

Bachelorarbeit

Christoph Zieger

Konzeption und Entwicklung von Embedded Software für
die Antriebssteuerung und Bahnführung eines
elektronischen Blindenführhunds

*Hochschule für Angewandte Wissenschaften Hamburg
Fakultät Technik und Informatik
Department Informations- und Elektrotechnik*

*Hamburg University of Applied Sciences
Faculty of Computer Science and Engineering
Department of Information and Electrical Engineering*

Christoph Zieger

Konzeption und Entwicklung von Embedded Software für die Antriebssteuerung und Bahnführung eines elektronischen Blindenführhunds

Bachelorarbeit eingereicht im Rahmen der Bachelorprüfung
im Bachelorstudiengang Elektro- und Informationstechnik
am Department Informations- und Elektrotechnik
der Fakultät Technik und Informatik
der Hochschule für Angewandte Wissenschaften Hamburg

Betreuender Prüfer: Prof. Dr. Lutz Leutelt
Zweitgutachter: Prof. Dr. Michael Erhard

Eingereicht am: 30. September 2021

Christoph Zieger

Thema der Arbeit

Konzeption und Entwicklung von Embedded Software für die Antriebssteuerung und Bahnführung eines elektronischen Blindenführhunds

Stichworte

Elektronischer Geschwindigkeitskontroller, Bürstenloser DC Motor (BLDC), Schrittmotor, Blindenführhund (EGD), Tiva (Microcontroller), Puls Breiten Modulation (PWM)

Kurzzusammenfassung

Mehrere Millionen Menschen weltweit sind aufgrund ihrer Sehbehinderung auf Alltagsunterstützung wie den Blindenführhund angewiesen. Der elektronische Blindenhund ist das Gegenstück zum realen Blindenführhund und ist ein mit der Hand geführtes Hilfsmittel, welches den Nutzer durch sein Antriebssystem und intelligente Elektronik im Alltag unterstützt. Ziel der Arbeit ist es, ein Antriebssystem zu erarbeiten, dass über eine geeignete Bahnführung in Form einer Fortbewegungsstrategie verfügt. Das Antriebssystem umfasst die Auswahl eines neuen Antriebs sowie die Art und Weise der Antriebssteuerung, wie auch die Umsetzung der Konzepte mithilfe von Embedded Software auf einem Mikrocontroller.

Christoph Zieger

Title of Thesis

Conception and development of Embedded Software for the drive control and path guidance of an electronic guide dog

Keywords

Electric Speed Controller ESC, Brushless DC Motor (BLDC), Steppermotor, Electronic Guide Dog (EGD), Tiva (Microcontroller), Puls Width Modulation (PWM)

Abstract

Several million people worldwide are dependent on assistance such as guide dogs due to their visual impairment. The electronic guide dog is the counterpart to the real guide dog and is a hand-guided aid that supports the user in through its drive system and intelligent electronics. The aim of the study is to develop a drive system that has a suitable path guidance in the form of a locomotion strategy. The drive system includes the selection of a new drive and the way of controlling the drive, as well as the implementation of the concepts with the help of embedded software on a microcontroller.

Inhaltsverzeichnis

Abbildungsverzeichnis	viii
Tabellenverzeichnis	xi
1 Einleitung	1
1.1 Hintergrund der Arbeit	1
1.2 Aufbau des elektronischen Blindenhunds	4
1.3 Ziel der Arbeit	6
1.4 Aufbau der Arbeit	6
2 Theoretische Grundlagen	7
2.1 Verwendete Abkürzungen	7
2.2 Verwendete Symbole	8
2.3 Elektrische Antriebe	10
2.3.1 BLDC Motor	11
2.3.2 Schrittmotor	13
2.3.3 Servomotor	15
2.3.4 Electric Speed Controllers (ESC) und Motortreiber	16
3 Stand der Technik	18
3.1 Electronic Guide Dog	18
3.2 Antriebssysteme für elektrische Fahrzeuge	22
4 Anforderungen an das Antriebssystem des EGD	25
4.1 Allgemeine Anforderungen	25
4.2 Anforderungen für die Auswahl einer neuen Antriebseinheit	26
4.3 Anforderungen an die Software	27
4.4 Übersicht zu den Anforderungen	28

5	Konzeptionierung	29
5.1	Konzept für Aufbau und Art des Antriebs	29
5.1.1	Wahl des Antriebskonzepts	30
5.1.2	Erstes Konzept - Zwei Antriebsmotoren ohne Lenkachse mit Differential	31
5.1.3	Zweites Konzept - Ein Antriebsmotor mit Lenkachse	33
5.2	Auswahl des Motors	36
5.2.1	Berechnung der benötigten Antriebsleistung	36
5.2.2	Liste der zur Auswahl stehenden Motoren	47
5.3	Konzeption und Softwareentwurf der neuen Fortbewegungsstrategie	50
5.3.1	Antriebssteuerung	50
5.3.2	Aktuelle Antriebssoftware des EGD	57
5.3.3	Anfahren und Abbremsen	63
5.3.4	Vorwärtsfahren	68
5.3.5	Kurvenfahrten	68
6	Praktischer Teil	81
6.1	Aufbau einer Teststation	82
6.1.1	Anschlüsse und Pinbelegung	84
6.1.2	Tiva TM4C1294 Mikrocontroller	86
6.1.3	Code Composer Studio (CCS)	86
6.2	Durchführung	86
6.2.1	Inbetriebnahme des Motors	86
6.2.2	Implementierung der Fortbewegungsstrategie auf dem Mikrocontroller	87
6.2.3	Implementierung in das Gesamtsystems des EGD	88
6.2.4	Testfahrten	88
7	Auswertung	95
7.1	Bewertungskriterien für Testfahrten	95
7.2	Bewertung der Testfahrten	96
7.3	Zusammenfassung der Ergebnisse	97
7.3.1	Erfolgreich umgesetzte Anforderungen	98
7.3.2	Nicht erfolgreich umgesetzte Anforderungen	98
7.4	Verbesserungsvorschläge	98
8	Fazit und Ausblick	100

Literaturverzeichnis	102
A Anhang	107
Selbstständigkeitserklärung	114

Abbildungsverzeichnis

1.1	Behinderungsarten in Deutschland (Stand 2019), insgesamt 7,9 Mio. Schwerbehinderte[37]	1
1.2	Frontansicht des elektronischen Blindenführhunds (EGD) mit altem Antriebssystem	4
1.3	Seitenansicht des elektronischen Blindenführhunds mit altem Antriebssystem	5
2.1	Grundaufbau eines elektrischen Motors[26]	11
2.2	Grundprinzip eines BLDC Innenläufer Motors[19]	12
2.3	Aufbau: Innen- und Außenläufer[24]	12
2.4	Pulssignal eines Schrittmotors	13
2.5	Aufbau Hybrid Schrittmotor[12]	14
2.6	Aufbau eines Servomotors[17]	15
2.7	ESC: Mit Wechselrichterschaltung für die Kommutierung[23]	16
3.1	Theia Projekt[25]	19
3.2	Blindbot mit Ballsystem als Antrieb. Entwickler: Alexander Bayer und Niklas Gutsmedl[29]	20
3.3	Projekt MYGO, Zeichnung des Blindenstock mit Antrieb am Stabsende. Entwickler: Sebastian Ritzler[20]	21
3.4	Belt Driven E-Skateboard[10]	22
3.5	Roboter Cheetah, von Boston Dynamics[13]	23
3.6	E-Skateboard mit HUB Motor[28]	23
3.7	HUB Motor mit und ohne Getriebe[31]	24
5.1	Zeichnung des allgemeinen Aufbaus mit Stab, Rädern, Antrieb und Benutzer - Front und Seitenperspektive	30
5.2	Aufbau des ersten Antriebkonzepts	31
5.3	Aufbau des ersten Antriebkonzepts mit Kugelgelenk	32
5.4	Aufbau des zweiten Antriebkonzepts	33
5.5	Aufbau des zweiten Antriebkonzepts mit Stützrad und Gelenk	34

5.6	Übersicht zu den Vor- und Nachteilen der zwei Antriebskonzepte	35
5.7	Schiefe Ebene ohne Haftreibung[27]	40
5.8	Schiefe Ebene mit Haftreibung F_R (rot)[27]	40
5.9	Kraftmesser für die Zugkraftmessung	42
5.10	Ausschnitt Schrittmotor - Auswahl	48
5.11	Schrittmotor[1] mit Treiber: Trinamic PD57-2-1076[15]	48
5.12	Aufbau des ersten Antriebkonzepts, seitliche Perspektive	49
5.13	Aufbau des ersten Antriebkonzepts, Draufsicht	49
5.14	Trinamic TMCM 1076 - Motortreiber mit Interface[6]	50
5.15	Schaltplan für den Motor-Driver (TMCM-1076)[5]	51
5.16	Übersicht zum Arbeitsbereich, sowie den Header- und .c Dateien	58
5.17	Flowchart des Programmstarts mit Endless-Loop	59
5.18	Ausschnitt zu der Antriebssoftware	60
5.19	Konzept - Algorithmus zum Anfahren des EGD	65
5.20	timerInterruptHandler(void), Algorithmus für das Anfahren des EGD . . .	66
5.21	Vogelperspektive einer Linkskurve mit Kurvenwinkel α	69
5.22	Linearer Zusammenhang zwischen α (X-Achse) und dem Innenradius r_i (Y-Achse)	71
5.23	Nichtlinearer Zusammenhang zwischen α (X-Achse) und dem Innenradius r_i (Y-Achse)	72
5.24	Differential der Motoren bei einer Kurvenfahrt	73
5.25	Vogelperspektive einer Linkskurve mit eingezeichnetem Benutzerweg l_{bogen} (grün)	75
5.26	Mögliche Verläufe von Kurvenfahrten	77
5.27	Flowchart zur Übersicht vom Funktionsablauf set_angle() bei einer Rechts- kurve	78
5.28	Funktionsablauf von set_angle() im Interrupthandler (<i>timerInterruptHand- ler()</i>) bei einer Rechtskurve	79
6.1	Schematischer-Schaltplan für Teststation, inklusive MC, Motor und Driver	82
6.2	Teststation mit Platine, Mikrocontroller und Bauelementen	83
A.1	CCS Projekt: watchdog, Übersicht des Arbeitsbereichs - Headerfiles und .c Dateien	108
A.2	Flowchart - Antriebssoftware des EGD	109

A.3	Flowchart zur Übersicht vom Funktionsablauf <code>set_angle()</code> bei einer Rechtskurve	110
A.4	Funktionsablauf von <code>set_angle()</code> im Interrupthandler (<i>timerInterruptHandler()</i>)	111
A.5	BLDC Motor Auswahl - Übersicht Bewertung	112
A.6	Schrittmotor Auswahl - Übersicht Bewertung	113

Tabellenverzeichnis

4.1	Physikalische Beschreibung der Eigenschaften und Anforderungen des EGD	28
5.1	Parameter für die Berechnung der Motorleistung	36
5.2	Beschreibung der Formelzeichen für die Berechnungen	37
5.3	Zusammenfassung der Rechenergebnisse	46
5.4	Trinamic TMC2102 - Interface Pinbelegung[5]	52
5.5	Zusammenfassung der Rechenergebnisse für Geschwindigkeit und Beschleunigung	64
6.1	Anschlussplan der Teststation	84
6.2	Anordnung der Motoren und Treiber im EGD	84
6.3	Anschlussplan für Motor 1	85
6.4	Anschlussplan für Motor 2	85
6.5	Weitere Pinbelegung des Microcontrollers	85
6.6	Übersicht der Funktionen für das Softwareprojekt EGD_Teststation . . .	87
6.7	Übersicht zu wichtigen <i>Defines bzw. Makros</i> für die EGD Teststation . . .	88
6.8	Strommessung des Motors bei 14 V <i>Betriebsspannung</i>	89
6.9	Ergebnisse zum Anfahren und Abbremsen	91
6.10	Ergebnisse zum Geradeausfahren	92
6.11	Ergebnisse zur Kurvenfahrt	93

1 Einleitung

1.1 Hintergrund der Arbeit

In Deutschland wurde im Jahr 2019 die Zahl der schwerbehinderten Menschen vom Statistischen Bundesamt auf ca. 7,9 Millionen Menschen geschätzt. Davon waren ca. 350.000 Menschen (ca. 4,5 %) schwer sehbehindert oder sogar blind.[9] Das folgende Kreisdiagramm 1.1 soll dies näher ausführen.

Behinderungsarten 2019

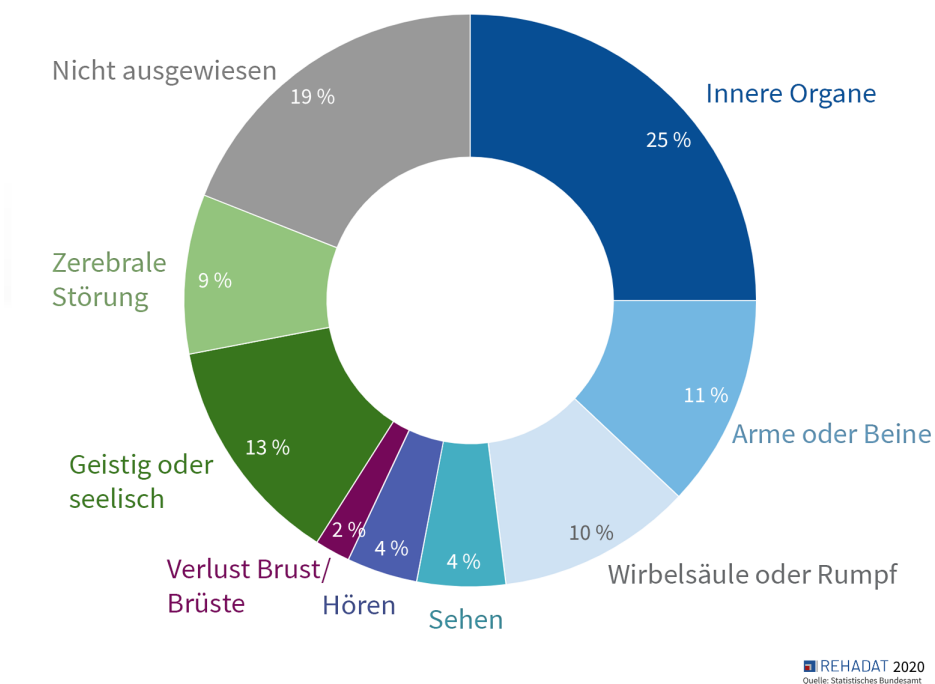


Abbildung 1.1: Behinderungsarten in Deutschland (Stand 2019), insgesamt 7,9 Mio. Schwerbehinderte[37]

In einem Bericht des Robert Koch Instituts (RKI) aus dem Jahre 2017 über Blindheit und Sehbehinderung heißt es, dass die World Health Organisation (WHO) weltweit die Anzahl der Sehbehinderten auf eine Zahl von 246 Millionen schätzt, dessen Visus¹ zwischen 0,3 und 0,05 liegt. Menschen dessen Visus kleiner als 0,05 war, wurden als blind eingestuft, dies waren ca. 39 Millionen Menschen weltweit.[21] Ein jung Erwachsener hat im Vergleich dazu einen Visus zwischen 1,0 und 2,0.[18]

Die globale Datenerfassung der Sehbehinderten und Blinden ist eine große Herausforderung, da jedes Land die Behinderung individuell definiert und der Erfassungsgrad somit nicht überall identisch ist.

Unterstützung und Hilfsmittel

Unterstützung und Hilfsmittel für Sehbehinderte gibt es in verschiedener Hinsicht. Eine finanzielle Unterstützung wird durch das Blindengeld oder durch die Krankenkassen ermöglicht. Weitere Beihilfe kommt zum anderen durch die Anwesenheit von Pflegepersonal zustande, als auch durch den Einsatz eines Blindenhunds oder anderen technischen Instrumenten wie z.B. einem Blindenstock, Diktiergerät oder Ähnlichem. In einer Umfrage zur Lebenssituation blinder Menschen wurden die wichtigsten Hilfsmittel für die befragten Sehbehinderten erwähnt. Insbesondere im Bereich der Mobilität sind die Blinden auf ein Hilfsmittel angewiesen. 86 % der Befragten waren auf einen Langstock bzw. Blindenstock angewiesen. Des Weiteren waren von allen Befragten ungefähr 85 % auf eine Begleitperson genau dann angewiesen, wenn sie sich auf unbekanntem Wegen befunden haben.[36] Daraus resultiert, dass der Großteil aller Sehbehinderten nicht nur an ein einfaches Hilfsmittel gebunden ist, sondern auch bei unbekanntem Strecken und Veränderung der gewohnten Umgebung mindestens eine Begleitperson oder ein ähnlich intelligentes System benötigten. Da Begleitpersonen, sowie Blindenhunde zum Teil nur schwer verfügbar sind und auch finanziell gedeckt werden müssen, stellt dies eine besonders große Herausforderung für die Sehbehinderten dar.

¹Der Visus ist ein Maß für die Sehschärfe eines Menschen, dessen Wert zwischen 0 und 1 liegen kann. Desto höher der Visus ist, desto besser ist die Sehschärfe des Menschen. Es ist möglich, dass ein Mensch einen Visus von über 1, also über 100 % haben kann.

Ein üblicher Blindenhund kann bis zu 30.000 Euro kosten.[33] Unter passenden Voraussetzungen kann hierbei die Krankenkasse die Kosten für den Hund übernehmen. Neben den hohen Kosten ist es auch wichtig, dass der Hund ausreichend trainiert wird, um somit zuverlässige Dienste zu gewährleisten. Ein Blindenführhund ist in der Lage bis zu 76 Hörzeichen und Befehle zu lernen. Auch ist es wichtig, dass die gewohnte und trainierte Umgebung des Blindenhunds nicht verändert wird. Baustellen oder andere Hindernisse, welche die Umgebung bzw. die Routen stören oder sogar blockieren, können den Blindenhund in der Orientierung und Führung behindern.

Einblick

Mit dieser Arbeit soll an einem bereits bestehenden elektronischen Hilfsmittel weiterentwickelt und gearbeitet werden. Dieses Hilfsmittel ist ein elektronischer Blindenführhund, bestehend aus einem Stab, Antrieb, Sensorik, sowie anderer intelligenter Elektronik und Software. Er dient zur Navigation, Wegführung und der allgemeinen Verbesserung der Mobilität eines Sehbehinderten.

1.2 Aufbau des elektronischen Blindenhunds

Der aktuelle Aufbau des elektronischen Blindenhunds besteht aus einem Stab, welcher von dem Benutzer gehalten wird und einem elektrischen Antrieb, für die Richtungsweisung des Blinden. Des Weiteren verfügt er über zusätzliche Sensorik, wie zum Beispiel dem Lidar Sensor und anderer Elektronik zur Erkennung von Gegenständen. Weitere Elektronik ermöglicht es dem Blinden über eine Spracheingabe, Zielvorgaben bzw. Navigationsanweisungen zu geben, um folglich zum gewünschten Ort zu gelangen. Das folgende Bild 1.2 veranschaulicht den elektronischen Blindenführhund mit dem alten Antriebssystem, aus der Frontperspektive. Der Antrieb ist am Ende des Stabes zu erkennen. Die zwei Räder werden über einen kleinen Brushless DC Motor angesteuert. Für die Lenkung ist ein Servomotor mit dazugehöriger Lenkachse verantwortlich.

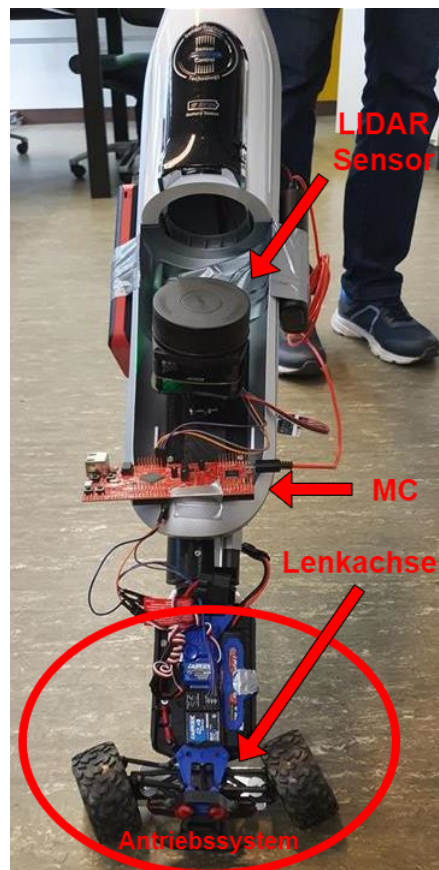


Abbildung 1.2: Frontansicht des elektronischen Blindenführhunds (EGD) mit altem Antriebssystem

1 Einleitung

Die folgende Abbildung 1.3 veranschaulicht den Prototypen des elektronischen Blindenführhunds aus der Seitenansicht.



Abbildung 1.3: Seitenansicht des elektronischen Blindenführhunds mit altem Antriebssystem

Auch hier ist der Stab, inklusive der Elektronik und elektrischem Antrieb zu sehen. Über den Lidar Sensor sollen Objekte auf dem Weg erkannt und folglich vom Antrieb umfahren werden.

1.3 Ziel der Arbeit

In dieser Arbeit soll ein Konzept sowie eine Software für die Antriebssteuerung und Bahnführung eines elektronischen Blindenhunds entwickelt werden. Darüber hinaus soll die neue Software auf einem Mikrocontroller in ein bereits bestehendes Projekt mit eingebunden werden und die alte Softwarelösung zur Antriebssteuerung und Bahnführung ersetzen. Für den Verlauf dieser Arbeit und die Implementierung der neuen Antriebssteuerung sowie der Bahnführung ist es daher notwendig, einen neuen Antrieb sowie ein passendes Konzept für eine Antriebsart zu finden.

1.4 Aufbau der Arbeit

Diese Arbeit ist wie folgt gegliedert. Beginnend mit Kapitel 2 werden die theoretischen Grundlagen dargelegt. Hierbei werden ähnliche Projekte vorgestellt und untersucht. Ein besonderer Fokus liegt hierbei auf den gewählten Antriebslösungen. Im dritten Kapitel wird der Stand der Technik behandelt, woraufhin sich das vierte Kapitel mit den Anforderungen an das Antriebssystem des Electronic Guide Dogs beschäftigt. Kapitel 5 beinhaltet die Konzeptionierung für den Aufbau, die Art des Antriebs sowie die Wahl des Motors und der Software. Das darauffolgende sechste Kapitel beschäftigt sich mit der praktischen Durchführung des Projekts, inklusive der Teststation und Testfahrten. Im vorletzten Kapitel (Kapitel 7) findet eine Auswertung der im praktischen Teil erhobenen Daten statt. Abgeschlossen wird die Arbeit in Kapitel 8 mit einem Fazit und Ausblick.

2 Theoretische Grundlagen

Für den weiteren Verlauf dieser Arbeit werden die wichtigsten Grundlagen im Folgenden erklärt. Beginnend mit den Abkürzungen und Symbolen für die späteren Berechnungen.

2.1 Verwendete Abkürzungen

BLDC:	Brush Less DC Motor / Bürstenloser Gleichstrom (DC) Motor
RKI :	Robert Koch Institut
Back EMF:	Elektromagnetische Gegenkraft
ESC:	Electric Speed Controller / Elektronischer Geschwindigkeits Regler
MCU:	Microcontroller Unit
MC:	Microkontroller
RC Auto:	Radio Controlled Car / Funkferngesteuertes Modellauto
CCS:	Code Composer Studio
PWM:	Puls Width Modulation
TIVA:	Kurzname für den verwendeten Microcontroller
EGD:	Electronic Guide Dog / Elektronischer Blindenführhund / Blindenhund
Blindenhund:	Abkürzung für elektronischer Blindenführhund
Guide Dog:	Abkürzung für elektronischer Blindenführhund

2.2 Verwendete Symbole

MS :	Microsteps - Mikroschritte
n_{teeth} :	Anzahl der Rotorzähne
SA :	Step Angle - Schrittwinkel
$STEP_{resol.}$:	Schrittauflösung des Stepper motors
RPM :	Rounds per Minute - Umdrehungen pro Minute
$n(t)$:	Umdrehungen pro Zeit
f_{step} :	Schrittfrequenz
v :	Geschwindigkeit
v_{start} :	Startgeschwindigkeit
$\Delta n_{interrupts}$:	Anzahl der aufgetretenen Interrupts
v_n :	Natürliche Schrittgeschwindigkeit
$v_{m,i}$:	Geschwindigkeit des innenliegenden Motors
$v_{m,a}$:	Geschwindigkeit des außenliegenden Motors
v_{curve} :	Differential-Geschwindigkeit für Kurvenfahrt
a_{curve} :	Beschleunigung des Innenrads während der Kurvenfahrt
Δt :	Dauer der Kurvenfahrt
a :	Beschleunigung des Blindenhunds beim Anfahren
ω :	Winkelgeschwindigkeit
$\dot{\omega}$:	Winkelbeschleunigung
g :	Erdbeschleunigung
F_{Roll} :	Rollreibungskraft
F_{Luft} :	Strömungswiderstandskraft
$F_{Steigung}$:	Steigungswiderstandskraft
μ_r :	Rollreibungskoeffizient
F_{Zug} :	Zugkraft des Benutzers
P_{Mech} :	Mechanische Leistung
η :	Wirkungsgrad
M :	Drehmoment
M_L :	Lastdrehmoment
$M_{\dot{\omega}}$:	Beschleunigungsdrehmoment
m :	Masse - Gewicht des Blindenhunds
D :	Durchmesser der Reifen

α :	Kurvenwinkel
β :	Haltewinkel des Benutzers
L_R :	Abstand der Räder zueinander
r_i :	Innenradius einer Kurvenfahrt
r_a :	Außenradius einer Kurvenfahrt
l_{ri} :	Länge der Innenbahnkurve
l_{ra} :	Länge der Außenbahnkurve
r_{bogen} :	Radius des Bogens
l_{bogen} :	Länge des Bogens, welchen der Benutzer bei einer Kurvenfahrt zurücklegen muss
I_{Motor} :	Stromaufnahme des Motors
U_q :	Versorgungsspannung des Motors

2.3 Elektrische Antriebe

Ein wesentlicher Teil dieser Arbeit beschränkt sich auf die Auswahl eines passenden Motors sowie die Ansteuerung für ein geeignetes Fahrverhalten. Die essenziell wichtigen Motoren sollen kurz mit ihren Eigenschaften und Funktionsweisen übersichtlich vorgestellt werden. Im Wesentlichen kommen für den Antrieb des EGD nur zwei Typen von Motoren in Frage. Zum einen sind dies die Brushless DC Motoren und zum anderen die Schrittmotoren.[16]

Elektrische Antriebe bzw. Maschinen gibt es in verschiedenen Arten[7]:

- Linearmotor
- Drehfeldmaschine
- Asynchronmaschine
- Synchronmaschine
- Gleichstrommaschine

Grundprinzip und Grundaufbau elektrischer Motoren

Die Aufgabe eines elektrischen Motors ist es, die zugeführte elektrische Energie P_{elek} in mechanische Energie P_{mech} umzuwandeln. Dabei gibt es vier wesentliche Eigenschaften, die den Motor charakterisieren:

- Drehmoment M in $[Nm]$
- Umdrehungsgeschwindigkeit RPM oder n in $[min^{-1}]$
- Leistung P_{mech} (Resultat aus Drehmoment und Umdrehungsgeschwindigkeit) in $[W]$
- Wirkungsgrad η

Das Drehmoment eines elektrischen Motors ist in der Regel von der Umdrehungsgeschwindigkeit abhängig. Damit ist auch die Leistung nicht zu jedem Punkt gleich. Viele Motoren haben einen speziellen Arbeitsbereich, in denen ihr Wirkungsgrad maximal ist und auch der Verschleiß am geringsten ist.

Grundaufbau einer elektrischen Maschine

Der Grundaufbau elektrischer Motoren ist meist sehr identisch. Dieser besteht grundsätzlich aus einem Stator, einem Rotor, einem Magneten, einem Kommutator und verschiedenen Anschlussklemmen. Die folgende Abbildung 2.1 soll dieses näher veranschaulichen.

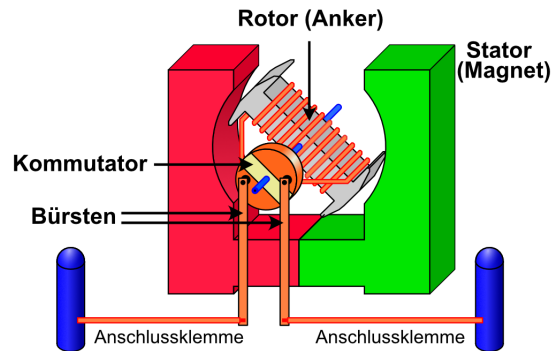


Abbildung 2.1: Grundaufbau eines elektrischen Motors[26]

Die verschiedenen Bauarten der Motoren haben spezielle Vor- und Nachteile und unterscheiden sich in ihrem Aufbau. Neben den genannten Bauelementen existieren auch weitere Bauteile, je nach Bauart des Motors, mit und ohne Schleifkontakte, mit verschiedenen Anschlussklemmen oder auch mit und ohne Permanentmagneten. Die Ausführungen sind zahlreich und liegen dem Einsatzgebiet zugrunde.

2.3.1 BLDC Motor

Der Brushless DC Motor ist eine Gleichstrommaschine und findet in vielen elektrischen Fahrzeugen seinen Einsatz.[19] Insbesondere im Modellbau und in *E Fahrzeugen* sind diese vorzufinden. Aufgebaut werden die BLDC Motoren meist in einem drei Phasen System (siehe Bild 2.2). Des weiteren unterscheidet man zwischen Innen- und Außenläufern (siehe Abbildung 2.3).

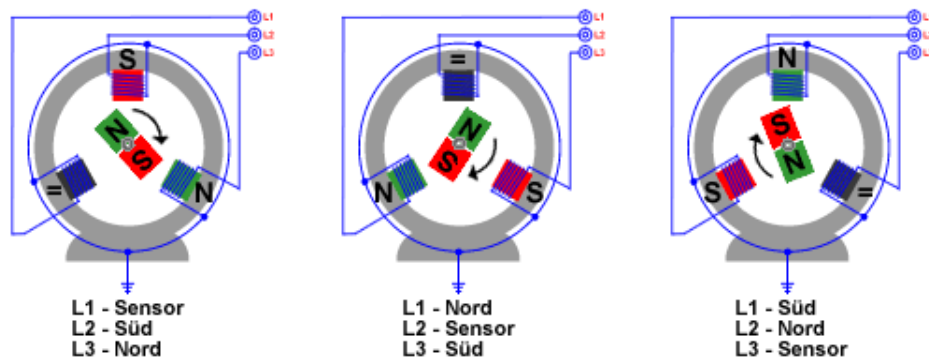


Abbildung 2.2: Grundprinzip eines BLDC Innenläufer Motors[19]

Auf dem folgenden Bild 2.3 sind zwei BLDC Motoren zu erkennen. Links ein Innenläufer und rechts ein Außenläufer. Beim Außenläufer befinden sich die Permanentmagneten auf dem äußeren drehenden Teil. Im Inneren wird das Drehfeld erzeugt.

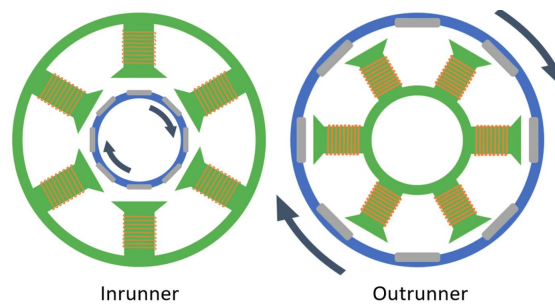


Abbildung 2.3: Aufbau: Innen- und Außenläufer[24]

Funktionsprinzip

Die Abbildung 2.2 veranschaulicht die Funktionsweise eines BLDCs. Das Prinzip der Kommutierung basiert auf der gezielten Ansteuerung der drei Spulen U,V und W. Bei einem Innenläufer werden die Phasen so bestromt, dass sich ein Drehfeld am Stator ergibt. Der innenliegende Permanentmagnet des Rotors würde in diesem Szenario dem Drehfeld des Stators folgen und sich entsprechend ausrichten.[19] Um dieses Drehfeld zu erzeugen benötigt es einen ESC (Electric Speed Controller), welcher im Verlauf erklärt wird.

Vorteile und Nachteile des BLDC

Die Brushless DC Motoren haben den Vorteil, dass sie ein sehr dynamisches und vibrationsfreies Fahrverhalten mit sehr hohen Drehzahlen ermöglichen können. Auch haben die BLDCs, im Vergleich zum Schrittmotor, kein Haltemoment und bieten somit eine frei drehbare Achse, sofern sie nicht angesteuert werden. Ein Nachteil ist jedoch das niedrige Drehmoment im unteren Drehzahlbereich.

2.3.2 Schrittmotor

Der Schrittmotor ist ein Motor welcher, im Vergleich zu einem BLDC keine Geschwindigkeit, sondern eine Position vorgibt. Er wird häufig in der Automation verwendet und ist dort meist in Robotern verbaut. Mithilfe von Funktionen wie dem *Microstepping* kann er sehr feine Auflösungen für hoch präzise Anwendungen, an der Welle umsetzen. Den Schrittmotor gibt es ähnlich wie den BLDC Motor in verschiedenen Ausführungen und Bauformen. Der im späteren Verlauf verwendete Schrittmotor ist ein zwei Phasen Hybrid Motor und gleichzeitig eines der meist verwendeten Modelle.

Funktionsprinzip

Anders wie bei einem BLDC Motor, macht der Schrittmotor mit jedem eingehenden Puls am Motortreiber einen sogenannten *Schritt*. Jeder Schritt bewegt die Welle des Antriebs um eine gewisse Gradzahl. Diese Schrittauflösung nennt man auch $STEP_{resolution}$. Der Aufbau eines typischen Pulssignals kann der folgenden Abbildung 2.4 exemplarisch entnommen werden.

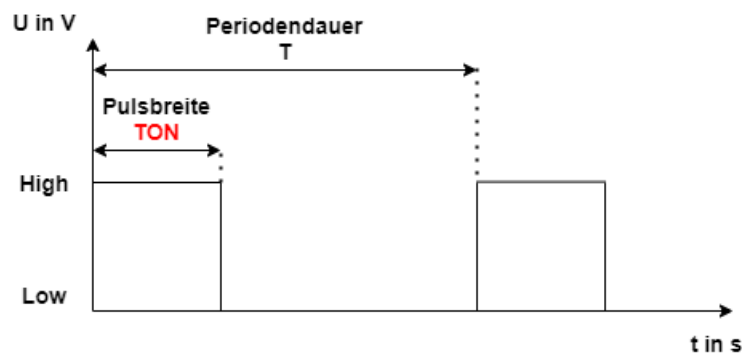


Abbildung 2.4: Pulssignal eines Schrittmotors

Die Geschwindigkeit des Schrittmotors kann über die Schrittfrequenz f_{step} bestimmt werden. Oftmals werden vom Hersteller Angaben zu Pulsbreite TON sowie der maximal zulässigen Schrittfrequenz $f_{step,max}$ gegeben. Die Pulshöhe beläuft dabei oftmals auf 5V. Der Aufbau eines Hybrid Schrittmotors kann der Abbildung 2.5 entnommen werden.

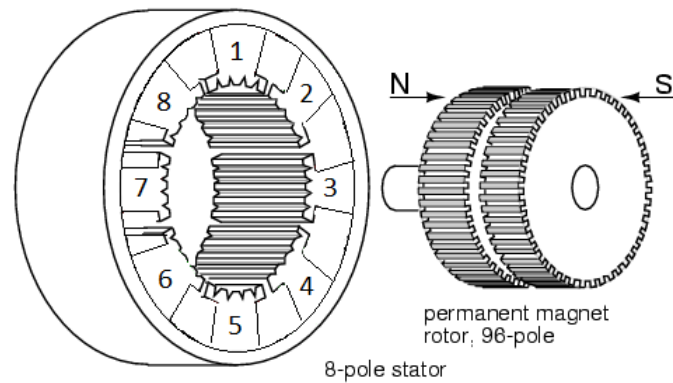


Abbildung 2.5: Aufbau Hybrid Schrittmotor[12]

Der Hybrid Motor ist eine Mischung aus einem Reluktanz-Schrittmotor und einem Permanentmagnet -Schrittmotor. [14] Zu erkennen auf der Abbildung 2.5 ist der Stator mit 8 Polen sowie der Rotor mit dem Permanentmagneten und einer Verzahnung n_{teeth} von 96 Polen. Diese Pole sind ausschlaggebend für den Schrittwinkel SA . In diesem Fall würde ein Schritt, im *Fullstep* Betrieb, $SA = \frac{360}{96} = 3.75^\circ$ entsprechen. Bei der Verwendung eines Schrittmotors ist darauf zu achten in welchem Betriebsmodus man sich befindet. Es gibt drei verschiedene Modi.

- Fullstep: Hohes Drehmoment, geringe Schrittauflösung
- Halfstep: Mittleres Drehmoment, mittlere Schrittauflösung
- Microstepping: Für hoch präzise Anwendungen, mit wenig Resonanzen aber kleineren Geschwindigkeiten

Der unterschied der Betriebszustände liegt in der Bestromung der Statorspulen und hat somit Einfluss auf die Schrittauflösung sowie das entstehende Drehmoment und die Drehgeschwindigkeit des Rotors.

Vorteile und Nachteile des Schrittmotors

Der Schrittmotor im Vergleich zum BLDC Motor, hat den Vorteil, dass dieser ein sehr hohes Drehmoment im unteren Drehzahl Bereich aufweist, dafür jedoch oftmals eine geringere maximale Drehzahl. Zusätzlich haben viele Schrittmotoren den Vorteil, dass diese relativ leise sind und eine sehr präzise Steuerung der Welle ermöglichen. Ein Nachteil ist jedoch die Stromaufnahme im Haltemoment. Das Haltemoment sorgt dafür, dass der Motor im Stillstand seine aktuelle Position beibehält. Intelligente Motortreiber können den Haltestrom und weitere Parameter begrenzen oder verändern.

2.3.3 Servomotor

Servomotoren sind Motoren die meist für eine bestimmte Auslenkung eines Systems zuständig sind. Oft werden Servomotoren in kleinen Lenksystemen eingebaut. Servomotoren oder auch Servos verfügen über ein Regelsystem (Servoregler) zum Überprüfen der aktuellen Position.

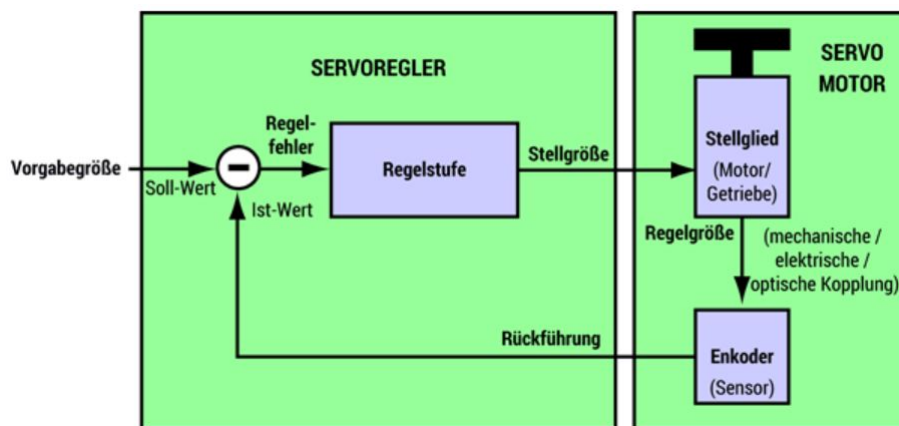


Abbildung 2.6: Aufbau eines Servomotors[17]

Mithilfe der Vorgabegröße (*Sollwert*), sowie des aktuellen Positionswerts *Ist-Wert*, wird eine Regeldifferenz ermittelt und mithilfe der Regelstufe des Servos, in eine passende Stellgröße umgewandelt (siehe Abbildung 2.6). Der Servomotor wird ähnlich wie ein BLDC mit einem PWM Signal über seinen Controller angesteuert.

2.3.4 Electric Speed Controllers (ESC) und Motortreiber

Die sogenannten ESCs oder auch Motortreiber, sind dafür zuständig die entsprechende Motoreinheit über die gezielte Ansteuerung der Motorphasen zum laufen zu bringen. Die ESCs kommen bei BLDC Motoren zum Einsatz und werden häufig über ein PWM Signal gesteuert, während die Motortreiber oder auch Motor Driver ihren Einsatz bei Schrittmotoren finden. Die Motortreiber für die Schrittmotoren haben oftmals zusätzliche Einstellungsmöglichkeiten, wie die Veränderung der Schrittweite oder die Begrenzung des Haltestroms. Auch wird der Treiber eines Stepper Motors nicht mit einem PWM Signal angesteuert, sondern mit einer Pulsfrequenz.

ESC

Der Electric Speed Controller arbeitet, je nach BLDC Aufbau, meist mit einer drei Phasen Brücke (siehe Abbildung 2.7).

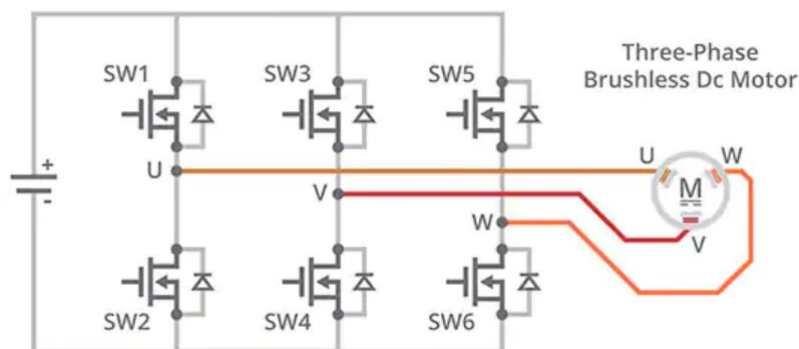


Abbildung 2.7: ESC: Mit Wechselrichterschaltung für die Kommutierung[23]

Der Controller bestromt die jeweilige Phase (U,V,W) mithilfe des Schließens der Schalter SW1, SW3 und SW5. Die Schalter bilden Paare, so dass wenn Schalter SW1 geöffnet wird, der dazugehörige Schalter SW2 geschlossen wird. Damit der Controller weiß, wann er welchen Schalter schließen muss gibt es zwei Möglichkeiten. Zum einen über das sogenannte Back EMF[23] (elektromagnetische Gegenkraft) und zum Anderen über die Informationen von einem am Stator angebrachten Hall Sensor. Viele ESCs im Modellbau arbeiten ohne Hall Sensoren, nur mit der Information des Back EMFs. Immer dann, wenn das sich drehende Magnetfeld des Rotors sich über eine Spule des Stators bewegt, wird

eine Spannung in diese induziert. Mit dieser Information kann der Controller dann die Position des Rotors bestimmen.

Motortreiber

Während der ESC seinen Einsatz bei dem BLDC Motor findet, ist der Motortreiber im allgemeinen für den Schrittmotor zuständig. Auch der Motortreiber für die Schrittmotoren arbeitet ähnlich wie ein normaler ESC und bestromt in einer bestimmten Abfolge die Spulen des Stators. Dennoch gibt es hier Unterschiede in der Schalttechnik. Die Schrittmotoren können zum einen im Bipolaren und zum anderen im Unipolaren Modus betrieben werden. Da die genau Funktionsweise und Beschaltung sehr aufwendig und für den Verlauf der Arbeit nicht von großer Bedeutung ist, wird im Folgenden darauf verzichtet eine genauere Erläuterung zu geben.

3 Stand der Technik

Im Folgenden wird ein Blick auf den Stand der Technik geworfen. Hierbei werden ähnliche Projekte des Electronic Guide Dog betrachtet und vorgestellt. Ziel soll es sein, herauszufinden wie Antriebe und Steuerung in ähnlichen Projekten aufgebaut und konzipiert worden sind, so dass mithilfe dieser Informationen eine neue Antriebssteuerung inklusive Motor entworfen und ausgewählt werden kann.

3.1 Electronic Guide Dog

Projekte oder voll einsatzfähige elektrische Blindenführhunde sind derzeit nur schwer zu finden. Dies liegt an den vielen Herausforderungen, mit denen solche konfrontiert werden. Vermutlich erst mit dem Fortschritt der Technik und somit der Minimierung der Risiken, werden in Zukunft mehrere Projekte dieser Art erscheinen. Im Vergleich dazu sind im Bereich des autonomen Fahrens unzählige Sicherheitssysteme verbaut, welche über viele Jahre entwickelt und getestet worden sind. Zusätzlich ist die Arbeitsumgebung von Kraftfahrzeugen und autonomen Fahrzeugen, durch gut markierte Straßenlinien und Kennzeichnung mithilfe von Schildern, deutlich besser im Vergleich zu einfachen Fußwegen. Obwohl viele dieser Voraussetzungen im Einsatzbereich sowie auch in der Technik des autonomen Fahrens deutlich besser sind, ist nur ein sehr kleiner Teil der neuen Fahrzeuge heutzutage vollständig auf das autonome Fahren eingestellt. Auch gibt es noch zahlreiche Sicherheitslücken und weiteren Forschungsbedarf in der Entwicklung.

Die folgenden Projekte eines elektronischen Hilfsmittels bzw. Blindenführhunds sollen veranschaulichen, welche verschiedene Konzeptionsmöglichkeiten es für die Gestaltung des Antriebssystems gibt.

Theia Projekt

In dem Theia Projekt, welches von einem Studenten der Universität Loughborough in England entworfen wurde, wird anders wie bei diesem Projekt zwar auch ein elektrischer Motor verwendet, allerdings nicht für einen herkömmlichen Antrieb zum Ziehen des Blinden. Diese Richtungsweisung geschieht über das Prinzip der Kreiselpräzission auch *Gyroscopic Precession* genannt. Ein solches *Gyroscop* wird oftmals in der Raumfahrttechnik in Verbindung mit Lagekontrollsystemen verwendet. Der Blinde hat somit nur ein kleines Gerät mit der integrierten Technologie in der Hand. Durch die intern drehende Mechanik mithilfe der Kreiselpräzission, hat der Sehbehinderte das Gefühl in eine Richtung gelenkt zu werden.[25]

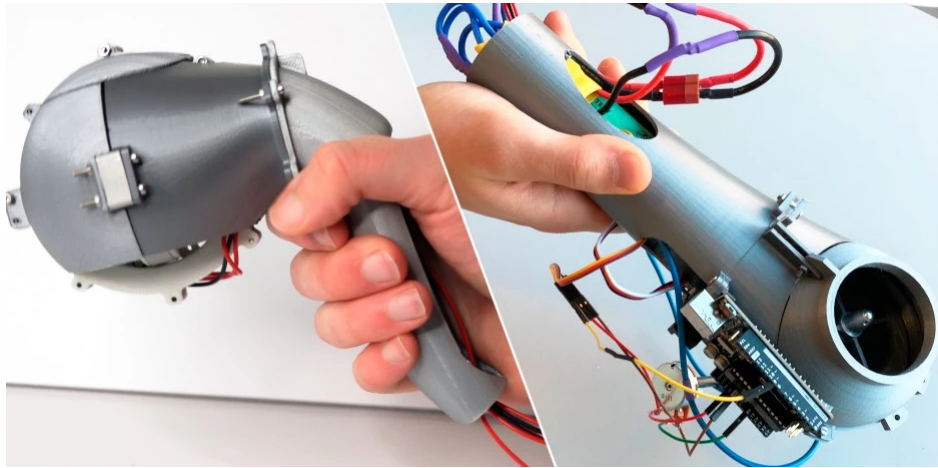


Abbildung 3.1: Theia Projekt[25]

Auf der Abbildung 3.1 ist der Prototyp zu erkennen. Der Motor für die Erzeugung der Kreiselpräzission befindet sich im inneren des Gehäuses.

Blindbot Projekt

In einem anderen Projekt dem sogenannten Blindbot aus dem Jahr 2016[29], haben dessen Erfinder Alexander Bayer und Niklas Gutsmedl, einen Blindenführhund entworfen, dessen Antrieb über ein Ballsystem läuft. Hierbei wird von zwei Motoren ein Ball angesteuert, so dass eine Richtungsänderung um volle 360 Grad ermöglicht wird. Genauere Infos zu den verwendeten Motoren, dem Antrieb oder auch zu dem Projekt selber

sind nicht weiter veröffentlicht worden. Im folgenden Bild 3.2 kann man den sogenannten Blindbot und dessen Antrieb erkennen.



Abbildung 3.2: Blindbot mit Ballsystem als Antrieb. Entwickler: Alexander Bayer und Niklas Gutmiedl[29]

MYGO Projekt

Im letzten vorzustellenden Projekt MYGO von Sebastian Ritzler[20], welches dem aktuellen Aufbau unseres EGD sehr ähnelt, wird ebenfalls ein langer Stab verwendet, der von einem Blinden gehalten wird. An dem Ende des Stabs ist eine kleine Antriebseinheit verbaut, die aus einem Radnarbenmotor besteht. Diese Einheit soll für den Zug des Blinden und die Richtungsweisung zuständig sein. Auch bei diesem Projekt wurden keine weiteren Details zu dem Gesamtaufbau und dessen Antriebseinheit gegeben. Das folgende Bild 3.3 soll den Prototypen MYGO veranschaulichen.



Abbildung 3.3: Projekt MYGO, Zeichnung des Blindenstock mit Antrieb am Stabsende.
Entwickler: Sebastian Ritzler[20]

Ein serienreifer voll funktionsfähiger elektronischer Blindenhund inklusive detaillierter Dokumentation, ist zur Zeit noch nicht zu finden. Die meisten Projekte befinden sich in der Anfangsphase sowie in der Konzeption und Entwicklung eines Prototypens. Für die Entwicklung eines neuen Antriebssystems, sollte daher ein genauerer Blick auf Antriebe geworfen werden, welche von der Größenordnung vergleichbar mit der des aktuellen Blindenführhunds sind. Dies wären zum Beispiel Antriebseinheiten für E-Skateboards oder E-Roller, genauso wie kleine Gartenroboter oder elektrische Rollatoren. Diese Geräte verfügen über elektrische Antriebe welche in der Lage sind hohe Lasten über längere Zeit zuverlässig und effizient zu transportieren.

3.2 Antriebssysteme für elektrische Fahrzeuge

Nachfolgend soll geschaut werden, welche Antriebe in welchen elektrischen Fahrzeugen verbaut werden. Dies soll einen Einblick zum aktuellen Stand der Technik geben. Außerdem kann dies bei der Auswahl eines neuen Antriebs von Vorteil sein.

Viele elektrische Fahrzeuge verwenden abhängig ihres Einsatzgebiets unterschiedliche Motoren und Antriebssysteme. Im Bereich der E-Mobilität sind die folgenden Varianten sehr vertreten:

- Radnabenmotor (HUB Motor)
- Direkte Umsetzung (Direct Drive)
- Motor mit Riemenantrieb (Belt Driven Motor)
- Motor mit Getriebe (With Gear)

Riemenantrieb mit BLDC Motor

Der im Kapitel 2.3.1 beschriebene BLDC Motor ist ein gängiger Elektromotor für viele elektrische Kleinfahrzeuge. Er ist der Nachfolger des Brushed DC Motors (Gleichstrom-Bürstenmotors) und bietet hohe Drehzahlen bei gutem Wirkungsgrad und geringem Verschleiß. Oftmals findet man solche BLDC Motoren in kleineren RC-Autos oder in E-Skateboards. Da das Drehmoment im unteren Drehzahlbereich oftmals nicht ausreichend ist, wird meist eine Untersetzung mithilfe eines Getriebes oder eines Riemens verbaut. Das folgende Bild 3.4 eines E-Skateboards verwendet ein BLDC Motor mit einer Untersetzung, die über einen Riemen realisiert wurde ist.



Abbildung 3.4: Belt Driven E-Skateboard[10]

Nicht nur die Verwendung eines Riemens bietet eine Änderung des Drehmoments, sondern bietet dies auch einen gewissen Schutz, da der Motor nicht unmittelbar an der Achse des Reifens befestigt ist. Speziell Elektromotoren sind sehr empfindlich und können durch starke Erschütterungen schnell beschädigt werden. Ein weiterer Einsatzort des BLDC Motors findet sich ähnlich wie beim Schrittmotor in der Robotik wieder.

Der Roboter namens *Cheetah* von Boston Dynamics, auf der Abbildung 3.5 verwendet für seinen speziellen Antrieb einen BLDC Motor.

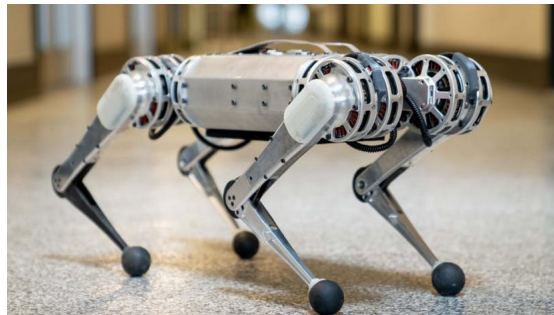


Abbildung 3.5: Roboter Cheetah, von Boston Dynamics[13]

Radnabenmotor - HUB Motor

Neben dem BLDC Motor mit Riemen, werden auch häufiger HUB Motoren verwendet. Insbesondere in E-Rollern, E-Bikes aber auch in E-Skateboards sind diese vorzufinden. Das folgende Bild 3.6 soll dies veranschaulichen.



Abbildung 3.6: E-Skateboard mit HUB Motor[28]

Der HUB Motor oder auch Radnabenmotor ist so konzipiert, dass dieser direkt in das Gehäuse des Reifens verbaut wird und die Radnabe trägt. Dies hat den Vorteil, dass das

System sehr kompakt ist und dank des Gehäuses oftmals auch eine geringere Geräuschentwicklung hat. Meist bieten die HUB Motoren auch hohe Umdrehungsgeschwindigkeiten und durch die Größe auch ein relativ hohes Drehmoment. Dafür haben diese jedoch meist einen höheren Verschleiß und eine starke Wärmeentwicklung. Besonders die Reifen können in solchen Systemen nicht sehr einfach gewechselt werden.[28]

Oftmals werden HUB Motoren als Außenläufer konzipiert, da die spezielle Bauform ein höheres Drehmoment ermöglicht. Um das Drehmoment noch weiter zu verbessern, gibt es auch HUB Motoren mit verbautem Getriebe. Der Aufbau eines HUB Motors mit und ohne Getriebe, ist im folgenden Bild 3.7 zu erkennen.



Abbildung 3.7: HUB Motor mit und ohne Getriebe[31]

Wie bereits erwähnt wurde, gibt es verschiedene Antriebssysteme, welche je nach Einsatz unterschiedliche Vor- und Nachteile mit sich bringen. Für die Auswahl eines passenden Motors und die Wahl der Antriebsart ist es wichtig zu wissen, welchen Anforderungen und Umständen der Motor unterliegt. Für eine grobe Auswahl ist es empfehlenswert die Motoren nach der benötigten Geschwindigkeit, des Drehmoments sowie der Größe und des Gewichts zu wählen. Auch kann im Anschluss auf den Anwendungsfall, die Geräuschentwicklung, die Bauform, sowie die Effizienz und Belastbarkeit eingegangen werden. Im nächsten Kapitel sollen die zentralen Anforderungen des Motorsystems für den elektronischen Blindenführhund näher besprochen werden.

4 Anforderungen an das Antriebssystem des EGD

In diesem Kapitel werden die wichtigsten Anforderungen an den Antrieb und die Software beschrieben. Speziell die Anforderungen für ein passendes und dynamisches Fahrverhalten sind in diesem Projekt schwer zu fassen. Dies liegt unter anderem daran, dass mit verschiedenen Benutzern andere Fahrverhalten auftreten. Besonders die Schrittgeschwindigkeit aber auch das Beschleunigungsverhalten und die Reaktionszeit variieren hier von Benutzer zu Benutzer sehr stark.

4.1 Allgemeine Anforderungen

- Maximale Geschwindigkeit $5 \frac{km}{h}$, entspricht der durchschnittlichen Schrittgeschwindigkeit eines Menschen[11]
- EGD muss beliebige Streckenprofile mit Kurven fahren können
- EGD muss Steigungen bis zu 9 Grad überwinden können. Die Steigung orientiert sich an den Vorgaben zur zulässigen genormten Rampensteilheit[30] und der größten Straßensteigung in Deutschland[38]
- EGD soll normale Straßen und glatte Oberflächen befahren können, dabei sollen Nässe, Sand und kleine Steinchen oder Dreck überwunden werden können

4.2 Anforderungen für die Auswahl einer neuen Antriebseinheit

Im Folgenden werden zentrale Anforderungen an die neue Antriebseinheit Stichwortartig aufgelistet.

- Auswahl eines geeigneten Motors
- Motor bzw. Antrieb muss die nötige mechanische Leistung aufbringen
- Maximalgewicht der gesamten Antriebseinheit 3 kg. Antriebseinheit darf nicht zu schwer werden, da ansonsten das maximale Gesamtgewicht des EGDs überschritten wird
- Maximales Gewicht des elektronischen Blindenführhunds 6 kg
- Maximale Betriebsspannung $\leq 24\text{ V}$. Für die Realisierung mit einem Akku, sollte die Spannung nicht zu groß sein
- Maximaler Dauerstrom des Antriebssystems $\leq 4\text{ A}$. Stromaufnahme sollte aufgrund der Verwendung eines Akkus nicht zu groß sein
- Antrieb muss leise sein und darf den Sehbehinderten in der Geräuschwahrnehmung nicht behindern
- Reifendurchmesser Minimum 8 cm. Die Räder dürfen nicht zu klein werden, da ansonsten die Auflagefläche zu gering ist und die Gefahr besteht, das kleinste Hindernisse nicht mehr überwunden werden können. Auch darf der Reifen nicht zu klein werden, zumal sonst mit steigender Umdrehungszahl auch der Geräuschpegel, durch die Umdrehungen des Motors zunimmt
- Höhe und Breite des Motors dürfen nicht über 7 cm schreiten, die Tiefe bzw. Länge (mit Welle) darf maximal 8 cm betragen. Der neue Antrieb muss so gewählt werden, dass er in dem Aufbau (siehe Kapitel 1.2 Abbildung 1.2) passend verbaut werden kann
- Die Welle des Motors muss einer radialen Dauerbelastung von mindestens 50 Newton standhalten können. Entspricht ungefähr der Belastung, die das Gesamtsystem auf den Motor ausübt

- Motor muss verfügbar bzw. schnell zu erwerben sein. Lange Lieferzeiten beeinflussen die Bearbeitung des Projekts und müssen daher vermieden werden
- Motoren inklusive Steuereinheit dürfen nicht über 500 Euro kosten (Gesamtbudget 500 Euro)

4.3 Anforderungen an die Software

- Software für die Antriebssteuerung und Bahnführung muss auf dem TIVA TM4C1294 Mikrocontroller implementiert werden können. Der TIVA wurde bereits für die alte Antriebseinheit verwendet und soll weiterhin die Grundlage für die Software darstellen
 - Software muss zusätzlich in das bestehende Projekt *watchdog* passend eingebaut werden können und die alte Antriebssteuerung ersetzen
- Mikrocontroller im Zusammenhang mit der Software müssen den Antriebsregler des Motors so ansteuern können, dass eine vorgegebene Geschwindigkeit am Motor umgesetzt werden kann
- Software muss Funktionen für die Umsetzung eines passenden Fahrverhaltens und einer Bahnführung implementieren. Das passende Fahrverhalten orientiert sich maßgeblich am Benutzer und wird im Kapitel 5.3 näher erklärt
- Mikrocontroller und Software sollen den Antriebsregler des Motors konfigurieren können z.B.:
 - Antriebsmodi - Wenn der Treiber über diese Funktion verfügt
 - Vorwärts - Rückwärts
 - Microsteps - bei Verwendung eines Schrittmotors
 - Strombegrenzung - Wenn der Treiber über diese Funktion verfügt
- Gegebenenfalls muss ein digitaler Regler implementiert werden, welche die Fahrtrichtung des EGD mithilfe eines Kompasssensor korrigiert

4.4 Übersicht zu den Anforderungen

Die folgende Tabelle 4.1 soll einen Überblick zu den Anforderungen und den dazugehörigen Formelzeichen geben.

Beschreibung	Zeichen	Wert	Einheit
Max. Gewicht	m_{EGD}	6	kg
Max. Geschwindigkeit	v_{EGD}	5	$\frac{km}{h}$
Min. Reifendurchmesser	D_{min}	8	cm
Max. Reifendurchmesser	D_{max}	12	cm
Max. Steigung / Winkel	$\alpha_{steigung}$	9	deg
Max. Spannung	$U_{q,max}$	24	V
Max. Stromaufnahme	$I_{motor,max}$	4	A

Tabelle 4.1: Physikalische Beschreibung der Eigenschaften und Anforderungen des EGD

5 Konzeptionierung

In diesem Teil soll ein Konzept für das Antriebssystem entwickelt werden, welches auf die gegebenen Anforderungen aus Kapitel 4 ausgelegt ist. Das Kapitel umfasst die folgenden drei wichtigen Bereiche:

- Auswahl der Antriebsart - Konzept für Aufbau und Art des Antriebs[35]
- Auswahl des Motors
- Konzeption und Softwareentwurf der neuen Fortbewegungsstrategie

Begonnen wird mit der Vorstellung einer Antriebsart für den EGD.

5.1 Konzept für Aufbau und Art des Antriebs

Der allgemeine Aufbau des EGD - Prototypen aus Kapitel 1 soll weiterhin nicht verändert werden. Für das neue Antriebssystem soll ausschließlich der untere Teil des EGD mit Rädern, Motoren, Lenkachse und der Software verändert werden.

Der aktuelle Prototyp auf dem Bild 1.2 verfügt über einen kleinen BLDC Motor aus einem RC Auto und ist für den hier beschriebenen Aufbau und die Anforderungen nicht passend ausgelegt. Hinzu kommt, dass der Motor sehr laut ist und nicht für eine hohe Achsenbelastung geschaffen ist.

Das nachstehende Bild 5.1 verdeutlicht den allgemeinen Aufbau des elektronischen Blindenhunds, inklusive des Benutzers.

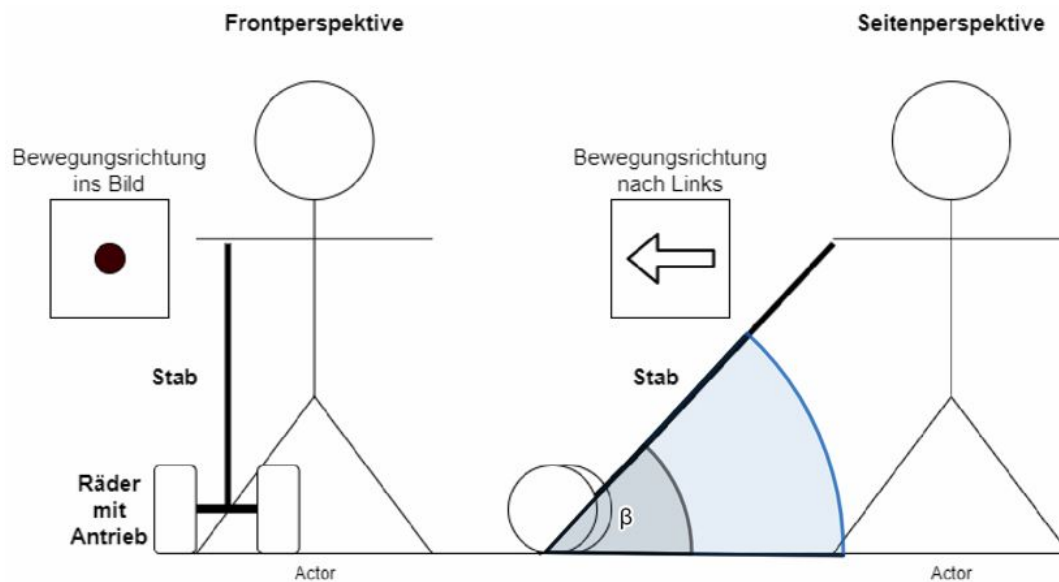


Abbildung 5.1: Zeichnung des allgemeinen Aufbaus mit Stab, Rädern, Antrieb und Benutzer - Front und Seitenperspektive

Auf der Abbildung 5.1 ist der elektronische Blindenhund als Zeichnung, mit Antrieb, Rädern, Stab und Benutzer zu sehen. Hierbei hält der Benutzer oder Sehbehinderte den EGD am Stabende fest und wird mithilfe des Antriebs navigiert. Des Weiteren ist ein Haltewinkel β zu erkennen. Dieser Winkel hängt von der Größe des Benutzers und der Länge des Stabes l_{stab} ab. Insbesondere für das zweite Antriebskonzept ist dieser Winkel zu berücksichtigen.

5.1.1 Wahl des Antriebskonzepts

Für das neue Antriebssystem kommen zwei Antriebsarten in Frage. Diese werden im Folgenden kurz vorgestellt und erläutert. Am Ende wird eine Auswahlentscheidung zu einem bestimmten Konzept getroffen, welches dann für die Entwicklung der Fortbewegungsstrategie und dessen Software verwendet wird.

5.1.2 Erstes Konzept - Zwei Antriebsmotoren ohne Lenkachse mit Differential

Dieses Konzept basiert auf zwei Antriebsmotoren, welche für den nötigen Antrieb und gleichzeitig auch für die Lenkung des EGDs zuständig sind. Die folgende Abbildung 5.2 soll dabei diesen Aufbau veranschaulichen. Neben dem Aufbau sind noch weitere Maßeinheiten wie der Durchmesser der Reifen D , die Länge des Stabes l_{stab} und der Abstand der zwei Reifen zueinander L_R zu erkennen. Diese Maßeinheiten sind ebenfalls stellvertretend für den Aufbau des zweiten Konzepts.

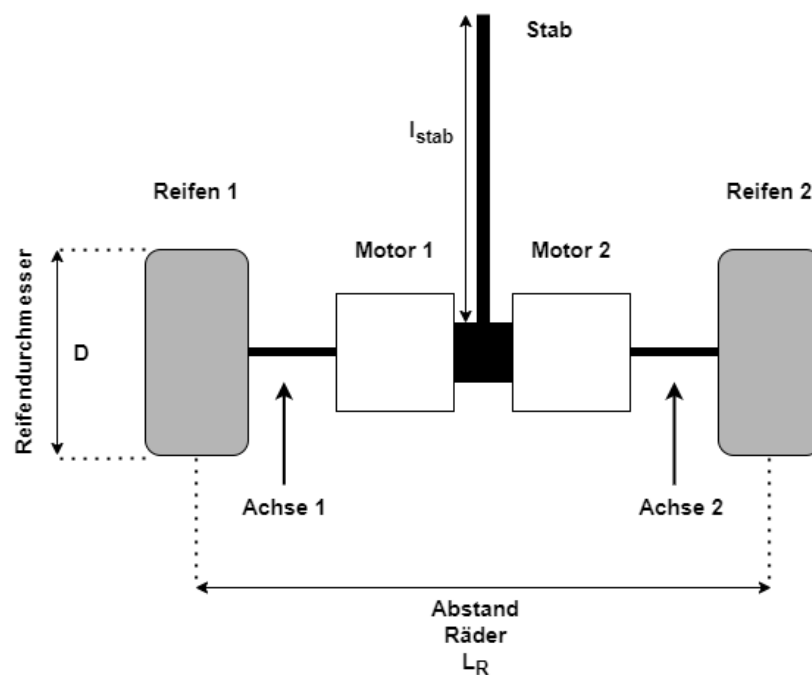


Abbildung 5.2: Aufbau des ersten Antriebkonzepts

Anhand der Abbildung 5.2 ist zu erkennen, dass das Antriebssystem aus zwei Motoren besteht, dessen Achsen nicht miteinander verbunden sind. Jeder Motor kann somit einzeln über die Software angesteuert und auf eine gewisse Geschwindigkeit gebracht werden. Verschiedene Fahrmanöver, wie zum Beispiel Kurvenfahrten, entstehen dann durch die passende Ansteuerung des jeweiligen Motortreibers oder Speed-Controllers. Dieser Controller verarbeitet das entsprechende Signal und treibt den dazugehörigen Motor sowie Reifen an.

Ein Vorteil dieses Aufbaus ist, dass die benötigte Antriebsleistung auf zwei Motoren verteilt wird und jeder Motor somit nur noch die Hälfte der Leistung erbringen muss. Da dieser Aufbau keine Lenkmechanik, sondern nur ein Gehäuse zum Befestigen der zwei Motoren und des Stabs benötigt, ist diese Konstruktion schneller und leichter, im Vergleich zum zweiten Konzept, aufzubauen. Außerdem wird kein weiterer Servomotor für die Lenkung benötigt. Auch spielt es keine Rolle in welchem Haltewinkel (β) (siehe Abbildung 5.1) der Stab vom Benutzer gehalten wird.

Ein Nachteil entsteht jedoch durch die noch starre Befestigung des Stabs. Dieser ist aktuell noch fest zwischen den zwei Motoren montiert. Sofern der EGD nun ein Lenkmanöver nach links oder rechts einleitet, wird der Benutzer automatisch in die jeweilig entgegengesetzte Richtung gezogen.

Mit einem passendem Gelenk könnte man diesem Problem entgegenwirken. Siehe dazu folgende Grafik 5.3:

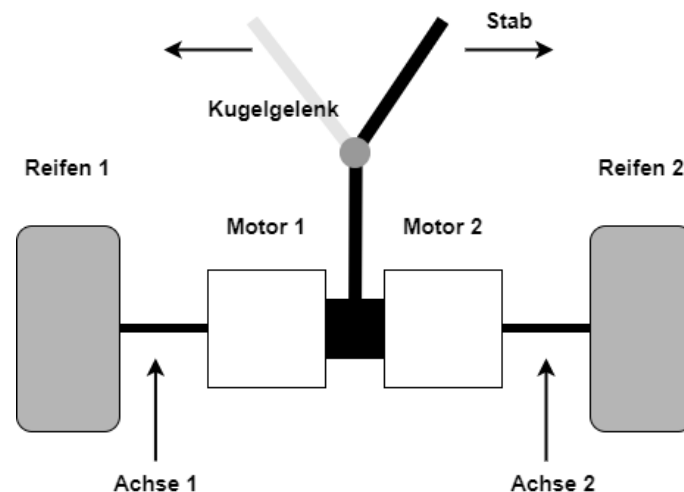


Abbildung 5.3: Aufbau des ersten Antriebskonzepts mit Kugelgelenk

In diesem Aufbau (Bild 5.3) wurde dem Stab ein Kugelgelenk hinzugefügt. Dies ermöglicht ein besseres Fahrverhalten, da der Benutzer bei einer Kurve nicht unmittelbar durch den vorher starren Stab, zur Drehung gezwungen wird.

5.1.3 Zweites Konzept - Ein Antriebsmotor mit Lenkachse

Das zweite Konzept ähnelt dem Antrieb eines einfachen PKWs, wobei hier jedoch nur zwei Räder zum Einsatz kommen, anstatt vier. Der Aufbau des zweiten Konzepts basiert auf einem Antriebsmotor und einem Servomotor, sowie einer Mechanik, die es ermöglicht die Radposition zu verstellen. Diese zwei Räder sind mechanisch mit der Antriebsachse und dem Motor verbunden und sind nur für den Antrieb zuständig. Die Lenkachse wird über einen Servomotor für das gewünschte Lenkmanöver in die Richtige Position gebracht. Siehe dazu die folgende Zeichnung des zweiten Konzepts (Abbildung 5.4).

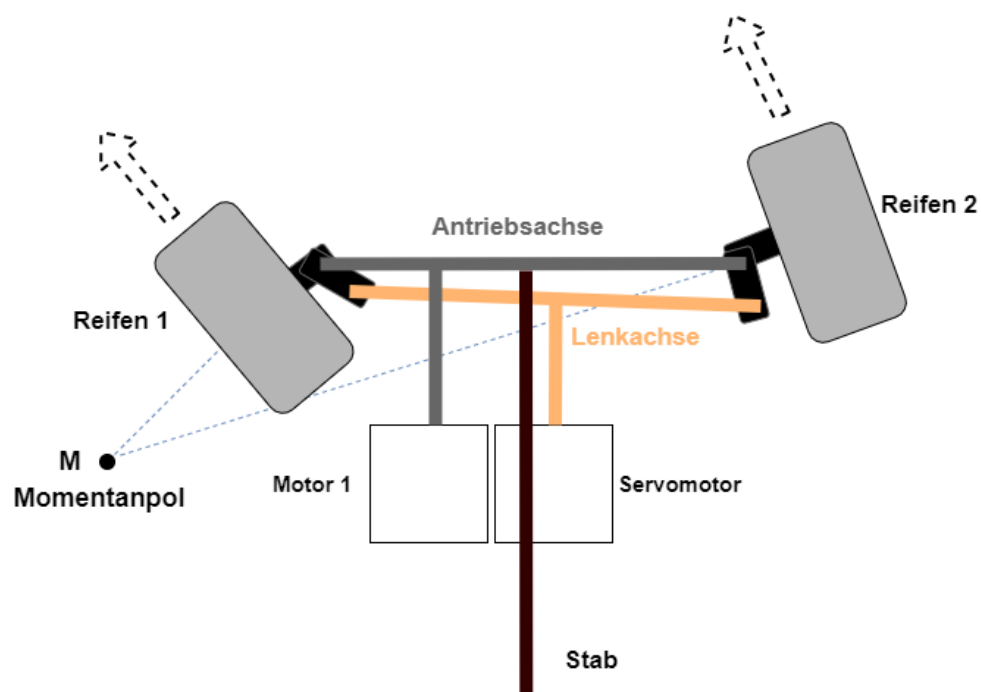


Abbildung 5.4: Aufbau des zweiten Antriebskonzepts

Auf der Abbildung 5.4 sind wie auch beim ersten Konzept die zwei Reifen sowie der Stab zu erkennen. Allerdings wird für diese Art von Aufbau nur ein Antriebsmotor verbaut. Anstatt des zweiten Antriebsmotors wird eine Lenkmechanik mit einem Servomotor benötigt, welcher diese Lenkeinheit ansteuert und in die gewünschte Position bringt. Maßeinheiten für Reifenabstände, Reifendurchmesser und Stablänge sind aus dem ersten Konzept zu übernehmen.

Dieses System hat den Vorteil, dass die Umsetzung der Lenkung durch die Lenkachse deutlich einfacher zu implementieren ist, wie im Vergleich zum ersten Konzept. Auch wird sich dadurch ein besseres Fahrverhalten ergeben, da sich der untere Teil der Antriebseinheit in verschiedene Richtungen neigen kann und der Benutzer bei einer Kurve nicht sofort nach außen gezogen wird.

Der Nachteil ist jedoch der erhöhte Konstruktionsaufwand insbesondere in Verbindung mit der Lenkachse. Eine Konstruktion würde demnach länger dauern, als ein einfaches Gehäuse zum Halten bzw. Befestigen der beiden Motoren. Da in diesem Aufbau nur ein Antriebsmotor verwendet wird, muss dieser auch die doppelte Leistung erbringen. Dieses kann bei der Wahl eines passenden Motors ebenfalls eine wichtige Rolle spielen. Zudem kommt hinzu, dass der bereits angesprochene Haltewinkel einen Einfluss auf die Lenkung hat. Denn sofern der Stab senkrecht zum Boden ($\beta = 90^\circ$) ist, werden die Reifen bei einer Neigung der Lenkachse nicht mehr wie vorgesehen am Boden aufliegen. Demnach müsste man dem Antriebssystem als Lösung ein weiteres Stützrad sowie ein Gelenk hinzufügen, so dass die Reifen immer denselben Aufwinklung zum Boden haben.

Die folgende Abbildung 5.5 veranschaulicht das zweite Konzept mit einem Stützrad und einem passenden Gelenk.

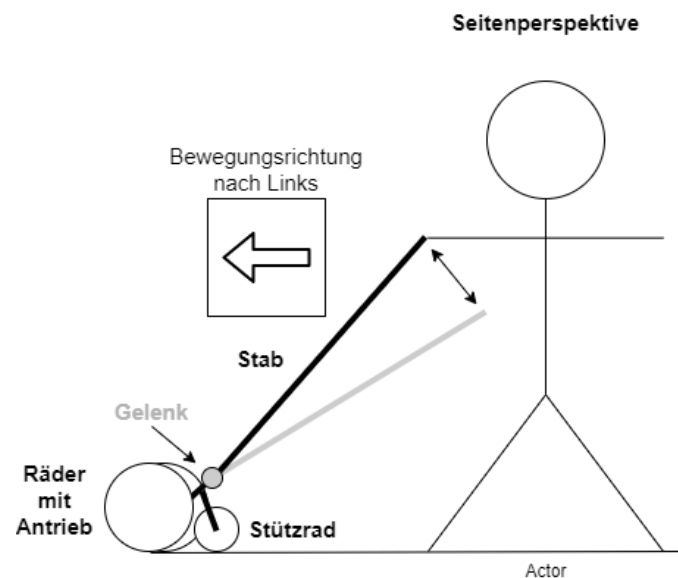


Abbildung 5.5: Aufbau des zweiten Antriebskonzepts mit Stützrad und Gelenk

Auf der Abbildung 5.5 ist zu sehen, dass durch den Einbau des Stützrades und des Gelenks sich ein gleichbleibender Winkel zwischen Boden und unterem Stabende einstellt.

Übersicht und Auswahl des Antriebkonzepts

Die Vor- und Nachteile der Antriebssysteme werden in der Abbildung 5.6 zusammengefasst.

	Konzept 1	Konzept 2
Vorteile	Geringer Konstruktionsaufwand	Lenkverhalten einfach zu implementieren
	Unabhängig vom Haltewinkel	Besseres Fahrverhalten durch Lenkachse
	Leistungsverteilung auf zwei Motoren	
Nachteile	Lenkung schwieriger zu implementieren	Antriebsleistung wird von einem Motor getragen
	Kurvenfahrten können unerwünschtes Lenkverhalten hervorrufen	Haltewinkel beeinflusst Fahrverhalten
		Größerer Konstruktionsaufwand

Abbildung 5.6: Übersicht zu den Vor- und Nachteilen der zwei Antriebskonzepte

In dieser Arbeit wurde sich für das erste Konzept entschieden. Ein wichtiger Anlass für diese Entscheidung war der geringere Konstruktionsaufwand sowie die damit einhergehende schnelle Produktion mit dem 3D Drucker und die Aufteilung der Gesamtmotorleistung auf zwei Antriebsmotoren.

5.2 Auswahl des Motors

In diesem Teil soll ein passender Motor für die genannten Anforderungen und das entsprechende Antriebskonzept 1 ausgewählt werden. Damit der Antrieb des elektronischen Blindenführhunds während des Einsatzes nicht durch schlechte Straßenverhältnisse oder die starke Zugkraft des Benutzers gestört wird, werden die Motoren stärker dimensioniert.

5.2.1 Berechnung der benötigten Antriebsleistung

Die Antriebsleistung ist eine wichtige Größe, die für die Auswahl des passenden Motors maßgeblich von Bedeutung ist.[32] Für die Berechnung der benötigten Antriebsleistung werden folgende Informationen in der Tabelle 5.1 bereitgestellt. Weitere Informationen sind im Kapitel 4 Anforderungen an das Antriebssystem, sowie im Kapitel 4.4 Übersicht zu den Anforderungen, zu finden. Um zu verhindern, dass der Motor im späteren Einsatz unterdimensioniert ist, werden in den darauffolgenden Berechnungen lediglich die Maximalwerte in Betracht gezogen.

Beschreibung	Zeichen	Wert	Einheit
Gewicht EGD	m_{EGD}	6	kg
Geschwindigkeit EGD	v_{EGD}	1.39	m/s
Reifendurchmesser	D	0.1	m
Radius Reifen	r	0.05	m
Steigung	$\alpha_{steigung}$	9	deg
Rollreibungskoeffizient	μ_r	0.15	
Luftdichte	ρ	1.22	kg/m^3
Umströmte Querschnittsfläche	A	0.3	m^2
Luftwiderstandszahl	c_w	0.3	

Tabelle 5.1: Parameter für die Berechnung der Motorleistung

Die Tabelle 5.1 listet die wichtigsten Größen für die Leistungsberechnung auf. Insbesondere bei der Wahl des entsprechenden Rollreibungskoeffizienten μ_r wurde sich an Mess- und Erfahrungswerten der Literatur orientiert. Zusätzlich wird dieser Koeffizient von verschiedenen Faktoren beeinflusst. Insbesondere die Straßenverhältnisse, sowie auch die Reifengröße (Auflagefläche) und das Material spielen eine wichtige Rolle.

Der verwendete Rollreibungskoeffizient c_r wurde wie folgt ermittelt.

Mit dem Rollreibungswert eines Reifens auf losem Sand von $c_{sand} = 0.3$ und einem Rollreibungswert eines Fahrradreifens auf einer Straße von $c_{strasse} = 0.004$, ergibt sich folgender Mittelwert für c_r :

$$c_r = \frac{c_{sand} + c_{strasse}}{2} = \frac{0.3 + 0.004}{2} = 0.152 \approx 0.15. \quad (5.1)$$

$$c_r = 0.15. \quad (5.2)$$

Bei dieser Berechnung wurde davon ausgegangen, dass der elektronische Blindenführhund auch auf schlechte Straßenverhältnisse mit Sand und anderen kleinen Hindernissen treffen kann. Daher wurde ein deutlich höherer Reibungswert angenommen wie unter optimalen Straßenbedingungen.

Zunächst soll noch eine Übersicht zu den im folgenden verwendeten Formelzeichen gegeben werden. Siehe Tabelle 5.2

Formelzeichen	Einheit	Beschreibung
U_{Rad}	m	Umfang Rad
n	min^{-1}	Umdrehungsgeschwindigkeit
v_{EGD}	m/s	Geschwindigkeit des EGD
ω	rad/s	Winkelgeschwindigkeit
$\dot{\omega}$	rad/s^2	Winkelbeschleunigung
F_{Roll}	N	Rollreibungskraft
μ_r		Rollreibungskoeffizient
$F_{Steigung}$	N	Steigungswiderstandskraft
$F_{H,res}$	N	Resultierende Hangabtriebskraft
F_{Luft}	N	Strömungswiderstandskraft
F_{Zug}	N	Zugwiderstandskraft
η		Wirkungsgrad des Motors
P_{mech}	W	Mechanische Leistung des Motors
P_e	W	Elektrisch zugeführte Leistung
M_L	Nm	Lastdrehmoment
$M_{\dot{\omega}}$	Nm	Beschleunigungsdrehmoment
M_{Rad}	Nm	Trägheitsdrehmoment

Tabelle 5.2: Beschreibung der Formelzeichen für die Berechnungen

Übersicht zu den Formeln für die Bestimmung der Leistung und des Drehmoments

Berechnung der Leistung P :

$$P = M \cdot 2 \cdot \pi \cdot n \cdot \frac{1}{60}. \quad (5.3)$$

Berechnung des Drehmoments M :

$$M = F \cdot r = \frac{P}{\omega}. \quad (5.4)$$

Berechnung der Drehfrequenz f mit dem Umfang des Reifens U :

$$f_{Rad} = \frac{v}{U}. \quad (5.5)$$

Berechnung der Umdrehungen pro Minute n :

$$n = f \cdot 60 \quad (5.6)$$

Berechnung der Winkelgeschwindigkeit:

$$\omega = 2\pi f. \quad (5.7)$$

Berechnung der Umdrehungs- und Winkelgeschwindigkeit

Im ersten Schritt wird die Umdrehungsgeschwindigkeit des Motors benötigt, dazu wird zuerst der Umfang des Reifens berechnet. Dieser ergibt sich wie folgt.

$$U_{Rad} = \pi \cdot D = \pi \cdot 0.1 \text{ m} = 0.314 \text{ m}. \quad (5.8)$$

$$U_{Rad} = 0.314 \text{ m}. \quad (5.9)$$

Damit kann die Umdrehungsgeschwindigkeit und die Winkelgeschwindigkeit bestimmt werden.

Für die Umdrehungsgeschwindigkeit n in $[min^{-1}]$, ergibt sich

$$n = \frac{v}{s} \cdot 60 \quad (5.10)$$

mit $v = v_{EGD}$ der Geschwindigkeit des EGD und $s = U_{Rad}$ dem Umfang des Reifens

$$= \frac{v_{EGD}}{U_{Rad}} \cdot 60 = \frac{1.39 \frac{m}{s}}{0.314 m} \cdot 60 = 265.6 \text{ min}^{-1} \approx 266 \text{ min}^{-1} \quad (5.11)$$

$$n = 266 \text{ min}^{-1} \quad (5.12)$$

und für die Winkelgeschwindigkeit

$$\omega = n \cdot \frac{2\pi}{60} = 265.6 \text{ min}^{-1} \cdot \frac{2\pi}{60} = 27.8 \frac{rad}{s}. \quad (5.13)$$

$$\omega = 27.8 \frac{rad}{s}. \quad (5.14)$$

Berechnung der Gesamtkraft

Im nächsten Schritt wird die Gesamtkraft F_{total} berechnet. Diese gibt Auskunft über die gesamte Kraft, die der Motor überwinden muss, damit eine Bewegung zustande kommen kann.

$$F_{total} = F_{Roll} + F_{Steigung} + F_{Zug} + F_{Luft} \quad (5.15)$$

Rollwiderstand und Steigungswiderstand können zusammengeführt werden. Die Grafik 5.7 veranschaulicht dabei eine schiefe Ebene mit der Hangabtriebskraft F_H , der Gewichtskraft F_G und der Normalkraft F_N , bei einer realen Steigung von α . In diesem Fall entspricht $\alpha = \alpha_{steigung}$. Insgesamt soll dies die Straßensteigung ohne Haftreibung repräsentieren. In der Grafik 5.8 soll zusätzlich noch davon ausgegangen werden, dass bei der schiefen Ebene eine Haftreibung F_R (rot) auftritt, die der Hangabtriebskraft entgegenwirkt.

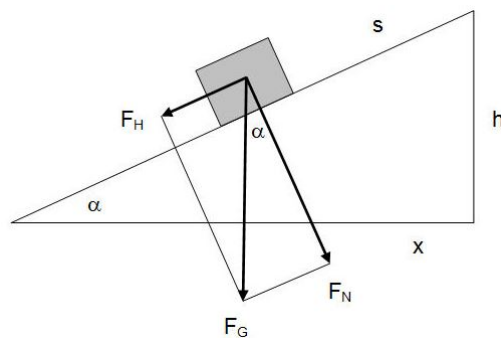


Abbildung 5.7: Schiefe Ebene ohne Haftreibung[27]

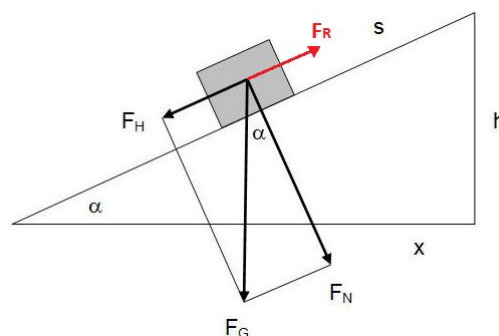


Abbildung 5.8: Schiefe Ebene mit Haftreibung F_R (rot)[27]

Hangabtriebskraft

Damit der Blindenhund diese Steigung inklusive der Haftreibung überwinden kann, muss er einen ausreichend starken Antrieb haben. Die dafür benötigte Kraft beschreibt die resultierende Hangabtriebskraft $F_{H,res}^{\vec{}}$. Da der Blindenhund einen Reifen besitzt, kann man anstatt dem Begriff der Haftreibung auch den Begriff der Rollreibung verwenden. Wichtig ist, dass für den Reibungskoeffizienten der passende Wert verwendet wird.

Die benötigte Kraft $F_{H,res}^{\vec{}}$, berechnet sich wie folgt:

$$F_{H,res}^{\vec{}} = \vec{F}_H + \vec{F}_R. \quad (5.16)$$

Mit \vec{F}_H :

$$\vec{F}_H = m_{EGD} \cdot g \cdot \sin(\alpha_{steigung}). \quad (5.17)$$

Mit \vec{F}_R :

$$\vec{F}_R = \mu_r \cdot \vec{F}_N \quad (5.18)$$

und

$$\vec{F}_N = m_{EGD} \cdot g \cdot \cos(\alpha_{steigung}). \quad (5.19)$$

Dies ergibt die resultierende Gleichung

$$F_{H,res}^{\vec{}} = m_{EGD} \cdot g \cdot \sin(\alpha_{steigung}) + \mu_r \cdot m_{EGD} \cdot g \cdot \cos(\alpha_{steigung}) \quad (5.20)$$

zusammengefasst zu

$$F_{H,res}^{\vec{}} = m_{EGD} \cdot g \cdot (\sin(\alpha_{steigung}) + \mu_r \cdot \cos(\alpha_{steigung})). \quad (5.21)$$

Das Einsetzen von Werten aus der Tabelle 5.1 ergibt dann folgendes:

$$F_{H,res}^{\vec{}} = F_{Roll} + F_{Steigung} = m_{EGD} \cdot g \cdot (\sin(\alpha_{steigung}) + \mu_r \cdot \cos(\alpha_{steigung})) \quad (5.22)$$

$$= 6 \text{ kg} \cdot 9,81 \frac{\text{m}}{\text{s}^2} \cdot (\sin(9^\circ) + 0.15 \cdot \cos(9^\circ)) = 17.92 \text{ N} \approx 18 \text{ N} \quad (5.23)$$

$$F_{H,res}^{\vec{}} = 18 \text{ N} \quad (5.24)$$

Berechnung der Zugkraft

Im nächsten Schritt soll die Zugkraft berechnet werden. Die Zugkraft beschreibt die Kraft, welche entgegengesetzt der Bewegungsrichtung wirkt und somit eine Art Zuglast bzw. Widerstand für den Antrieb darstellt. Die Zugkraft kann unter anderem dadurch zustande kommen, indem der Benutzer zu langsam geht oder sich an dem Blindenstab festhält. Auch variiert die Zugkraft von Benutzer zu Benutzer. Um einen Überblick über diese Kraft zu bekommen, wurde ein Kraftmesser für die Messung verwendet.

Siehe Grafik 5.9.



Abbildung 5.9: Kraftmesser für die Zugkraftmessung

Der Kraftmesser wurde hierbei von zwei Person gehalten, wobei eine Person den Sehbehinderten repräsentiert hat und die andere den Blindenführhund. Die sehbehinderte Person hat dabei an einer Seite des Kraftmessers festgehalten und wurde von der anderen Person geführt. Es wurden verschiedene Streckenabschnitte mit Kurven und Steigung durchlaufen. Die Messergebnisse wurden während des Versuchs notiert.

Für ein passendes Messergebnis der Zugkraft, wurde der Mittelwert aus den Maximalwerten genommen.

Dieser Mittelwert betrug 15 N .

Damit ergibt sich für die Zugkraft F_{Zug} folgendes:

$$F_{Zug} = 15\text{ N}. \quad (5.25)$$

Strömungswiderstandskraft

Für den letzten Schritt zur Berechnung der Gesamtkraft wurde die Strömungswiderstandskraft berechnet, welches die Kraft beschreibt, die der Antrieb leisten muss, um dem Luftwiderstand entgegenzuwirken. Hierbei ist jedoch zu sagen, dass der Luftwiderstand aufgrund der kleinen Angriffsfläche des Blindenhunds und der geringen Fahrgeschwindigkeiten zu vernachlässigen ist.

Die Folgende Berechnung soll veranschaulichen, dass die Strömungswiderstandskraft hierbei zu vernachlässigen ist.

$$F_{Luft} = 0.5 \cdot c_w \cdot A \cdot \rho \cdot (v_{EGD})^2 \quad (5.26)$$

Mit $c_w = 0.3$, $\rho = 1.22 \frac{\text{kg}}{\text{m}^3}$, sowie $A \approx 0.3\text{ m}^2$ und $v_{EGD} = 1.39 \frac{\text{m}}{\text{s}}$ ergibt sich für

$$F_{Luft} = 0.5 \cdot 0.3 \cdot 0.3\text{ m}^2 \cdot 1.22 \frac{\text{kg}}{\text{m}^3} \cdot 1.39 \frac{\text{m}}{\text{s}} = 0.1\text{ N} \approx 0\text{ N} \quad (5.27)$$

$$F_{Luft} = 0\text{ N} \quad (5.28)$$

Als gute Annäherung der umströmten Querschnittsfläche A des EGD, wurde mit einer Stablänge von 1 m und einer Breite von 0.3 m gerechnet. Auch wurde sich auf eine typische Luftwiderstandszahl c_w aus der Literatur bezogen.

Ergebnis der Gesamtkraft

Für die Gesamtkraft F_{total} ergibt sich somit

$$F_{total} = F_{Roll} + F_{Steigung} + F_{Zug} + F_{Luft}, \quad (5.29)$$

wobei $F_{Roll} + F_{Steigung} = 18 \text{ N}$, $F_{Zug} = 15 \text{ N}$ und $F_{Luft} = 0 \text{ N}$, so dass F_{total} sich als

$$F_{total} = 18 \text{ N} + 15 \text{ N} = 33 \text{ N}, \quad (5.30)$$

zu

$$F_{total} = 33 \text{ N}. \quad (5.31)$$

ergibt.

Berechnung der benötigten mechanischen Leistung

Mit der Gesamtkraft und der maximalen Geschwindigkeit des EGD, kann dann die mechanisch benötigte Leistung P_{mech} des Motors berechnet werden. Diese ist wie folgt zu bestimmen:

$$P_{mech} = F_{total} \cdot v_{EGD} = 33 \text{ N} \cdot 1.39 \frac{\text{m}}{\text{s}} = 45.87 \text{ W} \approx 46 \text{ W} \quad (5.32)$$

$$P_{mech} = 46 \text{ W}. \quad (5.33)$$

Berechnung des Drehmoments

Neben der Umdrehungszahl und der benötigten mechanischen Ausgangsleistung fehlt noch das Gesamtdrehmoment M_{ges} , welches der Motor aufbringen muss.[8] Das Gesamtdrehmoment M_{ges} setzt sich als Summe aus dem Lastdrehmoment M_L , dem Beschleunigungsdrehmoment $M_{\dot{\omega}}$ und dem Trägheitsdrehmoment M_{Rad} zusammen. Dieses kann wie folgt ermittelt werden:

$$M_{ges} = M_L + M_{\dot{\omega}} + M_{Rad}. \quad (5.34)$$

Zuerst soll das Lastdrehmoment M_L ermittelt werden,

$$M_L = \frac{P_{mech}}{\omega} = \frac{46 \text{ W}}{27.8 \frac{\text{rad}}{\text{s}}} = 1.65 \text{ Nm}. \quad (5.35)$$

$$M_L = 1.65 \text{ Nm}. \quad (5.36)$$

Zuzüglich des Lastdrehmoments M_L kommt noch das Beschleunigungsdrehmoment $M_{\dot{\omega}}$ und das Trägheitsdrehmoment der Räder M_{Rad} .

Die allgemeine Bestimmung des Beschleunigungsdrehmoments geschieht über die folgende Formel:

$$M_{\dot{\omega}} = m_{EGD} \cdot a \cdot r. \quad (5.37)$$

Sofern man davon ausgeht, dass der Mensch mit $a = 1 \frac{\text{m}}{\text{s}^2}$ beschleunigt, erhält man als Ergebnis für $M_{\dot{\omega}}$

$$M_{\dot{\omega}} = m_{EGD} \cdot a \cdot r = 6 \text{ kg} \cdot 1 \frac{\text{m}}{\text{s}^2} \cdot 0.05 \text{ m} = 0.3 \text{ Nm}. \quad (5.38)$$

Das zusätzliche Trägheitsdrehmoment der Räder M_{Rad} kann hierbei aufgrund der geringen Masse der Reifen und dessen geringem Radius vernachlässigt werden.

Damit ergibt sich das gesamte Drehmoment zu

$$M_{gesamt} = M_L + M_{\dot{\omega}} \quad (5.39)$$

$$M_{gesamt} = 1.65 \text{ Nm} + 0.165 \text{ Nm} = 1.95 \text{ Nm} \quad (5.40)$$

$$M_{gesamt} = 1.95 \text{ Nm}. \quad (5.41)$$

Die folgende Tabelle 5.3 soll eine Übersicht zu den Ergebnissen der Berechnungen geben.

Formelzeichen	Wert	Einheit
n	266	min^{-1}
ω	27.8	$\frac{rad}{s}$
P_{mech}	46	W
M_{ges}	1.95	Nm
$P_{mech, Motor1/2}$	23	W
$M_{ges, Motor1/2}$	0.975	Nm

Tabelle 5.3: Zusammenfassung der Rechenergebnisse

Die gesamte mechanische Leistung P_{mech} sowie das Gesamtdrehmoment M_{ges} werden durch Verwendung des Antriebkonzepts 1, mit zwei Antriebsmotoren, gleichmäßig aufgeteilt. Jeder Motor trägt somit zum Erhalt der gesamten Leistung und des gesamten Drehmoments zur Hälfte bei. Es ist davon auszugehen, dass die Werte für die Leistung und das Drehmoment überdimensioniert sind, da die Berechnungen unter *Worst Case Scenario* Bedingungen getätigt wurden.

5.2.2 Liste der zur Auswahl stehenden Motoren

Ein zusätzlich wichtiger Kernaspekt dieser Arbeit ist die Auswahl eines passenden Motors. Die erforderlichen Leistungsberechnungen wurden im vorherigen Schritt getätigt.

Die Schwierigkeit besteht nun darin, einen passenden Motor zu finden, welcher nicht nur die theoretischen Berechnungen erfüllt, sondern auch noch die folgenden Punkte mit einbezieht:

- Lieferbarkeit - Verfügbarkeit
- Leistung - Drehmoment - RPM
- Effizienz
- Lautstärke
- Dokumentation
- Verarbeitung
- Preis
- Größe - Gewicht
- Einsatzbereich

Diese Kriterien wurden nach Bedeutsamkeit prozentual gewichtet, in einer Tabelle übersichtlich zusammengeführt und mit einem 0 - 10 Punktesystem bewertet. Siehe dazu die Tabellen im Anhang (Tabelle BLDC Motorauswahl A.5 und Tabelle Schrittmotorauswahl A.6).

Die Auswahl des Motors geschah über die Endzusammenfassung der Punktzahl des jeweiligen Motors. Der Motor mit der höchsten Punktzahl wurde demnach für die Verwendung in Betracht gezogen.

Die folgende Abbildung 5.10 soll einen Ausschnitt der Tabelle zur Schrittmotorauswahl verdeutlichen.

Motor / ESC Einkaufsoptionen				
Info	Lieferung muss innerhalb von 1,5 Wochen geschehen	Muss die Anforderungen der Leistungsberechnung erfüllen		
Bewertung-Gewichtung	20,00%	17,50%	17,50%	12,50%
Kriterien / Anforderungen	Verfügbarkeit	Leistung/Drehmoment / RPM	Effizienz	Lautstärke
Produktname / Schrittmotor mit Regler	0-10	0-10	0-10	0-10
Spalte1	Spalte2	Spalte3	Spalte4	Spalte5
Trinamic PD57-2-1076	9	8	8	9
PD4-E591L42-E-65-7	7	9	8	6
EMMS-ST-57-M-S-G2	8	8	7	6
NEMA23-01	9	6	7	6

Abbildung 5.10: Ausschnitt Schrittmotor - Auswahl

Für den Verlauf der Arbeit wurde sich für einen Schrittmotor entschieden, da dieser im unteren Drehzahlbereich ein hohes Drehmoment aufweist und somit kein weiteres Getriebe verbaut werden muss. Auch war die Verfügbarkeit und die Dokumentation bei den Schrittmotoren, der unteren Preisklasse, im Vergleich zu den BLDC Motoren deutlich besser. Im Folgenden wurde sich für den Trinamic PD57-2-1076 entschieden.[5] Von denen für das aktuelle Antriebskonzept zwei Stück verbaut wurden (siehe Abbildung 5.11).

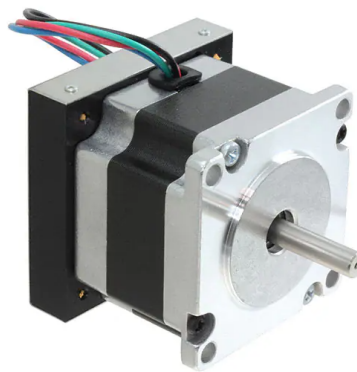


Abbildung 5.11: Schrittmotor[1] mit Treiber: Trinamic PD57-2-1076[15]

Umsetzung des Antriebkonzepts

Mithilfe der Informationen zum Antriebskonzept und der Auswahl des passenden Motors konnte dann ein mechanischer Aufbau gefertigt werden. Die folgenden Abbildungen 5.12 und 5.13 zeigen die Umsetzung des ersten Konzepts mit den zwei Antriebsmotoren. Das Gehäuse und die Räder, wurden extern gefertigt, so dass Räder und Motoren nur eingesteckt und verschraubt werden mussten. Für zukünftige Projekte, ist darauf zu achten, dass die Welle des Motors gut über die Radnabe mit dem Reifen befestigt wird. Weiche Materialien können dazu führen, dass die Radnabe durchdreht.

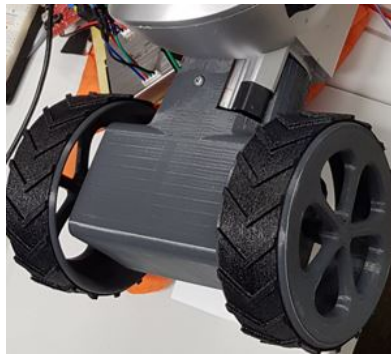


Abbildung 5.12: Aufbau des ersten Antriebkonzepts, seitliche Perspektive



Abbildung 5.13: Aufbau des ersten Antriebkonzepts, Draufsicht

Auf den Abbildungen ist das Gehäuse mit den zwei Rädern aus verschiedenen Perspektiven zu erkennen. Die Motoren befinden sich im Gehäuse und sind direkt über die Welle an den Reifen angeschlossen.

5.3 Konzeption und Softwareentwurf der neuen Fortbewegungsstrategie

Im Folgenden soll ein neues Konzept für eine Fortbewegungsstrategie des EGD entworfen werden, welches die alte Antriebseinheit sowie dessen Software ersetzt bzw. ergänzt. Aus diesem Konzept wird dann eine passende Softwarelösung entwickelt und in das bestehende Projekt implementiert. Damit die Fortbewegungsstrategie erfolgreich umgesetzt werden kann, muss zuerst eine Antriebssteuerung angefertigt werden. Diese wird im Folgenden genauer erläutert.

5.3.1 Antriebssteuerung

Die Antriebssteuerung spielt für die Umsetzung einer erfolgreichen Bahnführung eine bedeutsame Rolle. Sie ist dafür zuständig die benötigte Geschwindigkeit an der Welle des Motors einzustellen.

Die nachfolgende Grafik soll den aktuellen Motortreiber mit seinem Interface für die Beschaltung veranschaulichen.

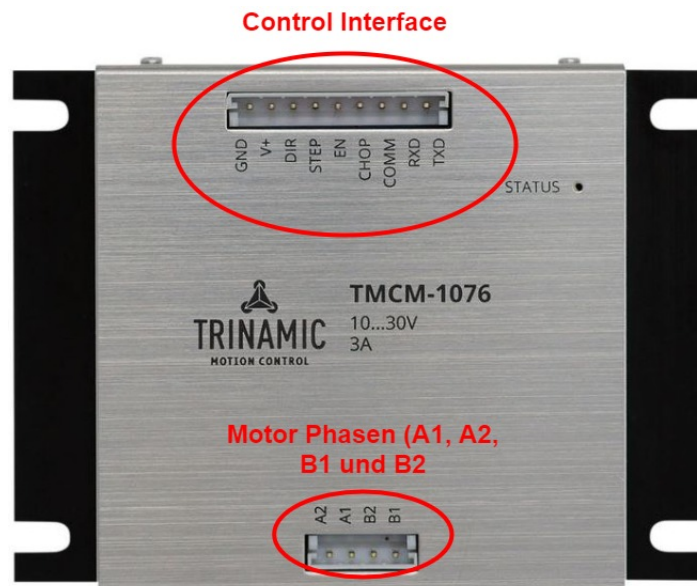


Abbildung 5.14: Trinamic TMC1076 - Motortreiber mit Interface[6]

Dazu kann auch im Folgenden (Abbildung 5.15) die dazugehörige Schaltung des Treibers betrachtet werden.

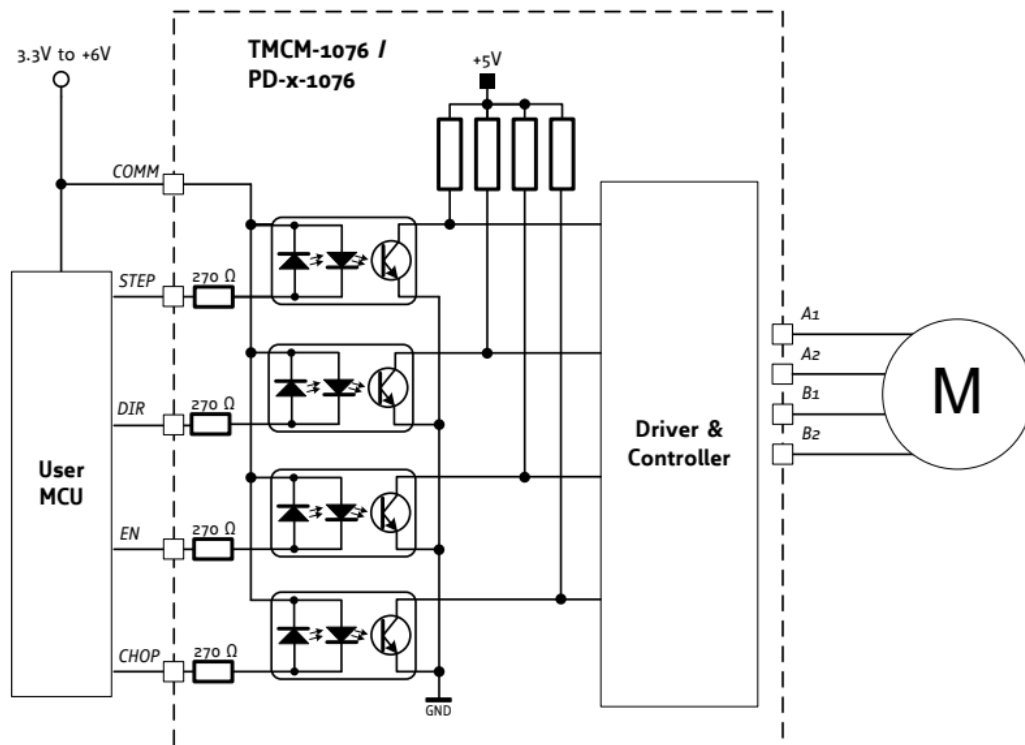


Abbildung 5.15: Schaltplan für den Motor-Driver (TMCM-1076)[5]

Das Interface bietet neben der Einstellung der Schrittgeschwindigkeit, auch die Möglichkeit die Drehrichtung mit *Step Direction* und die Betriebsmodi über *Chop* einzustellen.

Die Zusammenfassung des Interfaces, erfolgt über die folgende Tabelle 5.4.

Pin	Pin Name	Bedeutung
1	GND	
2	V+	10 V - 30 V (DC)
3	DIR	Drehrichtung
4	STEP	Pulsfrequenz für Schritte
5	EN	Aktiviert die H-Brücken
6	CHOP	Modi Auswahl (StealthChop oder SpreadCycle)
7	COMM	
8	RXD	UART Receive
9	TXD	UART Transmit

Tabelle 5.4: Trinamic TMC2130 - Interface Pinbelegung[5]

Der Motortreiber bietet neben den einfachen Einstellung über das Interface die weitere Option, mithilfe von PIN 8 *RXD* und PIN 9 *TXD* eine UART Verbindung aufzubauen. Diese ermöglicht eine Vielzahl von Einstellungen. Folgende sind von größerer Bedeutung:

- Microstep Resolution
- Betriebsmodi:
 - StealthChop (Leiser Bestriebsmodus)
 - SpreadCycle (Mehr Leistung)
 - CoolStep (Effizienter Betrieb)
- Maximum Current
- Halbstrombegrenzung
- Standbycurrent

Besonders wichtig für ein ruhiges und ruckelfreies Fahrverhalten mit möglichst wenig Resonanzen ist die Vermeidung von niedrigen Drehzahlen. Bei kleinen Drehzahlen bzw. Frequenzen im Bereich um 200 Hz sind diese oft vorzufinden. Um ein ruhiges Fahrverhalten zu ermöglichen und Resonanzen zu minimieren ist, es notwendig eine hohe Schrittauflösung (Microstep Resolution) zu haben. Zusätzlich kann auch ein Dämpfer

verbaut werden, welcher die Schwingungen weiterhin abfängt. Auch ist bei jeder Inbetriebnahme eines Schrittmotors darauf zu achten, dass nicht nur die Stromversorgung ausreichend dimensioniert ist, sondern auch der Treiber richtig betrieben wird. Dazu wurde sich hier auf das Datenblatt des Trinamic TMCM-1076 bezogen. Eine zu kleine bzw. zu große Versorgungsspannung kann ebenfalls zu Resonanzen oder Schäden am Motor führen. Für die Ansteuerung der Anschlüsse des Motortreibers (Abbildung 5.14) wurden Spannungspegel von Low 0 V und High 5 V verwendet. Weiterhin ist zu beachten, dass der Ausgangsstrom für die Ansteuerung des jeweiligen Pins (siehe Tabelle 5.4) zwischen 5 mA und 20 mA liegt. Die maximale Eingangsfrequenz für den PIN 3 (*STEP*) ist $f_{step,max}$ und lag bei einer Pulsbreite von $TON = 10 \mu s$ bei 45 kHz. Da es in der Praxis jedoch Resonanzen im höheren Frequenzbereich gab, wurde bereits bei 25 kHz das Limit gesetzt.

Aufbau einer UART Verbindung

Wie bereits erwähnt, verfügt der Motor-Driver über eine Vielzahl an Einstellungen, die über die serielle UART Schnittstelle konfiguriert werden können.[4] Aufgrund des hohen Zeitaufwands für die Entwicklung, Implementierung und Prüfung der einzelnen Funktionen, sowie der allgemeinen Dokumentation des Projekts, konnte für diese Anforderung kein Softwarecode oder eine ähnliche Lösung entwickelt werden. Informationen zur UART Verbindung und dem Aufbau des Datenpakets, sind dem Datenblatt des Motor-Treibers zu entnehmen.

Einstellen der Geschwindigkeit des EGD

Aus dem Kapitel 2 *Theoretische Grundlage* ist bekannt, dass die Controller oder Driver für Schrittmotoren mit einer Pulsfrequenz arbeiten. Im Folgenden soll ein Überblick zu den verwendeten Parametern und Formeln für die Motoransteuerung gegeben werden. Zuerst zu den Bedeutungen der Formelzeichen.

- f_{step} : Schrittfrequenz in $\frac{1}{s}$
- $n(t)$: Anzahl der Umdrehungen der Welle in Abhängigkeit der Zeit t
- $v_{EGD}(t)$: Geschwindigkeit des EGD
- n_{teeth} : Anzahl der Rotorzähne (Konstante des Schrittmotors)
- $SA = \frac{360^\circ}{n_{teeth}}$ in $[\frac{deg}{Step}]$ hier: $SA = 1.8 \frac{deg}{step}$ (Konstante des Schrittmotors)
- MS : Microsteps¹ hier: $MS = 16$ (Abhängig vom Betriebsmodus)
- $STEP_{resolution} = \frac{1}{MS \cdot n_{teeth}}$ Schrittauflösung des Steppermotors

Mit der allgemeinen Gleichung für die Berechnung der Geschwindigkeit,

$$v(t) = \frac{s}{t} = \frac{Strecke}{Zeit} \quad (5.42)$$

wobei die Strecke $s = \pi \cdot D$ dem Radumfang entspricht und die Frequenz f_{step} der Anzahl der Schritte pro Sekunde. Da jeder Puls bzw. Schritt nur ein Bruchteil einer vollen Umdrehung des Radumfangs ist, muss die Strecke $s = \pi \cdot D$ mit der Schrittauflösung $STEP_{resolution}$ und mit der Frequenz f_{step} multipliziert werden, um die Geschwindigkeit des EGDs zu ermitteln.

¹Zusätzlich zu den 200 Schritten (durch die Verzahnung), die der Motor für eine volle Umdrehung braucht, kann jeder Schritt in weiter 16 Microsteps unterteilt werden. Dies ermöglicht eine höhere Auflösung mit ruhigerem Fahrverhalten. Reduziert jedoch die maximale Umdrehungsgeschwindigkeit.

Für die Gleichungen werden die *Microsteps* (MS), der *Durchmesser* (D), die Anzahl der Rotorzähne (n_{teeth}) sowie der *Schrittwinkel* (SA) als konstant angenommen. Die Geschwindigkeit des EGDs ergibt sich somit zu:

$$v_{EGD}(f_{step}) = \pi \cdot D \cdot f_{step} \cdot STEP_{resolution}. \quad (5.43)$$

$$= \pi \cdot D \cdot f_{step} \cdot \frac{1}{MS \cdot n_{teeth}} = \frac{\pi D \cdot f_{step}}{MS \cdot n_{teeth}}. \quad (5.44)$$

$$= \frac{\pi D \cdot f_{step}}{MS \cdot \frac{360^\circ}{SA}} = \frac{f_{step} \cdot \pi D \cdot SA}{MS \cdot 360^\circ}. \quad (5.45)$$

$$\Rightarrow v_{EGD}(f_{step}) = \frac{f_{step} \cdot \pi D \cdot SA}{MS \cdot 360^\circ}. \quad (5.46)$$

Die Pulsfrequenz oder Schrittfrequenz erhält man durch Umstellen:

$$\Rightarrow f_{step}(v_{EGD}) = \frac{v_{EGD} \cdot MS \cdot 360^\circ}{\pi D \cdot SA}. \quad (5.47)$$

Für Messungen der Umdrehungen n pro Zeit t kann folgende Formel angewandt werden:

$$\Rightarrow n(t, f_{step}) = \frac{t \cdot f_{step}}{MS \cdot \frac{360^\circ}{SA}}. \quad (5.48)$$

Für die weiteren Versuche wird meist die Formel zur Einstellung der Geschwindigkeit des EGDs benötigt. Mit einem Durchmesser von $D = 0.1 \text{ m}$, $SA = 1.8 \frac{\text{deg}}{\text{step}}$ und $MS = 16$ ergibt sich dann für die Geschwindigkeit:

$$v_{EGD}(f_{step}) = \frac{f_{step} \cdot \pi D \cdot SA}{MS \cdot 360^\circ} = \frac{f_{step} \cdot \pi \cdot 0.1 \text{ m} \cdot 1.8 \frac{\text{deg}}{\text{step}}}{16 \cdot 360^\circ}. \quad (5.49)$$

$$\Rightarrow v_{EGD}(f_{step}) = f_{step} \cdot 9.817 \cdot 10^{-5}. \quad (5.50)$$

Sofern die Mikroschrittauflösung MS über die UART Verbindung verändert wird, ist darauf zu achten, dass in der Formel der Wert für MS entsprechend angepasst wird.

Funktion zur Änderung der Geschwindigkeit

Mithilfe der Formeln zur Geschwindigkeitseinstellung kann nun ein Algorithmus entwickelt werden, welcher die Geschwindigkeit des Blindenhunds verändert. Um die Geschwindigkeit des Blindenhunds einzustellen, muss der Motor-Driver auf seinem *Step* Eingang richtig angesteuert werden. Dazu siehe auch die Angaben des Herstellers bezüglich der nötigen Spannungs- und Strompegel sowie der zeitlichen Anforderungen. Um bei einem Schrittmotor eine Geschwindigkeit einzustellen, muss eine gewisse Anzahl an Pulsen in einer bestimmten Zeit an den Controller gesendet werden. Dies wird in diesem Fall über einen Ausgangspin am Mikrocontroller umgesetzt. Es wurde demnach ein Timer mit einem GPIO Port verbunden und so konfiguriert, dass Load und Matchvalue für die Pulsenerierung das erwünschte Zeitverhalten am MC Ausgang erzeugt haben. Die Pulsbreite betrug hierbei $TON = 10 \mu s$. Die Frequenz konnte über den Funktionsparameter der Geschwindigkeit (*double runningSpeed*) und der Formel 5.46 bestimmt werden. Die Funktionen zur Geschwindigkeitseinstellung heißen wie folgt:

- `void setSpeed_Motor_1(double runningSpeed)`
- `void setSpeed_Motor_2(double runningSpeed)`
- `void setRunningSpeed(double runningSpeed)`

Die Geschwindigkeit *runningSpeed* ist der Parameter der drei Funktionen und wird in der Einheit $\frac{km}{h}$ angegeben. Alle Funktionen stellen in Abhängigkeit vom Übergabewert, passend den Vorteiler, sowie Match- und Loadvalues für die Geschwindigkeit des Blindenhunds, am Mikrocontroller ein und starten nach ihrem Aufruf die dazugehörigen Timermodule. Während `void setSpeed_Motor_1()` und `void setSpeed_Motor_2()` die Geschwindigkeit eines einzigen Motors einstellen, setzt die Funktion `void setRunningSpeed()` beide Motoren auf die gleiche Geschwindigkeit.

5.3.2 Aktuelle Antriebssoftware des EGD

Die Datei *steering_driver.c* bildet zusammen mit der *steering_driver.h* die aktuelle Antriebssoftware und ist im CCS Projekt *watchdog* zu finden. Sie stellt die Grundlage für die Weiterentwicklung des neuen Antriebssystems dar und implementiert die zu überarbeitenden Funktionen für das neue Antriebssystem. Die neuen Funktionen müssen so aufgebaut und konzipiert werden, dass diese in das bestehende Softwareprojekt des EGD eingebunden werden können. Hierzu wurde zuerst eine Übersicht zur bestehenden Antriebssoftware erstellt. Diese Übersicht ist im Anhang unter Abbildung A.2 zu finden. Das Grundprinzip basiert auf der wiederkehrenden Abfrage eines *Moving Angles*, welcher den zulässigen Fahrwinkel für die Fortbewegung des Blindenhunds darstellt und über ein System zur Hinderniserkennung, mithilfe von Bildverarbeitung und Lidarsensorik ermittelt wird. Dieser Winkel kann zwischen -45 und 45 Grad liegen und wird im Verlauf dieser Arbeit mit α beziffert. Gemäß der Umgebung und des resultierenden Winkels, wird daraufhin ein Fahrmanöver über die Antriebseinheit eingeleitet.

Es gibt vier verschiedene Fahrmanöver:

- `void start_moving(steering_driver* steer_driver, int8_t final_speed)`
- `void set_speed(steering_driver* steer_driver, int8_t final_speed)`
- `void set_angle(steering_driver* steer_driver, int8_t angle)`
- `void stop_vehicle(steering_driver* steer_driver)`

Die Funktionen werden so angepasst, dass der Funktionsname erhalten blieb und nur die Funktionsdefinition verändert werden muss. Die Funktion `set_speed(steering_driver* steer_driver, int8_t final_speed)` wurde für das neue Antriebssystem durch eine andere Funktion ersetzt, so dass diese nicht mehr benötigt wird. Auch ist zu erkennen, dass die Funktionen für die Fahrmanöver maximal zwei Parameter übergeben bekommen. Zum einen der Pointer, `steering_driver* steer_driver`, welcher für die Antriebssteuerung unbedeutend ist und zum anderen der Wert für die Geschwindigkeit (`int8_t final_speed`) oder der Winkel (`int8_t angle`), für die Kurvenfahrt.

Des Weiteren verfügt das Projekt über mehrere Header und `.c` Files, in denen die Funktionen für den Antrieb, die Sensorik und die allgemeine Datenverarbeitung stehen. Die Herausforderung ist es, die richtige Schnittstelle zu finden, in welche die neue Software eingebunden werden muss. An dieser Stelle ist anzumerken, dass die aktuelle Software,

bestehend aus Funktionen, Variablenamen, Kommentaren und ähnlichem, in Englisch gefertigt wurde. Da dieses Projekt zukünftig weiter überarbeitet wird, wurde bei der Wahl neuer Funktionen darauf verzichtet die Sprache zu verändern. Funktionen, Kommentare und bildliche Veranschaulichungen sind daher nicht in Deutsch geschrieben worden. Das folgende Bild 5.16 soll dabei den Arbeitsbereich (Workspace) für die neue Software darstellen.

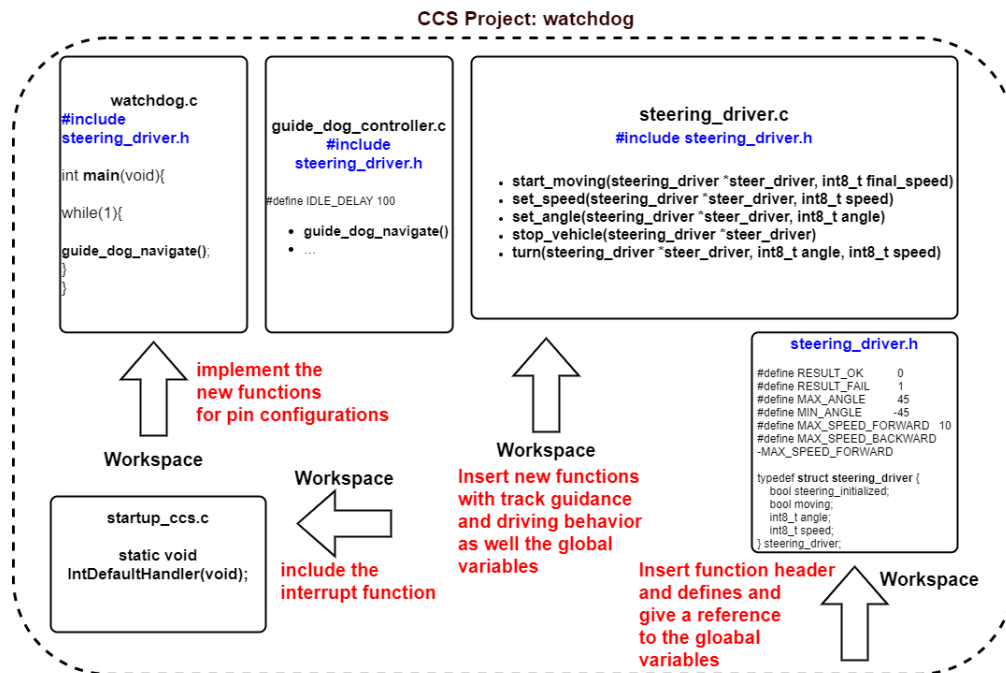


Abbildung 5.16: Übersicht zum Arbeitsbereich, sowie den Header- und .c Dateien

Im Anhang ist diese Übersicht in größer Darstellung zu finden (siehe Abbildung A.1).

Ablauf der aktuellen EGD Software

Mit der Anbindung des Mikrocontrollers an die Stromversorgung, wird die aktuelle Software des Guide Dogs über die *main* Funktion gestartet. Diese befindet sich in der *watchdog.c*. Nachdem Start und dem Aufruf der main-Funktion wird in eine Endlosschleife übergegangen, in welcher die Funktion *guide_dog_navigate()* wiederholend aufgerufen wird.

Die folgende Abbildung 5.17 soll den Programmablauf mit der Endlosschleife exemplarisch darstellen.

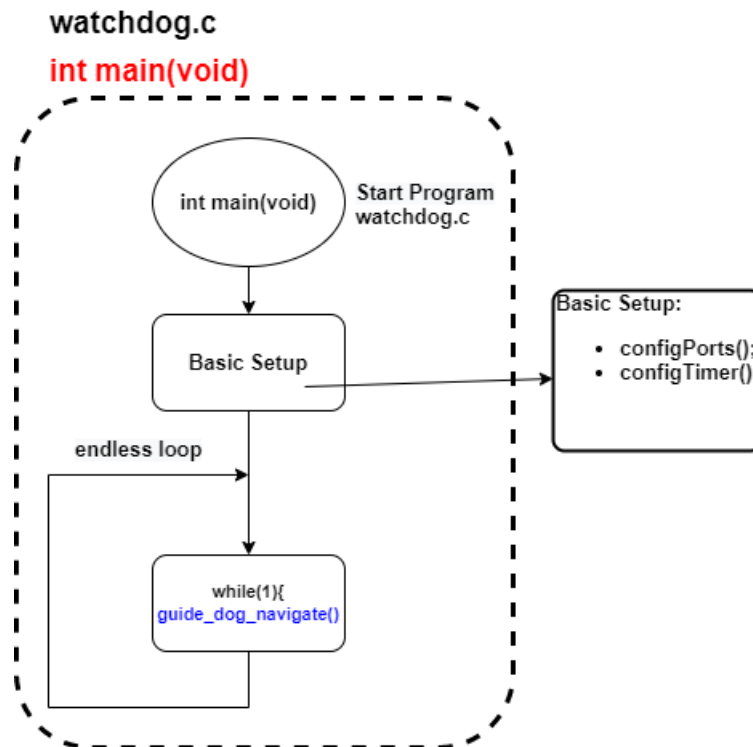


Abbildung 5.17: Flowchart des Programmstarts mit Endless-Loop

Die Funktion *guide_dog_navigate()* ist eine zentrale Funktion des gesamten Softwareprojekts und ruft unter anderem die Funktionen für die Sensorik und die Motorsteuerung auf. Um diesen Vorgang bildlich zu verdeutlichen, wird im Folgenden ein Teilausschnitt des Funktionsablaufs für die Motorsteuerung dargestellt. Der Ausschnitt bezieht sich auf die Ansteuerung der Antriebseinheit und somit auf die Fortbewegung des elektronischen Blindenhunds (siehe Abbildung 5.18). Das vollständige Flowchart ist im Anhang unter der Abbildung A.2 zu finden.

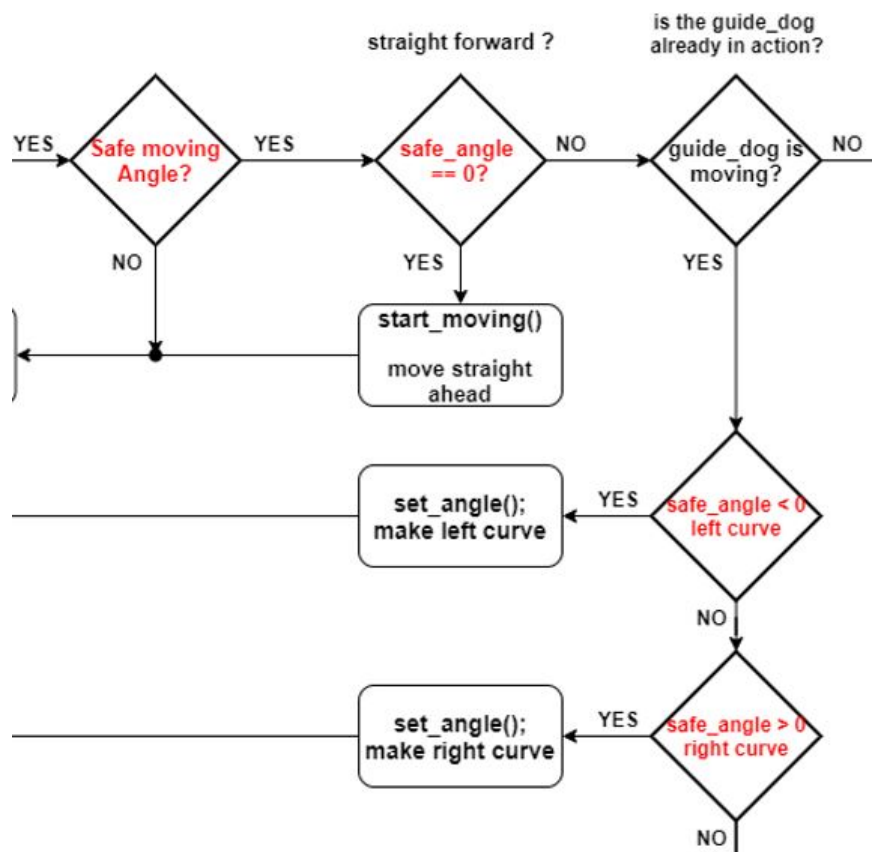


Abbildung 5.18: Ausschnitt zu der Antriebssoftware

Anhand der Abbildung 5.18 ist zu erkennen, dass die Software sukzessive den *safe_angle* auf bestimmte Werte abfragt und daraufhin ein neues Fahrmanöver einleitet. Ist zum Beispiel der Guide Dog in Bewegung und der Winkel < 0 Grad, so wird die Funktion *set_angle()* aufgerufen und der Guide Dog wird mithilfe des Winkels zu einem Linksmanöver aufgefordert. Nachdem das Manöver abgeschlossen wurde, verbleibt der EGD in seinem Zustand, bis ein neuer *safe_angle* mit entsprechendem Funktionsaufruf eintritt.

Im nächsten Schritt soll aufgrund dieser Basis die Fortbewegungsstrategie für die entsprechenden Fahrmanöver entwickelt werden.

Entwurf der Fortbewegungsstrategie

Für den Entwurf der Fortbewegungsstrategie gibt es verschiedene Möglichkeiten zur Umsetzung der Wegführung. Es gibt zum einen die Möglichkeit, dass der Blindenhund die Wegweisung und Navigation über eine fest vorgegebene Geschwindigkeit ausführt und zum anderen die Möglichkeit, dass der Blindenhund die Navigation über einen gewissen Kraftzug, durch die Vorgabe eines Drehmoments am Elektromotor durchführt.

Geschwindigkeitsvorgabe

Bei der ersten Variante, mit der Vorgabe einer Geschwindigkeit, wird der Benutzer durch den Antrieb gezwungen das Tempo des Motors mitzulaufen. Sofern der Benutzer die Geschwindigkeit verringert, würde der Motor versuchen die Geschwindigkeit dennoch aufrecht zu erhalten. Die Schwierigkeit liegt in der Einstellung der richtigen Geschwindigkeit für den Benutzer. Auch ist nicht vorherzusagen, ob dies ein angenehmes Fahrverhalten ermöglicht, da der Benutzer sich an die Geschwindigkeit des Blindenhunds kontinuierlich anpassen muss.

Drehmomentvorgabe

Bei der zweiten Variante, wird dem Motor ein gewissen Drehmoment vorgegeben mit dem der Benutzer gezogen bzw. geführt wird. Sollte der Benutzer die Geschwindigkeit verringern würde der Motor hierbei automatisch mit reduzierter Geschwindigkeit weiterlaufen, jedoch die Zugkraft aufrechterhalten. Die Problematik mit dem verwendeten Schrittmotor ist, dass dieser nur in der Lage ist eine Position bzw. eine Geschwindigkeit umzusetzen. Ein BLDC Motor könnte für diese Verfahren jedoch eingesetzt werden. Die Drehmomentkonstante des BLDC Motors ermöglicht die Einstellung eines festes Drehmoments mit einer einfachen Stromregelung.

Für den folgenden Verlauf dieses Projekts wird mit der ersten Variante und somit einer fest vorgegebenen Geschwindigkeit gearbeitet. Dieses liegt vor allem den Strukturen und Vorgaben der aktuellen Antriebssoftware zugrunde.

Für die Entwicklung und Umsetzung der Fortbewegungsstrategie kann man zwischen den vier folgende Manöver für die Bahnführung des elektronischen Blindenführhunds unterscheiden:

- Anfahren
- Vorwärtsfahren
- Kurvenfahrten
- Bremsen

Für den Entwurf der Fortbewegungsstrategie wurden vier verschiedene Statusmeldungen für den Blindenhund definiert. Dem elektronische Blindenhund wird zu jedem Zeitpunkt ein bestimmter Status zugeordnet. Diese heißen wie folgt:

- *start_moving_status*
- *set_angle_status*
- *stop_vehicle_status*
- *idle_status*

Die Statusmeldungen sind insofern von Bedeutung, da sie Auskunft über das aktuelle Fahrmanöver des Blindenhunds geben und damit für den Entwurf der Software essenziell sind. Ein Status kann sich immer dann ändern, wenn ein bereits laufendes Manöver beendet oder ein neues Manöver aufgerufen wird.

Aufbau der Funktionen

Die nachfolgenden Funktionen für die verschiedenen Fahrmanöver (Anfahren, Abbremsen und Kurvenfahrten) sind so aufgebaut, dass diese immer aus zwei Hauptfunktionen bestehen. Dabei ist die eine Hauptfunktion der Name des Manövers selber zum Beispiel *set_angle()* und leitet hierbei das Manöver und die ersten Initialisierungsschritte ein. Hingegen übernimmt die zweite Hauptfunktion (*timerInterruptHandler*) die Änderung der Geschwindigkeiten am Motortreiber. Außerdem ist die zweite Funktion Interrupt gesteuert und dient als eine Art Zeitgeber für die Änderung von zeitabhängigen Abläufen wie zum Beispiel der Beschleunigung.

5.3.3 Anfahren und Abbremsen

Das Anfahren des EGD geschieht über die Funktion `start_moving(double finalSpeed)`. Der Parameter `finalSpeed` gibt hierbei die Endgeschwindigkeit für den EGD vor. Die Funktion kann zu jeder Zeit erneut aufgerufen und überarbeitet oder abgebrochen werden. Das Anfahren ähnelt dem Abbremsen des EGDs sehr stark und erfordert ein gleichmäßiges Beschleunigen der Räder über eine gewisse Zeit. Für die Umsetzung der Beschleunigung muss eine passende Anzahl von Pulsen in der richtigen Zeit an die beiden Motor-Driver gesendet werden. Es ist darauf zu achten, dass durch die Anordnung der Motoren (180 Grad verdreht), die Drehrichtung eines der beiden Motoren umzukehren ist. Wichtig beim Anfahren ist, dass jeder Motor zur selben Zeit die gleiche Geschwindigkeit aufweist. Um ein möglichst realistisches Manöver zu erhalten, werden zuerst Kenndaten zu allgemeinen Schrittgeschwindigkeiten und Beschleunigungen benötigt. Die Schrittgeschwindigkeit bei Männern betrug laut Literatur im Schnitt $1.41 \frac{m}{s}$ und bei Frauen $1.27 \frac{m}{s}$. Der Mittelwert daraus sind $1.34 \frac{m}{s}$, dieses entspricht $4.8 \frac{km}{h}$.

Für verschiedene Benutzer und Fahrprofile kann zusätzlich noch eine Beziehung zwischen Körpergröße und der *natürlichen Gehgeschwindigkeit* v_n einbezogen werden. Diese Beziehung wurde von Bernd Rodewald und Hans Joachim Schlichting im Jahre 1988 wie folgt beschrieben[34]:

$$v_n = \frac{\sqrt{2 \cdot g}}{\pi} \sqrt{l}. \quad (5.51)$$

Mit g wird die Gravitation beschrieben. Diese soll hier $9,81 \frac{m}{s^2}$ betragen. Mit l wird die Beinlänge gemeint. Des Weiteren wird angenommen, dass ein Zusammenhang zwischen Beinlänge und Körpergröße besteht. Dabei soll die Beinlänge das 0.4 bis 0.5 fache der Körpergröße sein. Bei einer Durchschnittsgröße eines ausgewachsenen Mannes in Deutschland, mit einer Körpergröße von $1,8 m$, ergibt dies folgendes für die natürliche Gehgeschwindigkeit:

$$v_n = \frac{\sqrt{2 \cdot g}}{\pi} \sqrt{l} = \frac{\sqrt{2 \cdot 9.81 \frac{m}{s^2}}}{\pi} \cdot \sqrt{1.8 m \cdot 0.45} = 1.26 \frac{m}{s} = 4.56 \frac{km}{h}. \quad (5.52)$$

$$v_n \approx 4.6 \frac{km}{h}. \quad (5.53)$$

Diese Geschwindigkeit liegt im Rahmen der angenommenen durchschnittlichen Geschwindigkeit und ist somit passend.

Für die Beschleunigung beim Anfahren wurden folgende Werte ermittelt.

Der Blindenhund soll innerhalb von 2 Sekunden ($t_{beschl.}$) aus dem Stand auf seine maximale Schrittgeschwindigkeit (v_{EGD}) beschleunigen.

$$t_{beschl.} = \Delta t = 2 \text{ s} \quad (5.54)$$

und der maximalen Geschwindigkeit

$$\Delta v_{max} = 1.39 \frac{\text{m}}{\text{s}}, \quad (5.55)$$

ergibt folgendes für

$$a_{max} = a = \frac{\Delta v_{EGD}}{\Delta t_{beschl.}} = \frac{1.39 \frac{\text{m}}{\text{s}}}{2 \text{ s}} = 0.695 \frac{\text{m}}{\text{s}^2} \approx 0.7 \frac{\text{m}}{\text{s}^2}. \quad (5.56)$$

Ein typisches Beschleunigungsverhalten bei Fußgängern, während des Gehens, liegt bei $1 \frac{\text{m}}{\text{s}^2}$. Da es sich hierbei um einen sehbehinderten Benutzer handelt, können Geschwindigkeiten und Beschleunigungen grundsätzlich niedriger angesetzt werden. Somit wird die Beschleunigung auf $0,7 \frac{\text{m}}{\text{s}^2}$ vorerst festgelegt.

Zusammenfassung der Ergebnisse:

Parameter	Wert	Einheit
a	0.7	m/s^2
v_{EGD}	4.6	km/h

Tabelle 5.5: Zusammenfassung der Rechenergebnisse für Geschwindigkeit und Beschleunigung

Anfahren Algorithmus

Um einen Überblick zu bekommen, wie das Anfahren in der Funktion `start_moving()` umgesetzt wurde, ist im Folgenden eine Abbildung 5.19 mit einem Konzept zum Anfahren zu finden.

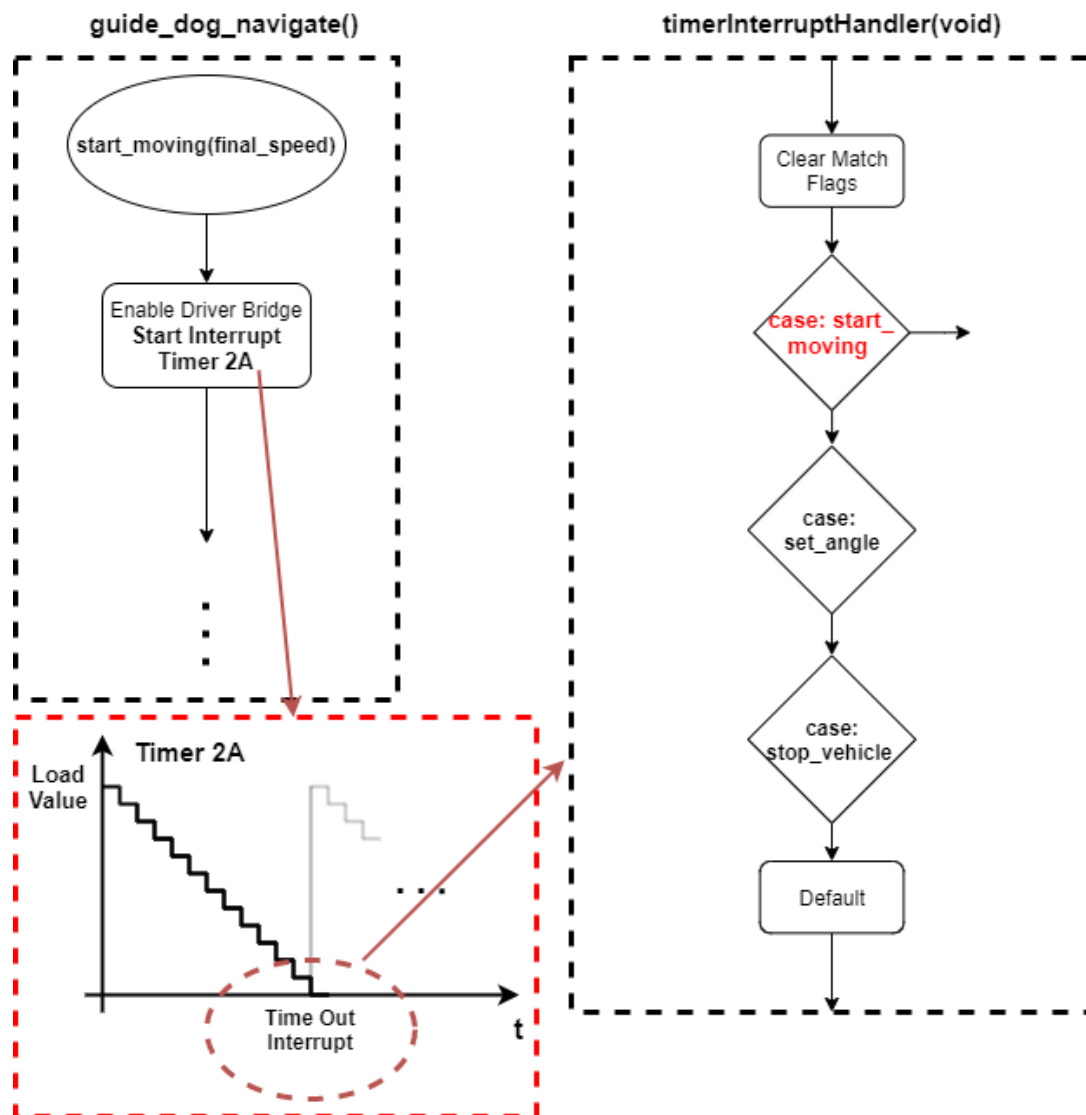


Abbildung 5.19: Konzept - Algorithmus zum Anfahren des EGD

Das Anfahren geschieht über die Funktion `start_moving()`, welche im Verlauf des Programms gestartet wird. Nachdem die Funktion aufgerufen wurde, wird die Driver Stage des Motor-Drivers aktiviert und es wird ein Timer gestartet. Dieser Timer erreicht nach einer vorgegebenen Zeit einen *Time Out* und löst daraufhin einen Interrupt aus. Zusätzlich wird entsprechend dem Manöver der aktuelle Status initialisiert. Nachdem der Interrupt aufgetreten ist, wird die dazugehörige Interrupt Funktion `timerInterruptHandler(void)` ausgeführt. Anschließend wählt die Funktion mit der Information über den

aktuellen Status, in einer *Switch Case* Anweisung das entsprechende Manöver aus und bearbeitet dieses.

Nachfolgend ist eine Abbildung 5.20 zum Ablauf der *Switch Case* Anweisung zu sehen.

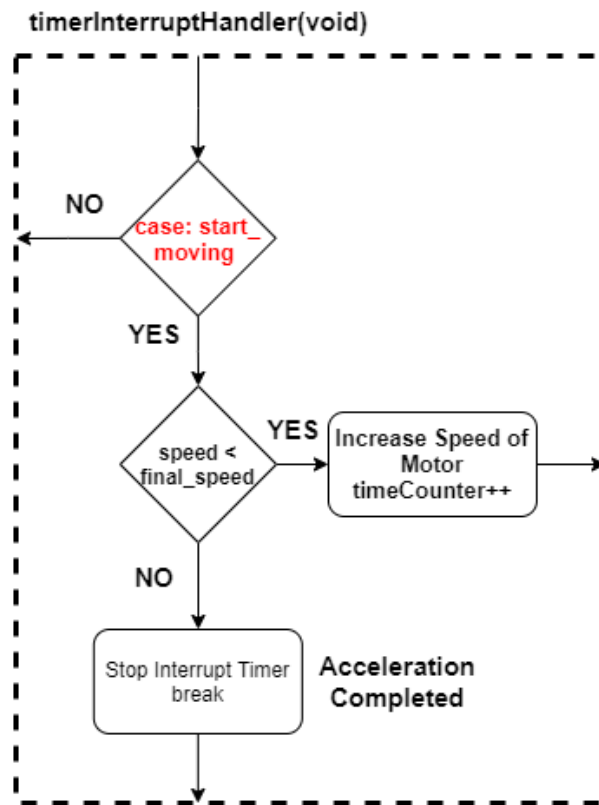


Abbildung 5.20: `timerInterruptHandler(void)`, Algorithmus für das Anfahren des EGD

Nachdem die Funktion `start_moving()` aufgerufen und ein Interrupt durch den Timer generiert wurde, wird in der Interrupt-Funktion durch eine Switch-Case Anweisung das Manöver ausgewählt und bearbeitet. In diesem Fall wird der Case `start_moving` gewählt. Nachdem dies passiert ist, wird mit jedem Interrupt gefragt, ob die aktuelle Geschwindigkeit `speed` kleiner als die finale Geschwindigkeit `final_speed` ist. Wenn dies zutrifft, wird die aktuelle Geschwindigkeit erhöht.

Die aktuelle Geschwindigkeit $v(t)$ wird nach der folgenden Formel erhöht:

$$v(t) = v_{start} + a \cdot \Delta t, \quad (5.57)$$

wobei die Beschleunigung a ist, die Anfangsgeschwindigkeit v_{start} und Δt über die Beziehung zwischen der Anzahl der Interrupts und der `samplingTime` bestimmt werden kann. Es ergibt sich:

$$v(t) = v_{start} + (timeCounter - timeCounterStartValue) \cdot samplingTime \cdot a. \quad (5.58)$$

Die Variable `timeCounter` zählt die Anzahl der Interrupts und wird mit jedem Aufruf der Interrupt-Funktion um eins erhöht. Die Variable `timeCounterStartValue` repräsentiert die Anzahl der Interrupts zum Startzeitpunkt des Funktionsaufrufs `start_moving()`. Die `samplingTime` entspricht der Zeitdifferenz zwischen zwei Interrupts. Wenn nun die Anzahl der Interrupts $\Delta n_{interrupt}$ bekannt ist und mit der Zeitdifferenz multipliziert wird, erhält man die zeitliche Veränderung.

Mit $\Delta n_{interrupt} = (timeCounter - timeCounterStartValue)$ kann dann die Gleichung vereinfacht werden

$$v(t) = v_{start} + \Delta n_{interrupt} \cdot samplingTime \cdot a. \quad (5.59)$$

Beim nächsten Interrupt, ist `timeCounter` um eins größer und die Geschwindigkeit kann daraufhin wieder neu berechnet und erhöht werden.

Bremsen

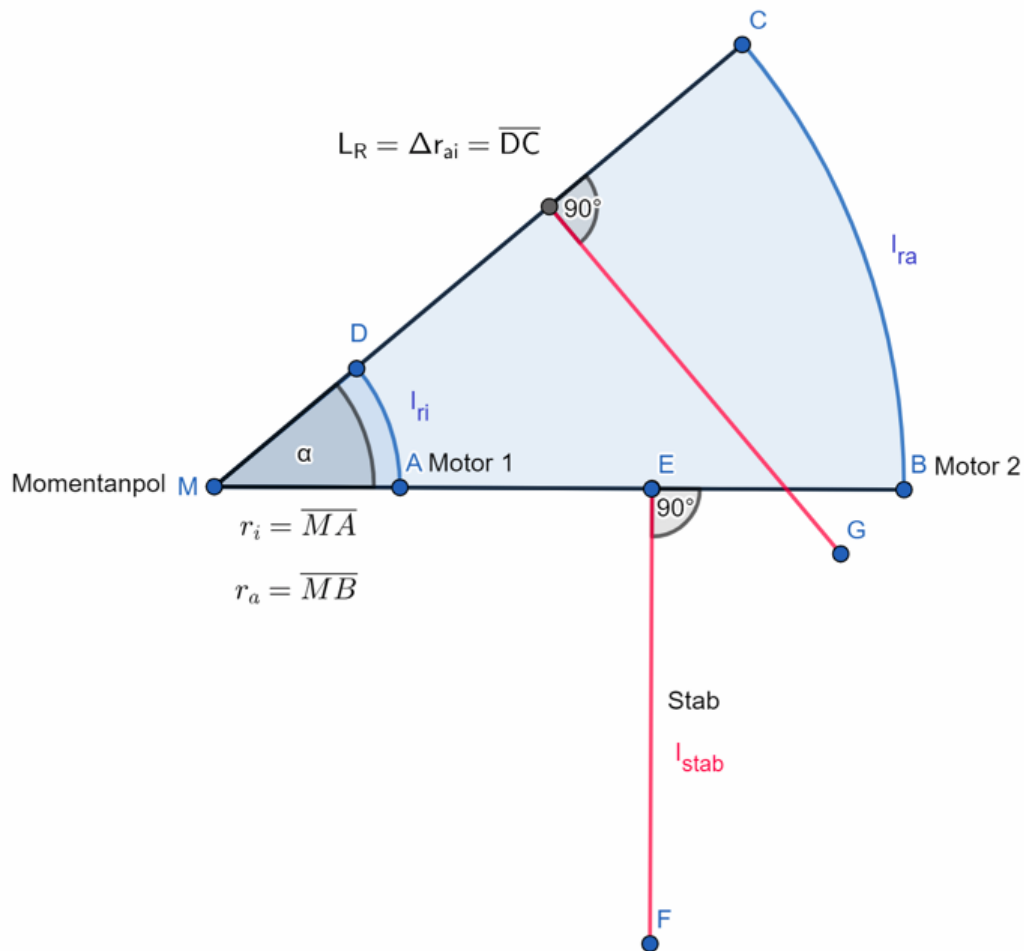
Das Abbremsen ist ähnlich aufgebaut wie die Funktion `start_moving()`, zum Beschleunigen des EGDs und wird durch die Funktion `stop_vehicle()` eingeleitet. Der Unterschied zur Funktion `start_moving()` ist, dass mit jedem Time-Out bzw. Interrupt die Geschwindigkeit des EGDs reduziert wird. Die Geschwindigkeit des Blindenhunds wird solange von der Startgeschwindigkeit reduziert, bis der EGD komplett anhält. Die Funktion kann zu jedem Zeitpunkt aufgerufen werden und unterbricht alle anderen Manöver.

5.3.4 Vorwärtsfahren

Für ein Manöver zum Geradeausfahren benötigt es lediglich eine finale Geschwindigkeit mit der beide Motoren gleichzeitig angetrieben werden. Der EGD fährt immer dann Geradeaus, wenn er in Bewegung ist und von der Software keine Anweisungen der Richtungsänderung, über *set_angle()*, oder eine Anweisung zum Anhalten *stop_vehicle()* bekommt. Die Problematik beim Geradeausfahren ist, dass kleinste Unterschiede zwischen den Geschwindigkeiten beider Motoren zu einer Abweichung der Spur führen würden. Die Implementierung eines Regelsystems, inklusive Kompasssensor, könnten dem entgegenwirken.

5.3.5 Kurvenfahrten

Die Implementierung eines Algorithmus für eine Kurvenfahrt ist im Vergleich zu den anderen Manövern am aufwändigsten. Dies liegt daran, dass eine passende Ansteuerung für das Differential der Motoren, für die Kurvenfahrt gefunden werden muss.[22] In diesem Fall muss auch eine weitere Beziehung zwischen dem Kurvenwinkel α und dem Innenradius der Kurve gefunden und getestet werden. Um dieses besser zu erklären soll in der nachfolgenden Abbildung 5.21 der elektronische Blindenhund aus der Vogelperspektive bei einer Kurvenfahrt, mit zugehörigen Strecken- und Winkelbezeichnungen, dargestellt werden.

Abbildung 5.21: Vogelperspektive einer Linkskurve mit Kurvenwinkel α

Die Abbildung 5.21 veranschaulicht die zurückgelegte Strecke des elektronischen Blindenhunds bei einer Linkskurve aus der Vogelperspektive. Das Prinzip basiert auf der unterschiedlichen Ansteuerung von Motor 1 und Motor 2.

Der Startpunkt liegt auf der horizontalen Linie mit den Punkten A und B. Der Punkt A kennzeichnet die Position des ersten Reifens bzw. die Position des ersten Motors. Wo hingegen der Punkt B die Position des zweiten Reifens markiert. Der Abstand zwischen Punkt A und Punkt B ist der bereits bekannte Reifenabstand L_R . Der Rote Strich kennzeichnet den Stab, welcher am Gehäuse des Blindenhunds befestigt ist und vom Benutzer gehalten wird. Das Ende des Kurvenmanövers kennzeichnen die Punkte D und C. Im Schnittpunkt der Geraden \overline{DC} und \overline{AB} ist der Momentanpol M und der Kurvenwinkel

α zu finden. Der Momentanpol beschreibt dabei den Drehpunkt eines starren Körpers bei einer ebenen Bewegung im Moment. Dieser Momentanpol oder Drehpunkt liegt in diesem Versuch immer auf der Linie von \overline{AB} , also auf der Linie der Antriebsachse.[22] Der Abstand vom Momentanpol zum Punkt A hängt von dem Innenradius r_i der Kurve ab, und hat Einfluss auf die zurückgelegte Strecke bei einem Kurvenmanöver. Das kleinste zu fahrende Kurvenmanöver wäre bei diesem Konzept eine Drehung aus dem Stand. Gefolgt von dem zweitkleinsten Manöver, bei dem der kleinste Innenradius $r_i = 0$ ist und der Momentanpol somit auf dem Punkt A oder Punkt B liegen würde. Hierbei würde nur ein einziger Motor in Bewegung sein. Die Strecken dieser Kurvenfahrten hängen vom Kurvenwinkel α und dem Innenradius r_i ab. Lediglich die Dauer der Kurvenfahrt ist variabel und wird durch die Geschwindigkeit des Motors bestimmt.

Wichtig für die Implementierung sind die bereits erwähnten Streckenlängen der Innen- und Außenbahnen der Kurve. Die Innenbahn bei einer Kurve ist l_{ri} und die Außenkurve ist l_{ra} . Das Verhältnis dieser beiden Längen beschreibt gleichzeitig das Verhältnis der Drehzahl der beiden Motoren (Motor 1 und Motor 2) bzw. deren Differential zueinander. Das Prinzip eines Differentials findet man in vielen Kraftfahrzeugen. Es ermöglicht, dass in Kurvenfahrten die Räder mit unterschiedlicher Geschwindigkeit und gleichbleibender Vortriebskraft drehen können.

Bestimmung des Kurvenradius

Maßgeblich für die Umsetzung eines erfolgreichen Kurvenmanövers, ist die Einstellung eines passenden Differentials der beiden Motoren. Hierfür muss der Kurveninnenradius r_i und die Geschwindigkeit, mit der die Kurve gefahren wird bekannt sein. Im diesem Fall wird die Geschwindigkeit für das Kurvenmanöver nur an einem Motor angepasst. Das Differential entsteht durch die Reduzierung der Geschwindigkeit des innenliegenden Motors. Der außenliegende Motor soll dabei mit der Anfangsgeschwindigkeit weiterlaufen. Dementsprechend soll der EGD mit der selben Geschwindigkeit aus der Kurve rausfahren, wie er auch in die Kurve reingefahren ist. Die Umsetzung eines benutzerfreundlichen Kurvenmanövers ist experimentell zu bestimmen und im praktischen Teil zu bewerten. Grundsätzlich gilt, für große Winkel wird eine enge bzw. scharfe Kurve gefahren und für kleine Winkel gegen null, wird eine sehr große Kurve gefahren. Es wird zu Anfang festgelegt, dass der kleinste Innenradius bei einer Kurve $r_{i,min} = 0.5 \text{ m}$ beträgt und der größte Innenradius $r_{i,max} = 4 \text{ m}$. Außerdem wird von einem linearen Zusammenhang zwischen Kurvenwinkel und Innenradius ausgegangen. Des Weiteren wurde durch die

Software des EGD ein vom Betrag maximaler Kurvenwinkel von $\alpha_{max} = 45^\circ$ vorgegeben. In den Testfahrten des Kapitels 6.2.4 soll dann geprüft werden, welche Einstellungen für eine Kurvenfahrt optimal sind. Folgender linearer Ansatz wurde verwendet (siehe auch Abbildung 5.22):

$$r_i(\alpha) = \frac{\Delta r}{\Delta \alpha} \cdot \alpha + r = \frac{r_{i,max} - r_{i,min}}{0^\circ - \alpha_{max}} \cdot \alpha + r_{i,max}. \quad (5.60)$$

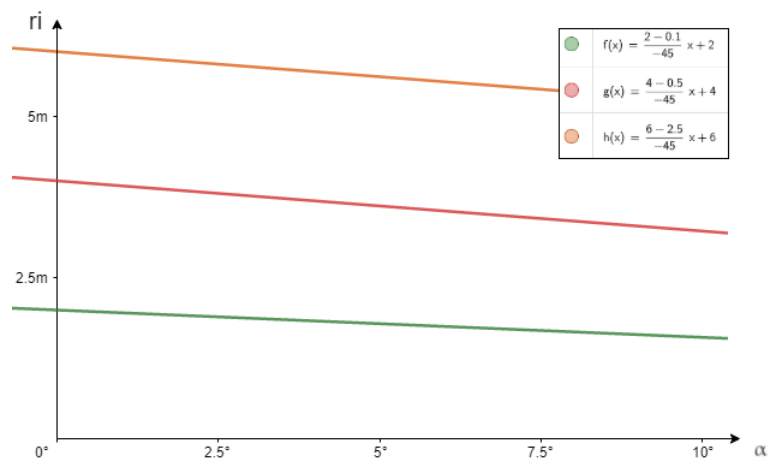


Abbildung 5.22: Linearer Zusammenhang zwischen α (X-Achse) und dem Innenradius r_i (Y-Achse)

Ein weiterer Ansatz für den Zusammenhang zwischen Kurvenwinkel und Innenradius könnte durch eine e-Funktion wie folgt implementiert werden.

$$r_i(\alpha, k) = \frac{e^{k(-\alpha + \alpha_{max})}}{x} + r_{i,min} \quad (5.61)$$

Mithilfe eines zusätzlichen Faktors k kann dann eine Funktionenschar erstellt und nach einem passendem Profil, für eine Kurvenfahrt gesucht werden (vgl. Abbildung 5.23). Der Winkel α liegt hierbei zwischen $0^\circ \leq \alpha \leq \alpha_{max}$. Die Besonderheit bei dieser Beziehung ist, dass der Innenradius $r_i(\alpha)$ für den Winkel $\alpha = 0^\circ$ gegen $r_i \rightarrow \infty$ gehen würde und für $r_i(45^\circ) = r_{i,min}$ ist. Dieser Zusammenhang passt, da bei einer Geradeausfahrt, der Winkel $\alpha = 0^\circ$ ist und der Innenradius gegen unendlich strebt.

Dazu siehe folgende Abbildung 5.23.

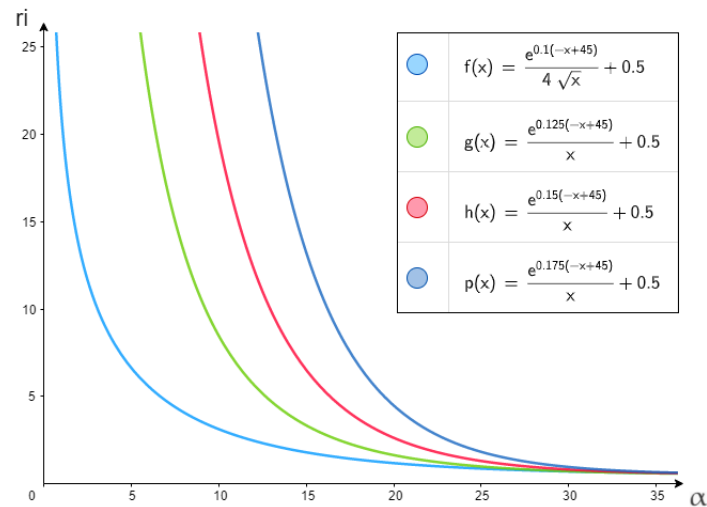


Abbildung 5.23: Nichtlinearer Zusammenhang zwischen α (X-Achse) und dem Innenradius r_i (Y-Achse)

Der außenradius r_a ergibt sich zwangsläufig über den Innenwinkel, da beide Motoren immer denselben Abstand L_R zueinander haben.

$$r_a(r_i) = r_i + L_R. \quad (5.62)$$

Bestimmung des Differentials

Dank des Zusammenhangs zwischen Kurvenwinkel und Innenradius, kann dann bestimmt werden, wie schnell sich der jeweilige Motor drehen muss. Zuerst wird der Quotient der Motorgeschwindigkeiten und Radien ermittelt:

$$\frac{v_{m,i}}{v_{m,a}} = \frac{r_i}{r_a} = \frac{l_{ri}}{l_{ra}} \quad (5.63)$$

Mit $v_{m,i}$ wird die Geschwindigkeit des innenliegenden Motors und mit $v_{m,a}$ die Geschwindigkeit des außenliegenden Motors beschrieben. Der Außenliegende Motor soll dabei immer mit der Schrittgeschwindigkeit laufen, so dass lediglich die Geschwindigkeit des innenliegenden Motors zu verändern ist. Damit der innenliegende Motor keinen ruckartigen Sprung zum Beginn des Kurvenmanövers machen muss, wird ein Beschleunigungsprofil

gefahren. Dieses muss so ausgelegt werden, dass das Streckenverhältnis von Innen- und Außenbahn wie erwähnt gleichbleibt.

Berechnung der Innen- und Außenbahnlänge:

Innenbahnlänge:

$$l_{ri}(r_i(\alpha)) = \frac{\alpha}{360} \cdot 2 \cdot \pi \cdot r_i. \quad (5.64)$$

Außenbahnlänge:

$$l_{ra}(r_a(r_i)) = \frac{\alpha}{360} \cdot 2 \cdot \pi \cdot r_a. \quad (5.65)$$

Berechnung des Beschleunigungsprofils mithilfe der folgenden Grafik 5.24.

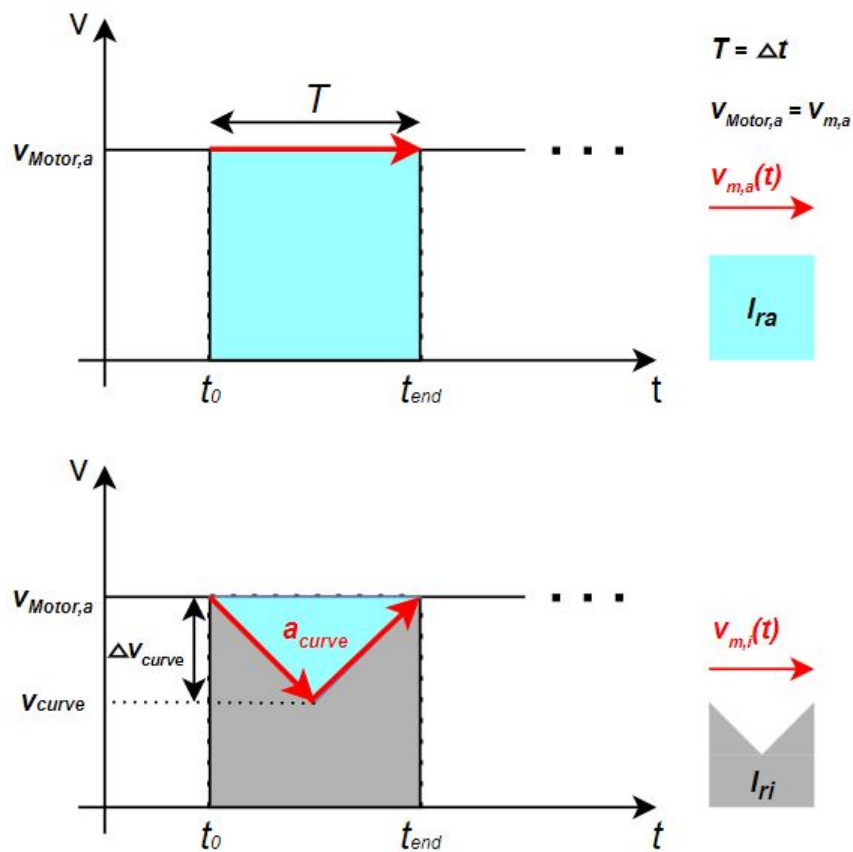


Abbildung 5.24: Differential der Motoren bei einer Kurvenfahrt

Auf der Abbildung 5.24 sind die Zeitverläufe der Geschwindigkeiten beider Motoren zu erkennen. Der obere Zeitverlauf (blau) veranschaulicht die Geschwindigkeitsänderung $v_{m,a}$ des außenliegenden Motors über die Zeit t . Diese Geschwindigkeit ist konstant, da es sich hierbei um den außenliegenden Motor handelt. Der untere Zeitverlauf veranschaulicht dabei die Geschwindigkeitsänderung des innenliegenden Motors $v_{m,i}$ (grau). Das Kurvenmanöver benötigt die Zeit $T = \Delta t$ und fängt bei t_0 an. Das Integral über dem Geschwindigkeitsverlauf ist die zurückgelegte Strecke. Es gibt die Außenstrecke l_{ra} (blau) und die Innenstrecke l_{ri} (grau). Wie bekannt, muss das Verhältnis der Kurvenlängen, also der Integrale über den Geschwindigkeitsverläufen von Außen- und Innenmotor gleichbleiben.

Für die Umsetzung in der Software werden im Folgenden die wichtigsten Beziehungen hergeleitet. Für die Kurvenfahrt wird die Beschleunigung des innenliegenden Motors a_{curve} , sowie die Endgeschwindigkeit v_{curve} und die Kurvenfahrtzeit Δt benötigt.

Bestimmung der Kurvenfahrtzeit Δt :

$$\Delta t(v_{m,a}; l_{ra}) = \frac{l_{ra}}{v_{m,a}}. \quad (5.66)$$

Bestimmung der Geschwindigkeit v_{curve} , mithilfe des Geschwindigkeitverlaufs $v_{m,i}$ aus Abbildung 5.24 (grau markierte Fläche). Das Integral über den Geschwindigkeitsverlauf $v_{m,i}$ muss der Strecke des Innenradius l_{ri} entsprechen:

$$\int_{t_0}^{t_{end}} v_{m,i}(t) dt \stackrel{!}{=} l_{ri}. \quad (5.67)$$

Mithilfe von einfachen geometrischen Flächenberechnungen über die Abbildung 5.24 kann l_{ri} wie folgt berechnet werden:

$$l_{ri} = (v_{m,a} - v_{curve}) \cdot \frac{\Delta t}{2} + v_{curve} \cdot \Delta t. \quad (5.68)$$

Im folgenden Schritt muss nur noch nach v_{curve} umgestellt werden, es ergibt sich

$$v_{curve} = \frac{l_{ri} - (v_{m,a} \cdot \frac{\Delta t}{2})}{0.5 \cdot \Delta t}. \quad (5.69)$$

Damit kann dann die Beschleunigung a_{curve} für die Kurvenfahrt des Innenmotors bestimmt werden.

$$a_{curve} = \frac{\Delta v_{curve}}{\Delta t} = \frac{v_{m,a} - v_{curve}}{\Delta t} = \frac{v_{m,a} - \frac{l_{ri} - (v_{m,a} \cdot \frac{\Delta t}{2})}{0.5 \cdot \Delta t}}{\Delta t} \quad (5.70)$$

$$a_{curve} = \frac{v_{m,a}}{\Delta t} - \frac{l_{ri} - (v_{m,a} \cdot \frac{\Delta t}{2})}{0.5 \cdot (\Delta t)^2} \quad (5.71)$$

$$a_{curve} = \frac{v_{m,a}}{\Delta t} - \frac{2 \cdot l_{ri} - (v_{m,a} \cdot \Delta t)}{(\Delta t)^2} \quad (5.72)$$

Auch wichtig zu erwähnen ist die zurückgelegte Strecke des Benutzers l_{bogen} , denn dieser wird wie bereits erwähnt bei einem Kurvenmanöver durch den Stab, auch über einen gewissen Radius r_{bogen} um den Drehpunkt M bewegt (vgl. Abbildung 5.25).

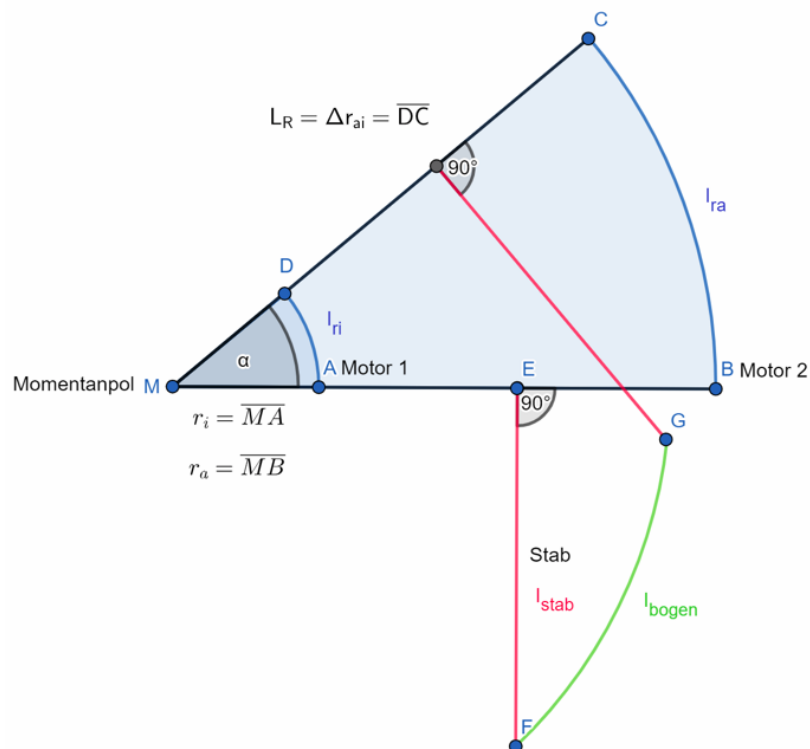


Abbildung 5.25: Vogelperspektive einer Linkskurve mit eingezeichnetem Benutzerweg l_{bogen} (grün)

Die grüne Kurve auf der Abbildung 5.25 stellt den Weg des Benutzers dar, welchen er tätigen muss, um dem Blindenhund bei einer Kurve zu folgen. Die Problematik liegt in der Länge des Stabs. Umso länger der Stab des Blindenhunds ist, umso größer wird auch der Weg, den dieser zurücklegen muss. Ein weiteres Problem könnte durch einen zu langen Stab entstehen, da dieser einen gewissen Hebel auf den Blindenhund ausübt und somit das Kurvenmanöver stört. Erst durch die Testfahrten kann ermittelt werden, welche Auswirkungen die Stablänge auf das Kurvenmanöver und das Fahrverhalten hat.

Die Strecke des Benutzers l_{bogen} bildet einen Kreisabschnitt mit dem Radius r_{bogen} und kann mit der folgenden Formel beschrieben werden:

$$l_{bogen} = \frac{\alpha}{360^\circ} \cdot 2 \cdot \pi \cdot r_{bogen}, \quad (5.73)$$

wobei sich r_{bogen} wie folgt ergibt

$$r_{bogen} = \sqrt{\left(r_i + \frac{L_R}{2}\right)^2 + (l_{stab})^2}. \quad (5.74)$$

Algorithmus für die Kurvenfahrt

Die Kurvenfahrt wird mit der Funktion `set_angle(double angle)` aufgerufen. Der Parameter `angle` gibt die gewünschte Richtung vor, wobei die Software einen maximalen Winkel von 45° und einen minimalen von -45° , aufgrund des Sichtfeldes zulässt. Ein positiver Winkel stellt eine Rechtskurve und ein negativer eine Linkskurve dar. Für die Implementierung einer Kurvenfahrt ist zu bedenken, dass es hierbei verschiedene Verläufe während der Laufzeit gibt. Es besteht die Möglichkeit, dass die Funktion zum Zeitpunkt t_1 aufgerufen und erfolgreich zum Zeitpunkt t_3 beendet wird (siehe Abbildung 5.26, schwarze Kurve). Des Weiteren besteht die Möglichkeit, dass die Funktion erneut zur Kurskorrektur, vor dem Beenden des Manövers zum Zeitpunkt t_2 aufgerufen wird (siehe Abbildung 5.26, blaue und rote Kurve). Auch kann das Manöver durch eine andere Funktion ganz unterbrochen werden.

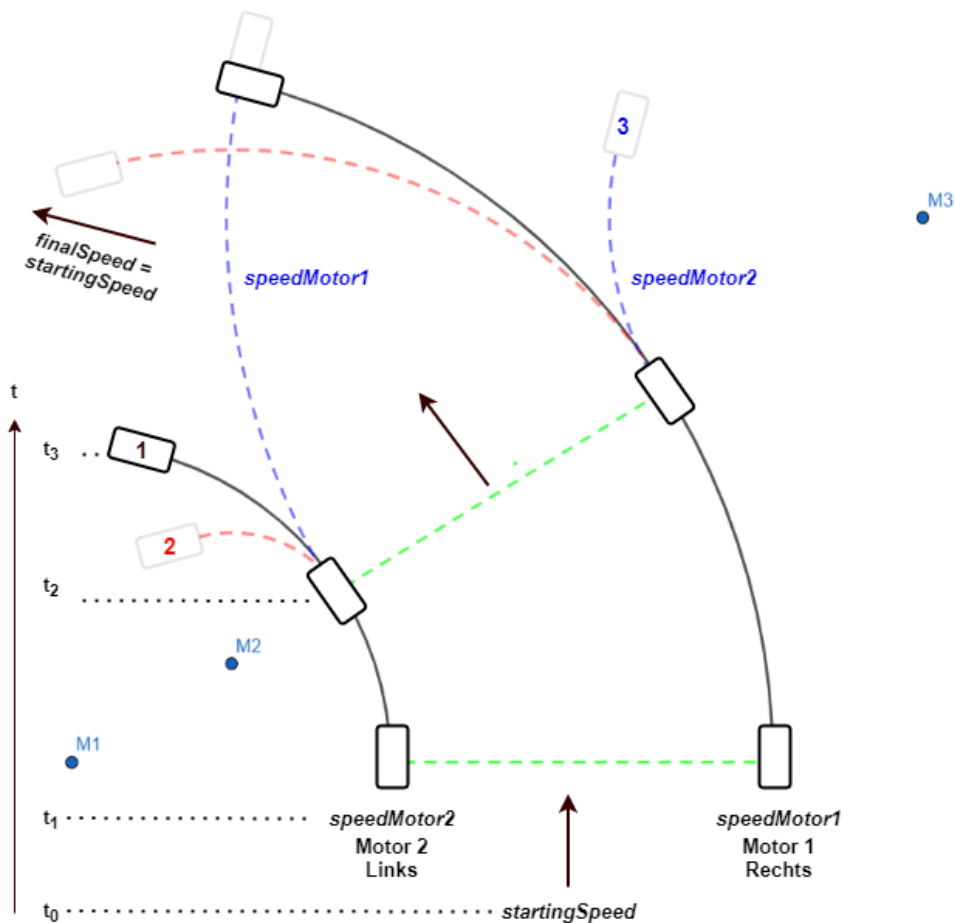


Abbildung 5.26: Mögliche Verläufe von Kurvenfahrten

Die auf Abbildung 5.26 zu sehenden Punkte M1, M2 und M3 stellen die Drehpunkte für die entsprechenden Kurvenbahnen (Bahn 1, Bahn 2 und Bahn 3) dar. Bevor ein Kurvenmanöver eingeleitet werden kann, muss der EGD in Bewegung sein. Dieses ist zum Zeitpunkt t_0 zu erkennen. Hierbei bewegt sich der Blindenhund mit einer Geschwindigkeit (*startingSpeed*) und geht zum Zeitpunkt t_1 in eine Linkskurve über. Der außenliegende Motor (hier Motor 1) übernimmt die Geschwindigkeit beim Start der Kurve und behält diese bis zum Zeitpunkt einer Änderung bei. Sofern das Manöver erneut zum Zeitpunkt t_2 , vor Abschluss des ersten Kurvenmanövers mit anderen Parametern (*angle*) aufgerufen wird, müssen die Werte für die Berechnung des Differential, zur Kurvenfahrt neu ausgelegt werden. Es ist dabei zu unterscheiden, ob der Blindenhund in eine neue Richtung übergeht, oder ob sich lediglich der Winkel in seiner aktuellen Richtung verändert. Die folgende Grafik 5.27 soll den Ablauf von *set_angle()* exemplarisch veranschaulichen.

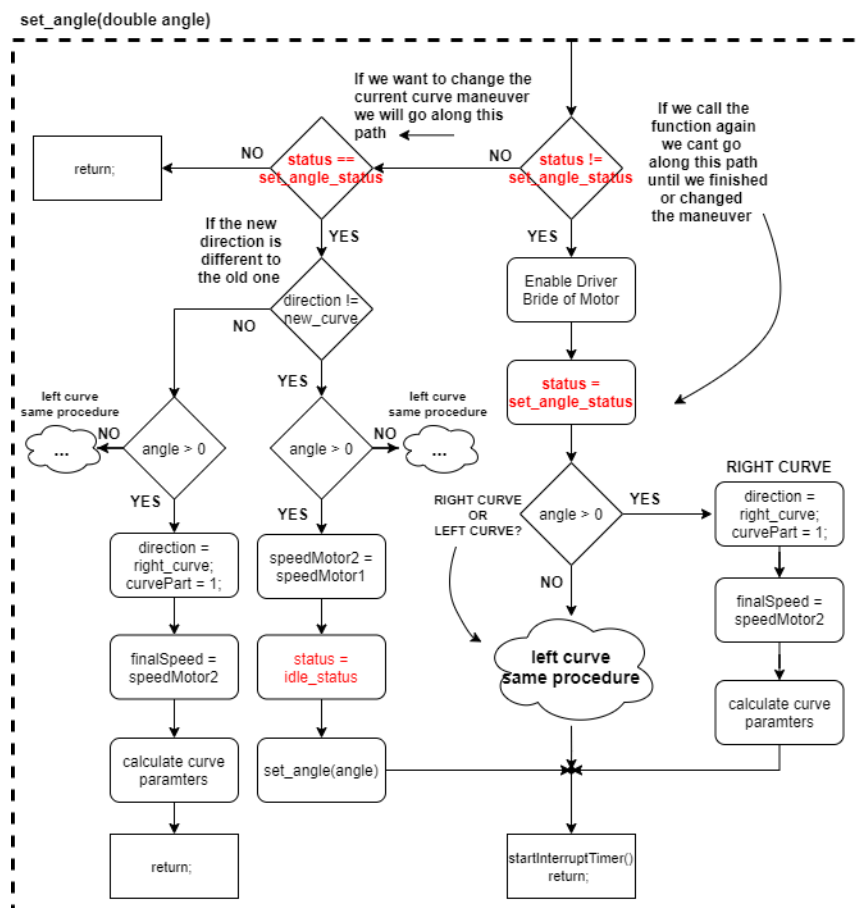


Abbildung 5.27: Flowchart zur Übersicht vom Funktionsablauf *set_angle()* bei einer Rechtskurve

Im Anhang ist die Abbildung 5.27 in vergrößerter Darstellung zu finden (siehe Anhang, Abbildung A.3).

Die Abbildung 5.28 veranschaulicht die Umsetzung des Kurvenmanövers mithilfe der Interrupt-Funktion *timerInterruptHandler()*. Die Abbildung ist in vergrößerter Variante im Anhang zu finden (siehe Abbildung A.4).

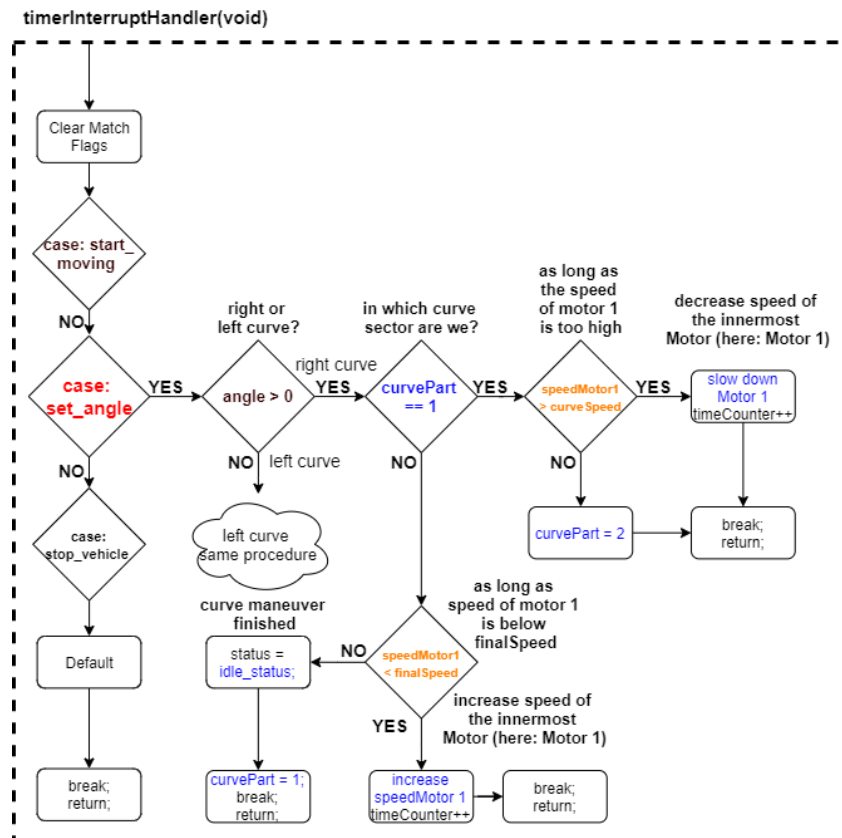


Abbildung 5.28: Funktionsablauf von *set_angle()* im Interrupthandler (*timerInterruptHandler()*) bei einer Rechtskurve

Das Kurvenmanöver wird in zwei Abschnitte unterteilt. Diese Abschnitte werden in der Software durch die Variable *curvePart* initialisiert. Nach der Richtungsbestimmung des Manövers, wird im ersten Kurvenabschnitt, die Geschwindigkeit des innenliegenden Motors solange reduziert bis die im voraus berechnete Geschwindigkeit (*curveSpeed*) erreicht wurde. Anschließend wird in *curvePart 2* übergegangen, in dem die Geschwindigkeit zurück auf die Anfangsgeschwindigkeit *startingSpeed* gefahren wird. Das Beschleunigungsmanöver für den innenliegenden Motor, während einer Kurvenfahrt ist symmetrisch aufgebaut (vgl. Abbildung 5.24), weshalb beide Phasen gleich viel Zeit in Anspruch nehmen

und die selbe Strecke zurücklegen. Die Berechnungen für die entsprechenden Parameter einer Kurvenfahrt sind im Kapitel 5.3.5 ab Seite 71 zu finden.

6 Praktischer Teil

In diesem Teil sollen insbesondere die Überlegungen und Konzepte aus dem Kapitel 5 praktisch umgesetzt werden. Das Kapitel umfasst den Aufbau einer Teststation, die Einrichtung der Entwicklungsumgebung, sowie die Pinbelegung des Hardwareaufbaus und die Inbetriebnahme der Motoren. Am Ende des Kapitels soll der elektronische Blindenführhund mit dem neuen Antriebssystem, im Hinblick auf die Anforderungen aus Kapitel 4, auf verschiedenen Strecken ausgetestet werden.

6.1 Aufbau einer Teststation

Im ersten Schritt wurde ein Aufbau für einen Prüfstand entwickelt. Mit diesem sollte es möglich sein, den Motor, sowie den Mikrocontroller auf verschiedene Funktionalitäten zu testen. Das folgende Bild 6.1 veranschaulicht den Schaltplan dieses Aufbaus.

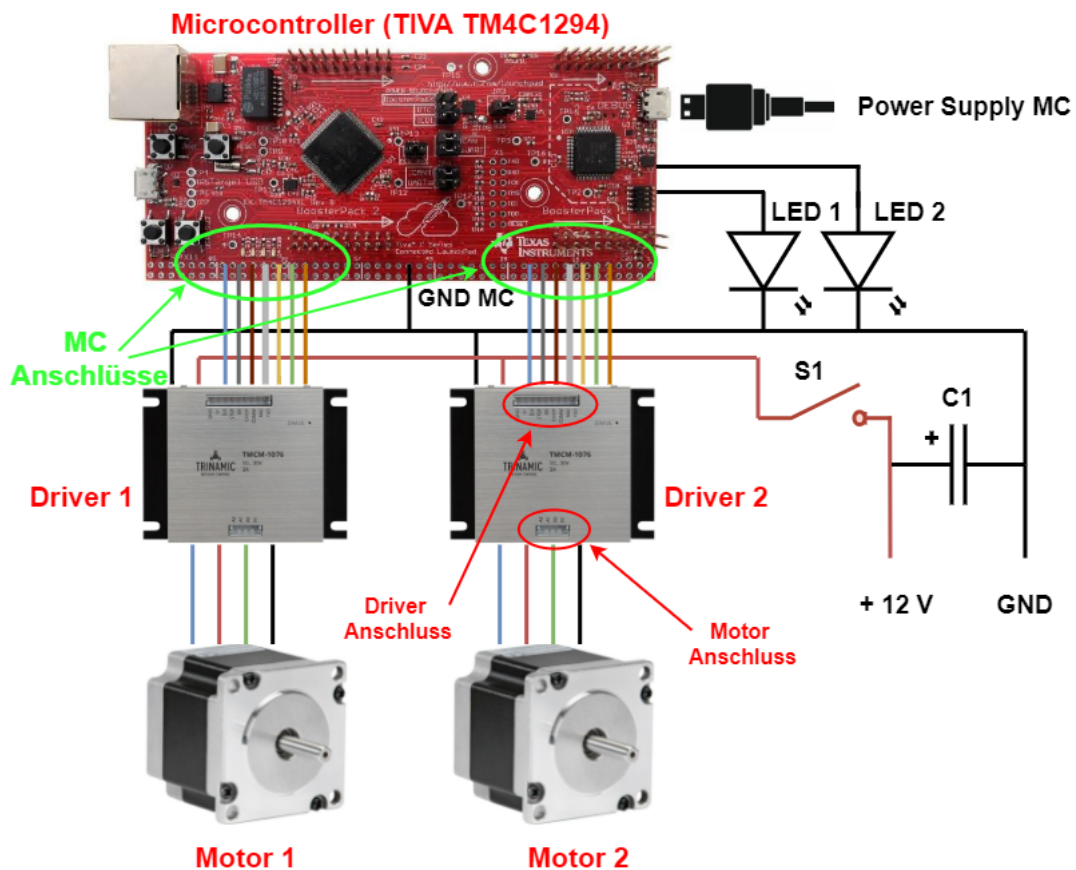


Abbildung 6.1: Schematischer-Schaltplan für Teststation, inklusive MC, Motor und Driver

Die Umsetzung des Schaltplans ist anhand von Abbildung 6.2 zu erkennen.

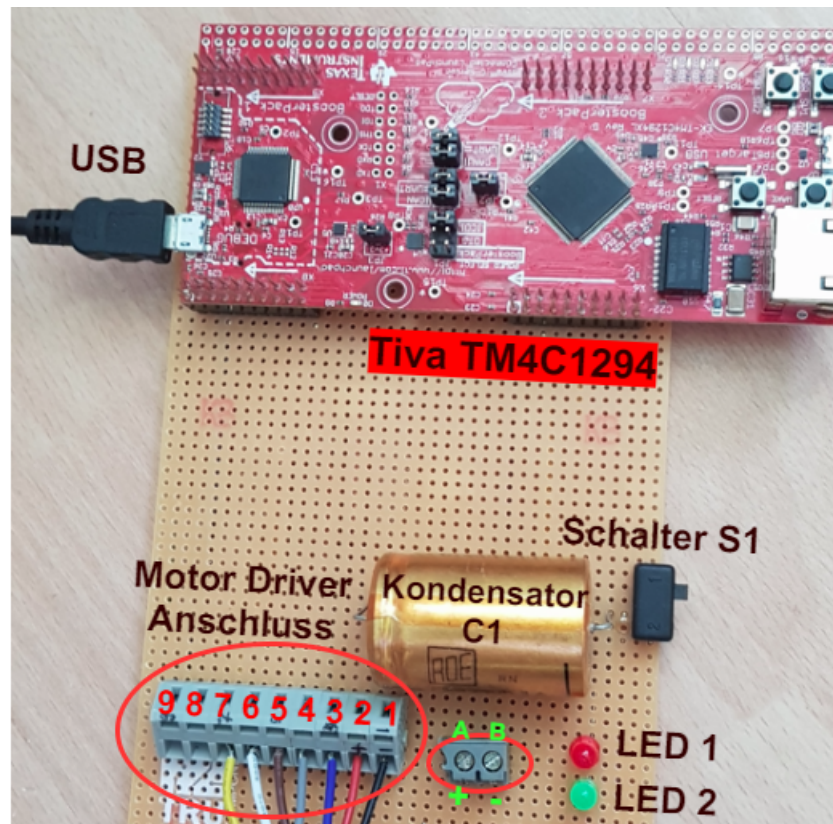


Abbildung 6.2: Teststation mit Platine, Mikrocontroller und Bauelementen

Folgende Bauelemente wurden verwendet:

- Elektrolyt Kondensator C1: 4600 μF
- Schalter S1 - An- und Abschalten der Spannungsversorgung
- LED 1: Rot Statuslampe 1
- LED 2: Grün Statuslampe 2
- Anschlussklemmen für Motor-Driver

Die Teststation ermöglicht den Anschluss eines Motortreibers über die Anschlussbuchse mit den 9 Pins. Im Verlauf des Projekts wurde dem Aufbau noch ein Potentiometer hinzugefügt, um verschiedene Eingaben während eines Testlaufs prüfen zu können. Auch ist

zu beachten, dass der Ground vom Microcontroller mit dem Ground für die Stromversorgung des Motors verbunden werden muss.

Im Verlauf der Arbeit wurde eine weitere Platine für den Einbau in den EGD konzipiert. Sie basiert auf demselben Schaltplan wie in Abbildung 6.2 und kann mithilfe des nächsten Kapitelabschnitts 6.1.1 und den Tabellen zur Pinbelegungen beschaltet werden.

6.1.1 Anschlüsse und Pinbelegung

Anschlussplan der Teststation

Anschlussklemme	Anschluss	Bedeutung
A	12 V	Stromversorgung Motor
B	GND	Masse der Stromversorgung
1	GND	GND Anschluss des Drivers
2	V+	10 V - 30 V (DC)
3	DIR	Drehrichtung
4	STEP	Pulsfrequenz für Schritte
5	EN	Aktiviert die H-Brücken
6	CHOP	Modi Auswahl (StealthChop oder SpreadCycle)
7	COMM	
8	RXD	UART Receive
9	TXD	UART Transmit

Tabelle 6.1: Anschlussplan der Teststation

Pin 1 bis 9 stehen für die Anschlüsse des Motor-Divers. Siehe dazu auch Tabelle 5.4

Anordnung der Motoren

Die folgende Tabelle 6.2 soll dabei die Anordnung der Motoren und Treiber für den Gesamtaufbau veranschaulichen.

Antrieb	Anordnung	Software Parameter [km/h]
Motor 1 und Driver/ Treiber 1	Rechts	<i>speedMotor1</i>
Motor 2 und Driver/ Treiber 2	Links	<i>speedMotor2</i>

Tabelle 6.2: Anordnung der Motoren und Treiber im EGD

Anschlussplan - Pinbelegung am Mikrocontroller (Abbildung 6.2)

Microcontroller Verbindung	Motor-Driver-1	Bedeutung
PC4	3: DIR	Drehrichtung
PB3 (Timer 5B)	4: STEP	Pulsfrequenz für Schritte
PC5	5: EN	Aktiviert die H-Brücken
PC6	6: CHOP	Modi (StealthChop o. SpreadCycle)
PC7	7: COMM	
PA6 / nicht in Benutzung	8: RXD	UART Receive
PA7 / nicht in Benutzung	9: TXD	UART Transmit

Tabelle 6.3: Anschlussplan für Motor 1

Microcontroller Verbindung	Motor-Driver-2	Bedeutung
PE0	3: DIR	Drehrichtung
PL4 (Timer 0A)	4: STEP	Pulsfrequenz für Schritte
PE1	5: EN	Aktiviert die H-Brücken
PE2	6: CHOP	Modi (StealthChop o. SpreadCycle)
PE3	7: COMM	
PA4 / nicht in Benutzung	8: RXD	UART Receive
PA5 / nicht in Benutzung	9: TXD	UART Transmit

Tabelle 6.4: Anschlussplan für Motor 2

Microcontroller Pin	Anschluss
PG1	Veraltet: Servo Motor RC-CAR (PWM5)
PK4	Veraltet: ESC RC-CAR (PWM6)
PP0 (U6Rx)	TX Lidar Sensor
PP1 (U6Tx)	RX Lidar Sensor
PE4	ADC-Analoger-Eingang (Potentiometer)

Tabelle 6.5: Weitere Pinbelegung des Microcontrollers

Für den Zusammenbau ist darauf zu achten, dass sämtliche Massen miteinander verbunden werden. Die Masse der Versorgungsspannung für den Motor muss eine Verbindung zur Masse für die Stromversorgung des MC haben.

6.1.2 Tiva TM4C1294 Mikrocontroller

Die zentrale Recheneinheit für das bestehende Projekt des elektronischen Blindenhunds ist der Microcontroller Tiva TM4C1294 von Texas Instruments.[2] Für die Umsetzung der neuen Embedded Software wurde dieser Controller verwendet. Der Controller ist auf dem Bild 6.2 zu sehen.

6.1.3 Code Composer Studio (CCS)

Als Entwicklungsumgebung für die Programmierung und die Umsetzung auf den Mikrocontroller dient die Software Code Composer Studio (verwendete Version: V9). Hierbei ist darauf zu achten, dass sowohl die Compiler Version[3] (verwendete Version: TI v20.2.5.LTS) als auch die Tiva Ware (verwendete Version: 2.2.0.295) auf einem aktuellen Stand sind. Zuzüglich müssen die Headerfiles des TM4C1294NCPDT für die Verwendung von den *Defines und Makros* der verschiedenen Register mit eingebunden werden. Die Headerdatei wird dann im späteren Softwareprojekt durch *include "inc/tm4c1294ncpdt.h"* dem Projekt hinzugefügt. Gegebenenfalls sollte auch geprüft werden, ob die serielle Schnittstelle, das *Stellaris In-Circuit Debug Interface*, vom Gerätemanager und in den Projekt-Properties, unter General, richtig konfiguriert wurden.

6.2 Durchführung

In der Durchführung wurde das neue Antriebssystem mithilfe der gefertigten Teststation sowie des elektronischen Blindenführhunds und der dazugehörigen Software auf Funktionalität geprüft.

6.2.1 Inbetriebnahme des Motors

Für die erste Inbetriebnahme wurde der Motor mittels eines Funktionsgenerators, mit Netzteil und Versorgungsspannung über das Treibermodul in Betrieb genommen. Gleichzeitig wurden Messungen zur Geräusch- und Geschwindigkeitsveränderung gemacht. Eine wichtige Erkenntnis bei der Inbetriebnahme ist, dass die Driverstage *EN* des Motortreibers während der Anschaltzeit *Power Up Time* low gesetzt sein muss. Erst nach wenigen Sekunden der Anschaltzeit kann der Motortreiber enabled werden.

Nachfolgend zu der Inbetriebnahme wurde das entwickelte Konzept für die Antriebssteuerung und Bahnführung auf dem Mikrocontroller implementiert. Hierbei wurden zwei verschiedene Projekte verwendet, die im nächsten Abschnitt näher erklärt werden.

6.2.2 Implementierung der Fortbewegungsstrategie auf dem Mikrocontroller

Für die Umsetzung der Software sowie weiterer Versuche zur Validierung der entwickelten Konzepte, wurde mit dem Teststand aus Kapitel 6.1 das Projekt *EGD_Teststation* in der Entwicklungsumgebung CCS erstellt. Das Projekt umfasst einige wichtige Funktionen die in der folgenden Tabelle 6.6 aufgelistet werden:

Funktions Name	Bedeutung
<i>void configPorts(void);</i>	Konfiguriert die GPIO Pins des Mikrocontrollers entsprechend der Pinbelegung aus Kapitel \ 6.1.1
<i>void configTimer(void);</i>	Konfiguriert die benötigten Timermodule für die Generierung der Schrittfrequenz
<i>void start_moving(double finalSpeed);</i>	Funktion zum Starten der Motoren. Funktion beschleunigt die Motoren auf finalSpeed (km pro Stunde)
<i>void stop_vehicle(void);</i>	Funktion zum Stoppen der Motoren
<i>void set_angle(double angle);</i>	Funktion für die Kurvenfahrt. Parameter gibt den Winkel (in Grad) der Kurvenfahrt an
<i>void setSpeed_Motor_1(double runningSpeed);</i>	Setzt die Geschwindigkeit von Motor 1 auf runningSpeed (in km pro Stunde)
<i>void setSpeed_Motor_2(double runningSpeed);</i>	Setzt die Geschwindigkeit von Motor 2 auf runningSpeed (in km pro Stunde)
<i>void setRunningSpeed(double runningSpeed);</i>	Setzt die Geschwindigkeit von Motor 1 und 2 auf runningSpeed (km pro Stunde)
<i>void startInterruptTimer(void);</i>	Startet den Timer für die Generierung der Interrupts
<i>void timerInterruptHandler(void);</i>	Wird aufgerufen wenn der Interrupt durch den Timer ausgelöst wurde.

Tabelle 6.6: Übersicht der Funktionen für das Softwareprojekt *EGD_Teststation*

Im Folgenden soll die Tabelle 6.7 eine Übersicht zu den wichtigsten *Defines* und *Makros* gegeben, welche das Fahrverhalten und die Antriebssteuerung des Blindenhunds beeinflussen:

Variable	Bedeutung	Default Value
TON	Bestimmt die Pulsbreite der Schrittfrequenz	10e-6
samplingTime	Bestimmt die Zeitdifferenz zwischen zweier direkt aufeinander folgender Interrupts (in Sekunden)	0.05
ri_min	Kleinster Innenradius für Kurvenfahrt (in Metern)	0.1
ra_max	Größter Außenradius für Kurvenfahrt(in Metern)	2
minSpeed	Kleinste Geschwindigkeit des EGD (in km/h)	0.1
maxSpeed	Höchste Geschwindigkeit des EGD (in km/h)	5.0
acceleration	Beschleunigung für Anfahren und Abbremsen (in m/s^2)	0.7

Tabelle 6.7: Übersicht zu wichtigen *Defines* bzw. *Makros* für die EGD Teststation

6.2.3 Implementierung in das Gesamtsystems des EGD

Nachdem die Implementierung und Umsetzung mit der Teststation erfolgreich abgeschlossen wurde, konnte an der Implementierung der Software in das Gesamtprojekt *watchdog* des EGDs weitergearbeitet werden. Besonders wichtig ist, dass geprüft wird, welche Anschlüsse und Variablennamen des Microcontrollers in der Software bereits verwendet werden (siehe dazu Kapitel 6.1.1 Tabelle 6.5). Nachdem dies ausfindig gemacht wurde, mussten nur noch die Funktionen entsprechend den Vorgaben aus der aktuellen EGD Software angepasst (siehe dazu Kapitel 5.3.2) und in den richtigen Dateien eingebunden werden (siehe dazu Anhang Abbildung A.1). Es ist weiterhin darauf zu achten, dass bei der Implementierung der Funktionen und Makros, in die verschiedenen Dateien, Referenzen über *include* Befehle gesetzt werden.

6.2.4 Testfahrten

Für die Validierung der entwickelten Software und des Antriebssystems, wurden im Folgenden der elektronische Blindenführhund mit dem neuen Antrieb getestet. Da durch einen Kurzschluss auf einer Platine während eines Testlaufs zwei essentielle Motor-Driver

kaputt gegangen sind, hat sich die praktische Umsetzung und Validierung stark verzögert. Die zwei Einheiten wurden nachbestellt, kamen jedoch durch Lieferprobleme verspätet an. Konzepte für Fahrverhalten und Software konnten in dieser Zeit nur theoretisch entwickelt und nur zum Teil mit einem Oszilloskop geprüft werden. Für die folgenden Testfahrten konnte somit aus zeitlichen Gründen kein geeigneter Messstand mehr aufgebaut werden, um eine umfangreiche und detaillierte Testreihe aufnehmen zu können.

Zuerst wurde eine Strommessung vollführt, um für den späteren Einbau eines Akkus die benötigte Kapazität abzuschätzen.

Strommessung

Die Messung des Stromes wurde mit und ohne Belastung der Antriebswelle vorgenommen. Die Belastung während der Fahrt wurde durch das Festhalten der Welle simuliert. Gemessen wurde der Effektivwert des Stromes, bei einer Betriebsspannung von 14 V. Da es sich bei dem Motor um einen Schrittmotor handelt, ist die Stromaufnahme in nahezu allen Belastungszuständen sehr ähnlich. Die folgenden Messwerte können dies bestätigen.

Belastung	RPM [min^{-1}]	I_{motor} [mA]
Nein	100	751
Nein	200	783
Ja	100	840
Ja	200	908
Mittelwert \bar{x}	150	820

Tabelle 6.8: Strommessung des Motors bei 14 V *Betriebsspannung*

Da es sich hierbei um eine grobe Abschätzung zur Einschätzung der Leistungsaufnahme handelt, ist es nicht von großer Relevanz eine umfangreiche und präzise Messreihe aufzustellen. Zur weiteren Reduzierung der Stromaufnahme können über die UART Verbindung interne Motortreiber-Einstellungen vorgenommen werden, die beispielsweise den Ruhestrom oder den Maximalstrom begrenzen.

Ergebnis der Strommessung

Die Stromaufnahme während des Betriebs, mit Belastung an der Welle, liegt unterhalb der Vorgaben zum maximalen Dauerstrom (vgl. Kapitel 4) und entspricht somit den Anforderungen. Damit entspricht die Leistungsaufnahme für einen Motor in etwa:

$$P_{zu} = I_{motor} \cdot U_q = 0.82 \text{ A} \cdot 14 \text{ V} = 11.48 \text{ W} \quad (6.1)$$

$$P_{zu} \approx 12 \text{ W}. \quad (6.2)$$

Die Gesamtleistung für die Antriebseinheit beträgt somit $P_{ges} = P_{zu} \cdot 2 = 24 \text{ W}$. Das Ergebnis kann jedoch durch die Belastung sowie die Erwärmung des Motors schnell verändert werden.

Teststrecke und Testablauf

Im nächsten Schritt werden die Testfahrten inklusive der Ergebnisse aufgeführt und zusammengestellt. Die Testfahrten wurden mit dem neuen Antriebssystem (siehe Kapitel 5.2.2 Abbildung 5.12) und der neuen Software vollzogen. Es handelt sich bei der Teststrecke ausschließlich um eine ebene Fahrbahn ohne Hindernisse mit einem Linoleum Boden. Eine umfangreiche Prüfung, mit verschiedenen Streckenabschnitten, konnte durch die limitierte Reichweite der Kabellänge, zur Stromversorgung des Systems nicht vollführt werden. Ein passender Akku für die Stromversorgung der Antriebseinheit war zu diesem Zeitpunkt noch nicht im EGD verbaut. Die Testung erfolgt in vier Schritten:

- Anfahren und Abbremsen
- Geradeausfahren
- Kurvenfahrten
- Gemischtes Fahren

Anfahren und Abbremsen

Das Anfahren und Abbremsen wurde über die zwei folgenden Funktionen eingeleitet:

- *start_moving(endSpeed)*
- *stop_moving(void)*

Über den Parameter *endSpeed* konnte die Endgeschwindigkeit des EGDs festgelegt werden. Der EGD hat nach Aufruf der Funktion *start_moving(endSpeed)* die Fahrt aufgenommen und solange beschleunigt bis er die Endgeschwindigkeit erreicht hat. Das Beschleunigungsprofil wurde nach den Berechnungen aus Kapitel 5.3.3 ausgelegt. Nach der Durchführung des Anfahrens wurde der EGD mit der Funktion *stop_moving(void)* angehalten. Diese Funktion bremst den EGD mit dem selben Beschleunigungsprofil wie beim Anfahren. Das Manöver wird solange ausgeführt bis der Blindenhund im Stillstand ist.

Die Ergebnisse sind der folgenden Tabelle zu entnehmen:

Manöver	Beschl. [m/s ²]	End. Geschw. [km/h]	Geräuscentw. [groß, mittel, klein]	Funktion Erfolg / Nicht	Fahrverhalten gut / mittel / schlecht
Anfahren	0.5	5	mittel	Erfolgreich	gut
Anfahren	0.7	5	mittel	Erfolgreich	gut
Anfahren	1	5	mittel	Erfolgreich	mittel
Abbremsen	0.5	0	mittel	Erfolgreich	gut
Abbremsen	0.7	0	mittel	Erfolgreich	gut
Abbremsen	1	0	mittel	Erfolgreich	gut

Tabelle 6.9: Ergebnisse zum Anfahren und Abbremsen

Die Messung zum Anfahren und Abbremsen konnte trotz der geringen Reichweite des Kabels erfolgreich durchgeführt werden. Festgestellt wurde, dass ein passendes Beschleunigungsverhalten in den theoretischen Berechnungen (siehe Kapitel 5.3.3) ermittelt wurde. Dies lag zwischen $0.5 \frac{m}{s^2}$ und $0.7 \frac{m}{s^2}$. Auch konnte festgestellt werden, dass beim Anfahren und Abbremsen die Geschwindigkeitsvorgabe kein Problem für die Führung des Benutzers dargestellt hat (vgl. Kapitel 5.3.2: Geschwindigkeitsvorgabe). Dies war nur dann ein Problem, wenn die Geschwindigkeit des Benutzers zu stark reduziert wurde. Bezüglich der Geräuscentwicklung, konnten Vibrationen und Resonanzen in allen Beschleunigungsprofilen festgestellt werden. Diese waren allerdings nicht zu gravierend und haben auch auf das Fahrverhalten beim Beschleunigen kein Einfluss gehabt. Um dennoch möglichen Motorschäden in zukünftigen Projekten und Testfahrten vorzubeugen, sollte an dieser Stelle gegebenenfalls nachgearbeitet werden.

Geradeausfahren

Neben dem Anfahren und Abbremsen wurde ebenfalls geprüft, wie gut der Blindenhund seine Spur halten konnte und welche Geschwindigkeiten angenehm für eine Führung ist. Hierfür wurden Geschwindigkeiten von $1 \frac{km}{h}$ bis $5 \frac{km}{h}$ getestet.

Die nächste Tabelle veranschaulicht die Ergebnisse zum Geradeausfahren:

Manöver	Beschl. [m/s ²]	End. Geschw. [km/h]	Geräuscentw. [groß, mittel, klein]	Funktion Erfolg / Nicht	Fahrverhalten gut / mittel / schlecht
Geradeaus	0.7	1	klein	Erfolgreich	schlecht
Geradeaus	0.7	2	klein	Erfolgreich	mittel
Geradeaus	0.7	3	mittel	Erfolgreich	gut
Geradeaus	0.7	4	mittel	Erfolgreich	gut
Geradeaus	0.7	5	groß	Erfolgreich	mittel

Tabelle 6.10: Ergebnisse zum Geradeausfahren

Auch hier stimmen die Ergebnisse mit den theoretischen Berechnungen überein. Die im Voraus berechneten Geschwindigkeiten beliefen auf $v_{EGD} = 4.7 \frac{km}{h}$. Die Messung hat ergeben, dass eine angenehme Geschwindigkeit im Bereich zwischen $3 \frac{km}{h}$ und $4 \frac{km}{h}$ liegt. Die Geräuscentwicklung ist hierbei etwas größer, wie im Vergleich zu den sehr niedrigen Geschwindigkeiten von $1 \frac{km}{h}$ bis $2 \frac{km}{h}$. Bei Geschwindigkeiten im Bereich von $5 \frac{km}{h}$ sind hierbei mehr Motorgeräusche wahrzunehmen. Das einbauen von Dämpfern könnte die Geräuscentwicklung reduzieren.

Kurvenfahrten

Die Kurvenfahrten wurden durch den Befehl `set_angle(angle)` eingeleitet. Mit dem Befehl `angle` konnte der gewünschte Winkel eingestellt werden. Zu beachten ist, dass der Blindenhund vorher auf eine gewisse Geschwindigkeit gefahren werden musste, um ein Kurvenmanöver auszuführen. Im Folgenden werden die Ergebnisse einer Rechtskurve mit verschiedenen Winkeln und Geschwindigkeiten aufgeführt. Die Variation der Geschwindigkeit soll dabei Auskunft über den Einfluss auf das Kurvenmanöver geben. Für eine Linkskurve konnten sehr ähnliche Messungen aufgenommen werden, weshalb diese im Folgenden nicht aufgeführt werden.

Die Messungen zu der Kurvenfahrt war aufgrund des bereits erwähnten Anschlusses an eine externe Stromquelle mit begrenzter Kabellänge nur bedingt möglich. Besonders der Kurvenradius war schwer zu messen. Trotzdem konnten zu den verschiedenen Winkeln

Manöver	Winkel [deg]	Geschw. [km/h]	Funktion Erfolg / Nicht	Fahrverhalten gut / mittel / schlecht
Kurvenfahrt	5	2	Erfolgreich	gut
Kurvenfahrt	15	2	Erfolgreich	mittel
Kurvenfahrt	25	2	Nicht erfolgreich	schlecht
Kurvenfahrt	35	2	Nicht erfolgreich	schlecht
Kurvenfahrt	45	2	Nicht erfolgreich	schlecht
Kurvenfahrt	5	4.5	Erfolgreich	gut
Kurvenfahrt	15	4.5	Erfolgreich	mittel
Kurvenfahrt	25	4.5	Nicht erfolgreich	schlecht
Kurvenfahrt	35	4.5	Nicht erfolgreich	schlecht
Kurvenfahrt	45	4.5	Nicht erfolgreich	schlecht

Tabelle 6.11: Ergebnisse zur Kurvenfahrt

und Geschwindigkeiten einige Werte ermittelt werden. Bezugnehmend auf kleine Winkel $\alpha \leq 5^\circ$, konnte der Blindenführhund gute Ergebnisse erzielen und ebenfalls ein angenehmes Fahrverhalten während der Kurve ermöglichen. Dahingegen konnten größerer Winkel $\alpha > 5^\circ$ bereits nur noch sehr schwierig umgesetzt werden. Auch konnte festgestellt werden, dass in dem Geschwindigkeitsbereich von $2 \frac{km}{h}$ bis $4.5 \frac{km}{h}$ die Geschwindigkeit keinen merkbaren Einfluss auf das Lenkverhalten hat. Die theoretischen Überlegungen aus Kapitel 5.1.2 haben sich als richtig erwiesen. Der bereits erwähnte Hebel durch den Stab am EGD hat die Lenkung bei kleinen Radien stark beeinträchtigt. Der Hebel wurde umso kleiner, desto geringer der Kurvenwinkel α war. Um die Ergebnisse zu optimieren wäre eine Möglichkeit, das Differential zu verändern. Hierbei kann versucht werden andere Beschleunigungsprofile zu fahren. Auch könnte mithilfe des Lösungsansatzes aus Kapitel 5.1.2 ein Kugelgelenk implementiert werden und in Folge dessen die Messung wiederholt werden. Sollte sich anschließend keine Verbesserung zeigen, so würde es sich empfehlen, auf das zweite Antriebskonzept umzusteigen (vgl. Kapitel 5.1.3, zweites Antriebskonzept mit Lenkachse).

Gemischtes Fahren

Anschließend sollte im letzten Schritt der Messung eine beliebige Abfolge an Fahrmanövern durchgeführt werden, um das Fahrverhalten in einer realen Situation möglichst präzise einschätzen zu können. Die Problematik bezieht sich hierbei wiederholt auf den eingeschränkten Fahrweg durch das Anschlusskabel. Trotzdem konnte festgestellt werden, dass eine beliebige Abfolge an Anfahren und Abbremsmanövern erfolgreich absolviert werden konnte. Lediglich bei den Kurven ist es noch nicht möglich schnelle Änderungen des Kurses vorzunehmen.

Bemerkungen

Wichtig zu bemerken ist, dass bei der Verwendung des Reifens darauf zu achten ist, dass dieser eine möglichst gute Auflagefläche mit viel Grip bietet. Dies erhöht zwar die Reibung und ist daher nicht so effizient, dahingegen wird allerdings das Fahrverhalten insbesondere in den Kurven, aber auch der Geräuschpegel verbessert.

7 Auswertung

Im Folgenden werden die Ergebnisse dieser Arbeit zusammengefasst und ausgewertet. Für die Auswertung spielen besonders die Testfahrten eine wichtige Rolle, da diese Auskunft über eine erfolgreiche oder nicht erfolgreiche Umsetzung der Software, als auch über die Ansteuerung der Antriebseinheit, sowie die Leistungsberechnung geben.

Es wird mit den Bewertungskriterien für die Testfahrten begonnen.

7.1 Bewertungskriterien für Testfahrten

Im Folgenden werden die Kriterien zur allgemeinen Bewertung festgelegt und erklärt. Die Bewertungskriterien für die Testfahrten orientieren sich maßgeblich an den Anforderungen aus dem Kapitel 4, insbesondere an den *Allgemeinen Anforderungen* und den *Anforderungen an die Antriebseinheit*. Für die folgende Bewertung ist es wichtig zu erwähnen, dass das Fahrverhalten nicht nur von dem Antriebssystem, sondern weitgehend auch vom Verhalten des Benutzers abhängig ist.

Getestet wurde das Antriebssystem des elektronischen Blindenhunds mit verschiedenen Benutzern und Fahrmanövern. Diese Testfahrten sind anhand des Fahrverhaltens und der Bahnführung bzw. Manövrierfähigkeit des EGDs zu beurteilen. Das Fahrverhalten umfasst dabei die Beschleunigung, das Abbremsen sowie die Geräuschentwicklung der Antriebseinheit. Die Bahnführung bzw. Manövrierfähigkeit, umfassen dagegen das Verhalten des EGDs während einer Kurvenfahrt. Auch ist zu bewerten, wie sich unterschiedliche Fahrmanöver auf den Benutzer auswirken und ob dieser dieses als benutzerfreundlich empfindet.

7.2 Bewertung der Testfahrten

Die Bewertung der Testfahrten mit verschiedenen Fahrmanövern sowie Benutzern wurden wie folgt umgesetzt:

Positive Ergebnisse des Fahrverhaltens

- Der neue Antrieb ist ausreichend stark dimensioniert. Steigungen und Zugkräfte des Benutzers können auch bei höheren Geschwindigkeiten überwunden werden.
- Geräuschpegel deutlich niedriger, im Vergleich zum Vorgängermodell.
- Beschleunigungsverhalten während des Anfahrens, sowie während des Abbremsens konnte verbessert und benutzerfreundlich gestaltet werden.
- Verbesserung der Belastbarkeit des Antriebssystem, da die neuen Motoren höhere Lasten aushalten können.
- Die Stromaufnahme konnte ebenfalls sehr niedrig gehalten werden, so dass mit dem neuen Antriebssystem eine längere Zeit gefahren werden kann.

Negative Ergebnisse des Fahrverhaltens

- Unpassende Geschwindigkeitsvorgabe des Antriebssystems bei Situationen, in denen der Benutzer die Geschwindigkeit reduziert.
- Vibration und Resonanzen während des Beschleunigens

Positive Ergebnisse der Bahnführung - Manövrierfähigkeit

- Kurvenfahrt mit Innenradius $r_i > 3.6 m$
 - Auslegung eines passenden Differentials mithilfe der Software war erfolgreich
 - Es wird ein konstanter Kreisbogenabschnitt gefahren
 - Kurvenfahrt ist Benutzerfreundlich

- Unter den genannten Testbedingungen (vgl. Kapitel 6.2.4), ist das Antriebssystem in der Lage den Kurs, auch ohne Regelungseinheit beizubehalten.

Negative Ergebnisse der Bahnführung - Manövrierfähigkeit

- Kurvenfahrt mit Innenradius $r_i \leq 3.6 \text{ m}$
 - Auslegung des Differentials muss ggf. optimiert werden
 - Kreisbogen kann nicht gefahren werden, da der Hebel durch den Stab des Benutzers zu groß ist
 - Schnelle hintereinander folgende Manöver konnten ebenfalls nicht vom Antriebssystem umgesetzt werden

Bewertung: Der Antrieb erfüllt die Leistungsanforderungen und ist deutlich leiser als das Vorgängermodell. Auch die Anforderungen für ein passendes Fahrverhalten während des Anfahrens und Abbremsens werden erfüllt. Ebenfalls ist die Stromaufnahme geringer und der Antrieb kann höheren Belastungen standhalten. Es gibt jedoch Probleme durch die feste Geschwindigkeitsvorgabe des Motors, in Situationen bei denen der Benutzer seine Geschwindigkeit stark erhöht oder reduziert. Auch ist die Motoreinheit beim beschleunigen lauter, im Vergleich zu einer Situation in der eine konstante Geschwindigkeit gefahren wird. Da die Manövrierfähigkeit und die Kurvenfahrten eine wichtige Rolle in der Führung eines Sehbehinderten spielen und aktuell noch deutliche Schwächen aufweisen, ist dies negativ für die Bahnführung zu bewerten.

7.3 Zusammenfassung der Ergebnisse

Die Testfahrten haben verdeutlicht, inwiefern der elektronische Blindenhund mit dem neuen Antriebssystem, die Anforderungen für die Bahnführung umsetzen kann. Die Ergebnisse wurden im vorigen Kapitel 7.2 *Bewertung der Testfahrten* ausgewertet und dargestellt.

Im Folgenden werden die erfolgreich und nicht erfolgreich umgesetzten Ergebnisse zur Software aufgelistet.

7.3.1 Erfolgreich umgesetzte Anforderungen

- Konzept für eine Bahnführung und eine Fortbewegungsstrategie wurden entwickelt und auf den MC implementiert
 - Software und Antriebssteuerung wurden erfolgreich auf dem TIVA implementiert
 - Software und Antriebssteuerung waren in der Lage den Motor auf gewisse Geschwindigkeiten zu beschleunigen und den Treiber des Motors in Teilen zu konfigurieren
 - Software beinhaltet Funktionen für verschiedene Fahrmanöver
 - Software bietet einfache Umstellung von Parametern für die Änderung des Fahrverhaltens
 - Software wurde an passender Stelle in das Gesamtprojekts (*watchdog*) des EGD integriert

7.3.2 Nicht erfolgreich umgesetzte Anforderungen

- Die Antriebssteuerung bietet nicht die Möglichkeit über eine UART Verbindung die internen Motorparameter des Drivers zu verändern.
- Es wurde keine Regelung für eine Kursabweichung integriert

7.4 Verbesserungsvorschläge

Aufgrund von zeitlichen Engpässen, konnten nicht alle Funktionen der neuen Software für die Antriebssteuerung und Bahnführung des elektronischen Blindenhunds getestet und validiert werden. Ebenso fehlen Testfahrten mit verschiedenen Straßenverhältnissen, wie z.B nasser Straße, oder starker Steigung. Demnach würde es zu neuen Erkenntnissen, sowie zur Verbesserung des Projekts führen, wenn diese Testfahrten durchgeführt und

die komplette Software getestet worden wäre. Auch die Algorithmen für das Fahrverhalten können optimiert werden, indem den Funktionen für die Antriebssteuerung mehr Informationen zu den Hindernissen und Strecken als Parameter übergeben werden. Dies erfordert allerdings eine Umstrukturierung der Schnittstelle für das Antriebssystem. Ratsam ist auch die Implementierung eines Kompassensors und eines digitalen Reglers, für die Verbesserung der Bahnführung. Des weiteren ist es empfehlenswert bei praktischen Projekten, wie auch bei dieser Arbeit, möglichst früh einen Teststand für eine praktische Umsetzung und Prüfung der Systemfunktionalität aufzubauen. In dieser Arbeit wurde bereits früh in der Anfangsphase ein Teststand aufgebaut, jedoch war es mit der ersten Version noch nicht möglich wichtig Funktionen wie das Fahrverhalten des EGDs zu prüfen. Ebenfalls hätte man auf zwei weitere Motoren für die Antriebssteuerung setzen können, so dass bei einem Fehlverhalten oder einem unerwünschten Defekt, unmittelbar weitergearbeitet werden kann.

8 Fazit und Ausblick

Ziel dieser Arbeit war die Konzeptionierung und Entwicklung einer neuen Software für die Antriebssteuerung und Bahnführung eines elektronischen Blindenführhunds. Darunter fällt auch die Auswahl eines neuen Antriebs, inklusive der Leistungsberechnung und Auswahl einer Antriebsart. Zusätzlich sollte das neue System in das bereits bestehende Projekt des EGD einsatzbereit und funktionsfähig implementiert werden. Bezugnehmend auf die Auswahl des neuen Antriebs sowie der zugehörigen Leistungsberechnung und der Antriebsart kann formuliert werden, dass dies zu Teilen erfolgreich umgesetzt werden konnte. Der verwendete Schrittmotor (Trinamic PD57-2-1076) konnte die Anforderungen zur Geräuschkentwicklung nicht vollständig erfüllen, da in gewissen Geschwindigkeitsbereichen Resonanzen und Schwingungen aufgetreten sind, welche den Sehbehinderten in der Geräuschwahrnehmung beeinträchtigen würden (vgl. Kapitel 6.2.4). Die Leistungsberechnung des Motors war hingegen erfolgreich und kann auf weitere Projekte übertragen werden (vgl. Kapitel 5.2.1). Außerdem war die Entwicklung einer Antriebssteuerung weitgehend erfolgreich. Ferner konnte der Microcontroller mithilfe der neuen Software den Motortreiber konfigurieren. Exemplarisch sind hier die Drehrichtung und die Regulierung der Betriebsmodi zu nennen (vgl. Kapitel 5.3.2). Andererseits ergaben sich Schwierigkeiten bei der Einstellung einer UART Verbindung für die interne Motorkonfiguration zur Einstellung der Schrittauflösung oder auch der Strombegrenzung. Des Weiteren konnte die Implementierung eines digitalen Reglers für die Regelung der Kursabweichung mithilfe eines Kompassensors nicht vorgenommen werden. Bezüglich der Bahnführung konnte das Antriebssystem mit zwei Antriebsmotoren und Differentiallenkung die Anforderungen von weiten Kurvenradien $r_i \geq 3.6 \text{ m}$ erfüllen (vgl. Kapitel 6.2.4). Kleinere Kurven mit Innenradius $r_i < 3.6 \text{ m}$ konnten hingegen kaum oder gar nicht umgesetzt werden, weil der Hebel durch den Stab des elektronischen Blindenhunds das Lenkmanöver zu stark beeinflusst hat. Die Software für die Fortbewegungsstrategie und Bahnführung ist trotz der mangelnden Manövrierfähigkeit größtenteils funktionsfähig, konnte jedoch nicht vollständig validiert werden. Abschließend lässt sich formulieren, dass die Umsetzung der Konzeption eines neuen Antriebssystems zu einigen Teilen erfolgreich gewesen ist, jedoch

in der Manövrierfähigkeit viele Schwächen aufweist und daher für den Alltagsgebrauch noch nicht vollständig einsatzbereit ist.

Im Folgenden wird ein kurzer Ausblick gegeben und mögliche Verbesserungsvorschläge zur Weiterentwicklung des Antriebssystems aufgezeigt. Für die Weiterentwicklung des Antriebssystems empfiehlt es sich die aktuelle Software für die Bahnführung, insbesondere für die Kurvenmanöver, zu optimieren und vollständig zu validieren. Um einen Überblick über die Gesamtfunktionalität zu erhalten, sollte ebenfalls eine vollständige Messung der Testfahrten mit verschiedenen Straßenverhältnissen vorgenommen werden. Nachfolgend kann zusätzlich eine Regelung für die Kursabweichung integriert werden. Außerdem sollte auch geprüft werden, ob andere Motoren, wie z. B. die BLDC Motoren oder Hub-Motoren in diesem Anwendungsgebiet geräuschärmer sind und eine bessere Fahrdynamik ermöglichen. Darüber hinaus sollte über eine Feedbackfunktion nachgedacht werden, die die aktuelle Benutzergeschwindigkeit mit der des Blindenhundes abgleicht und somit das Laufgefühl mit dem EDG optimiert. Des Weiteren kann bei unzureichender Bahnführung eine alternative Antriebsart verwendet werden. Hier bietet sich der Einbau einer Lenksachse an. Zusammenfassend kann gesagt werden, dass die Entwicklung eines elektronischen Blindenhundes mit vielfältigen Komplikationen verbunden ist, welche teilweise in dieser Arbeit überwunden wurden. Allein in Deutschland sind ca. 350.000 Menschen sehbehindert und könnten durch eine marktfähige Version einer elektronischen Orientierungshilfe in ihrem Alltag unterstützt werden, welches die gesellschaftliche Relevanz des Themas aufzeigt. Die genannten Forschungslücken bieten somit erste Orientierungspunkte für die weitere Arbeit an diesen Herausforderungen. Eine zufriedenstellende Lösung ist wissenschaftlich und gesellschaftlich erstrebenswert.

Literaturverzeichnis

- [1] *QMOT QSH5718 MANUAL*. – URL https://www.trinamic.com/fileadmin/assets/Products/Motors_Documents/QSH5718_manual.pdf; abgerufen am 12. August 2021.
- [2] *Tiva™ TM4C1294NCPDT Microcontroller*, 2014. – URL <https://www.ti.com/lit/ds/symlink/tm4c1294ncpdt.pdf>; abgerufen am 9. August 2021.
- [3] *CCSTUDIO*, 2021. – URL <https://www.ti.com/tool/CCSTUDIO>; abgerufen am 15. August 2021.
- [4] *PD57/60-1076 TMCL™ Firmware Manual*, 2021. – URL https://www.trinamic.com/fileadmin/assets/Products/Drives_Documents/PD-1076-TMCL_firmware_manual_fw1.02_rev1.02.pdf; abgerufen am 5. August 2021.
- [5] *PD57/60-x-1076 Hardware Manual*, 2021. – URL https://www.trinamic.com/fileadmin/assets/Products/Drives_Documents/PD-1076_hardware_manual_hw1.10_rev1.05.pdf; abgerufen am 5. August 2021.
- [6] *TMCM™-1076 Hardware Manual Hardware Version V1.10*, 2021. – URL https://www.trinamic.com/fileadmin/assets/Products/Modules_Documents/TMCM-1076_hardware_manual_hw1.10_rev1.02.pdf; abgerufen am 5. August 2021.
- [7] BINDER, Andreas: *Elektrische Maschinen und Antriebe*. Springer Vieweg, 2014. – ISBN 978-3-540-71850-5
- [8] BRAUN, Jan: *Formelsammlung*. Website. 2012. – URL https://www.maxongroup.de/medias/sys_master/8819062800414.pdf?attachment=true; abgerufen am 29. Juli 2021.

- [9] BUNDESAMT, Statistisches: *Arten der Behinderung*. Website. 2019. – URL https://www.destatis.de/DE/Themen/Gesellschaft-Umwelt/Gesundheit/Behinderte-Menschen/Publikationen/Downloads-Behinderte-Menschen/schwerbehinderte-2130510199004.pdf?__blob=publicationFile; abgerufen am 4. Juli 2021.
- [10] CANE, Mike: *Oren Returns With A DIY Belt-Drive Electric Skateboard*. Website. 2018. – URL <https://atomicsupermen.wordpress.com/2018/10/17/oren-returns-with-a-diy-belt-drive-electric-skateboard/>; abgerufen am 18. Juli 2021.
- [11] CHRISTIAN T. ERBSMEHL, Bettina B. und: *Bewegungsverhalten von Fußgängern im Straßenverkehr*. FAT Forschungsvereinigung Automobiltechnik e.V, 2014
- [12] COLVIN, James: *How to Use a Stepper Motor*. Website. 2021. – URL <https://digilent.com/blog/how-to-use-a-stepper-motor/>; abgerufen am 14. Juli 2021.
- [13] COWARD, Cameron: *Using Mini Cheetah-Style Brushless DC Motors for CNC Applications*. Website. 2019. – URL <https://www.hackster.io/news/using-mini-cheetah-style-brushless-dc-motors-for-cnc-applications-013598374c26>; abgerufen am 18. Juli 2021.
- [14] DICHLER, Werner: *Schrittmotor*. Website. 2011. – URL https://werner.dichler.at/?q=system/files/attachment/download_docs_schrittmotor.pdf; abgerufen am 14. Juli 2021.
- [15] ELECTRONICS, Digi K.: *PD57-1-1076*. Website. 2021. – URL <https://www.digikey.de/product-detail/de/trinamic-motion-control-gmbh/PD57-1-1076/1460-1324-ND/9607709>; abgerufen am 5. August 2021.
- [16] FAULHABER, Fritz: *Technische Informationen*. Website. 2021. – URL https://www.faulhaber.com/fileadmin/Import/Media/DE_TECHNICAL_INFORMATION.pdf; abgerufen am 2. August 2021.
- [17] GMBH, Kompendium INFOTIP S.: *ANALOGUE SCHALTUNGSTECHNIK SERVO*. Website. 2017. – URL <https://kompendium.infotip.de/servos.html>; abgerufen am 14. Juli 2021.

- [18] GUNTRAM KOMMERELL, Michael B. und: *Sehschärfebestimmung nach Europäischer Norm Wissenschaftliche Grundlagen und Möglichkeiten der automatischen Messung*. Website. 2002. – URL <https://www.uniklinik-freiburg.de/augenklinik/fpat/sehschaerfe.html>; abgerufen am 5. Juli 2021.
- [19] HELI-PLANET: *E- Motoren und Flugregler für Modellhelikopter*. Website. – URL https://www.heli-planet.com/motor_und_regler.html; abgerufen am 12. Juli 2021.
- [20] IDSA, Industrial Designers Society Of A.: *MYGO the first generation of a dynamic and interactive guidance in the urban environment*. Website. 2007. – URL <https://www.idsa.org/awards/idea/graduate-student-designs/mygo-first-generation-dynamic-and-interactive-guidance-urban>; abgerufen am 18. Juli 2021.
- [21] INSTITUT, Robert K.: *GBE-Themenheft Blindheit und Sehbehinderung*. Website. 2017. – URL https://www.rki.de/DE/Content/Gesundheitsmonitoring/Gesundheitsberichterstattung/GBEDownloadsT/blindheit.pdf?__blob=publicationFile#:~:text=Weltweit%20sch%C3%A4tzte%20die%20WHO%20f%C3%BCr,Visus%20%3C0%2C05; abgerufen am 4. Juli 2021.
- [22] ISERMANN, Rolf: *Fahrdynamik-Regelung*. Springer Vieweg, 2006. – ISBN 978-3-8348-9049-8
- [23] JASON, Kelly: *Wie lässt sich die Kommutierung eines BLDC-Motors am effektivsten durchführen?* Website. 2017. – URL <https://www.digikey.de/de/articles/what-is-the-most-effective-way-to-commutate-a-bldc-motor>; abgerufen am 15. Juli 2021.
- [24] JONER, Emrik: *How Brushless Motors Work and How to Test Them*. Website. 2020. – URL <https://www.tytorobotics.com/blogs/articles/how-brushless-motors-work>; abgerufen am 12. Juli 2021.
- [25] LAVERS, Nick: *Mobility device for the blind works like a handheld robotic guide dog*. Website. 2020. – URL <https://newatlas.com/robotics/theia-blind-handheld-robotic-guide-dog/>; abgerufen am 15. Juli 2021.

- [26] LEIFIPHYSIK: *Elektromotor*. Website. – URL <https://www.leifiphysik.de/elektrizitaetslehre/kraft-auf-stromleiter-e-motor/grundwissen/elektromotor>; abgerufen am 12. Juli 2021.
- [27] MEINSTEIN: *Die schiefe Ebene: Gewichtskraft, Hangkraft und Normalkraft*. Website. – URL https://www.uni-muenster.de/imperia/md/content/fachbereich_physik/didaktik_physik/publikationen/102_springen_gehen_laufen_2.pdf; abgerufen am 7. Juli 2021.
- [28] NICULA, David: *What's the Difference Between a Hub Drive and a Belt Drive Electric Skateboard?* Website. – URL <https://electricboardco.com/blogs/news/what-s-the-difference-between-a-hub-drive-and-a-belt-drive-motor-in-electric-skateboards>; abgerufen am 18. Juli 2021.
- [29] PRESSE, Bosch: *Jugend forscht Landeswettbewerb 2016 Baden-Württemberg*. Website. 2016. – URL <https://www.bosch-presse.de/pressportal/de/de/jugend-forscht-landeswettbewerb-2016-baden-wuerttemberg-45652.html>; abgerufen am 18. Juli 2021.
- [30] PÖSCHL, Sascha: *Rampensteigung - Erklärung*. Website. – URL https://www.helpster.de/rampensteigung-erklaerung_196938; abgerufen am 20. Juli 2021.
- [31] ROE, Dan: *E-Bike Motors, Explained*. Website. 2019. – URL <https://www.bicycling.com/bikes-gear/a25836248/electric-bike-motor/>; abgerufen am 18. Juli 2021.
- [32] SCHAAD, St: *Schrittmotor kurz erklärt*. Website. – URL https://wiki.bu.ost.ch/infoportal/_media/hardware/syssp/bauteile/schrittmotor_kurz_erklaert_d.pdf; abgerufen am 28. Juli 2021.
- [33] SCHILLMEIER, Monika Dr. med. vet.: *Der Blindenhund – Was muss er können? Was kostet er? Welche Rasse eignet sich?* Website. – URL <https://myhappypet.de/ratgeber/hund-und-katze/der-blindenhund-was-muss-er-koennen-was-kostet-er-welche-rasse-eignet-sich>; abgerufen am 6. Juli 2021.
- [34] SCHLICHTING, Bernd Rodewald Hans J.: *Springen, Gehen, Laufen*. Website. 1988. – URL <https://www.uni-muenster.de/imperia/md/>

- [content/fachbereich_physik/didaktik_physik/publikationen/102_springen_gehen_laufen_2.pdf](#); abgerufen am 14. August 2021.
- [35] SCHNETTLER, Achim Kampker Dirk Vallée A.: *Elektromobilität*. Springer Vieweg, 2018. – ISBN 978-3-662-53137-2
- [36] SEHBEHINDERTENVERBAND DBSV, Deutscher B. und: *Umfrage zu Lebenssituationen blinder Menschen*. Website. 2016. – URL <https://www.dbsv.org/umfrage-alltag.html>; abgerufen am 5. Juli 2021.
- [37] STATISTIK, Rehadat: *Arten der Behinderung*. Website. 2020. – URL <https://www.rehadat-statistik.de/statistiken/behinderung/behinderungsarten/uebersicht-aus-schwerbehindertenstatistik/>; abgerufen am 4. Juli 2021.
- [38] TREFZ, Henry: *Das ist die steilste Straße Deutschlands*. Website. 2016. – URL <https://www.thueringer-allgemeine.de/leben/land-und-leute/das-ist-die-steilste-strasse-deutschlands-id221898583.html>; abgerufen am 20. Juli 2021.

A Anhang

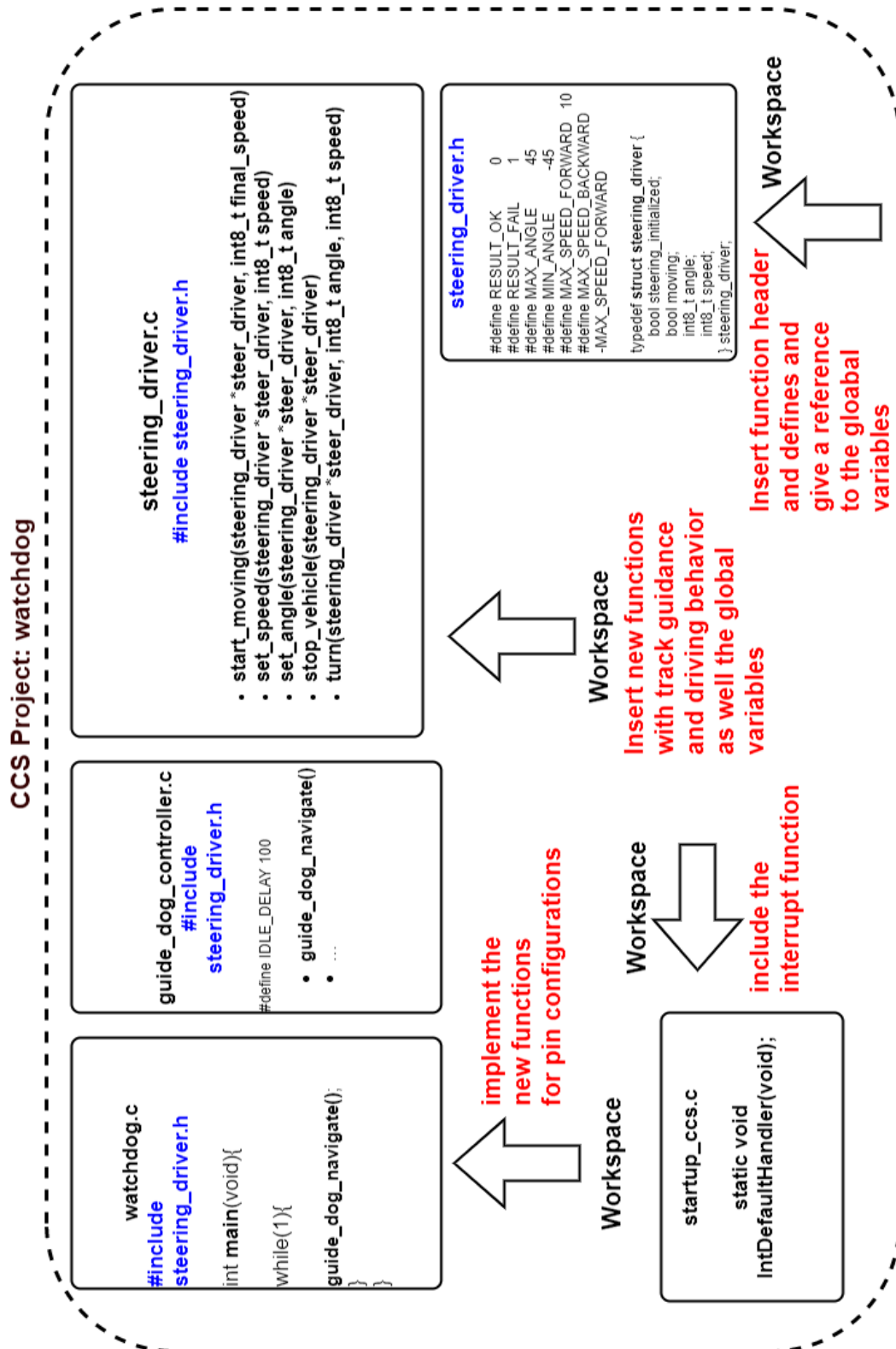
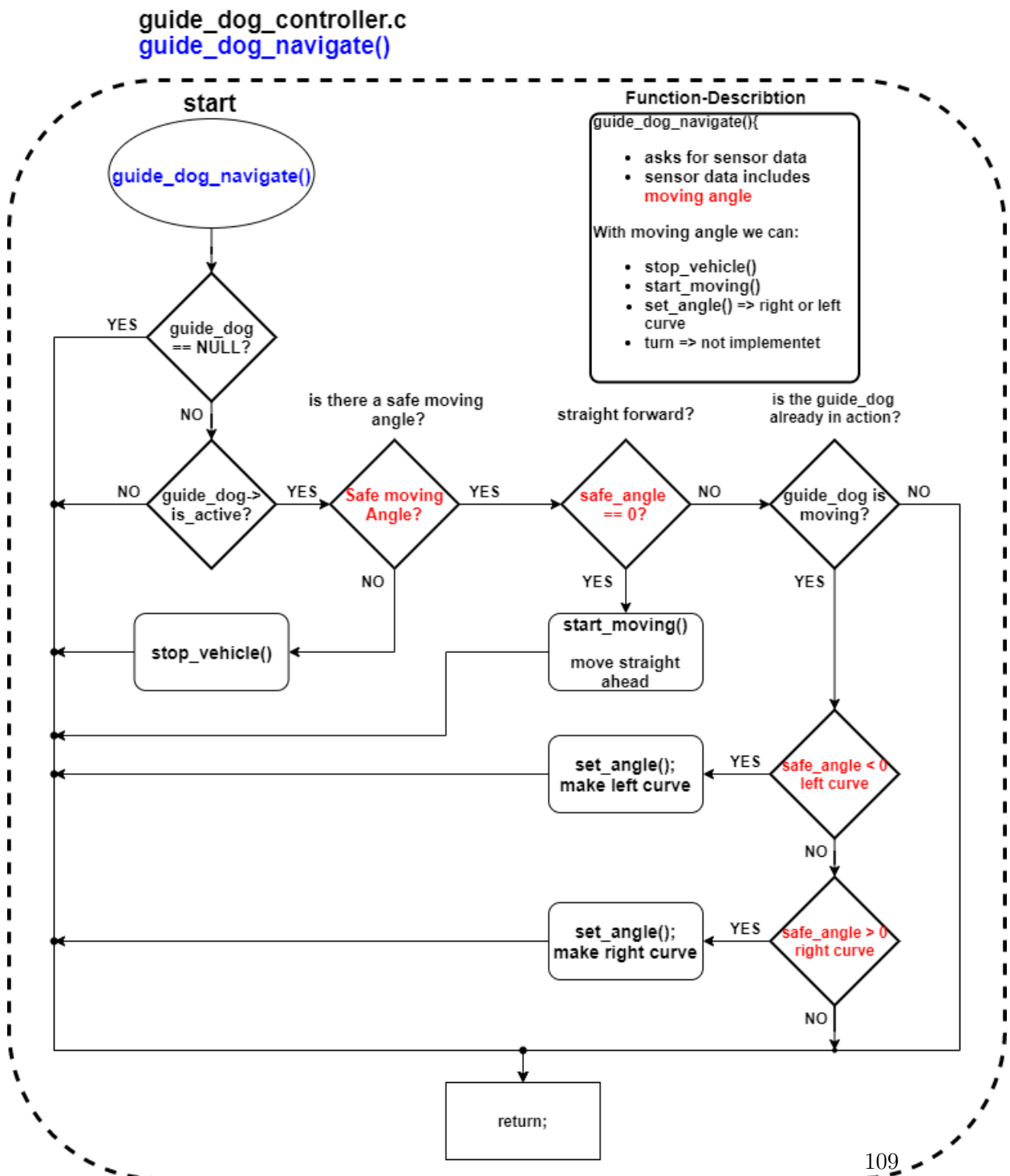


Abbildung A.1: CCS Projekt: watchdog, Übersicht des Arbeitsbereichs - Headerfiles und .c Dateien



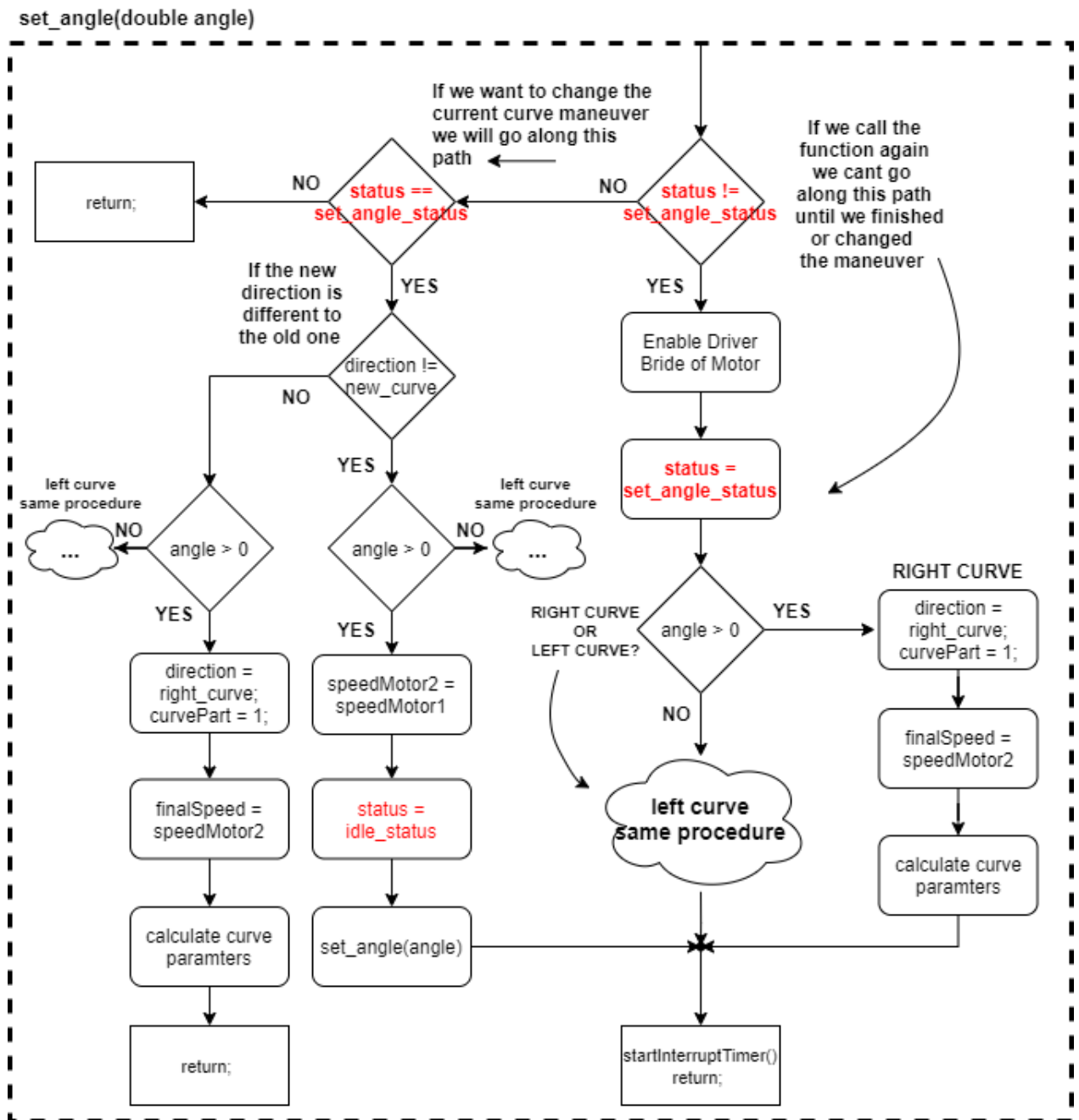


Abbildung A.3: Flowchart zur Übersicht vom Funktionsablauf set_angle() bei einer Rechtskurve

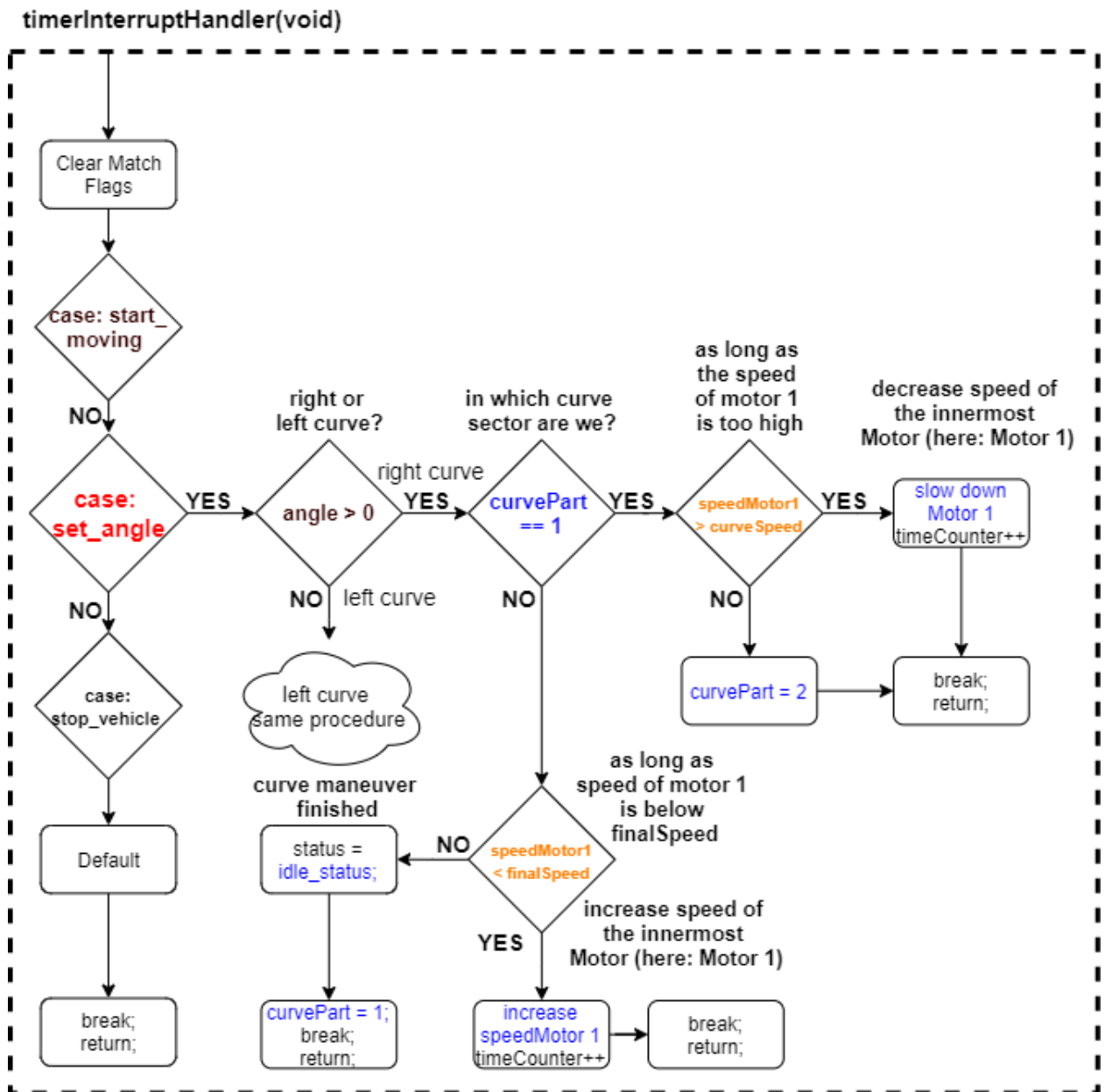


Abbildung A.4: Funktionsablauf von set_angle() im Interrupthandler (*timerInterruptHandler()*)

Motor / ESC Einkaufsoptionen											
Info	Lieferung muss innerhalb von 15 Wochen geschehen	Muss die Anforderungen der Leistungsberechnung erfüllen	17,50%	17,50%	10,00%	7,50%	7,50%	Preis	Dimension / Größe / Gewicht	2,50%	100,00%
Bewertung-Gewichtung	20,00%	17,50%	17,50%	17,50%	10,00%	7,50%	7,50%	7,50%	5,00%	2,50%	Summe / Bewertung
Kriterien / Anforderungen	Verfügbarkeit	Leistung/Drehmoment / RPM	Effizienz	Lautstärke	Dokumentation	Verarbeitung / Qualität	Verarbeitung / Qualität	Preis	Dimension / Größe / Gewicht	Einsatzbereich	Summe / Bewertung
Produktname / Motoren ESC Komplettsystem	0-10	0-10	0-10	0-10	0-10	0-10	0-10	0-10	0-10	0-10	0-10
Spalte1	Spalte2	Spalte3	Spalte4	Spalte5	Spalte6	Spalte7	Spalte8	Spalte9	Spalte10	Spalte11	
Turnigy	8	6	3	5	5	7	6	8	6	6	5,625
Hobbywing HW38020312 Xerun	9	5	3	5	6	5	5	7	6	6	5,675
Track ROAR	7	5	3	5	6	7	8	7	6	6	5,65
GoelRC	9	5	3	5	2	5	9	7	6	6	5,575

Abbildung A.5: BLDC Motor Auswahl - Übersicht Bewertung

Motor / ESC Einkaufsoptionen										
Info	Lieferung muss innerhalb von 1,5 Wochen erscheinen	Muss die Anforderungen der Leistungsrechnung erfüllen								
Bewertung-Gewichtung	20,00%	17,50%	17,50%	12,50%	10,00%	7,50%	7,50%	5,00%	2,50%	100,00%
Kriterien / Anforderungen	Verfügbarkeit	Leistung/Drehmoment / RPM	Effizienz	Lautstärke	Dokumentation	Verarbeitung / Qualität	Preis	Dimension / Größe / Gewicht	Einsatzbereich	Summe / Bewertung
Produktname / Schrittmotor mit Regler	0-10	0-10	0-10	0-10	0-10	0-10	0-10	0-10	0-10	0-10
Spalte1	Spalte2	Spalte3	Spalte4	Spalte5	Spalte6	Spalte7	Spalte8	Spalte9	Spalte10	Spalte11
Trinamic PD57-2-1076	9	8	8	9	9	8	6	8	5	8.2
PD4-ES91L42-E-65-7	7	9	8	6	9	9	4	4	5	7.325
EMMS-ST-57M-S-G2	8	8	7	6	8	8	6	6	5	7.25
NEMA23-01	9	6	7	6	7	8	8	6	5	7.15

Abbildung A.6: Schrittmotor Auswahl - Übersicht Bewertung

Erklärung zur selbstständigen Bearbeitung einer Abschlussarbeit

Hiermit versichere ich, dass ich die vorliegende Arbeit ohne fremde Hilfe selbständig verfasst und nur die angegebenen Hilfsmittel benutzt habe. Wörtlich oder dem Sinn nach aus anderen Werken entnommene Stellen sind unter Angabe der Quellen kenntlich gemacht.

Ort

Datum

Unterschrift im Original