A picture containing white

Description automatically generated

Projekt Mecanum

Ein Auto, das in jede Richtung fahren kann

Eine Projektarbeit von:

Vladimir Morozov

3iP Sek. Bäumlihof

3. Mai 2022

Begleitung von Herr Haris Fuchs

Inhaltsverzeichnis

[1. Vorwort 3](#_Toc102425289)

[2. Einleitung 3](#_Toc102425290)

[3. Hauptteil 4](#_Toc102425291)

[3.1. Die Vorbereitung 4](#_Toc102425292)

[3.2. Die Modellierung des Rades 4](#_Toc102425293)

[3.3. Das Programm 6](#_Toc102425294)

[3.4. Die Elektronik 10](#_Toc102425295)

[3.5. Die benützten Programme 11](#_Toc102425296)

[3.6. Der 3D-Drucker 12](#_Toc102425297)

[3.7. Das Mecanum-Rad 13](#_Toc102425298)

[4. Zusammenfassung und Reflexion 14](#_Toc102425299)

[5. Quellenverzeichnis 14](#_Toc102425300)

[5.1. Alle Komponente 15](#_Toc102425301)

[6. Anhang 15](#_Toc102425302)

[7. Verfasser und Quellennachweis 15](#_Toc102425303)

# Vorwort

Ich bedanke mich sehr bei meinem Betreuungslehrer Herr Fuchs und meinem Vater Oleksii Morozov. Beide haben mir sehr beim Projekt geholfen und mir auch sehr viele Ideen gebracht. Beim Herrn Fuchs bedanke ich mich am meisten für die Idee, wie die Mecanum-Räder modelliert werden sollten, und bei meinem Vater für die Realisierung des Projekts. Ohne diese beide Personen würde das Projekt nicht funktionieren oder nicht mal existieren. Indirekt bedanke ich mich auch bei Digi-Key für die Komponenten, Kabel und vieles mehr und auch bei Creality, die den 3D-Drucker Ender 3 v2 hergestellt haben, den ich benützt habe, um alle Modelle auszudrucken. Dieses Projekt hat ein halbes Jahr gedauert und war ziemlich anstrengend und mühsam, da es sehr viel mit Programmieren zu tun hat, aber auch die Modelle zu kreieren und auch testen. Insgesamt wurden mehr als 3 Tage (summiert) gebraucht, um alles auszudrucken, und mehr als 15 Stunden (summiert) für das Programmieren. Dieses Projekt war sehr interessant für mich, da es sehr viel mit Elektronik und Programmieren zu tun hatte.

# Einleitung

Ich habe während der zweiten Sekundarklasse schon die ganze Zeit überlegt, was ich so für ein Projekt machen sollte. Zuerst kam die Idee, einen Spinnenartigen Roboter zu kreieren, das sich sehr simpel nach vorne bewegt. Das war mir aber zu langweilig. Die nächste Idee wäre dann, einen 3D-Drucker selbst zu kreieren. Das wäre großartig, aber ich habe schon geahnt, dass es zu aufwändig wäre und zu schwierig für mich wäre, wenn ich noch nie wirklich ein Projekt gemacht habe. Dann stoss ich auf eine Idee. Da ich eh schon Videospiele spiele, denke ich, warum nicht selbst ein Videospiel zu kreieren. Dann bleib ich bei dieser Idee, bis wir angefangen haben zusammenzusammeln, was wir schon für Ideen hätten für unser Projekt. Jedoch während der Spezialwoche habe ich mich umentschieden. Mein Vater hat mir vorgeschlagen lieber etwas zu kreieren, das mit der Elektronik und Mechanik zu tun hat. Da hatte er auch recht. Ein Videospiel zu machen könnte man jederzeit erstellen, doch etwas zu kreieren, wo man etwas baut, dies kann man nicht jederzeit machen, da man nicht immer die Ressourcen dazu hat. Also musste ich kurz überlegen was ich machen sollte. Da kam etwas auf den Kopf. Ich habe einmal ein ferngesteuertes Auto gesehen, das in jede Richtung sich bewegen kann, ohne sich zu rotieren. Also habe ich nach dem Namen recherchiert, da ich früher den Namen nicht gewusst habe. Dieser Name trat dann auf: "Mecanum Wheel". Es inspirierte mich dann auch so ein Auto zu bauen, aber ohne fertiggemachte Räder, Körper oder auch Programm. Ich wollte alles selbst kreieren.

# Hauptteil

## Die Vorbereitung

Als ich das Projekt angefangen habe, habe ich zuerst eine Grobplanung gemacht, um zu wissen, was ich zuerst machen werde und was erst danach. Dies war schwierig nach Zeit einzuplanen, da man nie weiss, wie lange das Programmieren oder auch das Modellieren gehen würde. Das war das erste was gemacht wurde. Als ich die Grobplanung fertiggeschrieben habe, haben ich und Herr Fuchs angefangen zu überlegen, was ich für Materialien gebrauchen würde und wie die Räder aufgebaut sein könnten. Wir wussten schon das Grundprinzip des Aufbaus der Räder, das Einzige was wir machen mussten, war zu überlegen, wie wir das Rad so modellieren könnten, dass es nicht zu schwer werden würde. Ich habe auch geplant, dass das Auto über einen Microcontroller kontrolliert werden würde, das über Wi-Fi Daten übermitteln könnte, um zu befehlen, was es für Bewegungen ausführen sollte. All diese Aktionen auf dem Auto werden von einer Batterie an Bord angetrieben. Als dies dann besprochen wurde, geling ich an die Arbeit. Geplant war, dass ich jedes Mal, wenn ich mich mit dem Projekt beschäftigen würde, dass ich einige Zeit an dem Modellieren verbringe und einige Zeit beim Programmieren.

## Die Modellierung des Rades

A picture containing floor, shoes

Description automatically generatedA picture containing floor, shoes

Description automatically generatedIch habe angefangen den ersten Prototyp des Rades zu kreieren. Das Ziel war es, dass es rund genug war, um mehrere Räder auf ein Rad darauf zu stecken, damit es ohne zu viel Rütteln sich drehen könnte. Das erste Modell, als ich es ausdruckte, war ein totaler Misserfolg, da die Grösse des kleinen Rades zu klein war, und zu wenig Fläche hatte, um den Boden zu berühren. Das zweite jedoch war genau so wie es sein sollte. Das Rad hat so eine Fläche, dass es an jedem Punkt den Boden berührt, und die Halterung störte nicht. Unten sieht man die zwei unterschiedlichen ausgedruckten Räder:

Misserfolg Erfolg

A group of white objects on a wooden surface

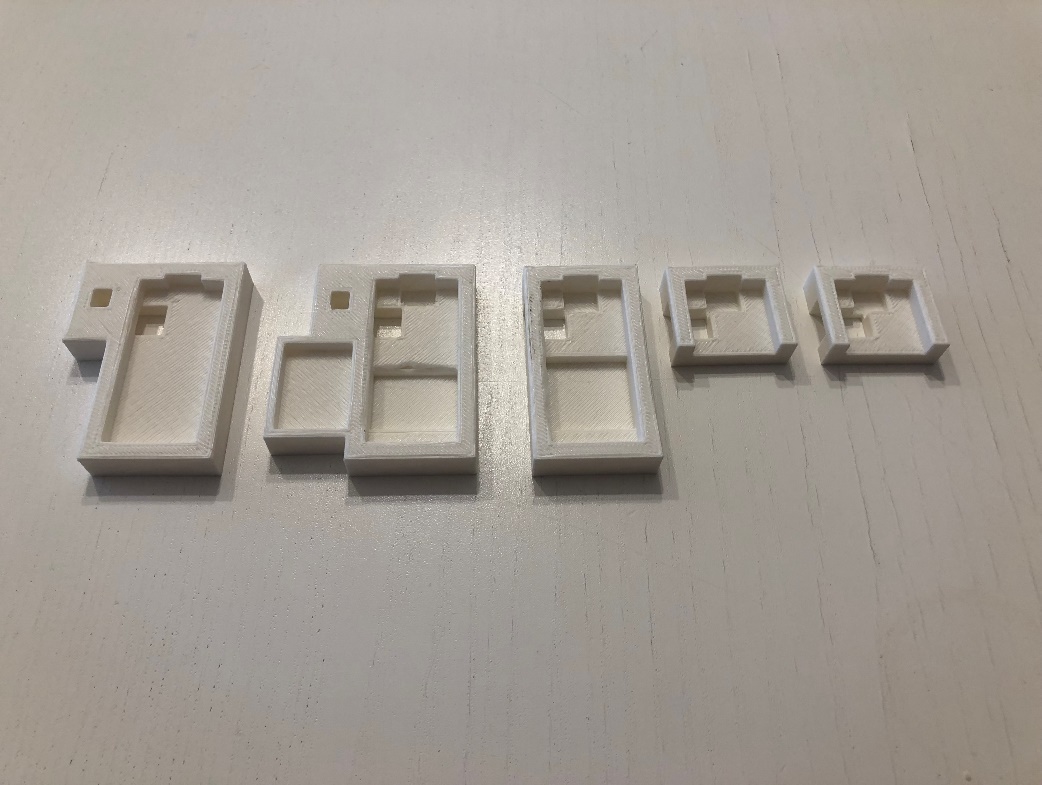
Description automatically generated with low confidenceAls ich das kleine Rad erfolgreich modelliert habe, habe ich alle kleinen Räder, Halterungen und das Gestell, um die kleinen Räder zu halten, ausgedruckt (insgesamt 8 kleine Räder und Halterungen und ein Gestell pro Rad, also insgesamt 32 kleine Räder und Halterungen, und vier Gestelle). Dies dauerte dann pro Rad-Set etwa 9 Stunden, um auf einem 3D-Drucker es auszudrucken. Als alles ausgedruckt wurde, habe ich mir zwei 4mm dicke Aluminium-Stäbe gekauft, um dann die kleinen Räder in den Halterungen fest zu heben. Ich habe die zwei Aluminium Stäbe in kleine Segmente abgesägt, diese dann in die Halterungen befestigen, und dann es mit Heissleim festkleben. Als alle kleinen Räder an den Halterungen befestigt wurden, klebte ich diese Halterungen auch mit Heisskleber an das Gestell. Somit hatte ich vier fertige Räder.

Nachdem alle Räder ausgedruckt wurden, ging es an den Körper weiter. Ich musste mir zuerst überlegen, wie die Motore gehalten werden müssten. Also habe ich mehrere Modelle kreiert und auch ausprobiert. Ein Modell, wo man den Motor einschiebt und eins wo man den Motor von oben einfügt und dann mit einem Deckel zudrückt. Meine erste Idee war das mit dem Einschieben, jedoch wäre es dann nicht zu fest am Halten. Dann habe ich mich auf die Idee mit dem Deckel entschieden. Einmal hat es nicht funktioniert, da es noch Teile Motor gab, die verhinderten den Motor völlig reinzukriegen. Also gab es dann zwei ausgedruckte Modelle, von dessen nur eins passend war. Diese drei ausgedruckten Modelle und den Deckel sieht man im unten angehängten Bild.

A picture containing wall, white, toilet

Description automatically generated

Nachdem als die Motor-Halterung aus dem Weg war, musste ich die richtigen Massen für die Komponenten wissen. Also habe ich auch nochmals mehrere Modelle zum Testen ausgedruckt und somit alles richtig abgemessen. Ich brauchte aber nicht wie früher ein paar Versuche, sondern ganze fünf Versuche. Jedes Mal passte etwas nicht und schliesslich hat es gepasst. Diese Versuche kann man im unten vorhandenen Bild sehen.



Nachdem alles richtig dann nachgemessen wurde, war der Körper fertig. Also habe ich es zum Drucken gesetzt, das dann 22 Stunden lang druckte. Als der 3D-Drucker es fertig ausgedruckt hatte, habe ich noch kontrolliert, ob alles passte. Und zum Glück passte alles. Wenn sogar nur etwas nicht passen würde, müsste ich alles wieder von Anfang aus ausdrucken.

## Das Programm

Das Programm war der wichtigste Teil des Projekts. Ohne des Programmes würde das Auto nicht mal wirklich fahren, gar funktionieren. Das Programm besteht aus zwei Teilen: Das Skript für den Microcontroller und einer Website. Das Skript für den Mikrokontroller kontrolliert alles Wichtiges, wie z.B. die Kontrollierung der Motoren, die Datenübermittlung und Wi-Fi Übertragung. Der Mikrokontroller ist in dem Fall das Gehirn des Autos. Und das stimmt auch. Dieser Mikrokontroller (Adafruit HUZZAH32 Feather Board) ist wie ein kleiner Computer, ohne die wichtigsten Sachen. Es kann Daten überarbeiten und auch viele verschiedene Operationen kontrollieren. Ein Bild davon sieht man unten.

A close-up of a circuit board

Description automatically generated with medium confidence

Das Skript besteht aus sechs Teilen: Die Pin-Konfigurierung (Zu welchem Kontakt was verbunden ist) und die Wi-Fi Konfigurierung, überprüfen, ob neue Daten für die Kontrollierung für die Motore empfangen worden sind, die Überarbeitung der empfangenen Datei, Daten für die Website and das nutzende Gerät schicken und den entsprechenden Dateien die Motoren anschalten bzw. ausschalten.

Text

Description automatically generatedDer wichtigste Teil ist die Überarbeitung der Datei, und entsprechend zu den empfangenen Dateien die Motore anschalten bzw. ausschalten. Diese Zwei Teile sieht man im unten vorhandenen Bild

Graphical user interface, text

Description automatically generated with medium confidence

Graphical user interface, text

Description automatically generated with medium confidenceSogar wenn man hier jetzt nichts versteht, kann man es ganz kurz erklären.

Dieses Abteil überprüft, ob es Daten empfangen hat. Wenn Daten empfangen worden sind, überarbeitet es diese Daten in Zustände (States). Es werden Daten übertragen in einem "XXXX" Format, wie z.B. "0101", "0221" oder sogar "0000". In diesem Abteil überprüft es die erste Stelle in diesem "XXXX" Format. In diesem Fall, wenn wir eine Datei mit "8521" empfangen würden, würde es die erste Zahl abchecken, in diesem Fall "8". Nach Zahl dieser Stelle, stellt es einen Motor Zustand zu aus/vorwärts/rückwärts. Dies macht es für jede dieser vier Zahlen und stellt für jede Zahl einen Zustand. Nachdem alles überarbeitet wurde, aktiviert es eine Funktion, die alle diese Zustände übernimmt. Dies wird dann im nächsten Abteil klar.

Text

Description automatically generated

Dieses Abteil übernimmt diese Zustände und schaltet zu jedem Zustand entsprechend einen Motor an oder aus. Bevor es aber entscheidet, welcher Motor angehen soll, und welcher nicht, schaltet es zuerst jeden Motor aus, damit keine Probleme auftauchen würden. All dies ist in der loop() Funktion. Die loop() Funktion ist einfach dazu dar, damit sich etwas ständig wiederholen kann. In diesem Fall, wenn sich ein Zustand gewechselt hat, wechselt sich dazu entsprechend der Zustand der Motoren.

Diagram

Description automatically generatedDann gibt es noch die Webseite, über die alle Dateien für die Bewegung verschickt werden. Diese Webseite besteht aus zwei Dateien: Dem HTML (index.html) und dem Programm des HTMLs (script.js (JavaScript)). Diese Webseite wird zum Gerät geschickt, das über Wi-Fi zum Mikrokontroller verbunden ist. Vorhanden sind der "index.html", der "script.js" aber auch die verschiedenen Bilder dazu. Diese Webseite sieht dann so aus.

## Die Elektronik

Es wurden Motoren, Mikrokontroller, Drivers und Batterien benützt. Das fertige Produkt hat normale DC Motoren. Jedoch als ich mit dem Projekt anfing, wollte ich Schrittmotoren benützen. Das großartige an Schrittmotoren ist, dass wie es schon in Namen steht, der Motor Schrittbewegung. Stattdessen dreht sich ein DC Motor, ohne zu stoppen, bis man es vom Strom abschliesst. Jedoch habe ich zwei Mal verschiedene Schrittmotoren gekauft, aber jeder Motor hatte seine Kontras. Die ersten Schrittmotoren waren stark genug, um zu drehen, jedoch waren sie zu langsam. Also habe ich dann neue gekauft die viel schneller waren, aber da waren die Motoren zu schwach. Also habe ich solche DC Motoren gekauft, dass ein Getriebe hat. Somit war der Motor nicht zu schnell, damit man es rechtzeitig stoppen konnte, ohne dass es weiterdreht.

A picture containing blue

Description automatically generated

Alle Motore werden von mehreren DRV8833 Drivern kontrolliert. Diese Driver dienen dazu, dass man bei den Motoren die Polarität wechseln könnte, und dass dieser somit in beide Richtungen sich drehen konnte. Dies war für das Auto sehr kritisch, da jedes Rad einzeln kontrolliert werden sollte.

A picture containing electronics, circuit

Description automatically generated

A picture containing text, battery

Description automatically generatedDas Auto wird von 5 NiMH AA Batterien angetrieben. Pro Batterie beinhaltet es 1.2V und 1900mAh, dass für so eine kleine Batterie ziemlich viel ist. Weil der Motor 6V verbrauchte, habe ich also 5 Batterien in serieller Verbindung zusammen Verbindung, damit es zu 6V ergibt. Als ich dann die Motore und den Mikrokontroller verbunden habe, habe ich sie parallel verbunden. Somit verliere ich nicht die Spannung, sondern den Strom.

## Die benützten Programme

Ich habe mehrere Programme benützt, um zu programmieren, modellieren oder auch Modelle auszudrucken. Die Namen wären: Arduino, Visual Studio Code, Shapr3D, Ultimaker Cura.

Arduino und Visual Studio Code habe ich benützt, um das Skript für den Mikrokontroller und die Website zu programmieren. Arduino ist eigentlich eine Sprache abgewandt von C++, die so umkonstruiert wurde, damit es geeignet wäre, um mit der Elektronik zu arbeiten. Visual Studio Code ist ein universales IDE, das für jede Programmiersprache geeignet ist.

Icon

Description automatically generated

Logo

Description automatically generated

Shapr3D und Ultimaker Cura wurden beide für Modelle aber auch das Ausdrucken von Modellen gebraucht. Ich habe Shapr3D benützt, weil es erstens gratis für Schüler ist, aber auch, weil es viel simpler ist als alle anderen Modellierungsprogramme. Ultimaker Cura war aber eher ein Tool, um eine Datei zu erstellen, dass die Schritte für den 3D-Drucker bezeichneten. Beide diese Programme haben mir unglaublich bei der Modellierung geholfen.

A picture containing text, sign

Description automatically generatedShape

Description automatically generated

## Der 3D-Drucker

Bei diesem Projekt habe ich den Creality Ender 3 v2 3D-Drucker benutzt. Es ist immer noch unglaublich, dass es heute solche Technik gibt, wo man selbst etwas zuhause als Ersatzteil oder etwas für den Alltag etwas schnell ausdrucken kann. Ohne diesen 3D-Drucker könnte ich nicht mal irgendwas machen, und anstatt für 30 Fr. 1’000g von Material kaufen kann und es bei mir selbst ausdrucken, anstatt dass ich mehrere von hunderten von Franken wegschmeissen würde und auch noch mehrere Monate warten würde. Deshalb fand ich es eine grosse Hilfe einen 3D-Drucker zu haben. Unten könnt ihr sehen, wie dieser 3D-Drucker aussieht. Wenn ihr euch fragt, wie viel so ein 3D-Drucker kostet: Es kostet zirka 250 bis 300 Franken.

Text, whiteboard

Description automatically generated

## Das Mecanum-Rad

A close-up of a steering wheel

Description automatically generated with medium confidenceLogo

Description automatically generated with medium confidenceDas Mecanum-Rad ist ein Rad, das erlaubt einem Fahrzeug omnidirektionell zu fahren, ohne mit einer Mechanischen Lenkung ausgestattet zu sein. Es wurde von dem schwedischen Erfinder Bengt Ilon erfunden, und wird deshalb auch als das Ilon-Rad bezeichnet. Es wurde von Bengt Ilon am 13. November 1972 als seine eigene Idee bezeichnet und ist deshalb auch der Patentinhaber dieses Rades. Im Gegensatz zum Allseitenrad (auch "Omni-Wheel") stehen die Rollen unter einem speziellen Winkel, und zwar unter einem 45˚ Winkel. Dies erlaubt es auch diagonal und auch seitlich zu fahren, ohne sich zu drehen. Am meisten werden solche Räder in engen Plätzen benützt, damit man keine Drehungen machen konnte, sondern gerade so fahren, dass es passt. Aber auch werden sie sehr viel in der Robotik benützt, um einfacher laste von einem Platz zu einem anderen zu bringen. Unten sieht man zwei Bilder: Ein konkretes Bild von so einem Rad und mit welchen Motor-Rotationen so ein Auto in welche Richtung sich bewegen kann.

# Zusammenfassung und Reflexion

Als ich das Auto fertiggemacht habe, war ich ein bisschen enttäuscht. Und zwar habe ich gedacht, dass es locker dann seitlich oder diagonal fahren könnte. Jedoch war das nicht der Fall. Es fand sich heraus, dass auch noch anging, wie die Reibung zwischen Rad und Boden sein sollte, wie die Räder in einem X sein sollten, dass der Abstand zwischen den Rollern zu gross war und vieles mehr. Aber da es mein erstes richtiges Projekt war, bin ich auch froh, dass ich so weit gekommen bin. All mein wissen über das Programmieren, aber auch ein bisschen über die Elektronik und Mechanik, kam endlich mal wirklich ins Spiel und ich konnte endlich mal selbst sehen, was ich so alles kann. Aber es ist nicht nur für mich, sondern an alle. Wenn ich nur sage: "Ja, kann programmieren und finde mich gut mit Technik aus.", das reicht nicht aus. Ich muss es zeigen können. Und das war der Beweis. Sogar wenn ich das Auto nicht so fertig geworden ist wie ich geplant hatte, habe ich immer noch sehr viel Großartiges und Interessantes gelernt.

Ich plane, wenn ich einmal Zeit haben werde, die Räder, aber auch allgemein vieles am Auto probieren zu verbessern. Und irgendwann mal, nach einer langen Zeit, wird das Auto endlich mal richtig funktionieren. Ich könnte sogar es probieren zu verkaufen, doch dies ist ein Gedanke für ein anderes mal.

# Quellenverzeichnis

Ich habe von diesem Tutorial gelernt, wie das Programm funktioniert, das auch alles Schritt nach Schritt erklärt: <https://shawnhymel.com/1882/how-to-create-a-web-server-with-websockets-using-an-esp32-in-arduino/>

In der unten gelinkten Dokumentation, das von mehreren professionellen Ingenieuren unterschrieben und genehmigt worden ist, wird alles über das Mecanum-Rad erklärt, und auch gezeigt, wie man selbst ein solches Rad modellieren kann: <https://commons.erau.edu/cgi/viewcontent.cgi?article=1268&context=edt>

Weitere Informationen über das Mecanum-Rad: <https://de.wikipedia.org/wiki/Mecanum-Rad>

## Alle Komponente

Mikrokontroller: <https://www.digikey.ch/en/products/detail/adafruit-industries-llc/3591/8119805>

Motor: <https://www.digikey.ch/en/products/detail/adafruit-industries-llc/3802/9342523?s=N4IgjCBcoLQBxVAYygMwIYBsDOBTANCAPZQDa4ArAEwIC6AvvYVWSAMxwAMLDQA>

Driver: <https://www.digikey.ch/de/products/detail/adafruit-industries-llc/3297/6419360>

Batterie: <https://www.digikey.ch/en/products/detail/panasonic-bsg/BK-200AAB9B/5067197>

Batteriehalter: <https://www.digikey.ch/en/products/detail/adafruit-industries-llc/3456/7320200>

# Anhang

Alles was ich selbst programmiert oder modelliert habe, kann man auf meiner Github-Page finden, und mit meiner Nachfrage auch kopieren, um es selbst zu bauen.

<https://github.com/Vinomakon/project_mecanum>

# Verfasser und Quellennachweis

Hiermit versischere ich Vladimir Morozov, dass ich alle Texte in meiner Dokumentation selbst verfasst habe.

Alle Texte, die nicht von mir stammen, habe ich mit einen Autorenhinweis gekennzeichnet. Sofern ich Gedanken, Ideen oder Textausschnitte aus anderen Texten oder Büchern übernommen habe, liegt eine Quellenangabe vor.

Ich weiss, dass, wenn ich gegen diese Vorgaben verstosse, meine Leistung abgewertet wird.