

System Design Project

Zwischenbericht WS 2015/2016

Der fühlende Psi Bot #14

13. Januar 2016

1 Gruppenmitglieder

- Rob Falkenstein (3530252) rob.falkenstein@freenet.de
- Yannik Rieder (3723679) yannik.rieder@gmx.de
- Fritz Gebhardt (4137115) fg@klige.de
- Moritz Schiltewolf (3803354) moritz.schiwo@gmail.com

2 Roboterkonzept

Bis zum Meilenstein bestand unser Grundkonzept darin, uns an die LEGO-Anleitung zu halten, um ein Grundgerüst zum Testen zu haben. An der Bahn merkten wir aber schnell, dass dies nicht genug sein würde. Deshalb wechselten wir die Räder aus, um eine höhere Stabilität gegenüber der Bodenwellen zu erreichen. Dies stellte sich als gute Idee heraus, da er nun problemlos über die Bodenwellen fahren konnte. Außerdem zeigte der Roboter keine Schwierigkeiten an kleinen Steigungen. Da er die steilen Passagen aber weder aufwärts noch abwärts meistern konnte, haben wir zum Ziel, den Schwerpunkt des Robos weiter nach unten zu verlagern, sowie das Stützrad auszutauschen, um eine höhere Stabilität zu erzielen. Um den Schwerpunkt richtig zu lagern, werden wir versuchen, den NXC Baustein horizontal in den Robo einzubauen. Dabei gibt es zwei Dinge zu beachten: Der Robo sollte auf der einen Seite möglichst tief und stabil liegen, auf der anderen Seite aber nicht auf den Bodenwellen oder an den Steigungen am Boden aufliegen. In Bezug auf die Fortbewegung werden wir uns weiterhin an recht großen Rädern orientieren, da diese ein hohes Maß an Widerstandsfähigkeit gegenüber der Bodenwellen angezeigt haben. Des Weiteren ist uns bei den ersten Fahrten mit dem Roboter aufgefallen, dass es ungünstig ist, die Lichtsensoren, wie ursprünglich geplant, eher mittig anzubringen, so dass

sie bei einem Wechsel von Schwarz zu Weiß reagierten. Bei dieser Bauweise hatte der Roboter in den Kurven teilweise Probleme. Um das Problem zu beheben haben wir die Sensoren nach außen verschoben und den sich auf diese Sensoren Code quasi invertiert, so dass nun beim Wechsel von Weiß zu Schwarz reagiert wird. Wir haben uns bisher nur konzeptuell Gedanken darüber gemacht, wie der Roboter den Ball aufnehmen, tragen und wieder ablegen soll, da wir es für sinnvoll erachtet haben, zunächst die grundlegenden Bewegungsmechanismen zu bauen und zu programmieren.

3 Softwarekonzept

Für das Softwarekonzept haben wir uns an den Tipp von Professor Leo Reindl gehalten. Die zentralen Parameter wurden an den Anfang des NXC-Codes ausgelagert und der Code in deren Folge entwickelt. Grundlegende Parameter sind bisher die Schwellenwerte für die Farberkennung sowie zentrale Geschwindigkeitsindizes. Unser momentaner Stand sind mehrere Codes, einmal zum Umdrehen, dann einen zum Greifen, einen zum Ablegen und ein letzter zum Befahren der schwarzen Linie. Diese sollen am Ende alle in einen Code zusammengeführt werden. Wir hoffen, dass dies uns keine Probleme bereiten wird; die Einzelerstellung der notwendigen Code-Teile hat für uns bisher aber einige Vorteile gehabt. Daneben versuchen wir, den Code möglichst ausführlich zu kommentieren, damit die anderen Teammitglieder keine Probleme haben, gemachte und in GitHub hochgeladenen Veränderungen nachvollziehen zu können.

4 Fortschritt

Am Ende des ersten Meilenstein-Treffen fährt der Roboter die schwarze Linie zuverlässig ab. Anfangs wurden zwei Lichtsensoren verwendet, die zentral und nah beieinander lagen, damit der Roboter bei Abweichen von der schwarzen Linie diesen Weg korrigieren konnte. Da dies bei den Bodenwellen aber starke Probleme verursachte, entschieden wir uns dazu, die beiden Lichtsensoren weiter auseinander zu platzieren und den Code zu ändern. Dadurch hatte der Roboter keine Probleme mehr, der schwarzen Linie zu folgen, da nur im Falle des Abweichens vom weißen Randmuster eine Korrektur vorgenommen wurde.

5 Fehleranalyse

Anfangs wurde der NXT-Computer über die NXT-Software von Lego programmiert. Theoretisch hätte der Line-Follow-Code funktionieren sollen, so dass der Roboter der schwarzen Linie folgt. Aus unerklärlichen Gründen tat er dies trotz verschiedener Programmierungscodes nicht. Deshalb haben

wir die Software durch Not-Exactly-C ausgetauscht. Mit dieser funktioniert das Verfolgen der Linie beim ersten Versuch. Beim ersten Versuch auf der Test-Strecke zeigen sich einige Probleme, da der Roboter bisher nur zur Verfolgung der schwarzen Linie programmiert wurde. Bei der Abfahrt verlieren die Lichtsensoren den Kontakt zur schwarzen Linie und der Roboter kippt nach vorne über. Bei der Auffahrt erreicht der Roboter nur die Hälfte und die Reifen drehen durch. Diesen Problemen wollen wir zuallererst durch den neuen Schwerpunkt des Roboters beikommen. Wir hoffen, dass dies genügt, damit dieser die Bahn zufriedenstellend abfahren kann. Sollte dies nicht der Fall sein, werden wir angewiesen sein, Räder mit einer höheren Haftreibung am Robo anzubringen. Dies wird sich in den kommenden Sitzungen herausstellen.

6 Weiteres Vorgehen

Die Programmiercodes für das Fahren und das Drehen funktionieren bisher ziemlich gut. Wenn wir den Roboter umbauen, müssen wir uns an der Höhe des Balles orientieren, um den Greifarm richtig einzusetzen. Dieser wird an den dritten Motor angeschlossen werden. Wie das codetechnisch funktionieren soll, haben wir noch nicht ganz geklärt. Es ist jedoch klar, dass wir den Berührungssensor benutzen müssen, um dem Greifarm anzuzeigen, wann dieser schließen soll.

7 Arbeitsteilung

Alle Mitglieder haben das Projekt zusammen bearbeitet. Beim bisherigen Zusammenbau lieferte jedes der Mitglieder seinen Teil. Der Code wurde bisher entweder über GitHub per SVN geteilt und bearbeitet oder von den an der Testbahn anwesenden Testern geändert.