

# Ballschussmaschine

Kjell Diegeler, Elektro- und Informationstechnik  
Otto-von-Guericke-Universität Magdeburg

**Zusammenfassung**—Der in diesem Seminar entwickelte Roboter ist ein fast vollautomatisierter fahrender Ballschussroboter. Die Umsetzung der Konstruktions- und Softwarekonzepte werden im Folgenden erläutert, wie auch die entstandenen Herausforderungen und deren Lösungen. Zum Schluss werden die Verwendungszwecke und mögliche Optimierungen diskutiert.

**Schlagwörter**—Automatisierung, Ballschuss, LEGO-Mindstorms, Programmierung, Roboter, Sensorik

## I. EINLEITUNG

WÄHREND des Praktikums war es die Aufgabe, einen Roboter zu entwickeln, neben der Hauptaufgabe die Grundlagen in MATLAB zu erlernen und die Sensoren und Motoren von LEGO kennenzulernen. Für die Entwicklung des Roboters ist eine gute räumliche Vorstellung von großem Vorteil.

Stabilität und Balance spielen eine große Rolle beim Entwickeln des Roboters. Bei fehlender Balance besteht die große Gefahr des Umkippens und starke Einschränkungen in der Funktion. Fehlende Stabilität ist allerdings das größere Problem, sollte dies nicht gegeben sein, können sich relevante Bauteile lösen und die Funktion des Roboters nicht gewährleisten. Die Ansteuerung der Sensoren und Motoren kann auch eine kleine Herausforderung darstellen. Die Verwendung eines LEGO-fremden Bauteils, wie zum Beispiel einer Webcam, kann schnell zu einer zeitaufwendigen Herausforderung werden.

## II. VORBETRACHTUNGEN

In diesem Abschnitt wird die Konzeptidee des Roboters erläutert, um auf Hinblick der Umsetzung Verständnis zu schaffen. Die Funktionen und benötigten Komponenten werden hierbei genauer vorgestellt.

### A. Was soll die Ballschussmaschine können?

Das Konzept des Ballschussroboters ist es, nachdem er gestartet wurde, soll der Roboter warten, bis der Ballschussapparat geladen ist. Daraufhin soll er sich fortbewegen und bei einem vordefinierten Abstand vor dem Ziel halten. Als Nächstes sucht die Maschine den Körper, der als Ziel dient und wenn dieser nicht mittig zum Roboter ausgerichtet ist, richtet sich der Roboter selbständig aus. Ist dies geschehen, erfolgt der Schuss und das Ziel wird getroffen.

### B. Die benötigten Komponenten für die Ballschussmaschine

Zur Umsetzung wurde ein LEGO-Mindstorms-Set und eine Webcam zur Verfügung gestellt. Das Set beinhaltet eine verschiedene Anzahl an Motoren, Sensoren und LEGO-Klemmbausteinen, die für die Umsetzung essentiell wichtig sind. Die benötigten Komponenten werden in Abbildung 1 dargestellt. Als Steuereinheit dient der NXT-Stein, welcher sämtliche Motoren und Sensoren steuert und wichtige Daten entnimmt. Der NXT-Baustein ist zu dem auch die Schnittstelle zum PC bzw. Laptop, wo das Programm abläuft. Die Motoren dienen zur Fortbewegung und zur automatischen Ausrichtung des Roboters, als auch zur Steuerung des Sicherungsstiftes des Ballschussapparates. Der Taster wurde dazu eingesetzt, um dem Motor, der den Sicherungsstift steuert, zu signalisieren, wann der Stift ausgefahren werden soll. Der Ultraschallsensor wurde zur Distanzmessung verwendet. Zum Schluss wurde eine Webcam zur Farberkennung verwendet und somit zur Zielsuche.



Abbildung 1. Verwendete Komponenten [1]–[5]

## III. UMSETZUNG

Im Folgenden wird die Umsetzung Schritt für Schritt erläutert. Zuerst wird der Programmablaufplan dargestellt. Danach wird Aufbau und Funktion der Konstruktion erklärt. Zum Schluss wird noch die Funktionsweise der Webcam erläutert.

### A. Programmablaufplan

In der Abbildung 2 wird für die Ballschussmaschine der Programmablaufplan dargestellt.

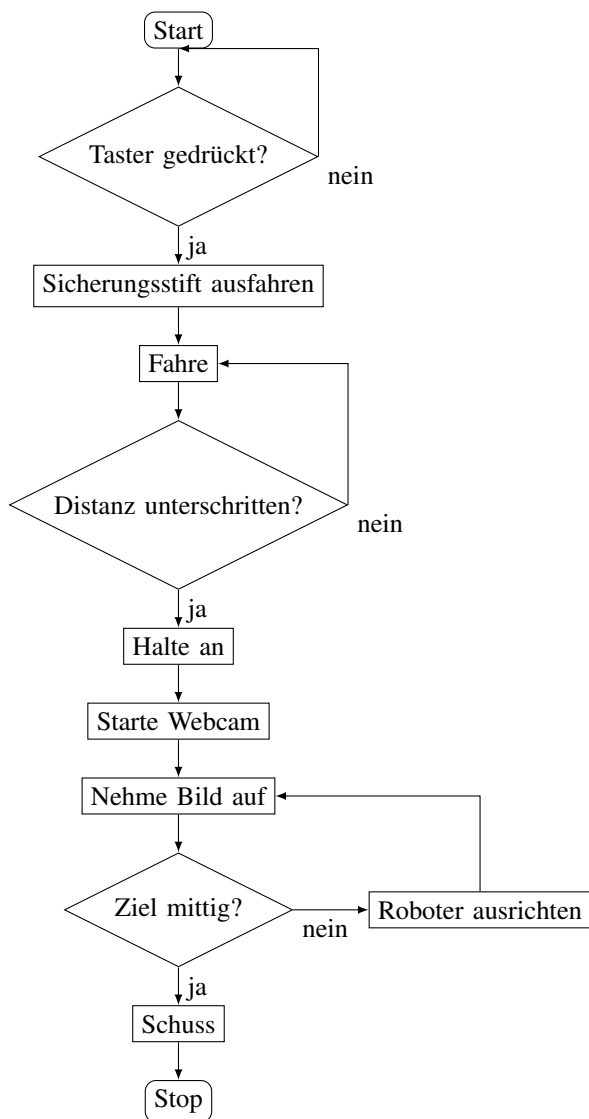


Abbildung 2. Beispielhafter Programmablaufplan zur Erklärung eines Verfolgungsalgorithmus

### B. Aufbau und Funktion der Ballschussmaschine

Der Roboter in Abbildung 3 besteht aus einem Fahrwerk, Ballschussapparat, einem Sicherungsstift und einer Webcam. Das Fahrwerk hat die Funktion, den Roboter zu fahren und später den Roboter auszurichten. Hierzu werden zwei von den drei verwendeten Motoren genutzt. Der Ballschussapparat besteht aus einer Schiene, einem Gummi und einer Nachladevorrichtung. Der Sicherungsstift, abgebildet in Abbildung 4, dient hierzu, um die Nachladevorrichtung an seinem Ort zu halten. Dieser wird gesteuert durch einen Tastsensor und einen weiteren Motor. Durch Betätigen des Schalters wird der Motor angesteuert und dieser bewegt sich dann um 180°. Ist der Sicherungsstift ausgefahren, macht das Programm eine Pause und gibt dann den Motoren vom Fahrwerk den Befehl zu starten. Der Ultraschallsensor sorgt ab dann, dass die Ballschussmaschine bei einer vordefinierten Distanz anhält. Dann wird die Webcam gestartet. Die ist dazu da, um ein Bild aufzunehmen und dabei die Farbe Rot herauszufiltern und

als Bild auszugeben. Mit den erhaltenen Daten wird nun die Position des Mittelpunktes des Zieles ermittelt und mit dem Mittelpunkt des Bildes abgeglichen. Sollte es zu einer Abweichung kommen, so wird über die Motoren des Fahrwerkes die Position in der jeweiligen Richtung abgeändert und erneut die Bedingungen gecheckt. Wenn die Bedingung erfüllt ist, wird jetzt der Motor des Sicherungsstiftes angesteuert. Dieser dreht sich um 180° und sorgt dafür, dass die Nachladevorrichtung durch das Gummi nach vorne schnallt und somit die Kugel abfeuert. Damit wäre das Programm beendet.

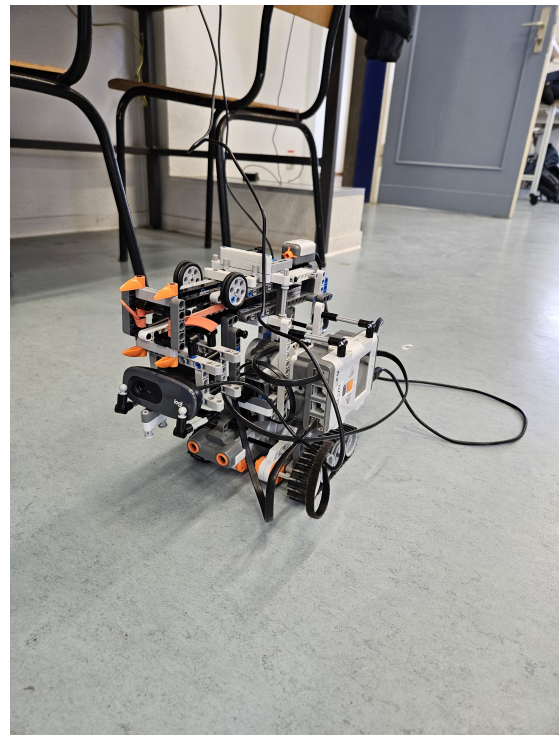


Abbildung 3. fertige Ballschussmaschine

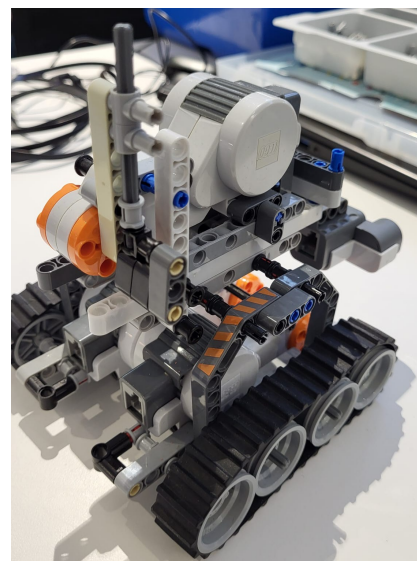


Abbildung 4. Sicherungsstift

### C. Benutzung einer Webcam

Die Benutzung einer Webcam stellte sich als kleine Hürde heraus. Der Quellcode wird unten dargestellt. Im ersten Schritt wird die Webcam als Objekt angelegt. Mit dem Befehl `snapshot` wird eine Einmalaufnahme der Webcam gemacht und die Farbwerte in einer  $480 \times 640 \times 3$  großen Matrix gespeichert. Mit diesen Werten kann man sich auch das Bild ausgeben lassen, mit Skalierung. Hierfür wird der Befehl `imagesc()` verwendet. Als nächsten relevanten Schritt wird eine 0-Maske über die Farbwerte erstellt und mithilfe eines Filters überall, wo die Farbwerte mit der Filterbedingung überschrieben, mit der Zahl 1. Dies kann man sich ebenfalls als Bild ausgeben lassen. Als Nächstes wird über den Suchbefehl `find()` die Koordinaten der ersten und letzten 1 ermittelt. Somit lässt sich dann über die Gleichung 1 die Koordinaten des Mittelpunktes des Zielobjektes ermitteln.

```
cam = webcam;
preview(cam)
img = snapshot(cam);
imagesc(img)
mask = zeros(size(img,1,2));
mask(img(:,:,1)>75 & img(:,:,2)<50 & img(:,:,3)<50) = 1;
figure(2)
imagesc(mask)
[r_a,c_a] = find(mask==1,1,'first');
[r_e,c_e] = find(mask==1,1,'last');
m = c_a + (c_e - c_a)/2
```

$$m = c_a + (c_e - c_a)/2 \quad (1)$$

### IV. DISKUSSION

Natürlich sind während der Umsetzung auch Probleme aufgetreten. Zum einen war die größte Schwierigkeit, die Konstruktion stabil zu gestalten, da das verwendete Gummi sehr viel Spannung erzeugte. So kam es des Öfteren vor, dass bei den ersten Testversuchen die Konstruktion mit zerschoss und man gezwungen war, die Maschine immer stabiler zu gestalten. Dies führte allerdings auch dazu, dass die ganze Konstruktion sehr einseitig belastet war, sodass die Gefahr des Umkippens sehr hoch war.

Dies wurde gelöst, indem man den NXT-Stein seitlich befestigt hat. Desweiteren kam die Frage auf, wie schnell man die Motoren ansprechen kann, ohne Fehler zu generieren, dies konnte nur durch Testen gelöst werden. Die Fixierung des Sicherungstiftes beim Einfahren gestaltete sich anfangs auch als schwierig, konnte aber durch bauliche Änderungen gesichert werden. Durch die Bauweise sieht die Konstruktion sehr militärisch aus, dies ist natürlich nicht beabsichtigt und kommt zustande, da es sich um ein Kettenfahrzeug handelt. Dies war leider die einfachste Lösung, um den Roboter fahren zu lassen und auch zu drehen.

### V. ZUSAMMENFASSUNG UND FAZIT

Am Ende konnte der Roboter, nachdem er geladen war, vollkommen autonom fahren, sein Ziel suchen und das Ziel mithilfe einer Kugel abschießen. Die Ballschussmaschine ist noch optimierbar. Zum einen könnte man noch die Konstruktion so abändern, dass die Maschine auch sich automatisch nach oben bzw. nach unten ausrichtet. Im Großen und Ganzen wurde das Projektseminar erfolgreich abgeschlossen und die wichtigen Grundlagen der Programmierung mit Matlab wurden mitgenommen.

### LITERATURVERZEICHNIS

- [1] BRICKIPEDIA: *NXT-Servomotor*. [https://brickipedia.fandom.com/wiki/9842\\_NXT\\_Servo\\_Motor](https://brickipedia.fandom.com/wiki/9842_NXT_Servo_Motor). Version: Februar 2024
- [2] BRICK-SHOP: *NXT-Stein*. <https://www.brick-shop.de/Elektrik-4951.html?language=de>. Version: Februar 2024
- [3] AMAZON: *Ultraschallsensor*. <https://www.amazon.de/LEGO-NXT-Ultraschallsensor-Ultrasonic-sensor/dp/B000PM8I8O>. Version: Februar 2024
- [4] LEGO: *Tastensor*. <https://www.lego.com/cdn/cs/set/assets/bltee56bc5628e9b8bf/9843.jpg>. Version: Februar 2024
- [5] AMAZON: *Logitech C270 Webcam*. <https://www.amazon.de/Logitech-C270-Webcam-720p-schwarz/dp/B01BGBJ8Y0>. Version: Februar 2024