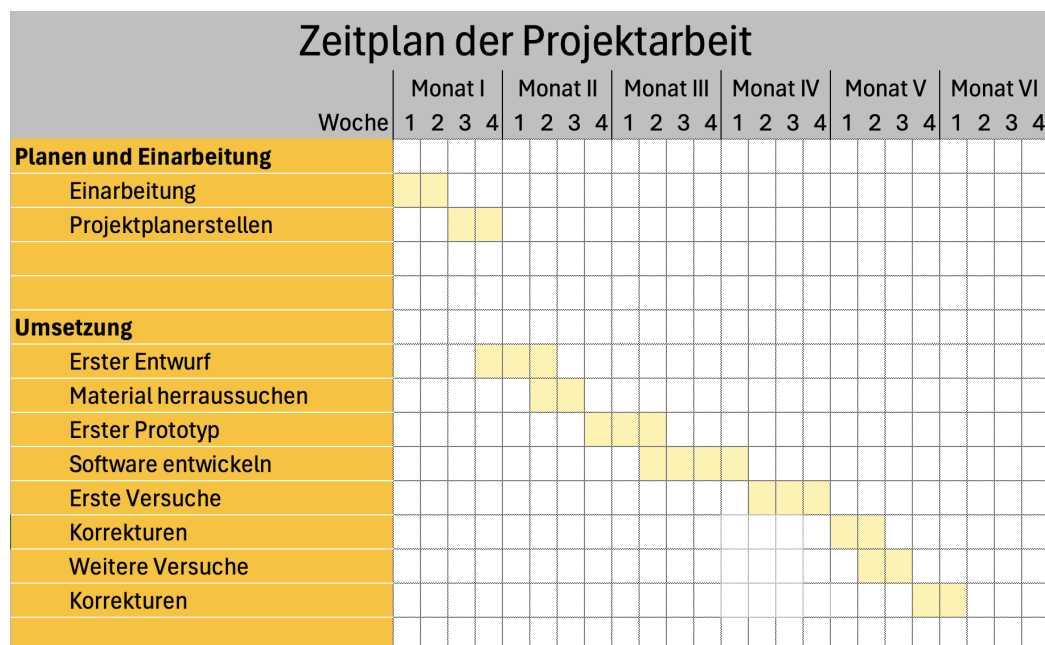


Abbildung 3.1: Zeitplan

4 Umsetzung und Ergebnisse

4.1 Voraussetzungen

Voraussetzung für dieses Projekt ist ein fundiertes Verständnis der Programmierung sowie Kreativität bei der Konstruktion. Diese Fähigkeiten bringen wir durch unsere abgeschlossene Ausbildung im Bereich Mechatronik und Elektronik mit. Darüber hinaus sind Kenntnisse in der Softwareentwicklung erforderlich, insbesondere im Hinblick auf die Programmierung des LEGO Spike Prime Systems, um den Roboter erfolgreich zu steuern und die Interaktion mit der Spielumgebung zu gewährleisten. Die Studienarbeit erstreckt sich über zwei Praxisphasen, was einem Zeitraum von insgesamt sechs Monaten entspricht. In dieser Zeit werden die theoretischen Grundlagen, die während des Studiums erlangt haben, in die Praxis umsetzen, um eine vollständige Robotiklösung zu entwickeln, die den Anforderungen des Projekts gerecht wird.

4.2 Aufbau

4.3 Software

4.3.1 Herausforderung der Programmierung

Die größte Herausforderung bei der Realisierung des Vier-Gewinnt-Roboters besteht in der begrenzten Rechenleistung des verwendeten Mikrocontrollers. Anders als bei leistungsstarken Computern, die in der Lage sind, vollständige Spielalgorithmen zu berechnen und dadurch eine perfekte Strategie zu verfolgen, muss der Roboter mit deutlich eingeschränkten Ressourcen auskommen. Ein umfassender Algorithmus, der jede mögliche Zugkombination im Vier-Gewinnt-Spiel analysiert, erfordert erheblichen Speicherplatz und Rechenleistung, da mit jedem neuen Zug die Anzahl der möglichen Spielverläufe exponentiell ansteigt. Auf einem leistungsstarken Computer wäre es theoretisch möglich, eine "perfekte Strategie zu entwickeln, die den gesamten Spielbaum durchläuft und immer den besten Zug auswählt. Der Mikrocontroller des Roboters hingegen hat nicht die Kapazität, all diese Berechnungen in angemessener Zeit durchzuführen. Aus diesem Grund kann der Roboter nur eine begrenzte Anzahl von Zügen im Voraus berechnen. Er muss sich auf Algorithmen stützen, die in der Lage sind, kurzfristige Entscheidungen zu treffen, anstatt langfristige Strategien zu verfolgen. Das bedeutet, dass der Roboter durch Heuristiken (d.h. Daumenregeln oder Annäherungen) gesteuert wird, die ihm helfen, gute, aber nicht immer optimale Züge zu machen. Diese Heuristiken können einfache Prinzipien wie das Verhindern eines unmittelbaren Sieges des Gegners oder das Setzen eigener Spielsteine in strategisch günstige Positionen umfassen.

5 Zusammenfassung

Auf zwei bis drei Seiten soll auf folgende Punkte eingegangen werden:

- Welches Ziel sollte erreicht werden
- Welches Vorgehen wurde gewählt
- Was wurde erreicht, zentrale Ergebnisse nennen, am besten quantitative Angaben machen
- Konnten die Ergebnisse nach kritischer Bewertung zum Erreichen des Ziels oder zur Problemlösung beitragen
- Ausblick

In der Zusammenfassung sind unbedingt klare Aussagen zum Ergebnis der Arbeit zu nennen. Üblicherweise können Ergebnisse nicht nur qualitativ, sondern auch quantitativ benannt werden, z. B. „...konnte eine Effizienzsteigerung von 12 % erreicht werden.“ oder „...konnte die Prüfdauer um 2 h verkürzt werden“.

Die Ergebnisse in der Zusammenfassung sollten selbstverständlich einen Bezug zu den in der Einleitung aufgeführten Fragestellungen und Zielen haben.

Abbildungsverzeichnis

| | | |
|-----|--------------------|---|
| 3.1 | Zeitplan | 6 |
|-----|--------------------|---|

Tabellenverzeichnis

| | |
|--|----|
| D.1 Anforderungsliste W-Wünsch F-Forderung | 29 |
|--|----|

A Ergänzungen

A.1 Details zu bestimmten theoretischen Grundlagen

A.2 Weitere Details, welche im Hauptteil den Lesefluss behindern

B Details zu Laboraufbauten und Messergebnissen

B.1 Versuchsanordnung

B.2 Liste der verwendeten Messgeräte

B.3 Übersicht der Messergebnisse

B.4 Schaltplan und Bild der Prototypenplatine

C Zusatzinformationen zu verwendeter Software

C.1 Struktogramm des Programmentwurfs

C.2 Wichtige Teile des Quellcodes

D Datenblätter

Auf den folgenden Seiten wird eine Möglichkeit gezeigt, wie aus einem anderen PDF-Dokument komplette Seiten übernommen werden können, z. B. zum Einbindungen von Datenblättern. Der Nachteil dieser Methode besteht darin, dass sämtliche Formateinstellungen (Kopfzeilen, Seitenzahlen, Ränder, etc.) auf diesen Seiten nicht angezeigt werden. Die Methode wird deshalb eher selten gewählt. Immerhin sorgt das Package „*pdfpages*“ für eine korrekte Seitenzahleinstellung auf den im Anschluss folgenden „nativen“ L^AT_EX-Seiten.

Eine bessere Alternative ist, einzelne Seiten mit „*\includegraphics*“ einzubinden.

Besprechungsnotizen

D.1 9.Oktober.2024

- Spieltheorie erfassen
- Speicherplatz im uController wird begrenzt sein
- Spielalgorithmus (wann wird geschaut wo Steine liegen - immer das ganze Feld abcannen?)
- Zeitplan erstellen
- <https://education.lego.com/de-de/downloads/spike-app/software/>

• Zeitplan

- KW42: Literaturrecherche, Erstellung eines groben Konzeptes (Skizzen, Funktionsweise)
- KW43: Zusammenbau des Roboters, Tests der mechanischen Komponenten (schrittweise Ansteuerung)
- KW44:

Anforderungsliste

Tabelle D.1: Anforderungsliste W-Wünsch F-Forderung
Nr **Anforderung an das System** **F/W**

| Allgemein | | |
|------------------|--|---|
| - | Lage der Steine erkennen | F |
| - | Das Ende des Speils erkennen | F |
| - | Beweglich - Steine in jede Spalte | F |
| - | Magazin für Steine | F |
| - | Immer nur ein Stein pro Spielzug | F |
| - | Abwarten bis der Gegner sein Zug beendet hat | F |
| - | Begrenzungen des Spielfeld erkennen | F |
| - | | F |
| - | | F |
| - | | F |