

Fachübergreifendes Projekt

Sommersemester 2011

Projekttitle

Line Follower „RoadRunner“

Verantwortliche Autoren / Matrikelnummer:



Manuel Schlager / S0910438048



Simon Infanger / S0910438038



LVA-Leiter:
Dipl. Ing. (FH) Raimund Edlinger

Studiengang:
AT09 / INIF

Abgabetermin:
-

Inhaltsverzeichnis:

1. Einleitung	4
2. Aufgabenstellung & Anforderungen	4
2.1 Line Follower Modus	4
2.2 Line Follower Enhanced Modus	5
2.2.1 Linienunterbrechung	6
2.2.2 Tunnel.....	6
2.2.3 Hindernis (Ziegelstein).....	7
3. Mechanik	7
3.1 Grundkonzept	7
3.2 Rahmen.....	7
3.2.1 Grundrahmen.....	7
3.2.2 Radaufhängung.....	9
3.2.3 Bereifung	10
3.2.4 Anmerkungen	11
3.3 Antrieb.....	11
3.3.1 Antriebsrad	12
3.3.2 Antriebsmotor.....	13
3.3.3 Getriebe	13
3.3.4 Anmerkungen	15
3.4 Lenkung.....	15
3.4.1 Lenkungsmotor.....	16
3.4.2 Anmerkungen	16
3.5 Gesamtkonzept	17
3.5.1 Catia - Gesamtkonstruktion.....	17
3.5.2 Bilder RoadRunner.....	18
4. Elektronik	19
4.1 Grundkonzept	19
4.2 Miniboard (Prozessorboard).....	19
4.2.1 Schaltplan	20
4.2.2 Bestückungsplan.....	21
4.2.3 Stückliste	21
4.2.4 Anmerkungen	22
4.3 Energieversorgung	22
4.3.1 Schaltplan	23
4.3.2 Bestückungsplan.....	24
4.3.3 Stückliste	25
4.3.4 Anmerkungen	25

4.4 Liniensensoren (Linienboard)	25
4.4.1 Schaltpläne	27
4.4.2 Bestückungspläne	29
4.4.3 Stückliste	29
4.4.4 Platinen	30
4.4.5 Anmerkungen	31
4.5 Abstandssensoren	31
4.5.1 Schaltplan	32
4.5.2 Bestückungsplan	32
4.5.3 Stückliste	33
4.5.4 Anmerkungen	33
5. Software	34
5.1 Line Follower- & Line Follower Enhanced Modus	34
5.1.1 Lenkungssteuerung	34
5.1.2 Antriebsansteuerung	36
5.1.3 Anmerkungen	38
5.2 Hindernisse im Enhanced Modus	38
5.2.1 Anmerkungen zur Linienunterbrechung	38
5.2.2 Anmerkungen zum Hindernis (Ziegelstein)	38
5.2.3 Anmerkungen zum Hindernis (Tunnel)	40
6. Gegneranalyse & Verbesserungsvorschläge	40
7. Quellenverzeichnis	40

1. Einleitung

Ziel des Projektes ist die Entwicklung eines Roboters (Abbildung 1.1), der selbstständig in der Lage ist, eine Linie zu verfolgen. Zusätzlich zu dieser Funktion soll es möglich sein, diverse Hindernisse mittels geeigneter Sensorik zu bewältigen. Da dieser Roboter im späteren Verlauf bei der RobotChallenge 2011 antreten soll, ist ein weiterer wichtiger Aspekt, dass dieser seine zu erfüllenden Aufgaben so schnell und zuverlässig wie möglich durchführt, um eine dementsprechende Leistung zu erbringen, um somit konkurrenzfähig zu sein.



Abb. 1.1: Line Follower „RoadRunner“

2. Aufgabenstellung & Anforderungen

2.1 Line Follower Modus

Der Line Follower muss grundsätzlich, wie bereits in der Einleitung angesprochen, möglichst schnell einer schwarzen Linie auf einem weißen Untergrund folgen (Abbildung 2.1.1).

Dazu muss es dem Roboter möglich sein, eine Abweichung von der optimalen Linie zu erkennen. Dafür ist der Einsatz einer geeigneten Sensorik (Sensorleiste mit 5 Reflexlichtschranken, welche an der Unterseite des Roboters angebracht ist) notwendig. Dadurch ist es möglich den Verlauf der Strecke zu detektieren und bei einer Änderung vom optimalen Kurs dementsprechend entgegen zu wirken.

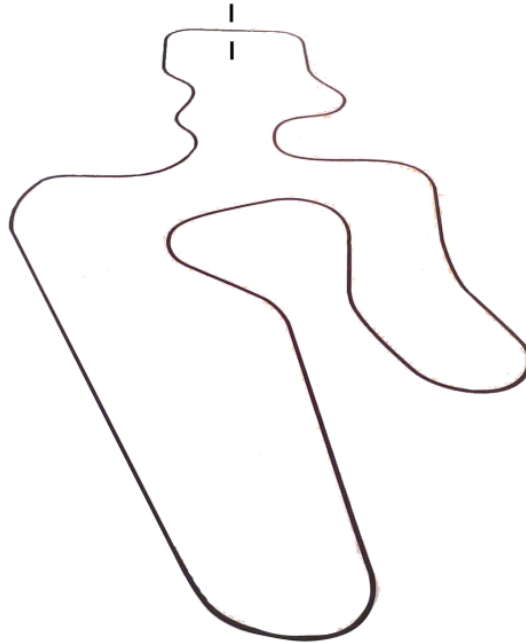


Abb. 2.1: Line Follower Strecke

Abbildung 2.1.1 zeigt die Line Follower Strecke der RobotChallenge 2011. Die Rennstrecke wird grundsätzlich immer erst am Tag des Wettbewerbes bekannt gegeben. Dies soll Chancengleichheit für alle Teams schaffen.

Grundsätzlich ist bekannt, dass die schwarze Linie eine Breite von ca. 1,5cm aufweist und daher für Testzwecke ein normales schwarzes Isolierband für eine eigene Strecke verwendet werden kann.

Im Gegensatz zum Vorjahr, wo die Streckenabschnitte auf 50 x 50 cm großen weißen Platten aufgeklebt waren, ist in diesem Jahr die gesamte Rennstrecke auf eine weiße Kunststoffplane aufgedruckt gewesen.

Neben der Strecke gibt es eine Sensorik (Lichtschranke) die den Start und das Ziel darstellt, durch die der Roboter durchfahren muss. Der Abstand zwischen dieser Sensorik beträgt 50 cm. Außerdem darf der Line Follower eine Länge von 30 cm und eine Breite von 50 cm (aufgrund der Start-Ziel Sensorik) nicht überschreiten.

2.2 Line Follower Enhanced Modus

In diesem Wettbewerb wird die Line Follower Strecke (Abbildung 2.1) um diverse Schwierigkeiten und zwar eine Linienunterbrechung, einen Tunnel und ein Hindernis (Ziegelstein) erweitert, zu sehen in Abbildung 2.2.

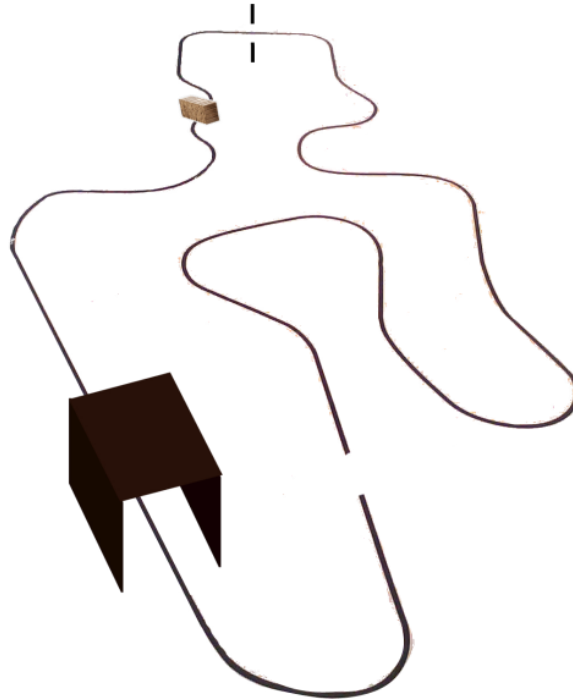


Abb. 2.2: Line Follower Enhanced Strecke

Aufgrund dieser Problematik ergeben sich neue Anforderungen an die Sensorik des Roboters bezüglich der Erkennung der Hindernisse (3 Abstandssensoren: 1x 10cm und 2x 40cm), um diese schnell und zuverlässig überwinden zu können.

Die tatsächlichen Positionen der Hindernisse in der Strecke sind, wie bereits erwähnt wurde, erst am Tag des Wettbewerbs bekannt geworden.

2.2.1 Linienunterbrechung

Die Unterbrechung der Linie, welche eine Länge von 10 cm aufweist, befindet sich laut den Line Follower Rules nicht in einer Kurve, sondern wie in Abbildung 2.2 zu sehen ist, auf einem geraden Abschnitt.

2.2.2 Tunnel

Die Linie führt außerdem durch einen Tunnel, der eine Länge und Breite von 50 cm aufweist (Abbildung 2.2). Der Sinn hinter diesem Hindernis besteht darin, dass der Roboter mit wechselnden Lichtverhältnissen zu Recht kommt.

2.2.3 Hindernis (Ziegelstein)

Die Linie wird außerdem durch ein weiteres Hindernis erweitert, welches einem Ziegelstein von der Dimension 25 x 12 x 6,5cm entspricht (Abbildung 2.2).

Dieser soll erkannt werden und mit dementsprechendem Ausweichmanöver umfahren werden. Nach dem Umfahren muss der Roboter wieder autonom auf die Linie zurück finden. Der Ziegel darf sowohl links als auch rechts umfahren werden. Der Line Follower darf allerdings nicht durch bzw. über den Ziegelstein fahren.

3. Mechanik

Das folgende Kapitel befasst sich mit der Entwicklung und dem Aufbau der einzelnen mechanischen Komponenten des Roboters. Diese Komponenten unterteilen sich grundsätzlich in den Rahmen, den Antrieb und die Lenkung. Eventuelle Anmerkungen zu den einzelnen Komponenten werden in den jeweiligen Unterkapiteln angeführt.

3.1 Grundkonzept

Nach der Konzeptionierungsphase stand fest, dass wir lediglich die Antriebseinheit und die Lenkung des Vorjahreskonzeptes „Arrow“ übernehmen werden. Die Lenkung sollte vorerst mit einem Servo verwirklicht werden. Diese Überlegung musste letztlich wegen zu langer Reaktionszeit des Servos verworfen werden. Da aus Beobachtungen der Videos von der RobotChallenge 2010 hervorging, dass das Konzept „Arrow“ mit Unter- und Übersteuern zu kämpfen hatte, lag das Hauptaugenmerk bei der Konstruktion auf Grippverbesserung und Gewichtseinsparung.

3.2 Rahmen

3.2.1 Grundrahmen

Bei der Konstruktion des Rahmens wurde darauf geachtet, dass bei geringstem Gewichtsaufwand höchstmögliche Flexibilität erreicht wird. Ein Hauptziel ist, dass die Lenkeinheit möglichst einfach und ohne viel Aufwand in der Position variabel ist.

Für die Stabilität sorgen unter anderem hauptsächlich zwei Aluminiumträger mit einer Materialstärke von 5mm.

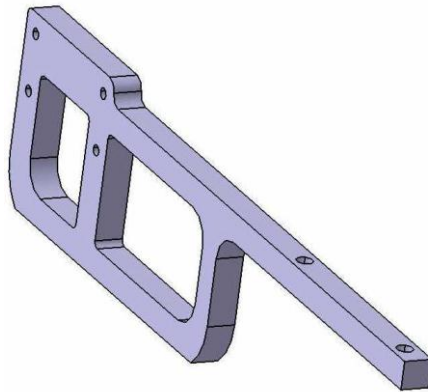


Abb. 3.1: Aluminiumträger

Hier wurde darauf geachtet, dass die Konstruktion möglichst leicht und kompakt bleibt. Unnötige Materialansammlungen sind, wo es möglich war, vermieden worden. Ein weiterer Punkt bei der Planung der beiden Seitenteile war die Minimierung der Bodenfreiheit. Um dies realisieren zu können, mussten die Abmessungen der weiteren Anbauteile wie Reifen, deren Aufhängung und Antriebseinheit bekannt sein. Die Verbindungen der Aluminiumträger wurden mit Karbonplatten realisiert. Diese haben den Vorteil extrem leicht und zugleich stabil zu sein.



Abb. 3.2: Bodenplatte aus Karbon

Da bekannter Weise der Akku eines der schwersten Bestandteile ist, musste er natürlich so tief wie möglich platziert werden. Aus diesem Grund wurde die Bodenplatte zugleich als Akkuablage verwendet. Über der Bodenplatte werden die Aluminiumseitenteile mit einer weiteren Karbonplatte zusammengehalten. Diese dient zugleich als Befestigungsebene der Elektronik.

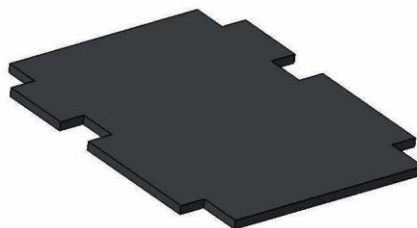


Abb. 3.3: Platinenbefestigungsplatte aus Karbon

An der Vorderseite des Rahmens wurde ebenfalls eine Karbonkonstruktion verwendet. Diese verwirklicht zugleich die Verbindung mit dem Lenkmotor und der Antriebseinheit.



Abb. 3.4: Lenkmotorhalterung aus Karbon

Die oben beschriebenen Einzelteile bilden zusammengefügt den Grundrahmen, an dem alle weiteren Bauteile befestigt werden können.

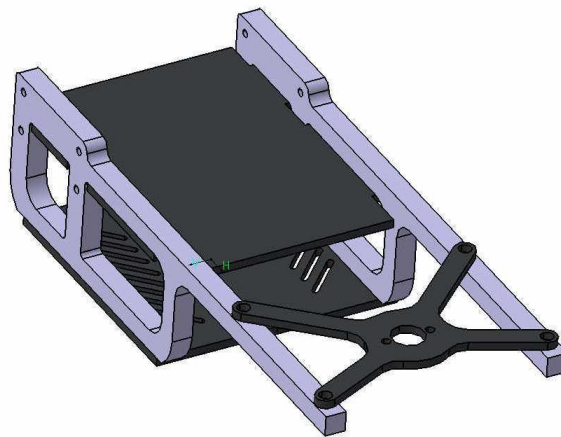


Abb. 3.5: Grundrahmen

Auf den zusammengesetzten Grundrahmen wurde weiterführend der Heckspoiler mit den integrierten Tippschaltern und dem Ein/Aus- Schalter befestigt.

3.2.2 Radaufhängung

Nach dem die Bereifungsart und vor allem die Dimension fest stand, konnte mit der Konstruktion der Radaufhängung begonnen werden. Da wir Reifen aus dem Modelbaubereich verwendet haben, musste die dort verwendete Radbefestigung und deren Lagerung in unsere Radaufhängungskonstruktion mit einbezogen werden.

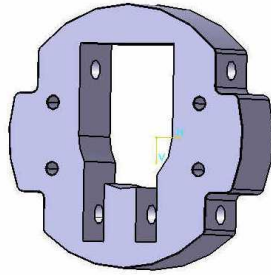


Abb. 3.6: Alu Radaufhängung

Um eine feste Verbindung mit der vorhandenen Modellbauradlagerung gewährleisten zu können, wurde diese in eine 1cm dicke Aluscheibe eingepasst. Da die Passung sehr exakt gemacht wurde, wurde die Radlagerung in den Alu-Block eingepresst. Durch das Einpressen wurde genügend Zusammenhalt erreicht und es konnte auf weitere Verschraubungen verzichtet werden.

Um den nötigen Platz für die Reifen zum Rahmen hin zu schaffen, wurde die Aufhängung mit PVC-Hülsen distanziert. Diese können natürlich durch längere Hülsen ersetzt werden, um eine Spurverbreiterung zu erreichen.

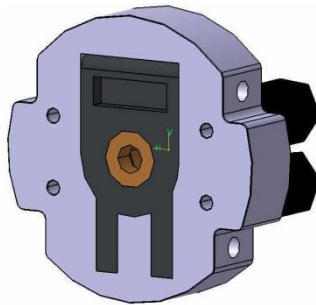


Abb. 3.7: Radaufhängung komplett

3.2.3 Bereifung

Die Bereifung ist einer der größten Schwachpunkte des Linienverfolgers Arrow gewesen. Unsere grundsätzliche Überlegung bei der Radauswahl fokussierte sich auf einen etwas größeren Raddurchmesser, da mit steigendem Durchmesser der Rollwiderstand sinkt. Zusätzlich sollte die Lauffläche aus einem Material bestehen, das auf glatten Oberflächen optimalen Grip bietet. Aus diesem Grund kamen als Lauffläche nur noch 2 Materialien (Moosgummi od. Silikon) in die engere Auswahl. Wir haben uns letztlich auf Reifen aus dem Modellbaubereich aus Moosgummi geeinigt. Die Wahl ist auf den Indore Moosgummireifen VTEC Heck 35 1/10 gefallen.



Abb. 3.8: Moosgummiräder

Moosgummi hat die Eigenschaft, dass mit steigendem Laufflächendruck die Haftung steigt. Ein weiterer Weg den Gripp zu erhöhen, wäre die Breite der Reifen zu reduzieren. Dies hätte ebenso mehr Druck pro Fläche zur Folge.

3.2.4 Anmerkungen

Der Rahmen erfüllte seine Anforderungen grundsätzlich überaus zufriedenstellend. Ein kleiner Verbesserungsvorschlag wäre jedoch eine Spurverbreiterung. Im Rennbetrieb stellte sich heraus, dass aufgrund der extrem gut haftenden Moosgummireifen, der Linienverfolger zum Überschlagen neigt. Durch eine breitere Spur würde sich das innere Rad nicht mehr vom Boden abheben.

Ein weiter Nachteil ist die schnelle Verschmutzung der Moosgummireifen. Die Reifen sind jedoch relativ simpel mit einem Flusen Roller oder mit einem Isolierband zu säubern.

3.3 Antrieb

Bei der Auslegung des Antriebes waren folgende Punkte zu beachten:

- Kompakte Bauweise, um das Trägheitsmoment für den Lenkmotor möglichst gering zu halten
- Auswahl eines passenden Antriebsmotors
- Wahl einer passenden Getriebeübersetzung
- Antriebswalze mit möglichst viel Gripp
- Lagerung der Antriebswalze

In unserem Fall wurde die Antriebseinheit des Vorjahreskonzeptes „Arrow“ übernommen, da diese bereits sämtliche oben angeführten Anforderungen erfüllte. Es wurden lediglich einige kleine Modifikationen durchgeführt.

3.3.1 Antriebsrad

Der Antrieb des Linienverfolgers „Arrow“ verwendete als Antriebsrad eine Paper Pickup Roller von HP (Bezeichnung: HP RM1-0037-020). Diese Papiereinzugsrolle wird in den Druckermodellen LJ 4000 4050 4500 verwendet.



Abb. 3.9: Paper Pickup Roller

Auf Grund der fortgeschrittenen Abnutzung der alten Druckerwalze mussten wir diese ersetzen. Ein weiterer Punkt auf den wir bei der Suche nach einer neuen Antriebswalze achteten, war die Festigkeit. Die bisherige Walze hatte einen etwas zu weichen Aufbau, der wie wir herausfanden, in Kurvenlagen nicht optimal war.

Als neue Antriebswalze wurde ein Paper Pickup Roller HP (Bezeichnung: RF5-3338-000CN) von den Druckermodellen 9000 9040 9050 MFP verwendet. Diese ist durch den Vollgummiaufbau härter und hat zusätzlich eine leichte Profilierung.

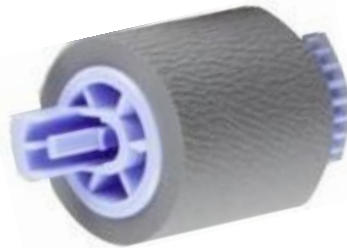


Abb. 3.10: Paper Pickup Roller

Die Wahl diese Vollgummiwalze zu verwenden, verringerte das untersteuernde Verhalten drastisch. Als Trägerkörper wurde anstatt des Plastikinnenteils eine Aluminiumwelle verwendet.

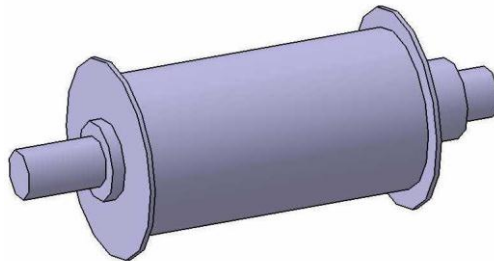


Abb. 3.11: Antriebswelle aus Alu

3.3.2 Antriebsmotor

Beim Motor fiel die Wahl auf einen Faulhaber Getriebemotor mit der Bezeichnung: 1717006SR. Als Untersetzungsgetriebe wurde ein Faulhabergetriebe mit der Bezeichnung 16/7 und einer Übersetzung von 3,71:1 verwendet. Um die Forderung für einen kompakten Antrieb zu verwirklichen, wurde der Antriebsmotor direkt hinter der Antriebsrolle platziert (Abbildung 3.14: Getriebe).

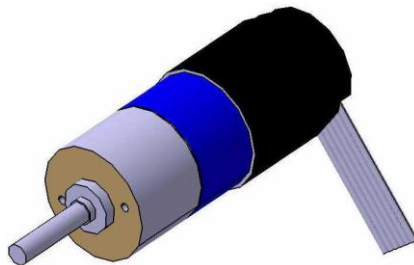


Abb. 3.12: Antriebsmotor

3.3.3 Getriebe

Um die Ausgangsdrehzahl des Antriebsmotors auf das gewünschte Maß zu reduzieren, wurde ein zweistufiges Übersetzungsverhältnis realisiert. Auf Grund der begrenzten Platzverhältnisse mussten relativ kleine Zahnräder mit einem Modul von 0,5 verwendet werden. Das mittlere Zahnrad wurde mit Hilfe eines Langloches in der seitlichen Karbonplatte in die gewünschte Position gebracht.

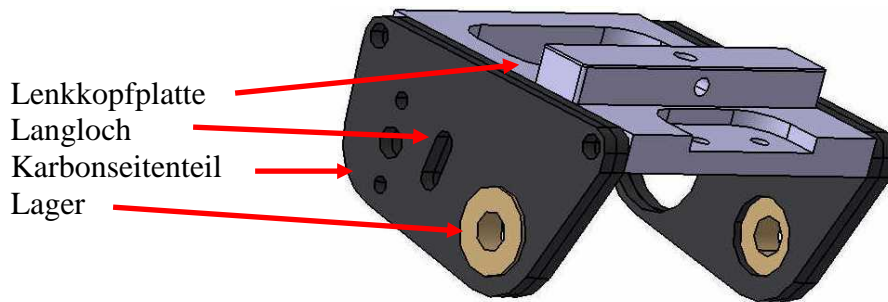


Abb. 3.13: Frontantrieb Grundrahmen mit Lagern

Durch das Langloch kann auch die Zahnflankenspeisung variiert werden, dies ist wichtig um die Leichtgängigkeit des Getriebes zu gewährleisten.

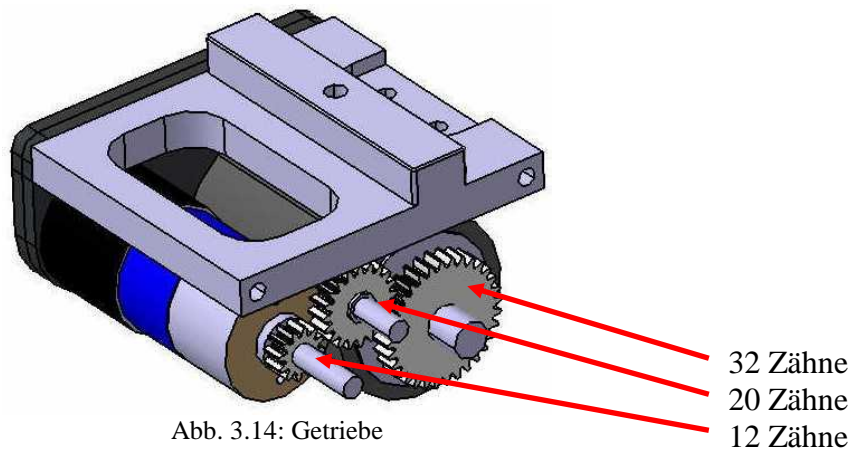


Abb. 3.14: Getriebe

An der Motorwelle wurde ein Zahnrad mit 12 Zähnen verbaut, an der Antriebswelle ein Zahnrad mit 32 Zähnen und als Zwischenrad kam ein 20 Zähne Zahnrad zum Einsatz. Als Gesamtuntersetzung resultiert daraus:

$$i_{\text{ges}} = 32 / 12 = 2,67$$

Nimmt man nun als maximale Drehzahl die angegebene nominale Drehzahl von 14000U/min vom Antriebsmotor an, so erhält man eine Drehzahl auf der Getriebewelle von:

$$n_{\text{Getriebe}} = 14000 / 3,71 = 3774\text{U/min}$$

Um die maximale Drehzahl der Antriebsrolle zu ermitteln, muss nur noch die Untersetzung des zweistufigen Getriebes mit eingerechnet werden:

$$n_{\text{Antrieb}} = 3774 / 2,67 = 1415\text{U/min}$$

Somit erhält man eine maximal mögliche Fahrgeschwindigkeit von 3,55m/s was ca. 13km/h entspricht.

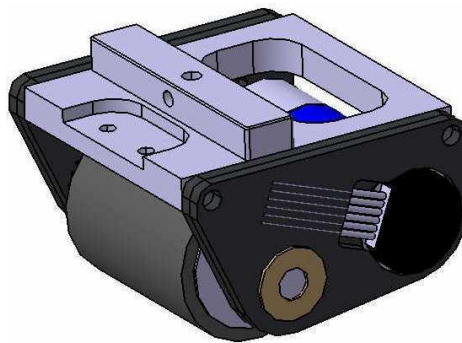


Abb. 3.15: Frontantrieb

3.3.4 Anmerkungen

Der Lenkkopf als Gesamtes erwies sich als sehr ausgereifte Konstruktion. Der einzige Optimierungspunkt in dieser Einheit wäre die Implementierung eines stärkeren Motors um besser beschleunigen zu können.

3.4 Lenkung

In der ursprünglichen Version wollten wir die Lenkung mit einem Servo aufbauen. Dies hätte für uns einige Vorteile in Bezug auf die Lenkungsansteuerung gehabt. Ein Beispiel für solch einen Vorteil ist, dass ein Servo seinen vorgegebenen Ausschlagwinkel exakt beibehält. Ein Motor hingegen kann auf Grund einer Massenträgheit überdrehen.

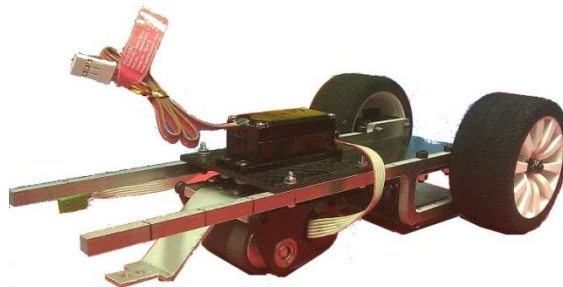


Abb. 3.16: Servo - Lenkeinheit

Leider stellte sich bei den ersten Testläufen heraus, dass ein Servo eine bei weitem zu lange Reaktionszeit zwischen Ansteuerung und Ausführung hat. Bei moderaten Geschwindigkeiten

funktionierte das System einwandfrei. Wurde jedoch die Geschwindigkeit gesteigert, begannen sich die Lenkbewegungen auf zu schaukeln. Grund dafür ist eine Reaktionszeit von ca. 20ms bei Standard-Servos. Aus diesem Grund musste wieder auf das vorhandene Lenksystem des Linienverfolgers „Arrow“ umgesattelt werden.

3.4.1 Lenkungsmotor

Beim Motor fiel die Wahl auf einen Faulhaber Getriebemotor mit der Bezeichnung: 1717006SR. Als Untersetzungsgetriebe wurde ein Faulhabergetriebe mit der Bezeichnung 16/7 und einer Untersetzung von 3,71:1 verwendet.

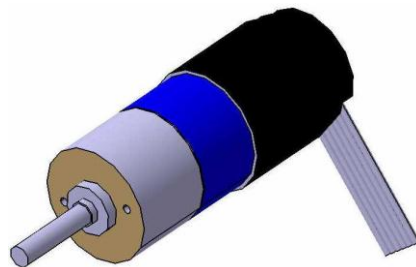


Abb. 3.17: Lenkmotor

Der Lenkmotor wurde, wie in der nachfolgenden Abbildung zu erkennen ist mit dem Grundrahmen (Abbildung 3.2.1.5: Grundrahmen) verschraubt. Die Lenkeinheit ist durch eine Bohrung und einer Fixierschraube mit der Motorwelle verbunden.

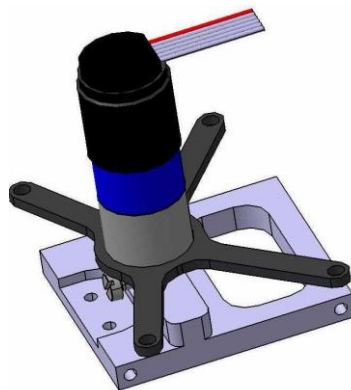


Abb. 3.18: Lenkeinheit

3.4.2 Anmerkungen

Um eine möglichst exaktes Lenkverhalten zu erreichen ist es wichtig, dass das Massenträgheitsmoment so gering wie möglich gehalten wird. Je geringere die Trägheit um

die Lenkachse ist, umso weniger schwingt sich die Lenkeinheit auf. In diesem Fall ist auf jedes unnötige Gramm im Bereich des Sensorboards zu verzichten.

3.5 Gesamtkonzept

Der folgende Abschnitt zeigt die Catia-Konstruktionen und Fotos des gesamten Roboters in verschiedensten Ansichten. Sämtliche Maße und Dimensionierungen sind in den beiliegenden Catia-Files ersichtlich.

3.5.1 Catia - Gesamtkonstruktion

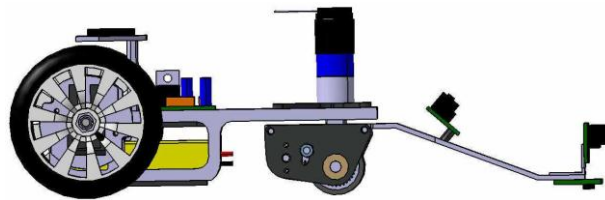


Abb. 3.19: Catia - Gesamtkonstruktion Seite

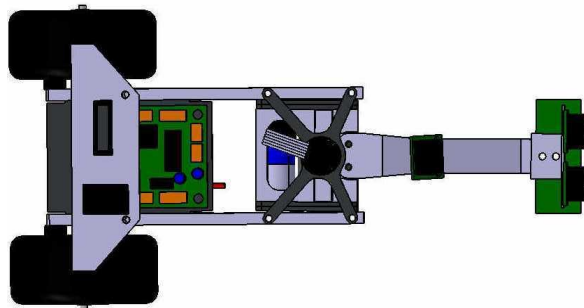


Abb. 3.20: Catia - Gesamtkonstruktion Draufsicht

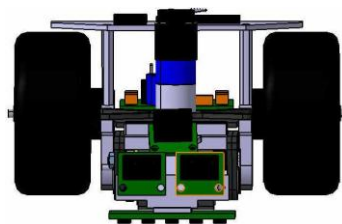


Abb. 3.21: Catia - Gesamtkonstruktion Front

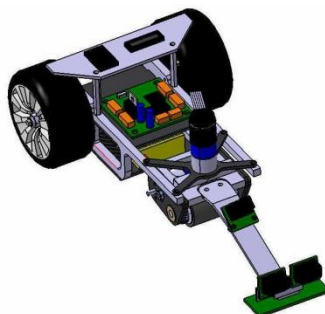


Abb. 3.22: Catia - Gesamtkonstruktion diagonal

3.5.2 Bilder RoadRunner

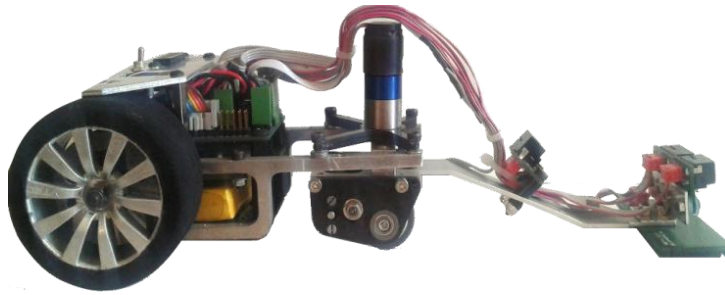


Abb. 3.23: RoadRunner Seite

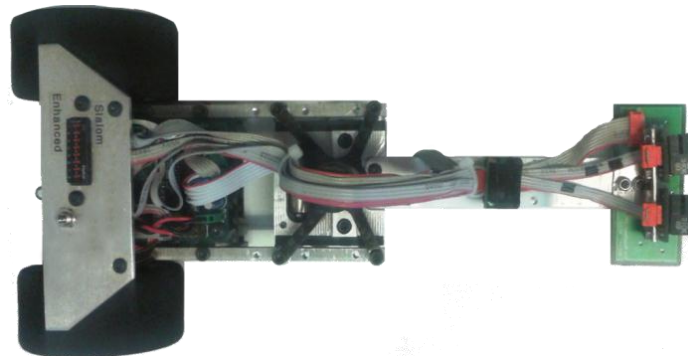


Abb. 3.24: RoadRunner Draufsicht

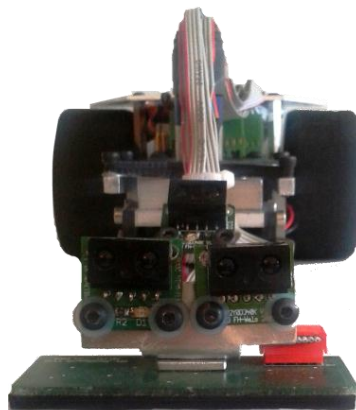


Abb. 3.25: RoadRunner Front



Abb. 3.26: RoadRunner diagonal

4. Elektronik

Das folgende Kapitel befasst sich mit der Entwicklung der Schaltpläne und den dazugehörigen Platinen der einzelnen elektronischen Komponenten des Roboters. Diese Komponenten unterteilen sich grundsätzlich in das Miniboard, die Spannungsversorgung und die Sensorik. Eventuelle Anmerkungen zu den einzelnen Komponenten werden in den jeweiligen Unterkapiteln angeführt.

Alle Schaltpläne und die dazugehörigen Platinen wurden mit dem Programm Target3001 erstellt.

4.1 Grundkonzept

Bereits während der Entwicklung des mechanischen Grundkonzeptes wurden erste Überlegungen bezüglich des elektronischen Grundkonzeptes getroffen.

Aufgrund der Übersichtlichkeit und der Erweiterbarkeit eines modularen Aufbaus sollte die Entwicklung vor allem in diese Richtung gehen. Daher wurde neben der entwickelten Spannungsversorgung und der Sensorik (Liniensensoren, Abstandssensoren) das modular aufgebaute Miniboard (Prozessorboard) von Herrn Ing. Zauner Michael BSc des Robo Racing Teams der FH-Wels verwendet.

Neben der Sensorik für die Linienerkennung und der Entscheidung anstelle eines 3Sensor- ein 5Sensor-Linienboard zu verwenden, wurden für den Line Follower Enhanced Modus noch 3 weitere Abstandssensoren angebracht. Diese schaffen die Möglichkeit dem Hindernis (Ziegelstein) ausweichen zu können.

4.2 Miniboard (Prozessorboard)

Die Gründe für die Verwendung des Miniboards neben der kompakten Aufbauweise, welche sich mittlerweile mehrfach bewiesen hat liegen vor allem in der Charakteristik des Boards:

- 8-bit Microcontroller der Serie ATmega644-20AU mit 64k Bytes
- ISP-Programmierinterface mit USART
- 4 ADC-Kontakte
- Alle frei zur Verfügung stehenden Ports auf Anschlüsse geführt
- Separater Anschluss für eine Spannungsversorgung
- Leuchtdiode für diverse Statusanzeigen

4.2.1 Schaltplan

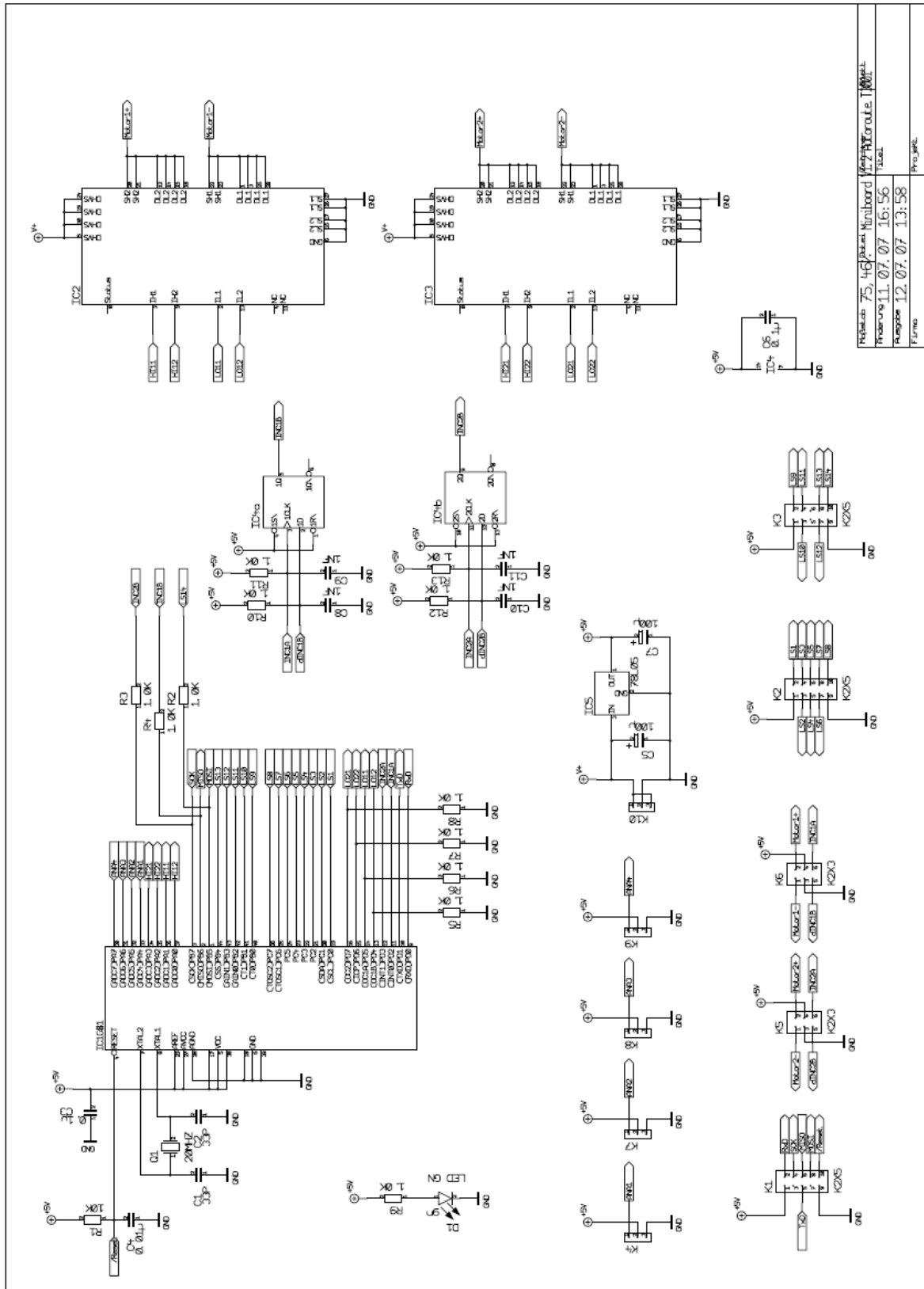


Abb. 4.1: Miniboard V1.2 - Schaltplan

4.2.2 Bestückungsplan

In den folgenden Abbildungen sind die Top- und Bottom-View, sowie eine 3D-Ansicht des Miniboards in der Version 1.2 abgebildet.

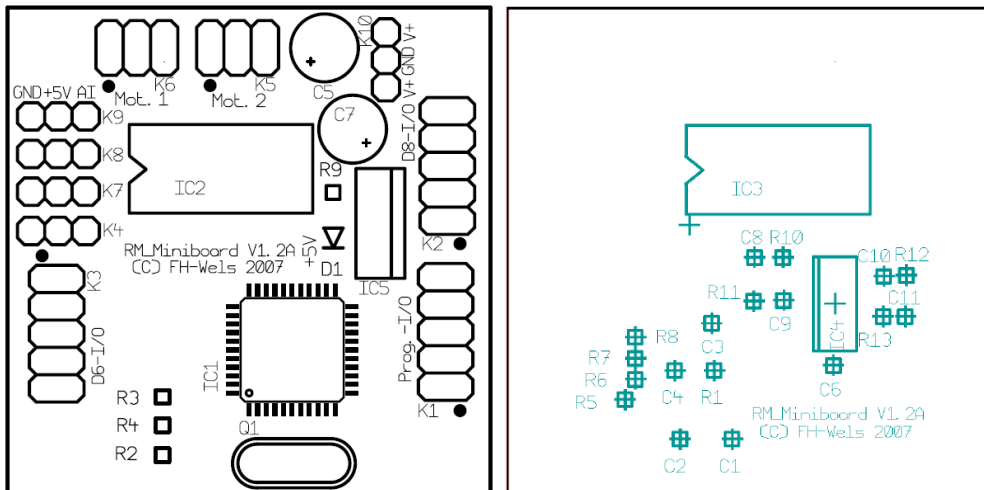


Abb. 4.2: Miniboard V1.2 - Bestückungsplan - Top & Bottom

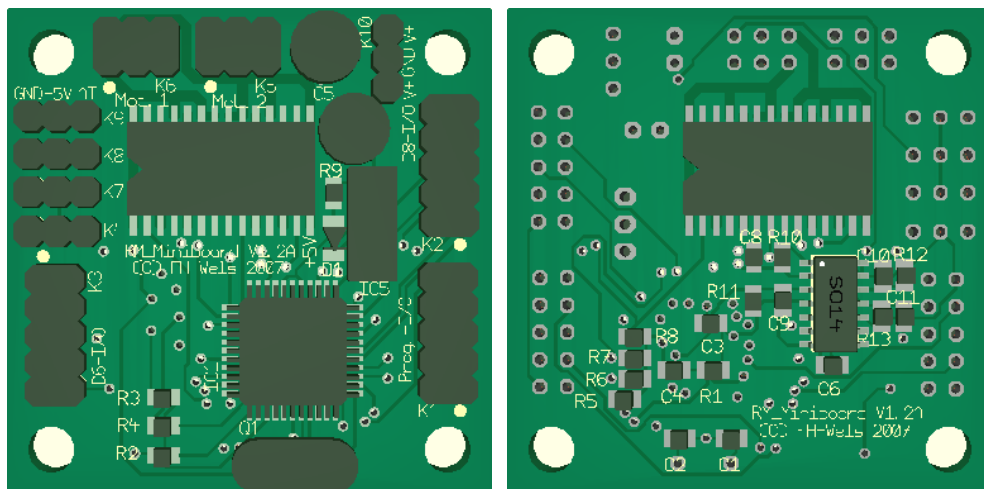


Abb. 4.3: Miniboard V1.2 - Bestückungsplan - 3D-Ansicht - Top & Bottom

4.2.3 Stückliste

In der folgenden Tabelle ist die Stückliste, welche für die Bestückung des Miniboards in der Version 1.2 verwendet wurde, abgebildet.

Name	Bauteilart	Wert	Gehäuse	Anzahl	Top	Bottom	Art
C1, C2	Keramikkondensator	33pF/50V	0805	2	0	2	SMD
C3, C6	Keramikkondensator	100nF/50V	0805	2	0	2	SMD

C4	Keramikkondensator	10nF/50V	0805	1	0	1	SMD
C8-C11	Keramikkondensator	1n5/50V	0805	4	0	4	SMD
C5, C7	Elko	100µF/16V	2,5-5	2	2	0	bedrahtet
D1	LED	grün	1206	1	1	0	SMD
IC1	Microcontroller	ATmega644-20AU	TQFP44	1	1	0	SMD
IC2, IC3	Leistungstreiber	BTS 7740 G	SO28	2	1	1	SMD
IC4	D-Flip-Flop	74AHCT74D	SO14	1	0	1	SMD
IC5	Spannungsregler	MC7805CT	TO-220	1	1	0	bedrahtet
K1-K3	Stiftleiste	2x5	-	3	3	0	bedrahtet
K4, K7-K10	Stiftleiste	1x3	-	5	5	0	bedrahtet
K5, K6	Stiftleiste	2x3	-	2	2	0	bedrahtet
Q1	Quarz	20MHz	HC-18U	1	1	0	bedrahtet
R1	Widerstand	10k	0805	1	0	1	SMD
R2-R7	Widerstand	1k	0805	12	4	8	SMD

Tab. 4.1: Miniboard V1.2 - Stückliste

4.2.4 Anmerkungen

Anstelle der Stiftleisten für die Komponenten K1-K10 auf dem Miniboard wären besser MOLEX-Stecker geeignet, da dadurch nicht nur die Einsteckproblematik, sondern auch die Möglichkeit des unbeabsichtigten Absteckens der Stiftleisten besteht.

Weiters wurde anstelle des Spannungsreglers vom Typ MC7805CT ein Schaltnetzteil mit 5V und 500mA eingebaut. Auf diese Komponente wird im Kapitel 4.3 Energieversorgung näher eingegangen.

4.3 Energieversorgung

Die Energieversorgung setzt sich aus einem im Modellbau verwendeten Lithium Polymer Akku aus 3 Zellen mit 11,1V und einem Schaltnetzteil von 5V und 500mA (Abbildungen 4.4 und 4.5) anstelle des Spannungsregler vom Typ MC7805CT zusammen. Um die Energieversorgung über einen längeren Zeitraum gewährleisten zu können, wurde ein Lithium Polymer Akku mit einer Kapazität von 620mAh eingebaut.

Diese beiden Komponenten stellen die benötigte Versorgungsspannung von mindestens 5V für das Miniboard zur Verfügung.

4.3.1 Schaltplan

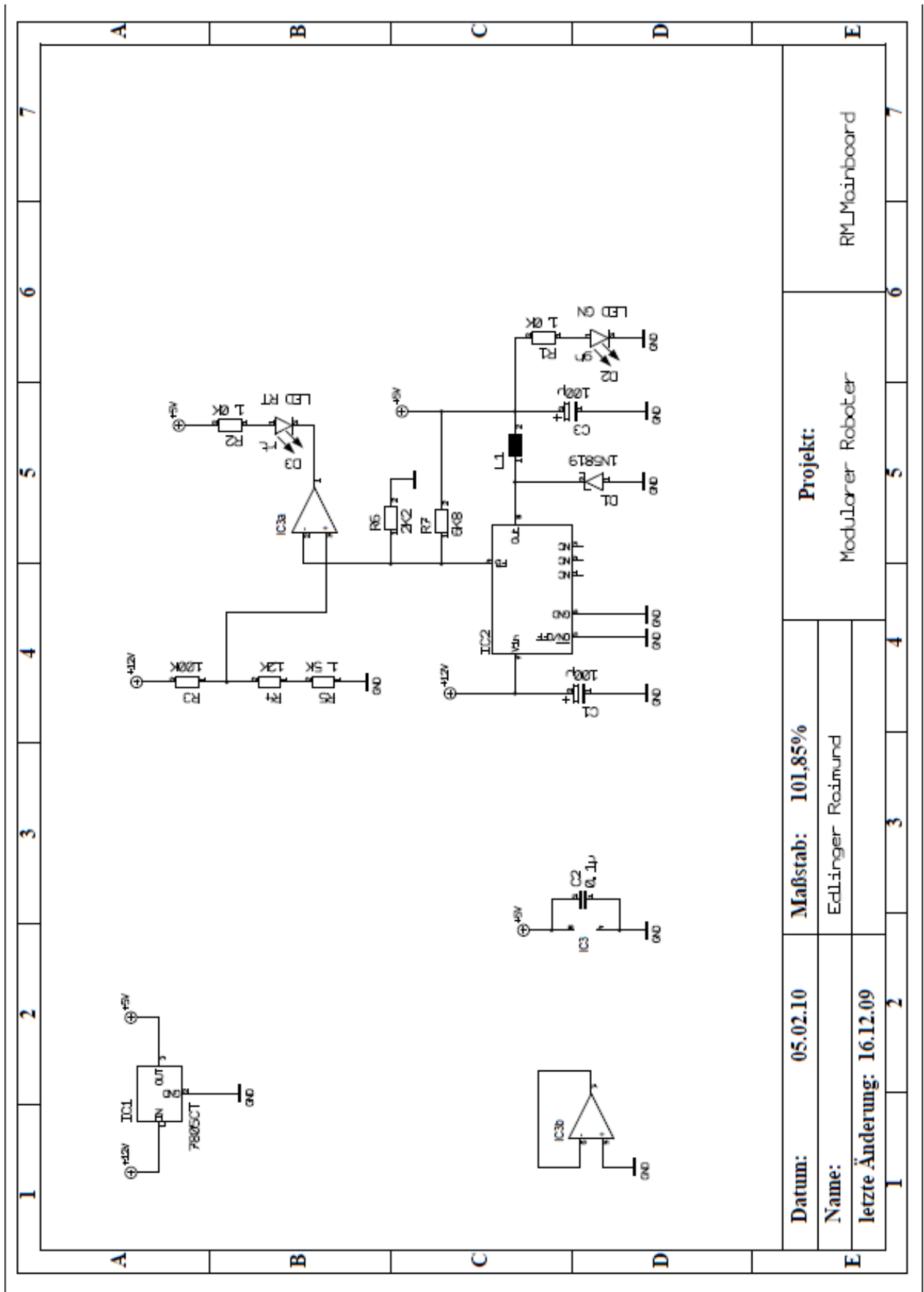


Abb. 4.4: Schaltnetzteil 5V 500mA - Schaltplan

4.3.2 Bestückungsplan

In den folgenden Abbildungen sind der Top-View und die 3D-Ansicht des Schaltnetzteils mit 5V und 500mA abgebildet.

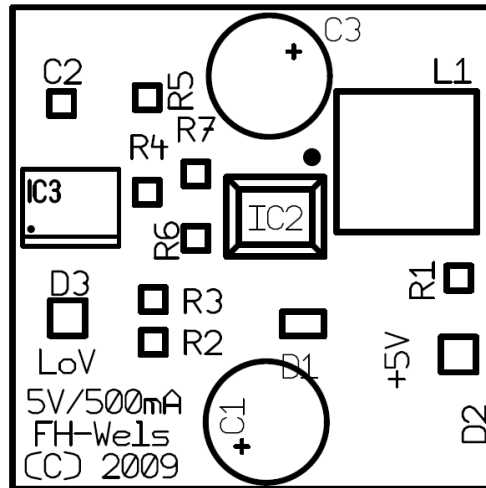


Abb. 4.5: Schaltnetzteil 5V 500mA - Bestückungsplan - Top

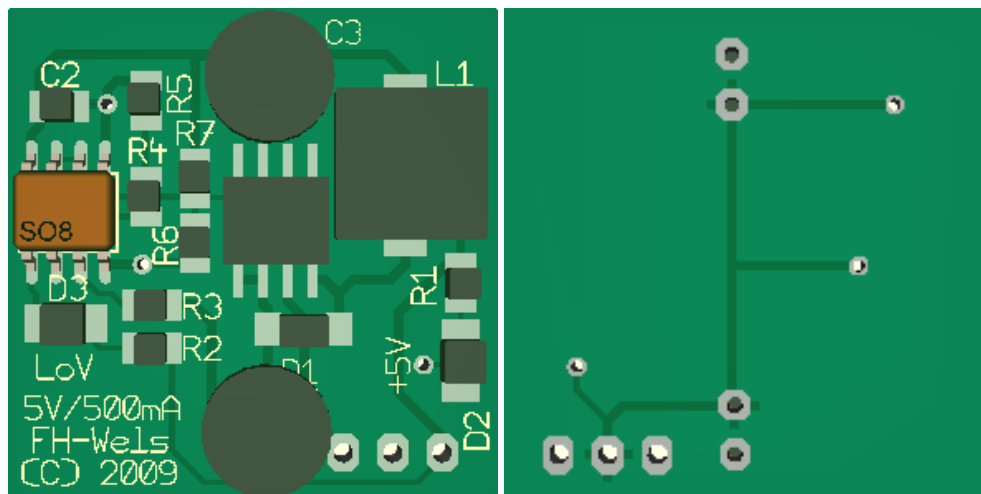


Abb. 4.6: Schaltnetzteil 5V 500mA - Bestückungsplan - 3D-Ansicht - Top & Bottom

4.3.3 Stückliste

In der folgenden Tabelle ist die Stückliste, welche für die Bestückung des Schaltnetzteils mit 5V und 500mA verwendet wurde, abgebildet.

Name	Bauteilart	Wert	Gehäuse	Anzahl	Top	Bottom	Art
C2	Keramikkondensator	100nF/50V	0805	1	1	0	SMD
C1, C3	Elko	100µF/16V	2,5-5	2	2	0	bedrahtet
L1	Spule	100µH	1028	1	1	0	SMD
D1	Schottky-Diode	1N5819	DO-214	1	1	0	SMD
D2	LED	grün	1206	1	1	0	SMD
D3	LED	rot	1206	1	1	0	SMD
IC1	Spannungsregler	MC7805CT	TO-220	1	1	0	bedrahtet
IC2	Spannungswandler	LM2594HV	DIL8	1	1	0	SMD
IC3	Universalverstärker	TS922ID	SO8	1	1	0	SMD
R1, R2	Widerstand	1k	0805	2	2	0	SMD
R3	Widerstand	100k	0805	1	1	0	SMD
R4	Widerstand	12k	0805	1	1	0	SMD
R5	Widerstand	1k5	0805	1	1	0	SMD
R6	Widerstand	2k2	0805	1	1	0	SMD
R7	Widerstand	6k8	0805	1	1	0	SMD

Tab. 4.2: Schaltnetzteil 5V 500mA - Stückliste

4.3.4 Anmerkungen

Das Schaltnetzteil wurde ebenfalls wieder als modulare Komponente entwickelt, was den Einsatz in verschiedenen Elektronikschaltungen ermöglicht. Bei dieser Komponente haben sich keine nennenswerten Schwierigkeiten ergeben.

4.4 Liniensensoren (Linienboard)

Für die Linienerkennung wurden zu Testzwecken jeweils ein Liniensensorboard mit 3 Sensoren (Abbildung 4.11) und ein weiteres mit 5 Sensoren (Abbildung 4.12) verwendet.

Durch diverse Versuche mit verschiedenen Armlängen und dem 3- und 5-sensorigen Linienboard ergab sich, dass es vor allem im Line Follower Enhanced Modus vorteilhaft ist, das Linienboard mit 5 Sensoren zu verwenden. Dies ist darauf zurückzuführen, dass

besonders bei der Linienunterbrechung das 5-sensorige Linienboard bessere Ergebnisse erzielt hatte.

Beim Aufbau der 5-sensorigen Platine ist darauf zu achten, dass der Abstand zwischen den Sensoren 12,7mm beträgt, wodurch sichergestellt wird, dass normalerweise immer mindestens ein Sensor ein Signal liefert. Weiters wurde der mittlere Sensor um 5,2mm nach vorne versetzt. Dadurch wird erreicht, dass im schlimmsten Fall also bei einem Knick von 90° dieser kein Signal mehr liefert und unter der Verwendung der Signale der anderen Sensoren schneller auf die Ausnahmesituation eingegangen werden kann.

Als Liniensensoren wurden die Infrarot-Reflexlichtschranken SFH 9201 der Firma Siemens (Abbildung 4.7) verwendet. Beschaltet werden diese so, dass die Signale (High- bzw. Low-Pegel) der Sensoren direkt am Miniboard angeschlossen werden können (Abbildung 4.8).

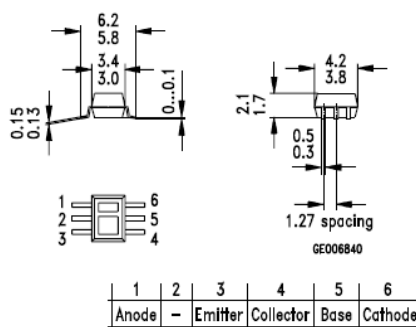


Abbildung 4.7: Liniensensor SFH 9201

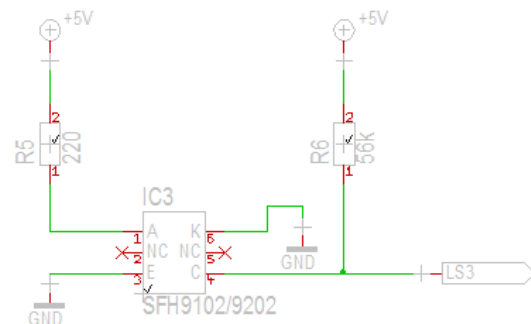


Abb. 4.8: Liniensensor SFH 9201- Beschaltung

4.4.1 Schaltpläne

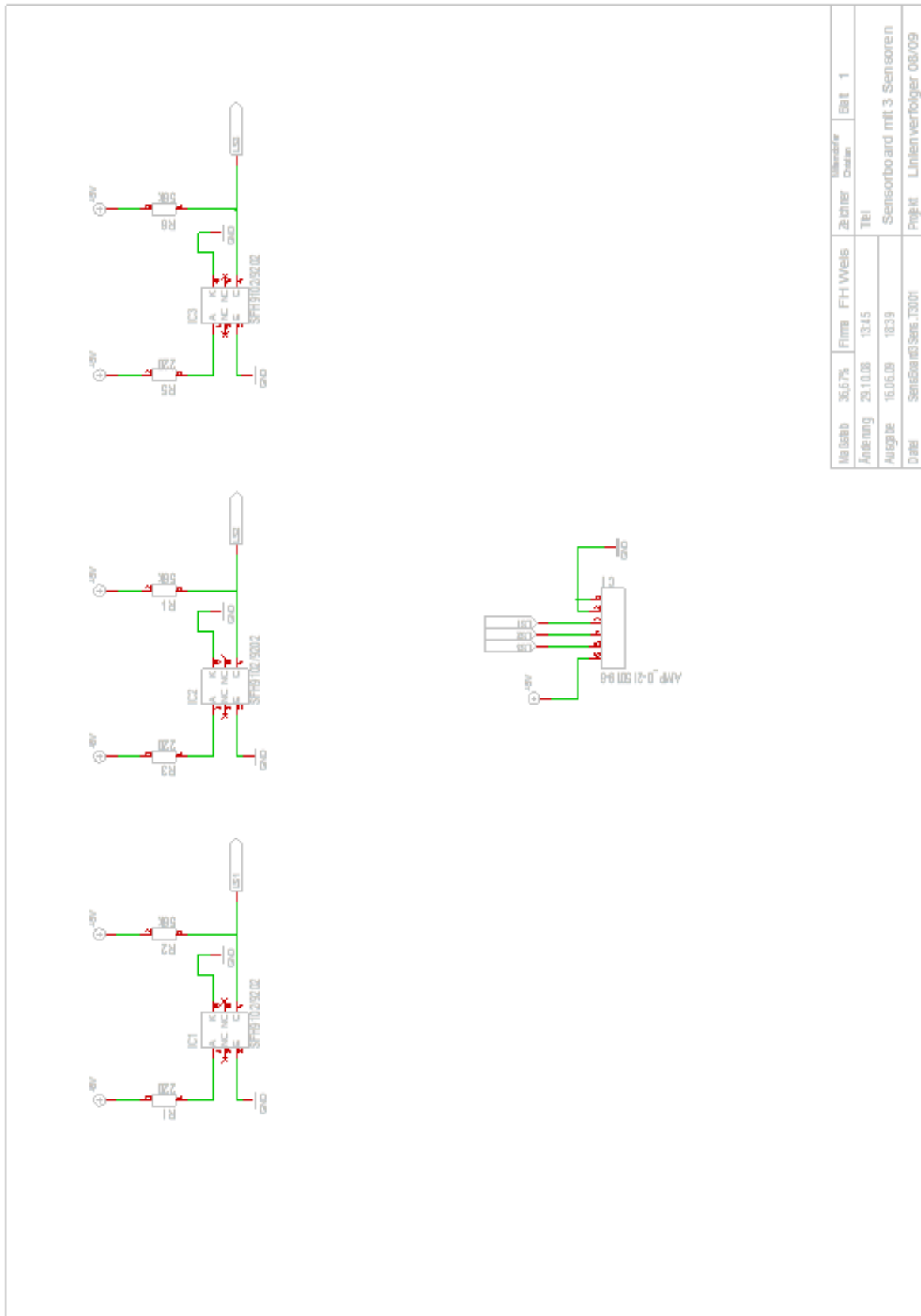


Abb. 4.9: Linienboard 3Sensor - Schaltungsplan

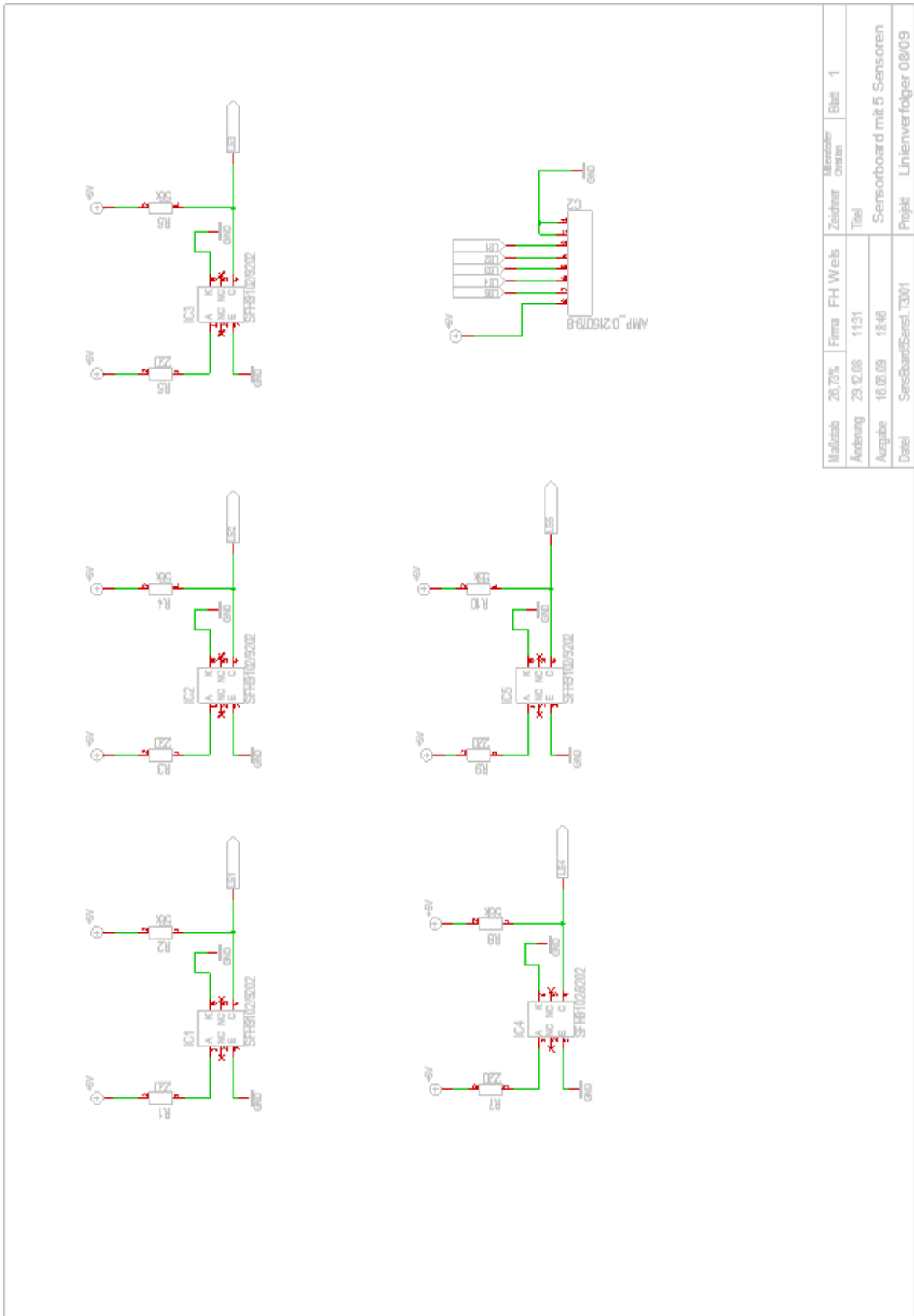


Abb. 4.10: Linienboard 5Sensor - Schaltungsplan

4.4.2 Bestückungspläne

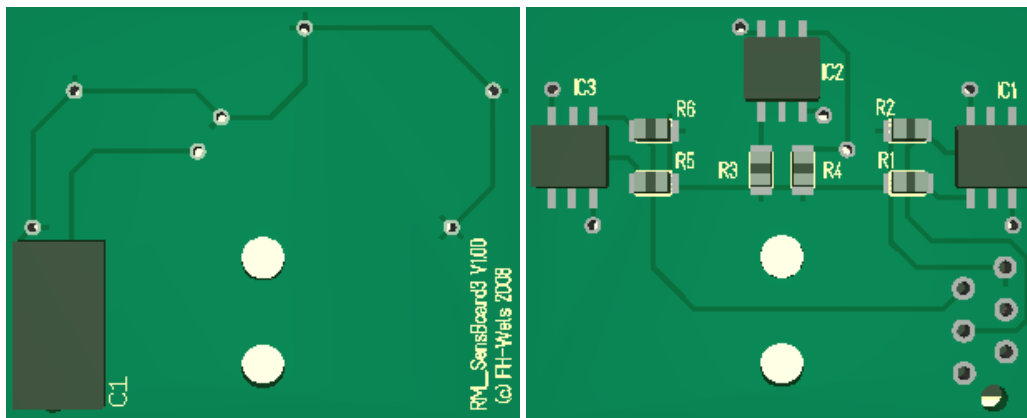


Abb. 4.11: Linienboard 3Sensor - 3D-Ansicht - Top & Bottom

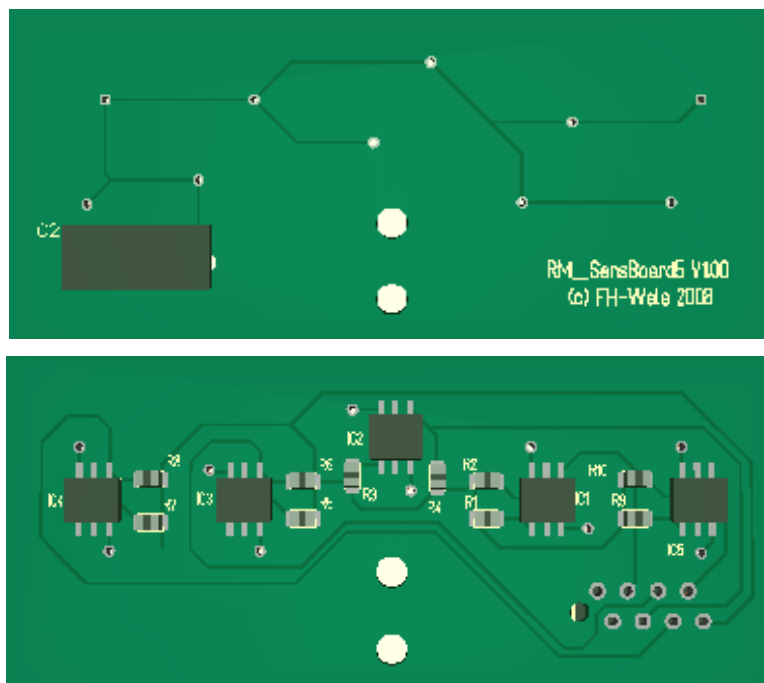


Abb. 4.12: Linienboard 5Sensor - 3D-Ansicht - Top & Bottom

4.4.3 Stückliste

In den folgenden Tabellen sind die Stücklisten, welche für die Bestückung des 3- und 5-sensorigen Linienboards verwendet wurden, abgebildet.

Name	Bauteilart	Wert	Gehäuse	Anzahl	Top	Bottom	Art
C1	Steckverbinder MICROMATCH 6 Pol	2x3	AMP	1	1	0	bedrahtet
IC1-IC3	Linienensor	SFH 9201	-	3	0	3	SMD
R1, R3, R5	Widerstand	220	0805	3	0	3	SMD
R2, R4, R6	Widerstand	56k	0805	3	0	3	SMD

Tab. 4.3: Linienboard 3Sensor - Stückliste

Name	Bauteilart	Wert	Gehäuse	Anzahl	Top	Bottom	Art
C2	Steckverbinder MICROMATCH 8 Pol	2x4	AMP	1	1	0	bedrahtet
IC1-IC5	Linienensor	SFH 9201	-	5	0	5	SMD
R1, R3, R5, R7, R9	Widerstand	220	0805	5	0	5	SMD
R2, R4, R6, R8, R10	Widerstand	56k	0805	5	0	5	SMD

Tab. 4.4: Linienboard 5Sensor - Stückliste

4.4.4 Platinen

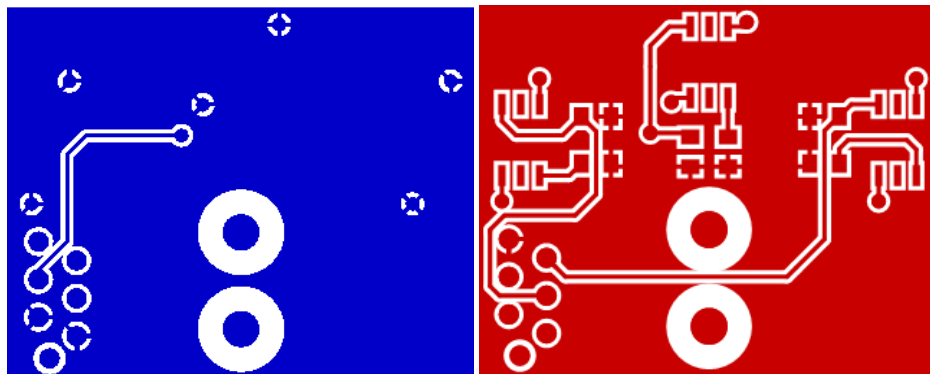


Abb. 4.13: Linienboard 3Sensor - Platine - Top & Bottom

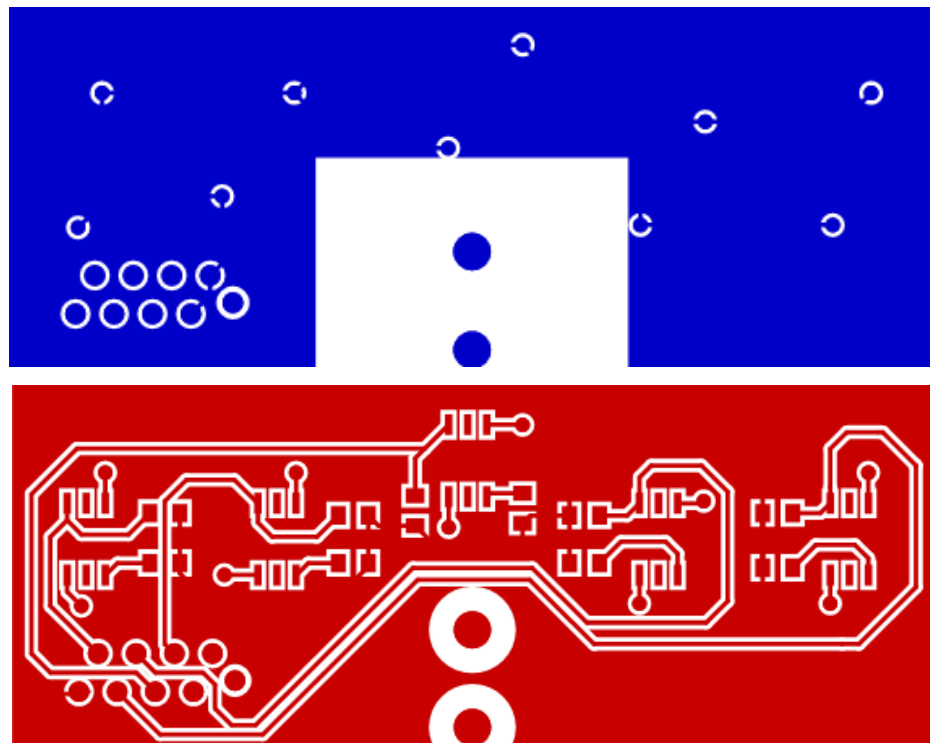


Abb. 4.14: Linienboard 5Sensor - Platine - Top & Bottom

4.4.5 Anmerkungen

Während der Versuche wurde festgestellt, dass die Reflexlichtschranken empfindlich auf den Einfall von Tageslicht reagieren, was zur Folge hatte, dass die korrekte Auswertung der Signale nicht mehr erfolgte. Aus diesem Grund wurde eine Gummilippe, welche auf der Platine angebracht wurde, verwendet.

Weiters wurde bei den Versuchen festgestellt, dass der angegebene Arbeitsabstand von 1-5mm der Infrarot-Reflexlichtschranke (Datenblatt), zu unterschiedlichem Verhalten der Sensoren führte. Aus diesem Grund ist es günstiger einen geringeren Abstand zu wählen.

4.5 Abstandssensoren

Um während des Line Follower Enhanced Modus den Hindernissen (Tunnel, Ziegelstein) ausweichen zu können, wurden 3 Abstandssensoren des Typs GP2Y0D310K und GP2Y0D340K der Firma SHARP (Abbildung 4.15) verwendet.



Abbildung 4.15: Abstandssensoren GP2Y0D310K, GP2Y0D340K

2 Sensoren (GP2Y0D310K, GP2Y0D340K) wurden für den Ziegelstein verwendet. Ein weiterer Sensor (GP2Y0D340K) wurde für die Tunneldurchfahrt verwendet, um sicherzustellen, dass die Sensoren für den Ziegelstein nicht frühzeitig die Ausweichroutine auslösen. Dieser Sensor ist leicht nach oben gerichtet und erkennt, ob es sich bei dem Hindernis um den Tunnel oder den Ziegelstein handelt.

Der digitale Abstandssensor GP2Y0D310K hat laut Datenblatt eine Einsatzweite von 10cm und der digitale Abstandssensor GP2Y0D340K hat laut Datenblatt eine Einsatzweite von 40cm. Beide Sensoren benötigen eine Versorgungsspannung von 5V, jedoch benötigen diese Sensoren auch noch eine externe Beschaltung (Abbildung 4.16).

Auch diese Schaltung wurde bereits von Herrn Ing. Zauner Michael BSc des Robo Racing Teams der FH-Wels als modulare Platine entwickelt und wurde aus diesem Grund für dieses Projekt verwendet.

4.5.1 Schaltplan

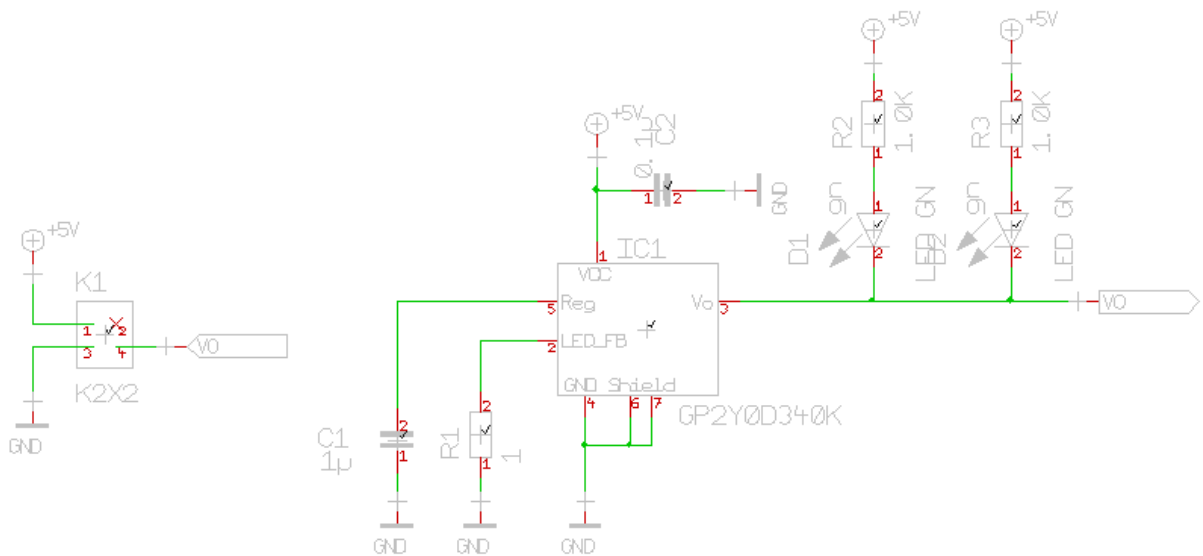


Abb. 4.16: GP2Y0D310K, GP2Y0D340K - Schaltplan

4.5.2 Bestückungsplan

In den folgenden Abbildungen sind die Top- und Bottom-View, sowie eine 3D-Ansicht der GP2Y0D310K und GP2Y0D340K abgebildet.

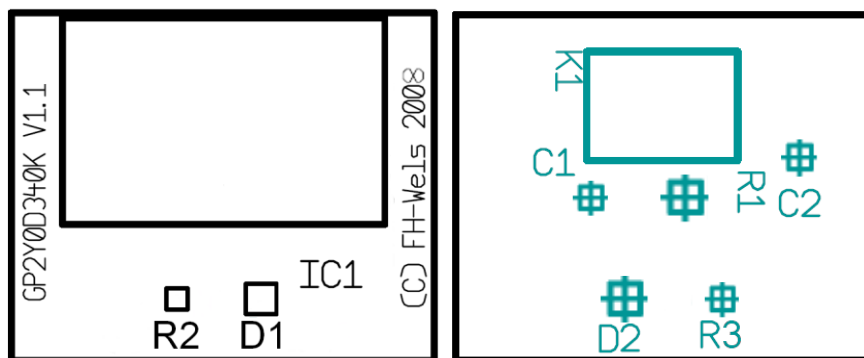


Abb. 4.17: GP2Y0D310K, GP2Y0D340K - Bestückungsplan - Top & Bottom

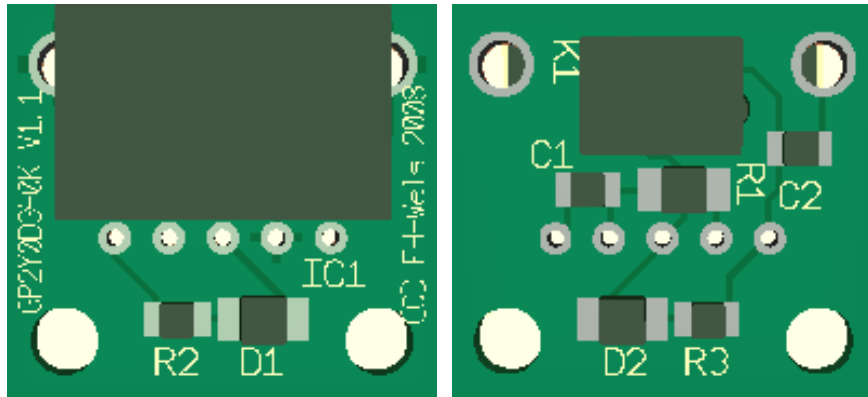


Abb. 4.18: GP2Y0D310K, GP2Y0D340K - Bestückungsplan - 3D-Ansicht - Top & Bottom

4.5.3 Stückliste

In der folgenden Tabelle ist die Stückliste, welche für die Bestückung der Abstandssensoren GP2Y0D310K und GP2Y0D340K verwendet wurde, abgebildet.

Name	Bauteilart	Wert	Gehäuse	Anzahl	Top	Bottom	Art
C1	Keramikkondensator	1 μ F/50V	0805	1	0	1	SMD
C2	Keramikkondensator	100nF/50V	0805	1	0	1	SMD
D1, D2	LED	grün	1206	2	1	1	SMD
K1	Steckverbinder MICROMATCH 4 Pol	2x2	AMP	1	0	1	bedrahtet
IC1	Abstandssensor	GP2Y0D310K, GP2Y0D340K	-	1	1	0	bedrahtet
R1	Widerstand	1	0805	1	0	1	SMD
R2, R3	Widerstand	1k	0805	2	1	1	SMD

Tab. 4.5: GP2Y0D310K, GP2Y0D340K - Stückliste

4.5.4 Anmerkungen

Während der Versuche wurde festgestellt, dass der analoge Abstandssensor GP2D120 sehr empfindlich auf den Einfall von Tageslicht reagierte und nicht mehr einwandfrei funktioniert. Aus diesem Grund wurde dieser durch die digitalen Abstandssensoren GP2Y0D310K und GP2Y0D340K ersetzt.

Auf Grund diverser Schwierigkeiten während der Versuche und während des Wettbewerbs wäre der Einsatz neuer Sensorik wie z.B. von Ultraschall bzw. mechanischer Sensoren in diesem Bereich empfehlenswert.

5. Software

Das folgende Kapitel befasst sich mit der Entwicklung der Software, welche für die einzelnen mechanischen und elektronischen Komponenten des Roboters benötigt werden. Diese Software unterteilt sich grundsätzlich in den Line Follower - und den Line Follower Enhanced Modus. Eventuelle Anmerkungen zu den Softwarekomponenten werden in den jeweiligen Unterkapiteln angeführt.

Die Softwareentwicklung des Roboters erfolgte in der imperativen Programmiersprache C. Die Entwicklungsumgebung setzt sich aus dem Programm Code Vision AVR 2.04.4a, für die Erstellung der Software und dem Programm AVR Studio 4.18, für die Programmierung des Roboters, zusammen.

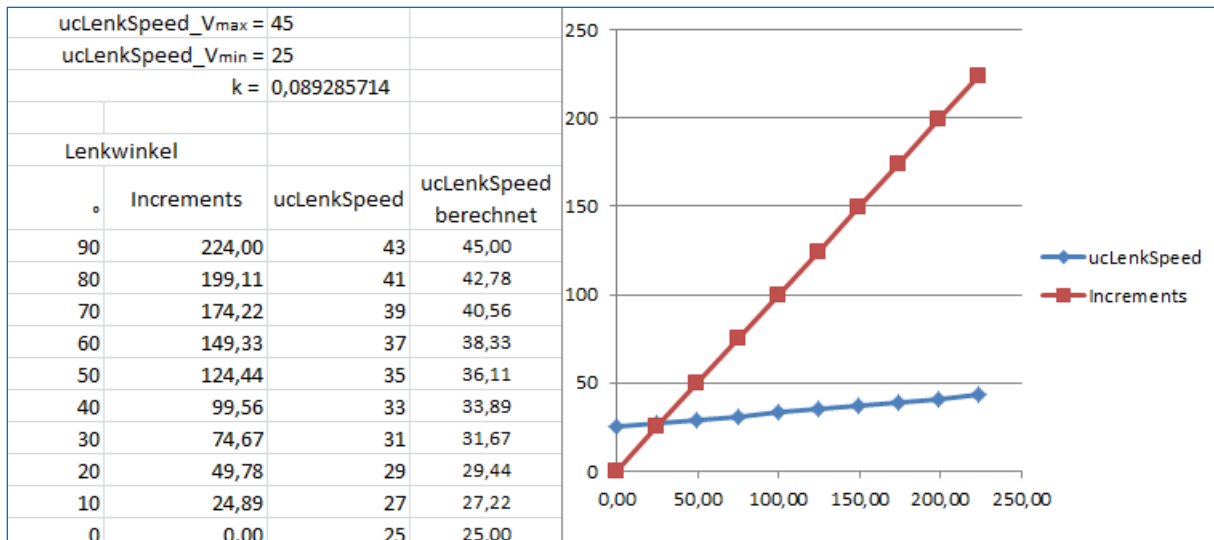
Da die Neuentwicklung aller Softwarekomponenten im Rahmen des fächerübergreifenden Projektes nicht möglich ist, wurden diverse Komponenten, wie z.B. die pulswidenmodulationsbasierte Ansteuerung der Motoren, die Entfernungsbestimmung für die verwendete Sensorik oder die Regelung der Motoren zur Verfügung gestellt.

5.1 Line Follower- & Line Follower Enhanced Modus

Die beiden Modi können über einen DIP - Schalter aktiviert werden. Es befinden sich alle Routinen für beide Betriebsarten im selben Code. Enhanced spezifische Abschnitte im Code werden durch die Abfrage des aktuellen Betriebsmodus ein- bzw. ausgeblendet.

5.1.1 Lenkungssteuerung

Die Lenkgeschwindigkeit wird in unserem Programm in Abhängigkeit des aktuellen Lenkwinkels entlang einer Geraden ($y = k * x + d$) ermittelt. Je größer der aktuelle Lenkwinkel ist, desto schneller wird der Lenkmotor angesteuert. Im Programm muss lediglich der maximale- und der minimale Lenkspeed eingegeben werden.



Ermittlung des Lenkspeeds:

```

// Motor 1 = Lenkung
// Lenkgeschwindigkeit setzen
// y = k * x + d (siehe beiliegendes Excel)
ucLenkSpeed = (unsigned char) (((fLenkSpeed_Max - fLenkSpeed_Min) / 224) * abs(s1ActualIncrements) + fLenkSpeed_Min);

```

Im folgenden Code Teil befinden sich die Abfragen für die Linienunterbrechung des Enhanced Modus sowie die Ermittlung der Richtung in die gelenkt werden muss. Bei einer Linienunterbrechung darf keine Lenkaktion ausgeführt werden. Hingegen bei einem aktiven linken oder rechten Sensor muss eine Lenkaktion erfolgen. Die Lenkrichtung wird mit Hilfe der aktuell aktiven Sensoren ermittelt. Ist der mittlere Sensor des Sensorbords aktiv, wird keine Lenkaktion ausgeführt.

```

/*
  IR (Infrarot) Sensor ein = 1
  IR (Infrarot) Sensor aus = 0

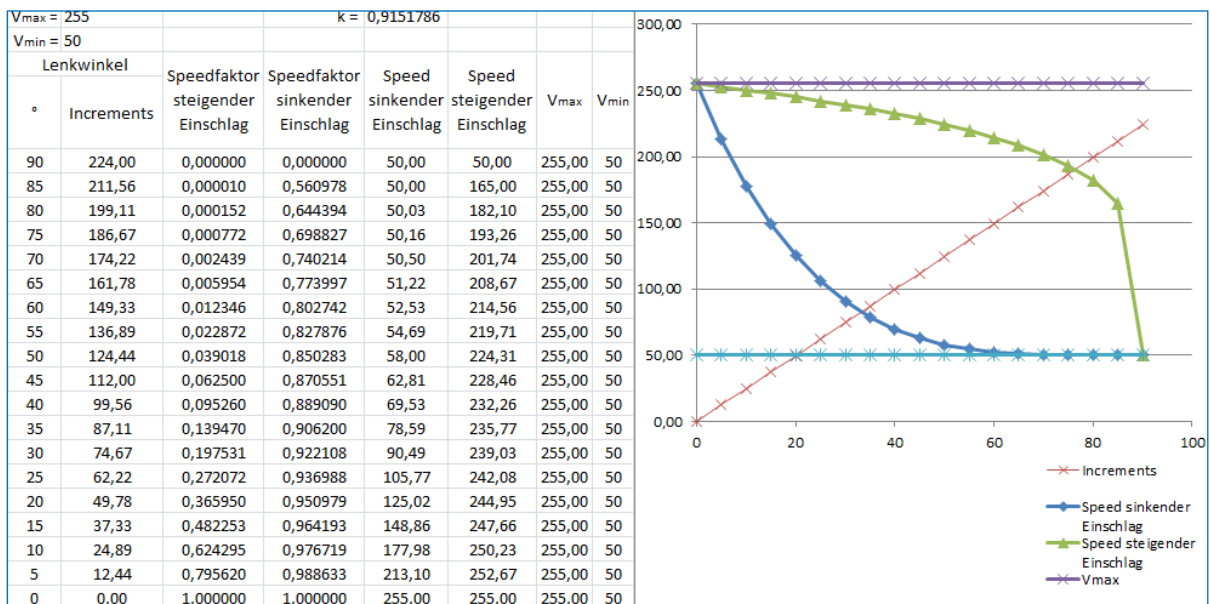
  |-----|
  | 2 | 1 | 0 | -1 | -2 |
  | L2_3BIT | L1_3BIT | M_3BIT | R1_3BIT | R2_3BIT |
  |-----|
  |-----|
  |-----|

*/
if((ENHANCED_SLALOM_MODE) &&
((scLetzter_AktiverSensor == 0) || (scLetzter_AktiverSensor == -1) || (scLetzter_AktiverSensor == 1)) &&
(L2_3BIT != 1) && (L1_3BIT != 1) && (M_3BIT != 1) && (R1_3BIT != 1) && (R2_3BIT != 1))
{
  SetMotor1(BACKWARD, WinkelInInkrementen(0));
  SetMotor2(FORWARD, fV_Max);
}
// Motor 1 (Lenkungsmotor)
// 1 Umdrehung = 896 Impulse => 90°/(360°/896) => 224 Impulse für 90°
// nach links lenken
else if((scAktiverSensor < 0) && (siActualIncrements <= 224))
{
  SetMotor1(FORWARD, ucLenkspeed);
  ucActualDir = FORWARD;
}
// nach rechts lenken
else if((scAktiverSensor > 0) && (siActualIncrements >= -224))
{
  SetMotor1(BACKWARD, ucLenkspeed);
  ucActualDir = BACKWARD;
}
else
{
  SetMotor1(FORWARD, 0);
}

```

5.1.2 Antriebsansteuerung

Bei der Geschwindigkeitsermittlung für den Antrieb haben wir versucht, eine Strategie zu verwirklichen, die in sämtlichen Motorsportarten angewendet wird. Bei der Einfahrt in eine Kurve wird die Geschwindigkeit augenblicklich drastisch verringert. Erkennt der Roboter jedoch einen Ausklang des Kurvenverlaufes, versucht er sofort aus der Kurve heraus zu beschleunigen.



Der anschließende Code beinhaltet im ersten Abschnitt die Geschwindigkeitsansteuerung bei verlorener Linie. Im unteren Teil befindet sich die lenkwinkelgebundene Geschwindigkeitsermittlung.

```

// Motor 2 (Antriebsmotor)
// Bei steigendem Lenkeinschlag wird die Geschwindigkeit (fSpeed) schneller gedrosselt
// Bei sinkendem Lenkeinschlag wird bereits aus der Kurve herausbeschleunigt
if(scLetzter_AktiverSensor == 2 && R2_3BIT != 1 && R1_3BIT != 1 && M_3BIT != 1)
{
    fSpeed = fV_Min;
}
else if(scLetzter_AktiverSensor == -2 && L2_3BIT != 1 && L1_3BIT != 1 && M_3BIT != 1)
{
    fSpeed = fV_Min;
}
else if(siPrior_ActualIncrements >= abs(siActualIncrements))
{
    // aus der Kurve hinausbeschleunigen setzen
    // k ist von den siActualIncrementen abhängig und nicht gerade
    fFaktor_sinkenderLenkwinkel = pow(fabs((fabs((float)siActualIncrements) - 224.0) / 224.0), 0.111);
    // y = k * x + d (siehe beiliegendes Excel)
    fSpeed = fFaktor_steigenderLenkwinkel * (fV_Max - fV_Min) + fV_Min;
}
else if (siPrior_ActualIncrements < abs(siActualIncrements))
{
    // geschwindigkeit beim einfahren in die Kurve stärker verringern
    // k ist von den siActualIncrementen abhängig und nicht gerade
    fFaktor_steigenderLenkwinkel = pow(fabs((fabs((float)siActualIncrements) - 224.0) / 224.0), 2.0);
    // y = k * x + d (siehe beiliegendes Excel)
    fSpeed = fFaktor_sinkenderLenkwinkel * (fV_Max - fV_Min) + fV_Min;
}

// Merken des aktuellen Lenkwinkels (in Inkrementen)
siPrior_ActualIncrements = abs(siActualIncrements);

```

Im Enhanced Modus wird der Speed je nach aktivem Abstandssensor auf einen vordefinierten Speed - Wert gesetzt. Dies soll eine immer gleichbleibende Startgeschwindigkeit für die Ausweichroutine gewährleisten.

```

/*
IR_DIGITAL_ZIEGEL_45 ist der digitale Distanzsensor, welcher den Ziegel erkennt
IR_DIGITAL_ZIEGEL_45 = PINA.4 (Printsteckerleiste K4), zu finden in Datei "gp2d120.h"

Geschwindigkeitsverringern bei aktivem IR_DIGITAL_ZIEGEL_45 im Enhanced Slalom - Mode,
um eine Geschwindigkeitsdrosselung vor dem Hindernis (Ziegel -> Ausweichroutine) zu bewirken
*/
if(enhanceReady)
{
    if(ENHANCED_SLALOM_MODE && IR_DIGITAL_ZIEGEL_45 && (siActualIncrements <= 50 && siActualIncrements >= -50))
    {
        fSpeed = 25; // 35
    }
    else if(ENHANCED_SLALOM_MODE && IR_DIGITAL_ZIEGEL_45)
    {
        fSpeed = 15; // 25
    }
}
// Setzen der Geschwindigkeit
SetMotor2(FORWARD, (unsigned char)fSpeed);
scLetzter_AktiverSensor = scAktiverSensor;
}

```

5.1.3 Anmerkungen

Eine Anregung zur Verbesserung der Geschwindigkeitsansteuerung wäre die Implementierung eines Reglers, der immer auf den gewünschten Speed hin regelt. Eine solche Regelung würde das Arbeiten mit der Geschwindigkeit in SI-Einheiten ermöglichen.

5.2 Hindernisse im Enhanced Modus

5.2.1 Anmerkungen zur Linienunterbrechung

Bei der Linienunterbrechung stellte sich heraus, dass unser Roboter teilweise Schwierigkeiten bei schräger Einfahrt in die Linienunterbrechung hat. Zurück zu führen ist dies auf das eher schmale Sensorboard. Der Code Teil für dieses Hindernis ist unter Punkt 5.1.1 Lenkungssteuerung veranschaulicht.

5.2.2 Anmerkungen zum Hindernis (Ziegelstein)

Die Ausweichroutine ist ein fix programmierter Programmteil. Die Ausweichrichtung wird auf Grund des Lenkwinkels beim Start der Routine festgelegt. Durch diese Möglichkeit wird immer der günstigere Weg zur Umfahrung des Hindernisses herangezogen.

```

// Funktion für die Ausweichroutine (Ziegel) des Enhanced Slalom - Mode
void StartSlalomEnhancedMode(signed int siActualIncrements)
{
    // Für Ausweichroutine rechts
    unsigned char ucRichtung1 = FORWARD;
    unsigned char ucRichtung2 = BACKWARD;
    signed int siSensor = -2;

    // Für Ausweichroutine links
    if(siActualIncrements >= 0)
    {
        ucRichtung1 = BACKWARD;
        ucRichtung2 = FORWARD;
        siSensor = 2;
    }

    SetMotor2(FORWARD,40);

    ucTimerIntervall1SecReached = 0;
    ucTimerIntervall1Sec = 300; // Timer auf 300ms setzen
    bStartTimer1Sec = 1; // Timer starten

    // 50° nach [rechts / links] lenken
    do
    {
        ucDriveOne(1, ucRichtung1, WinkelInInkremente(50), 180);
    }while(!ucTimerIntervall1SecReached);

    ucTimerIntervall1SecReached = 0;
    ucTimerIntervall1Sec = 300; // Timer auf 300ms setzen
    bStartTimer1Sec = 1; // Timer starten

    // 0° geradeaus lenken
    do
    {
        ucDriveOne(1, ucRichtung1, WinkelInInkremente(0), 180);
    }while(!ucTimerIntervall1SecReached);

    ucTimerIntervall1SecReached = 0;
    ucTimerIntervall1Sec = 200; // Timer auf 200ms setzen
    bStartTimer1Sec = 1; // Timer starten

    // 50° [links / rechts] lenken
    do
    {
        ucDriveOne(1, ucRichtung2, WinkelInInkremente(50), 180);
    }while(!ucTimerIntervall1SecReached);

    ucTimerIntervall1SecReached = 0;
    ucTimerIntervall1Sec = 100; // Timer auf 100ms setzen
    bStartTimer1Sec = 1; // Timer starten

    // 35° [links / rechts] lenken
    do
    {
        ucDriveOne(1, ucRichtung2, WinkelInInkremente(35), 180);
    }while(!ucTimerIntervall1SecReached);

    ucTimerIntervall1SecReached = 0;
    ucTimerIntervall1Sec = 100; // Timer auf 100ms setzen
    bStartTimer1Sec = 1; // Timer starten

    // 10° [links / rechts] lenken //20
    do
    {
        ucDriveOne(1, ucRichtung2, WinkelInInkremente(10), 180);
    }while(scRead3BitLinienSensor() != siSensor);

    //
    ucTimerIntervall1SecReached = 0;
    ucTimerIntervall1Sec = 75; // Timer auf 50ms setzen
    bStartTimer1Sec = 1; // Timer starten

    do
    {
        // Für Ausweichroutine rechts
        if(((L2_3BIT == 0) && (L1_3BIT == 0) && (M_3BIT == 0) && (R1_3BIT == 0) && (R2_3BIT == 0)) && siSensor == -2)
        {
            ucDriveOne(1, ucRichtung2, WinkelInInkremente(45), 180);
        }
        // Für Ausweichroutine links
        else if(((L2_3BIT == 0) && (L1_3BIT == 0) && (M_3BIT == 0) && (R1_3BIT == 0) && (R2_3BIT == 0)) && siSensor == -2)
        {
            ucDriveOne(1, ucRichtung1, WinkelInInkremente(45), 180);
        }
    }while(!ucTimerIntervall1SecReached);
}

```

5.2.3 Anmerkungen zum Hindernis (Tunnel)

Der Tunnel stellt für unser System keinerlei Behinderung dar. Deshalb kann auf den Sensorik Part für dieses Hindernis verzichtet werden.

6. Gegneranalyse & Verbesserungsvorschläge

Das Spitzenfeld bei der RobotChallenge 2011 setzte auf reine Platinenlösungen bei den Robotern. Die ersten beiden Plätze wurden von Linienverfolgern belegt, die sich mit Hilfe eines CPU-Lüfters am Boden ansaugten. Unser eher mechanisch belasteter Roboter kann noch geringfügig verbessert werden und seine aktuelle Positionierung im Mittelfeld verteidigen zu können. Ein Platz unter den Top 3 wird jedoch bei gleichbleibendem Gegnerfeld nicht möglich sein. Ein weiterer Aspekt zur Verbesserung wäre die Implementierung einer Bedienung via Fernbedienung.

Die Bedienung mit Hilfe eines LCD-Displays ist wünschenswert. Das Display sollte jedoch entfernbar realisiert werden, um unnötiges Gewicht zu sparen.

7. Quellenverzeichnis

- INNOC (Austrian Society for innovative Computer Science), 2011:
URL: <http://www.robotchallenge.org>
- Robo Racing Team der FH-Wels, 2011:
URL: <http://rrt.fh-wels.at>
- Modellbau Lindinger GmbH, 2011:
URL: <http://www.lindinger.at>
- Pauer Point Bürotechnik, 2011:
URL: <http://www.pauerpoint.at>
- Datenblätter, 2011:
URL: <http://www.datasheetcatalog.com>
URL: <http://www.alldatasheet.com>