



FAKULTA  
ŠTROJNÍ  
ČVUT V PRAZE

# Ústav mechaniky, biomechaniky a mechatroniky

Uchopovací mechanismus založený na  
kinematice lidské ruky

Gripping mechanism based on human hand  
kinematics

DIPLOMOVÁ PRÁCE

2022

**Bc. Viktor Budoš**

**Studijní program:** Aplikované vědy ve strojním inženýrství

**Studijní obor:** Mechatronika

**Vedoucí práce:** Ing. Jan Zavřel, Ph.D.

---

## Anotace

|                       |  |
|-----------------------|--|
| Jméno autora:         | Bc. Viktor Budoš   |
| Název práce           | Uchopovací mechanismus založený na kinematice lidské ruky  |
| Název v angličtině:   | Gripping mechanism based on humand hand kinematics   |
| Rok:                  | 2022   |
| Studijní program:     | Aplikované vědy ve strojním inženýrství  |
| Studijní obor:        | Mechatronika   |
| Ústav:                | Ústav mechaniky, biomechaniky a mechatroniky   |
| Vedoucí práce:        | Ing. Jan Zavřel, Ph.D.   |
| Bibliografické údaje: | Počet stran: 85<br>Počet obrázků: 124<br>Počet tabulek: 12   |
| Přílohy               | DVD obsahující kódy, simulační model a demonstrační videa.   |
| Klíčová slova         | Robotická ruka, Optimalizace úchopu, Genetický algoritmus, Senzory ohybu, Gyroskop, Arduino, Řídící rukavice, 3D tisk      |
| Key words:            | Robotic hand, Optimalization of gripping, Genetic algorithm, Flex sensors, Gyroscope, Arduino, Control glove, 3D printing. |

### Abstrakt:

Cíle práce je vytvoření rešerše uchopovacích mechanismů. Tvorba simulačního modelu v prostředí Matlab Simulink. Optimalizace simulačního modelu z hlediska úchopu zvolených předmětů. Tvorba řídicí rukavice pro ovládání simulačního modelu pomocí externího HW. Vytvoření reálného modelu za pomoci 3D tisku a jeho následným ovládáním.

### Abstract:

The goals of work are research of gripping mechanisms. Creating simulation model in environment Matlab Simulink. Optimalization of simulation model in the way of grasping chosen objects. Creating a controlling glove to control the simulation model via external hardware. Designing a real model using 3D printing technology.

---



## Prohlášení

Prohlašuji, že jsem tuto diplomovou vypracoval samostatně a použity byly pouze podklady uvedené v příloženém seznamu.

V Praze dne .....

.....

podpis

---



## Poděkování

Rád bych poděkoval své rodině, která mě podporovala při studiu. Dále bych rád poděkoval svému vedoucímu práce za vstřícnost a odborné vedení práce.

---

## Anotace

Diplomová práce se zabývá tvorbou uchopovacího mechanismu založeného na kinematice lidské ruky.

V první části je vypracována rešerše typů robotických hlavic a existujících modelů antropoidních mechanismů.

Ve druhé části je rozebrána tvorba modelu v prostředí Matlab Simulinku s využitím prostředí Simscape.

Ve třetí části je pro daný model provedena optimalizace z hlediska úchopů vybraných předmětů. Jako optimalizační metoda byl vybrán genetický algoritmus.

Ve čtvrté části je rozebrána tvorba řídicí rukavice, kde byl vybrán příslušný počítač a senzory. V kapitole je také rozebrána tvorba vlastních senzorů ohybu. Pro závěr kapitoly byly pořízeny video demonstrace řídicí rukavice, pomocí níž byl ovládán model simulační.

V poslední kapitole je rozebrána tvorba reálného modelu s pomocí 3D tisku, kdy je možné použití řídicí rukavice k jeho ovládní.

---

## Abstrakt

The diploma deals with making of gripping mechanism based on kinematics of human hand.

In the first part of the research are described robotics end-effectors and robotic anthropomorphic hands .

In the second part is described the proces of making simulation model in environment Matlab Simulink with using toolbox Simscape.

In the third part is created simulation model applied optimalization in the way of gripping certain objects. For this case was used a genetic algorithm.

In the following part was described proces of making a controll glove, where was also respectively choosen a microcontroller and sensors. In this part was also described a proces of making a bending sensors. For the conclusion was taken a video demonstration where was shown a controlling of simulation model via controlling glove.

The last part deal with creation of real model using 3D printing technology, where is possible to apply controlling glove to control the 3D printed model.

---

## Obsah

|  |    |
|--|----|
| 1. Úvod.....   | 1  |
| 2. Cíle práce .....  | 2  |
| 3. Teoretická část .....   | 2  |
| 3.1. Úchopné hlavice .....                                       | 2  |
| 3.1.1. Pasivní hlavice.....                                      | 2  |
| 3.1.2. Aktivní hlavice.....                                      | 4  |
| 3.2. Hlavice pro jemné manipulování .....                        | 7  |
| 3.3. Robotické hlavy s kinematikou lidské ruky.....              | 8  |
| 3.3.1. Robustní ruka.....  | 8  |
| 3.3.2. Biomimetická antropomorfní ruka .....                     | 9  |
| 3.3.3. Shadow hand .....   | 10 |
| 3.3.4. InMoov .....  | 12 |
| 4. Simulační model.....  | 13 |
| 4.1.1. Základní pohyby.....                                      | 13 |
| 4.2. Stavba modelu v prostředí Matlab a Simulinku/Simscape ..... | 16 |
| 4.2.1. Základní stavební bloky pro simulaci v Simscape .....     | 17 |
| 4.2.2. Modelování dlaně .....                                    | 18 |
| 4.2.3. Modelování prstů.....                                     | 19 |
| 4.2.4. Sestavení simulačního modelu celé ruky .....              | 22 |
| 4.3. Aktuace modelu.....   | 24 |
| 4.3.1. Komunikace bloků Simscape a Simulink .....                | 24 |
| 4.3.2. Aktuování článku 1.....                                   | 25 |
| 4.3.3. Aktuování článku 2 a 3 .....                              | 25 |
| 4.3.4. Torzní pružiny v prostředí Simscape .....                 | 27 |
| 5. Optimalizace .....  | 28 |
| 5.1. Simulační model .....                                       | 28 |
| 5.1.1. Buzení.....   | 28 |
| 5.1.2. Kontakt a vzdálenost .....                                | 29 |
| 5.1.3. Určení kontaktu .....                                     | 30 |
| 5.2. Optimalizační parametry.....                                | 30 |
| 5.3. Průběh optimalizace .....                                   | 31 |
| 5.3.1. Okrajové hodnoty parametrů .....                          | 31 |
| 5.3.2. Algoritmus .....  | 31 |
| 5.3.3. Popis optimalizace .....                                  | 33 |
| 5.3.4. Cílové funkce .....                                       | 36 |
| 5.4. Výsledky optimalizace pro úchop koule .....                 | 37 |
| 5.5. Výsledky optimalizace pro úchop kužele.....                 | 39 |
| 5.6. Výsledky optimalizace pro úchop krychle.....                | 41 |
| 5.7. Zhodnocení optimalizace .....                               | 44 |
| 6. Ovládání simulačního modelu pomocí externího HW.....          | 44 |
| 6.1. Mikropočítače a senzory .....                               | 44 |
| 6.2. Řídící rukavice.....  | 50 |



---

|        |  |    |
|--------|--|----|
| 6.3.   | Spojení řídicí rukavice s modelem..... | 54 |
| 6.4.   | Data ze senzorů natočení .....         | 55 |
| 6.5.   | Gyroskop .....                         | 57 |
| 6.6.   | Závěr kapitoly .....                   | 61 |
| 7.     | Fyzický experiment.....                | 63 |
| 7.1.   | Použité programy .....                 | 63 |
| 7.2.   | 3D tisk .....                          | 63 |
| 7.3.   | Vytvoření 3D modelů .....              | 64 |
| 7.3.1. | Dlaň.....                              | 64 |
| 7.3.2. | Provedení rotační vazby .....          | 66 |
| 7.3.3. | Tvorba článků prstů.....               | 66 |
| 7.3.4. | Torzni pružiny .....                   | 69 |
| 7.3.5. | Sestavení modelu.....                  | 70 |
| 7.3.6. | Stojan pro pohony.....                 | 71 |
| 8.     | Závěr.....                             | 80 |
|        | Seznam obrázků .....                   | 81 |
|        | Seznam tabulek .....                   | 83 |
|        | Použitá literatura.....                | 83 |

---



## 1. Úvod

Lidská ruka je komplexní uchopovací mechanismus, který se vyvíjel tisíce let evolucí, a jedná se proto o náročnou úlohu. Lidé se po staletí snaží vyrobit věrohodný funkční model lidské ruky, kde motivace plyne k postavení funkční protézy, nebo robotické hlavice v podobě a funkčnosti lidské ruky.

Robotických hlavic existuje mnoho a většinou jsou určeny k uchopování předmětů určitého tvaru a za předem stanovených podmínek. Změní-li se tvar předmětu či podmínky, robotická hlavice nemusí úkol zvládnout. Myšlenka použití antropoidní ruky je v její komplexnosti, lidská ruka dokáže uchopit skoro vše nehledě na geometrii, a to relativně s velkou silou stisku.

V dnešní době existuje celá řada robotických rukou, kde je snahou maximalizovat co nejvíce aspektů, kterými lidská ruka disponuje jako například agilitu, sílu stisku, nebo míru antropomorfismu.

Předmětem práce je postavit a optimalizovat simulační model robotické hlavice manipulátoru s kinematikou lidské ruky, kde se optimalizace modelu týká uchopování různých předmětů. S následným řízením modelu pomocí pohybů vlastní ruky. Poslední úkol práce je s pomocí 3D tisku vyrobit reálný model ruky.

## 2. Cíle práce

### Cíle práce jsou následující

- Prostudujte v současnosti používané modely ruky k uchopování.
- Navrhněte a dále optimalizujte vhodný model ruky pomocí simulace.
- Simulujte model ruky s možností ovládní pomocí externího HW v prostředí Matlab.
- Proveďte fyzický experiment s ovládaným mechanickým modelem.

## 3. Teoretická část

V teoretické části budou rozebrány jednotlivé typy robotických hlavic a existující antropoidní modely.

### 3.1. Úchopné hlavice

Úchopná hlavice je koncový prvek pohybového mechanismu, kde úchopnou hlavici můžeme rozdělit do několika skupin. Dělíme je podle toho, k jaké technologii ji použijeme, jakou sílu může hlavice vyvodit, nebo podle povahy styku s manipulovaným předmětem. V základu dělíme hlavice na aktivní a pasivní. [1]

#### 3.1.1. Pasivní hlavice

Pasivní hlavice jsou hlavice, které neovládají velikost úchopné síly při manipulaci s předmětem. Jako příklad můžeme uvést hlavici s přísavkou, permanentními magnety, odpruženými čelistmi, nebo gumový koncem. [1]

#### **Permanentní magnet**

Síla zdvihu je dána povahou parametry magnetu. Uvolnění předmětu se provádí pneumaticky, nebo najetím na doraz. Výhodou hlavice je jeho jednoduchá konstrukce. Robotickou hlavici s permanentním magnetem vidíme na obrázku 1.



Obrázek 1 Úchopná hlavice s permanentním magnetem [2]

### Deformační přísavky

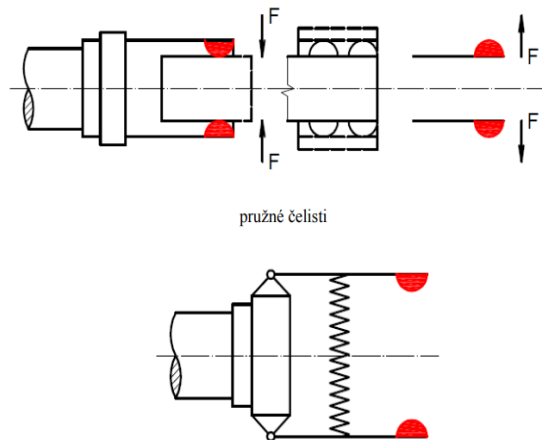
Robot přitlačí přísavky na manipulovaný předmět tak, že v nich vznikne podtlak. Uvolnění předmětu se provádí otevřením ventilu, který vyrovná tlak v přísavce s tlakem okolní. (Obrázek 2)



Obrázek 2 Přísavky pro robotickou hlavici [3]

## Čelisti

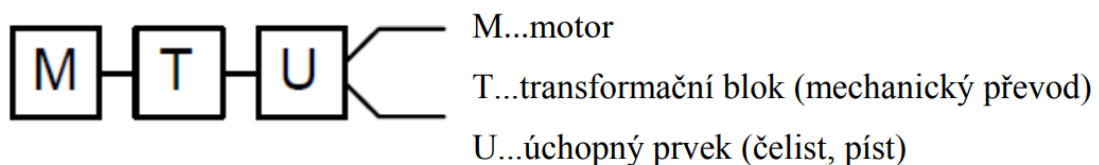
Jedná se o čistě mechanický prvek, který patří mezi nejjednodušší. Čelisti mohou být pevné, nebo odpružené.



Obrázek 3 Schéma hlavice s pevnou a odpruženou čelistí [1]

### 3.1.2. Aktivní hlavice

Aktivní hlavice oproti pasivní disponuje ovládáním velikosti úchopné síly. Tento typ hlavice můžeme složit z následujících bloků na obrázku 4.



Obrázek 4 Schéma bloků [1]

- Do bloku motoru můžeme zařadit – hydraulický, pneumatický, elektrický pohon.
- Převodový blok realizuje transformaci pohybu z motoru na úchopnou hlavici. Podle zvoleného typu převodu můžeme změnit smysl pohybu, rychlost a sílu. Do převodového bloku můžeme zařadit pákový, ozubený, vačkový, nebo šroubový převod.
- Uchopovací prvek je koncový prvek robotu, dělí se podle typu pohonu na elektromagnetický, elektrický, hydraulický, pneumatický či jejich kombinace. [1]

| Typ transform. bloku |      | Typ čelistí - vazba s transformačním blokem |                |
|----------------------|------|---|----------------|
|                      |      | Posuvné čelisti                             | Otočné čelisti |
| Páka                 | PTB  |   |                |
| Ozubení              | OTB  |   |                |
| Vačka                | VTB  |   |                |
| Šablona              | ŠATB |   |                |
| Šroub                | ŠRTB |   |                |

Obrázek 5 základní typy čelistí [1]

### Dvoučelistová hlavičce

Mezi nejzákladnější typy hlavic patří dvoučelistový mechanismus. Vyznačuje se jednoduchou konstrukcí a nízkou cenou. Vhodné pro manipulaci s předměty, které mají pravidelný tvar.



Obrázek 6 Příklad dvoučelistového robota [4]

### Tříčelistový

Dražší varianta než dvoučelistový mechanismus, ale disponuje lepším úchopem než dvoučelistový mechanismus. (Obrázek 7).



Obrázek 7 Tříčelistový mechanismus [5]

### Tříprstý mechanismus

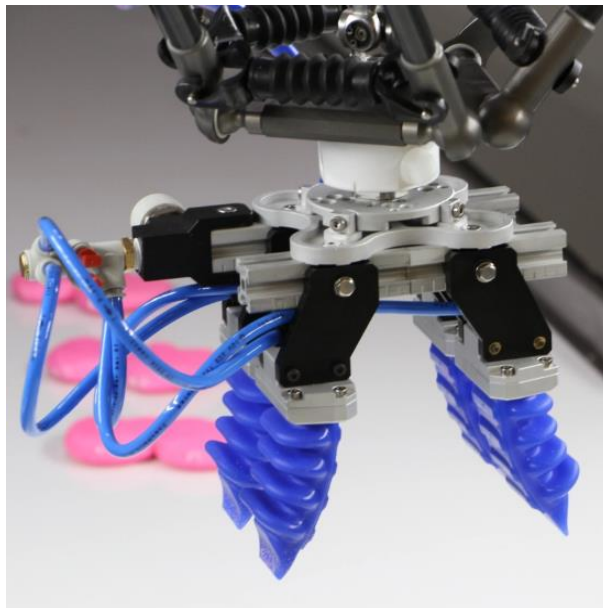
Mechanismus má místo pevných čelistí vložené rotační vazby tak, že připomíná lidské prsty. Mechanismus může potom lépe uchopovat předměty různých tvarů. Zmíněný model na obrázku 8 disponuje i silovými sensory na jednotlivých člácích prstů.



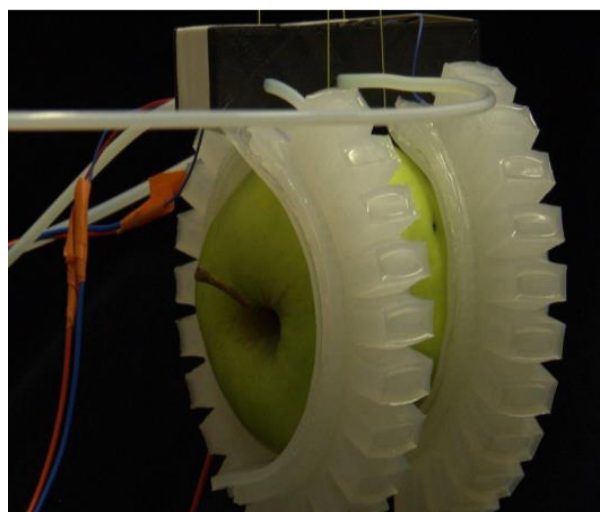
Obrázek 8 Tříprstý mechanismus [6]

### 3.2. Hlavice pro jemné manipulování

Jedná se o aktivní hlavice, které mají vzhledem k předešlým zmíněným hlavicím netradiční vzhled a způsob sevření. Jak vzhled napovídá, inspirací pro tento model byla chapadla chobotnice. Hlavice je vyrobena z poddajných materiálů jako je guma a neobsahuje žádné mechanické vazby. Chapadlo obsahuje několik kanálků, do kterých vstupuje tekutina a tím dochází k sevření. Tato hlavice nedisponuje velkou silou, ale je extrémně šetrná k uchopovanému předmětu. Proto je vhodná pro manipulaci například s ovocem a jinými potravinami, které jsou náchylné na otláčení. Hlavice pro jemnou manipulaci jsou na obrázcích 9 a 10.



*Obrázek 9 Hlavice pro jemnou manipulaci [22]*



*Obrázek 10 Hlavice pro jemnou manipulaci [21]*

### **3.3. Robotické hlavy s kinematikou lidské ruky**

Jedná se o typ robotické hlavice, která se vzhledem a kinematikou podobá lidské ruce. Lidská ruka je složitý nástroj, kterému se konstruktéři a výzkumní pracovníci snaží co nejvíce přiblížit. Lidská ruka je schopna jemné motoriky, takže se jedná o universální uchopovací hlavici, která zároveň dokáže vyvinout i značnou sílu. Například dle studie [7] dominantní ruka u muže disponuje silou stisku až 318 Newtonů. Naproti tomu dnešní modely dokážou vyvinout sílu stisku okolo desítek newtonů.

V dnešní době došlo k pokroku v oblasti antropomorfních rukou ať už v oblasti jemné motoriky, uchopování, nebo biomechaniky, a proto tyto robotické ruce mají potenciál proniknout více do průmyslu či zdravotnictví.

Rád bych zde uvedl některé existující modely antropoidních robotických rukou.

#### **3.3.1. Robustní ruka**

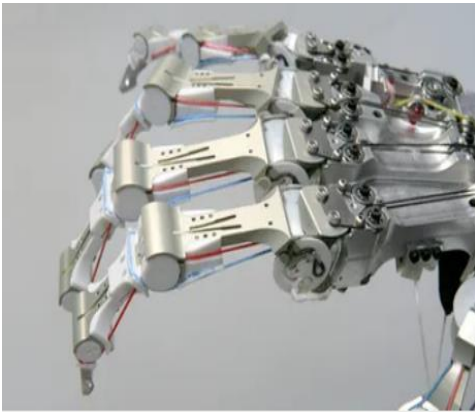
Ruka byla vyvinuta na institutu robotiky a mechatroniky v Německu (obrázek 11 a 12). Hlavním úkolem bylo vytvořit ruku, která bude schopna jemné manipulace a zároveň bude oproti ostatním modelům robustní.

Ruka se skládá z pěti prstů, které jsou propleteny třiceti osmi šlachami v podobě soustavy lanek, kdy každá šlacha má svůj vlastní motor. Ruka je schopna hýbat s každým článkem prstu individuálně. Každá šlacha má na svém konci, kromě motoru, také mechanismus s pružinou. Tento mechanismus zajišťuje právě tuhost ruky, kterou lze měnit. Ke změně tuhosti se využívá takzvaná antagonistická aktuace, kdy je každý článek poháněn dvěma lanky a pokud se oba motory otáčejí stejným směrem, dochází k pohybu. Pokud se motory otáčejí opačným směrem, narůstá tuhost.

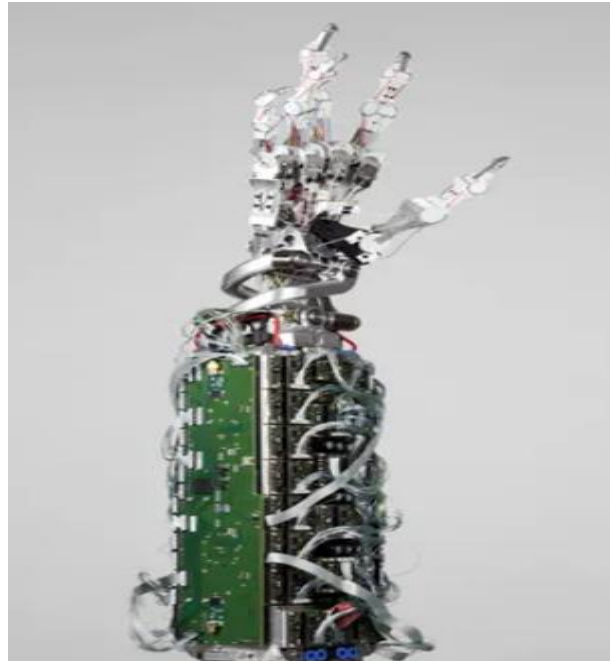
Ruka byla dokonce vystavena mnoha testům, kdy byla konfrontována nárazy kladivem či baseballovou pálkou a díky navýšení tuhosti článků ruka dokázala náraz absorbovat.

Ruka je řízena pomocí speciální rukavice, kdy je měřeno natočení kloubů operátora. Aby bylo možné jemné uchopování, ruka identifikuje, jestli je objekt měkký. Zda je objekt měkký identifikuje podle velikosti síly vyvinuté pružinami. [8]





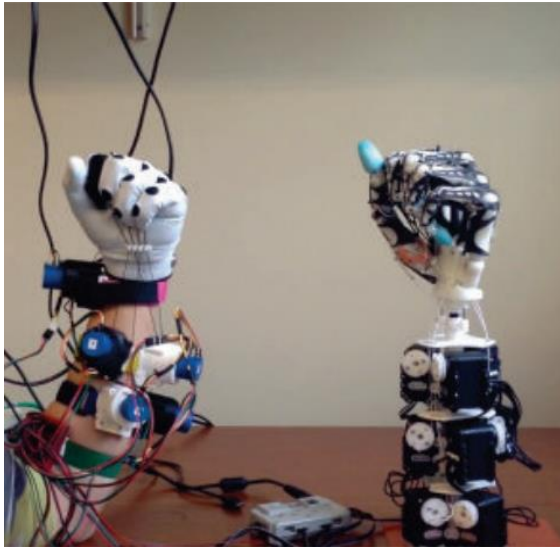
Obrázek 11 Robustní ruka [8]



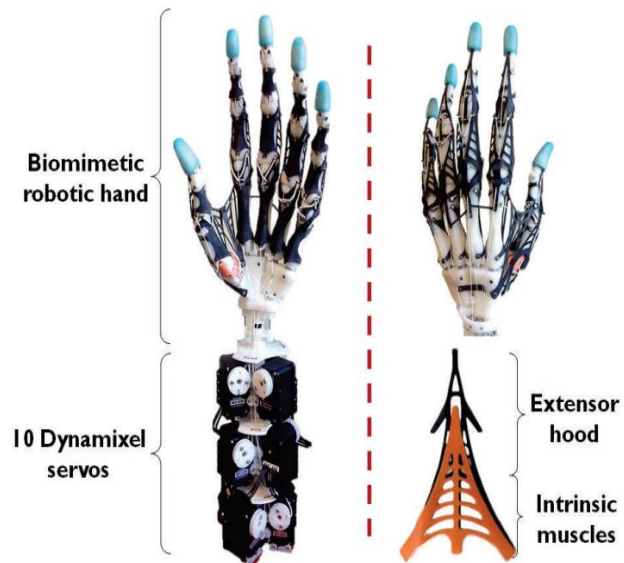
Obrázek 12 Robustní ruka [8]

### 3.3.2. Biomimetická antropomorfní ruka

Tento model ruky byl vyvinut na Washingtonské univerzitě, kdy záměrem bylo vyvinout ruku co nejvíce přesnou kinematice lidské ruky. Model vidíme na obrázcích 13 a 14. Základem bylo naskenování reálné lidské ruky a vytisknutí všech potřebných částí. Jejich spojení nebylo provedeno jako u klasických modelů, a to přidáním rigidní rotační vazby, ale pomocí vyrobených šlach. Tyto šlachy byly vyrobeny ze speciálních gumových pásků, které spojují dvě části dohromady, jako u reálné lidské ruky. K aktuaci bylo použito deset servomotorů. Ruka je ovládána speciální rukavicí, která udává natočení servopohonů. [9]



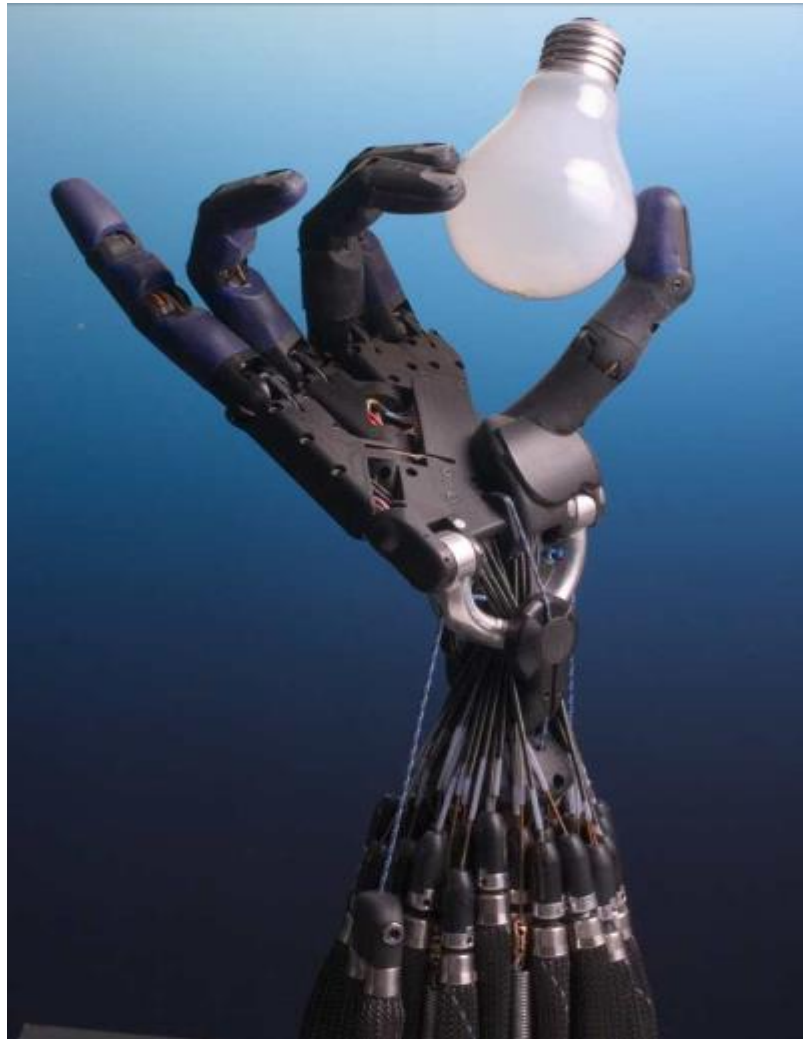
Obrázek 13 Biomimetic Anthropomorphic Robotic Hand [9]



Obrázek 14 Schéma – Biomimetic Anthropomorphic Robotic Hand [9]

### 3.3.3. Shadow hand

Další z modelů je ruka představená firmou Shadow Robot, která se rozměrově podobá lidské ruce (Obrázek 15). K pohonu prstů jsou použity pneumatické pohony, kterých se v modelu nachází čtyřicet. Díky nim dokáže ruka provést až dvacet čtyři různých pohybů, čímž se výrobci snažili co nejvíce napodobit pohyby lidské ruky. Na špičkách prstů se nachází senzory tlaku. Ruka je schopna udržet závaží až pět kilogramů a umožňuje jemné manipulace s předměty.



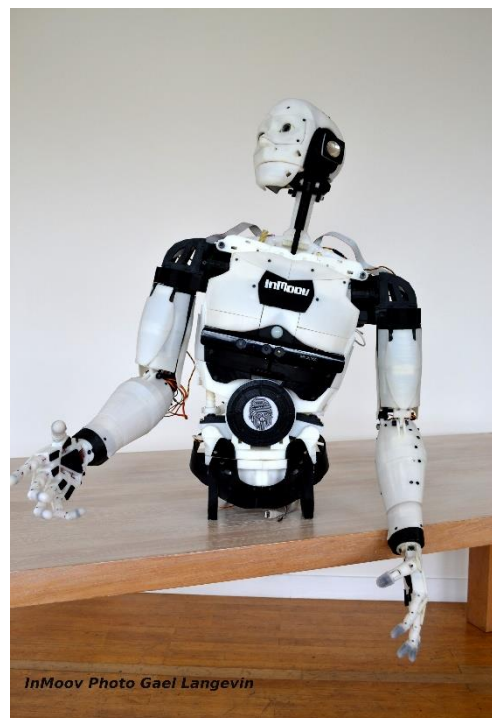
*Obrázek 15 Shadow hand [23]*

### 3.3.4. InMoov

Jedná se o open source projekt, založený francouzským sochařem Gael Langevin, který staví model funkčního robota za pomoci 3D tisku (obrázek 17). Součástí robota je samozřejmě ruka (Obrázek 16). Model je poháněn pomocí lan a servopohonů, kdy je přidána ke každému pohonu pružina pro vyvození předpětí. Na každý prst připadá pouze jeden servopohon, takže se jedná o plně podaktuovaný systém. Na dlani jsou navíc přidány gumové podložky pro zlepšení úchopu.



Obrázek 16 Inmoov detail ruky [24]

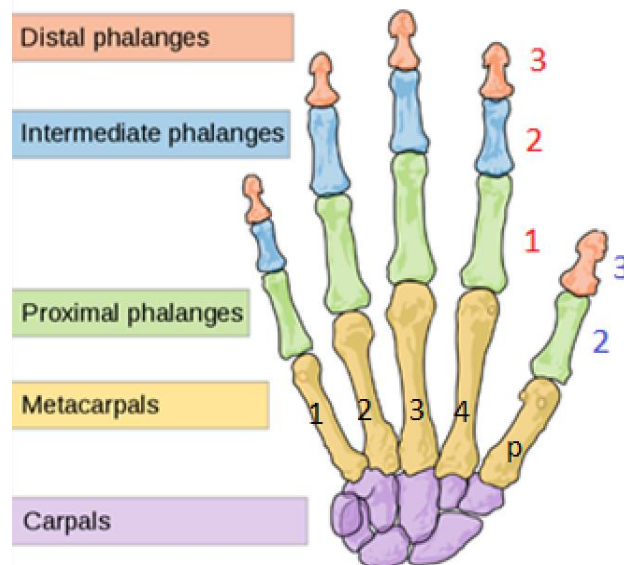


Obrázek 17 Inmoov robot [24]

## 4. Simulační model

Lidská ruka, jak již bylo zmíněno, je složitá a komplexní záležitost. Při tvorbě simulačního modelu musí být brán zřetel na to, že bude podle simulačního modelu zkonstruován model reálný. V rámci toho, aby to bylo možné, budou jisté stupně volnosti zanedbány.

V rámci modelu budou modelovány kosti distálních, mediálních a proximálních phalangů, karpální a metakarpální kosti budou modelovány v rámci jednoho tělesa s počátkem souřadnicového systému. Výjimkou bude metakarpální kost palce, která bude modelována spolu s ostatními částmi palce (Obrázek 18).



Obrázek 18 Kosti v lidské ruce [25]

### 4.1.1. Základní pohyby

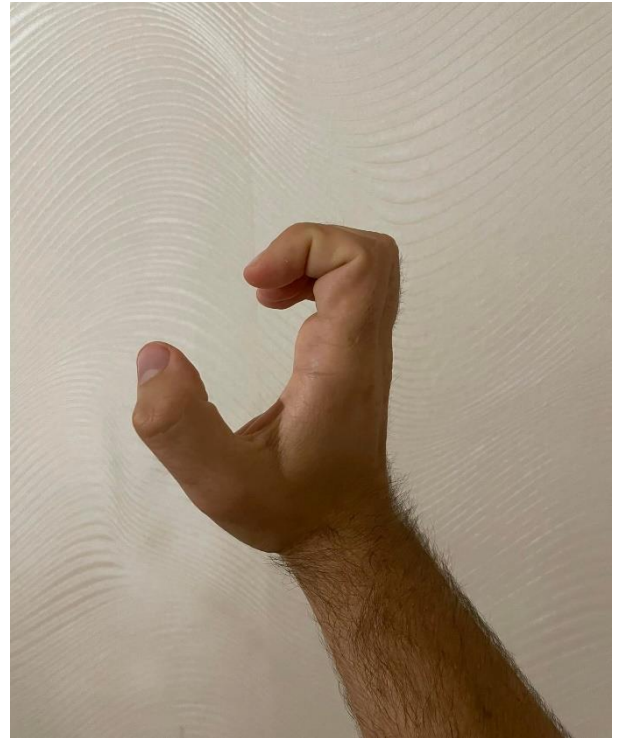
Při tvorbě modelu jsem uvažoval tyto pohyby. Na obrázku 18 byly pro jednoznačnost očíslovány červeně a jednotlivé články prstů, kde číslování modrou barvou značí články palce. Posléze byly očíslovány černě i jednotlivé prsty, kdy palci místo čísla přísluší písmeno p. Tyto označení jsou důležitá pro optimalizační část, kdy jednotlivé matice a vektory jsou označeny pouze pomocí těchto čísel/ písmen.

#### Sklápění prstů

Při sklápění prstů, myšleno malíček až ukazováček, zjistíme, že je tento pohyby složen ze dvou pohybů - (Obrázek 19 a 20). První dva články (3 a 2) dokážeme sklopit, aniž bychom sklopily spodní článek. Jelikož jedním pohybem naklápíme dva články, dvě tělesa, lze říci, že se jedná o podaktuovaný systém. Spodní článek dokážeme natočit nezávisle na zbylých dvou. Spojením těchto pohybů dokážeme věci sevřít, nebo vykonávat různá gesta. Tyto dva různé pohyby jsou zahrnuty v modelu.



*Obrázek 19 Pohyb prstů 1*



*Obrázek 20 Pohyb prstů 2*

### **Naklápění prstů vůči sobě**

Významným pomocníkem při uchopování předmětů je toto naklopení, kdy se ruka dokáže přizpůsobit velikosti předmětu za účelem jeho úchopu.

Tento pohyb, ač je důležitý, jsem zanedbal z hlediska složité implementace do fyzického modelu. (obrázek 21 a 22)



*Obrázek 21 Prsty v klidu*



*Obrázek 22 Natočení prstů do stran*

### Sklápění palce

Sklápění palce se také skládá ze dvou nezávislých pohybů. První pohyb na obrázku 24 je sklopení prvních dvou článků. A Na obrázku 23 je pohyb natočení posledního článku, kde se jedná o natočení metakarpální kosti palce.



Obrázek 24 Pohyb palce 1



Obrázek 25 Pohyb palce 2



Obrázek 23 Pohyb palce 3

### Naklápění palce

Palec dále dokážeme natáčet jako na obrázcích 26 a 27. Tento pohyb nakonec nebyl zaveden do modelu.



Obrázek 26 Pohyb palce 4



Obrázek 27 Pohyb palce 5

## 4.2. Stavba modelu v prostředí Matlab a Simulinku/Simscape

### Matlab

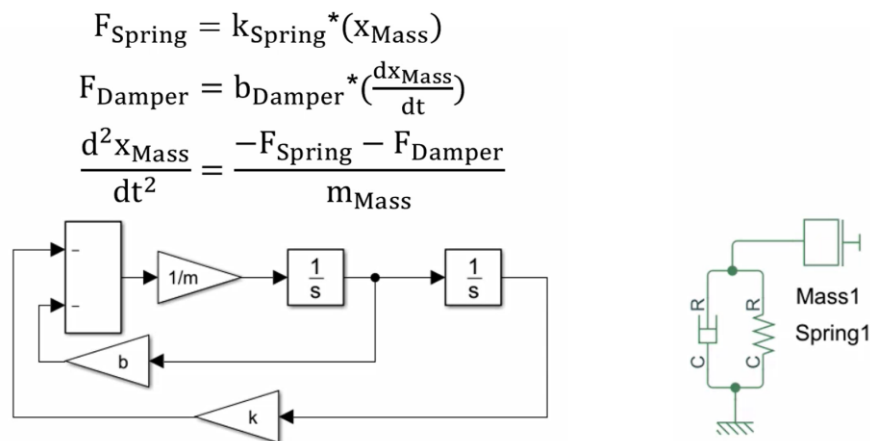
Matlab je programovací skriptovací jazyk čtvrté generace a je vyvíjen společností MathWorks. Zkratka jazyka matlab (matrix laboratory) znamená v překladu maticová laboratoř, což reflektuje skutečnost, že matlab používá matice jako hlavní strukturu. Umožňuje tedy manipulace s maticemi, vykreslování funkcí a dat, vytváření algoritmů a dokáže také komunikovat s jinými programy či hardwary jako například Arduino.

### Simulink

Jedná se o balíček matlabu, který byl vyvinut pro simulaci dynamických systémů. Funguje na bázi Matlabu, ale místo skriptovacího jazyka se používá grafické/blokové programování.

### Simscape

Jedná se o rozšíření pro Simulink. Kde v rámci tohoto rozšíření lze modelovat elektrické, hydraulické, pneumatické, mechanické a multifyzikální systémy v rámci programu Simulinku. Na obrázku 28 můžeme vidět rozdíl mezi modelováním hmoty a tlumiče s pružinou, kde na levo se nachází bloky sestavené v simulinku a napravo v simscapu. [10] [11]



Obrázek 28 Porovnání Simulink/Simscape [11]



#### 4.2.1. Základní stavební bloky pro simulaci v Simscape

Pro simulaci v Simscape jsou esenciální tyto tři bloky. (Obrázek 29)

##### Mechanism Configuration

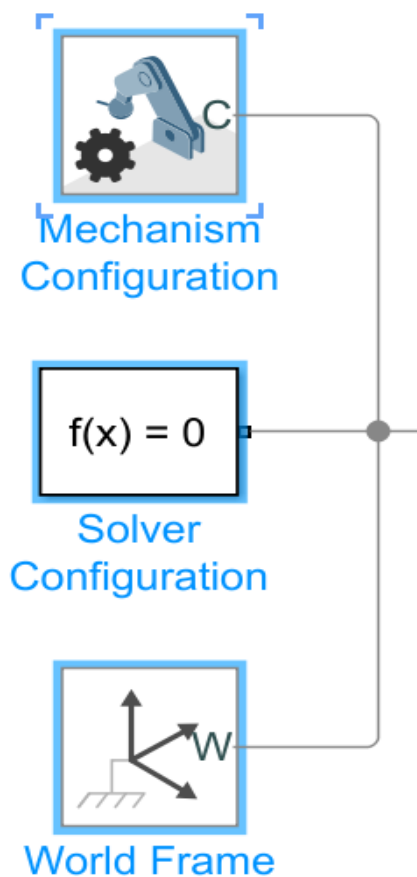
Tímhle blokem lze nastavit gravitační zrychlení pro všechny nadcházející stavební bloky v programu Simscape multibody.

##### Solver Configuration

Každý simulační model musí obsahovat právě jeden blok, který obsahuje informace o použitém řešiči, které model potřebuje před zahájením simulace.

##### World Frame

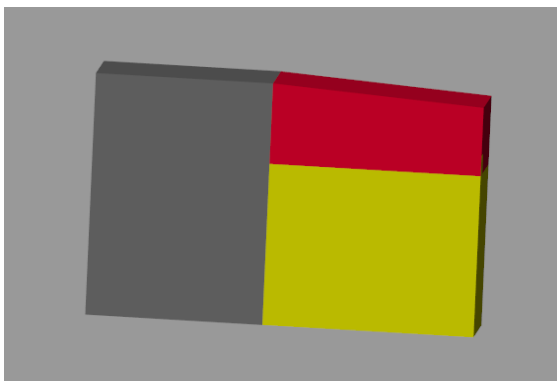
Poslední z trojice základních bloků je nastavení výchozího souřadnicového systému, z kterého budou vycházet ostatní souřadnicové systémy.



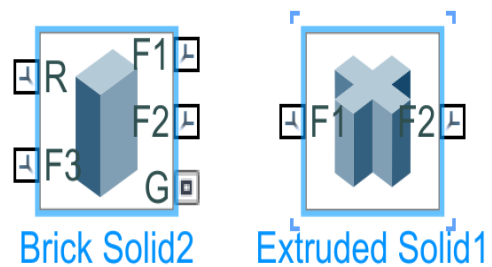
Obrázek 29 Základní bloky

#### 4.2.2. Modelování dlaně

Jako aproximaci dlaně jsem zvolil kvádr, který bude mít z jedné strany sraženou hranu, abych více napodobil tvar dlaně, která není z vrchu rovná. Dlaň se skládá z šedé části, která má rovný vrch a obsahuje počátek souřadnicového systému. Šikmou část jsem sestavil ze dvou bloků (Obrázek 30). Bloky byly sestaveny pomocí funkce brick solid a šikmá část pomocí extruded solid (Obrázek 31). Rozměry dlaně jsou  $L = 87 \text{ mm}$ , šikmá část klesá s úhlem  $\alpha = 5^\circ$ , délka rovné části  $L_r = 40 \text{ mm}$ , výška dlaně  $h = 60 \text{ mm}$  a tloušťka  $t = 15 \text{ mm}$ .



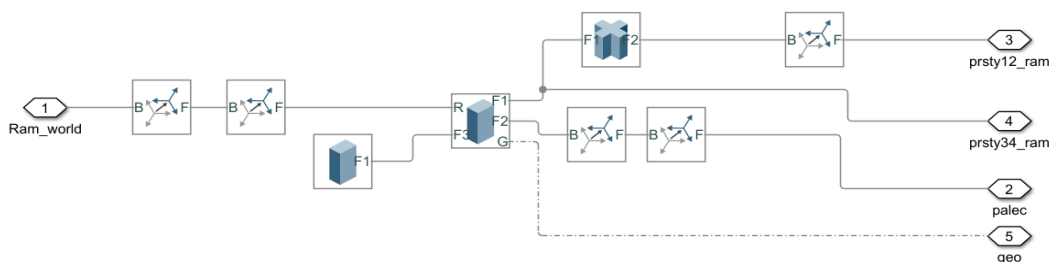
Obrázek 31 model dlaně



Obrázek 30 Bloky pro sestavení dlaně

Tabulka 1 Rozměry dlaně

|                          |                       |
|--------------------------|-----------------------|
| Šířka dlaně              | $L = 87 \text{ mm}$   |
| Úhel klesání šikmé části | $\alpha = 5^\circ$    |
| Výška dlaně              | $h = 60 \text{ mm}$   |
| Délka rovné části        | $L_r = 40 \text{ mm}$ |
| Tloušťka dlaně           | $t = 15 \text{ mm}$   |



Obrázek 32 Sub systém dlaně

Na obrázku 32 vidíme sub systém dlaně, který je zleva spojen s rámem a napravo jsou vyvedeny různé souřadnicové systémy. Ze solidů použitých k sestavě dlaně, jsou vyvedeny různé souřadnicové systémy nacházející se onom solidu ( $R, F1, F2, F3$ ). Použitím bloků Rigid Transform, který si můžeme představit jako transformační matici, použijeme k vyvedení souřadných systémů pro prsty.

Jedná se vlastně o předpřipravení souřadnicových systémů, kde vymodelovaný prst již jenom přiřadíme ke správnému portu. Například pro ukazováček, nebo prostředníček použijeme port *prsty\_12\_ram* (Obrázek 32). Tyto dva prsty budou od sebe posunuty, kdy jsem vzdálenost mezi všemi prsty volil 16 mm.

#### 4.2.3. Modelování prstů

Články budou modelovány jako válce, kde jsem vycházel z průměrů vlastních prstů. Zvolené délky a průměry jsou uvedeny v tabulce 2, 3 a 4. Číslování prstů jsem volil od dlaně směrem ke konečku prstu.

Tabulka 2 Průměry článků

|                | 1. článek | 2. článek | 3. článek |
|----------------|-----------|-----------|-----------|
| Průměry článků | 16 mm     | 14 mm     | 12 mm     |

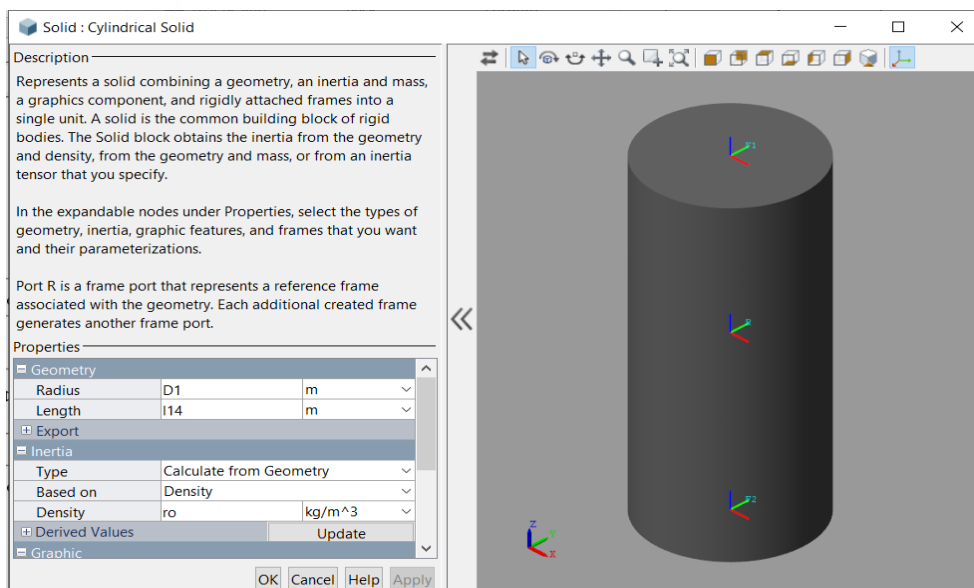
Tabulka 3 Délky druhých a třetích článků.

|              | 2. článek | 3. článek |
|--------------|-----------|-----------|
| Délky článků | 30 mm     | 28 mm     |

Tabulka 4 Délky 1. článků

|                 | Malíček | Prsteníček | Prostředníček | Ukazováček |
|-----------------|---------|------------|---------------|------------|
| Délky 1. článků | 30 mm   | 35 mm      | 45 mm         | 38 mm      |

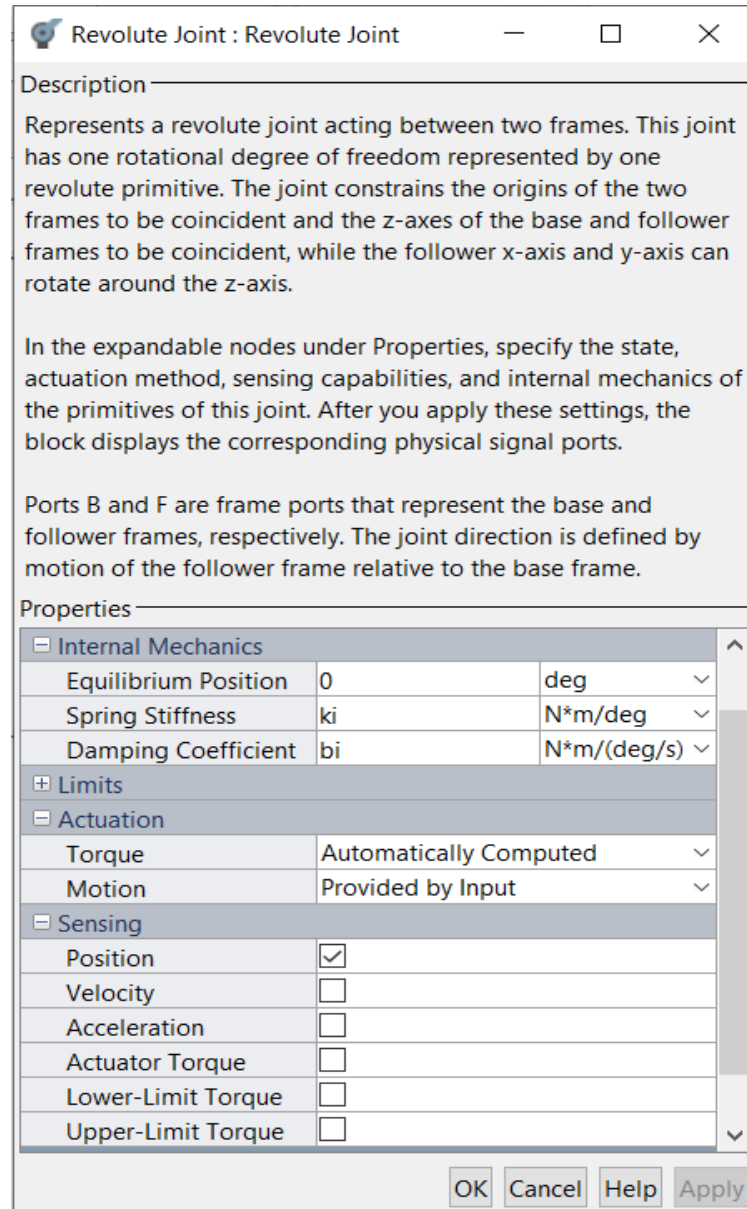
Jako hodnotu hustoty pro válec byla zvolena hodnota  $1600 \text{ kg m}^{-3}$ , která odpovídá hustotě plastu PLA, ze kterého bude daný model tisknut. Použitý blok pro tvorbu článku vidíme na obrázku 33.



Obrázek 33 Cylindrical solid

## Rotační vazba

Rotační vazbu mezi články zprostředkoval blok Revolute Joint. Blok přidá stupeň volnosti do systému, a to konkrétně rotaci pouze kolem osy Z. Je důležité, aby souřadnicové systémy na konci článku byly správně natočené, aby rotace mohla být uskutečněna v požadovaném směru.



Obrázek 34 Nastavení parametrů kloubu

Pro lepší simulační chování systému byly do vazeb zaneseny malé hodnoty tuhosti  $k_i$  a tlumení  $b_i$  o velikosti  $1e-6$  Nm/deg a  $1e-7$  Nm/(deg/s). Kloub lze také aktuovat, a to buď zadáváním pohybu, nebo zadáváním momentu. Lze také snímat natočení, zrychlení a tak dále viz. (Obrázek 34).

### Bříška prstů

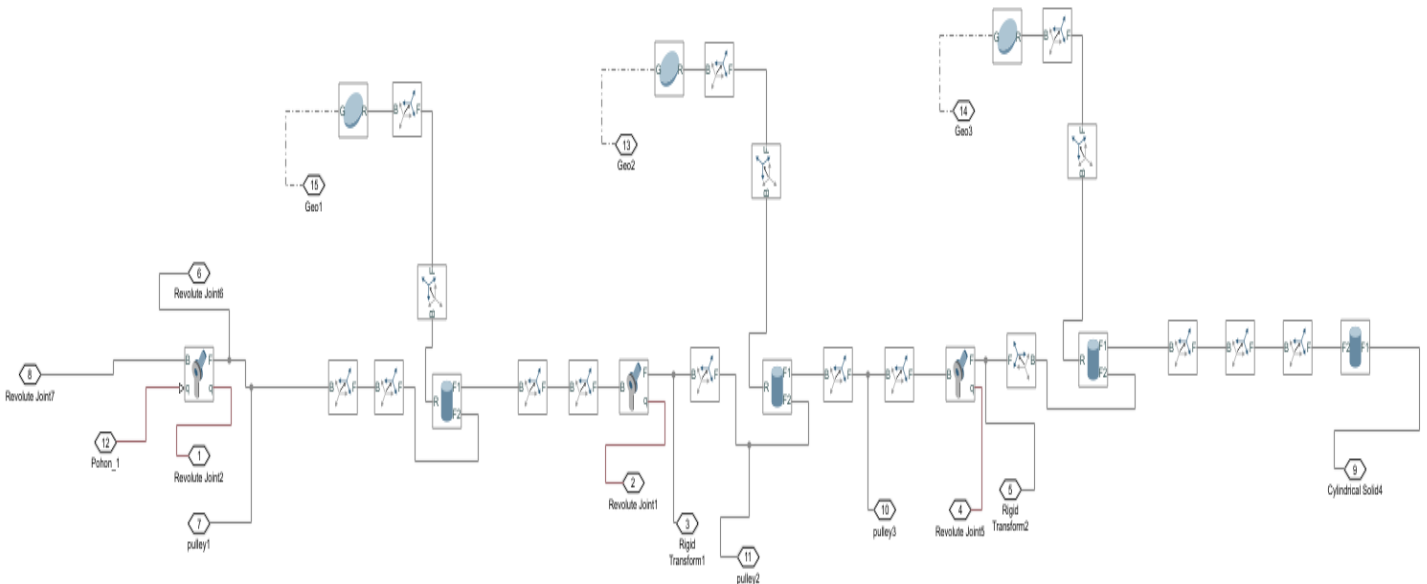
Jednou z komponent každého článku na prstu je cylindrický solid vložený do článku (Obrázek 35). Mezi tímto prvkem a prvkem uchopovaným je vyvedena geometrie, která je spojená pomocí silového bloku Spatial Contact Force, který měří vzdálenosti mezi vytvořenými geometriemi. Více v kapitole 5.1.2.



Obrázek 35 Ellipsoidal solid blok

### Sestavení prstu

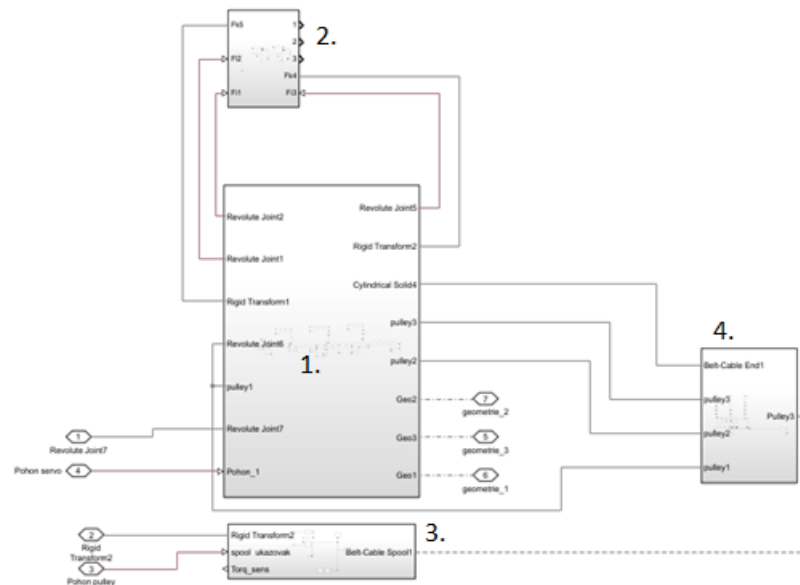
Po sestavení všech stavebních bloků prstu (Obrázek 33) a správném natočení osy z pomocí bloků Rigid Transform spojíme články rotační vazbou (Obrázek 34). Přidáním bříšek prstů (Obrázek 35) k jednotlivým článkům a přidáním koncového solidu pro uchycení lana (viz. kapitola 4.3), dostaneme stavební blokové schéma prstu (Obrázek 36). Náhled na 3D obrázek prstů můžeme vidět na obrázku 39.



Obrázek 36 Část subsystému prstů se stavebními bloky prstu

### Složení prstů z jednotlivých subsystémů

Na obrázku 37 můžeme vidět kompletní složení prstu z jeho jednotlivých subsystémů, které jsem pro přehlednost očísloval od jedné do čtyř. Kde subsystém 1 je subsystém se stavebními bloky (Obrázek 36). Subsystém 2 reprezentuje torzní pružiny v kloubech a je popsán v kapitole 4.3. Subsystémy 3 a 4 jsou také popsány v kapitole 4.3 a reprezentují systém lana, kladek a lanového bubnu.

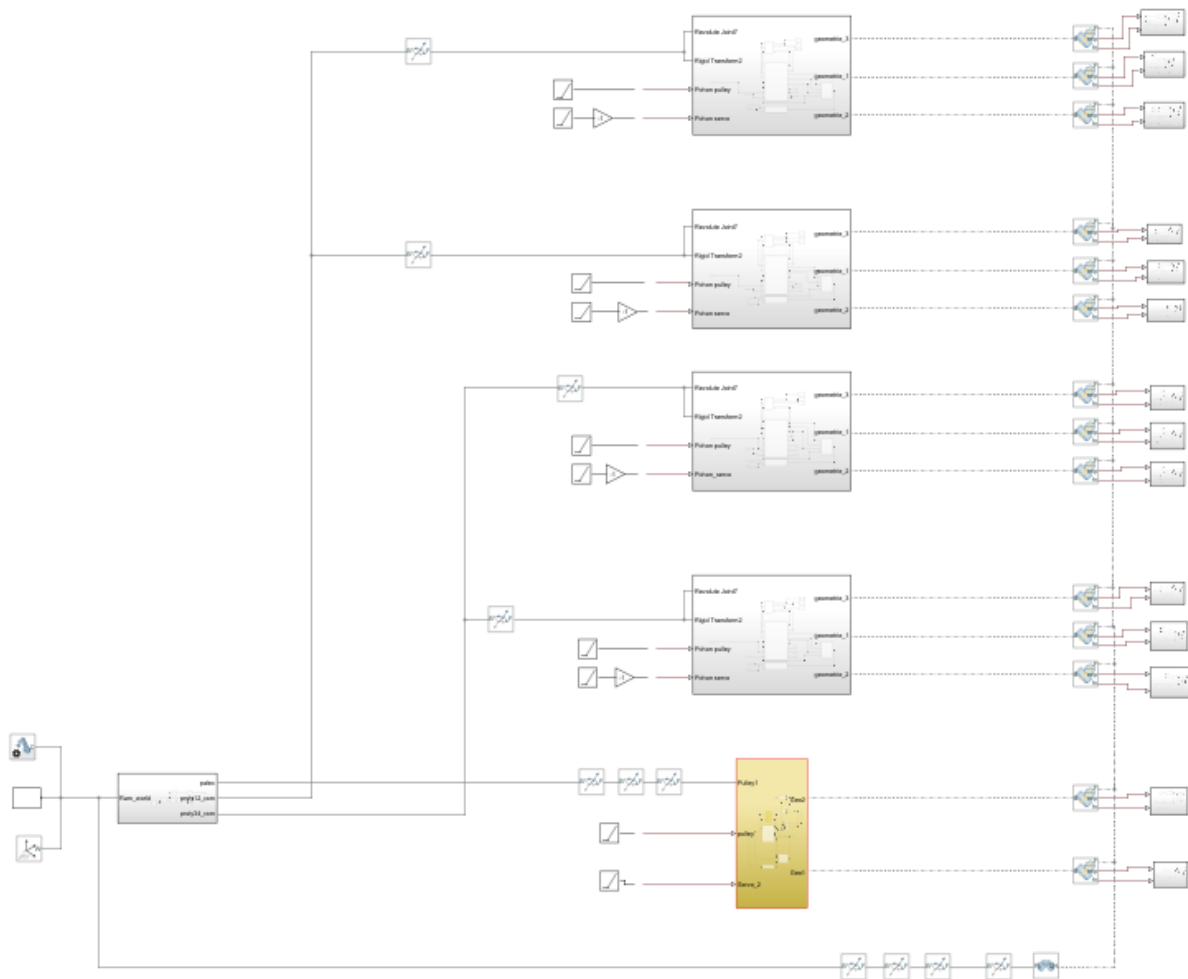


Obrázek 37 Subsystém prstu

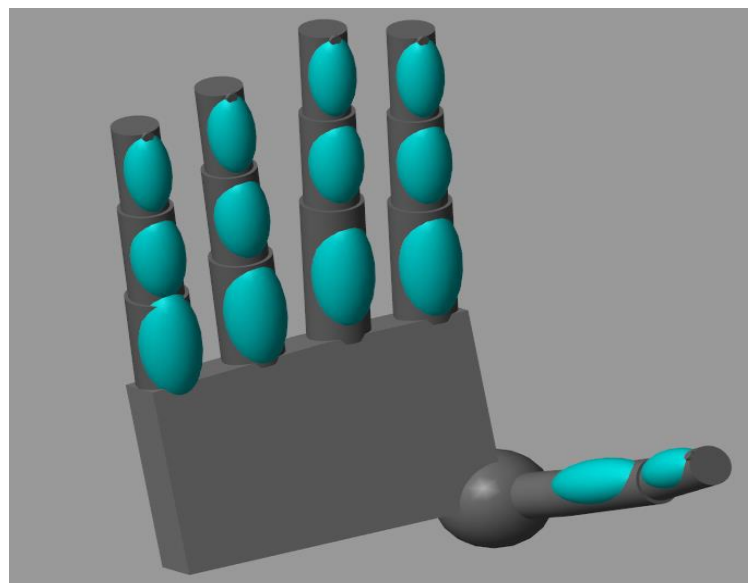
#### 4.2.4. Sestavení simulačního modelu celé ruky

Na obrázku 38 vidíme kompletní blokové složení simulačního modelu celé ruky a na obrázku 39 vidíme vizuální podobu simulačního modelu. V levé části obrázku 38 vidíme základní bloky, které jsem uvedl v kapitole 4.2.1. Vedle nich se nachází blok s modelem dlaně, který jsem zmínil v 4.2.2. V Pravé části vidíme subsystémy jednotlivých prstů, kde spodní subsystém reprezentuje palec. Nad palcem se nachází subsystém ukazováčku a horní subsystém reprezentuje malíček. Vnitřek těchto subsystémů je vidět na obrázku 37.

Z jednotlivých subsystémů vystupují čárkované čáry, které reprezentují geometrii předmětu. Tyto geometrie jsou důležité pro snímání kontaktu s předměty, které geometrii sdílí, více v kapitole 5.1.2.



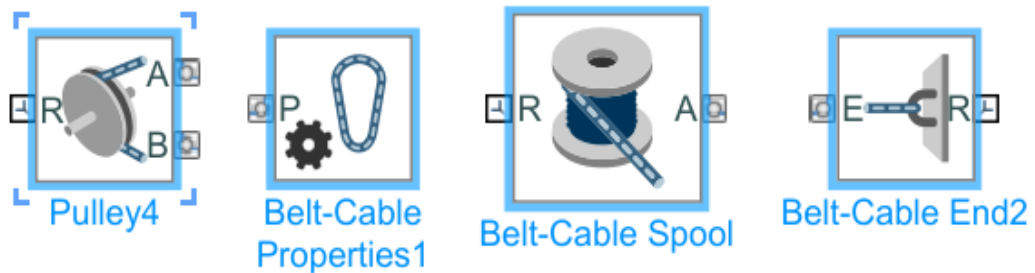
Obrázek 38 Simscape model – blokové schéma modelu dlaně



Obrázek 39 3D náhled simulačního modelu

### 4.3. Aktuace modelu

V kapitole 4.1.1 bylo uvedeno, že budou poháněny články 3 a 2 jedním pohonem a článek 1 bude mít pohon samostatný. Ke sklápění dvou článků bylo využito lano, jehož jeden konec bude upevněn na koncovém článku a druhý bude navíjen na buben. Lano povede skrze prsty a bude vedeno pomocí kladek, které Simscape toolbox umožňuje použít spolu s lanem a lanovým bubnem.



Obrázek 40 Bloky pro vedení lana

#### 4.3.1. Komunikace bloků Simscape a Simulink

Pro propojení mezi bloky Simulinku a Simscapu, je potřeba použít Simulink-PS Converter. Naopak pro připojení bloků z prostředí Simscape do bloků Simulinku je zapotřebí Simulink-PS Converter. Zahrnuje-li blok pohyb, je potřeba také v Converteru nastavit zahrnutí přepočtu prvních a druhých derivací.



Obrázek 41 Simulink  
to simscape converter

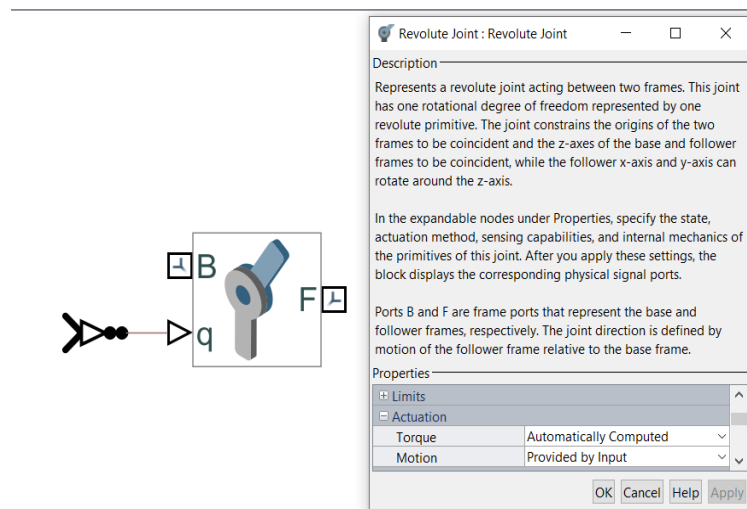


Obrázek 42 Simscape to simulink  
converter



### 4.3.2. Aktuování článku 1

Buzení prvního článku bude kinematické. Na obrázku 43 v bloku kloubu v sekci Actuation se nachází podsekce Torque (moment) a Motion (pohyb). Jelikož je buzení kinematické, pro pohyb zadáme Provided by Input (poskytnuto vstupem) a pro moment zvolíme jeho automatické dopočítávání. Aby bylo možné přivést vstup z bloků Simulinku, musíme předřadit před blok Simulink – PS Converter, a protože se jedná o pohyb, zahrneme v něm přepočítání první a druhé derivace, který je na obrázku 43 zvýrazněn tečkami za šipkou Converteru.



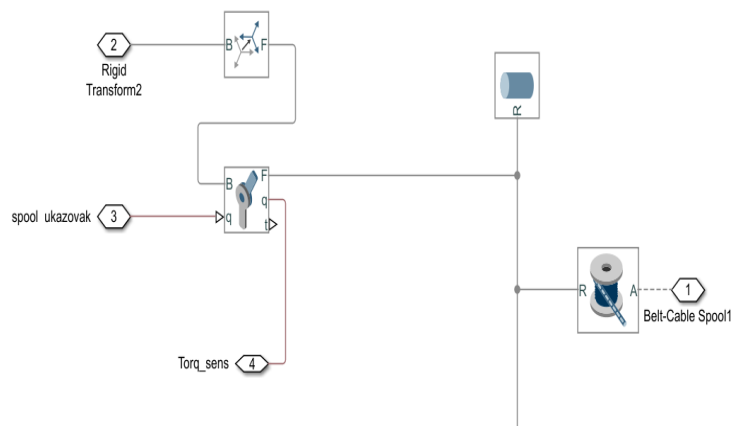
Obrázek 43 Aktuace kloubu

### 4.3.3. Aktuování článku 2 a 3

Lano samotné je tuhé a spojením jednotlivých bloků na obrázku 40 se lano automaticky vytvoří. Každé lano v systému musí mít svůj vlastní blok Belt-Cable Properties, který specifikuje použité lano.

#### Lanový buběn

Pro blok lanového bubně je třeba zvolit jeho průměr a přiřadit k němu fyzickou část, a to cylindrical solid. Pro model byl zvolen průměr bubně 16 mm. Dále je třeba přidat rotační kloub – revolute joint. Pomocí kloubu můžeme bubně pohánět a tím odvíjet či navíjet lano. Poslední zbývá jen připojit bubně k lanu. Kinematické buzení lana proběhne stejným nastavením kloubu jako v 4.3.2. Subsystem lanového bubně můžeme vidět na obrázku 44.

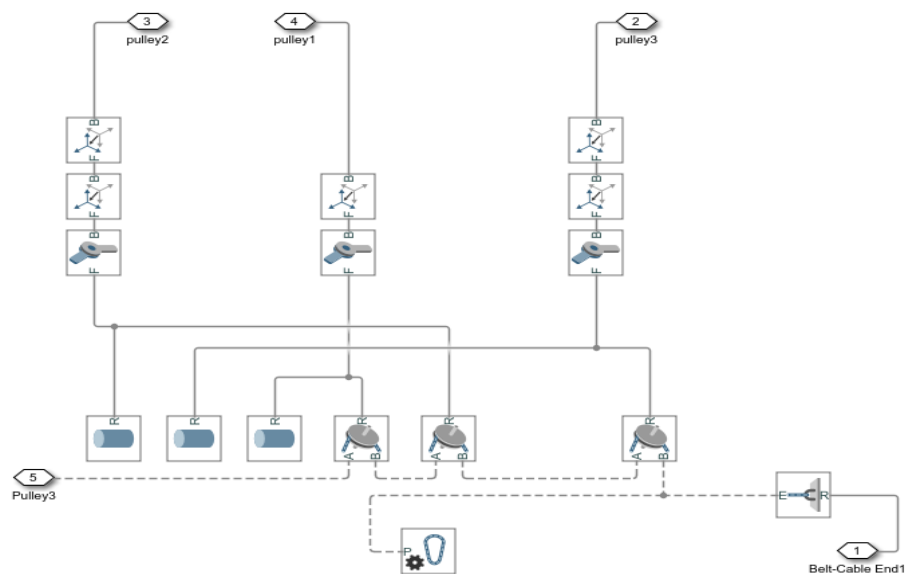


Obrázek 44 Subsystém lanového bubnu

### Kladka + ukotvení lana

Pomocí kladek, můžeme vést lano kudy potřebujeme. Pro použití tohoto bloku je nutné mu přiřadit rotační kloub a fyzickou část jako u lanového bubnu.

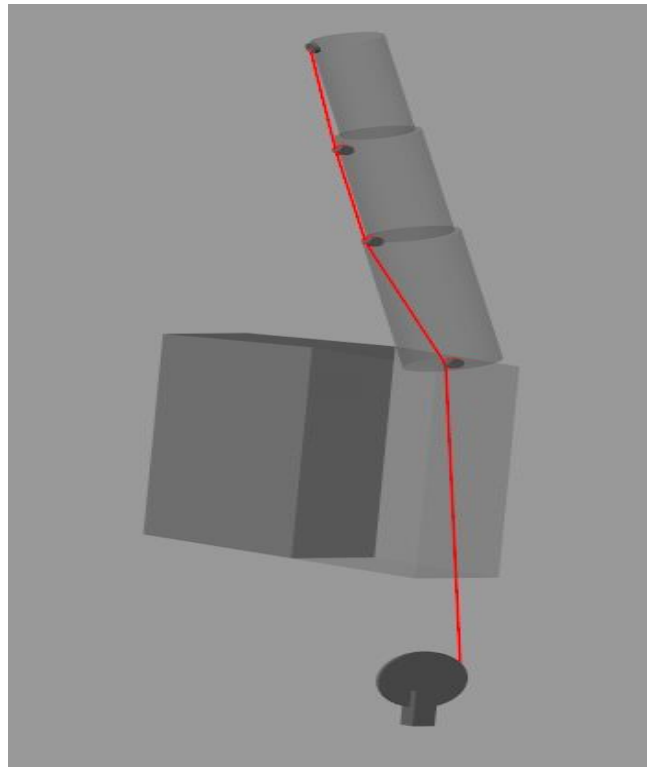
Nakonec je lano z kladek vyvedeno do koncového bloku Belt-Cable End, se kterým je svázáno viz. obrázek 45. Tento blok je umístěn na koncovém článku prstu (Obrázek 36 – poslední solid z prava) a díky němu je realizováno sklápění.



Obrázek 45 Schéma – Kladka + Belt-Cabe End

### Lanový okruh

Na obrázku 46 můžeme vidět sestavený lanový okruh. Úplně dole se nachází lanový buben, jehož natočením se navine část lana vedeného skrze kladky a vyvolá reakci pomocí koncového bloku (Belt-Cable End), a díky této reakci se dá prst do pohybu. Model je tedy buzen kinematicky zadáváním úhlu natočení lanového bubnu.

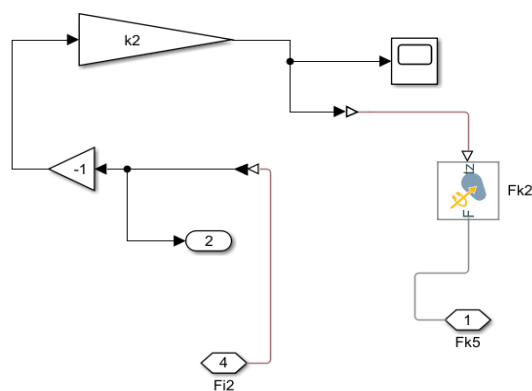


Obrázek 46 Prst-vedené lanového pohonu

#### 4.3.4. Torzní pružiny v prostředí Simscape

Vzhledem k tomu, že pohon z 4.3.3. je podaktuovaný, přidal jsem do prvních dvou kloubů torzní pružiny, které změnou tuhosti mění povahu sklápění.

Z kloubů je snímáno natočení v radiánech, které je poté násobenou tuhostí dané pružiny. Tato výsledná hodnota je pak poslána do bloku External Force and Torque viz. obrázek 47, která aplikuje moment na daný kloub v opačném směru pohybu.

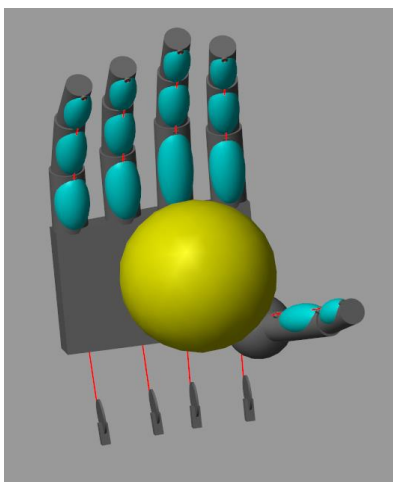


Obrázek 47 Simscape – pružina

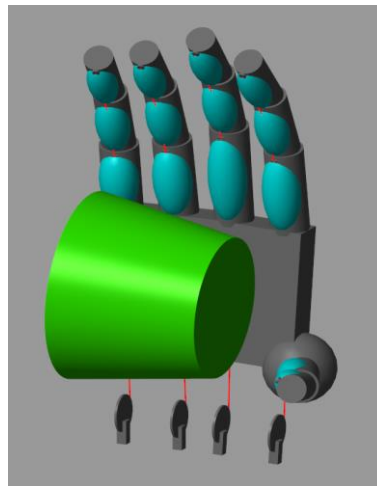
## 5. Optimalizace

V rámci práce byla použita optimalizace na daný simulační model, kde myšlenka optimalizace byla, aby simulační model ruky co nejlépe uchopil daný předmět. Dospěl jsem k tomu, že aby byl předmět uchopen co nejlépe, tak je důležité, aby se všechna bříška článků prstů dotkla předmětu zároveň, nejlépe v jeden okamžik. Aby nedocházelo k tomu, že se jeden článek dotkne výrazně dříve, než ostatní což by mohlo vést k posunutí předmětu a zmaření celého úchopu. Rozhodl jsem se tedy pro minimalizaci vzdáleností bříšek prstů od předmětů při prvním kontaktu s předmětem.

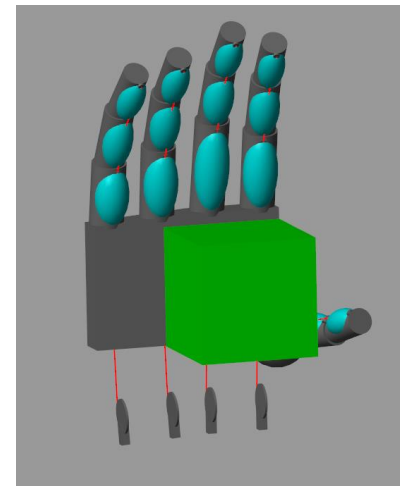
Uchopované předměty budou koule, komolý kužel a krychle.



Obrázek 48 Ruka – koule



Obrázek 50 Ruka – kužel



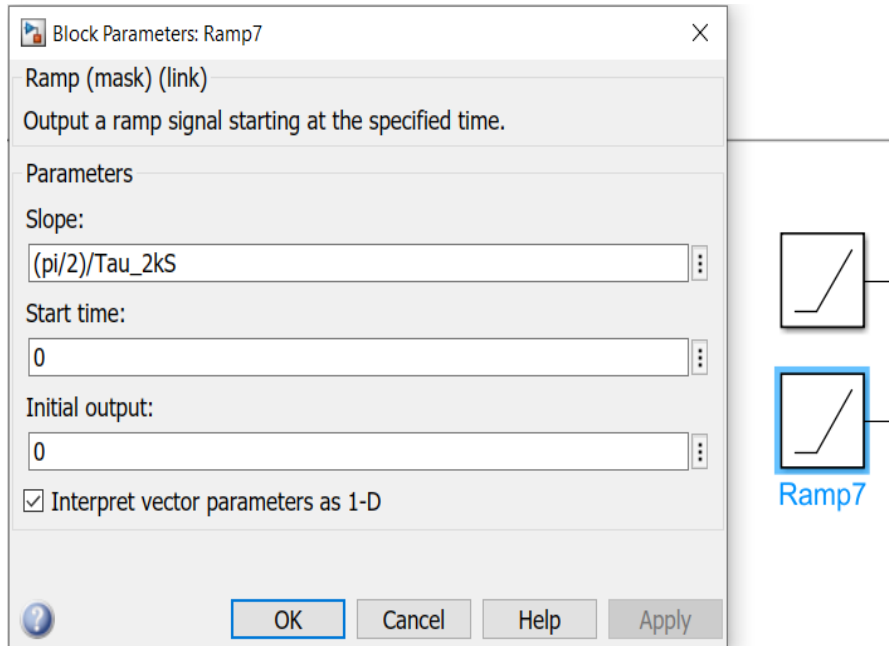
Obrázek 49 Ruka – krychle

### 5.1. Simulační model

Simulační model byl detailně rozebrán v kapitole 4. V rámci této podkapitoly pouze vytкну změny, které byly provedeny pro optimalizační model ruky.

#### 5.1.1. Buzení

V rámci optimalizace je kinematické buzení provedeno pomocí bloku ramp. Kde se nastaví úhel natočení, který se vydělí časem potřebným k vykonání natočení. Blok můžeme vidět na obrázku 51. Kde  $Tau_{2KS}$  na obrázku je jeden z optimalizačních parametrů o kterých bude hovořeno v kapitole 5.2.

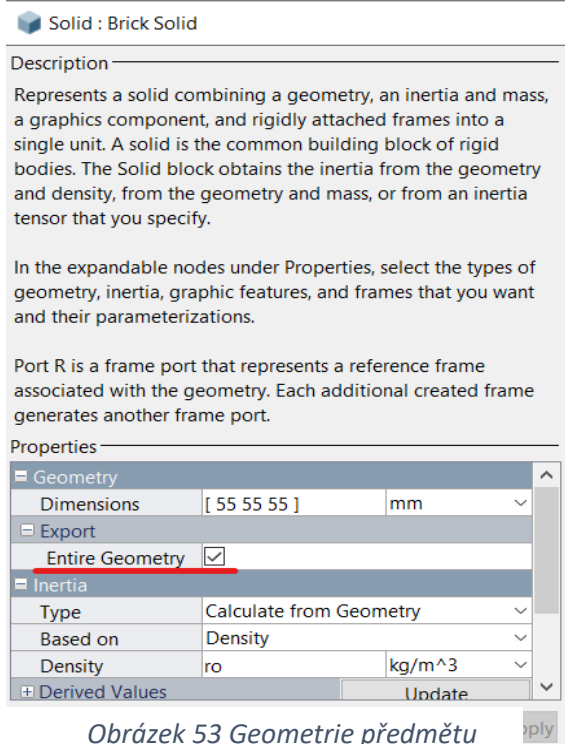


Obrázek 51 Ramp

### 5.1.2. Kontakt a vzdálenost

Aby se dala měřit vzdálenost mezi předměty a zároveň měřit síla vyvolaná kolizí dvou geometrií, je potřeba extrahovat geometrii obou předmětů (obrázek 52). Spojení obou geometrií umožní silový blok spatial contact force (obrázek 53), který funguje jako pružina s tuhostí a tlumič s tlumící konstantou. Geometrie jsou tedy spolu spojeny pomocí tlumiče s pružinou a vhodné volbě konstant potom odpovídají reakční účinky při kolizi

geometrií. V tomhle případě je síla stisku bezpředmětná, proto jsou voleny konstanty  $b = 1 \text{ Nsm}^{-1}$  a  $k = 1 \text{ Nm}^{-1}$ . Podstatné je, jestli kontakt nastal, nebo nenastal. Spojení geometrie uchopovaného předmětu a bříšek prstů vidíme na obrázku 38.



Obrázek 53 Geometrie předmětu

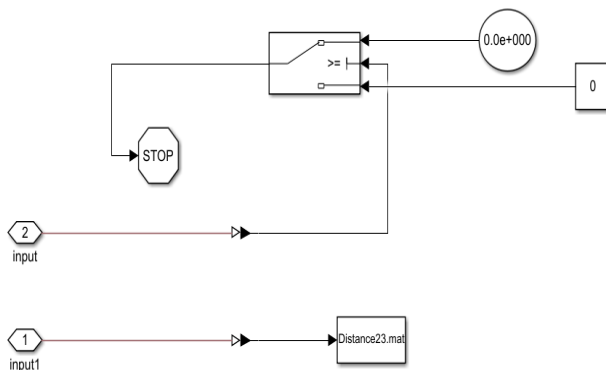


Spatial  
Contact Force

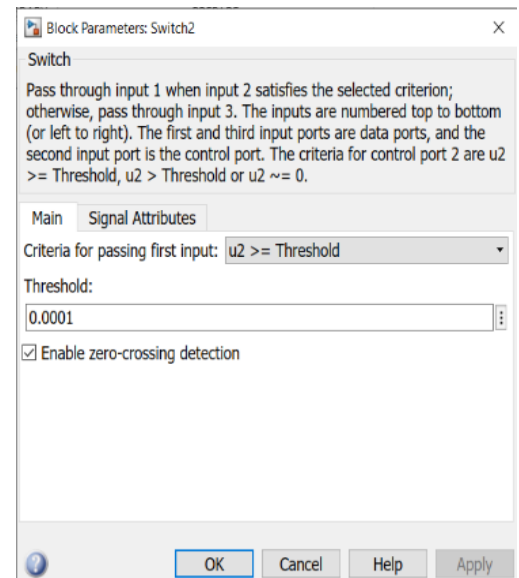
Obrázek 52 Spatial  
contact force

### 5.1.3. Určení kontaktu

Jestli nastal, nebo nenastal kontakt, rozhoduje subsystem vedle bloku Spatial Contact force. Z bloku je posíláno do subsystemu velikost normálové síly a vzdálenost mezi geometriemi. Vnitřek subsystemu vidíme na obrázku 54.



Obrázek 54 Sub systém – spatial contact force



Obrázek 55 if – switch

V subsystemu najdeme přepínač, který funguje jako podmínka *if*. Vlevo od přepínače je blok stop simulation, který zastaví simulaci, pokud je vstup do bloku nenulový. Do přepínače mohou vstoupit pouze dvě hodnoty – a to konstanta nula, nebo nenulová hodnota. Pro nenulovou hodnotu jsem zvolil blok Clock, který měří simulační čas. Simulace probíhá, dokud se pohony nenatočí o požadovaný úhel. Pokud dojde ke kolizi dvou geometrií, která vyvolá silovou reakci pošle se tato hodnota přímo do přepínače, který má v sobě podmínku. Pokud hodnota síly bude větší než 1e-4 Newtonů, přepínač přepne cesty z nulové konstanty na nenulovou a tím pádem se zastaví simulace. Hodnoty jsou během simulace až do jejího zastavení ukládány pomocí bloku To File, který se nachází ve spodní části na obrázku 54.

### 5.2. Optimalizační parametry

Jako jedny z optimalizačních parametrů jsem volil souřadnice uchopovaného předmětu. Souřadnice  $((x,y,z)\_předmět)$  reprezentují vzdálenost předmětu od souřadnicového systému dlaně. Dále byly jako optimalizační parametry voleny tuhosti torzních pružinek ( $K31, K32, K33, K34, K3\_thumb$ ) v posledním kloubu prstu, natočení palce vůči předmětu ( $Pfi, Pfiz$ ), délka prvního článku palce ( $P1\_length$ ) a v neposlední řadě jsem jako optimalizační parametry volil časy, za které se mají jednotlivé lanové bubny a vazby prvních článků natočit. Jelikož jsem optimalizoval každý předmět zvlášť, na jednu optimalizaci připadá 21 parametrů, které jsou zapsány v tabulce 5.

*Tabulka 5 Optimalizační parametry*

|           |      |           |       |         |       |
|-----------|------|-----------|-------|---------|-------|
| K3_thumb  | P(1) | x_předmět | P(9)  | Tau_1kS | P(17) |
| K34       | P(2) | y_předmět | P(10) | Tau_2kS | P(18) |
| K33       | P(3) | z_předmět | P(11) | Tau_3kS | P(19) |
| K32       | P(4) | Tau_1kP   | P(12) | Tau_4kS | P(20) |
| K31       | P(5) | Tau_2kP   | P(13) | Tau_pkS | P(21) |
| P1_length | P(6) | Tau_3kP   | P(14) |         |       |
| Pfi       | P(7) | Tau_4kP   | P(15) |         |       |
| Pfiz      | P(8) | Tau_pkP   | P(16) |         |       |

### 5.3. Průběh optimalizace

Vzhledem k tomu, že blok Spatial Contact Force v každém okamžiku měří vzdálenosti geometrií, při simulaci celého modelu ruky se v modelu nachází šestnáct těchto bloků. Takový počet bloků výrazně zpomalí simulaci. Pro plynulejší chod simulace bylo rozhodnuto, že bude pro daný vektor parametrů probíhat simulace každého prstu zvlášť.

#### 5.3.1. Okrajové hodnoty parametrů

Pro danou optimalizaci byly vytvořeny dva vektory  $lb$  a  $ub$  o velikosti  $1 \times P$ , kde  $P$  je počet optimalizačních parametrů (tabulka 5). Vektor  $lb$  značí spodní hodnoty intervalu a vektor  $ub$  značí horní hodnoty. Hodnoty příslušící jednotlivým prvkům vektorů jsou k nahlédnutí v souborech Bound\_V.m, Bound\_K.m, Bound\_Kr.m.

#### 5.3.2. Algoritmus

Protože nevíme, kde se nachází extrém optimalizační funkce, je proto volen genetický algoritmus, který je schopen prohledat množinu hodnot, která byla stanovena v 5.3.1.

##### Genetický algoritmus

Jedná se o heuristický přístup. Algoritmus je založený na principech přírodního výběru. Algoritmus náhodně vytvoří počáteční populaci složenou z jedinců. Jedinec je vektor parametrů, kterému přísluší hodnota fitness funkce. Podle velikosti fitness funkce se posuzuje jedincova zdatnost. Algoritmus pak náhodně z populace vybere sadu jedinců s vysokou zdatností. Po výběru jedinců nastává tvorba potomků, a to buď křížením, mutací, nebo ponecháním jedinců beze změny (reprodukční funkce). Z potomků posléze vznikne nová generace a proces se opakuje, dokud nenastane jedna z podmínek pro ukončení algoritmu.

## Křížení

Křížení je výměna genetického materiálu dvou jedinců s předem danou pravděpodobností. Používané druhy křížení jsou:

### 1. Aritmetické křížení

Kde  $r$  je náhodně volený váhový koeficient z intervalu  $(0, 1)$ .

$$\begin{aligned}x &= rx' + (1 - r)y' \\ y &= ry' + (1 - r)x'\end{aligned}$$

### 2. Prosté křížení

Dva vektory si mezi sebou s pravděpodobností  $r$  vymění své parametry.

Interval  $k \leq r \leq k + 1$  náleží  $m$  a  $r$  se nachází v intervalu  $(0, 1)$ .

$$\begin{aligned}x &= (x_1 \dots x_l \dots x_m) \\ x' &= (x_1 \dots x_l y_{l+1} \dots y_m)\end{aligned}$$

## Mutace

Náhodná změna parametrů jedince s předem danou pravděpodobností.

## Zvolené parametry pro algoritmus

Pro dané optimalizace byl zvolen maximální počet generací 40. Počet jedinců populace byl volen 200. A v případě, že by se hodnota fitness funkce málo měnila, byla nastavena tolerance  $1e-2$ . Zvolené optimalizační parametry algoritmu jsou vidět na obrázku 56.

```

options = gaoptimset('Display','iter',...
'PopulationSize',200,...
'Generations',40,...
'TolFun',1e-2,...
'TolCon',1e-2,...
'PlotFcns',@gaplotbestf);
[F,fval,exitflag,output, population] = ga(@Enter_Koule,21,[],[],[],[],lb,u

```

Obrázek 56 Genetický algoritmus



### 5.3.3. Popis optimalizace

Po spuštění skriptu s algoritmem započne načtení matice hraničních hodnot (kapitola 5.3.1) a vytvořením textového souboru pro ukládání hodnot viz. obrázek 57.

```
fileID = fopen('Values_KOULE.txt','w');
fprintf(fileID,'%4s %21s %11s %14s %12s %18s %12s %6s %14s %11s %12s');
fclose(fileID);
```

Obrázek 57 Vytvoření textového souboru

Dále se je spuštěn genetický algoritmus, který otevře příslušnou funkci, uloží algoritmem zvolené parametry do vektoru P a uloží dané parametry v Matlabu tak, aby si je simulace ze Simulinku mohla načíst. Dále se pustí funkce obsahující skripty, které spustí jednotlivé simulace prstů, které generují pro každý skript matice  $Distance_{ki}$  ( $k$  – číslo prstu,  $i$  – číslo článku) s koncovými vzdálenostmi od bříšek prstů k uchopovanému předmětu. V obrázku 59 nalezneme v matici změřených vzdáleností kladnou, nulovou, nebo zápornou vzdálenost. Na obrázku 59 vidíme počáteční kód, který slouží k vyhledání kladných, nulových, nebo záporných vzdáleností a k nim přiřadí příslušný simulační čas, při kterém nastaly. Tento čas je uložen do matice  $V_i$  ( $i$  = číslo prstu) na obrázku 58. Pokud ke kontaktu nedošlo, uloží se do vektoru  $V_i$  hodnota 5000, která slouží jako logická hodnota pro nedosažení kontaktu. Pro jeden prst tedy dostaneme vektor  $V_i$  s rozměrem 1 x (počet bříšek prstů). Tyto simulace se opakují pro každý prst zvlášť.

```
%% kontakt
for i = 1:size(Distancell,2)
    Touchl1 = Distancell(2,i);
    if Touchl1 <= 0;
        Rl1=Touchl1;

[Touchl1_pos]=find(Rl1 == Distancell(2,:));
T11 = Distancell(1,Touchl1_pos);
T11 = T11(1,1);
        break
    else
        T11 = 5000;
    end
end
```

$$V_1 = [T11;T12;T13];$$

Obrázek 58 Matice kandidátů pro prst 1 - malíček

Obrázek 59 kód – volba kandidáta

Po skončení všech simulací obdržíme tedy pět vektorů  $V_1..V_4$  a  $V_p$ . Dále (Obrázek 60) je kontrolováno, jestli matice neobsahuje pouze hodnoty 5000. Pokud ano, tak kontakt nenastal a funkce je penalizována hodnotou  $1e8$ . Jestli kontakt nastal, tak jsou hodnoty časů dále ukládány do matice  $Vo$ , která slouží pro uložení časů, kdy došlo ke kontaktu. Z matice  $Vo$  je posléze vzata minimální hodnota času a je označena jako  $Tmax$ .

```

Vin=[V_1; V_2; V_3; V_4; V_p];
Vo=[];
Compare2 = 0;
Compare1 = (1:size(Vin,1))
for i = 1:size(Vin,1)
    if Vin(i,1) ~= 5000
        Vo = [Vo;Vin(i,1)];

    else
        Compare2 =Compare2+1;
    end
end

if size(Compare1,2) == Compare2
    F_koule = 1e8;
else

Tmax = min(Vo)

```

Obrázek 60 kód – výběr nejmenšího času doteku

Jelikož je simulován každý prst zvlášť, tak při přiřazování času  $Tmax$  z obrázku 60 zjistíme, že se simulační čas  $Tmax$  v matici vzdáleností jiných simulací nenachází. Je to proto, že simulační časy dvou různých simulací nebudou nikdy naprosto stejné.

Proto v dalším kroku (Obrázek 61) je od každého časového kroku matic koncových doteků odečtena v absolutní hodně hodnota  $Tmax$ . Ke každému nejmenšímu časovému rozdílu přísluší hodnota vzdálenosti, která je vzata a převedena vynásobením  $1e3$  na mm. Takto selektované vzdálenosti jsou dále použity pro sestavení cílové funkce.

```

%% Dist1
DD11=[];
for i = 1:size(Distance11,2)
    D11(i) = abs(abs(Distance11(1,i))-Tmax);
    DD11= [DD11, D11(i)];
end
T1 = min(min(DD11));
[T1_pos] = find(T1 == DD11(1,:));
Dist11 = Distance11(2,T1_pos)*1e3;
if Dist11 < 0
    Dist11 =0;
end
Dist12 = Distance12(2,T1_pos)*1e3;
if Dist12 < 0
    Dist12 =0;
end
Dist13 = Distance13(2,T1_pos)*1e3;
if Dist13 < 0
    Dist13 =0;
end

```

Obrázek 61 kód – nalezení příslušného času a vzdálenosti

Název koncové vzdálenosti je shodný s názvem maticí vzdáleností *Distance\_ki* (k – číslo prstu, i – číslo článku), kde bylo zaměněno *Distance\_ki* za *Dist\_ki*. Sestavené cílové funkce je popsáno v kapitole 5.3.4.

Posledním krokem před stanovením cílové funkce bude pouze kontrola, jestli se palec nenacházel v kolizi s předmětem od začátku simulace, protože je optimalizováno počáteční natočení palce a poloha předmětu najednou. Může dojít k tomu, že se předmět s palcem v takové kolizi ocitnou. Pro ošetření tohoto problému bylo měřeno natočení článků prstů. Při spuštění simulace s kolizí se registruje kontakt ihned a velikosti matic natočení článků budou minimální. Proto při nesplnění podmínky (obrázek 62) velikosti matice natočení penalizují výslednou funkci hodnotou  $1e8$ .

```

%% Penalizace za kolizi
if size(Fi1,1) <= 80 || size(Fi2,1) <= 80 || size(Fi3,1) <= 80 || size(F:
    F_valec = 1e8;
end

```

Obrázek 62 Penalizace za kolizi

Jelikož hodnotu naší cílové funkce máme, funkce ji navrátí funkci na začátku a uloží veškeré hodnoty do textového souboru (obrázek 63).

```

A=[Fk;p(1);p(2);p(3);p(4);p(5);p(6);p(7);p(8);p(9);p(10);p(11);p(12);p(
fileID = fopen('Values_KOULE_V2.txt','a');
fprintf(fileID,'%16.4f %12.4f %12.4f %12.4f %12.4f %12.4f %12.4f %12.
fclose(fileID);

```

Obrázek 63 Ukládání hodnot z jednotlivých iterací

### 5.3.4. Cílové funkce

Cílovou funkci pro jeden prst sestavují jako součet jednotlivých cílových funkcí v kvadrátu. Cílová funkce je pak součtem těchto jednotlivých funkcí pro všechny články. Pro optimalizaci byly voleny váhové koeficienty (obrázek 64, 65, 66). Pro kouli byly použité váhové koeficienty tak, aby se co nejvíce minimalizovala vzdálenost pro články 2 a 1. Také byly váženy prsty celkově, a to sestupně od palce s prostředníčkem po malíček.

Pro předměty krychle a válec byly voleny stejné hodnoty váhových koeficientů, kde jejich použití bylo podobné jako u koule s rozdílem, že u jednotlivých článků byly koeficienty použity pouze pro články s číslem 3.

Na obrázku 66 je výsledná hodnota funkce nazvaná  $F\_valec$ , jedná se o cílovou funkci pro kužel, kde se zapomněl obměnit název, jelikož mělo jít původně o uchopení válce.

```

%% Cílová funkce s váhovými koeficienty

c1 = 10; c2 = 8; c3 = 5; c4 = 2;
D1 = ((Dist11*c1+Dist12*c4+Dist13).^2)*c1;
D2 = ((Dist21*c2+Dist22*c4+Dist23).^2)*c2;
D3 = ((Dist31*c3+Dist32*c4+Dist33).^2)*c3;
D4 = ((Dist41*c4+Dist42*c4+Dist43).^2)*c4;
Dp = ((Dist1p*c4+Dist2p).^2)*c4;

F_koule = D1+D2+D3+D4+Dp

```

Obrázek 64 Cílová funkce koule

```

%% Cílová funkce s váhovými koeficienty

c1 = 10; c2 = 8; c3 = 5; c4 = 2;
D1 = ((Dist11+Dist12+Dist13*c4).^2)*c4;
D2 = ((Dist21+Dist22+Dist23*c3).^2)*c3;
D3 = ((Dist31+Dist32+Dist33*c2).^2)*c2;
D4 = ((Dist41+Dist42+Dist43*c1).^2)*c1;
Dp = ((Dist1p+Dist2p*c1).^2)*c1;

F_krychle = D1+D2+D3+D4+Dp

```

Obrázek 65 Cílová funkce krychle

```
%% Cílová funkce s váhovými koeficienty

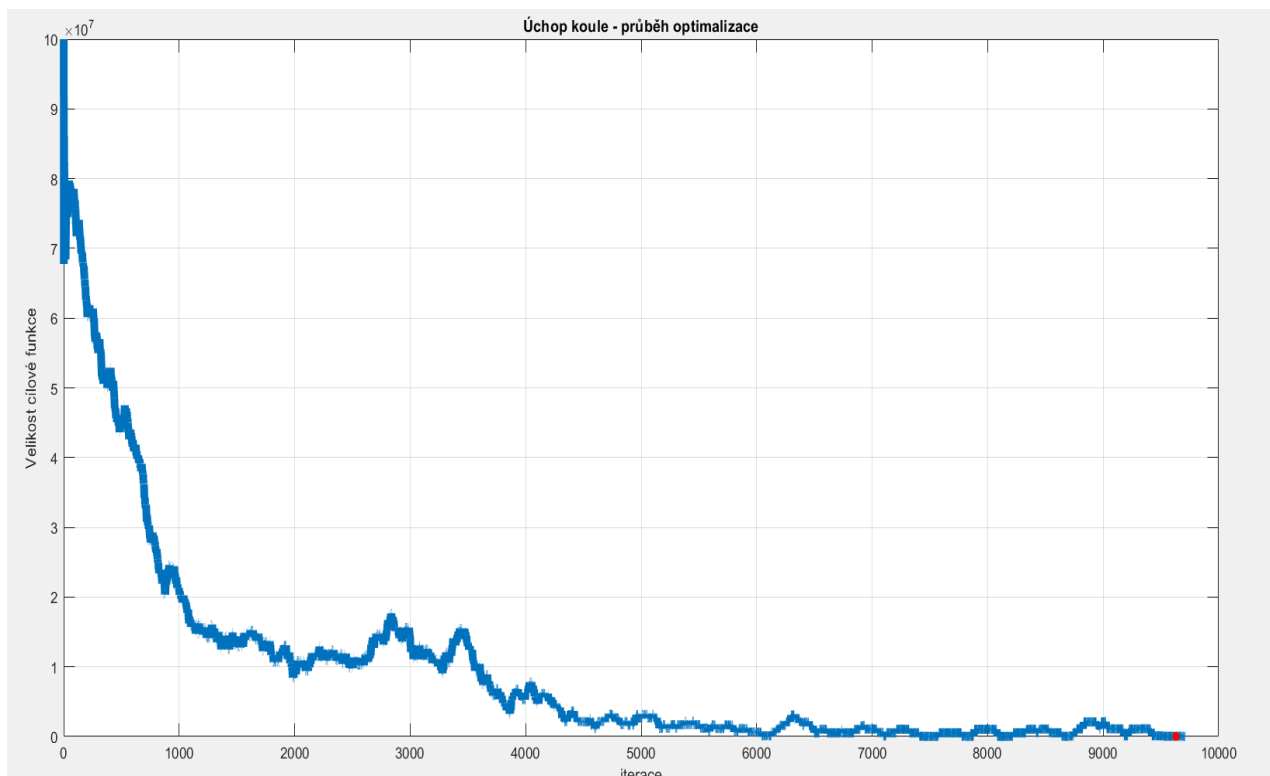
c1 = 10; c2 = 8; c3 = 5; c4 = 2;
D1 = ((Dist11+Dist12+Dist13*c1).^2)*c4;
D2 = ((Dist21+Dist22+Dist23*c1).^2)*c3;
D3 = ((Dist31+Dist32+Dist33*c1).^2)*c2;
D4 = ((Dist41+Dist42+Dist43*c1).^2)*c1;
Dp = ((Dist1p+Dist2p*c1).^2)*c1;
F_valec = D1+D2+D3+D4+Dp
```

Obrázek 66 Cílová funkce kužel

#### 5.4. Výsledky optimalizace pro úchop koule

Na obrázku 67 vidíme průběh optimalizace, kde na x ose se nachází počet iterací a na y ose k nim příslušné hodnoty cílové funkce. Z hlediska časové náročnosti si simulace vyžádala cca. 24 hodin čistého času, všechny tři simulace byly podobně časově náročné.

Simulace skončila dosažením maximálního počtu generací 40 a nejnižší hodnotu cílové funkce najdeme v obrázku 67 jako červenou tečku s iterací číslem 9630 s hodnotou cílové funkce  $F_{koule} = 15\,725\text{ mm}^2$ . Na obrázku 68 můžeme vidět obrázek ze simulace po uchopení předmětu. Průměr koule byl  $D = 70\text{ [mm]}$  což cca. odpovídá tenisovému míčku.

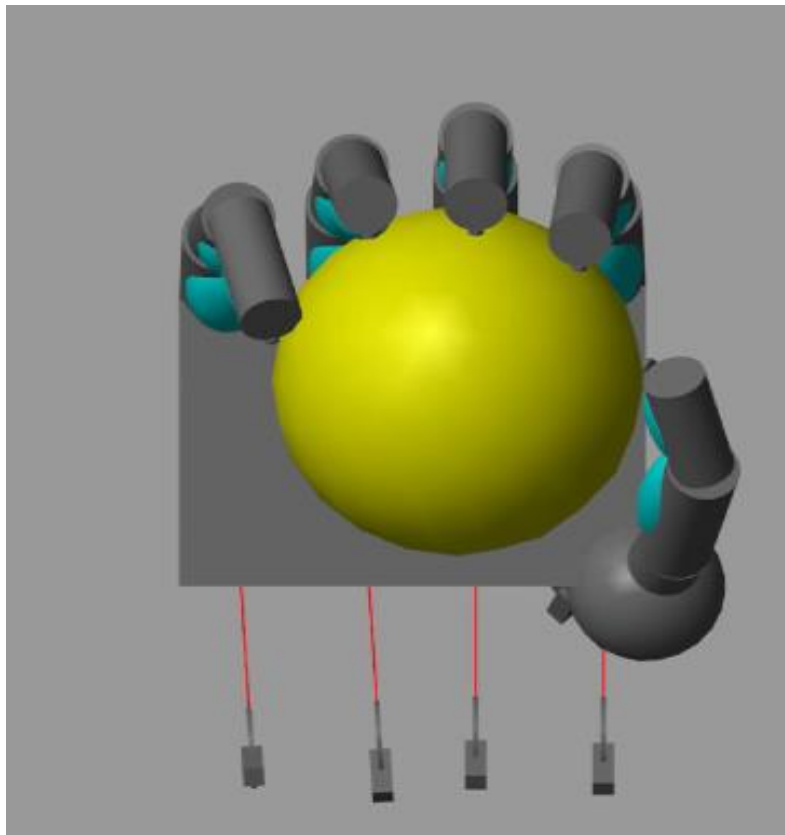


Obrázek 67 Průběh optimalizace úchopu koule

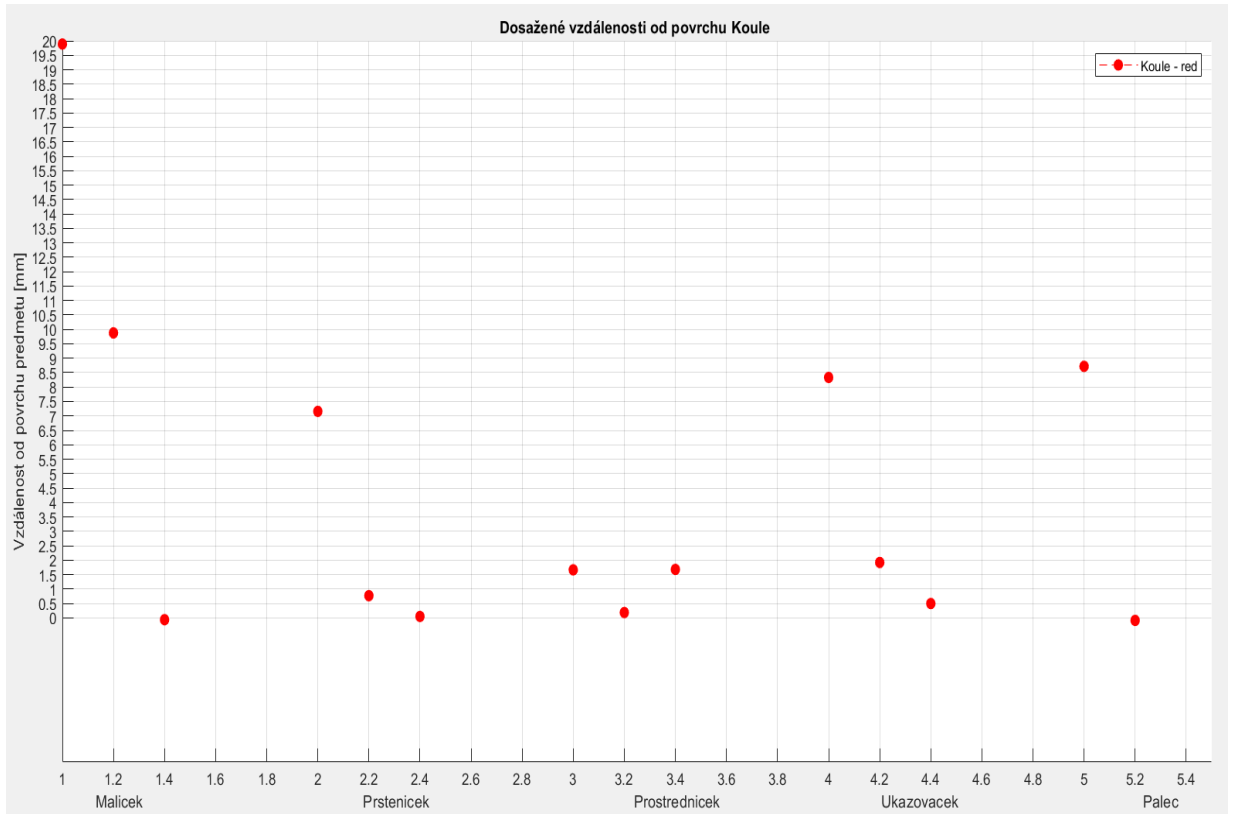
Na obrázku 69 jsou vyobrazeny vzdálenosti jednotlivých bříšek prstů od geometrie míčku. Číslování bříšek prstů jde zleva do prava (1–2–3). Vzhledem k použitým váhám, jde vidět, že články malíčku se nejvíce vzdalují od geometrie. Výsledné hodnoty optimalizace jsou k nahlédnutí v tabulce 6.

Tabulka 6 Optimalizované parametry pro úchop koule

|           |                              |           |                     |         |        |
|-----------|------------------------------|-----------|---------------------|---------|--------|
| K3_thumb  | $0.4499 \text{ Nm rad}^{-1}$ | x_předmět | $-15.34 \text{ mm}$ | Tau_1kS | 1.15 s |
| K34       | $0.8794 \text{ Nm rad}^{-1}$ | y_předmět | $-59.75 \text{ mm}$ | Tau_2kS | 1.29 s |
| K33       | $0.3005 \text{ Nm rad}^{-1}$ | z_předmět | $11.10 \text{ mm}$  | Tau_3kS | 1.31 s |
| K32       | $0.8253 \text{ Nm rad}^{-1}$ | Tau_1kP   | 1.84 s              | Tau_4kS | 1.18 s |
| K31       | $0.3410 \text{ Nm rad}^{-1}$ | Tau_2kP   | 3.92 s              | Tau_pkS | 3.55 s |
| P1_length | $21.93 \text{ mm}$           | Tau_3kP   | 2.59 s              |         |        |
| Pfi       | $53.56^\circ$                | Tau_4kP   | 3.21 s              |         |        |
| Pfiz      | $58.18^\circ$                | Tau_pkP   | 2.02 s              |         |        |



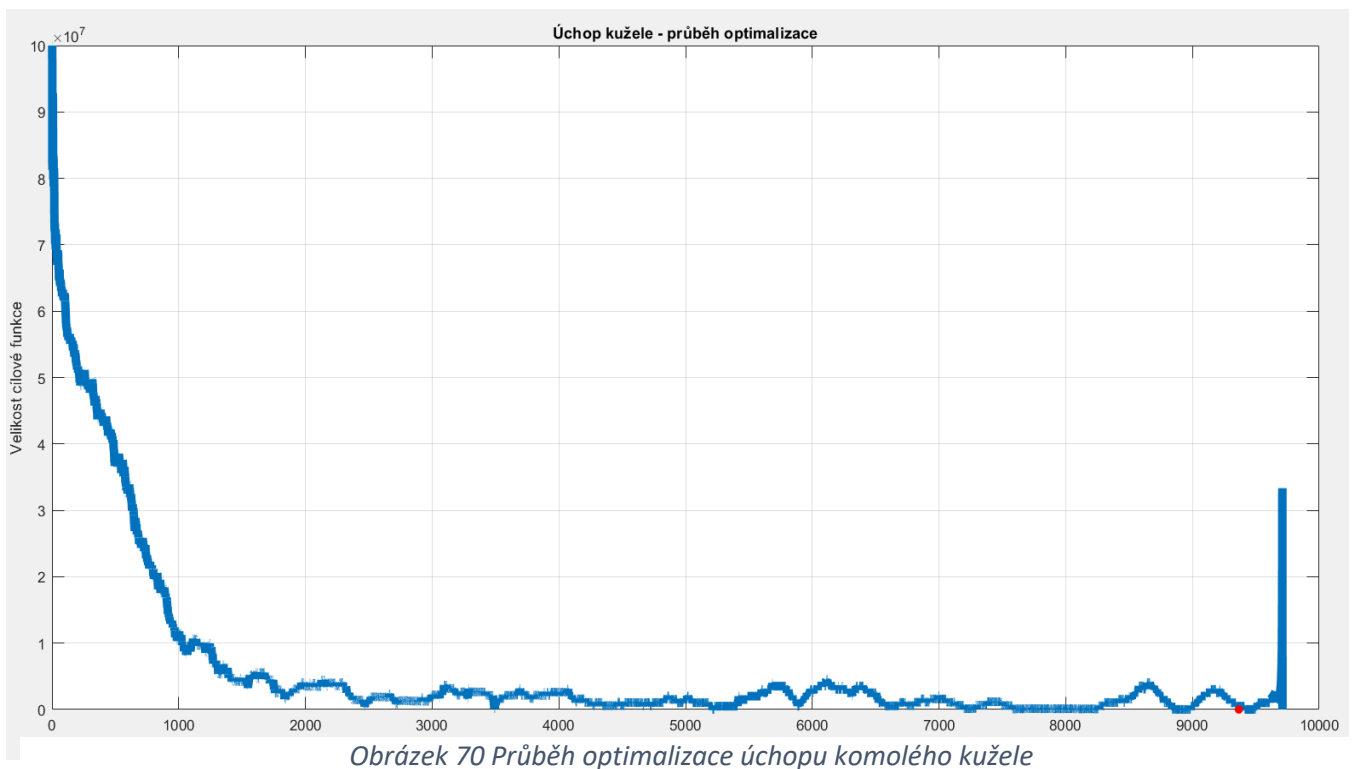
Obrázek 68 Úchop koule



Obrázek 69 Vzdálenosti bříšek prstů od geometrie míčku

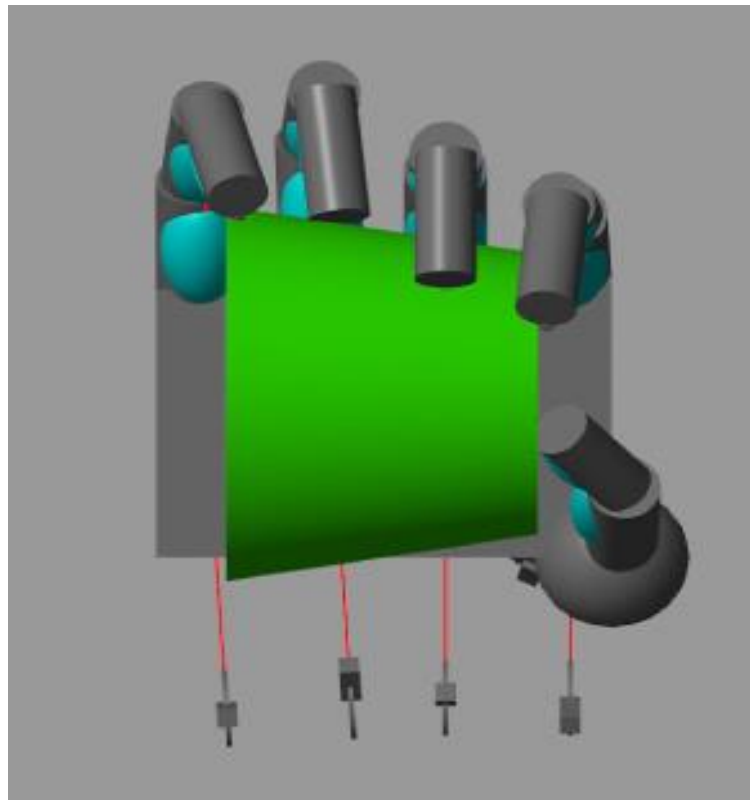
### 5.5. Výsledky optimalizace pro úchop kužele

Na obrázku 70 vidíme stejný průběh jako na obrázku 67, ale tentokrát se optimalizace týkala úchopu komolého kužele, který měl délku  $L = 60[mm]$  a kuželovitost 1:3.



Obrázek 70 Průběh optimalizace úchopu komolého kužele

Optimalizace skončila dosažením maximálního počtu generací a minimum našla na iteraci číslo 9366 a hodnotou cílové funkce  $F_{kužel} = 2\,521 [mm^2]$ . Minimum je vyznačeno na obrázku červenou tečkou. Váhové koeficienty na prsty byly zvoleny stejně jako výše u úchopu koule, ale váženy zde byly pouze první články prstů. Dosažené vzdálenosti od povrchu kužele jsou vyobrazeny na obrázku 72 a výsledné optimalizované parametry v tabulce 7.

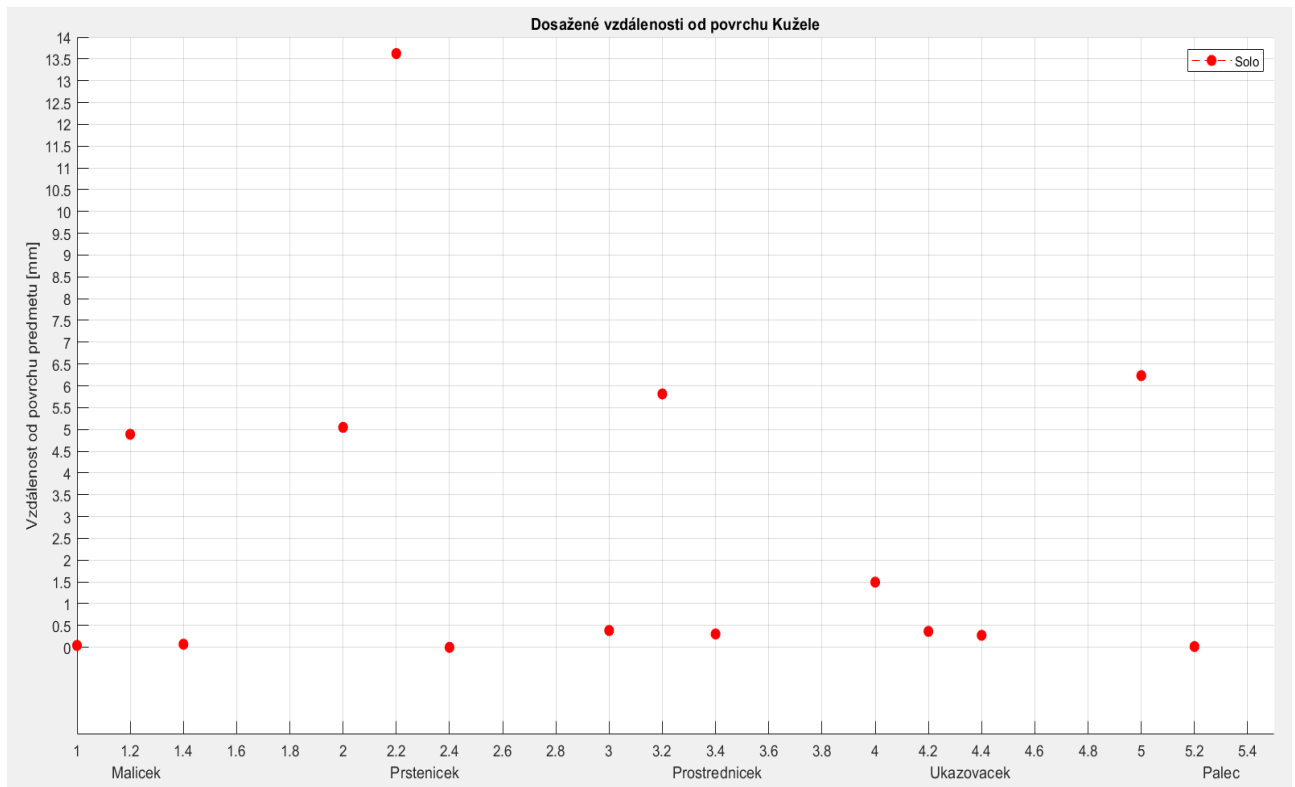


Obrázek 71 Úchop kužel

Tabulka 7 Optimalizované parametry pro úchop komolého kužele

|           |                              |           |                     |         |        |
|-----------|------------------------------|-----------|---------------------|---------|--------|
| K3_thumb  | $0.6301 \text{ Nm rad}^{-1}$ | x_předmět | $-24.51 \text{ mm}$ | Tau_1kS | 2.67 s |
| K34       | $0.6368 \text{ Nm rad}^{-1}$ | y_předmět | $-47.13 \text{ mm}$ | Tau_2kS | 4.53 s |
| K33       | $0.9649 \text{ Nm rad}^{-1}$ | z_předmět | $4.90 \text{ mm}$   | Tau_3kS | 1.51 s |
| K32       | $0.9781 \text{ Nm rad}^{-1}$ | Tau_1kP   | 1.66 s              | Tau_4kS | 1.23 s |
| K31       | $0.4713 \text{ Nm rad}^{-1}$ | Tau_2kP   | 1.06 s              | Tau_pkS | 3.23 s |
| P1_length | $13.86 \text{ mm}$           | Tau_3kP   | 1.41 s              |         |        |
| Pfi       | $31.01^\circ$                | Tau_4kP   | 2.13 s              |         |        |
| Pfiz      | $54.48^\circ$                | Tau_pkP   | 1.97 s              |         |        |

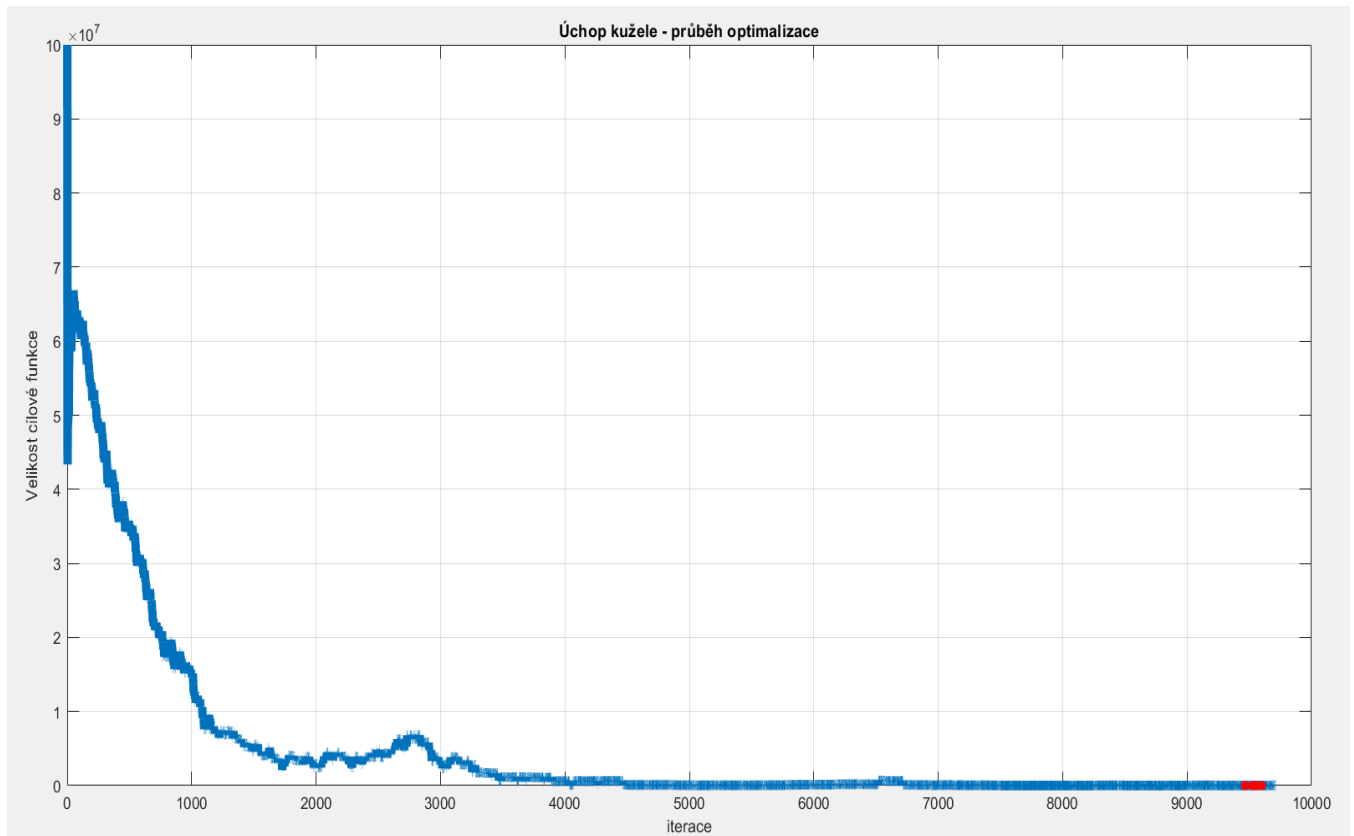




Obrázek 72 Vzdálenosti bříšek prstů od geometrie kužele

## 5.6. Výsledky optimalizace pro úchop krychle

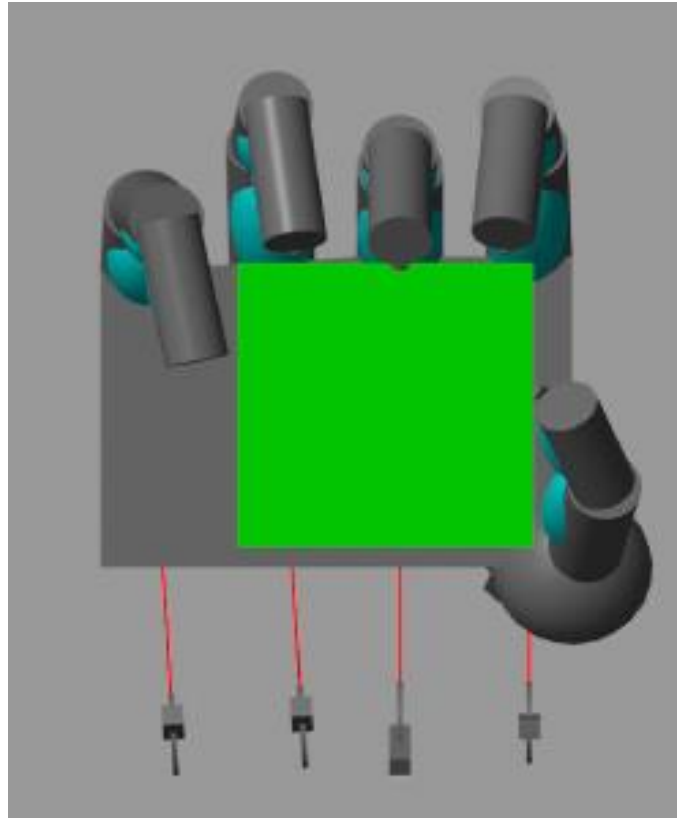
Na obrázku 73 vidíme stejný průběh jako na obrázku 67, ale tentokrát se optimalizace týkala úchopu krychle, která zhruba odpovídá velikosti Rubikovy kostky 3x3x3 s délkou strany rovnou  $a = 55 \text{ mm}$ . Globální minimum nabylo hodnoty  $F_{\text{krychle}} = 8311.8 \text{ [mm}^2\text{]}$ . Tato hodnota bylo nabyto hned čtyřikrát, jak můžeme vidět na obrázku 73, kde tři červené tečky vyobrazující toto minimum, a to v iteracích 9466, 9523, 9558 a 9601. Váhové koeficienty při simulaci krychle byly zvoleny stejně jako pro komolý kužel. Výsledky vzdáleností bříšek prstů od geometrie krychle vidíme na obrázku 75. Úchop krychle simulačním modelem můžeme vidět na obrázku 74 a výsledné parametry optimalizace můžeme vidět v tabulce 8.



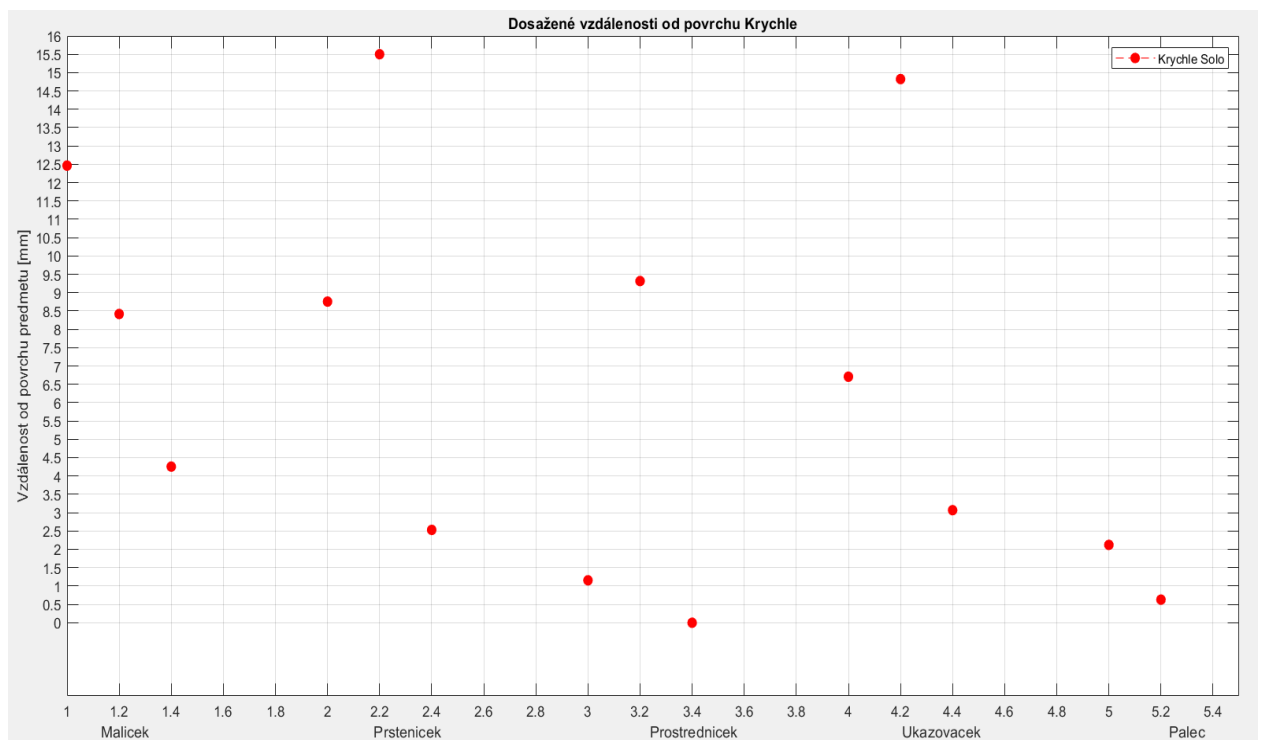
Obrázek 73 Průběh optimalizace úchopu krychle

Tabulka 8 Optimalizované parametry pro úchop krychle

|           |                              |           |                     |         |                   |
|-----------|------------------------------|-----------|---------------------|---------|-------------------|
| K3_thumb  | $0.1613 \text{ Nm rad}^{-1}$ | x_předmět | $-14.65 \text{ mm}$ | Tau_1kS | $1.072 \text{ s}$ |
| K34       | $0.8453 \text{ Nm rad}^{-1}$ | y_předmět | $-58.55 \text{ mm}$ | Tau_2kS | $1.606 \text{ s}$ |
| K33       | $0.8388 \text{ Nm rad}^{-1}$ | z_předmět | $1.37 \text{ mm}$   | Tau_3kS | $1.051 \text{ s}$ |
| K32       | $0.5592 \text{ Nm rad}^{-1}$ | Tau_1kP   | $1.030 \text{ s}$   | Tau_4kS | $1.348 \text{ s}$ |
| K31       | $0.9323 \text{ Nm rad}^{-1}$ | Tau_2kP   | $1.024 \text{ s}$   | Tau_pkS | $4.244 \text{ s}$ |
| P1_length | $11.82 \text{ mm}$           | Tau_3kP   | $3.278 \text{ s}$   |         |                   |
| Pfi       | $45.76^\circ$                | Tau_4kP   | $1.311 \text{ s}$   |         |                   |
| Pfiz      | $61.77^\circ$                | Tau_pkP   | $2.115 \text{ s}$   |         |                   |



Obrázek 74 Úchop krychle



Obrázek 75 Vzdálenosti bříšek prstů od geometrie krychle

## 5.7. Zhodnocení optimalizace

Byla provedena optimalizace pro tři různé předměty. Vzhledem k tomu, aby ruka uchopila daný předmět všemi bříšky článků současně, musela by se danému předmětu přizpůsobit geometrie prstů jako jejich délka či velikosti jejich bříšek. Proto je výsledkem minimum funkce, které lze s daným modelem a jeho rozměry dosáhnout.

Jelikož optimalizace neuvažuje sílu stisku, nelze výsledky této optimalizace použít pro manipulaci s těžkými předměty.

Videa ze simulace úchopu použité pro prezentaci na předmětu jsou dostupná v příloze.

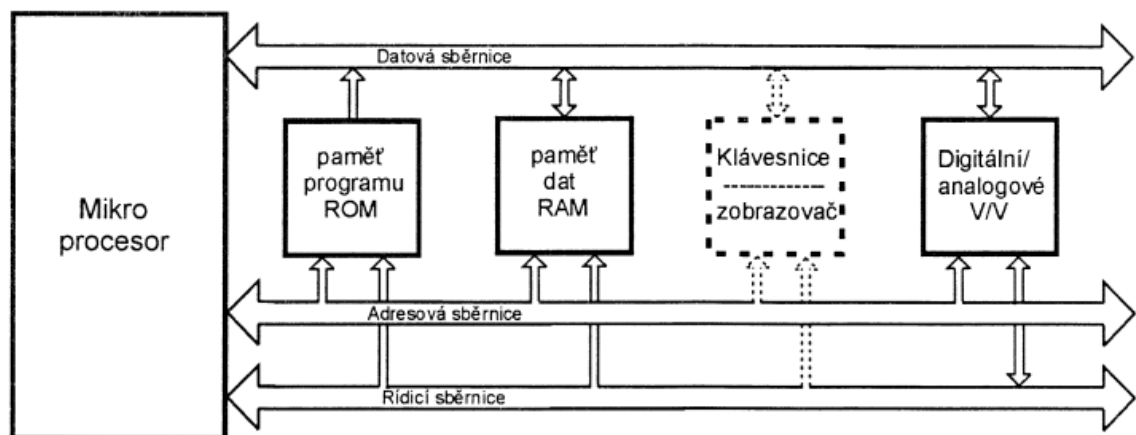
## 6. Ovládání simulačního modelu pomocí externího HW

K ovládání simulačního modelu bude uskutečněnou skrze platformu Arduino, kde po sériové lince budou posílány data ze senzorů do simulace v Simulinku – Simscape.

### 6.1. Mikropočítače a senzory

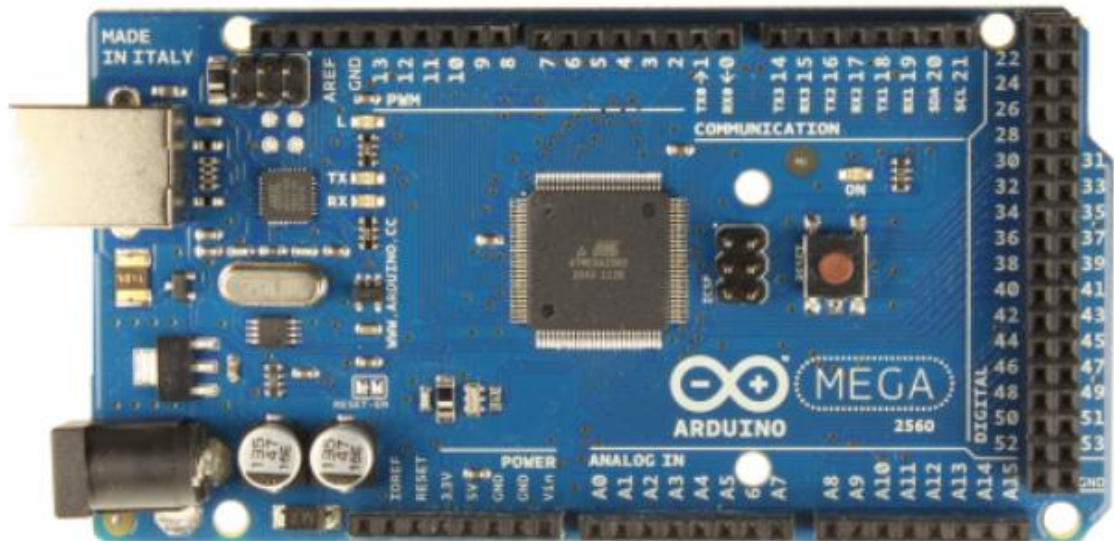
#### Mikropočítač

Jedná se o uspořádání logických obvodů, které vykonávají posloupnost povelů, které se načítají z paměti. Mikropočítač musí obsahovat procesor, paměť a vstupní/výstupní zařízení. Jednotlivé komponenty mikropočítače spolu komunikují po sběrnici. Blokové schéma vidíme na obrázku 76.



Obrázek 76 Blokové schéma mikropočítače [12]

Pro tuto diplomovou práci byla zvolena deska Arduino typu Mega 2560. Tato deska byla zvolena z důvodu dostatečného počtu analogových I/O pinů.



Obrázek 77 Arduino MEGA 2560 [13]

Parametry desky jsou uvedeny níže v tabulce 9.

Tabulka 9 Parametry desky Arduino MEGA 2560

|                          |                                   |
|--------------------------|-----------------------------------|
| Mikroprocesor            | ATmega2560                        |
| Provozní napětí          | 5V                                |
| Počet digitální I/O pinů | 54 z toho 15 je podporovaných PWM |
| Počet analogových pinů   | 16                                |
| Flash paměť              | 256 KB                            |
| Rychlost hodin           | 16 MHz                            |

### Senzory snímání úhlu natočení prstů

Pro snímání úhlu natočení jsem zvolil tzv. flex sensor (obrázek 78). V základě se jedná o proměnný rezistor, jehož odpor závisí na míře ohnutí senzoru. Protože je změna odporu proporcionální k velikosti ohnutí, lze odečítat změna napětí a lze tak stanovit míru jeho natočení.

- Konstrukce

Konstrukčně se konkrétně tenhle senzor skládá z fenolové pryskyřice, do které je vložen vodivý inkoust, který mění své elektrické vlastnosti při ohybu. Pro snímání těchto vlastností je po segmentech na inkoustu položený vodič.



Obrázek 78 Flex sensor [14]

- Vlastnosti

Odpor v klidu  $25\text{ k}\Omega$

Odpor v zatíženém stavu (60 až  $110\text{ k}\Omega$ )

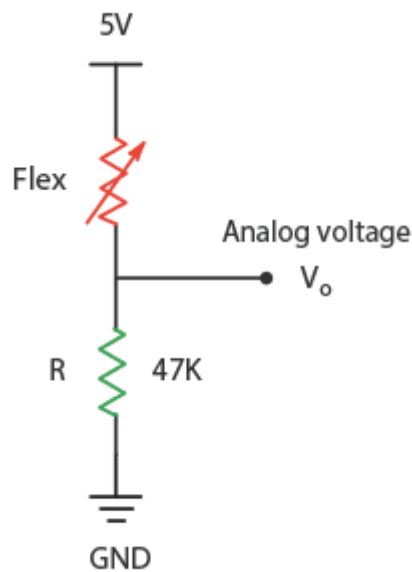
Velikosti senzoru ( $73,66 \times 6,35 \times 0,5\text{ mm}$ )

- Čtení senzoru

Pro přečtení změny napětí se připojí jeden vývod k pozitivnímu pólu  $5\text{ V}$  a k druhému vývodu sériově připojíme rezistor s neproměnným odporem. Z uzlu spojení mezi senzorem a odporem můžeme odečítat úbytek napětí na rezistoru. (Obrázek79).

Změnu napětí můžeme popsat rovnicí níže. Kde  $V_{cc}$  značí velikost napájecího napětí,  $R_{pulldown}$  značí velikost rezistoru, který je připojí k zemi a  $R_{flex}$  je měnící se odpor pásku způsobený jeho deformací. Výsledná změna napětí  $V_O$ , je pak odečtena mikrokontrolerem na analogovém vstupu.

$$V_O = V_{cc} * \left( \frac{R_{pulldown}}{R_{pulldown} + R_{flex}} \right)$$



Obrázek 79 Napěťový dělič [26]

Těchto senzorů jsem pořídil 5 a byly použity pro snímání natočení prstů, jako je na obrázku 20. Prvoplánově svírání prstů bylo myšleno jako kombinace pohybů z obrázků 19 a 20 a tudíž by byl pouze jeden aktuátor pro sklápění všech tří článků najednou. Během vypracovávání práce byl plán změněn a byl přidán aktuátor, aby se pohyby z obrázků 19 a 20 mohly konat nezávisle na sobě. Aby to bylo možné, bude zapotřebí snímat natočení spodních článků zvlášť, a to přidáním dalších pěti senzorů. Protože flex sensory (obrázek 78) jsou poněkud nákladné, rozhodl jsem se místo koupě dalších pěti senzorů udělat pokus a tyto senzory sám vyrobit.

### Výroba vlastních sensorů ohybu

Aby sensor měnil své elektrické vlastnosti, bude zapotřebí ho opatřit vodivým materiálem, který mění své elektrické vlastnosti při vnějším zatížení. Zvolil jsem levný materiál zvaný Velostat. Jedná se o materiál z polymerní folie, který je obalený uhlíkovými sazemi, které dělají folii elektricky vodivou. Černý proužek velostat folie můžeme vidět na obrázku 80 B).

Velostat folie byla nastříhaná na proužky, na které bylo potřeba z každé strany přilepit vodící prvek. Jako vodič jsem prvně zvolil dva měděné drátky, které vedly k neúspěchu. Po nahrazení drátků hliníkovou folií, která více přilne k povrchu, sensor začal konečně odpovídat na deformaci.

Nejprve byl na každý pásek napájen drátek pomocí cínové pájky (obrázek 80 A)). Následně byl na každý kus folie (obrázek 80 A) přilepena průhledná lepící páska. Na jeden, takto nalepený kus byl přidán pásek z velostatu, tak aby daný foliový proužek nepřechýlával. Druhý kus folie z páskou byl přilepen stejně z druhé strany (obrázek 80 C).



Obrázek 80 DIY sensor

### Změřené vlastnosti

Z charakteristik kupovaných senzorů vyplývá nárůst odporu se zatížením. U materiálu velostat odpor s ohnutím klesá. Pro představu jsem pomocí multimetru naměřil pokles z  $500\Omega$  na  $50\Omega$  při ohnutí. Díky lidskému faktoru se jednotlivé sensory od sebe trochu liší a uvedený pokles odporu je jen pro představu.

### Testování senzorů

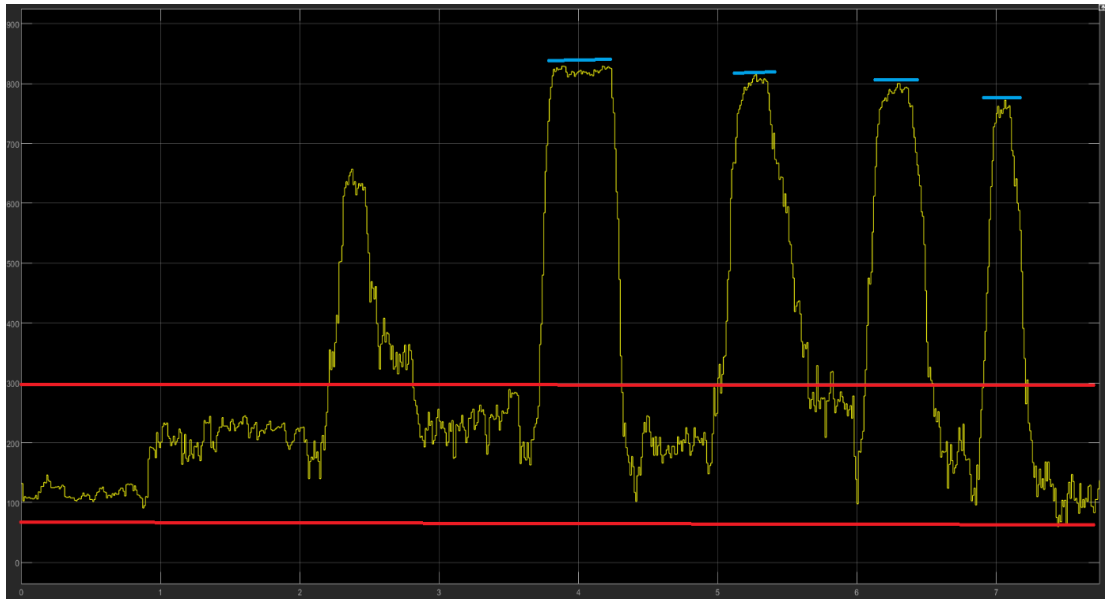
Na obrázku 81 vidíme záznam dat z jednoho vyrobeného senzorů. Kde bylo červeně ohraničeno pásmo, kdy byl sensor v klidu a modře byly zvýrazněny vrcholy, kde byl sensor ohnut téměř o  $90^\circ$ .

Na obrázku 82 vidíme k porovnání druhý vyrobený sensor. Červený proužek značí stav bez ohnutí a modrý ohnutí o  $90^\circ$ . Jde vidět, že některé hodnoty při ohnutí jsou srovnatelné s hodnotami naměřenými v klidu.

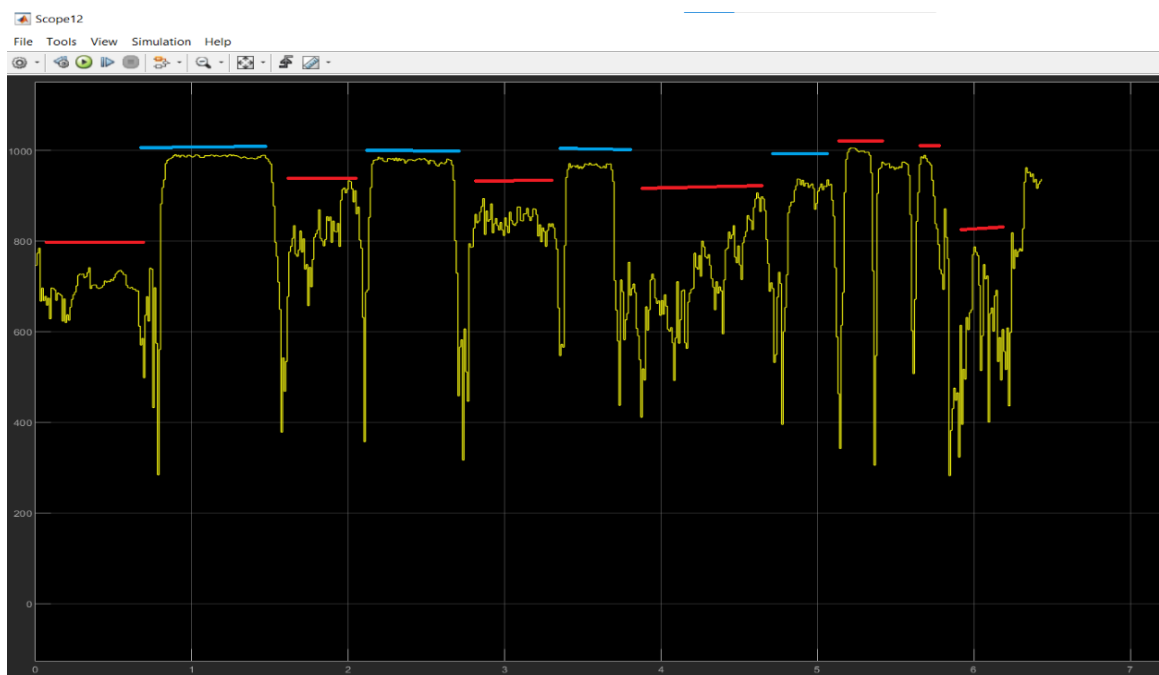
K porovnání na obrázku 83 vidíme průběh senzoru kupovaného, kde nenastává žádný markantní šum.

Senzory byly během testování nalepeny na rukavici pro relevantnost výsledků. Z obrázků 81 a 82 jde vidět, že i třes ruky dokáže rozhodit přijímaná data, nebo sklápění vedlejšího prstu.

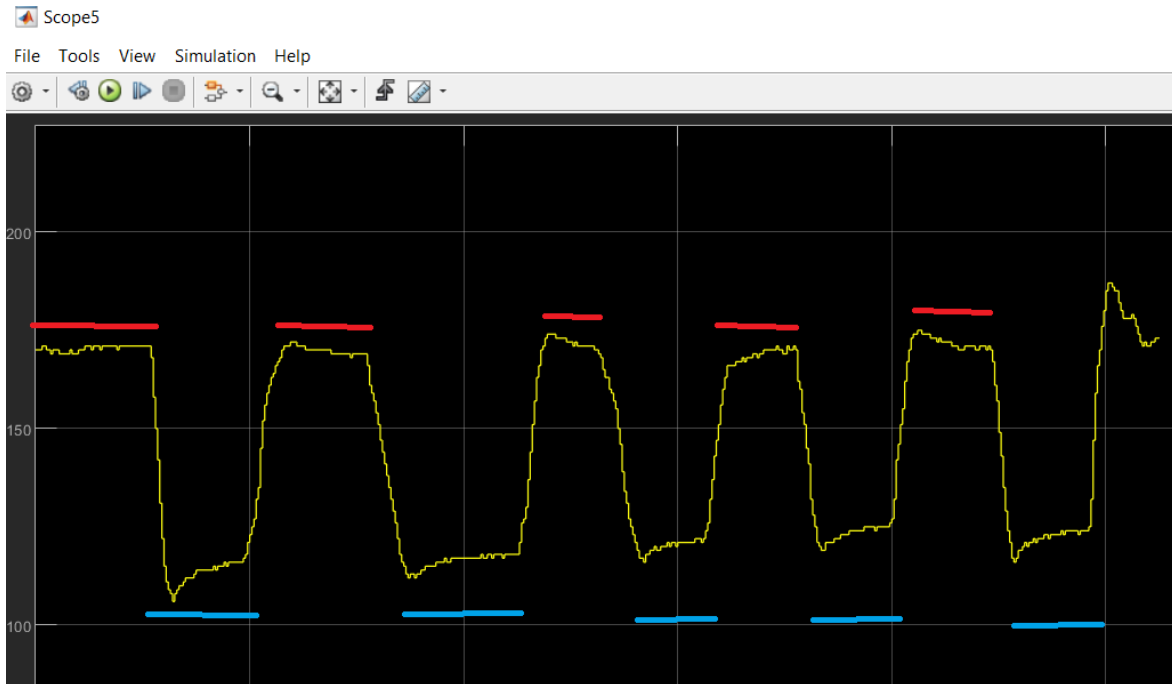




Obrázek 81 Data z vyrobeného senzoru 1



Obrázek 82 Data z vyrobeného senzoru 2



Obrázek 83 Data z koupeného senzoru

### Závěr kapitoly

V této kapitole byly rozebrány použité sensory k určení míry natočení. Z výsledků vyplývá, že pro přesnější snímání natočení by bylo výhodnější použití pouze sensorů kupovaných. Na druhou stranu výroba vlastních sensorů byla dobrá zkušenost a mohu říci, že se nejedná o lehkou úlohu.

### 6.2. Řídící rukavice

Pro vytvoření rukavice jsem zvolil klasickou zahradní rukavici (Obrázek 84). Na rukavici jsem posléze tavným lepidlem přilepil všechny výše zmíněné sensory a podstavec s pájitelným polem (Obrázek 87).



Obrázek 84 Rukavice [15]

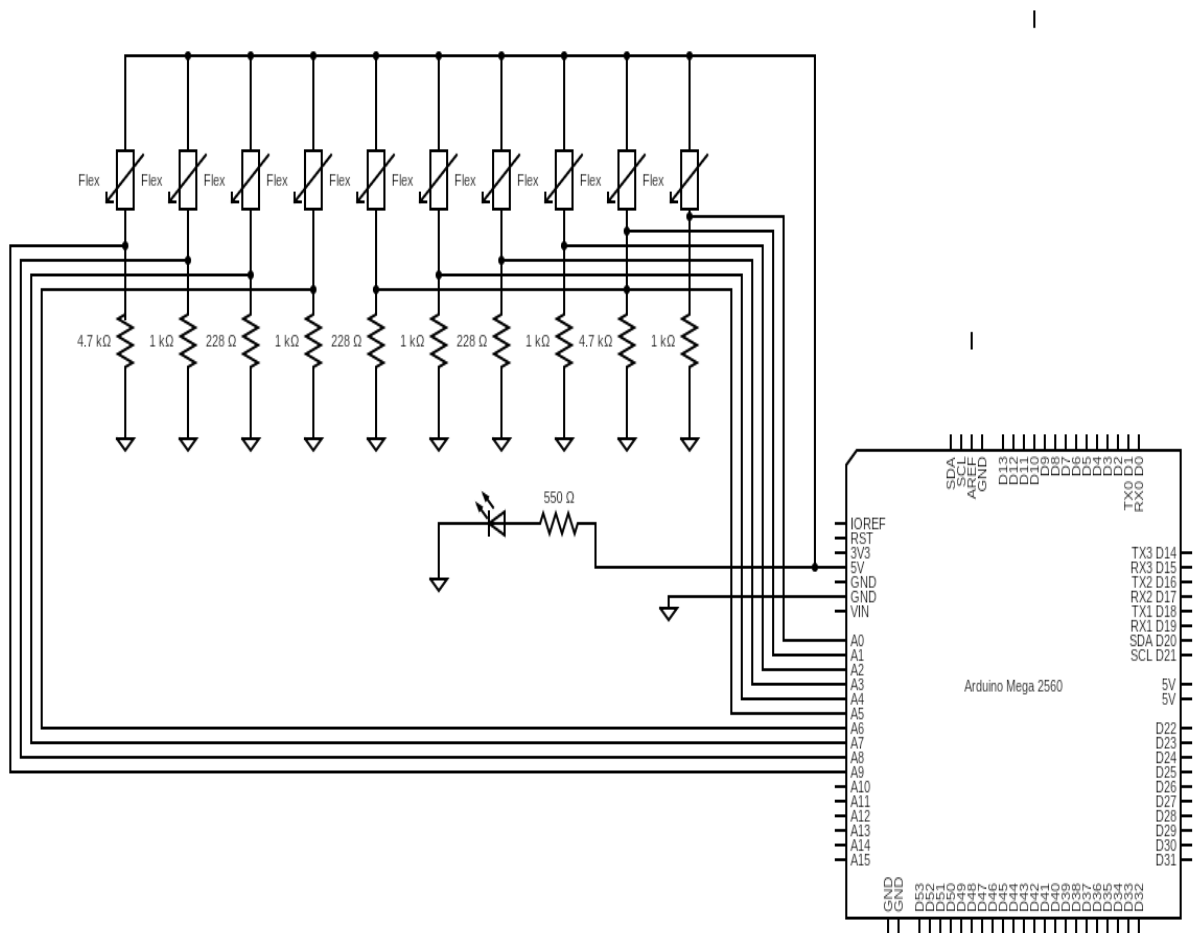
### Součástí rukavice

K rukavici byla dále přidána platforma pro zapojení senzorů. Platforma je složena z pájitelného pole, kde byly připájeny potřebné rezistory a piny k vytvoření napěťového děliče (Obrázek 79). Celé schéma můžeme vidět na obrázku 86 a desku s připájenými piny a rezistory na obrázku 85, kde byla posléze přidána LED k signalizaci stavu zapnuto/vypnuto. Dále byl pro desku vytisknut podstavec (Obrázek 87) na který byla připevněna deska. Podstavec byl posléze spolu s deskou pomocí lepidla připevněn na řídicí rukavici, kterou vidíme na obrázku 88.

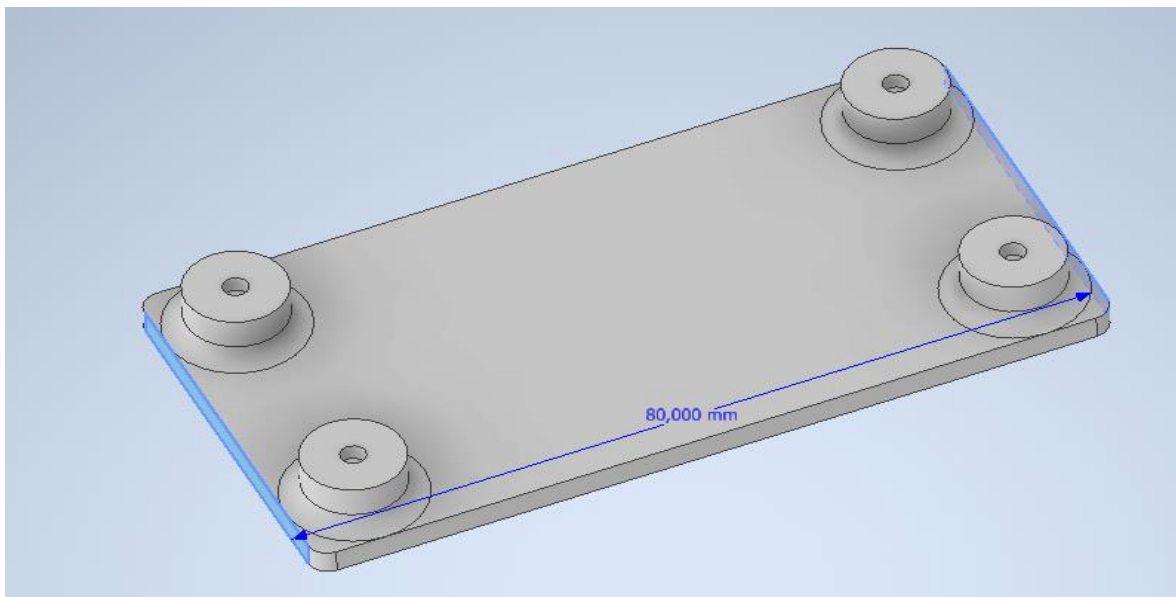
K vytvoření schématu na obrázku 86 byl využit webový editor circuit-diagram, který je dostupný z <https://www.circuit-diagram.org/>.



Obrázek 85 Pájecí pole



Obrázek 86 Schéma zapojení sensorů



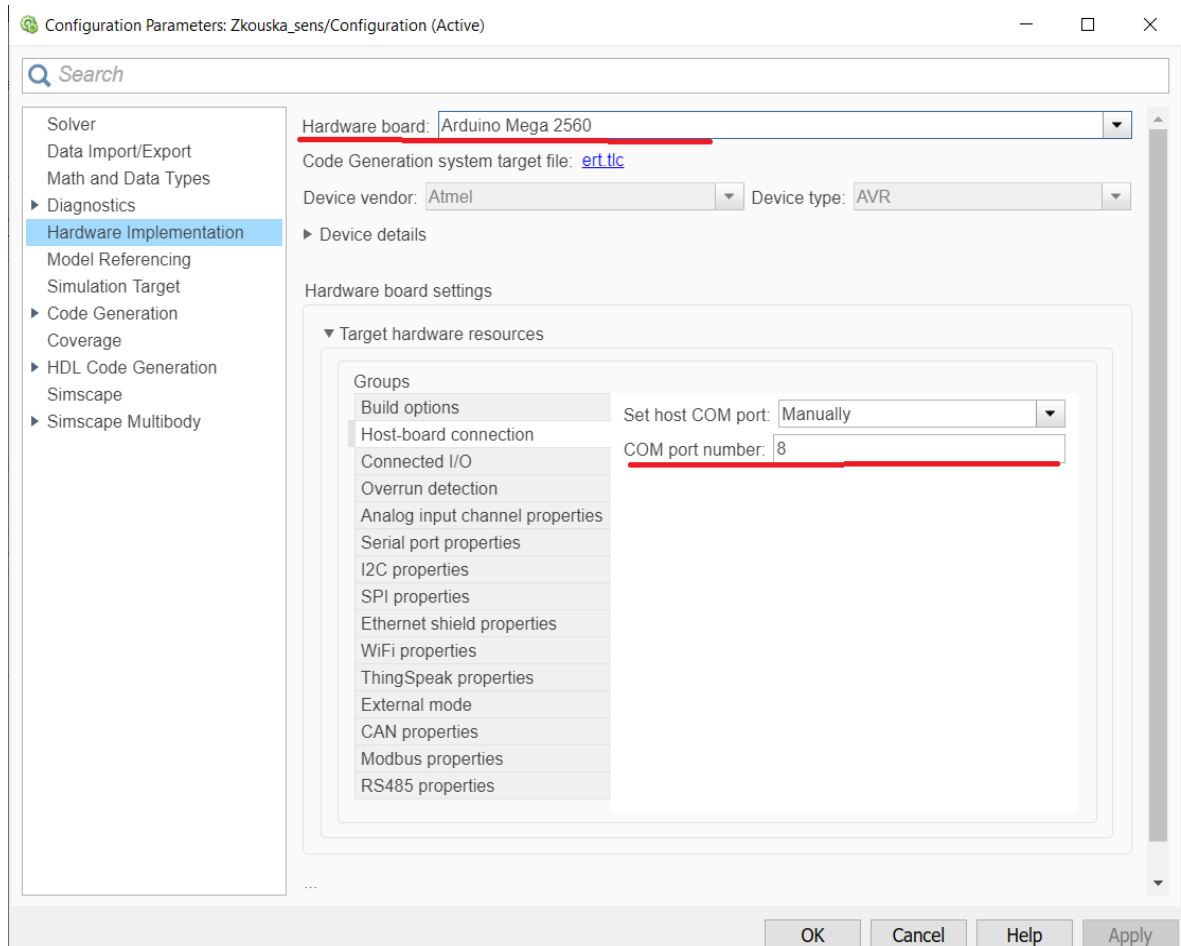
Obrázek 87 Podstavec pro pájitelné pole



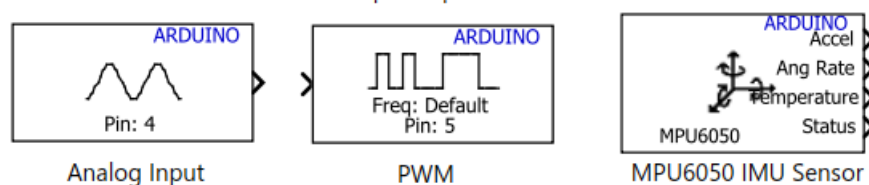
Obrázek 88 Řídicí rukavice

### 6.3. Spojení řídicí rukavice s modelem

Prvně je třeba do modelu implementovat používaný hardware. Arduino implementujeme rozkliknutím konfiguračních parametrů v Simulinku a poté v záložce Hardware Implementation zvolíme používanou desku a nastavíme port využívaný deskou (Obrázek 88).

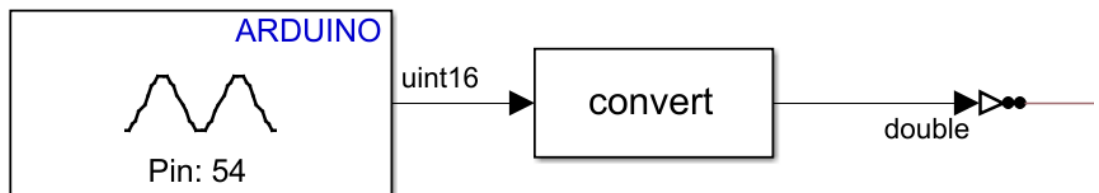


Obrázek 89 Implementace hardwaru do prostředí simulinku



Obrázek 90 Použité bloky Arduino/ Simulink

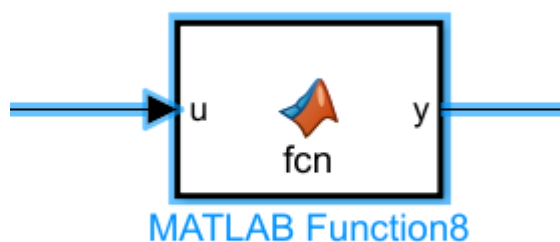
Ve vytvořeném simulačním modelu (Obrázek 38) je pro buzení modelu použit blok Ramp. Protože chceme model ovládat pohyby vlastní ruky skrze řídicí rukavici, je zapotřebí nahrát hodnoty ze senzorů do modelu. Na obrázku 90 vidíme pouze vybrané tři bloky z celé škály bloků vytvořeném v balíčku pro Arduino. Pomocí prvního bloku Analog Input nahrajeme analogové hodnoty ze senzorů, které můžeme vidět na obrázcích 81, 82 a 83. Protože datový typ dat posílaných z Arduina neodpovídá datovému typu vyžadovaným Simulinkem, vložíme mezi modelem a výstupem z Arduina converter, který změní typ vstupních dat na požadovaný (Obrázek. 91). Konkrétně, jak vidíme na obrázku, jedná se o datové typy uint16 (unsigned integer number – 16 bit) a double (double precision floating point - 64 bit).



Obrázek 91 Convert blok

#### 6.4. Data ze senzorů natočení

Analogové hodnoty ze senzorů vidíme na obrázku 81, 82 a 83. K převedení těchto hodnot na odpovídající hodnotu natočení byl použit blok MATLAB function (obrázek 92). Do tohoto bloku je posílána hodnota odečtená ze senzorů (u) a z té je posíláno odpovídající natočení v radiánech (y).



Obrázek 92 Matlab function

Jelikož odečtené hodnoty ze sensorů se nepohybují od nuly k určité maximální hodnotě, použil jsem tento blok k napsání jednoduché mapovací funkce. Pro každý sensor bylo provedeno měření a z něhož byly odečteny potřebné hodnoty a to maximální hodnota signálu při klidu ( $in\_max$ ) a při plném sevření ( $in\_min$ ). K těmto krajním hodnotám odpovídají hodnoty  $out\_max = 0 [rad]$  a  $out\_min = 3.14 [rad]$ . Spolu se vstupem ze sensorů  $u$  figurují tyto hodnoty v rovnici níže.

$$\varphi = (u - inmin.) * \frac{(outmax. - outmin.)}{(inmax. - inmin.)} + outmin. \dots [rad]$$

Jelikož senzory někdy mohou nabýt větších či menších hodnot, než jsou hraniční hodnoty  $in\_min$  a  $in\_max$ , zvláště u sensorů vyráběných, jsou dále hodnoty podrobeny kontrole, které v případě, že úhel natočení přesáhne požadovanou hodnotu, přepíše se tato hodnota na hodnotu hraniční. Funkce je vidět na obrázku 93 a celkové uspořádání bloků vidíme na obrázku 94.

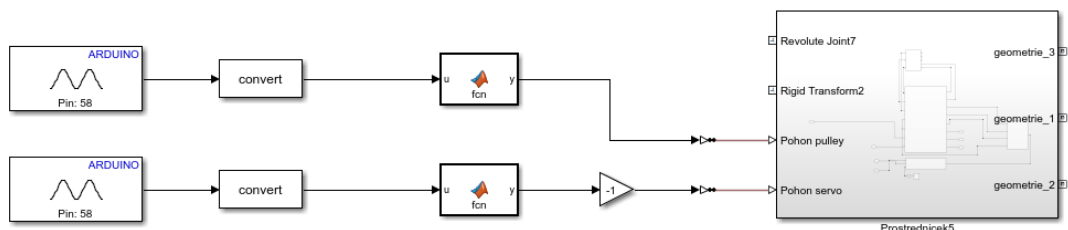
```
function y = fcn(u)

in_min = 210;
in_max = 270;

out_max = 0;
out_min = 3.14;
Fi=(u - in_min) * (out_max - out_min) / (in_max - in_min) + out_min;
if Fi < 0
    y = 0;
elseif Fi > 3.14
    y = 3.14;
else
    y = Fi;
end

end
```

Obrázek 93 Map function



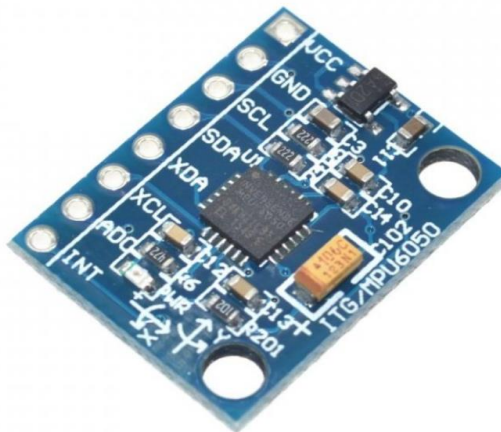
Obrázek 94 Celkové uspořádání



## 6.5. Gyroskop

V rámci ovládní Simscape modelu byla snaha otáčet model podle natočení vlastní ruky pomocí gyroskopu. Bohužel po úspěšné kalibraci gyroskopu se nepodařilo úspěšně aplikovat gyroskop do modelu. Model reagoval příliš opožděně na data z gyroskopu a výsledky nejsou použitelné. Z těchto důvodů po konzultaci s vedoucím této práce jsem se rozhodl gyroskop nepoužít. Na ukázkou uvedu aspoň postup, kterým jsem gyroskop zkalibroval.

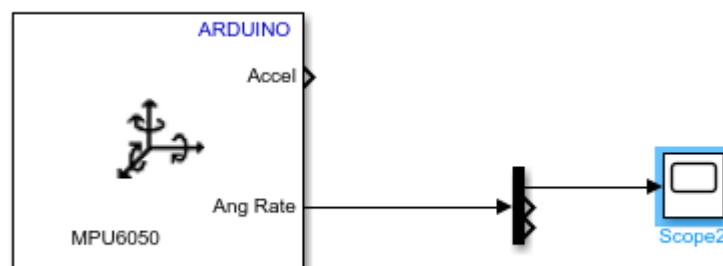
Použitý gyroskop byl MPU 6050 na obrázku 95.



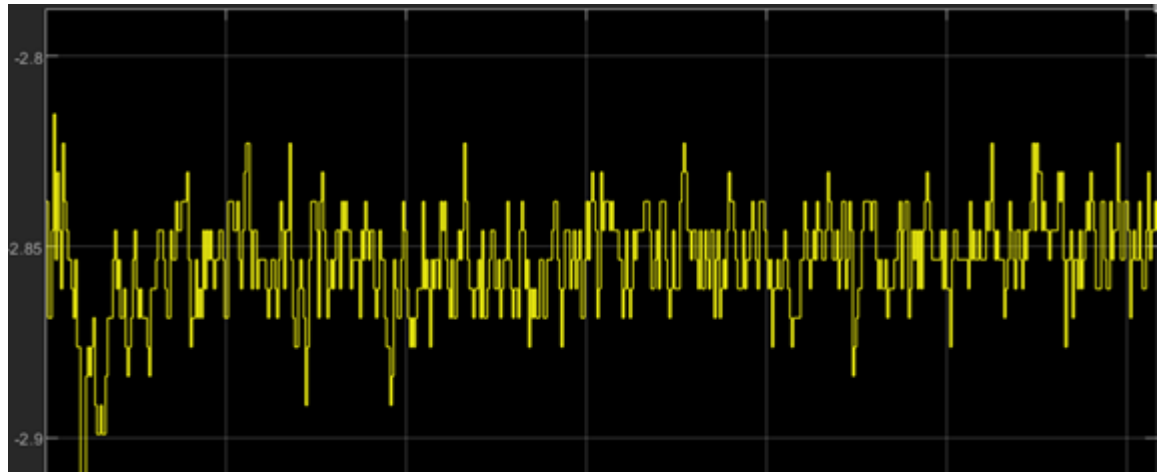
Obrázek 95 3-osý gyroskop s akcelerometrem

### Kalibrace

V simulinku jsem využil blok MPU 6050 IMU Sensor (obrázek 90) stavěný pro tento typ gyroskopu. Po připojení sensoru (obrázek 96) jsem zaznamenal data úhlové rychlosti (v klidu) kolem osy x.

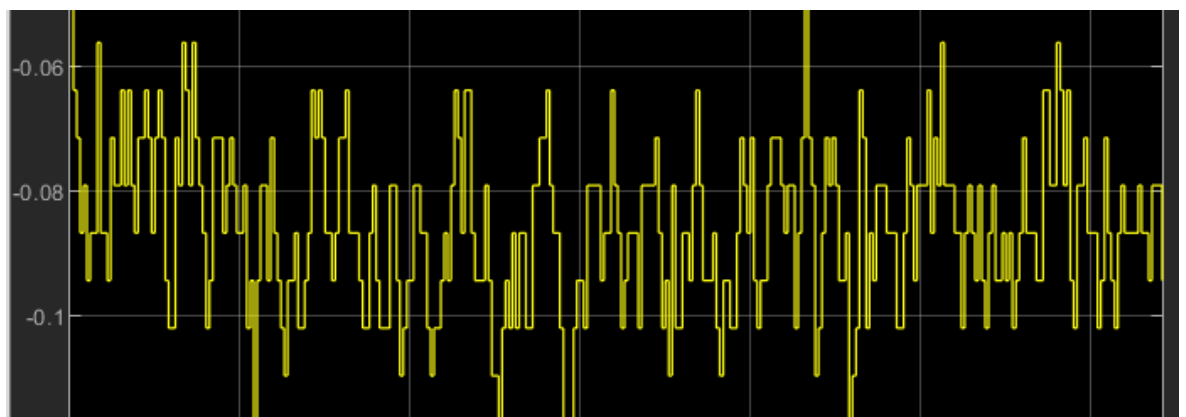


Obrázek 96 Gyroskop - Simulink



Obrázek 97 Data gyroskop\_2

Jelikož se data úhlové rychlosti při klidu pohybují okolo hodnoty  $-2.82$ , přičetl jsem k signálu hodnotu  $+2.82$ . Výsledná úhlová rychlost na obrázku 98 se již pohybuje v blízkosti požadované nulové hodnoty. V dalším kroku byla vytvořena funkce, v níž za pomoci jednoduché podmínky (obrázek 99) byl odstraněn přebytečný šum (za klidu).



Obrázek 98 Data gyroskop\_2

```

function y = fcn(u)

    if abs(u) < 0.2
        y = 0;

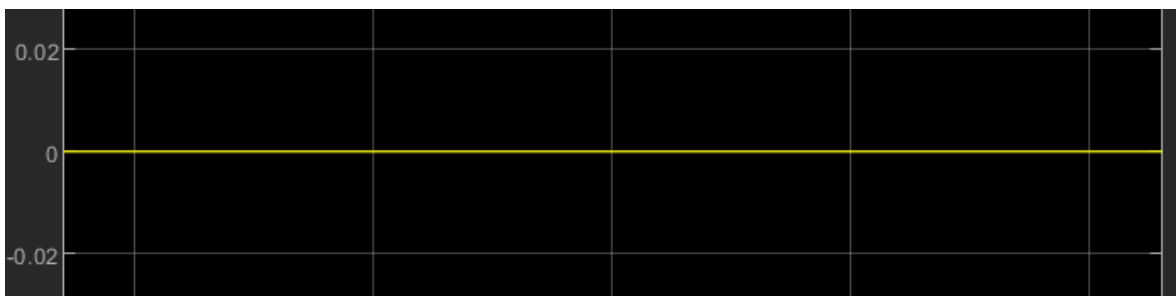
    else
        y = u;
    end

end

```

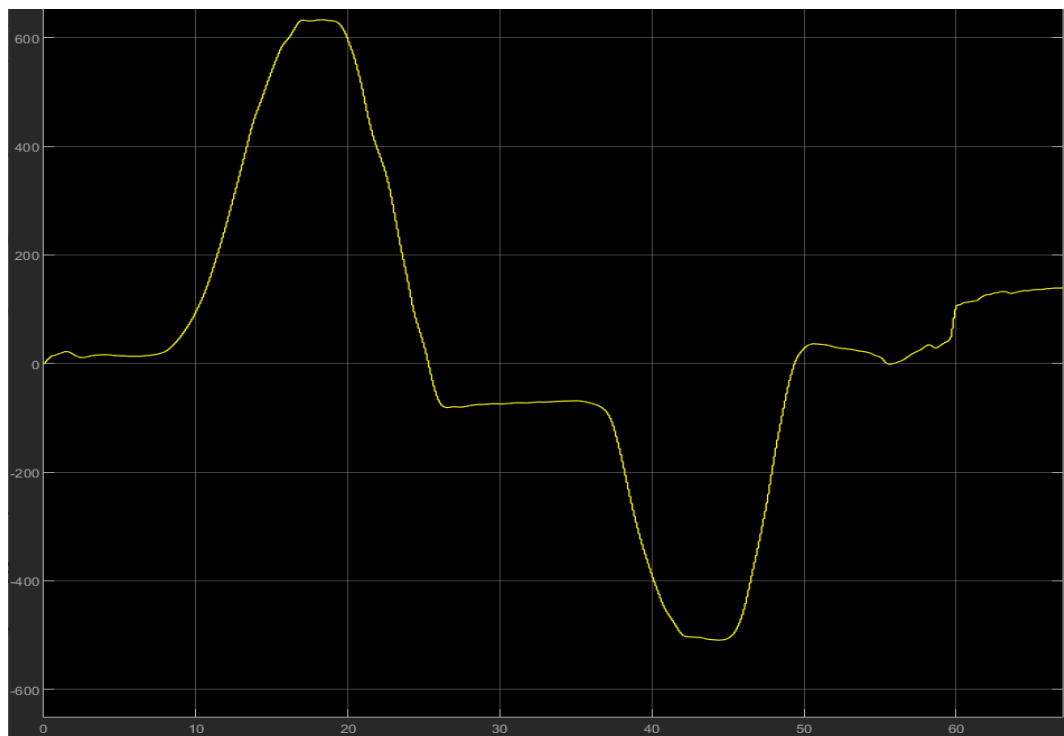
Obrázek 99 Gyroskop – funkce pro odstranění přebytečného šumu z gyroskopu

Výsledná úhlová rychlost v klidu po kalibraci (obrázek 100).

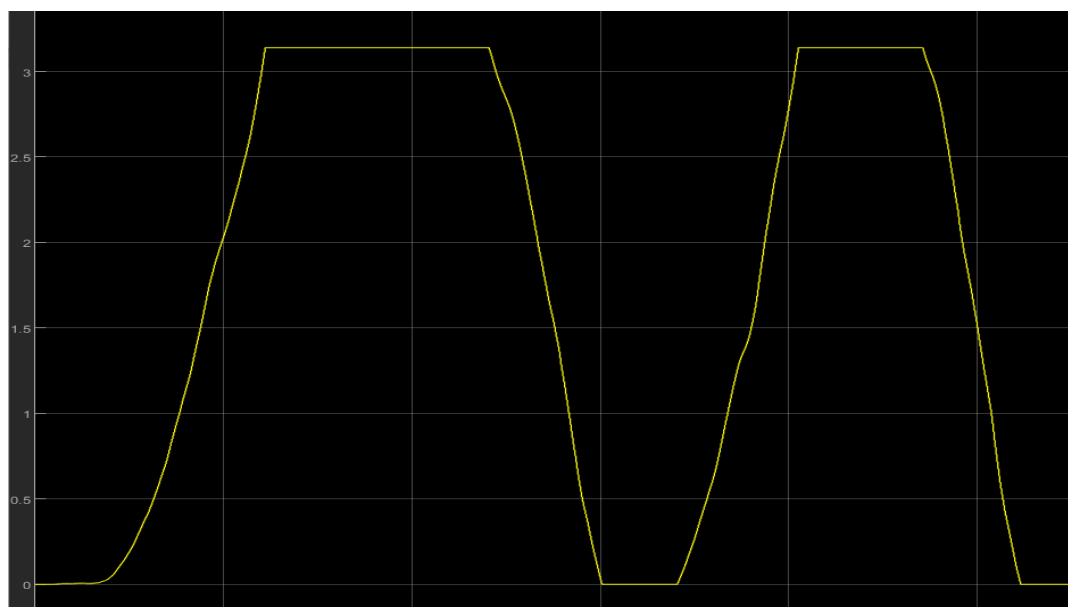


Obrázek 100 Data gyroskop\_3

Tato hodnota byla následně integrována pro získání úhlu natočení (obrázek 101). Gyroskop byl nakloněn o  $90^\circ$  a podruhé o  $-90^\circ$ , křivka tvarově odpovídá provedeným pohybům, ale jejich hodnota ne. Proto byla data zpracována podobně jako u senzorů natočení na obrázku 93. Výsledek je na obrázku 102 (kde hodnoty jsou uvedeny v radiánech), kde sensor byl dvakrát nakloněn z  $0^\circ$  do  $180^\circ$  což odpovídá vzhledem k naměřeným datům.



Obrázek 101 Data z natočení gyroskopu 1



Obrázek 102 Data z natočení gyroskopu 2

## 6.6. Závěr kapitoly

V této kapitole byla rozebrána tvorba řídicí rukavice. Úspěšně byly aplikovány sensory pro snímání míry natočení prstů, podařilo se také vytvořit a aplikovat vlastní sensory míry natočení prstů, ale bohužel se nepodařilo uspokojivě zprovoznit gyroskop na simulačním modelu. Závěrem kapitoly bych rád na videu, které je v příloze, demonstroval řízení modelu pomocí rukavice.

- One\_finger\_simulation\_1 – video

Toto video a všechna ostatní mají rozdělenou obrazovku, kde na jedné obrazovce se nachází simulační model a na druhé obrazovce ruka řídicí rukavicí. V simulačním modelu je použit pouze model jednoho prstu, a to pouze pro porovnání s demonstrací se simulačním modelem, který obsahuje všechny modely prstů. Jelikož dynamika pouze jednoho prstu není, tak náročná pro výpočet a model reaguje téměř real-time a lze, tak spolehlivě demonstrovat funkci vyrobeného a kupovaného senzoru, kdy senzor kupovaný ovlivňuje pohyb, který je na obrázku 20 a senzor vyrobený senzor ovlivňuje pohyb na obrázku 19.

- One\_finger\_simulation\_2 – video

V tomto video se nachází pouze palec, kdy byla snaha demonstrovat pohyby na obrázcích 25 a 23. Kde koupený senzor ovlivňuje pohyb konaný na obrázku 25 a vyrobený senzor ovlivňuje pohyb z obrázku 23. Jelikož se jedná pouze o simulaci palce, reakce modelu je téměř real-time.

- Whole\_hand\_simulation\_1 – video (Pouze kupované senzory)

V této demonstraci se nachází již celý simulační model, kde byly zapojeny pouze senzory zakoupené (obrázek 103 – modře označené). Pro provedení pohybu 19, je posílán poloviční signál z kupovaných senzorů do spodní rotační vazby. Signál je poloviční, protože lanový buben se při maximální výchylce natočí o  $180^\circ$  a spodní vazba pouze o  $90^\circ$ . Simulace je již díky náročnějšímu na výpočtu oproti minulým demonstracím značně zpožděná.

- Whole\_hand\_simulation\_2 – video (Všechny senzory)

Jelikož vyrobené senzory jsou více citlivé a reagují i na namáhání typu tah/tlak, tak docházelo k rušení signálu sousedních senzorů. Proto jednoduší byly použité senzory označeny na obrázku 103, kdy byly vyrobené senzory pro prostředníček a prsteníček odstraněny. Proto ukazováček a prostředníček sdílejí ve videu stejnou hodnotu natočení spodního článku. Také hodnota ze senzoru pro snímání ohybu spodního článku malíčku je sdílána se spodním článkem prsteníčku.



*Obrázek 103 Označené použité sensory na řídicí rukavici*

## 7. Fyzický experiment

Závěrem práce je postaven reálný model, ovládaný řídicí rukavicí na obrázku 88.

### 7.1. Použité programy

#### Autodesk Inventor 2020

Pro vytváření jednotlivých 3D dílů modelu byl použit parametrický modelářský program Autodesk Inventor 2020.

#### Ultimaker Cura

Jedná se o open source program využívaný k extrahování z vymodelovaných částí takzvaný G – kód, který je dále vložen do 3D tiskárny.

### 7.2. 3D tisk

K vytvoření modelu byla využita 3D tiskárna 3D – Ender 3 pro od firmy Creality. Materiál využitý pro tvorbu modelu jsem zvolil materiál PLA, který se vyznačuje teplotou tání ( 175 až 178 ) $^{\circ}$ C°. Materiál PLA je jednodušší pro tisk a nepotřebuje větší teploty oproti jiným dostupným materiálům jako například ABS.



Obrázek 104 Ender pro [16]

### 7.3. Vytvoření 3D modelů

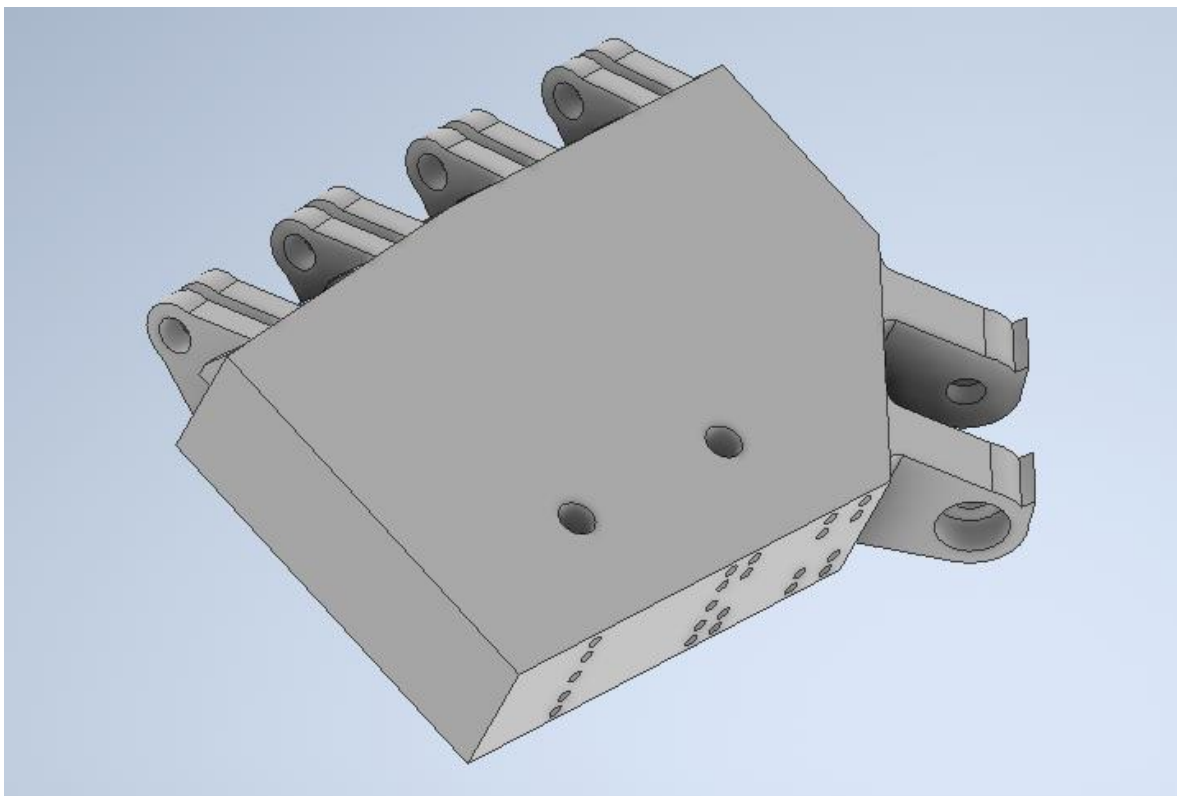
Pro vytvoření 3D modelů byla snaha zachovat některé hlavní rozměry ze simulačního modelu, a to délku prstů a rozměry dlaně.

#### 7.3.1. Dlaň

Pro rozměry dlaně byly použity hodnoty z tabulky 1. Některé hodnoty byly, ale z praktických a konstrukčních důvodů zvětšeny. Použité hodnoty reálného modelu jsou uvedeny v tabulce 10.

*Tabulka 10 Rozměry dlaně reálného modelu*

|                          |                       |
|--------------------------|-----------------------|
| Šířka dlaně              | $L = 94 [mm]$         |
| Úhel klesání šikmé části | $\alpha = 5 [^\circ]$ |
| Výška dlaně              | $h = 60 [mm]$         |
| Šířka rovné části        | $L_r = 50 [mm]$       |
| Tloušťka dlaně           | $t = 18 [mm]$         |

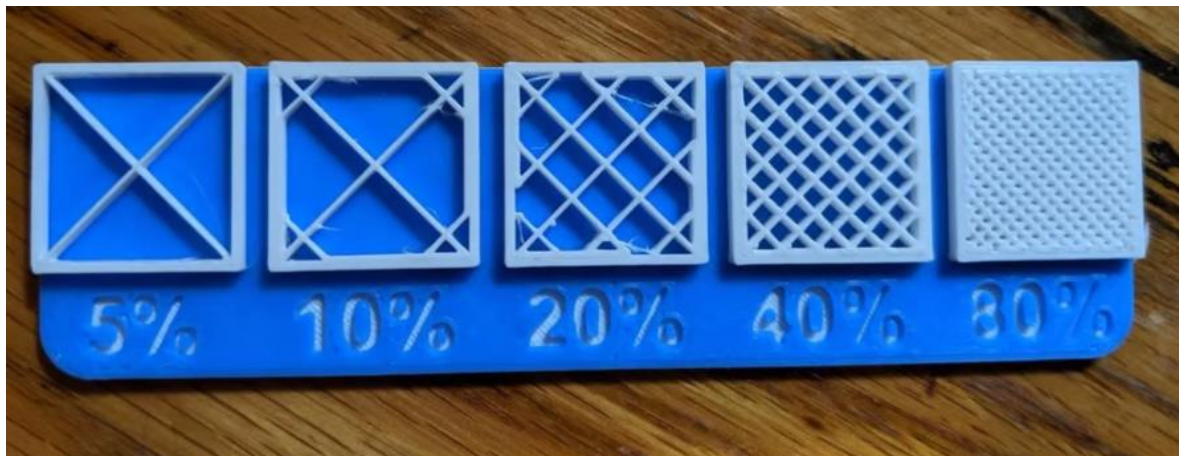


*Obrázek 105 CAD model dlaně*

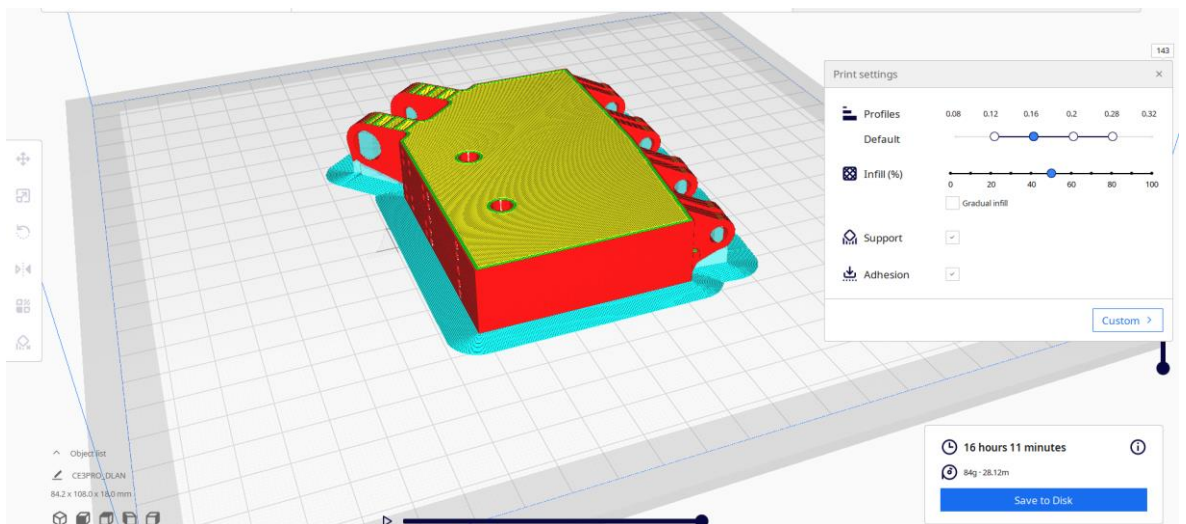


Po dokončení modelování dlaně, byl soubor exportován do formátu STL a nahrán do programu Ultimaker Cura. K velikosti modelu a velké četnosti míst s plným materiálem byla v programu Ultimaker Cura zvolena padesátiprocentní výplň materiálem. Vzhledem k úspoře času byly všechny části fyzického modelu vytisknuty s padesáti procentní výplní, kromě prstů, kde byla použita 100% výplň. Pro představu obrázek 106, kde jsou vytisknuty různé úrovně výplně. Z takto zpracovaného modelu (obrázek 107) je programem vygenerovaný G-kód pro tiskárnu.

Tisk dlaně trval cca. šestnáct hodin.



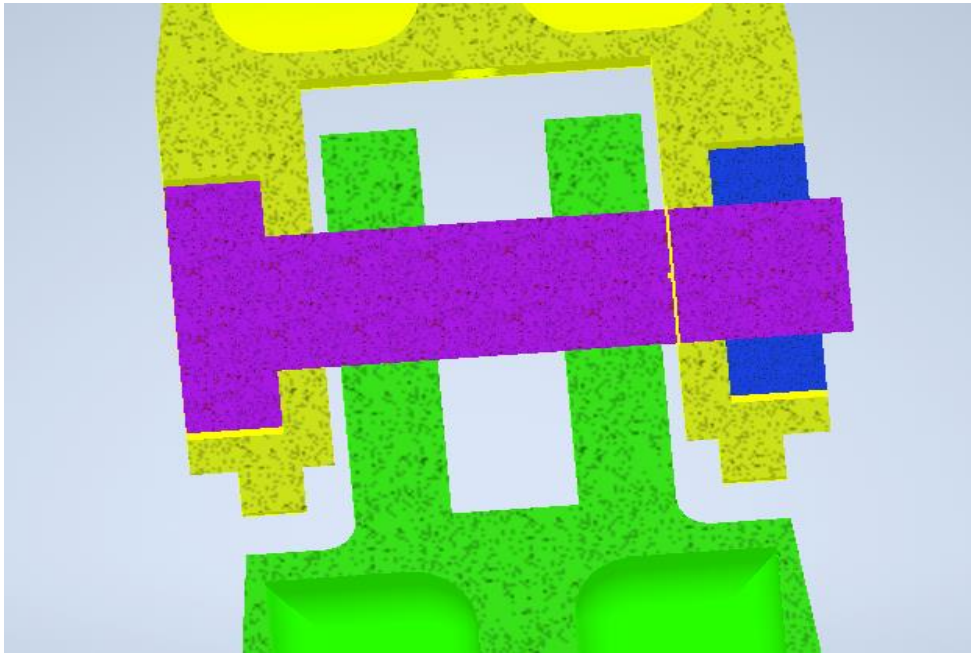
Obrázek 106 3D print infill [17]



Obrázek 107 Dlaň v programu Ultimaker Cura

### 7.3.2. Provedení rotační vazby

Pro rotační vazby na dlani byl použit jeden šroub pro dvě vazby najednou. Použit byl metrický šroub M5 x 50 s půlkulatou hlavou a křížovou drážkou DIN 7985 a pro ostatní vazby byl použit stejný šroub s délkou 30. Délky šroubů byly zkráceny, aby nevyčnívaly ze strany prstů.



Obrázek 108 Návrh vazby v modelu

### 7.3.3. Tvorba článků prstů

Délky článků prstů byly dodrženy dle tabulek 2,3 a 4. Pouze místo válcového profilu byly články nakonec pro jednoduchost modelovány z obdélníkového profilu. Doba tisku jednoho článku trvala cca. 1 až 2 hodiny.

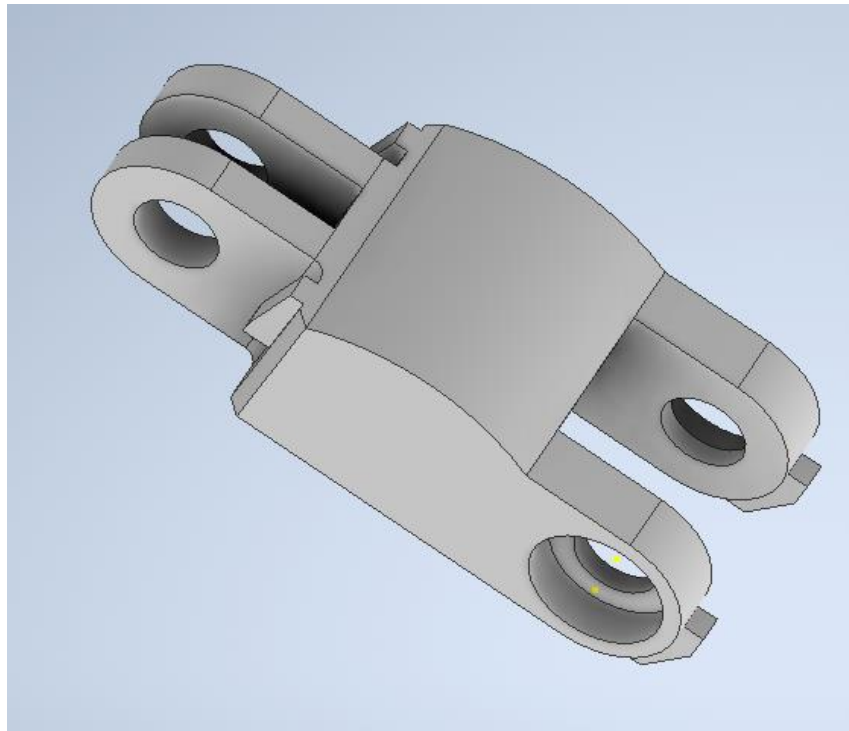
#### Pohony

Článek 2 a 3 budou poháněny pomocí lana, které bude vedeno přes otvor na obrázku 112.A) a uchyceno bude posledním článku pomocí uzlu. Lano se pak ukotví na řemenici (Obrázek 119), kde bude uchyceno pomocí šroubku.

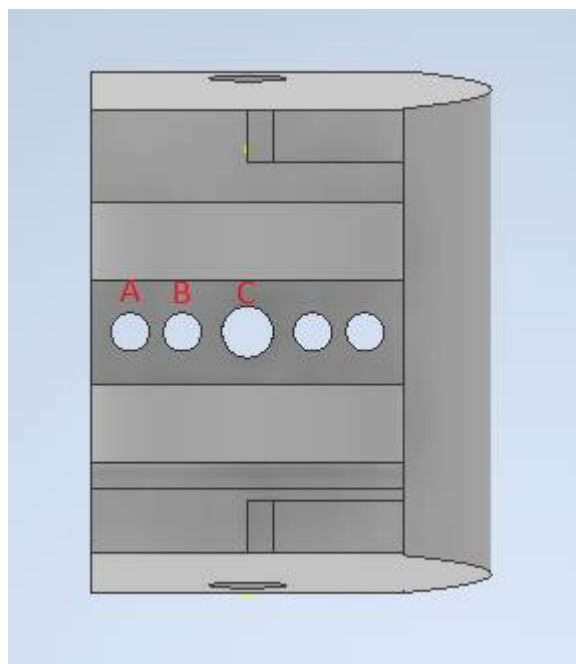
Do reálného modelu bylo nakonec přidáno i lano pro zpětný chod. Zpětný chod bude tedy realizován pomocí zpětného lana a torzních pružin v kloubech. Zpětné lano se provede kanálkem, který je zrcadlově na druhé straně prstu. Torzní pružiny se zachytí v kanálcích 112.C).

Protože spodní články budou mít také vlastní pohony, bylo zapotřebí vymyslet vhodný mechanismus na pohánění spodních článků. Na konec v rámci úspory času bylo rozhodnuto pro použití lan. Lano pro spodní článek povede kanálkem 112.B).

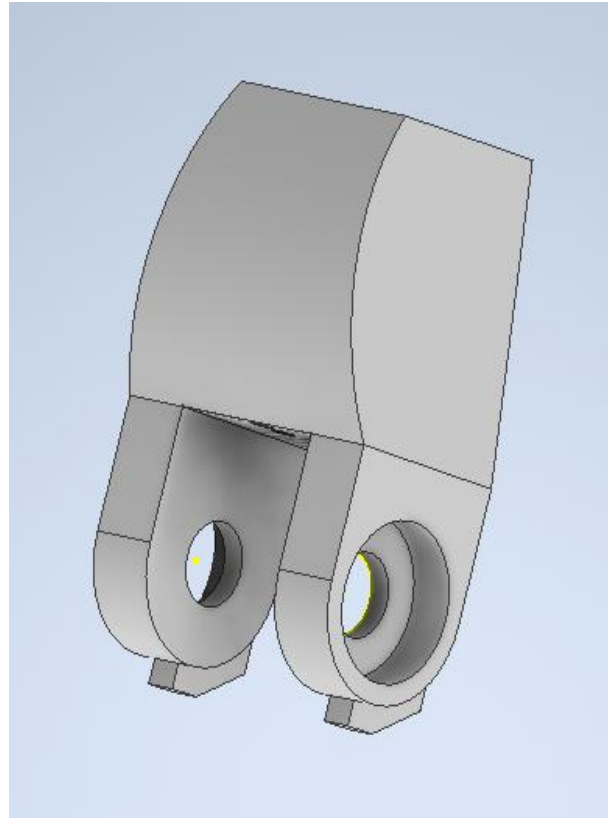
Výhodou použití lan pro aktuaci všech žádaných článků je zachování tloušťky modelu, kdy použití, například pákového převodu, by si vyžádalo více místa.



Obrázek 109 Model článku prstu



Obrázek 110 Jednotlivé kanálky pro vedení lan a uložení pružiny



*Obrázek 111 Koncový článek prstu*

### 7.3.4. Torzní pružiny

K realizaci torzních pružin v kloubech byl zakoupen pružinový drát o průměru  $d = 0.6 \text{ mm}$ . Podle vzorce níže, kde  $E$  je youngův modul pružnosti v tahu pro pružinový drát  $E = 206 \cdot 10^3 \text{ Nmm}^{-2}$ ,  $d$  je průměr drátu,  $N$  počet závitů a  $D$  značí vzdálenost od středu cívký ke středu vnějšího drátu cívký. [18]

Všechny torzní tuhosti v druhém kloubu byly nastaveny v optimalizaci na hodnotu  $1 \text{ Nm} \cdot \text{rad}^{-1}$ . Pro ostatní tuhosti byly ze všech tří optimalizací spočteny průměrné hodnoty, které jsou v tabulce níže. Pro zjednodušení můžeme říct, že hodnota tuhosti optimalizovaných torzních pružin je poloviční od těch ve druhém kloubu.

$$k = \frac{E \cdot d^4}{3667 \cdot D \cdot N}$$

Tabulka 11 Průměrné hodnoty torzních pružin

|          |  |
|----------|--|
| K3_thumb | 0.41 $[\text{Nm} \cdot \text{rad}^{-1}]$ |
| K34      | 0.78 $[\text{Nm} \cdot \text{rad}^{-1}]$ |
| K33      | 0.59 $[\text{Nm} \cdot \text{rad}^{-1}]$ |
| K32      | 0.52 $[\text{Nm} \cdot \text{rad}^{-1}]$ |
| K31      | 0.47 $[\text{Nm} \cdot \text{rad}^{-1}]$ |

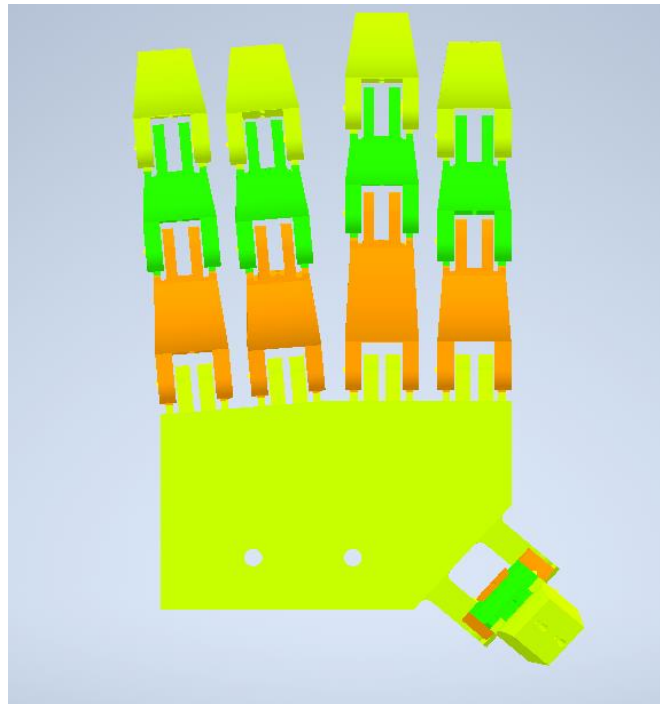
Hodnoty pro tvorbu jsou následující. Odhadovaná vzdálenost  $D = 5 \text{ mm}$  a použití pružinového drátu  $d = 0.6 \text{ mm}$  s  $E = 206 \cdot 10^3 \text{ Nmm}^{-2}$  byly pomocí vzorce výše navrhnuty 2 typy pružin. Pro pružinu ve třetím (horním) kloubu se třemi závitů je odhadovaná tuhost  $k_3 = 0.48 \text{ Nm rad}^{-1}$  a pro spodní se dvěma závitů  $k_2 = 0.78 \text{ Nm rad}^{-1}$ .



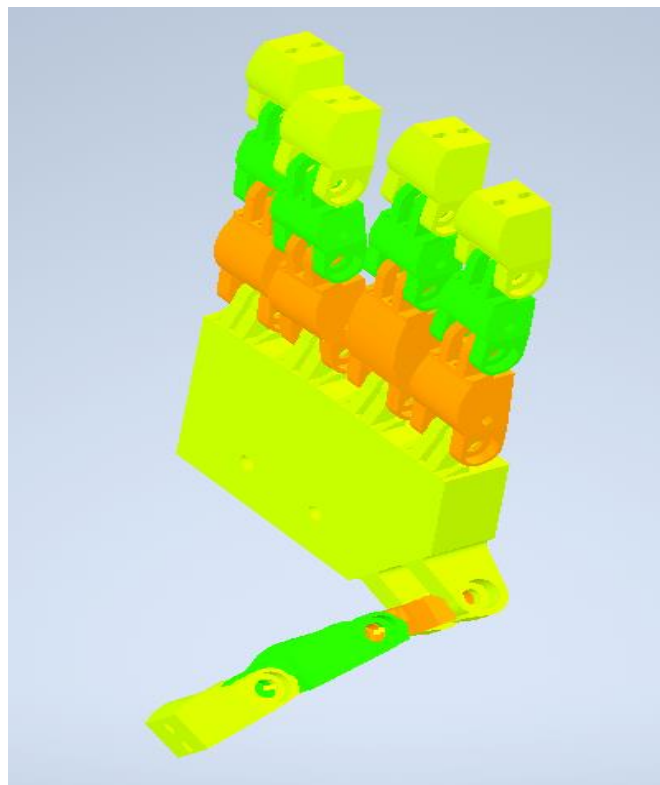
Obrázek 112 Vyrobená torzní pružina

### 7.3.5. Sestavení modelu

Sestavený model můžeme vidět na obrázcích 115 a 116.



Obrázek 113 Sestava\_model\_1



Obrázek 114 Sestava\_model\_2

### 7.3.6. Stojan pro pohony

K modelu ruky na obrázcích 113-114 byl vymodelován stojan, který bude pevně spojen s rukou a umožní tak použití servopohonů typu MG995 [19] s lany k aktuaci modelu. Použitý pohon je servopohon s kovovými převody a možným natočením o 180 °. Parametry servopohonu jsou uvedeny v tabulce 12.

Tabulka 12 Parametry servopohonu [19]

|  |                 |
|--|-----------------|
| Kroutící moment při napětí 4.8 [V] a 6 [V] | (1.2 – 1.5)[Nm] |
| Provozní napětí                            | (4.8 – 7.2)[V]  |
| Odebíraný proud naprázdno                  | $I = 10[mA]$    |
| Odebíraný přídržný proud                   | $I = 1.2 [A]$   |



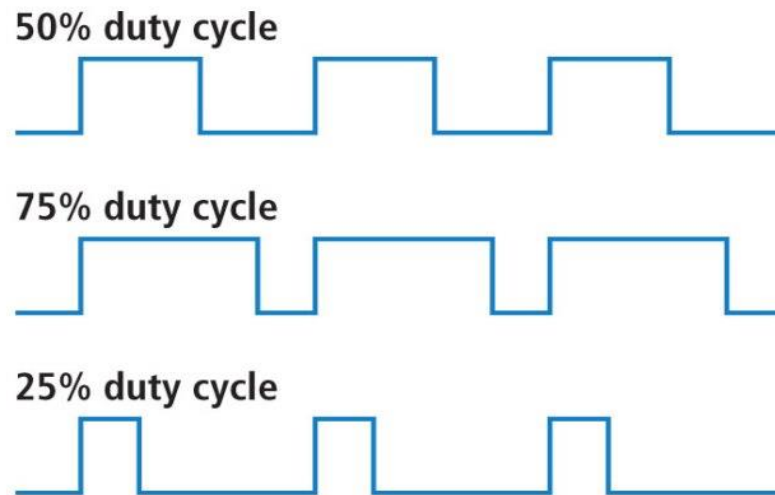
Obrázek 115 Servopohon [19]

#### Princip fungování servopohonu

Servopohon se skládá z DC motoru s potenciometrem a zabudovaným regulátorem. Servopohon obdrží PWM signál, který nese informaci o žádané poloze. Pomocí potenciometru má servopohon zpětnou vazbu o tom, v jakém stupni natočení se nachází. Regulační obvod pak zajistí, aby se servopohon natočil do požadované polohy za požadovaný čas.

- PWM

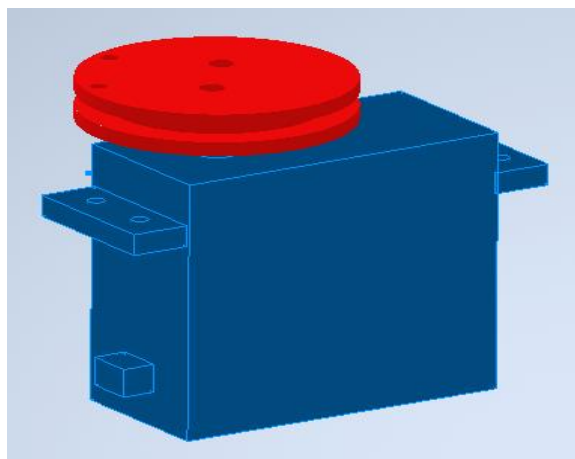
Pulzně šířková modulace je signál, který střídá dvě logické úrovně 1 a 0. V potaz se pak bere takzvaná střída [%], neboli poměr velikosti pulzu v 0 a 1 za jednu periodu. Příkladem, pokud je signál pouze logická 1, hodnota střídy bude 100 % a naopak. Jednotlivé střídy neboli duty cycle, které vidíme na obrázku 116.



Obrázek 116 Velikosti střídy [20]

### Řemenice

K modelu servomotoru byla dále vymodelovaná řemenice k uchycení ocelového lanka o průměru  $d_l = 0.6 \text{ mm}$ . Průměr řemenice byl zvětšen z  $16 \text{ mm}$  na  $25 \text{ mm}$  a to z důvodu uchycení řemenice na servomotor. Sestavu motoru s řemenicí vidíme na obrázku 117.

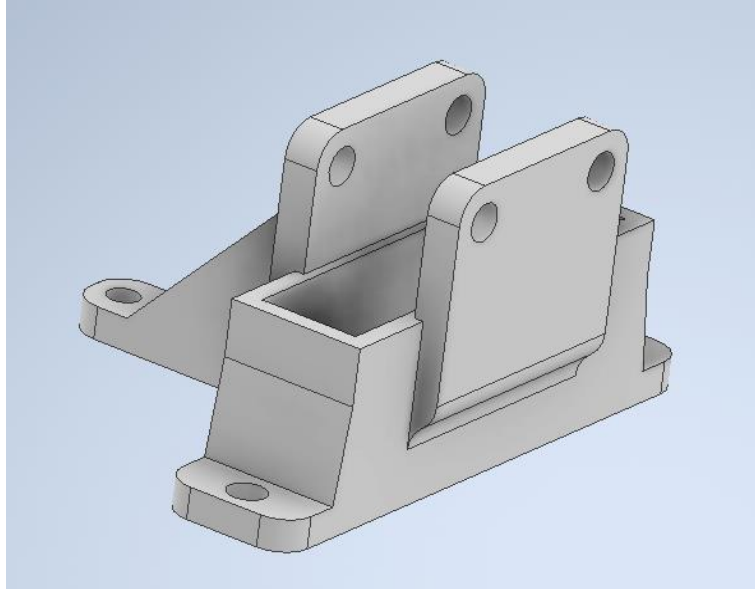


Obrázek 117 Řemenice/ Motor



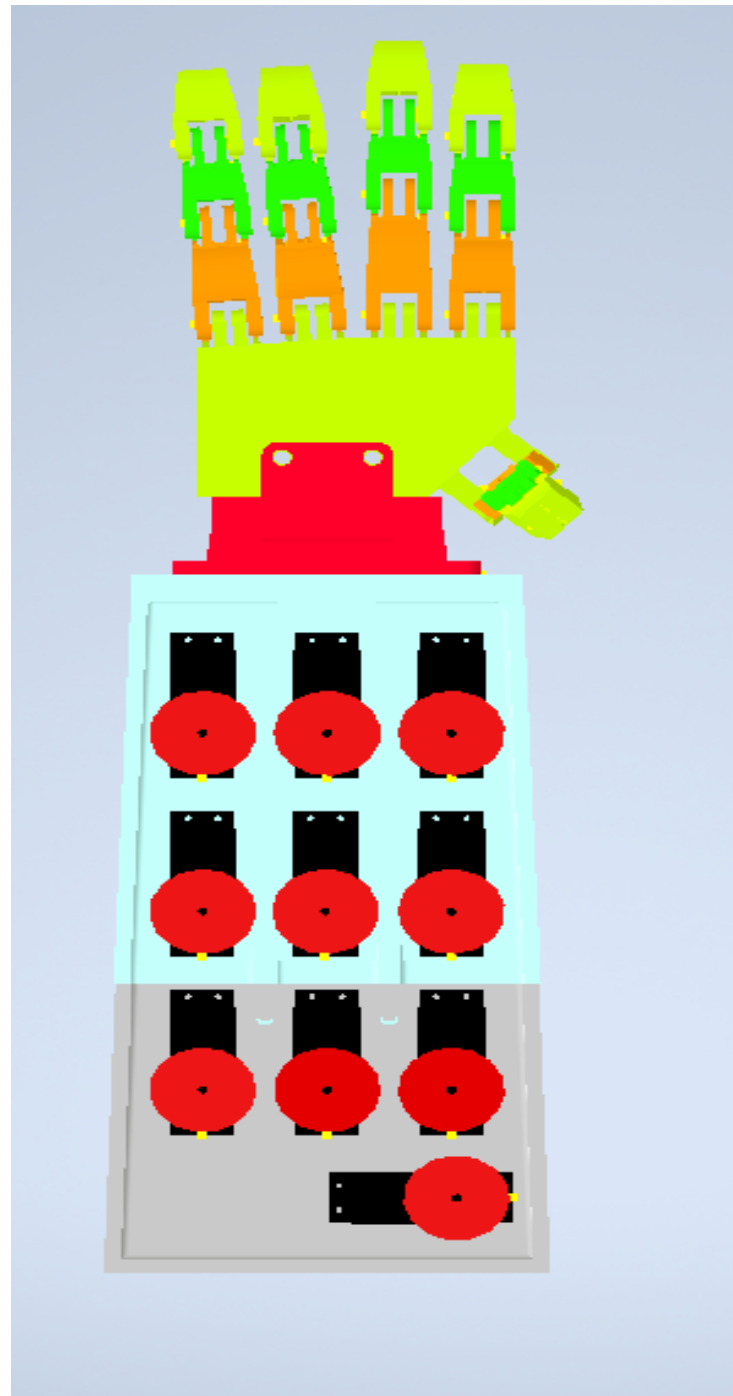
### Modelování stojanu

Aby ruka nebyla přimontována přímo ke stojanu s motory, vymodeloval jsem pro ni mezi prvek, který může reprezentovat zápěstí. Prvek pro spojení stojanu s rukou vidíme na obrázku 118. Doba tisku stojanu na obrázku trvala 5 hodin.

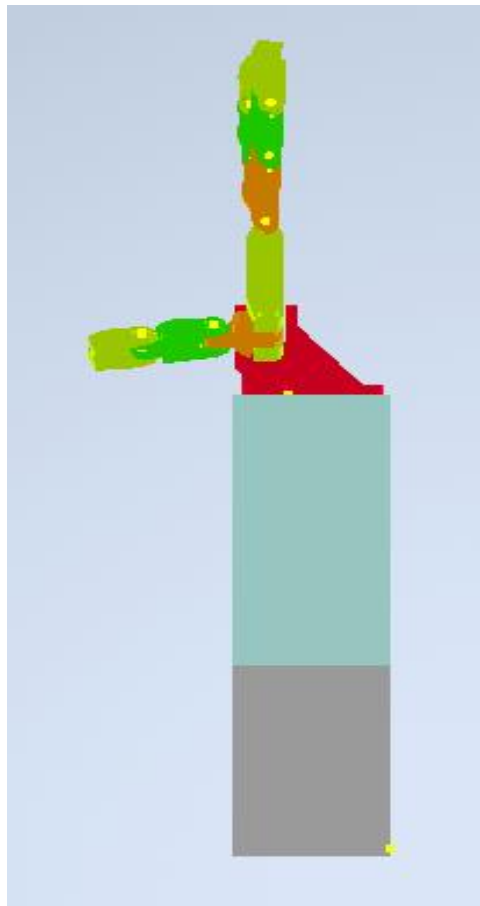


Obrázek 118 Spojovací prvek

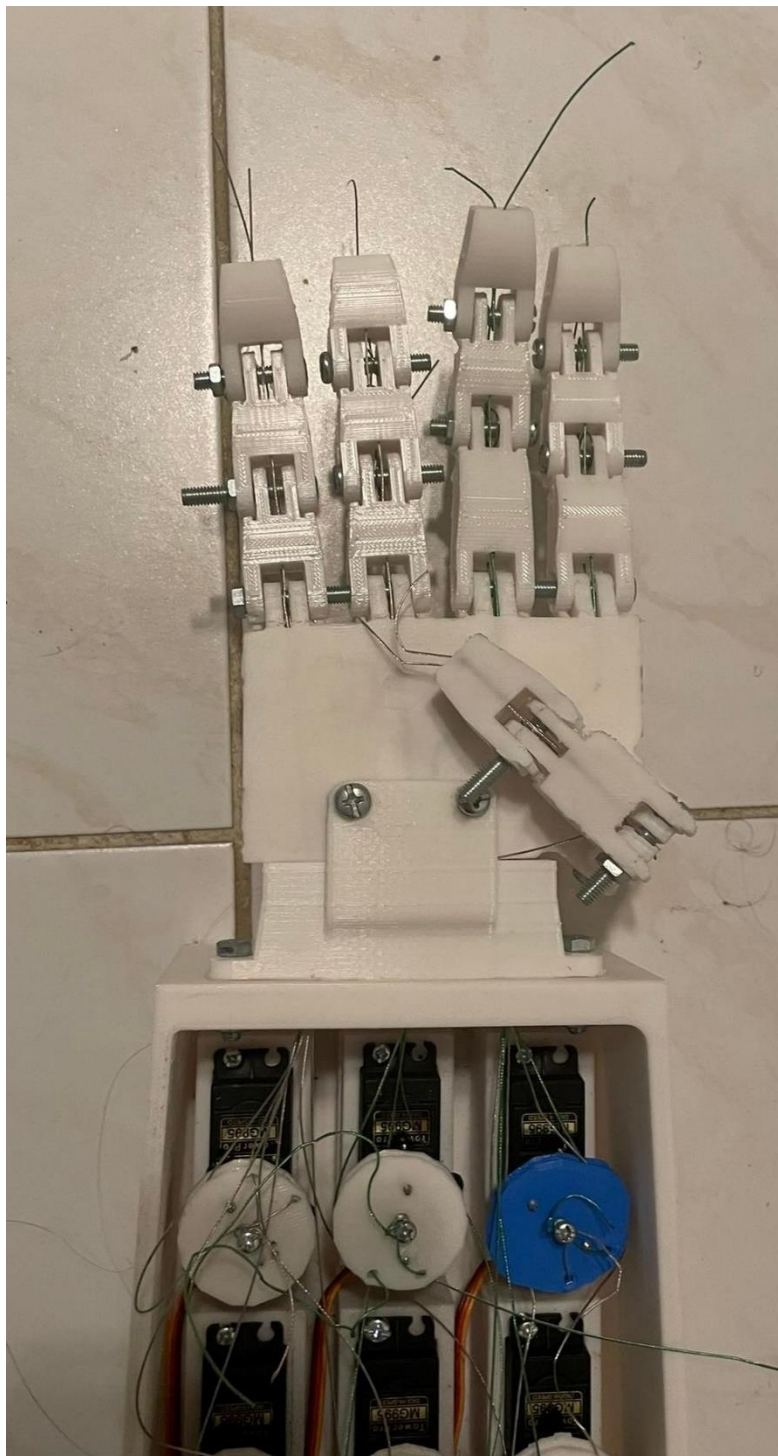
Samostatný stojan musel být díky své délce modelován na dva kusy, aby se vešel na plochu 3D tiskárny. Stojan obsahuje deset míst pro servopohony. Na konci stojanu jsou vymodelovány díry pro připevnění svorkovnice, kterými jsou vyvedeny kabely pro napájení a uzemnění. Z druhé strany svorkovnice se bude připojen zdroj elektrického proudu (obrázek 123). Ostatní vyvedené kabely z motorů pro PWM jsou zapojeny přímo do mikrokontroleru. Stojan spolu s ostatními částmi modelu můžeme vidět na obrázku 119 a 120. Doba tisku obou částí stojanu trvala dohromady dva dny.



Obrázek 119 Sestavený model – pohled 1



Obrázek 120 Sestavený model – pohled 2

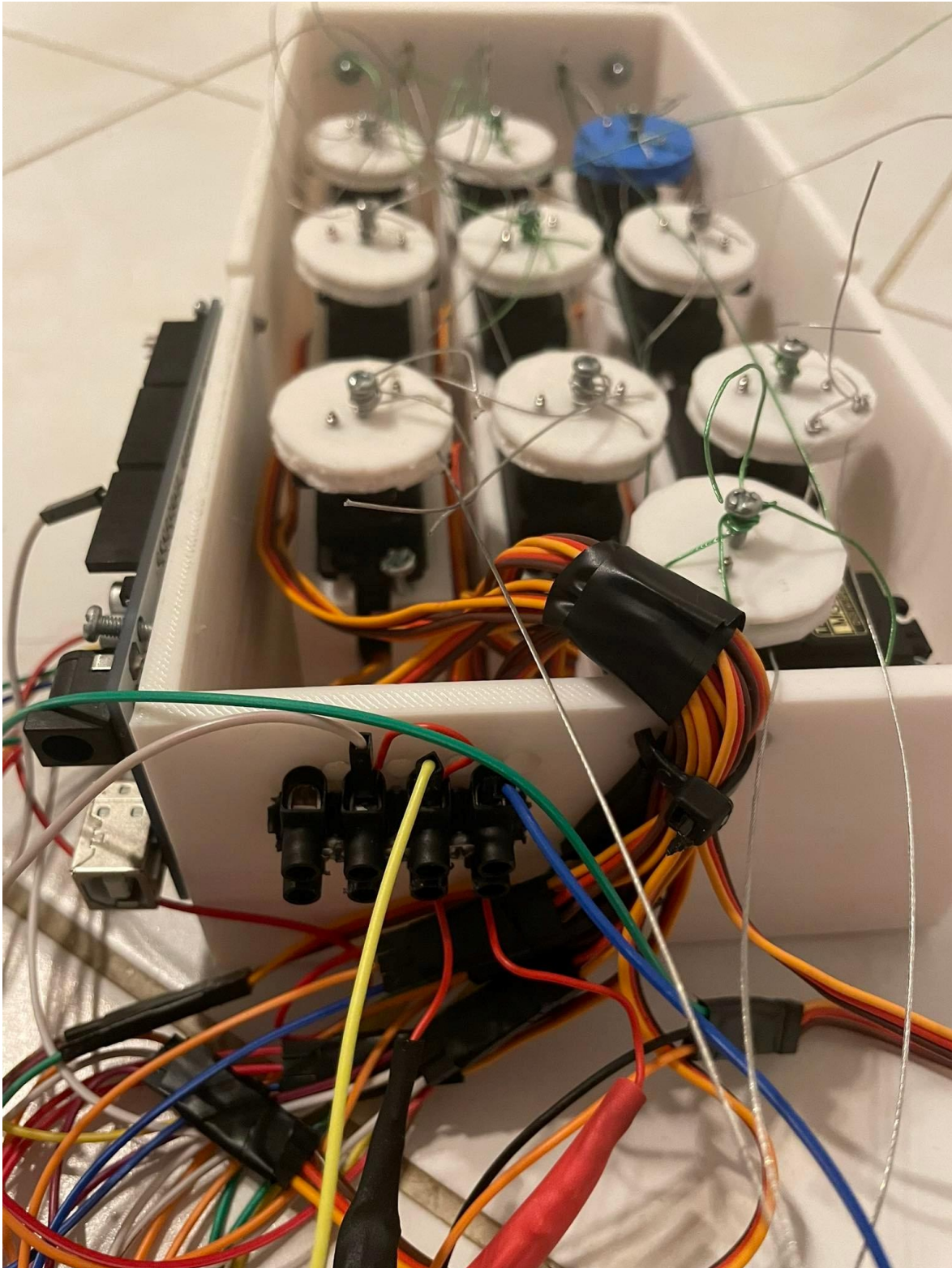


*Obrázek 121 Vytisknutí model\_pohled\_1*

Na obrázcích 121 a 122 vidíme kompletní sestavení modelu, kde bylo ještě zapotřebí zkrátit délku jednotlivých šroubků.

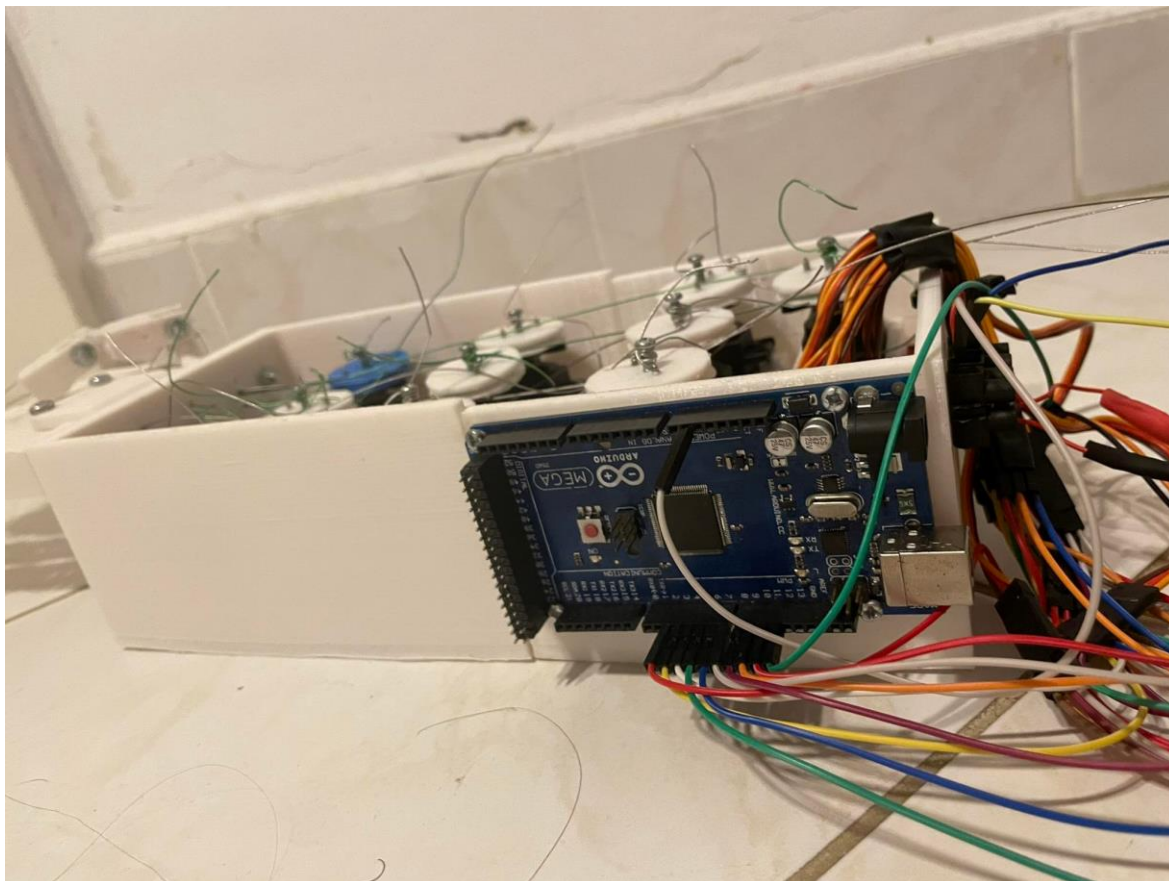


Obrázek 122 Vytisknutý model\_pohled\_2



*Obrázek 123 Elektrické zapojení*

Jednotlivé kabely jsou spájeny do jednoho, kde spoj pro kabely se zápornou polaritou je překryt černou bužírkou a pro kladnou polaritu bužírkou červenou. Takto jsou vyvedené kabely z motorů přivedeny na svorkovnici odkud také vychází dva kabely pro připojení zdroje. Důležité je také přivedení kabelu ground z Arduina do svorkovnice se zápornou polaritou.



*Obrázek 124 Umístění Arduina na modelu*

## 8. Závěr

V této práci jsem se zabýval vytvořením simulačního modelu, který měl sledovat kinematiku lidské ruky. Ze základních možných pohybů lidské ruky byly vybrány pohyby, které byly použity pro tuto práci.

V rámci optimalizace modelu byly vybrány tři předměty, a to koule, kužel a kvádr. Kritérium pro uchopení byla minimalizace vzdáleností povrchů bříšek prstů při prvním kontaktu. Jako optimalizační parametry byly voleny souřadnice uchopovaných předmětů vůči dlani ruky, torzních tuhostí pružin v kloubech prstů, délka spodního článku palce a jeho natočení a doba potřebná k natočení jednotlivých pohonů.

Dále byla vyrobena řídicí rukavice se senzory natočení prstů. Ke kupovaným sensorům pro rozšíření agility bylo vytvořeno dalších pět sensorů z materiálu Velostat. K zapojení sensorů bylo využito pájecí poje s rezistory, které bylo připevněno na podstavce, a nakonec spolu se senzory pomocí tavné pistole připevněno na rukavici. Záměrem bylo k rukavici připojit gyroskop, který se ani po kalibraci nepovedlo úspěšně aplikovat na simulační model, a proto nebyl aplikován ani do reálného modelu. Nakonec byla natočena natočena videa demonstrující ovládání simulačního modelu za pomoci řídicí rukavice.

Nakonec byl v prostředí Autodesk Inventoru navržen model dle parametrů modelu simulačního. Z modelů byl dále vytvořeny G-kódy, díky kterým byly jednotlivé díly ruky vytištěny. Do kloubů modelu byly vloženy vyrobené torzní pružiny a za pomoci ocelového lanka a servomotorů byl spolu s Arduinem a řídicí rukavicí vyrobený model rozpohybován.

Cíle stanovené v zadání práce byly úspěšně splněny. Bohužel ne všechny výsledky dosažené v téhle práci byly uspokojivé. V budoucnu bych rád do rukavice a modelu přidal možnost haptické zpětné vazby.



**SEZNAM OBRÁZKŮ**

|   |    |
|---|----|
| Obrázek 1 Úchopná hlavice s permanentním magnetem [2] .....           | 3  |
| Obrázek 2 Přípravky pro robotickou hlavici [3] .....                  | 3  |
| Obrázek 3 Schéma hlavice s pevnou a odpruženou čelistí [1] .....      | 4  |
| Obrázek 4 Schéma bloků [1] .....                                      | 4  |
| Obrázek 5 základní typy čelistí [1] .....                             | 5  |
| Obrázek 6 Příklad dvoučelistového robota [4] .....                    | 5  |
| Obrázek 7 Tříčelistový mechanismus [5] .....                          | 6  |
| Obrázek 8 Tříprstý mechanismus [6] .....                              | 6  |
| Obrázek 9 Hlavice pro jemnou manipulaci [22] .....                    | 7  |
| Obrázek 10 Hlavice pro jemnou manipulaci [21] .....                   | 7  |
| Obrázek 11 Robustní ruka [8] .....                                    | 9  |
| Obrázek 12 Robustní ruka [8] .....                                    | 9  |
| Obrázek 13 Biomimetic Anthropomorphic Robotic Hand [9] .....          | 10 |
| Obrázek 14 Schéma – Biomimetic Anthropomorphic Robotic Hand [9] ..... | 10 |
| Obrázek 15 Shadow hand [23] .....                                     | 11 |
| Obrázek 16 Inmoov detail ruky [24] .....                              | 12 |
| Obrázek 17 Inmoov robot [24] .....                                    | 12 |
| Obrázek 18 Kostí v lidské ruce [25] .....                             | 13 |
| Obrázek 19 Pohyb prstů 1 .....  | 14 |
| Obrázek 20 Pohyb prstů 2 .....  | 14 |
| Obrázek 21 Prsty v klidu .....  | 14 |
| Obrázek 22 Natočení prstů do stran .....                              | 14 |
| Obrázek 23 Pohyb palce 3 .....  | 15 |
| Obrázek 24 Pohyb palce 1 .....  | 15 |
| Obrázek 25 Pohyb palce 2 .....  | 15 |
| Obrázek 26 Pohyb palce 4 .....  | 15 |
| Obrázek 27 Pohyb palce 5 .....  | 15 |
| Obrázek 28 Porovnání Simulink/Simscape [11] .....                     | 16 |
| Obrázek 29 Základní bloky .....                                       | 17 |
| Obrázek 30 Bloky pro sestavení dlaně .....                            | 18 |
| Obrázek 31 model dlaně .....  | 18 |
| Obrázek 32 Sub systém dlaně .....                                     | 18 |
| Obrázek 33 Cylindrical solid .....                                    | 19 |
| Obrázek 34 Nastavení parametrů kloubu .....                           | 20 |
| Obrázek 35 Elipsoidální solid blok .....                              | 21 |
| Obrázek 36 Část subsystému prstů se stavebními bloky prstu .....      | 21 |
| Obrázek 37 Subsystém prstu .....                                      | 22 |
| Obrázek 38 Simscape model – blokové schéma modelu dlaně .....         | 23 |
| Obrázek 39 3D náhled simulačního modelu .....                         | 23 |
| Obrázek 40 Bloky pro vedení lana .....                                | 24 |
| Obrázek 41 Simulink to simscape converter .....                       | 24 |
| Obrázek 42 Simscape to simulink converter .....                       | 24 |
| Obrázek 43 Aktuace kloubu .....                                       | 25 |
| Obrázek 44 Subsystém lanového bubnu .....                             | 26 |
| Obrázek 45 Schéma – Kladka + Belt-Cable End .....                     | 26 |
| Obrázek 46 Prst-vedené lanového pohonu .....                          | 27 |
| Obrázek 47 Simscape – pružina .....                                   | 27 |
| Obrázek 48 Ruka – koule .....   | 28 |
| Obrázek 49 Ruka – krychle .....                                       | 28 |
| Obrázek 50 Ruka – kužel .....   | 28 |
| Obrázek 51 Ramp .....   | 29 |

|  |    |
|--|----|
| Obrázek 52 Spatial contact force.....  | 29 |
| Obrázek 53 Geometrie předmětu .....  | 29 |
| Obrázek 54 Sub systém – spatial contact force.....                             | 30 |
| Obrázek 55 if – switch .....   | 30 |
| Obrázek 56 Genetický algoritmus.....   | 32 |
| Obrázek 57 Vytvoření textového souboru.....                                    | 33 |
| Obrázek 58 Matice kandidátů pro prst 1 - malíček .....                         | 33 |
| Obrázek 59 kód – volba kandidáta .....   | 33 |
| Obrázek 60 kód – výběr nejmenšího času doteku.....                             | 34 |
| Obrázek 61 kód – nalezení příslušného času a vzdálenosti .....                 | 35 |
| Obrázek 62 Penalizace za kolizi .....  | 35 |
| Obrázek 63 Ukládání hodnot z jednotlivých iterací.....                         | 35 |
| Obrázek 64 Cílová funkce koule.....  | 36 |
| Obrázek 65 Cílová funkce krychle.....  | 36 |
| Obrázek 66 Cílová funkce kužel .....   | 37 |
| Obrázek 67 Průběh optimalizace úchopu koule .....                              | 37 |
| Obrázek 68 Úchop koule .....   | 38 |
| Obrázek 69 Vzdálenosti bříšek prstů od geometrie míčku .....                   | 39 |
| Obrázek 70 Průběh optimalizace úchopu komolého kužele .....                    | 39 |
| Obrázek 71 Úchop kužel .....   | 40 |
| Obrázek 72 Vzdálenosti bříšek prstů od geometrie kužele .....                  | 41 |
| Obrázek 73 Průběh optimalizace úchopu krychle .....                            | 42 |
| Obrázek 74 Úchop krychle.....  | 43 |
| Obrázek 75 Vzdálenosti bříšek prstů od geometrie krychle .....                 | 43 |
| Obrázek 76 Blokové schéma mikropočítače [12] .....                             | 44 |
| Obrázek 77 Arduino MEGA 2560 [13] .....  | 45 |
| Obrázek 78 Flex sensor [14] .....  | 46 |
| Obrázek 79 Napěťový dělič [26] .....   | 47 |
| Obrázek 80 DIY sensor .....  | 48 |
| Obrázek 81 Data z vyrobeného senzoru 1.....                                    | 49 |
| Obrázek 82 Data z vyrobeného senzoru 2.....                                    | 49 |
| Obrázek 83 Data z koupeného senzoru.....                                       | 50 |
| Obrázek 84 Rukavice [15] .....   | 50 |
| Obrázek 85 Pájecí pole .....   | 51 |
| Obrázek 86 Schéma zapojení sensorů .....                                       | 52 |
| Obrázek 87 Podstavec pro pájitelné pole .....                                  | 52 |
| Obrázek 88 Řídící rukavice.....  | 53 |
| Obrázek 89 Implementace hardwaru do prostředí simulinku .....                  | 54 |
| Obrázek 90 Použité bloky Arduino/ Simulink .....                               | 54 |
| Obrázek 91 Convert blok .....  | 55 |
| Obrázek 92 Matlab function.....  | 55 |
| Obrázek 93 Map function .....  | 56 |
| Obrázek 94 Celkové uspořádání .....  | 56 |
| Obrázek 95 3-osý gyroskop s akcelerometrem .....                               | 57 |
| Obrázek 96 Gyroskop - Simulink.....  | 57 |
| Obrázek 97 Data gyroskop_2.....  | 58 |
| Obrázek 98 Data gyroskop_2.....  | 58 |
| Obrázek 99 Gyroskop – funkce pro odstranění přebytečného šumu z gyroskopu..... | 59 |
| Obrázek 100 Data gyroskop_3.....   | 59 |
| Obrázek 101 Data z natočení gyroskopu 1 .....                                  | 60 |
| Obrázek 102 Data z natočení gyroskopu 2 .....                                  | 60 |
| Obrázek 103 Označené použité sensory na řídicí rukavici.....                   | 62 |

|   |    |
|---|----|
| Obrázek 104 Ender pro [16] .....                                      | 63 |
| Obrázek 105 CAD model dlaně.....                                      | 64 |
| Obrázek 106 3D print infill [17] .....                                | 65 |
| Obrázek 107 Dlaň v programu Ultimaker Cura.....                       | 65 |
| Obrázek 108 Návrh vazby v modelu .....                                | 66 |
| Obrázek 109 Model článku prstu.....                                   | 67 |
| Obrázek 110 Jednotlivé kanálky pro vedení lan a uložení pružiny ..... | 67 |
| Obrázek 111 Koncový článek prstu.....                                 | 68 |
| Obrázek 112 Vyrobená torzní pružina .....                             | 69 |
| Obrázek 113 Sestava_model_1 .....                                     | 70 |
| Obrázek 114 Sestava_model_2 .....                                     | 70 |
| Obrázek 115 Servopohon [19].....                                      | 71 |
| Obrázek 116 Velikosti střídý [20].....                                | 72 |
| Obrázek 117 Řemenice/ Motor .....                                     | 72 |
| Obrázek 118 Spojovací prvek .....                                     | 73 |
| Obrázek 119 Sestavený model – pohled 1 .....                          | 74 |
| Obrázek 120 Sestavený model – pohled 2 .....                          | 75 |
| Obrázek 121 Vytisknutí model_pohled_1 .....                           | 76 |
| Obrázek 122 Vytisknutý model_pohled_2 .....                           | 77 |
| Obrázek 123 Elektrické zapojení.....                                  | 78 |
| Obrázek 124 Umístění Arduina na modelu.....                           | 79 |

## SEZNAM TABULEK

|   |    |
|---|----|
| Tabulka 1 Rozměry dlaně .....                                     | 18 |
| Tabulka 2 Průměry článků .....                                    | 19 |
| Tabulka 3 Délky druhých a třetích článků.....                     | 19 |
| Tabulka 4 Délky 1. článků .....                                   | 19 |
| Tabulka 5 Optimalizační parametry.....                            | 31 |
| Tabulka 6 Optimalizované parametry pro úchop koule .....          | 38 |
| Tabulka 7 Optimalizované parametry pro úchop komolého kužele..... | 40 |
| Tabulka 8 Optimalizované parametry pro úchop krychle .....        | 42 |
| Tabulka 9 Parametry desky Arduino MEGA 2560.....                  | 45 |
| Tabulka 10 Rozměry dlaně reálného modelu .....                    | 64 |
| Tabulka 11 Průměrné hodnoty torzních pružin .....                 | 69 |
| Tabulka 12 Parametry servopohonu [19] .....                       | 71 |

## Použitá literatura

- [1] *Průmyslové roboty a manipulátory* [online]. Praha: ČVUT, 2012 [cit. 2022-08-15]. Dostupné z: Moodle. Skripta. ČVUT.
- [2] Magnetics. In: *Www.magnetics.com* [online]. Boyne City, USA: Industrial Magnetics [cit. 2022-07-29]. Dostupné z: <https://www.magnetics.com/product.asp?ProductID=222>
- [3] Vakuová přísavka. In: *Smc* [online]. Brno [cit. 2022-07-29]. Dostupné z: <https://www.smc.eu/cs-cz/produkty-a-podpora/skupiny-prvku/prisavky-pro-potravinarsky-prumysl>

- [4] RG2 Gripper. In: *Universal-robots* [online]. Dánsko, 2005 [cit. 2022-07-29]. Dostupné z: <https://www.universal-robots.com/plus/products/onrobot/rg2-gripper/>
- [5] Pneumatic gripper MPZ. In: *Directindustry* [online]. [cit. 2022-07-29]. Dostupné z: <https://www.directindustry.com/prod/schunk-gmbh-co-kg/product-7038-1841215.html>
- [6] 3-Finger Adaptive Robot Gripper. In: *Wevolver* [online]. [cit. 2022-07-29]. Dostupné z: <https://www.wevolver.com/specs/3-finger.adaptive.robot.gripper>
- [7] Ruční dynamometrie u účastníků mistrovství ČR v boxu v roce 2010. In: <https://www.fsps.muni.cz/> [online]. Bohunice: Fakulta sportovních studií MU [cit. 2022-07-29]. Dostupné z: <https://www.fsps.muni.cz/emuni/data/reader/book-10/06.html>
- [8] Building a Super Robust Robot Hand. In: *IEEE Spectrum* [online]. New York, USA: Institute of Electrical and Electronics Engineers, 2011 [cit. 2022-07-29]. Dostupné z: <https://spectrum.ieee.org/dlr-super-robust-robot-hand>
- [9] This Is the Most Amazing Biomimetic Anthropomorphic Robot Hand We've Ever Seen. In: *IEEE Spectrum* [online]. New York, USA: Institute of Electrical and Electronics Engineers, 2016 [cit. 2022-07-29]. Dostupné z: <https://spectrum.ieee.org/biomimetic-anthropomorphic-robot-hand>
- [10] MATLAB. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001 [cit. 2022-08-04]. Dostupné z: <https://cs.wikipedia.org/wiki/MATLAB>
- [11] Simscape. In: *Mathworks* [online]. USA [cit. 2022-08-04]. Dostupné z: <https://www.mathworks.com/products/simscape.html>
- [12] UHLÍŘ DRSC., Prof. Ing. Ivan. *ELEKTRICKÉ OBVODY A ELEKTRONIKA*. První. Praha: ČVUT, 2002. ISBN 978-80-01-03932-8.
- [13] ARDUINO MEGA 2560 MICROCONTROLLER REV 3 - BOXED ORIGINAL. In: *Robotpark* [online]. [cit. 2022-08-04]. Dostupné z: <https://www.robotpark.com/Arduino-MEGA-2560-En>
- [14] Tenzometr - senzor ohýbání. In: *Laskakit* [online]. [cit. 2022-08-04]. Dostupné z: [https://www.laskakit.cz/tenzometr-senzor-ohybani--2-2-/?gclid=CjwKCAjw3K2XBhAzEiwAmmgrAvmAWQLhDnewcIKUcHFEu5jZrPdGl6LXsGP6slj4V\\_FtWBNT-jnkqhoCOF8QAvD\\_BwE](https://www.laskakit.cz/tenzometr-senzor-ohybani--2-2-/?gclid=CjwKCAjw3K2XBhAzEiwAmmgrAvmAWQLhDnewcIKUcHFEu5jZrPdGl6LXsGP6slj4V_FtWBNT-jnkqhoCOF8QAvD_BwE)
- [15] Rukavice povrstvené Ardon PETRAX latex. In: *Inzep* [online]. [cit. 2022-08-04]. Dostupné z: [https://www.inzep.cz/rukavice-povrstvene-ardon-petrax-latex-reflexni-zluta-8-2?gclid=CjwKCAjw3K2XBhAzEiwAmmgrAha4xioFqy3tlqxovpBN8fSSF3PChiabmZUxwFemP8rjzBOU4I7i5RoCoesQAvD\\_BwE](https://www.inzep.cz/rukavice-povrstvene-ardon-petrax-latex-reflexni-zluta-8-2?gclid=CjwKCAjw3K2XBhAzEiwAmmgrAha4xioFqy3tlqxovpBN8fSSF3PChiabmZUxwFemP8rjzBOU4I7i5RoCoesQAvD_BwE)
- [16] Creality ENDER 3 PRO. In: *Alza* [online]. Česká Republika [cit. 2022-08-10]. Dostupné z: [https://www.alza.cz/creality-ender-3-pro-d5522742.htm?kampan=adw4\\_Alza\\_pla\\_all\\_pricekillers\\_pricekillers\\_c\\_9062893\\_\\_ENDER02\\_591993626872\\_~135778951096~&gclid=Cj0KCQjwrs2XBhDJARIsAHVymmRbP3qeEK1ImaqZ6rkXyvxgt4baIElaYTS5gB4Zxw8nYQANtELldwUaAuTKEALw\\_wcB](https://www.alza.cz/creality-ender-3-pro-d5522742.htm?kampan=adw4_Alza_pla_all_pricekillers_pricekillers_c_9062893__ENDER02_591993626872_~135778951096~&gclid=Cj0KCQjwrs2XBhDJARIsAHVymmRbP3qeEK1ImaqZ6rkXyvxgt4baIElaYTS5gB4Zxw8nYQANtELldwUaAuTKEALw_wcB)

- [17] 3D Printing Infill: The Basics Simply Explained. In: *All3dp* [online]. [cit. 2022-08-11].  
Dostupné z: <https://all3dp.com/2/infill-3d-printing-what-it-means-and-how-to-use-it/>
- [18] Calculate-spring-rate-torsion-spring. In: *Webshop.tevema.com* [online]. [cit. 2022-08-09].  
Dostupné z: <https://webshop.tevema.com/gb/blog/calculate-spring-rate-torsion-spring/>
- [19] Servo MG995 s kovovými převody 180°. In: *Laskakit* [online]. [cit. 2022-08-09]. Dostupné z:  
<https://www.laskakit.cz/servo-mg995-s-kovovymi-prevody-180---extra-silne/>
- [20] Pulse Width Modulation. In: *Https://learn.sparkfun.com/* [online]. [cit. 2022-08-09].  
Dostupné z: <https://learn.sparkfun.com/tutorials/pulse-width-modulation/duty-cycle>
- [21] Research.tuni.fi. In: *Sticky soft robots* [online]. Finland: Tampere University [cit. 2022-07-29]. Dostupné z: <https://research.tuni.fi/bmr-group/projects/sticky-soft-robots/>
- [22] How marine biology inspired Soft robotics industrial grippers. In: *Techcrunch* [online]. San Francisko, California, USA: Brian Heater, Lora Kolodny, 2017 [cit. 2022-07-29]. Dostupné z: <https://techcrunch.com/2017/04/01/soft-robotics-grippers/>
- [23] A New Robotic Hand. In: *Web.archive* [online]. Sarah Gingichashvili, 2007 [cit. 2022-07-29].  
Dostupné z:  
<https://web.archive.org/web/20120704120412/http://thefutureofthings.com/pod/269/a-new-robotic-hand.html>
- [24] Arm and forearm. In: *Inmoov* [online]. France: Gaël Langevin, 2012 [cit. 2022-07-29].  
Dostupné z: <https://inmoov.fr/>
- [25] *Bones of the human hand* [online]. In: . [cit. 2022-07-29]. Dostupné z:  
[https://en.wikipedia.org/wiki/Hand#/media/File:Scheme\\_human\\_hand\\_bones-en.svg](https://en.wikipedia.org/wiki/Hand#/media/File:Scheme_human_hand_bones-en.svg)
- [26] Interfacing Flex Sensor with Arduino. In: *Lastminuteengineers* [online]. [cit. 2022-08-14].  
Dostupné z: <https://lastminuteengineers.com/flex-sensor-arduino-tutorial/>