

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:	24. Dezember 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 24. Dezember 2024

Patrik Peters / Simon Gschell

Kurzfassung

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 verschiedenen 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, effektive Lösungen zu entwickeln.

Abstract

This student research project deals with the development of a robot for the game “Four Wins” using the LEGO Spike Prime system. The aim of the project is to design a robot that is capable of autonomously assuming the role of a human opponent in the game “Four Wins”. The robot should not only recognize the moves of the human opponent and react to them, but also independently place pieces on the playing field and ensure that the playing field is ready for the next move. This requires precise control of the robot, especially when placing the pieces, as well as the ability to monitor the playing field in order to recognize the position of the pieces already placed. The project involves various technical and organizational aspects. These include the mechanical design of the robot, including the mechanism for inserting the tiles and monitoring the playing field, as well as the development of the software that controls the moves and behavior of the robot. Another important point is the sensor technology: the robot must be able to detect the current game situation using color sensors in order to know which squares in the playing field are already occupied and where it can place its next piece. At the end of the project, a small tournament is organized in which the robot solutions developed by different teams compete against each other. This offers

Translated with DeepL.com (free version)

Inhaltsverzeichnis

1	Einleitung	1
2	Spieltheorie	2
2.1	Einleitung	2
2.2	Grundlagen	2
2.2.1	Geschichte	2
2.2.2	Regeln	2
2.3	spieltheoretische Analyse	3
2.3.1	kombinatorisches Spiel	3
2.4	Strategien und Taktiken	4
2.5	Algorithmen	4
2.5.1	Minimax-Algorithmus	4
2.5.2	Alpha-Beta-Algorithums	4
2.6	Heuristiken	4
2.7	Lösbarkeit und perfektes Spiel	4
2.8	Implementierungsaspekte für den LEGO Spike Roboter	4
2.9	Schlussfolgerung	5
3	Vorgehen	6
3.1	Vorgehensweise	6
3.2	Anforderung	7
3.3	Konzept	7
3.4	Nutzwertanalyse	7
3.5	Zeitplan	7

4	Umsetzung und Ergebnisse	8
4.1	Voraussetzungen	8
4.2	Aufbau	9
4.3	Software	9
4.3.1	Herausforderung der Programmierung	9
5	Zusammenfassung	10
	Literaturverzeichnis	11
	Abbildungsverzeichnis	13
	Tabellenverzeichnis	14
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	16
B.1	Versuchsanordnung	16
B.2	Liste der verwendeten Messgeräte	16
B.3	Übersicht der Messergebnisse	16
B.4	Schaltplan und Bild der Prototypenplatine	16
C	Zusatzinformationen zu verwendeter Software	17
C.1	Struktogramm des Programmentwurfs	17
C.2	Wichtige Teile des Quellcodes	17
D	Datenblätter	18
D.1	9.Oktober.2024	23
	Sachwortverzeichnis	25

1 Einleitung

2 Spieltheorie

2.1 Einleitung

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

2.2 Grundlagen

2.2.1 Geschichte

2.2.2 Regeln

Die Regeln für dieses Strategiespiel sind kinderleicht, was auch ein Argument für die große Beliebtheit bei Jung und Alt ist. Das Spiel besteht aus einem senkrecht stehenden Spielbrett und jeweils aus 21 gelben und roten runde Steine. Diese Steine werden abwechselnd in das hohle Spielbrett mit 42 Aussparungen, sieben Spalten und sechs Reihen, eingeworfen. Der Spieler kann somit eine Spalte auswählen und den Stein fallen lassen. Der eingeworfene Stein besetzt den untersten freien Platz dieser Spalte. Das Spiel wird dann gewonnen, wenn einer der Spieler vier oder mehr Steine in einer waagerechten, senkrechten oder diagonalen Reihe platzieren kann. Der andere

Spieler hat somit automatisch verloren. Kommt es zu keiner Bildung einer dieser Kombinationen, endet das Spiel bei Vergabe aller Steine in einem Remis.

2.3 spieltheoretische Analyse

2.3.1 kombinatorisches Spiel

Das 4 gewinnt Spiel ist ein kombinatorisches Spiel. Sie bieten einen idealen Rahmen, um Strategien, Entscheidungsfindung und Spieltheorie zu untersuchen. Auch weitere Spiele wie Schach, Dame, Mühle und Tic-Tac-Toe sind ebenfalls kombinatorische Spiele.

Diese folgende fünf Eigenschaften machen ein kombinatorisches Spiel aus:

1. **Zwei-Spieler-Struktur:** Kombinatorische Spiele sind auf zwei Spieler beschränkt, die abwechselnd Züge machen. Dies ermöglicht eine klare Analyse von Strategien und Gegenstrategien.
2. **Nullsummencharakter:** Der Gewinn eines Spielers entspricht exakt dem Verlust des anderen. Dies führt zu einer antagonistischen Spielsituation, in der die Interessen der Spieler direkt gegeneinander stehen.
3. **Endlichkeit:** Die begrenzte Anzahl von Zugmöglichkeiten und die Unmöglichkeit unendlicher Spielverläufe garantieren, dass jedes Spiel zu einem Abschluss kommt.
4. **Perfekte Information:** Beide Spieler haben zu jedem Zeitpunkt vollständige Kenntnis über den Spielstand. Dies eliminiert Unsicherheiten, die in Spielen mit verborgenen Informationen auftreten würden.
5. **Determinismus:** Der Ausschluss von Zufallselementen macht das Spiel vollständig vorhersehbar, sofern die Strategien der Spieler bekannt sind.

Durch diese genannten fünf Eigenschaften werden kombinatorische Spiele auch endliches ZweiPersonen-Nullsummenspiel mit perfekter Information" genannt.

2.3.2 Zermelos Bestimmtheitssatz

Zermelos Bestimmtheitssatz am Beispiel von „Vier gewinnt“ Der Bestimmtheitssatz von Ernst Zermelo zeigt, dass jedes kombinatorische Spiel – wie auch „Vier gewinnt“ – in eine von drei Kategorien eingeordnet werden kann:

1. **Der anziehende Spieler (Spieler 1) hat eine dominante Strategie:** Spieler 1 kann bei optimalem Spiel immer gewinnen.
2. **Der nachziehende Spieler (Spieler 2) hat eine dominante Strategie:** Spieler 2 kann bei optimalem Spiel immer gewinnen.
3. **Keiner der Spieler hat eine dominante Strategie:** Beide Spieler können bei optimalem Spiel ein Unentschieden erzwingen.

Auf „Vier gewinnt“ angewandt bedeutet dies, dass unabhängig von den individuellen Spielstilen der Spieler das Spiel theoretisch in einer dieser Kategorien liegt. Tatsächlich wurde durch Computersimulationen gezeigt, dass der anziehende Spieler (Spieler 1) bei optimaler Spielweise immer gewinnen kann. Damit gehört „Vier gewinnt“ zur ersten Kategorie.

Der Beweis des Bestimmtheitssatzes basiert auf Rückwärtsinduktion. Hierbei wird zuerst die letzte mögliche Zugfolge im Spiel analysiert:

Wenn ein Spieler im letzten Zug eine dominante Strategie hat (z. B. durch eine Reihe von vier Steinen gewinnen kann), ist der Gewinn gesichert. Man geht Zug für Zug rückwärts und prüft für jede vorherige Spielsituation, ob ein Spieler seinen Gegner zwingen kann, eine Situation herbeizuführen, in der er selbst gewinnt oder zumindest nicht verliert. Für „Vier gewinnt“ bedeutet das: Spieler 1 kann durch geschicktes Spiel

von Beginn an sicherstellen, dass er entweder direkt gewinnt oder Spieler 2 zu Zügen zwingt, die ihm langfristig keine Gewinnoptionen lassen.

Konsequenzen für „Vier gewinnt“ Aus dem Bestimmtheitssatz folgt, dass „Vier gewinnt“ nicht symmetrisch ist – der erste Spieler hat bei optimaler Spielweise immer einen Vorteil. Dies macht deutlich, dass die Siegchancen in solchen Spielen nicht gleich verteilt sein müssen.

Durch die Analyse mit Rückwärtsinduktion wurde festgestellt, dass Spieler 1 mit einer optimalen Strategie den Sieg garantieren kann. Spieler 2 kann nur dann gewinnen, wenn Spieler 1 einen Fehler macht. Somit gehört „Vier gewinnt“ zur ersten Kategorie des Bestimmtheitssatzes.

2.4 Strategien und Taktiken

2.5 Algorithmen

2.5.1 Minimax-Algorithmus

2.5.2 Alpha-Beta-Algorithums

2.6 Heuristiken

2.7 Lösbarkeit und perfektes Spiel

2.8 Implementierungsaspekte für den LEGO Spike Roboter

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.

2.9 Schlussfolgerung

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

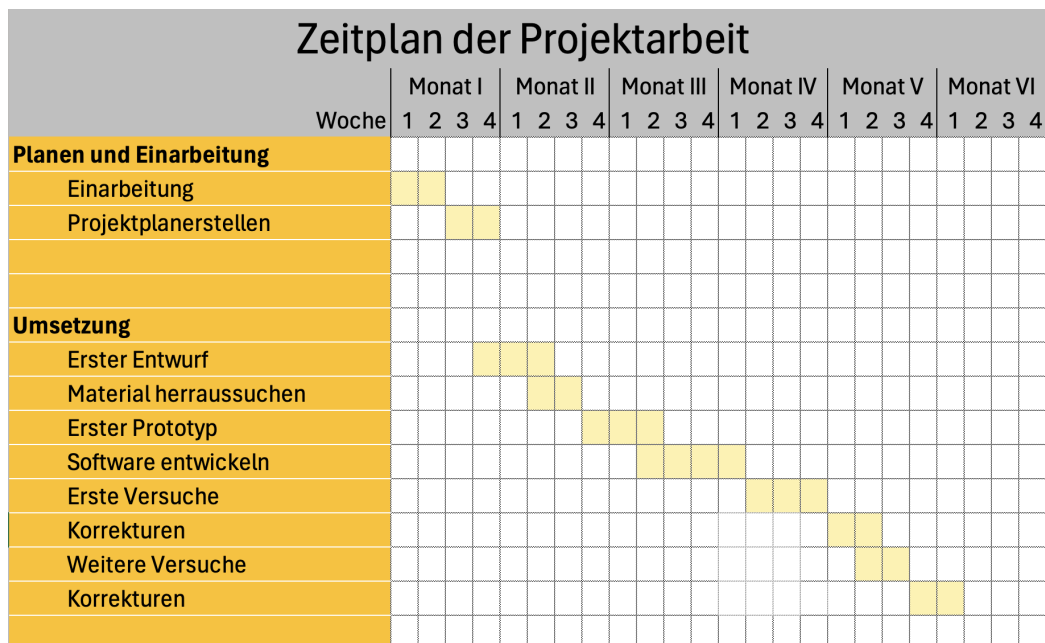


Abbildung 3.1: Zeitplan

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.

Literaturverzeichnis

- [BK97] A. Breiing und R. Knosala. *Bewerten technischer Systeme: Theoretische und methodische Grundlagen bewertungstechnischer Entscheidungshilfen*. Springer eBook Collection Computer Science and Engineering. Springer Berlin Heidelberg, 1997. ISBN: 9783642592294. DOI: 10.1007/978-3-642-59229-4.
- [Coh10] Mike Cohn. *User stories: für die agile Software-Entwicklung mit Scrum, XP u.a.* 1. Aufl. mitp, 2010. ISBN: 9783826658983.
- [Dua21] Duale Hochschule Baden-Württemberg, Fachkommission Technik. *Leitlinien für die Bearbeitung und Dokumentation der Module Praxisprojekt I bis III, Studienarbeit I / II, Bachelorarbeit*. Okt. 2021. URL: https://www.ravensburg.dhbw.de/fileadmin/user_upload/Dokumente/Dokumente_fuer_Studierende/191212_Leitlinien_Praxismodule_Studien_Bachelorarbeiten.pdf.
- [Lin22] Frank Lindenlauf. *Wissenschaftliche Arbeiten in den Ingenieur- und Naturwissenschaften: Ein praxisorientierter Leitfaden für Semester- und Abschlussarbeiten*. 1st ed. 2022. Springer Fachmedien Wiesbaden und Imprint Springer Spektrum, 2022. ISBN: 9783658367367. DOI: 10.1007/978-3-658-36736-7.
- [Mar20] Robert Martin. *Clean Agile – Die Essenz der agilen Softwareentwicklung*. 1st edition. mitp-Verlag und Safari, 2020. URL: <https://learning.oreilly.com/library/view/-/9783747501139/?ar>.

- [PR21] Klaus Pohl und Chris Rupp. *Basiswissen Requirements Engineering: Aus- und Weiterbildung nach IREB-Standard zum Certified Professional for Requirements Engineering Foundation Level*. 5., überarbeitete und aktualisierte Auflage. dpunkt Verlag, 2021. ISBN: 9783864908149.
- [Sch23] Nadine Schlüter. *Generic Systems Engineering: Ein methodischer Ansatz zur Komplexitätsbewältigung*. 3. Auflage 2023. Springer Berlin Heidelberg, 2023. ISBN: 9783662667897. DOI: 10.1007/978-3-662-66789-7.
- [Tip+19] Paul Allen Tipler u. a., Hrsg. *Physik: Für Studierende der Naturwissenschaften und Technik*. 8., korrigierte und erweiterte Auflage. Lehrbuch. Berlin: Springer Spektrum, 2019. ISBN: 9783662582800.
- [WRM22] Ralf Wirdemann, Astrid Ritscher und Johannes Mainusch. *Scrum mit User Stories*. 4., überarbeitete und erweiterte Auflage. Hanser eLibrary. Hanser, 2022. ISBN: 9783446474383. DOI: 10.3139/9783446474383. URL: <https://www.hanser-elibrary.com/doi/book/10.3139/9783446474383>.
- [Zan14] Christof Zangemeister. *Nutzwertanalyse in der Systemtechnik: Eine Methodik zur multidimensionalen Bewertung und Auswahl von Projektoptionen*. 5. Auflage 2014 (erweitert). Zangemeister & Partner, 2014. ISBN: 9783923264001.
- [Zie17] Julius Ziegler. „Optimale Trajektorienplanung für Automobile“. Dissertation. Karlsruhe: KIT Scientific Publishing und Karlsruher Institut für Technologie, 2017. URL: <http://dx.doi.org/10.5445/KSP/1000056530>.

Abbildungsverzeichnis

3.1	Zeitplan	7
-----	--------------------	---

Tabellenverzeichnis

D.1 Anforderungsliste W-Wünsch F-Forderung	24
--	----

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