

Johannes-Kepler-Gymnasiums

Besondere Lernleistung

Entwicklung und Inbetriebnahme eines Roboter-Bausatzes für
Schüler

Richard Roth

Chemnitz, 04.01.2016

Interner Betreuer: Herr Andreas Scheler

Externer Betreuer: M.Sc. Peter Weissig

Gliederung

| | |
|--|----------|
| 1. Einführung | Seite 3 |
| 1.1 Motivation | Seite 3 |
| 1.2 Ziel der Arbeit | Seite 4 |
| 2. Grundlagen | Seite 5 |
| 2.1 Sensoren | Seite 5 |
| 2.1.1 Bodensensoren | Seite 6 |
| 2.1.2 Radencoder | Seite 6 |
| 2.1.3 Distanzsensoren | Seite 9 |
| 2.1.4 Vergleich von SRF02 und SHARP Sensor | Seite 11 |
| 2.2 UART | Seite 11 |
| 2.3 Mikrocontroller | Seite 12 |
| 2.3.1 Hardware | Seite 12 |
| 2.3.2 Programmierung | Seite 14 |
| 3. Anforderungsanalyse | Seite 15 |
| 3.1 RoboSAX | Seite 15 |
| 3.2 WRO | Seite 16 |
| 3.3 FLL | Seite 17 |
| 3.4 Schülerpraktikum | Seite 18 |
| 3.5 Auswertung | Seite 18 |
| 4. Umsetzung | Seite 19 |
| 4.1 Konzept | Seite 19 |
| 4.2 Chassi | Seite 20 |
| 4.3 Ergebnis und Aussichten | Seite 25 |
| 5. Selbstständigkeitserklärung | Seite 27 |
| 6. Danksagung | Seite 27 |
| Literaturverzeichnis | Seite 28 |
| Anhang | Seite 32 |
| I Stückliste | Seite 32 |
| II Konstruktionspläne | Seite 33 |
| III Testprogramm | Seite 41 |

1. Einführung

1.1 Motivation

Seit einigen Jahren besuche ich die Robotik AG des Johannes Kepler Gymnasiums. Alle Teilnehmer der AG beschäftigen sich mit dem Aufbau eines Roboters, dessen Programmierung sowie mit der Einführung in die Elektrotechnik, Mechanik, Logik und Informatik.

Die im Rahmen der Arbeitsgemeinschaft zum Programmieren verwendeten Roboter sind meist Eigenbauten der Teilnehmer. Häufig nimmt der Bau eines solchen Roboters sehr viel Zeit in Anspruch. Allerdings findet die Arbeitsgemeinschaft Robotik während der Schulzeit nur einmal wöchentlich statt.

Bis zur Fertigstellung des Roboters muss die Programmierung zurücktreten. Dies kann den Zeitraum eines Schuljahres sehr leicht überschreiten. Dadurch ist die Zeit für den Bau eines Roboters stark begrenzt.

Zudem trägt die Robotik AG jedes Jahr den Roboter-Wettbewerb RoboSAX aus, wobei die AG aufgrund fehlender Roboter kaum teilnimmt.

Diese Sachverhalte sind für die Umsetzung der AG-Inhalte sehr hinderlich, sodass mir die Idee kam, einen einfachen, vorgefertigten Roboter-Bausatz für die Teilnehmer der Arbeitsgemeinschaft zu konzipieren und konstruieren.

Mein generelles Interesse an Technik, insbesondere der Konstruktion von Maschinen und deren Umsetzung unterstützt dieses Vorhaben. Zudem habe ich durch den Besuch der Robotik AG vielfältige Erfahrungen mit Robotern gesammelt und möchte gern die Schritte der Roboterentwicklung von der Planung bis zur Umsetzung komplett nachvollziehen können.

Ich erhoffe mir von der Beschäftigung mit dem Thema außerdem, meine bisherigen Erfahrungen als Grundlage für mein späteres Studium ausbauen zu können.

1.2 Ziel der Arbeit

Das Ziel der Arbeit ist die Konstruktion eines einfachen Roboters für Schüler. An diesen sollen die Teilnehmer das Programmieren ausprobieren und erlernen können. Dazu sollte der Roboter zu Anfang eine Grundausstattung an Aktorik und Sensorik besitzen, die einen problemlosen Einstieg in das Programmieren gewährleisten soll. Er sollte außerdem die Möglichkeit von Erweiterungen wie Ultraschall-Sensoren aufweisen. Um eine sichere Orientierung des Roboters zu gewährleisten, sind Abstands- sowie Boden- und Bewegungssensoren erforderlich.

Zur besseren Kommunikation des Roboters auch während der Fahrt bietet sich zudem ein Funkmodul an, welches zur drahtlose Datenübertragung genutzt wird.

Das Herzstück des Roboters soll das im Rahmen der AG Robotik bereits häufig genutzte „Keplerboard“ bilden, da es einer Vielzahl an Sensoren Platz bietet und sein Aufbau genau auf die Verwendung für derartige Zwecke zugeschnitten ist.

Wichtig ist die Konstruktion eines stabilen und robusten Roboters, dessen Aufbau und Montage in relativ wenigen, einfachen Schritten umsetzbar ist.

Um eine unproblematische Erweiterung des Roboters zu gewährleisten, sind festgelegte Schnittstellen und leichte Zerlegbarkeit unabdingbar. Dafür bietet sich eine Segmentierung des Roboters an.

Zudem sollte der Roboter für den Einsatz bei entsprechenden Wettbewerben geeignet sein. Um dies zu gewährleisten, soll eine Anforderungsanalyse erstellt werden, welche die Konstruktionsgrundlage bildet. Ein solcher Wettbewerb ist unter anderem der jährlich stattfindende Wettbewerb RoboSAX.

2. Grundlagen

Ein Roboter ist in Aufbau und Funktion vergleichbar mit einem Menschen. Die Nahrung des Roboters ist Energie, welche wir ihm über Akkus zuführen. Motoren bilden den Antrieb und sorgen für Bewegung, so wie Muskeln beim Menschen. Die menschlichen Sinne sind vergleichbar mit den Sensoren eines Roboters. Sie erfassen die Umwelt und geben die Daten weiter an das ‘Gehirn’ des Roboters - die CPU. Wie im Gehirn eines Menschen werden hier die eingehenden Daten verarbeitet und Entscheidungen getroffen. Von hier aus erfolgt die Steuerung des Roboters. Im Folgenden sollen nun die verwendeten Bauelemente kurz erklärt und vorgestellt werden.

2.1 Sensoren

Entsprechend dieses Vergleichs zwischen Robotern und Menschen sind Sensoren die Sinne des Roboters. Sie erfassen die Umwelt und werden vom Roboter zur Orientierung benötigt.

Die Aufgabe von Sensoren besteht in der Umwandlung physikalischer in elektrische Signale. Derartige Prozesse laufen zum Beispiel in Temperaturfühlern oder Feuchtigkeitssensoren ab.

Die Klassifikation der verschiedenen Sensoren erfolgt in der Unterscheidung von internen und externen Sensoren, welche nochmals in aktive und passive Sensoren unterteilt werden.

Interne Sensoren messen innerhalb des Systems. Beispielsweise sind Sensoren zur Messung des Ladezustandes einer Batterie oder Inkrementalgeber. Externe Sensoren führen hingegen Messungen in ihrer Umgebung aus, wie es bei Thermometern oder Bodensensoren der Fall ist.

Als passiv werden Sensoren bezeichnet, wenn sie die Umgebungsenergie für ihre Funktion nutzen, wie zum Beispiel in Kameras. Aktive Sensoren hingegen emittieren Energie und bestimmen daraufhin die Reaktion der Umgebung. Dieses System wird unter anderem bei Abstands- und Bodensensoren eingesetzt.^[28]

2.1.1 Bodensensoren

Bodensensoren geben Aufschluss über die Beschaffenheit oder die Farbe des Untergrundes. Um auf den Vergleich zum Menschen zurückzukommen, stellt ein Bodensensor den Seh- und Tastsinn eines Roboters dar.

Bodensensoren bestehen zumeist aus einer LED und einem Fotoelement. Die Beschaffenheit oder Farbe des Bodens sind ausschlaggebend dafür, wie viel Licht in Richtung Fototransistor reflektiert wird. Je mehr Licht reflektiert wird, desto mehr Strom kann fließen. Dabei sind der Abstand des Sensors zum Boden und die optische Eigenschaften des Untergrunds maßgebend für die Intensität des Signals. Das in Bodennähe zumeist geringe Störlicht sorgt für ein weitgehend unverfälschtes Messergebnis. ^[2]

Pololu QTR-1A Bodensensoren

Das Ausgangssignal des QTR-1A ist analog. Zwar kann dieses in ein digitales Signal umgewandelt werden, jedoch ist dies bei dem genutzten Mikrocontrollerboard nicht nötig.

Zum Arbeiten benötigt der Sensor 5 V und 17 mA. Die Ideale Höhe über dem Boden beträgt 3 mm, wobei die Ausgangsspannung bei dieser Höhe und einer schwarzen Linie 4,5 V beträgt. Bei einer Höhe von 9,5 mm ist das ausgehende Signal nur noch 1,2 V hoch. ^[2]

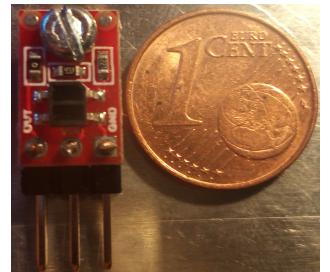
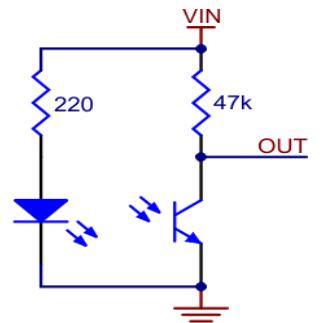


Bild 6-a: verbauter Sensor im Vergleich zu einem 1 Cent Stück



6-b: Schaltplan des QTR-1A Sensors.

Radencoder sind optische Sensoren zur Messung der

Umdrehungen eines Rades durch Beobachtung von Markierungen. ^[12]

Sie lassen sich in zwei Gruppen trennen: inkrementelle und absolute Radencoder. Erstere geben keine absolute Position an, sondern messen eine Winkeländerung. Absolute Radencoder hingegen können durch eine spezielle Codierung des Geberelementes innerhalb einer Umdrehung ihre konkrete Position angeben und behalten diese Daten auch im spannungslosen Zustand. ^[19] ^[28]

Da bei dem hier geplanten Vorhaben eine derart genaue Positionsbestimmung und -wiedergabe von geringer Bedeutung ist, werden inkrementelle Radencoder, auch Inkrementalgeber genannt, verwendet.

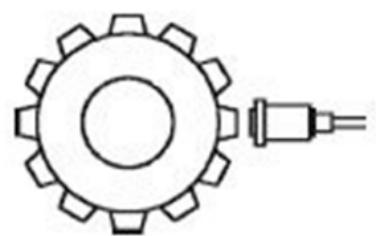
Zudem lassen sie sich in translatorische und rotatorische Encoder unterscheiden. Da bei diesem Projekt nur rotatorische Radencoder genutzt werden und die Funktionsprinzipien weitgehend gleich sind, wird hier nur auf rotatorische Radencoder eingegangen.

Die Anwendungsbereiche von Inkrementalgeber sind vielfältig. So findet man diese in Fließbändern, Fräsen und natürlich in Robotern. Sie dienen zur Drehzahl-, zur Geschwindigkeits- oder zur Positionsbestimmung.

Der Inkrementalgeber besteht aus einem Messelement und einem Geberelement, welches in gleichen Abständen unterteilt ist. Diese müssen den Umfang des Geberelementes im gleichen Maße aufteilen. Als Geberelemente werden oft Zahnräder oder Scheiben genutzt. Mit Hilfe des Inkrementalgebers sind somit Änderungen der Winkel- bzw. Wegposition messbar.

Durch die fortschreitende Bewegung der Segmente auf dem Geberelement wird ein Signal erzeugt. Das am Ende ausgehende Signal kann dabei analog oder digital sein. Die Abfolge der Signale zählt das Encoder-Interface und berechnet daraus den insgesamt zurückgelegte Weg. Da die Segmente keine genaue Aussage über den ursprünglichen Ort treffen, an welchem die Messung startete, wird von relativer Winkelbestimmung gesprochen.

Inkrementalgeber greifen auf unterschiedliche Techniken und Prinzipien zurück. Als erstes sei hier das induktive Prinzip genannt. Dabei wird ein Zahnrad als Geberelement genutzt. Die Anzahl der Zähne entspricht hier der Anzahl der Segmente. Die Signalgewinnung erfolgt bei diesem Prinzip durch die Änderung des induktiven Widerstands. Der dadurch entstehende sinusähnliche Spannungsverlauf wird dann zu einem Ausgangssignal verarbeitet. Vorteilhaft ist, dass das System ohne direkten mechanischen Kontakt auskommt. Aus diesem Grund findet er Anwendung in Autos oder Orten mit starken Erschütterungen und ungünstigen Umgebungsbedingungen.



7-a: Schema eines Inkrementalsgebers nach induktiven Prinzip.

Ein weiteres Prinzip ist das optoelektronische Prinzip. Hierbei fällt Licht durch Schlitze in der Scheibe oder wird durch kleine Spiegel reflektiert. Es fällt dann auf eine Fotozelle, welche das Licht in ein elektrisches Signal umwandelt. Das erzeugte Signal ist fast sinusförmig.

Das optoelektronische Prinzip ist das häufigst verwendete Prinzip bei Inkrementalgebern. ^[9]

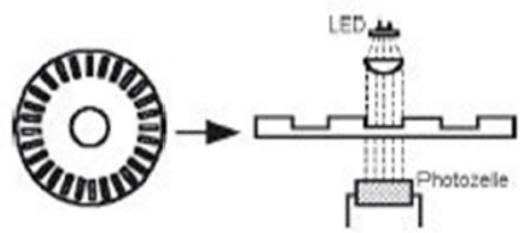
Zuletzt sei der Hall-Effekt Sensor genannt. Dabei wird eine von einem Steuerstrom durchflossene, dünne Metallplatte senkrecht von einem Magnetfeld durchdrungen. An den beiden Kanten der Platte lässt sich eine Spannung abgreifen. Diese Hall-Spannung entsteht durch die Ablenkung von Elektronen durch die Lorentzkraft. Der verwendete Radencoder der Firma Pololu arbeitet nach diesem Prinzip. ^[30]

Grundsätzlich ist anzumerken, dass Inkrementalgeber auf einen vorher definierten niedrig Pegel mit „logisch Null“ und einen hohen Pegel mit „logisch Eins“ reagieren. Diese Eigenschaft ermöglicht die einfache digitale Auswertung durch Mikrocontroller, welche somit keine Zwischenzustände verarbeiten müssen. Dementsprechend werden Ressourcen im Controller gespart.

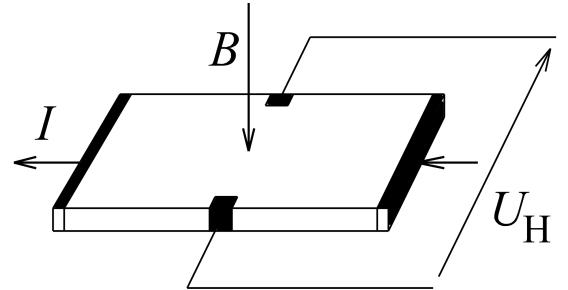
Pololu 48 CPR Radencoder

Das Geberelement ist eine Rad mit 48 Inkrementen, welche die gleiche Anzahl an Signalen pro Motorumdrehung erzeugen. Durch das verbaute Getriebe erhöht sich die Anzahl der Signale auf 1632 Signale pro Radumdrehung.

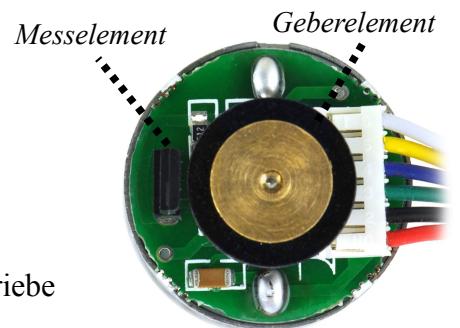
Der genannte Radencoder besitzt zwei Ausgänge, um die Drehrichtung bestimmen zu können. Dies geschieht durch die Positionierung eines



8-a: Schema eines Inkrementalgebers nach optoelektronischen Prinzip.



8-b: Schema des Hall-Effekts. Durchdringt das Magnetfeld B die Metallplatte, lässt sich eine Hall-Spannung U_H messen.



8-c: Der verwendete Radencoder der Firma Pololu.

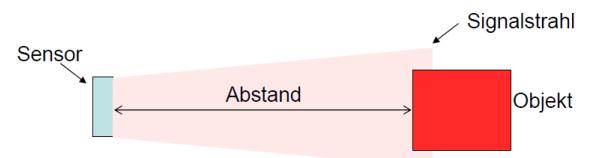
zweiten Messelement neben dem ersten. Dadurch werden zwei, um 90° versetzte Signale erzeugt.



9-a: Darstellung der beiden um 90° versetzte Signale des 48CPR

2.1.3 Distanzsensoren

Die Aufgabe von Distanzsensoren ist die Messung der Entfernung zu einem Gegenstand. Dabei gibt es im Wesentlichen drei verschiedene Möglichkeiten der Umsetzung: Messung der Signallaufzeit, der Phasendifferenz oder durch Triangulation. Die Messung der Signallaufzeit beruht auf der Bestimmung des Zeitunterschiedes zwischen aus- und eingehendem Signal.



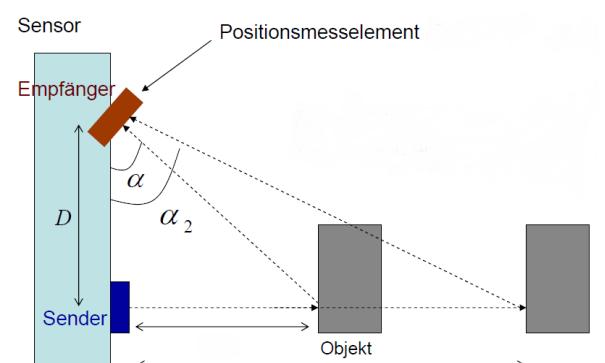
Dabei sollten Sender und Empfänger relativ nah beieinander liegen. Der Abstand zum Objekt berechnet sich aus der Hälfte der gemessenen Zeit multipliziert mit der Geschwindigkeit des gesendeten Signals. Ein Nachteil kann dabei die eventuell auftretende Mehrfachreflektion sein, wodurch sich der Wert der gemessenen Entfernung unrealistisch vergrößern kann.

9-b: Schema zur Entfernungsmessung mittels Signallaufzeit.

Bei der Messung der Phasendifferenz werden ebenfalls Signale emittiert und absorbiert.

Der Sensor misst die Phasenverschiebung der ausgesendeten und empfangenen Signale. Aus der Berechnung der Phasendifferenz kann nun der Weg berechnet werden.

Demgegenüber beruht die Triangulation auf der Winkelmessung. Ein Sender emittiert ein Signal in eine bestimmte Richtung, welches von einem Gegenstand reflektiert wird. Der Empfänger fängt das Signal ein und misst den Winkel zwischen



9-c: Schema zur Entfernungsmessung mittels Triangulation.

emittierten und absorbierten Signal. Daraus kann nun der Abstand zum Objekt berechnet werden. Ein Nachteil der Triangulation ist, dass das Signal lediglich aus einem einzelnen Strahl besteht und somit nur einen kleinen Bereich abdecken kann. Für die Abstandsmessung wird im Wesentlichen auf zwei verschiedene Medien zurückgegriffen: Infrarotlicht und Ultraschall. ^[28]

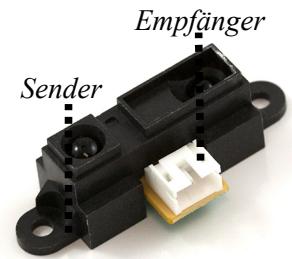
SHARP GP2Y0A21YK

Der Abstandssensoren der Firma SHARP funktioniert nach dem Prinzip der Triangulation, auf Basis der Infrarot Technik.

Das Ausgangssignal des SHARP-Sensors ist eine analoge Spannung zwischen 0,25V für Entferungen von 800mm und 2,30V für Entferungen von 100mm. ^[31]

Die Schnittstelle des Sensors ist leicht kompatibel und ermöglicht somit eine leichte Anbindung an das Mikrocontrollerboard. Die Kennlinie des SHARPs ist zudem nicht linear.

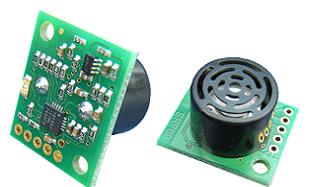
Da kontinuierlich gemessen wird ist der Stromverbrauch des Sensors relativ groß. Außerdem kann er lediglich im Rahmen eines kleinen Messwinkels von höchstens 3° messen. ^[3]



10-a: Bild des SHARP Sensors

Ultraschallsensor SRF02

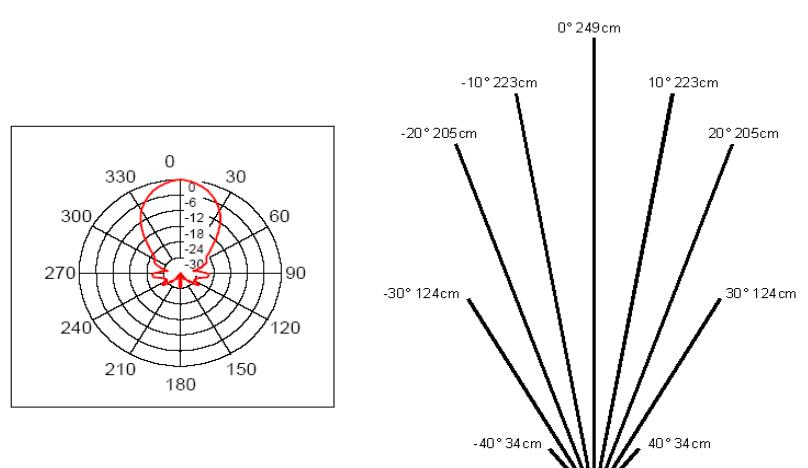
Der Ultraschallsensor SRF02 funktioniert nach dem Prinzip der Signallaufzeit. Die Frequenz dieses Sensors liegt bei 40 kHz. Das Ausgangssignal des Sensors ist Digital und er besitzt eine UART Schnittstelle. Er kann Gegenstände in einem Abstand zwischen 160mm und 6m detektieren. ^[33] ^[34]



10-b: Bild des SRF02.

2.1.4 Vergleich von SRF02 und SHARP Sensor

Im Gegensatz zum Sensor von SHARP besitzt der Ultraschallsensor einen weit gefächerten Messkegel, wohingegen der SHARP Sensor nur einen Strahl nach vorn aussendet.



11-a: Messkegel des SRF02.

Während der SHARP Sensor einen analogen Ausgang besitzt hat der Ultraschallsensor SRF02 einen digitalen Ausgang. Somit eignet sich der SHARP eher zur punktuellen Abstandsmessung und der SRF02 eher zum Erkennen von Hindernissen.

2.2 UART

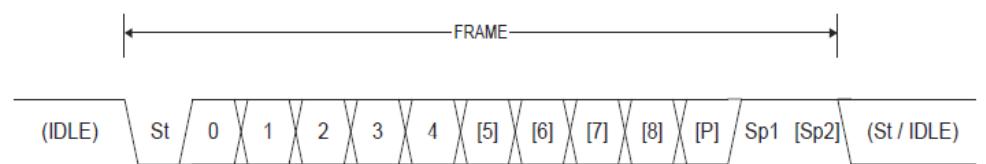
Die Kommunikation eines Roboters ist wiederum vergleichbar mit der eines Menschen. Über Nerven senden Sinne Informationen und Muskeln werden angesteuert. Beim Roboter hingegen erfolgt die Übertragung der Informationen der Sensoren und das Ansteuern der Motoren über Kabel. Menschen können untereinander mittels Sprache, Stil und Mimik kommunizieren. Bei Robotern ist dies etwas anders.

Die Kommunikation umfasst ebenfalls verschiedene Schnittstellen. Diese können seriell oder parallel auftreten. Beispiele für Roboter-Kommunikationssysteme sind SPI oder UART.

Beim UART, dem Universal Asynchronous Receiver Transmitter, handelt es sich um eine weit verbreitete serielle Schnittstelle mit asynchroner Übertragung. Er wandelt die parallelen Daten der Computers oder Chips in einen seriellen Datenstrom um und sendet diesen. Empfangene serielle Daten werden auf dem selben Weg in parallele Daten umgewandelt.

Bei dieser Übertragungsart ist es besonders wichtig, dass Sender und Empfänger über das gleiche Protokoll verfügen.

Zur Übermittlung werden lediglich drei Leitungen benötigt: eine zum Senden, eine für den Empfang und ein gemeinsamer Masseanschluss. Erfolgt über die genannten Leitungen keine Datenübermittlung, liegt ein „logisch Eins“ an. Werden Daten übertragen, wird zuerst ein Startbit gesendet, welches einem „logisch Null“ entspricht. Es folgen meist acht, selten neun Datenbits. Abschließend ist die Übermittlung eines Paritätsbits möglich, welches zur Fehlerüberprüfung genutzt wird. Die Datenübertragung wird mit einem Stopppbit beendet, welches immer einem „logischen Eins“ entspricht. ^[35]



12-a: Schema einer Datenübertragung nach UART.

St – Startbit

0-8 – Datenbits

P – Paritätsbit

Sp – Stopppbit

IDLE – Keine Übertragung

2.3 Mikrocontroller

2.3.1 Hardware

Der Mikrocontroller koordiniert den Roboter und wertet dessen Sensoren aus, ähnlich wie es das menschliche Gehirn tut. Dabei besteht die Hauptaufgabe in der Steuerung eines Roboters oder sonstiger elektronischer Schaltungen. Der Mikrocontroller muss eingehende Signale verarbeiten können und die angeschlossenen Geräte entsprechend steuern. Es gilt das E-V-A Prinzip: Eingabe, Verarbeitung und Ausgabe.

Ähnlich einem Computer besitzt ein Mikrocontroller ebenfalls „CPU“, RAM und ROM. Die CPU - Central Processing Unit - übernimmt Berechnungen und Durchführung logischer Operationen.

Der RAM - Random Access Memory - ist zuständig für das Speichern temporärer Daten und verfügt über eine geringe Schreib- und Lesezeit.

Der Read Only Memory, ROM, enthält das Programm. Auch nach Abschalten der Versorgungsspannung bleiben gespeicherte Daten erhalten, wobei jedoch eine längere Schreibzeit anfällt.

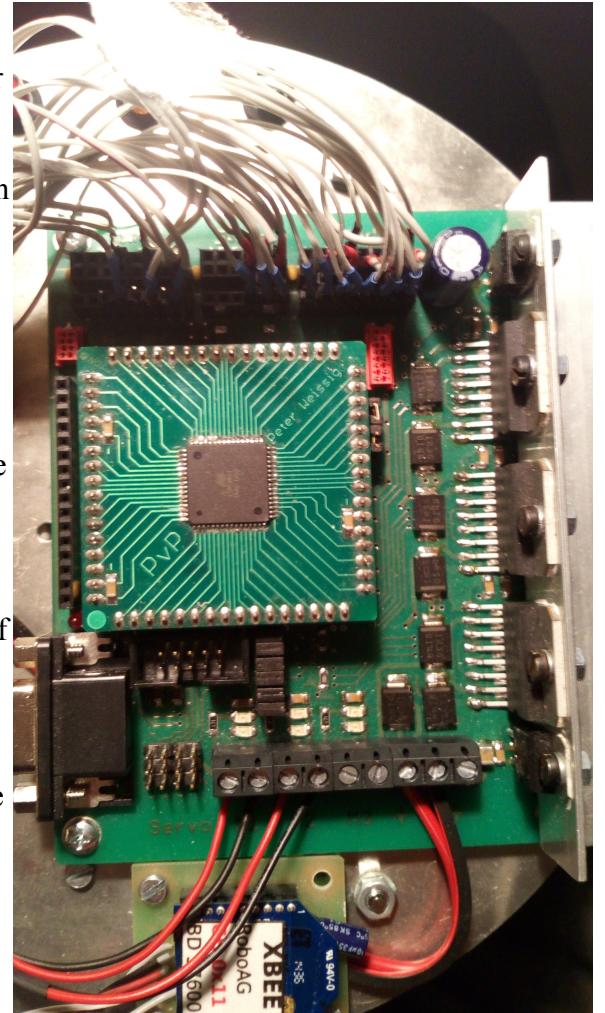
Von hoher Bedeutung bei Mikrocontrollern ist die Taktrate, welche die Arbeitsgeschwindigkeit des Controllers angibt. Die Peripherie, welche unter anderem Aussagen über die Kommunikationsmöglichkeiten des Controllers nach außen ermöglicht, bedarf gleichermaßen Beachtung.

Ein weiteres Unterscheidungsmerkmal von Mikrocontrollern ist zum Beispiel die Bitbreite. Sie gibt die Anzahl der pro Berechnung oder Verschiebung verwendeten Bits an. So wird bei einer Bitbreite von 1 Byte lediglich ein Zahlenbereich von 0 bis 255 abgedeckt.

Zwar können durch spezielle Software auch 16 oder 32 bit große Zahlen durchgeführt werden, jedoch wird das System dadurch langsamer.

Auch an den Bauformen lassen sich Controller unterscheiden. Zum einen gibt es THT - Controller. Diese lassen sich auf der Rückseite von Platinen verlöten, da sie kleine Beine aus Metall als Anschlüsse haben. Zum anderen gibt es SMD - Controller. Diese besitzen kleine Lötähnchen, wodurch diese Bauart direkt auf der Oberfläche der Platine verlötet werden kann.

Vorteile der Verwendung eines Mikrocontrollers liegt zum einen in einem sehr geringen Stromverbrauch und zum anderen in der sehr geringen Baugröße von etwa 1 x 1 Zentimeter. Darüber hinaus benötigt der Mikrocontroller auch kein eigenes, umfangreiches Betriebssystem. Das Programm kann direkt auf die angeschlossene Hardware zugreifen. Die Programmierung des Mikrocontrollers erfolgt über einen Compiler für z.B. C oder C++.



13-a: Atmega64 Mikrocontroller auf dem Mikrocontrollerboard „Keplerboard“.

Insgesamt ermöglicht ein Mikrocontroller bei geringen Kosten und einfachem Hardwarezugriff das Erfassen der eingehenden Daten, deren Verarbeitung sowie die Ausgabe von Signalen. Einschränkungen entstehen allenfalls durch fehlende Speicherverwaltung oder durch geringere Ressourcen gegenüber dem Computer. ^{[14][15][17]}

ATMEL mega 64

In dieser Arbeit wird als Mikrocontroller der ATmega64 der Firma ATMEL genutzt. Dieser verfügt über 64 kByte internen ROM sowie 4 kByte Arbeitsspeicher. Eine Erweiterung durch einen externen Speicher auf bis zu 64 kByte ist möglich.

Der im Keplerboard verwendete Chip wird als SMD-Bauteil gefertigt und ist 15 x 15 mm groß. Bei einer Betriebsspannung von 4,5 V bis 5,5 V besitzt er eine Taktfrequenz von bis zu 16 Mhz. Seine 53 programmierbaren Ein- und Ausgänge beinhalten unter anderem vier Timer, mehrere PWM-Kanäle sowie zwei UART-Schnittstellen.

Der genannte Mikrocontroller eignet sich aufgrund seiner vielseitigen Einsatzmöglichkeiten und der geringen Größe bei relativ großem Leistungsumfang in besonderem Maße für die Anwendung bei Roboter-Projekten wie diesem. Zudem besitzt er eine ausreichend große Anzahl an Ein- und Ausgängen, sodass auch für eventuelle Erweiterungen freie Pins zur Verfügung stehen. ^[1]



14-a: ATMEL mega64 Mikrocontroller. Gut zu erkennen sind die Lötfahnen.

2.3.2 Programmierung

Zum Erstellen eines Programms benutzt man verschiedene Programmiersprachen. Diese stellen Vorschriften, nach welche der Computer Rechnungen ausführen soll. Programmiersprachen lassen sich dabei in fünf Generationen einteilen: die Maschinensprache, die Assembler Sprache, die höhere Programmiersprache, die Fourth Generation Language (4GL) und zuletzt die Very High Level Language (VHLL). Aus einer Folge von Zahlen besteht die Maschinensprache. Sie ist schwer Programmierbar jedoch direkt übertragbar.

Bei Assembler steht eine symbolische Codierung für die Zahlencodes der Maschinensprache. Dabei ist Assembler an einen bestimmten Prozessor gebunden. Bei der höheren Programmiersprache werden die Algorithmen direkt aufgeschrieben und die Programme sind anwendungs- und maschinenunabhängig. Ein Beispiel für eine höhere Programmiersprache ist C oder C++.

Bei der 4GL handelt es sich um eine applikative, d.h. anwendungsbezogene, Programmiersprache. Damit ist man in der Lage komplexe Operationen zu lösen. Die letzte Generation, der VHLL, dient zur Beschreibung von Sachverhalten und findet vor allem bei künstlicher Intelligenz Anwendung. ^[18]

3. Anforderungsanalyse

Im Folgenden sollen einige Wettbewerbe speziell für von Schülern gebauten Robotern ansehen. Dies soll die Formulierung der Anforderungen an den Roboter ermöglichen. Dabei wird auf den RoboSAX Wettbewerb, die WRO und die FLL eingegangen. Zusätzlich zu diesen drei Wettbewerben möchte ich noch das Schülerpraktikum an der TU Chemnitz mit in meine Betrachtung einbeziehen.

3.1 RoboSAX

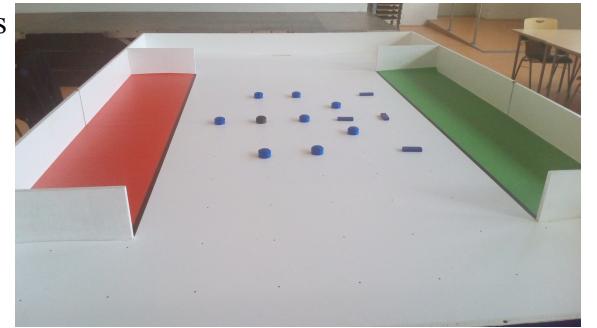
RoboSAX ist ein sächsischer Roboterwettbewerb für Schüler, welcher alljährlich von der Robotik AG des Johannes-Kepler-Gymnasiums veranstaltet wird. Dabei wird jedes Jahr eine andere Aufgabe an die teilnehmenden Schülerteams gestellt. Diese Aufgaben müssen sie auf dem zwei mal drei Meter großen Feld innerhalb von drei Minuten bewältigen. Sollte aufgrund fehlerhafter Programmierung oder ähnlichem ein Handeingriff von Nöten sein, so bringt dies je nach Stärke des Eingriffs Punktabzug. Auch werden nur Drehungen um die eigene Achse oder das Betätigen eines Tasters oder Schalters erlaubt. Dadurch wird ausgeschlossen, dass der Handeingriff Teil der vom Team ausgearbeiteten Strategie ist.

Die Aufgabe des Jahres 2016 nannte sich “Rettet die Schafe”. Dabei mussten die meist selbstgebauten Roboter blaue Objekte - die Schafe - in eine bestimmte Ablagezone verschieben. Je nach Anzahl der verschobenen Objekte wurden Punkte vergeben.

Ergänzend zur Aufgabenstellung des Jahres 2016 soll hier auch nochmal kurz auf die des Jahres 2015 eingehen. In dem genannten Jahr mussten möglichst viele Tennisbälle auf die Enden von Sackgassen eines Straßennetzes geschubst werden.

Aus diesen beiden Aufgabenstellungen lassen sich eine Vielzahl von

Anforderungen ableiten, welche ein Schülerroboter besitzen sollte. Zum einen sollte er Bodensensoren zur Erkennung von Ablagezonen oder sonstigen Straßen ähnlichen Markierungen auf dem Boden besitzen. Zum anderen auch Sensoren zur Erkennung von Hindernissen und Wänden wie Bumper oder Distanzsensoren. Auch Radencoder sind als Erweiterung vorstellbar, um z.B. die momentane Position des Roboters bestimmen zu können. ^{[22][23][24][25]}



16-a: Spielfeld des RoboSAX Wettbewerbs 2016.

3.2 World Robot Olympiad

Bei der WRO handelt es sich um einen internationalen Wettbewerb für Schüler von 8 bis 19 Jahren, um diese für naturwissenschaftlichen Fächer sowie für Ingenieurberufe oder IT-Berufe zu begeistern. Aufgabe ist es, einen Roboter mit dem LEGO MINDSTORMS zu bauen und zu programmieren.

Die Teams bestehen aus je zwei bis drei Schülern, zudem besitzt jedes Team einen Coach. Der Wettbewerb besteht aus drei unterschiedlichen Kategorien: Regular Category, Open Category und Football Category. Während sich bei den beiden ersten Kategorien die Aufgabenstellungen jedes Jahr ändern bleibt diese bei der Football Category immer gleich. ^[36]

Bei der Regular Category müssen die Teams mit ihren selbstgebauten Robotern verschiedene Aufgaben auf seinem 2,5m x 1,3m großem Spielfeld lösen. Besonderheit dabei ist, dass die Teams zwar mit selbstgebauten Robotern ihre Programmierung testen dürfen, zum Wettbewerb jedoch muss jedes Team einen neuen Roboter konstruieren. Im Jahr 2015 mussten die Roboter zum Beispiel verschiedenfarbige Steine nach einem bestimmten Schema einsammeln. Dabei gab die Farbe des Untergrundes, auf welchem

sich der Stein befand, den Ort des nächsten Stein bekannt. Von einem schwarzen Ablageplatz hingegen durfte kein Stein aufgesammelt werden. [39]

In der Open Category hieß die Aufgabe des Jahres 2014 “Roboter im Weltall”. Neben der Roboterentwicklung muss jedes Team zusätzlich eine Dokumentation und Plakate anfertigen. Der Roboter darf hier auch aus Lego-fremden Materialien erbaut werden und wird am Ende von einer Jury bewertet. [38]

Aufgabe in der Football Category ist es, Roboter zu bauen, welche Fußball spielen sollen. Dazu konstruieren die Teams zwei Roboter vollständig aus Lego, welche sie dann für das Fußballspiel programmieren sollen. Hierbei werden Infrarotsensoren zur Erkennung des Balls und Kompasssensoren zur Orientierung auf dem Spielfeld genutzt. [37]

Bei allen drei Kategorien zeigt sich das Antriebskonzept des Differenzialantriebes aufgrund der Vorteile bei der Navigation als am Idealsten. Auch Bodensensoren bieten sich zumindest bei den ersten beiden Kategorien an. Die in der letzten Kategorie geforderten Infrarot- und Kompasssensoren sind eher als Erweiterung zu erachten, da diese nur in speziellen Fällen eingesetzt werden. Zudem lässt sich mit Radencodern genauso wie beim RoboSAX die Position des Roboters auf dem Spielfeld schätzen.

3.3 First Lego League

Auch in der FLL ist es das oberste Ziel, Kinder und Jugendliche im Alter von 10 bis 16 an die Wissenschaft und Technik heranzuführen und sie dazu zu begeistern.

Jedes Team hat acht Wochen Zeit, sich mit der Aufgabe auseinanderzusetzen, einen passenden Roboter zu konstruieren und diesen dann zu programmieren. Die Roboter werden mit dem LEGO MINDSTORMS gebaut. Der Wettbewerb besteht aus mehreren Durchläufen, bei welchen die Teams zweieinhalb Minuten Zeit haben, möglichst viele der gestellten Aufgaben zu lösen. Dabei wird von Schiedsrichtern auf die Einhaltung der Regeln geachtet und der Roboter bewertet.

Das Thema des Wettbewerbs 2015/2016 hieß “Wege zur Bewältigung unseres Abfalls”. Hier war es Aufgabe der Teams, einen Roboter zu konstruieren, welcher unter anderem Gebäude einreißen und den dabei anfallenden Müll entsorgen sollte oder Autos recyceln sollten oder angefallenen Müll sortieren mussten. Natürlich bestand auch der Müll, das Auto und das Gebäude vollständig aus Lego. Das 3,5 m² große Spielfeld bestand aus

einer farbig bedruckten Matte, nur kleinere Abschnitte wurden durch schwarze Linien begrenzt, in welchem Objekte abgelegt oder abgeholt werden mussten. Erneut zeigt sich auch bei diesem Wettbewerb das Antriebskonzept des Differenzialantriebs als vorteilhaft. Auch Abstandssensoren sind von Vorteil, um nicht an der Spielfeldbegrenzung hängen zu bleiben und die Geräte, mit welchen man auf dem Spielfeld interagiert, zu erkennen. Des Weiteren sollte der Roboter über aufgabenspezifische Anbauten verfügen, welche ihm z.B. das Abreißen eines Gebäudes oder den Abtransport der Teile ermöglichen. Da diese Anbauten jedoch sehr spezifisch sind wird ihnen keine weitere Beachtung geschenkt. ^{[4][5][6][7][8]}

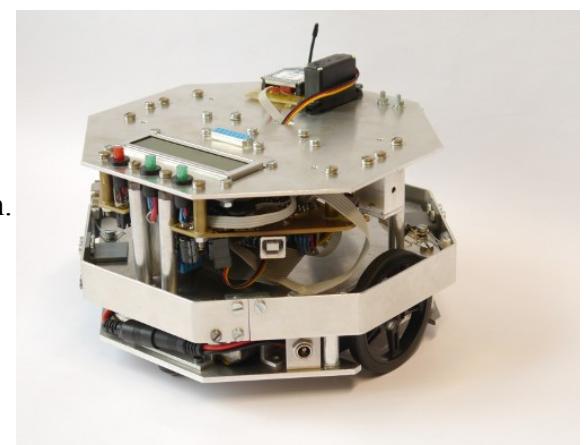
3.4 Schülerpraktikum an der TU-Chemnitz

Ergänzend zu den drei Wettbewerben soll nun noch kurz auf das Schülerpraktikum “Autonome Roboter” an der TU Chemnitz eingegangen werden.

Dieses wird für interessierte Schüler von der Fakultät für Elektrotechnik und Informationstechnik der TU Chemnitz angeboten. Dabei versuchen die Schüler genau wie die Studenten einen Roboter autonom durch ein Labyrinth zu navigieren. Hierbei wird ein eigens entwickelter Roboter genutzt. ^[26]

Er hat eine achteckige Form und besitzt zwei angetriebene Räder. Zudem besitzt er Radencoder, einen drehbaren sowie zwei feste SHARP Abstandssensor und

Bodensensoren zur Linienerkennung. Auch mit einem Display, einem USB-Anschluss und einem XBee Funkmodul zum leichteren Ausgeben von Daten ist er ausgerüstet. ^[27]



18-a: Für das Praktikum an der TU Chemnitz genutzte Roboter.

3.5 Abschließende Auswertung

Bei allen hier vorgestellten Wettbewerben und Aufgaben zeigte sich das Antriebskonzept der zwei unabhängig angetriebenen Räder als am Effektivsten. Durch einen solchen Differenzialantrieb ist es dem Roboter möglich, spontane Richtungsänderung zu vollziehen. Ein Beispiel dafür wäre das Drehen auf der Stelle. Daher ist es sinnvoll, den Roboter rund oder zumindest achteckig zu konstruieren, da sonst die Gefahr besteht, dass der Roboter bei einer Drehung an einer Wand oder ein Hindernis hängen bleibt. Hierbei ist es besser, einen eher kleinen Durchmesser zu wählen, um auch durch engere Gänge oder Gassen zu gelangen.

Auch bei den meisten Wettbewerben zeigte sich, dass Bodensensoren zur Erkennung von Linien und Abstandssensoren zur Erkennung von Hindernissen nützlich sind. Des Weiteren zeigt sich in den meisten Einsatzgebieten von derartigen Robotern, dass Radencoder zur Positionsschätzung hilfreich sind.

Als mögliche Zusatzausbauten für spezifische Aufgabenstellungen sind Kompasssensoren und Ultraschallsensoren zu nennen.

Nützlich ist das Xbee Funkmodul, da durch dieses der Roboter mit einem Computer kabellos kommunizieren kann. Beispielsweise kann man dies zum Auslesen von Daten oder zur Fehlersuche nutzen. Zudem kann über dieses Funkmodul auch eine Steuerung des Roboters erfolgen.

4. Umsetzung

4.1 Konzept

Aus den allgemeinen Anforderungen, welche im vorherigen Kapitel zusammengestellt wurden, lassen sich nun die spezifischen Anforderungen für diesen Roboter formulieren.

Der Roboter sollte zum einen mit zwei Motoren und einer Stützkugel, also mit einem Differentialantrieb, ausgestattet werden. Die Motorenachsen liegen hierbei auf einer Geraden. Dabei empfiehlt es sich, eine runde Grundform des Chassis zu wählen, um eine bessere Navigierbarkeit zu gewährleisten.

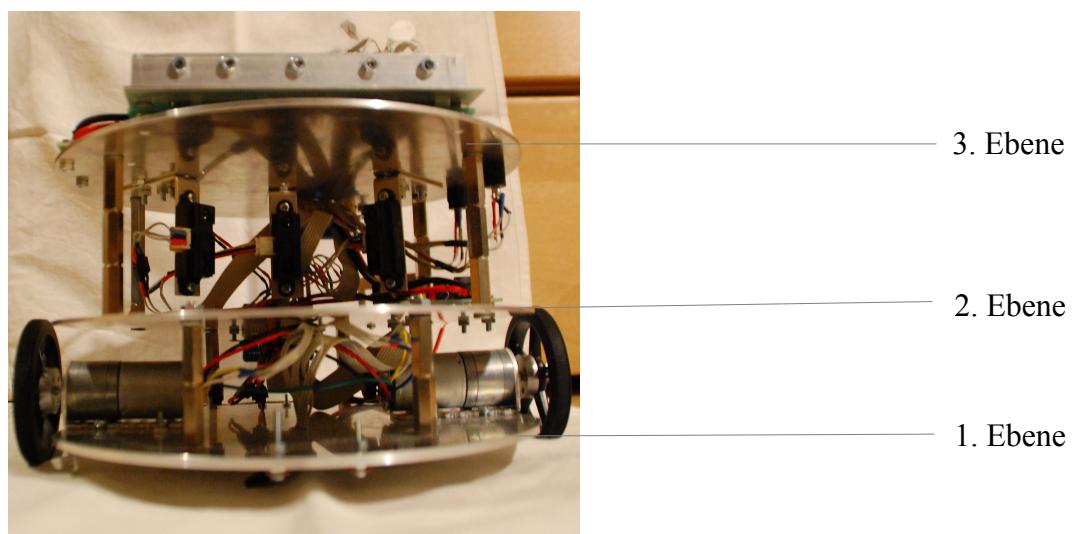
Des Weiteren muss eine einfache mechanische und elektrische Segmentierbarkeit gegeben sein. Dabei ist es wichtig die Sensoren und Motoren durch Steckverbindungen mit dem Controller zu verbinden. Dabei bieten sich D-SUB Verbinder für Datenkabel sowie einfache Steckverbinder für die Sensoren an.

Wie im Kapitel zuvor aufgeschlüsselt sollte der Roboter zudem über eine Grundausstattung an Sensoren verfügen: Boden- und Abstandssensoren sowie Inkrementalgeber für die Räder.

Eine weitere wichtige Anforderung an den Roboter ist die Kommunikation. Diese soll zum einen durch ein XBee - Modul gewährleistet werden, als auch durch einem RS232 Ausgang. Aufgrund der Verwendung eines XBee - Moduls kann auf ein Display verzichtet werden. In der AG wird das XBee - Modul bereits erfolgreich eingesetzt. Außerdem sollte der Roboter über Taster verfügen, die zur direkten Steuerung, z.B. zum Start eines Programms, verwendet werden. Als letztes ist auch eine Akkuladeanzeige ganz sinnvoll, einerseits um den Akku zu schonen, andererseits auch, um eine womöglich zu geringere Spannung als Fehlerquelle auszuschließen.

Der Aufbau des Roboters im Generellen sollte eher robust sein. Es sollten keine empfindlichen Elemente außerhalb des Roboters hervorstehen. Außerdem sollte der Aufbau leicht zu verstehen sein, um eventuelle Erweiterungen zu vereinfachen.

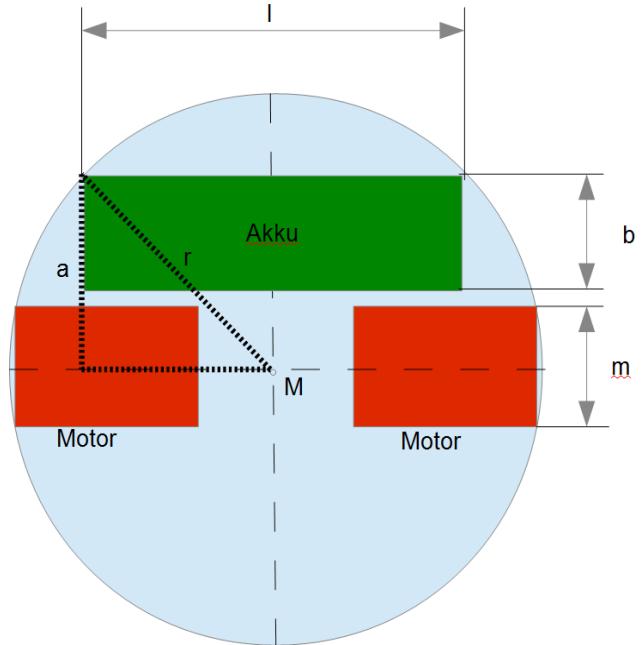
4.2 Chassi



20-a: Front des Prototypen.

Für die Gestaltung des Chassis ist der Durchmesser der Ebenen bedeutend. Dieser muss groß genug sein, dass sich die beiden Motoren berührungslos gegenüberliegen können und der Akkublock Platz findet. Zudem bedarf es mehrerer Stützen, um die beiden untersten Ebenen zu verbinden.

Da der Akku hinter den Motoren liegen soll, ist der Radius über das daraus entstehende Dreieck zu berechnen. Dieses bildet sich aus dem Mittelpunkt sowie einem Berührungs punkt des Akkus mit der Außenseite. Das Akkupack selber hat die Maße von 134 mm in der Länge (l), 46 mm in der Breite (b) und 23 mm in der Höhe. Um den minimalen Durchmesser zu berechnen bedarf es nun noch,



21-a: Schematische Darstellung der 1. Ebene mit den zur Berechnung benötigten Maßen.

wie auf der Skizze zu sehen, der Breite der Motoren (m) mit 25 mm. Zudem wird ein Mindestabstand von 5 mm angenommen, damit der Akku nicht direkt an den Motoren liegt und diese eventuell beschädigt. Da die Motorachsen auf dem Durchmesser der 1. Ebene liegen sollen, bedarf es nur der Hälfte ihrer Breite.

Aus der Addition der Breitenmaße sowie der 5 mm Mindestabstand ergibt sich die erste Kathete zur Berechnung des Radius. Die zweite ergibt sich aus der halben Länge des Akkupacks. Mit Hilfe des Satzes von Pythagoras lässt sich nun die Hypotenuse berechnen.

$$a = \frac{1}{2}m + b + 5\text{mm}$$

$$r = \sqrt{\left(\frac{1}{2}l\right)^2 + a^2}$$

Nun wird der berechnete Radius von 92,31mm noch auf die anderen Anforderungen hin untersucht. Zum einen ob bei der Annahme dieses Radius beide Motoren sich berührungslos gegenüber liegen können. Die Motoren besitzen eine Länge von je 63,5mm. Da ein Motor wesentlich kürzer ist als der angenommene Radius erschließt sich daraus, dass die Motoren sich auf jeden Fall berührungslos gegenüber liegen werden. Zum anderen ob das verwendete Keplerboard auf die dritten Ebene passt. Dabei ist es wichtig, dass die Diagonalen des Boards kleiner sind als der Durchmesser, damit die Platine nicht übersteht. Die Seitenkanten dieses sind 95,2 mm mal 119,8 mm. Auch hier wird wieder der Satz des Pythagoras verwendet:

$$d = \sqrt{95,2^2 + 119,8^2}$$

$$d = 153,02 \text{ mm}$$

Da der Durchmesser größer als die berechneten Diagonalen ist, besteht auch hier kein Problem. Um die Fertigung zu vereinfachen und etwas Toleranz zu gewährleisten wurden die Maße des Radius auf 100 mm aufgerundet.

Die Stützkugel des Roboters sollte möglichst weit hinten an der Grundplatte befestigt werden, damit die drei Auflagepunkte ein möglichst großes Dreieck bilden. Zudem sollte ein Großteil des Gewichts im hinteren Teil lokalisiert sein, um die Haftung nochmals zu

verbessern und zu sichern, dass der Roboter auch bei starker Verzögerung nicht kippt. An der ersten Ebene werden auch die mindestens vier Bodensensoren angebracht. Diese müssen drei Richtungen abdecken - zum einen nach vorn für die klassische Linienverfolgung, zum anderen jeweils zur linken und zur rechten Seite des Roboters für das Labyrinth. Um einer Linie zu folgen bedarf es vorn mindestens zwei Sensoren. An den beiden Seiten reicht je einer, um eine mögliche Kreuzung von Linien wie in einem Labyrinth zu registrieren.

Damit der Roboter die Möglichkeit hat abzubiegen, sollten die beiden seitlichen Sensoren nicht hinter den Motorenachsen befestigt werden. Auch sollte der Abstand



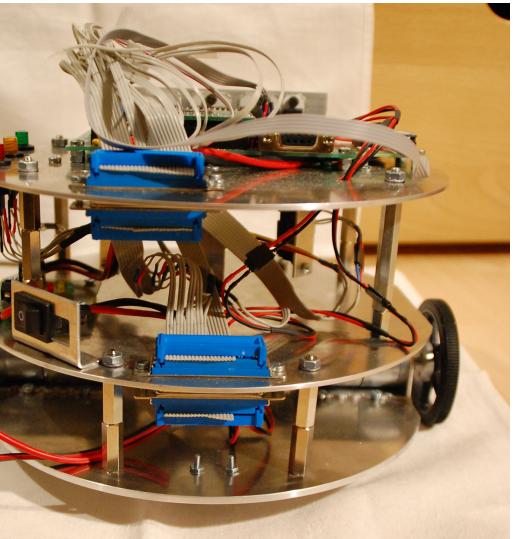
22-a: Bild des Roboters von unten. Gut zu sehen sind die vier roten Bodensensoren, deren Steckkontakte und die Stützkugel.

zum Ebenenmittelpunkt kurz sein, damit eine Kurvenfahrt der Drehung um die eigene Achse entspricht. Ideal wäre es dabei, wenn die Sensoren auf der Achse der beiden Motoren durch den Mittelpunkt lägen. Dies ist jedoch aufgrund der Motorbefestigungen schlecht möglich.

Um die Räder besser zu schützen wurden diese etwas in das Chassi hineingesetzt. Dazu wurde an beiden Seiten der Grundplatte die Rundung auf einer Länge von 60 mm senkrecht zu den Motorenachsen begradiert, sodass die Räder mit einem Durchmesser von jeweils 60 mm nicht allzu stark überstehen.

Die beiden Motoren sind so bemessen, dass sie den Roboter auf mindestens 20 cm/s beschleunigen kann. Dabei ist eine Leerlaufgeschwindigkeit der Motoren von 30 cm/s bis 50 cm/s ideal. Die Motoren von Pololu sind am Besten geeignet, da sie

zum einen mit 53,41 cm/s die Anforderung der Geschwindigkeit erfüllen, zum anderen



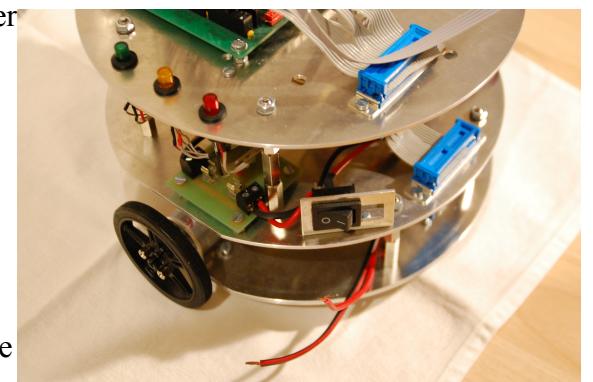
23-a: Blick von Hinten auf den Roboter. Gut zu sehen sind die blauen D-SUB Verbindungskabel.

ist bei diesem Modell der Inkrementalgeber bereits standardmäßig verbaut. Dadurch wird Platz gespart und eine höhere Messgenauigkeit als bei einem selbstgebauten Sensor erzielt.

Damit die erste und zweite Ebene des Roboters stabil verbunden werden kann, bedarf es mindestens drei Stützen, im Falle dieses Roboters vier. Zur Verbindung der Datenkabel wird ein D-SUB Stecker verwendet (siehe Bild 23-a).

Auf der zweiten Ebene ist der Hauptschalter sowie die Akku Ladestandsanzeige untergebracht (siehe Bild 23-b). Der Hauptschalter liegt direkt neben dem Durchgang für das Akkukabel, welches vom Akku kommt. Zudem sollten Position und Winkel derartig gestaltet sein, dass eine komfortable Benutzung gewährleistet ist.

Deshalb befindet sich der Schalter im hinteren Bereich des Roboters leicht schräg am Rand.



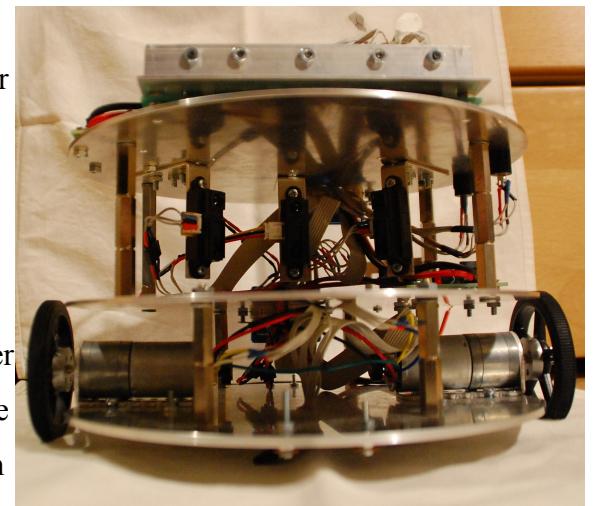
23-b: Bild der Ladenstandsanzeige und des Hauptschalters auf der 2. Ebene.

Bei der Platine zur Überprüfung des Akku-Ladezustands ist es zum einen wichtig, dass die Anzeige - LED gut sichtbar sind und zum anderen, dass man ohne die Demontage der oberen Ebenen an die Feinsicherung heran kommt. Deshalb befindet sich die Platine circa mittig am Rand des Roboters, hinter der Radachse auf der zweiten Ebene. Um den Akku zu schonen ist es des Weiteren wichtig, dass der Schalter im Schaltkreis vor der Platine kommt. Somit kann im ausgeschalteten Zustand kein Strom über den Ladestandsanzeiger fließen. Deshalb ist die gewählte Position direkt vor dem Schalter. Insgesamt dürfen diese beiden Elemente, Schalter und Platine, keine Behinderung für die drei SHARP Sensoren darstellen, welche an der Unterseite der dritten Ebene befestigt sind. Zudem wurden die Datenkabel in den hinteren Bereich verlegt und auch dort zum nächsten D-SUB Stecker geführt. Außerdem besitzt jeder Sensor eine weitere eigene Steckverbindung, um seinen Austausch zu vereinfachen.

Damit ein möglichst großer Winkel von den Abstandssensoren abgedeckt wird, wurden drei Sensoren genutzt, wobei einer direkt nach vorne misst, während die anderen beiden jeweils in einem Winkel von je 45° von diesem abstehen.

Da die Sensoren einen Messbereich von 100mm bis 800mm haben, sollten sie etwa 100mm in den Roboter herein gesetzt werden, damit der Roboter auch Gegenstände direkt vor sich wahrnehmen kann. Durch eine Vielzahl von Befestigungslöchern ist zudem eine Variabilität der Position der Befestigung der SHARP Sensoren gegeben. Auf der dritten und letzten Ebene sind sowohl drei Taster, als auch der Controller und ein XBee-Modul untergebracht. Die Platine des Mikrocontrollers ist so auf die Ebene gesetzt, dass am hinteren Ende einiges an Platz für den D-SUB Stecker der Datenkabel bleibt. Die vorderen Ecken des Kühlblechs für die Motortreiber schließen dabei nahezu bündig mit der Außenkante der Ebene ab.

Der RS232-Anschluss zeigt nach hinten, sodass der Roboter komfortabel angeschlossen werden kann. Zudem sind die Steckkontakte des Mikrocontrollerboards gut erreichbar. Kurzfristige Änderungen an der Verbindung zu den Sensoren oder Motoren können so gut vorgenommen werden. Auch für die gewollte Möglichkeit der Erweiterungen sind



24-a: Die drei SHARP Sensoren. Im Hintergrund sieht man die Datenkabel.

die Steckleisten des Boards gut zu erreichen.

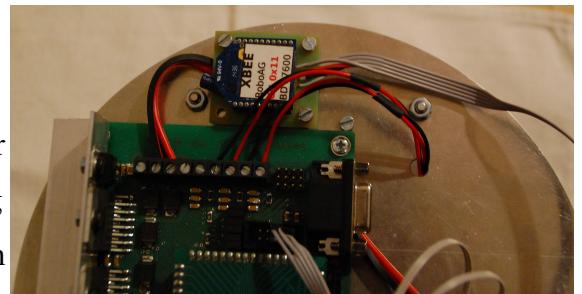
Die drei Taster sollten ebenfalls gut erreichbar liegen, weswegen sie hintereinander direkt neben das Board gesetzt wurden. Da die Taster beleuchtet sind können sie auch als weitere visuelle Ausgabe dienen. Wenn ein fertiges Programm auf den Roboter übertragen wird und dieses sofort startet können Schäden an Roboter, Übertragungskabel und Computer entstehen. Dem soll durch das Starten des Programms durch die Taster entgegengewirkt werden.

Zuletzt ist auch das X-Bee Modul auf dieser Ebene befestigt. Dieses sollte zur Sicherung einer guten Übertragung nicht zwischen den verschiedenen Ebenen liegen, da die Metallplatten das Modul abschirmen könnten und somit die Übertragung verschlechtern könnten.

Auch sind die Wege, wenn es direkt neben des Keplerboards liegt, sehr gering und der Kabelaufwand somit auch in Grenzen gehalten.



25-a: Blick auf die 3. Ebene mit Keplerboard (Mitte), Xbee Modul (Rechts) und Tastern (Links).



25-b: Blick von oben auf das Xbee Modul neben dem Keplerboard.

4.3 Ergebnis und Aussichten

Der Roboter ist erfolgreich zusammengebaut wurden. Auch wurden schon erste Testfahrten unternommen, um die Funktionsfähigkeit zu prüfen. Das Programm dazu befindet sich im Anhang.

Auch gibt es eine Auflistung der benötigten Teile sowie die genauen Konstruktionspläne, um das Bauen weiterer Roboter zu ermöglichen. Beides wurde der Arbeit ebenfalls angehängt. Alle geforderten Sensoren konnten untergebracht werden und auch das Funkmodul fand Platz.

Nach erfolgreichem Aufbau des Roboters lassen sich noch einige Dinge für weiterführende Arbeiten sagen:

Zum einen ließen sich weitere Beispielprogramme verfassen. Dazu sollten entsprechende Arbeitsblätter gestaltet werden, welche in das Thema einführen und das Erlernen der Programmiersprachen vereinfachen.

Zum Anderen könnten als Erweiterungen Ultraschallsensoren verbaut und angeschlossen werden.

Als weitere Erweiterung wäre auch eine 4. Display-Ebene denkbar. Momentan ist das XBee - Modul erstmal völlig ausreichend, für künftige Anwendungen kann aber auch Display praktisch sein.

Außerdem könnten zusätzliche Löcher in die Ebenenplatten gebohrt werden, um Erweiterungen einfacher zu befestigen.

Zudem könnte ein weiterer Mikrocontroller oder ein Sensor über UART, der RS232 Schnittstelle, angebunden werden.

5. Selbstständigkeitserklärung

Hiermit erkläre ich Roth, Richard, dass ich die vorliegende Arbeit selbstständig und ohne Benutzung anderer als der angegebenen Quellen angefertigt habe. Folgende Personen haben mich unterstützt: M.Sc. Weissig, Peter, Herr Scheler, Andreas. Die Arbeit hat in gleicher oder ähnlicher Form noch keiner anderen Prüfungsbehörde vorgelegen.

Datum

Unterschrift

6. Danksagung

Hiermit möchte ich mich vor allem bei meinen Betreuern M.Sc. Peter Weissig und Herrn Andreas Scheler für ihre Hilfe bedanken. Weiterhin danke ich allen Freunden und Bekannten für ihre Unterstützung.

Literaturverzeichnis

- 1 ATMEL mega64 Datenblatt. URL: http://www.atmel.com/Images/Atmel-2490-8-bit-AVR-Microcontroller-ATmega64-L_datasheet.pdf [01.12.2016]
- 2 Bodensensoren. URL: <https://www.pololu.com/product/2459> [21.12.2016]
- 3 Distanzsensoren. URL: <http://wiki.ctbot.de/index.php/Distanzsensoren> [24.10.2016]
- 4 FLL Allgemeines. URL: <http://www.first-lego-league.org/de/allgemeines/was-ist-fll.htm> [14.01.2016]
- 5 FLL Aufgabenstellung. URL: <http://www.first-lego-league.org/de/fll/robot-game/aufgaben.html> [20.01.2016]
- 6 FLL Regeln. URL: <http://www.first-lego-league.org/de/fll/robot-game/regeln.htm> [20.01.2016]
- 7 FLL Übersicht. URL: <http://www.first-lego-league.org/de/2015.html> [20.01.2016]
- 8 FLL Wettbewerbsinformationen. URL: http://www.first-lego-league.org/de/allgemeines/was-ist-fll.html?file=files/inhalt/fll_de/allgemeines/was_ist_fll/FLL_2015_Wettbewerbsinfos%20ausfuehrlich.pdf [14.01.2016]
- 9 Inkrementalgeber Allgemein. URL: http://www.imc-berlin.de/fileadmin/Public/Downloads/Whitepapers/WP_Inkrementalgeber_I.pdf [18.10.2016]
- 10 Inkrementalgeber Pololu. URL: <https://www.pololu.com/product/3542> [18.10.2016]
- 11 Inkrementalgeber Pololu 2. URL: <https://www.pololu.com/product/2284> [18.10.2016]
- 12 Inkrementalgeber Übersicht. URL: http://www.uni-ulm.de/fileadmin/website_uni_ulm/iui.inst.130/Arbeitsgruppen/Robotics/Robotik/Vorlesung_04.pdf [18.10.2016]
- 13 Mikrocontroller Allgemeines. URL: <http://einstieger.myavr.de/index.php?id=5> [30.10.2016]
- 14 Mikrocontroller Allgemeines 2. URL: <http://cclassic.users.sourceforge.net/downloads/doc/Mikrocontroller/Mikrocontroller.pdf> [30.10.2016]
- 15 Mikrocontroller Allgemeines 3. URL: https://fsinfo.cs.tu-dortmund.de/_media/ags/kolloquien/kollag_20151105_microcontroller-ag.pdf [30.10.2016]
- 16 Mikrocontroller Grundlagen. URL: <http://www.mikrocontroller.net/articles/Kategorie:Grundlagen> [30.10.2016]

- 17 Mikrocontroller Technik. URL: http://www.controllersandpcs.de/lehrarchiv/pdfs/automatisierungstechnik/MC_MS_01.pdf [30.10.2016]
- 18 Programmierung. URL: <http://www2.informatik.uni-halle.de/lehre/c/c12.html> [01.12.2016]
- 19 Radencoder Allgemein. URL: <https://www.posital.com/de/produkte/absolute-drehgeber/absolute-vs-incremental.php> [18.10.2016]
- 20 Radencoder Pololu. URL: <https://www.pololu.com/category/201/encoders> [18.10.2016]
- 21 Radencoder Pololu Datenblatt. URL: <https://www.pololu.com/file/0J1210/romi-encoder-dimension-diagram.pdf> [18.10.2016]
- 22 RoboSAX Allgemeines. URL: <http://www.robosax.de/index.php/ueber-robosax> [10.01.2016]
- 23 RoboSAX Aufgabenstellung. URL: <http://www.robosax.de/index.php/aktuelle-aufgabe> [10.01.2016]
- 24 RoboSAX Regelwerk 2015. URL: <http://www.robosax.de/data/2015/Regelwerk.pdf> [10.01.2016]
- 25 RoboSAX Regelwerk 2016. URL: <http://www.robosax.de/data/2016/Regelwerk.pdf> [10.01.2016]
- 26 Schülerpraktikum Allgemeines. URL: <https://www.tu-chemnitz.de/etit/proaut/lehre/schuelerpraktikum.html> [21.01.2016]
- 27 Schülerpraktikum Roboter. URL: <https://www.tu-chemnitz.de/etit/proaut/lehre/praktikumMobileRoboter/Doku/hardware.html> [21.01.2016]
- 28 Sensoren Allgemein. URL: https://www.uni-ulm.de/fileadmin/website_uni_ulm/iui.inst.130/Mitarbeiter/oubbati/RobotikWS1113/Folien/Sensoren.pdf [18.10.2016]
- 29 Sensoren Technik. URL: http://files.pepperl-fuchs.com/selector_files/navi/productInfo/doct/tdoct1797a_ger.pdf [18.10.2016]
- 30 Sensoren Technik 2. URL: http://www.eah-jena.de/~ploss/LEHRE/PDF/SENSOR_4.PDF [18.10.2016]
- 31 SHARP Datenblatt. URL: http://www.sharpsma.com/webfm_send/1489 [24.10.2016]
- 32 SHARP Distanzsensoren. URL: http://rn-wissen.de/wiki/index.php/Sensorarten#Sharp_Infrarotsensoren [24.10.2016]
- 33 Ultraschallsensor. URL: <http://www.robot-electronics.co.uk/htm/srf02tech.htm>

[21.12.2016]

34 Ultraschallsensor Technik. URL: <http://www.exp-tech.de/srf02-ultrasonic-ranger>

[21.12.2016]

35 UART. URL: <http://users.etch.haw-hamburg.de/users/schwarz/>

En/Lecture/Pej/Notes/Chap1.pdf [02.12.2016]

36 WRO Allgemeines. URL: <http://worldrobotolympiad.de/>

wro-allgemeine-informationen [11.01.2016]

37 WRO Football Category. URL: <http://worldrobotolympiad.de/wro-football-category>

[14.01.2016]

38 WRO Open Category. URL: <http://worldrobotolympiad.de/wro-open-category>

[11.01.2016]

39 WRO Regular Category. URL: <http://worldrobotolympiad.de/wro-regular-category>

[11.01.2016]

Bildquellen

- 6b <https://www.pololu.com/product/2458/pictures#lightbox-picture0J630> [03.01.2017]
- 7a <http://www.imc-berlin.de/fileadmin/Public/Downloads/Whitepapers/>
WP_Inkrementalgeber_I.pdf [02.01.2017]
- 8a <http://www.imc-berlin.de/fileadmin/Public/Downloads/Whitepapers/>
WP_Inkrementalgeber_I.pdf [02.01.2017]
- 8b <https://upload.wikimedia.org/wikipedia/commons/thumb/d/d5/Halleffekt.svg/2000px-Halleffekt.svg.png> [02.01.2017]
- 8c <https://www.pololu.com/product/2284/pictures#lightbox-picture0J3798> [02.01.2017]
- 9a <https://www.pololu.com/product/2284/pictures#lightbox-picture0J3481> [02.01.2017]
- 9b https://www.uni-ulm.de/fileadmin/website_uni_ulm/iui.inst.130/Mitarbeiter/oubbatti/RobotikWS1113/Folien/Sensoren.pdf [02.01.2017]
- 9c https://www.uni-ulm.de/fileadmin/website_uni_ulm/iui.inst.130/Mitarbeiter/oubbatti/RobotikWS1113/Folien/Sensoren.pdf [02.01.2017]
- 10a [https://cdn.sparkfun.com//assets/parts/1/8/4/00242-1.jpg](https://cdn.sparkfun.com/assets/parts/1/8/4/00242-1.jpg) [02.01.2017]
- 10b <http://www.exp-tech.de/media/catalog/product/cache/1/image/600x600/9df78eab33525d08d6e5fb8d27136e95/s/r/srf02c.png> [02.01.2017]
- 11a <http://www.robot-electronics.co.uk/htm/srf02tech.htm> [02.01.2017]
- 12a http://www.atmel.com/Images/Atmel-2490-8-bit-AVR-Microcontroller-ATmega64-L_datasheet.pdf
[02.01.2017]
- 14a <http://www.futurlec.com/Pictures/ATMEGA64-16AI.jpg> [02.01.2017]
- 18a <https://www.tu-chemnitz.de/etit/proaut/lehre/rsr/robong.jpg> [02.01.2016]

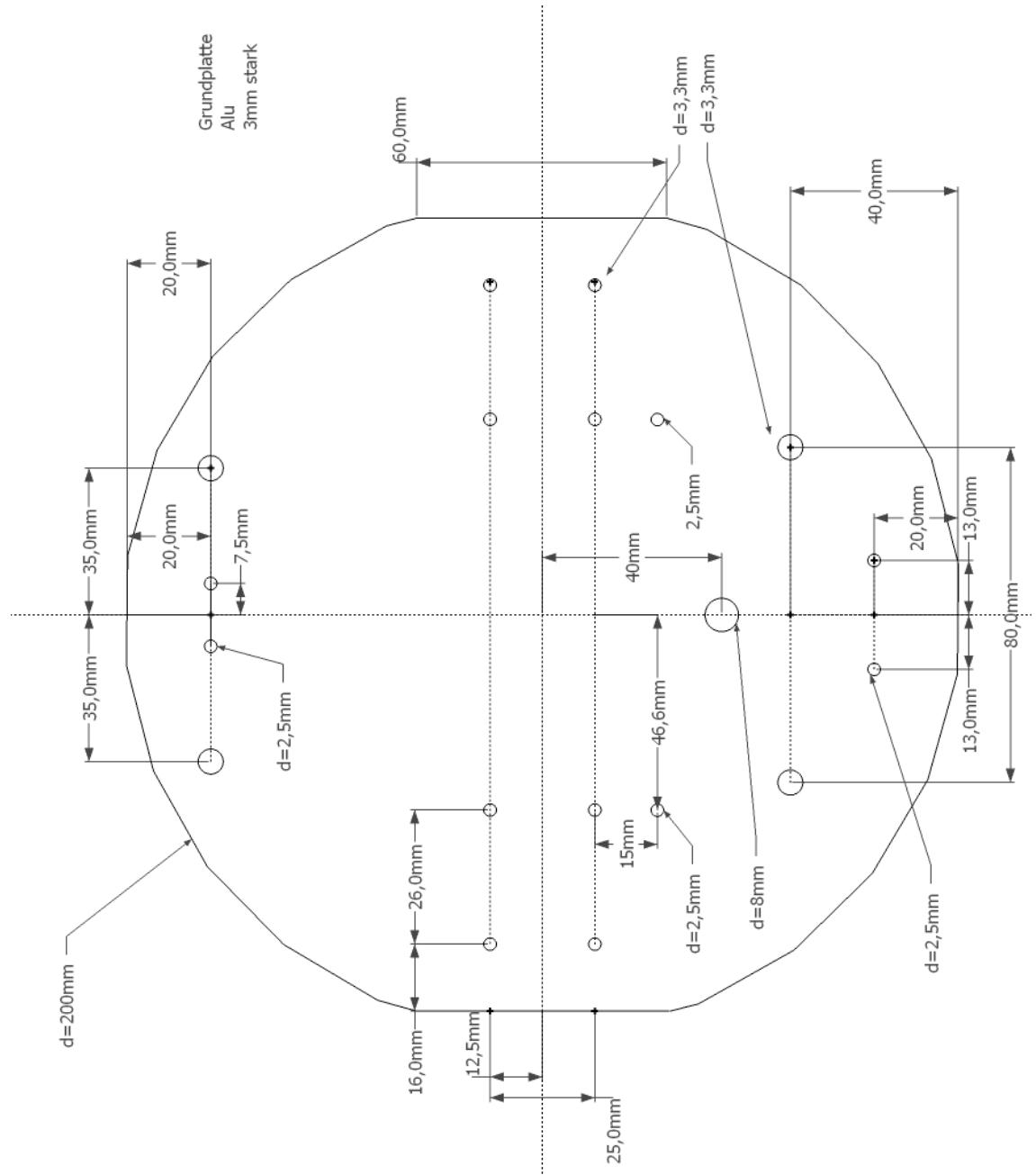
Anhang

I. Stückliste

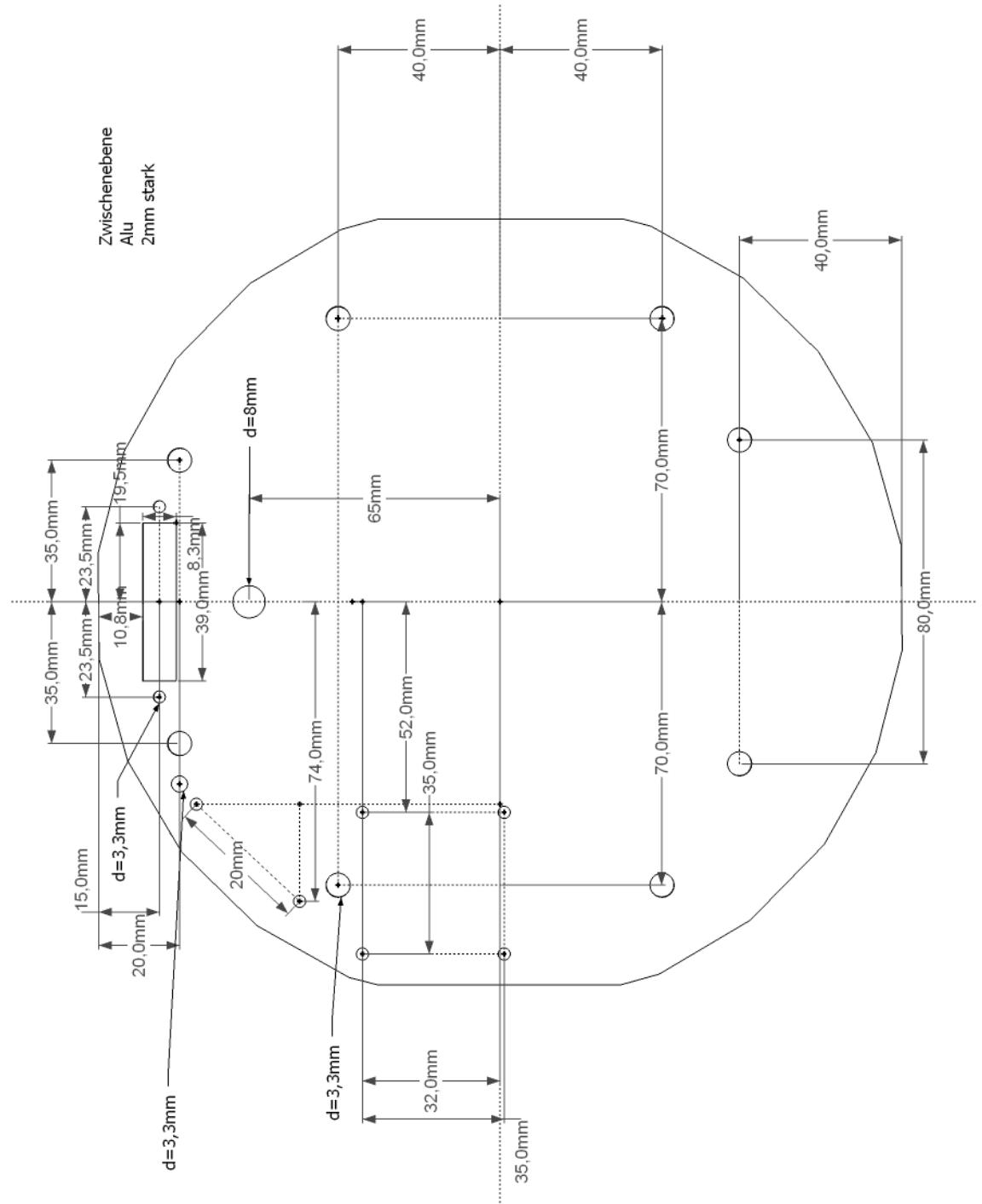
| Bezeichnung | Nummer | Einzelpreis | Anzahl | Händler |
|----------------------------|----------------|-------------|--------|----------|
| Schraube | M2x10 | - | 6 | - |
| Schraube | M3x6 | - | 8 | - |
| Schraube | M3x10 | - | 32 | - |
| Schraube | M3x15 | - | 12 | - |
| Mutter | M2 | - | 14 | - |
| Mutter | M3 | - | 70 | - |
| Distanzhülsen | DA 20MM | 0,16€ | 20 | Reichelt |
| D-SUB Buchse | D-SUB BU 25FB | 0,67€ | 2 | Reichelt |
| D-SUB Stecker | D-SUB ST 25FB | 0,67€ | 2 | Reichelt |
| Taster grün | 701236 | 2,79€ | 1 | Conrad |
| Taster rot | 701227 | 2,79€ | 1 | Conrad |
| Taster gelb | 701229 | 2,79€ | 1 | Conrad |
| Schrumpfschlauch schwarz | SDH 1,6 SW | 0,32€ | 30cm | Reichelt |
| Schrumpfschlauch blau | SDH 1,6 BL | 0,32€ | 16cm | Reichelt |
| Schrumpfschlauch rot | SDH 1,6 RT | 0,32€ | 25cm | Reichelt |
| Motor | 2284 | 32,50€ | 2 | Pololu |
| Bodensensor | 2458 | 3,50€ | 5 | Pololu |
| Rad | 1420 | 7,90€ | 2 | Pololu |
| Distanzsensor | GP2-1080 | 5,55€ | 3 | Reichelt |
| Mikrocontroller | Atmega64 | 6,99€ | 1 | Reichelt |
| Mikrocontrollerboard | Keplerboard | ca. 100€ | 1 | - |
| Xbee Modul | ZIGBEE Xbee CA | 31,10€ | 1 | Reichelt |
| Xbee Adapter | - | ca. 17€ | 1 | - |
| Akku Pack | XR6 3600 | 20,50€ | 1 | Reichelt |
| Spannungsüberwachung | - | ca. 20€ | 1 | - |
| Ebene 1 | - | - | 1 | - |
| Ebene 2 | - | - | 1 | - |
| Ebene 3 | - | - | 1 | - |
| Winkel Sensorbefestigung | - | - | 3 | - |
| Winkel Schalterbefestigung | - | - | 1 | - |

II Konstruktionspläne

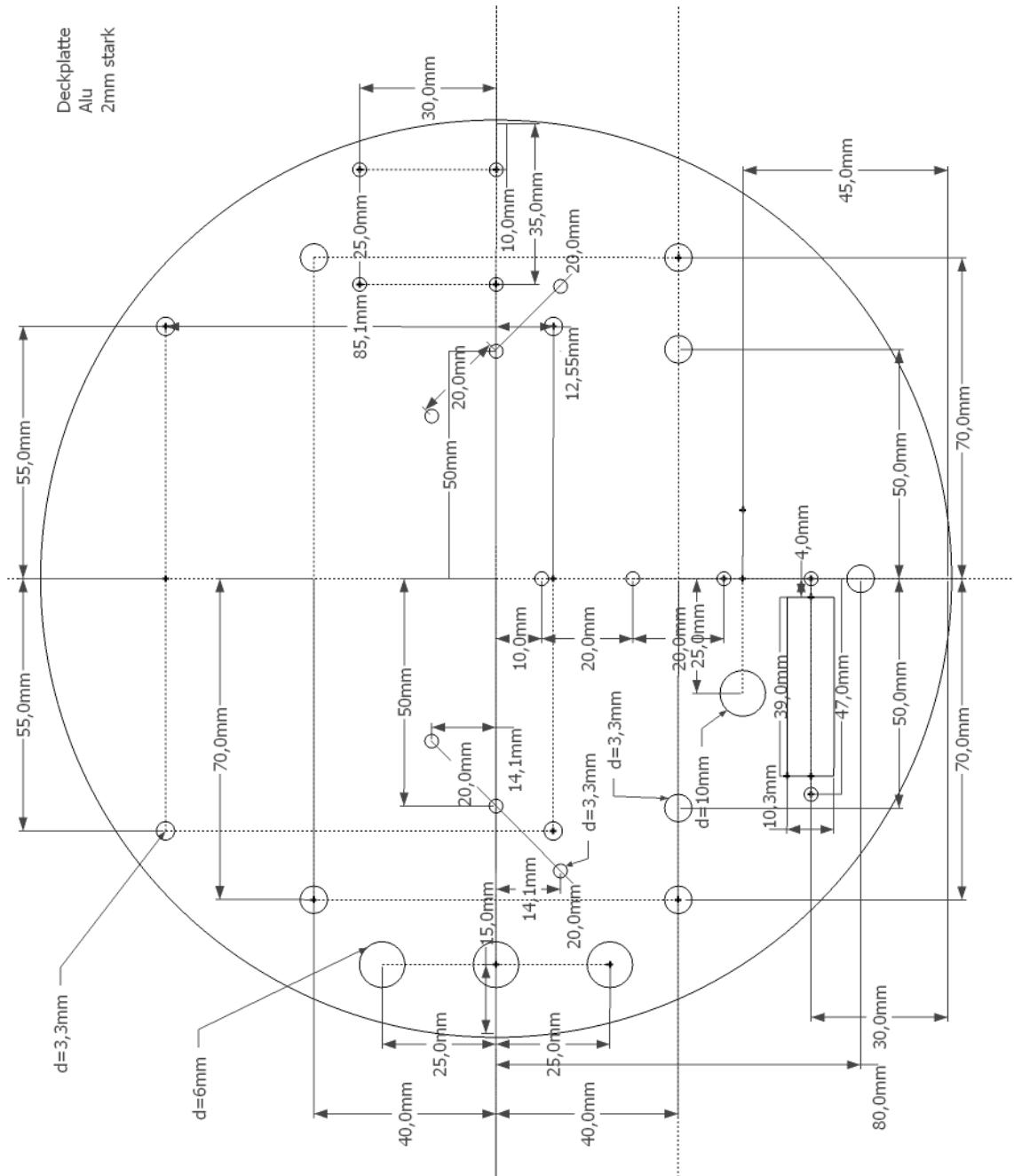
33-a: Ebene 1



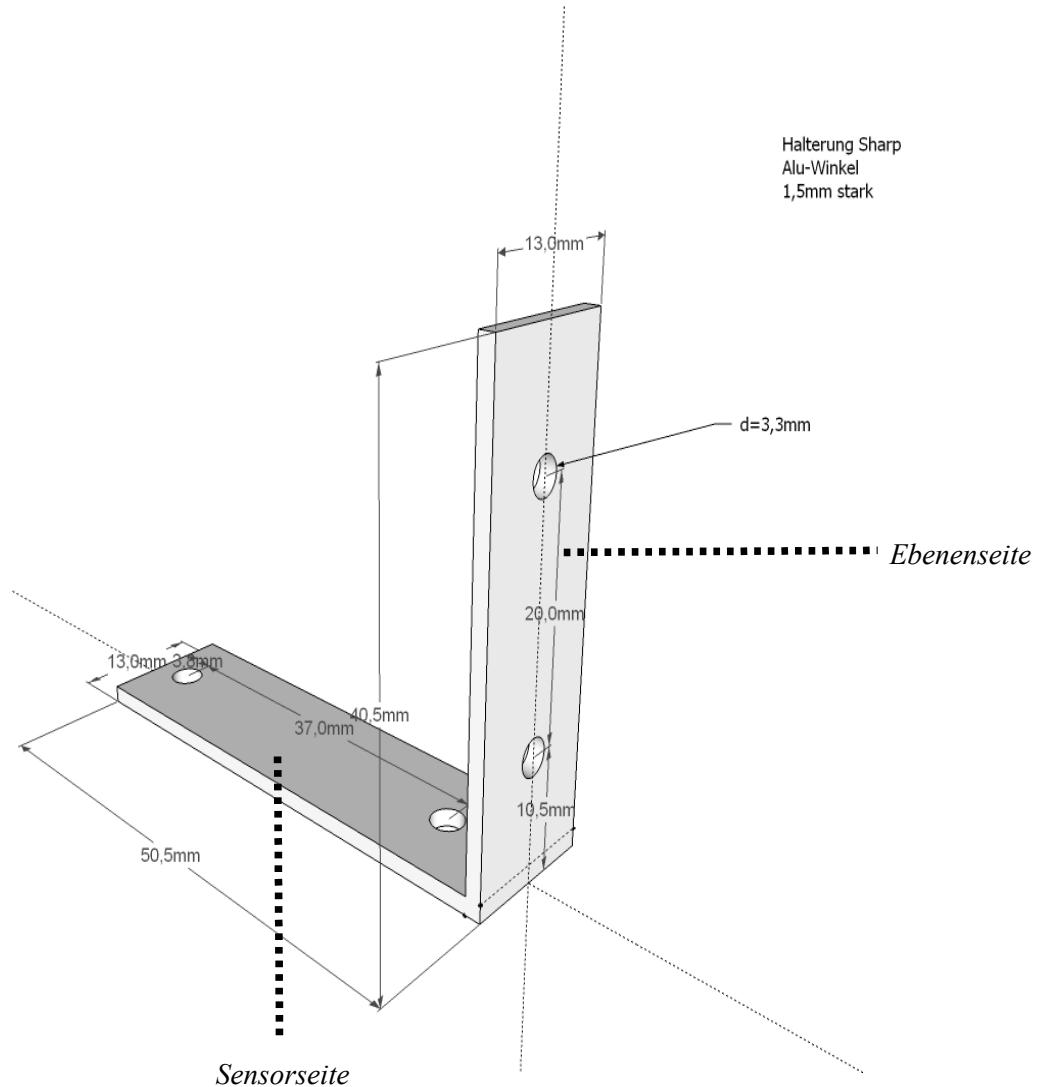
34-a: Ebene 2



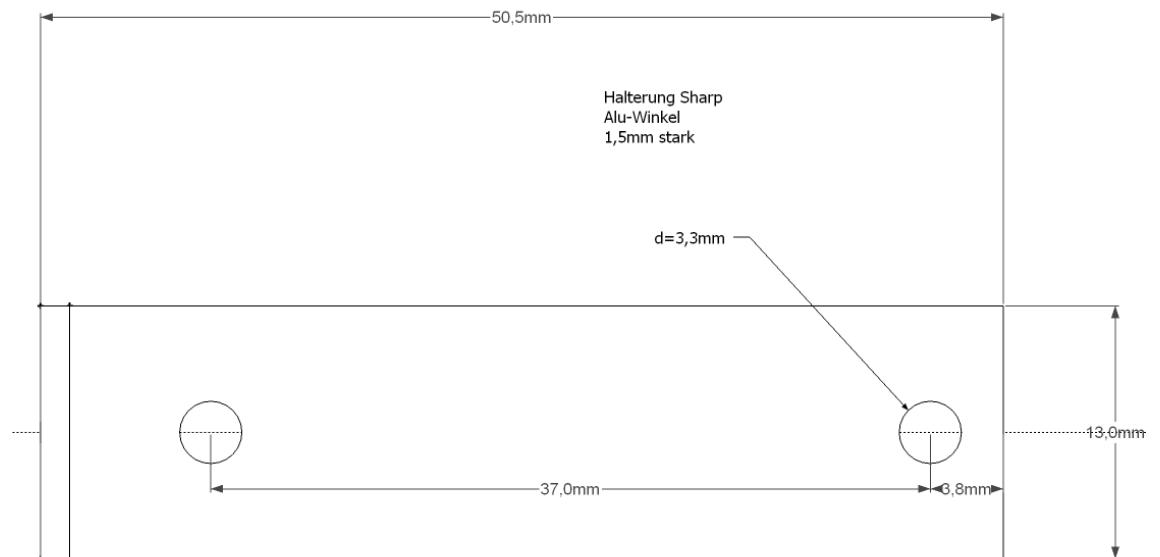
35-a: Ebene 3



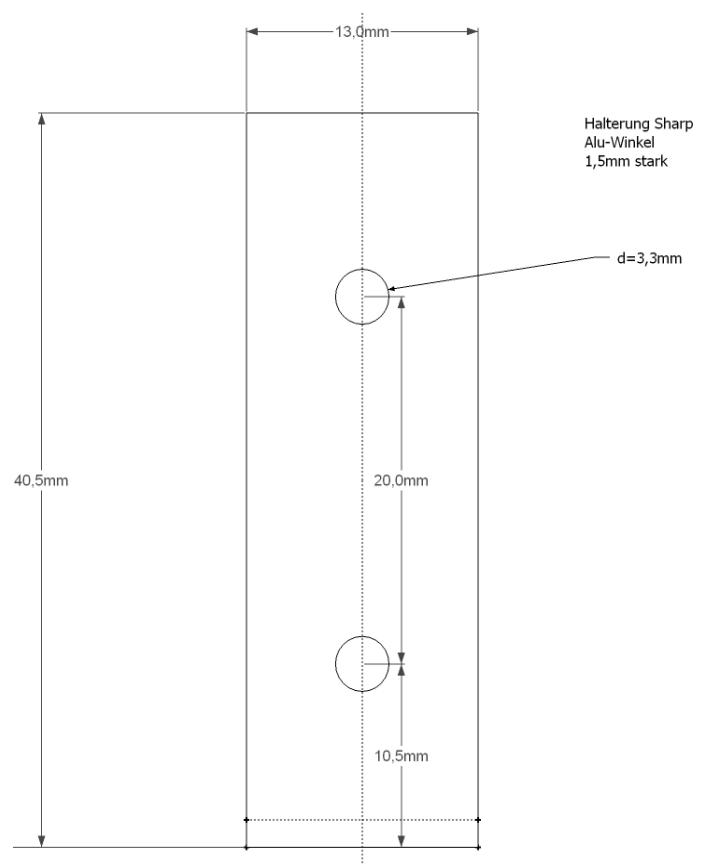
36-a: Winkel zur Befestigung der SHARP-Sensoren. 3D Ansicht.



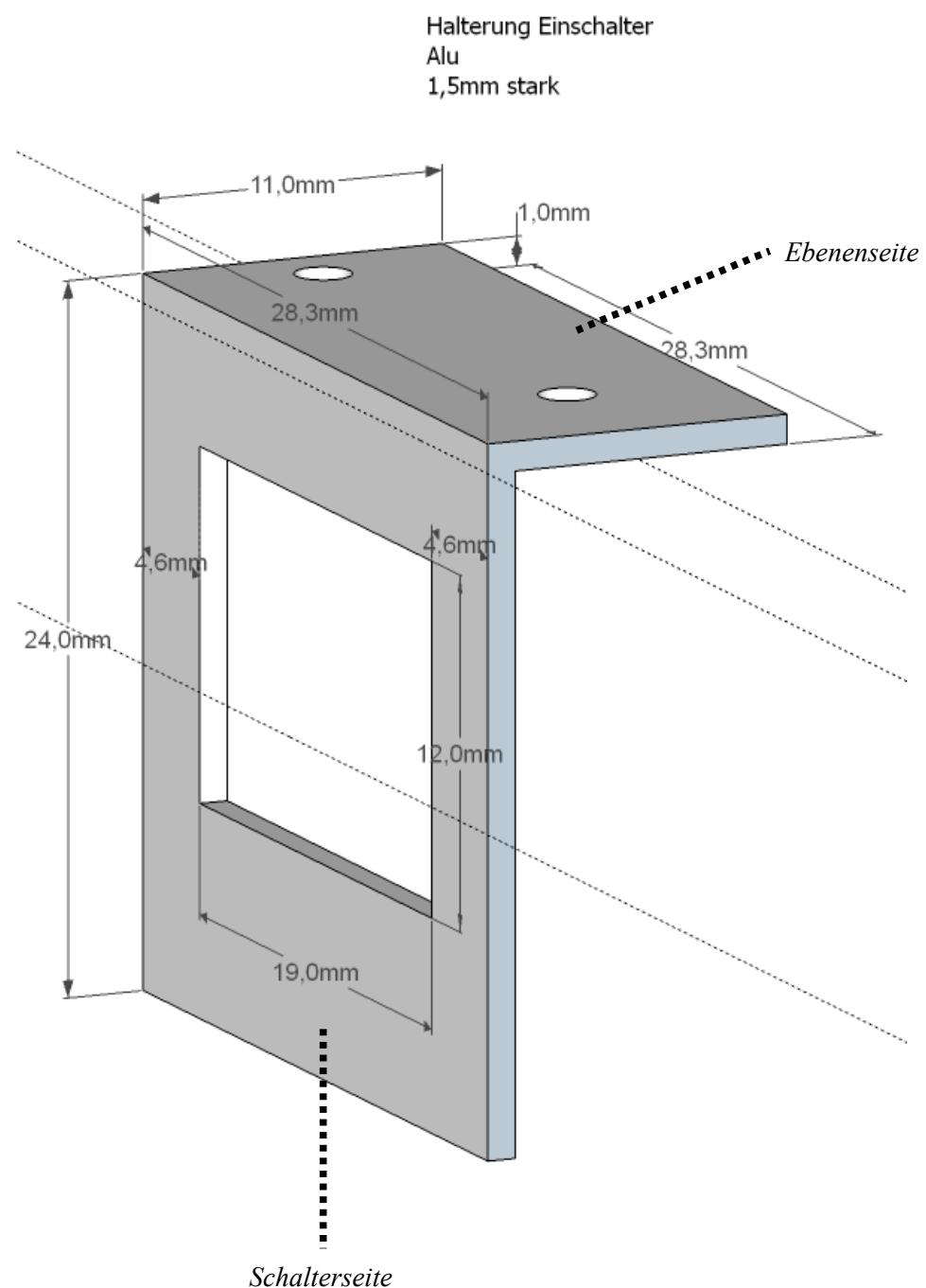
37-a: Winkel zur Befestigung der SHARP-Sensoren. Sensorseite.



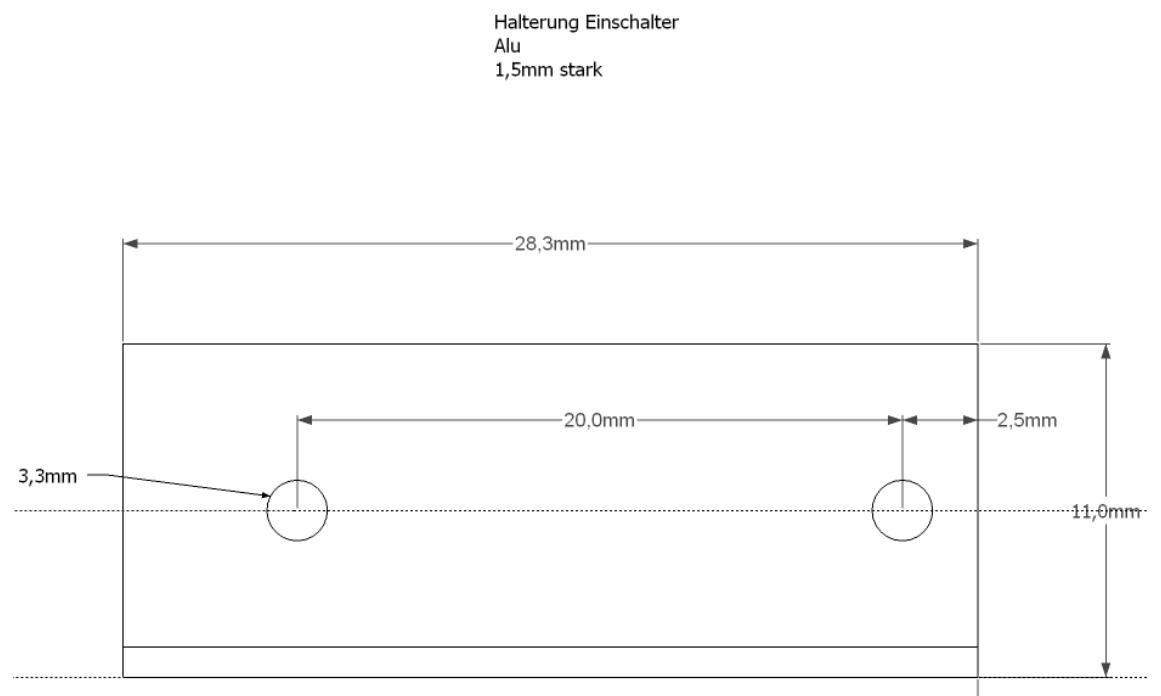
37-b: Winkel zur Befestigung der SHARP-Sensoren. Ebenenseite.



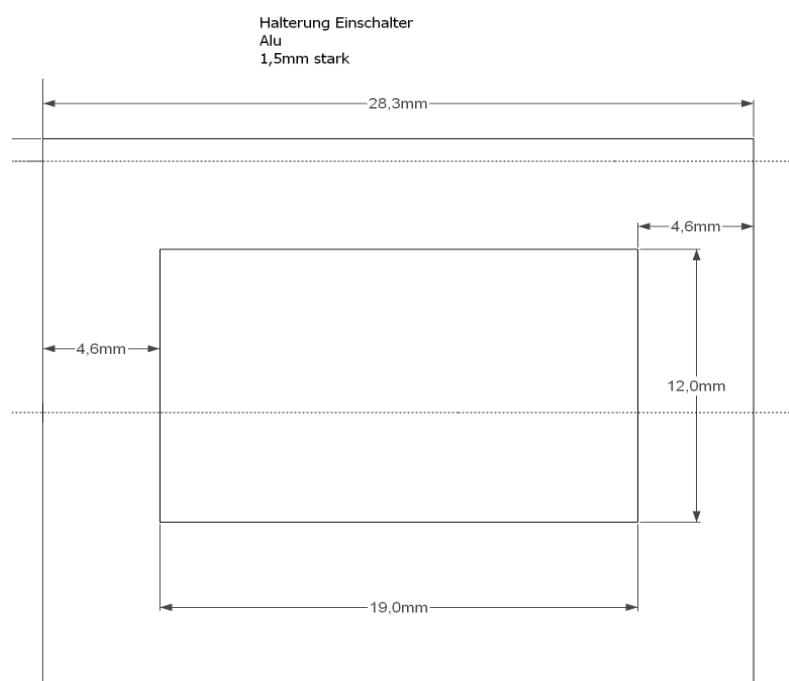
38-a: Winkel zur Befestigung des Hauptschalters. 3D Ansicht.



39-a: Winkel zur Befestigung des Hauptschalters. Ebenenseite



39-b: Winkel zur Befestigung des Hauptschalters. Schalterseite.



III Testprogramm

Dieses Programm steuert die beiden Motoren an und lässt den Roboter einige Sekunden fahren.

```
#include <inttypes.h>
#include "keplerboard.h"

#define M1_SPEED(x)  (x ? (PORTB |= _BV(5)) : (PORTB&= ~_BV(5)))
#define M1_DIR(x)    (x ? (PORTB |= _BV(2)) : (PORTB&= ~_BV(2)))
#define M2_SPEED(x)  (x ? (PORTB |= _BV(6)) : (PORTB&= ~_BV(6)))
#define M2_DIR(x)    (x ? (PORTB |= _BV(3)) : (PORTB&= ~_BV(3)))
#define M3_SPEED(x)  (x ? (PORTB |= _BV(7)) : (PORTB&= ~_BV(7)))
#define M3_DIR(x)    (x ? (PORTB |= _BV(4)) : (PORTB&= ~_BV(4)))
#define MALL_OFF()   (PORTB&= 0x03)

int main (void) {
    system_init();
    int i;
    i=1;
    while (1) {
        if (i<100) {
            M1_SPEED (1);
            M1_DIR (1);
            delay_ms (1000);
            MALL_OFF ();
            i++;
        }
        else {
            M1_SPEED (1);
            M1_DIR (1);
            delay_ms (1);
            MALL_OFF ();
        }
    }
    return (0);
}
```