

# Skorpion Puzzle - Aus Spielzeug, Spielzeug bauen

Tim Kasten, ETIT  
Otto-von-Guericke-Universität Magdeburg

**Abstract**—In einem Projektseminar für Elektrotechnik und Informationstechnik erhielten die Studierenden zwei Bauteilsets aus der LEGO®-Mindstorms® Produktreihe. Ihre Aufgabe bestand darin, einen funktionsfähigen Roboter zu entwickeln und die dazugehörige Software mithilfe von MATLAB zu programmieren. Im Zuge dieses Projekts wurde ein Roboter konstruiert, der eine innovative Art des Puzzelns ermöglichte. Dieser Roboter fokussiert sich nicht darauf, Bilder zusammenzusetzen, sondern auf das Erkennen von Melodien. Die Lieder wurden in einzelne Abschnitte zerlegt, wobei jedem Abschnitt eine bestimmte Farbe zugeordnet wurde. Der Roboter navigiert über diese farbigen Segmente, und wenn sie in der richtigen Reihenfolge angeordnet sind, spielt der Roboter das Lied korrekt ab. Auf diese Weise verfolgt der Roboter einen alternativen Ansatz im Vergleich zu herkömmlichen Puzzles, der interaktiver und spannender ist.

**Schlagwörter**—LEGO® Mindstorms®, Musik, Puzzle, Roboter, Spielzeug.

## I. EINLEITUNG

IN den letzten Jahren hat die Integration von Lernrobotern und vereinfachten Programmierumgebungen in Bildungseinrichtungen einen deutlichen Aufschwung erfahren. Durch den Einsatz von Roboterkits wie LEGO®-Mindstorms® wird es für Schülerinnen, Schüler und Studierende immer einfacher, erste Schritte in der Welt der Robotik zu unternehmen. Im Rahmen eines Projektseminars Elektrotechnik/Informationstechnik wurde ein Roboter entwickelt, der auf den LEGO®-Mindstorms®-EV3 Bausätzen basiert. Ziel war es, einen Roboter zu entwerfen, der Farben erkennen und je nach Farbe bestimmte Abschnitte bekannter Melodien abspielen kann. Dabei wurden Melodien aus verschiedenen Bereichen wie Videospielen mit „Zelda“, der Filmindustrie mit Star Wars und klassischer Musik mit „Für Elise“ ausgewählt. Die Realisierung erfolgte durch die Programmierung des EV3 mit MATLAB, dass über eine Schnittstelle der RWTH Aachen<sup>1</sup> die Steuerung verschiedener Sensoren und Motoren ermöglicht. Dieser Roboter bietet eine innovative Herangehensweise an das Puzzle-Konzept, indem er Melodieteile anstelle von Bildteilen verwendet. Auf diese Weise können die Nutzerinnen und Nutzer durch die Anordnung verschiedener Farbkarten eine vollständige Melodie erstellen. Zur Unterstützung des Farbkartenwechsels verfügt der Roboter über eine Fahrautomatik und eine Start-Stopp-Automatik, die die beiden Motoren für die Fortbewegung des Roboters steuern. Das Hauptziel dieses Projekts war es, Studierenden grundlegende Kenntnisse in den Bereichen Robotik,

Programmierung und Projektmanagement zu vermitteln. Dazu wurde den Teilnehmenden die in Zweiergruppen arbeiteten, zwei Kästen LEGO® sowie die Möglichkeit, den gesamten Prozess von der Ideenfindung über die Umsetzung bis hin zur abschließenden Präsentation eigenständig zu gestalten, gegeben.

## II. VORBETRACHTUNGEN

### A. LEGO®-Mindstorms®

LEGO®-Mindstorms® ist eine innovative Produktlinie von LEGO®, die gezielt entwickelt wurde, um das Bauen und Programmieren von Robotern zu erleichtern. Sie dient nicht nur Bildungszwecken, sondern auch der grundlegenden Entwicklung von Robotertechnologien. Die verschiedenen Kits enthalten eine Vielzahl modularer, Sensoren Motoren sowie die Typischen Legosteine, die es den Benutzern ermöglichen, ihre eigenen Roboter nach individuellen Vorstellungen zu gestalten und zu programmieren. Durch die Nutzung verschiedener Gerätenerationen mit unterschiedlichen Sensoren können Anwender die für ihre spezifischen Anforderungen am besten geeignete Technologie wählen.

In diesem Projekt wurde die EV3-Version von LEGO®-Mindstorms® eingesetzt. Die Programmierung erfolgt in der Regel über eine benutzerfreundliche grafische Programmiersprache, die von LEGO® entwickelt wurde und besonders für Anfänger\*innen zugänglich ist. Jedoch wurde im Rahmen dieses Projekts eine Schnittstelle der RWTH Aachen verwendet, um den Funktionsumfang zu erweitern. Diese Schnittstelle ermöglicht es, mithilfe von MATLAB-Code die Sensoren und Motoren des Roboters präzise anzusteuern und komplexere Projekte umzusetzen, die über die Möglichkeiten der einfachen „LEGO-Programmiersprache“ hinausgehen.

### B. Auswahl der Melodien

Aufgrund der begrenzten Leistungsfähigkeit des Lautsprechers und der Verarbeitungsgeschwindigkeit des Mindstorms-Prozessors ist es von entscheidender Bedeutung, Melodien sorgfältig auszuwählen. Diese müssen einerseits allgemein bekannt sein, um von allen potenziellen Benutzern wiedererkannt zu werden. Andererseits ist zu beachten, dass der EV3-Brick nur in der Lage ist, einen Ton zur gleichen Zeit abzuspielen. Dadurch können komplexere Tonüberlagerungen nicht wiedergegeben werden. Daher ist die Melodienauswahl auf bekannte und einfache Melodien beschränkt, die auch mit längeren

Tönen klar erkennbar sind. Als Ergebnis wurde sich für Melodien aus Star Wars, Zelda und für Das Lied „Für Elise“ entschieden.

Es ist jedoch wichtig zu anzumerken, dass diese Songauswahl nachträglich erweitert werden kann.

### C. Wiedergabe von Melodien

Um Tonausgaben auf dem EV3 zu ermöglichen, muss eine spezifische Anweisung an den Brick gesendet werden. Diese Anweisung enthält Informationen über die Frequenz eines bestimmten Tons, seine Dauer sowie die gewünschte Lautstärke. Um ganze Melodien abzuspielen, muss die Melodie zuerst in Frequenzen und Tonlängen zerlegt werden, damit sie anschließend schrittweise an den Brick gesendet werden kann. Hierfür können Umwandlungstabellen verwendet werden, die entsprechende Frequenzen für die verschiedenen Töne bereitstellen.

### D. Probleme mit den Sensoren

Die Sensoren, die für diesen Roboter erforderlich sind, bestehen aus zwei Tastsensoren und zwei Farbsensoren. Die Tastsensoren haben die Funktion, entweder anzuzeigen, ob sie im aktuellen Moment gedrückt sind, oder die Anzahl der bisherigen Betätigungen zu zählen. Letzteres wurde im Programm verwendet, um mit einem der Taster zwischen den Melodien zu wechseln und mit Hilfe des anderen Tasters die Wiedergabe sowie den Motor zu stoppen.

Bei der Umsetzung der Projektidee mussten insbesondere die Ungenauigkeiten der Farbsensoren berücksichtigt werden, da sie häufig unzuverlässige Ergebnisse lieferten. Insbesondere bei schlechten Lichtverhältnissen wurden Farben oft falsch erkannt. Um diesem Problem vorzubeugen, wurden zwei Sensoren eingebaut, die im Programm miteinander abgeglichen wurden.

Außerdem war ursprünglich geplant, einen Gyrosensor zu verwenden, um die Lenkung des Roboters zu überwachen. Dieser Sensor sollte einerseits den Drehwinkel und andererseits die Drehgeschwindigkeit messen. Allerdings wurde diese Idee aufgegeben, da der Gyrosensor fehlerhafte Werte lieferte und insgesamt ungenau war.

## III. DER CODE

Das Programm basiert auf einem sich nach jeder Iteration wiederholenden Schema, das im folgenden Ablaufdiagramm (Abb. 1) veranschaulicht wird. Zunächst muss der Benutzer mithilfe eines Tasters eine der drei Melodien (Star Wars, Für Elise, Zelda) auswählen. Da der Tastsensor auf zwei Modi beschränkt ist - einem Modus, in dem er nur während des Drückens ein Signal ausgibt, und einem Modus, in dem die absolute Anzahl der Betätigungen gezählt wird - muss zunächst in den absoluten Zählmodus gewechselt werden. Anschließend ermöglicht das Programm durch die Berechnung des Modullos  $n = \text{mod}(\text{tastsensorcount}, 3)$  einen Wechsel zwischen drei Modi, die jeweils einer Melodie entsprechen:

$$n = \begin{cases} 0, & \text{Starwars} \\ 1, & \text{Zelda} \\ 2, & \text{Für Elise} \end{cases}$$

Mithilfe von if-Statements wird jeder dieser Modi einer Melodie zugeordnet, die im weiteren Verlauf des Programms verwendet wird. Anschließend werden die beiden verwendeten Farbsensoren ausgelesen und dreimal abgeglichen. Falls die Werte der Sensoren bei den drei Abgleichen nicht übereinstimmen, wird der Wert eines der beiden Sensoren genutzt. Basierend darauf wird ein Teil der Melodie abgespielt, der durch Switch-Statements an die Farbausgabe der Farbsensoren gekoppelt ist. Eine typische Ausgabe wäre "blaue Farbe", die dann beispielsweise "case blue" und den entsprechenden Melodieausschnitt abspielt.

Solange die Melodie abgespielt wird, stoppt der Roboter, und erst wenn eine Melodie beendet ist, wird die Fahrt fortgesetzt. Sobald der Farbsensor eine neue Farbe erkennt, wird der dazu korrespondierende Melodieabschnitt abgespielt, und der Roboter stoppt erneut. Dieser Vorgang wiederholt sich, bis ein zweiter Taster, der nach dem gleichen Funktionsprinzip wie der Taster zum Wechseln der Melodie funktioniert, ein Stoppsignal auslöst und die Melodie beendet wird, während der Motor in den Bremsmodus wechselt.

Um das Programm nachträglich erweiterbar zu gestalten, wurden die einzelnen Melodien in separate Funktionen integriert. Dadurch ist es möglich, zusätzliche Melodien hinzuzufügen, solange sie das erforderliche Format für die einzelnen Töne einhalten:

"tone(Frequenz (in Hz), Lautstärke (in %), Länge (in ms))"

Um zu erkennen, ob der Roboter weiterfahren kann oder die Melodie noch zu Ende gespielt werden muss, wurde außerdem eine Funktion implementiert, die überprüft, ob er sich noch auf derselben Farbe befindet oder bereits auf einer neuen Farbe steht und deshalb stoppt, bis die Melodie vollständig abgespielt wurde.

## IV. KONSTRUKTION

Neben der Implementierung der Software war es von entscheidender Bedeutung, ein robustes Gerüst zu konstruieren, das sowohl mehrere Sensoren als auch zwei Motoren sicher halten konnte. In der ersten Version des Roboters wurde auf Ketten als Antrieb gesetzt sodass die Lenkung mittels einer klassischen „Panzersteuerrung“ realisiert werden konnte (Abb. 2), jedoch obwohl die Legoteile hervorragend für ihren Zweck als Spielzeug sind, sind sie ungeeignet für mechanisch präzise Aufgaben. Dies führte dazu, dass der Roboter nicht geradeaus fahren konnte und zufällig nach rechts oder links abwich. Aus diesem Grund wurden die Ketten durch vier Räder ersetzt, was es dem Roboter ermöglichte, präziser zu fahren und vorerst auf eine Lenkung verzichtet (Abb. 3).

Neben den unerlässlichen Komponenten war auch das Design ein zentraler Aspekt der Entwicklung. Ein ikonisches

Design steigert den Wiedererkennungswert erheblich und ist somit entscheidend für ein marktfähiges Produkt. Außerdem bieten die Optikelemente des Puzzles zusätzliche Stabilität und Griffigkeit trotz dessen das diese modular und leicht demontierbar sind. Letzteres war besonders hilfreich bei der Fehlerbehebung und Weiterentwicklung.

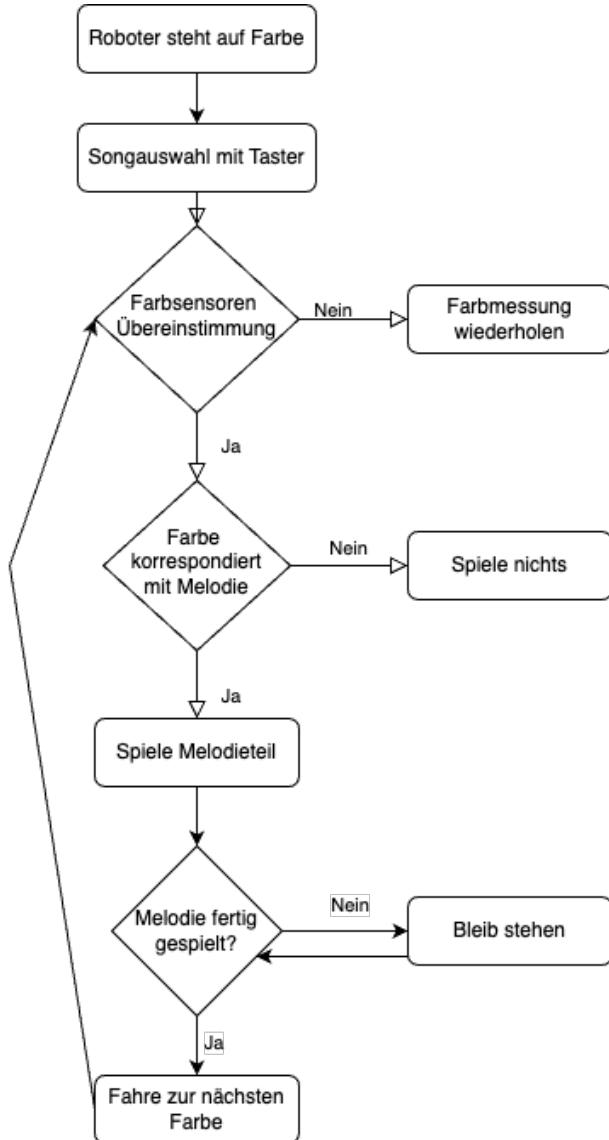


Abb. 1 Ablaufplan

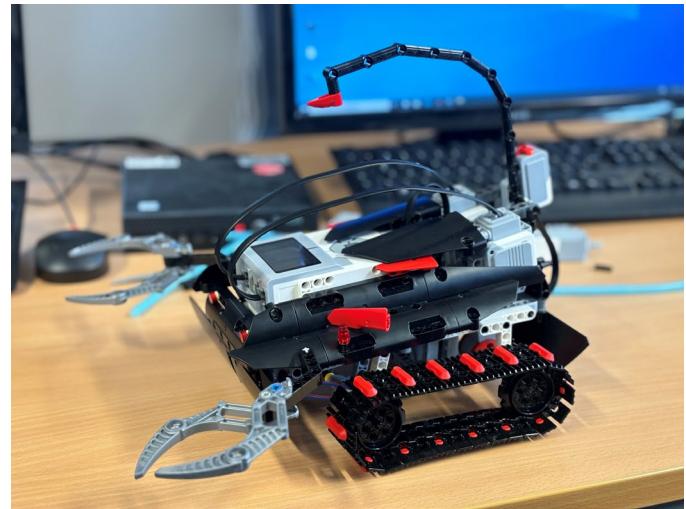


Abb. 2 Mark 2

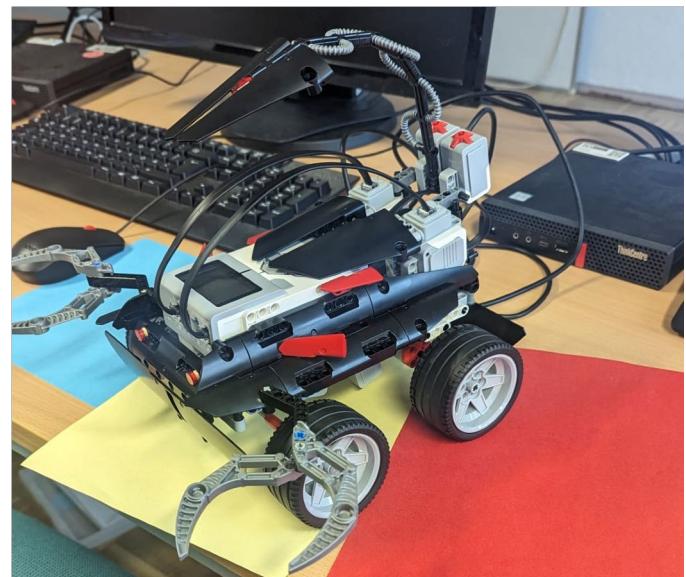


Abb. 3 Mark 6 / Finale Version

## V. PERSPEKTIVISCHE ERWEITERUNGEN

Aktuell sind drei Melodien integriert, jedoch kann diese Anzahl unbegrenzt erweitert werden, sofern für jede weitere Melodie Funktionen implementiert werden, die dem gleichen Schema folgen wie die bereits integrierten Melodien. Der begrenzende Faktor liegt dabei in der Steuerung zwischen den Melodien. Aufgrund der eingeschränkten Nutzbarkeit der Taster, wie bereits in den Vorüberlegungen erläutert, ist es nur komfortabel, eine geringe Anzahl an Melodien zu nutzen. Bei jedem Wechsel muss man durch verschiedene Melodien navigieren, ohne eine spezifische Melodie direkt auswählen zu können.

Eine weitere Möglichkeit zur Erweiterung besteht darin, einen externen Lautsprecher anzubringen. Dieser bietet nicht nur eine bessere Tonqualität als der im Steuerungsmodul vorhandene, sondern kann auch komplexere Tonabfolgen mit geringerer Latenz abspielen. Dadurch würde sich ein größeres Repertoire an Melodien eröffnen.

Für die weitere Implementierung von Melodien sollte außerdem eine Zusatzfunktion im Programm vorhanden sein, die es ermöglicht, aus fertigen Songs konvertierte und abspielbare Melodien zu erstellen. Obwohl dies eine aufwendige Erweiterung wäre, ist sie unumgänglich, um eine breitere Auswahl an Melodien zu bieten.

Durch eine Umgestaltung der Steuerung wäre es zukünftig auch möglich, weitere Fahrmodi einzuführen. Diese könnten beispielsweise eine Änderung der Fahrtrichtung in Korrespondenz mit der ausgewählten Farbe ermöglichen.

## VI. FAZIT

LEGO®-Mindstorms® bietet eine Vielzahl kompatibler Klemmbausteine, hervorragend für den Einstieg in die Welt der Roboterentwicklung. Dank der intuitiven Anwendung, ist der Entwicklung eigener Projekte, nur durch die begrenzte Flexibilität der Klemmbausteine im Einzelnen Grenzen gesetzt. Die größten Herausforderungen ergeben sich jedoch aus den Sensoren, die sowohl in ihrer Genauigkeit als auch in ihren Ausgabe-modi begrenzt sind. Dadurch sind komplexere Projektideen zwar eingeschränkt, aber durch die Integration zusätzlicher Mikrocontroller, Sensoren oder ähnlicher Komponenten durchaus realisierbar.

Die Einbindung von MATLAB funktioniert grundsätzlich gut, jedoch treten häufig Konnektivitätsprobleme auf, was zu einer erheblichen Zeitinvestition in die Fehlerbehebung während der Entwicklungsphase führt. Die größten Herausforderungen bei der Entwicklung dieses Roboters waren einerseits die Gewährleistung einer zuverlässigen Steuerung und andererseits der Umgang mit den Sensoren.

Vom Ansatz her war das Projekt ideal, um einen Einstieg in die Programmierung zu bieten, da eine Vielzahl grundlegender Programmierkonzepte implementiert werden musste. Zusätzlich forderte das Projekt gewisse Kenntnisse in Mechanik und Konstruktion um die gewünschten Roboter zu bauen, was über dem Rand des Projekts hinaus durchaus interessant für die Zukunft ist.

## VII. LITERATURVERZEICHNIS

- [1] Lehrstuhl für Bildverarbeitung RTWH Aachen, [Online]. Available: <https://www.lfb.rwth-aachen.de/de/academics/lectures/mindstorms/> [Zugriff am 08.03.2024].