

**Ferdinand Klug**

**Prototypentwicklung  
eines Rovers  
als Trägerplattform für  
Sensorsysteme**

**MASTERARBEIT**

zur Erlangung des akademischen Grades  
Diplom-Ingenieur/in

Masterstudium Geomatics Science



**Technische Universität Graz**

Betreuer:

Ao. Univ.-Prof. Dipl.-Ing. Dr.techn. Norbert Kühtreiber

Institut für Navigation

Graz, Juni 2013

Deutsche Fassung:  
Beschluss der Curricula-Kommission für Bachelor-, Master- und Diplomstudien vom 10.11.2008  
Genehmigung des Senates am 1.12.2008

## EIDESSTATTLICHE ERKLÄRUNG

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbstständig verfasst, andere als die angegebenen Quellen/Hilfsmittel nicht benutzt, und die den benutzten Quellen wörtlich und inhaltlich entnommenen Stellen als solche kenntlich gemacht habe.

Graz, am .....

.....

(Unterschrift)

Englische Fassung:

## STATUTORY DECLARATION

I declare that I have authored this thesis independently, that I have not used other than the declared sources / resources, and that I have explicitly marked all material which has been quoted either literally or by content from the used sources.

.....

date

.....

(signature)



# Danksagung

Mein größter Dank gilt meiner Freundin Carmen, die viel Geduld und Zeit aufgebracht hat und mir immer wieder motivierend zur Seite stand. Sie hat viel zum Gelingen dieser Arbeit beigetragen. Für die Unterstützung bei der schriftlichen Arbeit bedanke ich mich bei GR Mag. Heimo Schäfermann.

Für die gute Betreuung bedanke ich mich recht herzlich bei Ao. Univ.-Prof. Dipl.-Ing. Dr.techn. Norbert Kühnreiter. Er hat mir bei der Realisierung dieser Arbeit viel Freiraum gelassen und ist mir immer beratend zur Seite gestanden. Ein weiterer Dank gilt meinen Kollegen und dem gesamten Institut, dass sie mir die Realisierung dieser Arbeit ermöglicht und durch fachliche Diskussionen auf neue Aspekte aufmerksam gemacht haben.

Abschließend bedanke ich mich noch bei meinem Bruder Dipl.-Ing. Christoph Klug, als auch bei Ing. Martin Sorger, Hannes Legenstein und Fabian Herrmann, die mich besonders in den Bereichen der Softwareentwicklung, Mechanik und Datenübertragung beraten haben.

# Lizenzierung



Dieses Projekt ist lizenziert als Inhalt der Creative Commons Namensnennung - Nicht-kommerziell - keine Weitergabe.<sup>1</sup>

**Die Institute der Studienrichtung Geodäsie (Technische Universität Graz) dürfen:**  
das Werk bzw. den Inhalt für Forschungs- und Entwicklungsarbeiten frei Nutzen und die Resultate veröffentlichen.  
Es steht den Instituten frei Abwandlungen und Bearbeitungen des Werkes bzw. Inhaltes anzufertigen.

## **Zu den folgenden Bedingungen:**

*Namensnennung* — Sie müssen den Namen des Autors/Rechteinhabers in der von ihm festgelegten Weise nennen.

*Keine kommerzielle Nutzung* — Dieses Werk bzw. dieser Inhalt darf nicht für kommerzielle Zwecke verwendet werden.

*Keine Weitergabe* — Dieses Werk darf nur für institutsinterne Forschungs- und Entwicklungsarbeiten verwendet werden.

*Freier Zugriff* — Der Autor hat freien Zugriff auf die Forschungs- und Entwicklungsarbeiten sofern dieser daran beteiligt ist.

## **Wobei gilt:**

*Verzichtserklärung* — Jede der vorgenannten Bedingungen kann aufgehoben werden, sofern Sie die ausdrückliche Einwilligung des Rechteinhabers dazu erhalten.

*Public Domain (gemeinfreie oder nicht-schützbar Inhalte)* — Soweit das Werk, der Inhalt oder irgendein Teil davon zur Public Domain der jeweiligen Rechtsordnung gehört, wird dieser Status von der Lizenz in keiner Weise berührt.

*Sonstige Rechte* — Die Lizenz hat keinerlei Einfluss auf die folgenden Rechte:

- Die Rechte, die jedermann wegen der Schranken des Urheberrechts oder aufgrund gesetzlicher Erlaubnisse zustehen (in einigen Ländern als grundsätzliche Doktrin des fair use etabliert);
- Das Urheberpersönlichkeitsrecht des Rechteinhabers;
- Rechte anderer Personen, entweder am Lizenzgegenstand selber oder bezüglich seiner Verwendung, zum Beispiel für Werbung oder Privatsphärenschutz.

*Hinweis* — Im Falle einer Verbreitung müssen Sie anderen alle Lizenzbedingungen mitteilen, die für dieses Werk gelten. Am einfachsten ist es, an entsprechender Stelle einen Link auf diese Seite einzubinden.

---

<sup>1</sup>Angepasste Version der Creative Commons Lizenz (Basistext und Graphik von <http://creativecommons.org>)

# Kurzfassung

In dieser Masterarbeit wurde der Prototyp eines ferngesteuerten Rovers konstruiert, der modular aufgebaut, geländegängig und erweiterbar ist. Auch wurde eine Bodenstation mit dazugehöriger Antennennachführung und einer autonomen Steuersoftware entwickelt. Mit dieser Arbeit wurden neue Konzepte in der Datenübertragung und der Steuerung von Rovern erarbeitet. Diese Steuerung kann mittels 40Mhz RC-Signale über einen Computer erfolgen.

Der Großteil der Hardware wurde selbst geplant, gebaut, programmiert und getestet. Durch die Verwendung von industriellen WLAN-Modulen in Verbindung mit einem Autopiloten wurde das Datenübertragungskonzept realisiert.

Das Zusammenspiel von Hardware und Software, mit dessen implementiertem Berechnungskonzept der Nachführungsparameter, konnte durch Tests verifiziert werden. Sowohl die Datenübertragung, als auch das RC-Steuerungskonzept haben sich als funktionstüchtig erwiesen. Die Ergebnisse haben gezeigt, dass der Einsatz von Spezialelektronik nicht immer notwendig ist und durch die Verwendung von handelsüblichen Teilen oftmals umgangen werden kann, ohne die Funktionalität zu beeinflussen. Es sollte dennoch qualitativ hochwertiges Material verwendet werden. Dieses System würde sich zur Datenerfassung und als Testträger für verschiedene Sensoren besonders in solchen Gebieten eignen, wo die Gesundheit des Menschen gefährdet wäre.

# Abstract

In this master thesis the prototype of a modular built up, all-terrain and expandable remote controlled rover was designed. A ground station with necessary antenna tracking and an autonomous steering software was also developed. New concepts were worked out for the data transmission and rover control. This control can be carried out via a computer by means of 40 MHz radio controlled signals.

The major part of the hardware was planned, built, programmed and tested by the author. The data transmission concept was carried out by the use of industrial WLAN modules in connection with an autopilot.

The coordination of the hardware, software and the implemented calculation concept of the tracking parameters was verified by tests. Both, the data transmission and the re-control concept, turned out to be working.

The results have shown that the use of special electronics is not always necessary and can be avoided by the use of standard parts without influencing the functionality. Nevertheless, high-quality material should be used. This system would be particularly suitable for different sensors for the data acquisition and as a test carrier in areas where the health of humans would be endangered.

# Abkürzungen

**A/D** Analog/Digital

**AHRS** Attitude and heading reference system

**AM** Amplitudenmodulation

**APM** Autopilot Mega

**ASCII** American Standard Code for Information Interchange

**AV** Audio-Video

**bmvit** Bundesministerium für Verkehr, Innovation und Technologie

**DCM** Direct Cosine Matrix

**DOP** Delution of Precision

**DSS** Double Superhead System

**DSSS** Direct Sequence Spread Spectrum

**DVI** Digital Visual Interface

**EIA** Electronic Industries Alliance

**EIRP** Equivalent isotropically radiated power

**FHSS** Frequency Hopping Spread Spectrum

**FM** Frequenzmodulation

**FPV** Fly per Video

**GAN** Global Area Network

**GnD** Chassis Ground

**GNSS** Global Navigation Satellite System

**GPS** Global Positioning System

**HDMI** High Definition Multimedia Interface

**HDOP** Horizontal Delution of Precision

**ID** Identifier

**IMU** Inertial Measurement Unit  
**INS** Inertial navigation system  
**ISM** Industrial, Scientific and Medical Band

**LAN** Local Area Network  
**LED** Light Emitting Diode  
**LiPo** Lithium-Polymer  
**LLVTTL** Low-Low-Voltage-Transistor-Transistor-Logik  
**LSB** Least Significant Bit  
**LVTTTL** Low-Voltage-Transistor-Transistor-Logik

**MAD** Mean-absolute-deviation  
**MSB** Most Significant Bit

**NiCd** Nickel-Cadmium  
**NMEA** National Marine Electronics Association

**OSD** On Screen Display

**PCM** Pulse Code Modulation  
**PoE** Power over Ethernet  
**PPM** Pulse Position Modulation  
**PWM** Pulse Width Modulation

**RC** Radio Controll  
**RIA** Robot Institute of America  
**RSSI** Received Signal Strength Indication  
**RxD** Received Data  
**SSS** Single Superhead System  
**STD** Standard deviation  
**TTL** Transistor-Transistor-Logik  
**TxD** Transmitted Data

**UV** Unmaned Vehicle

**VDOP** Vertical Delution of Precision  
**VSPE** Virtual Serial Ports Emulator

**WAN** Wide Area Network  
**WLAN** Wireless Local Area Network

# Inhaltsverzeichnis

<b>Danksagung</b>	<b>ii</b>
<b>Lizenzierung</b>	<b>iii</b>
<b>Kurzfassung</b>	<b>iv</b>
<b>Abstract</b>	<b>v</b>
<b>Abkürzungen</b>	<b>vi</b>
<b>1 Einleitung</b>	<b>1</b>
1.1 Motivation und Zielvorgaben . . . . .	2
1.2 Struktur der Arbeit . . . . .	2
<b>2 Zielsetzungen und Grundgedanken</b>	<b>4</b>
2.1 Hardware . . . . .	4
2.1.1 Elektronik . . . . .	4
2.1.2 Rover . . . . .	9
2.1.3 Antennennachführung . . . . .	12
2.1.4 Bodenstation . . . . .	14
2.2 Software . . . . .	15
<b>3 Grundlagen</b>	<b>18</b>
3.1 UV-Autopilotssysteme . . . . .	18

3.2	Datenübertragung . . . . .	21
3.2.1	Datenübertragungsverfahren . . . . .	21
3.2.2	Protokolle . . . . .	34
3.3	Antennenausrichtung . . . . .	37
3.4	Antriebe . . . . .	39
3.4.1	Positionierungsantriebe . . . . .	39
3.4.2	Rover Antriebssystem . . . . .	40
<b>4</b>	<b>Hardwareaufbau</b>	<b>43</b>
4.1	Rover . . . . .	43
4.1.1	Chassis . . . . .	43
4.1.2	Trägerplattform . . . . .	48
4.1.3	Elektrik/Elektronik . . . . .	50
4.1.3.1	Spannungsversorgung . . . . .	51
4.1.3.2	Autopilot (ardupilot mega v2.0) . . . . .	53
4.2	Bodenstation . . . . .	55
4.2.1	Chassis . . . . .	55
4.2.2	Elektrik/Elektronik . . . . .	56
4.2.2.1	Spannungsversorgung . . . . .	56
4.3	Antennennachführung . . . . .	60
4.3.1	Chassis . . . . .	60
4.3.2	Elektrik/Elektronik . . . . .	64
4.3.2.1	Antrieb . . . . .	64
4.3.2.2	Positionierungs- und Ausrichtungselektronik . . . . .	66
4.3.2.3	Richtantenne . . . . .	69
4.4	Systemübergreifende Elektronik . . . . .	70



---

4.4.1	Roversteuerung . . . . .	70
4.4.2	Telemetrie- und Bilddatenübertragung . . . . .	73
<b>5</b>	<b>Berechnungen und Abschätzungen</b>	<b>76</b>
5.1	Datenaufbereitung . . . . .	77
5.2	Koordinatentransformationen . . . . .	79
5.3	Absolute Servoposition . . . . .	81
5.4	Antennenbuffer . . . . .	82
5.5	Abschätzungen . . . . .	85
<b>6</b>	<b>Software und Implementierung</b>	<b>89</b>
6.1	Externe Software . . . . .	89
6.1.1	Autopilot . . . . .	89
6.1.2	Zusatzsoftware Kommunikation . . . . .	91
6.1.3	Joystick-Steuerung . . . . .	93
6.2	C/C++ Tracking-Software . . . . .	94
6.2.1	Softwarestruktur Antennennachführung . . . . .	95
6.2.2	Communication Basics . . . . .	96
6.2.3	Application Basics . . . . .	101
6.2.4	Runtime Environment . . . . .	102
6.2.5	Main Application . . . . .	103
<b>7</b>	<b>Versuche, Messungen und Ergebnisse</b>	<b>106</b>
7.1	Nachführungsauflösung . . . . .	107
7.2	Validierung der „Home“-Position . . . . .	110
7.3	Kompassoffsets . . . . .	113
7.4	Systemtest . . . . .	115
7.5	Trajektorienimulation . . . . .	118

<b>8</b>	<b>Resümee und Ausblick</b>	<b>123</b>
<b>9</b>	<b>Technische Übersicht</b>	<b>127</b>
	<b>Tabellenverzeichnis</b>	<b>131</b>
	<b>Abbildungsverzeichnis</b>	<b>137</b>
	<b>Literaturangabe</b>	<b>139</b>
<b>Anhang</b>		<b>I</b>
A	Baupläne . . . . .	II
A.1	Rover . . . . .	II
A.2	Antennentracker . . . . .	X
A.3	Bodenstation . . . . .	XVIII
B	Kommunikations- und Elektronikschemata . . . . .	XXII
C	Stromlaufpläne . . . . .	XXIII
C.1	Rover . . . . .	XXIII
C.2	Bodenstation . . . . .	XXVII
D	Software Schemas . . . . .	XXXIV
E	Lieferfirmen und Kostenaufstellung . . . . .	XXXVII

# Kapitel 1

## Einleitung

Roboter gewinnen in heutiger Zeit zunehmend an Aufmerksamkeit, sei es im Bereich der Industrie, Raumfahrt, Medizin, Sicherheit oder Forschung. Bei all diesen Fahrzeugen spielt die Navigation und Zielführung eine wichtige Rolle. Dabei zählen derzeit die vollautonomen, humanen aber auch selbstlernenden Roboter zu den höchstentwickelten Systemen dieser Art. In diesen sind zahlreiche Sensoren eingebaut, die zur Positionierung und Datenerfassung verwendet werden. Roboter verwenden verschiedenste Methoden, um ihre Position, Orientierung und Raumlage zu bestimmen. Um diese Parameter zu erhalten und nutzbar zu machen, sind aufwändige Berechnungsalgorithmen und eine dementsprechende Hardware notwendig.

Der Begriff „Roboter“ wird vom ursprünglich tschechischen Wort „robota“ abgeleitet und bedeutet übersetzt (*Fron-*)Arbeit (nach Duden). In der Literatur gibt es unterschiedlichste Varianten, wie ein Roboter definiert wird. Beispielsweise definiert der Verein Deutscher Ingenieure einen Roboter als „nach DIN EN ISO 8373 automatisch geführte, mit drei oder mehr frei programmierbaren Bewegungsachsen ausgerüstete Mehrzweckmanipulatoren, die entweder ortsfest oder mobil in industriellen Anwendungen eingesetzt werden.“ (nach Hans-Jürgen Gevatter & Ulrich Grünhaupt, 2006). Das Robot Institute of America (RIA) hingegen definiert einen Roboter als „a reprogrammable, multifunctional manipulator designed to move material, parts, tools or specialized devices through various programmed motions for the performance of a variety of tasks.“ (nach Mohamed Oubbati, 2007)

Wie aus diesen Definitionen zu erkennen ist, verwendet der Roboter „Manipulatoren“ und verfügt über ein genau definiertes Aufgabengebiet. Ein Rover jedoch, wie er für diese Masterarbeit entwickelt wurde, verfügt über maßgebliche Unterschiede gegenüber einem Roboter. Der Rover ist nach Definition des Collins Dictionary's „a small remote-controlled vehicle which roams over rough, esp. extraterrestrial, terrain taking photographs, gathering rock and soil samples, etc.“ (nach <http://www.collinsdictionary.com/dictionary/english/rover>). Dieser Begriff beschreibt das entwickelte Fahrzeug sehr genau und wird daher auch im weiteren

Verlauf dieser Arbeit verwendet.

## 1.1 Motivation und Zielvorgaben

Die Idee für die Entwicklung des Prototypen eines ferngesteuerten Rovers mit einer Bodenstation und einer Antennennachführung entstand daraus, dass die Institute der Geodäsie an der TU Graz über keine Hardware im Bereich der Robotik verfügen. Die Thematik dieser Arbeit ist dahingehend interessant, da sie disziplinübergreifend ist und auch Kenntnisse in mehreren Fachbereichen wie beispielsweise Mechanik, Elektrotechnik, Elektronik, Nachrichtentechnik, Softwareentwicklung und natürlich der Navigation voraussetzen.

Bereits im Vorfeld wurde eine modulare Bauweise definiert, um Raum für Anpassungen und Erweiterungen zu gewährleisten. Als Ziel bei der Entwicklung und Realisierung des Vorhabens wurden die Geländegängigkeit, hohe Nutzlast und Reichweite, die Transportfähigkeit, das portable Design sowie eine zentrale Steuerung angestrebt.

Trotz der großzügigen Förderung durch den Forschungsförderfonds der TU-Graz, wurde versucht die finanziellen Ausgaben beim Projekt niedrig zu halten. Daher werden nach Möglichkeit Standardindustriekomponenten bzw. Eigenanfertigungen verwendet und nur wenige Spezialbauteile eingesetzt.

## 1.2 Struktur der Arbeit

Im Kapitel 2 werden verschiedene Grundgedanken und Konzepte zur Realisierung der Hardware und Software erläutert. Es wird hierbei auf die Vor- und Nachteile der unterschiedlichen Ansätze hingewiesen. Die im Projekt entstandene Hardware und Software gliedert sich in fünf Einzelmodule die teilweise unabhängig voneinander eingesetzt werden können.

Kapitel 3 behandelt die notwendigen Grundlagen zu Themen wie Autopilotsysteme, Datenübertragungen, Nachführungssysteme und Antriebssysteme. Der Schwerpunkt liegt hier auf den Grundlagen der Datenübertragung und der Nachführungssysteme. Für die Recherchen dieses Grundlagenkapitels wurde sowohl als Hauptquelle das Fachbuch „*Digitale Schnittstellen und Bussysteme*“ von „Friedrich Wittgruber“, diverse Fachzeitschriften wie „Flugmodell und Technik“, als auch unterschiedlichste Fachartikel aus dem Internet herangezogen.

Kapitel 4 beschreibt den Aufbau der Hardware und ist in vier Abschnitte unterteilt. Diese Abschnitte beschreiben die Konstruktion des Rovers, der Bodenstation, der Antennennachführung und der Elektronik. Das **Roverchassis** basiert auf einem Kettenantrieb und wurde nach einem Plansatzes von Christian Ulrich (<http://www.ulrichc.de/>) erstellt. Diese Hardware stellt

dabei das einzige Modul dar, welches nicht zur Gänze selbst entwickelt wurde. Der **Roveraufbau** ist eine vollständige Eigenentwicklung und beinhaltet diverse elektronische Komponenten. Zusätzlich zur RC-Steuereinheit mit der Motorregelung wurde ein Autopilotensystem zur Sensordatenerfassung eingebaut. Desweiteren befinden sich im Aufbau auch die Kommunikationskomponenten zur Übertragung der Autopilot- und Videodaten. Bei der Datenübertragung wurde ein erweitertes Konzept der WLAN-Verbindung realisiert. Die **Bodenstation** wurde als Steuerkonsole des Rovers und der Antennennachführung entwickelt. Sie dient zudem als Datenschnittstelle zwischen der Bodenstationssoftware und dem Rover. Die **Antennennachführung** ist eine vollständig neu designte Eigenkonstruktion und unterscheidet sich deutlich von handelsüblichen Systemen. Im Unterabschnitt Elektronik werden die systemübergreifenden Komponenten der Kommunikation und Steuerung genauer erklärt. Alle notwendigen Stromlaufpläne und Elektronikschaltungen werden hier dargestellt.

Die implementierten Algorithmen zur Berechnung der Nachführungsparameter, der robusten Datenaufbereitung, der Antennenbuffer und diverse Abschätzungen werden im Kapitel 5 beschrieben.

In Kapitel 6 wird die erstellte C/C++ Software erörtert. Zusätzlich wird ein Überblick von den unterschiedlichen externen Programmen gegeben. Die **Nachführungssoftware** entstand aus Gründen der Portabilität in der Programmiersprache C/C++. Die Software beinhaltet alle Berechnungsalgorithmen zur Bestimmung der Nachführungsparameter. Das Kapitel 7 beschreibt die durchgeführten Tests, Messungen und deren Auswertung. Auch wird in diesem Kapitel nicht nur auf die entstandenen Probleme bei den Tests hingewiesen, sondern auch Lösungsansätze dafür geboten.

Im Resümee wird nochmals kurz auf die Planung und Umsetzung des Projekts eingegangen und es werden noch zusätzliche Möglichkeiten für etwaige Verbesserungen und Weiterentwicklungen gemacht. Eine technische Zusammenfassung der einzelnen Hardwaremodule ist in Kapitel 9 zu finden.

Der Anhang beinhaltet alle erstellten Konstruktionspläne, Schemata, Stromlaufpläne und Elektronikschaltungen. Abschließend wird ein Auszug der entstandenen Kosten der Zulieferfirmen gegeben. Auf der beigelegten Daten-DVD befinden sich diverse Messdaten mit den dazugehörigen Matlab-Auswerteroutinen. Auch sind alle Pläne sowie die Nachführungs- und Simulationssoftware darauf gespeichert. Die Source Code Dokumentation der erstellten C/C++ Bibliotheken befindet sich ebenfalls auf dieser DVD.

Die Idee für die Entwicklung eines ferngesteuerten Rovers, der modular aufgebaut, geländegängig und beliebig erweiterbar ist, ist nicht neu. Der Forschungsstand hierfür ist im Bereich der Raumfahrt oder der militärischen Nutzung schon weit fortgeschritten, doch sind diese Unterlagen und Systeme jedoch unglaublich teuer und auch für die private Nutzung meist nicht freigegeben. In der Geodäsie sind ähnliche Projekte nur aus Luftbildaufnahmen von Drohnen bekannt. Aus diesem Grund bietet diese Arbeit ein großes Potential für nachstehende Untersuchungen und Weiterentwicklungen.

# Kapitel 2

## Zielsetzungen und Grundgedanken

### 2.1 Hardware

Das Basiskonzept entspricht dem eines mobilen Roboters, einer portablen Bodenstation sowie einer Antennennachführung. Der Rover soll als Träger für diverse Sensoren und Kameras dienen und noch dazu die Möglichkeit bieten, eventuelle Erweiterungen ohne Neubau durchzuführen. Die Bodenstation wird als Steuerzentrale sowie Auswerteeinheit der Sensordaten an einem zentralen Punkt ausgelegt. Zur Erhöhung der Mobilität, der Datendurchsatzrate sowie der Datenübertragungssicherheit soll eine Antennennachführung verwendet werden. Eine der Vorgaben war unter anderem ein modularer Aufbau, um genügend Spielraum für Weiterentwicklungen und Änderungen zu haben. Da das Projekt aus Eigenmitteln finanziert wurde, waren auch die Kosten ein wichtiger Faktor. Deshalb wurde darauf geachtet, möglichst solche Komponenten zu verwenden, welche bereits privat vorhanden waren oder als Standardwaren bei Händlern (günstig) zu erwerben sind. Dadurch konnten die Materialkosten enorm gesenkt werden und die Gesamtkosten blieben überschaubar. Die Auswahl entsprechender Hardwarekomponenten für die einzelnen Baugruppen wird ebenfalls in diesem Kapitel dargelegt.

#### 2.1.1 Elektronik

##### **Datenerfassung**

Für die erste Ausbaustufe des Projekts dient das Autopilotsystem lediglich zur Erfassung der Sensordaten. Der Autopilot soll für die Zukunft die autonome Steuerung ohne aufwändige Modifikation übernehmen können. Für einen offenen Softwarezugang sollte der Autopilot ein Open-Source-Produkt sein. Für die einfache Implementierung und Systemportabilität stellen die Art der Datenschnittstelle als auch des Datenprotokolls weitere Entscheidungskriterien dar. Eine offene Schnittstelle zur Anbindung verschiedener Sensoren sollte vorhanden sein. In diesem

Projekt wurde nach einem kostengünstigen Komplettpaket gesucht, das diese Anforderungen erfüllt. Nach intensive Recherchen konnten zwei Systeme gefunden werden, welche die zuvor genannten Kriterien auch erfüllen. Der Autopilot Mega (APM) von DIY-Drones (Abbildung: 2.1) und der pixhawk-PX4 (Abbildung: 2.2) von der ETH Zürich.

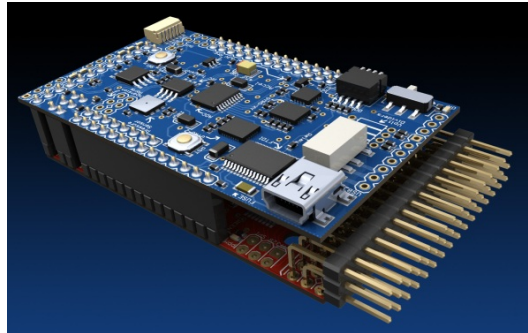


Abbildung 2.1: Autopilot Mega<sup>1</sup>

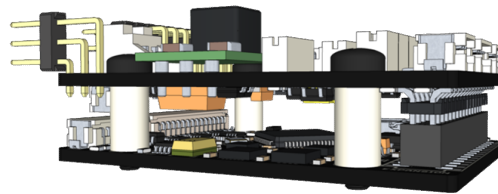


Abbildung 2.2: pixhawk-PX4<sup>2</sup>

Im Endeffekt fiel die Entscheidung auf den, da für dieses System bereits viele Informationen bezüglich Montage und auch Beispielprogrammierungen zu finden waren, (<http://code.google.com/p/ardupilot-mega/wiki/Introduction>). Zusätzlich bietet der Hersteller eine große Auswahl an externen Sensoren an. Das System liefert Messdaten wie Position, Raumlage, Ausrichtung und Geschwindigkeit. Mit einem Zusatzmodul kann der APM auch die Temperatur und den Luftdruck erfassen. Durch den offenen I2C-Bus können weitere Sensoren, wie Entfernungsmesser oder Spannungs-Stromwächter, mit dem System verbunden werden. Für den Basisaufbau wurden folgende Sensoren eingeplant:

- Beschleunigungssensor
- Neigungssensor
- Kompass
- GPS
- Geschwindigkeitssensor
- Strom- und Spannungssensor

<sup>1</sup><http://code.google.com/p/ardupilot-mega/wiki/Introduction>

<sup>2</sup><http://www.ulrichc.de/product/robotic/cu-chain-chassis/>

Teile dieser Sensoren sind beim Autopilotensystem bereits im Inertial Measurement Unit (IMU)-Shield integriert. Zur Erfassung von Geschwindigkeit, absoluter Position (GPS), Ausrichtung (Kompass) sowohl Strom und Spannung müssen Zusatzmodule montiert werden.

### Datenübertragung

Ein Schwerpunkt dieses Projektes stellt die sichere Datenübertragung der Telemetriewerte dar. Eine hohe Reichweite als auch eine große Datendurchsatzrate wurden in ihrer gesetzlichen Legalität an die Datenübertragung gefordert. Anfänglich wurden drei Frequenzen für die Kommunikation vorgesehen:

- 5,8 GHz-Bildübertragung
- 2,4 GHz-Datenübertragung
- 40 MHz-Steuerung

Dabei sollten die Bilddaten als Audio-Video (AV)-Signal im 5,8 GHz-Bereich übertragen werden. Dieses Frequenzband gilt jedoch als etwas störanfällig bei Sichtbehinderungen und wurde deshalb verworfen. Da das analoge Signal an der Bodenstation digitalisiert werden müsste, um es in die Software zu implementieren, wäre ein weiterer Nachteil. Daher wurde auf das 5,8GHz-Frequenzband vollständig verzichtet (Abbildung: 2.3).

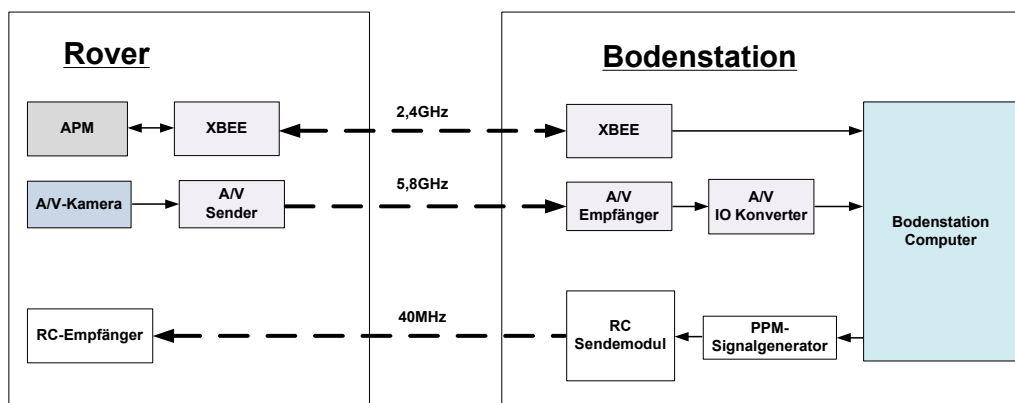


Abbildung 2.3: Kommunikationsdesign 1

Die Datenübertragung im 2.4 GHz-Band soll jedoch beibehalten werden. Hierfür waren zu Beginn die XBee-Module von „Digi“ vorgesehen (Abbildung: 2.4).

Diese Module haben den Vorteil, dass direkt eine RS232- bzw. Transistor-Transistor-Logik (TTL)-Kommunikation ermöglicht wird und diverse Adapterplatinen (Breakoutboards) leicht erhältlich sind. In weiterer Folge werden auch Bildinformationen übertragen und daher wäre neben den XBee-Modulen auch ein AV-System (z.B.: <http://www.globe-flight.de/24-GHz-A-V-Equipment>)

<sup>3</sup><http://www.digi.com/products/wireless-wired-embedded-solutions/zigbee-rf-modules/point-multipoint-rfmodules/xbee-wi-fi>



Abbildung 2.4: XBee Wireless Local Area Network (WLAN)-Modul<sup>3</sup>

notwendig. Hier müssten jedoch zusätzliche Antennen und Komponenten zur Stromversorgung angebracht werden. Aufgrund von zwei 2,4 GHz-Kanälen wäre deren gegenseitige Beeinflussung nicht abzusehen. Daher fiel die Entscheidung auf einen WLAN-Router der Firma Mikrotik (<http://routerboard.com/>), welcher über drei getrennte Local Area Network (LAN)-Schnittstellen verfügt. Da das Autopilotensystem keine LAN-Schnittstellen bereitstellt, kann dieser Router nicht direkt mit diesem verbunden werden.

Abbildung 2.5: XPort-Modul<sup>4</sup>

Die Umwandlung der APM-Daten soll mit einem XPort-Modul (Lantronix) erfolgen, welches TTL-Signale in ein entsprechendes Ethernet-Protokoll konvertiert (Abbildung: 2.5). Durch diesen Aufbau ist es möglich Daten- und Bildinformation über eine einzelne WLAN-Verbindung zu übertragen (Abbildung: 2.6). Diese Art der Konfiguration hat den Vorteil, dass nur eine Empfangsantenne notwendig ist.

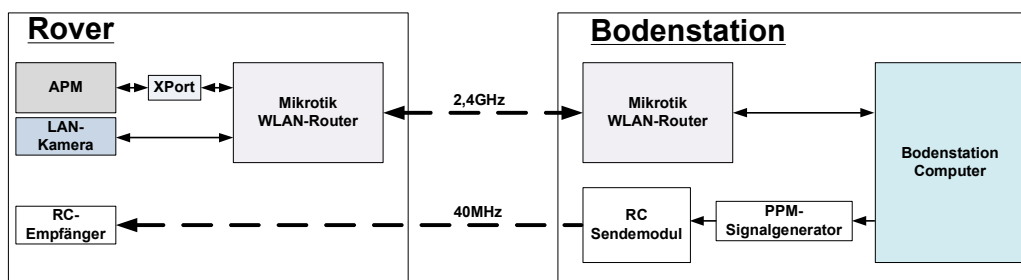


Abbildung 2.6: Kommunikationsdesign 2

<sup>4</sup><http://www.lantronix.com/device-networking/embedded-device-servers/xport.html>

## Steuerung

Um Schäden durch den Rover vorbeugen zu können, werden zwei unabhängige Steuerungssysteme benötigt. Die Hauptsteuerung erfolgt mit einem Joystick über den Bodenstationscomputer. Der RC-Pultsender ist als ein Backupsystem anzusehen. Die beiden Steuerungssysteme sollen demnach denselben Übertragungskanal nutzen und umschaltbar sein, damit eine der Steuerungen direkt die andere ablösen kann. Die Realisierung dieses Ansatzes ist mit der Herausforderung verknüpft, dass keine entsprechenden „Plug-and-Play“-Komponenten auf dem Markt zu finden sind. Die Möglichkeiten, um Steuerbefehle mit einer PC-Software zu übertragen sind vielfältig und sind jeweils mit eigenen Vor- und Nachteilen verbunden.

- Servocontroller mit Funkbrücke
- APM-Mavlink Protokoll
- Pulse Position Modulation (PPM)-Signalgenerator mit RC-Sendermodul

Der große Vorteil bei der Verwendung eines Servocontrollers mit einer Funkbrücke sind die geringen Limitierungen bei der Anzahl von Steuerkanälen. Die Kommunikationsprotokolle dieser Servocontroller verfügen über einen simplen Aufbau und sind daher leicht in die Steuerungssoftware implementierbar. Bei einer Direktverbindung mit dem Computer wird üblicherweise nur eine RS232-Schnittstelle benötigt oder virtuell angelegt. Eine solche Steuerung hat aber zwei große Nachteile: Zum einen das Problem der Legalität von Datenübertragungen über große Distanzen und zum anderen die fehlenden Möglichkeiten bei einem solchen System mit einem herkömmlichen RC-Sender einzugreifen.

Die Nutzung des Kommunikationsprotokolls vom Autopiloten (Mavlink) stellt die zweite Steuerungsmöglichkeit dar. Diese stellte sich jedoch durch die Komplexität des Protokolls als zu unsicher heraus und ist insofern auch für eine alleinige Verwendung nicht empfehlenswert. Der Vorteil des Systems ist, dass man jederzeit mit einem RC-Sender eingreifen kann.

Die Verwendung eines PPM-Signalgenerators ist die dritte und für dieses Projekt auch verwendete Alternative (Abbildung: 2.7). Der Vorteil dieser Steuerungsart ist die Möglichkeit der Kommunikation über ein Lehrer-Schüler-System am RC-Sender direkt mit einem Empfänger. Hierbei stellt der Computer den Schüler dar und übermittelt die Befehle an den Handsender. Die Firmen Flytron (<http://www.flytron.com/programmers/56-compufly-usbtppm-converter.html>) und Endurance (<http://www.endurance-rc.com/pctx.php>) bieten hierfür entsprechende Signalgeneratoren an.

Durch die Verwendung eines RC-Sendermoduls, das mit dem Signalgenerator verbunden wird, ist die Nutzung des Handsenders nicht mehr notwendig.

---

<sup>5</sup><http://www.flytron.com/programmers/56-compufly-usbtppm-converter.html>

Abbildung 2.7: PPM-Signalgenerator der Fa. Flydron<sup>5</sup>

### 2.1.2 Rover

Der Rover stellt eine Trägerplattform für die Sensoren, die Kameras sowie des Autopilotsystems dar. Daher mussten schon im Vorfeld diverse Rahmenbedingungen für die Auswahl eines geeigneten Systems festgelegt werden. Der Rover muss nicht nur eine entsprechend große Traglast für die verschiedenen Komponenten bieten, sondern auch geländetauglich sein. Auch das Antriebssystem muss für mögliche Erweiterungen genügend Leistungsreserven haben. Außerdem sollte eine gewisse Größenordnung nicht unterschritten werden. Im Vorfeld wurde nicht spezifiziert, ob die Fortbewegung des Rovers mit einem Rad- oder Kettenantrieb erfolgen sollte. Durch diese Vorgaben ergaben sich drei Baugruppen für den Rover.

- Chassis
- Trägerplattform
- Elektrik/Elektronik

#### Chassis

Bei den ersten Recherchen für ein geeignetes Chassis wurde festgestellt, dass ein käufliches Roverchassis mit den genannten Vorgaben den finanziellen Rahmen sprengen würde. Derartige Roversysteme liegen in der gewünschten Größenklasse bei mehreren tausend Euro. Aufgrund der hohen Kosten wurde in weiterer Folge entschieden, das Chassis, nach entsprechenden Vorgaben und Berechnungen, selbst zu bauen. Als Baumaterial wurde hauptsächlich leicht bearbeitbares Aluminium verwendet.

Nach intensiver Recherche wurden Baupläne für ein Radchassis (Abbildung: 2.8) und ein Kettenchassis (Abbildung: 2.9) gefunden, die auch für private Zwecke frei nachgebaut werden dürfen. Diese Chassisbaupläne wurden von Christian Ulrich unter <http://www.ulrichc.de/main/> veröffentlicht und erfüllen alle Anforderungen hinsichtlich Größe, Tragkraft, Geländegängigkeit und Materialbedarf. Die Baupläne sind nicht für einen 1:1 Nachbau ausgelegt, wurden jedoch so umfangreich und ausführlich erstellt, dass man ein vollständiges Roverchassis erhält.

<sup>6</sup><http://www.ulrichc.de/product/robotic/cu-wheel-chassis/>

<sup>7</sup><http://www.ulrichc.de/product/robotic/cu-chain-chassis/>

Abbildung 2.8: Radchassis nach UlrichC<sup>6</sup>Abbildung 2.9: Kettenchassis nach UlrichC<sup>7</sup>

Bei einer genaueren Betrachtung der verschiedenen Chassistypen entschied ich mich für das Kettenchassis mit Kippschwingenlaufwerk, einem Aluminiumwinkelrahmen, eine abgeänderte Flanschkettenvariation und eine abgeänderte Version des Schneckenantriebs. Einige Bauteile, wie im Basisplan beschrieben, sind nur schwer erhältlich und brachten deshalb Änderungen in der Konstruktion mit sich.

### Trägerplattform

Die Trägerplattform sollte als eigenständiges Modul gebaut werden, um diesen bei wechselnden Anforderungen an den Rover leichter austauschen zu können. Auch bei möglichen Änderungen in der Elektrik, Elektronik und Sensorik muss die Plattform dementsprechend kompatibel sein. Die Hauptelektronik wie Autopilot mit Sensoren, Steuerelektronik und Kommunikation müssen ebenfalls in der Plattform Platz finden. Alle Bedienungs- und Verbindungselemente sollen für mögliche Erweiterungen frei zugänglich sein. Um Beschädigungen durch Feuchtigkeit oder feste Fremdkörper zu vermeiden, wird die eigentliche Elektronik zusätzlich gekapselt. Für verschiedene Tests und eventuellen Anpassungen muss auch die Elektronik modular aufgebaut werden. Als Basismaterial wird wieder Aluminium und beschichtetes Sperrholz verwendet.

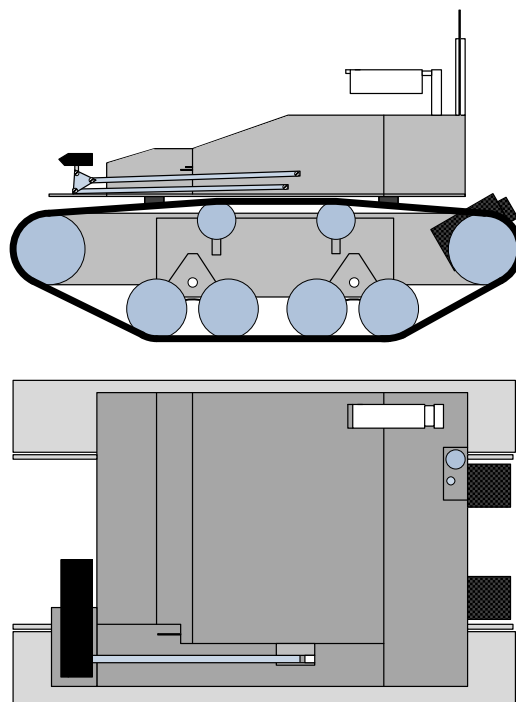


Abbildung 2.10: Design des Chassis mit Trägerplattform

Abbildung 2.10 zeigt das Design des Roverchassis sowie der montierten Trägerplattform.

### 2.1.3 Antennennachführung

Wie bereits beschrieben wird eine Richtfunkantenne zur Erhöhung der Reichweite bei der Datenübertragung verwendet werden. Da diese Antenne auf die Empfangsstation ausgerichtet werden muss, wird eine entsprechende Mechanik zum Einsatz kommen. Diese Mechanik erlaubt eine Ein- oder Zweiachsbewegung der Antenne in der Horizontal- und bzw. oder in der Vertikalebene. Es sind verschiedene Nachführungssysteme aus dem Fly per Video (FPV)-Sektor erhältlich, die jedoch mehrere konstruktionsbedingte Nachteile haben können. Die Montageflächen liegen bei solchen Systemen entweder außerhalb der Neigungsachse (Abbildung: 2.11.a), wodurch sich große Hebelkräfte ergeben können oder sie haben eine innenliegende Montagekonsole wie in Abbildung: 2.11.b gezeigt, wodurch der Einsatz mehrerer Antennen nur schwer möglich ist.

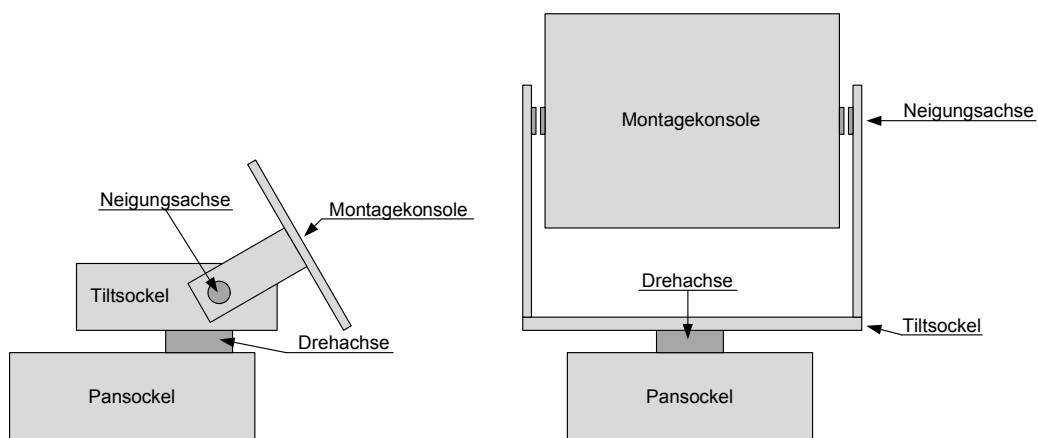


Abbildung 2.11: a) aussenliegende Montagekonsole; b) innenliegende Montagekonsole

Da die Nachführungssysteme nur selten eine innenliegende Kabelführung erlauben ist ebenfalls ein Nachteil der Konstruktion. Wegen den freiliegenden Verbindungsleitungen kann es leicht zu Beschädigungen kommen. Aufgrund von hohen Anschaffungskosten und der konstruktionsbedingten Nachteilen solcher Nachführungssysteme, ist die Eigenentwicklung einer selbständigen Mechanik ratsam. Die Vorgaben an die Mechanik sind:

- Dreh- und Neigungsachse
- seitlich aussenliegende Montagekonsolen
- geschlossenes Gehäuse
- innenliegende Leitungsführung

Aus diesen Überlegungen heraus entstand ein neues Design für eine Nachführungsmechanik (Abbildung: 2.12).

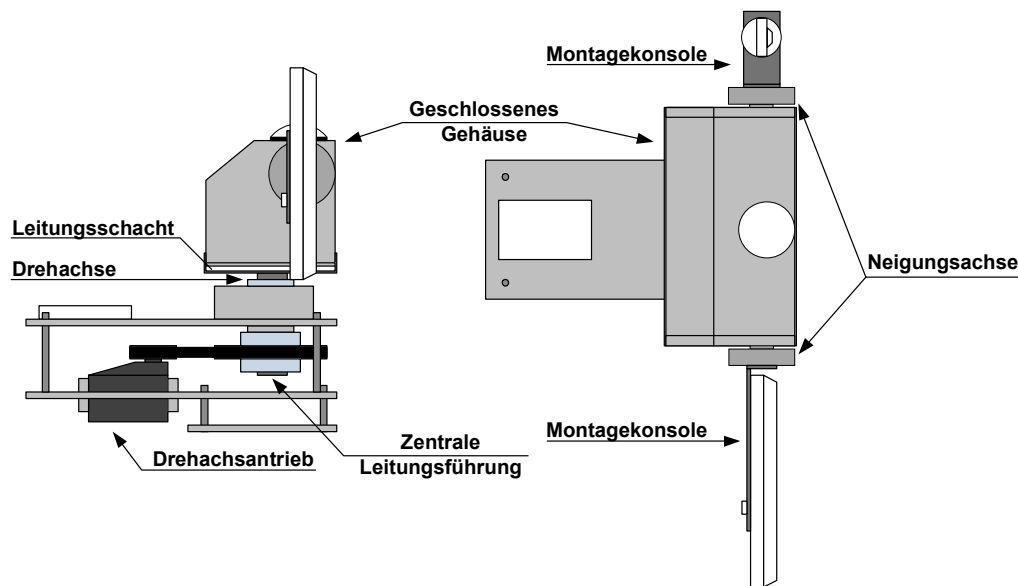


Abbildung 2.12: Mechanikkonzept

Diese Mechanik bietet zudem die Möglichkeit, die Montagekonsolen an den jeweiligen Verwendungszweck anzupassen. Zudem erlauben die in der Drehachse geführten Leitungen eine mehrmalige horizontale Drehung der Neigungsachse.

### Nachführungsparameter

Zur Bestimmung der Dreh- und Neigungswinkel wird eine erweiterte Form des Positionierungsverfahrens (Kapitel: 3.3) verwendet. Die Erfassung der Nachführungsposition und der Ausrichtung soll unabhängig vom Rover durch einen Global Positioning System (GPS)-Empfänger bzw. einem Magnetfeldkompass erfolgen. Dadurch ist es zusätzlich möglich, die Nachführung auch für andere Aufgaben wie beispielsweise Satellitentracking zu verwenden. Die Roverdaten werden durch eine herkömmliche WLAN-Verbindung übertragen. Die Berechnung der Nachführungswinkel sowie die Ansteuerung der Nachführung erfolgt ausschließlich durch den Bodenstationscomputer. Da am Computer ein direkter Zugriff auf die Rohdaten der einzelnen Sensoren erfolgen kann, ergeben sich weitere Auswertungsmöglichkeiten.

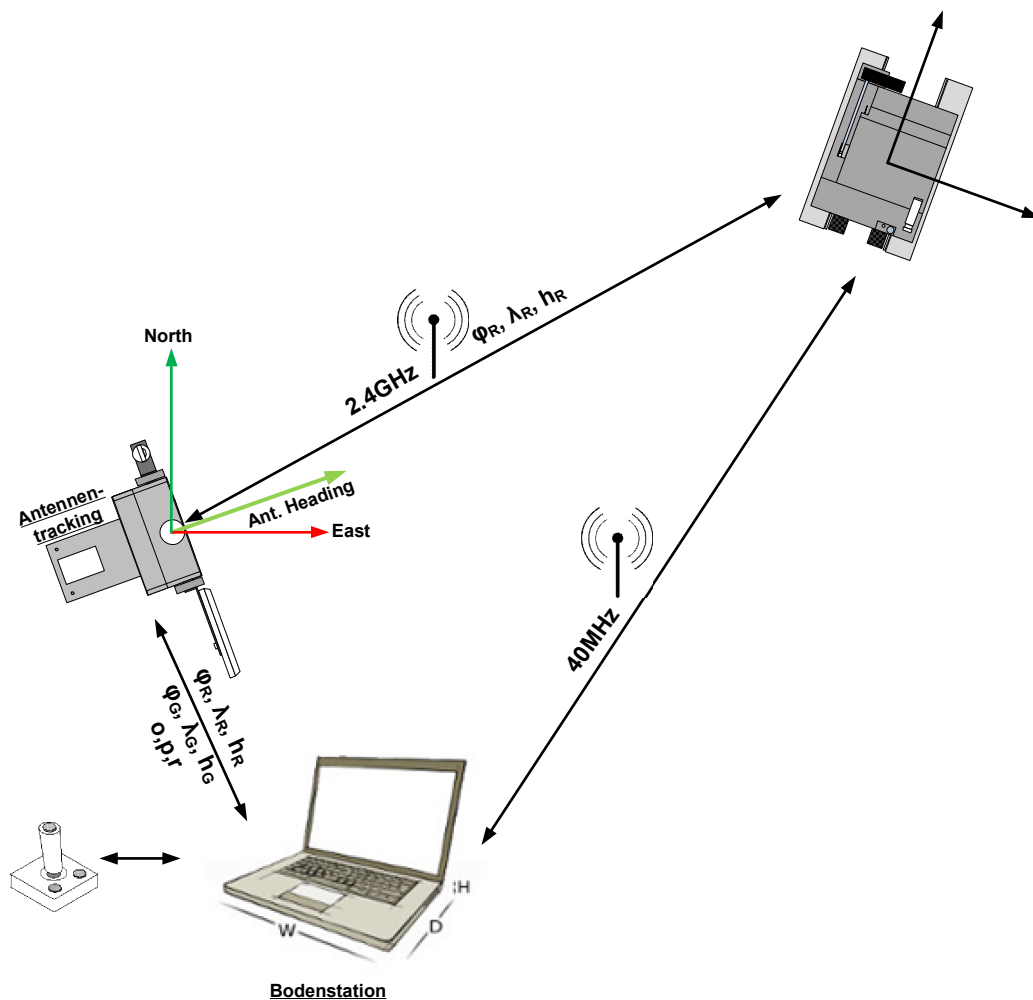


Abbildung 2.13: Antennentracking

Abbildung 2.13 zeigt das vollständige Design der Datenübertragung und der Steuerung.

#### 2.1.4 Bodenstation

Da im Bereich der Telemetrie-Datenübertragung ein anderer Ansatz verfolgt wird, ist die Verwendung einer handelsüblichen Bodenstation nicht möglich. Diese Bodenstationen sind nicht für den zentralen Einsatz eines Computers ausgelegt. Die Bodenstation für dieses Projekt musste völlig neu konstruiert werden, da sie nicht nur zur Ansteuerung der Antennennachführung sondern auch als Steuerzentrale verwendet werden sollte. (Abbildung 2.14).

Im Chassis wurden alle Komponenten für die Bodenstation untergebracht. Die Anschlüsse der externen Geräte befinden sich an einem Anschlussterminal. Mit entsprechenden Kippschaltern können die einzelnen Elektronikmodule separat geschaltet werden. Das Design der Datenübertragung und der Steuerung wurde bereits im Kapitel 2.1.1 beschrieben und deshalb



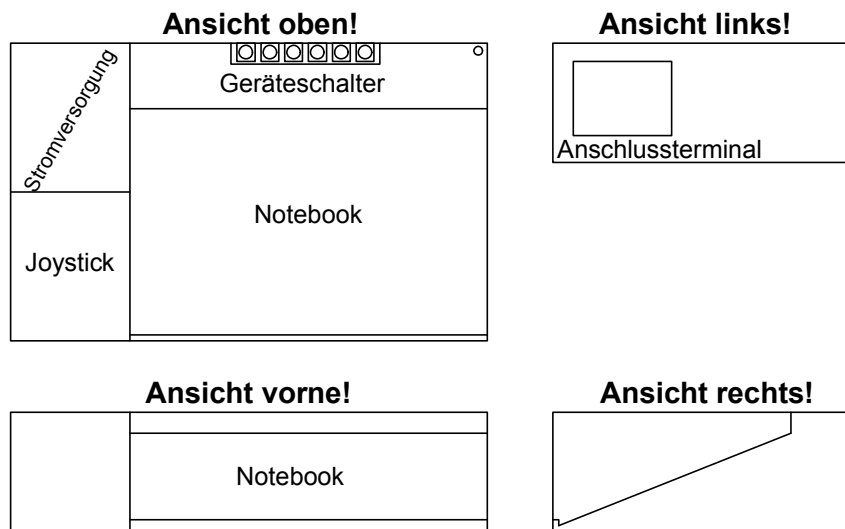


Abbildung 2.14: Konzept Bodenstationschassis

nicht erneut angeführt.

## 2.2 Software

Der Autopilot verwendet zur Kommunikation ein MAVLink-Datenformat, und soll mit der Software „APM Planner“ oder „QGroundcontrol“ visualisiert werden. Diese Softwarepakete sind frei erhältlich und für die Kommunikation mit dem vorgesehenen Autopiloten ausgelegt. Da beide Softwarepakete nur mit einigem Aufwand für die Steuerung der Antennennachführung erweitert werden können, wurde entschieden, dass eine eigenständige Trackersoftware erstellt wird und parallel zur Visualisierung betrieben wird. Das bedeutet, dass die Roverdaten in zwei unabhängige Datenstreams aufgesplittet werden müssen.

Für die Trackersoftware wurden im Vorfeld verschiedene Anforderungen definiert. Als Programmiersprache sollte C/C++ und die Entwicklungsumgebung Visual Studio verwendet werden. Das Ziel ist wie bei den mechanischen und elektronischen Komponenten, eine möglichst modulare Programmstruktur. Die Software soll so gestaltet sein, dass Erweiterungen leicht möglich sind und einzelne Programmstrukturen wiederverwendet werden können. Zur weiteren Vereinfachung der Portabilität sind diese einzelnen Software-Module als statische Bibliotheken zu kompilieren. Das Softwaredesign sieht dabei fünf Hierachiestufen für die vollständige Steuersoftware der Antennennachführung vor (Abbildung 2.15).

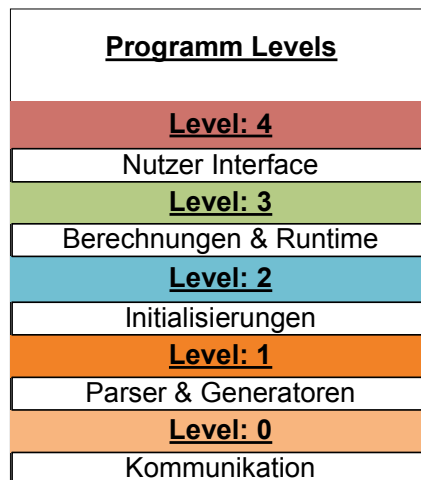


Abbildung 2.15: Konzeptaufbau Software

Die prinzipiellen Anforderungen an die Software sind:

- Einlesen und Parsen der GPS-Positionen der Antennennachführung
- Einlesen und Parsen der Kompassdaten der Antennennachführung
- Einlesen und Parsen der Roverdaten
- Ausreisserdetektion der Positions- und Orientierungsdaten
- Visualisierung diverser Parameter
- Berechnung der Antennenausrichtungparameter
- Steuerung der Antennennachführung

Um den Programmieraufwand in einem entsprechenden Rahmen zu halten wird versucht vorhandene oder frei verfügbare Programmteile bzw. Bibliotheken zu nutzen und diese an die Software anzupassen. Desweiteren soll zur Signalgenerierung der RC-Steuerung die vorhandene „Flydron“-Software verwendet werden.

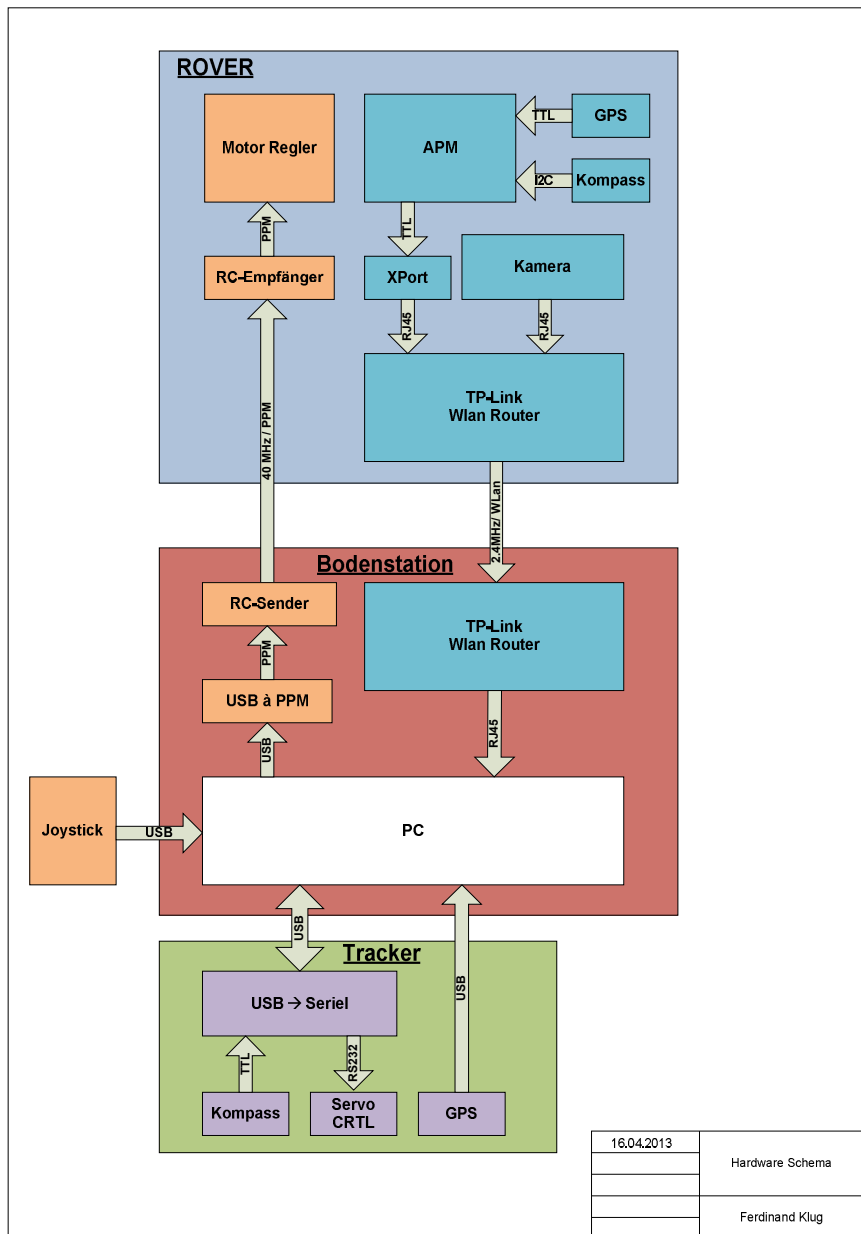


Abbildung 2.16: Konzeptaufbau Kommunikation

Abbildung 2.16 zeigt das schlussendliche Konzept mit den Kommunikationspfaden und den Elektronikmodulen. Zusätzlich beinhaltet das Schema die verschiedenen Kommunikationspegel und Schnittstellen des vollständigen Systems.

# Kapitel 3

## Grundlagen

### 3.1 UV-Autopilotsysteme

Ein Autopilotsystem ist das Kernelement vieler Verbewegungsmittel. Diese Systeme sind in der Raum-, Luft- und Schifffahrt als auch versuchsweise in Autos zu finden. Ein Autopilotsystem übernimmt die autonome Steuerung eines Fahrzeuges nach definierten Regeln. Dabei sind die Anforderungen an das System je nach Einsatzgebiet sehr unterschiedlich. Die Bücher „Navigation“ von Bernhard Hofmann-Wellenhof *et al.* (2003) und „Unmanned Rotorcraft Systems“ von Guowei Cai *et al.* (2011) geben einen detaillierten Einblick in diese Thematik und in die Berechnungsschema zur Inertialnavigation.

In diesem Abschnitt werden die Grundlagen anhand von Autopilotsystemen beschrieben, die im Modellbau und der Robotik üblich sind. Haupteinsatzgebiet dieser Autopilotsysteme sind Quadro-, Hexa-, Oktokopter, Modellflugzeuge oder Modellhubschrauber. Beim Begriff Autopilotsystem wird implizit von einer autonomen Steuerung im jeweiligen Fahrzeug ausgegangen. Durch die Verwendung derselben Elektronik wird im Modellbaubereich jedoch kein Unterschied zwischen der Lagestabilisierung und der autonomen Steuerung gemacht.

Ein Autopilotsystem besteht immer aus einer Prozessoreinheit sowie diversen Schnittstellen und Sensoren. Die eingesetzten Sensoren werden durch das Aufgabengebiet des Systems bestimmt und können mit den Schnittstellen und der Software meist erweitert werden. Wird das System zur reinen Lagestabilisierung verwendet, ist die Verwendung von gyroskopischen Sensoren ausreichend. Soll das System zur autonomen Steuerung verwendet werden, sind zumindest zusätzliche Beschleunigungssensoren nötig. Die Kombination aus gyroskopischen Sensoren und Beschleunigungssensoren wird auch als Inertial Measurement Unit (IMU) bezeichnet. Aufgrund des Driftverhaltens solcher IMU's können auch weitere Sensoren wie Magnetometer und GPS oder GNSS zur Unterstützung eingesetzt werden. Im Bereich der Quadro-, Hexa-, und

Octocopter- Lagestabilisierung können auch optische Sensoren zum Einsatz kommen. Ein solches, erweitertes System wird in der Literatur (David Titterton & John Weston, 2004) auch als Attitude and heading reference system (AHRS) beschrieben. Die Kombination aus einer IMU mit einem AHRS und einer Navigationsberechnung wird als Inertial navigation system (INS) bezeichnet (Abbildung: 3.1).

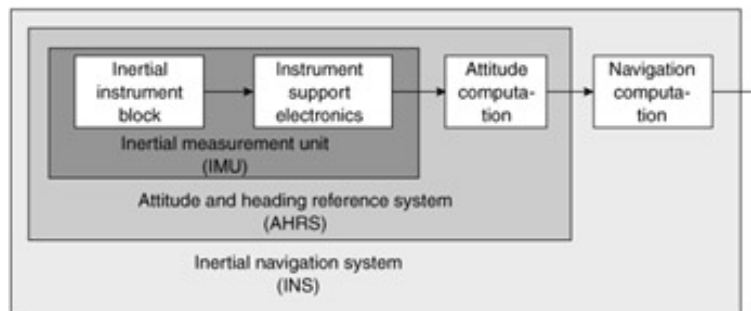


Abbildung 3.1: Blockdiagramm Inertial navigation system<sup>1</sup>

Die derzeit erhältlichen Autopilotensysteme, welche in Dronen oder im Bereich des Modellbaus eingesetzt werden, bieten meist Schnittstellen für verschiedenste Sensoren. Solche externen Sensoren sind beispielsweise:

- Strom-/Spannungssensor
- Barometer
- Temperatursensor
- Staudrucksensor
- optische, akustische, elektrische Distanzsensoren
- uvm.

Mit den Sensordaten errechnet die Proessoreinheit die aktuelle Position, die Raumlage, die Geschwindigkeit und die Rollraten. Vergleicht man diese Daten mit Zielpunktkoordinaten, lassen sich in einem Regelzyklus, Steuerbefehle für die Aktoren berechnen, welche das Fahrzeug steuern. Je nach Anwendung und Prozessorleistung werden zwei Algorithmen für die Berechnungen verwendet. Der Kalman-Filter wird zur Schätzung von zeitlich variablen Zufallsgrößen verwendet und geht auf einen stufenweisen Ausgleich zurück (Abbildung: 3.2) (nach Roland Pail, 2007). Bei der Verwendung eines Kalman-Filters ist der Vorteil, dass Datenlücken mit der Prädiktion überbrückt werden und eine vollbesetzte Kovarianzmatrix geschätzt wird.

<sup>1</sup><http://www.globalspec.com/reference/26565/203279/chapter-9-strapdown-system-technology>

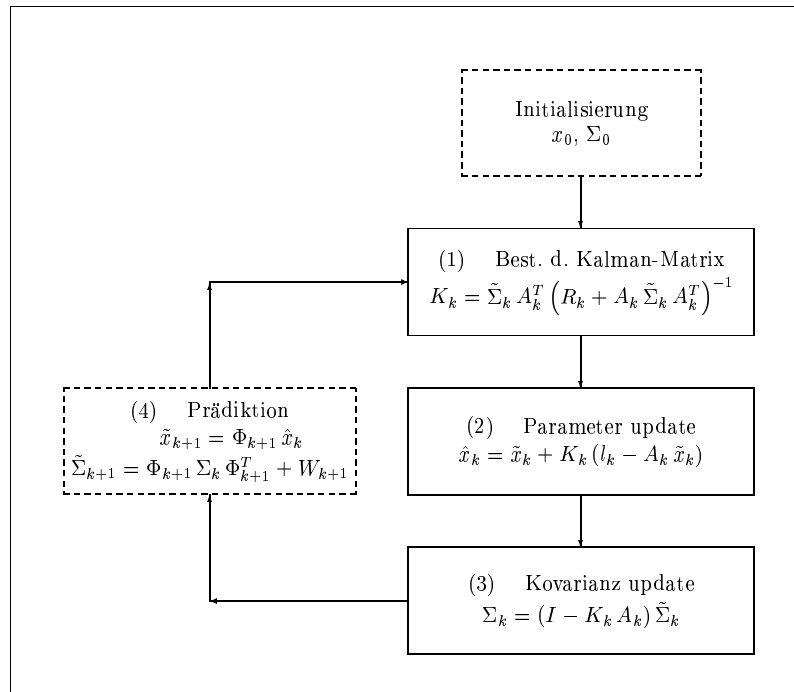


Abbildung 3.2: Berechnungsschema Kalman Filter Algorithmus<sup>2</sup>

Die Alternative zum rechenaufwändigen Kalman-Filter stellt der Direct Cosine Matrix (DCM)-Komplementär-Filter Algorithmus dar. Wie aus Abbildung: 3.3 ersichtlich, verwendet dieser Algorithmus nur die aktuell vorliegenden Daten. Es werden keine Prädiktionen oder Kovarianzen geschätzt, wodurch es bei Datenlücken keine Überbrückung gibt. Im Gegensatz zum Kalman-Filter, der auf der Ausgleichsrechnung basiert, wird in diesem Algorithmus mit Rotationsmatrizen gerechnet. Einen guten Einstieg in das Berechnungsschema liefern die Publikationen von: Erick Macias *et al.* (2012), William Premerlani & Paul Bizard (2009) und Starlino (2001).

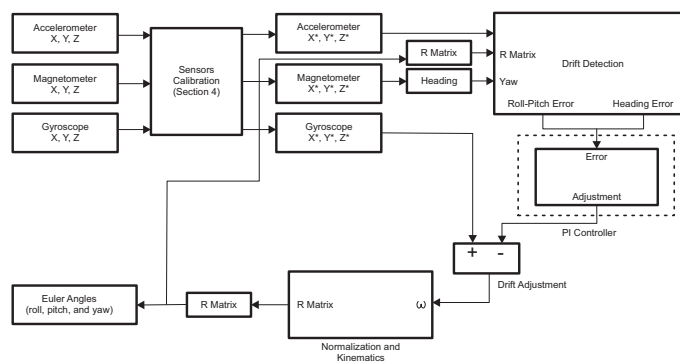


Abbildung 3.3: Berechnungsschema DCM-Komplementär Filter Algorithmus<sup>3</sup>

Der geringere Prozessieraufwand ist auch ein Grund, weshalb gerade im Bereich der Unmanned Vehicle (UV)-Autopilotensysteme der DCM-komplementär-Filter Algorithmus bevorzugt wird.

<sup>2</sup>Roland Pail (2007)

<sup>3</sup>Erick Macias *et al.* (2012)

## 3.2 Datenübertragung

Bei der vorliegenden Diplomarbeit hat die Datenübertragung einen sehr hohen Stellenwert. Daher wird in den entsprechenden Kapiteln (2.1.1; 4.4; 6) auf die tatsächliche Implementierung eingegangen. Dieses Kapitel soll lediglich einen Überblick über die verschiedenen genutzten Systeme geben, da gerade in diesem Bereich sehr gute und einschlägige Referenzen vorhanden sind. (siehe Klaus Dembowski, 2007, Hans-Jürgen Gevatter & Ulrich Grünhaupt, 2006, Ralf Gessler & Thomas Krause, 2009, Lammert Bies, 2010, u.v.m)

Das Kapitel beinhaltet die benötigten Grundlagen zur seriellen Datenkommunikation, die grundsätzliche Handhabung von Datenprotokollen sowie Datenübertragungsverfahren. Detaillierter werden die Übertragungsverfahren des Radio Controll (RC)- Steuerungssignales sowie die RS232 bzw. TTL Datenschnittstellen behandelt. Einen kurzen Einblick in die technischen Anforderungen an das jeweilige System bietet ebenfalls der Abschnitt 3.2.1. Um Begriffsverwirrungen vorzubeugen, wird der Begriff der „seriellen Schnittstelle“ nur für die physikalische Übertragungsmethode verwendet, nicht jedoch für einen speziellen Standard (3.2.1).

Im ersten Abschnitt 3.2.1, werden allgemein die Datenübertragungsverfahren und Trägerfrequenzen erläutert. In den folgenden Unterabschnitten wird auf die tatsächlich verwendeten Verfahren eingegangen. Der zweite Abschnitt gibt einen Überblick über die verwendeten Datenprotokolle, sowie deren Handhabung. Eine detailliertere Beschreibung der Verarbeitung der eingehenden Datenpakete wird im Kapitel 6 gegeben.

### 3.2.1 Datenübertragungsverfahren

Ein Kernpunkt dieser Diplomarbeit stellt die sichere Datenübertragung sowie Steuerbefehlsübertragung dar. Aufgrund der Größe und Masse des entwickelten Rovers ist der Steuerung nochmals eine erhöhte Aufmerksamkeit zu schenken. In Österreich sind lediglich einige wenige Frequenzbänder zur allgemeinen Nutzung unter Limitierungen hinsichtlich der Abstrahlleistung (Equivalent isotropically radiated power (EIRP)) freigegeben. Für spezielle Anwendungen, beispielsweise Amateurfunk, wurden diese Richtlinien etwas entschärft, jedoch auch nur nach einer erfolgreich abgelegten, amtlichen Amateurfunkprüfung. Als teilweise frei nutzbare Frequenzbänder gelten in Österreich (nach: Bundesministerium für Verkehr, Innovation und Technologie (bmvit); <http://www.bmvit.gv.at/telekommunikation/recht/aut/verordnungen/fnv.html>):

- 25 MHz-Band
- 433 MHz-Band

- 868 MHz-Band
- 2,4 GHz-Band (Industrial, Scientific and Medical Band (ISM) oder Wireless Local Area Network (WLAN))
- 5,8 GHz-Band (ISM oder WLAN)

Gerade im Bereich des ISM- oder WLAN-Bandbereichs sind strenge Richtlinien in Bezug auf Abstrahlleistungen, Protokollierung und auch Modulierung der Nutzdaten vorhanden. Die niedrigeren Frequenzbänder im MHz-Bereich, sind lediglich durch die Sendeleistung limitiert. Frequenzbänder wie das 35 MHz-Band und eingeschränkt das 40 MHz-Band sind ausschließlich dem RC-Modellbetrieb vorbehalten.

Die Limitierungen, beispielsweise der Abstrahlleistung, sind dahingehend als durchaus sinnvoll zu erachten, da es ansonsten kaum mehr möglich wäre, drahtlose Kommunikation oder Steuerungen, aufgrund von Störungen anderer Sendeanlagen zu realisieren. Wesentlich wichtigere Entscheidungshilfen für den Nutzer zur Auswahl einer im gesetzlichen Rahmen geeigneten Frequenz, sind die erreichbaren Reichweiten und Datendurchsatzraten, sowie die Ausfall- bzw. Störungsempfindlichkeiten. Als Faustregel kann man annehmen, je höher die Frequenz, desto höher wird die Datendurchsatzrate, desto geringer wird die Reichweite und schlussendlich, desto höher wird der Protokollaufwand. Ein positiver Effekt bei den aufwendigeren Protokollimplementierungen ist, dass diese Datenübertragung als relativ störungsempfindlich gegenüber natürlichen Störquellen angesehen wird (WLAN). Ausgenommen sind mutwillige Störungen oder Beeinflussungen des Signals durch Dritte.

Für die weiterführende Arbeit wurden letztendlich ein 40 MHz-Frequenzband sowie ein 2,4 GHz-Band gewählt. Die Gründe für diese Entscheidung werden in Kapitel 4 genauer erläutert, gehen jedoch mit der Verfügbarkeit sowie den Kosten einher. Bei den verwendeten Datenübertragungsverfahren sowie deren Datenprotokolle werden folgende, einführende Grundlagen erläutert.

- RC
- WLAN
- Serielle Schnittstelle

In der oben erwähnten Aufzählung sind die Sonderformen der Datenübertragung, wie unterschiedliche Signalpegel, nicht separat angeführt. Auch wird in dieser Arbeit nicht auf die Grundlagen der WLAN-Kommunikation und ihrer Standards im Speziellen eingegangen, da dies den Rahmen meiner Ausführungen sprengen würde.



### RC-Steuerung

Bei einer RC-Steuerung handelt es sich um eine kabellose Datenübertragungsmethode, welche in erster Linie im Modellbau verwendet wird. Dabei sind diese Fernsteuerungen meist auf wenige Schalt- oder Steuerkanäle beschränkt. Diese Art der Steuerung ist auf eine hohe Reichweite, sowie eine möglichst hohe Störuneempfindlichkeit ausgelegt, da es bei einem Signalverlust schnell zur Zerstörung eines Modells oder zu Unfällen mit enormen Schäden kommen kann. In Österreich sind derzeit vier Frequenzen für den Betrieb einer RC-Steuerung vorgesehen (siehe Seite: 24). Die tatsächlichen Modulationsverfahren zur Informationsübertragung sind hierbei stark von den Herstellern abhängig, beruhen jedoch auf zwei Basismodulationstypen: Zum Einen die Amplitudenmodulation (AM) und zum Anderen die Frequenzmodulation (FM).

Bei der AM wird die Amplitude der Trägerfrequenz durch die zu übertragende Information beeinflusst (Abbildung: 3.4). Dieses Modulationsverfahren ist jedoch sehr störanfällig, da eine Änderung der Signalstärke eine scheinbar geänderte Information darstellt. Durch diese Störanfälligkeit findet diese Art der Modulation nur in preisgünstigen Modellspielzeugen Verwendung.

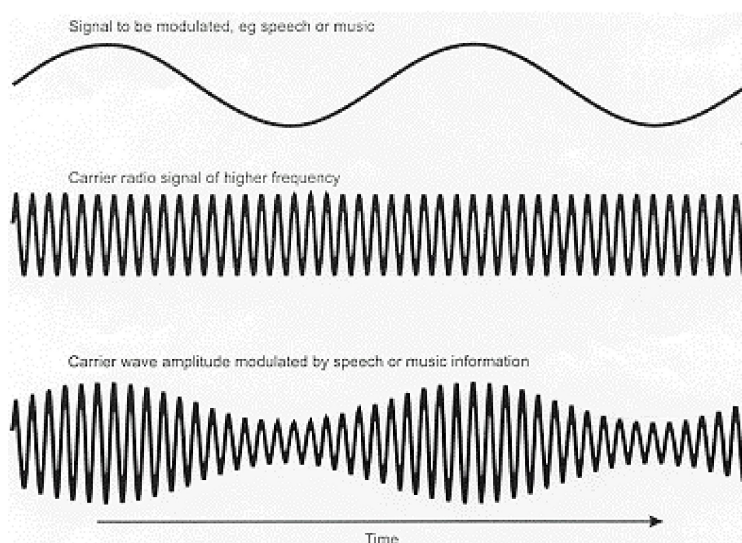
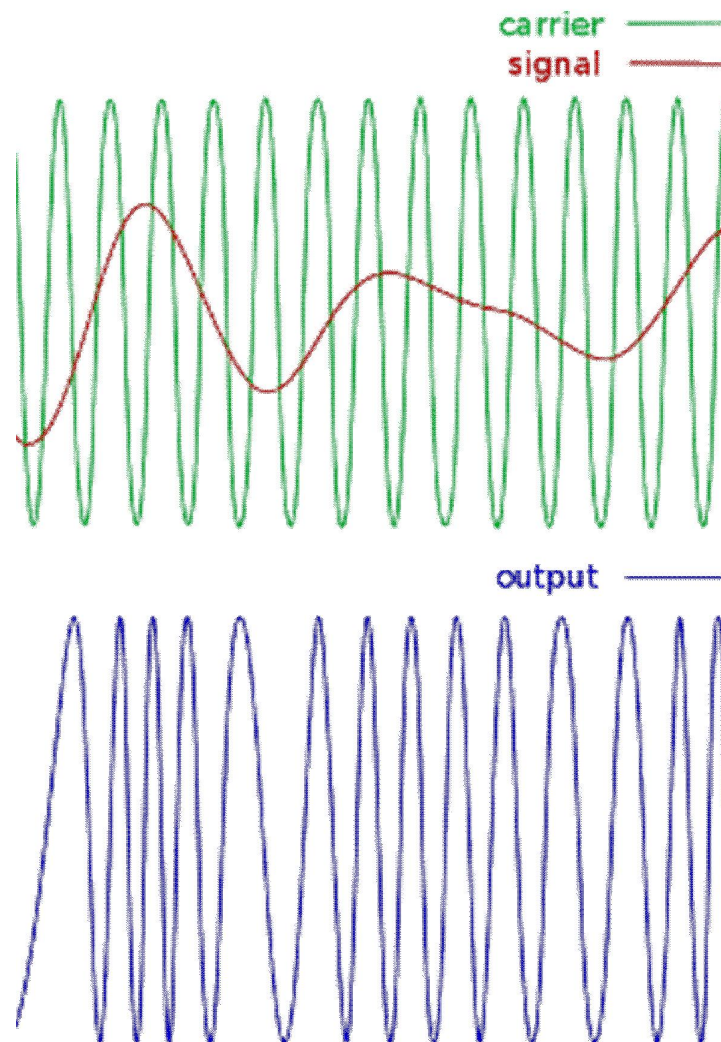


Abbildung 3.4: Amplitudenmodulation<sup>4</sup>

Eine beliebtere Anwendung ist die FM (siehe Abbildung: 3.5) mit ihren unterschiedlichen vom Hersteller abhängigen Varianten. Bei dieser Modulation wird die Trägerfrequenz durch die zu übertragenden Information beeinflusst. Da die Frequenz selbst die Information trägt, ist dieses Verfahren nicht anfällig auf mögliche Änderungen in der Signalstärke. Es ist jedoch zu beachten, dass man für die Datenübertragung keine exakte Frequenz, sondern ein Frequenzband benötigt.

<sup>4</sup><http://tele-information.com/wp-content/uploads/2012/10/Amplitude-modulation-AM.gif>

<sup>5</sup><http://tele-information.com/wp-content/uploads/2012/10/Frequency-Modulation.png>

Abbildung 3.5: Frequenzmodulation<sup>5</sup>

Als Trägerfrequenzen werden im Modellbau folgende vier Basisfrequenzen mit unterschiedlichsten Kodierungsverfahren angewendet:

- 25 MHz (Spielzeug, Auto, Schiffe)
- 35 MHz (Auto, Schiffe, Flugzeuge, Hubschrauber)
- 40 MHz (Auto, Schiffe, eingeschränkt; Flugzeuge, Hubschrauber)
- 2.4 GHz (Auto, Schiffe, Flugzeuge, Hubschrauber)

Die im MHz-Bereich arbeitenden RC-Sender sind genau genormt. Da die einzelnen Senderkanäle bei diesen Anlagen über Quarze reguliert werden, muss jeder Nutzer sicherstellen, dass sich innerhalb der Reichweite eines Senders nur ein Sendefrequenzband für dessen Nutzung befindet. Sollten zwei oder mehrere Nutzer dasselbe Frequenzband belegen, kommt es zu gravierenden Störungen bzw. Übersteuerungen. Im schlimmsten Fall führt das zu einer Zerstörung

des Modells. Zusätzlich sind die 25 MHz-Sendeanlagen nur auf Boden gestützte Fahrzeuge beschränkt, da diese Anlagen sehr störanfällig sind. Hingegen sind die 35 MHz- und 40 MHz-Anlagen auch für Flugzeuge oder Hubschrauber freigegeben. In Österreich sind im 35 MHz-Band 19-Sendefrequenzen (Kanal Nummer: 61-79) und im 40 MHz-Band 4-Sendefrequenzen (Kanal Nummer: 50-53) für den Flugbetrieb zugelassen. Diese Senderanlagen verwenden die zuvor beschriebene FM zur Datenübertragung. Durch die unterschiedlichen Umsetzungen der Kodierung, gibt es eine Vielzahl an verwendeten Abkürzungen und Begriffen. Zu nennen sei hier das FM Single Superhead System (SSS) der Firma Graupner oder das FM Double Superhead System (DSS) der Firma Futaba. Alle Senderanlagen können ausschließlich analoge Signale ausstrahlen, sie besitzen keine Empfangseinheit. Somit ist nur eine unidirektionale Kommunikation möglich.

Wie eben beschrieben, handelt es sich bei einer RC-Steuerung um eine analoge Datenübertragung. Die zu übertragende Information wird als PPM-Frame bezeichnet. Dabei handelt es sich um die Summe einzelner Steuerimpulse, welche auch Pulse Width Modulation (PWM) genannt werden. Jedes dieser PPM-Frames hat eine Länge von 22.5 ms und ist unterteilt in eine Startinformation, in mehrere Kanalinformationen, sowie einer Stoppinformation. Eine Kanalinformation hat eine Pulsdauer von  $0.7 \leq ms \leq 1.7$  und die dazugehörige Stoppinformation hat eine Pulsdauer von 0.3 ms (siehe: 3.6). Je nach Sender-Empfänger-Hersteller gibt es zwei unterschiedliche Signalpegel, den positiven PPM-Frame und den negativen PPM-Frame. Der positive PPM-Frame bedeutet, dass die Kanalinformation den Pegelstand High und die Stoppinformation in den Pegelstand Low geht. Das negative PPM-Frame ist hingegen genau invertiert.

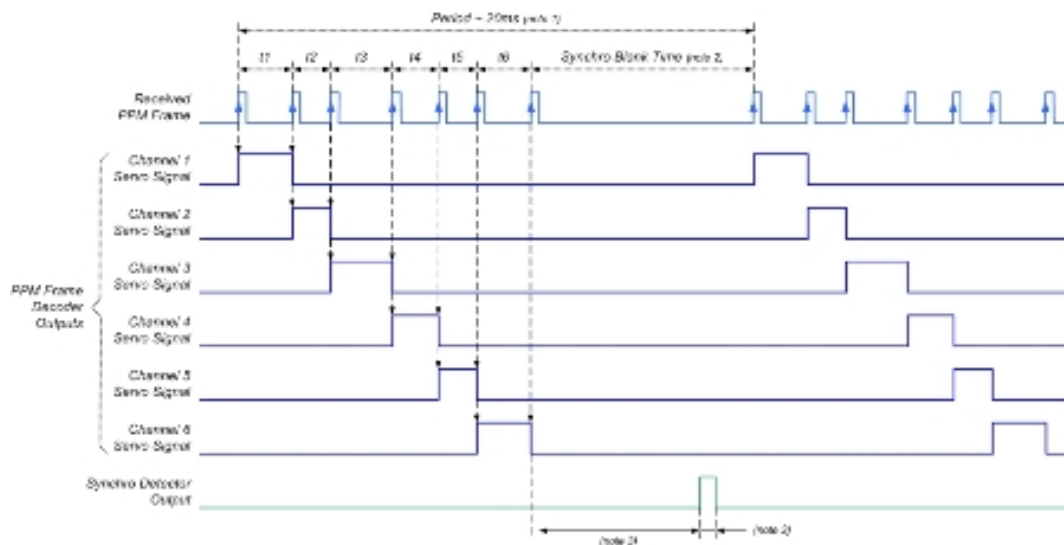


Abbildung 3.6: Negatives PPM-Frame<sup>6</sup>

Eine große Erneuerung in diesem Bereich stellt die 2.4 GHz-Technologie dar. Diese Senderanlagen verwenden ein digitales Übertragungsprotokoll und sind größtenteils auch für bidirektionale Kommunikation ausgelegt. Wie auch für die im MHz-Bereich arbeitenden Sender-

<sup>6</sup><http://diydrones.com>

anlagen gibt es keine einheitlichen Standards oder Normen für Übertragungsprotokolle und die eigentliche Datenübertragung. Es müssen lediglich die gesetzlichen Richtlinien hinsichtlich Abstrahlleistungen für die Verwendung von 2,4 GHz-Anlagen eingehalten werden. In Europa stehen für eine 2,4 GHz RC-Senderanlage Frequenzen von  $2,4 \leq GHz \leq 2.4835$  zur Verfügung. Diese Grundvoraussetzungen machen es nötig, spezielle Übertragungsverfahren zu entwickeln, um den sicheren Betrieb zu gewährleisten. Abbildung 3.7 zeigt eine Messung des Rauschpegels ohne den Einfluss eines WLAN-Routers oder einer RC-Anlage.

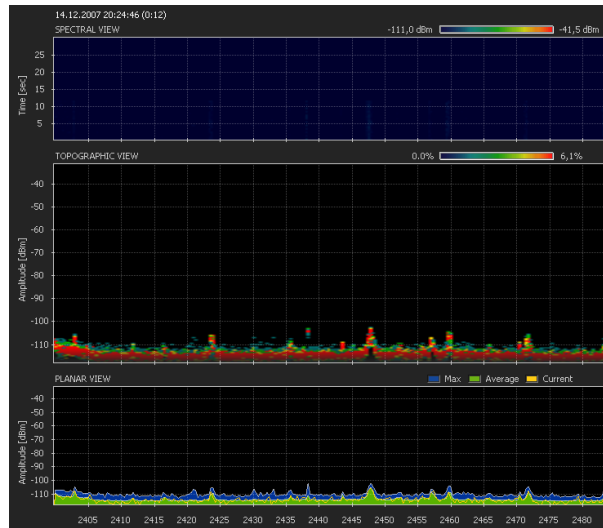


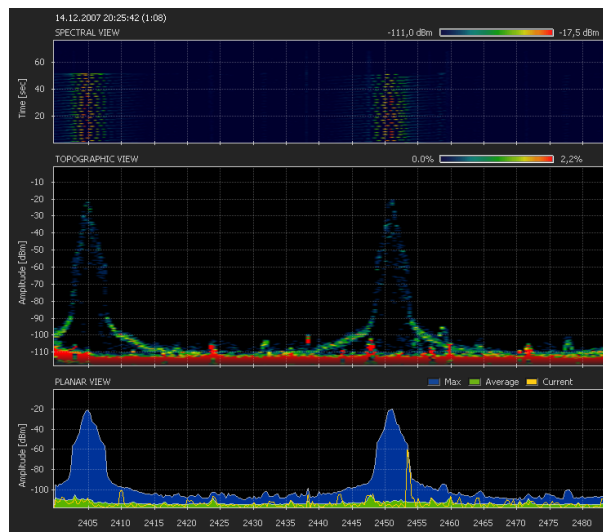
Abbildung 3.7: 2.4 GHz Ruhepegel<sup>7</sup>

Da in diesem Frequenzbereich auch die WLAN-Datenkommunikation stattfindet, ist es nötig, die Sender- und Empfangsanlagen genau aufeinander abzustimmen. Diese Abstimmung legt das Frequenzband fest, welches zur eigentlichen Informationsübermittlung herangezogen wird. Das „einloggen“ zwischen Sender und Empfänger kann mittels dreier verschiedener Methoden erfolgen. Bei den hier genannten Verfahren, müssen Sender und Empfänger das jeweilige System unterstützen:

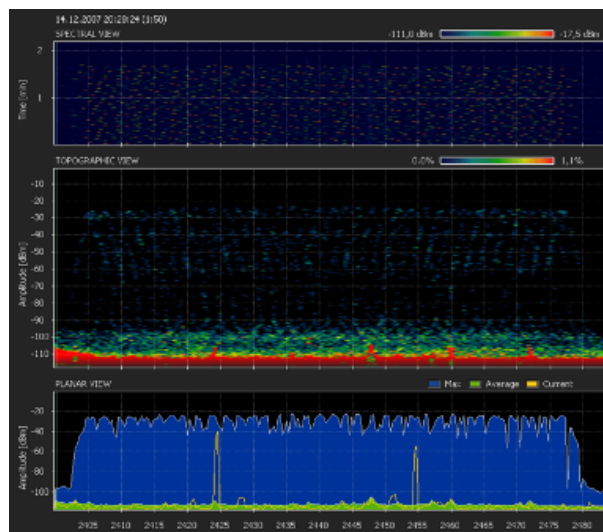
- Direct Sequence Spread Spectrum (DSSS)
- Frequency Hopping Spread Spectrum (FHSS)
- Hybride Formen

Beim DSSS wird zur eigentlichen Information eine vordefinierte Bitfolge aufmoduliert, wodurch das Frequenzband gespreizt wird (siehe Abbildung: 3.8). Die Information bleibt bei diesem Verfahren unterhalb des Umgebungsrauschpegels. Die Senderanlage scannt das gesamte 2.4 GHz-Spektrum und wählt eine freie Frequenz aus, welche anschließend belegt wird. Der Empfänger filtert die Informationen aus dem empfangenen Signal heraus. Stellt sich heraus, dass das Signal gestört wird, wird auf eine andere Frequenz gewechselt.

<sup>7</sup><http://wiki.rc-heli-fan.org/index.php/Datei:Ruhe.png>

Abbildung 3.8: 2.4GHz DSSS-Pegel<sup>8</sup>

Das FHSS geht hingegen einen völlig anderen Weg zur Datenübertragung. Dabei wird die Information nicht durch eine Spreizung des Signals um die Basisfrequenz übermittelt, sondern es werden viele einzelne Kanäle nacheinander verwendet (Abbildung: 3.9). Dabei durchlaufen Sender und Empfänger das gesamte Frequenzband. Sender und Empfänger müssen synchron die einzelnen Kanäle wechseln. Ein weiterer Unterschied zum DSSS ist, dass das Signal über dem Rauschpegel liegen muss, um eine Kommunikation möglich zu machen.

Abbildung 3.9: 2.4GHz FHSS-Pegel<sup>9</sup>

Bei den hybriden Formen kommt es zu einer Vermischung beider Systeme. Das bedeutet, dass ein Sender ein Spreizbandsignal zur Informationsübertragung verwendet und die Basisfrequenz kontinuierlich ändert. Allgemein wird den Anlagen im 2.4 GHz-Bereich eine hohe Störunempfindlichkeit nachgesagt, welche zusätzlich durch verschiedene Fehlerkorrekturalgo-

<sup>8</sup><http://wiki.rc-heli-fan.org/index.php/Datei:DX7.png>

<sup>9</sup><http://wiki.rc-heli-fan.org/index.php/Datei:FASST.png>

rithmen unterstützt werden. Da die 2.4GHz-RC-Übertragungstechnologie von Hersteller zu Hersteller unterschiedlich umgesetzt wird, ist kaum eine detaillierte Beschreibung des tatsächlich angewandten Systems zu finden.

Die Steuerinformation, ein Pulse Code Modulation (PCM)-Frame, vom Sender wird mittels eines Encoders an den FM-Modulator weitergeleitet. Von Seiten des Empfängers wird das Eingangssignal mittels Decoder wiederum in ein PPM-Signal aufgesplittet und an den einzelnen Steuerkanälen ausgegeben (Abbildung: 3.10). Dieses PPM-Signal wird von den Aktoren (Servo, Motorsteller) verwendet und in eine entsprechende Dreh- oder Linearbewegung umgewandelt. Wie schon bei der 2.4 GHz-RC-Übertragungstechnologie, sind auch hier die Kodierungs- bzw. Dekodierungsalgorithmen zwischen den Herstellern unterschiedlich und ebenso ein Betriebsgeheimnis. Die PCM beruht auf der Grundlage, dass ein analoges Eingangssignal mit einem Analog/Digital (A/D)-Wandler in eine entsprechende, digitale Bitfolge umgewandelt wird. Diese Bitfolge wird anschließend auf das Trägersignal aufmoduliert, am Empfänger demoduliert und in ein PPM-Signal umgewandelt. Das PPM-Signal kann anschließend von den Aktoren verwendet werden. Nähere Informationen siehe:

- rc-heli-fan wiki (2012)
- Jörg Schamuhn (2008, FMT S.140ff)
- Manfred Dieter Kotting (2008, FMT S.140ff)

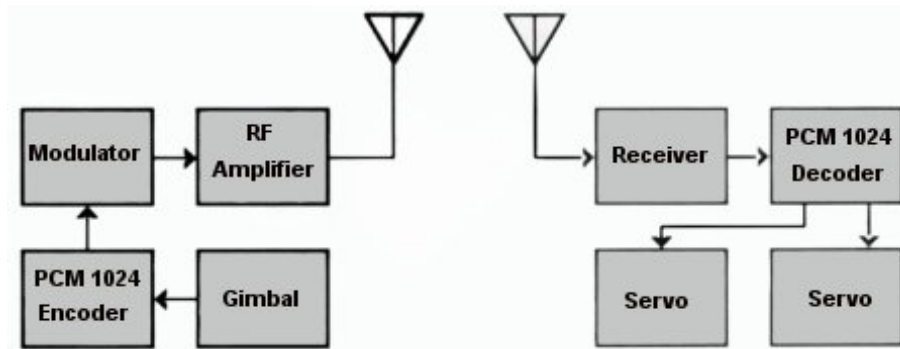


Abbildung 3.10: 2.4GHz PCM-Signalweg<sup>10</sup>

<sup>10</sup><http://www.futaba-rc.com/faq/faq-pcm1024.html>



### Serielle Datenübertragung

Der Begriff „serielle Datenübertragung“ wird im Allgemeinen mit einer RS-232 (Electronic Industries Alliance (EIA)-232)- Schnittstelle identifiziert. Eine serielle Schnittstelle ist jedoch nicht auf einen speziellen Schnittstellenstandard wie den vorhin genannten festgelegt, sondern definiert lediglich das Verfahren, wie Daten übertragen werden. Seriell bedeutet in der Kommunikationstechnik, dass eine Binärfolge, bitweise über ein und dieselbe Leitung oder einen Kanal übertragen wird. Im Gegensatz dazu werden bei einer parallelen Schnittstelle vier, acht oder mehr Bits, gleichzeitig über separate Datenleitungen übertragen. In der Datenkommunikation ist ein Großteil der gängigen Schnittstellen seriell. Die bekanntesten Vertreter (laut Wikipedia) sind:

- RS-232, 485, TTL
- USB
- div. Bildübertragungssysteme (High Definition Multimedia Interface (HDMI), Digital Visual Interface (DVI))
- Bussystem (LAN, WLAN, Bluetooth)
- uvm.

Die einzelnen Übertragungssysteme unterscheiden sich in erster Linie durch unterschiedliche Übertragungsgeschwindigkeiten, sowie in der verwendeten Kodierung der Daten. Im weiteren Verlauf wird sowohl auf den, in dieser Arbeit verwendeten RS-232-Standard mit den unterschiedlichen Logikpegeln, als auch auf den WLAN-Standard, eingegangen.

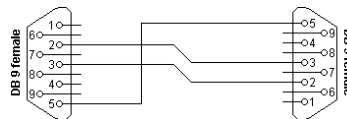
### RS232, TTL, LVTTL

Dieser Abschnitt behandelt die Grundlagen der klassischen „seriellen Schnittstelle“ (RS-232), sowie die beiden gängigsten Abwandlungen (TTL, LVTTL). Die beiden letzteren Schnittstellen unterscheiden sich von der RS-232, durch andere Signalpegel, sowie einer positiven Informationslogik. Da diese Schnittstellen in erster Linie zur Kommunikation mit elektronischen Baugruppen verwendet werden, wurde von Seiten der Industrie versucht, die Signalpegel an die Spannungspegel der gängigsten Mikroprozessoren anzupassen, um diese direkt miteinander verbinden zu können. Es ist dabei zu beachten, dass eine solche elektronische Baugruppe **nicht** direkt mit einer eventuell vorhandenen Computer RS-232-Schnittstelle verbunden werden darf, da eine solche Verbindung das Gerät aufgrund der höheren Spannung zerstören könnte. Um eine TTL-Schnittstelle mit einem Computer direkt zu verbinden, benötigt man einen zusätzlichen Pegelwandler. Dieser stimmt die beiden Signalpegel aufeinander ab und invertiert die Informationslogik. In Tabelle (3.1) sieht man die verschiedenen Signalpegel. Der Unterschied von TTL und Low-Voltage-Transistor-Transistor-Logik (LVTTL) wird noch später genauer beschrieben.

Tabelle 3.1: Signalpegel in Volt

Technologie	Empfangen		Senden	
	VLow	VHigh	VLow	VHigh
EIA-232	$-15 \geq -3$	$+3 \leq +15$	$-15 \geq -5$	$+5 \leq +15$
TTL 5 V	$0 \geq +0,8$	$+2 \leq +5$	$0 \geq +0,4$	$+2,4 \leq +5$
LVTTL 3,3 V	$0 \geq +0,8$	$+2 \leq +5$	$0 \geq +0,4$	$+2,4 \leq +5$

Die zuvor genannten Schnittstellen können bereits mit einer Drei-Leiter-Verkabelung genutzt werden, was der Minimalkonfektionierung einer seriellen Datenübertragung entspricht. Man benötigt für eine einfache Datenübertragung zwischen zwei Geräten eine Chassis Ground (GnD)-, eine Received Data (RxD)-, sowie eine Transmitted Data (TxD)-Leitung. Sollen beispielsweise zwei Computer miteinander verbunden werden, zeigt Abbildung 3.11 eine derartige Datenverbindungsleitung. Unterschiedlichste Darstellungen von Kabelkonfektionierungen sind unter [http://www.lammertbies.nl/comm/cable/de\\_RS-232.html](http://www.lammertbies.nl/comm/cable/de_RS-232.html) zu finden.

Abbildung 3.11: RS-232 Nullmodemverbindung<sup>11</sup>

Eine TTL- oder LVTTL-Schnittstelle besitzt im Regelfall einen zusätzlichen Leiter zur Spannungsversorgung des angeschlossenen Gerätes. Im Allgemeinen finden drei verschiedene, stabilisierte Spannungsversorgungen Verwendung. Die TTL-Schnittstelle liefert eine stabilisierte 5 V-Spannung, die LVTTL-Schnittstelle eine 3.3V-Spannung und die Low-Low-Voltage-Transistor-Transistor-Logik (LLVTTL)-Schnittstelle eine 1.8 V-Spannung, wobei die LLVTTL-Schnittstelle kaum Verwendung findet. Ein weiterer Unterschied zur RS232-Schnittstelle stellt der Logikpegel zur Informationsübertragung dar. Wie bereits erwähnt, verwendet die RS232-Schnittstelle einen negativen Logikpegel. Das bedeutet, dass Signalpegelspannungen von  $-15V \geq -3V$  als logisch „1“ und  $+3V \leq +15V$  als logisch „0“ definiert sind (siehe Tabelle: 3.1). Der Zwischenbereich von  $-3V \geq +3V$  ist nicht definiert (Abbildung: 3.12.a). Bei der TTL-Schnittstelle wird ein positiver Signalpegel verwendet (Abbildung: 3.12.b), was dem invertierten Signal entspricht.

<sup>11</sup>Lammert Bies (2010)



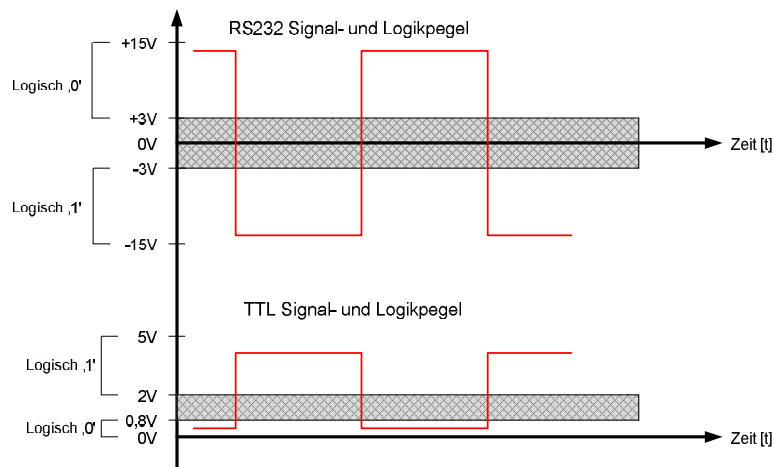
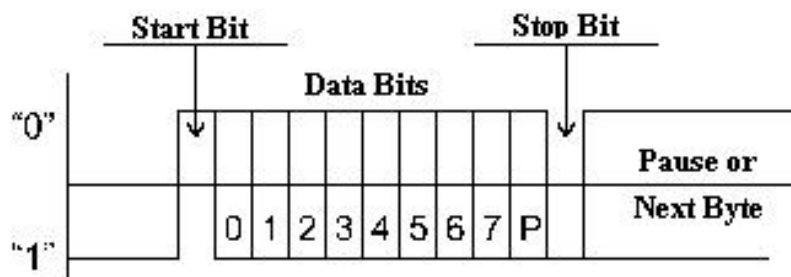


Abbildung 3.12: Signal und Logikpegel: a) RS-232 vs., b) TTL, LVTTTL

### Übertragungsprotokoll

Wie bereits in der Einleitung in diesem Kapitel erwähnt wurde, werden bei einer seriellen Datenübertragung die Informationen bitweise übermittelt. Die RS-232- und TTL-Schnittstelle verwenden dabei dasselbe Übertragungsprotokoll, daher sind in weiterer Folge immer beide Schnittstellen gleichbedeutend angesprochen. Die Datenrate wird hierbei in Baud oder Bit/sec angegeben. Diese Datenrate entspricht nicht der Informationsrate, da bei diesem Protokoll zusätzliche Informationen bzw. Steuersymbole benötigt werden. Die gängigsten Datenraten sind 4800, 9600, 19200, 38400, 57600 und 115200. Das Datenpaket besteht dabei immer aus einem Startbit, 5-8 Datenbits (Least Significant Bit (LSB) → Most Significant Bit (MSB))(üblicherweise 7-8), einem möglichen Paritätsbit und 1-2 Stoppbits (Abbildung: 3.13).

Abbildung 3.13: Serielles Datenpaket<sup>12</sup>

Ein solches Datenpaket nennt man auch ein „Wort“. Für die Darstellung einer solchen Information gibt es verschiedene Möglichkeiten, wobei die binäre Darstellung der tatsächlichen physikalischen Übertragung entspricht. Üblicherweise werden je nach Anwendungsbereich Hexadezimal, Dezimal oder Extended American Standard Code for Information Interchange (ASCII) verwendet. Hierbei handelt es sich um die Darstellungsmethoden, welche die Lesbarkeit erhöhen.

<sup>12</sup><http://www.scienceprog.com/controlling-external-devices-using-com-port-communications-programmed-using-vb-language/>

- Binär (0,1)
- Hexadezimal (0x00 → 0xFF)
- Dezimal (0 → 127 (255))
- ASCII (NUL → DEL (Space))
- Extended Ascii

Bei dieser Art der Datenübertragung müssen Sender und Empfänger dieselben Einstellungen im Bezug auf die Datenrate und der Paketlänge aufweisen, da es ansonsten zu Datenverlusten kommt. Bei der Verwendung von externen Geräten ist es zusätzlich nötig, den Aufbau der Information zu kennen, da es andernfalls nicht möglich ist, diese zu interpretieren oder decodieren. Die in dieser Arbeit verwendeten Protokolle (3.2.2), können auch als Beispiele für andere Umsetzungen und Interpretationen, der eigentlichen Datenbits, herangezogen werden.

Einen weiteren Aspekt bei der Datenübertragung, stellt die Übertragungssicherheit, oder die Möglichkeit fehlerhafte Bits festzustellen, dar. In den meisten Fällen werden bei dieser Art der Datenübertragung keine aufwändigen schnittstellenspezifischen Fehlerüberprüfungen durchgeführt. Dabei wird bei der Verbindungseinrichtung an beiden Endgeräten die Parität auf „Even“ (gerade) oder „Odd“ (ungerade) eingestellt. Es ist zu beachten, dass viele Endgeräte keine Paritätsüberprüfung implementiert haben. Daher ist in jedem Fall die Schnittstellenspezifikation genau zu beachten. Das Prinzip der Paritätsprüfung ist so einfach wie auch effektiv, jedoch hat diese auch eklatante Schwachstellen. Es wird von Seiten des Senders die Anzahl der Einsen im Datenpaket gezählt und im Falle einer geraden Anzahl, sowie mit „Even“-Parität, das Paritätsbit auf Null gesetzt. Sollte das nächste Datenpaket nun eine ungerade Anzahl an Einsen haben, wird das Paritätsbit auf Eins gesetzt. Da dem Empfänger bekannt ist, ob es sich um eine „Even“- oder „Odd“- Parität handelt, kann das Datenpaket leicht auf ein fehlerhaftes Bit hin geprüft werden. Die Zustandsänderung eines Bits entlang des Signalweges von Eins auf Null oder umgekehrt nennt man „flippen“. Durch zu lange oder falsch verlegte Datenleitungen kann es zu einem solchen Flipp eines Bits kommen (siehe Friedrich Wittgruber, 2006). Die Paritätsprüfung jedoch versagt, wenn zwei Bits innerhalb eines Datenpaketes flippen, weil die Parität erhalten bleibt. Wird durch die Paritätsprüfung ein fehlerhaftes Datenpaket erkannt, ist es nicht möglich, das fehlerhafte Bit ausfindig zu machen oder zu reparieren. Die einzige Möglichkeit besteht darin, dass fehlerhafte Paket erneut anzufordern.

An dieser Stelle werden u.a. Friedrich Wittgruber (2006) und Hans-Peter Messmer & Klaus Dembowski (2003) als weiterführende Literatur genannt, da diese einen detaillierteren Einblick in diverse Datenübertragungsverfahren erlauben.

### Übertragungspfade

Übertragungspfade oder auch das Übertragungsmedium definiert die Art der physikalischen Geräteverbindung. Die physikalischen Verbindungen können über eine feste Verkabelung oder eine drahtlose Anbindung realisiert werden. Dabei wird die technische Umsetzung durch die zu überbrückende Distanz definiert. Diese Übertragungspfade werden in drei große Gruppen unterteilt:

- Übertragung über große und sehr große Distanzen (bsp. Global Area Network (GAN))
- Übertragung über kurze bis mittlere Distanzen (bsp. Local Area Network (LAN), Wide Area Network (WAN))
- Drahtlose Übertragungen (bsp. Wireless Local Area Network (WLAN))

Die erste Gruppe kann als nationale, internationale oder globale Datenübertragung interpretiert werden und wird nach heutigem Stand der Technik meist durch Glasfaserverbindungen umgesetzt. Die zweite, für den Nahbereich ausgelegte Gruppe, wird häufig durch Kupferdatenleitungen, seltener durch Glasfaserverbindungen, realisiert. Da in dieser Arbeit keine Datenübertragungen mit Glasfaserverbindungen benötigt wurden, wird im weiteren Verlauf dieser Arbeit auch nicht näher darauf eingegangen.

**Datenleitungen** für die Kommunikation besitzen einen speziellen Aufbau um die heutigen Datendurchsatzraten ermöglichen zu können. Datenleitungen besitzen je nach Ausführung eine oder mehrere Abschirmungen, welche Beeinflussungen durch Einkoppeleffekte verringern sollen. Diese Schirmung wird oft auch als Signal-GnD mitverwendet. Die Limitierungen für Datenleitungen ohne Zusatzequipment (Repeater) liegen in der Leitungslänge sowie in der Art der Verlegung. In der Literatur wird bei Standard-Datenleitungen (Patchleitungen) eine maximale Länge von 5-10 m angegeben (siehe Klaus Dembowski, 2007). Diese Längen sind in erster Linie durch den Innenwiderstand und den damit verbundenen Spannungsabfall in der Leitung begründet. Bei längeren Distanzen ist die Wahrscheinlichkeit höher, dass der Signalpegel einen undefinierten Wert erreicht. Diese Einflüsse kommen in erster Linie bei Datenübertragungsverfahren zu tragen, welche auf Spannungspegel beruhen. Verfahren, die auf den Strompegel zurückgreifen, sind davon weniger beeinflusst. Wie zuvor schon erwähnt ist die Verlegung solcher Datenleitungen ein weiteres Kriterium für die Längeneinschränkung. Wird eine Datenleitung beispielsweise in einem Bündel mit Energieleitungen verlegt, kann es durchaus vorkommen, dass durch Einkoppeleffekte der Signalpegel beeinflusst wird, und somit ein Informationsbit „flippert“. Detailliertere Informationen finden sich in den entsprechenden **ÖVE-ÖNORMen**.

Die dritte und mittlerweile im Endnutzerebereich meistverbreitetste Datenübertragungsmethode, stellt die **WLAN**-Verbindung dar. Diese Art der Verbindung, unter Verwendung des entsprechenden Equipments, eignet sich mittlerweile auch für Datenübertragungen mit Distanzen von

mehreren Kilometern. Um solche Distanzen im gesetzlichen Rahmen überbrücken zu können, sind vor allem hohe Ansprüche an die Sende-Empfangsantennen zu stellen. Auch können entsprechende Repeater eingesetzt werden, was jedoch für die vorliegende Arbeit nicht von Bedeutung ist. Wie in der Einleitung dieses Kapitels schon angeführt, werden Limitierungen der Reichweite durch eine gesetzliche Regelung der maximalen Abstrahlleistung von 100 mW (EIRP) (siehe [http://www.wavesurf.at/wlan/index.php?option=com\\_content&task=view&id=16&Itemid=57](http://www.wavesurf.at/wlan/index.php?option=com_content&task=view&id=16&Itemid=57)) festgelegt. Einen weiteren Limitierungsfaktor stellt das Antennendesign dar. Handelsübliche WLAN-Router verwenden eine  $\lambda/4$ -Rundstrahlantenne, welche in alle Richtungen dieselbe Energie abstrahlt und bei freier Sicht eine stabile Datenübertragung von ca. 100-300 m erlaubt. Da diese Reichweiten für viele Anwendungen nicht ausreichend sind und auch die Freisicht nicht immer gegeben ist, werden sogenannte Richtfunk- oder Patchantennen eingesetzt, welche ihre Abstrahlenergie gerichtet bündeln. Richtfunkantennen müssen immer auf die entsprechende Empfangsstation ausgerichtet werden, um eine gesicherte Datenübertragung zu gewährleisten. Die Energieverteilung solcher Antennen wird üblicherweise in Polarplots dargestellt (siehe Kapitel: 4).

### 3.2.2 Protokolle

Die in dieser Arbeit verwendeten Protokolle und Konzepte der Prozessierung werden in diesem Unterkapitel behandelt. Wie im vorherigen Abschnitt beschrieben, werden bei einer RS-232- bzw. TTL-Schnittstelle 7–8 Datenbits übertragen, die ein Datenwort ergeben. Werden mehrere Wörter aneinander gereiht, entsteht daraus ein Datenprotokoll oder Datenframe. Der Aufbau eines solchen Protokolls ist je nach Endgerät verschieden und wird den jeweiligen Anforderungen angepasst. Die wohl bekanntesten Vertreter eines solchen Protokolls sind die Protokollsammlungen aus dem Bereich der Netzwerktechnik (HTTP, FTP, TCP, ...). Hinter dieser Ansammlung an Protokollen steht das ISO/OSI 7-Schichtenmodell, wie es für den Datenverkehr im Internet verwendet wird (siehe Mark A. Dye *et al.*, 2008). Da diese Protokolle diese Arbeit nur am Rande betreffen, wird hier auch nicht speziell darauf eingegangen. Die hier verwendeten, deutlich einfacher aufgebauten Datenprotokolle, beschränken sich auf die Bitübertragungsschicht sowie dem Übergeordneten spezifischen Protokoll.

Die Definition der Kommunikation ist ein wesentlicher Parameter der Protokollimplementierung. Prinzipiell wird zwischen drei Gruppen unterschieden:

- Unidirektional (Endgerät sendet Permanent)
- Bidirektional (Endgerät sendet nur auf Anfrage)
- Hybride Formen (Endgerät sendet allgemeine Daten permanent und spezifische Daten auf Anfrage)

**Unidirektional** bedeutet in diesem Zusammenhang, dass ein Endgerät permanent Daten sendet. Da es sich bei den Endgeräten fast ausschließlich um Mikroprozessoren mit nachgeschalteten Sensoren handelt, kann diese Art der Übertragung als eine Endlosschleife angesehen werden. Wurde einmal eine Verbindung zwischen dem Nutzgerät (Computer, übergeordneter Prozessor) aufgebaut, benötigen diese Geräte keine weiteren Befehle mehr, sondern senden ihre Daten nach einem festgelegten Schema. In diesem Zusammenhang kommt der spezifische Aufbau eines Protokolls besonders zu tragen. Ein häufig verwendetes Beispiel ist dabei das National Marine Electronics Association (NMEA)-Protokoll. Der Aufbau eines solchen Protokolls folgt immer demselben Schema. Dabei beginnt ein solches Protokoll (Frame) immer mit einem eindeutigen Startzeichen, gefolgt von einer Frame-Kennzeichnung. Diese Kennzeichnung gibt an, um welche Daten es sich handelt. Dies ist in erster Linie bei Geräten wichtig, die unterschiedlichen Datenframes verwenden, da die folgenden Daten identifiziert und entsprechend decodiert werden müssen. Anschließend folgen die eigentlichen Nutzdaten und meist eine Kontrollsumme über einen im Protokoll definierten Bereich. Der zuvor genannte Aufbau ist nur eine Beispielimplementierung. Meist verwendet jeder Sensor- oder Aktorhersteller ein eigenes Protokoll. Zwischen der Kennzeichnung und dem eigentlichen Datenpaket können auch weitere Informationen eingebettet werden, wie zum Beispiel die Anzahl der Datenbytes. Je mehr Information in ein solches Frame eingebettet wird, umso wahrscheinlicher fallen einzelne Pakete durch einen Fehler bei der Decodierung auf und werden verworfen. Im Allgemeinen wird versucht, möglichst kleine Datenframes mit unterschiedlichen Kennzeichnungen zu verwenden. Abbildung 3.14 zeigt den Aufbau eines einfachen Datenpakets, das zum Beispiel von einem GPS-Empfänger gesendet wird.

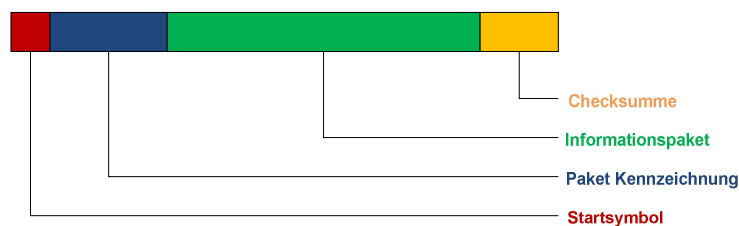


Abbildung 3.14: Unidirektionales Datenpaket

Fordert das Gerät die benötigten Informationen beim Endgerät an, wird der Begriff „**bidi-rektionale Kommunikation**“ verwendet. Das Empfängergerät weiß, welche Daten ankommen müssen. Daher wird oft auf ein eindeutiges Startsymbol, eine Frame-Kennzeichnung und eine Checksumme verzichtet. Diese Art der Kommunikation ist meist zielgerichteter als eine unidirektionale Kommunikation und daher einfacher zu realisieren. Das Konzept ist insofern simpel, da der Datenempfänger ein Zeichen oder eine Zeichenfolge an das Endgerät schickt und auf

die entsprechende Antwort wartet. Daten werden nur angefordert, wenn sie benötigt werden. Dadurch entsteht kaum ein zusätzlicher Datenüberfluss.

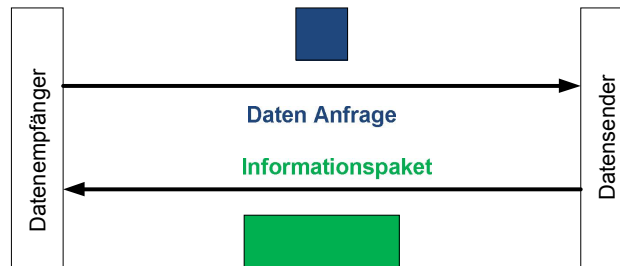


Abbildung 3.15: Bidirektionales Datenpaket

**Hybride Formen** verwenden hingegen beide Konzepte. Allgemeine Daten oder Informationen werden dabei kontinuierlich an den Nutzer gesendet und spezielle Daten, wie beispielsweise Einstellungen eines Endgerätes oder auch spezielle Datensätze, werden nur auf Anfrage übermittelt. Ähnlich wie bei der bidirektionalen Kommunikation, gibt es auch hier kaum Informationsüberladungen. Als Beispiel kann hier ein einfacher, neigungskompensierter Kompass angeführt werden. Dem Nutzer ist in erster Linie die horizontale Ausrichtung des Kompasses wichtig. Es ist also nicht nötig, diese Information kontinuierlich über eine bidirektionale Verbindung anzufordern. Diese Daten werden dem Nutzer mit einem unidirektionalen Protokoll zur Verfügung gestellt. Ist der Nutzer an den Neigungsdaten des Kompasses interessiert, wird eine definierte Zeichenfolge an das Gerät gesendet. Der elektronische Kompass verarbeitet die Neigungsdaten in ein unidirektionales Protokoll mit einer eigenen Kennzeichnung und sendet das Paket zwischen den kontinuierlichen Daten an den Nutzer. Dass der Nutzer und die Datenleitung mit unnötigen Informationen überhäuft werden, wird durch diesen Vorgang vermieden.

Die zuvor genannten Kommunikationsprotokolle und Verfahren, beziehen sich in dieser Darstellung auf eine reine Punkt-zu-Punkt Verbindung. Das bedeutet, dass eine Schnittstelle genau ein Endgerät bedient. Sollen mehrere Geräte mit einer Schnittstelle verwendet werden, spricht man von einem Netzwerk oder Bussystem. Die Kommunikation zwischen den Geräten beschränkt sich wiederum auf die drei oben genannten Verfahren, jedoch kommen einige Erweiterungen hinsichtlich Kollisionsverwaltung, eindeutige Adressierung und dergleichen hinzu. So erhält jedes Gerät eine eindeutig identifizierbare Kennzeichnung, die zusätzlich in das Protokoll eingebunden wird. Des Weiteren müssen Vorkehrungen getroffen werden, welches Gerät wann, was, wohin sendet und welche Informationen für welches Gerät bestimmt sind. Ein geeignetes Nachschlagewerk für die komplexe Thematik der Netzwerk- und Bustechnologie bietet das Buch Klaus Dembowski (2007).

### 3.3 Antennenausrichtung

Da in vielen Bereichen die Reichweite bei der drahtlosen Datenübertragung eine große Rolle spielt, wird in diesem Absatz die Verwendung von Richtfunkantennen und deren Ausrichtung beschrieben. Richtfunkantennen haben einen auf wenige Grad begrenzten Wirkungsbereich und müssen aus diesem Grund auf die entsprechende Gegenstation ausgerichtet werden. Um diese Antennen entsprechend ausrichten zu können, werden der Dreh- und Neigungswinkel zur Gegenstation benötigt. Handelt es sich bei der Gegenstation um ein bewegtes Objekt, ist es nötig, eine automatische Nachführung zu verwenden. Zur Bestimmung der Dreh- bzw. Neigungswinkel einer solchen automatischen Nachführung stehen zwei Konzepte zur Verfügung. Diese beiden Konzepte verfolgen völlig unterschiedliche Lösungsansätze.

#### Signalpegelmessung

Beim Ansatz der Signalpegelmessung werden mehrere Empfänger und Antennen für ein einziges Signal benötigt. Der Empfänger mit dem stärksten Empfangssignal (Received Signal Strength Indication (RSSI)) dient als die Hauptempfangsstation. Wird je nach Signalstärke zwischen mehreren Empfängern umgeschaltet, wird der Begriff „Diversity“ verwendet. Werden drei oder mehr Richtfunkantennen mit entsprechender Ausrichtung verwendet, kann daraus ein Winkel abgeleitet werden. Abbildung 3.16 rechts oben) zeigt eine Antennenanordnung bei der zwischen drei horizontalen Sektoren unterschieden wird. Bewegt sich das Zielobjekt in den Empfangsbereich eines anderen Sektors, übernimmt automatisch der jeweilige Empfänger. Da der gerade verwendete Empfänger bekannt ist, kann das Antennenarray so weit nachgeführt werden, dass sich die Gegenstation wieder im Bereich des mittleren Empfängers befindet.

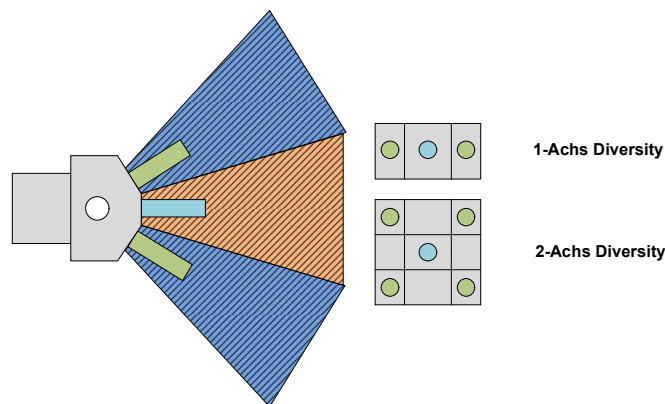


Abbildung 3.16: Diversity Antennenanordnung

Soll eine Zweiachsnachführung umgesetzt werden, benötigt man mindestens fünf Antennen für die zusätzliche Bestimmung der Vertikalkomponente. Der Vorteil dieses Systems ist die Ersparnis aufwändiger Positionsrechnungen. Der Nachteil jedoch ist die große Bauform der Antennennachführung sowie die Verwendung von mehreren Empfängern. Auch sind solche Systeme nicht käuflich.

### Positionierungsverfahren

Der gebräuchlichere Lösungsansatz verwendet, zur Bestimmung der Winkel, ein Positionierungsverfahren. Durch die Startposition des Rovers werden die Trackerposition und dessen Ausrichtung festgelegt. Die Roverposition muss anschließend kontinuierlich an die Nachführungselektronik übermittelt werden. Diese Nachführungselektronik berechnet die Horizontal- und Vertikalwinkel (Abbildungen: 3.17 a und 3.17 b) und steuert anschließend die entsprechenden Antriebe der Dreh- und Neigungsachse. Wie schon beim Ansatz der Signalpegelmessung ist auch hier eine Antennennachführung bei einem Signalverlust nicht mehr möglich. Meist wird dieses Konzept der Nachführung in Verbindung mit einem FPV-System genutzt, da im Rover das zusätzliche Equipment bereits vorhanden ist.

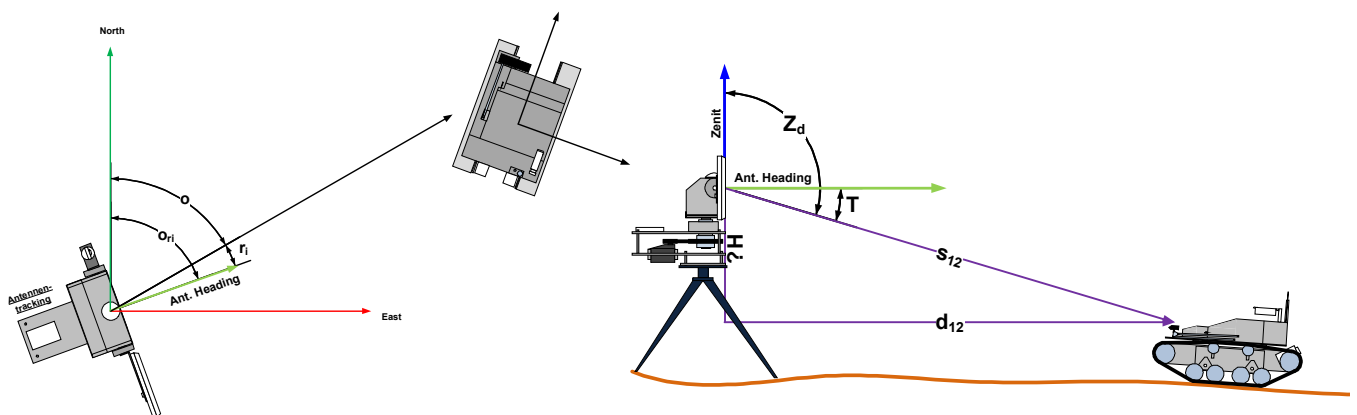


Abbildung 3.17: a) Drehachsenkomponente, b) Neigungsachsenkomponente

Als Telemetriedaten kommen hier Global Navigation Satellite System (GNSS)-Positionslösungen zum Einsatz, die mit einer On Screen Display (OSD)-Elektronik auf die Tonspur aufmoduliert werden ([http://fpv-community.de/wiki/index.php?title=Einstieg\\_f%C3%BCr\\_FPV-Neulinge](http://fpv-community.de/wiki/index.php?title=Einstieg_f%C3%BCr_FPV-Neulinge)). Diese Telemetriedaten benötigen eine spezielle Auswerteelektronik um sie für die Weiterverarbeitung nutzbar zu machen (Abbildung: 3.18).



Abbildung 3.18: Telemetrie Auswerteeinheit AV-Signalübertragung<sup>13</sup>

<sup>13</sup><http://www.globe-flight.de/EzAntennenTracking-Controller>



## 3.4 Antriebe

Neben der Datenübertragung stellt auch die Auswahl der Antriebskomponenten des Rovers, sowie der Antennennachführung, eine nicht zu unterschätzende Herausforderung dar. In diesem Kapitel werden nicht nur unterschiedliche Antriebssysteme genannt, sondern auch auf deren Vor- und Nachteile hingewiesen. Die eigentliche Auswahl beruht auf Erfahrungswerte, Versuche und der Verfügbarkeit. Zur schlussendlichen Auswahl der Antriebe, flossen noch zusätzliche Kriterien wie Betriebsstrom, Betriebsspannung, Haltemomente und die Verfügbarkeit entsprechender Motorregler ein.

### 3.4.1 Positionierungsantriebe

Für Nachführungs- bzw. Stellaufgaben sind im Allgemeinen Schrittmotoren mit einer entsprechenden Steuerkarte im Einsatz. Diese Antriebstechnologie erlaubt ein sehr genaues Positionieren, hat aber den großen Nachteil, dass sie sehr teuer und aufwändig anzusteuern ist. Eine Alternative stellen Servomotoren dar. Diese Antriebe sind in verschiedensten Größenordnungen erhältlich und mit einer einfachen Elektronik leicht anzusteuern. Sie bieten für diverse Verwendungszwecke eine ausreichende Positionierungsgenauigkeit. Speziell die Modellbauservos stellen in dieser Kategorie eine günstige Alternative dar. Es ist jedoch von Nachteil, dass Modellbauservoantriebe einen begrenzten Wirkungsbereich (üblicherweise zwischen  $-45^\circ$  ( $-60^\circ$ ) und  $45^\circ$  ( $60^\circ$ )) aufweisen und sind somit nicht für Dauerdrehungen konzipiert. Jedoch stellt der begrenzte Wirkungsbereich dieser Antriebe kein Problem dar, insofern bei der Antennennachführung auf die Dauerdrehung aufgrund der Leitungsverlegung verzichtet wird.

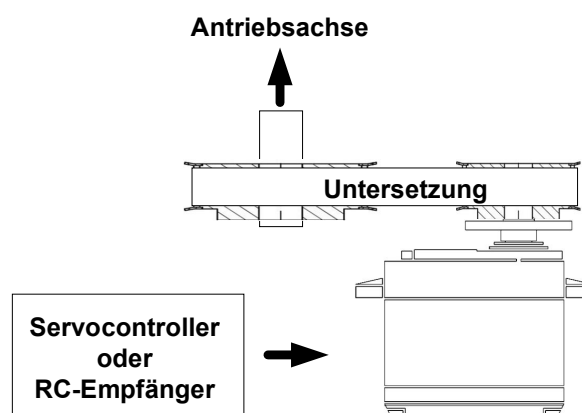


Abbildung 3.19: Antriebskomponenten Antennennachführung

Die Ansteuerung der Dreh- und Neigungsachsen kann mit Servoantrieben aus dem Modellbau realisiert werden (Abbildung: 3.19). Diese Servoantriebe benötigen zur Positionierung, die in 3.2.1 erwähnten PWM-Signale, welche durch einen Servocontroller generiert werden. Servopositionen sind normalerweise durch Pulsweiten definiert, wobei eine Pulsweite von  $1500\mu\text{sec}$

die Neutralposition darstellt. Die Auflösung  $\frac{[^\circ]}{[\mu\text{sec}]}$  ist meist durch den maximalen Ausschlag, sowie dessen Pulsweite definiert. Bei vielen Servos werden diese Daten jedoch nicht zur Verfügung gestellt. Dadurch muss die eigentliche Servoaufösung durch Versuche herausgefunden werden. Dies wird dahingehend begründet, dass im Modellbau folgende Parameter einen höheren Stellenwert besitzen:

- Maximaler Endausschlag in Grad
- Laufzeit bis Endausschlag
- Stell-bzw. Haltemoment

Die tatsächliche Auflösung der Servopositionen hat dabei eine untergeordnete Rolle und ist nicht nur durch den Servo festgelegt, sondern ebenfalls durch den Servocontroller und dessen Ausgangsaufösung. Die Näherungsweise berechneten Auflösungen werden für die verwendeten Servoantriebe in Kapitel 4 angeführt. Die tatsächlichen Werte weichen jedoch durch Ungenauigkeiten bzw. Spiel in den Untersetzungen ab und müssen durch Tests festgelegt werden.

### 3.4.2 Rover Antriebssystem

Der Antrieb eines Rovers oder Roboters umfasst mehrere Komponenten und ist von unterschiedlichsten Faktoren abhängig. Das Betreiben solcher Fahrzeuge erfolgt in erster Linie mittels Gleichspannung, da diese in Akkus gespeichert werden kann. Die benötigten Antriebskomponenten mit dessen Schema eines Schneckengetriebemotor und einer Untersetzung, werden in Abbildung 3.20 gezeigt.

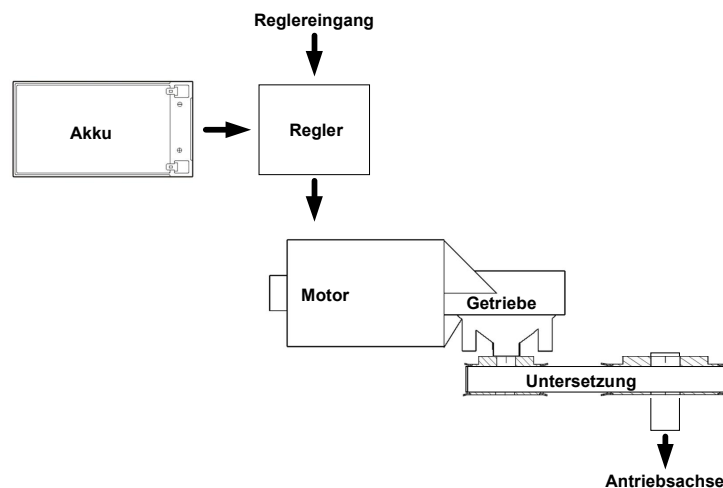


Abbildung 3.20: Antriebskomponenten Rover

Da ein solches Getriebe eine Selbsthemmung besitzt, ist keine zusätzliche Bremsvorrichtung von Nöten. Folgende Kriterien sind bei der Auswahl des richtigen Antriebs eines Rovers wichtig:

- Gewicht des Rovers
- Angetriebe Achsen
- Durchmesser der Antriebsräder
- Art der Fortbewegung (Räder, Raupen)
- Endgeschwindigkeit
- Steigfähigkeit
- Spannungsversorgung
- Leistungsreserven

Gleichstrommotoren besitzen ein hohes Anlaufmoment und zeichnen sich durch einfache Regelbarkeit aus. Wichtig für die Auswahl eines solchen Motors sind die Betriebsspannung, sowie die Verfügbarkeit entsprechender Motorsteller. Übliche Betriebsspannungen sind 12 V, 24 V, 48 V und 72 V. Für die Robotik sind vor allem die 12 V- und 24 V-Antriebe vorzuziehen, da es ausreichend viel Industrieware gibt und entsprechende Motorsteller kostengünstig erhältlich sind. Der 24 V-Antrieb ist in erster Linie bei größeren Leistungsklassen empfehlenswert, da sich nach  $P = U * I$  mit doppelter Spannung der Motorstrom halbiert. Das macht sich vor allem in der Anschaffung des Motorstellers bemerkbar. Ein weiterer Punkt sind die Leistungsreserven. Bei jeder Antriebsart ist darauf zu achten, dass der Motor nicht dauerhaft im Endbereich seiner Leistung betrieben wird.

Ein weiterer Aspekt der Antriebskomponenten stellt das Getriebe dar. Da Gleichstrommotoren üblicherweise eine hohe Drehzahl aufweisen, jedoch wenig Drehmoment liefern, ist es in den meisten Fällen nötig, die Drehzahl auf ein verwendbares Maß zu bringen. Aufgrund der Vielfalt der Möglichkeiten, wird an dieser Stelle lediglich auf zwei Überbegriffe verwiesen. Die erste Möglichkeit stellt einen Direktantrieb dar. Dabei wird die Motorleistung mittels einer Zahnrad- oder Zahnriemenuntersetzung auf die Antriebswelle übertragen. Bei großen Rovern ist darauf zu achten, dass dieser Antrieb keine Selbsthemmung besitzt. Das wiederum bedeutet, dass eine entsprechende Bremsvorrichtung einkalkuliert werden muss. Als zweite Möglichkeit bietet sich in diesem Zusammenhang ein Motor mit Schneckengetriebe an, die eine solche „integrierte“ Bremswirkung besitzt. Zusätzlich ist es oft nötig, eine Unter- oder Übersetzung (Formel: 3.1 und 3.2 nach Gustav Niemann & Hans Winter, 2003) einzuplanen, um die entsprechenden Drehzahlen und, in weiterer Folge, die gewünschte Geschwindigkeit des Rovers zu erreichen. Bei der Abschätzung des Antriebsmotors wird auf [http://www.rn-wissen.de/index.php/Motorkraft\\_berechnen](http://www.rn-wissen.de/index.php/Motorkraft_berechnen)

bzw. <http://www.robotshop.com/dc-motor-selection.html> verwiesen, welche ein entsprechendes Online-Tool zur Motorberechnung zur Verfügung stellt.

$$\begin{aligned} i_n &= \frac{n_a}{n_b} \\ i_T &= \frac{T_b}{T_a} \end{aligned} \quad (3.1)$$

$$\begin{aligned} i_{insLangsame} &= n_b < n_a \\ i_{insSchnelle} &= n_b > n_a \end{aligned} \quad (3.2)$$

Der Regler oder Motorsteller ist ein Bauteil, der einem Gleichstrommotor, entsprechend der Reglereinstellung, eine definierte Spannung zur Verfügung stellt. Bei der Auswahl eines Motorstellers müssen vor allem der Spannungsbereich, sowie die maximale Strombelastbarkeit auf den verwendeten Motor abgestimmt sein. Die einfachsten Motorregler stellen Schleifwiderstandsregler dar, welche die überschüssige Energie durch eine Widerstandsbrücke abbauen. Diese Regler weisen daher eine sehr hohe Verlustleistung auf. Besser eignen sich Regler, die auf einer Leistungselektronik beruhen. Diese weisen eine vergleichsmäßig geringe Verlustleistung auf und besitzen zudem eine genauere Regeleigenschaft.

Als Energiespeicher für größere Roboter und Rover sind vor allem geschlossene Blei-Gel-Akkus eine gute Alternative. Diese Akkus besitzen eine hohe Leistungsdichte, sind darüber hinaus leicht erhältlich und unproblematisch beim Ladevorgang. Die große Bauform sowie das hohe Eigengewicht stellen hier aber einen großen Nachteil dar. Handelsübliche Nickel-Cadmium (NiCd)- oder Lithium-Polymer (LiPo)-Akkus sind weitere Optionen um Energie zu speichern. Diese haben im Vergleich zur den Blei-Gel-Akkus eine noch höhere Leistungsdichte, sind jedoch um ein vielfaches teurer und erfordern einen weitaus höheren Aufwand beim Ladevorgang. Zudem sind LiPo-Akkus sehr empfindlich im Bezug auf Temperaturschwankungen sowie auf Unter- und Überladung. NiCd-Akkus sind im Vergleich zu LiPo-Akkus unempfindlich, jedoch bei annähernd gleicher Leistungsdichte um vieles kostspieliger. Zudem ist dieser anfällig auf den sogenannten Memory-Effekt.

# Kapitel 4

## Hardwareaufbau

### 4.1 Rover

Der Rover wird zur Datenerfassung benötigt und liefert zusätzlich zu den Rohdaten des Autopiloten auch Bildinformationen. Aufgrund der modularen Bauweise kann der Rover verschiedenste Sensoren aufnehmen und die Daten direkt an die Bodenstation senden. Im gesamten Projekt wurde versucht auf Spezialanfertigungen zu verzichten und stattdessen leicht erhältliche Industrieware zu verwenden, welche kostengünstig wiederzubeschaffen sind. Im folgenden Kapitel wird der Aufbau des Chassis, der Trägerplattform und der Elektrik bzw. Elektronik beschrieben.

#### 4.1.1 Chassis

Wie bereits im Kapitel 2.1.2 beschrieben, wurde bei der Planung und beim Bau des Chassis auf einen Plansatz von Christian Ulrich (<http://www.ulrichc.de/>) zurückgegriffen. Da diese Pläne für den nichtkommerziellen Einsatz frei zur Verfügung stehen (<http://www.ulrichc.de/main/copyright.de.htm>) musste das Chassis nicht vollständig neu konstruiert werden. Für den Nachbau stehen hierfür Konstruktionspläne, Bauteillisten und eine Kurzdokumentation zur Verfügung. Deshalb wird im weiteren Verlauf des Kapitels nicht auf Ulrichs Baupläne selbst eingegangen, sondern auf die vorgenommenen Änderungen und notwendigen Anpassungen des Chassis für dieses Projekt. Abbildung 4.1 zeigt das vollständige 3D-Modell des Chassis.

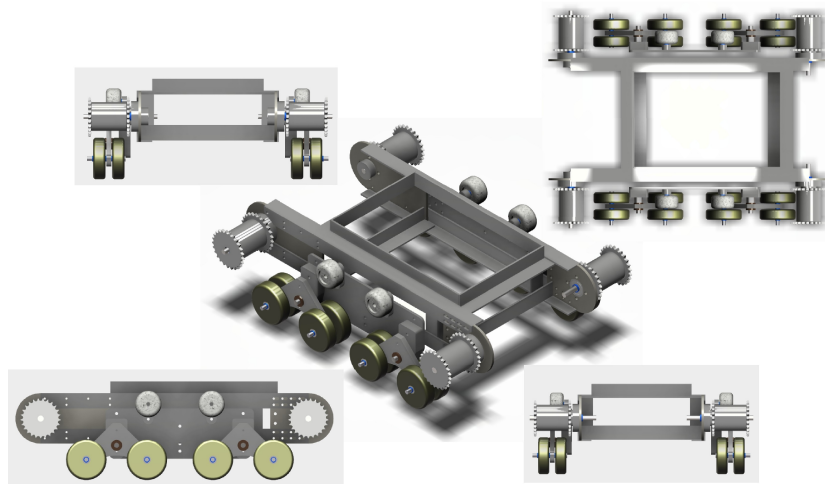


Abbildung 4.1: Modell des Roverchassis

### Laufkette

Um die Kettenräder nicht auszufräsen und die Kettenplatten nicht durch die Kettenglieder zu schrauben wurde ein abgeändertes Kettendesign mit Winkellaschen-Rollketten verwendet. Dadurch ist es möglich die Kettenräder ohne Nachbearbeitung zu montieren und die Kettenplatten auf die Winkellaschen der Kette zu schrauben (Abbildung: 4.2).

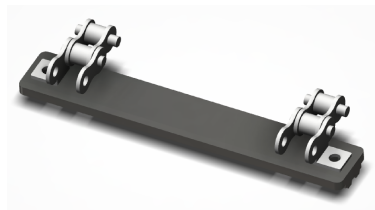


Abbildung 4.2: Modell einer Kettenplatte mit Rollenkettenglieder

Durch die Kettenräder aus Azetalharz wurden eine höhere Laufruhe und ein geringerer Geräuschpegel erreicht, da es keinen Stahl-zu-Stahl Übergang zwischen Kettenrad und Rollenkette gibt. Die Konstruktion hat sich nach den ersten Fahrversuchen bewährt und es sind keine sichtbaren Abnutzungserscheinungen bei höheren Belastungen aufgetreten. Die tatsächliche Lebensdauer der Azetalharzräder kann jedoch nur durch Langzeittests ermittelt werden.

Tabelle 4.1: Technische Daten je Laufkette und Kettenräder

Name	Material	Dimensionen	Zähne/Anzahl	Hersteller
Kettenrad	Azetalharz	3/8x7/32"	4x25Z	Fa. Mädler
Rollenkette	Stahl	3/8x7/32"	2x142 Glieder	Fa. Mädler
Kettenplatte	Aluminium	105x12x4 mm	71Stk.	
Verschraubungen	Stahl	M3x15	2x71Stk.	Fa. Ruku

In Tabelle 4.1 sind die Dimensionierungen der Kettenräder und der Laufkette dargestellt. Die Länge der Laufkette kann je nach Gehäuserahmenlänge und Kettenraddurchmesser variieren und muss dementsprechend angepasst werden.

### Antriebs- und Umlaufachse

Die Antriebs- und Umlaufachse sind Teile des Roverlaufwerks und dieses besteht aus vier unterschiedlichen Modulen:

- den Überlaufrollen
- den Laufrollen
- der Umlaufachse
- der Antriebsachse

Die Überlaufrollen und die Laufrollen und deren Aufhängungen wurden gemäß dem Bauplan von Ulrich gefertigt. Ebenso die Achslagerung der Antriebsachse mit dem Gehäuseflansch konnten direkt übernommen werden. Bei der Antriebsachse wurden der Kettenradaufnehmer und die Antriebswelle auf das neue, und breitere Kettendesign angepasst.

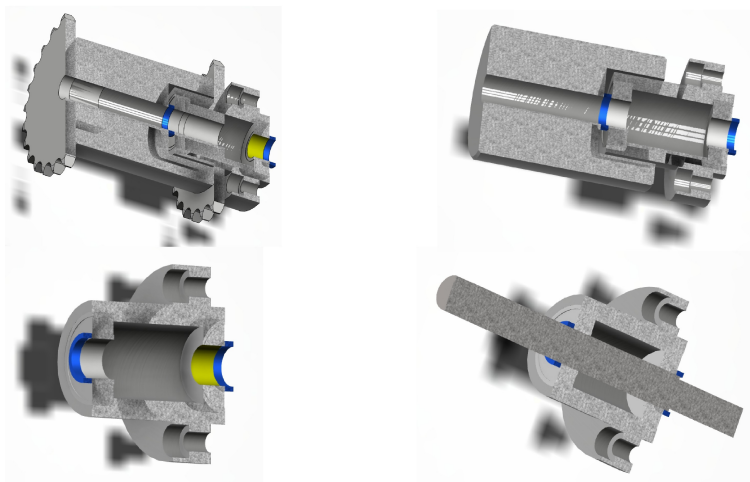


Abbildung 4.3: Schnittmodelle der hinteren Antriebsachse

Abbildung 4.3 zeigt das Schnittmodell der Antriebsachse ohne die Zahnriemenscheibe.

Im Gegensatz zur Antriebsachse ist die vordere Umlaufachse eine komplette Eigenkonstruktion. Die Achswelle wurde starr ausgeführt und ist somit direkt mit dem Gehäuseflansch verschraubt. Die Lagerung der Umlaufachse ist direkt in den Kettenradaufnehmer eingebaut (Abbildung: 4.4). Die vollständige vordere Umlaufachse kann leicht getauscht werden, indem lediglich nur eine Zentralmutter gelöst werden muss.

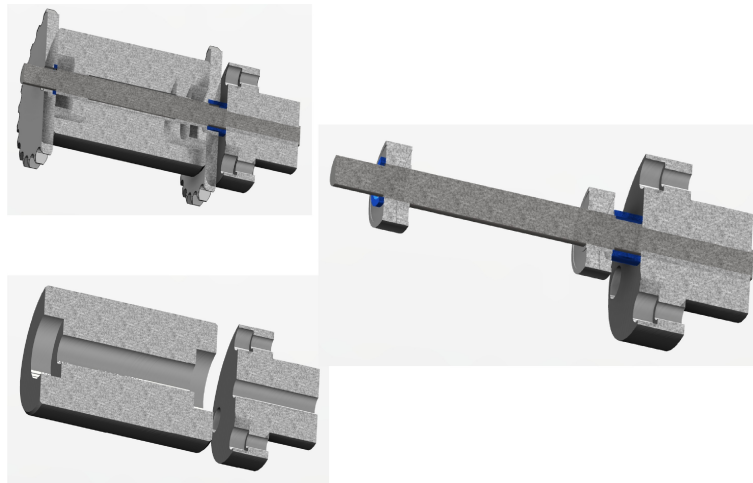


Abbildung 4.4: Schnittmodelle der vorderen Kettenachse

### Antrieb

Die beiden Antriebswellen werden mit Schneckengetriebemotoren und einer Zahnriemenunter-setzungen angetrieben. Als Antriebsmotor kommen zwei Industriemotoren der Marke „Nidec“ zum Einsatz. Während des Baus war die tatsächliche Belastung der Motoren noch nicht bekannt und deshalb wurden gezielt die stärkeren Motoren verwendet.

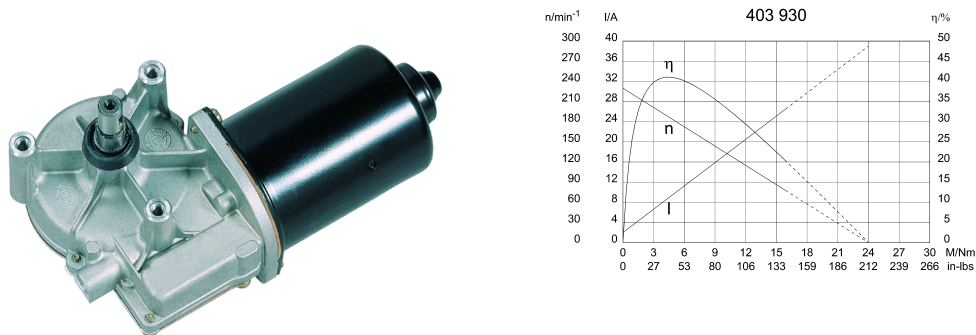


Abbildung 4.5: a) Antriebsmotor; b) Motorkennlinien

Für die Montage im Roverchassis wurde eine Montageplatte aus 4mm Aluminium und M6-Gewindestiften angefertigt (Abbildung: 4.6 a). Da es bei dieser Motorausführung keine Links-Rechtsausführung gibt, musste jeder Motor separat in das Roverchassis eingepasst werden (Abbildung: 4.7). Die Kraftübertragung von der Motorwelle zur Antriebswelle übernimmt ein 16mm breiter Zahnriemen.

Diese Zahnriemenuntersetzung eignet sich insofern sehr gut für die Kraftübertragung in einen Rover da die Zahnriemenscheiben gegebenenfalls leicht zu wechseln sind. Somit ist es mit geringem Aufwand möglich den Rover an verschiedene Geländeumgebungen anzupassen.

Die Zahnriemenscheiben wurden für eine Untersetzung von 1,45 ausgelegt, woraus sich für



die Motorwelle eine 32-Zahnscheibe und für die Antriebswelle eine 22-Zahnscheibe ergibt. Diese Zahnriemenscheiben wurden entsprechend den Wellenformen angepasst, montiert und durch Passstifte gesichert.



Abbildung 4.6: a) Antriebskomponenten; b) montierte Antriebseinheit

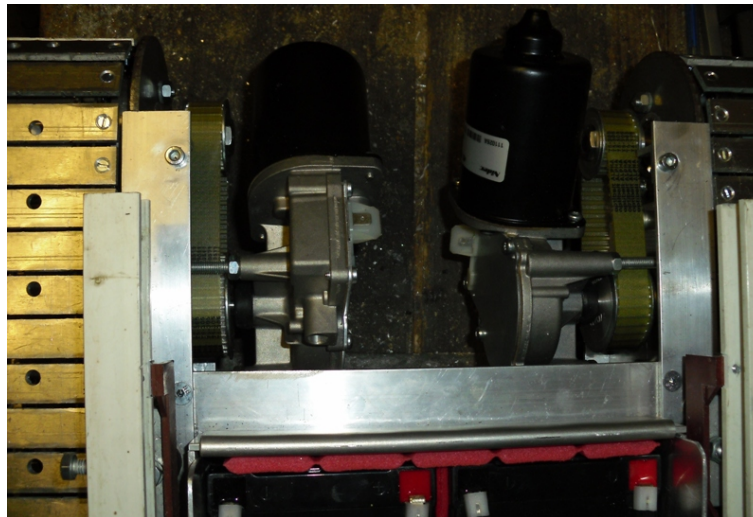


Abbildung 4.7: vollständiger Roverantrieb

### 4.1.2 Trägerplattform

Die Trägerplattform enthält die gesamte Elektronik zur Steuerung, Kommunikation und Sensordatenerfassung (APM). Bei der Konstruktion wurde darauf geachtet den niedrigen Schwerpunkt beizubehalten und es wurde eine möglichst große ebene Fläche auf der Abdeckung geschaffen. Somit können auf dieser Fläche diverse externe Sensoren oder zusätzlicher Aufbauten (z.B.: Prismen) montiert werden (Abbildung: 4.8 a und b). Durch die hauptsächliche Verwendung von beschichtetem Sperrholz, 2mm Aluminiumblech und Aluminiumwinkel konnte ein geringes Eigengewicht der Trägerplattform erreicht werden. Wie schon bei der offenen Bauweise der Zahnriemenuntersetzung wurde auch beim Aufbau auf eine wartungsfreundliche Konstruktion Wert gelegt, um jedmögliche Änderungen unkompliziert durchführen zu können.

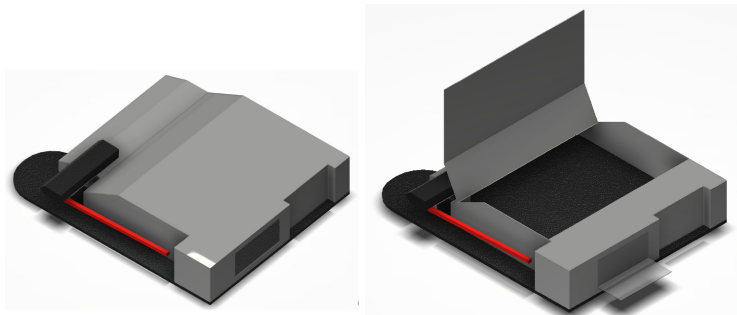


Abbildung 4.8: a) Geschlossene Wartungsklappen; b) Offene Wartungsklappen

Die für die Navigation notwendigen Sensoren wie: GPS, Magnetfeldkompass und Barometer, sind in einem aufklappbaren Sensorturm montiert (Abbildung: 4.9 a und b). Dieser Sensorturm erreicht im ausgeklapptem Zustand eine Höhe von  $\approx 65\text{cm}$ . Somit werden die Einflüsse durch die Motoren, die Elektronik und den Antennen auf den Magnetfeldkompass reduziert. Die Staudrucksonden zur Geschwindigkeitserfassung wurden in eine Vertiefung der Trägerplattform montiert, damit sie dem direkten Fahrtwind ausgesetzt aber dennoch vor Verschmutzung oder Beschädigungen geschützt sind (Abbildung: 4.9 c).



Abbildung 4.9: a) Sensorturm; b) Navigationselektronik; c) Staudrucksonden

Um die empfindlichen Elektronikmodule durch schädliche äußere Einflüsse zu schützen, wurden diese in einen eigens dafür konstruierten Rahmen eingebaut, der leicht aus der Trägerplattform herausgenommen und somit auch separat verwendet werden kann (Abbildung: 4.10).

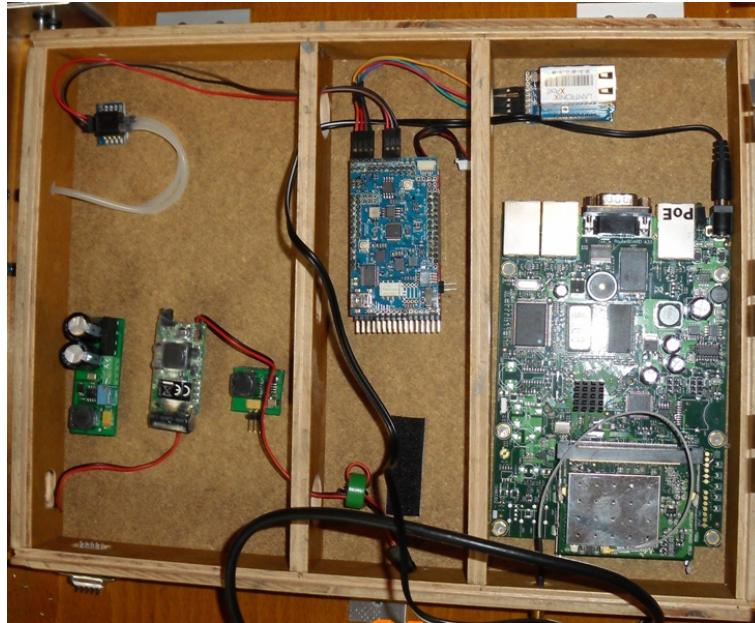


Abbildung 4.10: Elektronik Schutzrahmen

### 4.1.3 Elektrik/Elektronik

Die Roverelektrik und Elektronik wurde beim Bau in vier Abschnitte unterteilt: der Hochstromkreis zur Versorgung der Antriebsmotoren, die Elektronik-Stromversorgung, die Sensorikverkabelung und die Steuer- bzw. Regelelektronik. Die Hauptenergieversorgung kann durch einen Hauptschalter im Bedienterminal (Abbildung: 4.11 b) und einem Not-Aus auf der Trägerplattform (Abbildung: 4.11 a) getrennt werden.

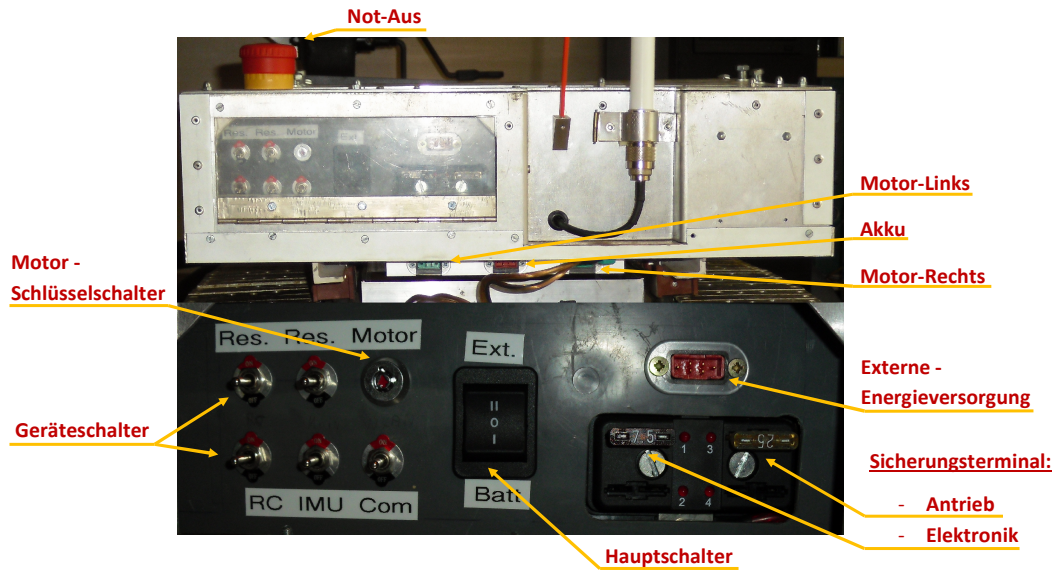


Abbildung 4.11: a) Anschlüsse und Not-Aus; b) Rover Bedienterminal

Durch drei Hochstromsteckverbindungen (60 A) wird die Trägerplattform mit dem Roverchassis verbunden (Abbildung: 4.11 a). Einer dieser Anschlüsse wird zur Energieübertragung zwischen Akku und Trägerplattform verwendet. Die zwei weiteren Verbindungen werden für den Anschluss der Antriebsmotoren benötigt und sind nur in Verbindung mit dem Antriebsakku (Hauptschalter in Position I) aktiv. Um ein unabsichtliches Anlaufen der Motoren zu verhindern wurde ein Schlüsselschalter für die Motoren eingebaut. Wird der Hauptschalter auf „Position II“ geschaltet, erfolgt die Energieversorgung extern. Bei dieser Schalterstellung wird nur die Hauptelektronik mit Energie versorgt und man hat dadurch den Vorteil, dass bei Wartungs- oder Programmierarbeiten weder Akku noch Roverchassis montiert sein müssen.

4.1.3.1 Spannungsversorgung

Die Hauptenergieversorgung erfolgt mit vier Blei-Gel-Akkus die jeweils 12 V und 13 Ah leisten. Diese Akkus werden wie Abbildung 4.12 gezeigt zu einem 24 V/26 Ah Akku-Pack zusammengeschlossen.

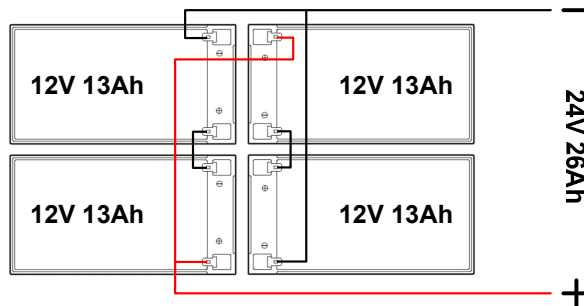


Abbildung 4.12: Schaltplan Akkupack

Wegen der hohen Lastströme kann der Hauptschalter nicht direkt für eine Umschaltung zwischen Akkubetrieb und externer Energieversorgung verwendet werden. Deshalb wurden für alle Hochstromkreise Relais mit einer Dauerstrombelastbarkeit von 50 A verwendet.

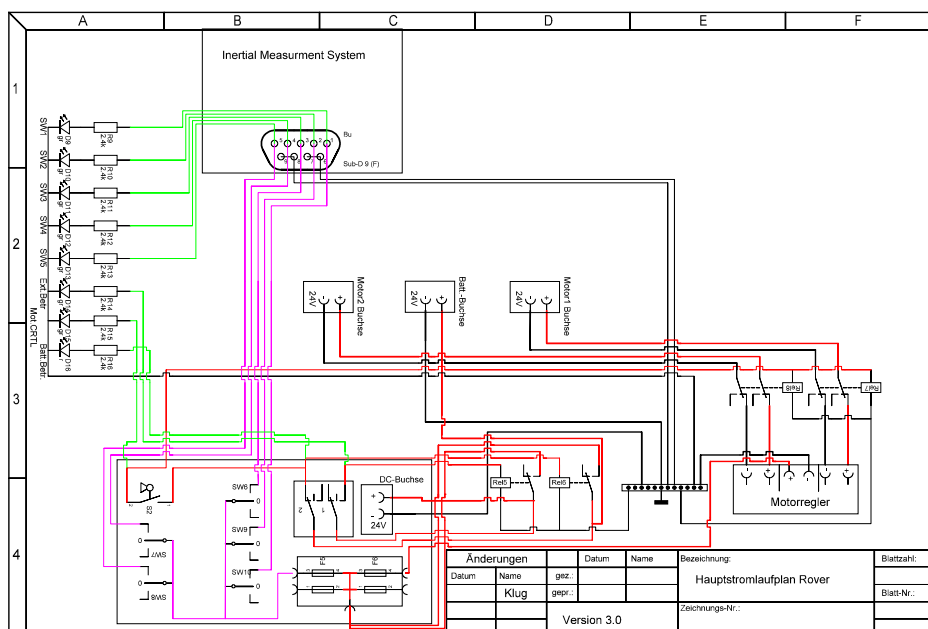


Abbildung 4.13: Stromlaufplan Energiekreis

Abbildung 4.13 zeigt den Schaltplan der Hauptenergieversorgung und der Steuerstromkreise. Wie schon bei der Bodenstation (Abschnitt: 4.2) wurde auch beim Rover eine Betriebsstatusanzeige aus Light Emitting Diode (LED)'s angefertigt. Durch diese Anzeige wird die Art der Energieversorgung und der Betriebsstatus einzelner Elektronikmodule signalisiert.



Den zweiten Abschnitt der Spannungsversorgung stellt der Anschluss der verwendeten Elektronikmodule im Schutzrahmen (Abbildung: 4.14) dar.

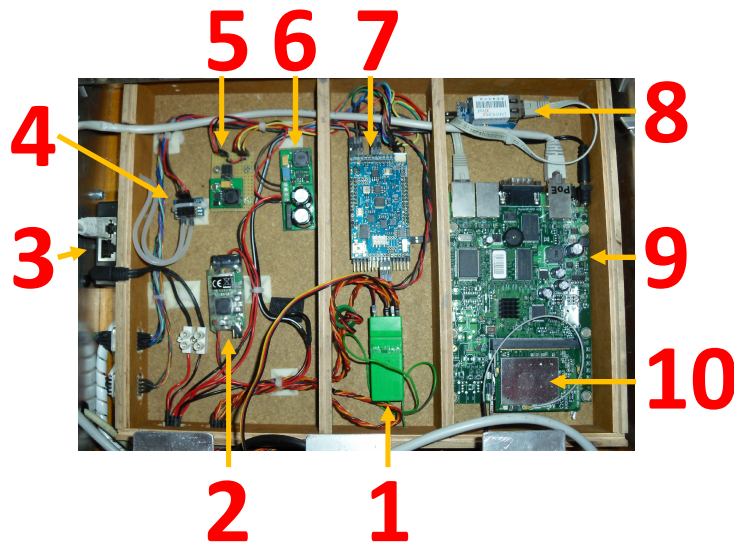


Abbildung 4.14: Schutzrahmen mit Elektronikkomponenten

Da die einzelnen Elektronikmodule (Abbildung: 4.14-Modulnummer: 1, 3, 7-10) unterschiedliche Betriebsspannungen benötigen, mussten diese durch verschiedene Spannungswandler (Abbildung: 4.14-Modulnummer: 2, 5, 6) generiert werden. In Tabelle: 4.2 werden die einzelnen Modulspannungen aufgeschlüsselt dargestellt.

Tabelle 4.2: Spannungen Elektronikmodule

Elektronikmodul	Betriebsspannung	Abbildung: 4.14-Modulnummer
RC-Empfänger	5 VDC	1
XPort-Kommunikationsmodul	5 VDC	8
Autopilot (APM)	9 VDC	7
WLAN-Modul	24 VDC	9 und 10
Kamera	24 VDC	3

Die technischen Spezifikationen der Spannungswandler und das Anschlussschema der Elektronikmodule wird in Tabelle 4.3 bzw. Abbildung 4.15 gezeigt.

Tabelle 4.3: Gleichspannungswandler Rover

Name	$U_{in}$	$U_{out}$	$I_{max}$	Abbildung: 4.14-Modulnummer
Step-Down	max. 60 VDC	1.2-37 VDC	1000 mA	6
BEC-Wandler	8-26 VDC	5 VDC oder 6 VDC	5000 mA	2
Step-Down W7805	max. 60 VDC	5 VDC	1000 mA	5

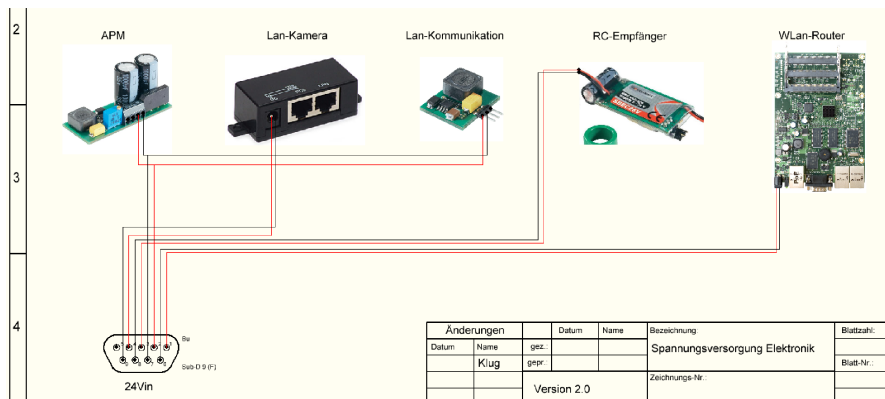


Abbildung 4.15: Anschlussschema Rover-Elektronik

Der Spannungseingang des Elektronikschutzrahmens erfolgt mit einer DB-9 Steckverbindung (vgl. Abbildung: 4.13 und 4.15).

#### 4.1.3.2 Autopilot (ardupilot mega v2.0)

In diesem Projekt wird das Autopilotensystem „Ardupilot Mega“ von „DIY-Drones“ eingesetzt. Dieser Autopilot dient als reine Sensor- und Kommunikationsschnittstelle zwischen dem Rover und der Bodenstation. Die Möglichkeit zur Erweiterung der autonomen Roversteuerung wurde schon bei der Konstruktion berücksichtigt und ist durch das Tauschen von zwei Steckverbindungen und einem Firmwareupdate leicht umzusetzen. Bei der aktuellen Konfiguration ist die Motorsteuerung direkt mit dem RC-Empfänger verbunden. Um den Autopiloten einzubinden müssen die Datenleitungen des RC-Empfängers und des Motorreglers an die Ein- und Ausgänge des APM-Systems angeschlossen werden.

Das Hauptsystem besteht aus zwei zusammensteckbaren Platinen, die verschiedene Anschlüsse für Kommunikation, Sensorik, Schalt- bzw. Regelaufgaben integriert haben. Die eine Platine wird für den Hauptprozessor und die Steuersignalgenerierung verwendet. Die zweite Platine, das IMU-Shield, ist die eigentliche Datenschnittstelle zwischen den Sensoren und dem Hauptprozessor. Mit der Open Source Arduino-Software und der Autopilotfirmware ist es möglich, das System nach individuellen Anforderungen zu programmieren und zu erweitern. Die im diesem Projekt verwendeten Elektronikmodule des Autopilotensystems und der Sensoren sind in Tabelle: 4.4 zusammengefasst.

Tabelle 4.4: Autopilot Elektronikmodule

Modul	Bezeichnung
Autopilot Mega	ArduPilot Mega - Arduino Mega compatible UAV Controller with ATmega 2560
Inertial Measurement Unit-Shield	ArduPilot Mega IMU Shield/Oil Pan
GPS-Modul	MediaTek MT3329 GPS V2.0
Barometer-Modul	Absolute Pressure and Temperature Sensor BMP085
Staudruck-Modul	Airspeed Kit with MPXV7002DP
Kompass-Modul	HMC5883L Triple Axis Magnetometer
Strom-/Spannungs-Sensor	AttoPilot Voltage and Current Sense Breakout 90A
Kommunikations-Modul	XtreamBee Board

Der hier verwendete Autopilot war nur als Bausatz erhältlich und wurde nach dem offiziellen „ArduPlane repository“ (DIY Drones, 2013) komplettiert. Auch die Verkabelung der externen Sensoren konnte größtenteils nach diesem Leitfaden gefertigt werden. Abbildung 4.16 zeigt den vollständigen Anschlussplan der Sensoren mit dem APM-System. Änderungen waren nur beim Kommunikationsmodul durch die Verwendung des „Lantronix“-XPort’s notwendig.

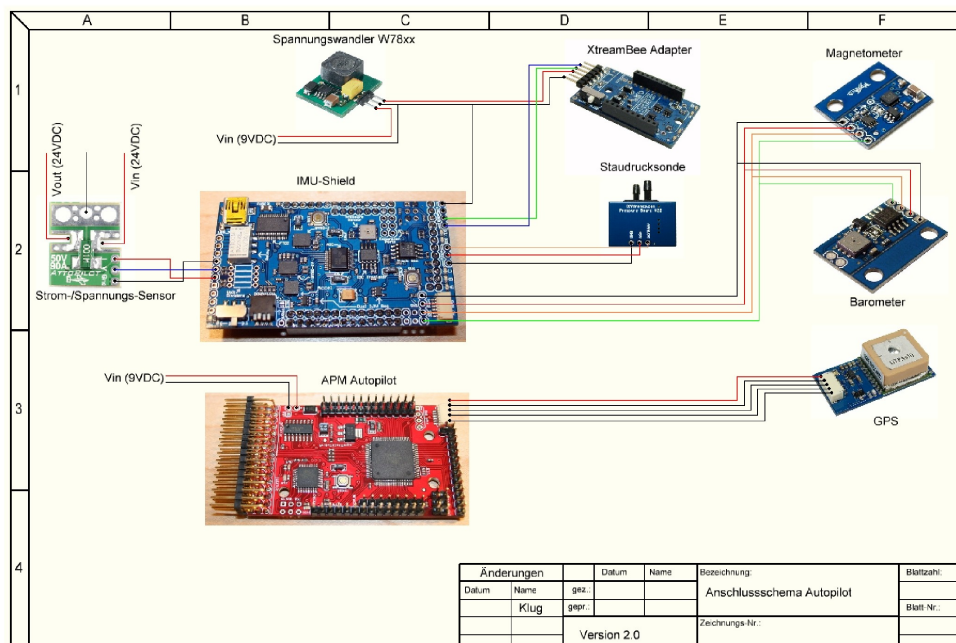


Abbildung 4.16: Stromlaufplan Roversensoren

Das XPort hat eine Stromaufnahme von 100mA, jedoch ist die APM-Schnittstelle auf 50 mA Ausgangsstrom begrenzt. Deshalb ist beim Anschluss an das Autopilot Mega (APM)-System darauf zu achten, dass beim XPort-Modul eine eigene Spannungsversorgung notwendig ist. Wird diese externe Spannungsversorgung nicht verwendet, ist keine ordnungsgemäße Kommunikation möglich.



## 4.2 Bodenstation

Die Bodenstation wird neben der Erfassung und Auswertung von Roverdaten auch für dessen Steuerung und der Berechnung und Ansteuerung der Antennennachführung verwendet. Sowohl die notwendige Elektrik und Elektronik als auch der Laptop und der Steuerungsjoystick sind in die Bodenstation integriert. Nicht alle Elektronikmodule werden für die einzelnen Tests und der Programmierung benötigt und können deshalb separat geschaltet werden. Um den Betriebszustand anzuzeigen, wurde eine entsprechende LED-Platine angefertigt. Alle notwendigen Anschlüsse für die Spannungsversorgung und der Kommunikation mit der Antennennachführung sind in einem Terminal zusammengefasst.

### 4.2.1 Chassis

Das Chassis der Bodenstation besteht aus einem 8mm bespannten Sperrholz (Abbildung: 4.17a). Diese Bodenstation verfügt über vier separate Fächer zur räumlichen Trennung der einzelnen Komponenten (Abbildung: 4.17b).

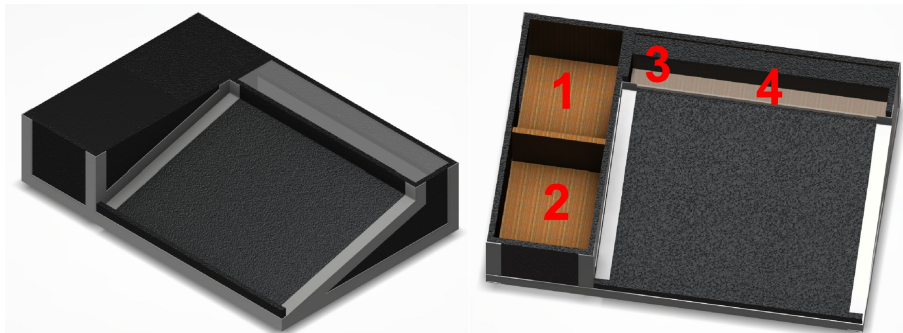


Abbildung 4.17: a) 3D-Konstruktionsplan Bodenstation; b) Ansicht Kammern

Nur wenige Handgriffe sind notwendig um die eingebaute Elektronik zu erreichen, was eine einfache Wartung gewährleistet. Die Kammern wurden so konstruiert, dass die Energieversorgung getrennt (Abbildung: 4.17b Nr.1) von der Steuer- bzw. Kommunikationselektronik (Abbildung: 4.17b Nr.4) untergebracht wurde. Die übrigen Kammern werden als Stauraum für den Joystick (Abbildung: 4.17b Nr.2), RC-Antenne (Abbildung: 4.17b Nr.3) und für Verbindungsleitungen genutzt. Die Maße der Bodenstation sind ( $L \times B \times H$  mm) 620 x 420 x 135 (ohne Antennen). Das Anschlussterminal beinhaltet alle notwendigen Steckverbindungen die zum Anschluss der externen Geräte benötigt werden (Abbildung: 4.18 und Tabelle: 4.5).

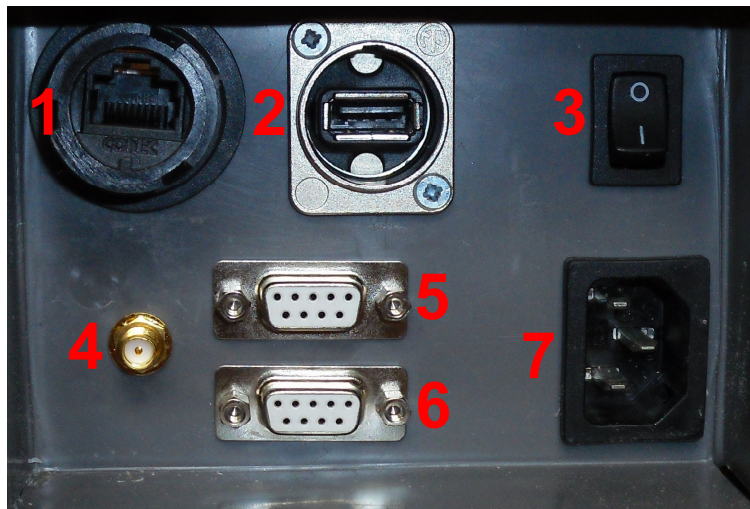


Abbildung 4.18: Bodenstation Anschlussterminal

Tabelle 4.5: Anschlüsse Bodenstation

Anschluss	Steckverbindung	Anschlussnummer (Abbildung: 4.18)
Kamera-Ausgang	RJ-45	1
GPS-Eingang	USB	2
230V-Hauptschalter	Einbaugeräteschalter	3
WLAN-Eingang	SMA-RP	4
Kompass-Eingang	DB-9	5
Servo-Ausgang	DB-9	6
230V-Eingang	Kaltgeräteeinbaubuchse	7

## 4.2.2 Elektrik/Elektronik

Die Elektronik und Elektrik der Bodenstation sind vollständig in das Chassis integriert und sind somit vor schädlichen Einflüssen weitgehend geschützt. Bei der Elektronik handelt es sich um alle Komponenten zur Steuerung des Rovers, der Ansteuerung der Antennennachführung und den Kommunikationskomponenten. Die Steuer- und Kommunikationselektronik wird noch im Kapitel 4.4 ausführlicher behandelt.

### 4.2.2.1 Spannungsversorgung

Die Spannungsversorgung in der Bodenstation ist in drei Stromkreise geteilt. Der erste Stromkreis ist der Netzanschluss mit 230 V-Wechselspannung. Diese Spannung wird mit einem Schaltnetzteil in eine 12 V-Gleichspannung umgewandelt und geht damit in den zweiten Stromkreis, dem Gleichspannungs-Hauptstromkreis, (Abbildung: 4.19) über. Der dritte Stromkreis übernimmt die Versorgung des USB-Hubs und des Computers. Da der USB-Hub bis zu 5000 mA Stromaufnahme hat, musste ein eigenes 5 V-Schaltnetzteil verwendet werden. Auch der Laptop wird mit einem eigenen Schaltnetzteil versorgt.

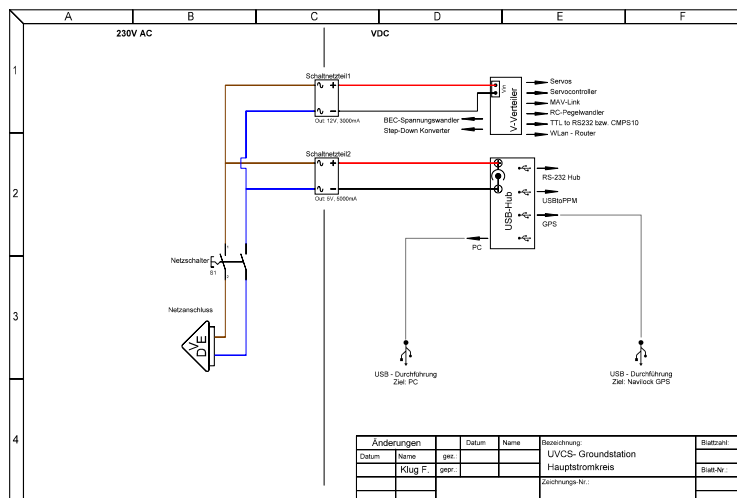


Abbildung 4.19: Stromlaufplan 230 V

Abbildung 4.19 zeigt den Schaltplan des 230 V-Stromkreis zum USB-Hub und zum Gleichspannungsverteiler. Da verschiedene Gleichspannungen notwendig sind, kommen sogenannte Step-Down-Konverter oder DC-DC-Wandler zum Einsatz. Teilweise mussten die Elektronikmodule selbst angefertigt werden, da für diese speziellen Spannungen keine Fertigbauteile erhältlich sind. Für den Bau der Step-Down-Wandler wurde ein einstellbarer LM317-Wandler verwendet (National Semiconductor, 2011). Um die notwendige Ausgangsspannung zu erhalten, muss der Widerstand **R4** aus Abbildung 4.20 entsprechend der Formel: 5.33 ausgewählt werden.

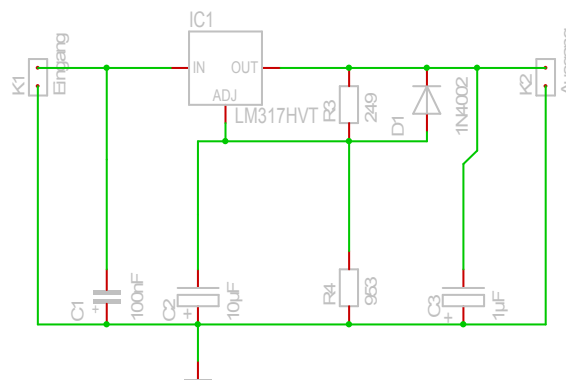


Abbildung 4.20: Grundsaltung LM317 Step-Down Wandler

Mit einer angefertigten Hauptverteilerplatine werden die Elektronikkomponenten nicht direkt an die Spannungswandler angeschlossen. Diese Platine verfügt für die unterschiedlichen Spannungspegel über separate, steckbare Eingänge und Ausgänge. Abbildung 4.21 zeigt das Anschlussschema der Spannungswandler mit der Hauptverteilerplatine.

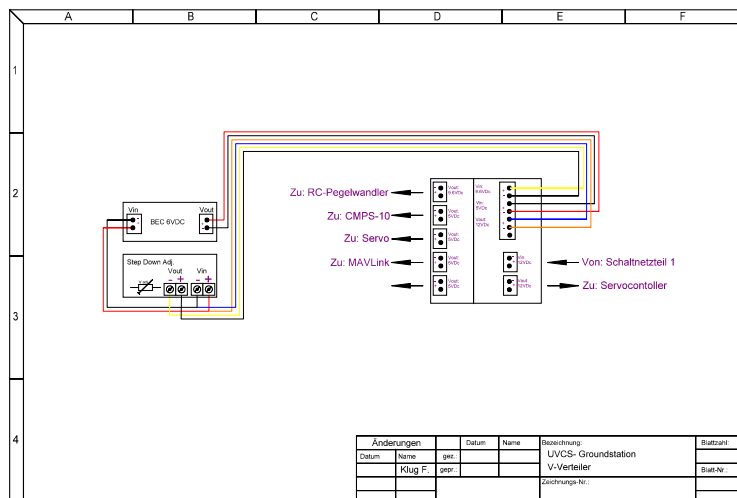


Abbildung 4.21: Spannungswandler und Anschlusschema Spannungsverteiler

Die Hauptverteilerplatine wurde so gefertigt, dass es eine zusätzliche Steckleiste für ein Schaltermodul und einer LED-Anzeige gibt (Abbildung: 4.22). Mit dem Schaltermodul kann jedes Steuer- und Kommunikationselement einzeln mit Spannung versorgt werden, was besonders für die Funktionstests und für die Programmierung von Vorteil ist. Da die Elektronikmodule fest im Chassis eingebaut und dadurch nicht direkt zugänglich sind, wird der Betriebszustand der Einzelmodule durch LED's angezeigt.

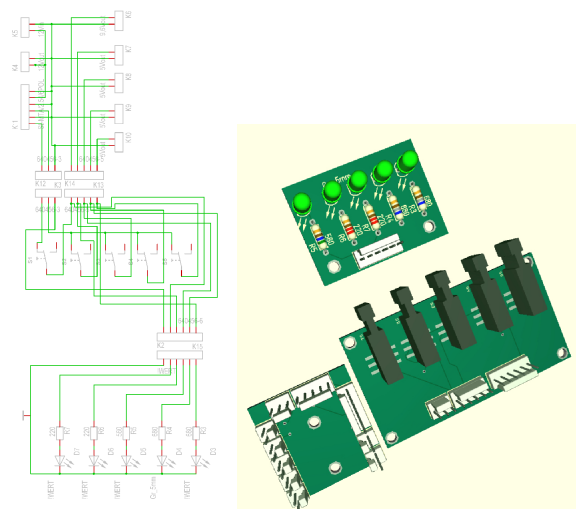


Abbildung 4.22: a) Stromlaufplan; b) Einzelplatinen Spannungsverteiler

Die für die Anzeige notwendigen Vorwiderstände wurden nach Formel 5.31 abgeschätzt und entsprechend ausgewählt. Tabelle 4.6 stellt die in der Bodenstation verwendeten Spannungswandler mit den Basisspezifikationen dar. In Tabelle 4.7 sind die Eingangsspannungen der einzelnen Elektronikmodule zusammenfassend dargestellt.

Tabelle 4.6: Gleichspannungswandler Bodenstation

Name	$U_{in}$	$U_{out}$	$I_{max}$	Hersteller
MeanWell	85-264 VAc	12 VDC	5000 mA	Fa. ELV-Electronics
Step-Down	max. 60 VDC	1.2-37 VDC	1000 mA	Eigenbau mit LM317
BEC-Wandler	8-26 VDC	5 VDC oder 6 VDC	5000 mA	Fa. Conrad-Electronics
Levelshifter	12 VDC	9.6 VDC	1000 mA	Eigenbau mit LM317
	3.8 VDC (Signal)	6 VDC (Steuereingang)		
		9.6 VDC (Signal)		

Tabelle 4.7: Spannungen Elektronikmodule

Elektronikmodul	Eingangsspannung
WLAN-Modul	12 VDC
Levelshifter	9.6 VDC
Servocontroller	12 VDC
Servoversorgung, Kompassmodul, RS232-Hub	5 VDC

Die Schaltpläne zu den einzelnen Elektronikmodulen sind in den jeweiligen Unterkapiteln 4.4.2 (*Telemetrie- und Bilddatenübertragung*) und 4.4.1 (*Roversteuerung*) abgebildet.

## 4.3 Antennennachführung

Wie bereits in Kapitel 2.1.3 beschrieben, wird die Antennennachführung zur Ausrichtung der Richtantennen auf den Rover benötigt. Es werden zwei unabhängige Achsen (Neigungs- und Drehachse) angesteuert. Dieses Kapitel beschreibt die mechanische Umsetzung sowie die verwendete Elektronik der Antennennachführung. Für die Antriebskomponenten der Nachführung wurde das Gewicht von zwei Patchantennen als Richtwert verwendet. Der mechanische Aufbau ist eine vollständige Eigenkonstruktion und besteht zu einem Großteil aus Aluminium.

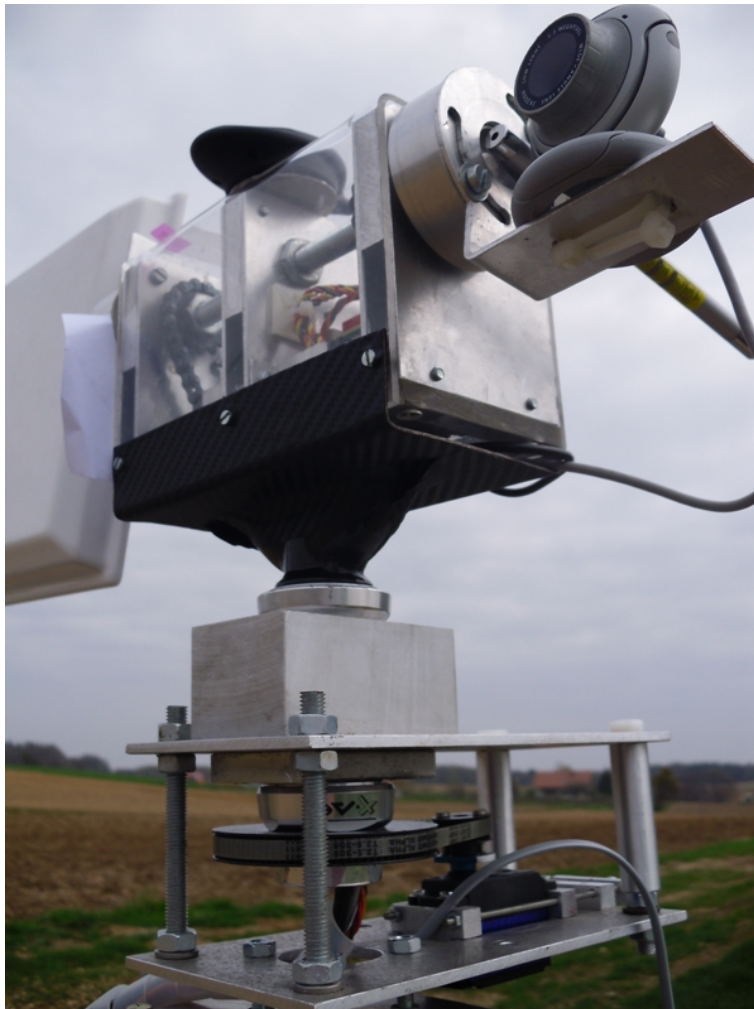


Abbildung 4.23: Antennennachführung

### 4.3.1 Chassis

Das Chassis der Nachführung unterteilt sich in eine Dreh- (Abbildung: 4.24a) und eine Neigungseinheit (Abbildung: 4.24b), die separat konzipiert und gebaut wurden. Durch diese Unterteilung ist es durchaus möglich bei einem alternativen Einsatz des Chassis nur die Drehachse zu verwenden und auf den Aufbau der Neigungseinheit zu verzichten.

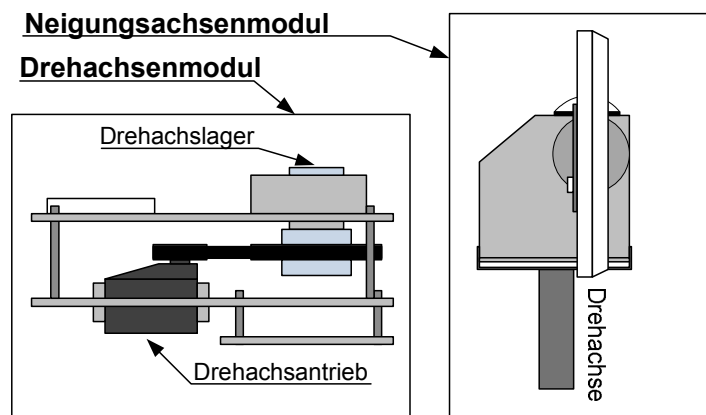


Abbildung 4.24: a) Drehachsenmodul; b) Neigungsachsenmodul

### Drehachse (Pan)

Das Drehachsen-Chassis wurde mit Hilfe einer geeigneten Halterung am Kamerastativ angebracht. Dieses Drehachsen-Chassis besteht aus einer 3mm Aluminium-Grundplatte zur Pan-Servoaufnahme und einer Lagerplatte zur Drehachslagerung. Die Grundplatte wurde dabei so angefertigt, dass die Servobefestigung gleichzeitig auch als Spannvorrichtung dient. Diese Spannvorrichtung ist deshalb erforderlich, da die Drehachse mit einem Zahnriemen angetrieben wird. Die Zahnriemenscheiben bestehen aus Kunststoff und wurden als Fertigteile, mit einem passenden Untersetzungsverhältnis, von der Firma Mädler bezogen. Das Untersetzungsverhältnis für die Drehachse ist mit  $\approx 1 : 3,5$  festgelegt. Das Pan-Servo lässt insgesamt eine  $\approx \pm 2520^\circ$ -Drehung zu, das entspricht 7 Umdrehungen. Dafür wurde eine möglichst große Untersetzung gewählt. Jedoch für eine gewünschte  $360^\circ$ -Drehung wäre eine Untersetzung von  $\approx 1 : 7$  notwendig, was aber aus Platzmangel nicht umgesetzt werden kann. Deswegen wird die restliche Einschränkung der Drehfreiheit durch die Software erzielt (Kapitel: 5.5).

Abbildung 4.25: a) Lenkgabel; b) Lenkkopflager<sup>1</sup>

Für die Lagerung der Drehachse wurde ein Fahrradlenkkopflager (Abbildung: 4.25b) verwendet, das in einen Aluminiumblock eingearbeitet wurde. Als Drehachse kommt eine Fahrradlenkgabel (Abbildung: 4.25a) zum Einsatz, die "auf den Kopf" gestellt eingebaut und angepasst wurde. Diese Art der Lagerung garantiert zum einen eine einfache, kostengünstige und spielfreie

<sup>1</sup>[http://de.wikipedia.org/wiki/Steuersatz\\_%28Fahrrad%29](http://de.wikipedia.org/wiki/Steuersatz_%28Fahrrad%29)



Montage, und zum anderen hat sie den großen Vorteil, dass es sich um eine Hohlwelle handelt. Durch diese Hohlwelle ist es möglich, alle Verbindungsleitungen durch sie hindurch zu führen, wodurch keine Probleme mit der Leitungsverlegung entstehen.

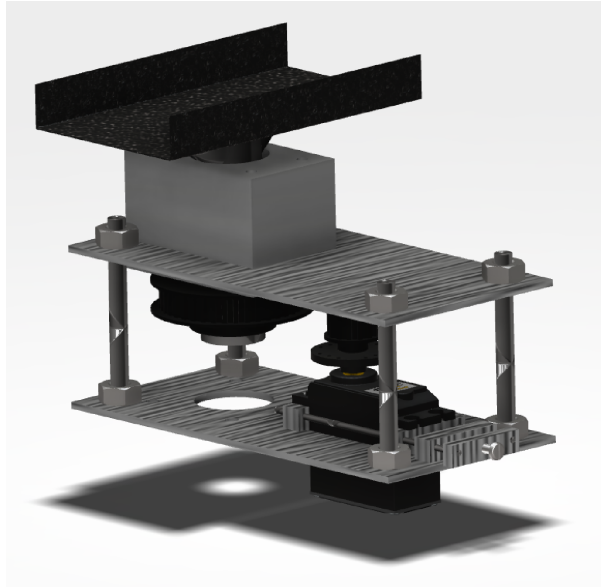


Abbildung 4.26: 3D-Konstruktionsplan Pan-Modul

### Neigungsachse (Tilt)

Der Aufbau der Neigungsachslagerung besteht aus drei Kugellagern. Zum einen besteht sie pro Kugellager aus zwei 2mm Alu-Abdeckblechen und zum anderen aus einem 10mm Flachaluminium. Jedes dieser Kugellager wurde in das Flachaluminium eingepresst (Abbildung: 4.27) und mit den Abdeckblechen verschraubt.

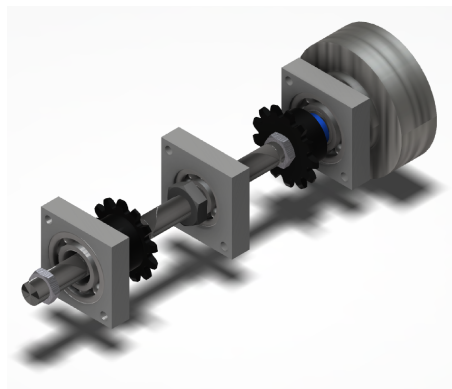


Abbildung 4.27: 3D-Konstruktionsplan Tilt-Lagereinheit

Dadurch entstehen drei Lagerböcke, die auf eine 8mm Polymethylmethacrylatplatte (Plexiglas) montiert wurden. Diese Grundplatte enthält neben der Kabelbaumdurchführung auch die Servohalterungen für die zwei Neigungsservos. Als Neigungsachswelle wird eine M10-Gewindestange verwendet. Um das Verschieben der Welle zu verhindern, wurden M10-Muttern mit einem Bund versehen und verschraubt. Auf der Welle und den beiden Servos wurden Ket-



tenräder montiert, welche die Neigungsachse antreiben. An den beiden freiliegenden Enden der Neigungswelle wurden zwei Befestigungsflansche montiert, um die Montagekonsolen aufzunehmen. Sowohl die Antenne als auch die Kamera wurden an den Montagekonsolen befestigt. Wie schon die Grundplatte wurde auch die Abdeckung der Mechanik aus einer Polymethylmethacrylatplatte (Plexiglas) gefertigt. Abschließend wurde der gesamte Neigungsachsenaufsatz auf der Befestigungsplatte der Drehachse montiert.

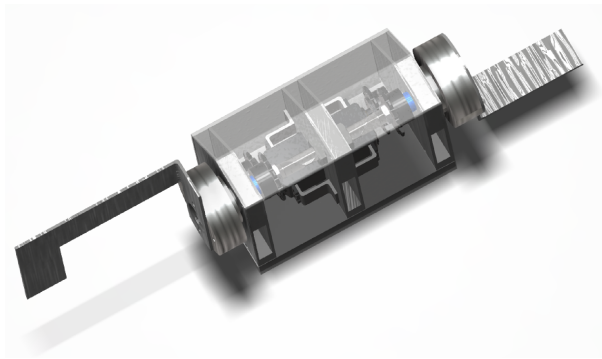


Abbildung 4.28: 3D-Konstruktionsplan Tilt-Modul

Im Gegensatz zu gängigen Konstruktionen erlaubt dieses Design eine größere Flexibilität bei der Montage des Equipments (Abbildung: 2.11a und b). Der große Vorteil bei dieser speziellen Konstruktion, bei der ein angepasster Adapter zwischen den Befestigungsflansche und dem Equipment verwendet wurde, ist, dass der Adapter bei einem Tausch des Equipment leicht ausgewechselt werden kann. Jedoch werden bei der Drehung um die Pan-Achse dieser Konstruktion Fliehkräfte freigesetzt. Doch durch den stabilen Aufbau und den Antrieb werden diese Fliehkräfte aufgenommen (Abbildung: 4.29).

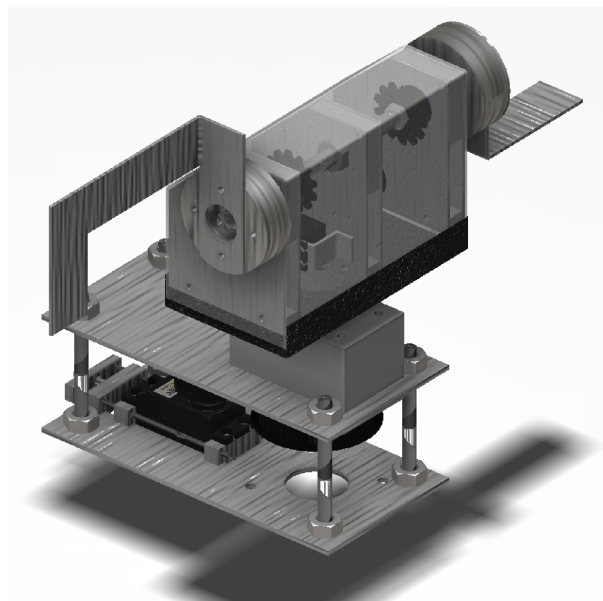


Abbildung 4.29: 3D-Konstruktionsplan Antennennachführung

## 4.3.2 Elektrik/Elektronik

### 4.3.2.1 Antrieb

Wie bereits beschrieben wurden für den Antrieb der beiden Achsen handelsübliche RC-Servos verwendet. Die Neigungsachse wird von zwei Servos des Herstellers „Hitec“ (Abbildung: 4.30b) angetrieben. Die Belastungen der Neigungsachsen konnten zu Baubeginn noch nicht exakt bestimmt werden und da auch die Standard-Modellbauservos nur über eine beschränkte Belastungsgrenze ( $\approx 0.85 \text{ Nm}$ ) verfügen kamen die oben genannten Servos zum Einsatz. Bei der Neigungswelle handelt es sich um eine durchgehende Welle, daher mussten beide Antriebsservos synchronisiert und aufgrund der Einbaulage, in der Drehrichtung angepasst werden. Ein Servo-Verzögerungsbaustein (Abbildung: 4.30a) der Firma „Conrad“ übernimmt dabei die Synchronisation und die Drehrichtungsanpassung. Durch diese Elektronik wird nur eine Anschlussleitung für beide Servos benötigt. Das bringt den Vorteil, dass nur ein Ausgangskanal des Servocontrollers belegt wird. Der Drehachsenantrieb erfolgt mit einem Rs-10-Segelwindenservo (Abbildung: 4.30c) vom Hersteller „Modelcraft“. Dieser Servo besitzt ein Stellmoment von  $1.2 \text{ N/m}$  ( $\approx 0.122 \text{ Kg/m}$ ) was mit der Untersetzung ausreichend ist.



Abbildung 4.30: a) Servo-Inverter; b) Neigungsachsenservo Hitec HS-645MG; c) Drehachsenservo Modelcraft RS-10<sup>3</sup>

Zur Kraftübertragung auf die Neigungs- und Drehachswelle wurden geeignete Zahnriemen- und Kettenantriebsscheiben verwendet.

<sup>3</sup><http://www.conrad.de>

### Antriebssteuerung (Pololu, 2006)

Die Servos verwenden zur Positionierung ein PPM-Signal, das mit einem Servocontroller generiert wird. Die dafür eingesetzte Elektronik wurde von der Firma „Pololu“ (Abbildung: 4.31) hergestellt und kann acht Servokanäle mit einer RS232-Schnittstelle ansteuern.



Abbildung 4.31: Pololu 8-Kanal Servocontroller<sup>4</sup>

Bei diesem Controller ist von Vorteil, dass sowohl die Logik als auch die Servoversorgung einen eigenen Spannungseingang haben. Bei großen Servoströmen wird hierbei eine Beschädigung der Sterelektronik entgegengewirkt. Abbildung 4.32 zeigt den Anschluss der Servos, als auch der beiden Spannungseingänge.

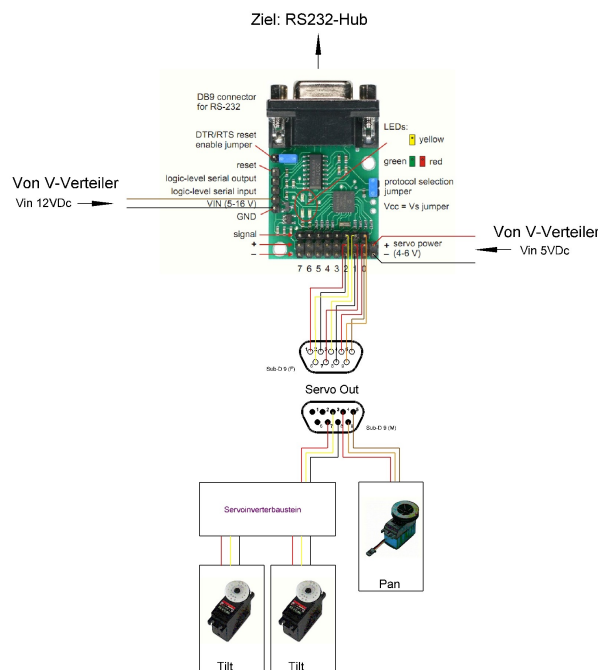


Abbildung 4.32: Antennennachführung Antriebsschaltplan

Der Controller kann mit weiteren Controllern verbunden werden, wodurch ein gleichzeitiger Betrieb von bis zu 254 Servos mit einer RS232-Schnittstelle möglich ist. Für die Ansteuerung kann zwischen zwei unterschiedlichen Protokollen gewählt werden. Das „MiniSSCII-Protokoll“ ist eine vereinfachte Form des „Pololu-Protokolls“ ((Pololu, 2006)). In diesem Modus sind weniger Funktionen implementiert als im „Pololu-Protokoll“ und die erreichbare Auflösung ist auf

<sup>4</sup><http://www.pololu.com/catalog/product/727>

254-Schritte begrenzt (Abbildung: 4.33b).

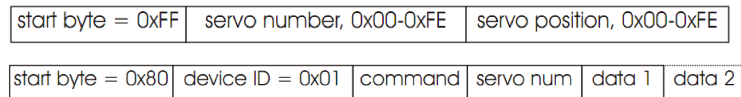


Abbildung 4.33: a) MiniSSC II Protokollstruktur; b) Pololu-Mode Protokollstruktur<sup>5</sup>

Abbildung 4.33a zeigt die im Projekt implementierte Protokollstruktur. Es sind neben verschiedenen Kommandos auch zwei Datenbytes für die Positionierung vorgesehen. Dadurch wird der Wirkungsbereich auf 32767-Einzelschritte aufgelöst. Die Protokollauswahl kann durch einen Jumper direkt am Controllerboard vorgenommen werden. Die vollständige Protokollaufschlüsselung ist dem Manual zu entnehmen.

#### 4.3.2.2 Positionierungs- und Ausrichtungselektronik

Wie schon im Konzept festgelegt, wird für die Bestimmung des Neigungs- und Richtungswinkels der Antenne das Positionierungsverfahren (2.1.3) verwendet. Hierfür müssen die Position und die Orientierung der Antennennachführung bekannt sein. Die Orientierungs- und Positionsbestimmung soll jedoch nicht mit dem Autopilotensystem des Rovers, sondern unabhängig von diesem erfolgen. Aus diesem Grund wurde ein Ein-Frequenz-USB-GPS-Empfänger (Abbildung: 4.34b) und ein neigungskompensierter Magnetfeldkompass (Abbildung: 4.34a) in die Antennennachführung eingebaut.

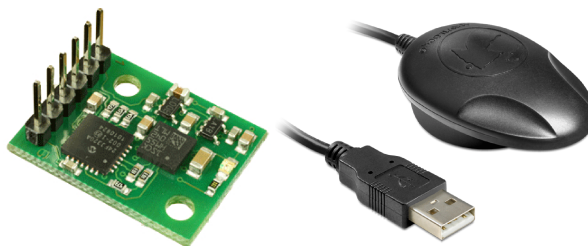


Abbildung 4.34: a) Devantech CMPS10 Kompassmodul; b) Navilock NL-302U GPS-Empfänger<sup>6</sup>

#### CMPS10 Kompassmodul Devantech (2012)

Das CMPS10-Kompassmodul wird für die Bestimmung der Orientierung der Antennennachführung verwendet. Es handelt sich bei diesem Modul um einen neigungskompensierten Drei-Achs-Magnetfeldkompass, der zur Neigungskompensation die Erdanziehungskraft mit einem Drei-Achs-Beschleunigungssensor misst. Dadurch ist auch bei einer Horizontalabweichung des Kompasses eine korrekte Richtungsbestimmung möglich. Einflüsse wie Deviation (Kompass Nord → Magnetisch Nord) oder Variation (Magnetisch Nord → Geographisch Nord) (Abbildung:

<sup>5</sup>(Pololu, 2006)

<sup>6</sup>a)Devantech (2012); b)Navilock (2013)

4.35) sind bei der Datenprozessierung noch zu berücksichtigen. Es ist auch zu beachten, dass es durch magnetfeldbeeinflussende Metalle in der direkten Umgebung des Kompassmoduls zu Missweisungen kommen kann.

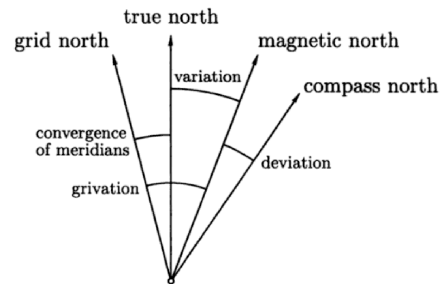


Abbildung 4.35: Transformationswinkel Nordrichtung<sup>7</sup>

Die Herstellerangaben (<http://www.robot-electronics.co.uk/acatalog/Compass.html>) für die Auflösung und Genauigkeit zeigt die Tabelle 4.8.

Tabelle 4.8: CMPS10 Herstellerangaben

Richtungsauflösung	0.1°
Neigungsauflösung	1°
Genauigkeit Horizontal	0.5%
Genauigkeit bis ±60°	0.5%

Angesichts des Öffnungswinkels der Richtantennen wurde die Richtungsgenauigkeit als ausreichend eingestuft. Da der Sensor über drei unterschiedliche Schnittstellen verfügt, kann dieser mit einem Computer oder einer anderen Elektronik kommunizieren. Abbildung 4.36 gibt das verwendete Anschlusschema des Kompassmoduls an.

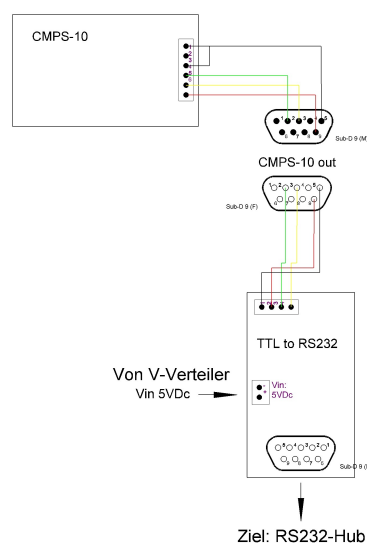


Abbildung 4.36: CMPS10 Kompassmodul Anschlussplan

<sup>7</sup>(Bernhard Hofmann-Wellenhof *et al.*, 2003)

Für das Projekt wurde die TTL-Schnittstelle verwendet und daher musste ein entsprechender Signalwandler zwischen dem Modul und der RS232-Schnittstelle geschaltet werden. Würde der Kompass direkt mit der RS232-Schnittstelle verbunden werden, würde dieser durch die unterschiedlichen Spannungspegel beschädigt oder gar zerstört werden (Kapitel: 3.2.1; Tabelle: 3.1).

Die Datenübertragung erfolgt mit einer bidirektionalen Kommunikation (Kapitel: 3.2.2 Abschnitt: Bidirektional). In Tabelle 4.9 sind die Kommandosequenzen für die notwendigen Daten zusammengefasst.

Tabelle 4.9: Kommandosequenzen CMPS-10 Kompassmodul

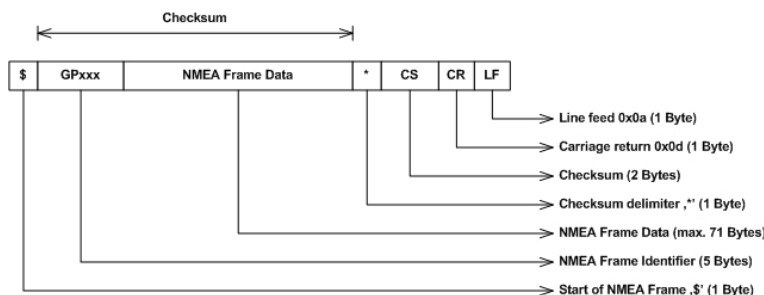
Kommando	Name	Datenbytes	Datenbeschreibung
0x13	Yaw-Winkel 16Bit	2	16-Bit, High Byte first
0x14	Pitch-Winkel	1	8-Bit, ±0 – 85°
0x15	Roll-Winkel	1	8-Bit, ±0 – 85°
0x23	Yaw-, Pitch-, Rollwinkel	4	16-Bit (High Byte first), 8-Bit, 8-Bit

Die Softwareimplementierung zur Datenkonvertierung wird in Kapitel 6.2.2 genauer beschrieben.

### Navilock GPS-Empfänger

Zur absoluten Positionsbestimmung der Antennennachführung kommt ein „Navilock“ GPS-Empfänger zum Einsatz. Es handelt sich um einen Ein-Frequenz-USB-Empfänger mit einem SiRFstarIII-Chipsatz und einer eingebauten Aktiv-Antenne. Die Positionsgenauigkeit wird von „Navilock“ mit 10 Meter (2D-RMS) angegeben (<http://www.navilock.de/produkte/G.61422/dokumente.html>).

Der GPS-Empfänger verwendet für die Datenübertragung ein NMEA-0183-Protokoll und eine virtuelle RS232-Schnittstelle (Abbildung: 4.37). Bei einer Verbindung des Empfängers mit dem Computer beginnt sofort die Datenübertragung. Es handelt es sich daher um eine unidirektionale Kommunikation (Kapitel: 3.2.2, Abschnitt: Unidirektional).



Example:  
\$GPGSV,3,1,12,29,14,099,,18,44,124,28,03,45,292,34,21,64,067,24\*78

Abbildung 4.37: NMEA-0183 Datenprotokoll<sup>8</sup>

<sup>8</sup><http://www.ralf-hochhausen.de/wp-content/uploads/2012/03/NMEAFrame.png>

Ein entsprechender Parser, der die NMEA-0183-Daten in ein lesbares Format umwandelt, wird in Kapitel 6.2.2 erklärt.

#### 4.3.2.3 Richtantenne

Als Richtantenne wurde eine Patchantenne der Firma „Airwin“ (Abbildung: 4.38a) verwendet. Richtantennen zeichnen sich durch eine gebündelte Energieabstrahlung aus. Der Verstärkungsfaktor (Gain) ist dabei ein Maß für die Richtwirkung. Bei der Verwendung von Antennen mit einem hohen Gain sind große Übertragungreichweiten möglich. Ein hoher Gain-Wert bedeutet jedoch auch eine hohe Richtungsempfindlichkeit. Hierfür muss ein Mittelweg zwischen dem Antennengewinn und dem Richtbereich gefunden werden.



Abbildung 4.38: a) Airwin Richtantenne<sup>9</sup>; b) Antennenkonsole

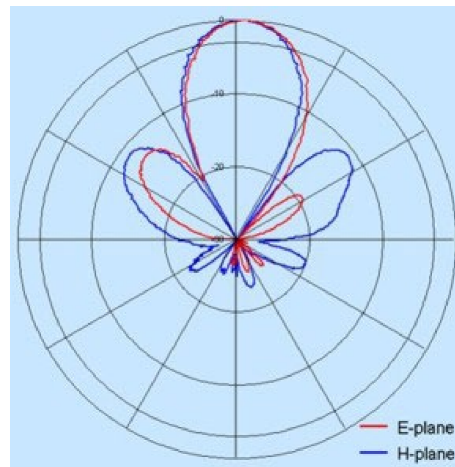
Die technischen Spezifikationen des Herstellers „Airwin“ sind in Tabelle 4.10 ([www.triotronik.com](http://www.triotronik.com)) zusammenfassend dargestellt.

Frequenzbereich	2400-2500 MHz
Verstärkung	16 dBi
3dB Abstrahlung V/H	28°/28°
Abmessungen: L x B x H	205 x 205 x 23mm

Abbildung 4.39 zeigt die Abstrahlcharakteristik der Horizontal- und Vertikalebene bei einem Energieverlust von 3 dB. Die verwendete Antenne wurde an die Montagekonsole der Antennennachführung angeschlossen (Abbildung: 4.38b). Die Anschlussleitung konnte hinsichtlich der Konstruktion zwischen dem Dreh- und Neigungsachsmodul zur Hohldrehachse geführt werden.

<sup>9</sup>[www.triotronik.com](http://www.triotronik.com)

<sup>10</sup>[www.triotronik.com](http://www.triotronik.com)

Abbildung 4.39: Polarplot Richtantenne<sup>10</sup>

## 4.4 Systemübergreifende Elektronik

Die systemübergreifende Elektronik behandelt die Kommunikation und die Steuerung des Rovers und der Bodenstation. Abbildung 4.40 zeigt das in Kapitel 2.1.1 beschriebene Kommunikationsschema.

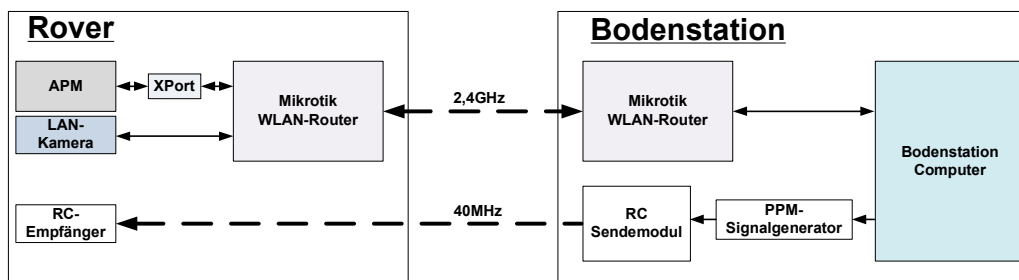


Abbildung 4.40: Kommunikations Konzept

### 4.4.1 Roversteuerung

Die Roversteuerung wird mit einer RC-Signalübertragung durchgeführt. Für die Bodenstation gibt es keine Fertigelektronik, deshalb musste ein eigenes System aus verschiedenen Einzelmodulen zusammgebaut werden. Diese Einzelmodule bestehen aus einem PPM-Signalgenerator, einer Pegelanpassungsplatine sowie ein 40 MHz-Sendemodul von „Graupner“. Beim Rover selbst kommen ein handelsüblicher RC-Empfänger und ein Dualer Motorregler zum Einsatz.

Wie bereits in Kapitel 2.1.1 beschrieben, wird für die Steuerung des Rovers der Signalgenerator von „CompuFly“ verwendet. Mit diesem Generator können sowohl positive als auch



negative PPM-Signale mit dem Computer generiert werden. Die Anzahl der Steuerkanäle wird mit der Steuersoftware eingestellt. Das ist vor allem für den verwendeten Empfänger wichtig da dieser auf eine Kanalanzahl abgestimmt ist. Dieser Signalgenerator ist aber nicht für einen direkten Anschluss an ein Sendemodul konzipiert und wird daher normalerweise über eine Lehrer-Schüler-Verbindung mit einem RC-Sender verbunden. Diese Lehrer-Schüler-Systeme verwenden je nach Hersteller, Signalspannungen zwischen 3.6 V und 4.8 V. Bei diesem Projekt wird jedoch kein Handsender verwendet, sondern es wird das HF-Sendermodul direkt mit dem Signalgenerator verbunden. Aber auch das HF-Sendermodul ist nicht für eine solche Verbindung ausgelegt. Damit nun aber ein fehlerfreier Betrieb möglich ist, benötigt man einerseits zwei unterschiedliche Spannungsversorgungen und andererseits eine höhere Signalspannung als vom Signalgenerator selbst erzeugt werden kann. Daher wurde eine eigene Platine angefertigt, welche sowohl die verschiedenen Spannungen als auch den notwendigen Signalpegel bereitstellt (Abbildung 4.41).

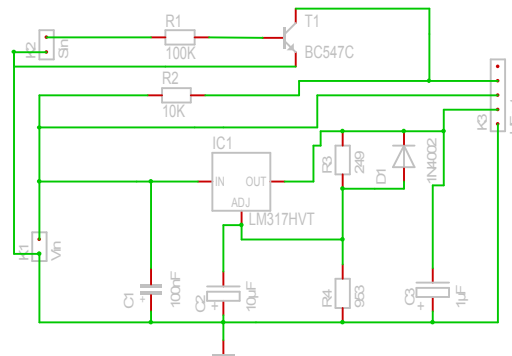


Abbildung 4.41: 40 MHz-Anschlussmodul

Bei der Signalpegelplatine wird eine Transistorschaltung verwendet, die das PPM-Signal auf eine Spannung von  $\approx 9,6$  V anhebt und invertiert. Zusätzlich ist ein Konstant-Spannungsregler integriert um die 9,6 V-Versorgungsspannung aus dem einstellbaren Step-Down-Konverter auf die benötigten 6 V zu bringen. Abbildung 4.42 zeigt das Anschlussschema der 40 MHz Anschlussplatine und des 40 MHz-Sendermoduls. Das Anschlussmodul hat einen eigenen Eingang für die Spannungsversorgung und einen eigenen Eingang für den PPM-Signalgenerator. Um das Anschlussmodul auch für andere Sendermodule verwenden zu können, wurde der Widerstand **R4** nicht direkt auf die Platine gelötet sondern mit einem IC-Sockel steckbar ausgeführt und nach Formel: 5.33 dimensioniert. In Tabelle 4.11 wird die Pinbelegung für die Eingangs- und Ausgangs-Steckerleisten dargestellt. Wurden die drei Module zusammengeschlossen, ersetzt der Computer vollständig einen RC-Handsender.

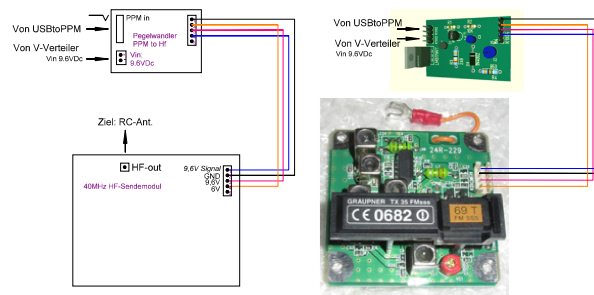


Abbildung 4.42: Anschlussschema 40MHz-Sendermodul

Tabelle 4.11: 40MHz-Anschlussmodul Pinbelegung

Steckleiste	Pin 1	Pin 2	Pin 3	Pin 4	Pin 5
K1 (USB2PPM-Eingang)	GnD	PPM 3.8 V	x	x	x
K2 (Spannung Eingang)	GnD	12 V	x	x	x
K3 (Sendermodul Ausgang)	GnD	6 V	9,6 V	PPM-Signal 9,6 V	x

Im Rover wird ein RC-Empfänger der Firma „Simprop“ eingesetzt. Dieser RC-Empfänger hat 6-Servokanäle, wovon einer zur Spannungsversorgung verwendet wird. Der Rover benötigt nur zwei Steuerkanäle für den Motorregler, deshalb werden im Endeffekt nur drei dieser Kanäle verwendet.

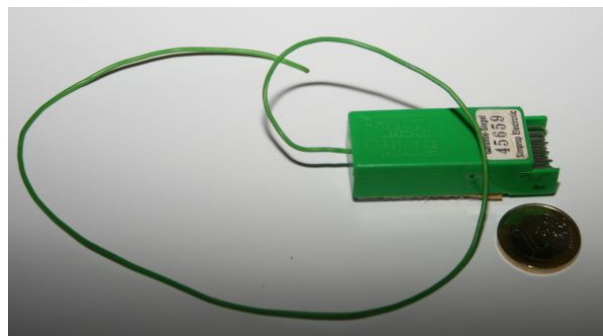


Abbildung 4.43: Simprop Pico2000 Empfänger

Für diese Anwendung ist die Antenne des Empfängers zu kurz weil sie vorwiegend im Aluminiumgehäuse der Trägerplattform verlegt werden musste. Deshalb wurde eine Antennenverlängerung angefertigt. Dabei ist vor allem auf die Wellenlängen 4.1 der Trägerfrequenzen zu achten, da sich aus deren die Antennenlängen berechnen.

$$\lambda = \frac{c}{f}; \tag{4.1}$$

Die besten Empfangseigenschaften werden erreicht, wenn die Antennen- und Wellenlänge einander gleicht. Jedoch bei niedrigeren Sendefrequenzen würde eine solche Antenne nach der Gleichung 4.1 mehrere Meter haben. Für ein praktikableres Längenmaß wird die Wellenlänge

durch einen ganzzahligen Teiler dividiert (Formel 4.2).

$$w = \frac{\lambda}{x}; \quad x = 2, 4, 8, 16, \dots \quad (4.2)$$

Aufgrund der Berechnungsergebnisse wurde eine Antenne mit einer aktiven Länge von  $\lambda/4$  verwendet. Diese aktive Länge entspricht bei einer Frequenz von 40 MHz  $\approx 1875$  mm.

Die Steuerung der Antriebsmotoren wird mit einem Dual-Motorregler der Firma „Sabertooth“ durchgeführt. Dieser Motorregler zeichnet sich durch eine hohe Dauerstrombelastbarkeit von 25 A (Kurzzeitig bis zur 50 A) und einer schnellen Reaktionszeit aus. Es ist auch ein „Failsafe“ integriert der bei einem Steuersignalverlust die Motoren stoppt. Da dieser Motorregler für die Robotik entwickelt wurde, bietet er verschiedene Regelmodi. Beide Motoren können entweder unabhängig voneinander über zwei Kanäle, oder in einem Differentialmodus gesteuert werden.

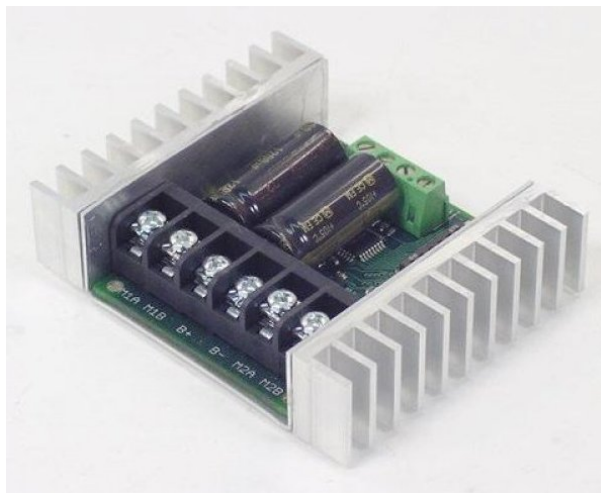


Abbildung 4.44: Dualer Motorregler<sup>11</sup>

Die Einstellungen des DIP-Schalters und die verschiedenen Anschlussmöglichkeiten werden im Manual des Motorreglers (DimensionEngineering, 2007) ausführlich erläutert.

#### 4.4.2 Telemetrie- und Bilddatenübertragung

Zur Telemetrie- und Bilddatenübertragung kommt das Routermodul (Abbildung 4.45a) mit einer WLAN-Steckkarte (Abbildung: 4.45b) von „Mikrotik“ zum Einsatz. Diese Router verfügen über einen hohen Grad zur freien Parametrisierung. Dadurch kann jeder dieser drei LAN-Ports direkt mit den drei LAN-Ports der Gegenstation verbunden werden. Diese Art der Portkonfiguration wird auch als Tunneling bezeichnet.

<sup>11</sup><http://nodna.de/Sabertooth-225A-regenerativer-Motor-Controller>

<sup>12</sup><http://routerboard.com/>

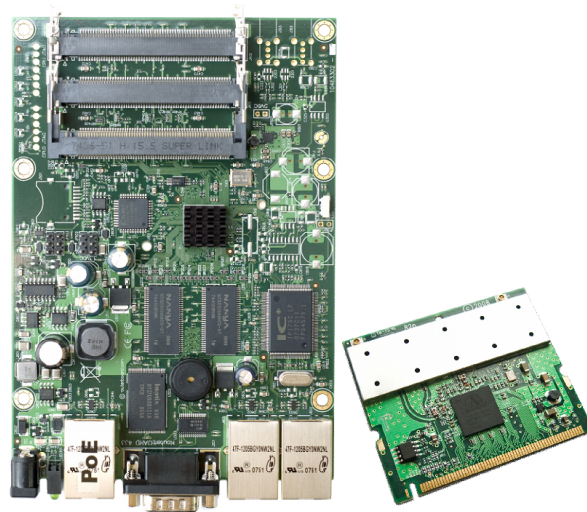


Abbildung 4.45: a) LAN-Board; b) WLAN-Card<sup>12</sup>

In diesem Projekt wurden zwei dieser Tunnel für die Datenübertragung konfiguriert. Diese Einstellung wurde in erster Linie für Testzwecke gewählt, um eine Trennung der Daten auf Seiten der Hardware durchzuführen. Einer der LAN-Ports wird für die Telemetriedatenübertragung mit dem „Lantronix XPort“ verbunden. Der zweite Port wird für den Anschluss einer LAN-Kamera verwendet. Da die Kamera über Power over Ethernet (PoE) mit Strom versorgt wird, musste ein geeigneter Adapter zwischengeschaltet werden (Abbildung 4.46).

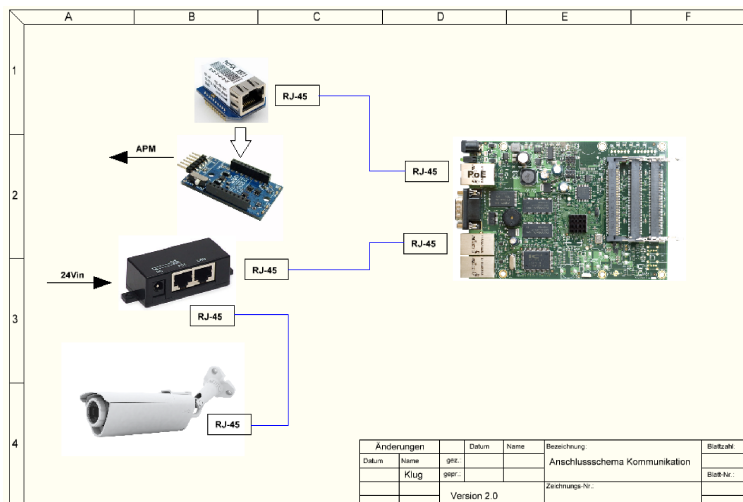


Abbildung 4.46: Anschlussschema Rover-Kommunikation

Das „Lantronix XPort“ ist ein TTL (seriell) zu „Ethernet“ Konverter. Der vollständige Schnittstellenwandler, zwischen der Autopilot-Kommunikationsschnittstelle und dem Router-board, besteht aus drei Hardwarekomponenten. Das „XtreamBee“-Board (Abbildung 4.47a) fungiert normalerweise als direkte Schnittstelle zwischen dem Autopilotensystem und einem „XBee“-Modul. Dieses „XtreamBee“-Board kann mit einem „XBee“-WLAN-Modul oder mit einem „XPort zu XBee“-Konverter (Abbildung 4.47b) bestückt werden. Dieser „XPort zu XBee“-Konverter

ist eine reine Adapterplatine, welche die verschiedenen Steckersysteme von XBee und XPort (Abbildung 4.47c) aufeinander abstimmt.

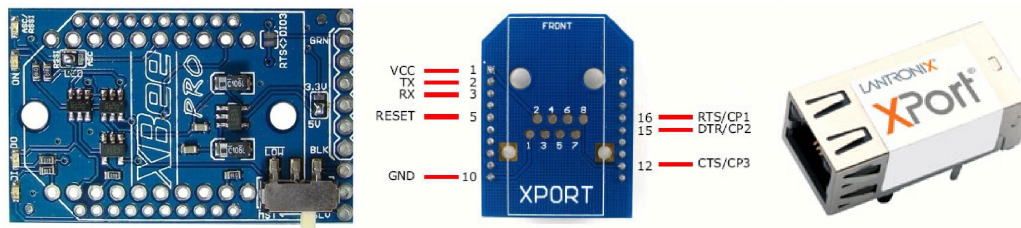


Abbildung 4.47: a) XtreamBee Board; b) XPort zu XBee Konverter; c) XPort<sup>13</sup>

Beim Anschluss an das Autopilot Mega (APM)-System ist zu bedenken, dass das „XPort“-Modul eine eigene Spannungsversorgung benötigt. Der maximale Ausgangsstrom an der APM-Schnittstelle ist mit 50 mA begrenzt, jedoch hat das XPort eine Stromaufnahme von  $\approx 100 \text{ mA}$ . Wird diese externe Spannungsversorgung nicht verwendet, kommt es zu Spannungseinbrüchen und verhindert eine Kommunikation. Der diesbezügliche Schaltplan ist in Kapitel 4.1.3.1 zu finden.

<sup>13</sup>a) [http://www.lipoly.de/index.php?main\\_page=product\\_info&cPath=880\\_1912\\_2496&products\\_id=153218](http://www.lipoly.de/index.php?main_page=product_info&cPath=880_1912_2496&products_id=153218)  
 b) <http://nodna.de/Droids-XBee-Simple-Board>  
 c) <http://www.lantronix.com/device-networking/embedded-device-servers/xport.html>

# Kapitel 5

## Berechnungen und Abschätzungen

In den folgenden Abschnitten werden die benötigten Berechnungen, die zur Datenaufbereitung (Abschnitt 5.1) und zur Basislinienbestimmung (Abschnitt 5.2) herangezogen wurden, behandelt. Weiter werden im Abschnitt 5.3 die Azimut- und Zenitwinkel in das Zahlenformat des Servocontrollers umgerechnet. Abschnitt 5.4 behandelt die benötigten Bufferwerte für die Dreh- und Neigungsachse, um ein „ruhiges“ Verhalten (Kapitel 2) der Antennennachführung zu garantieren. Abbildung 5.1 zeigt überblicksmäßig das Berechnungsschema zur Positionierung der Richtantenne.

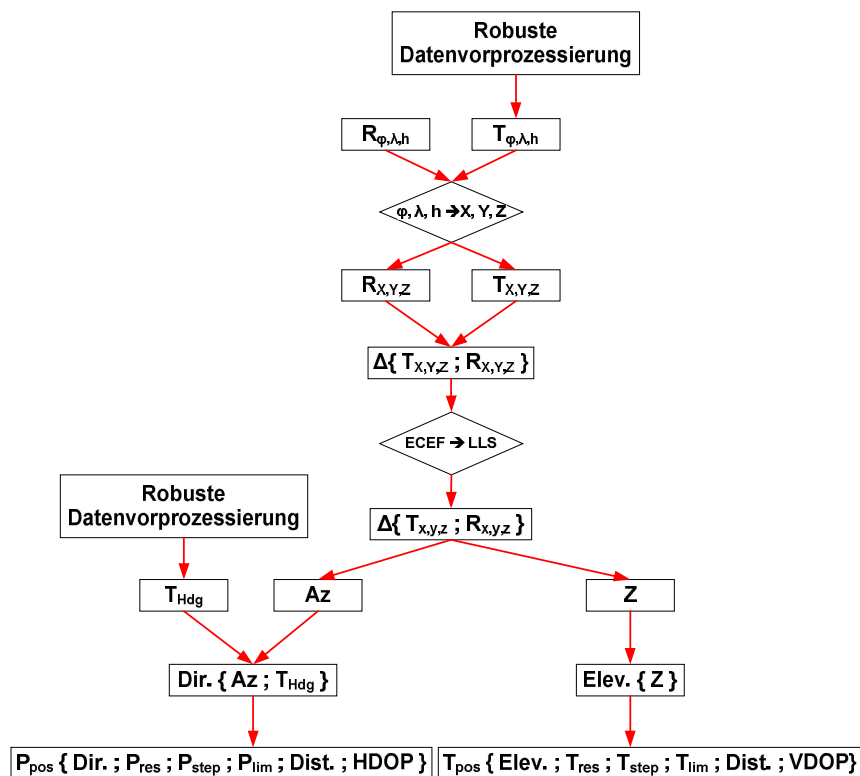


Abbildung 5.1: Antennennachführung Berechnungsschema

## 5.1 Datenaufbereitung

Der Abschnitt Datenaufbereitung behandelt die robuste Ausreißerdetektion sowie die folgende Datenaufbereitung. Der Algorithmus findet bei der Trackerpositionierung und Orientierung Verwendung. Die Position der Trackereinheit definiert dabei das lokale Koordinatensystem, das zur weiteren Berechnung der Ausrichtung der Dreh- bzw. Neigungsachse herangezogen wird. Das Basiskonzept der gesamten Antennenpositionierung wird in Kapitel 2 genauer erläutert.

Die robuste Datenaufbereitung geht dabei auf die in Friedrich Brunner & Andreas Wieser, 2009 beschriebene Methode der Ordnungsstatistik zurück. Dabei wird bei der Ausreißerdetektion nicht auf die  $3\sigma$ -Methode zurückgegriffen, sondern der Weg über die Berechnung des MAD gewählt, da dieser Algorithmus robuster gegenüber Ausreißern reagiert. Die Berechnung der Datenaufbereitung lässt sich in acht Schritten durchführen:

- Sortieren der Ausgangsdaten
- Bestimmung Median
- Berechnung Verbesserungen
- Sortieren der Verbesserungen
- Bestimmung Mean-absolute-deviation (MAD)
- Berechnung Standard deviation (STD)
- Berechnung obere/untere Grenze
- Ausreisser detektieren
- Berechnung Mittelwert
- Berechnung STD

Das Sortieren der Daten erfolgt in aufsteigender Reihenfolge, daher auch der Begriff der Ordnungsstatistik. Der Median wird auch der „mittlere Wert“ der Datenreihe genannt. Dieser Wert wird durch einen möglicherweise vorhandenen Ausreißer weniger beeinflusst als der herkömmliche Mittelwert und ist damit stabiler. Hierbei muss die auftretende Anzahl der Ausreißer in der Datenreihe geringer sein als die Hälfte der gesamten Beobachtungen (Formel: 5.1). Ansonsten können die Ausreißer nicht mehr von den Nutzdaten unterschieden werden. Da meist viele Messwerte zur Verfügung stehen, lässt sich diese Bedingung im Allgemeinen erfüllen.

$$Anzahl_{Ausreisser} < \frac{Anzahl_{Beobachtungen}}{2} \quad (5.1)$$

Der Median ist dabei definiert durch  $\tilde{y} = F^{-1}(0.5)$  was dem 0,5-Quantil entspricht. Eine einfachere Interpretation ist in Formel 5.2 dargestellt:

$$\tilde{y} = \begin{cases} y_{(n+1)/2}; & n = \text{ungerade} \\ 0.5 * (y_{n/2} + y_{n/2+1}); & n = \text{gerade} \end{cases} \quad (5.2)$$

Der MAD errechnet sich aus den geordneten Verbesserungen, die aus der Differenz des Medians und den Datenpunkten gebildet werden. (Formel: 5.3)

$$MAD = \text{median}[|\tilde{y} - y_n|]; \quad (5.3)$$

Die STD des MAD wird dabei wie folgt berechnet (Formel: 5.4):

$$s = 1/z_{1-0.5/2} * MAD; \quad \text{mit} \quad 1/z_{1-0.5/2} \approx 1,483 \quad (5.4)$$

Daraufhin werden mit der bekannten  $3\sigma$ -Methode die Unter- als auch Obergrenze der Messreihe festgelegt (Formel: 5.5).

$$\begin{aligned} y_{min} &= \tilde{y} - 3 * s \\ y_{max} &= \tilde{y} + 3 * s \end{aligned} \quad (5.5)$$

Alle Messwerte, welche sich nicht innerhalb dieser Grenzen befinden, müssen anschließend als Ausreißer eliminiert werden. Um schließlich einen stabilen Wert, sowie dessen Standardabweichung zu erhalten, wird das arithmetische Mittel gebildet. (Formeln: 5.6 und 5.7)

$$\bar{y} = \frac{1}{n} * \sum y_i \quad (5.6)$$

$$s^{[\circ]} = \sqrt{\frac{1}{n-1} * \sum y_i - \bar{y}} \quad (5.7)$$

Diese Methode der Datenaufbereitung wurde für die Festlegung der Position der Antennennachführung (Home) verwendet. Diese definiert in weiterer Folge das lokale Koordinatensystem. Da die Datenaufbereitung der GPS-Positionskomponenten ( $\varphi, \lambda$ ) bei der Bodenstation in ellipsoidischen Koordinaten (WGS84) erfolgt, wurde im weiteren die Standardabweichung der Grundgesamtheit  $\mathbf{s}$  in ein metrisches Maß umgewandelt. Dadurch kann auf die Qualität der von Ausreißer befreiten Position geschlossen werden (Formeln: 5.8 u. 5.9 nach [http://fmepedia.safe.com/articles/How\\_To/Calculating-accurate-length-in-meters-for-lat-long-coordinate-systems](http://fmepedia.safe.com/articles/How_To/Calculating-accurate-length-in-meters-for-lat-long-coordinate-systems)).



$$s_\varphi[m] = s_\varphi[rad] * (111132.92 - 559.82 * \cos(2 * \varphi_{ref}) + 1.175 * \cos(4 * \varphi_{ref})) \quad (5.8)$$

$$s_\lambda[m] = s_\lambda[rad] * (111412.84 * \cos \varphi_{ref} - 93.5 * \cos 3 * \varphi_{ref}) \quad (5.9)$$

Die absolute Punktlagegenauigkeit wird aus den DOP-Werten geschätzt (Formel: 5.10 nach Bernhard Hofmann-Wellenhof *et al.* (2008)). Der DOP-Wert beschreibt die Satellitengeometrie einer GPS-Positionslösung. Daher kann auf den Punktfehler geschlossen werden. Da angenommen werden kann, dass sich die Satellitengeometrie auf Seiten des Rovers sowie bei der Bodenstation aufgrund der relativ kurzen Distanz gleichmäßig verändert, muss er für die weiterführenden Berechnungen nur ein Mal angesetzt werden.

$$\sigma_{DOP}[m] = \sqrt{DOP^2 * 5.3^2} \quad (5.10)$$

Der konstante Multiplikator (5.3<sup>2</sup>) in Gleichung 5.10 stellt die Fehlereinflüsse entlang des Signalweges (Ionosphäre, Troposphäre, Relativistische Einflüsse,...) dar. Daher wird  $\sigma_{DOP}$  kontinuierlich aus den Roverdaten für die Horizontal- und Vertikalkomponente berechnet und anschließend als absolute Punktlageunsicherheit in die Abschätzung der Antennenbuffer (Abschnitt: 5.4) miteinbezogen.

## 5.2 Koordinatentransformationen

Wie bereits in der Einleitung erwähnt, geht die Positionierung der Antennen auf ein Basislinienverfahren der Rover- ↔ Antennenpositionen zurück. Dieser Abschnitt behandelt den Algorithmus zur Bestimmung der horizontalen (PAN, Azimut) und auch vertikalen (Tilt, Elevationswinkel) Ausrichtung der Antennen. Die Antennenposition und Orientierung definiert hierbei das lokale Koordinatensystem. Die im globalen WGS84 vorliegenden, aufbereiteten (Abschnitt: 5.1 (*Datenaufbereitung*)) Daten der Antennennachführung sowie die Echtzeitdaten des Rovers werden zur Azimuth- bzw. Zenitwinkelberechnung in das lokale Koordinatensystem transformiert (Abbildung: 5.1). Der Ursprung dieses Koordinatensystems liegt in der Antennennachführungsposition. Die anschließend folgenden Formeln zeigen die Schrittweise Berechnung des Azimuths bzw. des Zenitwinkels. Zu beachten ist,  $\mathbf{i}$  stellt die Position der Antennennachführung und  $\mathbf{j}$  die Position des Rovers dar.

**1.Schritt:** Umrechnung der ellipsoidischen Koordinaten in kartesische Koordinaten:

$$\begin{aligned} X &= (N + h) \cos \varphi \cos \lambda \\ Y &= (N + h) \cos \varphi \sin \lambda \\ Z &= \left( \frac{b^2}{a^2} N + h \right) \sin \varphi \end{aligned} \quad (5.11)$$

mit dem Normalkrümmungsradius:

$$N = \frac{a^2}{\sqrt{a^2 \cos^2 \varphi + b^2 \sin^2 \varphi}} \quad (5.12)$$

**2.Schritt:** Berechnung des normierten Differenzvektors (Antennennachführung, Rover):

$$s_{ij} = \sqrt{(X_j - X_i)^2 + (Y_j - Y_i)^2 + (Z_j - Z_i)^2} \quad (5.13)$$

$$\begin{aligned} \|\Delta X_{ij}\| &= \frac{X_j - X_i}{s_{ij}} \\ \|\Delta Y_{ij}\| &= \frac{Y_j - Y_i}{s_{ij}} \\ \|\Delta Z_{ij}\| &= \frac{Z_j - Z_i}{s_{ij}} \end{aligned} \quad (5.14)$$

**3.Schritt:** Aufstellen der Rotationsmatrix (Lokal  $\Rightarrow$  Global):

$$\mathbf{R}_{ij} = \begin{bmatrix} -\sin \varphi_i \cos \lambda_i & -\sin \lambda_i & \cos \varphi_i \cos \lambda_i \\ -\sin \varphi_i \sin \lambda_i & \cos \lambda_i & \cos \varphi_i \sin \lambda_i \\ \cos \varphi_i & 0 & \sin \varphi_i \end{bmatrix} \quad (5.15)$$

**4.Schritt:** Berechnung des lokalen Basislinienvektors (Matrizenschreibweise):

$$\mathbf{x}_{ij}^{Lokal} = \mathbf{R}_{ij}^T * \|\Delta \mathbf{X}_{ij}\| \quad (5.16)$$

**Alternativ zu Schritt 3-4:** Berechnung des lokalen Basislinienvektors (Gleichungssystem):

$$\begin{aligned} x_{ij}^{Lokal} &= -\sin \varphi_i * \cos \lambda_i * \|\Delta X_{ij}\| - \sin \varphi_i \sin \lambda_i * \|\Delta Y_{ij}\| + \cos \varphi_i * \|\Delta Z_{ij}\| \\ y_{ij}^{Lokal} &= -\sin \lambda_i * \|\Delta X_{ij}\| + \cos \lambda_i * \|\Delta Y_{ij}\| \\ z_{ij}^{Lokal} &= \cos \varphi_i \cos \lambda_i * \|\Delta X_{ij}\| + \cos \varphi_i \sin \lambda_i * \|\Delta Y_{ij}\| + \sin \varphi_i * \|\Delta Z_{ij}\| \end{aligned} \quad (5.17)$$

**5a.Schritt:** Berechnung Azimuth und Zenitdistanz:

$$\alpha_{ij} = \arctan \frac{y_{ij}^{Lokal}}{x_{ij}^{Lokal}}$$

$$\zeta_{ij} = \arctan \frac{\sqrt{(x_{ij}^{Lokal})^2 + (y_{ij}^{Lokal})^2}}{z_{ij}^{Lokal}} \quad (5.18)$$

### 5.3 Absolute Servoposition

Bei der Bestimmung der absoluten Servoposition werden die in 5.2 berechneten Azimuth- und Zenitwinkel in Servolaufzeiten bzw. Servoschritte  $\mathbf{S}_{step}$  umgerechnet. Um diese für die Drehachse berechnen zu können ist es nötig, die Orientierung (Heading) der Antennennachführung zu kennen (Formel: 5.19). Die Orientierung wird mit einem Magnetfeldkompass bestimmt, der im Kapitel 4.3 genauer beschrieben ist. Bei der Bestimmung der Servoschritte für die Neigungsachse wird lediglich der Komplementärwinkel des Zenitwinkels (Elevationswinkel) verwendet. Wobei hier die vom Kapitel 3.4.1 angedeutete Auflösung der Dreh- bzw. Neigungsservoantriebe zum tragen kommt.

$$Pan_{Step} = \left\lceil \frac{\alpha_{ij} - Hdgi}{Pan_{Res}} \right\rceil \quad (5.19)$$

$$Tilt_{Step} = \left\lceil \frac{90 - \zeta_{ij}}{Tilt_{Res}} \right\rceil \quad (5.20)$$

Bei der Berechnung der Drehachsen- und der Neigungsachsen- Servoschritte ist zusätzlich zu beachten, dass eine, wie in Kapitel 4.2 beschriebene, Richtfunkantenne (Patchantenne) Verwendung findet. Üblicherweise besitzen diese Antennen vier Sektionen, die zur Datenübertragung geeignet sind (siehe Polarplot). Da versucht werden sollte, immer im Bereich der Main-Front-Loop zu verbleiben, ist bei der Implementierung darauf zu achten. Weiter muss auf die maximalen Ausschläge der Antennennachführung geachtet werden, da angesichts der Leitungsführungen sowie der verwendeten Servomotoren (Kapitel: 3.4.1) keine „Endlos-Drehungen“ möglich sind. In diesem Projekt wurden die Maximalausschläge der Drehachse auf  $\pm 180^\circ$  bzw. für die Neigungsachse auf  $-30^\circ \leq \leq +60^\circ$  festgelegt. Aus diesem Grund muss versucht werden, dass sich der Rover innerhalb diesen Bereiches bewegt. Sollte dieser sich außerhalb des Bereiches befinden, wird die „schwächere“ Main-Back-Loob verwendet.

## 5.4 Antennenbuffer

Der Antennenbuffer wurde deshalb eingeführt, um ein möglichst ruhiges Nachführungsverhalten der Antenne zu gewährleisten. Er beschreibt einen Kegel, ausgehend von der Richtantenne, innerhalb dessen sich der Rover befinden darf, ohne dass eine Nachführung stattfindet. Dieser Bufferkegel wird definiert durch die Distanz  $s_{ij}$  zum Ziel, sowie durch die in Formel 5.10 berechneten, absoluten Punktlageunsicherheiten  $\sigma_{DOP}$ .

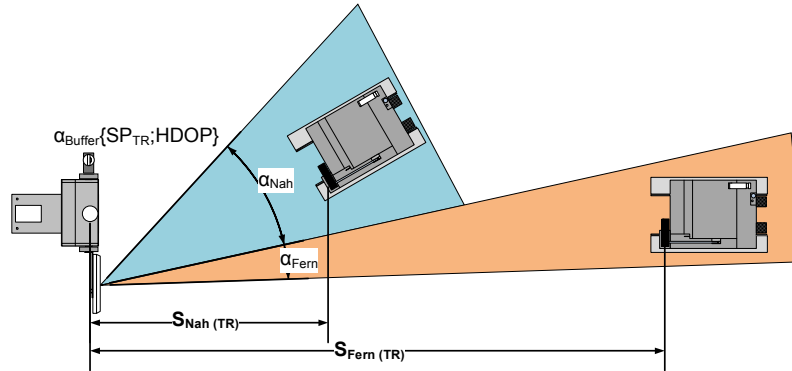


Abbildung 5.2: Buffer der Drehachse

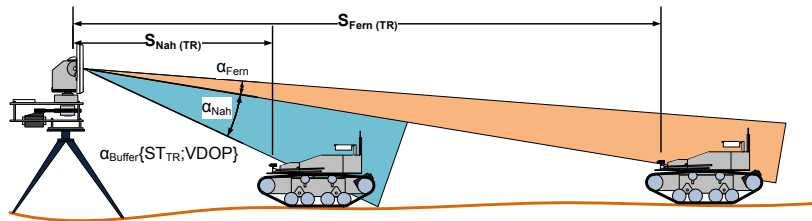


Abbildung 5.3: Buffer der Neigungsachse

Die Formeln 5.21 und 5.22 stellen die Berechnung des Bufferbereichs für die Dreh- als auch für die Neigungsachse dar. Sowohl die in Abschnitt 5.1 berechneten Standardabweichungen als auch die in Abschnitt 5.2 berechneten Distanzen, werden bei der Bufferberechnung miteinbezogen. Die Servoschritte müssen als Integerwerte vorliegen und da das Ergebnis aber für gewöhnlich eine Gleitkommazahl darstellt, muss diese zusätzlich abgerundet werden.

$$Pan_{Buffer} = \left[ \frac{\arctan \left( \frac{\sqrt{HDOP_j^2 * 5,3^2 + \sigma_{TH}^2}}{s_{ij}} \right)}{Pan_{Res}} \right] \quad (5.21)$$

$$Tilt_{Buffer} = \left[ \frac{\arctan \left( \frac{\sqrt{VDOP_j^2 * 5,3^2 + \sigma_{TV}^2}}{s_{ij}} \right)}{Tilt_{Res}} \right] \quad (5.22)$$

Aufgrund der verwendeten Low-Cost GPS-Empfänger (Kapitel: 4.3), kann dieser Kegel in horizontale sowie vertikale Richtung große Ausmaße annehmen (Abbildung: 5.4 und 5.5). Da die verwendete Antenne (Kapitel: 4.2) einen entsprechend großen Öffnungswinkel der Forward-Main-Loop besitzt und die Sendeleistung mit abnehmender Distanz zunimmt, werden die großen Bufferwerte in weiterer Folge als nicht kritisch erachtet.

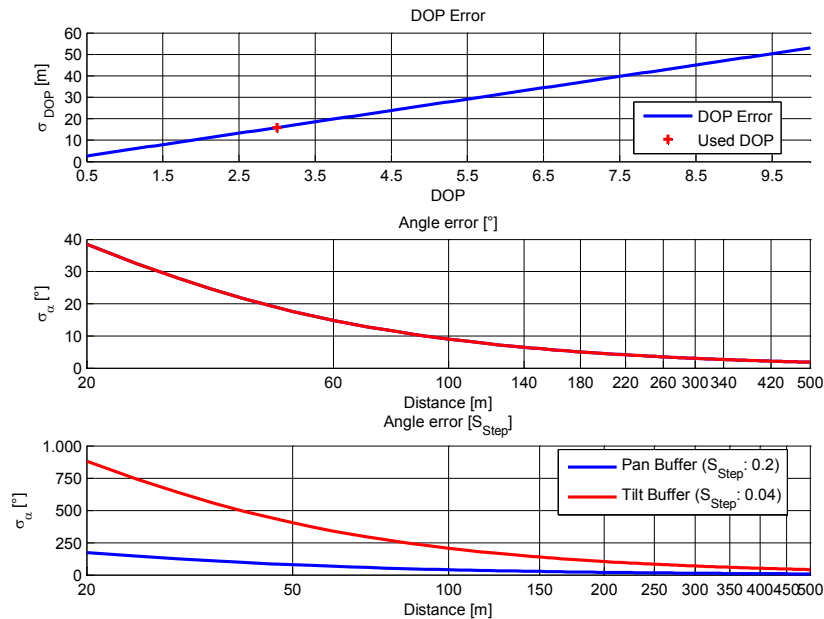


Abbildung 5.4: Buffer in Abhängigkeit: Distanz, DOP, Servoaufösung: a) DOP in Meter; b) Bufferbereich in Grad; c) Bufferbereich in abs. Servoschritten

Abbildung 5.4b zeigt eine Abschätzung der zu erwartenden Bufferwerte bei einem DOP-Wert von 3 in der Einheit Grad. Es ist zu erkennen, dass sich der Bufferbereich bis zu einer Distanz von ca. 30 m, außerhalb des Öffnungswinkels der Richtantenne (ca.  $\pm 25^\circ$ ) befindet. Da es sich allerdings um eine sehr geringe Distanz handelt und dieser Bereich zusätzlich in eine der beiden Forward-Side-Loops fällt, sind hier keine Übertragungsfehler zu erwarten. Da eine Standardantenne normalerweise bei freier Sicht eine Reichweite bis zu 300 m hat, muss bei einer Distanz von 100 m keine Beeinflussung der Datenübertragungsqualität durch eine Antennenfehlausrichtung befürchtet werden. Es ist jedoch darauf zu achten, dass die Antenne nicht „quer“ zum Rover ausgerichtet ist, da es sich dabei um den „toten“ Winkel handelt. Jedoch müssen durch Versuche genaue Werte zur maximalen Antennenfehlausrichtung festgelegt werden. Abbildung 5.4c zeigt die zuvor beschriebenen Bufferwerte in den jeweiligen absoluten Servopositionen.

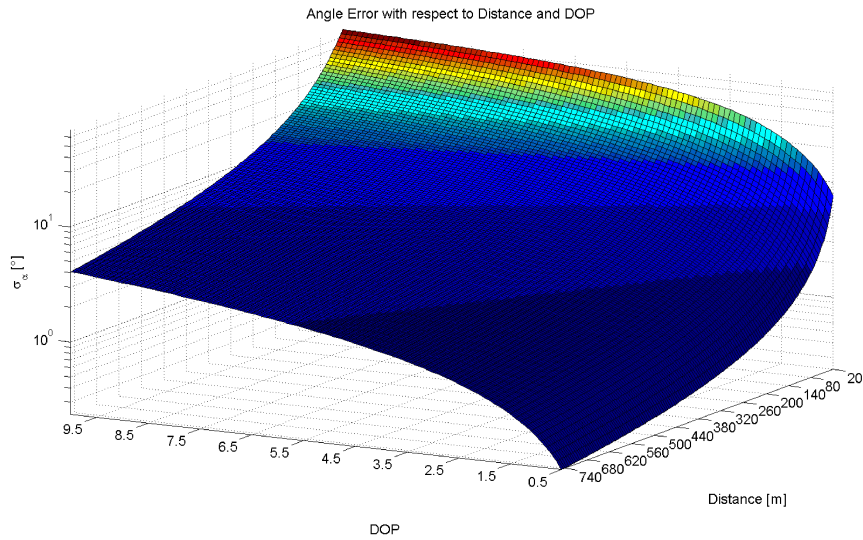


Abbildung 5.5: 3D Winkelfehler: Distanz, DOP

Abbildung 5.5 zeigt die in Abbildung 5.4a und b gezeigten Fehlereinflüsse in einer dreidimensionalen Ansicht. Es muss jedoch festgehalten werden, dass Positionslösungen mit einem DOP-Wert von über 3 verworfen werden sollen.

## 5.5 Abschätzungen

Dieser Abschnitt behandelt Abschätzungen bezüglich der Konstruktion, dem Bau und der Berechnungen.

### Antrieb Antennennachführung

Wie bereits mehrfach beschrieben, handelt es sich bei den Antriebskomponenten der Antennennachführung um handelsübliche Modellbauservos. Beim Antrieb der Drehachse handelt es sich um ein sogenanntes Segelwindenservo, welches mehrere Umdrehungen zulässt. Das Drehachsenservo besitzt einen maximalen Drehbereich von  $2520^\circ$ , die Drehbewegung sollte jedoch auf  $180^\circ$  beschränkt sein. Zum einen muss eine Übersetzung verwendet werden, und zum anderen eine Softwareeinschränkung implementiert werden. Aufgrund der Konstruktion sowie der Verfügbarkeit der Bauteile konnte keine Übersetzung mit einem Verhältnis von  $1 : 14$  realisiert werden. Das Antriebsrad musste konstruktionsbedingt einen Mindestdurchmesser von  $24 \text{ mm}$  aufweisen. Wird diese Einschränkung berücksichtigt, wird für das zuvor genannte Verhältnis ein Abtriebsraddurchmesser von  $\approx 330 \text{ mm}$  benötigt. Ein Zahnriemenrad mit einem solchen Durchmesser ist nur schwer erhältlich oder müsste extra angefertigt werden. Für dieses Projekt wurde wie in Kapitel 4.3 beschrieben, eine Übersetzung von  $\approx 1 : 3.43$  verwendet, wobei der Drehbereich durch die Software weiter eingeschränkt wird. Die Servos der Neigungsachse weisen einen Standarddrehwinkel von  $\pm 45^\circ$ . Es wurde hier keine Übersetzung auf  $\pm 90^\circ$  angestrebt. Der Neigungsachsenantrieb wurde so eingestellt, dass sich ein Endausschlag von  $+60^\circ$  und  $-30^\circ$  ergibt. Dieser Ausschlag ist nach ersten Überlegungen ausreichend, da es sich gegenwärtig um einen bodengestützten Rover handelt.

### Auflösung Antennennachführung

Bei der Abschätzung der Auflösung für die Dreh- und Neigungsachse werden die technischen Daten der jeweiligen Servos sowie des Servocontrollers herangezogen. Tabelle 5.1 zeigt die verschiedenen verfügbaren Datensätze für die benötigten Antriebskomponenten der Antennennachführung. Die Angaben variieren je nach Hersteller. Dies wirkt sich insofern auf die Abschätzungen aus, da keine einheitliche Umrechnung für die Dreh- und Neigungsachse verwendet werden kann.

Formel 5.23 beschreibt die Abschätzung der Drehachsenauflösung. Die Übersetzung wurde bereits eingerechnet um auf die tatsächliche Auflösung zu kommen. Bei dieser Berechnung fließen die internen Servocontrollerschritte ein.

$$d_{pan} = \frac{\alpha_{rs10}}{R_{step} * i_{rs10}} = \frac{7 * 360^\circ}{(500 - 5500) * 3.428} \approx 0,15^\circ \quad (5.23)$$

Formel 5.24 beschreibt die Abschätzung der Neigungsachsenauflösung. Im Vergleich zur

Tabelle 5.1: Servocontroller Parameter

Servocontroller		
PWM-Auflösung	$d_{sc}$	0.5 $\mu sec$
PWM-Spannweite	$R_{sc}$	0.25 – 2.75 $ms$
Schritt-Spannweite	$R_{step}$	500-5500
Drehachsenantrieb (RS10)		
Arbeitsbereich	$\alpha_{rs10}$	7 * 360°
Untersetzung	$i_{rs10}$	1 : 3.428
Neigungsachsenantrieb (HS-645)		
Arbeitsbereich	$\alpha_{hs645}$	±45°
PWM-Spannweite	$R_{hs645}$	400 $\mu sec$ für 45°

Drehachsenauflösung werden hierfür nicht die internen Servocontrollerschritte, sondern die Puls-Weiten-Auflösung zur Berechnung herangezogen. Die Angabe der PWM-Spannweite stellt für Servos die Standardwerte dar.

$$d_{tilt} = \frac{\alpha_{hs645} * 2}{(R_{hs645}/d_{sc} * 2)} = \frac{45^\circ * 2}{(400/0.5) * 2} \approx 0,06^\circ \quad (5.24)$$

Wie bereits in Kapitel 3.4.1 beschrieben sind die berechneten Werte aus 5.23 und 5.24 bloß Richtwerte. Die tatsächliche Auflösung wurde erst durch Testläufe der Nachführung festgestellt.

### Untersetzung Rover

Die Untersetzungsbestimmung wurde erst nach der Motorauswahl durchgeführt. Diese Untersetzung stellt dabei lediglich ein Beispiel dar und kann innerhalb eines gewissen Rahmens angepasst werden. Der Rover wurde aus Sicherheitsgründen langsam ausgelegt. Tabelle 5.2 listet die Rahmendaten für die Bestimmung dieser Untersetzung auf.

Tabelle 5.2: Rover Parameter

Antriebsrad	$d$	76 $mm$
Motordrehzahl	$n$	200 $U/min$
Endgeschwindigkeit	$v$	≈ 5 – 7 $km/h$

$$\omega = \frac{2 * \pi * n}{60} = \frac{2 * \pi * 200}{60} \approx 21 \left[ \frac{rad}{sec} \right] \quad (5.25)$$

$$v = \omega * \frac{d}{2} * i \quad (5.26)$$

$$i = \frac{v}{\omega * (d/2)} = \frac{5}{21 * 0.038 * 3.6} \approx 1.74 \quad (5.27)$$

Mit einer Untersetzung aus Formel 5.27 sollte der Rover eine Geschwindigkeit von ≈ 5  $\frac{km}{h}$  erreichen. Hierbei wurden hierbei jedoch keine Verluste oder Lastmomente miteinkalkuliert. Die



Formeln 5.25 bis 5.27 wurden nur als Referenzwerte herangezogen. Die tatsächliche Umsetzung der Antriebseinheit ist in Kapitel 4.1 genauer erläutert.

### Akkudimensionierung Rover

Wie auch bei den Abschätzungen zum Antrieb wurden die folgenden Werte als Referenzwerte herangezogen und stellen keine exakte Kalkulation dar. Die Vorgaben zur Akkukapazitätsberechnung sind in Tabelle 5.3 zu sehen. Der mittlere Stromverbrauch ist dabei eine Abschätzung des Verbrauchs aller Komponenten, ausgenommen der Motoren. Bei den Motoren wurde eine relativ hohe Strombelastung angesetzt, um genügend Reserven vorhanden zu haben.

Tabelle 5.3: Rover Vorgaben

Spannung	U	24 V
Motorleistung	P	2*250 W
mittl. Stromverbrauch Elektronik	I	≈ 5 A
Laufzeit	t	1 h

$$I_{Antrieb} = \frac{P}{U} = \frac{2 * 250}{24} = 20 \text{ A} \quad (5.28)$$

$$I_{ges} = \sum I_{Verbraucher} \approx 25 \text{ A} \quad (5.29)$$

$$C = I_{ges} * t = 25 * 1 \approx 25 \text{ Ah} \quad (5.30)$$

Die Formeln 5.28 bis 5.30 zeigen die benötigte Akkukapazität, welche zur eigentlichen Akkudimensionierung verwendet wurde. Bei der endgültigen Auswahl der Akkus musste zusätzlich auf die Bauform und die Baugrößen geachtet werden. Auch der Kostenfaktor spielt hier eine wesentliche Rolle.

### Widerstandsabschätzung LED-Anzeigen

Beim gesamten Projekt wurden verschiedene LED's für die Anzeigen verwendet. Da LED's nur einen geringen Toleranzbereich in Bezug auf die Versorgungsspannung (nach Erwin Böhmer, 1998) haben, mussten diese mit einem Spannungsteiler ausgestattet werden. Der Vorwiderstand begrenzt die Spannung auf ein für die LED's akzeptables Maß. Die Berechnung dieser Vorwiderstände erfolgte mit Formel 5.31.

$$R_{Vorwiderstand} = \frac{U_{Betriebsspannung} - U_{LED}}{I_{LED}} \quad (5.31)$$

Tabelle 5.4 zeigt die Vorwiderstände für die verschiedenen Betriebsspannungen und LED's.

Tabelle 5.4: LED-Vorwiderstandswerte ( $I_{LED} \approx 20 \text{ mA}$ )

Farbe	Vorwiderstand nach E-Reihe		
	5 V	12 V	24 V
Rot (1.6 V, 20 mA)	$\approx 220 \Omega$	$\approx 560 \Omega$	$\approx 1.2 \text{ k}\Omega$
Grün (2.1 V, 20 mA)	$\approx 150 \Omega$	$\approx 560 \Omega$	$\approx 1.2 \text{ k}\Omega$

Diese Werte stimmen insofern nicht genau mit den in Formel 5.31 berechneten Werten überein, da sie bereits auf die nächsten verfügbaren Werte (nach: (Klaus Beuth, 2003)) bezogen wurden.

### Widerstandsabschätzung Spannungsregler LM317

Für die Spannungsversorgung der Elektronik wurden verschiedene Spannungsregler angefertigt. Bei der Basisschaltung des LM317 Spannungsreglers kann die Ausgangsspannung mit dem Widerstandswert am ADJ-Pin festgelegt werden. Formel 5.32 zeigt die Berechnung der Ausgangsspannung mit einem Korrekturterm ([http://www.acoustica.org.uk/t/3pin\\_reg\\_notes1.html](http://www.acoustica.org.uk/t/3pin_reg_notes1.html)).

$$V_{out} = 1.25 * \left(1 + \frac{R2}{R1}\right) + I_{ADJ} * R2 \quad (5.32)$$

Der Term  $I_{ADJ} * R2$  stellt einen Korrekturterm für den internen Reglerstromverbrauch dar und hat eine Größenordnung von  $\approx 100 \mu\text{A}$ . Daher kann dieser Korrekturterm für die Abschätzung vernachlässigt werden. Nach dem Auflösen der Formel nach  $R2$  erhält man einen guten Näherungswert für den einstellbaren Widerstand (Formel: 5.33).

$$R2 \approx R1 * \left(\frac{V_{out}}{1.25} - 1\right) \quad (5.33)$$

# Kapitel 6

## Software und Implementierung

### 6.1 Externe Software

Dieser Abschnitt beschreibt die unterschiedlichen Zusatzprogramme, die für die Programmierung des Autopiloten, die Visualisierung der Roverdaten, die verschiedenen Kommunikationsmodule und für die Steuerung des Rovers verwendet werden. Diese Softwarepakete sind frei erhältlich.

#### 6.1.1 Autopilot

Der APM-Autopilot ist mit einem Atmega2560 als Hauptprozessor bestückt. Dieser Prozessor verwendet einen Arduino Bootloader und kann in C/C++ programmiert werden. Durch die Vielseitigkeit der Elektronik sind verschiedenste Firmware-Versionen für den Autopiloten vorhanden. Die effektivste Möglichkeit einer Firmwareänderung ist die Verwendung der „Groundcontrol“-Software APM-Missionplanner. Diese Software hat einen Firmware-Uploader implementiert und verschiedene Softwareversionen gespeichert. Darunter sind Firmware-Versionen für Flächenflugzeuge, Hubschrauber, Mehrrotorsysteme und Rover gespeichert (Abbildung 6.1). Mit dieser Software können auch die einzelnen Sensoren unabhängig voneinander getestet werden.

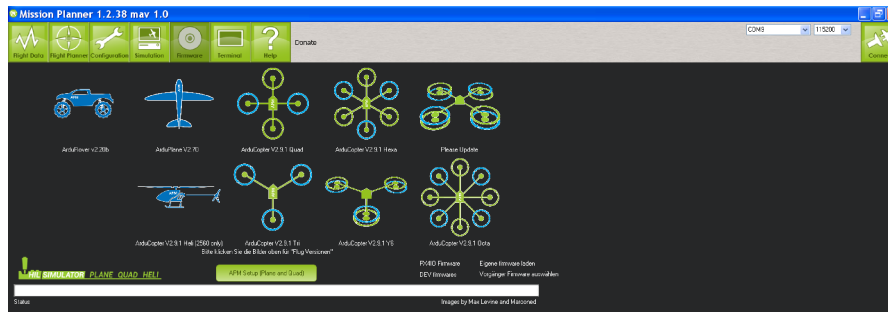


Abbildung 6.1: APM Planner Mav1.0

Für umfangreichere Änderungen an der Firmware ist die „Arduino“-Software als Programmierumgebung durchaus empfehlenswert. Wie schon wie der „Groundcontrol“-Software APM-Missionplanner ist auch diese Software ist mit einem kompatiblen Bootloader ausgestattet und kann aber zusätzlich mit verschiedensten Prozessoren verwendet werden. Gängige Firmware-Versionen sind frei als „Source Code“ erhältlich und die Nutzer können diese je nach Anforderung anpassen bzw. entsprechend umprogrammieren.

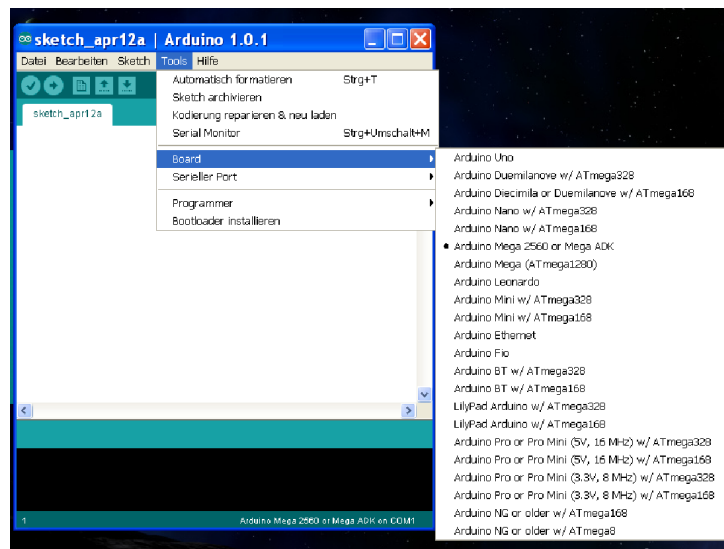


Abbildung 6.2: Kompatible Prozessoren für Arduino Software

Die Basisstruktur eines Arduino-Programms besteht mindestens aus zwei Methoden, einer „void setup()“-Methode und einer „void loop()“-Methode. Die „Setup“-Methode wird nur beim Einschalten bzw. bei einem Neustart des Prozessors aufgerufen und initialisiert dabei Variablen und Bibliotheken. In der „Loop“-Methode wird das Hauptprogramm in einer Endlosschleife ausgeführt (Arduino Homepage: <http://arduino.cc/en/Guide/HomePage>). Um Sensoren oder Aktoren einzubinden, werden meist zusätzliche Header-Bibliotheken bereitgestellt, die in das Hauptprogramm implementiert werden.

Möchte der Nutzer nur die Parameteranpassungen einer bestehenden Autopilotfirmware

ändern, ist hier die Verwendung der APM-„Groundcontrol“-Software oder der „QGroundControl“-Software empfehlenswert. Mit beiden Softwarepaketen können die voreingestellten Parameter aus dem EEPROM ausgelesen, verändert und geschrieben werden. Auch die Echtzeitvisualisierung und Datenspeicherung ist mit beiden Programmen möglich.



Abbildung 6.3: QGroundControl

Abbildung 6.3 zeigt die verwendete Visualisierungssoftware „QGroundControl“ in Verbindung mit dem Autopiloten. Mit diesem Programm können die vom Rover eingehenden Daten sowohl in einem Binärformat als auch in einem ASCII-Format gespeichert werden.

### 6.1.2 Zusatzsoftware Kommunikation

Wie in Kapitel 4.4.2 beschrieben, erfolgt die Übertragung der Roverdaten mit einer WLAN-Verbindung und dem „XPort“-Modul als Autopilotschnittstelle. Das Modul ist ein „embedded“ System, welches einerseits eine vollständige Netzwerkanbindung erlaubt und andererseits über eine TTL-Schnittstelle verfügt. Der integrierte Webserver ermöglicht eine einfache Konfiguration des Moduls (Abbildung: 6.4) über einen Web-Browser nach Wahl. Die „CBR“-Software und der „DeviceInstaller“ werden benötigt, um am Computer ein virtuelles COM-Port anzulegen und zu konfigurieren (Abbildung: 6.4a und c).

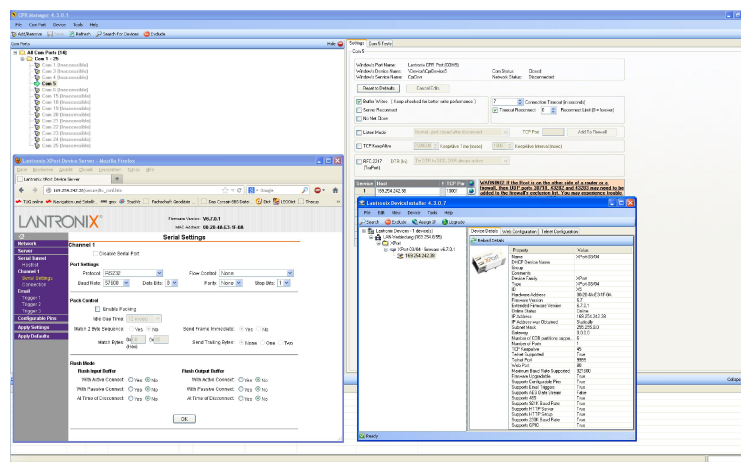


Abbildung 6.4: a) CPR-Manager; b) XPort-Device Server; c) Lantronix DeviceInstaller

Der übertragene Roverdatenstream muss aufgrund der beiden unabhängigen Softwarepakete zur Visualisierung und zur Berechnung der Antennenparameter aufgeteilt werden. Hierfür wird die Software Virtual Serial Ports Emulator (VSPE) verwendet (Abbildung: 6.5b).

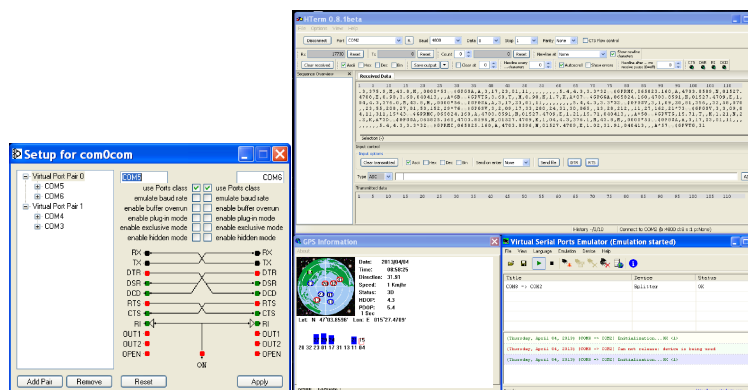


Abbildung 6.5: a) „com0com“-Emulator; b) COM-Port Splitter (VSPE)

In Abbildung 6.5.b) wird eine Beispielanwendung des COM-Port Splitters dargestellt. Das Programm zeichnet die eingehenden Daten des GPS-Empfängers auf und „streamt“ sie auf ein offenes Port. Deshalb können mehrere Programme auf ein und denselben Port zugreifen. Abbildung 6.5.a) zeigt eine weitere Software, die auf dem Computer virtuelle COM-Ports emuliert. Diese Ports können so konfiguriert werden, dass eine Nullmodem-Verbindung innerhalb des Computers aufgebaut wird, ohne dass zwei physikalische RS-232-Schnittstellen zur Verfügung stehen müssen. Dieses Programm wurde in erster Linie für Tests an der Kommunikationsbibliothek (Abschnitt: 6.2.2) verwendet.

### 6.1.3 Joystick-Steuerung

Die Steuerung der RC-Komponenten des Rovers erfolgt mit der Software von „Flytron“. Die Daten des Joysticks werden in die Software eingelesen, konvertiert und an den PPM-Signalgenerator gesendet.

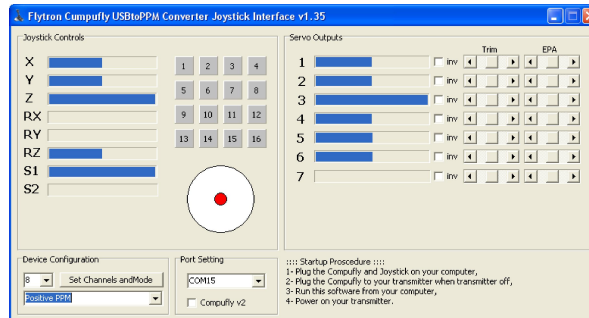


Abbildung 6.6: CompuFly von Flytron

Das Programm ermöglicht es, alle empfangerspezifischen Einstellungen wie die Anzahl der Kanäle, das Invertieren und Trimmen der einzelnen Kanäle und die Art der Signalgenerierung (positives- oder negatives- PPM-Frame) vorzunehmen. Dadurch ist die Hardware, in Verbindung mit dieser Software, für eine große Anzahl verschiedener Empfänger geeignet.

## 6.2 C/C++ Tracking-Software

Bei der Softwareimplementierung zur Steuerung der Antennennachführung wurde, das in Kapitel 2.2 beschriebene Konzept, umgesetzt. Zusätzlich wurden weitere Anforderungen an die Software gestellt. Diese beziehen sich nicht auf die einzelnen Berechnungs- und Steueraufgaben der Bibliotheken, sondern an die softwaretechnische Implementierung. Zu diesen Anforderungen zählen:

- Fehlerbehandlung
- Multithread tauglichkeit
- Dokumentation

Diese drei Forderungen stellten hauptsächlich für die Adaption des Open Source Codes eine Herausforderung dar. Diese mussten nämlich teilweise an die gewünschten Anforderungen angepasst werden. Die erstellten Dokumentationen sind entweder auf der beiliegenden Daten-CD zu finden oder es wird auf die entsprechende Onlinedokumentation verwiesen. Diese Dokumentationen enthalten neben den einzelnen Funktionsbeschreibungen auch Testimplementierungen, um eine alternative Verwendung möglichst einfach zu gestalten.

Bei der Programmierung wurden zusätzlich auf drei freie Bibliothekssammlungen zurückgegriffen. Diese Kernbibliotheken wurden für das Multithreading, die Fehlerbehandlung, die statistischen Berechnungen und die Visualisierung der Ergebnisse verwendet. Folgende Kernbibliotheken wurden eingebunden:

- Boost-library
- GNU-Scientific Library (GSL)
- Visualization Toolkit (VTK)

Die „Boost“-Bibliothek dient als Basis für die Kommunikation, die Fehlerbehandlung und das Multithreading. Eine detaillierte Beschreibung bietet die Aufzeichnungen unter <http://www.boost.org/doc/>. Die Verwendungsmöglichkeiten und auch Beispielimplementierungen werden unter <http://www.highscore.de/cpp/boost/> ausführlich behandelt.

Bei der „GNU-Scientific-Library“ handelt es sich um eine mathematische Bibliothek für numerische Berechnungen. Sie beinhaltet neben den verwendeten statistischen Berechnungen viele weitere Berechnungsalgorithmen. Auch hier wird auf die Onlinedokumentation unter <http://www.gnu.org/software/gsl/> bzw. [http://www.gnu.org/software/gsl/manual/html\\_node/](http://www.gnu.org/software/gsl/manual/html_node/) zur Implementierung verwiesen.



Das „Visualization Toolkit“ wird im Projekt für die graphischen Darstellungen zur robusten Datenaufbereitung verwendet. Neben der ausführlichen Onlinedokumentation (<http://www.vtk.org/doc/release/5.10/html/>) wird auch eine große Beispielsammlung (<http://www.vtk.org/Wiki/VTK/Examples/Cxx>) angeboten.

Auf diese Bibliothekssammlungen wird in weiterer Folge nicht mehr genauer eingegangen.

### 6.2.1 Softwarestruktur Antennennachführung

Bei der Programmierung der Trackersoftware wurde darauf geachtet, die einzelnen Programmteile und Bibliotheken möglichst gekapselt zu erstellen. Auch wenn der Programmieraufwand dabei um vieles größer war, stand doch die Wiederverwendbarkeit einzelner Programmteile im Vordergrund. Die Softwarestruktur wurde in untergeordnete Levels (Abbildung 6.7 linker Balken) und den übergeordneten Layers (Abbildung 6.7 rechter Balken) geteilt. Die Applikation und die statischen Bibliotheken sind in Abbildung 6.7 in der mittleren Blockstruktur dargestellt.

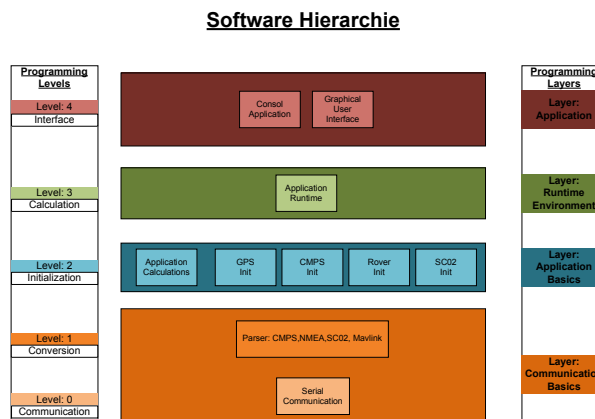


Abbildung 6.7: Software Hierarchiestufen

Tabelle 6.1: Software Hierarchie Beschreibung

Level	Layer	Beschreibung
Communication	Communication Basics	Aufbauen und Verwalten der RS232-Kommunikationen
Conversion	Communication Basics	Konvertierung der Eingangsdaten in ein einheitliches Format
Initialization	Application Basics	Initialisieren der Sensoren/Aktoren Ausreisserdetektion und Visualisierung
Calculation	Runtime Environment	Echtzeit Berechnung der Antennenparameter und Steuerung
Interface	Application	Nutzeroberfläche

Die Levels des ersten Layers „Communication Basics“ werden in weiterer Folge unter eine LGPL-Lizenz gestellt und sind somit zur öffentlichen Weiterverwendung freigegeben. Diese Bibliotheken bestehen teilweise aus externem Open Source Code mit der oben genannten Lizenz.

### 6.2.2 Communication Basics

Der Layer „Communication Basics“ beinhaltet alle kommunikationsspezifischen Routinen und greift dabei auf die Boost-Kernbibliothek zu. Diese Programmteile ermöglichen den Aufbau und das Verwalten von RS232-Verbindungen und das Konvertieren (parsen) der ein- bzw. ausgehenden Daten in ein geeignetes Format. Für diesen Layer wurde auch auf frei verfügbare Source Codes zurückgegriffen und diese ins Gesamtprojekt eingebunden. Teilweise aber waren diese freien Source Codes nicht mit dem verwendeten Betriebssystem kompatibel, dennoch konnten einzelne Basisroutinen davon verwendet werden.

#### Kommunikation

Die im Projekt am häufigsten verwendete Kommunikationsmethode greift auf die RS232 Schnittstelle zu. Daher wurde auf eine einfach zu implementierende Bibliothek mit möglichst flexibler Anwendung Wert gelegt. Eingesetzt wurde hier die auf die „Boost“-Bibliothek aufbauende Wrapper-Bibliothek, siehe unter: [http://www.webalice.it/fede.tft/serial\\_port/serial\\_port.html](http://www.webalice.it/fede.tft/serial_port/serial_port.html). Diese Wrapper-Bibliothek ist für ein Linux-System programmiert, wurde aber für dieses Projekt an das verwendete Windows-System angepasst.

Die Bibliothek beinhaltet vier verschiedene serielle Kommunikationsmethoden wovon in diesem Projekt die „Timeout-Serial“-Verbindung verwendet wird. Diese Verbindungart bricht bei einem auftretenden Lese- oder Schreibfehler oder einer plötzlichen Unterbrechung der Kommunikation die Verbindung ab, ohne dass es zu einem sogenannten „Dead-Lock“ oder „Blocking“ kommt (Narain Gehani & William D. Roome, 1993). Als ein „Dead-Lock“ wird ein Prozess beschrieben, der in eine Endlosschleife übergeht und nicht vom Nutzer abgebrochen werden kann. Dadurch werden unnötig viele Ressourcen eines Computers oder Mikrocontrollers belegt und ein Programmabsturz ist möglich. Hingegen wird beim „Blocking“ ein Programmzustand erreicht, der das Programm während der Laufzeit blockiert. Dieses „Blocking“ kann in der Kommunikation mit externen Geräten auftreten, wenn keine softwaretechnischen Routinen implementiert sind, die einen solchen Zustand abbrechen.

Weitere Vorteile dieser Kommunikationsbibliothek sind eine durchgängige Fehlerbehandlung und die „Multithread“-Tauglichkeit.



### MAVLink Parser

Das „MAVLink“-Protokoll ist die Kommunikationsschnittstelle zwischen einer Bodenstationssoftware und dem Autopilotensystem. Dieses Protokoll ist in verschiedene Datenframes aufgeteilt. Ein „MAVLink“-Frame hat zwischen 8 und 263 Bytes und besteht neben den eigentlichen Datenbytes für die Information aus 8 zusätzlichen Kontroll- und Steuerbytes (Abbildung: 6.9).

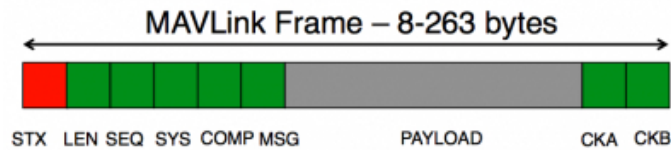


Abbildung 6.9: MAVLink Frame<sup>2</sup>

Der Autopilot beginnt sofort nach der Verbindungsaufnahme mit dem Senden der Nachrichten. Das Protokoll beinhaltet mehr als 200 unterschiedliche Nachrichten und Befehlssätze, die mit den Bytes 4 und 5 aus Abbildung 6.9 unterschieden werden. Um Fehler in den Datenpaketen von der Empfangsstation zu erkennen, wird eine 2 Byte lange Checksumme vom Sender berechnet und an den Datenstream gehängt (CKA, CKB aus Abbildung: 6.9).

Die Steuersoftware der Antennennachführung benötigt von den vielen unterschiedlichen Nachrichten nur zwei spezielle Datenframes. Die Nachricht mit der ID24 beinhaltet die rohen GPS-Positionen und weitere Zusatzinformationen wie HDOP, VDOP, GPS-Geschwindigkeit und die Anzahl der sichtbaren Satelliten. Die Nachricht mit der ID33 beinhaltet die gefilterten und fusionierten Daten aus den GPS-Positionen und den Beschleunigungssensoren. Die vollständige Aufschlüsselung der implementierten Nachrichten und Befehle ist unter <https://pixhawk.ethz.ch/mavlink/> zu finden. Die frei verfügbare, Header basierte Bibliothek verfügt über alle notwendigen Funktionen, um den ankommenden Datenstream zu entschlüsseln und in ein lesbares Format zu konvertieren. Sie bietet ebenso die Möglichkeit, Befehle in ein MAVLink-Frame zu verschlüsseln, um es anschließend an den Autopiloten zu senden. Der gesamte Informationsaustausch basiert daher auf einer hybriden Form der Kommunikation (Kapitel: 3.2.2). Die originale Bibliothek verfügt über keine Implementierungsbeispiele, deshalb wurde ein eigenes Beispiel an das Programmverzeichnis hinzugefügt. Die aktuelle, unveränderte Version der Bibliothek ist unter <https://github.com/mavlink/mavlink/downloads> verfügbar.

<sup>2</sup><http://qgroundcontrol.org/mavlink/start>

## Servo Parser

Der Servo Parser ist eine Bibliothek, die aus einzelnen Dezimalwerten den Datenstream für den Servocontroller generiert. Abbildung 6.10 zeigt die verwendete Protokollstruktur für den Servocontroller und wurde ursprünglich für einen „Arduino“-Mikroprozessor von Christopher Wojno programmiert. Die vollständigen Protokollmöglichkeiten werden im Servocontroller-Manual Pololu (2006) eingehend beschrieben. In diesem Projekt wird für die Steuerung der Antennennachführung nur das Kommando für das Setzen der absoluten Position benötigt (Source Code: 6.1). Bei der Programmierung wurden jedoch alle Funktionen des Servocontrollers in der Bibliothek umgesetzt.

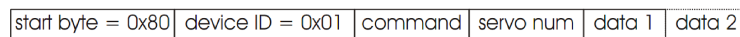


Abbildung 6.10: Pololu Protocol<sup>3</sup>

Listing 6.1: Generierung Pololu-Protokoll String<sup>4</sup>

```
Commands.cmd[0] = 0x80;
Commands.cmd[1] = 0x01;
Commands.cmd[2] = 0x04;
Commands.cmd[3] = (unsigned char) servo_number;
Commands.cmd[4] = ((unsigned char)(position >> 7) & 0xFF);
Commands.cmd[5] = ((unsigned char)(0x7f & position) & 0xFF);
```

Durch die Inkompatibilität von „Arduino“-Bibliotheken und der verwendeten Windows Plattform mussten auch hier Anpassungen vorgenommen werden. Aus der „Arduino“-Bibliothek wurden nur die rudimentären Funktionen entnommen und mit einer Fehlerbehandlung sowie einer Dokumentation erweitert. Zusätzlich wurde dem Projekt ein Beispiel hinzugefügt, welches aus dezimalen Eingabewerten den Datenstream entsprechend dem „Pololu“-Protokoll generiert und ausgibt.

Tabelle 6.2: Beispiel: Setzen einer absoluten Position

Byte	Dezimal 0	Pololu-Protokoll
Startbyte	128	0x80
Geräte-ID	1	0x01
Kommando	4	0x04
Servo Nummer	1	0x01
Servo Position	3496	0x1b (High-Byte) 0x28 (Low-Byte)

Tabelle 6.2 zeigt als Musterbeispiel die notwendigen Werte für das Setzen einer absoluten Position des Servos am Ausgang 1. Die generierten Hexadezimalwerte aus der dritten Spalte werden, beginnend vom Startbyte, an den Servocontroller gesendet. Der Servocontroller entnimmt aus dem eingehenden Datenstream die anzusteuern Servonummer und die gewünschte Servoposition. Diese Servoposition wird anschließend vom Mikroprozessor des Servocontrollers in eine Pulsdauer umgerechnet und an den jeweiligen Ausgang angelegt.

<sup>3</sup>Pololu (2006)

<sup>4</sup><https://github.com/wojnosystems/pololu-ms8sc-pololu-mode>  
und Klaus Schmaranz (2002)

## Kompass Parser

Der Kompass Parser wandelt die eingehenden Hexadezimaldaten des CMPS-Kompassmoduls (Kapitel: 4.3.2.2) in ein dezimales Format um. Zur Berechnung der Orientierung der Antennennachführung wird die in Tabelle 4.9 dargestellte Kommandosequenz „0x23“ verwendet. Durch diese Sequenz werden vom Kompassmodul vier Datenbytes an den Computer gesendet. Diese Datenbytes beinhalten je ein „signed character“-Byte zur Darstellung des Roll- und Pitchwinkels und zwei „unsigned character“-Bytes für den Yawwinkel. Die Umrechnung der Roll- und Pitchdaten kann direkt über:

Listing 6.2: 1-Byte CMPS-Typkonvertierung

```
tmp = (long)cByte;
if(tmp<32768) {iByte = tmp;}
else {iByte = tmp-65536;}
```

erfolgen.

Bei der Umrechnung der Yawdaten müssen der Aufbau und der Zusammenhang der beiden Bytes bekannt sein. Das erste eingehende Byte ist vom Hersteller als das „Highbyte“ und das Zweite als „Lowbyte“ definiert. Die Umrechnung in einen Dezimalwert erfolgt durch:

Listing 6.3: 2-Byte CMPS-Typkonvertierung

```
iHigh = ((int)cmps_answer[0]) & 0xFF;
iLow = ((int)cmps_answer[1]) & 0xFF;
iSum = (iHigh << 8) | iLow;
```

Werden bei dem Kompassmodul die rohen Magnetfeld- oder Beschleunigungsdaten angefordert, so wird jeder Wert in einer „16-Bit signed“-Darstellung übermittelt. Dadurch ergibt sich für den Datenwert eine 15-Bit-Darstellung. Das 16te-Bit wird für das Vorzeichen verwendet.

### 6.2.3 Application Basics

Der Layer „Application Basics“ enthält alle notwendigen Bibliotheken und Routinen zur Initialisierung der Endgeräte. Ein vollständiger Initialisierungsdurchlauf erfolgt in fünf Schritten die parallel oder sequenziell ablaufen (Abbildung: 6.11). Jeder dieser Schritte kann mit den erstellten Testprogrammen auch einzeln durchgeführt werden.

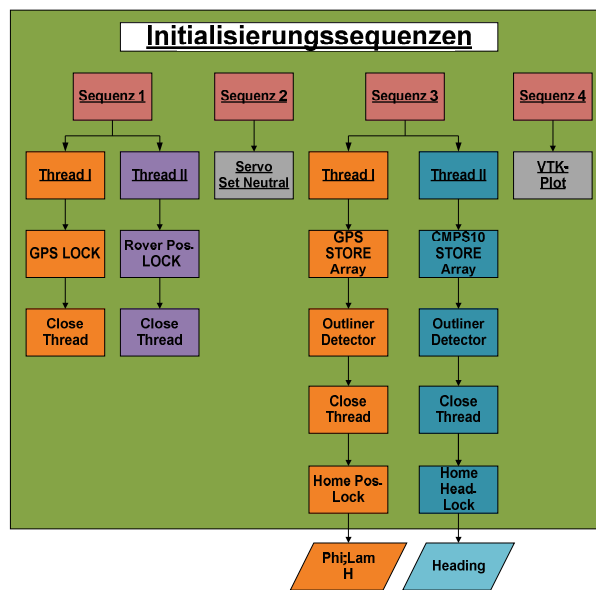


Abbildung 6.11: Softwareschema Initialisierung

Beim ersten Schritt wird der GPS-Empfänger der Bodenstation und des Rovers gestartet. Erst nach einem positiven 3D-Lock beider Empfänger beginnt der nächste Initialisierungsschritt. Der zweite Initialisierungsschritt bringt die Antriebsservos der Antennennachführung in die Neutralposition. Ist die Nachführung ausgerichtet, starten die nächsten Sequenzen.

Bei der dritten und vierten Sequenz werden parallel die GPS-Positionen und Kompassdaten der Antennennachführung aufgezeichnet und eine robuste Datenaufbereitung durchgeführt. Bei diesem Schritt wird die „Home“-Position definiert und muss möglicherweise mehrmals durchgeführt werden. Das Ergebnis der Berechnungen ist einerseits die ausreißerbefreite Position bzw. Orientierung der Antennennachführung und andererseits eine Fehlerabschätzung der berechneten Daten (Kapitel: 5.1). Diese Fehlerabschätzung wird in weiterer Folge zur Echtzeitbestimmung des Antennenbuffers benötigt (Kapitel: 5.4).

Im fünften Schritt werden die Rohdaten und die Grenzen der robusten Datenaufbereitung visuell dargestellt. Ab diesem Punkt kann der Nutzer entscheiden, ob eine erneute Initialisierung der „Home“-Position mit anderen Parametern durchgeführt werden soll. Ist keine erneute Berechnung mehr notwendig ist der Initialisierungsvorgang abgeschlossen und die Software geht in den Betriebsmodus über.

### 6.2.4 Runtime Environment

Die „Runtime“-Bibliothek wird nach der Initialisierung aufgerufen und ist für die autonome Steuerung der Antennenausrichtung zuständig. In Abbildung 6.12 wird das Ablaufschema der Software gezeigt. Diese Bibliothek empfängt kontinuierlich die Positionsdaten (Kapitel: 6.2.2) des Rovers und berechnet selbstständig in einer Schleife die zur Steuerung benötigten Parameter.

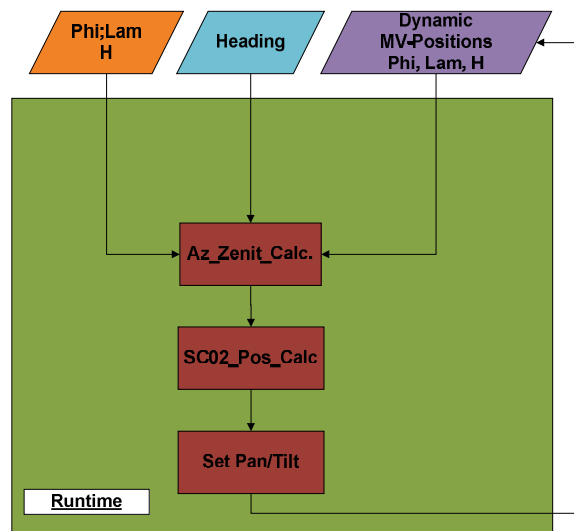


Abbildung 6.12: Softwareschema Laufzeit

Wie die Abbildung 6.12 zeigt, ist zu den kontinuierlichen Roverdaten für die Bibliothek zusätzlich auch die Position und die Orientierung der Antennennachführung notwendig. Diese Eingangsparameter können durch Initialisierung (Kapitel: 6.2.3) oder durch manuelle Eingabe zur Verfügung gestellt werden. Durch dieses Programmierkonzept kann diese Bibliothek auch ohne die Initialisierungsumgebung wiederverwendet werden. Die Roverdaten werden durch eine bibliotheksinternen Routine selbstständig abgefragt und aktualisiert. Bei jedem neuen Roverdatensatz wird eine Azimuth- und Zenit-Distanzberechnung durchgeführt. Diese Parameter und auch die Fehlerabschätzungen werden anschließend zur Berechnung der Bufferwerte und der Ausrichtungsparameter herangezogen. Nach diesen Berechnungen werden die Bufferwerte und die Ausrichtungsparameter in die Einheit des Servocontrollers für die Neigungs- und Drehachse konvertiert. Zusätzlich erfolgt eine Prüfung, ob die Nachführungsparameter außerhalb des Bufferbereichs liegt und eine Nachführung erfolgen muss (Kapitel: 5.4). Ist diese notwendig, wird die entsprechende Kommandosequenz (Kapitel: 6.2.2) generiert und an den Servocontoller gesendet.

In die Bibliothek sind zusätzliche Routinen implementiert, die eine kontinuierliche Ausgabe der Kompass- und GPS-Daten der Bodenstationssensorik erlauben. Diese Routinen wurden bisher nur für Testzwecke angewendet und sind nicht in das Nutzerinterface integriert.



### 6.2.5 Main Application

Das Hauptprogramm kann als Knotenpunkt der einzelnen Bibliotheken gesehen werden und liegt als eine Konsolenanwendung vor. Die Anwendung wurde so programmiert, dass der Nutzer durch alle Menüpunkte geführt wird und fehlerhafte Eingaben gemeldet werden. Wird beispielsweise ein falscher Verbindungsparameter angegeben oder das Gerät wurde nicht angeschlossen, wird der Fehler abgefangen und das Gerät nicht mit der Software verbunden (Abbildung 6.13).

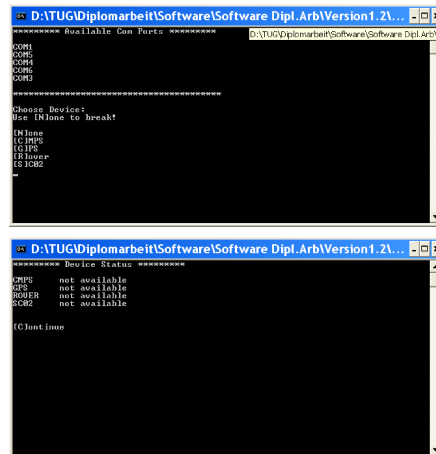


Abbildung 6.13: a) Geräteauswahl; b) Verbindungstest

Mit der Software können auch einzelne Geräte ausgewählt und getestet werden. Der Betriebsstatus kann durch die manuelle Eingabe der Position und Orientierung in Verbindung mit dem Rover überprüft werden. Vor der Initialisierung der Endgeräte ist es möglich, alle Parameter und Programmoptionen zu ändern (Abbildung 6.14).

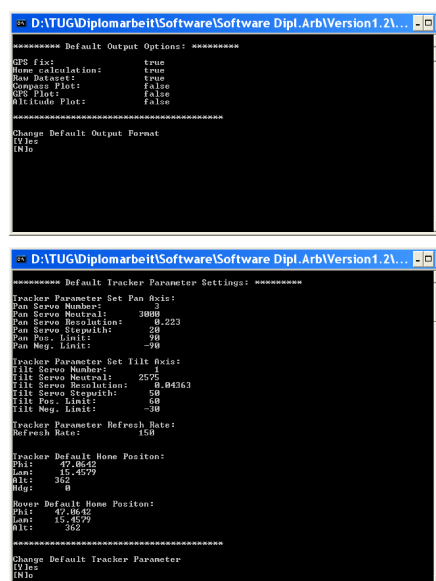


Abbildung 6.14: a) Ausgabeoptionen; b) Grundeinstellungen Antennennachführung

Für einen Gerätetest wurde nur ein Kompassmodul mit der Software verbunden. Abbildung 6.15a zeigt die Konsolenausgabe des Verbindungsstatus der implementierten Endgeräte. In Abbildung 6.15b werden die Grundeinstellungen zum Initialisierungsvorgang dargestellt. Diese Parameter können für jedes verbundene Gerät separat eingestellt und geändert werden.

```

D:\TUG\Diplomarbeit\Software\Software Dipl.Arb\Version1.21... - - x
***** Device Status *****
CHPS    running
GPS     not available
ROSER   not available
SOB2    not available

[Continue]

D:\TUG\Diplomarbeit\Software\Software Dipl.Arb\Version1.21... - - x
***** Default Device Calculation Properties: *****
CHPS:
Data Field Size: 10
Accuracy: 2
Sampling Intery.: 250
Deviation: 39.7
Deviation: 1.31667

*****
Change Default Device Initialization Properties
IV:es
IM:lo
-

```

Abbildung 6.15: a) Verbundene Geräte; b) Grundeinstellungen Kompassinitialisierung

Wurden alle gewünschten Einstellungen vorgenommen, wird die Initialisierung der gewählten Endgeräte durchgeführt. Dabei werden die einzelnen Datensamples sowie die Berechnungsergebnisse auf der Konsole dargestellt (Abbildung 6.16a). Bei der graphischen Ausgabe werden die Berechnungsergebnisse der robusten Datenaufbereitung wie in Abbildung 6.16b geplottet.

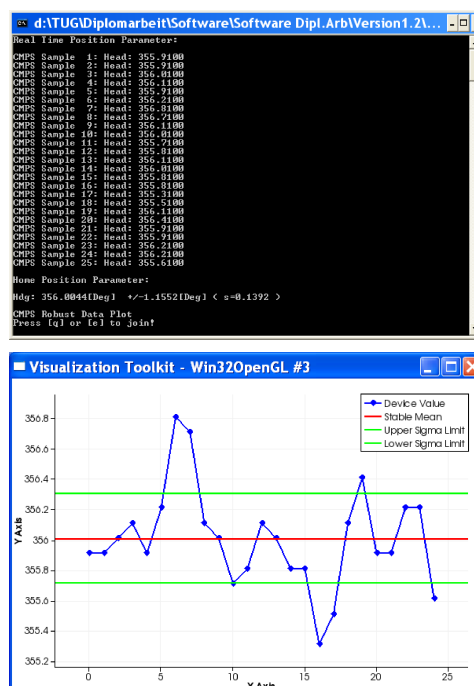


Abbildung 6.16: a) Echtzeit Datenausgabe; b) VTK-Ergebnisplot

Diese graphische Darstellung steht derzeit als Liniplot für die Kompass-Initialisierung und für die GPS-Höheninformation zur Verfügung. Weiters ist ein Scatterplot für die GPS-Positionsdaten implementiert.

Für die autonome Antennenansteuerung muss zumindest eine Verbindung zum Rover aufgebaut sein. Auf der Konsole werden dann laufend die aktualisierten Berechnungsergebnisse wie:

- Positionsdaten der Antennennachführung
- Positionsdaten des Rovers
- Nachführungsparameter

abgebildet.

Für Datenspeicherung, Bestimmung der Nachführungsauflösung und weiteren Tests wurde das Hauptprogramm in vereinfachte Versionen abgeändert. Dadurch nutzen diese nur einen Teil der Routinen des Hauptprogramms und sind somit leichter und schneller zu verwenden. Diese abgeänderten Versionen erlauben keine Parameteränderung über die Konsole und können dadurch nur mit einem Gerät kommunizieren.

# Kapitel 7

## Versuche, Messungen und Ergebnisse

Es wurden mehrere Tests und Messungen durchgeführt, um die Funktionstüchtigkeit der einzelnen Komponenten zu evaluieren. In diesem Kapitel werden die durchgeführten Messungen und Tests genau erläutert. Alle Tests wurden mit Ausnahme der Nachführungsauflösung und der Validierung der „Home“-Position auf offenem Gelände durchgeführt (Abbildung: 7.1).



Abbildung 7.1: Test- und Versuchsgelände<sup>1</sup>

Die einzelnen Komponenten wurden schon während der Bauzeit auf ihre Funktionalität hin geprüft. Durch diese Vorgehensweise konnten Mängel in den einzelnen Modulen rechtzeitig erkannt und beseitigt werden.

<sup>1</sup><http://gis2.stmk.gv.at>

## 7.1 Nachführungsauflösung

In Kapitel 5.5 wurden die Auflösungen für die Antennennachführung anhand der Servo-Parameter bereits abgeschätzt. Die ersten Versuche zeigten jedoch, dass diese Kennwerte stark von den tatsächlichen Werten abweichen. Daher wurden die Auflösungen durch Relativmessungen in der Horizontal- und Vertikalebene unabhängig ermittelt. Bei den Messungen wurde ein Maßband an einer ebenen Fläche fixiert und die Nachführung in einer definierten Distanz positioniert (Abbildung: 7.2). Die Markierung am Maßband erfolgte dabei mit einem Laserpointer der auf einer der Montagekonsolen montiert wurde. Die Exzentrizitäten die sich mit dieser Art der Montage ergeben wurden bei diesem Versuch vernachlässigt.

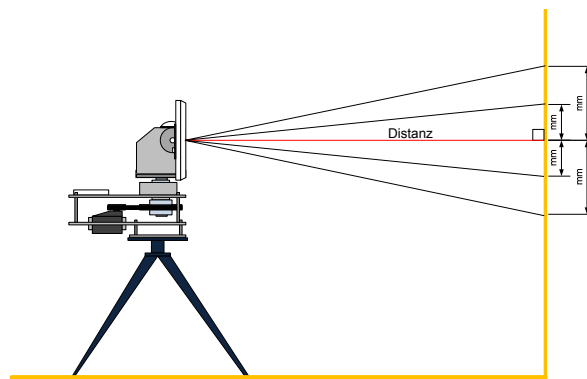


Abbildung 7.2: Messschema Neigungsachsenauflösung

Die Nullmessung ist der Wert der Neutralstellung von der Antennennachführung. Anschließend wurde die jeweilige Achse um eine äquidistante Anzahl an Servo-Schritten verstellt und der Wert am Maßband abgelesen. Mit der Distanz der Drehachse zum Maßband und der Differenz zur Nullmessung kann der Winkel berechnet werden. Dieser Winkel steht in Abhängigkeit zu den Servo-Schritten und der Neutralposition und in weiterer Folge zur Schrittauflösung. Die Formeln 7.1 bis 7.6 beschreiben den Berechnungsvorgang zur Ermittlung der Einzelschrittauflösung und dessen Standardabweichung (Lothar Sachs & Jürgen Heddrich, 2006). In Tabelle 7.1 sind die Indizes der Berechnungen erläutert.

Tabelle 7.1: Berechnungsindizes	
Indizes	Erklärung
i	Messwert
j	Messreihe
n	Anzahl der Messwerte
SRW	Servo Schrittweite
SR	Servo Einzelschritt
D	Distanz (Achse-Maßband)

$$\Delta H_i^j = H_i - H_0^j \quad (7.1)$$

$$\alpha_{i(STW)}^j = \arctan\left(\frac{\Delta H_i^j}{D}\right) \quad (7.2)$$

$$\bar{\alpha}_{ST}^j = \text{median}\left[\frac{\bar{\alpha}_{i(STW)}^j}{STW_i^j}\right] \quad (7.3)$$

$$s_{\bar{\alpha}_{ST}}^j = \text{std}\left[\frac{\bar{\alpha}_{i(STW)}^j}{STW_i^j}\right] \quad (7.4)$$

$$\bar{\alpha}_{ST} = \text{mean}(\bar{\alpha}_{ST}^j) \quad (7.5)$$

$$s_{\bar{\alpha}_{ST}} = \sqrt{\frac{1}{n-1} * \left[ \sum (n_j - 1) * s_{\bar{\alpha}_{ST}^j}^2 + \sum n_j * (\bar{\alpha}_{ST}^j - \bar{\alpha}_{ST})^2 \right]} \quad (7.6)$$

Zur Berechnung der Auflösung wurden mehrere Messreihen je Achse mit unterschiedlichen Distanzen durchgeführt. Nach jeder Messung wurde die jeweilige Achse in die Neutralposition zurückgestellt, um anschließend das nächste Vielfache der Schrittweite zu messen. Bei der Neigungsachse wurde dabei eine Schrittweite von 100-Schritten für jede Messung gewählt. Die obere und untere Grenze wurde mit 2000 und 3000 Servo-Positionen festgelegt. Die Drehachse wurde mit einer Schrittweite von 20-Schritten und zwischen den Grenzen von 2900 bzw. 3100 Servo-Positionen gemessen.

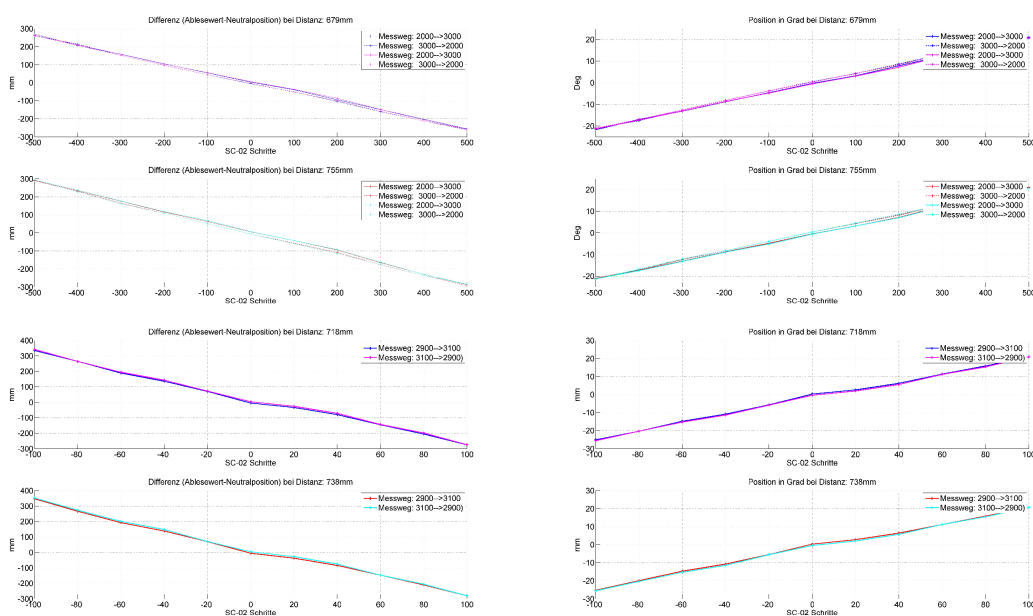


Abbildung 7.3: oben) Neigungsachse: Messwerte, Grad; unten) Drehachse: Messwerte, Grad

In Abbildung 7.3 sind die reduzierten Messreihen für beide Distanzen und Achsen dargestellt. Aus der Grafik wird ersichtlich, dass die Messreihen Hystereseeffekte und geringe Offsets aufweisen. Diese Effekte lassen sich für die Neigungsachse auf das Umkehrspiel in der Kraftübertragung des Kettenantriebes zurückführen. Bei der Drehachse wird vermutet, dass dieser Effekt durch eine geringe Abweichung, der Wellenbohrung von der Orthogonalen herrührt. Dadurch entstehen unterschiedliche Spannungen in der Zahnriemenuntersetzung.

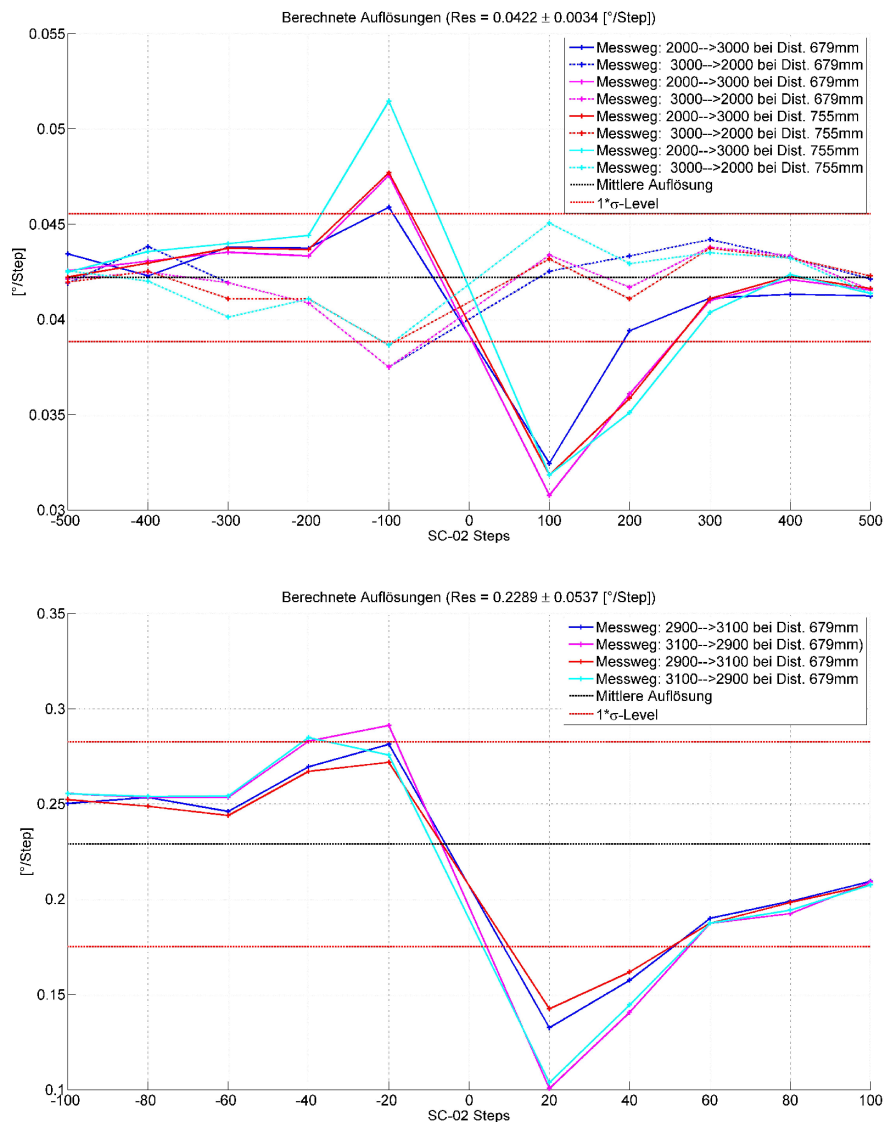


Abbildung 7.4: a) Auflösung Neigungsachse; b) Auflösung Drehachse

In Abbildung 7.4 werden die aus den reduzierten Messwerten berechneten Auflösungen grafisch dargestellt. Aus der Berechnung des Mittelwerts der vorliegenden Daten ergab sich eine Auflösung von  $0.0422$   $[\frac{^{\circ}}{\text{Step}}]$  für die Neigungsachse und eine Drehachsauflösung von  $0.23$   $[\frac{^{\circ}}{\text{Step}}]$  mit den Standardabweichungen von  $0.0034$   $[\frac{^{\circ}}{\text{Step}}]$  bzw.  $0.054$   $[\frac{^{\circ}}{\text{Step}}]$ .



## 7.2 Validierung der „Home“-Position

Zur Validierung der Nachführungsposition wurden Vergleichsmessungen zwischen bekannten Punkten und den aufbereiteten GPS-Daten durchgeführt. Als Referenzpunkte wurden die Messpfeiler auf dem Dach des Geodäsiegebäudes der Technischen Universität Graz verwendet (Abbildung: 7.5).

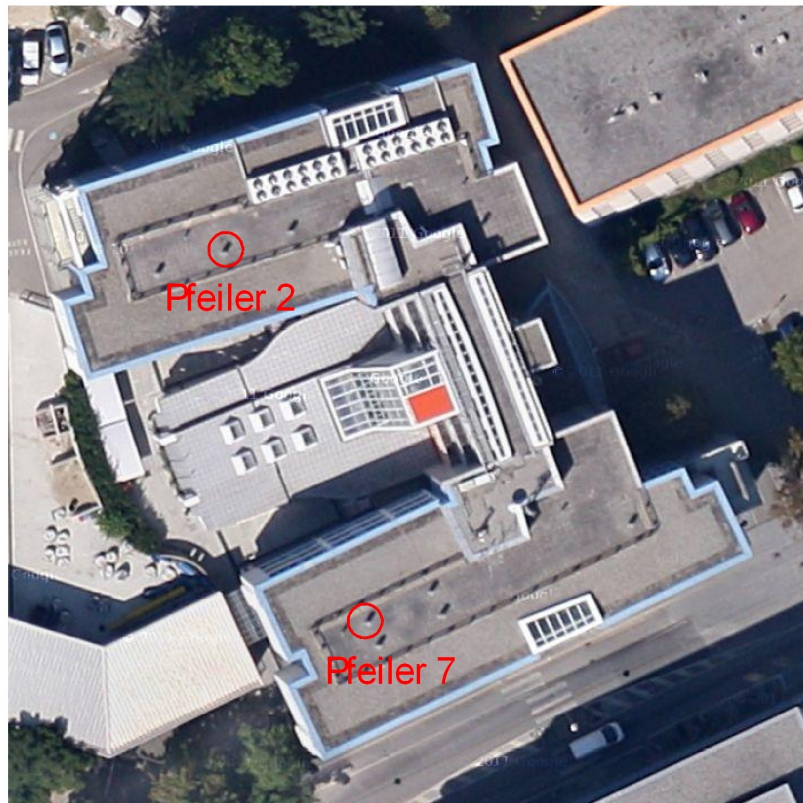


Abbildung 7.5: Messpositionen zur GPS-Validierung<sup>2</sup>

Tabelle 7.2: Koordinaten der Messpfeiler

Messpfeiler	$\varphi$ [°]	$\lambda$ [°]	h [m]
2	47.0641691700	15.4529530553	378.1234
7	47.0644673952	15.4527823937	378.1171

Die Koordinaten (Tabelle: 7.2) dieser Messpfeiler sind hochgenau bekannt. Im Vergleich zu den erwarteten Genauigkeiten der Nachführungs-GPS-Positionen können diese Koordinaten als fehlerfrei angenommen werden. Durch die exponierte Lage der Messpfeiler wurden nur geringe Abschattungseffekte bei den GPS-Messungen erwartet. Multipatheffekte können jedoch hier nicht gänzlich ausgeschlossen werden.

<sup>2</sup><http://maps.google.at/maps?hl=de&tab=wl>



Es wurden auf beiden Pfeilern mehrere Messreihen zu je 200 Messpunkten durchgeführt. Der direkte Vergleich mit den Pfeilerpositionen wurde mit den ausreißerbefreiten Positionslösungen durchgeführt.

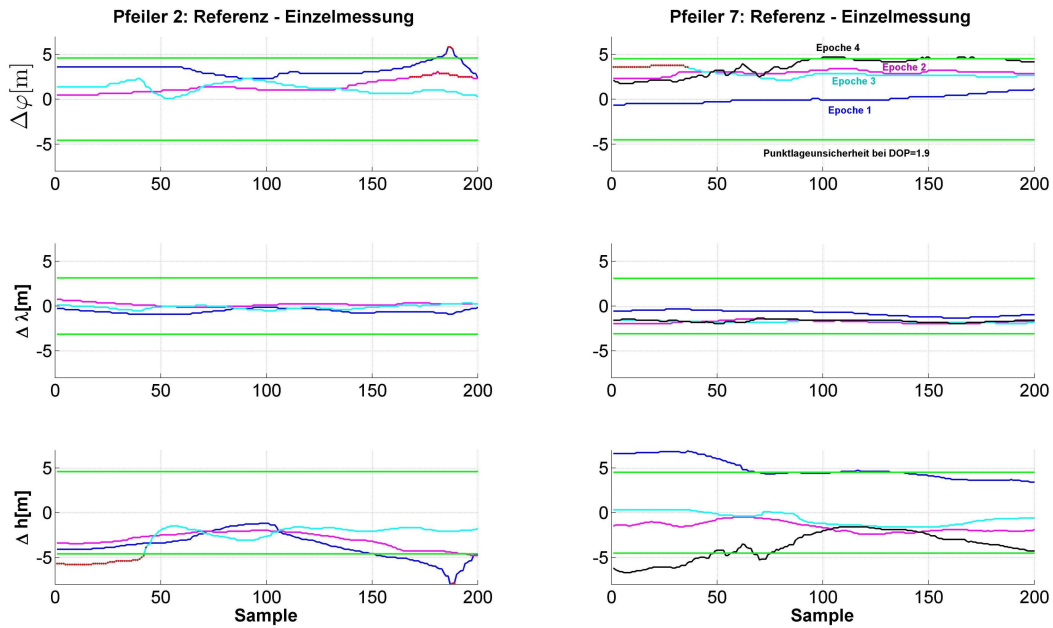


Abbildung 7.6: Differenzen der Einzelmessungen zur Referenzposition

Abbildung 7.6 zeigt die Differenzen der Einzelpunktmessungen zur Referenzposition. Die roten Punkte stellen dabei die  $3 \cdot \sigma$  Ausreisser (robuste Ausreisserdetektion) der Zeitreihe dar. Aus der Grafik ist erkennbar, dass die Abweichungen in Ost-Westrichtung deutlich geringer sind als in der Nord-Südrichtung. Diese Tatsache lässt sich durch die Verteilung der Satelliten zum Messzeitpunkt erklären (Abbildung: 7.7).

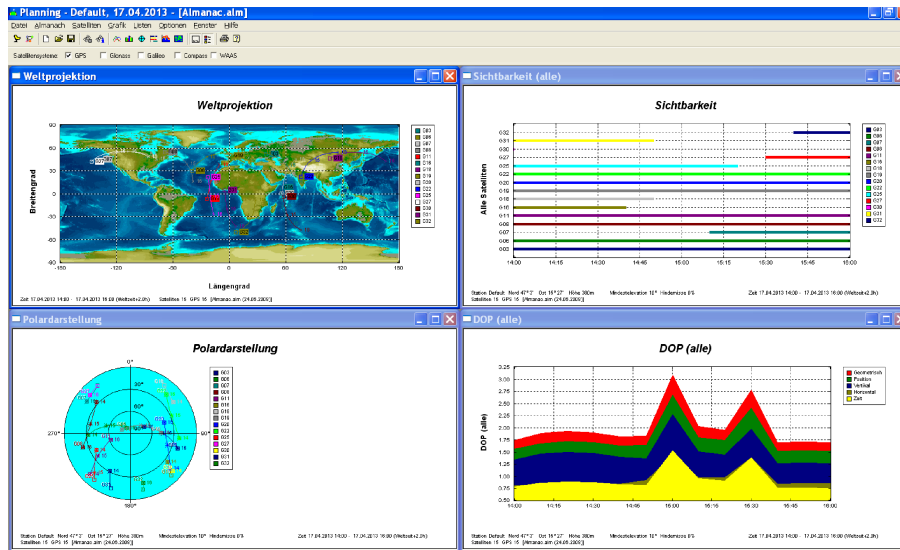


Abbildung 7.7: Satellitensichtbarkeit zur Messzeit

Aus der Polardarstellung in Abbildung 7.7 erkennt man, dass die Satelliten zum Messzeitpunkt größtenteils östlich und westlich der Messposition stationiert waren. Diese Konfiguration ergibt sich aus dem Orbitdesign des GPS-Systems und der Messposition. Das Orbitdesign des GPS-Systems ist in der Arbeit „GNSS-Global Navigations Systems“ von Bernhard Hofmann-Wellenhof *et al.* (2008) ausführlich beschrieben.

Tabelle 7.3: Pfeiler2 Ausreißerbefreite Mittelwerte

Messreihe	$\varphi$ [°]	$\lambda$ [°]	h [m]
1	47.064138 (s=0.59 m)	15.452962 (s=0.35 m)	381.9 (s=1.41 m)
2	47.064159 (s=0.51 m)	15.452950 (s=0.29 m)	380.8 (s=0.84 m)
3	47.064158 (s=0.57 m)	15.452955 (s=0.32 m)	380.0 (s=0.46 m)

Tabelle 7.4: Pfeiler7 Ausreißerbefreite Mittelwerte

Messreihe	$\varphi$ [°]	$\lambda$ [°]	h [m]
1	47.064469 (s=0.45 m)	15.452791 (s=0.46 m)	373.9 (s=1.13 m)
2	47.064440 (s=0.30 m)	15.452805 (s=0.26 m)	379.9 (s=0.58 m)
3	47.064443 (s=0.24 m)	15.452806 (s=0.18 m)	378.8 (s=0.71 m)
4	47.064429 (s=1.00 m)	15.452804 (s=0.22 m)	381.3 (s=1.58 m)

In den Tabellen 7.3 und 7.4 sind die Ergebnisse der robusten Ausreißerdetektion für jede Messreihe dargestellt. Die geringen Standardabweichungen lassen sich nur durch die kurze Messdauer von *approx 3.5 min* erklären und sind nicht als die Genauigkeit des GPS-Empfängers zu interpretieren. Driften und eine Satellitenkonstellationsänderung gehen durch die kurze Messdauer nur in geringem Maß in die Messungen ein. Um einen direkten Vergleich mit der angedachten Anwendung herstellen zu können wurde diese Messdauer bewusst gewählt.

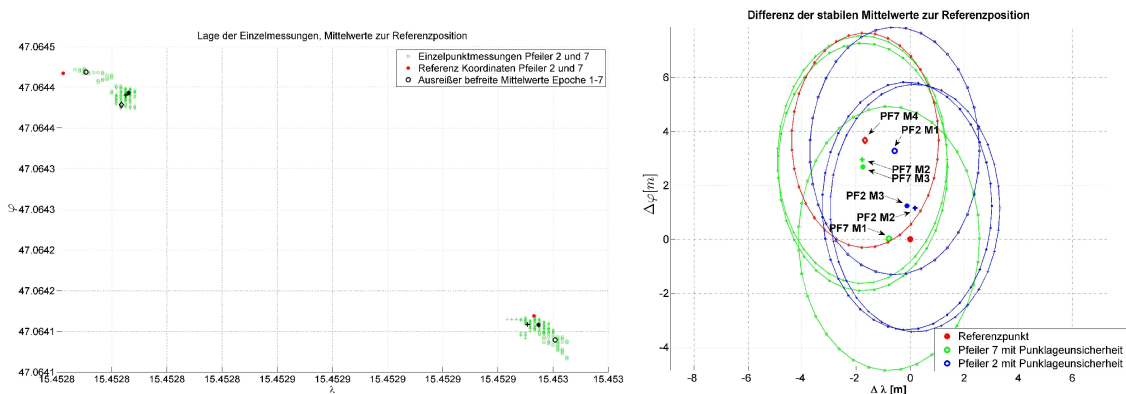


Abbildung 7.8: a) Messwerte absolut zur Referenz; b) Mittelwert relativ zur Referenz

In Abbildung 7.8a) sind die Einzelmesspunkte mit deren robusten Mittelwerten und die Referenzpositionen mit den Absolutkoordinaten dargestellt. Abbildung 7.8b) zeigt die Differenzen der robusten Mittelwerte zu den Referenzpositionen. Die Ellipsen stellen die absolute Punktlageunsicherheit dar, welche nach Formel 5.10 berechnet wurden.

Auffällig ist die Abweichung am Pfeiler Nr. 7 aus der vierten Messreihe. Bei dieser Messung fällt der tatsächliche Wert nicht in den Unsicherheitsbereich der Messreihe. Das kann durch

eine Veränderung der Satellitenkonstellation und den dadurch aufgetretenen Multipatheffekten erklärt werden.

In diesem Test sollte in erster Linie die Funktionalität der robusten Ausreißerdetektion überprüft werden. Wie schon im vorherigen Absatz beschrieben, ist im Feldeinsatz darauf zu achten, dass genügend Nutzdaten vorhanden sind. Ansonsten kann es leicht zu Fehlinterpretationen kommen. Auch konnte gezeigt werden, dass sich die tatsächliche Position im dem Bereich befindet, der sich aus den DOP-Werten berechneten Messunsicherheiten ergibt.

### 7.3 Kompassoffsets

In Kapitel 4.3.2.2 wurde bereits der grundlegende Aufbau und das Protokoll des Kompasses beschrieben. Die Bestimmung der Deklination, auch Variation genannt, erfolgte mit einem Onlineprogramm der „National Oceanic and Atmospheric Administration“. Diese Organisation stellt unter anderem Berechnungsprogramme für den Geomagnetismus zur Verfügung, aus denen die Deklination abgeleitet werden kann (Abbildung: 7.9).

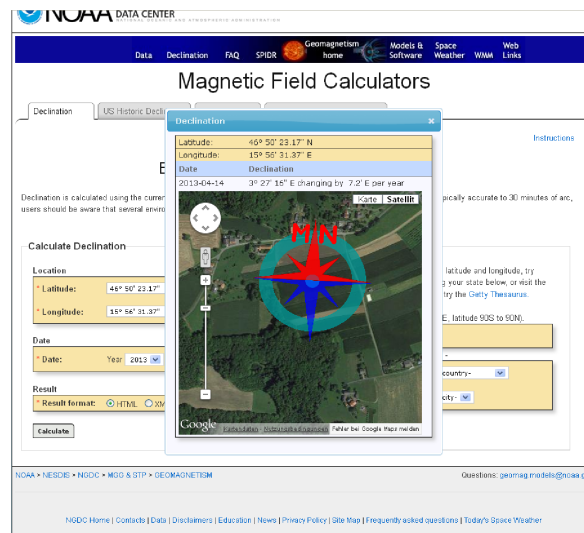


Abbildung 7.9: Magnetisch Nord Deklination<sup>3</sup>

Dieser Offset wurde bei der Nachführungssoftware bereits als Initialisierungswert für die Deklination festgelegt, sodass die rohen Kompassdaten entsprechend korrigiert werden. Der CMPS-10 Kompass sollte zusätzlich zu den drei Attitude-Parametern auch eine „Hard-Iron“-Kalibrierung implementiert haben. Bei dieser Kalibrierung wird der Kompass in 90°-Schritten um die Yaw-Achse gedreht. Die Firmware berechnet aus den Differenzen des feststehenden und des bewegten Magnetfeldes anschließend Korrekturwerte für die Magnetfeldsensoren. Die Berechnungsalgorithmen sind in den technischen Berichten der verschiedenen Chiphersteller

<sup>3</sup><http://www.ngdc.noaa.gov/geomag/geomag.shtml>

veröffentlicht (Talat Ozyagcilar, 2012 und STMicroelectronics, 2010).

Beim Versuch wurde jedoch festgestellt, dass bei dieser Kalibrierung mit dem eingebauten CMPS-10 Kompass die Korrekturparameter nicht übernommen werden und der Sensor einen Fehlercode zurücksendet. Durch die Einbaulage in der Antennennachführung und das magnetfeldbeeinflussende Metall ist eine solche „Hard-Iron“-Kalibrierung unumgänglich um Missweisungen zu vermeiden.

Dieser Fehler konnte schlussendlich auf die verwendete Firmware zurückgeführt werden. Die Herausgabe der Skalierungsparameter für den Magnetfeldsensor und für die Beschleunigungssensoren konnte nicht erreicht werden. Daher war es nicht möglich aus den Rohdaten des Kompassmoduls eine eigene Berechnung durchzuführen. Die Herstellerfirma bietet auch keine neue Firmware mit einem Updatekonzept an, wodurch dieser Kompass nur bedingt einsetzbar ist.

Für eine Weiterentwicklung dieses Projekts wäre es sinnvoller, das Kompassmodul durch einen Sensor von Honeywell HMC5883L auszutauschen und die Berechnungen mit dem Algorithmus von Talat Ozyagcilar (2012) und STMicroelectronics (2010) in die Nachführungssoftware zu implementieren. Bis zur Implementierung des Algorithmus wird die Orientierung der Nachführung über einen Kompass der Firma „Recta“ bestimmt. Dabei ist zu beachten, dass auch dieser Kompass nicht den magnetfeldbeeinflussenden Metallen ausgesetzt wird, da es hier ebenfalls zu gravierenden Missweisungen kommt (Abbildung: 7.10).



Abbildung 7.10: Missweisung durch einen Gewindebolzen

Da die Antennennachführung teilweise aus solchen Metallen besteht, muss die Orientierung durch das Ausrichten entlang der Längskante aus mindestens 30 cm Entfernung bestimmt werden.



## 7.4 Systemtest

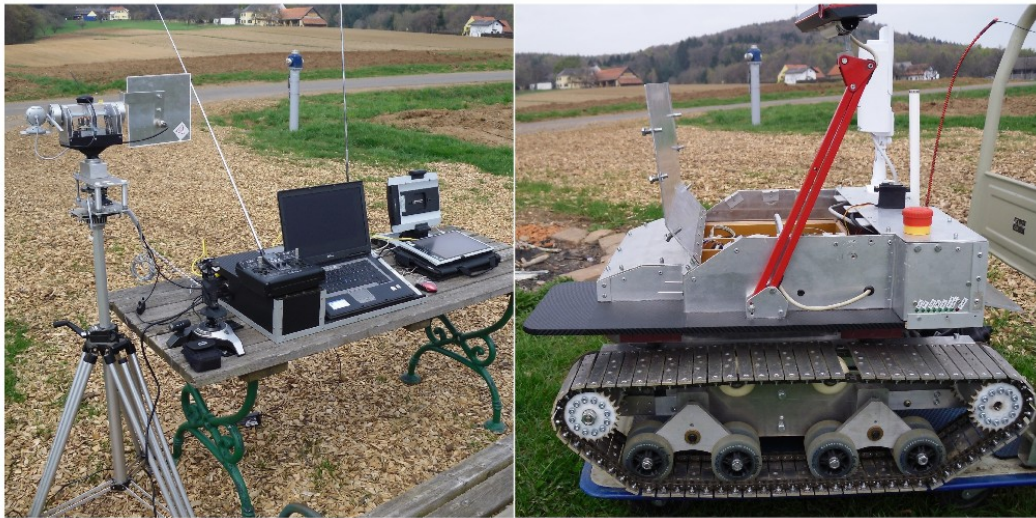


Abbildung 7.11: Systemtest Aufbau

Beim Systemtest wurden alle Hardware- und Softwareteile einem realen Feldtest unterzogen. Dabei wurden in einzelnen Schritten die verschiedenen Systeme überprüft und auf ihre Funktionalität geprüft. Im ersten Testdurchgang wurde versucht, die Reichweite der Steuerung und der Datenübertragung zu ermitteln.

Beim Steuerungstest wurde der Motorregler nicht direkt an den Empfänger angeschlossen, sondern ein Standardservo als Indikator genutzt. Für die Ermittlung der RC-Reichweite in Bodennähe wurde die Trägerplattform von der Bodenstation solange entfernt, bis beim Servo Störeinflüsse ersichtlich werden. Diese Störeinflüsse machen sich durch ein „zittern“ des Servos bemerkbar. Dieses „Zittern“ ist nicht die maximal erreichbare Reichweite, sondern kündigt lediglich einen bevorstehenden Signalabbruch an. Die Maximalreichweite ist dann erreicht, wenn die Steuersignale nicht mehr ordnungsgemäß an den Servo weitergegeben werden können. Auch ist die erreichbare Reichweite bei einem bodengestützten Rover kürzer als bei einer Flugdrone, was durch die stärkere Bodenreflexionen der Steuersignale zu erklären ist. Um einen Vergleichswert zu erhalten, wurde dieser Test einerseits mit der Bodenstation und andererseits mit einer handelsüblichen RC-Fernbedienung durchgeführt. Die erzielten Reichweiten können insofern direkt verglichen werden, als dass im RC-Handsender und in der Bodenstation dieselben Sendermodule zum Einsatz kommen.

Bei diesem Test stellte sich jedoch heraus, dass die Reichweiten mit dem 40 MHz-System deutlich kürzer sind als die ursprünglich angenommenen 200 Meter. Bereits nach  $\approx 45\text{m}$  zeigte sich ein deutliches „Zittern“ beim Testservo. Da dieses Verhalten auch bei einem Motorregler denselben Effekt hätte, würden die beiden Antriebsmotoren unkontrolliert anlaufen und stoppen. Daher wurde in weiterer Folge bei dem Steuerungstest die maximale Distanz für den Rover

auf  $\approx 40$  m festgelegt. Durch den Vergleichstest mit dem RC-Handsender und der Bodenstation können eventuelle Fehler in der selbstentwickelten Elektronik erkannt werden. Da sich in beiden Fällen die Reichweiten nicht deutlich voneinander unterscheiden, kann es nicht an der RC-Elektronik der Bodenstation liegen. Die geringe Reichweite der RC-Steuerung kann durch drei mögliche Ursachen erklärt werden:

1. **Fehlerhafter Empfänger:** Beim verwendeten Empfänger handelt es sich um ein bereits gebrauchtes Modell. Da zum Zeitpunkt der Versuche kein passendes Ersatzgerät zur Verfügung stand, konnte dieser Fehler bei den Testfahrten nicht ausgeschlossen werden.
2. **Antennenverlegung:** Durch die Antennenverlegung können Störungen durch die Elektronik auftreten. Da das System ein analoges Steuersignal verwendet, ist es anfällig auf Einflüsse die in dieser Frequenz auftreten.
3. **Störungen durch den Antrieb:** Da der Antrieb mit Bürstenmotoren ausgestattet ist, können durch das Bürstenfeuer hochfrequente Störsignale entstehen. Auch hier würde der Empfänger auf diese Signale reagieren. Da die Motoren jedoch durch ihre Einbaulage eine gewisse Distanz zum Empfänger aufweisen, wird dieser Einfluss als unwahrscheinlich angenommen.

Durch die begrenzten RC-Reichweiten konnte kein Langstreckentest mit dem vollständigen System durchgeführt werden. Im Gegensatz zur verwendeten analogen RC-Signalübertragung sind bei einer WLAN-Datenübertragung Sicherheitsroutinen implementiert, die falsche Datenpakete erkennen und repariert oder neu anfordern. Bei der WLAN-Übertragung kündigt sich ein Signalabbruch durch eine geringer werdende Datendurchsatzrate und einem abnehmenden Signalpegel an. Die Datenübertragungen der Roverpositionen zeigten bei den Testdurchläufen keine Störungen oder Durchsatzrateneinbrüche. Auch die Roverbildinformationen wurden bei den Testläufen ordnungsgemäß übertragen und konnten mit einem Web-Browser als Livestream dargestellt werden.

In diesem Zusammenhang hat sich gezeigt, dass das analoge 40 MHz-Sendersystem für dieses Projekt nicht praktikabel ist. Da die Steuerung sehr sensibel auf Fremdeinflüsse reagiert, könnte der Rover nur in näherer Umgebung der Bodenstation betrieben werden. Als Lösungsansatz für dieses Problem wird hier auf ein 2,4 GHz-System (Kapitel: 3.2.1) verwiesen. Wird das Steuerungs- und Datenübertragungssystem nur für Fahrzeuge benutzt die am Boden betrieben werden, ist auch die Verwendung der WLAN-Verbindung als Steuerbefehlsträger denkbar. Dieses alleinige System ist jedoch von Rechtswegen her nicht für Luftfahrzeuge erlaubt.

Innerhalb des Reichweitenradius der Steuerung reagierte der Rover sehr präzise auf die Steuerbefehle. Die eingestellte lineare Regelkurve stellte sich jedoch als zu sensitiv heraus, was sich in einem ruckartigen Anlaufen der Antriebsmotoren zeigte. Daher wurde beim zweiten Steue-

rungstest der Motorregler nicht mit dem linearen Regelmodus sondern mit dem exponentiellen Regelmodus betrieben. Diese Einstellung wirkt wie ein „Sanftanlauf“, wodurch die gesamte Antriebseinheit, der Motorregler und der Akku geringeren Belastungen ausgesetzt wird. Bei einem Volllasttest über mehrere Minuten zeigte sich, dass die Motoren und der Regler nur lauwarm wurden, was für die Antriebsdimensionierung spricht.

Die Bilddaten der „Low-Cost“-Kamera konnten ohne Probleme als Livestream dargestellt werden. Dabei fiel jedoch auf, dass für einen Livestream die Bilddaten von der Kamera nicht schnell genug komprimiert werden können. Die daraus entstehen Latenzzeiten in der Datenkonvertierung äußern sich im ruckartigen Bildverlauf. Der Kameravertreiber machte allerdings bereits im Vorfeld darauf aufmerksam, dass es sich nicht um eine hochwertige Kamera handelt und das ein solches Verhalten durchaus zu erwarten sei. Da eine hochwertige LAN-Kamera preislich bei mehreren hundert Euro liegt, wurde vorerst auf einen Austausch dieser verzichtet.

Das Konzept der Datenübertragung mit dem „XPort“-Modul in Verbindung mit dem eingebauten WLAN-Modul hat sich als eine flexible Methode herausgestellt um mehrere Datenstreams an eine entfernte Empfangsstation zu senden. Es kam während des gesamten Testverlaufs zu keinen Datenlücken oder Verbindungsunterbrechungen. Hierfür steht der tatsächliche Langstreckentest noch aus.

## 7.5 Trajektoriensimulation

Aus den zuvor beschriebenen Abschnitten geht hervor, dass die Steuerung des Rovers einer Weiterentwicklung bedarf, damit die angestrebte Reichweite von mehreren hundert Metern erreicht werden kann. Um die Antennennachführung dennoch unter möglichst realistischen Bedingungen zu testen, wurde ein Simulationsprogramm erstellt, welches die Koordinaten einer Trajektorie von einem Datenfile einliest und die Nachführung ansteuert. Die gewählte Route wird in Abbildung 7.12 gezeigt.

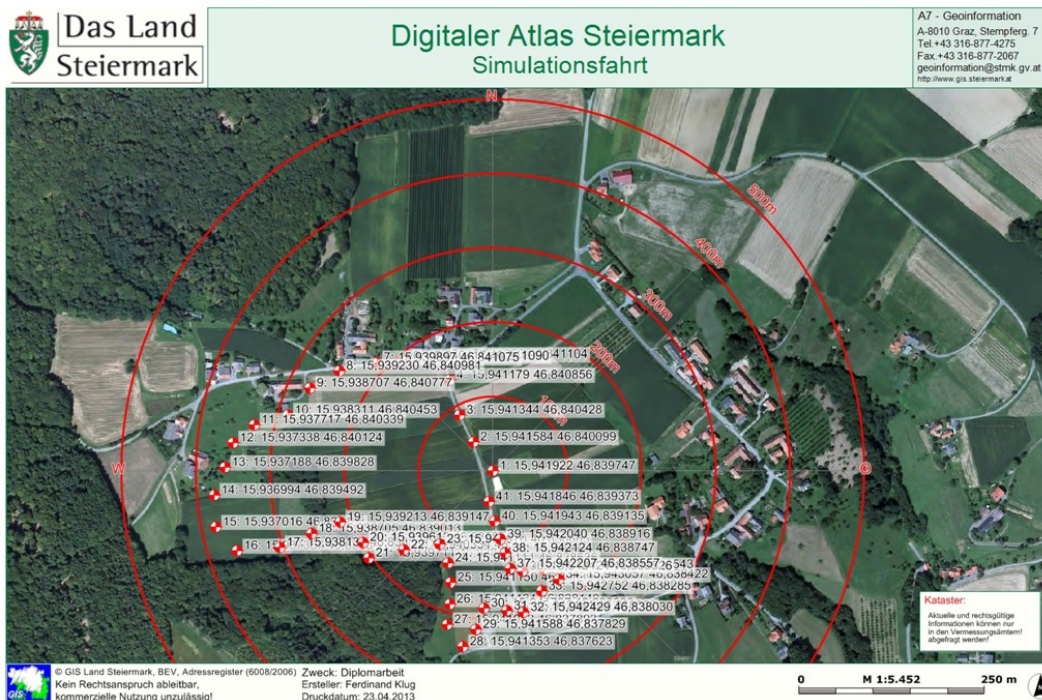


Abbildung 7.12: Trajektorie für Nachführungssimulation<sup>4</sup>

Für die Simulation wurde eine Route in einem Umkreis von  $\approx 380 \text{ m}$  gewählt. Das Datenfile beinhaltet die Koordinatensätze im WGS84-System. Da im Testgelände nur geringe Höhenunterschiede vorhanden waren, wurde diese Information für die Simulation adaptiert. Durch diese Adaption wird ein Höhenunterschied von  $\approx 60 \text{ m}$  erlangt, wobei sich der Ausgangswert an der Höhe der Nachführung orientiert (Abbildung: 7.13). Außerdem wurden die benötigten DOP-Werte der einzelnen Koordinatenpunkte festgelegt. Auch hier orientiert sich der Startwert an den DOP-Werten der Nachführungsposition (Abbildung: 7.13). Hier wurde zusätzlich versucht, die Werte auf die Einflüsse, wie Abschattungen durch einen Wald, (Abbildung: 7.12 links unten) abzustimmen. Das Datenfile enthält für die Simulation 40 Wegpunkte (Abbildung: 7.12) und die „Home“-Position mit der Orientierung. Die Berechnung und Nachführung erfolgt nach einem festen Zeitintervall von  $\approx 30 \text{ sec}$ . Dieses Zeitintervall wurde aus der mittleren Distanz zwischen den Wegpunkten ( $\approx 45 \text{ m}$ ) und der Rovergeschwindigkeit

<sup>4</sup>[www.gis.steiermark.at](http://www.gis.steiermark.at)



(5 km/h) errechnet.

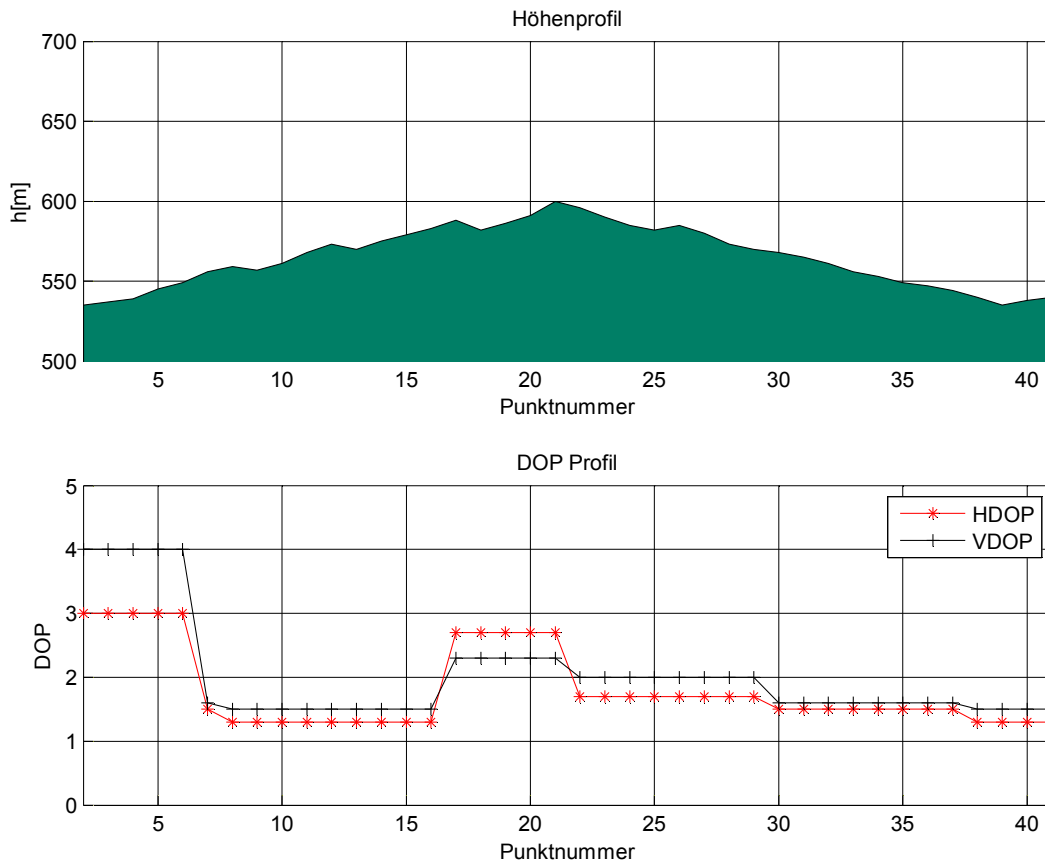
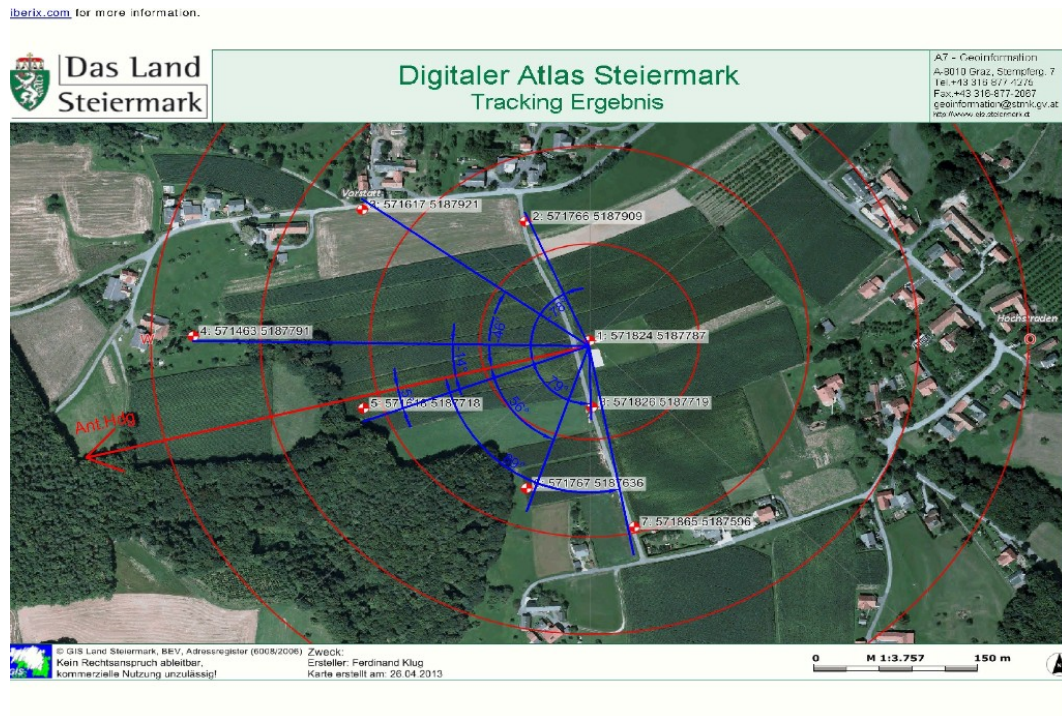


Abbildung 7.13: Höhen- und DOP-Profil für Nachführungssimulation

Mit dieser Simulation ist es möglich, die Antennennachführung und die Software unter verschiedensten Bedingungen zu testen. Wird zum Beispiel ein Gebiet mit großen Abschattungen getestet, kann das Datenfile durch das Ändern der entsprechenden Werte leicht an diese Gegebenheiten angepasst werden.

Bei der Simulation sind Probleme in der Nachführungssoftware mit den programmierten Bibliotheken entstanden. Diese haben sich in „Dead-Locks“ (Kapitel: 6.2.2) und falschen Variablenzuweisungen gezeigt. Nach der Behebung dieser Programmierfehler wurde die Nachführung in der Simulation korrekt angesteuert. Die berechneten Steuerparameter für sieben Beispiel-punkte sind in Abbildung 7.14 dargestellt.

<sup>5</sup>www.gis.steiermark.at edit mit ACAD

Abbildung 7.14: Beispielpunkte mit Nachführungswinkel<sup>5</sup>

Es wurden für die Darstellung einige Koordinaten aus der Trajektorie herausgegriffen und mit den Nachführungsparametern und den Bufferwerten in der Horizontalebene versehen. Dieses Vorgehen ist für die Vertikalebene nicht möglich, da es sich bei den Höheninformationen um Fiktivwerte handelt.

Tabelle 7.5: Simulationseinstellungen „Home“-Position

Koordinaten ( $\sigma_{robust}$ )		
$\varphi$	46.839747°	0.2301 m
$\lambda$	15.941922°	0.1618 m
h	536m	0.4885 m
Orientierung		
$\alpha$	257.5°	1.2°

In Tabelle 7.5 sind die Softwareparameter für die Nachführungsposition und Orientierung angegeben. Die Genauigkeitsangaben stammen aus einem Datenfile für die Validierung der „Home“-Position (Kapitel: 7.2). Wobei hier noch die  $1 * \sigma$ -Ausreissermethode verwendet wurde.

Zur Validierung der angesteuerten Achsen wurde beim Test eine Graduierung bei der Antennennachführung angebracht. Die abgelesenen Werte wurden anschließend mit den berechneten Werten verglichen.

```

***** Default Device Calculation Properties: *****
SC02:
Pan Servo Nr.: 3
Pan Neutral Pos.: 3000
Tilt Servo Nr.: 1
Tilt Neutral Pos.: 2575
*****

***** Default Tracker Parameter Settings: *****

Tracker Parameter Set Pan Axis:
Pan Servo Number: 3
Pan Servo Neutral: 3000
Pan Servo Resolution: 0.2289
Pan Servo Stepwidth: 20
Pan Pos. Limit: 90
Pan Neg. Limit: -90

Tracker Parameter Set Tilt Axis:
Tilt Servo Number: 1
Tilt Servo Neutral: 2575
Tilt Servo Resolution: 0.0422
Tilt Servo Stepwidth: 50
Tilt Pos. Limit: 60
Tilt Neg. Limit: -30

Tracker Parameter Refresh Rate:
Refresh Rate: 150

Tracker Default Home Positon:
Phi: 46.8397
Lam: 15.9419
Alt: 536
Hdg: 257.5

Rover Default Home Positon:
Phi: 46.8398
Lam: 15.942
Alt: 535
*****

```

Abbildung 7.15: Software Nachführungseinstellungen Simulation

Abbildung 7.15 zeigt die eingestellten Nachführungsparameter. Die Auflösungen für die Dreh- und Neigungsachse wurden aus den Resultaten von Kapitel 7.1 entnommen. In der Abbildung ist auch der Initialwert bzw. die untere Grenze der Bufferwerte (Pan Servo Stepwidth; Tilt Servo Stepwidth) ersichtlich. Aus diesen Werten ergibt sich ein minimaler Bufferbereich von  $\approx 4.5^\circ$  für die Drehachse und  $\approx 2^\circ$  für die Neigungsachse. Diese unteren Grenzen wurden eingeführt, um ein permanentes Nachführen bei schon geringster Positionsänderungen zu vermeiden.

Tabelle 7.6 zeigt die aus der „Home“-Position (Tabelle: 7.5), den Beispielkoordinaten (Abbildung 7.14) und den Nachführungs-Einstellungen (Abbildung 7.15) berechneten Nachführungs-Parameter. Bei genauerer Betrachtung der Werte lässt sich das Zusammenspiel der DOP-Werte und der berechneten Distanz auf die Ergebnisse der Nachführungsparameter erkennen. Dadurch wird gezeigt, dass die Berechnung der Parameter korrekt ist.

---

<sup>6</sup>Pkt. x(y); Abbildung: 7.14 (7.12)

Tabelle 7.6: Beispielpunkte, Roverkoordinaten und Nachführungsparameter<sup>6</sup>

Pkt. 2(4)	
$\varphi[^\circ]; \lambda[^\circ]; h[m]$	46.840856; 15.941179; 539
HDOP; VDOP	3; 4
Distanz $s_{RT}$ [m]	135.718
Panwinkel; Tiltwinkel [°]	-77.8; 1.3
Panwinkel; Tiltwinkel [ $ServoSTEP$ ]	2659; 2575
Panbuffer; Tiltbuffer [ $ServoSTEP$ ]	31; 221
Pkt. 3(8)	
$\varphi[^\circ]; \lambda[^\circ]; h[m]$	46.840981; 15.939230; 559
HDOP; VDOP	1.3; 1.5
Distanz $s_{RT}$ [m]	248.018
Panwinkel; Tiltwinkel [°]	-46.2; 5.3
Panwinkel; Tiltwinkel [ $ServoSTEP$ ]	2800; 2725
Panbuffer; Tiltbuffer [ $ServoSTEP$ ]	20; 50
Pkt. 4(13)	
$\varphi[^\circ]; \lambda[^\circ]; h[m]$	46.839828; 15.937188; 570
HDOP; VDOP	1.3; 1.5
Distanz $s_{RT}$ [m]	362.820
Panwinkel; Tiltwinkel [°]	-13.9; 5.4
Panwinkel; Tiltwinkel [ $ServoSTEP$ ]	2940; 2725
Panbuffer; Tiltbuffer [ $ServoSTEP$ ]	20; 50
Pkt. 5(19)	
$\varphi[^\circ]; \lambda[^\circ]; h[m]$	46.839147; 15.939213; 586
HDOP; VDOP	2.7; 2.3
Distanz $s_{RT}$ [m]	222.825
Panwinkel; Tiltwinkel [°]	5.4; 13.0
Panwinkel; Tiltwinkel [ $ServoSTEP$ ]	3020; 2875
Panbuffer; Tiltbuffer [ $ServoSTEP$ ]	20; 75
Pkt. 6(25)	
$\varphi[^\circ]; \lambda[^\circ]; h[m]$	46.838398; 15.941150; 582
HDOP; VDOP	1.7; 2.0
Distanz $s_{RT}$ [m]	167.548
Panwinkel; Tiltwinkel [°]	56.1; 15.9
Panwinkel; Tiltwinkel [ $ServoSTEP$ ]	3240; 2919
Panbuffer; Tiltbuffer [ $ServoSTEP$ ]	20; 86
Pkt. 7(32)	
$\varphi[^\circ]; \lambda[^\circ]; h[m]$	46.838030; 15.942429; 561
HDOP; VDOP	1.5; 1.6
Distanz $s_{RT}$ [m]	196.347
Panwinkel; Tiltwinkel [°]	89.0; 7.3
Panwinkel; Tiltwinkel [ $ServoSTEP$ ]	3380; 2752
Panbuffer; Tiltbuffer [ $ServoSTEP$ ]	20; 59
Pkt. 8(40)	
$\varphi[^\circ]; \lambda[^\circ]; h[m]$	46.839135; 15.941943; 538
HDOP; VDOP	1.3; 1.5
Distanz $s_{RT}$ [m]	68.081
Panwinkel; Tiltwinkel [°]	78.8; 1.7
Panwinkel; Tiltwinkel [ $ServoSTEP$ ]	3336; 2575
Panbuffer; Tiltbuffer [ $ServoSTEP$ ]	28; 159

# Kapitel 8

## Resümee und Ausblick

Ziel dieser Masterarbeit war es, den Prototypen eines ferngesteuerten Rovers mit der dazugehörigen Bodenstation für eine große Reichweite und unter Verwendung von „Low-Cost“-Komponenten zu entwickeln. Weitere Anforderungen an das Prototypensystem waren eine leichte Erweiterbarkeit aller Module, die Nutzung eines Autopilotsystems und die Verwendung von Open Source Programmierumgebungen.

Mit diesen Anforderungen an das System wurden zu Beginn der Arbeit diverse Autopilotsysteme, Kommunikationskomponenten und mechanische Bauteile verglichen und ausgewählt. Bei der Realisierung der Prototypen waren zusätzliche hardware- und softwaretechnische Komponenten nötig, die im Verlauf des Projekts neu entwickelt und angefertigt werden mussten.

Für die Datenübertragung des Autopilotsystems und die Positionierung der Antennennachführung wurde im Vorfeld ein Konzept der gängigen Nachführungen aus dem FPV-Sektor gewählt. Dieses Konzept diente als Ausgangspunkt für die Eigenentwicklung und blieb nach massiven Änderungen und Anpassungen lediglich in ihren Basisfunktionen bestehen. Die Berechnungsalgorithmen zur autonomen Nachführung der WLAN-Richtantenne wurden aus gängigen geographischen Berechnungen abgeleitet und in die Steuersoftware eingebunden.

Aufgrund intensiver Recherchen und der Planung schon vor Beginn der Bauphase kam es bei der Umsetzung der Hardware zu keinerlei Problemen. Durch die laufend durchgeführten Tests der einzelnen Komponenten in der Elektronik und der Software, konnten technische Probleme zum großen Teil behoben werden.

Bei der technischen Umsetzung des vollständigen Systems entstanden vier separate Module. Diese wurden dahingehend entwickelt, dass die einzelnen Module auch unabhängig voneinander betrieben werden können.

- Rover mit einem Ardupilot Mega Autopilotensystem und einer 40 MHz RC-Steuerung
- WLAN-Antennennachführung mit automer Positions- und Orientierungsbestimmung
- Bodenstation mit Übertragungs- und Steuerelektronik
- C/C++ Steuersoftware der Antennennachführung

Die entwickelten Prototypen können ohne umfangreiche Änderungen an der Hardware semi-autonom betrieben werden. Die Autopilotfirmware ist jedoch bei einer derartigen Verwendung an den Rover anzupassen. Bei den Simulationen und Feldtests zeigte sich, dass die Prototypen des Rovers, der Antennennachführung und der Steuersoftware gut zusammenarbeiten und dass das Berechnungskonzept der Nachführungsparameter als erfolgreich betrachtet werden kann. Stichhaltige Aussagen über die Genauigkeit und Präzision der Antennennachführung können jedoch nur weiterführende Versuche liefern. Auch die WLAN-Datenübertragung zwischen dem Rover und der Bodenstation wies bei den Tests weder bei der Bildübertragung noch beim Datenstream des Autopiloten Aussetzer oder Einbußen in der Durchsatzrate auf. Die Elektronik der Bodenstation zeigte bei den Versuchen keine Schwächen, jedoch wurde das Fehlen einer autonomen Stromversorgung bei den Tests als störend empfunden, da derzeit noch ein 230 V-Netzanschluss verwendet wird.

Bei den Testfahrten wurden jedoch zwei Probleme festgestellt, die im Verlauf dieser Arbeit noch nicht gelöst werden konnten und einer Weiterentwicklung bedürften. So zeigte sich, dass die erzielten Reichweiten mit der verwendeten 40 MHz RC-Steuerung des Rovers deutlich unter den erwarteten liegen. Diese Einbußen in der Reichweiten zeigten sich in der Form von Signalstörungen und Unterbrechungen außerhalb eines Radius von  $\approx 45 \text{ m}$ . Um einen Fehler in der PPM-Signalgenerierung, der selbst entwickelten Pegelanpassungsplatine und dem 40 MHz-Sendemodul auszuschließen, wurden Vergleichstests mit einem handelsüblichen RC-Handsender durchgeführt. Diese haben annähernd zu denselben Reichweiten geführt. Durch dieses Verhalten ist eine exakte Steuerung des Rovers in einem größeren Umkreis derzeit nicht möglich. Innerhalb dieses Bereichs reagiert der Kettenantrieb des Rovers sehr exakt auf die Steuerbefehle und erlaubt ein präzises Navigieren. Um jedoch Schäden am Roverantrieb durch unkontrolliertes Anlaufen der Motoren zu vermeiden, wurde auf weitere Tests außerhalb dieses Empfangsbereichs, bis zur Änderung der Steuerkomponenten, verzichtet. Ein Lösungsansatz wäre hier die Verwendung eines 2.4 GHz RC-Systems. Die Dimensionierung des Antriebs und die leistungsfähigen Akkus haben sich im Feldtest bewährt. Auch bei Vollastbetrieb zeigten sich keinerlei Überhitzungserscheinungen oder Leistungseinbrüche des Antriebs.

Die zweite Schwierigkeit trat bei der Orientierungsbestimmung der Nachführung auf, was auf die fehlerhafte Firmware der verwendeten Kompasselektronik zurückgeführt wurde. Es stellte sich bei den Versuchen heraus, dass der Kompass nicht auf die umgebenden Magnetfeldbeeinflussungen hin kalibriert werden kann. Mit dieser Elektronik war es nicht möglich eine Nord-

richtung zu bestimmen. Daher wurde für die Feldversuche die Ausrichtung der Nachführung mit einem Standardkompass durchgeführt. Die Bestimmung der „Home“-Position mit einem eigenen GPS-Modul auf der Nachführung ist dahingehend vorteilhaft, als dass die Nachführung auch für andere Zwecke zu verwenden ist. Auch die durch den Servoantrieb erreichte Auflösung der Dreh- und Neigungsachse mit  $0.230^\circ$  bzw. mit  $0.054^\circ$  ist für die Richtantenne mit einem Öffnungswinkel von  $28^\circ$  ausreichend. Es ist jedoch bei der Auflösungsbestimmung ein Umkehrspiel im Kettenantrieb der Neigungsachse zu erkennen. Durch die Verwendung eines Zahnriemenantriebs als Kraftübertragung könnte dieses Umkehrspiel vermieden werden.

Trotz der, bei den Versuchen und der Probefahrt aufgetretenen Schwierigkeiten, kann die Entwicklung der Prototypen als Erfolg beschrieben werden. Für ein voll operationelles System sind die einzelnen Module in einigen Punkten zu überarbeiten bzw. zu verbessern. Im Laufe dieser Masterarbeit sind für diese Überarbeitungen und Verbesserungen schon einige Lösungsansätze entstanden.

### **Verbesserungen und Empfehlungen:**

- *2.4 GHz RC-Komponenten:* Dieser Ansatz kann zur Lösung des Problems mit der Steuerreichweite herangezogen werden. Dabei sind der Empfänger im Rover und das Sendemodul in der Bodenstation auszutauschen. Durch Versuche zeigt sich dann, ob das Problem der geringen Reichweite beseitigt wurde.
- *Rover-Autopilot:* Für dieses Projekt wurde eine Standardfirmware verwendet, da alle benötigten Komponenten vorhanden waren. Soll der Rover zukünftig im semiautonomen Modus betrieben werden, muss die Firmware an das verwendete Fahrzeug angepasst werden.
- *Rover-Laufwerk:* Bei den Testfahrten hat sich gezeigt, dass das Kettenlaufwerk sehr stabil aber auch kaum eine Federung aufweist. Dadurch werden die Bewegungen des Laufwerks fast direkt an die Trägerplattform übertragen. Es wird für die empfindliche Elektronik empfohlen, das Laufwerk neu zu designen und dieses mit einer weicheren Federung zu versehen.
- *Rover-Kamera:* Es hat sich herausgestellt, dass sich die verwendete „Low-Cost“-LAN-Kamera nicht für das Steuern des Rovers eignet, da die Reaktionsgeschwindigkeit nicht ausreichend ist. Hier ist ein Austausch der Kamera notwendig, um einen „flüssigeren“ Bildverlauf zu erreichen.
- *Nachführung-Kompass:* Das verwendete Kompassmodul CMPS10 hat sich als unpraktikable Lösung für das Projekt herausgestellt. Hier wird die Verwendung eines Kompassmoduls empfohlen, das die Rohdaten an den Computer schickt. Die Berechnung der Orientierungsparameter sind anschließend im Hauptprogramm durchzuführen.

- *Bodenstation-Spannungsversorgung*: Die Verwendung von 230 VAC hat sich für die Prototypenentwicklung bewährt, ist jedoch für den Feldeinsatz oft nur schwer zu realisieren. Daher wäre die Erweiterung um einen 12 VDC-Eingang von Vorteil.
- *Software-Oberfläche*: Die Verwendung der Konsoleneingabe für die Software hat sich durch die zahlreichen Einstellmöglichkeiten als zeitaufwändig und unübersichtlich erwiesen. Die Programmierung einer grafischen Benutzeroberfläche würde die Übersichtlichkeit der Programmooptionen enorm steigern.
- *Software-Joysticksteuerung*: Die derzeitige Steuerung der RC-Komponenten erfolgt mit einem Open Source Java Programm. Bei der Konvertierung dieses Codes in C/C++ und der Implementierung in das Hauptprogramm könnte das vollständige System auf einem „embedded system“ (z.B.: Raspberry) betrieben werden. Dadurch könnte die Baugröße der Bodenstation verkleinert und damit portabler gestaltet werden.

Meiner Meinung nach liegt das größte Potential dieses Systems in der Kombination des Lastrovers, des Open Source Autopilot und der Bodenstation. Durch die modulare Bauweise könnten sich zukünftig auch spezielle Anwendungsbereiche aus der Geodäsie ergeben. Denkbar wären hier das Testen von Sensoren, GNSS-Systemen oder auch Geländeaufnahmen im autonomen Modus.

Da die Prototypen auch nach Abschluss dieser Arbeit privat weiterentwickelt und verbessert werden, soll ein einfach zu handhabendes, leicht erweiterbares und operationelles System entstehen. Der Rover soll zukünftig als semiautonomer Versuchsträger für verschiedenste Sensorsysteme dienen und einen Echtzeitdatenstream zur Bodenstation ermöglichen.

Abschließend bleibt zu sagen, dass das Prototypensystem durch die Verwendung von günstigen Komponenten voll funktionsfähig entwickelt und gebaut wurde und die Material- und Entwicklungskosten 4000 Euro nicht überschritten.



# Kapitel 9

## Technische Übersicht



Abbildung 9.1: Antennennachführung

Tabelle 9.1: Kerndaten Antennennachführung

LxBxH (ohne Antenne)	200x200x300	mm
LxBxH (mit Antenne)	200x500x380	mm
Gewicht	4	kg
Pan Wirkbereich	$\pm 90$	$^{\circ}$
Pan Auflösung	0.229 ( $\pm 0.054$ )	$^{\circ}/Step$
Tilt Wirkbereich	$-30 \leq \leq +60$	$^{\circ}$
Tilt Auflösung	0.042 ( $\pm 0.003$ )	$^{\circ}/Step$
Anschlüsse	2xRS-232 2xUSB 1xWLAN	
Geräte	GPS-Empfänger Elektronischer Kompass Pan Servo 2xTilt Servo Airwin Antenne 16dBi Webcam	

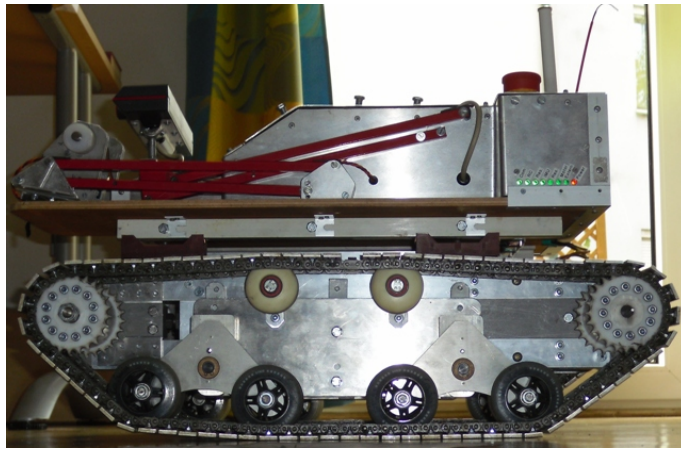


Abbildung 9.2: Rover

Tabelle 9.2: Kerndaten Rover

LxBxH	600x500x300	mm
Gewicht	45	kg
Geschwindigkeit	5	km/h
Antriebsleistung	2x250	W
Spannungsversorgung	4x12 (12)	VDC (Ah)
Autopilot	Ardupilot Mega 2.0 3Achs Gyro 3Achs Acc 3Achs Magnetometer Temperatur Barometer diff. Drucksensor Spannungssensor Stromsensor	
Kamera	Ubiquiti AirCam	
RC	2x25A Motorregler 40MHz-Empfänger	
Datenübertragung	Microtik RB433 Airwin 5dBi Lantronix Xport	



Abbildung 9.3: Bodenstation

Tabelle 9.3: Kerndaten Bodenstation

LxBxH	600x400x150	mm
Gewicht	12	kg
Spannungsversorgung	230	VAC
Anschlüsse	Kaltgerätebuchse	
	RJ-45 Buchse	
	2xRS-232	
	1xUSB	
	RC-Antenne WLAN	
Geräte	USB-Hub	
	RS-232-Hub	
	40MHz-Sendemodul	
	Pegelanpassungsplatine	
	USB2PPM Generator	
	TTL2RS-232 Konverter	
	Mikrotik RB433	
	2xNetzgeräte	
Joystick		

# Tabellenverzeichnis

3.1	Signalpegel in Volt . . . . .	30
4.1	Technische Daten je Laufkette und Kettenräder . . . . .	44
4.2	Spannungen Elektronikmodule . . . . .	52
4.3	Gleichspannungswandler Rover . . . . .	52
4.4	Autopilot Elektronikmodule . . . . .	54
4.5	Anschlüsse Bodenstation . . . . .	56
4.6	Gleichspannungswandler Bodenstation . . . . .	59
4.7	Spannungen Elektronikmodule . . . . .	59
4.8	CMPS10 Herstellerangaben . . . . .	67
4.9	Kommandosequenzen CMPS-10 Kompassmodul . . . . .	68
4.10	Richtantenne Spezifikationen . . . . .	69
4.11	40MHz-Anschlussmodul Pinbelegung . . . . .	72
5.1	Servocontroller Parameter . . . . .	86
5.2	Rover Parameter . . . . .	86
5.3	Rover Vorgaben . . . . .	87
5.4	LED-Vorwiderstandswerte ( $I_{LED} \approx 20 \text{ mA}$ ) . . . . .	88
6.1	Software Hierarchie Beschreibung . . . . .	95
6.2	Beispiel: Setzen einer absoluten Position . . . . .	99

---

7.1	Berechnungsindizesse . . . . .	107
7.2	Koordinaten der Messpfeiler . . . . .	110
7.3	Pfeiler2 Ausreißerbefreite Mittelwerte . . . . .	112
7.4	Pfeiler7 Ausreißerbefreite Mittelwerte . . . . .	112
7.5	Simulationseinstellungen „Home“-Position . . . . .	120
7.6	Beispielpunkte, Roverkoordinaten und Nachführungsparameter <sup>1</sup> . . . . .	122
9.1	Kerndaten Antennennachführung . . . . .	127
9.2	Kerndaten Rover . . . . .	128
9.3	Kerndaten Bodenstation . . . . .	129

# Abbildungsverzeichnis

2.1	Autopilot Mega <sup>2</sup> . . . . .	5
2.2	pixhawk-PX4 <sup>3</sup> . . . . .	5
2.3	Kommunikationsdesign 1 . . . . .	6
2.4	XBee WLAN-Modul <sup>4</sup> . . . . .	7
2.5	XPort-Modul <sup>5</sup> . . . . .	7
2.6	Kommunikationsdesign 2 . . . . .	7
2.7	PPM-Signalgenerator der Fa. Flydron <sup>6</sup> . . . . .	9
2.8	Radchassis nach UlrichC <sup>7</sup> . . . . .	10
2.9	Kettenschassis nach UlrichC <sup>8</sup> . . . . .	10
2.10	Design des Chassis mit Trägerplattform . . . . .	11
2.11	a) aussenliegende Montagekonsole; b) innenliegende Montagekonsole . . . . .	12
2.12	Mechanikkonzept . . . . .	13
2.13	Antennentracking . . . . .	14
2.14	Konzept Bodenstationschassis . . . . .	15
2.15	Konzeptaufbau Software . . . . .	16
2.16	Konzeptaufbau Kommunikation . . . . .	17
3.1	Blockdiagramm Inertial navigation system <sup>9</sup> . . . . .	19
3.2	Berechnungsschema Kalman Filter Algorithmus <sup>10</sup> . . . . .	20
3.3	Berechnungsschema DCM-Komplementär Filter Algorithmus <sup>11</sup> . . . . .	20

3.4	Amplitudenmodulation <sup>12</sup> . . . . .	23
3.5	Frequenzmodulation <sup>13</sup> . . . . .	24
3.6	Negatives PPM-Frame <sup>14</sup> . . . . .	25
3.7	2.4 GHz Ruhepegel <sup>15</sup> . . . . .	26
3.8	2.4GHz DSSS-Pegel <sup>16</sup> . . . . .	27
3.9	2.4GHz FHSS-Pegel <sup>17</sup> . . . . .	27
3.10	2.4GHz PCM-Signalweg <sup>18</sup> . . . . .	28
3.11	RS-232 Nullmodemverbindung <sup>19</sup> . . . . .	30
3.12	Signal und Logikpegel: a) RS-232 vs., b) TTL, LVTTTL . . . . .	31
3.13	Seriellles Datenpaket <sup>20</sup> . . . . .	31
3.14	Unidirektionales Datenpaket . . . . .	35
3.15	Bidirektionales Datenpaket . . . . .	36
3.16	Diversity Antennentracking . . . . .	37
3.17	a) Drehachsenkomponente, b) Neigungsachsenkomponente . . . . .	38
3.18	Telemetrie Auswerteeinheit AV-Signalübertragung <sup>21</sup> . . . . .	38
3.19	Antriebskomponenten Antennennachführung . . . . .	39
3.20	Antriebskomponenten Rover . . . . .	40
4.1	Modell des Roverchassis . . . . .	44
4.2	Modell einer Kettenplatte mit Rollenkettenglieder . . . . .	44
4.3	Schnittmodelle der hinteren Antriebsachse . . . . .	45
4.4	Schnittmodelle der vorderen Kettenachse . . . . .	46
4.5	a) Antriebsmotor; b) Motorkennlinien . . . . .	46
4.6	a) Antriebskomponenten; b) montierte Antriebseinheit . . . . .	47
4.7	vollständiger Roverantrieb . . . . .	47
4.8	a) Geschlossene Wartungsklappen; b) Offene Wartungsklappen . . . . .	48

4.9 a) Sensorturm; b) Navigationselektronik; c) Staudrucksonden . . . . . 48

4.10 Elektronik Schutzrahmen . . . . . 49

4.11 a) Anschlüsse und Not-Aus; b)Rover Bedienterminal . . . . . 50

4.12 Schaltplan Akkupack . . . . . 51

4.13 Stromlaufplan Energiekreis . . . . . 51

4.14 Schutzrahmen mit Elektronikkomponenten . . . . . 52

4.15 Anschlusschema Rover-Elektronik . . . . . 53

4.16 Stromlaufplan Roversensoren . . . . . 54

4.17 a) 3D-Konstruktionsplan Bodenstation; b) Ansicht Kammern . . . . . 55

4.18 Bodenstation Anschlussterminal . . . . . 56

4.19 Stromlaufplan 230 V . . . . . 57

4.20 Grundschtung LM317 Step-Down Wandler . . . . . 57

4.21 Spannungswandler und Anschlusschema Spannungsverteiler . . . . . 58

4.22 a) Stromlaufplan; b) Einzelplatinen Spannungsverteiler . . . . . 58

4.23 Antennennachführung . . . . . 60

4.24 a) Drehachsmodul; b) Neigungsachsmodul . . . . . 61

4.25 a) Lenkgabel; b) Lenkkopflager<sup>22</sup> . . . . . 61

4.26 3D-Konstruktionsplan Pan-Modul . . . . . 62

4.27 3D-Konstruktionsplan Tilt-Lagereinheit . . . . . 62

4.28 3D-Konstruktionsplan Tilt-Modul . . . . . 63

4.29 3D-Konstruktionsplan Antennennachführung . . . . . 63

4.30 a) Servo-Inverter; b) Neigungsachsenservo Hitec HS-645MG; c) Drehachsenservo Modelcraft RS-10<sup>23</sup> . . . . . 64

4.31 Pololu 8-Kanal Servocontroller<sup>24</sup> . . . . . 65

4.32 Antennennachführung Antriebsschaltplan . . . . . 65

4.33 a) MiniSSC II Protokollstruktur; b) Pololu-Mode Protokollstruktur<sup>25</sup> . . . . . 66



4.34 a) Devantech CMPS10 Kompassmodul; b) Navilock NL-302U GPS-Empfänger <sup>26</sup> . . . . .	66
4.35 Transformationswinkel Nordrichtung <sup>27</sup> . . . . .	67
4.36 CMPS10 Kompassmodul Anschlussplan . . . . .	67
4.37 NMEA-0183 Datenprotokoll <sup>28</sup> . . . . .	68
4.38 a) Airwin Richtantenne <sup>29</sup> ; b) Antennenkonsole . . . . .	69
4.39 Polarplot Richtantenne <sup>30</sup> . . . . .	70
4.40 Kommunikations Konzept . . . . .	70
4.41 40 MHz-Anschlussmodul . . . . .	71
4.42 Anschlussschema 40MHz-Sendermodul . . . . .	72
4.43 Simprop Pico2000 Empfänger . . . . .	72
4.44 Dualer Motorregler <sup>31</sup> . . . . .	73
4.45 a) LAN-Board; b) WLAN-Card <sup>32</sup> . . . . .	74
4.46 Anschlussschema Rover-Kommunikation . . . . .	74
4.47 a) XstreamBee Board; b) XPort zu XBee Konverter; c) XPort <sup>33</sup> . . . . .	75
5.1 Antennennachführung Berechnungsschema . . . . .	76
5.2 Buffer der Drehachse . . . . .	82
5.3 Buffer der Neigungsachse . . . . .	82
5.4 Buffer in Abhängigkeit: Distanz, DOP, Servoauflösung: a) DOP in Meter; b) Bufferbereich in Grad; c) Bufferbereich in abs. Servoschritten . . . . .	83
5.5 3D Winkelfehler: Distanz, DOP . . . . .	84
6.1 APM Planner Mav1.0 . . . . .	90
6.2 Kompatible Prozessoren für Arduino Software . . . . .	90
6.3 QGroundControl . . . . .	91
6.4 a) CPR-Manager; b) XPort-Device Server; c) Lantronix DeviceInstaller . . . . .	92
6.5 a) „com0com“-Emulator; b) COM-Port Splitter (VSPE) . . . . .	92

6.6 CompuFly von Flytron . . . . . 93

6.7 Software Hierarchiestufen . . . . . 95

6.8 a) GPGGA- und; b)GPGSA-Datensatz aus NMEA-Protokol<sup>34</sup> . . . . . 97

6.9 MAVLink Frame<sup>35</sup> . . . . . 98

6.10 Pololu Protocol<sup>36</sup> . . . . . 99

6.11 Softwareschema Initialisierung . . . . . 101

6.12 Softwareschema Laufzeit . . . . . 102

6.13 a) Geräteauswahl; b) Verbindungstest . . . . . 103

6.14 a) Ausgabeoptionen; b) Grundeinstellungen Antennennachführung . . . . . 103

6.15 a) Verbundene Geräte; b) Grundeinstellungen Kompassinitialisierung . . . . . 104

6.16 a) Echtzeit Datenausgabe; b) VTK-Ergebnisplot . . . . . 104

7.1 Test- und Versuchsgelände<sup>37</sup> . . . . . 106

7.2 Messschema Neigungsachsenauflösung . . . . . 107

7.3 oben) Neigungsachse: Messwerte, Grad; unten) Drehachse: Messwerte, Grad . . . . . 108

7.4 a) Auflösung Neigungsachse; b) Auflösung Drehachse . . . . . 109

7.5 Messpositionen zur GPS-Validierung<sup>38</sup> . . . . . 110

7.6 Differenzen der Einzelmessungen zur Referenzposition . . . . . 111

7.7 Satellitensichtbarkeit zur Messzeit . . . . . 111

7.8 a) Messwerte absolut zur Referenz; b) Mittelwert relativ zur Referenz . . . . . 112

7.9 Magnetisch Nord Deklination<sup>39</sup> . . . . . 113

7.10 Missweisung durch einen Gewindebolzen . . . . . 114

7.11 Systemtest Aufbau . . . . . 115

7.12 Trajektorie für Nachführungssimulation<sup>40</sup> . . . . . 118

7.13 Höhen- und DOP-Profil für Nachführungssimulation . . . . . 119

7.14 Beispielpunkte mit Nachführungswinkel<sup>41</sup> . . . . . 120

7.15 Software Nachführungseinstellungen Simulation . . . . . 121

9.1 Antennennachführung . . . . . 127

9.2 Rover . . . . . 128

9.3 Bodenstation . . . . . 129

# Literaturverzeichnis

- BERNHARD HOFMANN-WELLENHOF, KLAUS LEGAT, & MANFRED WIESER. 2003. *Navigation*. SpringerWienNewYork.
- BERNHARD HOFMANN-WELLENHOF, HERBERT LICHTENEGGER, & ELMAR WASLE. 2008. *GNSS Global Navigation Satellite Systems*. SpringerWienNewYork.
- DAVID TITTERTON, & JOHN WESTON. 2004. *Strapdown Inertial Navigation Technology*. IET.
- DEVANTECH. 2012. *CMPS10-Tilt Compensated Compass Module*. Robot Electronics.
- DIMENSIONENGINEERING. 2007 (7). *Sabertooth 2x25 User's Guide*.  
<http://www.dimensionengineering.com/datasheets/Sabertooth2x25v2.pdf>.
- DIY DRONES. 2013. *Ardupilot Mega Wiki*.  
<http://code.google.com/p/ardupilot-mega/wiki/home>.
- ERICK MACIAS, DANIEL TORRES, & SOURABH RAVINDRAN. 2012. *Nine-Axix Sensor Fusion Using the Direction Cosine Matrix Algorithm on the MSP430F5xx Family*.  
<http://www.ti.com/lit/an/slaa518a/slaa518a.pdf>.
- ERWIN BÖHMER. 1998. *Elemente der angewandten Elektronik*. vieweg.
- FRIEDRICH BRUNNER, & ANDREAS WIESER. 2009. *Ingenieurgeodäsie*. Lehrveranstaltungsskriptum.
- FRIEDRICH WITTGRUBER. 2006. *Digitale Schnittstellen und Bussysteme*. Vol. 2. vieweg.
- GUOWEI CAI, TONG HENG LEE, & BEN M. CHEN. 2011. *Unmanned Rotorcraft Systems*. Springer.
- GUSTAV NIEMANN, & HANS WINTER. 2003. *Maschinenelemente Band2: Getriebe allgemein, Zahnradgetriebe - Grundlagen, Stirnradgetriebe*. Springer.
- HANS-JÜRGEN GEVATTER, & ULRICH GRÜNHAUPT. 2006. *Handbuch der Mess- und Automatisierungstechnik in der Produktion*. Springer.
- HANS-PETER MESSMER, & KLAUS DEMBOWSKI. 2003. *PC-Hardwarebuch*. Vol. 7. Addison-Wesley.
- JÖRG SCHAMUHN. 2008. 2,4GHz und die Geheimnisse einer neuen Technologie. *Flugmodell und technik*, **3**(3), 140–143.
- KLAUS BEUTH. 2003. *Bauelemente Elektronik*. Vogel.
- KLAUS DEMBOWSKI. 2007. *Lokale Netze Handbuch der kompletten Netzwerktechnik*. Addison-Wesley.
- KLAUS SCHMARANZ. 2002. *Softwareentwicklung in C*. Springer.

- LAMMERT BIES. 2010. *Interfacing*.  
<http://www.lammertbies.nl/comm/index.html>.
- LOTHAR SACHS, & JÜRGEN HEDDRICH. 2006. *Angewandte Statistik*. Springer.
- MANFRED DIETER KOTTING. 2008. 2,4GHz-Systeme. *Flugmodell und technik*, **7**(7), 140–144.
- MARK A. DYE, RICK McDONALD, & ANTOON W. RUFI. 2008. *Netzwerkgrundlagen-CCNA Exploration Companion Guide*. Addison-Wesley.
- MOHAMED OUBBATI. 2007. *Robotik*. Institut für Neuroinformatik, Universität Ulm.
- NARAIN GEHANI, & WILLIAM D. ROOME. 1993. *The concurrent C programming language*. Psilicon Press.
- NATIONAL SEMICONDUCTOR. 2011 (02). *LM317 3-Terminal Adjustable Regulator*.  
<http://www.ee.buffalo.edu/courses/elab/LM117.pdf>.
- NAVILOCK. 2013. *Navilock NL-302U USB GPS Empfänger*.  
[http://www.navilock.de/produkte/G\\_61422/merkmale.html](http://www.navilock.de/produkte/G_61422/merkmale.html).
- POLOLU. 2006. *Serial 8-Servo Controller User's Guide*. Pololu.
- RALF GESSLER, & THOMAS KRAUSE. 2009. *Wireless Netzwerke für den Nahbereich*. Vieweg+Teubner.
- RC-HELI-FAN WIKI. 2012. *Fernsteuerung*.  
<http://wiki.rc-heli-fan.org/index.php/Fernsteuerung>.
- ROLAND PAIL. 2007. *Parameterschätzung*. Lehrveranstaltungsskriptum.
- STARLINO. 2001. *DCM Tutorial-An Introduction to Orientation Kinematics*.  
[http://www.starlino.com/dcm\\_tutorial.html](http://www.starlino.com/dcm_tutorial.html).
- STMICROELECTRONICS. 2010. *Using LSM303DLH for a tilt compensated electronic compass*. Tech. rept. STMicroelectronics.
- TALAT OZYAGCILAR. 2012. *Implementing a Tilt-Compensated eCompass using Accelerometer and Magnetometer Sensors*. Tech. rept. Freescale Semiconductor.
- WILLIAM PREMERLANI, & PAUL BIZARD. 2009. *Direction Cosine Matrix IMU: Theory*.  
<http://gentlenav.googlecode.com/files/DCMDraft2.pdf>.

# Anhang

Alle im Projekt angefertigten Konstruktionszeichnungen entstanden mit einer Schulungsversion (**Autocad 2012** und **2013**) von **Autodesk**. Für die 3D-Graphiken wurde ebenfalls die Software von **Autodesk** (**Inventor Fusion 2012** und **2013**) verwendet.

Schaltpläne und Stromlaufpläne sind mit dem Freeware Programm **SPlan 7.0** von **Abacom** gezeichnet. Für Elektronische Schaltungen kam auch das Freeware Programm **Target 3001!** von **PCB** zum Einsatz.

Die zahlreichen Schematas entstanden mit der **Microsoft** Software **Visio 2007**.

Als Programmierumgebung wurde die **Software Visual Studio 2008** von **Microsoft** verwendet.

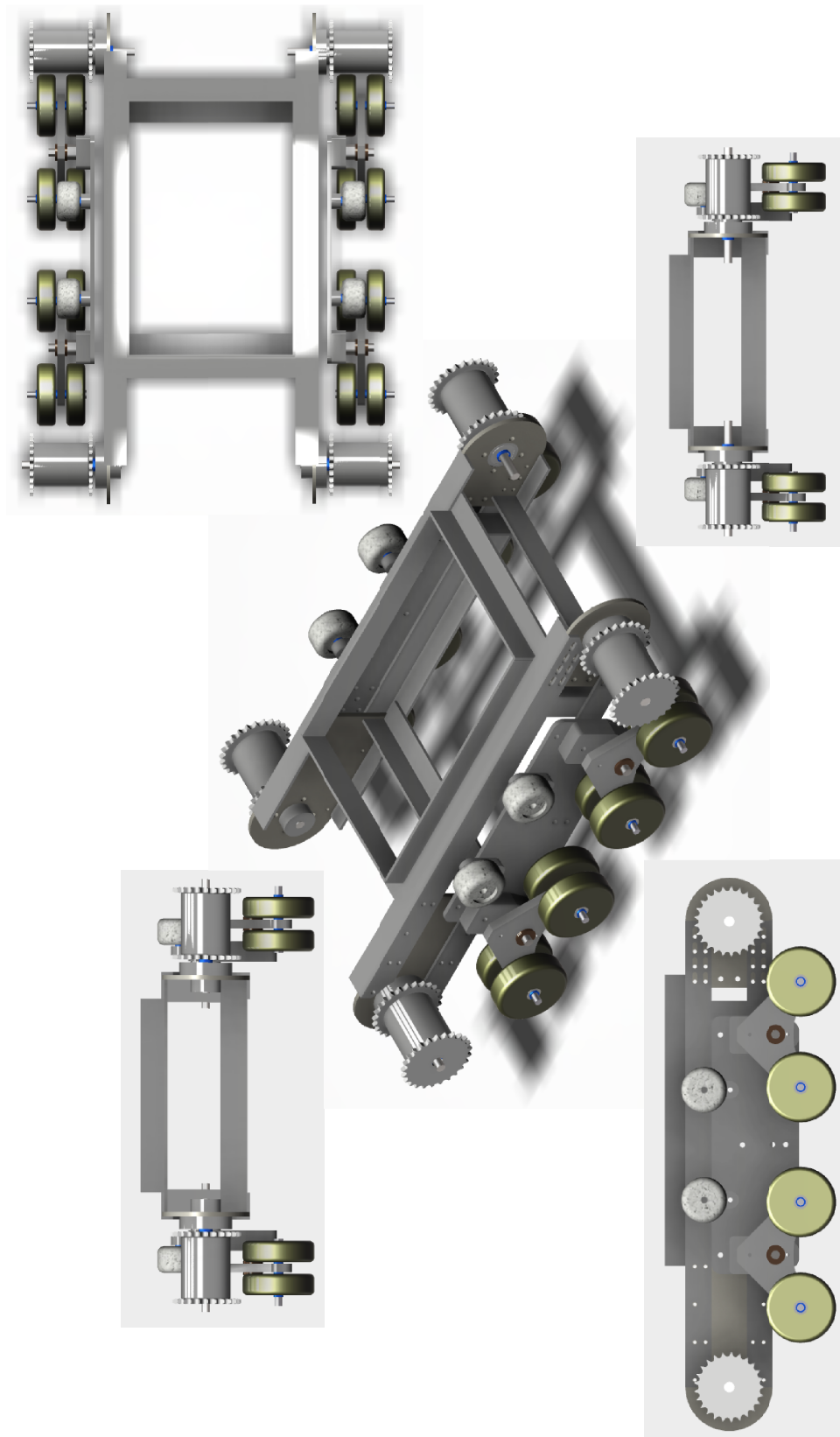
Die Source Code Dokumentation entstand mit dem Freeware Programm **Doxygen**.

Die im Anhang abgebildeten Baupläne, Schaltpläne und Schematas zeigen einen Großteil der verwendeten Unterlagen für die Mechanik, Elektrik und Elektronik. Der Basisplan des Rovers ist, wie bereits angemerkt, unter [www.ulrichc.de](http://www.ulrichc.de) zu finden. Die Baupläne für den Rover beinhalten lediglich die geänderten Konstruktionszeichnungen.

Die umfangreiche Sammlung an Softwaredokumentationen, Datenblätter, Manuals, Messdaten und Fotos sind auf der beigelegten DVD abgelegt. Auch der Source Code der verschiedenen Softwareversionen und diverse Hilfsprogramme befinden sich auf dieser DVD.

## A Baupläne

### A.1 Rover



**10mm Alu**

**20mm Alu**

**4mm Alu**

Zust.	Änderung	Datum	Name	Urspr.:	Datum	Name	Urspr.:	Benennung	
								Kettenplatten, Kippschwinge, Kippschwingenaufnehmer	
								Zeichnungsnummer	Blatt
								Ers. f.:	Bl.
								Ers. d.:	








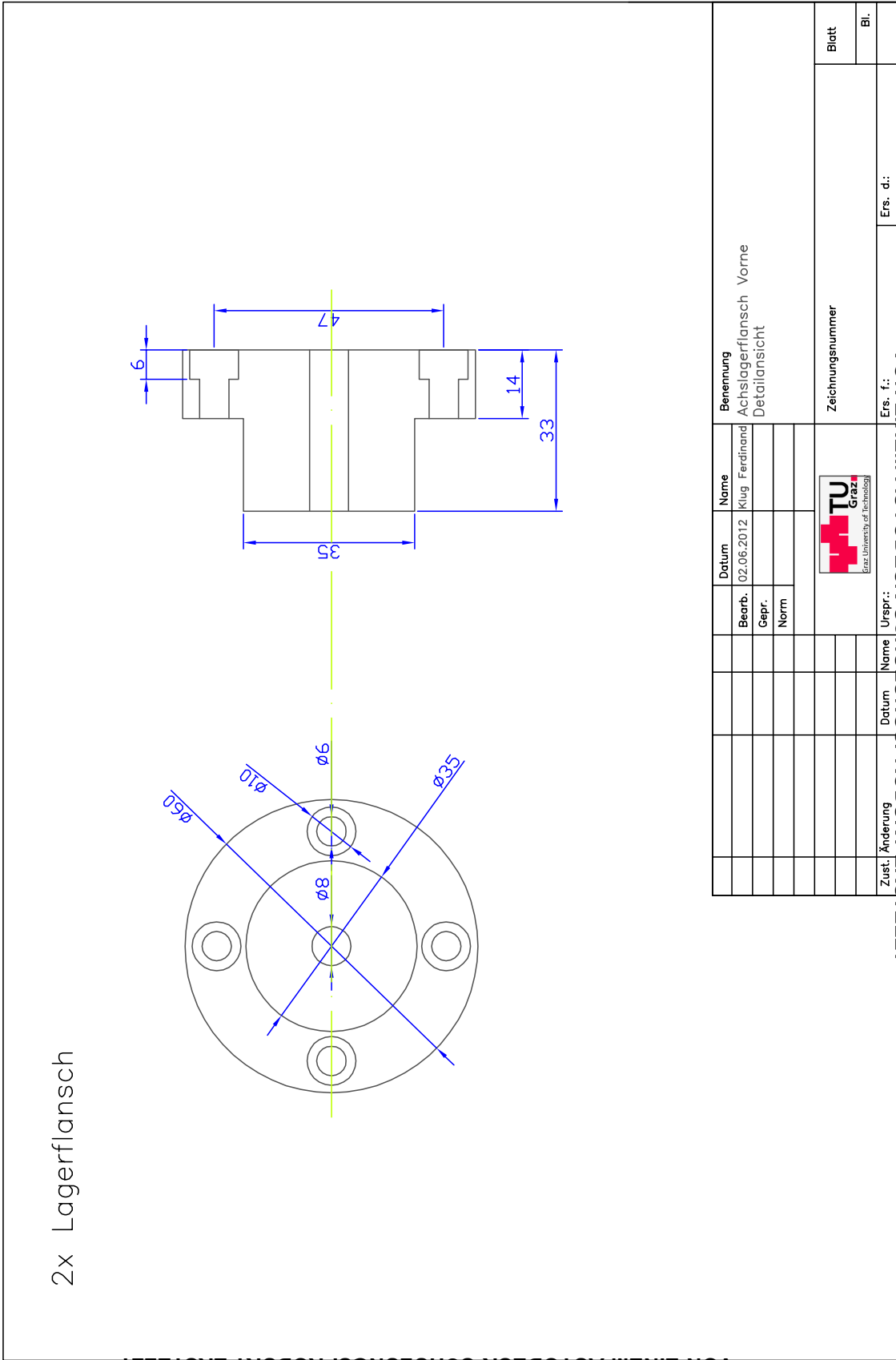


The drawing shows a chain drive assembly with the following components labeled from left to right:

- Achslagerflansch vorne**: Front axle bearing flange.
- Spacer**: A small cylindrical component between the bearing flange and the chainring.
- Kettenrad**: The front chainring.
- Lager 6001 –SKF**: A ball bearing supporting the chainring.
- Kettenspacer**: A larger central component that houses the bearings and the chainring.
- Kettenrad**: The rear chainring.
- Lager 6001 –SKF**: A ball bearing supporting the rear chainring.
- Kettenrad**: The rear chainring.

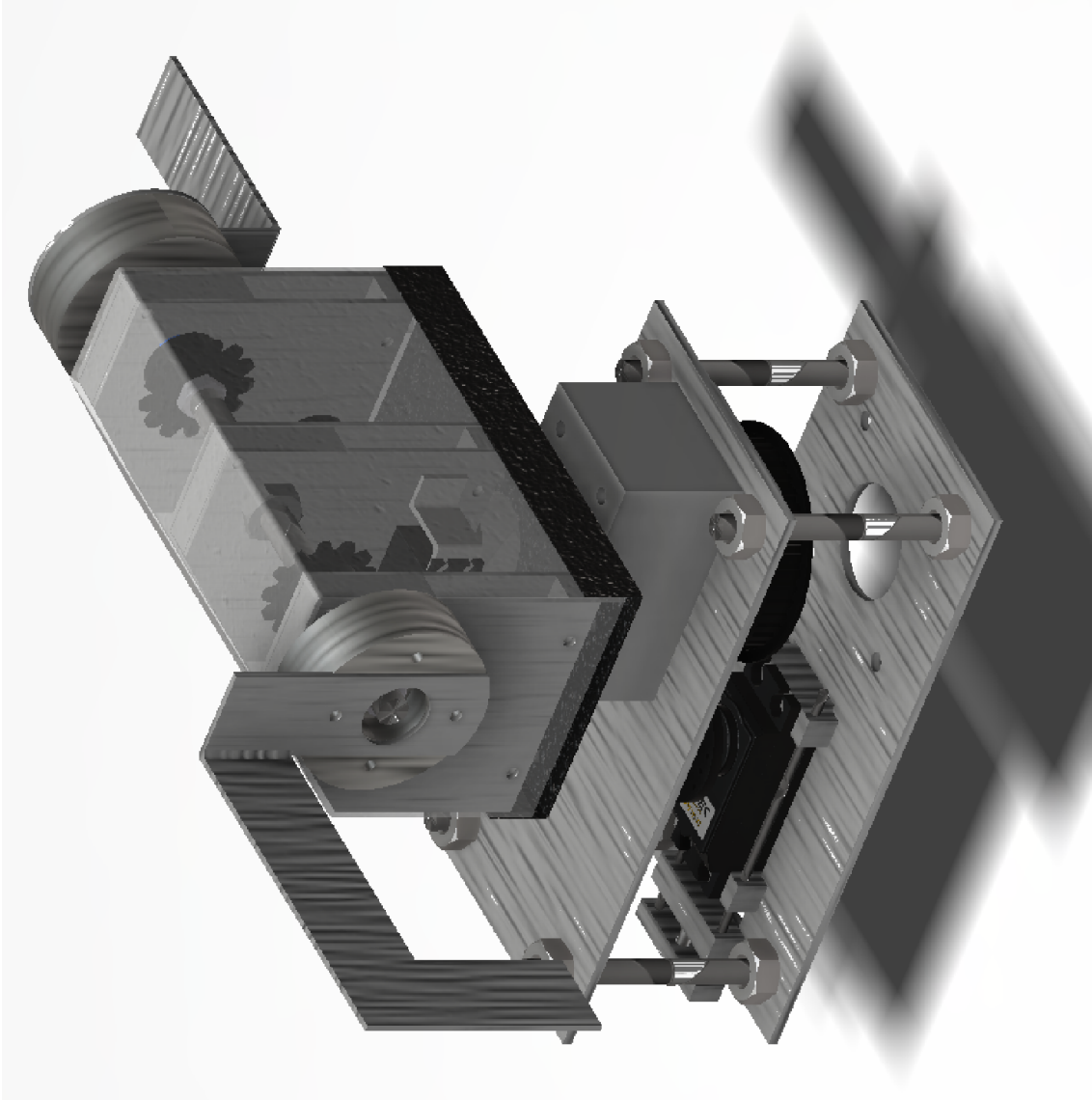
The drawing is a technical drawing with a vertical centerline, indicating it is a half-view of a symmetric assembly. The components are shown in perspective and cross-section views.

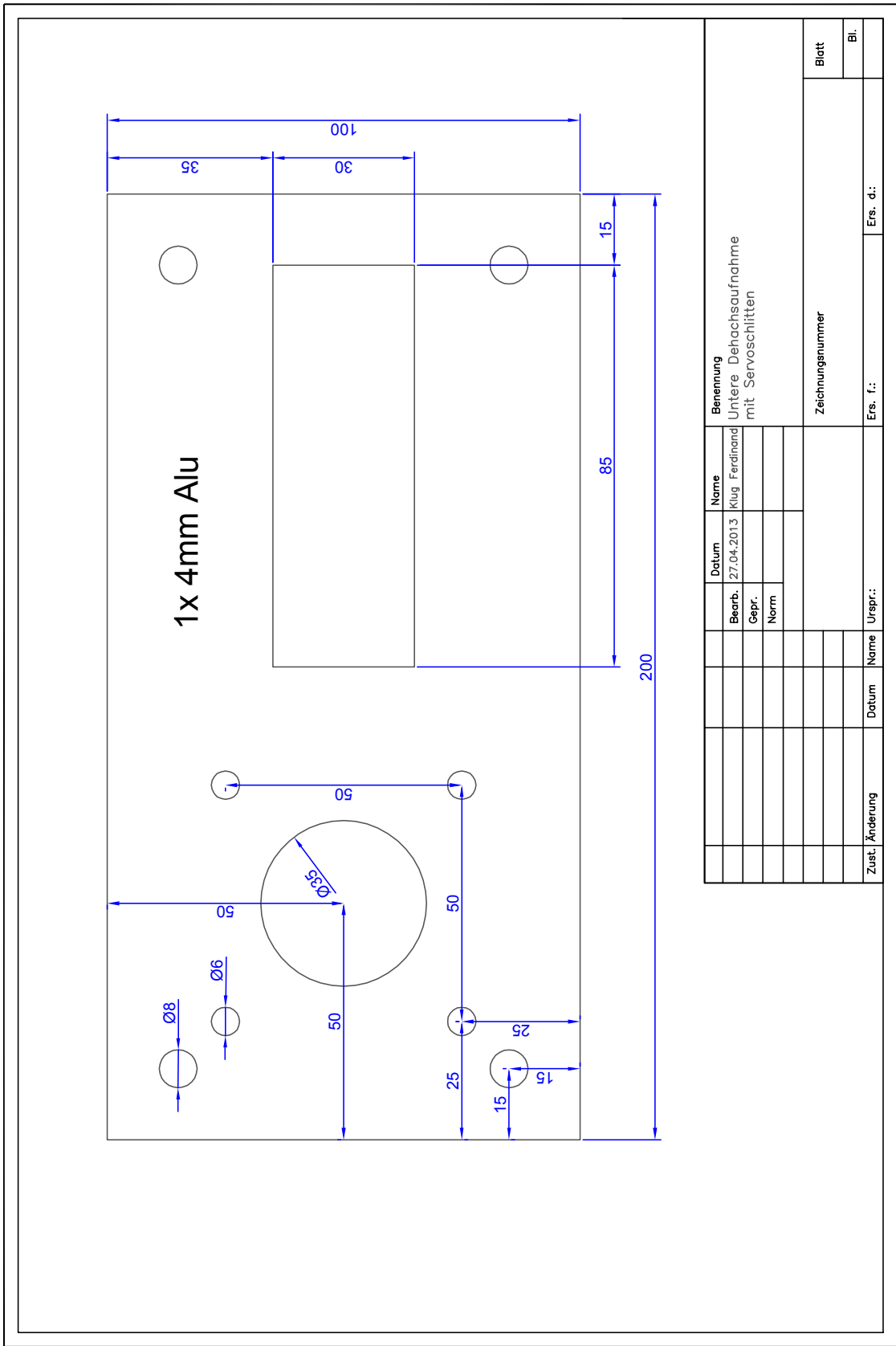
Zust.	Änderung	Datum	Name	Urspr.:	Ers. f.:	Ers. d.:	Blatt
							Blatt
				 TU Graz Graz University of Technology		Blatt	
				Benennung		Bl.	
				Datum		Ers. d.:	
				Name		Ers. f.:	
				02.06.2012		Zusammenbau Einzelteile	
				Klug, Ferdinand		Antrieb vorne	
				Bearb.			
				Gepr.			
				Norm			





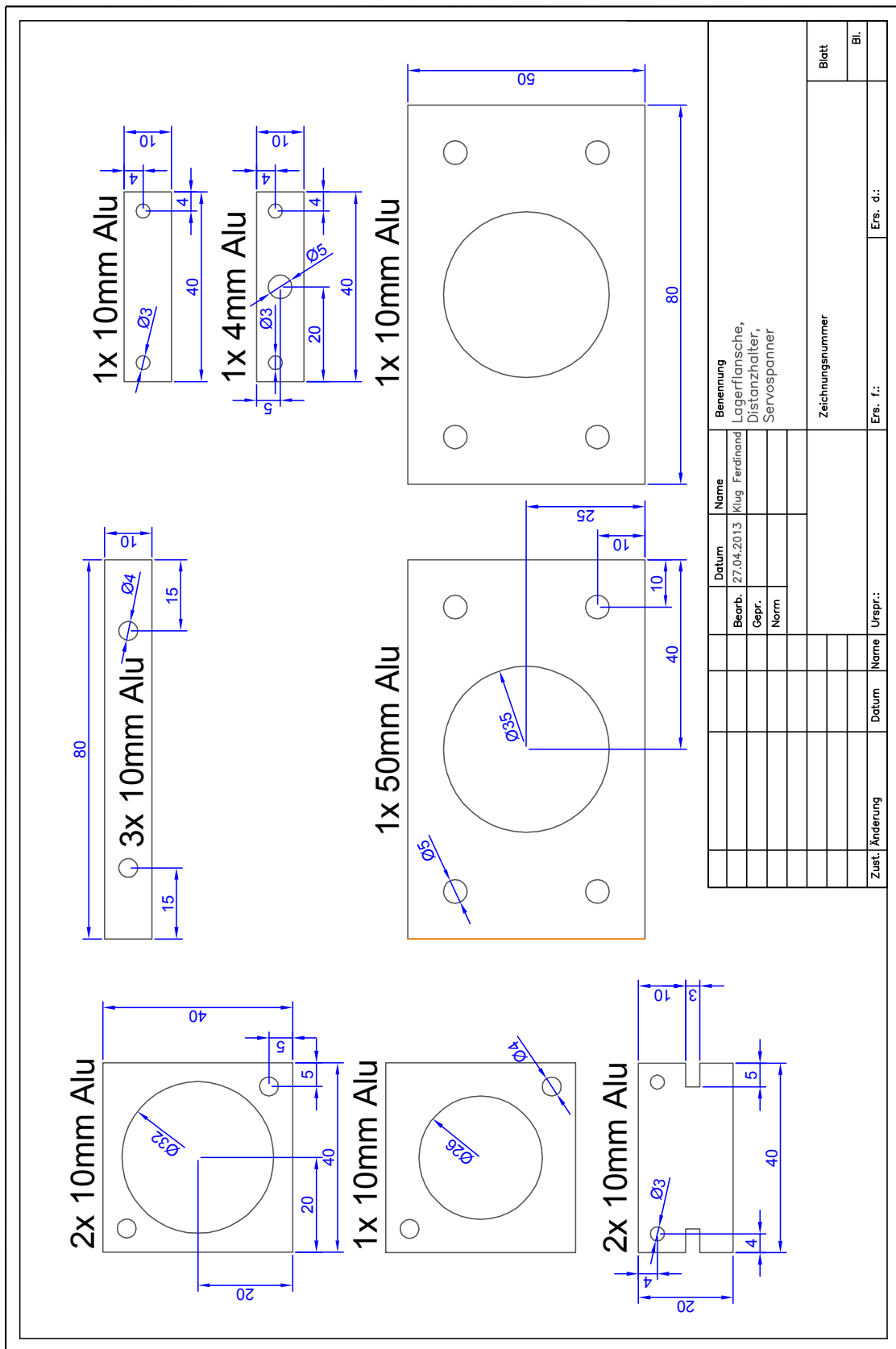
A.2 Antennentracker



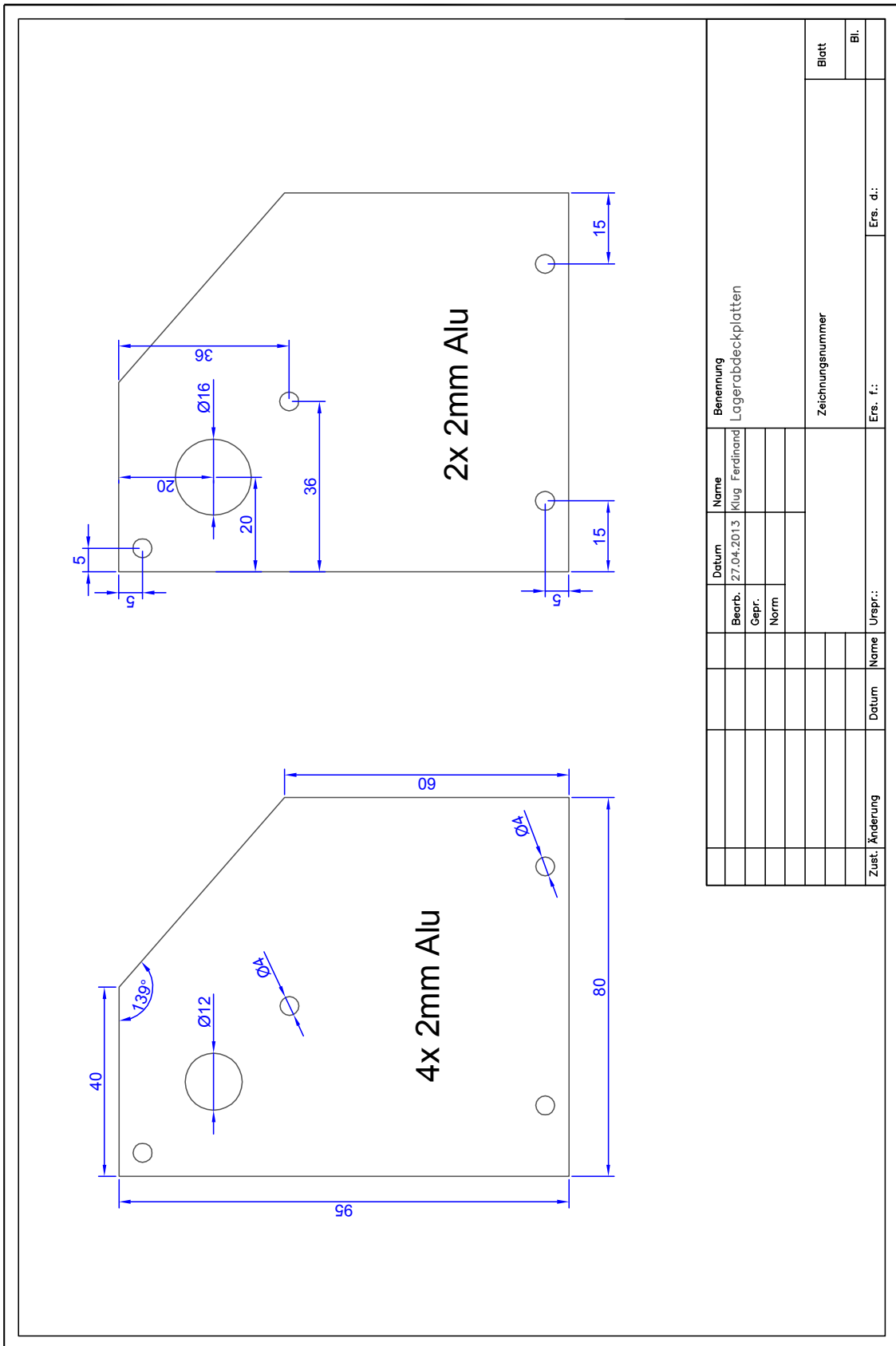




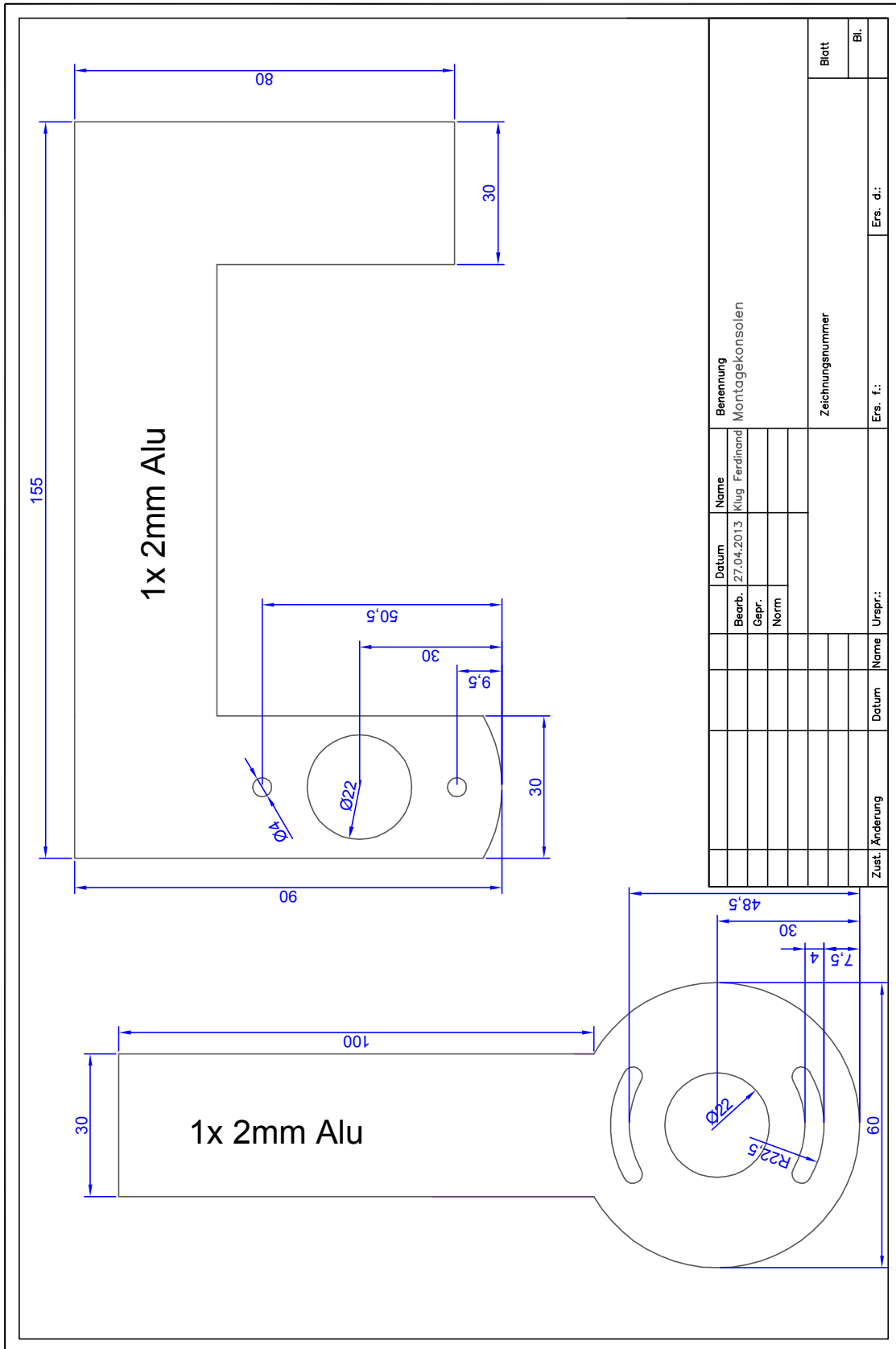




Zust.		Änderung		Datum	Name	Urspr.:	Ers. f.:	Ers. d.:	Blatt	
									Bl.	
									Bl.	
									Bl.	
									Bl.	
									Bl.	
									Bl.	
									Bl.	
									Bl.	
									Bl.	
									Bl.	

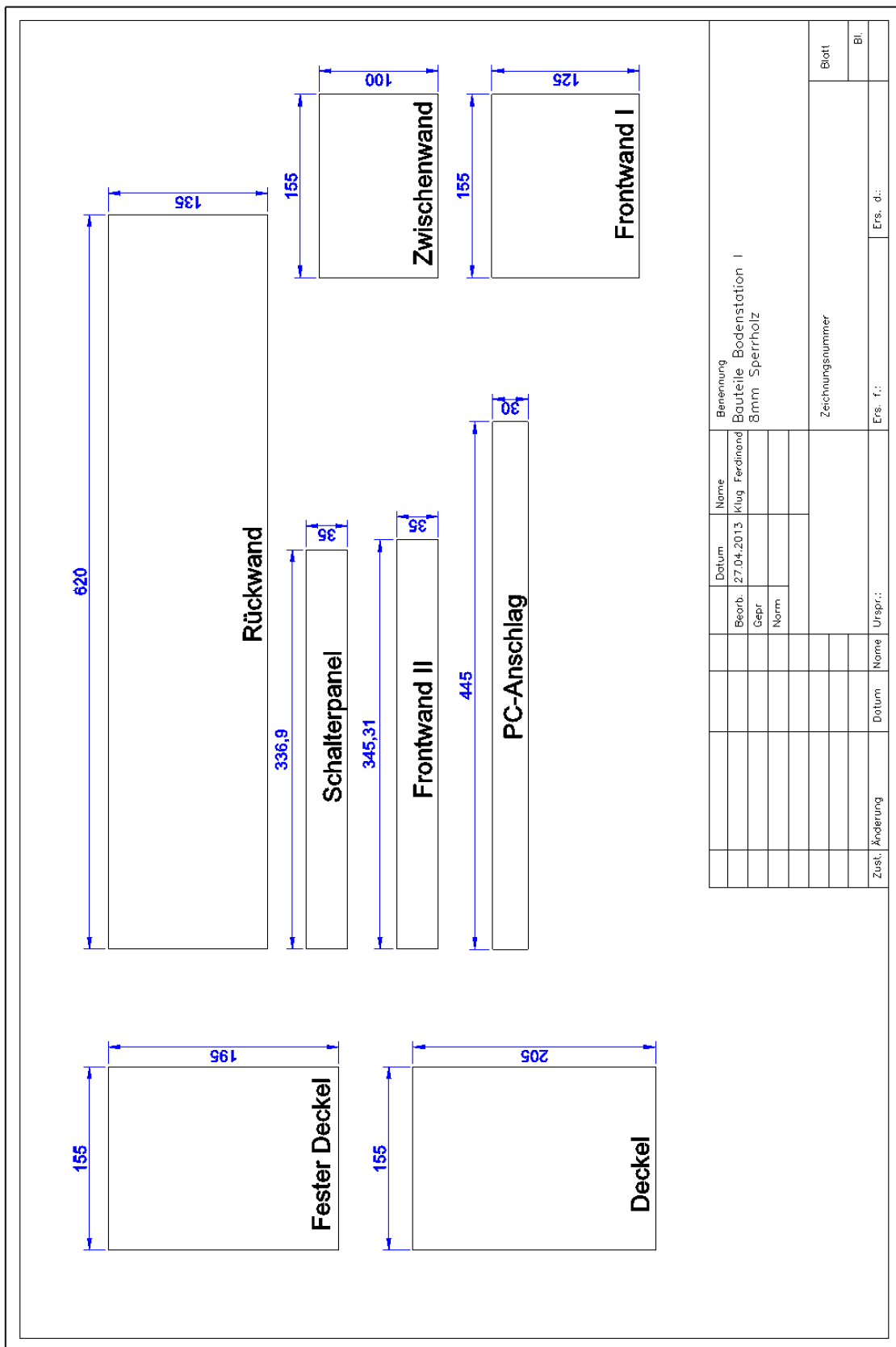




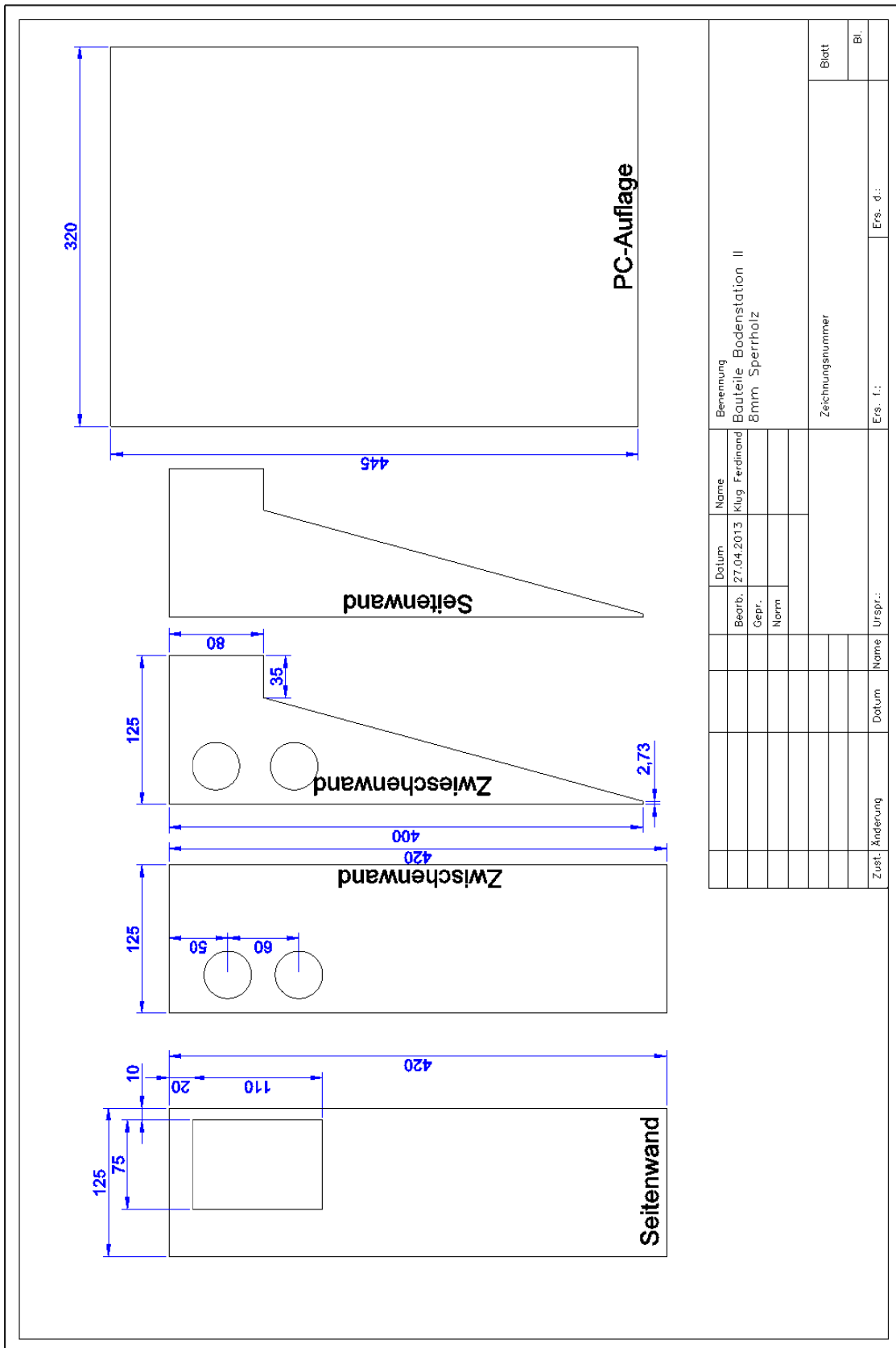


A.3 Bodenstation



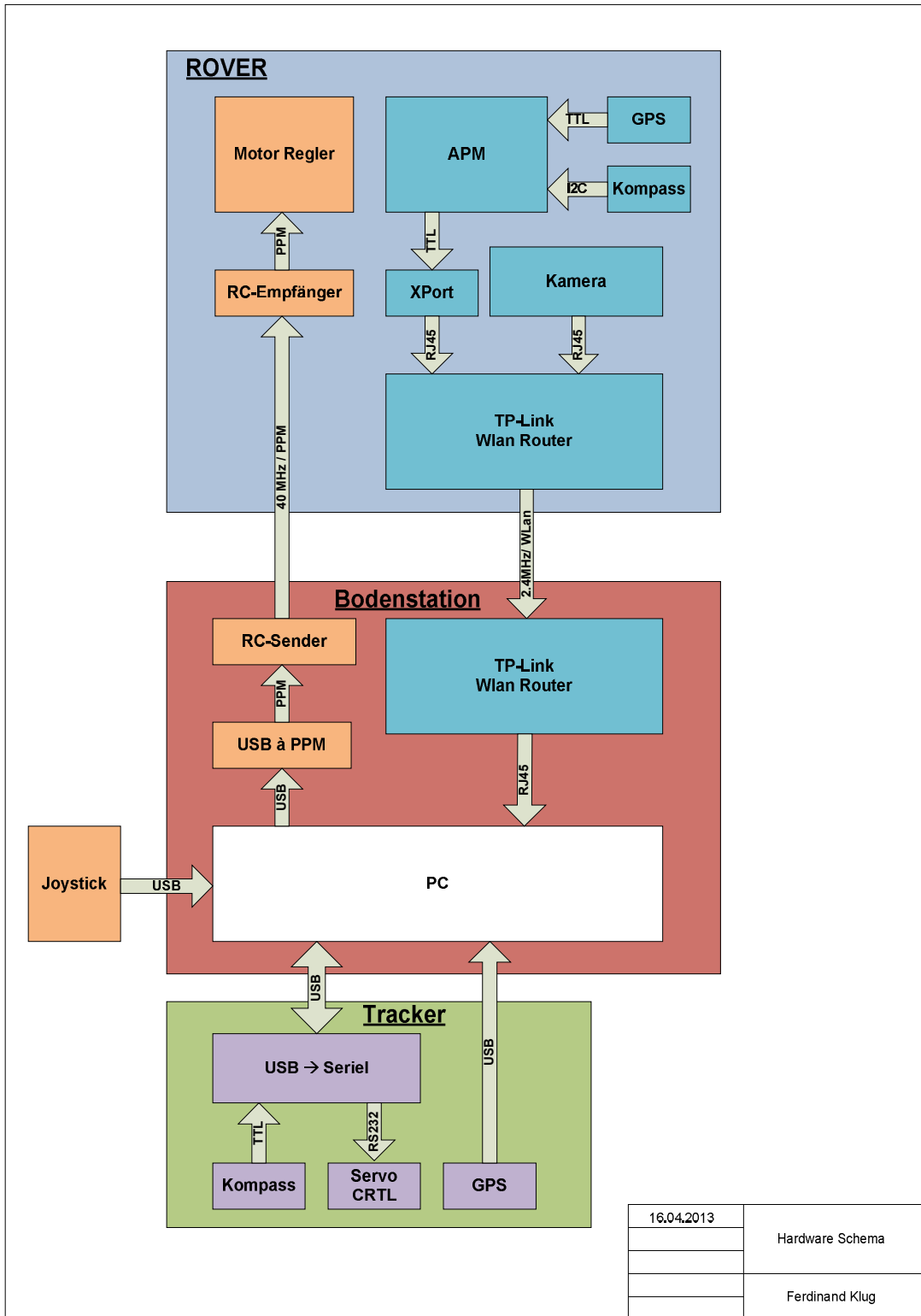






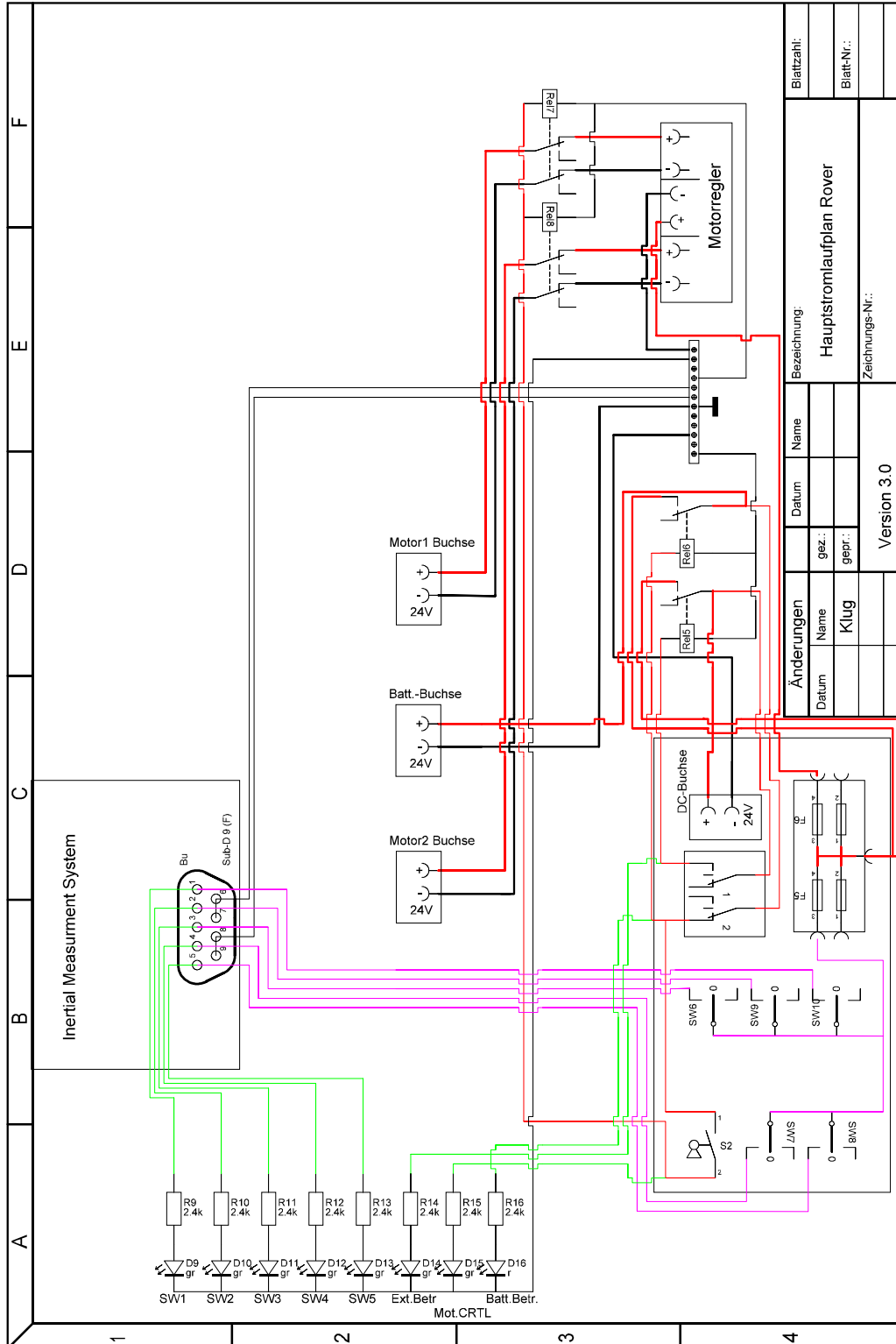
Zust.	Änderung	Datum	Name	Urspr.:	Datum	Name	Urspr.:	Datum	Name	Benennung
										Bauteile Bodenstation III
										8mm Sperrholz
										Zeichnungsnummer
										Ers. f.:
										Ers. d.:
										Blott
										Bl.

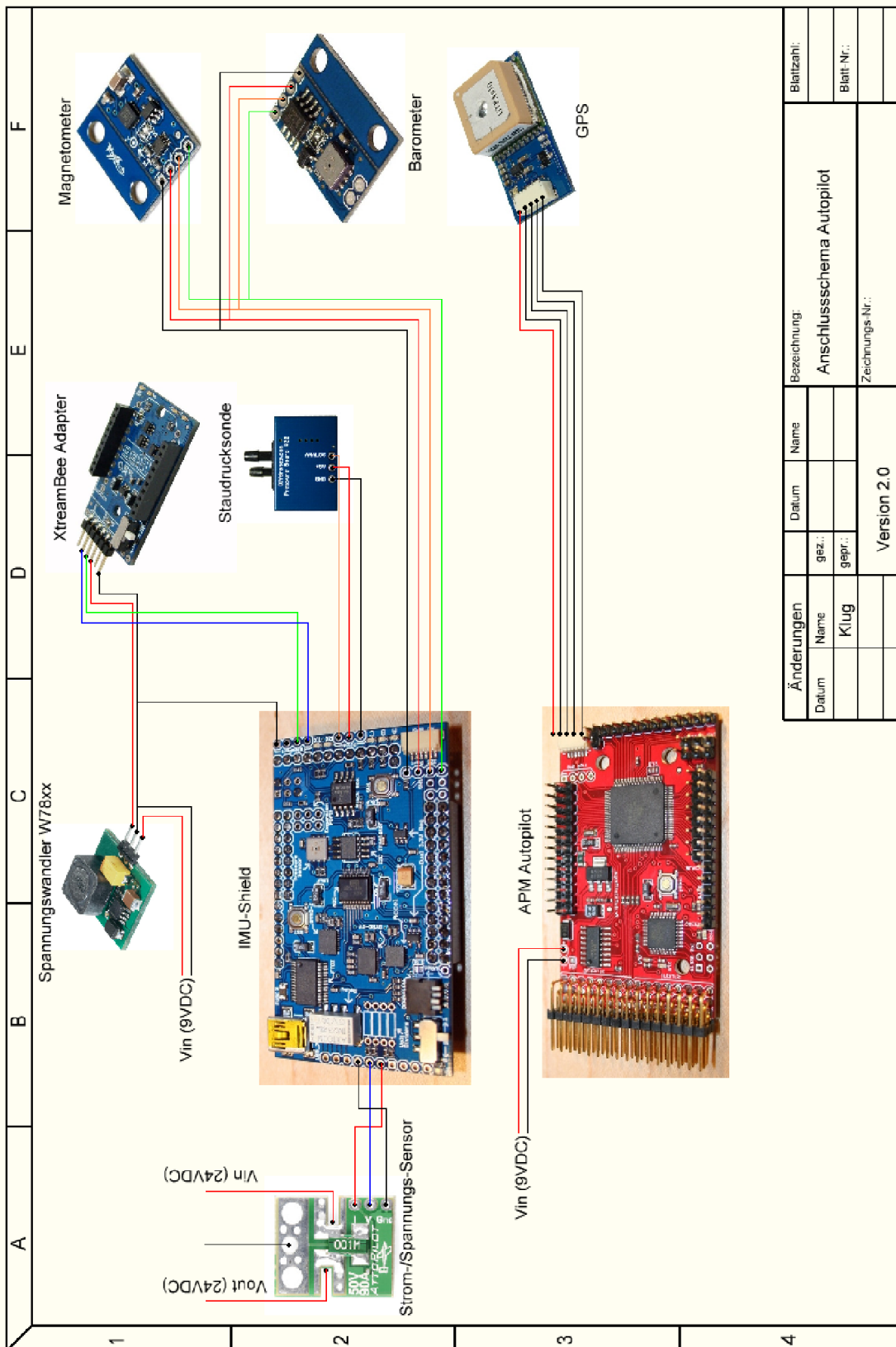
## B Kommunikations- und Elektronikschem

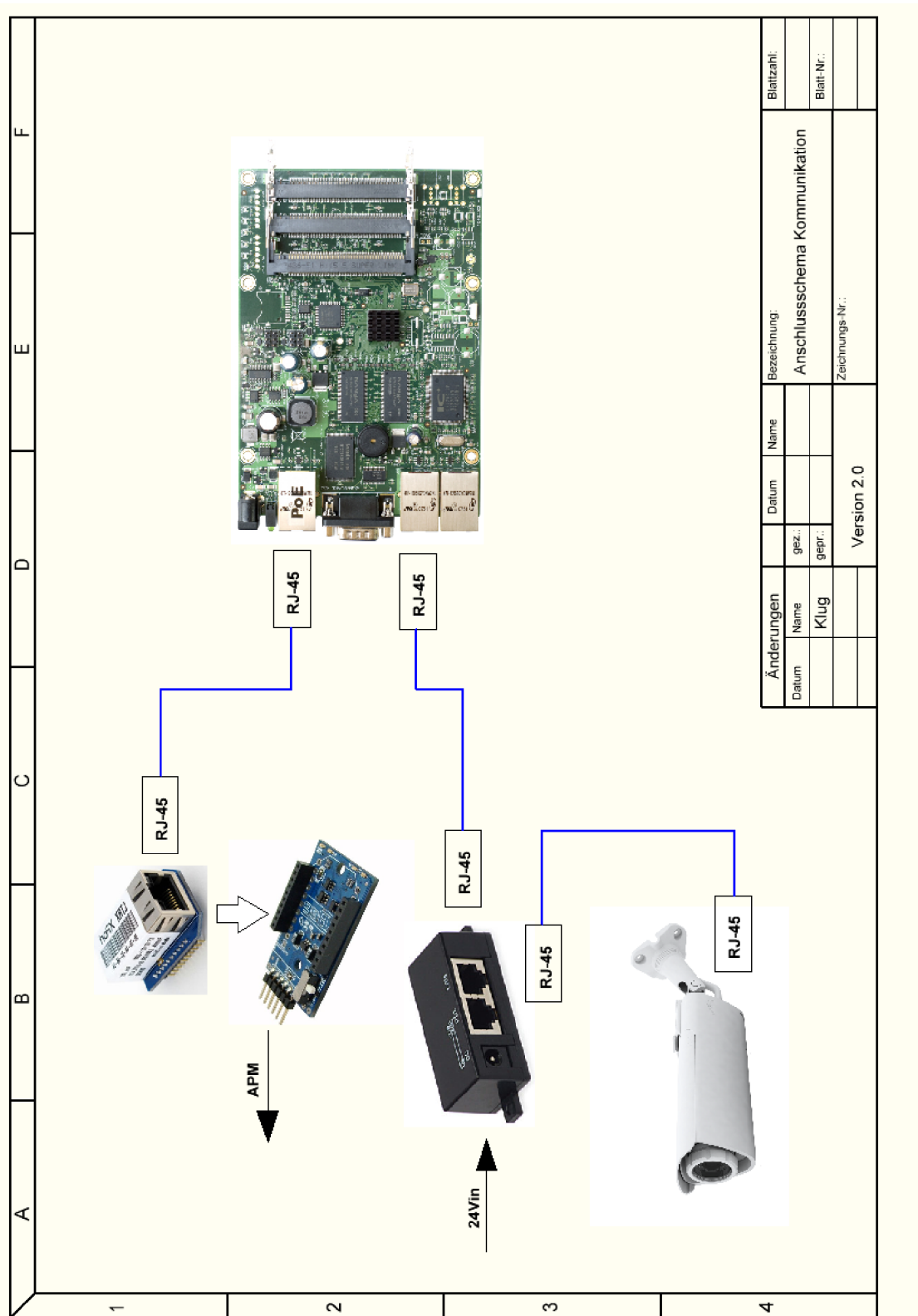


# C Stromlaufpläne

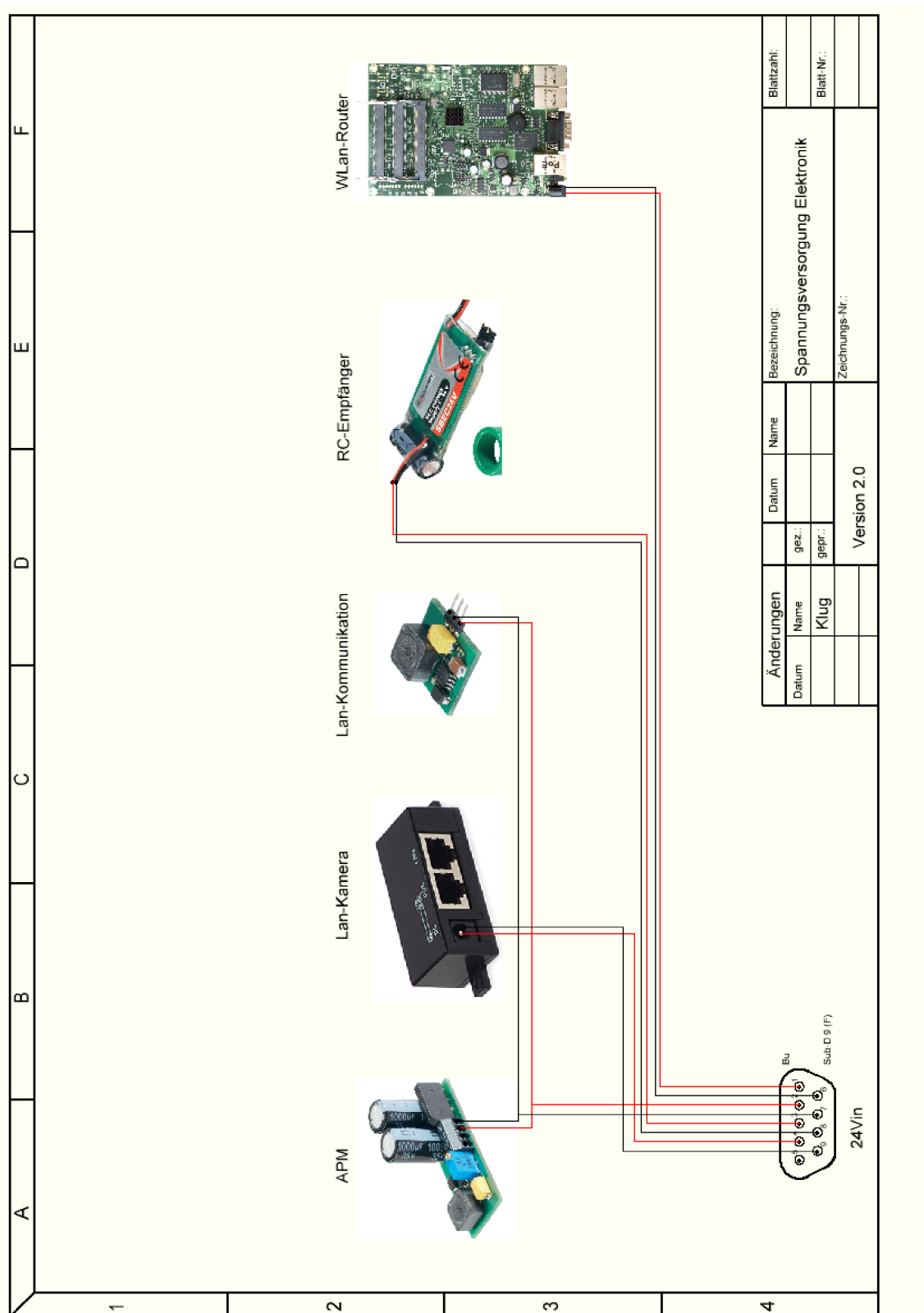
## C.1 Rover



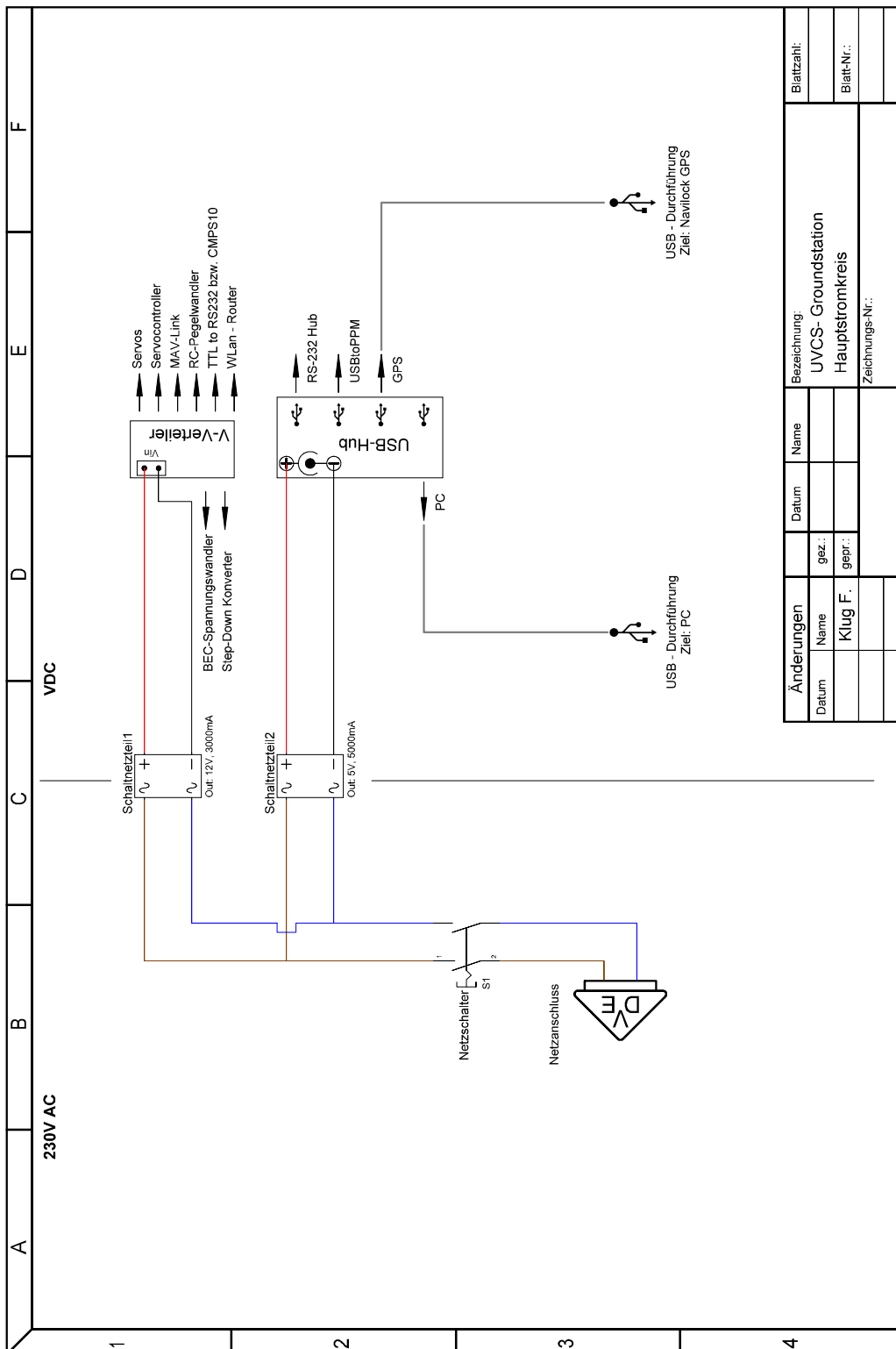




Änderungen		Datum	Name	Bezeichnung:	Blattzahl:
	Name	gez.:		Anschlusschema Kommunikation	Blatt Nr.:
	Klug	gepr.:			Blatt Nr.:
				Zeichnungs-Nr.:	
				Version 2.0	

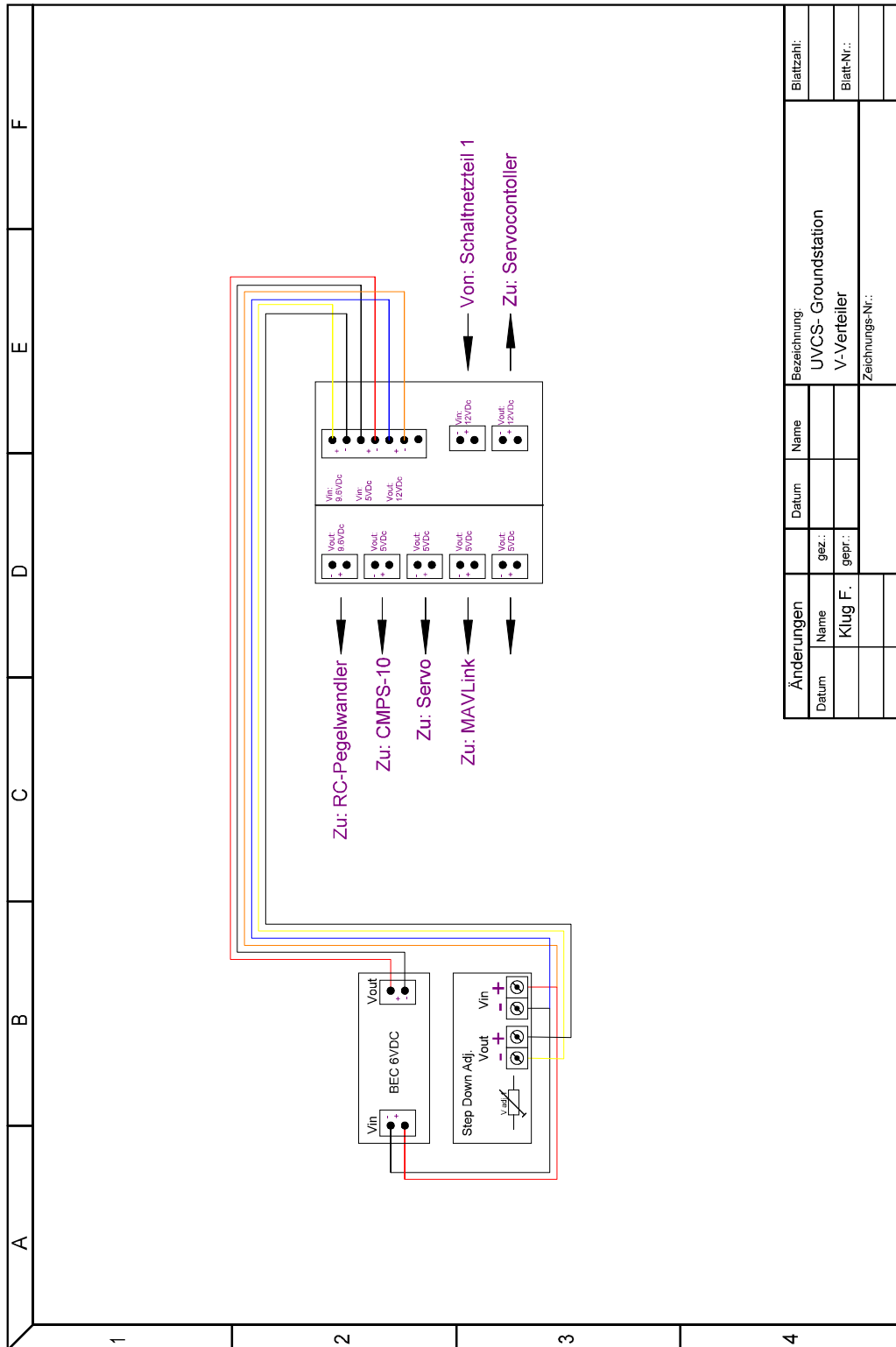


C.2 Bodenstation

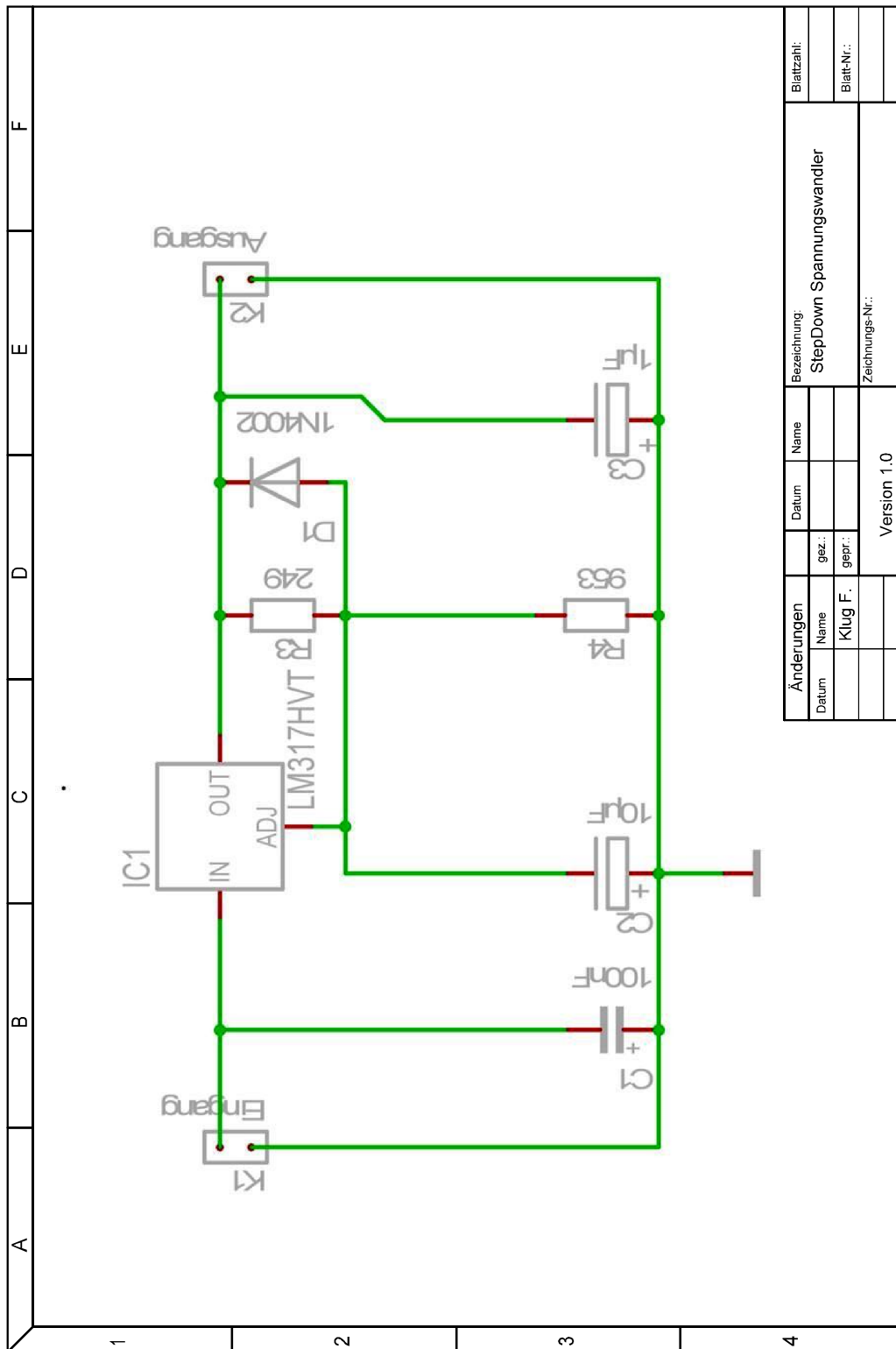


Änderungen		Datum		Name		Bezeichnung:		Blattzahl:	
							UVCS- Grundstation		
							Hauptstromkreis		
							Zeichnungs-Nr.:		

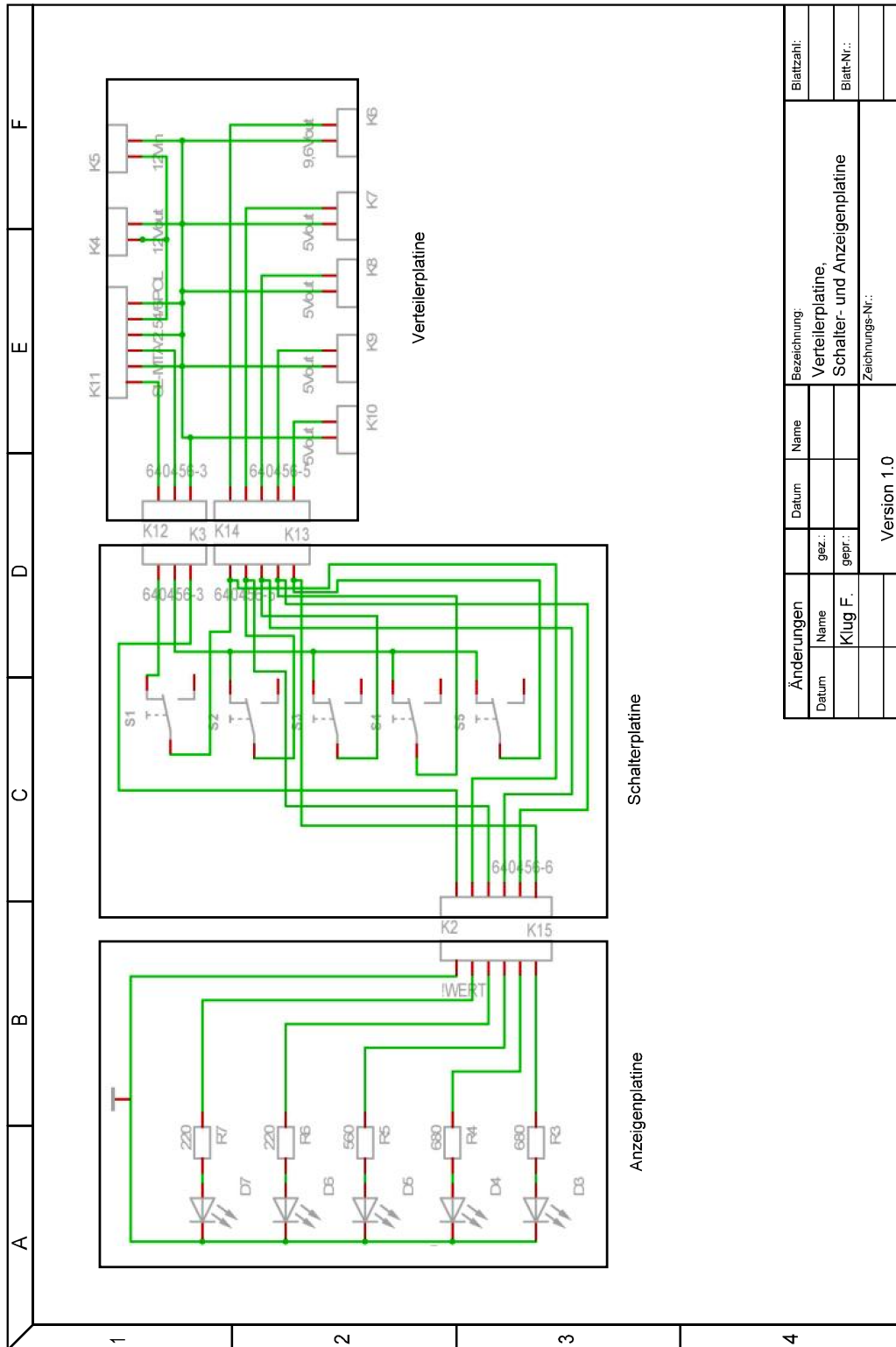


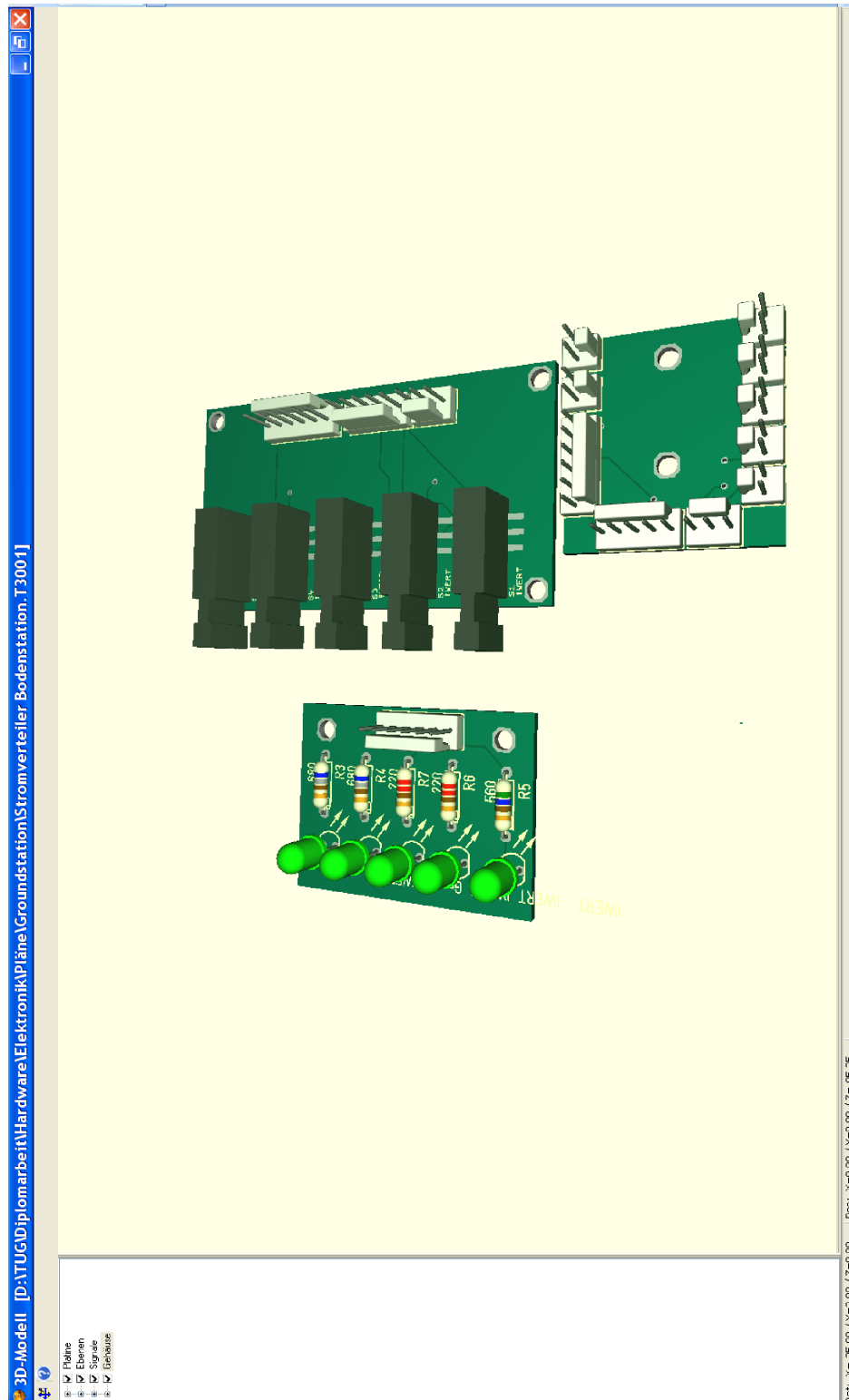


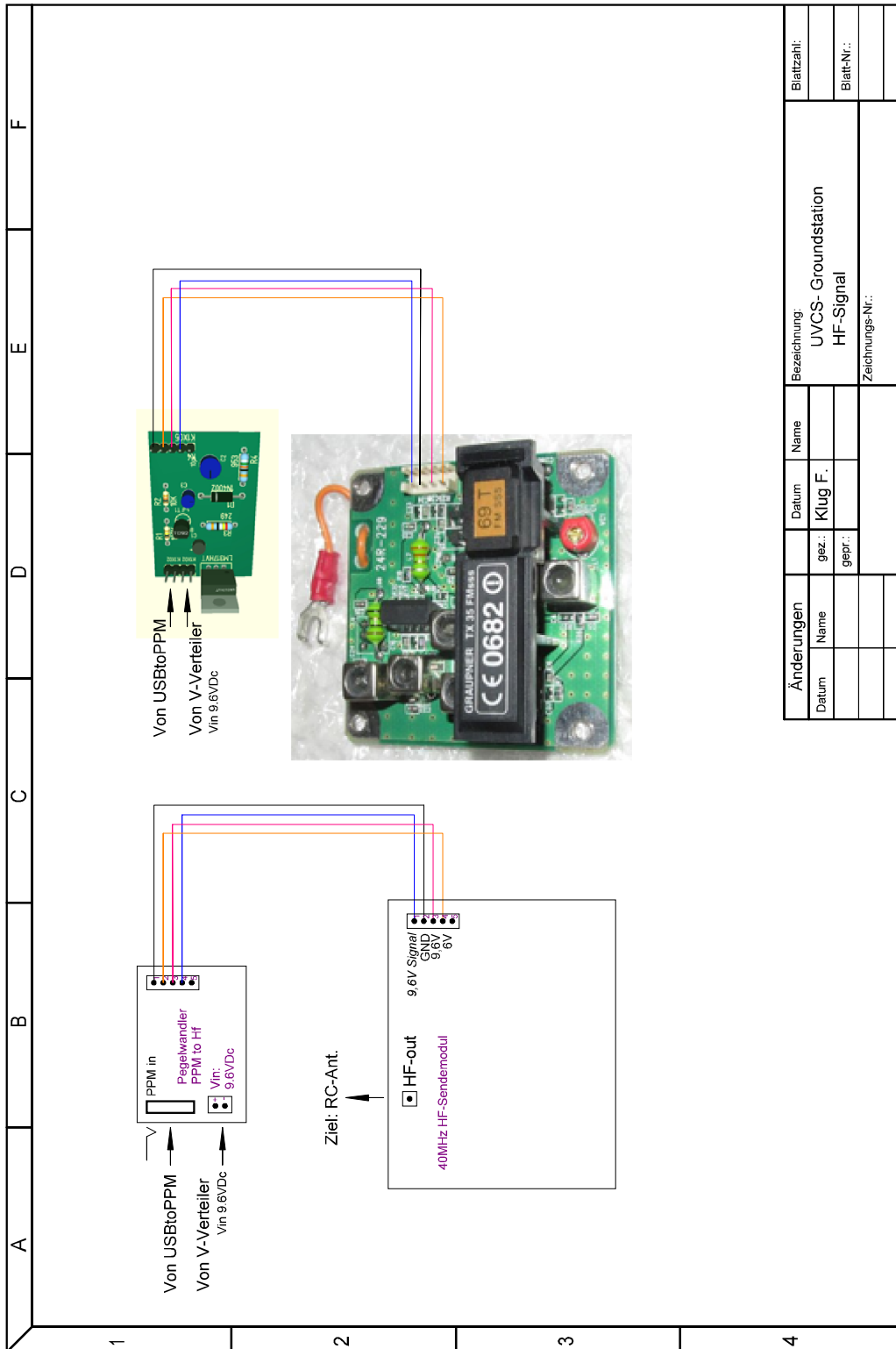
Änderungen		Datum	Name	Bezeichnung:		Blattzahl:
Datum	Name	gez.:		UVCS- Grundstation		
	Klug F.	gepr.:		V-Verteiler		Blatt-Nr.:
				Zeichnungs-Nr.:		

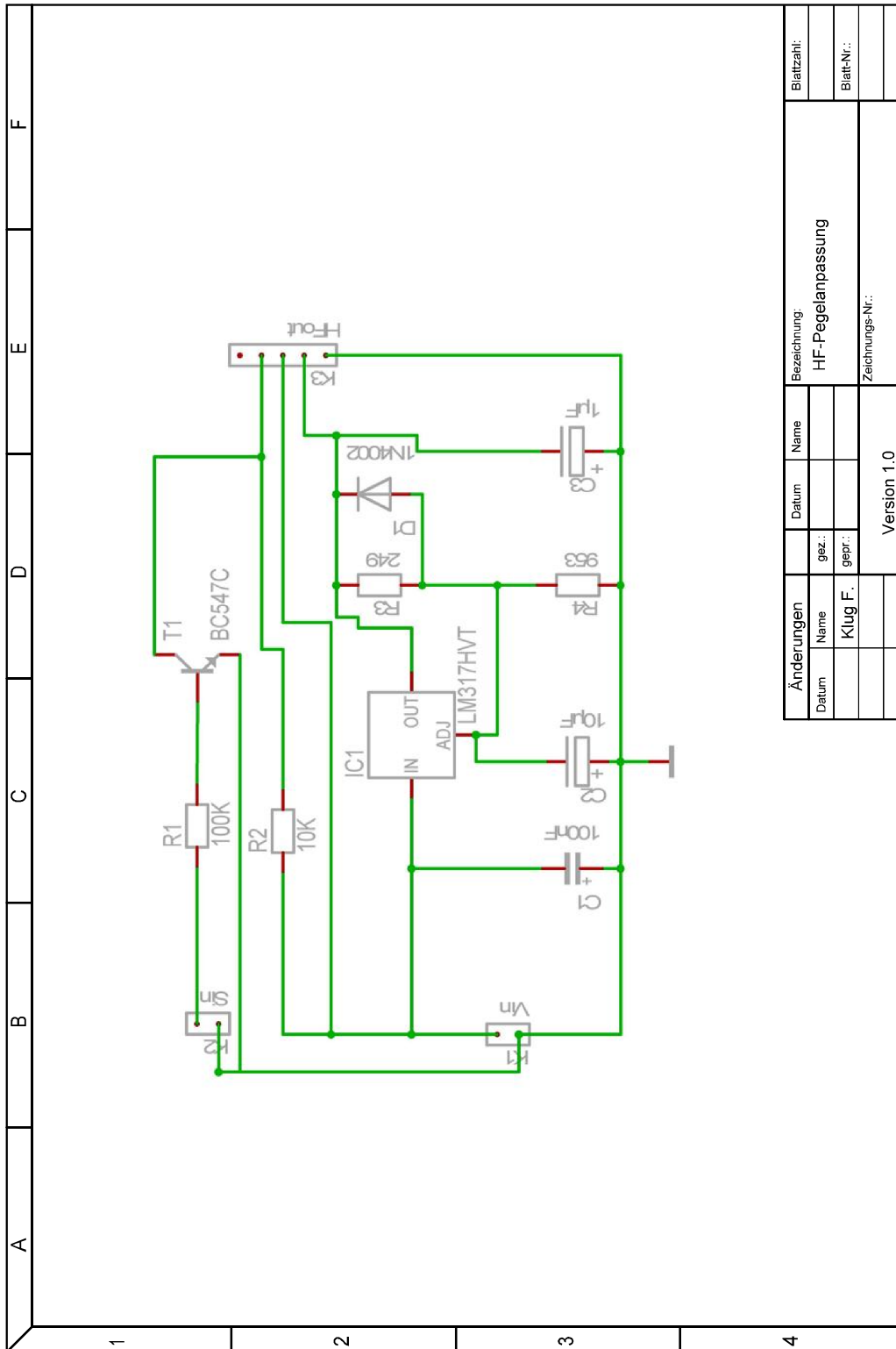


Änderungen		Datum	Name	Bezeichnung:		Blattzahl:
Datum	Name	gez.:	Klug F.	StepDown Spannungswandler		
		gepr.:		Zeichnungs-Nr.:		Blatt-Nr.:
			Version 1.0			



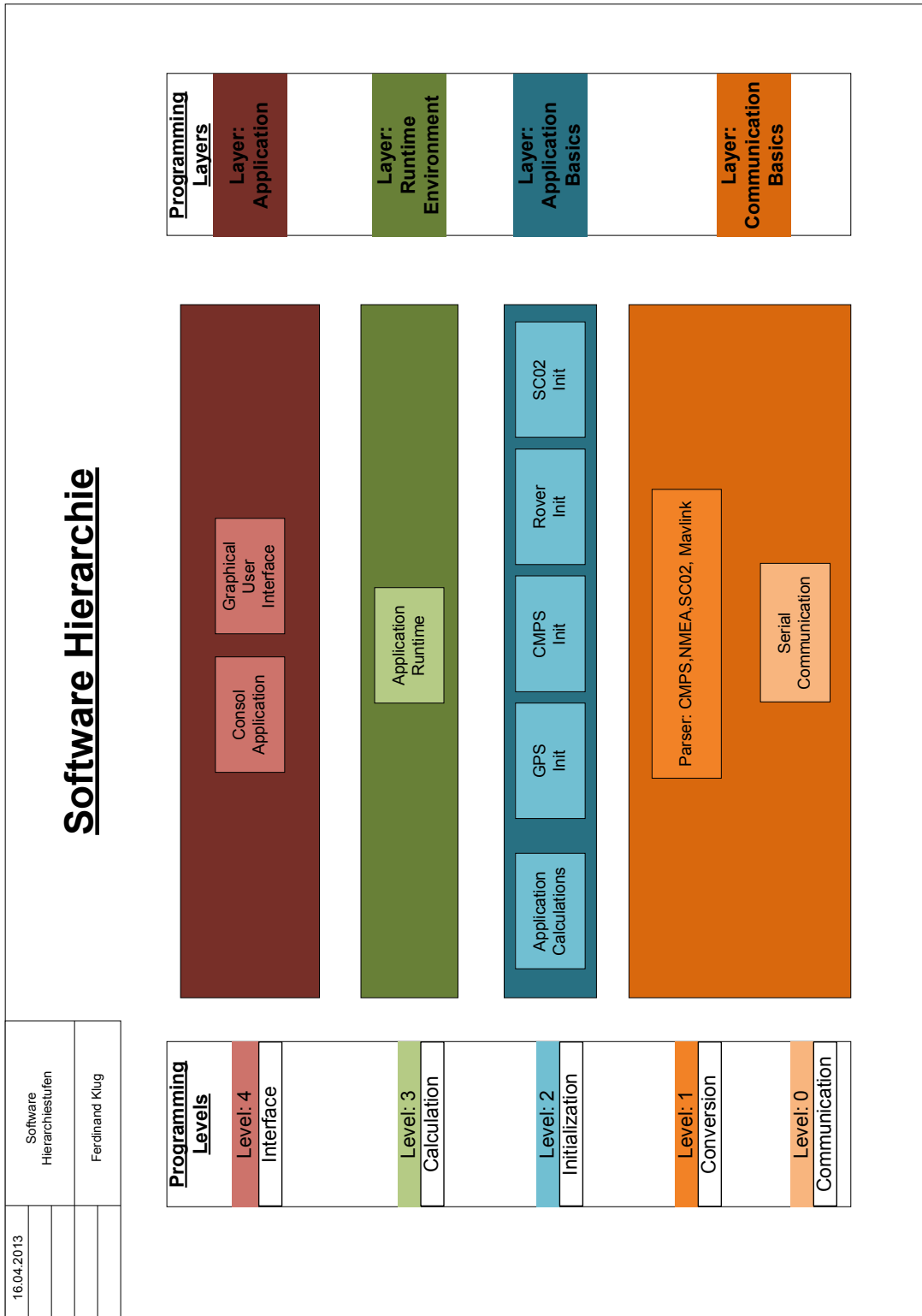






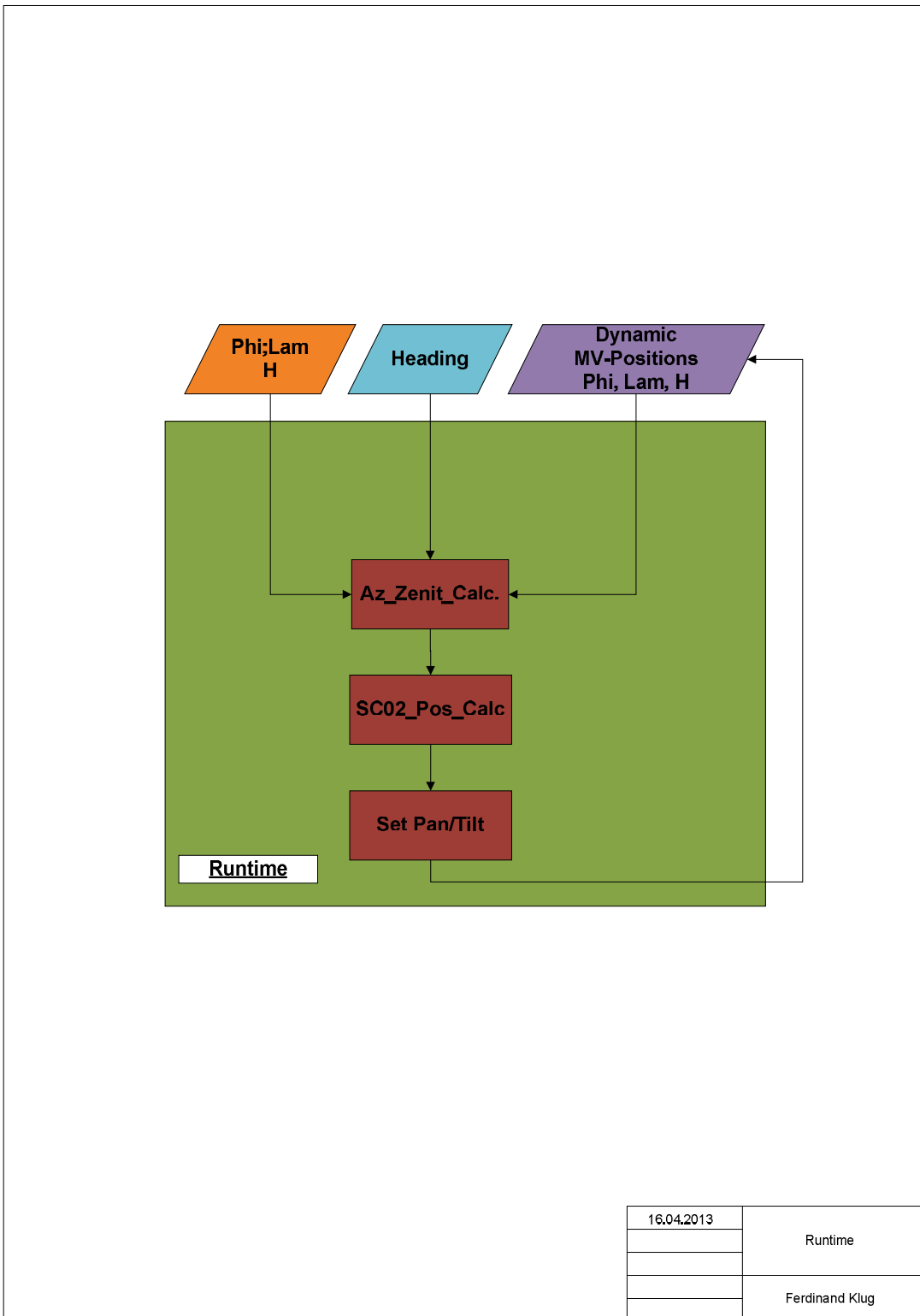
Änderungen		Datum	Name	Bezeichnung:		Blattzahl:
Datum	Name	gez.:		HF-Pegelanpassung		
	Klug F.	gepr.:		Zeichnungs-Nr.:		Blatt-Nr.:
				Version 1.0		

## D Software Schemas









## E Lieferfirmen und Kostenaufstellung

<b>Firma</b>	<b>Kosten (€)</b>	<b>Material</b>
Ach Autocolour	134,95	Dekorfolie
amazon	41,33	Akkuladegerät
B&T	55,26	Halbzeuge Aluminium
Causemann	24	RC-Antenne
conrad	568,87	Elektronik, Kleinmaterial
elv	21,42	Spannungswandler
Gutekunst	43,63	Federn
hochstraden.net	685,46	Kommunikationsmodule
Legenstein - Mädler	136,8	Achsen, Wellen
lipoly	504,33	Autopilot, Elektronik
lugitsch	22	Elektrik
neuhold	60,5	Elektronikzubehör
noDNA	202,98	Elektronik
readymaderc	60,38	RC-Elektronik
ruku	121,13	Schraubenmaterial
Schweighofer	135,87	RC-Komponenten
skateshop	32,79	Laufrollen
SOS electronic	95,04	Lantronix Module
spectra	171,36	RS232-Switch
zf austria	163	Kettenmaterial
zultner	84,52	Halbzeuge Aluminium
Knapp	229,39	Motoren
<b>Summe (€)</b>	<b>3595,01</b>	