

# 3D Laserscanner für mobilen Roboter

**Industriearbeit PAIND+E1**

im Auftrag des Industriepartners

**RUAG AG**

an der

Hochschule Luzern Technik & Architektur

im Studiengang Elektrotechnik

**Schwerpunkt**

Signalverarbeitung & Kommunikation,  
Automation & Embedded Systems

**Dozent:** Björn Jensen

**Experte:** Markus Thalmann

**Eingereicht von:** Daniel Zimmermann

**Matrikelnummer:** 15-465-271

**Datum der Abgabe:** 22.12.2017

**Klassifikation:** Rücksprache

# Eigenständigkeitserklärung

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbstständig angefertigt und keine anderen als die angegebenen Hilfsmittel verwendet habe. Sämtliche verwendeten Textausschnitte, Zitate oder Inhalte anderer Verfasser wurden ausdrücklich als solche gekennzeichnet.

Wolfenschiessen, den 22.12.2017

Daniel Zimmermann

# Abstract

This documentation is a result of the project Modul PAIND+E1 at the Lucerne School of Engineering and Architecture for the industry partner RUAG AG.

The following chapters contains the full experiences, results and descriptions during the project from september to dezember 2017. The body of the documentation is subdivided in different phases and reflects the timeline of the Project.

The first part is a summation of the results during the information research phase. It contains the knowledge about the available Sensors, the potential hardware components and the necessary Software to implement the solution after the functional specifications.

There are three concepts created, which have different approaches. The first concept called "plattform" is based on a "regulation experiment", which can turn the plattform in a wide range of angle, while using servo motors. This concept was no longer pursue, because the two other concepts were more suitable.

The two other concepts are based on a turning endlessly "tower". The difference between the two concepts is the position of the signal processing unit. In the unrotated version, the unit are below in a static case. Only the 3D-Sensor is rotating for mapping. In the rotated version, the signal processing unit in the case is also rotating, and only the Interface to the packpot is static.

The main content is about the realisided concept, which is the last called concept before. The realisation phase describes the process, how the case and the electronic parts are mounted. In a seperate topic, it describes, how the Software is implemented and how it works together with the hardware.

In the end a short reflection summarised the largest challenges during the project and how to solve them. It also reflects the project management and give a little outlook.

# Einleitung

Im Forschungszweig der Robotik entstehen ständig neue und revolutionäre Technologien. Dabei steht die Transformation des weitgehend selbstständigen Roboters im Vordergrund. Diverse Pilotprojekte beweisen bereits heute, dass durch eine komplexe Abstimmung hoch präziser Sensoren die kognitiven und sensorischen Fähigkeiten des Menschen nachgeahmt, wenn nicht sogar übertroffen werden können. Ein gutes Beispiel für diese Transformation sind mobile Roboter wie der iRobot Packbot. Durch entsprechende Logik und Sensorik können die geländegängigen Roboter dem Menschen einen enormen Dienst erweisen. In für Menschen unzugängliche oder nur unter hohem Gefahrenpotential begehbarer Orte wie Kriegsgebieten, von Naturkatastrophen geschädigten oder radioaktiv verstrahlten Umgebungen können sie Aufgaben bewältigen, welche dem Menschen alleine unmöglich erscheinen.

Durch die zunehmende Rechenleistung von Computern und den daraus resultierenden Datenmengen entsteht nun auch die Möglichkeiten mittels Robotern detaillierte Modelle der erwähnten Einsatzgebieten zu erstellen. An diesem Punkt setzt nun die Aufgabenstellung des PAIND+E1 an. Es soll ein low-cost Prototyp eines 3D-Laser-Modul entwickelt werden, mit welchen eine 3D-Karte der Umgebung möglichst detailliert modelliert werden kann. Dabei soll einerseits die Frage geklärt werden, welche Konfiguration eine bestmögliche Modellierung der Umgebung bietet und anderseits, mit welchen Mitteln eine Realisierung möglich ist.

Nach Erhalt der Aufgabenstellung galt es anfänglich, ein 3D-Laser-Modul mit einem bestehenden 2D-Laser zu realisieren. Beim Projektbeginn im September 2017 wurde dies von Dr. Björn Jensen abgeändert, da nun ein 3D-Laserscanner für diese Aufgabe zur Verfügung stand. Der 3D-Laserscanner besitzt gegenüber dem 2D Laser der Vorteil, dass bereits räumliche Messdaten zur Echtzeit übermittelt werden können. Zudem besitzt dieser auch einen bedeutend größeren Messbereich, welcher für die Erstellung von Umgebungskarten nötig ist. Beim zu erarbeitende Projekt handelt es sich um eine Realisierung eines funktionsfähigen Prototypen. Der Prototyp soll sich um eine Achse drehen und die Daten dem mobilen

Roboter einmal pro Umdrehung zur Verfügung stellen. Das entwickelte Laser-Modul soll im Rahmen der Arbeit auf dem Packbot-Roboter getestet werden.

Ziel des Projektes ist die Realisierung eines 3D-Laser Moduls. Dabei wird die gesamte Hardware mit den gewählten Komponenten zusammengebaut. Die Software wird durch bestehende Codepakete und eigenen Erweiterungen auf die Aufgabenstellung angepasst. In erster Priorität soll damit 3D Mapping in Echtzeit betrieben werden können. Das Modul wird mit dem bestehenden 3D-Laserscanner der Marke Velodyne des Typs VLP-16 realisiert. Dabei soll eine möglichst grosse räumliche Abdeckung der Umgebung erreicht werden. Diese wird in einer möglichst detaillierten Punktwolke modelliert. Zweite Priorität ist die Hinderniserkennung in Frontrichtung. Dazu muss in Frontrichtung eine detaillierte Punktwolke ermittelt werden können. Das Modul soll einerseits auf dem Packbot nutzbar, sowie auch eigenständig einsetzbar sein. Das Pflichtenheft im Anhang A, grenzt die Aufgabenstellung auf weitere Punkte ein.

Für die Aufgabenstellung eignet sich ein strukturierter Projektphasenablauf. Dabei werden nacheinander die Phasen Initialisierung, Informationsbeschaffung, Konzeption, Realisierung und die Testphase durchlaufen. Im Anhang A Pflichtenheft werden abgeschätzter Aufwand, Arbeitsmittel und zu erwartende Ergebnisse erläutert.

Der Inhalt der Dokumentation richtet sich nach den zu erarbeitenden Projektphasen. Im Anhang B ist ein detaillierter Projektplan angefügt, welcher einzelne Arbeitspakete und das Zeitmanagement aufzeigt. Im Kapitel 5 werden dazu noch Erläuterungen getätigt.

# Inhaltsverzeichnis

<b>1 Informationsbeschaffung</b>	<b>1</b>
1.1 Entfernungsmessung . . . . .	1
1.1.1 Velodyne VLP-16 Puck . . . . .	1
1.2 Stand der Technik . . . . .	4
1.2.1 IMM MSAS Team MSS Warschau . . . . .	4
1.2.2 Hector Tracker Team Hector Darmstadt . . . . .	6
1.2.3 Schlussfolgerung . . . . .	8
1.3 Software . . . . .	8
1.3.1 ROS Robot Operating System . . . . .	9
1.3.2 Velodyne Package . . . . .	10
1.4 Datenverarbeitung . . . . .	10
1.4.1 Raspberry Pi 2 & 3 . . . . .	10
1.4.2 Banana Pi M3 . . . . .	11
1.4.3 Odroid C2 & XU4 . . . . .	11
1.4.4 Up Board Squared . . . . .	12
1.4.5 Schlussfolgerung . . . . .	12
1.5 Antriebsmöglichkeiten . . . . .	12
1.5.1 Schrittmotor . . . . .	13
1.5.2 Gleichstrommotor . . . . .	13
1.6 Absolute Positionsbestimmung . . . . .	14
1.7 LED und Photodiode . . . . .	14
1.8 Sensor QRE 1113 . . . . .	14
1.9 Speisung und Verkabelung . . . . .	15
1.9.1 Abwärtswandler . . . . .	15
1.9.2 Schleifring . . . . .	16
1.10 Zwischenfazit . . . . .	16
<b>2 Konzeption</b>	<b>17</b>
2.1 Konzeptionsgrundlage . . . . .	17
2.2 Variante 1: Plattform . . . . .	17

2.3	Variante 2: Turm stationär . . . . .	19
2.4	Variante 3: Turm rotierend . . . . .	21
2.5	Evaluation der Komponenten . . . . .	22
2.5.1	Datenverarbeitung: Raspberry Pi 3 . . . . .	22
2.5.2	Antrieb: Pololu . . . . .	23
2.5.3	Positionsbestimmung: QR1113 . . . . .	23
2.5.4	Drehübertager: SENRING SNE01-24GV . . . . .	23
2.6	Zwischenfazit . . . . .	23
<b>3</b>	<b>Realisierung</b>	<b>24</b>
3.1	Hardware . . . . .	24
3.1.1	mechanische Komponenten & Gehäuse . . . . .	24
3.1.2	elektrische Komponenten . . . . .	26
3.2	Software . . . . .	26
3.2.1	3D Mapping . . . . .	26
3.2.2	Motorenansteuerung . . . . .	26
3.3	Zwischenfazit . . . . .	26
<b>4</b>	<b>Tests</b>	<b>27</b>
4.1	Testprotokolle . . . . .	27
4.2	Testergebnisse . . . . .	29
4.3	Fazit . . . . .	29
<b>5</b>	<b>Reflektion</b>	<b>30</b>
5.1	Schlussfolgerungen . . . . .	30
5.2	Erläuterungen Projektmanagement . . . . .	30
5.3	Ausblick . . . . .	30
5.4	Schlusswort . . . . .	30
5.5	Danksagung . . . . .	30
<b>Abbildungsverzeichnis</b>		<b>I</b>
<b>Tabellenverzeichnis</b>		<b>II</b>
<b>Quelltextverzeichnis</b>		<b>III</b>

<b>Glossar</b>	<b>IV</b>
<b>Literaturverzeichnis</b>	<b>VI</b>
<b>Onlinequellen</b>	<b>VII</b>
<b>Anhang A Pflichtenheft</b>	<b>VIII</b>
<b>Anhang B Projektmanagement</b>	<b>IX</b>
B.1 Diagramm . . . . .	IX
B.2 Tabelle . . . . .	IX
B.3 Screenshot . . . . .	IX
B.4 Graph . . . . .	IX

# Kapitel 1

## Informationsbeschaffung

In einer ersten Phase wurde ein Zeitraum zur Informationsbeschaffung festgelegt. Dieser Abschnitt ist einerseits für die Themeneinarbeitung und anderseits für die Abgrenzung der Aufgabe und der Ziele erforderlich.

Nachfolgend werden die wichtigsten Erkenntnisse der Informationsbeschaffung erläutert, die maßgebend für die Konzeption in Kapitel 2 und die Realisierung in Kapitel 3 sind. Dabei werden zu einzelnen Komponenten und Verfahren Stellung genommen und eruiert, ob diese sich für das Projekt eignen. Des Weiteren werden relevante Software erläutert, welche für die Realisierung nötig sind. Ein weiterer Abschnitt behandelt bereits bestehende Lösungen, die den Stand der Technik aufzeigen.

### 1.1 Entfernungsmessung

In diesem Unterkapitel wird der bestehende 3D-Laserscanner Velodyne VLP-16 analysiert. Dabei werden wichtige Spezifikationen erläutert. Anhand der Spezifikationen und Schnittstellen des Velodyne werden in den nächsten Unterkapitel weitere Komponenten eruiert.

#### 1.1.1 Velodyne VLP-16 Puck

Beim Velodyne VLP-16 Puck handelt es sich um einen Echtzeit 3D-Laserscanner, der auf dem Light Detection And Ranging (LIDAR)-Verfahren basiert. Nachfolgende Angaben entstammen dem Datenblatt, wenn nicht anders referenziert. [LiD16]

Der VLP-16 bietet insgesamt 16 Laser-/Detektorpaare, die in Abbildung 1.1 ersichtlich sind. Mit diesen wird in horizontaler Lage eine Messbereich von  $360^\circ$  erreicht. Dies wird dadurch ermöglicht, dass der Laserscanner sich intern mit 5 - 20 Rotationen pro Sekunde um die eigene Achse dreht. Die Rotationsgeschwindigkeit ist einstellbar. Dabei wird mit einer horizontalen Auflösung von  $0.1^\circ$  –  $0.4^\circ$  gerechnet. Diese Auflösung begründet sich auf die nötige Zeit eines Messdurchgangs. Um alle 16 Laser abzufeuern und wieder zu entladen dauert es  $56 \mu\text{s}$ . Da die 16 Laser mit je  $2^\circ$  Unterschied ausgerichtet sind ergibt sich daraus eine vertikalen Messbereich von  $30^\circ$  mit einer vertikalen Auflösung von  $2^\circ$ . Ein wichtiger Punkt ist somit, dass im stationären Zustand zwischen den Laserstrahlen keine Messpunkte ermittelt werden können. Um dies zu ermöglichen, muss der Laserscanner seine Position verändern können.

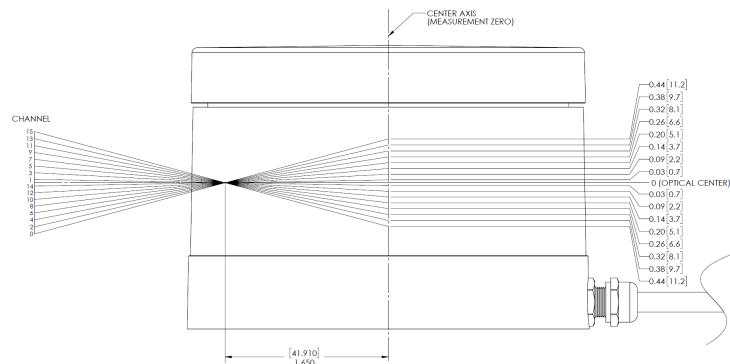


Abbildung 1.1: Laserstrahlen des Velodyne VLP-16  
[LiD16]

Ein relevante Eigenschaft dieses Laserscanners ist die grosse Messdistanz, die Distanzen zwischen 0.43 m bis 100 m ermöglicht. Dabei ist die typische Toleranz  $+/- 3 \text{ cm}$ . Der Reflektionsgrad wird in 256-bit Auflösung angegeben, d.h. dass der Sensor aus der zurückgesendeten Laserimpulsen die Intensität messen kann.

Der VLP-16 benötigt eine separate Interface Box, mit der die Speisung und die Datenschnittstellen zu einem 8-adrigen Kabel zusammengeführt werden. In Abbildung 1.2 ist dieses Kabel mit Pinbelegung dargestellt. Dabei werden die Adern 1-4 für die Ethernet Datenübertragung benötigt. Die Adern 5 und 6 sind nur bei zugeschaltetem Global Positioning System (GPS) nötig, ansonsten sind diese unbenutzt. Die stabilisierte 12 Volt Spannung wird über die Adern 7 und 8 zugeführt.

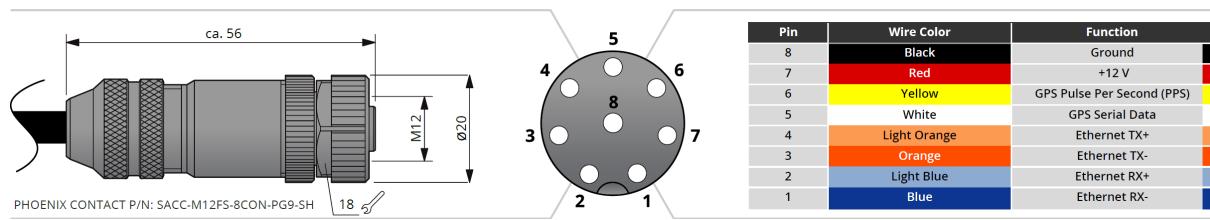


Abbildung 1.2: Anschluss und Kabelbelegung

[LiD16]

Die Interface Box ist in Abbildung 1.3 ersichtlich. Diese besitzt folgende Anschlüsse; eine 12 Volt Speisung, eine Ethernet RJ45-Anschluss und eine GPS Schnittstelle. Für die typische Leistungsaufnahme des Sensors wird 8 Watt angegeben.

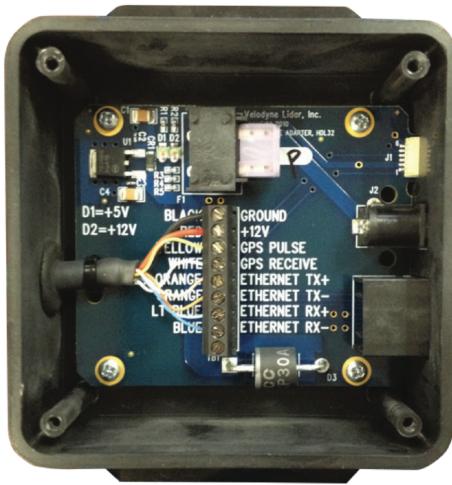


Abbildung 1.3: Ansicht auf die Interface Box

[LiD16]

Über die 100 Mbps Ethernetverbindung werden die Daten- und Positionsdaten vom Velodyne an den Computer übermittelt. Dabei werden für die zwei verschiedenen User Data Protocol (UDP) Pakete die Ports 2368 und 8308 gebraucht. Nachfolgend wird in Abbildung 1.4 der Aufbau eines Datenpakets dargestellt. Jedes Paket besitzt einen 42-Byte Header und einen Datenblock, der aus Laserrückgabewert, kalibrierten Reflektionsgrad, Azimutwert und Zeitstempel besteht.

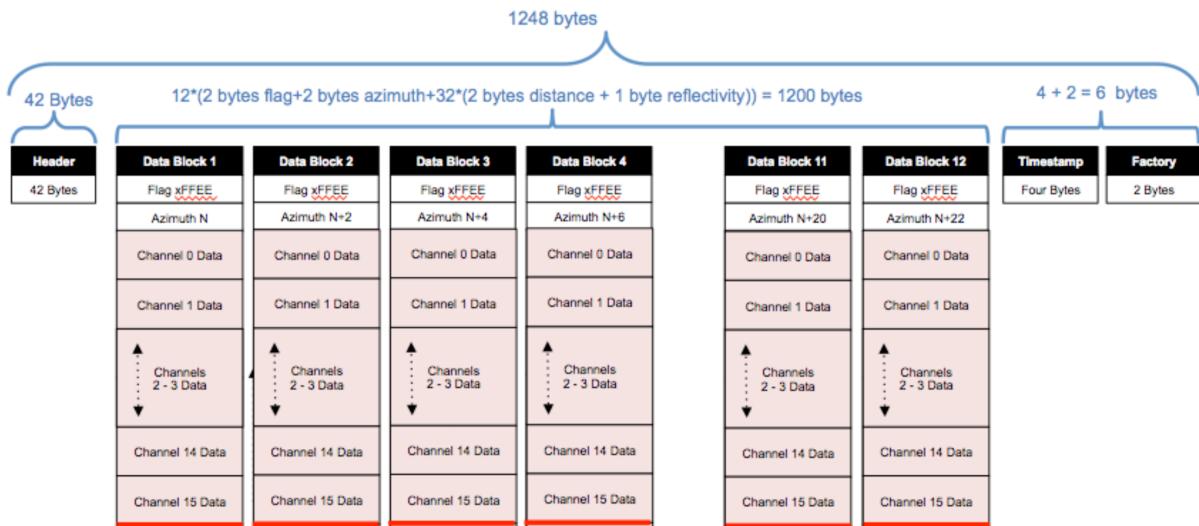


Abbildung 1.4: Aufbau Datenpaket

[LiD16]

## 1.2 Stand der Technik

Dieses Kapitel dient als Vorstudie über den Einsatz des Velodyne VLP-16 durch bereits bestehende Projekte. Dabei werden zwei verschiedene Konfigurationen betrachtet und dazu entsprechend Vor- und Nachteile erläutert. Es handelt sich hierbei um zwei Teams, welche an der European Robotic Hackathon (EnRich) 2017 teilgenommen haben und als State-of-the-Art betrachtet werden.

### 1.2.1 IMM MSAS Team MSS Warschau

Das Institute of Mathematical Machines (IMM) in Warschau hat den Velodyne VLP-16 an einer endlos drehenden Konstruktion befestigt. Dabei ist der Sensor nicht in der üblichen Lage (Ausrichtung XY-Ebene), sondern um 90° abgedreht (Ausrichtung YZ Ebene). In Abbildung 1.5 ist die entsprechende Konfiguration abgebildet.



Abbildung 1.5: Roboter des Team IMM an der EnRich  
[Bed17]

In der nachfolgenden Betrachtung wird vom Koordinatensystem in der Abbildung 1.5 ausgegangen. Das Team nutzt bei dieser Konfiguration die begrenzte Auflösung des Sensors besser aus. Die vertikale und horizontale Auflösung wechseln dabei. Ist die Konstruktion nicht drehend, kann er in der Vertikalen den Bereich  $360^\circ$  mit der Auflösung von  $0.01^\circ$  -  $0.04^\circ$  messen. In der Horizontalen kann der Bereich  $30^\circ$  mit  $2^\circ$  aufgelöst werden.

Der Vorteil dieser Konfiguration wird jedoch erst durch die Rotation um die Z-Achse deutlich. Wird der Sensor nun kontinuierlich um die Z-Achse gedreht, verschieben sich die 16 horizontalen Laserstrahlen mit der Umdrehungsgeschwindigkeit der Konstruktion. Einerseits bewegt sich der  $30^\circ$  grosse Messkegel durch den Raum und kann somit  $360^\circ$  in der Horizontalen vermessen. Anderseits kann durch die interne Rotation des Sensors  $360^\circ$  in der Vertikalen ausgemessen werden. Die schlechtere Auflösung von  $2^\circ$  kann somit kompensiert werden. Dies Konfiguration ermöglicht eine detaillierte Messung, da jeder Punkt im Raum von 16 Laserstrahlen mit der Auflösung zwischen  $0.1^\circ$   $0.4^\circ$  durchlaufen wird.

Die oben erläuterte Betrachtung gilt jedoch nur, wenn die Umdrehungsgeschwindigkeit der Konstruktion bedeutend langsamer als die interne Umdrehungsgeschwindigkeit des Sensors ist. Ansonsten besteht die Gefahr, dass die Reflektion des Laserstrahls nicht detektiert werden kann. Die Auflösung ist direkt von der Umdrehungsgeschwindigkeit abhängig. Es wird zusätzlich davon ausgegangen, dass keine Translation, d.h. Bewegung des Roboters, stattfindet. Diese Konfiguration eignet sich somit, wenn das zu vermessende Gelände bekannt ist. Dadurch können vordefinierte Positionen angesteuert werden, an denen eine halbe Umdrehung ausreicht, um eine räumliche Messung an der definierten Position zu erstellen.

Weiter wird in obigen Betrachtung die Abweichung zur Drehachse komplett vernachlässigt. Es entsteht ein Messfehler, der dem Abstand des Sensors zur Drehachse entspricht. Ein weiterer Messfehler kann durch Translation entstehen, wenn sich das Fahrzeug bewegt. Diese zwei Messfehler lassen sich jedoch durch geometrische Transformation in ein festes Koordinatensystem kompensiert. Die Schwierigkeit dabei ist, die exakten Messewerte mittels einem Drehencoder und einer Inertiale Messeinheit (IMU) zum jeweiligen Zeitpunkt zu ermitteln. In der nachfolgenden Matrix werden die Variablen der Translation mit  $t$ , Skalierung mit  $s$  und die Drehung um die Z-Achse mit dem Winkel  $\alpha$  angegeben.

$$\begin{bmatrix} s_x * \cos\alpha & s_x * -\sin\alpha & 0 & t_x \\ s_y * \sin\alpha & s_y * \cos\alpha & 0 & t_y \\ 0 & 0 & s_z & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} x' \\ y' \\ z' \\ w \end{bmatrix}$$

Ein Nachteil für die Aufgabenstellung ist, dass bei dieser Konfiguration im freien Feld viele Messpunkte ins Leere messen, da gerade himmelwärts (In die Z-Achse) keine Reflektion entsteht. Auch Messpunkte in Richtung negative Z-Achse müssen gefiltert werden, da ansonsten ständig die Oberfläche des Roboters selbst reflektiert wird.

### 1.2.2 Hector Tracker Team Hector Darmstadt

Das Team Hector der Universität Darmstadt besitzt auf dem Hector Tracker eine weitere Konfigurationsmöglichkeit. Auch dieses Team arbeitet mit einem endlos drehenden Konstruktion. Bei nachfolgenden Betrachtungen wird vom Koordinatensystem in Abbildung

1.6 ausgegangen. Der Velodyne VLP-16 befindet sich  $45^\circ$  abgeneigt zur XZ-Ebene. Dabei ist die Lage des Sensors zentral auf der Drehachse Z.

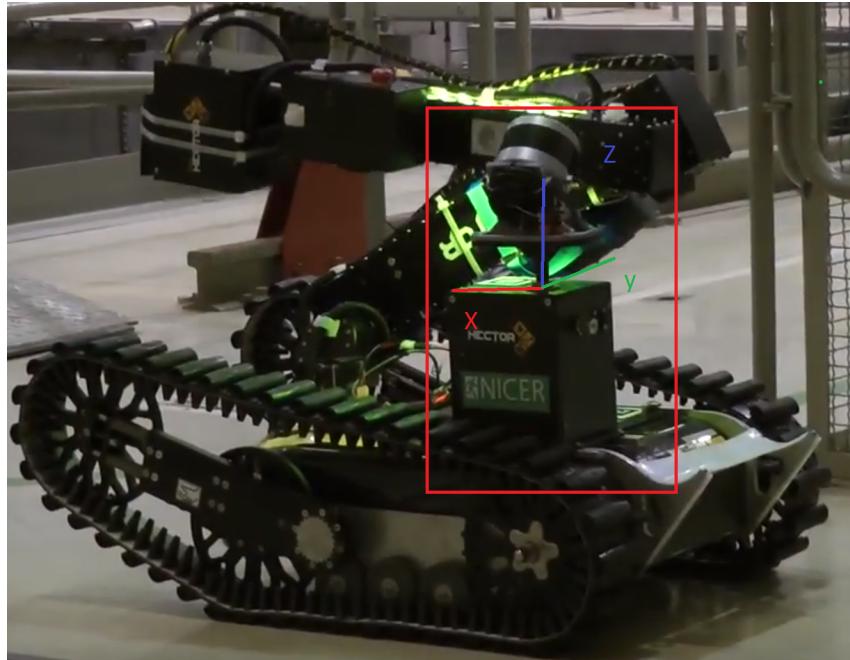


Abbildung 1.6: Roboter des Team Hector an der EnRich  
[Koh17]

Im stationären Zustand kann durch diese Konfiguration keine Verbesserung der Auflösung erreicht werden. Die Auflösungen bleiben erhalten, sind jedoch nun entsprechend der Neigung um  $45^\circ$  verschoben. Durch die Neigung des Sensors wird der Raum nicht gleichmäßig vermessen. Die räumliche Vermessung wird auch hier erst mit der Rotation um die Z-Achse ermöglicht. Es kann während einer Umdrehung somit durch die Neigung von  $45^\circ$  und dem  $30^\circ$  Kegel des Sensors eine Abdeckung in der Vertikalen von  $240^\circ$  erreicht werden. Die horizontale Abdeckung bleibt hierbei weiterhin  $360^\circ$ , da der Sensor die interne Rotation vollführt.

Gegenüber einer festen nicht drehenden Lösung besitzt diese Konfiguration somit den Vorteil, dass Messpunkte, welche sich zwischen den Laserstrahlen befinden, durch Drehen erreichen lassen. Dabei wird der  $30^\circ$  Messkegel durch die Rotation ständig im vertikalen Bereich hin- und hergeschoben.

Auch bei dieser Betrachtung muss die Umdrehungsgeschwindigkeit der Konstruktion bedeutend langsamer als die interne Umdrehungsgeschwindigkeit des Sensor sein. Die Auflösung bei dieser Konstruktion ist direkt zur Umdrehungsgeschwindigkeit abhängig. Des Weiteren muss bei dieser Konfiguration die Translation, d.h. die Bewegung des Roboters, noch separat betrachtet werden.

Im Vergleich zum Projekt der IMM entstehen bei dieser Konfiguration weniger Messpunkte in Richtung des Roboters und in Richtung Z-Achse (himmelwärts). Dies ist für Rauminterne Messungen unvorteilhaft. Für die Messung im freien Feld ist diese Konfiguration besser geeignet, sofern sich keine hohen Objekte nahe dem Sensor befinden.

### 1.2.3 Schlussfolgerung

An der EnRich 2017 nutzen mehrere Teams den Velodyne VLP-16 mit unterschiedlichen Konfigurationen. Die zwei betrachteten Konfigurationen nutzen die Möglichkeit einer endlos drehenden mechanischen Konstruktion. Im Unterkapitel 1.5 werden mögliche Antriebsmöglichkeiten für eine solche Konstruktion evaluiert. Eine endlos drehende Konstruktion benötigt eine spezielles Handling der Kabelführung, daher wird im Unterkapitel ?? die Möglichkeit eines Schleiftrings beschrieben. Beide Konfigurationen besitzen Vor- und Nachteile bei der Umgebungserkennung. Um eine möglichst detaillierte Umgebungswolke zu erstellen müssen vorwiegend Translation und Rotation um die Drehachse kompensiert werden.

## 1.3 Software

In diesem Kapitel wird die notwendige Software beschrieben. Es erläutert einerseits das Robot Operating System (ROS) und dessen Funktion im Projekt. Daneben werden kurz weitere Softwareapplikationen und -packages erwähnt, welche für die Aufgabenstellung nützlich sind.

### 1.3.1 ROS Robot Operating System

Die gesamte Kommunikation mit Sensoren und Aktoren findet auf dem Packbot mit ROS statt. Da das 3D-Laser-Modul auf dem Packpot, wie auch selbstständig funktionieren soll, ist ROS nicht zwingend einsetzbar. Dennoch wurde für die Aufgabenstellung ROS gewählt. Im Zusammenhang mit Velodyne und 3D Mapping bietet ROS neben Visualisierungstools und bereits bestehender Treiber-Packages auch die Integrationen der Point Cloud Library (PCL). Dies ist eine umfangreiche Library, welche viele Funktionen und Codebeispiele bereit stellt.

Grundsätzlich wird ROS, wegen seiner Nähe zu Linux Distributionen, auf einem Ubuntu Betriebssystem aufgesetzt und ist ein Software-Framework, dass die Programmiersprachen C++ und Python nutzt. Diese in 2007 entwickelte Open Source Software erhielt in den letzten Jahren ständig neue und überarbeitete Versionen. In der nachfolgenden Abbildung sind die aktuellen Distributionen ersichtlich.

Distro	Release date	Poster	Turtle, turtle in tutorial	EOL date
ROS Melodic Morenia	May, 2018	TBD	Tuttle, turtle in tutorial	May, 2023
ROS Lunar Loggerhead	May 23rd, 2017			May, 2019
ROS Kinetic Kame (Recommended)	May 23rd, 2016			April, 2021 (Xenial EOL)
ROS Jade Turtle	May 23rd, 2015			May, 2017
ROS Indigo Igloo	July 22nd, 2014			April, 2019 (Trusty EOL)

Abbildung 1.7: aktuelle Distributionen von ROS [Jos17]

Die Empfehlung von ROS und diversen Literaturen liegt hierbei bei der neusten Distribution ROS "Kinetic Kame". [Jos17] Es handelt sich hierbei um eine Langzeitversion von ROS, welche bis 2021 Unterstützung bietet. Im Zug der ersten Versuchen mit ROS wurde auf einem Laptop mit AMD64-Architektur gearbeitet. Auf diesem wurde ROS Kinetic Kame vollumfänglich ermöglicht. Für das 3D-Lasermodul wird jedoch ein einbaubaren Einplatinencomputer benötigt, welche im 1.4 genauer betrachtet werden.

Während der Einarbeitung mit Kinetik Kame konnten einige Nachteile der Distribution festgestellt werden. Es wurde festgestellt, dass diverse Packages nur für AMD64-

Architekturen zur Verfügung stehen und nicht für ARM-Architekturen. Dies ist ein wesentlicher Nachteil, da die meisten Einplatinencomputer mit ARM-Architekturen arbeiten.

### 1.3.2 Velodyne Package

Wie bereits im vorherigen Kapitel erwähnt besitzt ROS ein Treiber-Package für den Velodyne VLP-16. In der Abbildung 1.8 ist der Aufbau dieses Treibers mit dem ROS-Tool rqt-graph visualisiert.

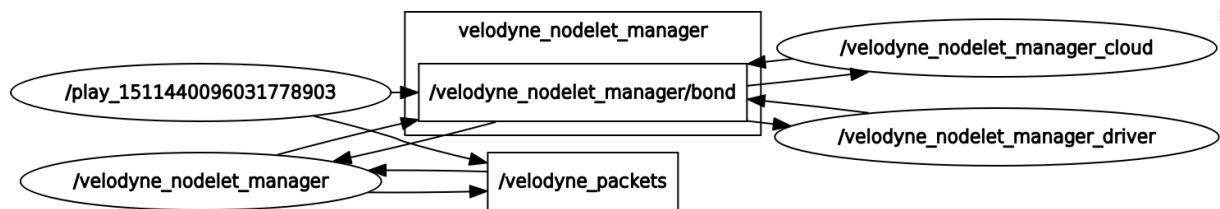


Abbildung 1.8: Velodyne Treiber aus rqt-graph

## 1.4 Datenverarbeitung

Um die Datenmenge zu verarbeiten und die Ansteuerung der Komponenten zu realisieren, eignen sich Einplatinencomputer. Einplatinencomputer besitzen den Vorteil, dass sie gegenüber üblichen Mikroprozessoren grössere Speichermöglichkeiten, höhere Prozessorleistung und bootbare Betriebssysteme ermöglichen. Gegenüber Barebone PC (XPC) besitzen sie den Vorteil, dass sie Hardware näher sind, indem sie direkt ansteuerbare Pins besitzen (GPIO). Aus diesem Grund werden nachfolgend diverse Einplatinencomputer betrachtet. Kriterien bei der Auswahl eines geeigneten Boards sind Prozessorleistung, Preis, Ethernet-Schnittstelle, Speichermöglichkeit, GPIO Verfügbarkeit und die verbaubare Dimension.

### 1.4.1 Raspberry Pi 2 & 3

Das Raspberry Pi ist eines der bekanntesten Einplatinencomputer und bietet daher eine grosse Community. Da bereits ein Raspberry Pi 2 zur Verfügung gestanden ist, konnten die ersten Erfahrungen mit einem Raspberry Pi gemacht werden. Das Raspberry bietet

zusammen mit ROS Kinetic Kame und Ubuntu Mate LTS 16.04 eine mögliche Lösung für die Datenverarbeitung. Das Raspberry Pi 2 bzw. 3 basiert auf einem Broadcom System-on-a-Chip (SOC) und ist mit einem ARM Cortex A7 bzw. A53 Prozessor mit vier Kernen ausgestattet. Die Taktfrequenz liegt bei diesen lediglich bei 1.2 GHz. Beide Modelle besitzen 1 GB RAM. Das Betriebssystem wird auf einer Secure Disk Memory Card (SD)-Card gebootet.

### 1.4.2 Banana Pi M3

Der Banana Pi M3 bietet zur Zeit (Stand Oktober 2017) die höchste Performance bei Einplatinencomputern mit ARM-Architektur durch den Allwinner A83T Achtkern-Prozessor, der mit 1.8 GHz taktet. Neben USB-Anschlüssen bietet es eine SATA-USB-Schnittstelle, die den internen 8 GB embedded Multimedia Card (eMMC)-Speicher um bis zu 2 TB erweitern lässt. Es bietet auch direkt integrierte WLAN-Schnittstellen, sowie eine RJ45 Gigabit Netzwerkschnittstelle. Zum Betreiben des Banana PI M3 benötigen man ein 5V DC-Netzteil. Der Preis eines BananaPi M3 liegt momentan bei ca 90. Franken. Da im Zusammenhang mit Banana Pi M3 und Ubuntu mehrfach Komplikationen und Probleme veröffentlicht werden. [TEA16];

### 1.4.3 Odroid C2 & XU4

In diversen Literaturen (siehe [Jos17] Kapitel 4) werden neben dem Raspberry Pi, Odroid Boards als empfohlene Einplatinencomputer aufgelistet. Dabei stehen die aktuelle Modelle Odroid-C2 oder Odroid-XU4 zur Verfügung. Sie bieten eine höhere Prozessorleistung, 1.5 GHz bzw. 2 GHz mit je 2 Gigabyte Random Access Memory (RAM). Betriebssysteme können via SD oder eMMC gebootet werden. Beide Boards sind jedoch noch nicht lange auf dem Markt und bieten in vielen Anwendungen nur Beta-Versionen. Vor allem die Unterstützung von Ubuntu LTS 16.04 ist nicht restlos geklärt. Der Preis dieser Boards ist um die 80 - 90 Franken.

### 1.4.4 Up Board Squared

Der Up Board Squared Einplatinencomputer unterscheidet sich wesentlich von den bisherig betrachteten Boards. Der Prozessor arbeitet nicht mit ARM-Architektur, sondern mit AMD64-Architektur. Somit können Ubuntu und ROS Distributionen vollumfänglich genutzt werden. Es besitzt mit Intel Pentium ein vier kerniger Prozessor und taktet mit 2.5 GHz. Zusätzlich bietet es einen separaten 500 MHz Grafikprozessor (GPU) und bis zu 8 GB RAM. Es bietet neben USB, eine zweifache RJ45-Schnittstelle und auch ein GPIO Pin Layout, welches dem Raspberry Pi entspricht. Der Preis eines Up Board Squared kostet je nach Ausführung zwischen 180-380 Fr.

### 1.4.5 Schlussfolgerung

Für die Aufgabenstellung eignet sich lediglich das Raspberry Pi 2 oder 3, um ROS mit Ubuntu LTS 16.04 zu betreiben. Das Banana Pi und die Odroid Boards sind im Kriterium Speichermöglichkeit und Prozessorleistung besser geeignet, können jedoch wegen fehlender Betriebssystem-Kompatibilität nicht genutzt werden. Einzig das Up Board Squared bieten in fast allen Kriterien eine denkbare Lösung. Aus Kostengründen wurde jedoch das Raspberry Pi als genügend bewertet.

## 1.5 Antriebsmöglichkeiten

Um das Produkt um eine Achse drehen zu lassen, müssen Motoren eingesetzt werden. Nachfolgend sind zwei verschiedene Motorenarten geschildert, um die Einsatzmöglichkeit zu klären. Wichtige Kriterien für die Aufgabenstellung sind einerseits die Ansteuerung und die Dimension. Anderseits muss die Möglichkeit bestehen die Winkeländerung zu erzielen.

### 1.5.1 Schrittmotor

Der Schrittmotor ist eine Antriebsmöglichkeit, welche für das Projekt in Frage kommt. Es gibt sehr kostengünstige und kompakt dimensionierte Motoren dieser Art. Ein interessanter Aspekt ist das gezielt Steuern des Motors. Durch einen Stromimpuls bewegt sich ein Schrittmotor nur einen festgelegten Winkelschritt weiter. Er kann bereits ohne zusätzliche Sensorik definierte Schritte anfahren, aus denen die Winkeländerung eruiert werden kann. Schrittmotoren besitzen die Eigenschaft, dass in der Ruhelage ein Haltemoment entsteht. Diese Eigenschaft wird jedoch für die Aufgabenstellung nicht zwingend benötigt.

Nachteilig für die Aufgabenstellung am Schrittmotor ist der höhere Stromverbrauch, vor allem zum Aufbringen des Haltemoments. Da nur ein Schritt vollführt werden kann, wenn das entsprechende Drehmoment nicht überschritten wird, müsste dieses sorgfältig berechnet werden. Ein bedeutender Nachteil im Zusammenhang mit der Aufgabenstellung ist, dass durch Schrittverluste die Winkeländerung nicht mehr quantitativ ermittelt werden kann. Ein weiterer Nachteil ist das verhältnismäßig hohe Gewicht. Dies ist kein Kriterium für die Aufgabenstellung, sollte jedoch bei der Realisierung beachtet werden.

Um mit einem Mikrocontroller einen Schrittmotor anzusteuern, empfiehlt sich ein Schrittmotorentreiber. Mit solchen Treibern lässt sich der Motor mittels 2 Steuerpins rotieren. Um die Drehgeschwindigkeit zu senken, bieten diese Treiber die Möglichkeit, die Schritte in 2, 4, 8 und 16 Teilschritte zu senken, um dadurch eine feinere Bewegung zu ermöglichen.

### 1.5.2 Gleichstrommotor

Als Alternative zum Schrittmotor bietet sich ein üblicher Gleichstrommotor. Diese Motoren sind für viele Einsatzbereiche geeignet und es gibt sie in verschiedenen Größen und Umdrehungszahlen. Im Gegensatz zu Schrittmotoren laufen Gleichstrommotoren kontinuierlich, aufgrund eines Stroms, der durch die Wicklungen fliesst. Nachteilig ist somit, dass weder die genaue Anzahl Umdrehungen noch die momentane Phasenlage bekannt ist. Es gibt jedoch eine Vielzahl an Möglichkeiten diese zu eruieren. Einerseits gibt es die Möglichkeit mittels Hall-Sensoren oder mittels Quadraturencodern, die aktuelle Drehrichtung und die Drehzahl zu ermitteln. Um dies zu ermöglichen, braucht es zusätzlich einen Vierquadrantensteller (H-Brücke), sowie eine Sensorlogik, damit der Sensor mittels

Pulsweitenmodulation (PWM) angesteuert werden kann. Im Preissegment bis 40 Fr. gibt es leistungsfähige Gleichstrommotoren für den Anwendungsbereich.

## 1.6 Absolute Positionsbestimmung

Damit beim 3D Mapping eine feste Koordinatenachse erstellt werden kann, benötigt eine drehende Konstruktion eine absolute Positionsbestimmung. Nachfolgend sind Varianten erläutert, welche sich dafür eignen.

## 1.7 LED und Photodiode

Eine simple Variante ist der Einsatz einer LED, welche im rotierenden Teil konstant leuchtet. Dabei wird die Photodiode an den stationären Teil angebracht und als Nullpunkt genutzt. Durch das Licht der LED wird ein Strom an der Photodiode übertragen und dies kann mit einer einfachen Schaltung über einen GPIO gemessen werden. Nachteilig bei dieser Variante ist der Lichtkegel der LED. Die LED müsste so verbaut werden, dass ein fokussierter Lichtstrahl auf die Photodiode fällt. Zudem ist diese Variante sehr stark abhängig von Umgebungslicht. Ansonsten kann der Nullpunkt stark abweichen. Diese Variante bedingt das mittels einer weiteren Komponente die Winkeländerung gemessen werden kann.

## 1.8 Sensor QRE 1113

Eine weiter ähnliche Variante ist der Einsatz eines QRE 1113 Sensor. Dieser Sensor basiert auf dem Prinzip der Infrarotreflektion auf. Dabei werden dunkle Oberflächen schlechter reflektiert, als helle Oberflächen. Diese Eigenschaft wird genutzt, um den Nullpunkt zu detektieren. Auch diese Variante bedingt das mittels einer weiteren Komponente die Winkeländerung gemessen werden kann. Eine weitere Überlegung ist die Aneinanderreihung von diesen Sensoren. Dies bietet die Möglichkeit einen Quadraturencoder Binär oder Gray-Code, wie in Abbildung 1.9 zu realisieren.

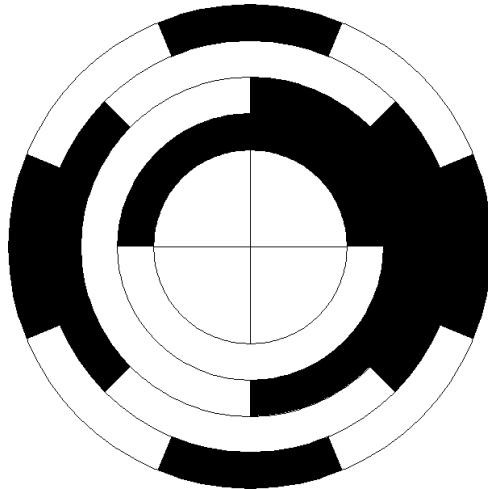


Abbildung 1.9: Quadraturencoder mit 4 Sensoren

Es können mit  $N$  Sensoren  $2^N$  Zustände unterschieden werden. Aus der Auflösung kann der Winkel berechnet werden. Für diese Aufgabe werden jedoch auch  $N$  GPIOs benötigt. Auch diese Variante ist stark abhängig vom Umgebungslicht und zusätzlich können die einzelnen Sensoren einander durch Streuung beeinflussen. Um dies zu minimieren sollte eine solche Variante mittels Gray-Code realisiert werden, damit nur immer eine gleichzeitige Zustandsänderung stattfindet. Die Abtastung der Zustände müsste nach dem Nyquist-Shannon-Abtasttheorem mit mindestens doppelter Abtastfrequenz gemessen werden.

## 1.9 Speisung und Verkabelung

### 1.9.1 Abwärtswandler

Der Abwärtswandler D24V50F5 von Pololu eignet sich sehr gut für die Stromversorgung von Einplatinencomputer. Er bietet eine stabile 5V-Ausgangsspannung bis zu einer ausgangsseitigen Belastung von 8 Ampere. Der Wirkungsgrad beläuft dabei auf über 90 Prozent bei einer eingesetzten Betriebsspannung von 12 Volt. Diese Angaben gehen aus Abbildung 1.10 hervor, die aus dem Datenblatt des Abwärtswandler entstammen.

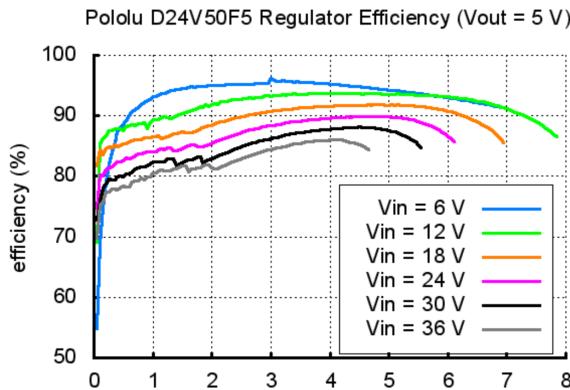


Abbildung 1.10: Wirkungsgrad D24V50F5 [D24V50F5]

### 1.9.2 Schleifring

Da übliche Kabel nur für einige wenige Verdrehungen ausgelegt sind und so bei endlos drehenden Vorrichtungen kaputt gehen würden, werden Schleifring als Drehübertrager eingesetzt. Ein Schleifring überträgt das Eingangssignal über eine Bürste und bildet somit einen Gleitkontakt. Da vermehrt nicht nur Speisung, sondern auch Datenübertragung über Schleifringe geführt werden, wächst dieses Sortiment stetig. Es gibt bereits einige wenige Hersteller aus dem amerikanischen und asiatischen Markt, welche Ethernet-Übertragung für Schleifringe vertreiben. Ein Ethernet-Schleifring, der zusätzlich Leistungsübertragung bis 5A zulässt kostet nach Abklärungen bei diversen Herstellern zwischen 120 - 500 Fr.

## 1.10 Zwischenfazit

Für die Einarbeitung in das Projekt ist die Informationsbeschaffung bzw. Recherche ein wesentlicher Bestandteil. Dabei werden wichtige Grundlagen für die Konzeption erstellt. Mittels Schrittmotoren oder Gleichstrommotoren kann der Velodyne gedreht werden. Für die Datenverarbeitung stehen mehrere Einplatinencomputer zur Verfügung, dabei muss jedoch die Integration in das System gewährleistet sein. Für endlos drehende Konfigurationen wird zusätzlich ein Schleifring benötigt, da ansonsten die Kabelführung nicht gewährleistet ist.

# Kapitel 2

## Konzeption

Dieses Kapitel beschreibt die Resultate und Ergebnisse aus der Konzeptionsphase. Nachfolgend werden auf die verschiedenen Konzeptionsvarianten eingegangen. Dabei werden Überlegungen und Bezüge zur Aufgabenstellung gemacht. Jede Variante wird abschliessend mit einem kurzen Fazit beurteilt

### 2.1 Konzeptionsgrundlage

Als Konzeptionsgrundlage dient das Pflichtenheft, welches im Anhang A einsehbar ist. Wie darin erwähnt, soll das Modul möglichst frei drehend konzipiert werden. Dies ist das ausschlaggebendste Kriterium für die Bauform des Moduls. Da der Velodyne VLP-16, wie in Kapitel 1.1.1 erläutert, einen begrenzten vertikalen Öffnungswinkel besitzt, wurde bei den folgenden Konzepten die Möglichkeit eines beweglichen Moduls geprüft. Durch eine grösseren Öffnungswinkel kann während der Bewegung der Raum besser ausgemessen werden. Ein weiteres relevantes Kriterium ist die Einsatzmöglichkeit des Modul. Es soll einerseits auf dem Packbot, sowie auch als eigenständiges Produkt funktionieren. Daher sind als Schnittstellen einen Speiseanschluss, welcher 12 Volt Gleichstrom liefert und eine Ethernet RJ45-Anschluss nötig. Nachfolgend sind auf diesen Grundlagen drei Varianten dargelegt.

### 2.2 Variante 1: Plattform

Die Variante 1: Plattform ist in der Skizze in Abbildung 2.1 ersichtlich. Bei dieser Konstruktion werden alle elektronischen Komponenten, welche für die Signalverarbeitung und

die Energieversorgung nötig sind, in einem rechteckigen Gehäuse im unteren Teil verbaut. Die Interface Box des Velodyne VLP-16 wird auch in diesem Gehäuse untergebracht. Lediglich der Laserscanner VLP-16 befindet sich ausserhalb des Gehäuses auf der Plattform oberhalb des Gehäuses.

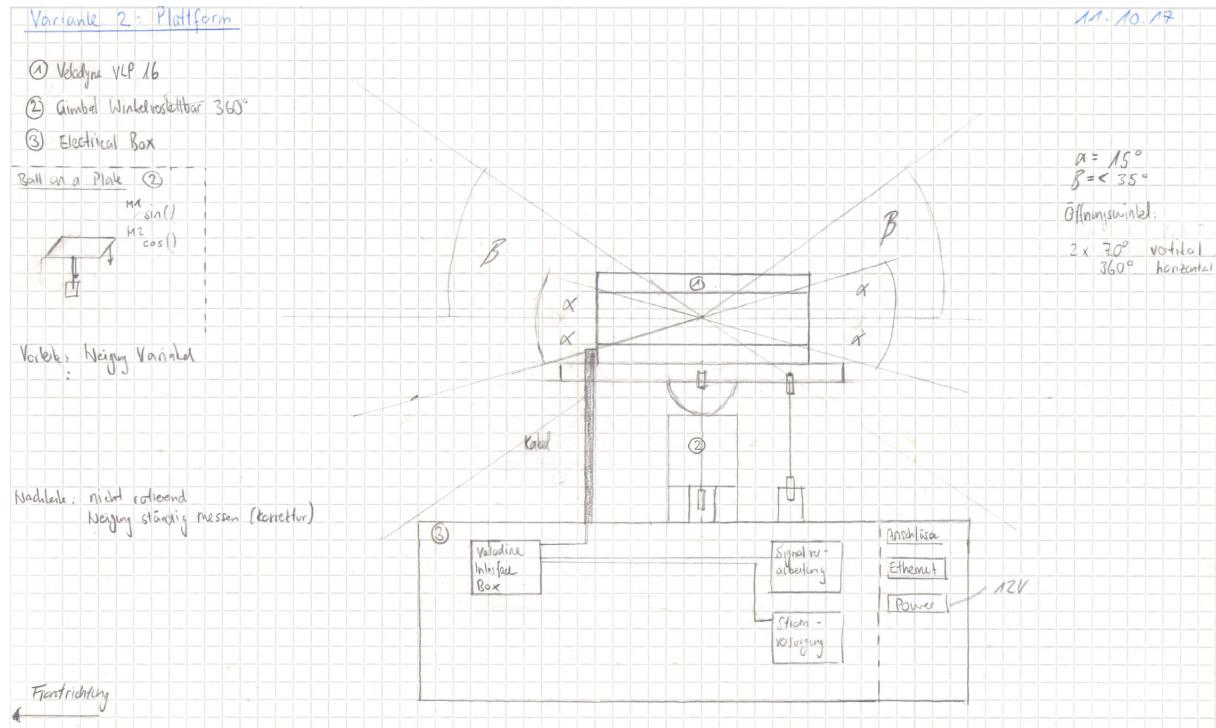


Abbildung 2.1: Skizze Variante 1

Die Eigenheit dieser Konzeption ist die Plattform, auf welcher sich der Velodyne VLP-16 befindet. Der Einsatz dieser Plattform erklärt sich durch ein bekanntes Regelungsexperiment namens "Ball on a Plate". In Abbildung 2.2 ist eine CAD-Zeichnung eines solchen Regelungsexperiment dargestellt. Bei "Ball on a Plate" werden zwei Servomotoren, welche je mit einem Gelenk mit einer Seite der Plattform verbunden sind, angesteuert. Indem die Servomotoren mittig auf der Seite mit der rechteckigen Plattform verbunden sind, kann durch Drehen der Servomotoren die Plattform zweiachsig geneigt werden. Im Regelungsexperiment kann durch Zuhilfenahme einer PID-Regelung ein Ball darauf balanciert werden. [Ben15]

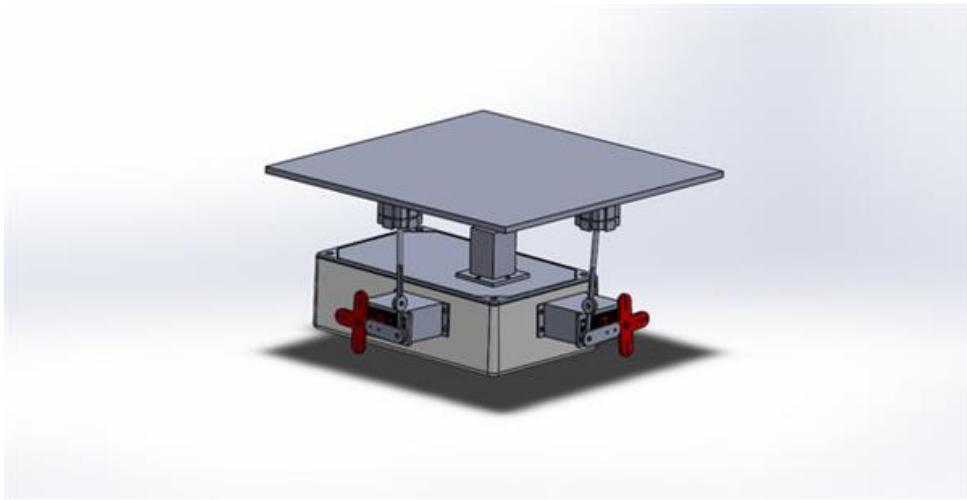


Abbildung 2.2: Ball on a Plate CAD [Ben15]

Diese Überlegungen bieten die Möglichkeit den Laserscanner VLP-16, welcher in der Konzeption auf der Plattform befestigt ist, durch das Ansteuern der Servomotoren in alle Richtungen zu neigen. Somit kann der Öffnungswinkel in jede Richtung vergrößert werden. Die Grenzen liegen dabei im wesentlichen bei der mechanischen Begrenzung.

Hauptproblematik bei dieser Konzeption ist die zusätzlich nötige Sensorik. Durch die Neigung der Plattform kann ohne zusätzliche Sensorik nicht auf die Messausrichtung zurück geschlossen werden. Dies erschwert die Visualisierung der 3D-Laserscanner Messdaten. Die Plattform müsste mittels einer eigenen IMU, welche aus Gyrosensor, Accelerometer und Magnetometer besteht, erweitert werden. Zusätzlich ist die Realisierung der Konzeption sehr stark abhängig von den mechanischen Komponenten, da die Verbindungsstifte von Servomotor und Grundplatte die Winkeländerung festlegen.

Während der Besprechung mit Herr Jensen vom wurde diese Konzeption verworfen und nicht mehr weiter vertieft.

ZEIT

## 2.3 Variante 2: Turm stationär

Das zweite Konzept ist eine Turmartige Konstruktion, die in Abbildung 2.3 dargestellt ist. Die Idee dazu lieferte das Unterkapitel 1.2.1 aus dem Kapitel Informationsbeschaffung.

Dabei befindet sich der Velodyne VLP-16 abgesetzt von den elektrischen Komponenten in der Höhe. Dies ermöglicht einen größeren Erfassungswinkel.

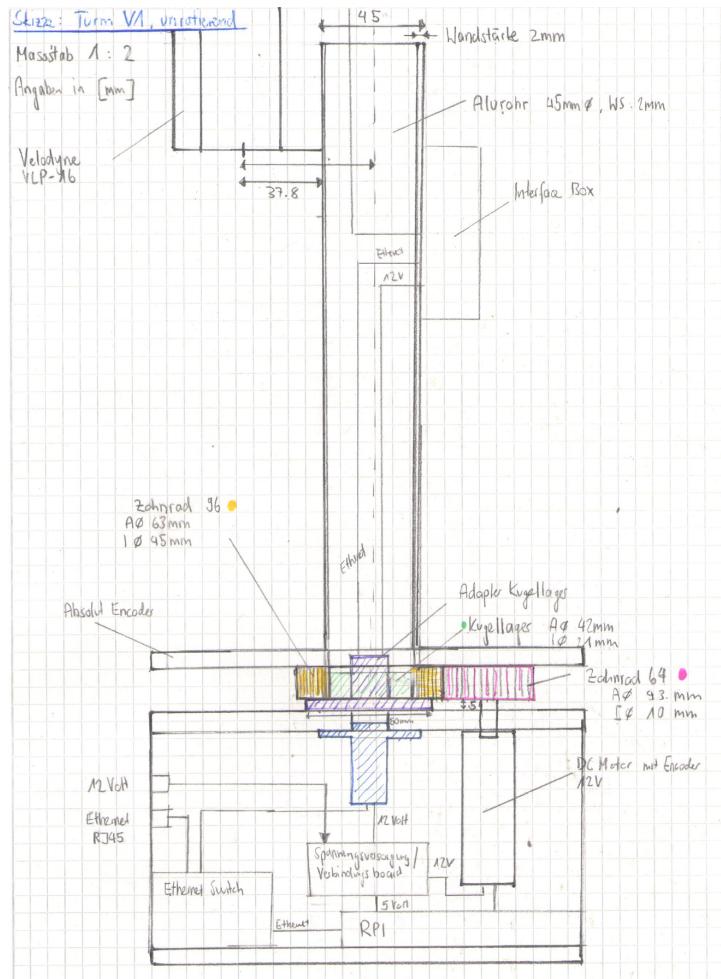


Abbildung 2.3: Skizze Variante 2

Eigenheit dieses Konzept ist, dass sich der Turm endlos drehen lässt. Diese Konfiguration verbessert die Problematik des Öffnungswinkel, welche in 1.1.1 erläutert ist. Durch die interne Rotation des VLP-16 können räumliche Scans durchgeführt werden, wie bereits in Unterkapitel ?? erläutert. Wird der Sensor nun zusätzlich um eine Achse gedreht können so auch Raumpunkte gemessen werden, welche ansonsten nicht erreichbar sind.

In dieser Konfiguration wird der VLP-16 mit einem Gleichstrommotor, der über zwei Zahnräder mit dem Aluohr verbunden ist, gedreht. Der untere Teil des Moduls, in dem

sich die Datenverarbeitung befindet, ist stationär. Mittels einem Adapter, an dem ein Kugellager befestigt ist, kann ein möglichst geringe Reibung erzielt werden.

Die gesamte Datenverarbeitung und Ansteuerung wird im stationären Teil getätig. Die Kabelverbindung des VLP-16 wird über einen Schleifring nach unten geführt. Die Interface Box wird in der Skizze im stationären Teil verbaut. Es ist jedoch auch möglich diese direkt gegenüber des Velodyne zu befestigen.

Nachteil dieser Konfiguration ist, dass es schlecht erweiterbar ist. Da der Schleifring begrenzte Kabeldurchführungen besitzt können keine weiteren Komponenten wie beispielsweise die 3D Kamera Theta hinzugefügt werden. Ein weiterer Nachteil ist die Problematik, wenn die Rotationsgeschwindigkeit des Turm auf mehrere Umdrehungen pro Sekunde ansteigt. Es können fehlerhafte Messresultate beim VLP-16 entstehen, daher muss die Drehgeschwindigkeit des Turms verhältnismässig langsam gegenüber der Drehgeschwindigkeit VLP-16 sein.

Die mechanische Konstruktion ermöglicht auch die Umpositionierung des Velodynes. Wird der Sensor oberhalb des Alurohr befestigt, kann auch die zweite betrachtete Konfiguration des Teams Hector in Unterkapitel 1.2.2 ausgetestet werden.

## 2.4 Variante 3: Turm rotierend

Das dritte Konzept ist wiederum eine Turmartige Konstruktion. Eine Skizze ist in Abbildung 2.4 ersichtlich. Es handelt sich hier um eine abgeänderte Version von Variante 2. Dabei befindet sich der Velodyne VLP-16 erneut abgesetzt von den elektrischen Komponenten in der Höhe. Die wesentlichste Änderung liegt darin, dass nun auch die Datenverarbeitung im unteren Teil mitdreht. Dies wurde ermöglicht indem alle mechanischen Drehkomponenten und der DC Motor umgedreht wurden. Somit kann das Problem der Erweiterbarkeit des vorherigen Konzepts gelöst werden. Die gesamte Elektronik und Sensorik dreht nun mit. Lediglich die Schnittstellen Ethernet und 12 Volt Speisung führen über den Schleifring zum Packpot. Für diese Konfiguration wird lediglich zusätzlich ein Alusockel benötigt, welche das Alurohr mit dem Gehäuse verbindet.

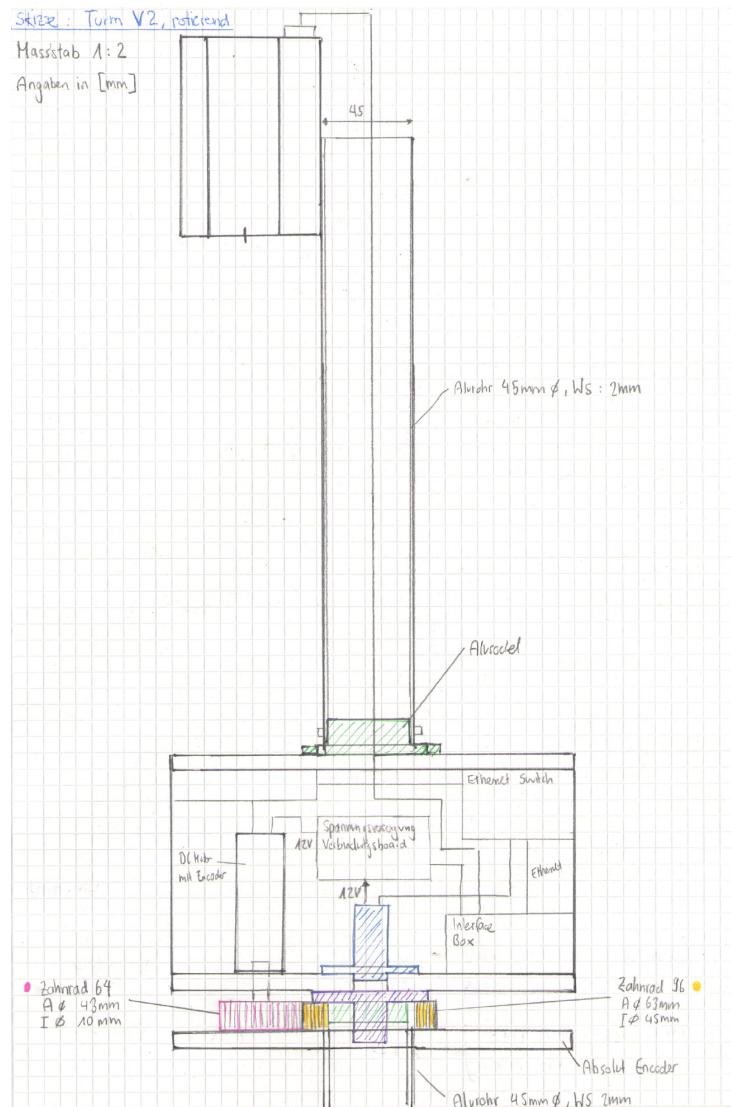


Abbildung 2.4: Skizze Variante 3

## 2.5 Evaluation der Komponenten

### 2.5.1 Datenverarbeitung: Raspberry Pi 3

im Unterkapitel 1.4 wurden bereits diverse Einplatinencomputer betrachtet und bewertet. Für die Konzeption wurde aus Performancegründen das Raspberry Pi 3 dem Raspberry 2

vorgezogen.

### **2.5.2 Antrieb: Pololu**

### **2.5.3 Positionsbestimmung: QR1113**

### **2.5.4 Drehübertager: SENRING SNE01-24GV**

Wie bereits in Unterkapitel 1.9.2 erläutert ist die Auswahl von Ethernet-Schleifringen bisher noch klein. Es wurde der SENRING SNE01-24GV für die Aufgabenstellung evaluiert. Dieser Schleifring ist einerseits mit 120 Fr. der kostengünstigste Schleifring und bietet die erforderlichen Übertagungsmöglichkeiten von Ethernet. Zusätzlich besitzt er 8 weitere Kabeldurchführungen, von denen 2 Kabel mit 5 Ampere und 6 Kabel mit 2 Ampere belastet werden können. [Sen15]

## **2.6 Zwischenfazit**

# Kapitel 3

## Realisierung

Dieses Kapitel beschreibt die Realisierung des Prototyps. Der Prototyp ist eine überarbeitete Version der Konzeptvariante 2 , siehe 2.3. Es mussten Änderungen bei verschiedenen Komponenten durchgeführt werden, diese werden in den nachfolgenden Unterkapitel 3.1.1 erläutert.

### 3.1 Hardware

Da beide Projekte, welche im Unterkapitel 1.2 eine endlos drehende Konstruktion nutzten, sowie die Aufgabenstellung im Anhang A eine drehende Konstruktion vorgibt, wurde eine endlos drehende mechanische Konstruktion gewählt.

#### 3.1.1 mechanische Komponenten & Gehäuse

Um eine möglichst einfache und schnelle Lösung zu realisieren, wurden die mechanischen Komponenten nach Verfügbarkeit ausgewählt. Die Masse des Alurohrs und der Zahnräder waren größtenteils durch die Rösser des Kugellagers gegeben. Das Kugellager wurde so gewählt, dass ein Ethernet RJ45 Stecker hindurchgeführt werden kann, da der gewählte Schleifring ein vorkonfektionierten RJ45-Stecker besitzt.

Das Kugellager wurde direkt auf das Alurohr gepresst und zusätzlich mit Schrauben verkeilt.

Die Zahnräder wurden mit dem Tool Onshape gelayoutet und im FabLaab mit einem 3D-Drucker erzeugt. Aus den vorgegebenen Dimensionen wurde ein Übersetzungsverhältnis der Zahnräder von 1.5 gewählt.

Dabei wurden in einer ersten Version

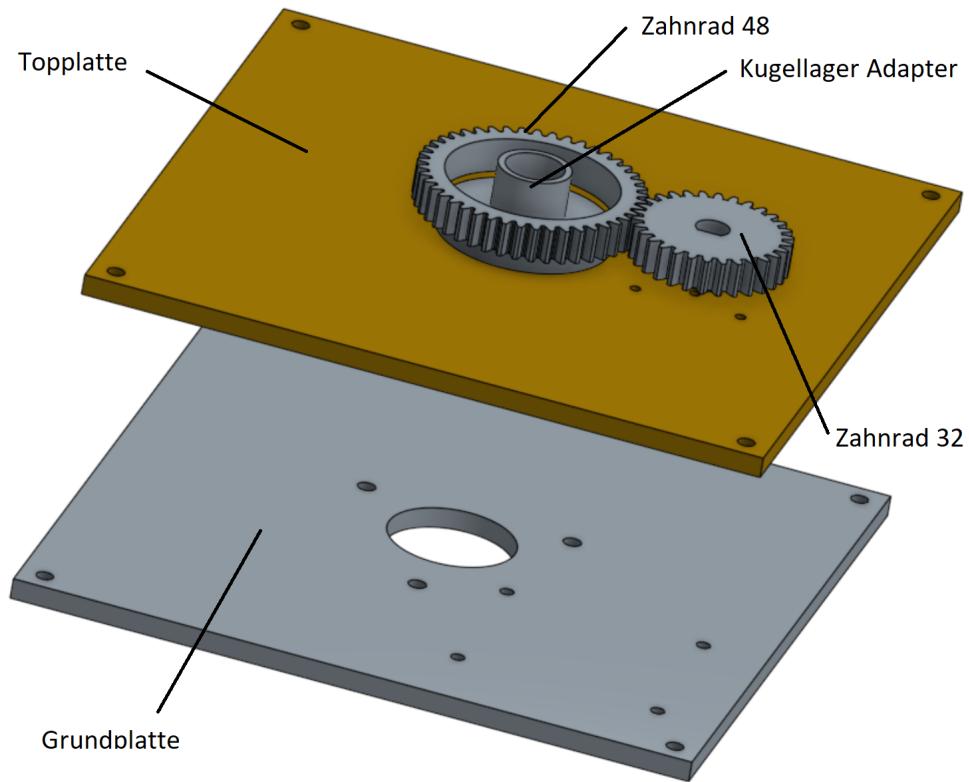


Abbildung 3.1: zusammengefügte mechanische Komponenten

Die Grösse der Platten wurde so gewählt, dass alle Komponenten dazwischen verbaut werden können. Daher wurden zwei MDF-Platten mit den Massen 200mm x 200mm x 6mm (HxBxT) im Fablab erstellt. Die Grund- und Topplatten wurden in OnShape gelayoutet un

### **3.1.2 elektrische Komponenten**

## **3.2 Software**

### **3.2.1 3D Mapping**

### **3.2.2 Motorenansteuerung**

## **3.3 Zwischenfazit**

# Kapitel 4

## Tests

### 4.1 Testprotokolle

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut

metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

- First itemtext

- Second itemtext
- Last itemtext
- First itemtext
- Second itemtext

## 4.2 Testergebnisse

## 4.3 Fazit

# Kapitel 5

## Reflektion

### 5.1 Schlussfolgerungen

### 5.2 Erläuterungen Projektmanagement

### 5.3 Ausblick

### 5.4 Schlusswort

### 5.5 Danksagung

An dieser Stelle möchte ich mir herzlich bedanken, die mich bei der Anfertigung dieser Arbeit unterstützt haben.

Zuallererst gebührt der Dank an Dr. Björn Jensen, der mich bei dieser Industriearbeit tatkräftige unterstützt hat, sowie mit wertvollen Hinweisen und schnellen Rückmeldungen zur Seite gestanden ist. Mein Dank geht auch an Jonas Räber, der mir eine grosse Hilfe für die Einarbeitung mit ROS war. Ebenfalls bedanken möchte ich mich bei den zwei Gegenleser Andreas Zimmermann und Angela Burch für die textuelle und inhaltliche Analyse der Dokumentation.

Zuletzt noch besten Dank an D

# Abbildungsverzeichnis

1.1	Laserstrahlen des Velodyne VLP-16 . . . . .	2
1.2	Anschluss und Kabelbelegung . . . . .	3
1.3	Ansicht auf die Interfacebox . . . . .	3
1.4	Aufbau Datenpaket . . . . .	4
1.5	Roboter des Team IMM EnRich . . . . .	5
1.6	Roboter des Team Hector EnRich . . . . .	7
1.7	aktuelle Distributionen von ROS . . . . .	9
1.8	Velodyne Treiber aus rqt-graph . . . . .	10
1.9	Quadraturencoder mit 4 Sensoren . . . . .	15
1.10	Wirkungsgrad D24V50F5 . . . . .	16
2.1	Skizze Variante 1 . . . . .	18
2.2	Ball on a Plate CAD [Ben15] . . . . .	19
2.3	Skizze Variante 2 . . . . .	20
2.4	Skizze Variante 3 . . . . .	22
3.1	zusammengefügte mechanische Komponenten . . . . .	25

# Tabellenverzeichnis

# Quelltextverzeichnis

# Glossar

ARM	Acorn Risc Machine verbreitetes Mikroprozessor-Design, das grösstenteils 32Bit-Prozessoren beruht (ARMv7)
AMD64	X64 Prozessor-Architektur der handelsüblichen 64-Bit-PC und Server
BLAM	Berkeley Localization And Mapping ist ein Open-Source Softwarepaket für LiDAR-basierte Echtzeit 3D
BSD	Berkley Software Distribution frei verwendbare Lizenz, auch als Vorlage für kommerzielle Produkte
eMMC	embedded Multimedia Card Digitales Speichermedium, das nach dem Prinzip der Flash Speicherung arbeitet
EnRich	European Robotic Hackathon weltweit erste und einzige Robotik Wettbewerb mit realen Szenarios
GPS	Global Positioning System globales Navigationssatellitensystem zur Positionsbestimmung
GPU	Grafikprozessor spezialisierter Prozessor für Graphikanwendungen
IMU	Inertiale Messeinheit ist eine Kombination mehrerer Trägheitssensoren
LIDAR	Light Detection And Ranging Verfahren zur optischen Distanzmessung
PWM	Pulsweitenmodulation Verfahren

RAM	Random Access Memory Verfahren
ROS	Robot Operating System Verfahren
SD	Secure Disk Memory Card digitales Speichermedium, das nach dem Prinzip der Flash-Speicherung arbeitet
SLAM	Simultaneous Localisation and Mapping Verfahren
SOC	System-on-a-Chip Integration der Funktionen eines programmierbaren elektronischen Systems auf einem Chip
UDP	User Data Protocol verbindungsloses Netzwerkprotokoll zur Versendung von Datagrammen von IP-basierten Rechnernetzen
XPC	Barebone PC komakte unvollständige Computerdesign, meist mit vielen Schnittstellen und guter Erweiterbarkeit

# Literaturverzeichnis

- [Jos17] Lentin Joseph. *ROS Robotics Projects*. Packt Publishing, März 2017. ISBN: 978-1-78355-472-0. URL: <http://proquest.tech.safaribooksonline.de/book/hardware/9781783554713>.

# Onlinequellen

- [Bed17] Janusz Bedkowski. *Enrich Team Informationen*. Juli 2017. URL: <http://enrich.european-robotics.eu/> (besucht am 13.11.2017).
- [Ben15] Benedict. *Student's 3D printed balancing device is a whole new ball game*. Dez. 2015. URL: [www.3ders.org/articles/20151213-students-3d-printed-balancing-device-is-a-whole-new-ball-game.html](http://www.3ders.org/articles/20151213-students-3d-printed-balancing-device-is-a-whole-new-ball-game.html) (besucht am 06.10.2017).
- [Koh17] Stefan Kohlbrecher. *Enrich Team Informationen*. Juli 2017. URL: <http://enrich.european-robotics.eu/> (besucht am 13.11.2017).
- [LiD16] Velodyne LiDAR. *VLP-16 Manual*. März 2016. URL: <http://velodynelidar.com/vlp-16.html> (besucht am 28.09.2017).
- [Sen15] Senring. *SNE01 series Mini 1000M Ethernet slip ring*. Dez. 2015. URL: <http://www.senring.com/ethernet-slip-rings/SNE01.html> (besucht am 03.10.2017).
- [TEA16] BPI TEAM. *Banana Pi M3 Ubuntu Image*. Apr. 2016. URL: <http://forum.banana-pi.org/t/bpi-m3-new-image-bpi-m3-ubuntu-16-04-beta-mate/1440> (besucht am 29.09.2017).

# Anhang A

# **Anhang B**

**B.1 Diagramm**

**B.2 Tabelle**

**B.3 Screenshot**

**B.4 Graph**