



Projektarbeit ~ Team BOTlike

26.08.2020 - 15.06.2021

Tobias Schmitz, Max Hennes, Ida Dreisbach
Projektkurs Informatik

Einführung

Für den diesjährigen Projektkurs im Fach Informatik haben wir einen wettkampffähigen Soccer-Roboter in der Kategorie 1-gegen-1 gebaut und programmiert. Dazu haben wir uns an die [Vorgaben des RoboCups](#) gehalten, da unser Ziel eine Teilnahme dort war. Leider war dies coronabedingt nicht möglich. Allerdings organisieren wir innerhalb des Projektkurses einen Wettbewerb, wie wir dort abschneiden wissen wir zum jetzigen Zeitpunkt noch nicht. In den folgenden Ausführungen stellen wir unseren Roboter vor.

Roboter (Aufbau)

Hardware

[Mainboard](#)

→ beinhaltet ESP32 (Prozessor), Kompass, Motortreiber, CANBus

[Adapterplatte \(modelliert mit OPENScad\)](#)

→ beinhaltet den Stromschalter

[IR Ring](#)

[Adapterplatte1 \(Platine\)](#)

[Adapterplatte2 \(Platine\)](#)

[Bodenplatte](#)

→ beinhaltet Motoren, Räder, Akku, Ballschale

Software

Hauptprogramme:

[Main 1](#) bis [Main 10](#)

Programm für den Infrarotring:

[IR-Ring](#)

Idee

Bei der Entwicklung des Roboters haben wir uns an denen der letzten Jahre orientiert. Wir haben uns für einen zylinderartigen Aufbau mit drei Rädern entschieden. Drei Räder hielten wir für sinnvoll, da auf einem unebenen Feld alle der drei Räder einen Berührungspunkt haben, wohingegen bei vier Rädern eines in der Luft hängen und damit ein Störfaktor sein könnte. Die schwereren Teile, insbesondere die Akkus, haben wir im unteren Teil des Roboters platziert, um für größere Stabilität zu sorgen, da der Roboter bei einem niedriger platziertem Massenmittelpunkt weniger leicht umgestoßen werden kann. Wir haben uns gegen eine Kamera entschieden, da diese deutlich zeitaufwändiger ist als eine Ortung durch Infrarotsensoren. Zur Orientierung haben wir uns für den Einbau eines Kompasses entschieden, da uns diese Idee, insbesondere für die Umsetzung in der Software, einfach erschien.

Umsetzung

Hardware

Um an erste Teile zu kommen haben wir Teile eines alten Roboters recycelt: Dazu gehören die Bodenplatte, die Vorrichtungen für die Räder, die Motoren und die LED-Platine. In die Bodenplatte haben wir Löcher zur Befestigung der Akkus gebohrt. Dann haben wir die Motoren und Radbefestigungen, die wir vorher an für die Räder vergrößert hatten, an der Bodenplatte befestigt. Die Räder, die wir verwendet haben, sind eine Maßanfertigung für die Roboter-AG, die uns Herr Stiebel ausgehändigt hat. Durch ihren Aufbau, mit weiteren orthogonalen kleinen Rädern, ist der Roboter zu sauberen Bewegungen in alle Richtungen fähig. Darauf haben wir eine ehemalige LED-Platine montiert, die wir zu einer Adapterplatte umfunktionierten. Sie schließt den unteren Teil des Roboters ab und dient als Aufbaumöglichkeit für den oberen Teil des Roboters. Um die Infrarot-Platine mit der Adapterplatte zu verbinden, mussten wir einen weitere Adapterplatte dazwischen einbauen, um das Problem der verschiedenen Schraubenlayouts zu lösen. Die Infrarot-Platine dient dazu den Ball, der Infrarotwellen abstrahlt, zu orten. Die Infrarot-Platine haben wir mit einer von uns passend modellierten Adapterplatte mit dem Mainboard verbunden, um erneut Schraubenlayouts auszugleichen. Die [Adapterplatte](#) haben wir in OpenSCAD modelliert und mit Hilfe eines 3D-Druckers gedruckt. Auf dem Mainboard befinden sich ein ESP32, die Motortreiber, der Kompass, vier Knopfschalter, der CAN-Bus und der Akkuanschluss. Der CAN-Bus und die Infrarot-Das Kabel zu den Akkus haben wir durchtrennt und einen Schalter eingebaut, um den Stromfluss jederzeit kontrollieren zu können.

Software

Die Entwicklung unseres Mainprogramms ist in der Veränderung in den verschiedenen Versionen (Main 1 - Main 10) erkennbar. Wir haben uns zunächst Gedanken darüber gemacht in welcher Reihenfolge der Roboter, was machen muss. Unsere Idee war dabei, dass der Roboter den Ball zunächst erkennen muss und dann in dessen Richtung fährt. Wenn er den Ball dann hat, soll er in Richtung Tor fahren.

Um zu wissen wohin der Roboter gerade gedreht ist, muss der Kompass zunächst ausgerichtet werden. Dies macht der Roboter nur einmal im Setup Code. Da er zu Beginn des Spiels immer in Richtung Tor gedreht startet, liegt das Tor also in Richtung 0. Um den Ball zu finden muss der ESP32 mit dem CANBus kommunizieren, da der CANBus dem IR-Ring Anfragen schickt. Dafür haben wir das IR-Ring Programm der Bohlebots benutzt. Der CANBus nennt uns daraufhin Richtungen von 0-255 (1 Byte). Diese haben wir folgendermaßen umgerechnet:

```

if (irpaket<15)  { ball=-8; }           // hier sind die Werte ungültig
else
{
  irpaket = irpaket-16;
  irpaket=irpaket/16;                  // auf sechzehn Richtungen aufteilen
  switch (irpaket)                    // auf die 6 Richtungen von bohlebots.h aufteilen
  {
    case 0: ball = -2; break;
    case 1: ball = -2; break;
    case 2: ball = -2; break;
    case 3: ball = -1; break;
    case 4: ball = -1; break;
    case 5: ball = -1; break;
    case 6: ball = 0; break;           // vorne genauer als hinten
    case 7: ball = 1; break;
    case 8: ball = 1; break;
    case 9: ball = 1; break;
    case 10: ball = 2; break;
    case 11: ball = 2; break;
    case 12: ball = 2; break;
    case 13: ball = 2; break;
    case 14: ball = 3; break;
    case 15: ball = 3; break;
  }
}

```



Leider hat unser Roboter hinten links eine blinde Stelle das konnten wir bislang allerdings noch nicht klären.

Wenn der Roboter den Ball dann gefunden hat soll er in Richtung Ball fahren. Um möglichst viele Tore zu machen, muss der Roboter immer in Richtung des Tors gucken, denn wir haben noch keinen Zugang zu den Daten unseres Tasters. Das haben wir versucht durch Drehungen zu erreichen, bislang klappt das allerdings nur ungenau. Genauere Erläuterungen kann man den Kommentaren in den Mainprogrammen entnehmen. Sowohl Main9 und Main10 haben ihre eigenen Vorteile und werden von uns deshalb als gleichwertige Programme für den Wettbewerb genutzt.