



**FACHHOCHSCHUL-BACHELORSTUDIENGANG**  
Automatisierungstechnik / Industrielle Informatik

---

**Programmierung einer HMI für einen Roboter für die EUROBOT<sup>open</sup> 2010**

**ALS BACHELORARBEIT EINGEREICHT**

**zur Erlangung des akademischen Grades**

**Bachelor of Science in Engineering**

**von**

**Christian Mitterndorfer**

**Juni 2010**

---

Betreuung der Bachelorarbeit durch

Prof.(FH) DI Walter Rokitansky

Ich erkläre ehrenwörtlich, dass ich die vorliegende Arbeit selbstständig und ohne fremde Hilfe verfasst, andere als die angegebenen Quellen nicht benutzt, die den benutzten Quellen entnommenen Stellen als solche kenntlich gemacht habe und dass diese Arbeit mit der vom Begutachter beurteilten Arbeit übereinstimmt.

Die Arbeit wurde bisher in gleicher oder ähnlicher Form keiner anderen Prüfungsbehörde vorgelegt und auch nicht veröffentlicht.

.....  
<<Vorname Name>>

<<Wohnort>>, <<Datum>>

## **KURZFASSUNG**

Seit dem Jahr 2007 nimmt das RoboRacingTeam der Fachhochschule Wels an dem internationalen Roboterwettbewerb EUROBOT<sup>open</sup> teil.

Diese Arbeit beschäftigt sich mit der Mensch – Maschinen – Interaktion mittels LCD und Tastatur auf dem Roboter. Sie beschreibt die verwendeten Hardwarekomponenten und deren Aufgaben. Auch gibt sie einen Überblick über die Menüführung und die Menünavigation.

Weiters werden Menüpunkte hinsichtlich ihrer Funktionalität für die diesjährige EUROBOT<sup>open</sup> beschrieben.

Des Weiteren wird die Implementierung der Menüführung, der Informationsspeicherung über das Ausschalten hinaus sowie die Kommunikation zwischen Mainboard und LCD erläutert. Zum Schluss werden weiterführende Möglichkeiten dieses Projektes vorgestellt um auch in Zukunft konkurrenzfähig zu bleiben.

# INHALTSVERZEICHNIS

1	MOTIVATION .....	1
1.1	EUROBOT <sup>open</sup> .....	1
1.1.1	EUROBOT <sup>open</sup> Thema 2010.....	1
2	AUFBAU .....	3
2.1	Eingabegerät .....	3
2.1.1	Tasterbelegung .....	3
2.2	Ausgabegerät .....	4
2.2.1	Versorgung .....	5
2.2.2	LCD – Interface .....	5
2.2.3	Programmier – Interface.....	6
2.2.4	Tastatur – Interface.....	7
2.2.5	SPI – Interface .....	7
3	Realisierung .....	8
3.1	Menüstruktur .....	8
3.2	Speichern auf EEPROM.....	9
3.3	SPI – Meldungen .....	10
3.3.1	Daten werden an LCD – Modul gesendet .....	10
3.3.2	Daten werden vom LCD – Modul abgefragt.....	10
3.3.3	Meldungsnummern.....	11
4	Menü.....	11
4.1	Menüstruktur und Menüführung .....	11
4.2	Startbildschirm .....	14
4.3	Info – Menü .....	14
4.3.1	Seite 1 .....	14
4.3.2	Seite 2.....	15
4.3.3	Seite 3.....	15
4.3.4	Seite 4.....	15
4.4	Einstellungs - Menü.....	16
4.4.1	Strategie.....	17
4.4.2	Farbe.....	17

4.4.3	Kameras.....	18
4.4.4	Absolutpositionsmessung.....	19
4.4.5	Testmenü .....	19
4.4.6	Funkmodul .....	19
4.4.7	Parameter.....	19
5	ZUSAMMENFASSUNG UND AUSBLICK.....	21
6	LITERATUR .....	22
6.1	Bücherliste.....	22
6.2	Bachelorarbeiten.....	22
6.3	Internet.....	22

## ABBILDUNGSVERZEICHNIS

Abbildung 1-1: Spielfeld EUROBOT <sup>open</sup> Thema 2010.....	2
Abbildung 2-1: Tastatur .....	3
Abbildung 2-2: LCD – Display der Firma Displaytec.....	4
Abbildung 2-3: LCD – Board des modularen Systems .....	4
Abbildung 2-4: Blockschaltbild des LCD .....	5
Abbildung 2-5: Die SPI – Schnittstelle zur Datenübertragung zwischen AVR - Controllern ...	7
Abbildung 3-1: Aufbau der SPI – Meldung beim Senden von Daten vom Master zum Slave	10
Abbildung 3-2: Aufbau der SPI – Meldung beim Senden von Daten vom Slave zum Master	10
Abbildung 4-1: Menüstruktur und Menüführung (INFO – Menü).....	12
Abbildung 4-2: Menüstruktur und Menüführung (Einstellungs – Menü).....	13
Abbildung 4-3: Startbildschirm.....	14
Abbildung 4-4: Info – Menü Seite 1.....	14
Abbildung 4-5: Info – Menü Seite 2.....	15
Abbildung 4-6: Info – Menü Seite 3.....	15
Abbildung 4-7: Info – Menü Seite 4.....	16
Abbildung 4-8: Einstellungs – Menü 1 .....	16
Abbildung 4-9: Einstellungs – Menü 2 .....	17
Abbildung 4-10: Einstellungs – Menü 3 .....	17
Abbildung 4-11: Einstellungs – Menü 3 .....	17
Abbildung 4-12: Strategie – Auswahlmenü .....	17
Abbildung 4-13: Farbe – Auswahlmenü .....	18
Abbildung 4-14: Kamera – Einstellmenü 1.....	18
Abbildung 4-15: Kamera – Einstellmenü 2.....	18
Abbildung 4-16: Kamera – Einstellmenü 3.....	18
Abbildung 4-17: Absolutposition – Einstellmenü.....	19
Abbildung 4-18: Funkmodul – Einstellmenü .....	19

## TABELLENVERZEICHNIS

Tabelle 1-1: Spielelemente und ihre Punkte.....	1
Tabelle 2-1: Steckerbelegung der Versorgung .....	5
Tabelle 2-2: Steckerbelegung des LCD – Interface.....	6
Tabelle 2-3: Steckerbelegung des Programmier – Interface.....	6
Tabelle 2-4: Steckerbelegung des Tastatur – Interface.....	7
Tabelle 2-5: Steckerbelegung des SPI – Interface .....	8

# 1 MOTIVATION

Seit dem Jahr 2007 nimmt das RoboRacingTeam der Fachhochschule Wels an dem internationalen Roboterwettbewerb EUROBOT<sup>open</sup> teil.

In den letzten Jahren konnten immer beachtliche Erfolge verbucht werden. Um an diese Erfolge auch dieses Jahr anzuknüpfen, ist es von großer Bedeutung den Roboter schnell und einfach bedienen zu können. Für diesen Zweck wurde eine geeignete Mensch – Maschinen – Interaktion – Schnittstelle geschaffen, um den Roboter in moderater Zeit auf die verschiedensten Bedingungen einzustellen.

## 1.1 EUROBOT<sup>open</sup><sup>1</sup>

Die EUROBOT<sup>open</sup> findet jedes Jahr Ende Mai statt. Teilnahmeberechtigt sind Teams aus allen Ländern.

Charakteristisch für die EUROBOT<sup>open</sup> ist, dass sich die Aufgabenstellungen und die Regeln Jahr für Jahr ändern. Die Aufgabenstellung und die Regeln werden jedes Jahr Ende September bekannt gegeben, wodurch sich eine Entwicklungszeit von 8 Monaten ergibt. Grundlegend ist, dass die Entwicklungszeit für alle Teams gleich ist. Lediglich die Größe des Spieltisches mit 210cm x 300cm und die Spieldauer mit 90sec. bleiben jedes Jahr gleich.

### 1.1.1 EUROBOT<sup>open</sup> Thema 2010<sup>2</sup>

Das Thema der EUROBOT<sup>open</sup> 2010 ist „Feed the world“. Aufgabe des Roboters ist es, drei verschiedene Spielelemente, welche drei verschiedene Nahrungsmittel darstellen, aufzunehmen und zu einer Entladestelle zu bringen. Jedes Spielelement bringt verschiedene Punkte. Je mehr Gewicht das Element hat, und desto schwerer es aufzunehmen ist, desto mehr Punkte bringt es. Tabelle 1-1 zeigt eine Übersicht der Spielelemente, welche Nahrungsmittel sie darstellen und wie viele Punkte sie bringen.

Nahrungsmittel	Spielelement	Gewicht (g) / Punkte
Tomaten	Bälle / Rot	je 150
Maiskolben	Zylinder	je 250
Orangen	Bälle / Orange	je 300

*Tabelle 1-1: Spielelemente und ihre Punkte*

---

<sup>1</sup> [eurobot, 2010]

<sup>2</sup> [Rules E2010, 2009]

Der Roboter, der in 90 sec. die meisten Punkte sammelt und auch bei der Entladestelle auslehrt, gewinnt. Abbildung 1-1 zeigt die Verteilung der Spielelemente auf dem Spielfeld.

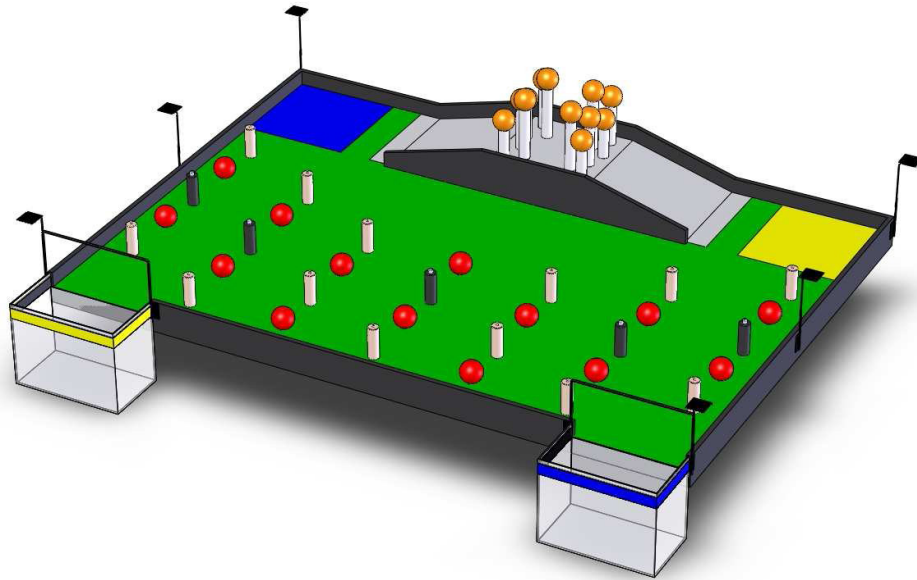


Abbildung 1-1: Spielfeld EUROBOT<sup>open</sup> Thema 2010

Wie aus Abbildung 1-1 ersichtlich gibt es zwei Spielfarben, Gelb und Blau. Die Entladestellen sind jeweils gegenüber der Startzone.

Eine Schwierigkeit besteht darin, dass die Orangen, welche die meisten Punkte bringen, auf einem erhöhten Podest auf „Bäumen“ platziert sind. Der Roboter muss einerseits die Steigung der Rampe überwinden und andererseits einen Mechanismus besitzen die Orangen von den Bäumen zu holen. Eine weitere Schwierigkeit besteht darin, dass es schwarze und weiße Zylinder am Spielfeld gibt. Während die weißen Zylinder zum Einsammeln sind und dadurch Punkte bringen, sind die schwarzen am Tisch festgeschraubt und bringen keine Punkte. Viel mehr besteht die Gefahr, dass der Roboter an den schwarzen Zylindern hängen bleibt oder durch einen hängen gebliebenen gegnerischen Roboter blockiert wird. Weiters ist die Aufnahme der Tomaten relativ schwierig, da sie frei herumrollen können und ihre Position dadurch nicht fix definiert ist.

## 2 AUFBAU

Die Komponenten der Mensch – Maschinen – Interaktion für den Roboter bestehen im Wesentlichen aus einem Eingabegerät, der Tastatur, und einem Ausgabegerät, dem LCD. Diese beiden Elemente sind Teil des modularen Systems<sup>3</sup> der FH – Wels und werden in Folgendem näher beschrieben.

### 2.1 Eingabegerät

Grundlegend wird das Menü mit einer Vier – Taster – Tastatur der Firma Storm – Interface<sup>4</sup> bedient. Dabei wurde bei der Belegung der Taster besonders auf die Menüführung (siehe 3) Rücksicht genommen. Der Anschluss der Tastatur erfolgt über einen fünfpoligen Stecker, wobei Pin 1 der gemeinsame Anschluss für alle Taster und Pin 2 bis 5 die einzelnen Taster sind. Eine hardwaremäßige Tasterentprellung ist nicht vorgesehen, dies muss softwaremäßig realisiert werden. Auf dem LCD – Modul sind die Tastereingänge mit Pull – Down Widerständen verbunden, die Taster sind also High – Aktiv.

#### 2.1.1 Tasterbelegung

Die Tasterbelegung ist wie folgt:

- Enter – Taster um vom Menü aus in ein Untermenü zu gelangen oder um eine Änderung zu bestätigen.
- Pfeil – Auf – Ab – Taster um im Menü zu navigieren oder um eine Einstellung zu ändern.
- Escape – Taster um von einem Untermenü in das Höhergelegene Menü zu gelangen oder um eine Änderung zu verwerfen.



Abbildung 2-1: Tastatur

---

<sup>3</sup> [Zauner, 2009]

<sup>4</sup> [storm-interface, 2010]

## 2.2 Ausgabegerät

Als Ausgabegerät wurde das Display 204A der Firma Displaytech Ltd<sup>5</sup>. gewählt (siehe Abbildung 2-2: LCD – Display der Firma Displaytech). Bei diesem Display handelt es sich um ein Standard – Display mit 4 Zeilen mit jeweils 20 Zeichen. Zur Ansteuerung dieses Displays ist im modularen System ein LCD – Board vorgesehen (siehe Abbildung 2-3: LCD – Board des modularen Systems). Dieses LCD – Board bietet des Weiteren folgende Schnittstellen.

- Versorgung
- LCD – Interface
- Programmier – Interface
- Tastatur – Interface
- SPI – Interface

Diese Schnittstellen sind in Folgendem genauer beschrieben.

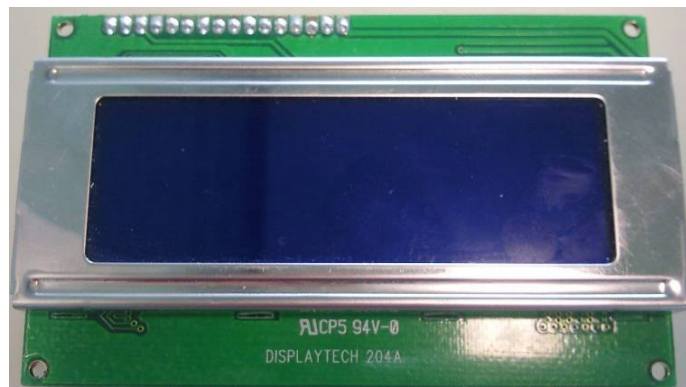


Abbildung 2-2: LCD – Display der Firma Displaytec

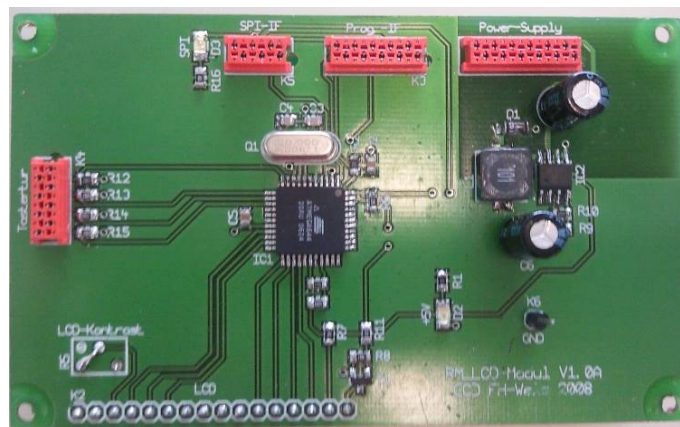


Abbildung 2-3: LCD – Board des modularen Systems

---

<sup>5</sup> [displaytech, 2010]

## 2.2.1 Versorgung

An diesem Interface wird die Akkuspannung zur Verfügung gestellt. Die 5V Spannungsversorgung für die Elektronik werden auf dem Board über einen Schaltregler bereitgestellt. Durch einen Spannungsteiler auf dem LCD – Board kann die Akkuspannung mit einen Analogeingang des Mikrocontrollers gemessen werden. Die Steckerbelegung ist Tabelle 2-1 zu entnehmen.

Pin		Funktion
Nummer	Bezeichnung	
1,3,5,7,9,11	Akku+	Pluspol der Versorgung
2,4,6,8,10,12	GND	Masse der Versorgung

Tabelle 2-1: Steckerbelegung der Versorgung

## 2.2.2 LCD – Interface

Abbildung 2-4 zeigt das Blockschaltbild des LCD. Der Controllerchip ST7066U von Sitronix<sup>6</sup> ist kompatibel zum Controllerchip HD44780 von Hitachi der sich als Standard durchgesetzt hat. Die Steckerbelegung des LCD – Interface ist Tabelle 2-2 zu entnehmen.

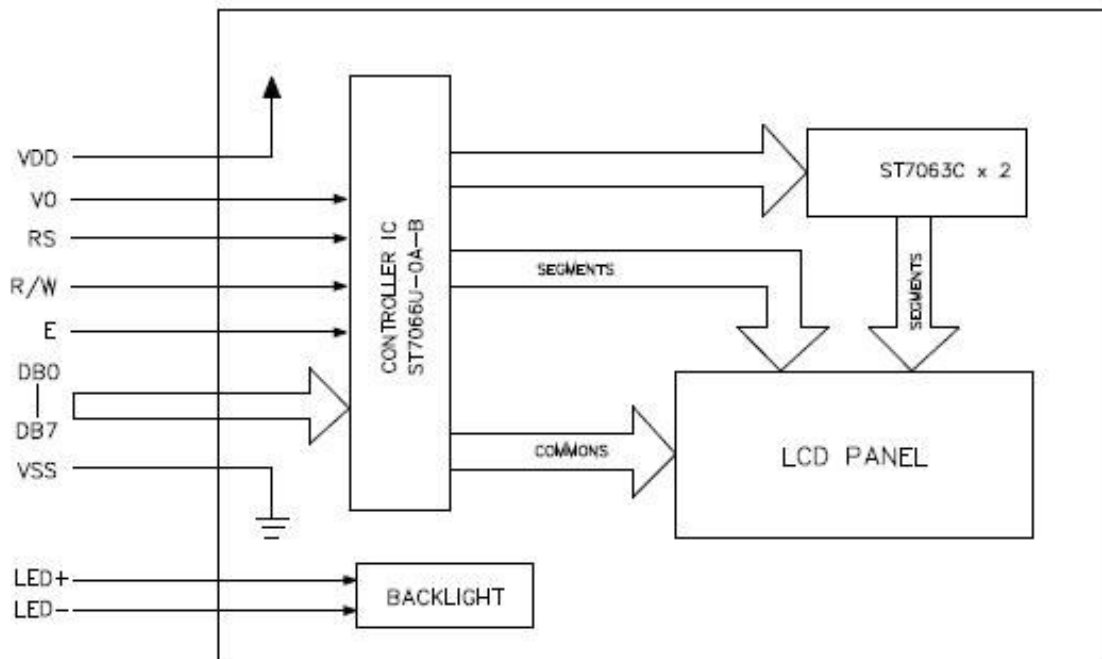


Abbildung 2-4: Blockschaltbild des LCD

<sup>6</sup> [Sitronix, 2010]

Pin		Funktion
Nummer	Bezeichnung	
1	GND	Masse der Versorgung
2	+5V	Pluspol der Versorgung
3	VO	Kontrasteinstellung des LCD - Display
4	RS	Registerauswahl 0 --> Befehlsregister 1 --> Datenregister
5	R/W	Read / Write 0 --> Schreiben 1 --> Lesen
6	EN	Enable
7,8,9,10	DB0..3	Datenleitung 0..3 (im 4-Bit Modus nicht verwendet)
11,12,13,14	DB4..7	Datenleitung 4..7
15	A	Anode der Hintergrundbeleuchtung
16	K	Kathode der Hintergrundbeleuchtung

*Tabelle 2-2: Steckerbelegung des LCD – Interface*

### 2.2.3 Programmier – Interface

Der Atmel – Mikrocontroller ATmega644 erlaubt eine In – System – Programmierung über die SPI – Anschlüsse (siehe 2.2.5). Weiters sind auf das Programmierinterface auch die Send- und Empfangsleitung der Debug – Schnittstelle geführt. Tabelle 2-3 zeigt die Belegung des Programmier – Interface.

Pin		Funktion
Nummer	Bezeichnung	
1	+5V	Pluspol der Versorgung
2	RxD	Empfangsleitung der Debug - Schnittstelle
4	SCK	Clock - Leitung der Programmier - Schnittstelle
5	TxD	Sendeleitung der Debug - Schnittstelle
6	MISO	Master In Slave Out - Leitung der Programmier - Schnittstelle
8	MOSI	Master Out Slave In der Programmier - Schnittstelle
9	GND	Masse
10	/Reset	Reset - Leitung der Programmier - Schnittstelle (Low - Aktiv)
3,7	NC	nicht in Verwendung

*Tabelle 2-3: Steckerbelegung des Programmier – Interface*

## 2.2.4 Tastatur – Interface

Wie bereits beschrieben besitzt, die Tastatur einen gemeinsamen Anschluss und vier Anschlüsse für die Taster. Tabelle 2-4 zeigt die Belegung des Tastatur – Interface auf dem LCD – Modul.

Pin		Funktion
Nummer	Bezeichnung	
1,3,5,7	+5V	Pluspol der Versorgung
2	T1	Taster 1 --> Escape - Taster
4	T2	Taster 2 --> Pfeil - Auf - Taster
6	T3	Taster 3 --> Pfeil - Ab - Taster
8	T4	Taster 4 --> Enter - Taster

Tabelle 2-4: Steckerbelegung des Tastatur – Interface

## 2.2.5 SPI – Interface<sup>7</sup>

Die SPI – Schnittstelle (Serial Peripheral Interface) arbeitet synchron mit einer gemeinsamen Taktleitung. Dieser Takt beträgt beim modularen System 625kBaude. Im Wesentlichen handelt es sich um eine Master – Slave – Verbindung wobei der Master den Zeitpunkt der Übertragung bestimmt, indem er am Ausgang SCK (Shift Clock) den Schiebetak an den Slave gibt. Beim modularen System der FH – Wels ist das Mainboard als Master vorgesehen, wodurch das LCD im Slave – Betrieb arbeitet. Abbildung 2-5 zeigt den prinzipiellen Aufbau dieser Schnittstelle.

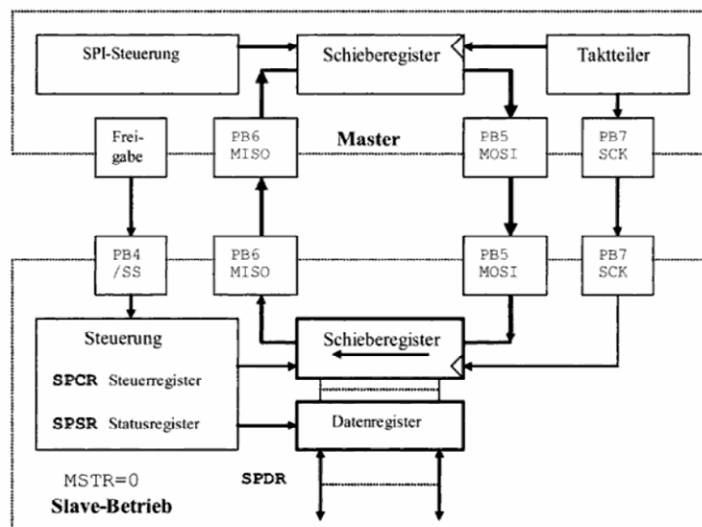


Abbildung 2-5: Die SPI – Schnittstelle zur Datenübertragung zwischen AVR - Controllern

<sup>7</sup> [Schmitt, 2007]

Die Schnittstelle besteht eigentlich aus zwei Schieberegistern. Ein Low – Zustand am SS - (Slave Select) Ausgang des Masters startet die Übertragung. Der Master und der Slave schreiben die zu sendenden Daten in die Schieberegister. Durch den gemeinsamen Takt werden die Daten über den Master Out – Slave In Pin (MOSI) vom Master zum Slave, und über den Master In – Slave Out Pin (MISO) übertragen.

Beim Anschluss mehrerer Slaves an einen Master, wie bei diesem Roboter, entsteht ein Bussystem, bei dem der Master jeweils einen Slave durch den Eingang SS freigibt.

Wie bereits erwähnt ist es möglich, den Mikrocontroller über die SPI – Schnittstelle In – System zu programmieren. In diesem Fall arbeitet das Programmiergerät als Master und der zu programmierende Baustein als Slave.

Tabelle 2-5 zeigt die Belegung des SPI – Interface am LCD – Modul.

Pin		Funktion
Nummer	Bezeichnung	
1	/SS	Slave – Select (Low - Aktiv)
2	SCK	Synchroner Clock der SPI - Schnittstelle
3	MISO	Master In Slave Out zur Datenübertragung zum Master
4	MOSI	Master Out Slave In zur Datenübertragung zum Slave
5,6	GND	Masse

*Tabelle 2-5: Steckerbelegung des SPI – Interface*

### 3 Realisierung

Aufbauend auf dem kooperativen Multitasking System<sup>8</sup> der FH – Wels wurde das LCD in C mit der Entwicklungsumgebung CodeVisionAVR<sup>9</sup> programmiert. Die wichtigsten Programmteile sind in Folgendem beschrieben.

#### 3.1 Menüstruktur

Um das Programm so übersichtlich wie möglich zu gestalten wurde für jedes Menü ein eigener Task angelegt. Jedes Menü das aufgerufen wird beendet den aufrufenden Task, wodurch die Anzahl der parallel laufenden Task gering gehalten wird.

Für die Menüstruktur selbst wurde eine Struktur angelegt.

---

<sup>8</sup> [Zauner, 2009]

<sup>9</sup> [codevision, 2010]

```
struct Menue
{
flash unsigned char *ucText;
unsigned char pprev;
unsigned char prev;
unsigned char next;
unsigned char nnext;
void (*callUnterprogramm)(void);
};

typedef struct Menue stMenue;
```

Der Datentyp struct fasst Elemente verschiedener Datentypen zusammen<sup>10</sup>. In diesem Fall besteht die Struktur aus einem Zeiger auf einen Text im Flash – Speicher. Weiters werden die Menüpunkte 2 Schritte zurück und 2 Schritte nach vor und ein Zeiger auf ein eventuell aufzurufendes Unterprogramm abgespeichert. Um beim Anlegen einer Menüvariable nicht immer das Schlüsselwort struct anführen zu müssen wird mit typedef struct ein benutzerdefinierter Strukturtyp definiert.

### 3.2 Speichern auf EEPROM

Für die Funktionalität des Roboters ist es von Vorteil wenn die getätigten Einstellungen auch nach dem Ausschalten gespeichert bleiben. Aus diesem Grund werden alle wichtigen Daten auf dem EEPROM der Mikrocontrollers gespeichert. Der ATmega664 stellt einen 2kByte großen EEPROM – Speicher zur Verfügung<sup>11</sup>.

Unter einen EEPROM versteht man einen durch den Anwender programmierbaren, elektrisch löschbaren Festwertspeicher<sup>12</sup>. Der Zugriff ist mit der Entwicklungsumgebung CodeVisionAVR denkbar einfach. Das Schlüsselwort „eeprom“ vor einer Variable reicht aus um dem Compiler mitzuteilen, dass sich diese Variable im EEPROM – Speicher befindet. Der Compiler übernimmt dann die Zugriffsverwaltung. Dies verleitet jedoch dazu, das EEPROM zu oft zu beschreiben. Die Lebensdauer des EEPROM ist lt. Datenblatt  $10^5$  Schriebzyklen.

---

<sup>10</sup> [Schmitt, 2007]

<sup>11</sup> [Atmel, 2010]

<sup>12</sup> [Tietze & Schenk, 1999]

Bei einer lt. Datenblatt typischen Programmierzeit von 3,3ms könnten diese  $10^5$  Schreibzyklen bereits nach 330 Sekunden verbraucht sein. Aus diesem Grund wird das EEPROM des LCD – Moduls immer nur dann beschrieben, wenn tatsächlich eine Änderung eines Parameters stattgefunden hat.

### 3.3 SPI – Meldungen

Die Kommunikation zwischen LCD – Modul und Mainboard erfolgt in beide Richtungen. Einerseits muss das LCD – Board Daten vom Mainboard verarbeiten, andererseits muss das LCD – Board auch Daten an das Mainboard senden. Um diese Übertragungen so einheitlich wie möglich zu gestalten wurde für beide Fälle ein einheitliches Protokoll vereinbart. Egal ob Senden oder Empfangen, die Meldungen bestehen immer aus drei Byte. Das erste Byte legt fest ob Daten gesendet werden oder ob Daten empfangen werden.

- Daten werden an LCD – Modul gesendet: 1
- Daten werden vom LCD – Modul abgefragt: 0

Der Aufbau des Protokolls beim Senden oder Empfangen ist in Folgendem näher beschrieben.

#### 3.3.1 Daten werden an LCD – Modul gesendet

Abbildung 3-1 zeigt den Fall das Daten an das LCD – Modul gesendet werden. Durch das zweite Byte teilt der Master dem Slave mit, welche Daten gesendet werden. Das dritte Byte sind dann die eigentlichen Daten.

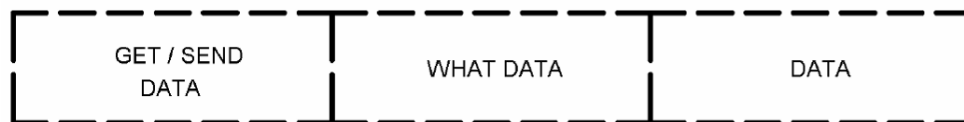


Abbildung 3-1: Aufbau der SPI – Meldung beim Senden von Daten vom Master zum Slave

#### 3.3.2 Daten werden vom LCD – Modul abgefragt

Abbildung 3-2 zeigt den Fall das Daten vom LCD – Modul abgefragt werden. Auch in diesem Fall gibt das zweite Byte Auskunft darüber, welche Daten abgefragt werden. Das dritte Byte dann ist ein Leerbyte in welchem die gewünschten Daten vom Slave an den Master gesendet werden.

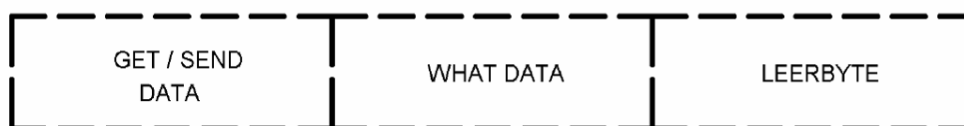


Abbildung 3-2: Aufbau der SPI – Meldung beim Senden von Daten vom Slave zum Master

### 3.3.3 Meldungsnummern

Folgende Meldungsnummern (zweites Byte) wurden vereinbart.

#### Daten werden an LCD – Modul gesendet:

- (1) Spielzeit
- (2) Konstellation der Zylinder
- (3) X – Position des Roboters
- (4) Y – Position des Roboters

#### Daten werden vom LCD – Modul abgefragt:

- |                             |                 |           |
|-----------------------------|-----------------|-----------|
| (1) Absolutpositionsmessung | 0 → nicht aktiv | 1 → aktiv |
| (2) Nummer der Strategie    | 0 – 9           |           |
| (3) mittlere Kamera         | 0 → nicht aktiv | 1 → aktiv |
| (4) seitliche Kamera        | 0 → nicht aktiv | 1 → aktiv |
| (5) Orangenkamera           | 0 → nicht aktiv | 1 → aktiv |
| (6) Kanal des Funkmoduls    | 0 – 9           |           |
| (7) Spielfarbe              | 1 → Blau        | 2 → Gelb  |

## 4 Menü

Da das Regelwerk der Eurobot<sup>open</sup> vor jedem Match lediglich 3 Minuten Vorbereitungszeit vorsieht, und in diesen 3 Minuten nicht nur der eigene Roboter sondern auch die Kameras sowie die Gegnererkennung konfiguriert werden müssen, ist es von Vorteil, eine einfache Menüführung zu haben um den Roboter schnell auf die verschiedenen Gegebenheiten einzustellen. Weiters sollten alle Einstellungen einfach und schnell zu kontrollieren sein. In Hinsicht auf diese Aspekte wurde folgende Menüstruktur und Menüführung entwickelt.

### 4.1 Menüstruktur und Menüführung

Das Menü soll, wie oben beschrieben, zwei Aufgaben erfüllen. Einerseits sollen alle Einstellungen schnell kontrollierbar sein, andererseits sollen Änderungen in möglichst kurzer Zeit realisiert werden können. Um das zu gewährleisten, wurde Info- und Einstellungsmenü thematisch getrennt.

Die Menüführung erfolgt über die vier Taster wie unter Punkt 2.1.1 Tasterbelegung beschrieben. Abbildung 4-1 und 4-2 zeigen eine grobe Übersicht über Menüpunkte und Menüführung.

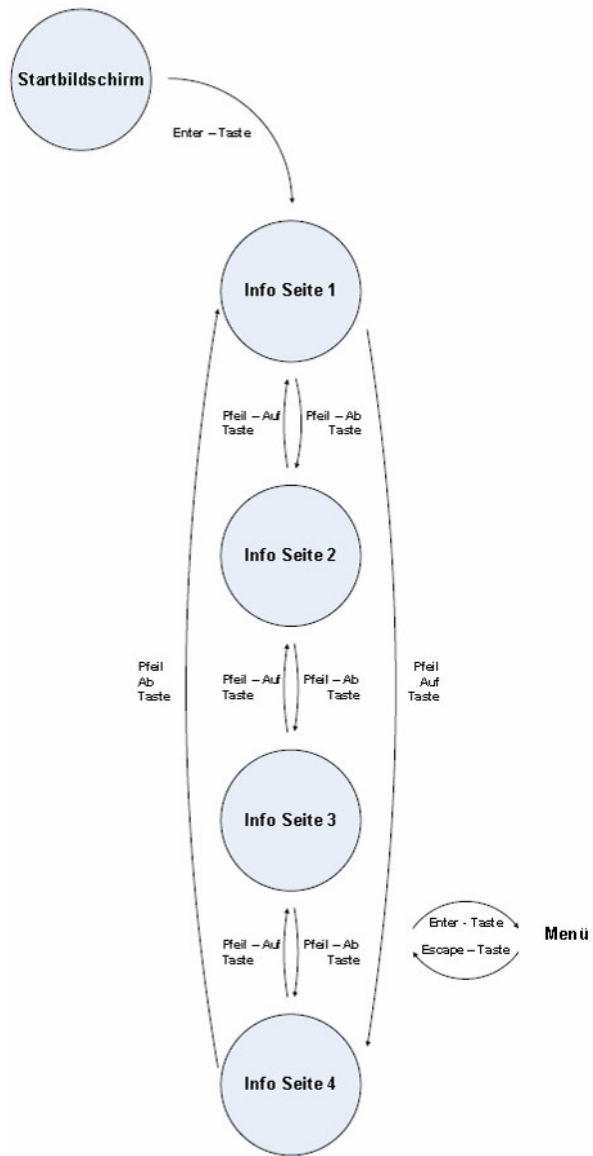


Abbildung 4-1: Menüstruktur und Menüführung (INFO - Menü)

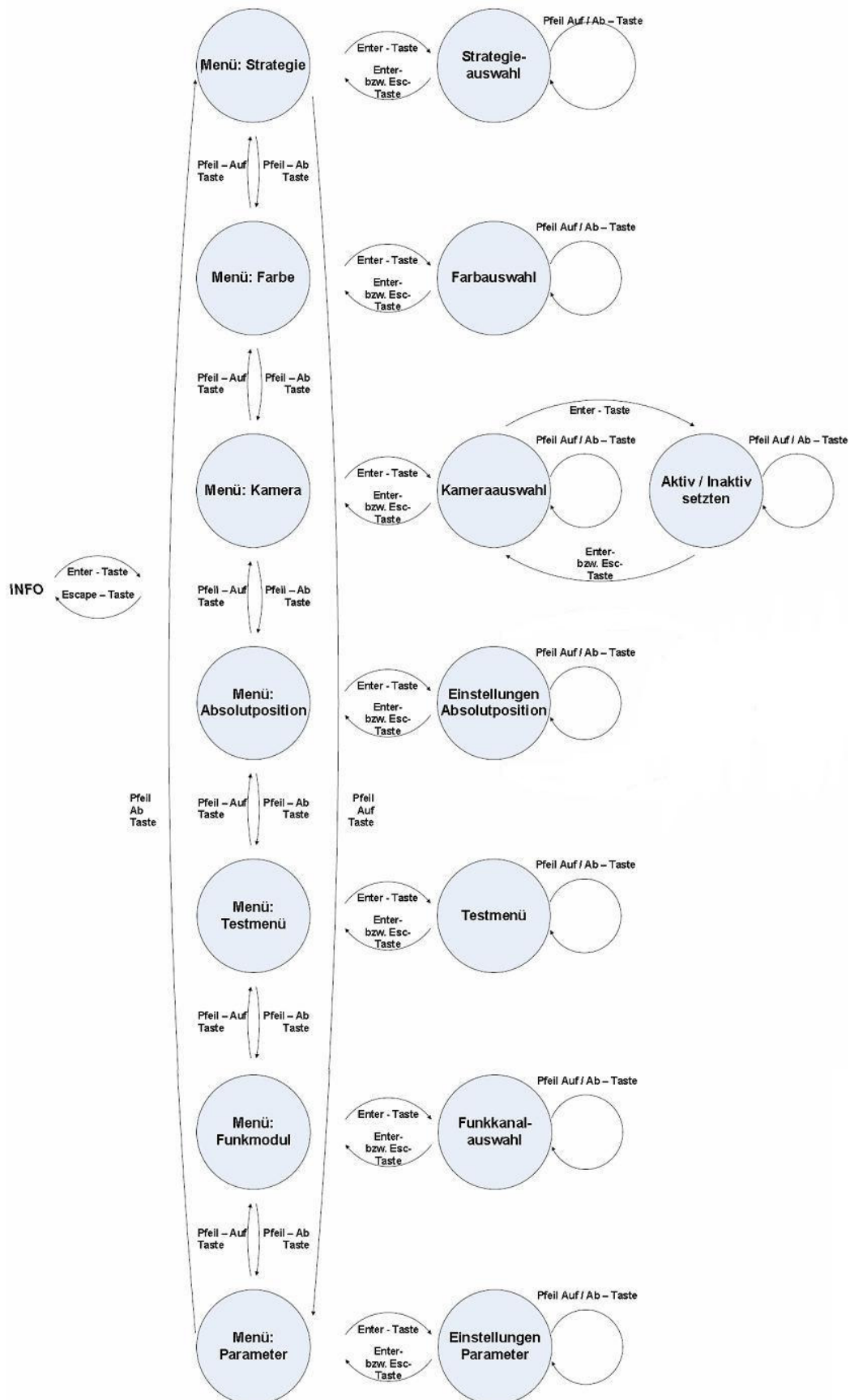


Abbildung 4-2: Menüstruktur und Menüführung (Einstellungs - Menü)

Die einzelnen Positionen sind in Folgendem beschrieben.

## 4.2 Startbildschirm

Nach dem Einschalten des Roboters wird der in Abbildung 4-3 gezeigte Startbildschirm angezeigt. Dieser hat für die Funktion des Roboters keine Bedeutung sondern gibt nur Auskunft über den Bewerb, in diesem Fall „Eurobot 2010 – Feed the World“, und über die Funktionalität des LCD. Nachdem dieser Bildschirm mit der Enter – Taster verlassen wurde kann dieser Bildschirm nicht wieder aufgerufen werden.



Abbildung 4-3: Startbildschirm

## 4.3 Info – Menü

Im Info – Menü sind alle wichtigen Einstellungen wiedergegeben. Um die Vielzahl von Informationen unterzubringen besteht dieses Menü aus 4 Seiten, wobei die Informationen der Wichtigkeit nach sortiert sind.

Eine wichtige Funktion des Info – Menüs ist, anzuzeigen ob getätigte Änderungen auch an das Mainboard weitergegeben wurden. Wenn am LCD eine Änderung getätigt wird, wird diese im Info – Menü mit einem \* gekennzeichnet. Dieses Zeichen wird gelöscht wenn das Mainboard die Änderung abgefragt, und somit übernommen, hat.

### 4.3.1 Seite 1

Abbildung 4-4 zeigt Seite 1 des Info – Menüs. Es wurde festgelegt das Akkuspannung, eingestellte Strategie und die restliche Spielzeit die wichtigsten Parameter sind und deshalb auf der ersten Seite angezeigt werden.



Abbildung 4-4: Info – Menü Seite 1

### 4.3.2 Seite 2

Abbildung 4-5 zeigt Seite 2 des Info – Menüs. Die Konstellation der Zylinder sowie die X- und Y- Position des Roboters sind Informationen deren Wichtigkeit am Anfang als hoch eingestuft wurde. Da diese Information jedoch von anderen Faktoren wie Kameras<sup>13</sup> und Absolutpositionsmessung<sup>14</sup> abhängig war, konnte diese Funktion aus Zeitgründen nicht mehr realisiert werden.



Abbildung 4-5: Info – Menü Seite 2

### 4.3.3 Seite 3

Abbildung 4-6 zeigt Seite 3 des Info – Menüs. Wie bereits beschrieben stehen vor jedem Match lediglich 3 Minuten zur Verfügung um den Roboter und die Kameras zu konfigurieren. Vor allem bei den Kameras stellt das ein Problem dar. Im Vorfeld konnte nicht genau gesagt werden ob diese 3 Minuten ausreichen um die Kameras zu konfigurieren, weshalb Softwaremäßig jede Kamera einzeln deaktiviert werden konnte. Abbildung 4-6 zeigt den Fall, dass keine Kamera aktiv ist, jedoch ist die Einstellung für die Orangenkamera vom Mainboard noch nicht übernommen.



Abbildung 4-6: Info – Menü Seite 3

### 4.3.4 Seite 4

Abbildung 4-7 zeigt Seite 4 des Info – Menüs. Wie bereits erwähnt war die Absolutpositionsmessung Teil eines anderen Projektes. Auf Grund des Zeitdruckes war im

---

<sup>13</sup> [Muckenhumer, 2010]

<sup>14</sup> [Stotter, 2010]

Vorfeld nicht sichergestellt ob die Absolutpositionsmessung termingerecht fertig gestellt werden kann, weshalb auch diese mittels Software deaktiviert werden konnte.

Weiters war bei dem Eurobot<sup>open</sup> Roboter ein Funkmodul in Verwendung. Durch die Erfahrungen der Vorjahre war bekannt, dass auch andere Teams dieses Funkmodul in Verwendung haben. Um nun Kollisionen im Funkverkehr zu vermeiden, konnte man über das Menü den Funkkanal des Moduls ändern.

Auch die eingestellte Farbe wird auf der letzten Seite des Info – Menüs angezeigt. Diese Information ist neben denen auf Info – Seite 1 eine der wichtigsten, jedoch steht sie am Roboter auch als Led – Anzeige zur Verfügung, weshalb sie am LCD nur auf der vierten und damit letzten Seite angezeigt wird.



Abbildung 4-7: Info – Menü Seite 4

## 4.4 Einstellungs - Menü

Im diesem Menü können alle Einstellungen vorgenommen werden. Dadurch ist es möglich, den Roboter schnell auf die verschiedenen Gegebenheiten einzustellen.

Um die Übersicht zu erhöhen wird die aktuelle Cursorstellung durch das Zeichen > gekennzeichnet. Die Ausführung ist ringförmig, das heißt man kommt vom letzten Punkt durch einen Tasterdruck wieder auf den ersten Punkt und umgekehrt.

Im Vorfeld wurden 7 Einstellungspunkte festgelegt. Bei der Reihung der Punkte wurde auf die Wichtigkeit Rücksicht genommen. Abbildung 4-8 bis 4-11 zeigen die einzelnen Menüpunkte.



Abbildung 4-8: Einstellungs – Menü 1



Abbildung 4-9: Einstellungs – Menü 2



Abbildung 4-10: Einstellungs – Menü 3



Abbildung 4-11: Einstellungs – Menü 3

Die einzelnen Menüpunkte sind in Folgendem beschrieben.

#### 4.4.1 Strategie

Das Strategie – Auswahlmenü ist wie das Einstellungs-menü aufgebaut. Auch hier wird die aktuelle Cursorposition und somit die Auswahl durch > angezeigt.



Abbildung 4-12: Strategie – Auswahlmenü

#### 4.4.2 Farbe

Das Farbe – Auswahlmenü ist wie das Einstellungs-menü aufgebaut. Auch hier wird die aktuelle Cursorposition und somit die Auswahl durch > angezeigt.



Abbildung 4-13: Farbe – Auswahlmenü

#### 4.4.3 Kameras

Der Aufbau dieses Menüs weicht von den anderen etwas ab. Zuerst wird die einzustellende Kamera ausgewählt (siehe Abbildung 4-14 und 4-16). Will man die Einstellung einer Kamera ändern, muss die gewünschte Kamera mit der Enter – Taster ausgewählt werden. Das \* - Symbol an der Seite des Kameranamens zeigt an, dass sich diese Kamera nun im Änderungsmodus befindet (siehe Abbildung 4-15). Das > - Symbol zeigt an welche Einstellung gerade ausgewählt ist.



Abbildung 4-14: Kamera – Einstellmenü 1



Abbildung 4-15: Kamera – Einstellmenü 2



Abbildung 4-16: Kamera – Einstellmenü 3

#### 4.4.4 Absolutpositionsmessung

Aufgebaut wie z.B. das Farbmenü kann in diesem Menü die Absolutpositionsmessung aktiv bzw. inaktiv gesetzt werden.



Abbildung 4-17: Absolutposition – Einstellmenü

#### 4.4.5 Testmenü

Dieses Menü wurde vorgesehen um die Funktionalität des Roboters testen zu können. So sollten in diesem Menü Aktoren angesteuert und Sensoren ausgelesen werden können. Im Verlauf der Entwicklung stellte sich allerdings heraus, dass dies für die Aufgabenstellung in diesem Jahr nicht von Bedeutung ist, weshalb dieses Menü auch nicht ausprogrammiert wurde. Jedoch ist eine Erweiterung im nächsten Jahr jederzeit denkbar.

#### 4.4.6 Funkmodul

Wie bereits beschrieben haben die vergangenen Jahre gezeigt, dass es durchaus möglich ist, dass andere Teams das selbe Funkmodul verwenden. Um Fehler bei der Übertragung zu vermeiden, wurde die Möglichkeit geschaffen, den Funkkanal umzustellen. Dies ist in diesem Menü möglich.



Abbildung 4-18: Funkmodul – Einstellmenü

#### 4.4.7 Parameter

Um Einstellungen tätigen zu können die durch keinen oben beschriebenen Menüpunkt abgedeckt wird, wurde dieses Menü vorgesehen. Im Verlauf der Entwicklung stellte sich allerdings heraus, dass dies für die diesjährige Aufgabenstellung nicht von Bedeutung ist,

Programmierung einer HMI für einen Roboter für die EUROBOT<sup>open</sup> 2010

weshalb dieses Menü auch nicht ausprogrammiert wurde. Jedoch ist eine Erweiterung im nächsten Jahr jederzeit denkbar.

## 5 ZUSAMMENFASSUNG UND AUSBLICK

Durch die neu programmierte Mensch – Maschinen – Interaktionsschnittstelle ist es möglich, den Roboter der Fachhochschule Wels für die EUROBOT<sup>open</sup> in moderater Zeit an die verschiedensten äußeren Bedingungen anzupassen. Auf Grund des Regelwerkes der EUROBOT<sup>open</sup> muss dies in 3 Minuten geschehen, weshalb auf eine einfache und klare Menüstruktur geachtet wurde. Weiters wurde ein großes Augenmerk darauf gelegt, dass die Software übersichtlich gestaltet ist, sodass einer Verwendung im nächsten Jahr durch kleinere Abänderungen nichts im Wege stehen dürfte. Auch sind einige Menüpunkte zwar vorgesehen, aber aus Zeitgründen nicht mehr programmiert worden, weshalb auch in dieser Hinsicht sicherlich weiterentwickelt werden kann. Durch die einheitlichen SPI – Meldungen passt sich die LCD – Software sehr gut dem modularen System der FH – Wels an, was auch von Anfang an beabsichtigt war.

Trotz einiger Probleme bei der Entwicklung des Roboters sowie beim Bewerb ist die HMI jederzeit zuverlässig gelaufen, weshalb eine Weiterentwicklung durchaus sinnvoll erscheint. Denkbar wäre ein größeres LCD um noch mehr Informationen auf einen Blick zu ermöglichen. Grundsätzlich sollte aber für eine HMI immer gelten: KISS (Keep it simple and straightforward).

## 6 LITERATUR

### 6.1 Bücherliste

[Schmitt, 2007] Schmitt, Günter: Mikrocomputertechnik mit Controllern der Atmel AVR-RISC-Familie

[Tietze - Schenk, 1999] Ulrich, Tietze - Christoph, Schenk: Halbleiter – Schaltungstechnik

### 6.2 Bachelorarbeiten

[Zauner, 2009] Zauner, Michael: Bachelorarbeit 1 - Modulares und skalierbares elektronisches System für autonome Roboter

[Muckenhumer, 2010] Muckenhumer, Bernhard: Bachelorarbeit 1 - Anbindung und Adaptierung einer mikrocontrollergesteuerten Kamera an einen autonomen Roboter

[Stotter, 2010] Stotter, Bernhard: Bachelorarbeit 2 – Entwicklung der Software eines absoluten Positionsbestimmungssystemsystems für mobile autonome Roboter

### 6.3 Internet

[Rules E2010, 2009]: E2010\_rules\_and\_drawing\_EN.pdf  
Stand 24.09.2009

[eurobot, 2010] <http://www.eurobot.org/> Stand 14.03.2010

[storm-interface, 2010] <http://www.storm-interface.com/> Stand 07.06.2010

[displaytech, 2010] <http://www.displaytech.com.hk/> Stand 07.06.2010

[Sitronix, 2010] <http://www.sitronix.com.tw/> Stand 07.06.2010

[codevision, 2010] <http://www.codevision.be/> Stand 14.06.2010

[Atmel, 2010] <http://www.atmel.com/> Stand 14.06.2010