

Hochschule für Angewandte Wissenschaften Hamburg  
(HAW Hamburg)

Bericht

zum Thema

# Wahlpflichtprojekt Mobile Roboter

vorgelegt von : Cemil Yazici(1746542), Cem Karakaya(1618772)

betreut von : Prof. Dr. rer. nat. Kai von Luck,  
Prof. Dr. rer. nat. Gunter Klemke

Hamburg, 01.03.2007

# Inhaltsverzeichnis

<b>1. PROJEKT-BESCHREIBUNG</b>	<b>3</b>
1.1 Aufgabenstellung	4
1.2 Spielweise der Robots	4
1.3 Aksenboard	5
<b>2. VORGEHENSWEISE</b>	<b>9</b>
2.1. Batterien	9
2.2 vor der Programmierung	9
2.3. Programmierung	9
2.4. Motoren	10
2.5. Sensoren	10
2.6. Schranke	12
2.7. Design	13
<b>3. FAZIT</b>	<b>14</b>

# 1. Projekt-Beschreibung

In diesem Projekt geht es im wesentlichen darum, ein sinnvolles Design für einen mobilen Roboter zu entwickeln. In unserem Fall handelt es sich hier um einen Fussball-Roboter.

Das Interessante an diesem Projekt ist es, dass man die Abhängigkeit zwischen Hardware und Software sieht. Wobei man sich vorher genau überlegen sollte, ob das, was man sich softwaretechnisch vornimmt, auch hardwaretechnisch realisieren kann und umgekehrt.

Die Techniken und Methoden die in diesem Projekt angewendet und erlernt werden, sind nicht nur für das theoretische geeignet, sondern auch für die Umsetzungen in der Praxis. Vor allem der Praktische Teil gibt einen fließenden Übergang für das spätere Arbeitsleben.

Das schöne an diesem Projekt ist es auch, dass man nicht nur etwas softwaretechnisch herstellt, was von sich aus zwar funktioniert, aber es muss sich auch in der Hardware beweisen, daher hat man auch direkten Kontakt mit der hardwaretechnischen Seite. Damit alles Reibungslos funktioniert. Infolgedessen kommt man auch mit der elektrotechnischen Lehre in Berührung, so dass man sich nicht nur auf eine Sache fixiert, daraus folgt dass man quasi sagen kann das nicht nur der softwaretechnische Bereich ausreichend ist, sondern man muss die softwaretechnischen, hardwaretechnischen und elektrotechnischen Teile als Ganzes betrachtet.

Als Beispiel: normalerweise haben wir Praktika, wo man entweder mehr hardwaretechnisch oder mehr softwaretechnisch arbeitet, d.h. im Webdesign wird in softwaretechnisch gearbeitet und im Microcontroller Bereich in hardwaretechnisch gearbeitet. Und hier hat man die Möglichkeit sich in beide Bereiche hineinzusteigern, um mal auch über den Tellerrand zur schauen. Da ja das Projekt gleichermaßen für Studierende der angewandten Informatik und der technischen Informatik geeignet ist. Die folgenden Vorgehensweisen sind nur Empfehlungen.

## **1.1 Aufgabenstellung**

In diesem Projekt soll ein fussballspielender Roboter auf Basis des 8-Bit Aksen- Boards gebaut und programmiert werden. Er darf maximal so groß wie ein DIN-A4-Blatt sein. Am Ende des Semesters soll er gegen die anderen Robots antreten.

Das Antreten erfolgt in einem gewissen Rahmen, mit bestimmten Regeln, die vorgegeben sind. Die genauen Regeln werden noch gemeinsam mit allen Studierenden und den Betreuern der Veranstaltung festgelegt. Unsere Empfehlung wäre, die Regeln so früh wie möglich zu bestimmen, damit Ihr dem entsprechend arbeiten könnt und später nichts ändern müsst.

## **1.2 Spielweise der Robots**

Die Roboter treten in einem Feld(siehe Bild unten) gegeneinander auf. Das Feld in dem gegen einander angetreten wird, ist abgestimmt auf die Roboter und deren Sensoren.

Das Antreten der Roboter erfolgt eins zu eins, also kein Gruppen spiel. Am Anfang des Spiels werden die jeweiligen Roboter mit Gesicht zum Tor gestellt, d.h. Rücken an Rücken. Der Ball wird in die Mitte des Feldes gelegt, dabei müssen sich die Roboter beim Start um 180 Grad drehen und so schnell wie möglich auf den Ball zufahren. Es ist ja klar, dass der Roboter der am schnellsten ist, die beste Möglichkeit hat den Ball als erster zu kriegen, um auf das Tor losgehen zu können, weil ja die Anzahl der geschossenen Tore entscheidet ist.

Man sollte nicht vergessen das am Anfang beide Roboter auf den Ball zu rasen. Es sollte also versucht werden gleich nach dem man den Ball hat, den Gegenspieler Rechts oder Links auszudribbeln, bzw. auszuweichen, um mögliche Kollisionen oder Ineinanderkeilung der Robots zu verhindern.

Diese Funktion sollte generell bei der Modellierung berücksichtigt werden. Je besser man das unter Kontrolle hat, desto mehr Tore können die Robots schießen. Dadurch wird auch gleichzeitig verhindert, dass irgendwelche Teile an den Robots abfallen. Außerdem würde es im Allgemeinen ein schöneres und anschauliches Spiel geben, dass auch dann mehr Spaß macht.

Bild vom Spielfeld:



### **1.3 Aksenboard**

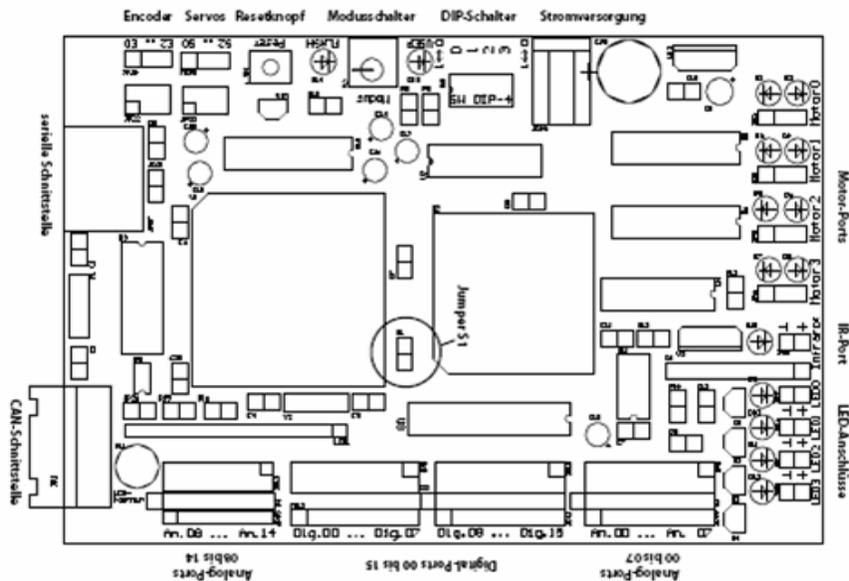
Das AKSEN-Board ist mit vielen Eingängen für Sensoren (digital und analog), digitalen Ausgängen, einem Mikrocontroller der 8051-Familie und einem CAN-Bus-Interface (optional) den gestiegenen Anforderungen der Robotik und der Embedded Systems sowohl in der Lehre als auch im professionellen Einsatz bestens gewachsen.

Das AKSEN-Board realisiert 2 Anwendungsfälle:

1. als eigenständiger Controller für reaktive Robots (ähnlich 6.270, RCX, C-Control etc.)
2. als Motor- und Sensorserver in einem System



Schemazeichnung des AKSEN-Boards :



Das Board wird mit einer Spannung im Bereich von 5 bis 8 Volt betrieben. Die Stromversorgung erfolgt über die Anschlussbuchse rechts neben den DIP-Schaltern. Es ist auf die richtige Polung zu achten. Der Masse-Pol ist innen, die Spannung außen angeschlossen.

Die Spannungsversorgung auf dem AKSEN-Board ist zweigeteilt. Die Versorgung für Servos, Motoren und LEDs ist nicht strombegrenzt und nur von der Leitungsbreite auf dem Board abhängig. Die Spannung ist mit der an der Stromversorgungsbuchse identisch. Alle weiteren Komponenten und Ausgänge werden über einen Spannungsregler auf einen Maximalstrom von 1 Ampere begrenzt.

Die Digitale Ein- und Ausgänge der Ports eignen sich zum Anschluss von digitalen Sensoren, wie z.B. Tastern.

Die analogen Eingänge der analogen Ports messen die am Port anliegende Spannung und stellen den Spannungswert als Zahl zwischen 0 (=0V) und 255 (=5V). Sie sind daher für analoge Sensoren, wie IR-Empfänger-Dioden, Photosensoren oder Optokoppler, bestens geeignet.

Die Motor-Ports sind für Anwendungen mit hohen Lasten geeignet. Pro Port kann maximal ein Ampere Strom bereitgestellt werden. Über die Funktionen der AKSEN-Bibliothek lassen sich die Drehrichtung und die Geschwindigkeit des Motors (in zehn Stufen) ändern.

Die Servo-Ports sind identisch mit den analogen und digitalen Anschlüssen. Die Signal-Pins der Servo-Ports geben ein pulsweitenmoduliertes Signal ab, mit dem die Stellung eines Servos gesteuert werden kann. Die Pulsweite kann über die Servo-Befehle der AKSEN-Bibliothek eingestellt werden.

Diesen Port haben wir für die Schranke benutzt, siehe dazu nächsten Abschnitt Vorgehensweise unter Punkt 6.

Das Verbindungskabel vom AKSEN-Board zum Computer ist ein normales vieradriges Kabel mit einem RJ11-Stecker an dem einen und einem 9-poligen SUB-D-Stecker an dem anderen Ende.

Die serielle Schnittstelle des AKSEN-Boards läuft mit einer Geschwindigkeit von 9600 Baud. Es werden 8 Bit breite Datenworte ohne Parität und mit einem Stoppbit benutzt. Die AKSENBibliothek enthält Funktionen, die eine Kommunikation über die serielle Schnittstelle erlaubt.

## **2. Vorgehensweise**

Hier einige Tipps zur Vorgehensweise :

### **2.1. Batterien**

Beschriftet eure Batterien (z. B. mit eurem Roboternamen) oder am besten kauft ihr euch welche. Denn wir hatten sehr oft das Problem, das unsere Batterien nicht da waren. Manchmal wurden sie auch nicht aufgeladen, obwohl sie im Ladegerät steckten. Das hatte natürlich zur Folge, dass wir an dem Tag nicht effektiv arbeiten konnten. Keine Batterien, keine Testmöglichkeit. ;)

### **2.2 vor der Programmierung**

Bevor ihr anfängt zu programmieren, sollte ihr euch das Achsen-Board-Handbuch gut durchlesen. Dort steht nämlich, wie man die einzelnen Komponenten ansteuern kann.

### **2.3. Programmierung**

Programmiert in kleinen Häppchen und testet. Nicht in große Brocken programmieren, das kann in die Hose gehen.

## 2.4. Motoren

Bei den Motoren solltet ihr nicht so viele Schwierigkeiten haben. Spart euch die Zeit für die Sensoren.



## 2.5. Sensoren

Bei den Sensoren hat man die größten Probleme, weil sich bei verschiedenen Lichtverhältnissen, sich auch die Sensoren anders verhalten. Generell würden wir euch empfehlen programmiert ein Sensor nach dem anderen.

- (a) die Abstandssensoren, damit ihr euch nicht an den Wänden anstößt.
- (b) die Ballsensoren, um den Ball zu finden.
- (c) die Torsensoren, um das Tor zu lokalisieren.

Prüft auf jedem Fall alle Sensoren auf ihre Reichweite. Wenn Ihr eine zu kurze Reichweite habt, bringen euch die Sensoren nichts. Aber da wird euch Herr Luck noch einiges zu sagen haben.

Sharp-Sensoren:



Bild zu 5.a)

---

Ball-Sensoren:

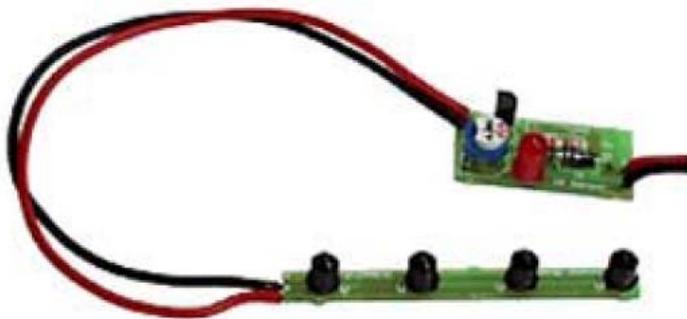


Bild zu 5.b)

Tor-Sensoren:

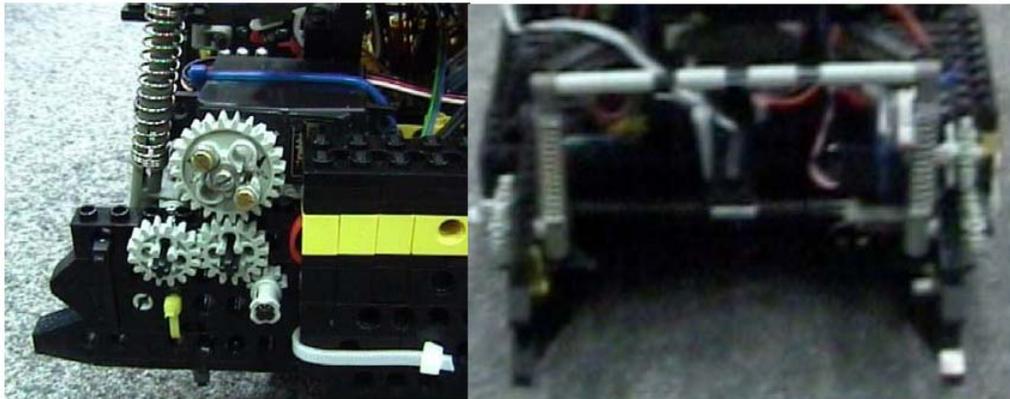


Bild zu 5.c)

## **2.6. Schranke**

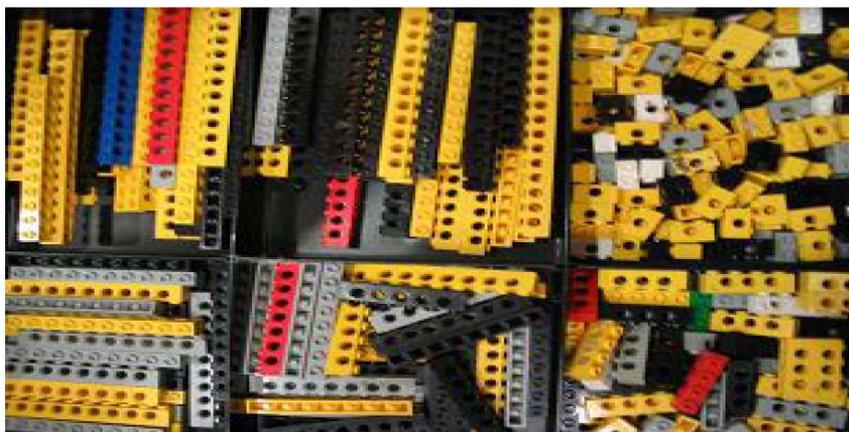
Falls ihr auf den Gedanken kommt, eine Schranke einzubauen, um den Ball zu halten. Überlegt auch das zweimal, denn es gibt genug Neider, die euch ein Strich durch die Rechnung machen. Also unsere Empfehlung, nur wenn ihr genug Zeit habt, baut eine ein.

Bilder von der Schranke (Seitenansicht und Frontansicht):



## 2.7. Design

Zum Design des Roboters würden wir euch empfehlen, baut es so das die Kabel nicht stören. Der Roboter sollte stabil genug sein, damit er nicht gleich auseinanderfällt, zur Not mit Kabelbinder. Baut einen stabilen Griff ein, weil die Schiedsrichter während des Spiels, sehr oft die Roboter auf neutrale Positionen setzen müssen. Wichtig ist, dass der Roboter nicht schön aussehen soll, sondern dass er schnell ist, dass die Sensoren richtig sitzen und es Tore schießt. Wenn das Design einmal steht, solltet ihr keine Zeit für das Redesign verschwenden.



## 3. Fazit

Ihr werdet sehr viel Spaß am Projekt haben, wenn ihr euch an unsere Vorschläge haltet. Sonst könnte es sehr schnell in Stress ausarten. Am Anfang hat es den Anschein, dass man viel Zeit hat, der Schein trügt. Wenn Ihr aber zu den angegebenen Zeiten kommt und arbeitet, müsste es reichen.

Der fertige Fussball-Roboter könnte dann so aussehen:



Zum Schluss noch einige nützliches Link mit Berichten und Bildern:

<http://users.informatik.haw-hamburg.de/~kvl/events.html>

[http://ots.fh-brandenburg.de/downloads/aksen/CD1.04\\_B00.965\\_FW1.4\\_FL0.3\\_H1.04/Handbuch/Handbuch.pdf](http://ots.fh-brandenburg.de/downloads/aksen/CD1.04_B00.965_FW1.4_FL0.3_H1.04/Handbuch/Handbuch.pdf)

Cemil Yazici und Cem Karakaya