

# Fußballspielende Roboter

## **Dokumentation**

Für das 6. Semester  
Projekt Autonome Mobile Roboter

Der Studienrichtung Intelligente Systeme  
An der Fachhochschule Brandenburg

Sommersemester 2005

Regina Wehren  
20022028

Katja Orłowski  
20022007

# 1. Inhaltverzeichnis

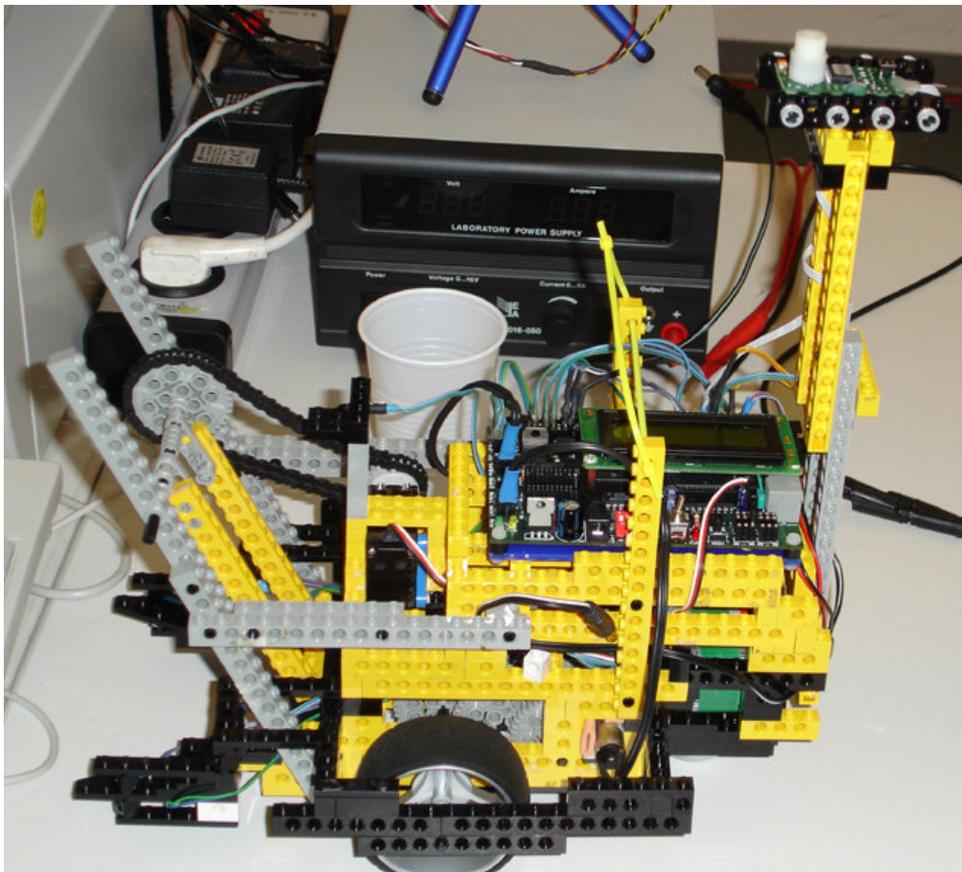
1. Inhaltverzeichnis.....	2
2. Einleitung.....	3
3. Vorstellung .....	3
4. Hardware.....	4
5. Sensoren .....	4
6. Software .....	5
7. Verbesserungen .....	5
8. Anhang: Quelltext.....	5

## 2. Einleitung

Als Studentinnen der Studienrichtung Intelligente Systeme liegen unsere Interessen im Bereich der Künstlichen Intelligenz (KI). Ziel des Projekts „Autonome Mobile Systeme“ sind Lehre und Forschung zu „Intelligenten Fähigkeiten“ wie Wahrnehmen der Umwelt und anderen Fähigkeiten wie Sehen

## 3. Vorstellung

Unser Roboter trägt den Namen Yellow Panther. Er erhielt seinen Namen auf Grund seiner Farbe (überwiegend gelbe Legosteine) und der Hoffnung, dass er recht schnell reagiert und den Ball in das Tor schießt.



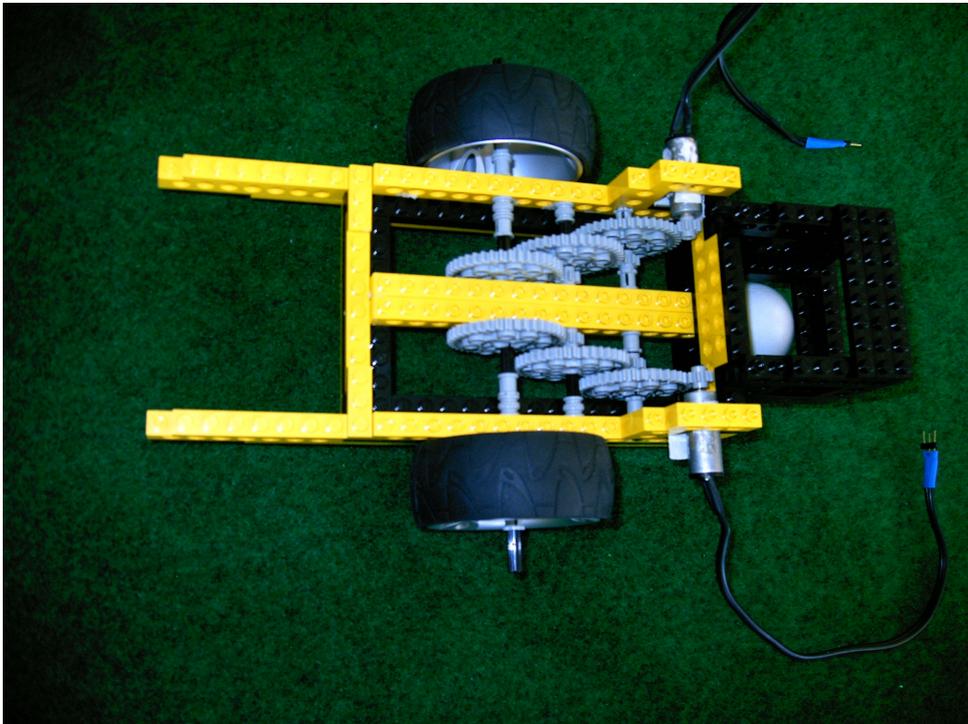
Der Roboter unterteilt sich in die einzelnen Hauptbestandteile:

- Chassis mit Getriebe
- Schussvorrichtung
- Kompass
- AKSEN Board Befestigung

Als besonders gelungen, empfinden wir die Schussvorrichtung und das stabile Gehäuse.

## 4. Hardware

Auf jeder Seite werden ein Modelbaumotor und 6 Zahnräder verwendet. Die Übersetzung des Getriebes liegt bei 1 zu 125. Zur Stabilisation und Wendigkeit des Getriebes wird ein Tischtennisball genutzt. Dazu wird der Ball unter der Akkuladefläche fixiert.



## 5. Sensoren

Für die Torerkennung wird ein Kompass benutzt, der auf dem dafür vorgesehenen Turm angebracht ist. Dieser Turm ist notwendig, da der Kompass nicht in der Nähe von Störquellen wie AKSEN Board, Sensoren und Motoren befestigt werden darf. Vor jeder Benutzung wird der Kompass initialisiert, wobei der Norden immer die Zielrichtung (Tor) anvisiert.

Für die Wandererkennung werden auf jeder Seite (vorn) jeweils ein Infrarot-Empfänger und –Sender befestigt. Diese sind im Winkel von ca. 45° Grad zu jeweiligen Seite ausgerichtet. Dabei wird einmal das Umgebungslicht gemessen (LED ist aus) und danach das reflektierte Licht der LED (LED ist an) der Wand. Diese beiden Werte werden voneinander subtrahiert und mit einem Schwellwert verglichen. Wird der Schwellwert überschritten, ist eine Wand in der Nähe und der Roboter muss dementsprechend ausweichen.

Für die Ballerkennung werden sieben Infrarot-Empfänger benutzt. Einer ist für das Empfangen des Umgebungslichts zuständig und ist somit oberhalb des AKSEN Board montiert. Die anderen sechs Empfänger sind zu Paaren zusammengefasst und an der Front links, mittig und rechts in Höhe der Ballmitte angebracht.

Die empfangenen Werte der Sensorpaare werden von dem Wert des Umgebungslichts abgezogen. Anschließend wird das Maximum des jeweiligen Sensorpaars und das Maximum dieser drei Maxima bestimmt. Überschreitet dieses Maximum einen Schwellwert, sieht der Roboter den Ball rechts, links oder mittig und reagiert dementsprechend.

## **6. Software**

Das Programm von Yellow Panther besitzt eine Zustandsmaschine. Dafür wurde ein Datentyp generiert, welcher aus einer Enumeration, die verschiedene Zustände enthält, besteht (Anhang Zeile 60-70). Für jeden möglichen Zustand existiert eine Zustandsfunktion (do\_\*()) Anhang Zeile 401ff.). Des weiteren gibt es Funktionen die zum Auffinden des Balles (searchBall()), des Tores (searchGoal2()) und der Wand (searchWall()) dienen (Anhang ab Zeile 222).

Jede mögliche simple Bewegung des Roboters ist als eigenständige Funktion definiert (Anhang Abschnitt Bewegungsmethoden Zeile 84ff.). Außerdem besitzt das Programm viele Konstanten um einen hardcodierten Quelltext und das damit aufwendige Ändern zu vermeiden.

## **7. Verbesserungen**

Im Hinblick auf die Revanche im Februar 2006 in Hamburg sollten noch einige Verbesserungen an Yellow Panther vorgenommen werden. Hierzu zählt die Analyse und Verbesserung des zu hohen Stromverbrauchs. Des weiteren muss die Arbeit der Sensorik zur Erkennung des Balls verfeinert werden. In diesem Zusammenhang muss der Einfluss der Umgebung analysiert und so gut wie möglich eingeschränkt werden.

## **8. Anhang: Quelltext**

Siehe Robocup.c