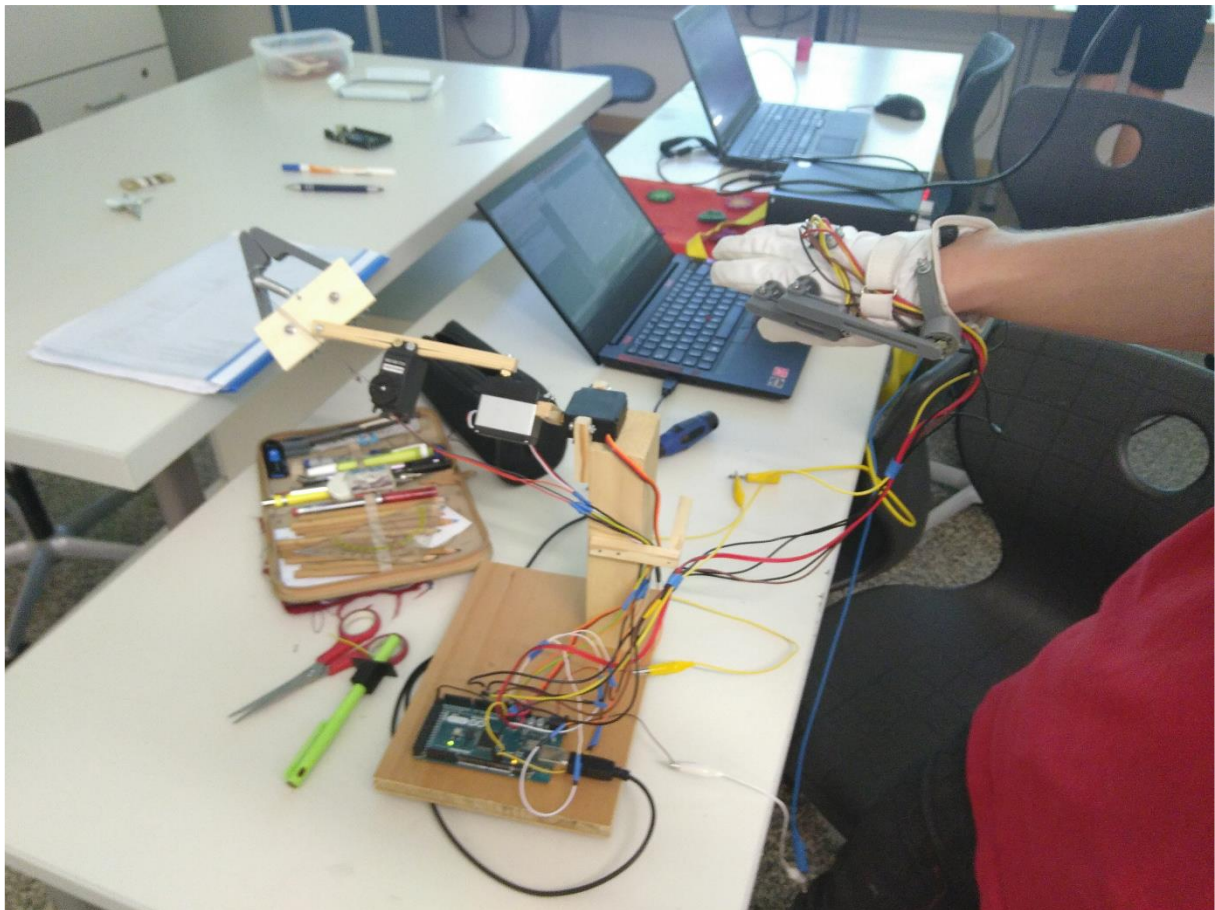


# Robotergreifer, der sich wie eine Hand bewegt

---

*Ein SIA Projekt von Fabian Exel am  
Gustav Stresemann Gymnasium Fellbach-Schmidlen.*



## Inhalt

Inhalt .....	2
Projektziele.....	3
Realisierung.....	3
Stückliste.....	5
Anwendungsbeispiele.....	5
Zusätzliche Verbesserungen, Erweiterungen kann man machen: .....	5
Schaltpläne .....	6
Aufbau verschiedener Bauteile / Komponenten .....	7
1. Servo .....	7
2. Beschleunigungssensor.....	9
3. Zahnräder .....	10
4. Greifer .....	12
5. Potentiometer .....	13
6. Sonstiges .....	13
Programmcode: für Arduino.....	14
Literaturverzeichnis.....	17
Abbildungsverzeichnis .....	17

## Projektziele

### Projektziel:

Bau eines Roboterarms, dessen Steuerung durch die menschlichen Hand-Bewegungen mittels eines mit Sensoren versehenen Handschuhs erfolgt. Die Sensorsignale werden von einem Mikrocomputer (Arduino) verarbeitet, der damit die Stellmotoren des Roboterarmes steuert. Wenn möglich soll der Roboterarm gezielt kleine Gegenstände bewegen und heben können.

### Teilziele:

- **Roboterarm** mit mindestens 3 Motoren darstellen:  
Konzept inkl. Kinematik, Auslegung, Konstruktion (tw. 3D-CAD), Rohteilfertigung (u.a. 3D-Druck), mechanische Fertigung, Zusammenbau
- **Handschuh mit Sensoren**, der Bewegungen des Menschen erkennt:  
Konzept, Auslegung der Sensoren und el. Schaltung, Zusammenbau
- **Arduino-Programm** zum Auslesen der Handschuhbewegungen:  
I/O Steuerung der Sensorsignale, Datenverarbeitung, Algorithmus zur Bewegungssteuerung
- **Zusammenspiel Handschuh – Roboterarm:**  
I/O Steuerung der Stellmotoren, Kalibrierung, Greifersteuerung

## Realisierung

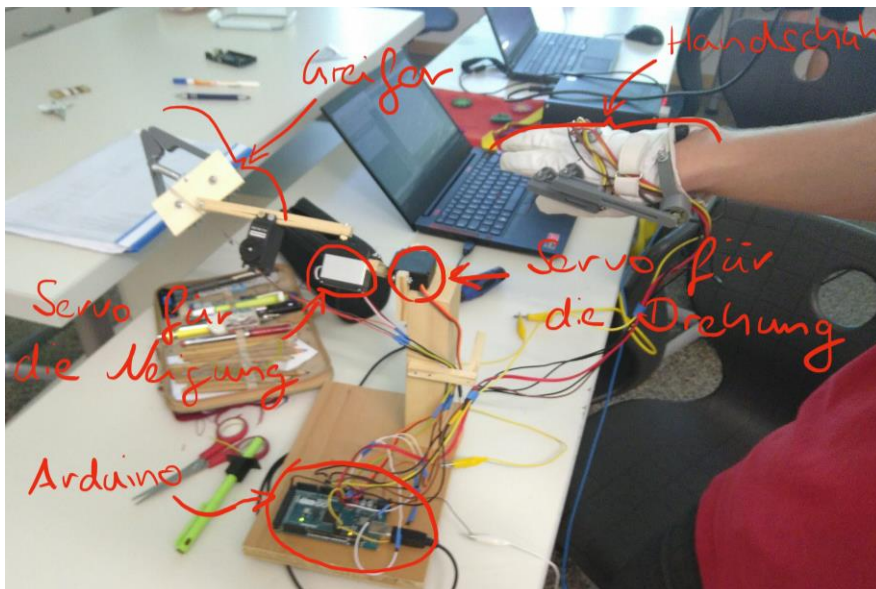


Abbildung 1: Der fertige Roboterarm mit Handschuh: Aufbau (oberes Bild)



Abbildung 2: Der Handschuh: Aufbau (rechtes Bild)

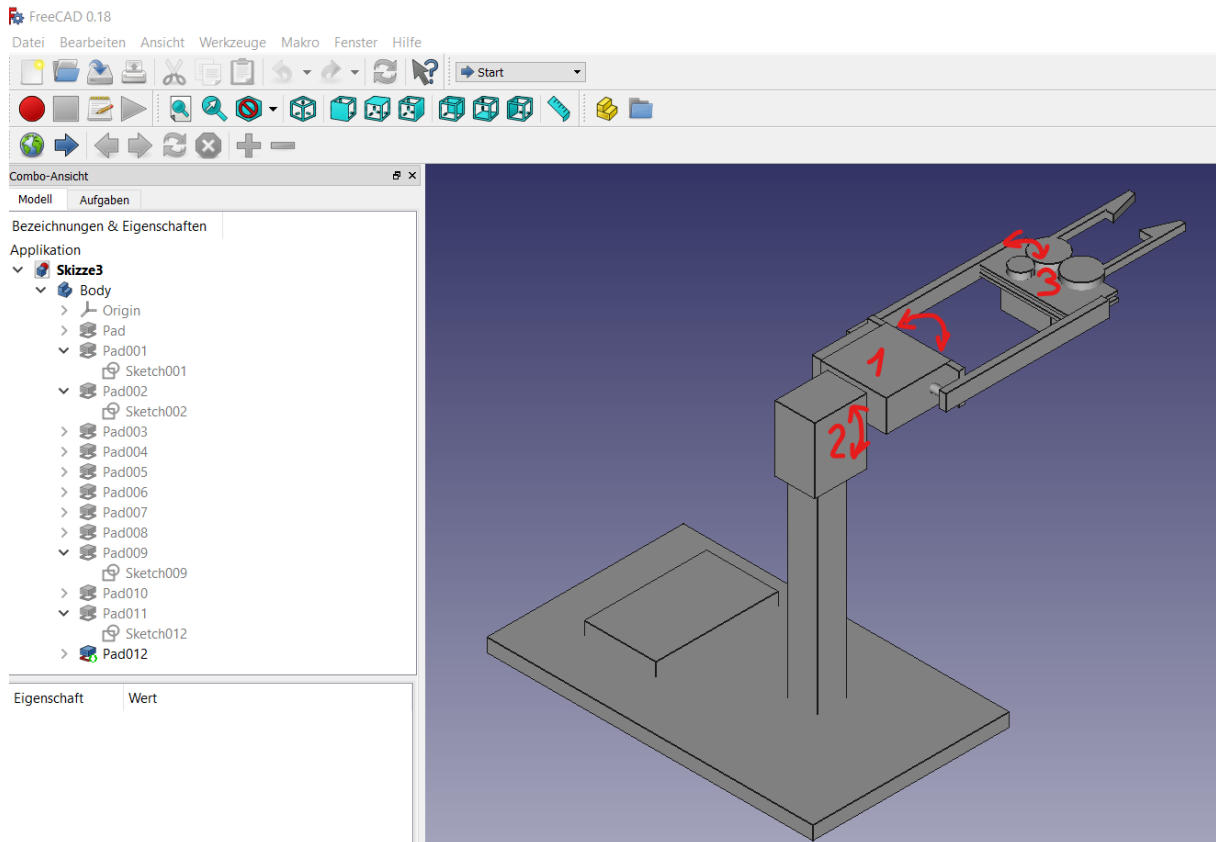
Alle Teilziele wurden erreicht:

Die Bauteile sind eigenhändig in 3D konstruiert, und am eigenen 3D-Drucker ausgedruckt worden.

### **Roboterarm:**

Der Greifer hat jetzt 3 durch Arduino angesteuerte Servomotoren mit gekoppelten Lagesensoren:

- Motor 1: Drehung des Greifarmes, Lagemessung (Winkel) mit Neigungssensor
- Motor 2: Längsbewegung nach unten und nach oben, Lagemessung (Winkel) mit Neigungssensor
- Motor 3: öffnet den Greifarm, Lagemessung (Winkel) mit Potentiometer



**Abbildung 3: 3D-Konstruktion des Roboterarms (Free-CAD)**

### **Handschuh:**

Integration von drei Sensoren in einen handelsüblichen Reithandschuh (besonders gute Passform):

- Potentiometer (Poti) 1: registriert Bewegung des Daumes, für die Steuerung des Greifers (Motor 1).
- Potentiometer (Poti) 2: nimmt Neigungsbewegungen der Hand auf, um eigentlich den entsprechenden Servo zu steuern, doch dieser wurde durch den dritten Sensor überflüssig, dem
- Beschleunigungssensor: drei Achsen: X, Y, Z;  
registriert die Drehung (Motor 2) und Neigung (Motor 3) der Hand

### **Arduino-Steuerungssoftware:**

Softwarecode zur Steuerung ist funktional dargestellt. Mögliche Fehler durch falsche Sensorsignale und damit falsche Steuersignale an den Neigungsservo sind im Gesamt-Testbetrieb aufgefallen. Korrekturprogramm für falsche Eingaben danach im Programmcode integriert worden (Siehe Kapitel...). Ev. Austausch des Servomotors könnte die Funktion verbessern.

## Stückliste

Teil	Menge	Preis einzelnd	Preis zusammen
Servo	3	15,00 €	45,00 €
Holzplatte	1	2,00 €	2,00 €
Ständer	1	1,00 €	1,00 €
Arduino Mega	1	60,00 €	60,00 €
Handschuh	1	5,00 €	5,00 €
Kabel		3,00 €	3,00 €
diverses anderes Holz		5,00 €	5,00 €
Metallplatten		7,00 €	7,00 €
Metallstangen		3,00 €	3,00 €
10kΩ Potentiometer	2	0,50 €	1,00 €
Neigungssensor	1	0,50 €	0,50 €
diverse Schrauben		1,00 €	1,00 €
Zahnräder (3d-Druck)	3	0,20 €	0,60 €
		<b>Summe:</b>	<b>134,10 €</b>

\* Alle Angaben sind Schätzwerte, da die Sachen überwiegend schon vorhanden sind

## Anwendungsbeispiele

Dieser Roboter ist vielfältig einsetzbar, Adaption an den jeweiligen Einsatzzweck ist notwendig. Mögliche Szenarien wären:

- Fachärzten, die in einem anderen Ort / Land sitzen und eine Reise für eine oder beide Seiten nicht möglich ist, hilft dieser Roboter, weil sie damit auch aus der Ferne / dem Ausland operieren können (Voraussetzung ist eine zusätzliche Internetverbindung)
- Als Hilfe beim Herumhantieren mit gefährlichen Stoffen, z.B.: Chemikalien, Bomben (entschärfen), Giftige Unfälle (z.B.: Tschernobyl Gau)
- Nutzung als modernes Kunstobjekt
- Hilfe für Leute, die nicht mehr aufstehen können, wollen (z.B.: bettlägerig im Altersheim)
- Möglichkeit für Lehrer, Dinge an die Tafel schreiben zu können, und trotzdem zur Klasse schauen zu können – man skaliert das Geschriebene einfach hoch
- In jetzigem Zustand Spiele, wo man z.B. Türme mit Hilfe von diesem Roboter stapeln muss.

Fazit: Alles, was man aus der Ferne steuern möchte mit einem Controller, der wie eine Menschenhand geformt ist.

### Zusätzliche Verbesserungen, Erweiterungen kann man machen:

- Kabellose Verbindung zwischen dem Handschuh und dem Roboter
- Anschluss ans Internet, um längere Strecken zu überbrücken
- Besseres Zusammenspiel Handschuh – Roboter
- Mehr Achsen, damit der Roboter sich besser bewegen, zum Beispiel unten eine Schiene, worüber sich der Roboter bewegen kann.



## Schaltpläne

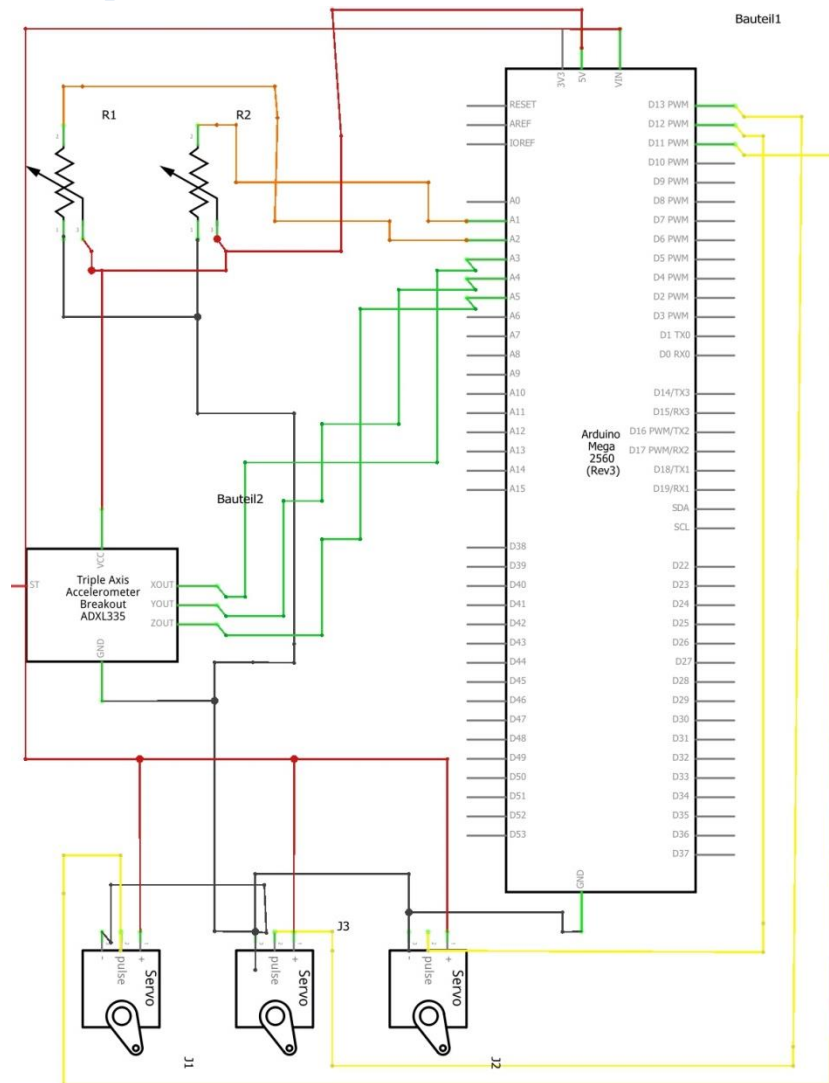


Abbildung 4: Schaltplan des Roboters

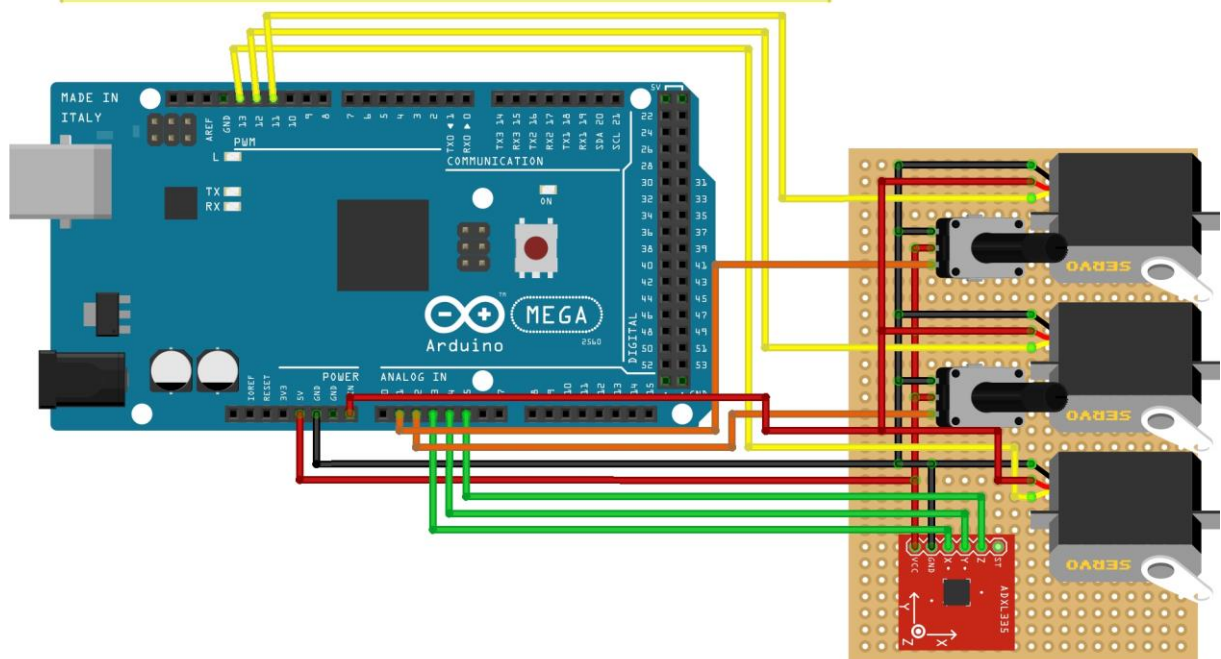


Abbildung 5: Schaltplan des Roboters als andere Darstellungsweise; die Farben der Kabel sind gleich geblieben

## Aufbau verschiedener Bauteile / Komponenten

### 1. Servo



Abbildung 6: Servo: Aufbau (oomlout, 13.06.2019), Bearbeitet

Mein Roboter besteht aus drei Servos, deshalb ist es wichtig zu wissen, wie so ein Servo überhaupt funktioniert. Zu allererst sind Servos generell Elektromotoren (im Modellbau vor allem Gleichstrommotoren) mit einem Rotorlagersensor. Servos kommen überall zum Einsatz, wo es um Regelungen bestimmter Zustände geht, d.h. für Präzision oder Geschwindigkeit verwendet werden.

Drei Ausführungsarten von Servos sind zu unterscheiden:

- Servomotor als reiner Hilfsmotor,
- Servoantrieb: Servomotor kombiniert mit dem Servoregler (Steuerungseinheit mit Software),
- kompletter Servo (teilweise mit Getriebe).

Positionsmessung gibt es zwei Möglichkeiten: Erfassung am Drehgeber, also dem Elektromotor.

- Servostellung direkt erfasst (die ganze Zeit wird gemessen – Absolutwertgeber)
- Servostellung indirekt erfasst (von der Anfangsposition an wird gerechnet – Inkrementalgeber)

**Regelkreis:** (oomlout, 13.06.2019)

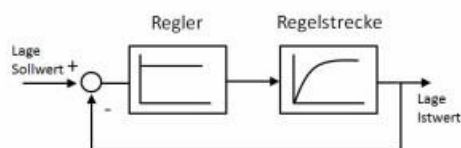
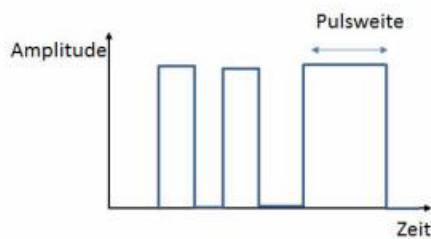


Abbildung 7: Regelkreis eines Servomotors

„In der Servoelektronik wird dann die aktuelle Rotorposition mit der Sollposition verglichen. Aus der Abweichung zwischen Soll und Ist-Wert wird im Anschluss ein Bewegungssignal (Stromsteuersignal) für den Servomotor berechnet. Dieser Ablauf des **Soll-Istwert Abgleiches** wiederholt sich dauerhaft und es

wird so lange ein neues Steuersignal generiert bis der aktuelle Wert im Toleranzrahmen des Sollwertes rangiert.“

### **Pulsweitenmodulation:**



**Abbildung 8: s-t-Diagramm von der PWM**

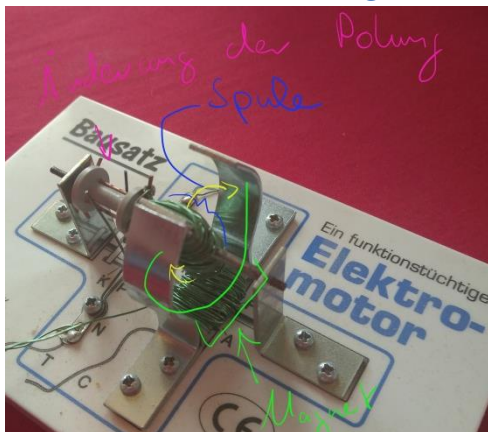
indem man sehr schnell die max. Spannung an und wieder ausschaltet (z.B.: 5V in einer Sekunde 10x an und aus = 0,1V). Das ist bei Microcontroller wichtig, weil sie von ihrer Hardware in der Regel aus nicht die Möglichkeit haben, eine geringere mittlere Spannung zu schalten.

*Servomotor Steuerung durch PWM-Signal (PulsweitenModulation) → vgl. (oomlout, 13.06.2019)*

Die Stellung des Servos wird als PWM-Befehl ausgeführt. PWM ist eine andere Möglichkeit, eine geringere Spannung darzustellen, ohne einen Widerstand. Der Vorteil bei dieser Methode ist, dass die eigentliche Spannung nur auf einen Zeitabschnitt aufgeteilt wird, ohne durch (wie es z.B. bei

Widerständen der Fall ist) eine Verringerung der Spannung. Man bekommt hier also eine geringere mittlere Spannung,

### **Der Elektromotor – ein wichtiger Bestandteil eines Servos**



**Abbildung 9: Beispiel eines Elektromotors – Hier wird anstatt einer kompletten Umpolung der Spule nur der Stromfluss durch die Spule kurz ausgesetzt – der Motor dreht sich dadurch hier in diesem Beispiel etwas langsamer.**

Ein Elektromotor besteht aus einer sich drehenden Achse mit Spule und Eisenkern (beide Dinge um ein stärkeres Magnetfeld zu erzeugen), welcher halb umgeben ist von einem Magneten. Dadurch dass Strom durch den Motor fließt, gibt es jetzt auf einmal jeweils zwei Magnetfelder. Das eine Magnetfeld ist ständig konstant, das ist der grün eingezeichnete Magnet, der andere Magnet ändert sich dagegen ständig, das ist bei der blauen eingezeichneten Spule. Dadurch, dass sich bei der Spule ständig die Stromrichtung ändert, wird die eine Seite mal zum grün eingezeichneten Magneten gelenkt, die andere Seite weg, weil diese Seite dieselbe Polung hat wie der grün eingezeichnete Magnet. Durch die Änderung der Stromrichtung wird jetzt die Spule umgepolt, das bedeutet, dass die selben Pole sich nun gegenüber stehen - Spule und Magnet. Deshalb gehen die beide Magnete auseinander, es bewegt sich jedoch nur die Spule, weil der andere Magnet fest ist. Die andere Seite der Spule hat dagegen nicht dieselbe Polung wie der grün eingezeichnete Magnet und möchte deshalb zu dem Magneten. Durch die Fliehkraft dreht es sich immer weiter. Die Polung kann man zum Beispiel durch Schleifkontakte ändern.



## 2. Beschleunigungssensor

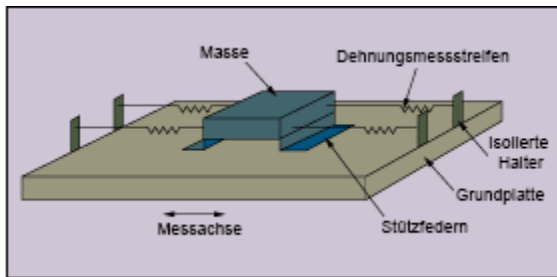


Abbildung 10: Aufbau eines Beschleunigungssensors  
(Beschleunigungssensor, 20.6.2019)



Abbildung 11: Unser  
Beschleunigungssensor kann  
Verschiebungen in 3 Achsen messen:  
X, Y & Z

Beschleunigungssensoren sind Sensoren, die Drehungen, bzw. Bewegung einer Masse durch Dehnungsmessstreifen registrieren, welche wie eine Feder gestreckt oder gestaucht werden können. Je nachdem, wie stark die Dehnungsmessstreifen gestaucht oder gestreckt sind, verhält sich elektrische Ladung proportional zur auf die Masse einwirkenden Kraft, ähnlich wie ein Widerstand. Deshalb kann man diesen Sensor als Beschleunigungssensor wie auch als Sensor verwenden, der die Drehung eines Körpers erfasst, weil dann anstatt der Beschleunigungskraft die Erdanziehung an der Masse zieht. Die Stützfedern in der Mitte stabilisieren die Masse nur noch, damit sie nicht von der Messachse in ungewollte Richtungen geht, die dafür sorgen würde, dass gemessener Strom und die einwirkende Kraft nicht mehr proportional zueinander sind. Vgl. (Beschleunigungssensor, 20.6.2019)

### 3. Zahnräder

Zahnräder sind dafür da, Momente zu übertragen und Geschwindigkeiten über die Übersetzung zu ändern. Zahnräder haben eine spezielle Konstruktion = Evolventenverzahnung, um möglichst gut zu funktionieren.

Da war die Idee, dass man sich einen Kreis (das Zahnrad) nimmt und den erst einmal rollen lässt. Dann schaut man sich mehrere Punkte an (siehe Schaubild 2 nächste Seite) – hier steht der Punkt M für den Mittelpunkt des Kreises, den Punkt braucht man, um anzuzeigen, wie weit der Kreis jetzt gerollt war. Der andere Punkt ist A, der ganz außen vom Kreis liegt. Jetzt bewegt man den Kreis etwas nach rechts und schaut, wie die beiden Punkte A und M liegen. Was auffällt ist, dass zwar der Mittelpunkt M immer konstant in gleichen Abständen nach rechts wandert, während A sich in einer Art Kurve bewegt.

Was hat diese Erkenntnis jetzt mit Zahnrädern zu tun? Man weiß jetzt, wie sich verschiedene Punkte im Kreis bewegen, wodurch man weiß, wie sich ein Kreis abrollt. Ein Zahnrad ist zwar an sich nicht direkt ein Kreis (außer man füllt dessen Zahnlücken mit etwas gleichmäßig auf), aber wenn zwei Zahnräder zusammen kommen, sollen sie bei der Stelle, wo sie ineinander greifen, wieder so wirken, als wären die beiden Zahnräder zwei übereinander gelappte Kreise, d.h. sie sollen sich gegenseitig ergänzen. Wäre es anders, dann würden sich die Zahnräder nicht richtig abrollen und sich wahrscheinlich verhaken. So kommt man jetzt auf die Form.

Was jetzt auch noch wichtig ist, ist der Winkel  $\alpha$ . Er gibt an, in welchen Winkel die linke Kurve (also die Kurve, die zum Beispiel vom Punkt A erzeugt wurden ist) und der Mittelpunkt zueinander stehen. Hier als Linie ist dann die Tangente gemeint beim Punkt A. Durch ausprobieren haben Konstrukteure gesehen, dass sich wohl ein  $20^\circ$  Winkel am besten eignet.

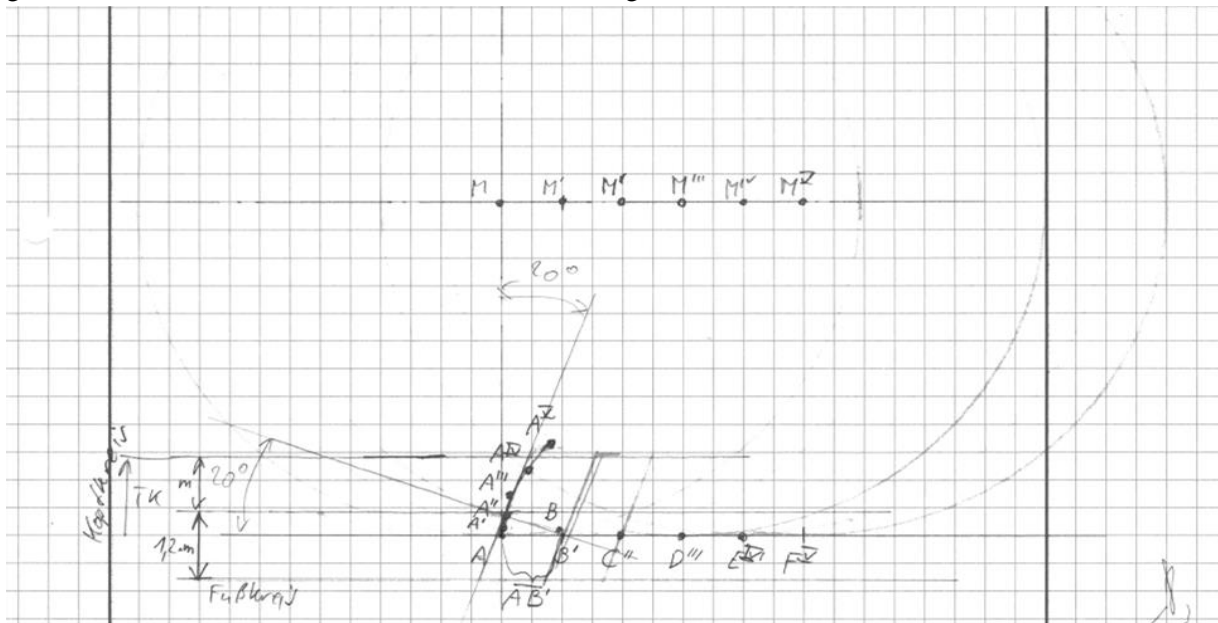


Abbildung 12: Die Fortbewegung eines Rades, woraus man dann die Zahnform her ableiten kann

Bei einem Zahnradpaar (Verbindung von 2 Zahnrädern) sollte man als weiteren Punkt noch wissen, wie viele Zähne jetzt jeweils ein Zahnrad hat. Dafür rechnet man bei beiden Zahnrädern den Durchmesser des mittleren Kreises (Teilkreis d) geteilt durch den Abstand äußerer Kreis / innerer Kreis (m) des Zahnrades und vergleicht jetzt die beiden.

Man muss bei der Rechnung einige Sachen beachten: der Abstand äußerer Kreis / Teilkreis (Modul m) bzw. Teilkreis / innerer Kreis ( $1,2 \cdot \text{Modul } m$ ) muss immer bei beiden Zahnrädern der selbe sein, damit die Zähne auch optimal in einander greifen können, und bei beiden Zahnrädern muss als Ergebnis der Rechnung immer eine natürliche Zahl heraus kommen, weil es keine halben / negativen Zahnräder gibt. Zum Ausprobieren für meine Zahnräder habe ich eine Tabelle erstellt, mit den Durchmessern, die

meine Zahnräder haben sollen (ich habe später nur kleinere genommen, weil aufgefallen ist, dass ansonsten der Platz nicht reichen wird / der Greifer viel zu groß wird. Die Rechnung ist aber im Ganzen nach demselben Prinzip nach verlaufen).

Berechnung der Zähnezah:  $Z = \text{Teilkreisdurchmesser } d / \text{Modul } m$

Teilkreis / Modul m	m=5	m=4	m=3	m=2,5
$d_1=50\text{mm}$	$Z_1=50/5=10$	$Z_1=50/4=12,5$	$Z_1=50/3=16,8$	$Z_1=50/2,5=20$
$d_2=70\text{mm}$	$Z_2=70/5=14$	$Z_2=70/4=17,5$	$Z_2=70/3=23,3$	$Z_2=70/2,5=28$

Man kann jetzt hier sehen, dass man für den Modul m nicht 4 oder 3 nehmen sollte, sondern eher 5 oder 2,5. m=5 ist dennoch nicht gerade zu empfehlen, weil hier die Zähneanzahl recht gering ist (Grenzzähnezah 13, bei weniger Zähnen wird durch Hinterschnitt der Zahnfuß geschwächt). Generell kann man sagen, dass sich Zahnräder mit mehr Zähnen besser und ruhiger drehen lassen, weil mehr Zähne ineinander greifen können. Entwickler von Zahnrädern sollten also darauf achten, die Zahnbreite und den Zahneingriff nicht zu klein zu machen.

Wie am Anfang des Textes beschrieben können Zahnräder über eine Übersetzung die Geschwindigkeit ändern. Dies kann man dann auch mit den Formeln berechnen:

$i = \text{Umfang größeres Zahnrad} / \text{Umfang kleineres Zahnrad}$

$i = \text{Anzahl der Zähne großes Zahnrad} / \text{Anzahl der Zähne kleines Zahnrad}$

(wenn man von einem großen Zahnrad auf ein kleineres geht; wenn man von einem kleinen Zahnrad zu einem großem Zahnrad übersetzen möchte vertauscht man einfach die beiden Quotienten).

Bei beiden Formeln wird dasselbe Ergebnis herauskommen. Das Ergebnis kann man dann mit der Drehgeschwindigkeit  $v$  multiplizieren, und weiß dann, wie schnell sich das Zahnrad dreht.

Anders ausgedrückt: Das Ergebnis drückt die prozentuale Änderung der Drehgeschwindigkeit aus.

Wenn man von einem kleinen Zahnrad auf ein größeres geht, dann wird die Geschwindigkeit kleiner ( $i$  muss also  $< 1$  sein); anders herum wird sie natürlich schneller ( $i$  muss also  $> 1$  sein).

Zahnrad GROß: $U=70\text{cm}$ ; $d=14$	Schnellere Übersetzung	$0,7\text{m} / 0,5\text{m} = 14 / 10 = 1,4$
Zahnrad klein: $U=50\text{cm}$ ; $d=10$	Kleinere Übersetzung	$0,5\text{m} / 0,7\text{m} = 10 / 14 \sim 0,7$

Am Ende stellte sich heraus, dass ich mir eigentlich nicht eine solche Mühe machen muss, weil nach (wiederholtem) herumprobieren in FreeCAD hat die Zahnradfunktion wieder funktioniert (nach Funktions-Recherche im Internet). Jetzt musste ich nur noch die Anzahl der Zähne bestimmen und das Programm hat mir das Zahnrad an die passende Größe angepasst. Mein Fehler war es, dass ich keinen Körper erstellt habe und deshalb das Programm nicht wusste, wohin es das Zahnrad jetzt basteln soll. Immerhin habe ich bei der Sache mit Zahnrädern selbst sehr viel gelernt, was ich später bestimmt auch noch brauchen werde.

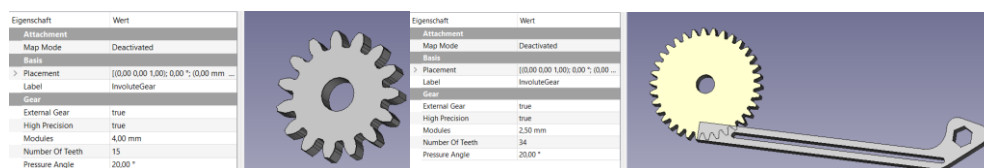


Abbildung 13: Eine Vorstufe eines Greiferarms (FreeCAD)

Abbildung 14: Ein in FreeCAD konstruiertes Zahnrad

#### 4. Greifer

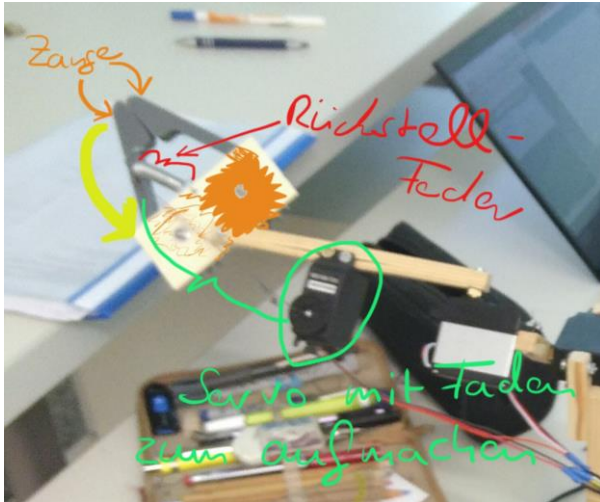


Abbildung 15: Der Greifer: Aufbau

##### Greifer-Funktionsweise:

###### - Greiferzangen öffnen:

Der Greifer funktioniert im Grunde so, dass ein Motor an einem Faden zieht und so die eine Seite der Zange öffnet, weil der Faden praktisch nach hinten gezogen wird. Somit muss sich die eine Seite der Zange bewegen, um den Kräften standzuhalten (alternativ könnte auch der Faden reißen, passiert aber nicht, weil die einzelnen Seiten der Zange drehbar sind). Die andere Seite der Zange dreht sich aber zur gleichen Zeit mit,

weil diese Seite über zwei Zahnräder (in der Zeichnung orange angedeutet) mit in die andere Richtung zum Öffnen bewegt wird.

###### - Greiferzangen schließen:

Da der Servo kann den Greifer nicht mehr schließen kann, weil der Faden flexibel und die Reibungskräfte wahrscheinlich zu groß sein werden, muss eine Feder das erledigen. Diese Feder zieht die ganze Zeit am Greifer entgegen der Richtung, in der der Servo die Zange zieht. Das ist auch wichtig, damit der Faden immer gespannt bleibt, weil der Servo mit einem durchhängenden Faden keine Kraft ausüben kann. Die Feder zieht ansonsten so lange am Greifer, bis er geschlossen ist, weil die beiden Zangenstücke ein Weiterkommen verhindern.

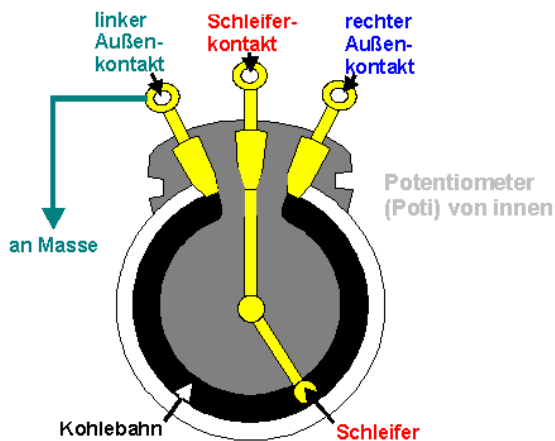
Wichtig ist nur, dass der Servo eine höhere Öffnungskraft aufbringen kann als die Schließ-Feder; und das kann man wie folgt feststellen (ich habe es dann experimentell festgestellt, indem ich verschiedene Federn eingesetzt und ausprobiert habe):

$$F_{\text{Feder}} = D * s_0$$

Das heißt aber auch, dass je länger eine Feder gespannt wird, desto höher ist auch deren Federkraft, weshalb es immer gilt, immer eine Feder zu finden und einzubauen, die möglichst entspannt, wenn der Greifer geschlossen ist.

## 5. Potentiometer

Abbildung 16: Aufbau eines Potentiometers (<http://www.rockprojekt.de/E-Gitarre/images/Poti.gif>)



Ein Potentiometer ist ein veränderlicher Widerstand, je nach Einstellung hat er einen höheren Widerstand, mal einen kleineren Widerstand. Er wird verwendet, um Sachen wie zum Beispiel Lautstärke eines Radios linear einstellen zu können. Im Inneren befindet sich dazu eine Kohlebahn, welche immer gleich viel Spannung zulässt, jedoch wird der Widerstand umso höher, desto mehr Kohlebahn der Strom überwinden muss. Über diese Kohlebahn fährt ein Schleifer, der dann irgendwo den Strom auf die Kohlebahn bringt. Diesen Schleifer dreht man auch über Drehregler zum Beispiel beim Radio, und sagt so, wie viel Strom fließen soll. Es gibt drei Anschlüsse, weil der

Potentiometer eigentlich mehr als Spannungsteiler, anstatt als Widerstand arbeitet, d.h. dass der Potentiometer die Spannung nicht vernichtet, sondern aufteilt. Um geringe Energieverluste zu haben wird daher geraten, einen Kontakt zur Masse zu verbinden. Der Mittlere ist in der Regel der Pluspol und der letzte ist schließlich der, wo man die entsprechende Wunschspannung haben möchte.

## 6. Sonstiges

### Verkabelungstricks

Bei der Verkabelung wurde, wie das letzte Mal schon beschrieben, darauf geachtet, dass sich die Kabel nicht verheddern. Ein speziell konstruierter kleiner Arm verhindert das Verheddern. Die Kabel sind hierbei in einer kleinen Führung aus zwei kleinen Hölzern, wo sie von alleine nicht rauskommen können, sich aber noch etwas bewegen können. Das ist wichtig, damit wenn der Roboter sich bewegt, kein Kabel abreißt, welches zum Beispiel am Servo ganz vorne am Greifer dran ist. Im Prinzip müsste



man sogar ein Gewicht dran machen, um das Kabel an einem Punkt nach unten zu ziehen, damit das Kabel zum einem nicht durchhängen kann, und sich dadurch verheddern kann, zum anderen sich immer noch bewegen kann, falls sich der Arm zum Beispiel „streckt“. Das Prinzip wurde schon bei Oberleitungen bei Zügen verwendet, wo das Kabel auch immer gespannt sein muss, da es sonst reißen könnte. Da bewegt sich das Kabel aber nur, weil im Sommer das Kabel durch die Wärme länger wird, weil die Atome des Kabels mehr Platz benötigen. Ob das Gewicht noch nötig wird, wird sich in nächster Zeit noch zeigen. Vorerst wird aber kein Gewicht eingebaut, solange es auch nicht gebraucht wird.

Abbildung 17: Das selbe Problem wie bei der Oberleitung könnte es auch bei meinem Roboter geben – hier wird es gelöst durch extra Gewichte (Ausdehnung von Festkörpern, 16.1.2019)



## Programmcode: für Arduino

### Notizen, Erklärungen:

- //Notiz
  - /\* Notiz über mehrere Zeilen \*/
- ➔ Durch diese Formatierung ist es möglich, den Programmcode direkt in die Arduino IDE einzufügen, und er wird durch das Farbige übersichtlicher (Arduino, V.1.8.9)

```
#include <Servo.h>
/* Einbindung der Servo-Bibliothek, damit das Programmieren der Servos leichter wird */

// Pinfestlegung der Sensoren, acc ist die Abkürzung für den Neigungssensor
int accZPin=0;    //Z-Achse
int accYPin=1;    //Y-Achse
int accXPin=2;    //X-Achse
int onePot=3;     //Greifer-Potentiometer
int twoPot=4;     //Neigungs-Potentiometer

/* Variablenerstellung für die unterschiedlichen Gelenke des Roboters – Namen nach Gelenk */
int greifer;
int greifer_alt = 0; // was [...]_alt bedeutet wird unten beschrieben
int dreher;
int dreher_alt = 0;
int neigung;
int neigung_alt = 0;

//Variablenfestlegung für die benötigte Pause einer Servobewegung
int pause_greifer;
int pause_dreher;
int pause_neigung;

//Namensfestlegung für die verschiedenen Servos
Servo first;    //Greifer
Servo second;   //Dreh
Servo third;    //Neigung

//Code, der einmal ausgeführt wird:
void setup() {
/* Baudrate einstellen – die Geschwindigkeit, womit der Arduino mit einem PC redet */
  Serial.begin(9600);
//Den Servos jeweils ihren Pin zuweisen
  first.attach(10);
  second.attach(11);
  third.attach(9);
}
```

```

//Code, der ständig wiederholt wird:
void loop() {
  /* zuerst wird zum leichteren Kalibrieren (einstellen) des Neigungssensor ihre Werte ausgelesen und an
  den PC geschickt */
  Serial.print(" X=");
  Serial.print(analogRead(accXPin));
  Serial.print(" Y=");
  Serial.print(analogRead(accYPin));
  Serial.print(" Z=");
  Serial.println(analogRead(accZPin));

  //Hier wird jetzt in diesem Block der Greifer-Servo gesteuert.
  greifer = map(analogRead(onePot), 215, 300, 30, 190); /* Hier wird der Greifer-Potentiometer ausgelesen
  und in die Werte für den Servo umgerechnet – die Variable „Greifer“ nimmt diesen Wert an. In der Klam-
  mer werden zuerst die Werte vom Sensor von 215 – 300 in den Winkel für den Servo von 30°-190° umge-
  rechnet. (Arduino, V.1.8.9)*/
  Serial.print(" Servo Greifer: "); //Winkel des Servos an den Computer senden
  Serial.print(greifer);          /* Immer zuerst was angezeigt wird (voriger Befehl), danach die Variabel */
  pause_greifer = abs(greifer - greifer_alt) * 3; /* Die Verzögerung für die Drehung wird ständig neu an-
  gepasst, je nachdem, wie groß die Winkeldifferenz ist. Dafür wird die differenz der alten Winkeldifferenz
  und der neuen genommen und mit drei multipliziert. Die neue Variable für diesen Vorgang hat immer die
  Vorsilbe „pause“. */
  if (pause_greifer <= 600 && greifer > 0){ /* Wenn der Winkelunterschied kleiner als 200° ist (da vorher
  mit drei Multipliziert wurden ist, muss man jetzt durch drei teilen, um auf diese Zahl zu kommen) und der
  Servowinkel nicht negativ ist, soll der Servo bewegt werden – das mache ich, damit Fehlmessungen oder
  falsche Werte weniger im Betrieb auffallen – eine Sicherheitsabfrage. */
    first.write(greifer); //Greifer-Servo bewegt sich zu dem vorgegebenen Winkel
    delay(pause_dreher); //Pause für die Bewegung
  }
  greifer_alt = greifer; /* den aktuellen Winkel als alten Winkel nehmen, damit in der nächsten Ausführ-
  ung des Codes der Wert wieder neu berechnet werden kann. */

  //Hier wird jetzt in diesem Block der Dreher-Servo gesteuert, funktioniert im Grunde so wie der Obige,
  Variablen und Zahlen wurden aktualisiert
  dreher = map(analogRead(accZPin), 250, 450, 30, 330); // Umrechnung der Werte
  Serial.print(" Servo Dreher: "); //Ausgabe des Servo-Winkels an den PC
  Serial.print(dreher);
  pause_dreher = abs(dreher - dreher_alt) * 3; //Pause berechnen
  if (pause_dreher <= 80 && dreher > 0){ //Sicherheitsabfrage
    second.write(dreher); //Dreh-Servo bewegen
    delay(pause_dreher); //Dynamische Pause
  }
  dreher_alt = dreher; //Werte aktualisieren

  /* Dieser Block funktioniert genauso wie die beiden darüber – die Variablen und Zahlen sind nur für den
  Neigungssensor angepasst wurden */
  neigung = map(analogRead(accYPin), 270, 410, 30, 330);
  Serial.print(" Servo Neigung: ");
  Serial.print(neigung);

```

```
pause_neigung = abs(neigung - neigung_alt) * 5;  
if (pause_neigung <= 80 && neigung > 0){  
  third.write(neigung);  
  delay(pause_neigung);  
}  
neigung_alt = neigung;  
}
```

## Literaturverzeichnis

Arduino. (V.1.8.9). *Referenz*. Arduino IDE.

Unbekannter Autor (16.1.2019). *Ausdehnung von Festkörpern*. [https://www.schule-bw.de/faecher-und-schularten/mathematisch-naturwissenschaftliche-faecher/physik/unterrichtsmaterialien/waermelehre/ausdehnung/ausdehnung\\_festkoerper.htm](https://www.schule-bw.de/faecher-und-schularten/mathematisch-naturwissenschaftliche-faecher/physik/unterrichtsmaterialien/waermelehre/ausdehnung/ausdehnung_festkoerper.htm).

Unbekannter Autor (20.6.2019). *Beschleunigungssensor*. <https://www.omega.de/prodinfo/beschleunigungsmesser-vibrationsaufnehmer.html>.

oomlout. (13.06.2019). *Servomotor einfach erklärt!* <https://www.precifast.de/servomotor-funktion-ansteuern-anschliessen/> : CNC Blog.

Unbekannter Autor (12.06.2019). *Servomotoren*. <http://home.teleos-web.de/vsteinkamp/antriebe/servo/servomotoren.htm>.

## Abbildungsverzeichnis

Abbildung 1: Der fertige Roboterarm mit Handschuh: Aufbau (oberes Bild) .....	3
Abbildung 2: Der Handschuh: Aufbau (rechtes Bild) .....	3
Abbildung 3: 3D-Konstruktion des Roboters (Free-CAD) .....	4
Abbildung 4: Schaltplan des Roboters .....	6
Abbildung 5: Schaltplan des Roboters als andere Darstellungsweise; die Farben der Kabel sind gleich geblieben .....	6
Abbildung 6: Servo: Aufbau (oomlout, 13.06.2019), Bearbeitet .....	7
Abbildung 7: Regelkreis eines Servomotors .....	7
Abbildung 8: s-t-Diagramm von der PWM .....	8
Abbildung 9: Beispiel eines Elektromotors – Hier wird anstatt einer kompletten Umpolung der Spule nur der Stromfluss durch die Spule kurz ausgesetzt – der Motor dreht sich dadurch hier in diesem Beispiel etwas langsamer. ....	8
Abbildung 10: Aufbau eines Beschleunigungssensors (Beschleunigungssensor, 20.6.2019) .....	9
Abbildung 11: Unser Beschleunigungssensor kann Verschiebungen in 3 Achsen messen: X, Y & Z ....	9
Abbildung 12: Die Fortbewegung eines Rades, woraus man dann die Zahnform her ableiten kann ....	10
Abbildung 13: Eine Vorstufe eines Greiferarms (FreeCAD) .....	11
Abbildung 14: Ein in FreeCAD konstruiertes Zahnrad .....	11
Abbildung 15: Der Greifer: Aufbau .....	12
Abbildung 16: Aufbau eines Potentiometers ( <a href="http://www.rockprojekt.de/E-Gitarre/images/Poti.gif">http://www.rockprojekt.de/E-Gitarre/images/Poti.gif</a> ) ..	13
Abbildung 17: Das selbe Problem wie bei der Oberleitung könnte es auch bei meinem Roboter geben – hier wird es gelöst durch extra Gewichte (Ausdehnung von Festkörpern, 16.1.2019) .....	13