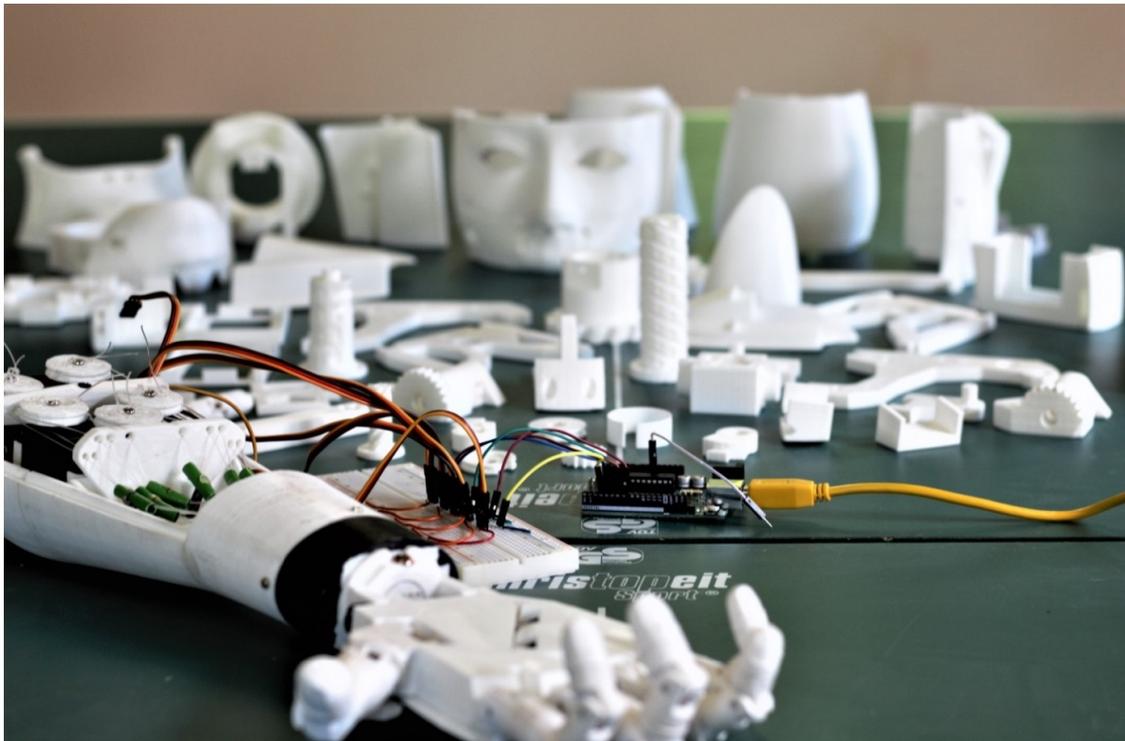


# Cogito Ergo Moveo Brachium

Jaú Gretler

2019



Maturaarbeit am Literaturgymnasium Rämibühl, Klasse 6c  
Betreuung: Manuel Benz

Ich, Jaú Gretler, bestätige, dass ich diese Arbeit selbst geleistet habe, dass sie kein Plagiat und auch keine Fälschung ist, dass alle übernommenen Teile korrekt erwähnt, zitiert und bibliografiert sind und ich nur die erwähnten Hilfsmittel verwendet habe. Ich bin von den Konsequenzen, die eine Nichteinhaltung dieser Punkte nach sich zieht, in Kenntnis gesetzt worden.

Ort, Datum: ----- Unterschrift: -----



# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>1</b>
1.1	Zielsetzung . . . . .	1
1.2	Aufbau . . . . .	1
1.3	Inspirationen . . . . .	2
<b>2</b>	<b>Einführung</b>	<b>6</b>
2.1	3D-Druck . . . . .	6
2.2	Die elektronischen Steuerungseinheiten . . . . .	16
2.3	Biometrische Datenerfassung . . . . .	19
<b>3</b>	<b>Verschiedene Möglichkeiten der Ansteuerung des Armes</b>	<b>23</b>
3.1	Per Gedankenkraft . . . . .	23
3.2	Per PS4-Controller . . . . .	24
3.3	Per Bewegungsübertragung . . . . .	27
<b>4</b>	<b>Endprodukt und Anwendungsmöglichkeiten</b>	<b>32</b>
4.1	Endprodukt . . . . .	32
4.2	Anwendungsmöglichkeiten . . . . .	33
<b>5</b>	<b>Reflexion</b>	<b>39</b>
5.1	Erfahrungen . . . . .	39
5.2	Originalität . . . . .	42
5.3	Kritik am eigenen Arbeitsprozess . . . . .	43
5.4	Perspektiven . . . . .	46
<b>6</b>	<b>Quellen- und Literaturverzeichnis</b>	<b>50</b>
<b>7</b>	<b>Anhang</b>	<b>51</b>
<b>A</b>	<b>Versuch der Ansteuerung per Gedankenkraft</b>	<b>51</b>
<b>B</b>	<b>Diskussion Projekt InMoov</b>	<b>53</b>
<b>C</b>	<b>TekkieLab</b>	<b>58</b>
<b>D</b>	<b>Arbeitsmoral</b>	<b>59</b>

# 1 Einleitung

## 1.1 Zielsetzung

Ziel dieser Arbeit war zum einen, meine Kenntnisse der Steuereinheit Arduino, des *Einplatinen-Computers*<sup>1</sup> RPP und der Programmiersprachen C++ und Python zu vertiefen. Zum andern wollte ich das erlernte Wissen aus Informatik und Elektrotechnik auf diese oder jene Weise mit dem eigenen Körper, sprich mit biometrischen Daten, welche sich vom selben erfassen liessen, verbinden. Im ersten Schritt beschloss ich die Konstruktion eines mechanischen humanoiden Armes als Hardware-Angelpunkt meiner Arbeit festzulegen. Das Design der einzelnen Bauteile und das mechanische Konzept, wie die Fingergelenke, das Hand- und das Ellbogengelenk motorisiert werden sollten, entnahm ich der Arbeit des französischen Designers und Bildhauers Gael Langevin. Dann wandte ich mich der Frage zu, welche biometrischen Daten ich verwenden könnte, um den Arm anzusteuern.

## 1.2 Aufbau

Im ersten Teil dieser Arbeit, werde ich die einzelnen Bauteile und Technologien vorstellen, welche ich verwendete bzw. deren Verständnis nötig war, um einerseits den humanoiden Arm bauen zu können und andererseits das Erfassen der biometrischen Daten zu ermöglichen. Dem Thema 3D-Druck werde ich viel Platz einräumen, da auch ein beachtlicher Anteil der für diese Arbeit verwendeten Zeit in dieses Thema investiert wurde.

Im zweiten Teil werde ich drei Versuche der Ansteuerung präsentieren. Der erste Versuch beinhaltet die Ansteuerung über Gedankenkraft, der zweite diejenige durch einen PS4-Controller und der dritte diejenige über Bewegungsübertragung.

Im dritten Teil werde ich das Endprodukt beschreiben und mögliche Anwendungsmöglichkeiten skizzieren.

---

<sup>1</sup>Bei einem Einplatinen-Computer befinden sich sämtliche zum Betrieb nötigen elektronischen Komponenten auf einer einzigen Platine. Eine Platine ist eine meist mit Kupfer oder Silber beschichtete Platte, welche in der Elektrotechnik dazu verwendet wird verschiedene Bauelemente elektronisch miteinander zu verbinden, meistens unter Einsatz von Lötzinn[1].

Die kritische Analyse meines Arbeitsprozesses und das Aufzeigen möglicher Perspektiven, was die Zukunft und die Erweiterbarkeit meines Produktes betrifft, folgt im vierten Teil. Zum anderen werde ich auch auf meine Erfahrungen und auf weiterführende Themen eingehen.

Um die Lektüre dieser Arbeit so gewinnbringend wie möglich zu gestalten, werde ich versuchen, alle Begriffe, welche Personen ohne spezifische Affinität zu Informations- und Elektrotechnik vielleicht nicht kennen, entweder im Vorherein oder in den Fussnoten verständlicher zu formulieren. Im letzteren Falle sind die Wörter jeweils *kursiv* gedruckt.

## 1.3 Inspirationen

### **inmoov.fr**

Im Frühjahr 2018 entdeckte ich auf der Website `thingiverse.com` Bauteile einer 3D-druckbaren Roboterhand. Die Recherche über den Designer der Roboterhand führte mich zur Website `inmoov.fr`. Das *Open-Source-Projekt*<sup>2</sup> dessen Beschreibung ich auf der Website las, befasst sich mit dem Bau und der Programmierung des Oberkörpers eines lebensgrossen humanoiden Roboters welcher den gleichen Namen wie das Projekt trägt; InMoov.

In dieser Arbeit werde ich diesen Namen häufig verwenden. Wenige Male wird er für das ganze Projekt stehen, in den allermeisten Fällen jedoch für den Roboter selbst, entweder für die Summe seiner Teile oder für ein ihm angehöriges Körperglied. Die oben erwähnte Website `thingiverse.com` beinhaltet eine der grössten öffentlich zugänglichen Sammlungen an 3D-Modellen, welche alle im *Datei-Format STL*<sup>3</sup> gratis herunterladbar sind.

---

<sup>2</sup>In der Informatik spricht man von Open-Source, wenn der Quell-Code einer Anwendung oder eines Programmes bekannt sprich öffentlich zugänglich ist. In der Technikindustrie verlangt der Begriff Open-Source nach den Konstruktionsplänen oder anderer Dokumentation, welche den Nachbau des als Open-Source bezeichneten Objektes erlaubt.

<sup>3</sup>Die Abkürzung STL steht in diesem Kontext für Standard Triangulation Language und benennt einen Dateitypen, welcher im Prinzip für eine bestimmte 3D-Druck-Technik, die Stereolithographie, konzipiert wurde. Auf diese Technik werde ich im Kapitel 3D-Druck näher eingehen. Der Dateitypus eignet sich für dreidimensionale Körper, welche über ein geschlossenes Volumen verfügen. Die Oberfläche des beschriebenen Körpers wird in Dreiecke aufgeteilt, daher der Name Triangulation

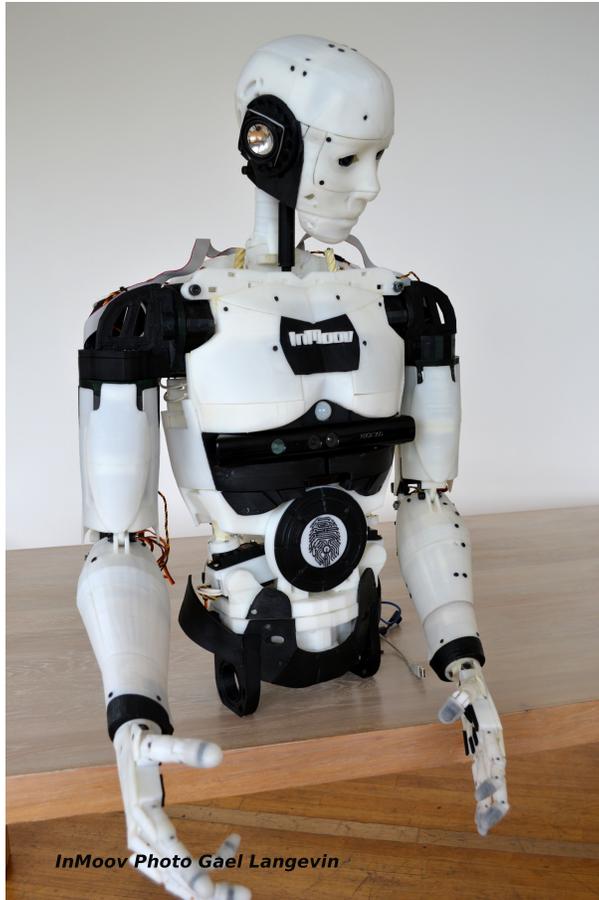


Abbildung 1: InMoov Oberkörper, Photo von Gael Langevin

Abgesehen von Stücken von Fischerleine, welche als Sehnen fungieren, sind alle nicht-elektronischen Bauteile des Roboters 3D-druckbar. Alles, was man benötigt, ist für circa 1500 CHF erwerbbar, Kostenstand 2018. Ausgeschlossen bei dieser Rechnung ist der Zugang zu einem 3D-Drucker, welcher natürlich essentiell ist für die Durchführung des Projektes. Das minimale Druckvolumen, welches benötigt wird, beträgt  $12 \times 12 \times 12 \text{ cm}^3$  und der Preis für einen Drucker mit diesem eher bescheidenen Bauraum beginnt im niedrigen dreistelligen Bereich.

Die Bewegungen, welche sich mit den insgesamt zweiunddreissig *Servos*<sup>4</sup> ausführen lassen, sind zwar nicht mit dem Bewegungsspielraum eines Hu-

---

<sup>4</sup>Servos sind klassischerweise Gleichstrom-Motoren, welche einen Rotor schrittweise mit einem Drehwinkel von  $180^\circ$  um sich selbst rotieren lassen. Wird ihr Winkel nach Ausführung des Befehls verändert, können sie diesen korrigieren.

manoiden wie beispielsweise “Asimo” [2] von Honda zu vergleichen, ganz zu schweigen von jenem eines Arms aus Fleisch und Blut; trotzdem kann man ihnen eine gewisse Flüssigkeit und Eleganz nicht absprechen. Konkret gibt es im Unterarm sechs Servos: Fünf um die “Sehnen” (Nylonfäden oder Fischerleinen) anzuspannen und somit die Finger zu strecken oder anzuziehen, und einen, um das Handgelenk zu rotieren. Im Oberarm gibt es einen weiteren, welcher das Beugen oder Strecken des Armes ermöglicht. Auch in der Schulter sitzen zwei Servos, welche einerseits das Heben und Senken der Schultern, aber auch die Rotation des Armes um die Schulter als Mittelpunkt bewerkstelligen können.

Interessierte finden im Anhang noch eine genauere Beschreibung des Projektes inklusive der Diskussion der Vor- und Nachteile des selbigen. Da ich zum Beginn dieser Arbeit über einen 3D-Drucker mit ausreichendem Druckvolumen verfügte, beschloss ich, in einem ersten Anlauf *einen* motorisierten Finger des Roboters zu bauen. Als dies gelang, begann ich Teile des Arms zu drucken und so fand die Konstruktion des Armes ihren Anfang.

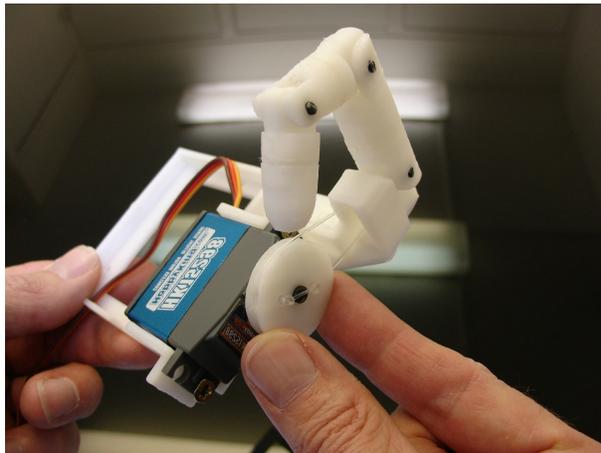


Abbildung 2: InMoov “Finger-Starter”,  
Photo von Gael Langevin

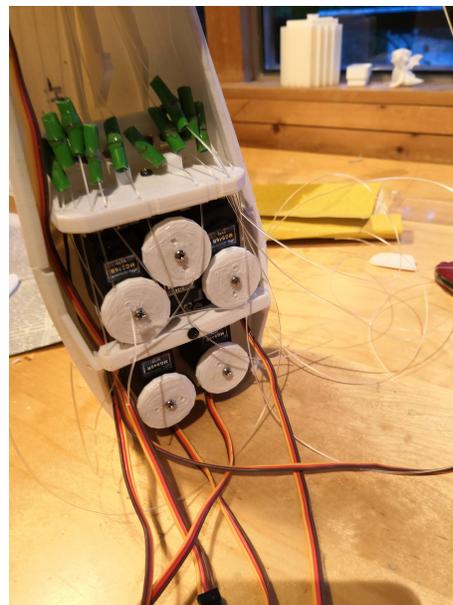


Abbildung 3: Nylonfäden fungieren als Sehnen im Unterarm

## Marc Frei

Ich las in einem Zeitungsartikel[3] über das Projekt eines Schweizer Künstlers namens M. Frei, in welchem er “per Gedankenkraft” eine Lampe zu steuern versuchte und es auch vermochte. Um dies zu bewerkstelligen, verwendet er ein Messgerät, um die elektrische Aktivität seines Hirnes aufzuzeichnen. Diese gemessenen Daten übermittelt er per Bluetooth an ein elektrotechnisches Steuerelement, auf welches ich später zu sprechen komme, welches die Stromzufuhr für die Glühbirne an- oder abschalten kann.

Auf meine Anfrage hin lud er mich an seinen Arbeitsplatz zu einem Interview ein, um mir sein Projekt zu zeigen. Die technische Umsetzung seiner Idee werde ich im Kapitel 3.1 erläutern. Um seine Idee in die Tat umzusetzen stützte er sich neben seinen Kenntnissen und Erfahrungen in der Elektrotechnik auch auf den Beitrag von Sophie Kravitz auf der Seite [sparkfun.com](http://sparkfun.com)<sup>5</sup>.

Eines der Projekte auf dieser Website befasst sich mit dem “Hacken” eines Gerätes namens “MindWave” der kalifornischen Firma NeuroSky.[4] In diesem findet sich eine Anleitung, wie man die Messungen von Hirnstromaktivitäten über Bluetooth vom MindWave zum besagtem Steuerelement übermitteln kann. Diese Anleitung macht sich M. Frei zunutze, um die Glühbirne zu manipulieren.

Während unseres knapp einstündigen Treffens erläuterte er mir, wie der Signalfluss etwa aussieht und was für Teile benötigt werden, und demonstrierte mir anschliessend die Funktionsweise. Ich zog das MindWave Headset über und versuchte angestrengt, mich mental so weit zu entspannen, dass die Lampe angehe.

Nach etwa einer halben Minute schlug ich die Augen auf, betrachtete die leuchtende Glühbirne und fühlte, wie eine neue Idee für die Erweiterung meiner Maturaarbeit geboren war.

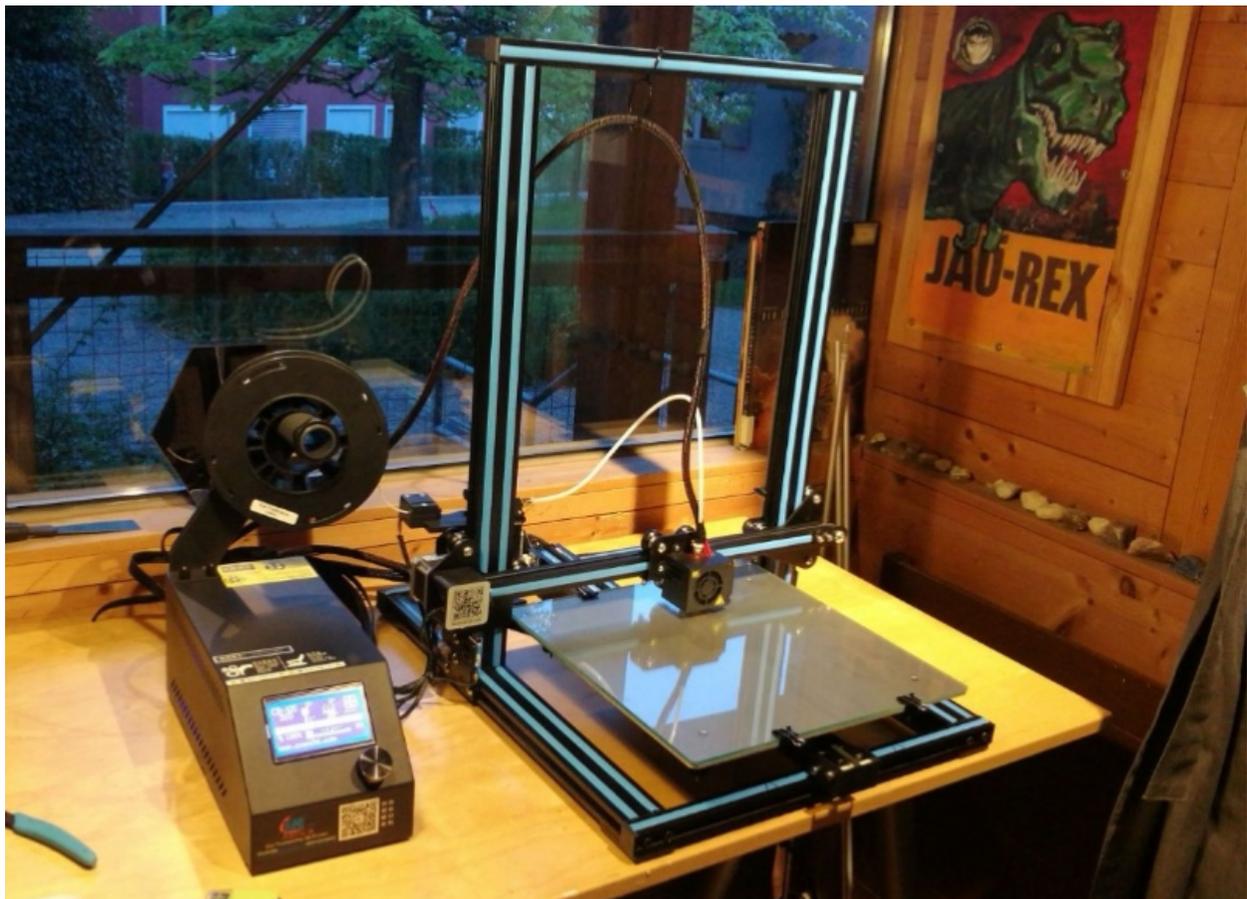
---

<sup>5</sup>Diese Plattform vertreibt zum einen elektronische Bauteile, aber sammelt und publiziert zum anderen auch Projektideen und stellt elektrotechnisches Fachwissen zur Verfügung.

## 2 Einführung

### 2.1 3D-Druck

Was viele Personen ohne eigene 3D-Druck-Erfahrung unterschätzen, ist dass das 3D-Drucken eine eigene Disziplin darstellt. Es gibt zahllose Foren, Bücher und Communities, die sich mit der Konstruktion, der Wartung und der Optimierung von 3D-Druck-Systemen auseinandersetzen. Denn es gibt viele Szenarien, wie ein Druckvorgang scheitern, aber nur wenige wie er gelingen kann. Bevor ich auf diese Szenarien zu sprechen komme, möchte ich zunächst in wenigen Worten erklären was ein 3D-Drucker ist und zu was er im Stande ist.<sup>6</sup>



*Abbildung 4: 3D-Drucker Creality CR10s, Quelle: Mein Zimmer*

---

<sup>6</sup>Dabei stütze ich mich primär auf eigene Erfahrungen, sekundär bzw. als Nachschlagewerk auf das Buch "Die neue Welt des 3D-Druckens" Lipson 2014

## Was ist ein 3D-Drucker?

Als 3D-Drucker werden Maschinen bezeichnet, die mit Hilfe Computer-gestützter Software aus verschiedensten flüssigen oder festen Materialien schichtweise ein dreidimensionales Objekt erschaffen basierend auf der Vorlage eines digitalen 3D-Modells.

## Verwendbare Materialien Stand Oktober 2018

Die kontinuierliche Weiterentwicklung bestehender 3D-Drucktechnologien gepaart mit der Erfindung neuer Technologien führte im Laufe der letzten zwei Jahrzehnte dazu, dass ein immer breiter werdendes Spektrum an Materialien genutzt werden kann, um 3D zu drucken. Es folgt eine Liste zum 3D-Druck verwendbarer Stoffe und Materialien, welche aber keinen Anspruch auf Vollständigkeit erhebt.

- Kunststoffe: eine Vielzahl von Thermoplasten, darunter die beiden prominentesten PLA und ABS.
- Metalle: verschiedene Legierungen, Gold, Silber, Stahl, Titan, Edelmetalle, Edelstahl, Nickel, Kobalt, Chrom, Kupfer, Bronze, Aluminium
- Ausserdem: Keramik, Gips, Wachs, "Bio Ink" [5], Nahrungsmittel[6]

Die beiden *Filamente*<sup>7</sup>, welche eine zentrale Rolle in der Konstruktion des Armes spielen, möchte ich hier etwas näher beschreiben:

PLA ist die Bezeichnung für Polylactid und für eines der aktuell gebräuchlichsten Filamente für den Hobby-Gebrauch von 3D-Druckern. Der Schmelzpunkt liegt bei circa 150° Celsius und eignet sich somit perfekt für einen Druckkopf wie den meinigen, welcher bei Temperaturen von bis zu 260° Celsius zu operieren vermag. Der Gründer des InMoov Projektes empfiehlt zwar den Gebrauch von ABS, welches ich weiter unten genauer beschreibe, trotzdem habe ich mich bei fast allen Teilen für den Gebrauch von PLA entschieden.

---

<sup>7</sup>Vergleichbar mit der Druckerpatrone in 2D Druckern. Als Filament bezeichnet man den thermoplastischen Kunststoff welcher beim FDM 3D-Druckverfahren dazu dient, konstant in flüssiger Form schichtweise auf der Druckplattform aufgetragen zu werden. Der terminologische Unterschied zum Begriff Material äussert sich darin, dass im Kontext des 3D-Druckens ein Filament meist dünn aufgerollt auf einer Spule aufbewahrt wird. Ein Thermoplast ist ein Kunststoff, welcher seine Form bei einer bestimmten Temperatur verändert.

Der Hauptunterschied zwischen PLA und ABS liegt meiner Erfahrung nach in der Bruchfestigkeit bzw. der Sprödigkeit.

ABS ist die Abkürzung für Acrylnitril-Butadien-Copolymer und besitzt einen für 3D-Druckzwecke ebenfalls gut gelegenen Schmelzpunkt von 110° Celsius. Neben der schon erwähnten erhöhten Zerbrechlichkeit gibt es jedoch auch einen nennenswerten Vorteil von ABS. Wenn man nämlich zwei ABS-Stücke in Aceton taucht und die beiden dann zusammenfügt, erhält man glatte, bis hin zu nicht mehr wahrnehmbaren Übergängen. Diesen Prozess nennt man auch Kaltschweissen.

### **Drei populäre 3D Drucktechnologien**

Seit die älteste 3D-Druck-Technik, die Stereolithographie, 1983 zum ersten mal in die Praxis umgesetzt wurde, vergrößerte sich nicht nur die Bandbreite der verwendbaren Stoffe um ein Vielfaches, sondern kamen zuerst für die Industrie und etwas zeitversetzt für den Privatgebrauch neue Verfahren auf den Markt. Die folgenden drei Verfahren erachte ich für den privaten oder semi-professionellen Gebrauch als die populärsten.<sup>8</sup>

- FDM: Fused Deposition Modelling

Materialien, welche sich unter Hitzeeinfluss verflüssigen werden schichtweise aufeinander aufgetragen. Das Material, in diesem Kontext Filament genannt wird dabei über den Druckkopf (Extruder) in flüssiger Form auf die Druckplatte aufgetragen und erkaltet dann in der gewünschten Position. Das ist die bekannteste, effizienteste und am einfachsten zu handhabende Drucktechnik für nicht-professionelle Anwender.

---

<sup>8</sup>In diesem Kapitel stütze ich mich primär auf Erfahrungswerte aus dem eigenen 3D-Druck-Alltag ab. Sekundär ist das Buch [7] anzugeben, welches ich immer wieder konsultierte um Aussagen zu verifizieren.

Vorteile:

- Wahrscheinlich die günstigste 3D-Drucktechnik.
- Weite Bandbreite an verwendbaren Filamenten mit diversen Eigenschaften wie Leitfähigkeit, Fluoreszenz und Farbenvielfalt.
- Technisch leichter umzusetzen als andere Techniken.
- Mehrere Druckköpfe können simultan arbeiten.

Nachteile:

- Grössere Druckaufträge werden schnell zeitintensiv.
  - Fehleranfälligkeit steigt rasant mit abnehmender Konstruktionspräzision oder Qualitätsklasse des Druckers.
  - Stützstrukturen werden nötig bei Überhängen über  $45^\circ$ .
  - Nur Materialien verwendbar, die sich durch einen Druckkopf pressen lassen.
- SLA: Stereolithographie  
In einem Becken voller Kunstharz erhöht sich der Füllstand graduell, bei jedem Schritt wird die Form des gewünschten Objektes mit UV-Licht abgestrahlt und erhärtet sich dann schlussendlich, so dass am Schluss nur die gewünschte Form übrig bleibt.

Vorteile:

- Mehrere Laser können simultan arbeiten.
- Präzision von 10 Mikrometern.

Nachteile:

- Druckprozess kann von für die Atemwege schädlichen Dämpfen begleitet werden.
  - Nur ein Material verfügbar.
- SLS: Selective Laser Sintering  
Ein Material liegt in Pulverform vor und wird in einer Druckkammer gleichmässig verteilt. Es wird mit einem Laserstrahl geschmolzen und erhärtet sich dann ebenfalls in der gewünschten Form.

Vorteile:

- Dem momentan schmelzenden Pulver unterliegende Pulverschichten ersetzen Stützstrukturen.
- Übrig gebliebenes Pulver kann im Gegensatz zum klassischen FDM-Filament effizient wiederverwertet werden.
- Pulverform erlaubt Einsatz von Stahl, Erzen und Titan.

Nachteile:

- Oberfläche eher porös als glatt.
- Es kann jeweils nur mit einem Pulver gedruckt werden.
- Explosionsgefahr verlangt hermetisch geschlossene, mit Stickstoff gefüllte Kammer.

Die Aussagen bezüglich der erzielbaren Materialeigenschaften sind insofern mit Vorsicht zu geniessen, als dass Eigenschaften wie beispielsweise Bruchfestigkeit oder Verformbarkeit nicht nur von der verwendeten Technik, sondern auch massgeblich vom spezifischen Material und der Arbeitsumgebung abhängen.

Befindet sich ein FDM-Drucker beispielsweise in einer thermisch kontrollierten Umgebung, wie etwa einer Plexiglashülle, kann er die vom Druckkopf emittierte Wärme viel besser bewahren. Fehlt diese Hülle, löst das Aufsteigen der durch den Druckkopf erhitzten Luft Konvektionsströme am Druckkopf aus, es wird also kalte Luft angezogen, welche auf diejenige Stelle trifft, an welcher der Druckkopf die Druckplatte berührt. Während normalerweise die Schicht, auf welche die neue Schicht an geschmolzenem Kunststoff aufgetragen wird bereits grösstenteils abgekühlt ist, führt die zusätzlich gespeicherte Wärme dazu, dass die Temperatur der unteren Schicht sich nah bei der aufzutragenden Schicht befindet, was zu einer stärkeren Bindung und somit zu erhöhter Stabilität bzw. Bruchfestigkeit führt. Umgekehrt verringert die auftreffende Kälte den Zusammenhalt zwischen den Schichten.

Besonders bei der Diskussion der Stabilität der Produkte verschiedener Verfahren ist zu erwähnen, dass sich mit allen oben genannten Technologien sehr hohe Stabilitätsgrade erzielen lassen, wenngleich SLA und SLS dank ihrer Unabhängigkeit von der thermostatischen Umgebung diesbezüglich einen Vorteil aufweisen.

## **Arbeitsschritte mit einem 3D-Drucker**

Der Prozess von der Idee bis zum fertigen Objekt läuft über folgende Etappen ab.

1. Seinen Gedanken Form geben und digital eine dreidimensionale Zeichnung, sprich ein 3D-Modell von ihnen anfertigen. Programme welche dies ermöglichen heissen CAD-(Computer Aided Design)-Programme. Ein bekanntes und benutzerfreundliches Beispiel wäre die Software TinkerCAD. Das 3D-Modell darf zwar hohl sein, es muss aber über eine geschlossene lückenlose Aussenwand verfügen. Tut es dies nicht, dann funktioniert der nächste Schritt nicht.
2. Das gezeichnete oder heruntergeladene 3D-Modell in eine sogenannte Slicing Software laden. Das Dateiformat ist bei diesem Schritt noch STL. Dort kann das Modell noch einmal verformt werden, meist durch Ska-

lierung, Rotation oder Vervielfachung. Des Weiteren können eine Vielzahl von Drucker-Einstellungen konfiguriert werden, angefangen bei der Temperatur des Druckkopfes und der Heizplatte, über die Dichte und Auflösung des zu druckenden Objektes bis hin zur Geschwindigkeit des Ventilators, der den Druckkopf kühlt, oder dem Höchstwinkel, bis zu welchem die Objekte geneigt sein dürfen, bevor Stützstrukturen mitgedruckt werden.

Die wichtigste Aufgabe dieses Programmes ist es, das Modell, nachdem es zufriedenstellend platziert, skaliert und rotiert wurde und sämtliche Einstellungen angepasst wurden, zu "slicen". Zu Deutsch: Das Programm fertigt feine Scheiben, Querschnitte vom 3D-Modell an. Mittlerweilen liegt die im Format .gcode vor. Dieses Format wird oft im Bereich des CAM (Computer Aided Manufacturing) verwendet, wenn einer Maschine Informationen gegeben werden müssen, wie sie einen bestimmten Herstellungsprozess auszuführen hat.

3. Das geslicete Modell per SD-Karte, USB oder WLAN an den Drucker übergeben, welcher dann im Verlauf von Minuten, Stunden oder Tagen Schicht für Schicht aufträgt bis das gewünschte Modell fertig ist.

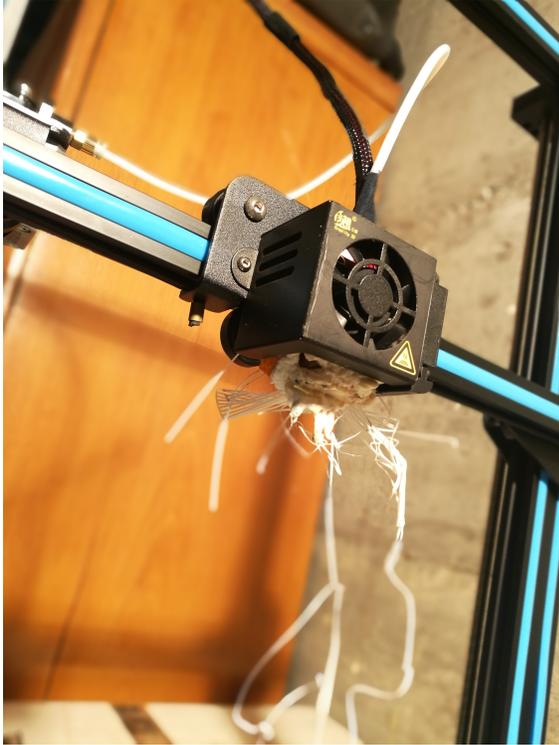
Das Slicing Programm, welches ich verwende, ist ein relativ bekanntes und eher einfach zu bedienendes namens Cura. Es gibt etwa fünfhundert Variablen, welche man konfigurieren kann. Jede einzelne beeinflusst den Druckprozess und die Qualität des Ergebnisses. Das Erstellen eines 3D-Modells ist eine Wissenschaft für sich, mit welcher ich bei dieser Arbeit nur am Rande zu tun hatte. Wie oben bereits erwähnt, hat der Entwickler des Projektes InMoov alle STL-Files hochgeladen, weshalb ich den Design-Teil weitestgehend überspringen konnte.

Einmal nur nutzte ich die Software "Blender", um ein 3D-Modell aller Fingerglieder in seine Einzelteile zu zerschneiden, um nicht für das Drucken einer einzelnen Fingerkuppe den ganzen Finger produzieren zu müssen. Blender ist aber kein typisches CAD-Programm, da der Gestaltungsprozess ein destruktiver ist, das heisst, im Regelfall hat man ein Objekt vor sich, an welchem

man Änderungen vornimmt, während bei klassischen CAD-Programmen eine Skizze konstruktiv ausgebaut wird.

Dieser einfache Eingriff demonstrierte mir bereits die Komplexität solcher CAD-Programme und bestätigte mich in der Entscheidung, mich gänzlich auf die verfügbaren Modelle zu beschränken.

Bis ich aber in der Lage war, Bauteile am Laufband zu produzieren, verging circa ein Monat und musste sehr viel Filament weggeworfen werden. Das Problem beim Drucken ist, dass es in manchen Fällen nicht offensichtlich ist, was schiefgelaufen ist. Meistens fällt einem bei einem Kontrollgang auf, dass das Filament kreuz und quer durch den Raum gedruckt wird und überhaupt keine erkennbare Form am Entstehen ist. Mindestens zwei Mal entdeckte ich nach mehr als zwanzig, einmal gar vierzig Stunden Druckzeit, dass eine Verschiebung der Druckplatte von wenigen Millimetern dazu geführt hatte, dass sich die Schicht-für-Schicht-Auftragung auf fünfzehn Zentimetern Höhe ebenfalls verschoben hatte und die somit deformierten Teile unbrauchbar waren. Drei Male bis jetzt geschah es, dass ich den Drucker qualmend vorfand und ein beissender Rauch ankündigte, dass wieder einige Stunden "Defilamentieren" auf mich warteten. (Abbildung 5 und 6).



*Abbildung 5: Filamentstau im Druckkopfgehäuse 1*



*Abbildung 6: Filamentstau im Druckkopfgehäuse 2*

Das ist ein frei erfundener Begriff für eine mühsame Tätigkeit, welche auf solche Unglücke folgte. Die erstgenannte Problematik der willkürlichen Filament-Extrusion ist nämlich nur das Anfangsstadium des Problems. Das Filament beginnt schnell sich zu stauen und das hohle Druckkopfgehäuse auszufüllen, in welchem sich aber noch der 250° Celsius heiße Druckkopf befindet. Dieser schmilzt das sich kontinuierlich aufstauende Filament, so dass ein wachsender Klumpen entsteht, welcher das Metallgehäuse, die Kunststoffkabelschläuche, den Druckkopf und die freiliegenden Drähte zu einer einzigen Masse fusioniert. Diese Masse zu “defilamentieren” ist stets ein zeit- und materialintensiver Prozess, welcher nach zweifacher Wiederholung bereits das Ersetzen des gesamten Druckkopfgehäuses inkl. Druckkopf forderte.

Ein weiteres Thema ist die Adhäsion, also das Mass, in welchem das Filament an der Druckplatte kleben bleibt, und welche hauptsächlich von den folgenden Faktoren abhängig ist.

Faktor	Beschreibung
Nähe des Druckkopfes zur Druckplatte	0.06 – 0.2mm bzw. Die Dicke eines 80g DIN A4 Papiers wird oft als Richtwert genannt.
Initialtemperatur dieser beiden	Je höher die Temperaturen desto besser die Haftung des Filamentes am Druckbett.
Initialgeschwindigkeit des Druckkopfes	Je schneller der Druckkopf desto weniger Zeit hat das Filament, um am Druckbett haften zu bleiben.
Adhäsionstyp	Verschiedene Typen haben unterschiedlichen Einfluss auf die Adhäsion, das “Raft” verstärkt die Adhäsion während bspw. der Brim lediglich die Extrusion testet.
Art der Stützstrukturen	Wenn ein Teil des Druckobjektes mehr als 45° überhängt, kann es ohne Stützstrukturen abbrechen/absacken und das Objekt kippen.
Oberflächenbeschaffenheit des Druckbettes	Leichtes Abschleifen des Glases kann die Haftung der Druckobjekte verstärken.
Druckbett-Material	Eine sogenannte Polyetherimidplatte zwischen Glas und Druckobjekt kann die Haftung stark verbessern.
Adhäsionsverstärkende Mittel	Haarspray, Leimstift, Printafix und weitere Stoffe können, wenn auch mit Putzaufwand verbunden, die Haftung verstärken.

Initialgeschwindigkeit und -Temperatur beziehen sich auf den Zeitraum, in welchem die erste Schicht gedruckt wird, und der Typus bezeichnet die Form und Zusammensetzung einer fakultativen Stützplatte, welche zwischen dem Druckobjekt und der Druckplatte gedruckt wird, um den Zusammenhalt derselben zu fördern. Jeder der schon einmal mit 3D-Druckern gearbeitet hat, kann bestätigen, dass der Produktionserfolg jedes noch so kleinen Objektes hauptsächlich mit der Adhäsion steht oder fällt. Die erhöhte Adhäsion im Vergleich zu ABS ist, nebst bereits erwähntem, ein gewichtiger Grund wieso ich es bevorzuge, mit PLA zu drucken.

## 2.2 Die elektronischen Steuerungseinheiten

### Arduino

Arduino ist der Name eines Open-Source Projektes welches hauptsächlich den gleichnamigen Mikrocontroller und die Entwicklungsumgebung *Arduino-IDE*<sup>9</sup> beinhaltet. Sinn und Zweck dahinter ist es, Nutzern, welchen sich dieser sonst schwer gestaltet, Zugang zum Programmieren und zu einem elektrotechnischen Basiswissen zu verschaffen . Wenn ich in dieser Arbeit den Begriff “Der Arduino” verwende, beziehe ich mich stets falls nicht anders deklariert auf den Arduino in seiner Funktion als Mikrocontroller. Die zentrale Hardware des Arduinos besteht aus einer Platine, auf welcher sich ein Schaltkreis befindet, welcher unter anderem über einen Mikroprozessor, Arbeitsspeicher, mehrere In- und Output Stellen und einen Reset-Knopf verfügt. Klassischerweise wird der Programmcode über eine USB-Schnittstelle auf das Arduino-board geladen.

---

<sup>9</sup>IDE steht für Integrated Development Environment und ist eine Bezeichnung für die Software welche verwendet wird, um Programmcodes zu schreiben. Oft enthalten solche Programme Hilfestellungen, z.B. automatische Fehlersuche oder Wortvorschläge.

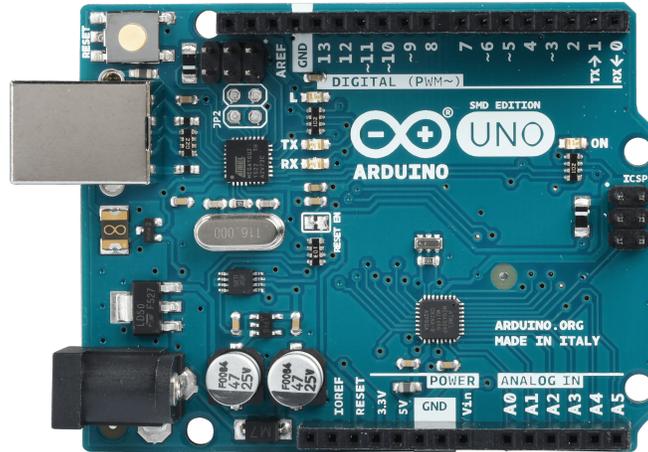


Abbildung 7: Arduino Uno Mikrocontroller

Anschlüsse wie Ethernet, USB und HDMI sind durchaus auch mit einem Arduino<sup>10</sup> Board realisierbar, einfach über Umwege. Eine beliebte Gattung von Umwegen sind die sogenannten Shields, also Platinen, deren Schaltkreise so ausgerichtet sind, dass man die ganze Platine auf den Arduino aufstecken kann. In den aufgesteckten Schaltkreisen können alle möglichen Ausgänge oder Eingänge integriert sein. Neben der Erweiterbarkeit bietet Arduino eine starke und hilfsbereite Community, in deren Foren man auf zahllose Fragen Antworten erhält, und, fast noch wichtiger, gibt es für ein weites Spektrum an Problemstellungen bezüglich Anwendungen mit Arduino bereits ein Projekt mit Schaltkreisplan und Code-Exempel.

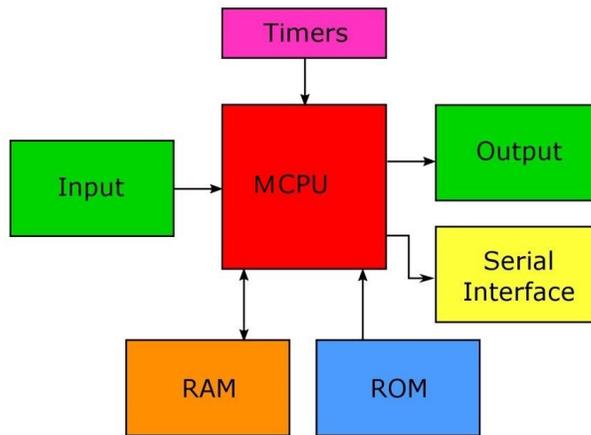
## RPP

Als ich das Projekt Arduino beschrieb, verwendete ich die Begriffe Mikrocontroller und Mikro-Prozessor. Der wesentliche Unterschied zwischen den beiden ist<sup>11</sup>, dass es sich beim Mikrocontroller (oder *MCU*) um eine Ansammlung elektrotechnischer Komponenten auf dem gleichen Chip handelt und der Mikroprozessor (oder *MPU*) einer dieser Komponenten ist[8][9].

<sup>10</sup>Bild Quelle: [https://cdn-reichert.de/bilder/web/xxl.ws/B300/ARDUINO\\_UNO\\_A06.png](https://cdn-reichert.de/bilder/web/xxl.ws/B300/ARDUINO_UNO_A06.png)

<sup>11</sup>Sehr stark vereinfacht

Mikroprozessor sitzt auf eigenem Chip, kann aber mit anderen Komponenten interagieren



Mikrocontroller beinhaltet einen Mikroprozessor



Abbildung 8: Unterschied zwischen Mikrocontroller und Mikroprozessor

Ein Mikroprozessor<sup>12</sup>, ist wie es der Name sagt, ein kleiner Prozessor, also die Komponente auf einem Mikrocontroller, welcher den darauf geladenen Programm-Code exekutiert und die darin enthaltenen Berechnungen anstellt.<sup>13</sup> Der Mikrocontroller hingegen *beinhaltet* einen Mikroprozessor.

Während der Arduino das Zusammenspiel von Sensoren, Motoren und einfacher Programmierung orchestrieren kann ist ein Raspberry Pi, oder kurz RPP, ein vollwertiger kleiner Computer, dessen klassische Ausführungen über einen HDMI-Ausgang, 4 USB-Anschlüsse, einen analogen Audio-Output, einen MicroSD-Karten-Slot, eine Stromzufuhr und einen Ethernet-Ausgang verfügt.

Um den Raspberry Pi in Betrieb zu nehmen, kann man eine Tastatur, eine Maus und einen Monitor anschliessen und das Betriebssystem darauf laden. Bezüglich des Betriebesystems gibt es verschiedene Möglichkeiten, wie etwa eine Variante von Windows 10; ich entschied mich jedoch für das von der RPP-Community empfohlene Linux-basierte “NOOBS”, eine vereinfachte Form des Standards “Raspberrian”.

<sup>12</sup>Quelle Bild: <https://www.microcontrollertips.com/microcontrollers-vs-microprocessors-whats-difference/>

<sup>13</sup>stark vereinfacht

Aus den folgenden Gründen kann der RPP als potenter als der Arduino beschrieben werden. Zum einen erlaubt das Betriebssystem eine effiziente Datenverwaltung und zugleich ermöglicht die Google-Chrome Variante “Chromium” den Zugang ins Internet . Gleichzeitig ist es aber auch möglich, den RPP als Mikrocontroller zu verwenden, in dem von den 40 *GPIO*<sup>14</sup>-*Pins*<sup>15</sup> Gebrauch gemacht wird.

## 2.3 Biometrische Datenerfassung

### Messung der Hirnaktivität

Die Technologie, welcher dieser Messung zugrunde liegt, ist die Elektroenzephalografie, kurz EEG. Ein EEG-Messgerät misst mit einer Auflösung von mehreren Zentimetern die elektrischen Veränderungen, welche bei der Informationsverarbeitung in einem Gehirn entstehen. Jedes Neuron kann, wird es angeregt oder gehemmt, diese Information über seine Zustandsänderung als Aktionspotential<sup>16</sup> über kleine Verästelungen an das nächste Neuron weitergeben. Somit entsteht ein Netzwerk elektrischer Signale.

Das EEG-Messgerät kann dessen Intensität messen bzw. bestimmen, an welchen Punkten, auf wenige Zentimeter genau, wie viel elektrische Aktivität herrscht. Diese Signale, welche durch die Neuronen, genaugenommen ihren langen Vorsätzen, den Axonen entlangrauschen, tun dies in Rhythmen. Diese Rhythmen können sehr unterschiedlichen ausfallen, sie variieren je nach Art der Aktivität, welche das Gehirn oder die betrachteten Regionen darin gerade ausführen. Wenn man bei diesen Rhythmen die Frequenz und die Amplitude betrachtet, so kann man daraus eine Welle interpretieren.

Die kalifornische Firma “NeuroSky” verkauft unter anderem ein Produkt namens “MindWave Mobile”. Dieses Gerät, welches gewisse Ähnlichkeit mit einer Stirnlampe aufweist, soll ein mobiles EEG-Messgerät sein. Im folgenden werde ich unter dem Namen Headset darauf Bezug nehmen.

---

<sup>14</sup>GPIO steht für “General Purpose Input/Output” und lässt den Nutzer wie es der Name verspricht über 40 Pins Signale empfangen oder schreiben.

<sup>15</sup>Pins sind meist kleine Stäbe aus einem leitenden Material, welche entweder ausgelesen oder auf welche geschrieben werden kann.

<sup>16</sup>Ein Aktionspotential ist Teil der Signalübertragung zwischen zwei Neuronen. Die Information, die es überträgt, ist entweder Erhöhung oder Verminderung des Potentials des Signals, weitergeschickt zu werden.



*Abbildung 9: Elektroenzephalografie Messgerät "MindWave Mobile" der Firma "NeuroSky"*

In einer kleinen Schnalle, welche am linken Ohrläppchen angeklippt wird, befindet sich eine Elektrode, sprich ein Sensor, welcher elektrische Signale messen kann. Das Gegenstück dazu befindet sich in einem vorstehenden Kunststoffstück, welches oberhalb der linken Augenbraue befestigt wird. Diese zwei Elektroden sind nun in der Lage, die elektrische Aktivität im Hirn zwischen ihnen zu messen.

Wie weiter oben beschrieben, kann man die Rhythmen, in welchen die elektrischen Signale auftreten, als Wellen interpretieren. Beim EEG werden die typischsten Wellen mit den Lettern des griechischen Alphabetes klassifiziert, genauer gesagt mit Alpha, Beta, Gamma, Delta und Theta. Delta-Wellen werden im traumlosen Tiefschlaf gemessen und haben eine tiefe Frequenz von 0-3 Hertz, das heisst, dass bei der Betrachtung des Signalrhythmus gar kein oder nur drei Signale pro Sekunde beobachtet werden können. Den Gegensatz dazu bilden die Gamma-Wellen, welche Frequenzen zwischen 24 und 100 HZ aufweisen und mit erhöhter Aufmerksamkeit und Konzentration gleichgesetzt werden. Die Messwerte dieser Frequenzen und somit der Wellen können in einer Software von NeuroSky angezeigt werden, wie in Abbildung 10 zu sehen ist[4].

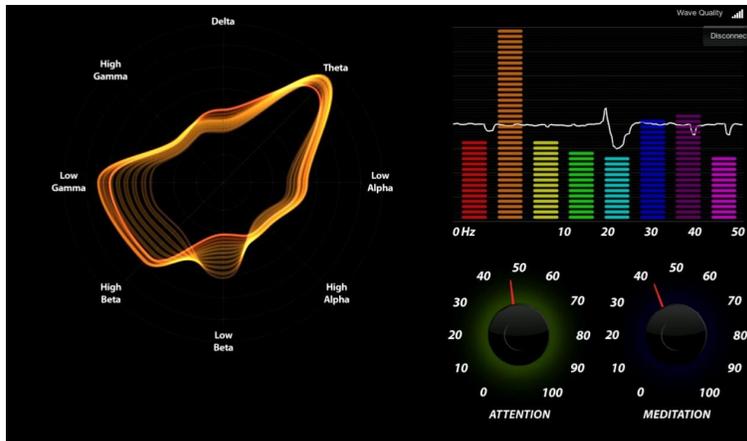


Abbildung 10: Software von NeuroSky welche durch EEG gemessene Hirnwellenaktivität graphisch repräsentieren soll

Das zuletzt genannte Programm<sup>17</sup> stellt die Intensität der verschiedenen Hirnwellen als sternförmige Animation dar und erhebt somit Anspruch darauf, Aussagen über deren tatsächliches Erscheinen im Gehirn des Nutzers treffen zu können. Ich musste feststellen, dass die graphische Darstellung meiner Hirnaktivität in den allermeisten Fällen wenig bis keinen Sinn ergab. Das einzige, das im Programm zuverlässig abgelesen werden konnte, war der Grad meiner mentalen Angespanntheit. Entspannte ich mich beispielsweise mehrere Minuten lang gedanklich oder zählte von 100 rückwärts bis 0, war in den meisten Fällen im Programm zu lesen, dass mein “Aufmerksamkeitswert” tatsächlich relativ niedrig war. Löste ich dann hingegen eine herausfordernde Matheaufgabe oder versuchte ich mich krampfhaft an etwas zu erinnern, stieg der Aufmerksamkeitswert zuverlässig auf ein vergleichbar hohes Niveau. Der Aufmerksamkeitswert ist ein quantitativer Ausdruck für die Anzahl Aktionspotentiale, welche zwischen den beiden Elektroden auftreten.

Die entscheidende Beobachtung, die meine Skepsis gegenüber der Relevanz der Outputs der oben genannten Hirnwellen schürte, war jene, dass, als ich das Headset für kurze Zeit ablegte, die verschiedenen Hirnwellenwerte trotzdem mehrere Minuten lang willkürlich rauschten. Dies war im Endeffekt aber irrelevant, da ich mir als Ziel setzte, nur die Veränderung der gesamten elektrischen Signalintensität als Output zu verwenden.<sup>18</sup>

<sup>17</sup>Bild Quelle:<https://www.futurashop.it/neurosky-MindWave-mobile-6168-MindWave>

<sup>18</sup>Kurz gesagt stellte ich mir vor, dass ich beispielsweise eine LED-Lampe im Inneren eines Auges des Roboters

## Messung der Muskelaktivität

Analog zur Elektroenzephalographie misst auch die Elektromyographie (EMG) die elektrischen Signale in einem Gewebe. Im Unterschied zum EEG, welches diese Messung am Hirn vornimmt konzentriert sich das EMG auf Muskeln. (griechisch: Enzephalon - Gehirn, Myo - Muskel). Dabei können auch Aussagen darüber getroffen werden, an welcher Stelle eines Muskels wie viel Aktivität herrscht. Kurz, verschiedene Muskelzustände sprich Bewegungen oder Positionen führen zu unterschiedlichen Signalen.

Wie NeuroSky führt auch die Firma “Thalmic Labs” ein Angebot im Sortiment, welches das Lesen von biometrischen Daten für den Hobby-Gebrauch ermöglichen soll. Das “Myo Gesture Control”- Armband macht sich das EMG zunutze, um dem Nutzer verschiedene Interaktionen mit einer weiten Bandbreite von Geräten zu erlauben. Das Armband kann nach einer gewissen Trainingszeit fünf Gesten des Trägers unterscheiden, wie z.B. das Ballen der Hand zu einer Faust oder das Zusammendrücken von Daumen und Zeigefinger. Mit diesen Gesten kann der Nutzer dann etwa in der Musikplaylist des Smartphones navigieren oder, dank dem Gyroskop (Orientations- und Beschleunigungssensor), den Cursor auf seinem Bildschirm bewegen.

---

anschalten könnte, indem ich mich entweder mental an- oder entspanne.

## 3 Verschiedene Möglichkeiten der Ansteuerung des Armes

### 3.1 Per Gedankenkraft

Die *Signalkette*<sup>19</sup>, welche ich mit der Benutzung des MindWave Mobiles hauptsächlich anvisierte, hätte folgendermassen aussehen sollen. Das Ursprungssignal entsteht im Hirn und kann beeinflusst werden, indem der Träger des Headsets sich aktiv mental entspannt oder fokussiert. Die reduzierte oder erhöhte neuronale Aktivität vermindert die Anzahl an Aktionspotentialen, welche zwischen der Stirn und dem linken Ohrläppchen (die beiden Orte an welchen Elektroden platziert wurden) des Trägers stattfinden, was in einer Messung tieferer oder höherer Aufmerksamkeitswerte resultiert.

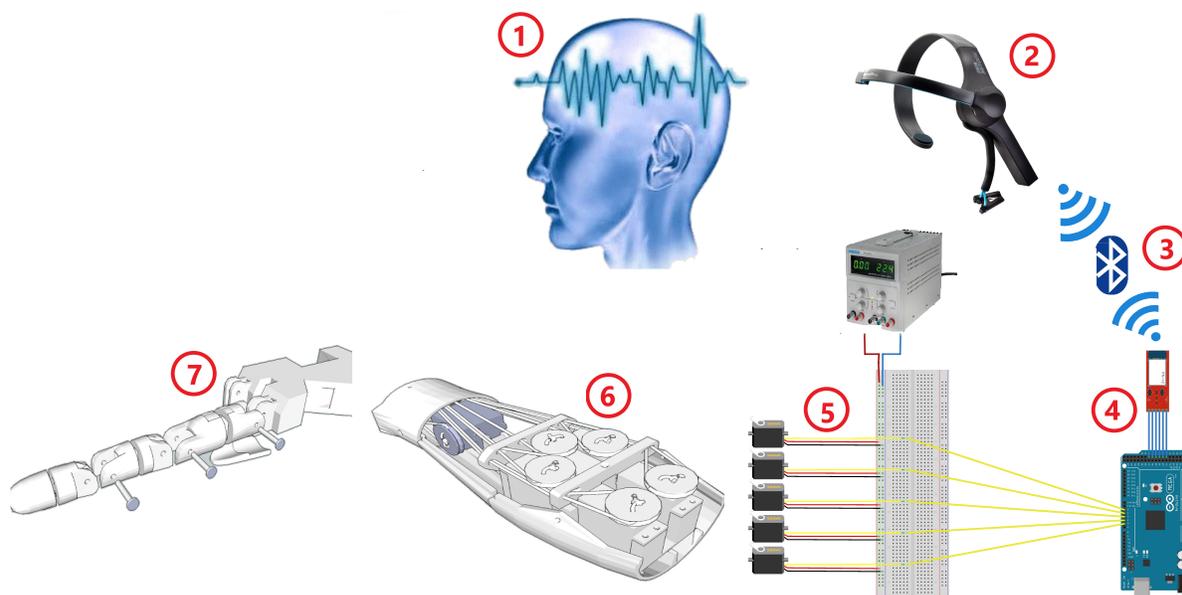


Abbildung 11: Schema der Signalübertragung über das MindWave Mobile. 1) Ursprungssignal: Unterschiedliche Grade mentalen Fokus resultieren in unterschiedlichen Hirnwellen, 2.) MindWave Mobile misst über zwei Elektroden die Hirnwellen, 3.) Das MindWave sendet die Information der Hirnwelle per Bluetooth an das Bluetooth Modul RN-42, 4.) Das Bluetooth-Modul leitet die Information an das Arduino-Board weiter, 5.) Der Code auf dem Arduino-Board gibt den Befehl an die Servos weiter, 6.) Die Servos spannen/entspannen die Nylonfäden im Arm, 7.) Die Nylonfäden ziehen den Finger an oder rollen ihn ein.

<sup>19</sup>Weg, welchen eine Information oder ein Signal zwischen seinem Ursprung und seiner finalen Destination zurücklegt.

Dieser Messwert wird dann per Bluetooth an den Mikrocontroller Arduino geschickt, welcher die eingehenden Signale berechnet und dann den Anweisungen des geladenen Codes folgend elektrische Signale an die ihm angeschlossenen Servos schickt. Die Servos spannen bzw. entspannen die Nylon-Sehnen im InMoov-Arm oder initiieren über ein Zahnradwerk die Rotation des Handgelenkes .

Das Ursprungssignal entsteht also im Hirn, wandert über verschiedene Etappen zur Endstation, dem Servo. Diese Signalkette scheiterte aber an der Kommunikation zwischen dem Headset und dem Arduino. Genaueres kann im Anhang unter dem Punkt A nachgelesen werden.

## 3.2 Per PS4-Controller

Für die Signalübertragung zwischen dem PS4-Controller und dem RPP verwendete ich *Treiber*<sup>20</sup> und Bibliotheken aus der Code-Sammlung eines GitHub-Beitrages namens DS4DRV.[10]Darin enthalten war ein Python-Code, welcher nach der erfolgreichen Implementierung der Kommunikationsprotokolle zwischen RPP und PS4-Controller die 14 Achsen des PS4-Controllers darstellte und mit einem Wert zwischen -1 und 1 versah.

Der PS4-Controller besitzt unter anderem zwei Joysticks und zweimal 8 Knöpfe, welche die Joysticks in Vierergrüppchen links und rechts aussen flankieren. Die Knopfwerte wurden als 0 oder 1 angegeben, je nach Zustand des Knopfes. Ein Joystick wurde auf zwei Achsen aufgeteilt, das heisst vorne/-mitte/hinten entsprach den Werten -1 / 0 / 1, ebenso wie links/mitte/rechts einer der Wert -1 / 0 / 1 annehmen konnte.

Den Anfangspunkt der Signalkette bildet nun der PS4-Controller, auf welchem durch die Manipulation von Tasten und Joysticks 14 Werte entsendet werden können. Das Signal wird per Bluetooth auf den RPP übertragen, wo es vom Python-Code ausgelesen wird.

---

<sup>20</sup>Ein Treiber ist vereinfacht gesagt ein Programm auf einem System, das für die Befehlsweiterleitung an ein angeschlossenes Modul verantwortlich ist, in diesem Fall die Bluetooth-Schnittstelle am RPP.

Dieser Code teilt dem Nutzer einerseits die Zustandsveränderungen der 14 aufgefangenen Achsen-Werte mit und übersetzt bzw. schreibt diese andererseits in digitale Outputs für die GPIO-Pins.

Um diese zuletzt genannten Funktionen ausführen zu können, muss der Pin jedoch vorher als Input beziehungsweise Output deklariert werden. Beides kann entweder analog oder digital geschehen. Grob gesagt bedeutet digital in diesem Kontext, dass der Wert entweder AN oder AUS bzw. HIGH oder LOW annehmen kann, während bei analog eine theoretisch unendliche Anzahl an Werten innerhalb eines Bereiches angenommen werden kann.

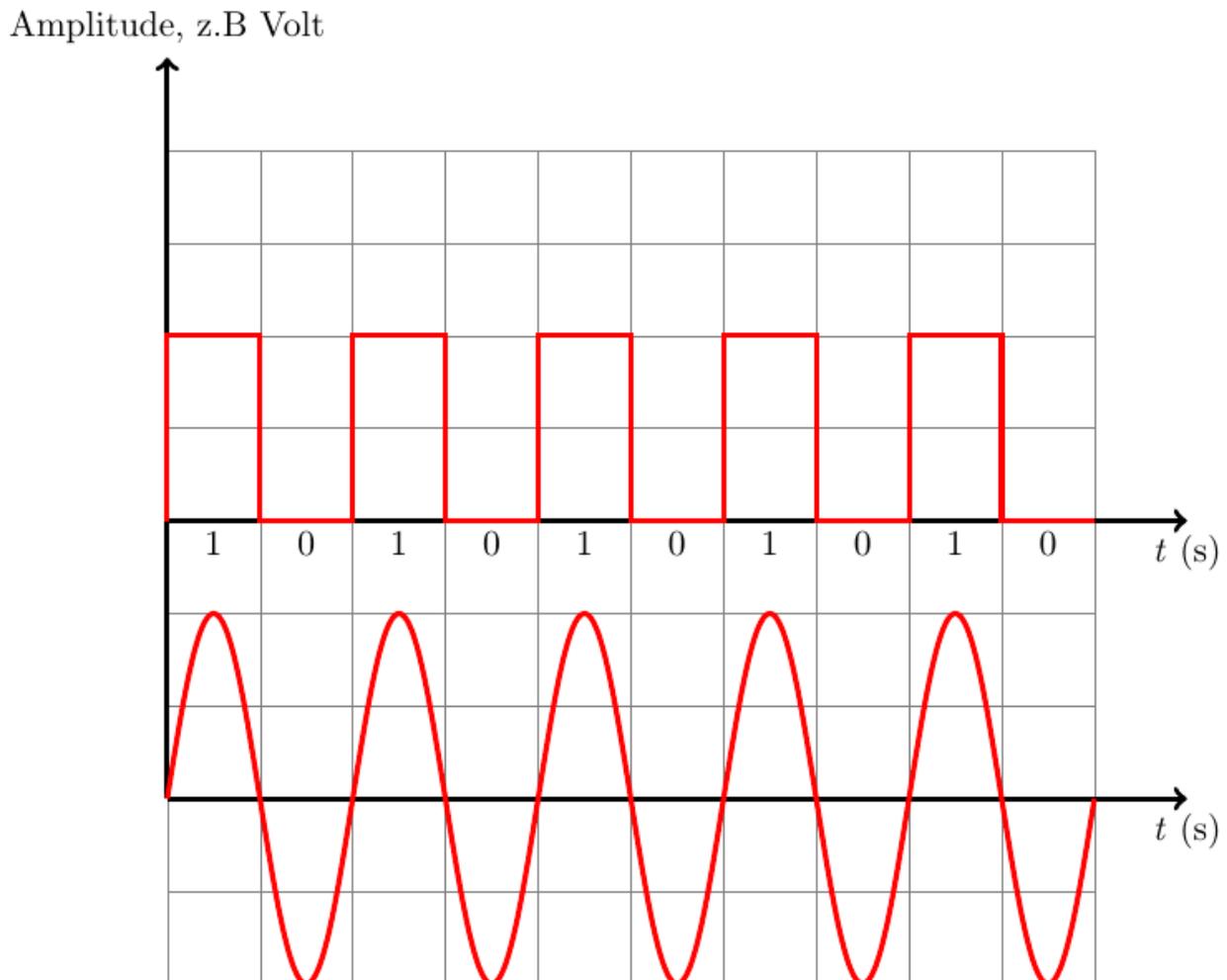


Abbildung 12: Sehr stark vereinfachter Graph eines digitalen und eines analogen Signals, über eine Zeit  $t$ (s) und mit einer Amplitude  $y$ . Handelte es sich um ein Signal von oder zu einem Arduino-GPIO-Pin, wäre die Amplitude in Volt angegeben.

Diese Pins leiten das Signal über ein Jumper-Wire, entweder über den Umweg einer *Steckplatine*<sup>21</sup> oder direkt an das Arduino-Board weiter, welches die Signale dann in seinem eigenen Code interpretiert und danach die ausgerechneten Befehle an die Servos weitergibt, die zu guter Letzt die Gliedmassen bewegen.

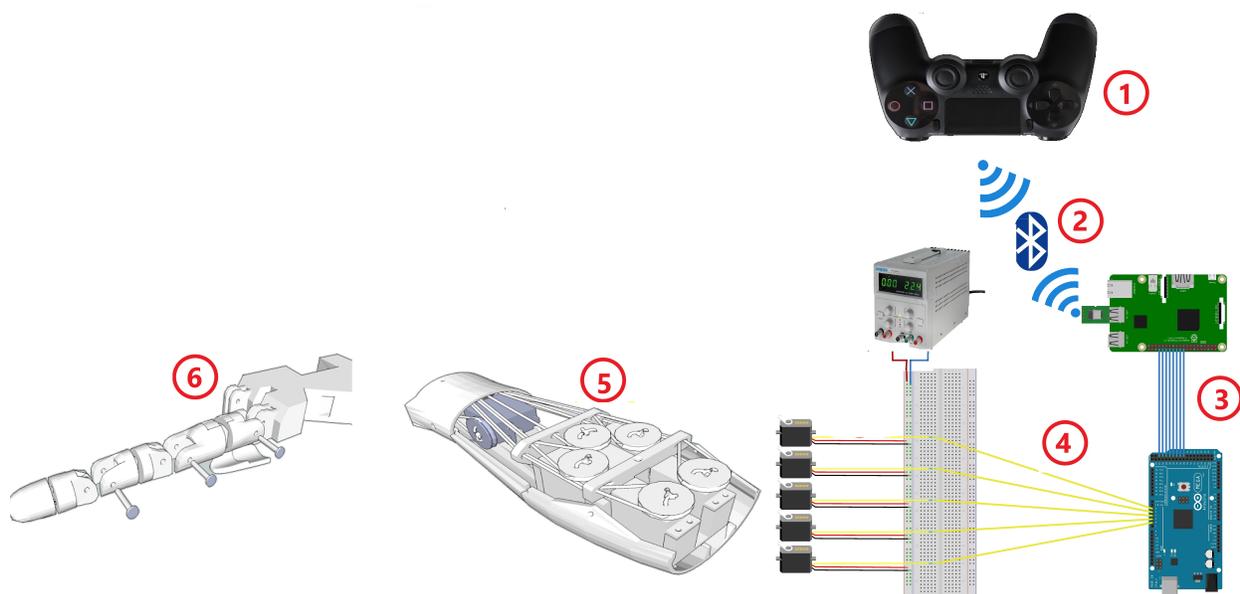


Abbildung 13: Schema der Signalübertragung über den PS4-Controller, 1.) Ursprungssignal: 14 verschiedene Manipulationen am Controller können zu 14 verschiedenen Signalen führen, 2.) PS4-Controller sendet Achsenwerte per Bluetooth an den RPP, 3.) Der RPP sendet die Information der Achsenwerte über das Flachbandkabel an das Arduino-Board, 4.) Der Code auf dem Arduino-Board gibt den Befehl an die Servos weiter, 5.) Die Servos spannen/entspannen die Nylonfäden im Arm, 6.) Die Nylonfäden ziehen den Finger an oder rollen ihn ein.

Nachdem erste Tests mit nur einem Knopf gelangen, liessen sich schlussendlich 8 Knöpfe anschliessen, deren Zustandsänderung bei 8 verschiedenen GPIO Pins einen Stromfluss zur Folge hatte. Zur Erinnerung, ein GPIO Pin ist in diesem Fall ein Pin am RPP, der (General Purpose) Input oder Output sein kann.

<sup>21</sup>Eine Steckplatine unterscheidet sich insofern von den "allgemeinen" Platinen, als dass die verschiedenen Elemente nicht angelötet werden müssen sondern, wie der Name es sagt in die Platine hereingesteckt werden können

Somit war es möglich, zum einen die fünf Finger, zum anderen die Handgelenkrotation des InMoov-Arms und zusätzlich zwei Gesten an die Knöpfe zu koppeln, welche in Abbildung 14 zu sehen sind<sup>22</sup>. Die verschiedenen Verzögerungen, mit welcher die einzelnen Signaletappen stattfanden, akkumulierten sich aber noch zu einer Verzögerung von ca. 0.75-1 Sekunden. Optimierungsansätze finden sich im Anhang.

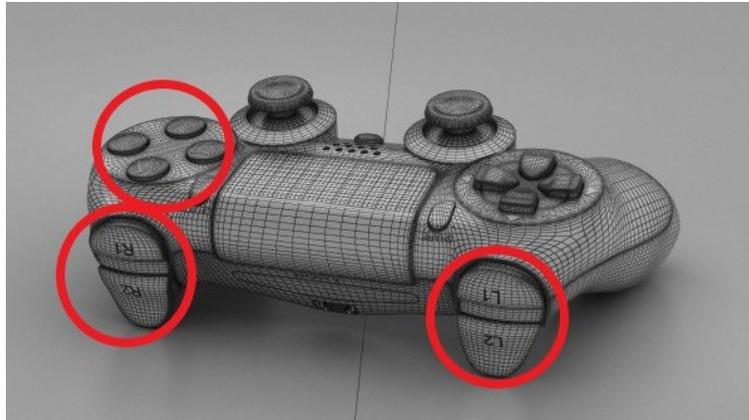


Abbildung 14: Sony Dualshock Controller 4, die 8 rot umkreisten Knöpfe sind an Funktionen des Armes gekoppelt

### 3.3 Per Bewegungsübertragung

Die Signalkette, welche ich mit Hilfe des Myo Gesture Control Armbandes - welches ich im Folgenden als Myo bezeichne - erstellte, glich der des MindWaves bis auf wenige Unterschiede. Das Signal hat seinen Ursprung im Motor Cortex des Gehirns. Dies ist die Region im Hirn, in welcher motorische Abläufe, sprich Bewegungen entstehen.

Dieses Signal gelangt durch ein Motoneuron an die Muskeln an meinem Unterarm. Einige Zentimeter unter dem Ellbogen sitzt das Myo-Armband und misst per EMG diese Signale. Im nächsten Schritt wird das Signal an einen Bluetooth-“Dongle” weitergegeben, welchen man über eine USB-Schnittstelle in das zu kontrollierende Gerät einstecken kann. Dieser Dongle ist im Wesentlichen ein vereinfachter Zugang zum Bluetooth Signal, da die oben ge-

---

<sup>22</sup>Quelle: <https://hum3d.com/>

nannten Konfigurationen darauf bereits eingestellt sind, was in einem Plug-and-Play-Verfahren resultiert. (Plug-and-Play: Einstecken und ohne komplizierteres Einstellen benutzen).

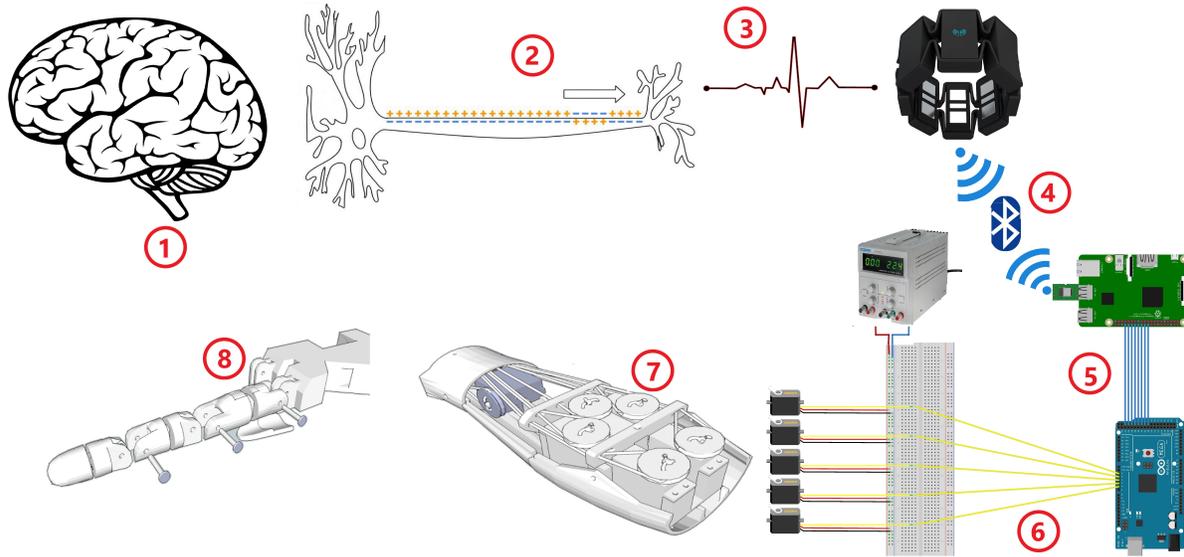


Abbildung 15: Schema der Signalübertragung über das Myo-Armband, 1.) Ursprungssignal: Befehl, die Handgeste zu verändern entsteht im Motor Cortex, 2.) Signal wird von Motoneuronen vom Gehirn zum Muskel übermittelt, 3.) Myo-Armband misst per EMG die elektrischen Veränderungen am Muskel, 4.) Myo kommuniziert per Bluetooth Signal an RPP, 5.) Der RPP sendet die Information der Muskelveränderung an das Arduino-Board, 6.) Der Code auf dem Arduino-Board gibt den Befehl an die Servos weiter, 7.) Die Servos spannen/entspannen die Nylonfäden im Arm, 8.) Die Nylonfäden ziehen den Finger an oder rollen ihn ein.

Leider besitzt das Arduino Board keine USB-Schnittstelle. Zur Umgehung dieses Engpasses ersann ich verschiedene Lösungen. Zum einen hoffte ich, durch das Anstecken eines Arduino-USB-Shields den Bluetooth-Dongle an das System anschliessen zu können. Einen USB-Shield kann man als Schaltkreis auf einer Platine betrachten, in welchem eine USB-Schnittstelle integriert ist und welchen man auf den Arduino draufstecken kann, womit man die Schaltkreise fusioniert.

Das Problem dabei wäre, dass das Verfahren eben doch nicht hundert Prozent Plug-and-Play ist, da man das Myo zuerst noch kalibrieren muss, ihm vereinfacht gesagt die Möglichkeit geben muss, sich an den individuellen Trägerarm anzupassen. Dieser Prozess erfordert das Öffnen eines Kalibrierungsfensters am Monitor des Gerätes, in welchem der Donggle steckt, was wiederum die Notwendigkeit eines Bildschirms nach sich zieht, ganz zu schweigen von einer Grafikkarte; beides Elemente, die sich über Umwege schon an den Arduino anschliessen lassen, aber weder billig noch einfach zu handhaben sind. Diese Feststellung machte mir klar, dass ich im Prinzip einen kleinen Computer benötige. Dies war der Moment, als ich begann, mich mit Raspberry Pi zu beschäftigen.

Ich recherchierte auf Github und anderen Informatik-Foren nach Projekten, in welchen die Kommunikationsprotokolle für die Bluetooth-Verbindung des Myos bereits vorhanden waren. Ich stiess auf die Code-Sammlung von Fernando Consetino[11], welcher auf seiner Website [fernandocosentino.net](http://fernandocosentino.net) eine Reihe von Codes zur Verfügung stellte. Diese beinhalteten die bereits erwähnten Kommunikationsprotokolle und auch mehrere Applikationen für den Gebrauch des Myo-Armbandes mit einem PC.

Fernando's Anliegen war es, den Applikationsmanager "MyoConnect", welcher von der Firma Thalmic Labs zur Installation gratis angeboten wird, aber nur für Windows und Mac OS verfügbar ist, auch auf die Plattform Linux zu bringen. Da das Betriebssystem des RPPs Linux-basiert ist, und Fernandos Code in Python geschrieben ist, eignete sich seine Code-Sammlung für den Einsatz in meiner geplanten Signalkette. Der Code dieses Applikationsmanagers ist wie folgt aufgeteilt. Es gibt einen Haupt-Code namens `myo_raw.py`. Dieser initiiert die Bluetooth-Kommunikation mit dem Myo und stellt die eingehenden Werte graphisch dar. Grundsätzlich können fünf verschiedene Posen (Handgesten) gelesen werden:

1. Ausstrecken der Finger
2. Ballen der Finger zur Faust
3. Pincetten-Griff
4. Neigen der Hand gegen aussen *und* gegen innen

Diese Posen werden aber nicht vom Myo als solche gelesen, sondern in der Software, in meinem Fall im Python-Code berechnet. Dazu werden die Werte genommen, welche an den Elektroden des Myo-Armbandes gemessen werden. Diese kommen wie oben erwähnt dadurch zu Stande, dass unterschiedliche Handgesten unterschiedlichen Muskelkontraktionen vorangehen.

Diese Variation der Muskelaktivität wird durch unterschiedliche Signale in den Motoneuronen ausgelöst. Diese Signale sind elektrisch und können vom EMG erfasst werden. In Fernandos Code ist die Möglichkeit enthalten, neben den genannten 5 Standard-Posen willkürliche Bewegungen der Hand aufzuzeichnen, zu benennen und dann zu verwenden. Konkret, kann man dem Programm sagen, es solle innert eines bestimmten Zeitraums, in welchem das Armband getragen wird, die Daten erfassen und als neue Pose interpretieren.

Wenn man dem Programm nur genügend Variationen dieser neuen Bewegung gibt, wird es diese irgendwann zuverlässig erkennen können. Zudem verfügt das Myo auch über ein Gyrometer, welches ihm erlaubt, Beschleunigungen im dreidimensionalen Raum zu erfassen und somit dem Programm mitzuteilen, in welche Richtung sich der der Arm des Trägers gerade bewegt. Kombiniert man nun diese beiden Möglichkeiten, steht einem ein weites Spektrum an möglichen Hand bzw. Armbewegungen zur Erzeugung von Signalen zur Verfügung.

Ich entschied mich dazu nur die fünf Standard-Posen zu wählen, alles andere hätte den zeitlichen Rahmen gesprengt. Neben dem Hauptcode `myo_raw.py` ist das zweitwichtigste Element in Fernandos Code-Sammlung der `PyoManager`, welcher, dem Vorbild der Originalsoftware folgend, die Anwendungen mit dem Myo zusammenfassend darstellen soll und die Möglichkeit bietet, die Anwendungen an- und abzuschalten.

Solch eine Anwendung könnte beispielsweise aus der Navigation des Maus-Cursors über den Bildschirm, dem Vor- und Zurückspulen in einem YouTube-Video oder der Steuerung eines fernsteuerbaren Autos bestehen. Im Unterschied zur Original-Software, bei welcher man zwei solcher Anwendungen fix vorgegeben hat, kann man die Anwendungen für diesen Manager frei programmieren und sie über den Manager einschalten. Der Versuch, solch eine

Mini-Applikation zu schreiben, welche die vom Myo eingehenden Signale digital auf die GPIO-Pins übersetzte, scheiterte daran, dass der PyoManager im Dateiformat .pyc (kompilierter Python-Code) vorlag und es nicht möglich war, ihn zu dekompileieren.

Doch gelang es, Code-Passagen in den Hauptcode zu schreiben, welche die eingehenden Gesten erkannten, jeder Geste einen GPIO-Pin zuzuweisen und die Information somit digital weitergeben zu können. Von den GPIO-Pins des RPPs führt ein Flachbandkabel die Signale zu einem Arduino Mega Board, wo digitale Input Pins diejenigen Eingänge auslesen, bei welchen ein Signal vom RPP ankommen und die Ausführung einer Geste signalisieren könnte. Der Code auf dem Arduino prüft kontinuierlich, ob dies der Fall ist und setzt die Information der Gestenänderung wiederum in Signale um, welche an die Servos geschickt werden, um die Geste auszuführen.

Die Gesten, welche das Myo am zuverlässigsten erkennen kann, sind der Pincetten-Griff und das Ballen der Hand zur Faust. Die Signalkette zwischen Motor Cortex und der InMoov Hand ist also hergestellt und funktioniert im Falle des Pincetten-Griffs und der Faust auch in etwa 7 von 10 Fällen. Auch bei dieser Ansteuerungsmethode gibt es eine Verzögerung der Signalübertragung von 0.5 Sekunden. Optimierungsansätze finden sich im Anhang.

## 4 Endprodukt und Anwendungsmöglichkeiten

### 4.1 Endprodukt

Zum Zeitpunkt der Abgabe dieser Arbeit umfasst das Endprodukt die folgenden Elemente. Im Zentrum steht der rechte Arm des InMoovs, welcher bis zum Schultergelenk konstruiert ist. Die Finger der rechten Hand lassen sich alle maximal einrollen auch bis zum Anschlagpunkt strecken, mit Ausnahme des Mittelfingers dessen Fingerkuppe noch leicht angezogen ist. Auch die Rotation des Handgelenkes funktioniert einwandfrei. Die bis jetzt genannten Gelenke sind allesamt mit Knöpfen des PS4-Controllers gekoppelt. Das Anziehen des Unterarmes hat als isolierte Bewegung schon funktioniert, wurde aber bis jetzt weder an einen PS4-Controller gekoppelt noch in Kombination mit anderen Bewegungen verwendet.

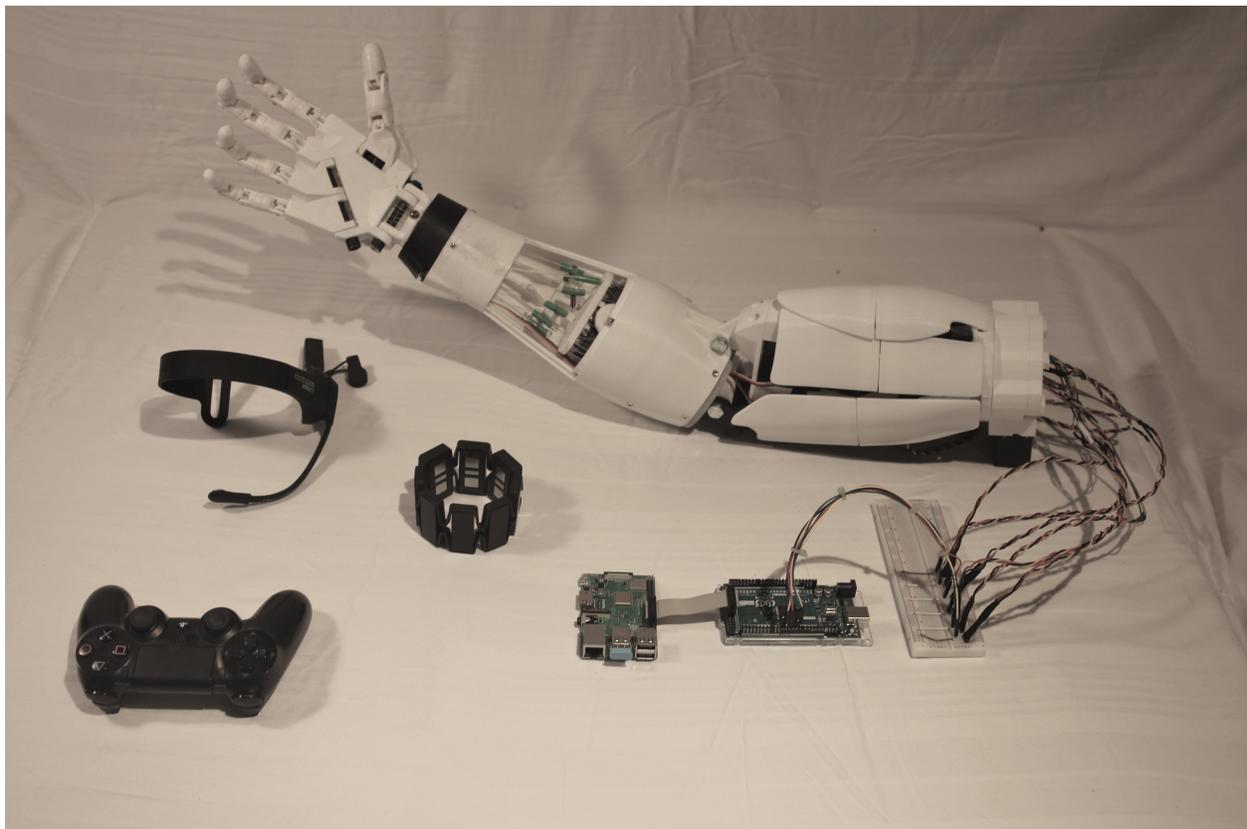


Abbildung 16: Endprodukt, bestehend aus dem rechten Arm des InMoovs und zwei funktionierenden Ansteuerungsarten.

Folgendes kann noch zum Ellbogengelenk gesagt werden: Der Servo, welcher den Unterarm über ein Gewinde anziehen kann, wurde von mir gemäss den Instruktionen auf `inmoov.fr` modifiziert. Ich löste den Potentiometer aus ihm heraus und platzierte ihn am Ellbogengelenk. Ein Potentiometer ist ein elektrotechnisches Bauelement, welches durch Drehen oder Verschieben seinen Wert verändern kann. Die Servos, welche ich verwende, haben eine Range von  $180^\circ$ . Der Rotor dreht aus zwei Gründen nicht um  $360^\circ$ . Erstens gibt es auf einem der Zahnräder in der Übersetzung zwischen dem Gleichstrommotor und dem Outputzahnrad eine mechanische Barriere, die ein Drehen der Übersetzung um mehr als  $180^\circ$  verhindert. Zweitens sorgt besagter Potentiometer dafür, dass bei jeder Veränderung des Drehwinkels ein Feedback gegeben wird, wodurch die Elektronik im Servo immer bestimmen kann, an welcher Position der Servo sich gerade befindet. Entfernt man nun aber diese mechanische Blockade und löst auch den Potentiometer heraus, kann sich der Servo um mehr als  $180^\circ$  drehen. Beziehungsweise ist es nun so, dass die maximale Drehzahl immer noch von der mechanischen Manipulation des Potentiometers abhängt, diese Manipulation aber durch einen Servo-externen Prozess stattfinden kann.

Der Potentiometer wird nun so am Ellbogengelenk installiert, dass er seine maximale Auslenkung erreicht sobald der InMoov-Ellbogen entweder maximal ausgestreckt oder maximal eingeknickt ist. Beschränkte sich die Maximalauslenkung des Potentiometers auf die  $180^\circ$  des Outputzahnrades, wäre es nicht möglich, das Gewinde, welches den Ellbogen ein- und ausfahren kann, genügend lange anzutreiben, um die durch die Architektur des Ellbogengelenkes gegebene maximale Beugung oder Streckung zu erreichen.

## 4.2 Anwendungsmöglichkeiten

Obwohl zwei der drei technischen Ansätze, welche ich in dieser Arbeit verfolgte, theoretisch funktionierten, handelt es sich bei dem Endprodukt um den Prototypen einer Idee. Im Kapitel “Perspektiven”, werde ich erläutern, wie die nächsten Schritte Hardware- und Softwaretechnisch aussehen könnten. Nun möchte ich aber die Anwendungen skizzieren, welche sich mit einer wei-

terentwickelten Version meiner Arbeit realisieren lassen könnten.

Es geht dabei massgeblich um die Prothesentechnologie. Sowohl der 3D-Druck, wie auch das Myo-Armband kommen in diesem Bereich zur Anwendung. 3D-gedruckte Prothesen sind vor allem eines: Sie befinden sich in einem ganz anderen Preissegment als Silikon-basierte Hightech-Prothesen. Ohne Krankenkasse, zahlen Patienten für eine myoelektrische technisch avancierte Unterarmprothese inklusive künstlicher Hand Preise von bis zu 450'000 US Dollar (Kosten über 5 Jahre)<sup>23</sup>. Natürlich gibt es auch bei 3D-gedruckten Prothesen ein Spektrum an Qualität und Funktionalität, doch grundsätzlich besitzen diese Prothesen einen hohen Grad an Flexibilität sowohl bei der Planung wie auch der Umsetzung.

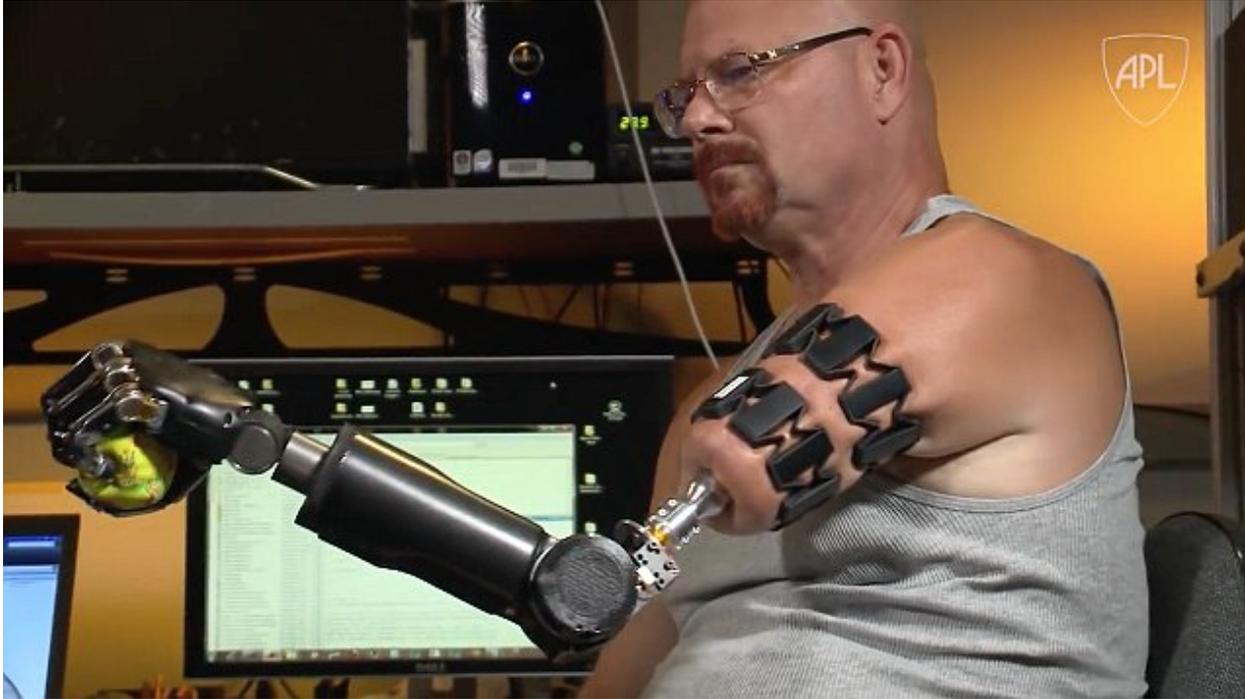
Den Einsatz des Myo-Armbandes in der Prothesentechnologie wird eindrücklich am Beispiel des Patienten Johnny Matheny demonstriert. Dieser verlor seinen linken Arm bis unterhalb des Ellbogens in Folge einer Krebserkrankung. Mit finanzieller Unterstützung der U.S Army entwickelte ein Forscher-Team am “Johns Hopkins Applied Physics Laboratory” mit dem Myo-Armband eine myoelektrische High-End Armprothese[13][14][15]. Die Motorik und Sensorik des künstlichen Armes von Thalmic Labs sind meinem um einige Grössenordnungen überlegen, die Software dahinter hat aber vieles gemeinsam mit dem Vorgehen, welches ich unter Punkt 3.4 beschreibe.<sup>24</sup>

Dabei wird, zumindest in der hauseigenen Forschung, nicht mit 3D-druckbaren Elementen gearbeitet. Verschiedene Einzelpersonen und Forscherteams haben aber bereits den 3D-Druck und das Myo-Armband kombiniert. Ich zähle mich selbst nicht dazu, denn das Ziel dieser Arbeit war es, einen motorisierten Roboterarm zu bauen, welcher später an eine Roboterschulter und einen Roboterkopf angeschlossen werden sollte und der nicht dafür konzipiert wurde, mit einem menschlichen Unterarm zu interagieren.

---

<sup>23</sup>Durchschnittliche Kosten einer myoelektronischen Armprothese über 5 Jahre [12]

<sup>24</sup>Abbildung 17 Quelle: <https://geekoutsw.com/2016/10/15/top-10-mech-humans/>



*Abbildung 17: Johnny Matheny mit Prothese, die durch das Myo kontrolliert wird.*

Bei genauerer Überlegung besitzt aber selbst mein Endprodukt in dieser Hinsicht ein gewisses Potential, wenn auch nicht für den Einsatz am Menschen. Trotzdem möchte ich kurz beschreiben, wie ein solcher Anschluss meines künstlichen Unterarmes an einen menschlichen Armstumpf aussehen könnte. Die Servos, welche die Fingerbewegungen und die Handgelenkrotation ermöglichen, sitzen im Unterarm beziehungsweise im Handgelenk des künstlichen Armes. Daher würde der Einsatz des Unterarms alleine diese Körpergliedmassen substituieren können.

Nun kommt es darauf an, wo der Armstumpf endet. Reicht er über den Ellbogen hinaus, müssten beim künstlichen Unterarm die Servos tiefer in den Unterarm geschoben werden. Wäre der Armstumpf kürzer als bis zum Ellbogen, müsste eine Vorrichtung am Stumpf befestigt werden, welche als Ellbogengelenk fungiert, sprich den künstlichen und den organischen Arm verbindet. Es müsste aber genug Oberarm vorhanden sein, um 1-2 Myo-Armbänder anbringen zu können. Zusätzlich müsste die Stromversorgung über einen Akku oder eine Batterie entweder im künstlichen Arm selbst, am Körper des Patienten oder gegebenenfalls in einem Rucksack untergebracht werden.

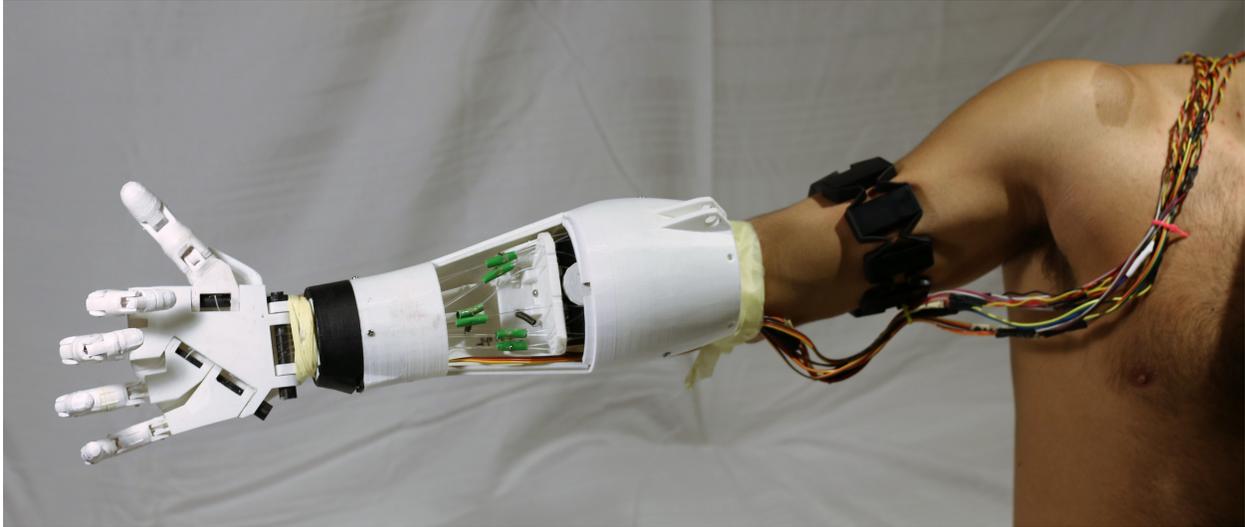
Nun könnte der Patient, wie im Kapitel 2.3 beschrieben, mit dem Myo Armband Gesten trainieren. Im Unterschied zum im Kapitel 2.3 beschriebenen Vorgehen, würde er dabei aber nicht mit einer Hand Gesten ausführen und diese aufzeichnen lassen, sondern versuchen, sich die auszuführenden Gesten vorzustellen, und die dabei entstehenden Nervenimpulse, welche an der Messstelle ankommen, mit dem Myo messen. Wären nun wiederum genug Daten vorhanden, aufgrund derer bestimmten Impulsmuster relativ genau eine gewünschte Geste oder Bewegung zugeordnet werden könnte, könnte dem Prozedere im Kapitel 2.3 gefolgt werden, um es dem Patienten zu ermöglichen, mit diesen trainierten Nervenimpulsen den InMoov-Unterarm zu kontrollieren.

Diese Prothese wäre natürlich auch nicht gratis, die Material- und Hardware-Kosten beliefen sich ungefähr auf<sup>25</sup>:

Bauelement	Preis (CHF)
Myo Armband	250
Filament	20
Servos	60
12 Ampèrestunden-Akku	40
Arduino Nano	23
Verlängerungskabel, Schrauben und Klebstoff	10
Fischerleine	5
<b>Total</b>	<b>408</b>

408 CHF also, und Zugang zu einem 3D-Drucker bräuchte man auch noch. Es wird aber deutlich, dass man sich in einer ganz anderen Preisklasse befindet. Zwar auch in einer tieferen Qualitätsklasse, in welcher der Funktionalität relativ schnell Grenzen gesetzt sind, etwa durch die Maximalnutzlast, welche sich von Kunststoffingern und -gelenken tragen lässt.

<sup>25</sup>Dies sind die Preise welche ich selbst zahlte, abgesehen vom Arduino Nano, dessen Preis ich der Herstellerwebseite entnahm[16]



*Abbildung 18: Inmoov-Arm in Kombination mit Myo als Prothese*

Auch das Drehmoment der im Arm verwendeten MG94 Servos würde nicht ausreichen, um Objekte mit einem Gewicht von mehr als einem Kilo zu tragen. Ich denke, dass in der Prothesentechnik die Frage “Was ist technisch möglich?” zwar durchaus wichtig ist, aber dass die Frage “Was kann ich mir leisten?” für den überwiegenden Grossteil der Patienten das relevantere Thema darstellt. In diesem Sinne möchte ich einerseits aus dieser kleinen Diskussion der Einsatzmöglichkeiten meines Armes den Schluss ziehen, dass der InMoov-Arm in Kombination mit dem Myo-Armband zwar eine bescheidene Funktionalität aber dafür preislich eine nennenswerte Alternative offerieren kann.

Andererseits möchte ich in einem letzten Beispiel illustrieren, wie billig es werden kann, wenn man auf die Motorisierung verzichtet, aber dennoch Zugang zu einem 3D-Drucker hat. Zudem muss es sich um einen Armstumpf handeln, bei welchem Teile des Unterarms vorhanden sind. In diesem Falle könnte man auf der Website [thingiverse.com](https://www.thingiverse.com), aus einer grossen Auswahl von Prothesen Modellen beispielsweise das Modell für eine Rechte Hand von “Atomic-Labs<sup>26</sup>” auswählen. Um sie in der Originalgrösse des Modelles zu drucken, bräuchte es 0.286 kg Filament. Mit dem Material PLA würde das notwendige Filament 10.02 CHF kosten<sup>27</sup>. Es würde sich dabei um eine Low-

<sup>26</sup>Quelle: <https://www.thingiverse.com/thing:2199054>

<sup>27</sup>Bei 40 Prozent Infill und einer Auflösung von 0.15mm. Berechnung durch Cura

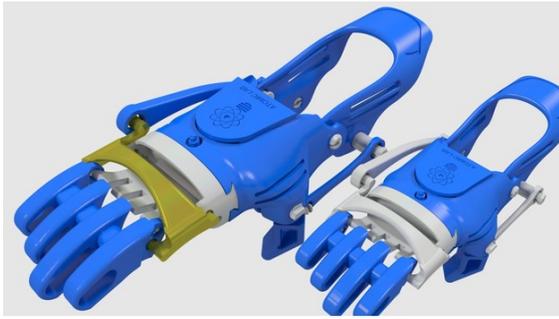


Abbildung 19: 3D-Modell einer Low-End Prothese von "Atomic Lab"



Abbildung 20: Patientin mit Prothese von "Atomic Lab"

End Passiv-Prothese handeln, welche gerade mal das Ballen der künstlichen Hand zur Faust ermöglichen könnte.

Zuletzt kann man sagen, dass sich mit 3D-Druck und dem Myo vielleicht keine High-End Exoprothesen<sup>28</sup> anfertigen lassen, aber das am anderen Ende der Preisskala mit diesen beiden Technologien durchaus Lösungen konstruiert werden können, welche vielen Patienten ein kleines Stück Bewegungsfreiheit zurückgeben und ihren Alltag bis zu einem gewissen Grade erleichtern können.

---

<sup>28</sup> "Befindet sich die Prothese außerhalb des Körpers wie bei künstlichen Gliedmaßen, spricht man von einer Exoprothese[17]"

# 5 Reflexion

## 5.1 Erfahrungen

### Handwerklich: Materialeigenschaften, Werkzeugbedienung

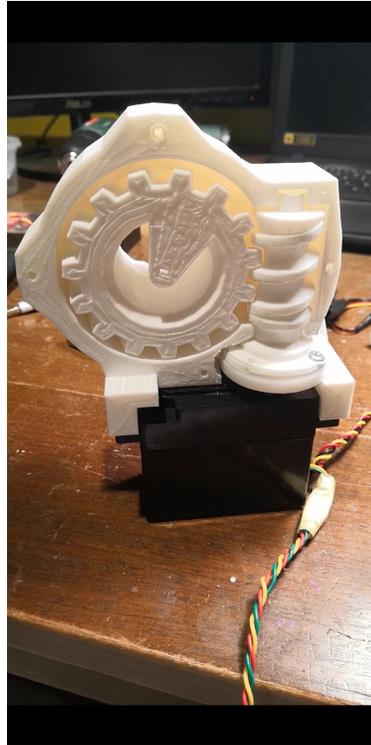
Verschiedene Individuen, gegenüber welchen ich den Zusammenhang zwischen meiner Arbeit und den Instruktionen auf `inmoov.fr` erläuterte, äusserten die Ansicht, dass sich die Montage mit derjenigen eines Lego-Bausets vergleichen liesse. Dies stimmt insofern, als klassische Lego-Bausteine aus dem Thermoplasten ABS bestehen, womit sie eine chemische Übereinstimmung teilen mit beispielsweise den Fingersegmenten an der Hand meines InMoovs.

Dort finden die Gemeinsamkeiten meines Erachtens aber auch schon ein Ende, denn nicht ein einziges Bauteil des InMoovs liess sich einfach auf das andere draufstecken. An jedem Einzelnen musste noch entweder geschliffen, gebohrt, gefeilt, mit Aceton kaltgeschweisst oder mit dem LötKolben geschmolzen werden. Ich ziehe meinen Hut vor dem Engagement von Gael Langevin und der Komplexität seines Projektes, doch was ich wie weiter unten genauer beschrieben definitiv als verbesserungswürdig erachte, sind die Übergänge zwischen den einzelnen Bauteilen, spezifischer die Präzision mit welcher beispielsweise Gewinde ineinander greifen oder Zylinder in Öffnungen passen.

Betrachtet man beispielsweise eine Öffnung mit 8mm Durchmesser und will durch diese einen Zylinder von 8mm Durchmesser hindurchschieben, erscheint einem diese Aufgabe auf den ersten Blick eher simpel. Wie viele Stunden man aber die Öffnung mit Feile und LötKolben vergrössern, den Zylinder abschleifen oder mit dem Japanmesser beschnitzen muss und wie viele Bauteile man zerschleisst, bevor das Gelenk oder die Kraftübertragung schliesslich “reibungslos” funktionieren, dies ahnt man erst auf den zweiten Blick.



*Abbildung 21: Gewinde, Handwerkliche Bearbeitung*



*Abbildung 22: Gewinde, Software Bearbeitung*

Obwohl das Gewinde nach meiner energischen Bearbeitung zu funktionieren begann, entschloss ich mich schlussendlich dennoch dazu, den Effekt, welchen ich handwerklich erzielt hatte, mit Skalieren der 3D-Modelle im Slicer zu replizieren, was zu einem befriedigenderen Resultat führte.

Wie ich selber bei verschiedenen Gelegenheiten schmerzhaft herausfinden musste, ist ABS um ein Vielfaches zerbrechlicher als sein Artgenosse PLA. Vergleicht man aber das Kaltschweißen von ABS und die damit einhergehende Stabilität und die ästhetische Qualität der Übergänge mit Experimenten, in welchen PLA-Stücke mit Araldit aneinandergeklebt werden müssen, dann wiegen diese Vorteile die verminderte Bruchfestigkeit von ABS auf. Es wäre theoretisch möglich, denselben Kaltschweisseffekt auch bei PLA zu erzielen unter Einflussnahme der Chemikalien Dichlormethan oder Ethylacetat, wie mir der Chemielaborant am LG erklärte.

Ich entschied mich jedoch für eine pragmatische Kombination beider Materialien, ohne die im letzten Satz genannten Möglichkeiten anzuwenden, da ich zuversichtlich war, mit PLA, ABS und Aceton die gewünschten Ergebnisse erzielen zu können.

Gegenstand dieser Überlegungen waren hauptsächlich die Finger der rechten Hand des InMoovs. Als ich diese zusammenbaute, musste ich fünf Generationen derselben ausdrucken, bis mir die Konstruktion gelang, ohne dass immer wieder Fingerglieder abbrachen. Einerseits musste um die einzelnen Glieder herum gebohrt werden, um die Gelenk-Bolzen-Öffnungen auf ein Minimum vergrössern zu können. Andererseits stellten die Finger auch sehr exponierte Körperteile dar und mussten daher auch ein Mindestmass an Ästhetik zu bieten haben. Nach zwei Generationen zerbrochener Fingergelenke und zu vielen ästhetischen Kompromissen entschied ich mich trotz der erhöhten Sprödigkeit des Materials dazu, auf ABS zu wechseln. Mit dessen Verwendung war es möglich, die Fingerglieder beinahe nahtlos aneinanderzufügen. Nach insgesamt fünf Generationen schaffte ich es schliesslich die vollendeten Finger an der Handfläche zu befestigen. Zuletzt finde ich das Drucken mit ABS auch deshalb eher lästig, weil es sich unter Temperaturschwankungen gerne zusammenzieht oder aber ausdehnt, eine Eigenschaft derentwegen sich einige fertig wirkende Drucke als wertlos entpuppten.

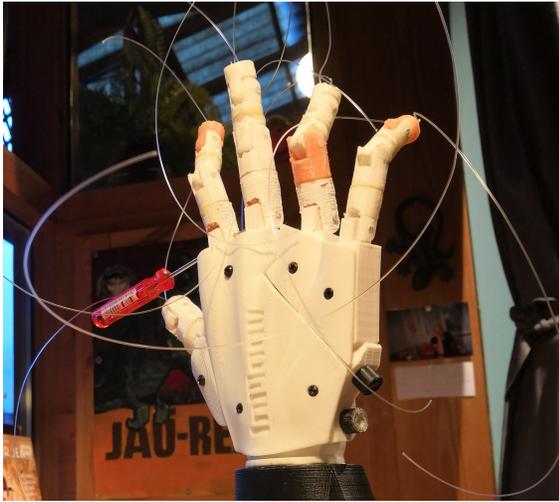


Abbildung 23: Zu viele ästhetische Kompromisse bei Verwendung von PLA



Abbildung 24: 5 Generationen Finger, die erste in der obersten Reihe, die fünfte zu unterst. Die, welche unten fehlen, sind entweder zerbrochen oder eingebaut worden. Rechts unten liegen noch drei der ersten "fertigen" Generation, welche ästhetisch ebenfalls nicht überzeugten

## 5.2 Originalität

Was die Originalität meiner Arbeit betrifft, gilt es Verschiedenes klar zu deklarieren. Wie schon im Kapitel Inspiration erwähnt, stammt das Design des Armes, seiner Einzelteile und die Weise, wie diese auf mechanischer Ebene miteinander interagieren, fast ausschliesslich von der Website [inmoov.fr](http://inmoov.fr) mit wenigen Ausnahmen, bei welchen ich optimierte STL-Files von [thingiverse.com](http://thingiverse.com) heruntergeladen habe.

Die Idee, den Arm in irgendeiner Weise über die Gedanken anzusteuern ist zwar nicht in diesen Worten auf [sparkfun.com](http://sparkfun.com) zu finden. Allerdings hat mich das Gedankengut dieser Website via Marc Frei sehr stark beeinflusst, ganz zu schweigen davon, dass dieses Gedankengut die Prämisse für die Ansteuerung per EEG stellte, auch wenn dieser Ansatz schlussendlich misslang.

Die beiden essentiellen Code-Sammlungen, welche als Grundlage dienten, um die Ansteuerung per EMG bzw. PS4-Controller zu ermöglichen, stammen beide aus verschiedenen GitHub-Projekten, wie im Quellenverzeichnis nachzulesen ist.

Die folgende Tabelle soll aufzeigen, wie es sich in Bezug auf die verschiedenen Ansteuerungsarten der jeweiligen Code-Originalität verhält. DS4 steht dabei für die Ansteuerung über den Controller, Myo für diejenige über das Armband und MindWave für diejenige über das EEG-Headset.

	DS4	Myo	MindWave
Eigenleistung	Arduino	Arduino	
Modifiziert	RPP	RPP	Arduino
Fremdeigentum	Treiber		

Es muss ergänzt werden, dass es sich bei den für den RPP modifizierten Codes um jeweils etwa ein bis zwei Dutzend von mir eingefügte Zeilen handelt, der Rest jedoch ebenfalls geistiges Fremdeigentum ist.

## 5.3 Kritik am eigenen Arbeitsprozess

### Organisation

In Retrospektive muss ich zugeben, dass meine Organisation trotz des eher frühen Beginns der Arbeit (April 2018) etwas zu wünschen übrig liess. Ich führte und führe noch immer mehrere Excel-Sheets, in welchen ich die bisherigen und die geplanten Kosten dokumentiere. Ich hielt und halte ein Arbeitsjournal, in welchem ich meine Sorgen und Erfolge niederschrieb und ich verwaltete ein grosses Arsenal an Schnellverschluss-Säckchen, Kisten und Boxen für all die gedruckten Teile.

Trotzdem hätte es beispielsweise bei der Aufbewahrung der Bauteile auf logistischer Ebene noch Raum für Optimierung gegeben. An Stelle des Word-Dokumentes, in welchem ich festhielt, welches Teil gerade wo und wie gedruckt wurde oder worden war, hätte eine Excel-Datei eine viel übersichtlichere Tabelle liefern können. Das Plastiksack-System der Druckobjekte reichte zwar durchaus aus, um in den produktivsten Phasen des Druckers die Wellen von neuen Teilen effizient sortieren zu können.

Als ich aber zwei Monate später nach dem vorläufigen Ableben meines eigenen Druckers vor der Aufgabe stand, aus dem Meer der Plastiksäckchen die Information herauszufischen, welche Teile ich in welchem Zustand besass,

war dies mit mühsamem Suchen verbunden, welches auch das Worddokument nicht verhindern konnte, weil ich darauf aufzuschreiben vergessen hatte, dass sich mehrere Teile der letzten Ladung als unbrauchbar erwiesen hatten.

## **Katalogisierung**

Ich katalogisierte und katalogisiere die Teile in Plastiksäckchen mit den Merkmalen: Generation, Körperteil, Material, Objektdichte und Zustand der Funktionalität. Leider hielt ich es lange Zeit nicht für nötig, diese Säckchen auch nach diesen Merkmalen zu gruppieren was vereinzelt zu verlangsamten Suchprozessen führte.

## **Ansatz**

Bei einer zukünftigen Arbeit sicher anders entscheiden würde ich mich hinsichtlich der Wahl zwischen einer eher oberflächlichen Abhandlung vieler Themen und der Vertiefung eines bestimmten Gebietes. Es war zwar äusserst spannend, mir sowohl Kenntnisse der Elektrotechnik, der Materialtechnik als auch der Informatik anzueignen, jedoch muss ich ehrlich zugeben, dass ich das Gefühl verspüre, nur leicht an der Oberfläche der jeweiligen Themen gekratzt zu haben.

In der Informatik reicht mein Wissen, um Servos anzusteuern und den Arduino mit dem RPP kommunizieren zu lassen, der genaue Aufbau der von GitHub verwendeten Codes entzieht sich jedoch meinem Verständnis. Ich kenne die elektrotechnischen Grundlagen, welche nötig sind um einen Schaltkreis mit Servos, LEDs, Sensoren und Stromversorgung zu prototypen.

Als aber bei der Präsentation meines Projektes bei Conrad<sup>29</sup> die durch die im Laden laufenden elektronischen Geräte verursachte elektromagnetische Strahlung durch Interferenzen meine Signalübertragungen verunmöglichte, war ich, nachdem eine improvisierte Kabelabdeckung mit Aluminiumfolie scheiterte, ratlos.

---

<sup>29</sup>Elektrowarenverkäufer, welcher einen Wettbewerb organisierte an welchem ich teilnahm. Genauerer im Anhang unter Punkt C

Auch in der Materialwissenschaft lernte ich durch den Umgang mit dem 3D-Drucker viel, allerdings bezieht sich dieses Wissen bzw. diese Erfahrung auf lediglich zwei Filamente.

In diesem Sinne wäre es sicher auch gewinnbringend gewesen, sich auf eines dieser Gebiete zu beschränken und in diesem ein vergleichsweise tieferes Verständnis zu erlangen.

## 5.4 Perspektiven

Im Folgenden werde ich besprechen, wie ich mit dem Projekt fortzufahren gedenke. Dabei werden mindestens drei Ebenen parallel verfolgt werden.

Einerseits wurde bis anhin nur ein Bruchteil des Hardware-Potentials des Projektes InMoov ausgeschöpft. Der Besitz der Konstruktionserfahrungen des rechten Armes sowie der meisten Bauteile des linken Armes laden dazu ein, den linken Arm ebenfalls zu bauen. Daneben habe ich den Grossteil der Teile für den Torso und den Kopf bereits gedruckt, weswegen die Verbindung der beiden Arme ein wichtiges und spannendes nächstes Zwischenziel darstellt. Diese Pläne setzen natürlich voraus, dass entweder der eigene 3D-Drucker repariert oder auf einen anderen Zugriff ausgeübt werden kann.

Ein weiterer Bereich, in welchem beachtliches Verbesserungspotenzial besteht, sind die Signalübertragungsgeschwindigkeiten, RPP- Myo und RPP-PS4-Controller. Diese liessen sich folgendermassen beschleunigen. Im Falle des Codes, welchen ich für die Kommunikation zwischen Myo und RPP Pi einsetzte, hatte ich vollen Einblick in den Code und konnte verzögernde Sequenzen löschen oder beschleunigen. Im Falle des Treibers auf der anderen Seite, welcher dem RPP erlaubte, sich über Bluetooth mit dem PS4-Controller zu verbinden, handelte es sich um einen bereits kompilierten Maschinencode, in welchen ich keine Einsicht hatte und somit nicht ablesen konnte, ob Verzögerungen enthalten waren.

In einem Test, in welchem ich die Signalübertragungsgeschwindigkeiten mit LEDs testete, stellte sich heraus, dass die Verzögerungen sowohl beim Myo als auch beim Controller weder bei der Verbindung Arduino-Servo noch bei der Verbindung Myo-RPP bzw. DS4-RPP stattfanden. In Anbetracht der Tatsache, dass der RPP eine Tendenz zu Abstürzen und Überhitzungen besass, folgerte ich, dass der RPP-CPU mit der Ausführung der beiden Codes überfordert war. Dies wiederum könnte wie folgt gelöst werden. Zum einen kann man auf dem RPP Kühlmodule installieren, welche die Hitze, welche der CPU produziert, besser ableiten können.

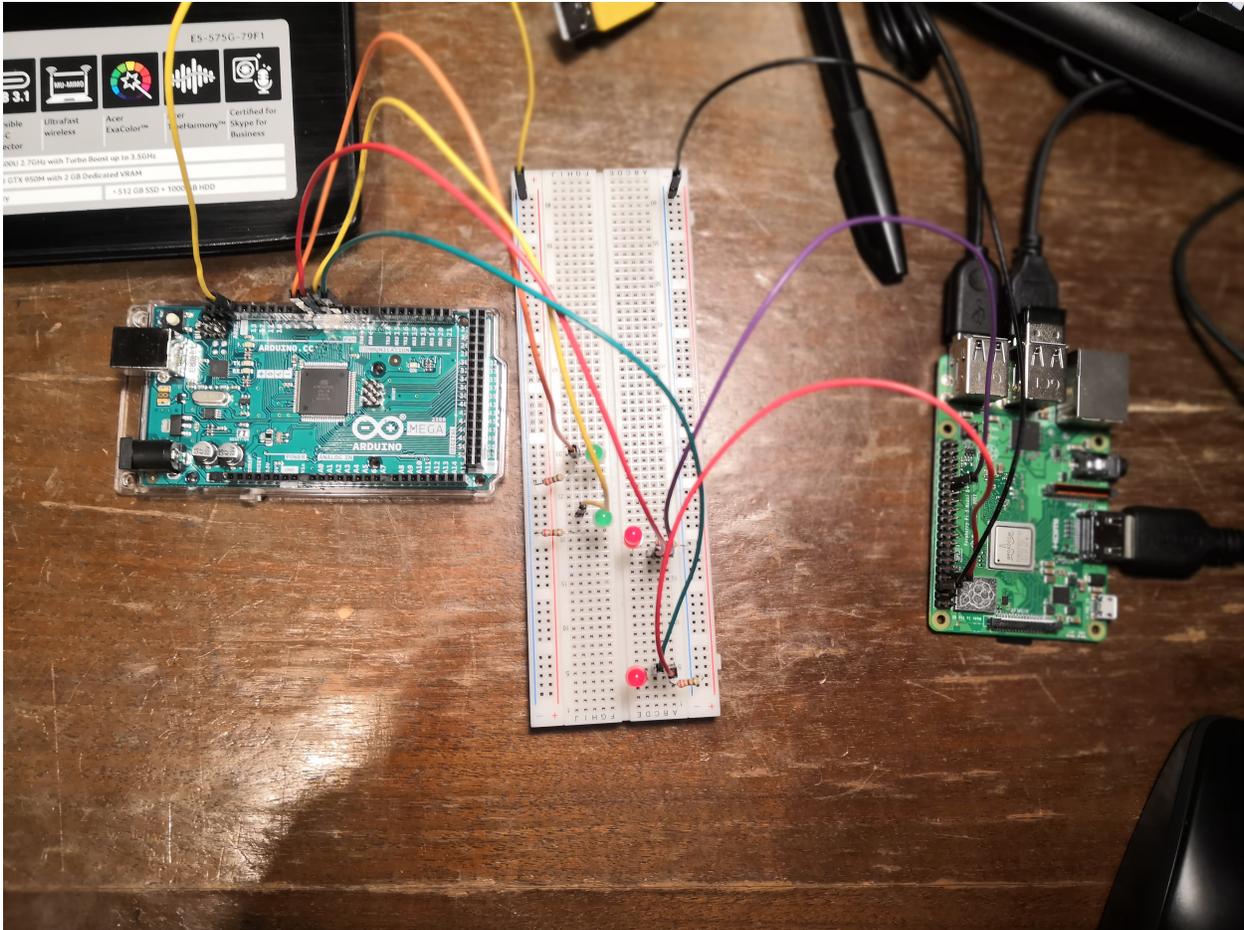


Abbildung 25: Test der Kommunikation zwischen Arduino und RPP mit Hilfe von LED-Lämpchen

Zum anderen gibt es die Möglichkeit, den Rechenprozess für die Exekution der Codes nicht auf dem RPP durchzuführen, sondern auf dem Laptop, welcher statt der 1.2 GHZ des RPP 3.5 GHZ liefern kann<sup>30</sup>

Dazu kann man über die sogenannte SSH (Secure Shell) auf den RPP zugreifen. Der Zugriff über SSH von einem System über ein anderes System kann im Falle des RPP über Wlan oder Ethernet erfolgen, man benötigt dazu lediglich die IP-Adresse und das Passwort des RPP. Danach kann der Rechenprozess verlagert werden.

<sup>30</sup>Tatsächlich kommen diese 3.5 GHZ aber nur beim sogenannten Übertakten (Prozessor wird intensiver gebraucht als Hersteller das empfiehlt) zu Stande, bei Tests mit Benchmark-Programmen würde der Wert deutlich darunterliegen. Insofern läge die Prozessorleistungsdifferenz vermutlich tiefer

Grundsätzlich würde dies zwar schon über das Windows-Terminal funktionieren. Benutzt man jedoch ein Linux-System, um auf das Linux-basierte Betriebssystem des RPP zuzugreifen, kann man dieselben Kommandos verwenden, was das Vorgehen etwas intuitiver gestaltet. Um also Linux von einem Windows-Betriebssystem starten zu können, kann man eine sogenannte Virtual-Box einrichten. Diese erlaubt es dem Nutzer, verschiedene Betriebssysteme zu simulieren, so dass die Systeme auf die Hardware des Laptops zugreifen können, aber dennoch ihre spezifischen Vorteile behalten.

Eine mögliche Beschleunigung der Signalübertragung würde sich also vermutlich erzielen lassen, wenn der Rechenprozess auf meinen Laptop verlagert würde, indem im Terminal in Linux in der Virtual-Box über SSH auf den RPP zugegriffen würde. Dieser Verbesserungsansatz wäre jedoch sicher nicht temporärer Natur, da das Endziel ja die maximale Mobilität des InMooves beinhaltet welche nicht gewährleistet ist, wenn der Rechenprozess ausserhalb seines Körpers stattfindet.

Als letzte Massnahme zur Beschleunigung der Kommunikation, kommt die Funktion `attachInterrupt()` in der Arduino-Umgebung in Frage. Bei dieser Funktion wird einem digitalen Aus- oder Eingang ein sogenannter Interrupt hinzugefügt. Um zu verstehen was dieser nun bewirkt, muss man sich kurz den Hauptteil meines Arduino-Codes vor Augen führen. Dieser besteht aus einer Reihe von If-statements. Das heisst, der Code prüft Schritt für Schritt ob eine Bedingung (in diesem Fall das Eintreffen eines elektrischen Signales auf einen Pin) erfüllt ist und führt je nachdem welche Bedingung erfüllt wurde eine andere Konsequenz aus.

Diese Loop-Funktion kann schnell sein, wenn nur wenige Bedingungen geprüft werden. Sobald es aber 10 oder mehr sind, muss man mit gewissen Verzögerungen rechnen. Die Zeit, welche der Code benötigt, um wieder beim Prüfen der obersten Bedingung anzulangen ist sozusagen ein toter temporaler Winkel, da in diesen Milli- oder Zehntelsekunden die Bedingung ungeprüft bleibt.

Um dieser Verzögerung vorzubeugen, kann man am digitalen Input-Pin also einen Interrupt platzieren. Dieser wird, unabhängig davon, welche Zeile des Codes gerade ausgeführt wird, auf eine Zustandsänderung des Pins reagieren und die gewünschte Konsequenz ausführen. Natürlich gibt es auch bei den Interrupts eine obere quantitative Grenze. Bevor man aber mehrere Tausend solcher Interrupts verwendet sollten diese sich positiv auf die Signalübertragungszeit auswirken<sup>31</sup>. Weiteres Optimierungspotential bietet auch die Elektronik am InMoov Arm. Da ich insgesamt doch nur etwa 18 Kabel durch den Oberarm ziehe und diese dann, jetzt während des Prototypings, an die Steckplatine anschliesse, besitzt das Ganze momentan noch einen gewissen Grad an Übersichtlichkeit. Sobald aber noch mehrere Dutzend weitere Kabel angeschlossen werden und vorausgesetzt der Rechenprozess ist nicht extern gelagert, wird es nötig sein, statt des Prototypen via Steckplatinen die Verkabelung an richtigen Platinen anzulöten.

Bis anhin verdrehte und lötete ich die Kabelstränge selbst, ab einer gewissen Quantität wäre es von Vorteil die Kabel für Strom, Erdung und Signal nicht mehr in einem einzigen Kabelstrang zu führen, sondern die Stränge aufzuteilen. In dem nämlich z.B. die Signalübertragungen über Flachbandkabel stattfinden, kann Platz gespart und Übersichtlichkeit geschaffen werden. Die Flachbandkabel auch für die Stromversorgung zu verwenden, wie es Gael vorschlägt, ist aber gefährlich, da ein Flachbandkabel relativ zu seinem Durchmesser viel weniger Strom übertragen kann ohne zu schmelzen als ein einzelnes plastikummanteltes Kupferkabel.

Zum Schluss besteht natürlich auch noch der Ehrgeiz, die Ansteuerung über das EEG doch noch zu realisieren. Im Anhang erkläre ich, weshalb die Verbindung zwischen Arduino und EEG-Headset fehlschlägt. Liesse sich analog zum Myo und dem DS4 das Kommunikationsprotokoll für die Bluetooth-Verbindung mit dem RPP finden und implementieren, wäre es vielleicht sogar möglich, alle drei Ansteuerungsarten zu kombinieren. Es lassen sich nämlich sowohl die Python-Codes wie auch die C++ Arduino-Codes zu jeweils einem zusammenfassen. Um diesen Kollektiv-Code dann aber zu prozessieren, bräuchte es auf der Seite des RPPs sicherlich mehr Rechenleistung.

---

<sup>31</sup>Allerdings muss angefügt werden, dass genau durch diese höhere Priorisierung der Interrupts über andere Funktionen, es sein kann, dass ihre Einfügung komplexe Folgen nach sich zieht, welche weder gut erkennbar noch positiver Natur sein müssen.

## 6 Quellen- und Literaturverzeichnis

- [1] W. Foundation, *Einplatinencomputer*, Zuletzt bearbeitet: 13.19.2018. Adresse: <https://de.wikipedia.org/wiki/Einplatinencomputer> (besucht am 10.12.2018).
- [2] S. Robotics, *Preis des humanoiden "Asimov" von Honda*, 2016. Adresse: <https://www.smashingrobotics.com/thirteen-advanced-humanoid-robots-for-sale-today/> (besucht am 09.12.2018).
- [3] 2. Minuten, *Schweizer Künstler steuert Lampe mit Gedankenkraft*, Erscheinungsdatum: 05.06.2018. Adresse: <https://www.20min.ch/digital/news/story/Schweizer-Kuenstler-steuert-Lampe-mit-GedankenkraftSchweizer-Kuenstler-steuert-Lampe-mit-Gedankenkraft-14524063> (besucht am 10.12.2018).
- [4] S. Kravitz, *Hackers in Residence - Hacking MindWave Mobile*, 2013. Adresse: <https://learn.sparkfun.com/tutorials/hackers-in-residence---hacking-mindwave-mobile/all> (besucht am 09.12.2018).
- [5] C. Norotte, F. S. Marga, L. E. Niklason und G. Forgacs, "Scaffold-free vascular tissue engineering using bioprinting", *Biomaterials*, Jg. 30, Nr. 30, S. 5910–5917, Okt. 2009. DOI: 10.1016/j.biomaterials.2009.06.034.
- [6] D. Black, *The latest in kitchen tools: 3-D printers, robot chefs, and the Digital Chocolatier*, Zuletzt bearbeitet: 09.14.2010. Adresse: [https://www.thestar.com/life/2010/09/14/the\\_latest\\_in\\_kitchen\\_tools\\_3d\\_printers\\_robot\\_chefs\\_and\\_the\\_digital\\_chocolatier.html](https://www.thestar.com/life/2010/09/14/the_latest_in_kitchen_tools_3d_printers_robot_chefs_and_the_digital_chocolatier.html) (besucht am 10.12.2018).
- [7] H. L. und Melba Kurman, *Die neue Welt des 3D-Drucks*, 1. Auflage. 2014.
- [8] W. Foundation, *Microcontroller*, Zuletzt bearbeitet: 05.12.2018. Adresse: <https://en.wikipedia.org/wiki/Microcontroller> (besucht am 10.12.2018).
- [9] H. Choudhary, *Difference between Microprocessor and Microcontroller*. Adresse: <https://www.engineersgarage.com/tutorials/difference-between-microprocessor-and-microcontroller> (besucht am 10.12.2018).
- [10] C. Rosell, *A Sony DualShock 4 userspace driver for Linux*, 2013. Adresse: <https://github.com/chrippa/ds4drv> (besucht am 04.12.2018).
- [11] F. Cosentino, *A linux alternative to what MyoConnect scripting does under Windows*. 2012. Adresse: <http://www.fernandocosentino.net/pyoconnect/> (besucht am 04.12.2018).
- [12] D. K. B. et al., "Prosthetic cost projections for servicemembers with major limb loss from Vietnam and OIF/OEF", *The Journal of Rehabilitation Research and Development*, Jg. 47, Nr. 4, S. 387, 2010. DOI: 10.1682/jrrd.2009.04.0037.
- [13] C. News, *Johnny Matheny's prosthetic arm controlled by Thalmic Lab armband*, Zuletzt bearbeitet: 24.11.2015. Adresse: <https://www.cbc.ca/news/canada/kitchener-waterloo/johnny-matheny-robotic-arm-uses-thalmic-labs-aramband-1.3334199> (besucht am 11.12.2018).
- [14] M. Starr, *Myo armbands used to control prosthetic arm*, Zuletzt bearbeitet: 18.01.2016. Adresse: <https://www.cnet.com/news/myo-arambands-used-to-control-prosthetic-arm/> (besucht am 11.12.2018).

- [15] J. Erikson, *Thought-Controlled Robotic Arm 'Makes a Big Negative a Whole Lot Better'*, Zuletzt bearbeitet: 05.07.2013. Adresse: <https://uanews.arizona.edu/story/thought-controlled-robotic-arm-makes-a-big-negative-a-whole-lot-better> (besucht am 11.12.2018).
- [16] Arduino, *Preis Arduino Nano*. Adresse: <https://store.arduino.cc/arduino-genuino/boards-modules> (besucht am 09.12.2018).
- [17] W. Foundation, *Prothese*, Zuletzt bearbeitet: 01.10.2018. Adresse: <https://de.wikipedia.org/wiki/Prothese> (besucht am 11.12.2018).
- [18] —, *Toleranz (Technik)*, Zuletzt bearbeitet: 27.10.2018. Adresse: [https://de.wikipedia.org/wiki/Toleranz\\_\(Technik\)](https://de.wikipedia.org/wiki/Toleranz_(Technik)) (besucht am 09.12.2018).

## 7 Anhang

### A Versuch der Ansteuerung per Gedankenkraft

Von den drei Ansätzen, welche ich für die Ansteuerung des Armes testete, funktionierte der für diese Arbeit namensgebende nicht. In diesem Kapitel werde ich die Gründe dafür erläutern. Zunächst die Rekapitulation der geplanten Signalkette:

Das Ursprungssignal entsteht im Gehirn, entweder durch mentale Ent- oder Anspannung. Jedes Neuron kann, falls es angeregt oder gehemmt wird, diese Information über seine Zustandsänderung als Aktionspotential über kleine Verästelungen an das nächste Neuron weitergeben. Somit entsteht ein Netzwerk elektrischer Signale. Das EEG-Messgerät kann dessen Intensität messen bzw. bestimmen, an welchen Punkten, auf wenige Zentimeter genau, wie viel elektrische Aktivität herrscht.

Damit kann es die elektrischen Veränderungen, welche bei der Informationsverarbeitung entstehen, über zwei Elektroden messen und aus besagter Frequenz und Amplitude die momentane Hirnwellenaktivität berechnen. Diese Information wird sogleich via Bluetooth an das *Bluesmirf RN-41 Bluetooth-*

*Modul*<sup>32</sup> gesendet und gelangt von dort über eine serielle Schnittstelle zum

---

<sup>32</sup>Modul bedeutet in diesem Zusammenhang ganz simpel, dass eine elektrotechnische Komponente an ein anderes System gekoppelt wird die den Namen RN-41 trägt

Prozessor des Mikrocontrollers Arduino. Dieser Prozessor entscheidet unter Einfluss des programmierten Codes, ob eine Schwelle des Aufmerksamkeitswertes unterschritten wird, und schickt elektrische Signale an die ihm angeschlossenen Servos. Das Ursprungssignal entsteht also im Hirn und wandert über verschiedene Etappen zur Endstation, dem Servo.

Als schwierig stellte sich die Signalübertragung zwischen dem MindWave-Headset und dem RN-42 heraus. Um zwei Geräte über Bluetooth miteinander zu verbinden, bedarf es einiger Parameter, welche bestimmt werden müssen. Zum einen muss die Slave-Master-Beziehung klar definiert sein, soll heißen, ein Gerät ist der Master und kann die Verbindung initiieren und den Datenaustausch kontrollieren, während dem anderen Partizipanten die Rolle des Slaves zukommt, in welcher dem Namen entsprechend überhaupt nichts kontrolliert wird und das Gerät nur auf Befehle reagieren kann.

Des Weiteren muss die sogenannte BAUD-Rate bei beiden Teilnehmern deckungsgleich sein. Die BAUD-Rate ist ein Mass für die Geschwindigkeit der Informationsübertragung in einem Datenkanal, also dafür, wie viele Bits pro Sekunde übertragen werden. Obwohl theoretisch eine willkürliche BAUD-Rate möglich wäre, gibt es einige per Konvention festgelegte Raten, von 110 aufsteigend bis hin zu 921'600. Beim Arduino-Board selbst lassen sich die gewünschten Einstellungen direkt in den Code hineinschreiben, beim Mind-Wave Headset jedoch kann man nur recherchieren oder das Datenblatt des Herstellers konsultieren um herauszufinden welche Modi aktiv sind.

Mit der Android-Applikation "S2 Terminal for Bluetooth"<sup>33</sup> kann man ein Smartphone mit dem Bluetooth-Modul verbinden und dieses Kommunikationsprotokoll manuell einrichten. Als ich letzten Endes die Signalkette komplett errichtet hatte, sowohl die Hardware wie auch die Software betreffend, stellte ich fest, dass keine Daten beim Arduino ankamen. Ich verifizierte zahlreiche Male die Stabilität der anderen Signaletappen und kam zum Schluss, dass die Daten des Headsets gar nie beim RN-41 ankamen.

---

<sup>33</sup>Quelle: Google Play Store

Ich lieb das MindWave-Headset von Marc Frei aus, um zu testen, ob das Problem bei meinem Headset liege. Als es aber nicht möglich war, Freis Headset auch nur mit meinem PC zu verbinden, weil das Verbindungsprogramm einen Fehler aufwies, wandte ich mich anderen Ansteuerungsplänen zu.

## B Diskussion Projekt InMoov

Wie zu Beginn bereits erwähnt, stiess ich beim stöbern auf der Website `thingiverse.com` auf modifizierte Teile einer humanoiden Roboterhand, wurde neugierig und folgte dem Link auf die Seite `inmoov.fr`. Auf dieser Website hat der französische Designer Gael Langevin sein Projekt der Weltöffentlichkeit zugänglich gemacht.

Mr. Langevin, der Gründer und die Antriebskraft hinter dem Projekt, wollte ursprünglich eigentlich “nur” eine futuristisch aussehende Armprothese bauen im Zuge eines Design-Auftrages. Obwohl er den Auftrag als solchen nicht erfüllen konnte, beschloss er die dem Auftrag zugrunde liegenden Ideen und 3D-Designs zu veröffentlichen und legte somit im Januar 2012 den Grundstein für die weltweite Beteiligung an der Entwicklung des “Open source 3D-printed life size robot”.

Neben den 3D-Modellen postete er auf `inmoov.fr` auch Konstruktionsanleitungen in Form von Bildgalerien und YouTube-Videos und schuf ein Forum, in welchem Probleme und Ideen von allen Nutzern gelöst und diskutiert werden konnten. Auch Code-Vorlagen für das Testen einzelner Servos oder für die Sprachsteuerung findet man auf der Website. Es werden hauptsächlich die Arduino IDE und das Programm MyRobotLab unterstützt.

Er entwarf den gesamten Oberkörper eines humanoiden Roboters und stattete ihn mit verschiedenen Funktionen aus. Diese Funktionen beinhalten einen Tastsinn in Form von Drucksensoren in den Fingerspitzen und einen Sehsinn, der sich aus drei Komponenten zusammensetzt:

- einer Kamera im Auge
- einem PIR-Sensor, welcher ähnlich einer Fledermaus die Distanz zu umliegenden Objekten messen kann
- einer Xbox-Kinect, welche mit Hilfe einer Tiefen-Kamera sich bewegende Objekte überwachen kann

Neben den Sensoren zur vereinfachten Wahrnehmung von visuellen, auditorischen und taktilen Signalen, welche dem InMoov zur Verfügung stehen, liessen sich noch weitere Temperatur-, Luftdruck-, Infrarot- oder Hygrosensoren einbauen. Auch haben verschiedenste Makers Teile optimiert und sie dann per [thingiverse.com](http://thingiverse.com) veröffentlicht oder gar eigene Anleitungen und Gedankenanstösse an Gael weitergegeben, welcher sie dann in die Website integriert. Zuletzt lässt sich auch die von Gael vorgeschlagene Ansteuerung durch Arduino-Boards durch viele andere Mikroprozessoren ersetzen wie z.B. Raspberry Pi.

## Kritik

Im vollen Bewusstsein, dass meine Arbeit in der Form wie sie vorliegt nicht möglich gewesen wäre ohne Gael Langevins Lebenswerk, möchte ich dennoch einige Punkte ansprechen, die meines Erachtens anders gelöst hätten werden können. Ich halte seine künSTLerischen Fähigkeiten in Ehren, welche für eine hochwertige humanoide Erscheinung sorgen, wenn man Teile des InMoovs oder ihn als Ganzes vor Augen hat. Jedoch beschlich mich im Laufe der Konstruktion der Verdacht, dass dem Design oftmals ein ähnlich wenn nicht gar gleich hoher Stellenwert wie der Funktionalität beigemessen wurde.

Der Übergang zwischen den einzelnen “RobParts2-5”, Teilen des Unterarms, ist äusserst fein und falls richtig geleimt oder mit Aceton kaltgeschweisst auch fast nicht zu erkennen, jedoch hätte hier eine andere Verbindung, z.B. ein Stecksystem meiner Meinung nach für mehr Stabilität gesorgt. Tatsächlich finden im Oberarm dann einige Verbindungen mittels einfacher Stecksysteme statt, diese sind Gegenstand meines nächsten Kritikpunktes, der Präzision.

In der Technik spricht man von sogenannten Toleranzen. “Die Toleranz bezeichnet den Zustand eines Systems, in dem eine von einer störenden Einwirkung verursachte Abweichung vom Normalzustand (noch) keine Gegenregulierung oder Gegenmaßnahme notwendig macht oder zur Folge hat.[18]”

Beim Übergang zwischen “LowArmBase” und “HighArmBase” im Bizeps entsprechen die Durchmesser der Öffnungen genau den Durchmessern der Teile, die in sie hineingesteckt werden sollten. Dies hat zur Folge, dass mit einer konventionellen Druckpräzision von 0.1 – 0.15mm die Teile nicht aufeinander passen werden. Die Störung des Normalzustandes wäre in diesem Beispiel die Abweichung vom 3D-Modell des Bauteils, bedingt durch die unzureichende Auflösung beim Drucken. Toleranz würde nun heissen, das 3D-Modell im Vornherein auf eine Grösse zu skalieren, bei welcher trotz einer gewissen Abweichung der konventionellen Druckpräzision dennoch die Funktionalität des Objektes garantiert wäre.

In diesem Falle kann mit dem Japan-Messer nachgeholfen oder können die Übergangsstücke erhitzt werden, was schlussendlich zu einem einigermaßen stabilen Ergebnis führt. Falls aber beispielsweise ein Zylinder in einer Öffnung steckt und in dieser rotieren soll, dann muss die ganze Öffnung gleichmässig vergrössert oder der Zylinder abgeschliffen werden, was Zeit kostet. Toleranz könnte auch bedeuten, im Namen der besagten STL-File oder in der dazugehörigen Konstruktionsanweisung zu erwähnen wie man das 3D-Modell noch im Slicer so bearbeiten (z.B. skalieren) könnte, dass solche Nachbearbeitungen obsolet wären. Es würde nur schon helfen, wenn bei allen Teilen stehen würde, ob eine Schichtdicke von 0.25mm ausreicht oder ob es sich um einen Fall für 0.06mm handelt.

Diese teils etwas unpräzisen Angaben führten in zahlreichen Fällen dazu, dass Bauteile schlichtweg inkompatibel miteinander waren und eine arbeitsintensive Nachbearbeitung von Nöten war. Als letzten Kritikpunkt kann man benennen, dass sich das Projekt zwar an Einsteiger in die Welt der Elektrotechnik richtet, trotzdem aber sehr oft Nicht-Laienwissen vorausgesetzt wird um einzelne Ziele zu erreichen.

Das Verbinden aller elektronischen Komponenten zum Beispiel und die Stromversorgung derer wird zwar angesprochen, aber nicht in einer Masse, in welcher man darum herum käme sich noch einmal selbst in die Theorie dahinter zu vertiefen.

## Chancen

Nach diesen kritisierenden Worten ist es umso wichtiger, die Gründe zu betonen, weshalb das Projekt InMoov dennoch eine hervorragende Idee ist und weshalb es meiner Meinung nach einen ernstzunehmenden positiven Einfluss auf eine ganze Generation technikaffiner Einzelpersonen und Gruppen hat. Wie ich bei eigenem Leib spürte, ist die Robotik ein immens komplexes Gebiet, welches Mindestkenntnisse in Mathematik, Materialwissenschaften, Informatik und Elektrotechnik voraussetzt, um überhaupt die ersten kleinen Projekte realisieren zu können.

Die Aneignung dieses Wissens läuft Gefahr, als sehr mühselig oder gar schwer erreichbar fehldiagnostiziert zu werden. Natürlich muss Zeit investiert und Theorie verinnerlicht werden, aber wie es meine ersten kleinen Erfolge bei der Umsetzung dieses Projektes demonstrieren, bedarf es keineswegs eines Ingenieursstudium, um die ersten Schritte zu wagen. Gael Langevin ist es gelungen ein sehr komplexes Unterfangen, die Konstruktion eines Humanoiden in einer Masse zu vereinfachen und Fachleuten zugänglich zu machen, dass sich im Prinzip jeder, welcher sich die Zeit nehmen kann, die oben genannten Themen ein wenig zu studieren und das nötige Kleingeld besitzt, in der Lage ist, einen lebensgrossen Roboter zu bauen!

Den Begriff Kleingeld finde ich passend, da wie zu Beginn bereits erwähnt die Entwicklung eines Roboters sich auch im siebenstelligen Bereich befinden kann anstatt im ganz knappen vierstelligen. Die teuersten Komponenten sind die Servos, welche allein etwa 750 CHF kosten und die circa fünf bis sieben Kilogramm Filament, deren Kosten sich ebenfalls auf mehrere hundert Franken belaufen.

Diese Zahlen mögen im ersten Moment eine abschreckende Wirkung haben, jedoch sollte man sie zunächst einmal in Relation setzen mit den Gesamtkosten der Entwicklung anderer humanoiden Roboter. Würde man beispielsweise den Humanoiden “Asimo” von Honda kaufen wollen, hinge das Preisschild bei 2'500'000 US Dollar[2].

Diese überschaubaren Kosten sind eine Kernqualität des Projektes InMoov. Dutzende, wenn nicht hunderte Einzelpersonen und Gruppen auf der ganzen Welt, bestehend aus sogenannten “Makers” (modischer Anglizismus für Hobbybastler), Schulklassen und Universitäten haben sich dem Bau des InMoovs verschrieben.

Des Weiteren ist die Strukturierung der ganzen Website und des Projektverlaufes ziemlich intuitiv verständlich und führt zu einer Lernkurve die meistens parallel zu den benötigten Kenntnissen verläuft. Zuletzt ergeben sich viele Vorteile des Projektes aus seiner Open-Source Natur. Wäre der InMoov ein Produkt einer Firma, mit unveröffentlichtem oder unzugänglichem Quellcode oder inkompatibler Hardware à la Apple, dann wäre beispielsweise die Integration des RPPs ein schwierigeres Vorhaben geworden. Genau die Tatsache, dass das Projekt auch hardware-technisch Open-Source ist, sorgt für eine konstante Optimierung seitens der Community, welche ihre Verbesserungsvorschläge auf Thingiverse publiziert oder sie gar auf `inmoov.fr` in als Projektbestandteil bezeichnen darf.

Ich und hunderte andere Individuen und Gruppen sind Gael also zu grossem Dank verpflichtet, seine Idee nicht flächendeckend monetarisiert und stattdessen die Welt der Robotik der Welt zugänglich gemacht zu haben. Einen gewissen finanziellen Mehrwert besitzt `inmoov.fr` aber trotzdem, zumindest für ihn, da er selber designte Platinen auf der Website zum Verkauf anbietet und auch immer wieder Spenden in Form von Geld oder Filament erhält.

## C TekkieLab

Der Elektronikwarenhändler Conrad veranstaltete im Herbst 2018 den sogenannten TekkieLab Wettbewerb. Jeder Schweizer konnte sein eigenes “Tekkie”-Projekt oder das einer Gruppe in einem E-Mail an Conrad beschreiben und dafür argumentieren, wieso sich das Projekt besonders gut eignen würde, um mit 1000 CHF subventioniert zu werden.

Optimistisch verfasste ich ein E-Mail an das TekkieLab-Team von Conrad und präsentierte das Konzept meiner Maturaarbeit, begleitet von einigen Bildern derselben. Das TekkieLab-Team bedankte sich und kündigte an, dass von Hunderten Projekten 9 in die nächste Runde gekommen wären und ich einer davon wäre. Im nächsten Schritt wurden die Projektbeschriebe dieser 9 auf verschiedenen Social-Media-Kanälen Conrads hochgeladen und zur Abstimmung freigegeben mit dem Vermerk, dass die Top Drei 1000.- in Form von Conrad-Guthaben erhalten würden, um ihre Projekte umzusetzen.

Ich mobilisierte Freunde, Verwandte und Bekannte und hatte schlussendlich genug Stimmen, um in die nächste Runde zu gelangen. Mit den 1000 CHF, welche ich zur Realisierung meiner Idee erhielt tat ich vorallem zwei Dinge. Zum einen kaufte ich mehrere Kilogramm Filament ein, da dieses essentiell war für die Grundbausteine meiner Arbeit, und die 35 CHF die ich bei Conrad “bezahlte”, sich im unteren Preissegment dieses eher teuren Guts befanden. Des Weiteren richtete ich mir eine kleine Werkstatt in meinem Zimmer ein.

Neben eher billigen Instrumenten wie Seidenschneider, Japanmesser und diversen Zangen besitze ich nun auch einige hochwertigere Instrumente, wie eine digitale Lötstation, ein Voltmeter (Spannungs/Stromstärken-Messgerät), ein Schraubenzieherset und ein Labornetzteil. Am 24. November wurden ich und die zwei Mit-Finalisten in die Conrad-Filiale Dietlikon eingeladen, um unsere Projekte zu präsentieren. Leider empfing ich aufgrund der hohen Dichte elektromagnetischer Strahlung im Geschäft Interferenzen bei der Signalübertragung zwischen Arduino Mega und RPP und konnte so mein Projekt nicht präsentieren.

Auf den darauffolgenden Mittwoch war die Deadline für die finale Stimmrunde angesetzt. Der Gewinner würde wieder 1000.- ausgezahlt bekommen. Es konnte jedoch kein klarer Gewinner gekürt werden. Mein Projekt hatte zwar am meisten Stimmen erhalten, jedoch hatte sich herausgestellt, dass bei einem der Teilnehmer etwa 800 von 950 Stimmen einen sehr verdächtigen Eindruck machten.

Conrad beschloss daher, sich angesichts der Unklarheiten und der zeitlichen Nähe zur Deadline an eine äusserst diplomatische und faire Vorgehensweise zu halten, nämlich wurden allen Finalisten 1000 CHF in Form eines Conrad-Gutscheines ausbezahlt.

## **D   Arbeitsmoral**

Eine weitere Grösse, mit welcher ich lernen musste einen Umgang zu finden, war die Motivation. Zu Beginn dieser Arbeit tat ich den in meinen Augen wichtigsten Schritt, was die Planung anbelangt. Ich entschied mich für ein Projekt, welches mehrere meiner Interessenbereiche verband und welches auch die Aneignung von Kompetenzen mit vielfältigem Nutzen voraussetzte.

Obwohl ich von mir behaupten kann, diese Entscheidung kein einziges Mal bereut zu haben gab es Momente, in welchen ich meine Gedulds- und Konzentrationskapazitäten ausgeschöpft wähnte. Vor allem die Male, wenn ich viel Zeit und Mühe investiert hatte, um eine Signalkette aufzustellen und dann nach zahlreichen vergeblichen Fehlersuchversuchen einsehen musste, dass es nicht funktionieren würde. Auch die Augenblicke bei der Bearbeitung eines gedruckten Bauteiles mit der Bohrmaschine oder bei der Montage verschiedener Elemente, wenn ich das unverwechselbare leise Knacken hörte, das nur bedeuten konnte, dass ein Bauteil zum xten Male an der Übergangsstelle einen Bruch aufwies, verlangten Frustrationstoleranz.

Ebenso die Montage der Finger, welche sich über fünf Teile-Generationen erstreckte und Mal für Mal daran scheiterte, dass ich es feinmotorisch einfach nicht hinbekam, eine Öffnung von 3mm auf eine Öffnung von 4mm aufzubohren, ohne dass ich das Material sprengte. Im Laufe der Arbeit entwickelte ich eine Resistenz gegenüber Rückschlägen.

Deprimierten mich die ersten paar Dutzend bis Hundert Error-Meldungen und das Trümmerfeld gescheiterter Filamentklumpen erst noch, stachelten sie mich schon bald dazu an, solange dran zu bleiben, bis entweder eine Lösung oder eine Alternative gefunden wurde.

Ich beobachte Tag für Tag in meiner Arbeit, dass sich der Arbeitsprozess meistens folgendermassen gliedert. Ich habe eine Idee, ich möchte sie umsetzen und stosse auf eine Kette von Problemen. Die Hauptarbeit, welche ich ab diesem Zeitpunkt leiste, besteht darin, mich Glied für Glied an dieser Problemkette entlang zu arbeiten. Würde der Optimismus fehlen, hinter dem nächsten Problem das Ende der Kette zu sehen und dies stundenlang, dann bliebe die Idee eine Idee.