



Středoškolská technika 2025

Setkání a prezentace prací středoškolských studentů na ČVUT

SCARA robot

Bruno Karli

**Střední škola EDUCHEM
Okružní 128, 435 13 Meziboří**

Anotace

Tento projekt se zaměřuje na vývoj a konstrukci SCARA robota určeného ke kreslení. Robot byl vytištěn na 3D tiskárně a využívá krokové motory k přesnému řízení pohybu. Řídicí jednotkou je mikrokontroler ESP32, který zajišťuje zpracování instrukcí a ovládání jednotlivých os. Software umožňuje interpretaci zadaných souřadnic a přesný pohyb ramene robota v definovaném pracovním prostoru. Projekt je zaměřen na efektivní řízení pohybu a optimalizaci přesnosti kreslení s využitím dostupných komponent a moderních metod řízení.

Obsah

1	Úvod a cíle práce	4
2	Představa o realizaci projektu	5
3	Realizace projektu	6
3.1	Realizace obvodu pro robota	
4	Použitá elektronika	7
4.1	Popis použité elektroniky/součástek	8
5	Tvorba obvodu	13
6	Tvorba Plošného spoje	14
7	Použité součástky	16
8	Popis navržených dílů	17
9	Programování	36
10	Závěrečné úpravy	37
11	Představa o budoucím vylepšení	38
12	Seznam použité literatury	39

1 Úvod a cíle práce

-Pro svou závěrečnou práci jsem se byl rozhodnutý vytvořit na mé poměry poměrně složitý projekt, ze kterého vzešel SCARA Robot.

Mým cílem bylo zrealizovat robota, který bude schopen kreslit pomocí fixy, nějaké obrazce. V mé práci jsem pracoval hned v několika prostředích, která pro mě byla zcela nová, například:

- Programování C++

- Modelovací program Onshape

- Multisim

- EasyEDA

- Návrhy mechanických součástí (ozubená kolečka, převodovky)

2 Představa o realizaci projektu

Mé představy o tomto projektu byly ze začátku dosti naivní a přehnané. Ze začátku jsem měl v plánu vytvořit šesti osého robota. Nicméně jsem z tohoto nápadu jsem zhruba po měsíci upustil, protože bych na to neměl čas a také by to pro mě bylo extrémně náročné. Následně jsem se tedy rozhodl udělat jednodušší variantu, a tou je SCARA Robot, neboli v mém případě dvou ramenný robot, který je schopen kreslit. Také jsem mým cílem bylo aby můj program byl daleko rozvinutější a zvládal počítat i s jinými tvary obrazců, které by následně robot kreslil. Bohužel i přes mé zcela intenzivní snažení, které mi zabralo zhruba sedm měsíců jsem nebyl schopen docílit mého plnohodnotného plánu pro tento projekt.

3 Realizace projektu

S realizací mého projektu jsem začal již zhruba v polovině letních prázdnin, a nadále jsem pokračoval v novém předmětu konstrukčních úloh. Začátkem samotného projektu nebylo přímo navrhování určitých součástek v CAD programu Onshape, nýbrž zorientování v tomto programu, kde jsem následně navrhoval všechny moje součástky. Také jsem sbíral znalosti v oblasti programování v jazyce Wiring, což je upravený jazyk C++ pro IOT vývojové desky. Z počátku jsem využíval IOT desku Arduino UNO, nicméně se později ukázalo, že samotné Arduino UNO nestačí. Rozhodl jsem se použít IOT desku ESP32 dev module, která má na rozdíl od 8 bitového Arduina 32 bitový procesor a spousty dalších funkcí.

3.1 Realizace obvodu pro robota

Při návrhu samotného obvodu jsem z počátku používal již zmiňované Arduino, mi přestalo v průběhu vývoje projektu stačit. Má prvotní úvaha byla taková, že z Arduina budu vysílat pouze signály do tranzistorů, které následně budou otvírat jednotlivé propojení.

Nicméně po zakoupení ESP32 a Laboratorního zdroje jsem se rozhodl napřímo ovládat vstupy/výstupy celého obvodu pomocí již zmíněného ESP32. Hlavní důvod tohoto rozhodnutí pramenil z toho, že jsem lehce upravil obvod robota a také z toho, že ESP32 má větší možnost proudového odběru.

4 Použitá elektronika

V mém obvodu pro robota jsem použil poměrně jednoduché součástky, kterými jsou:

-Krokový motor: Nemá 17 – 33 mm / 2x

-Stepper Drivers: DRV8825 / 2x

-Servo motor: SG90 – 9 g

-Koncový spínač: End Stop Creality / 2x

-IOT vývojová deska: ESP32 dev module

-Kondenzátor: 100 μ F

Pro dočasné uspořádání jsem použil tzv. Breadboard, kterou jsem následně nahradil plošným spojem.
(viz. **5 Tvorba obvodu**)

4.1 Popis elektroniky

Krokový motor: Nema 17 – 33mm



Tento mnou zvolený krokový motor využívá napětí 12 -24 V 1-1.5A

Krokový motor: Nema 17 – 33mm je velice oblíbenou variantou krokových motorů zejména v 3D tiskárnách.

Krokový motor funguje na principu elektromagnetické indukce a je řízen přesnými elektrickými pulzy, které určují jeho pohyb. Jeho základní principy jsou:

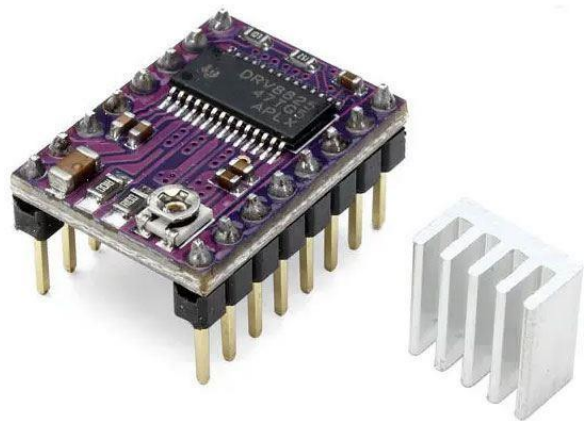
1. Rotor a stator:

- Rotor (točivá část) je obvykle tvořen permanentními magnety nebo měkkou magnetickou ocelí.
- Stator (pevná část) obsahuje vinuté cívky (fáze), které vytvářejí magnetická pole.

2. Krokový pohyb:

- Ovladač motoru (driver) přepíná napětí do jednotlivých cívek ve správném pořadí, čímž vytváří magnetické pole, které přitahuje rotor do určité polohy.
- Každý pulz posune rotor o jeden pevně daný krok (např. $1,8^\circ$ u standardních NEMA 17, což znamená 200 kroků na jednu otáčku).

Stepper Driver: DRV8825



Pro můj projekt jsem vybral hodně oblíbený typ driverů, líbila se mi možnost využití 1/32 nastavení mikrokrokování což v praxi znamená, že počet kroků na jednu otáčku je 32 násobkem základní hodnoty motoru. Tyto drivery jsou v obvodu napájené 20V a je potřeba nastavit omezovač proudu. Tento obvod funguje na přepínání výstupního signálu, který řídí elektromagnetické pole cívky krokového motoru. Jelikož tento obvod pracuje poměrně rychle s frekvencí 32kHz, což znamená, že driver musí mít alespoň vlastní pasivní chladič.

Servomotor: SG90 – 9g



Tuto verzi servomotoru jsem vybral z jediných důvodů, kterými jsou zejména ten, že jsem ho měl k dispozici doma, a dalším skvělým důvodem je cena. Také bychom se mohli bavit o váze, která činí pouze 9 gramů, což se zcela hodí na konec robotické ruky. Dále je tento malý a levný servo motor schopen vyvinout obrovský točivý moment 1.5kg/cm. Tomuto servomotoru stačí pouhé napětí 3.3V což ho činí velice efektivním v řešení mého obvodu, protože ho lze napájet přímo z ESP32.

Jak funguje:

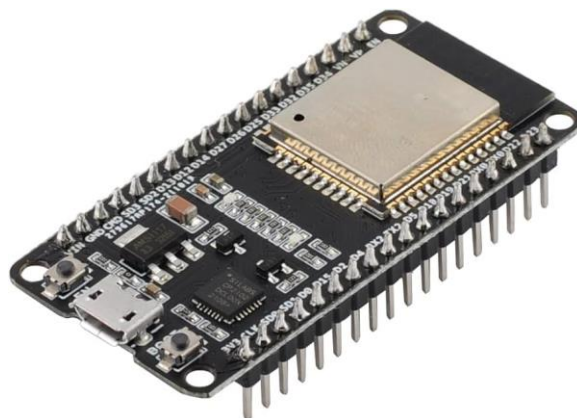
Servomotor funguje na principu zpětné vazby. Obsahuje elektromotor, řídicí elektroniku a snímač polohy (např. potenciometr). Řídicí signál (PWM) určuje požadovanou polohu hřídele, elektronika porovnává aktuální polohu se žádoucí a upravuje otáčky motoru tak, aby dosáhl cílové polohy.

Koncový spínač: End Stop Creality



Při výběru koncových spínačů jsem sáhl po End Stop Creality, protože jsem je měl doma z rozbité staré tiskárny, a jejich použití je zcela primitivní.

IOT vývojová deska: ESP32 dev module



Po zjištění že Arduino UNO není schopno řešit komplexnější operace a zjištění že je pomalé jsem zde vybral jasnou volbu pro tento projekt co se týče řídicí jednotky. 32 bitový systém, výrazné rozšíření množství použitelných digitálních pinů. Možnostmi využití internetu a Bluetooth, větším proudovým odběrem a na své poměry velmi levnou cenou je tato IOT vývojová deska ESP32 velice příjemná volba.

5 Tvorba obvodu

Při tvorbě obvodu jsem ze začátku bojoval s nedostatkem proudu z IOT desky Arduino UNO (200 mA) jsem zamýšlel použití tranzistorů pro otvírání cest k jednotlivým součástkám, nicméně jsem se rozhodl použít ESP32 se špičkovým proudovým odběrem až 1A což je více než dostatečné.

Drivery:

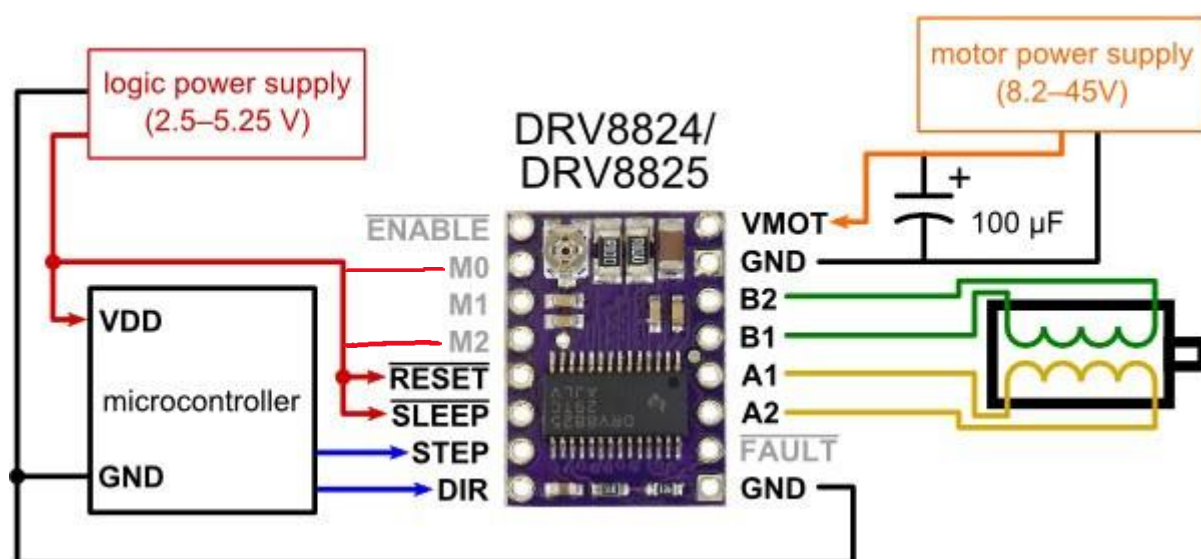
Ze začátku jsem byl otevřen vytvoření obvodu, přes který by uživatel mohl měnit mikrokrokování na datových vstupech driverů DRV8825 M0, M1, M2, ale vzhledem k nutnému přidání šesti dalších datových pinů jsem se rozhodl tak neučinit, hlavně díky poměrně náročnému návrhu plošného spoje.

Většina driverů má možnost využívat funkci mikrokrokování, zde například využívám 1/32 Step což v praxi znamená že když krokový motor má 200 kroků na 360° neboli na jednu celou otáčku znamená to, že výsledná hodnota kroků na 360° bude činit $32 \times 200 = 6400$ kroků. Tuto hodnotu budu nadále zvětšovat skrze převodovky.

Tabulka kombinací pro Mikrokroky:

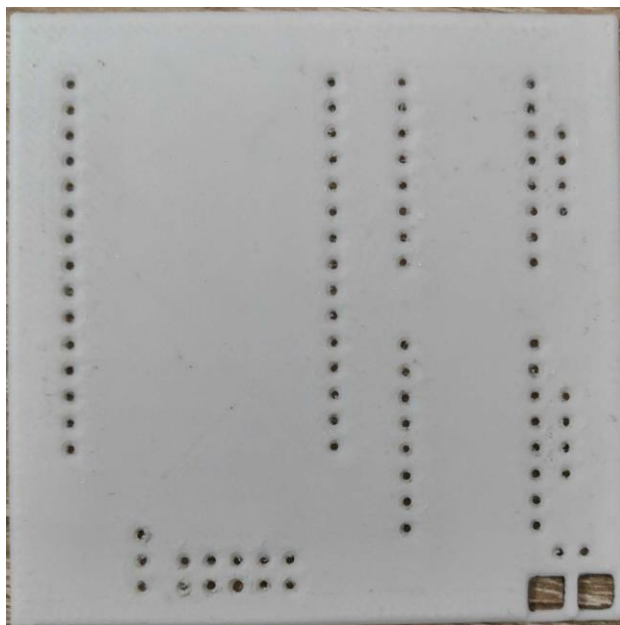
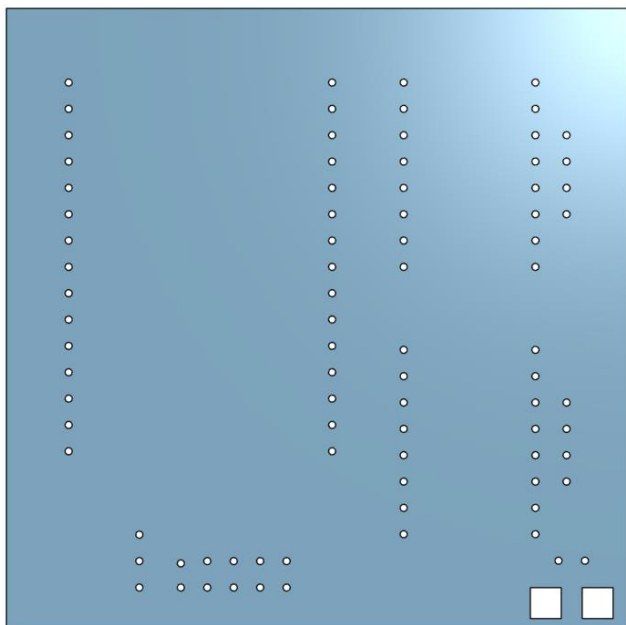
M0	M1	M2	Mikrokroky
0	0	0	Full Step
1	0	0	1/2 Step
0	1	0	1/4 Step
1	1	0	1/8 Step
0	0	1	1/16 Step
1	0	1	1/32 Step

Schématické zapojení driveru

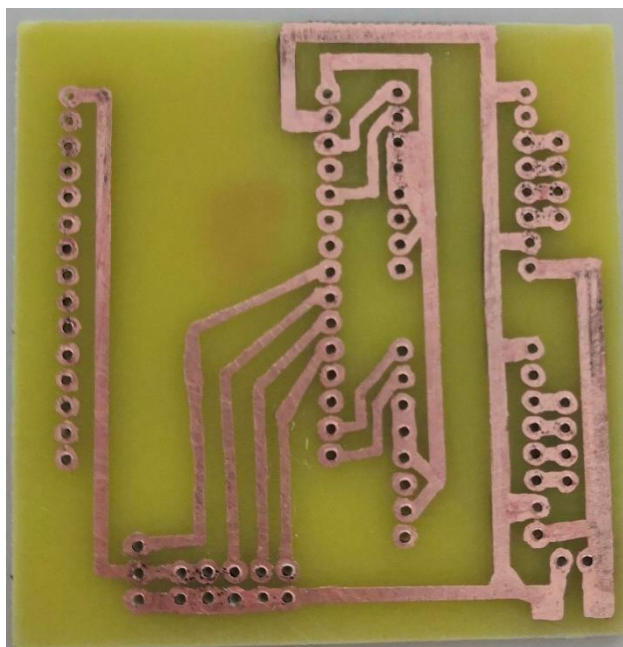


6 Tvora plošného spoje

Pro lepší optimalizaci a funkčnost jsem se rozhodl začlenit do tohoto projektu vlastní plošný spoj (PCB). Navrhl jsem jej pro kompatibilitu s patnicemi pro následné zasunutí driverů a ESP32 s možností jednoduché výměny v případě nutnosti. Pro větší přesnost mě napadlo využít 3D tiskárnu a udělat si šablonu s přesně odměřenými díry a následné přilepení této destičky na destičku plošného spoje pomocí elektrikářské pásky.



Po následném vyvrtání potřebných děr jsem nakreslil spoje fixou a dal jsem destičku leptat.



Po vyleptání destičky jsem ji osadil male to female connectorem, který byl použit pro přímé propojení s plošným spojem a následné inserci ESP32 pro jednodušší aplikaci.



7 Použité součástky

-Měděná podložka:

-8x12x1 (2)

-10x16x1 (7)

-Železné trubičky (osy):

-8x32,7

-10x75,9

-10x22,5

-10x32 (2)

-10x45,7

-12x27,5

-12x19,5 (3)

-12x34

-Šroub

-DIN 7985 M3x10 (38)

-DIN 7985 M3x13 (8)

-DIN 965 M3x40 (8)

-DIN 7985 M5x30 (4)

-DIN 912 M5x40 (4)

-DIN 965 M3x16 (2)

-DIN 912 M3x12 (3)

-Matka

-DIN 934 M3 (59)

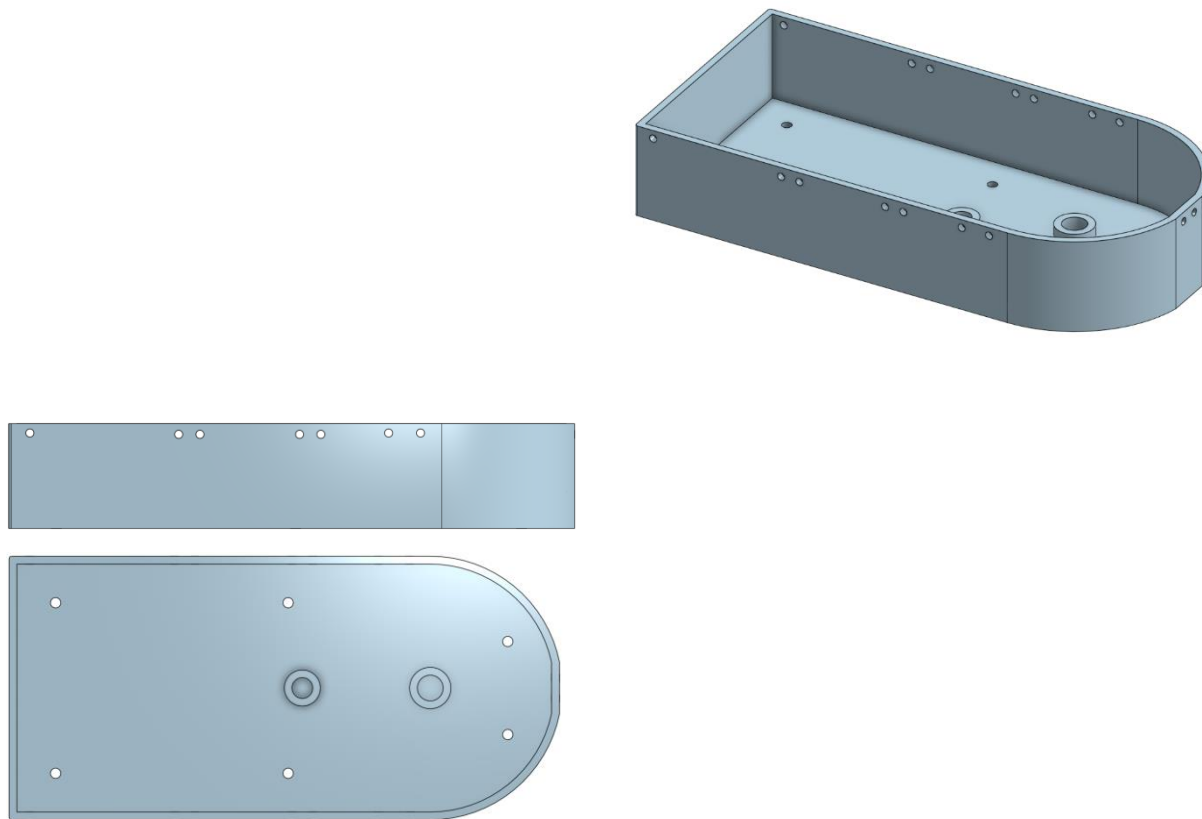
-DIN 934 M5 (8)

-Ložiska

-623: 10x3x4 (12)

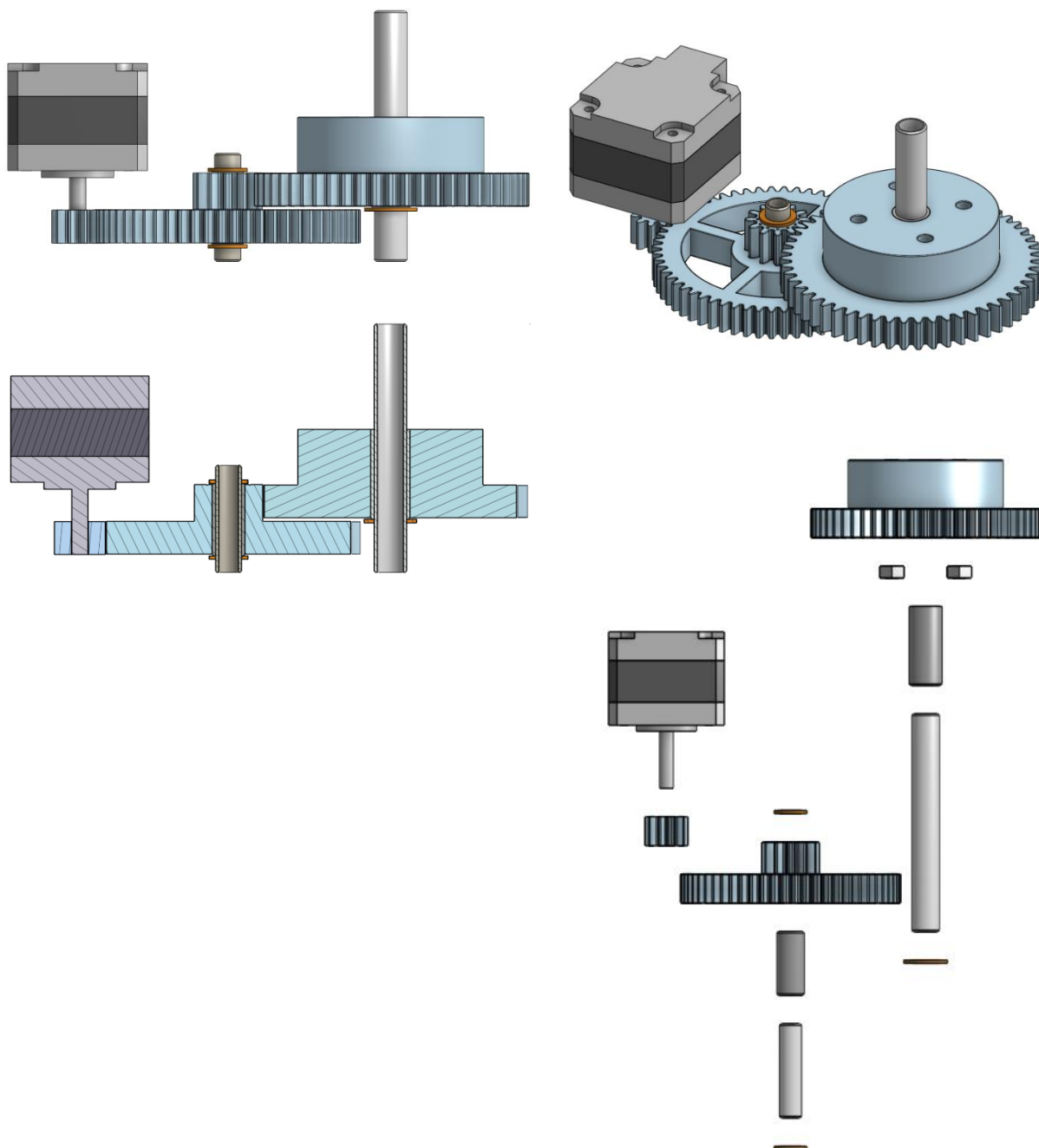
8 Popis navržených dílů

Díl 1 - Podstava



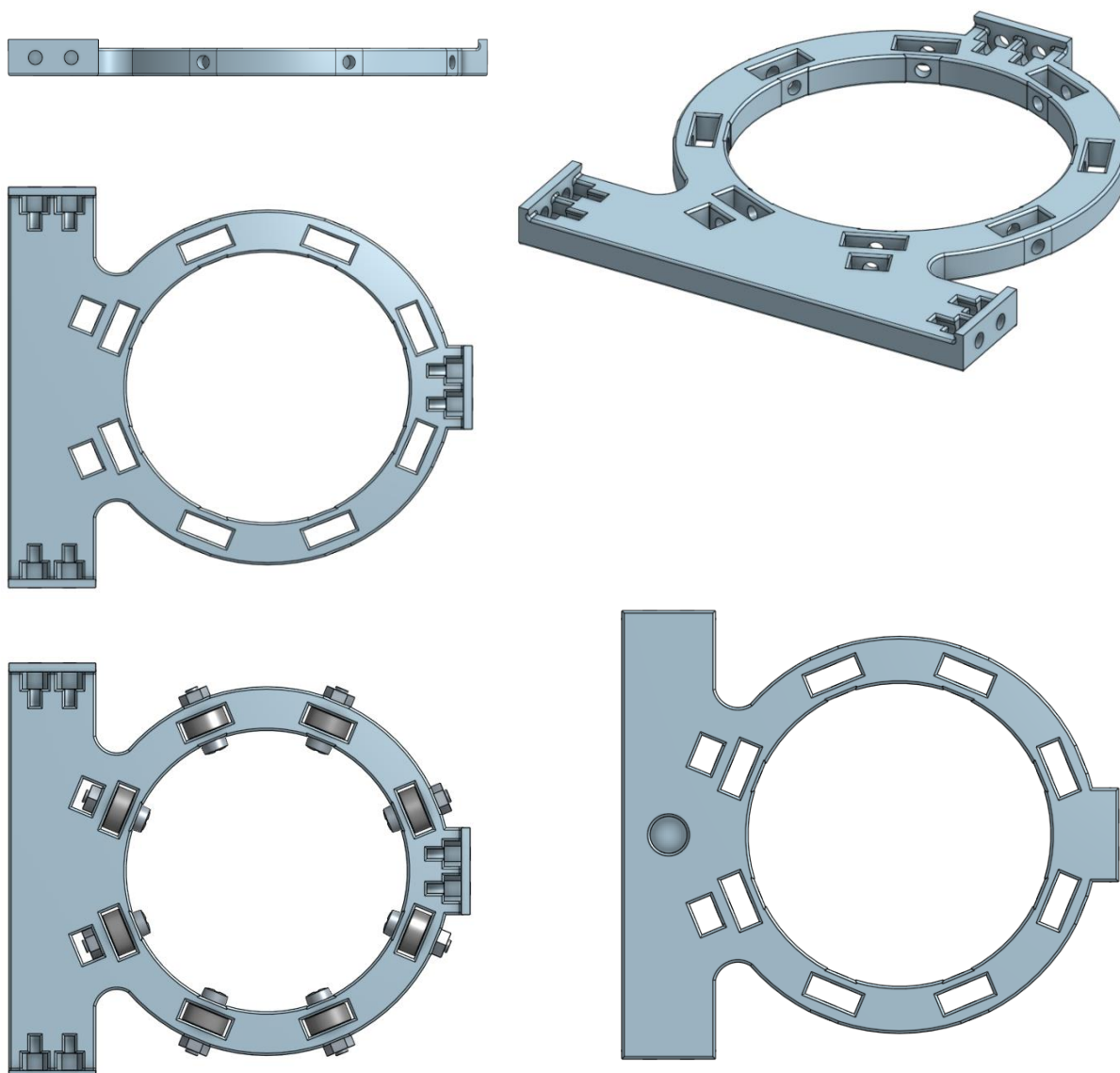
Podstava je jedna z nejdůležitějších součástí celé konstrukce, drží celou váhu a musí být dostatečně široká a dlouhá, aby se v žádném úhlu robot nemohl překlopit. Pro případy, kde by bylo potřeba větší zátěže na konci ramene jsem myslel také na možnost přišroubování tohoto dílu. Při návrhu bylo nutné myslet také na zakomponování a fixaci os pro převodovku, to jsou ty „vystouplé díry“.

Díl 2 – Převodovka 1



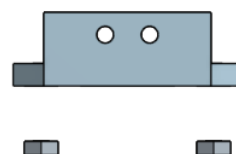
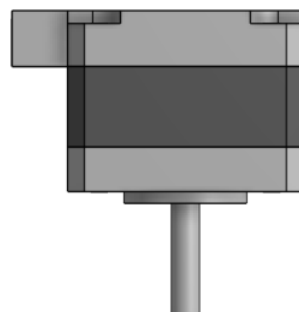
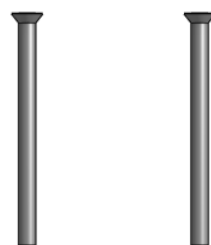
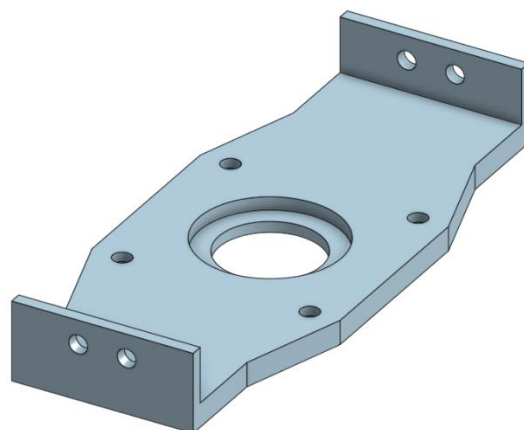
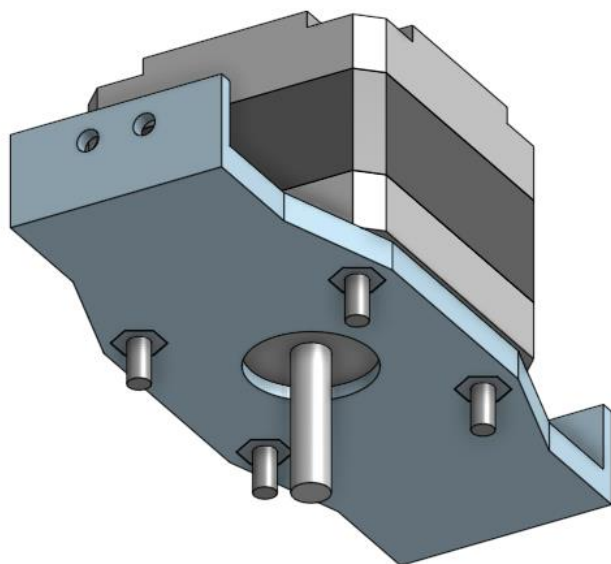
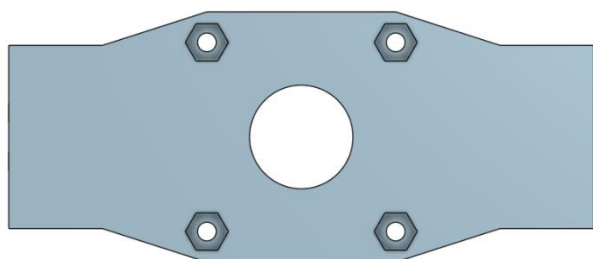
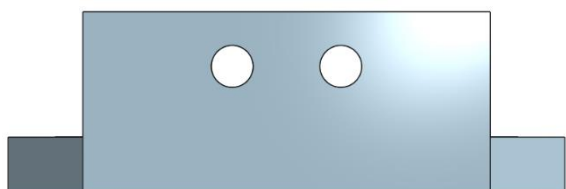
Převodovka je nejdůležitějším mechanickým subjektem v celé konstrukci a pro správnou optimalizaci pohybu robota. Tato převodovka pro první kloub je navržena pro převod 1:27 tedy pro otočení o jeden stupeň je nutno vykonat 480 kroků. Tato převodovka je uchycena do podstavy. Za pomoci nasunutí do děr pro trubičky.

Díl 3 – Ring 1

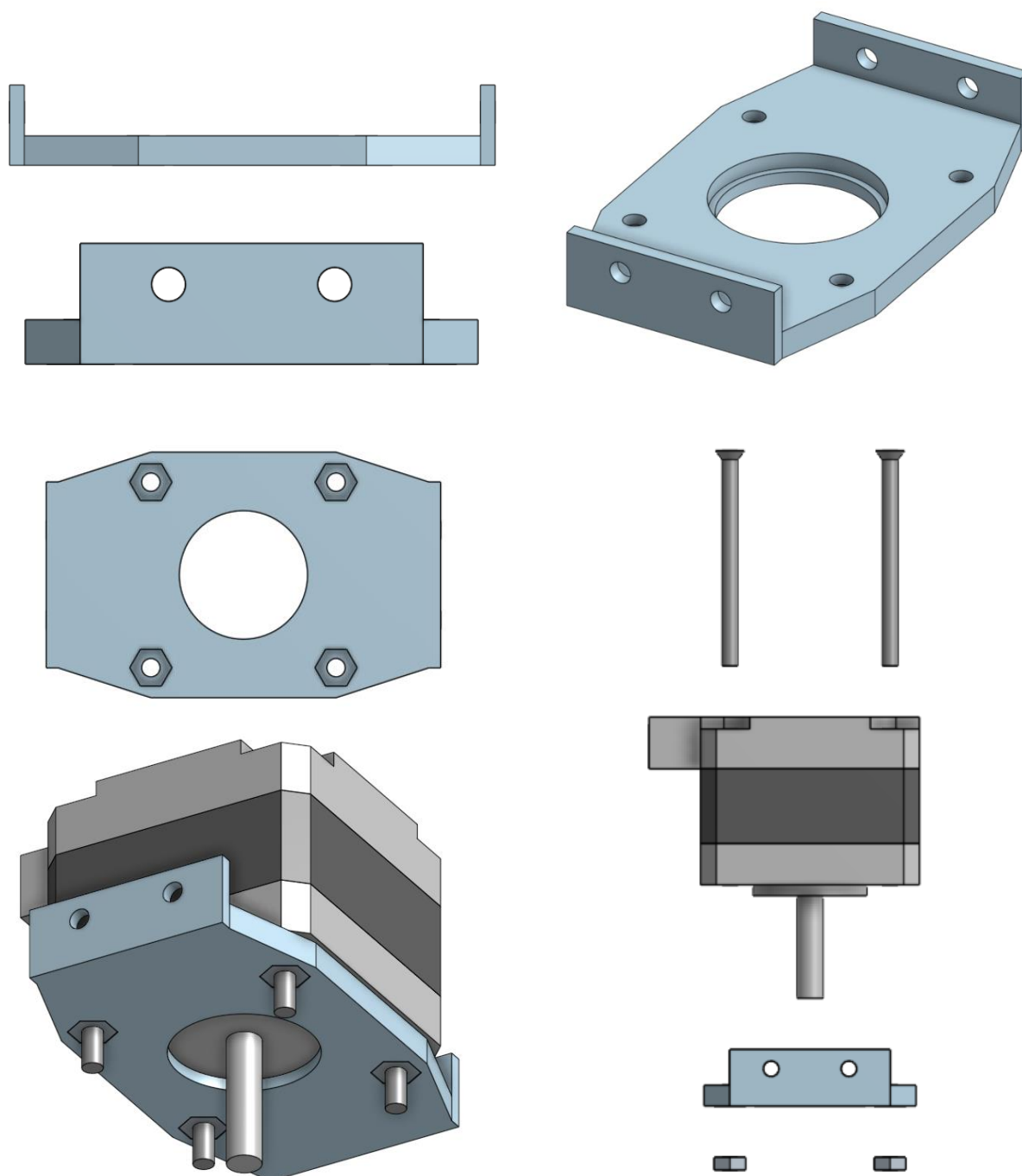


Tato součástka slouží k upevnění výstupního ozubeného kolečka z první převodovky. Pomocí dvanácti ložisek jsem zajistil dostatečnou mobilitu a stabilitu pro pohyb prvního kloubu. Tato součástka vlastně slouží jako velké ložisko, nicméně ložisko takového rozměru je velmi drahé a pro účely projektu postačí takováto aparatura.

Díl 4 – Držák motoru 1

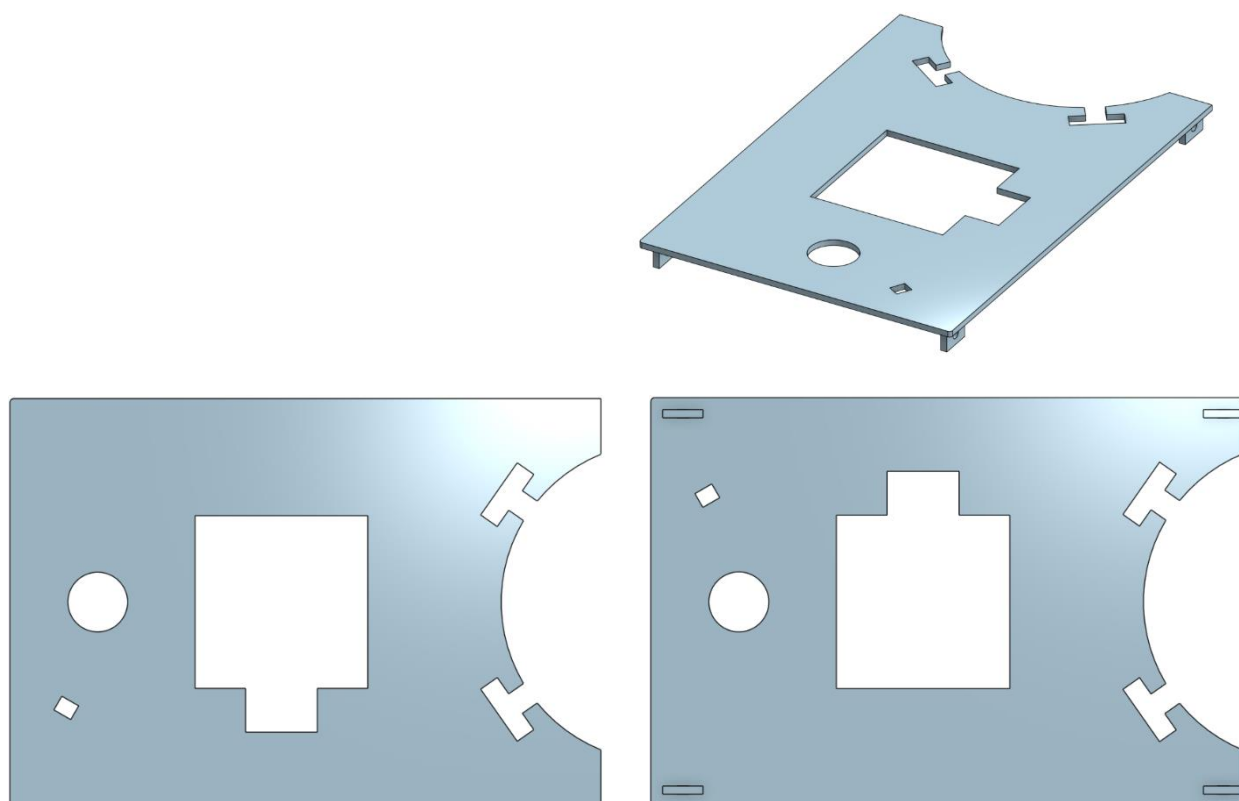


Díl 5 – Držák motoru 2

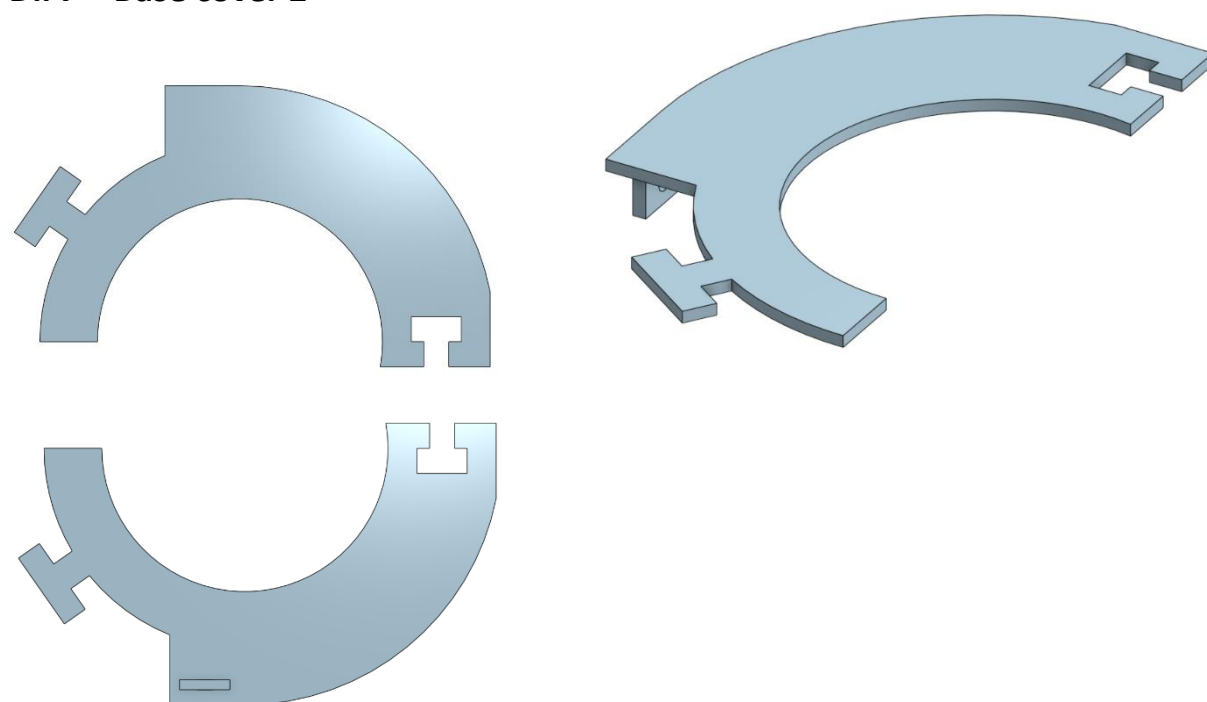


Tato součástka (**Díl 4** a **Díl 5**) slouží k uchycení krokového motoru na konstrukci. Součástka by měla být dostatečně robustní, tak aby vydržela případnou torzy vyvinutou odporem převodovky. Nicméně pro účely tohoto projektu není třeba navrhovat extra robustní konstrukci. Skrze otvor na středu je možné přímo prostrčit první ozubené kolečko na ose motoru bez nutné demontáže. Motor je připevněn pomocí čtyř šroubů M3x40, které jsou našroubovány skrze konstrukci motoru a následně do matky, která se nachází v předem navrhnutých otvorech na součástce.

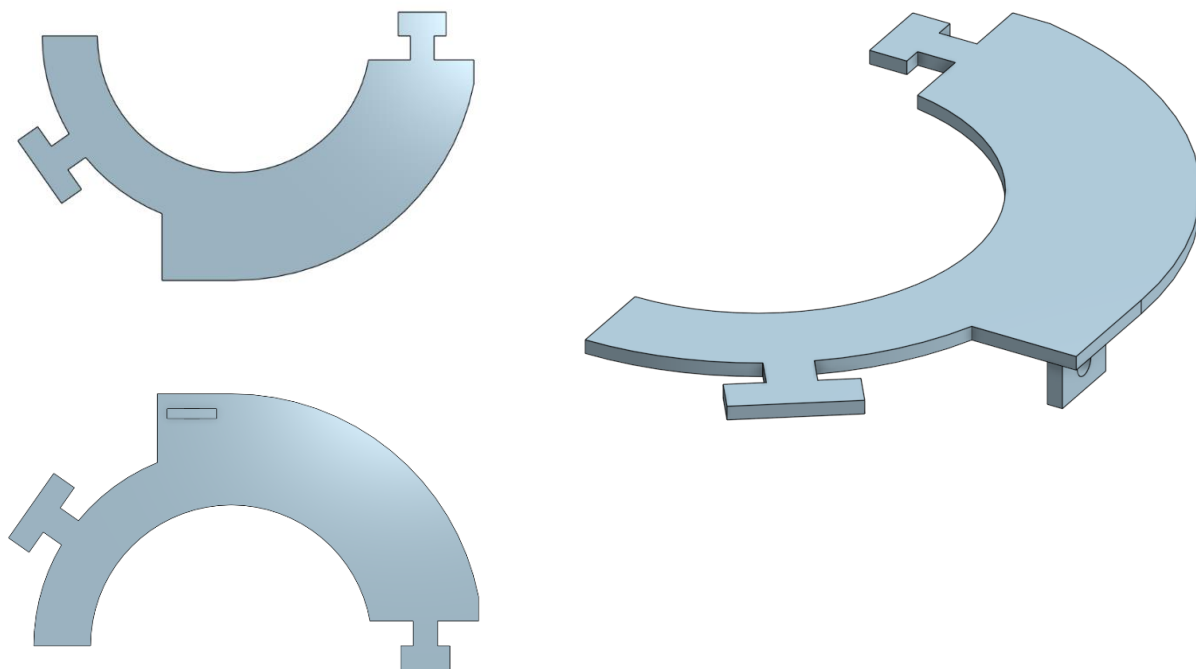
Díl 6 – Base cover 1



Díl 7 – Base cover 2

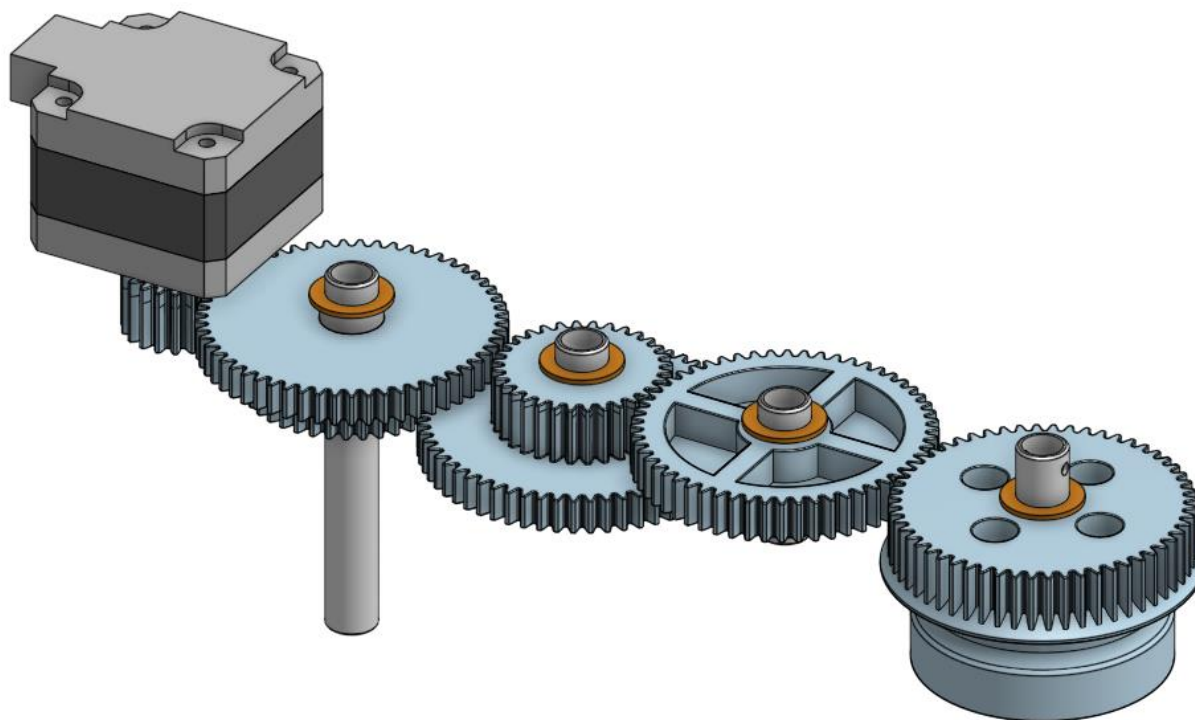


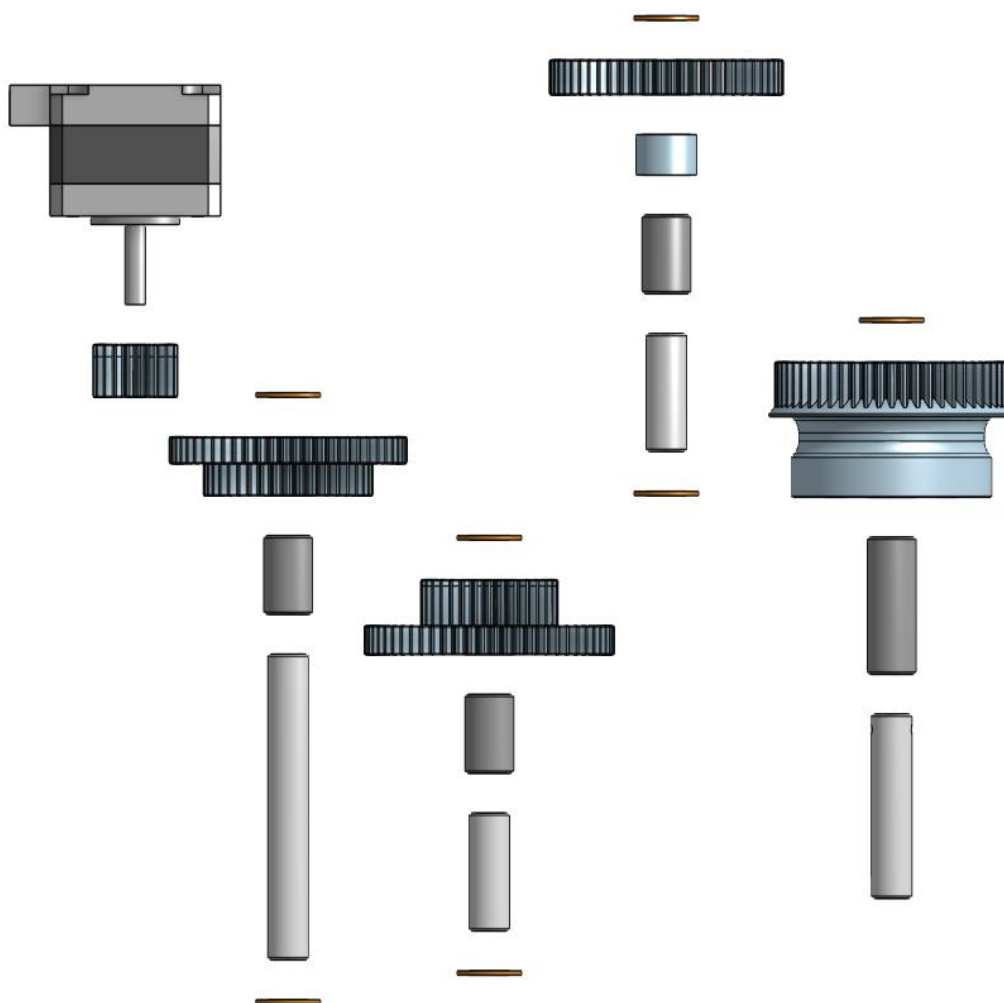
Díl 8 – Base cover 3



Díly (6, 7, 8) jsou určeny pro případné vniknutí předmětů do převodovky a také pro větší estetiku konstrukce robota. V průběhu navrhování dalších dílů pro robota jsem bohužel musel odstranit **Díl 6**, kvůli pozdějším součástkám a celkové přístupnosti zejména při montáži jednotlivých dílů. Tyto tři díly se namontují pomocí šroubů M3x10 přímo do podstavy.

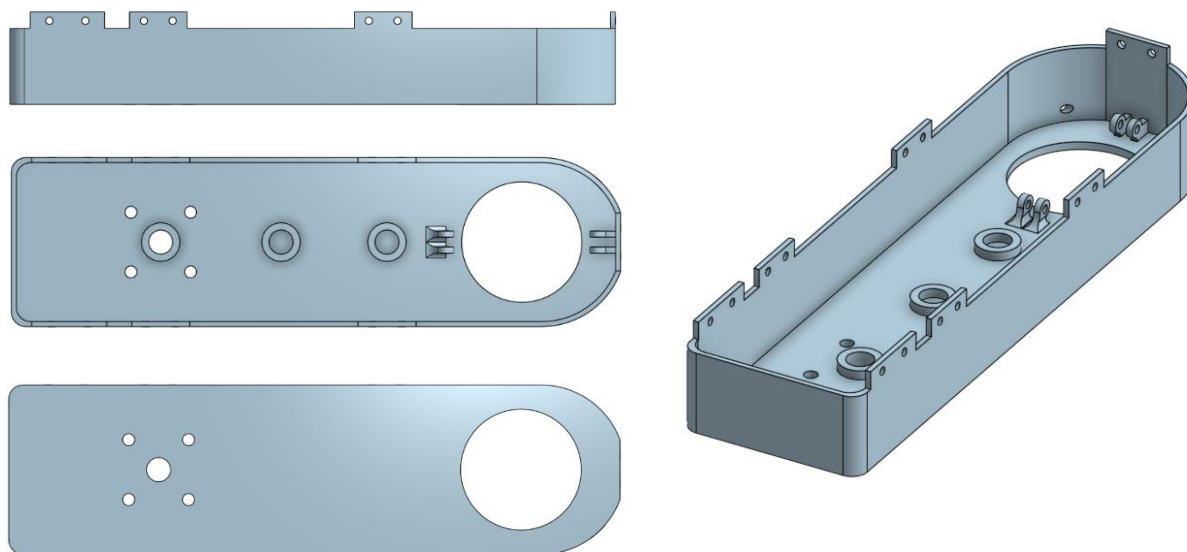
Díl 9 – Převodovka 2





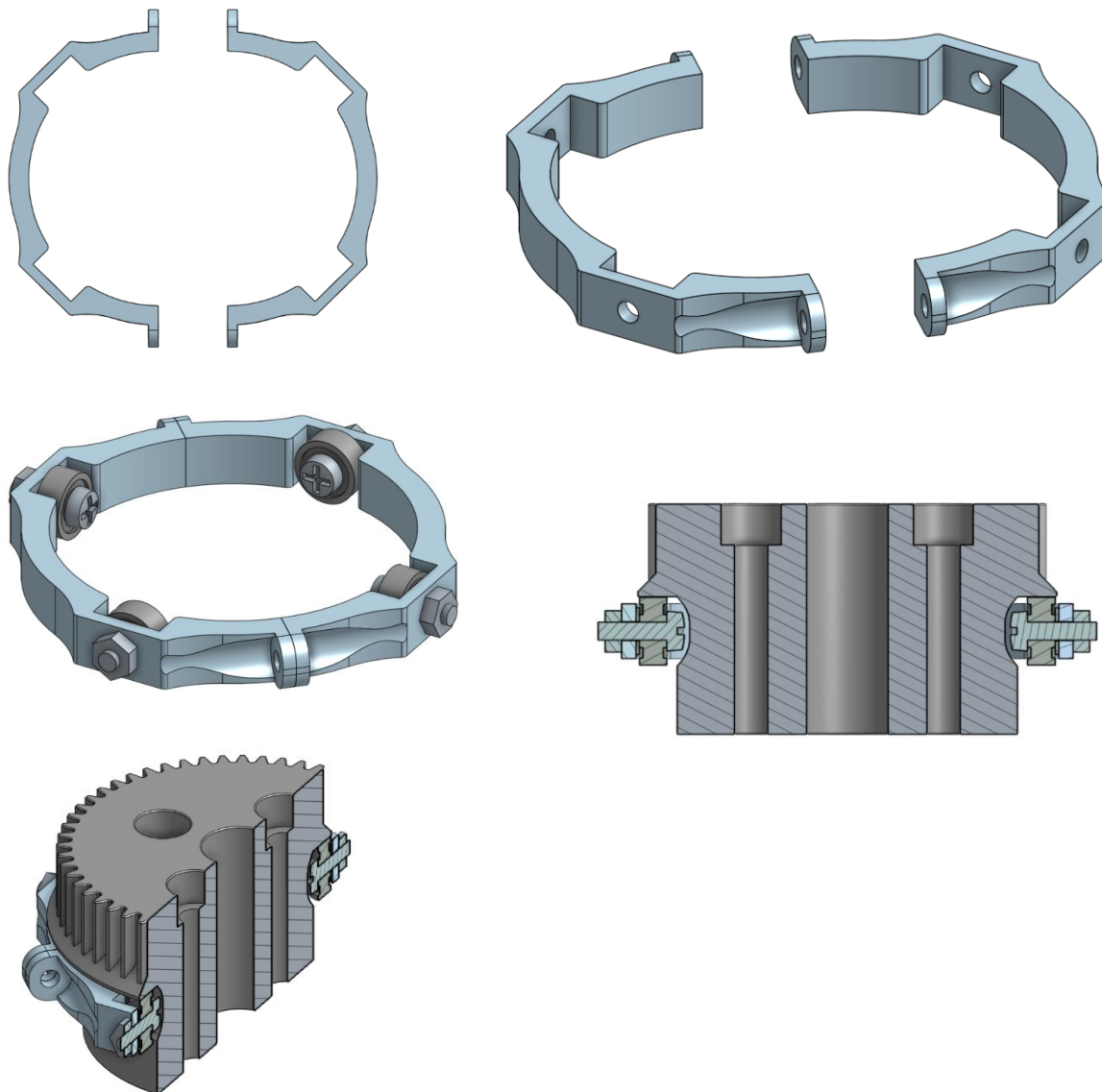
Při návrhu převodovky 2 jsem musel zakomponovat důležitý faktor celého mechanismu, který spočíval v délce celé převodovky, musel jsem proto kombinovat dva různé moduly ozubených koleček, tak abych dosáhl požadovaných vzdáleností. Tato převodovka má převod 1:7,875 což znamená 140 kroků pro jeden stupeň.

Díl 10 – Střední část ramene



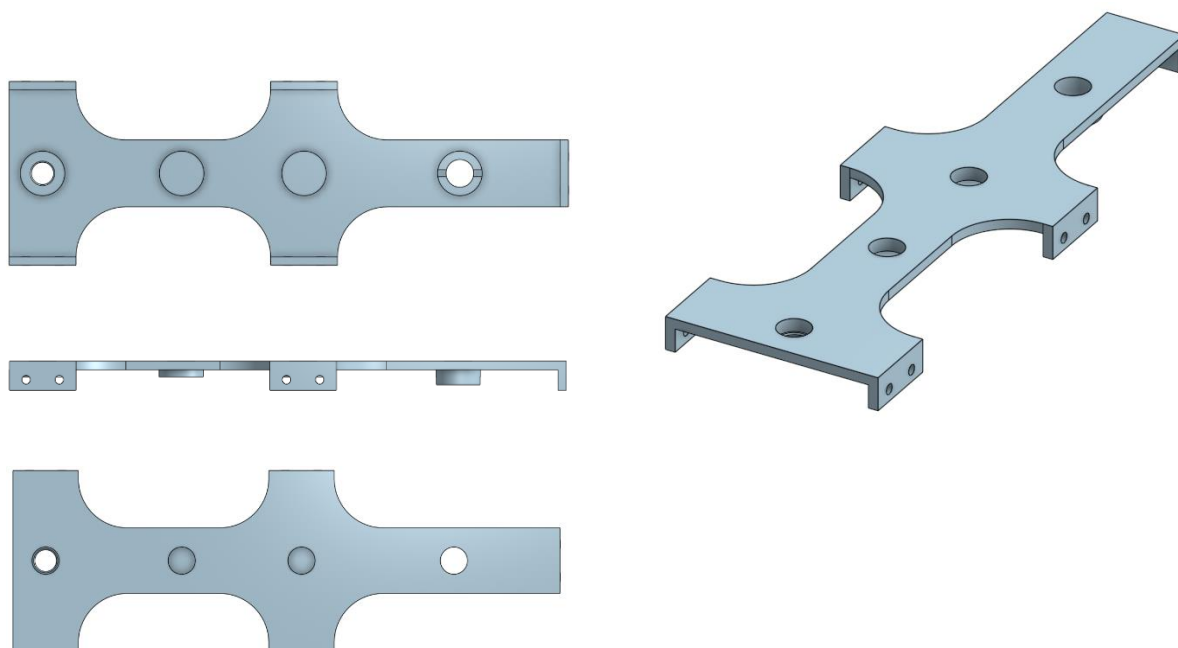
Díl 10 byl poměrně dost složitý pro návrh. Musel jsem zde počítat s mnou zvolenou délkou ramene a velikostí návrhu převodovky. Při návrhu této součástky jsem si uvědomil, že by bylo dobré využít protiváhu, tak aby se konec ramene nepřevažoval a nebyla tak zapříčiněna nerovnoměrná zátěž na výstup z převodovky na podstavě. Opět na tomto dílu jsou navrženy „díry“ pro nosné osy převodovky a úchytné pacičky pro uchycení výstupu z převodovky 2.

Díl 11 – Ring 2



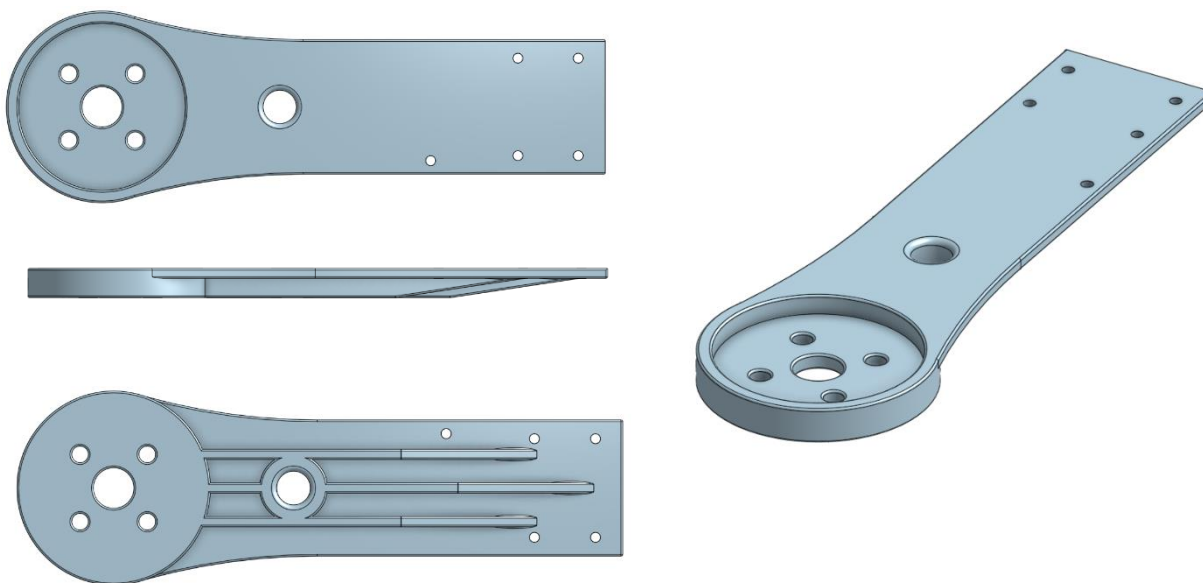
Pro zajištění výstupního kolečka z převodovky a také ze střední části ramene, jsem navrhl **Ring 2**. Za pomoci čtyř malých ložisek jsem byl schopen dosáhnout podobného efektu, jako kdybych použil jedno velké podél obvodu. Čtyři ložiska také stačí pro solidní stabilitu kloubu.

Díl 12 - Páteř



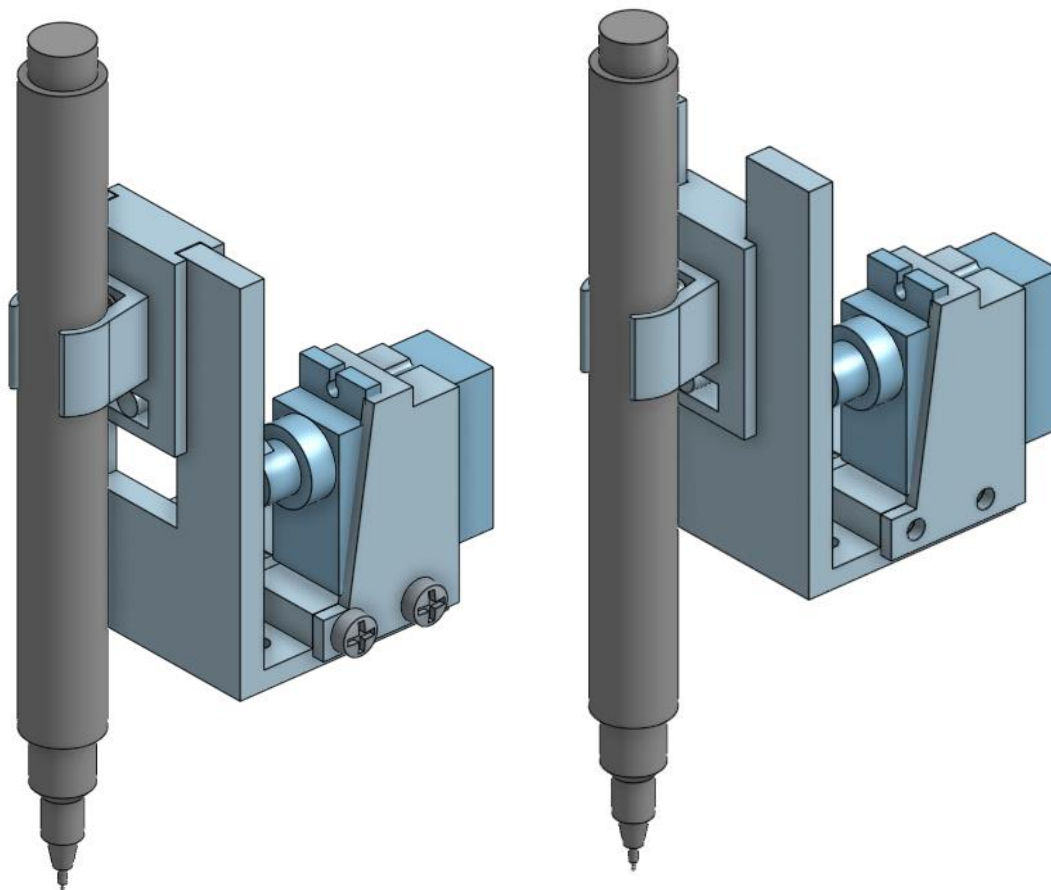
Součástka zvaná páteř slouží k uchycení os **převodovky 2**. Pro účely projektu stačí takováto kombinace převodovky a uchycení. Nicméně správné řešení by spočívalo v přidání ložisek na konce osy.

Díl 13 – Arm end

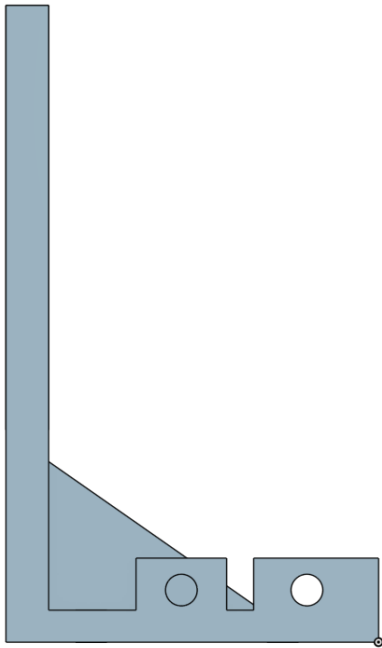
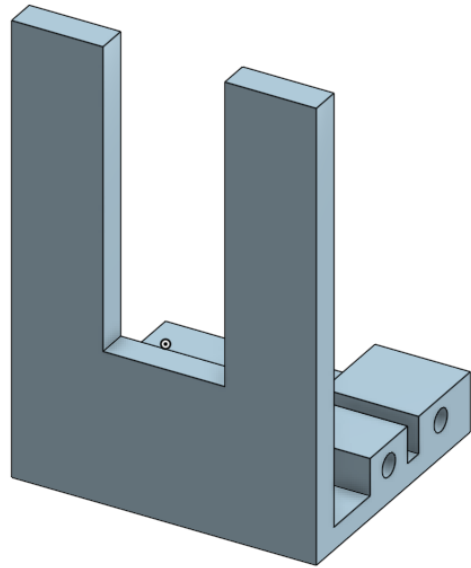
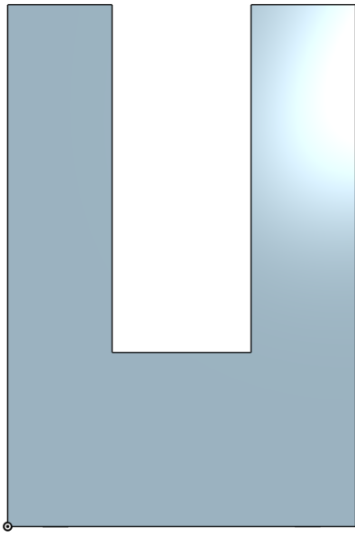


Tento díl slouží jako konstrukce stabilní konstrukce z druhého kloubu na konec robotického ramene. Konstrukce je navržena možná až moc robustně, nicméně to bylo trochu i cílem pro celkovou stabilitu na konci ramene.

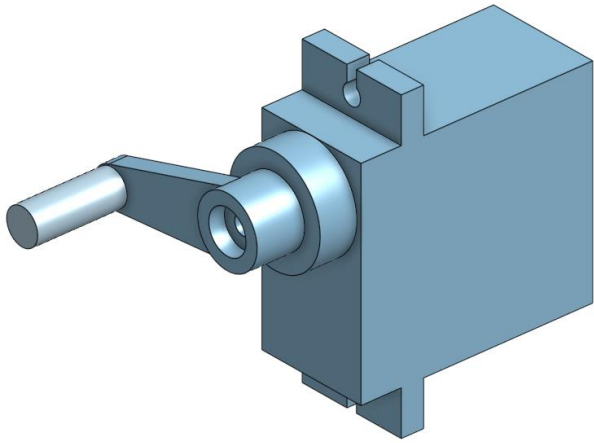
Díl 14 – Marker mechanismus



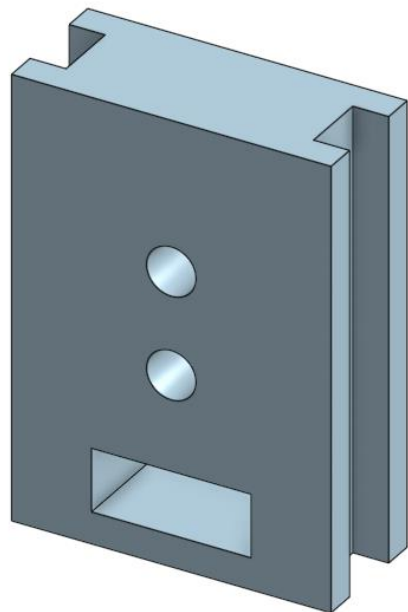
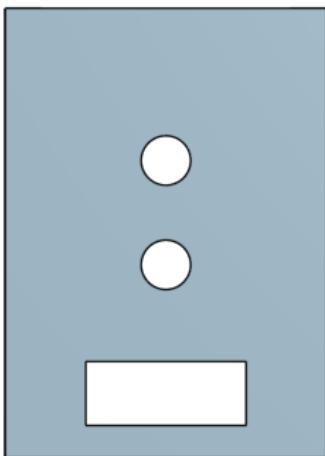
Marker mechanismus – Stand



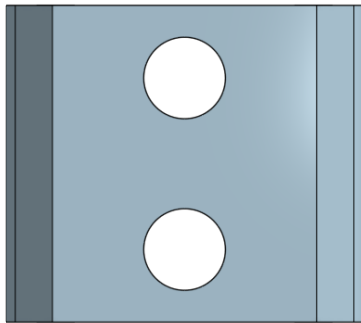
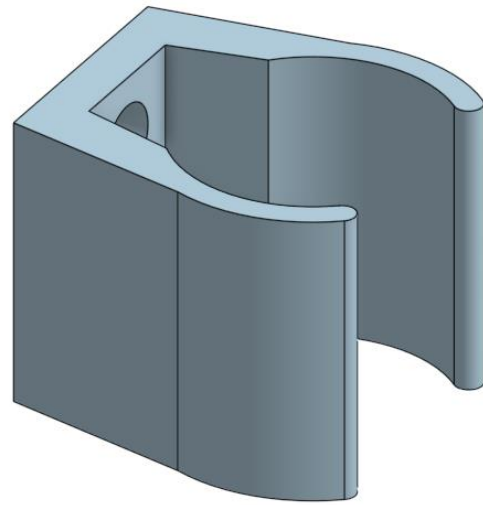
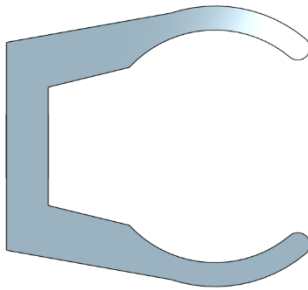
Marker mechanismus – Plug



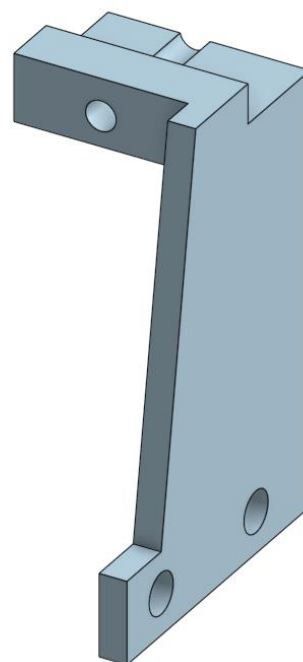
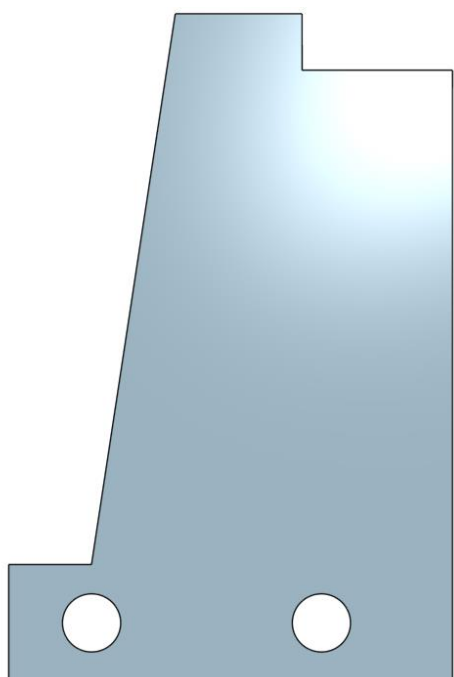
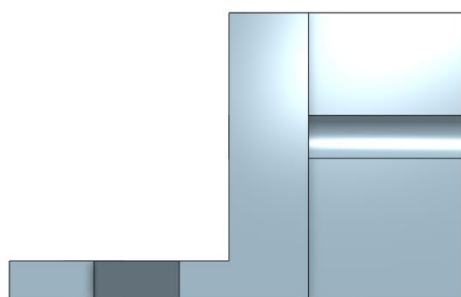
Marker mechanismus – Slider



Marker mechanismus – Marker Holder

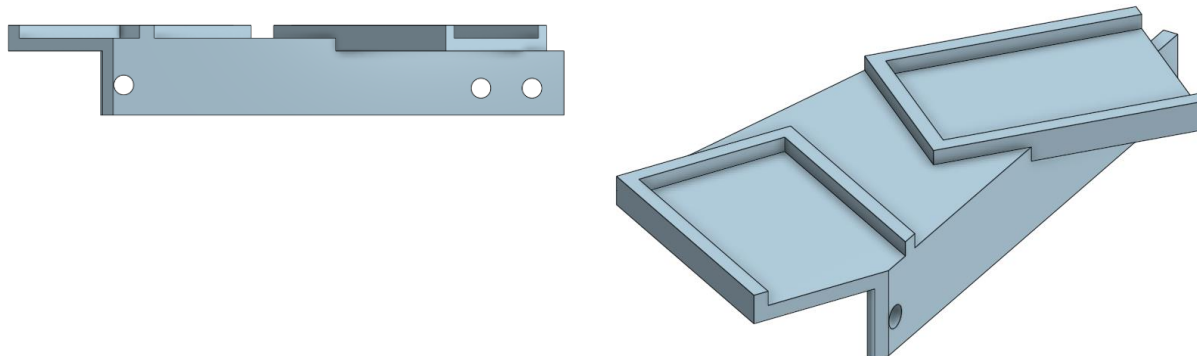


Marker mechanism – Servo Holder



Pro koncové zařízení **Marker mechanismus** jsem zakomponoval servomotor, který zvedá fixu v závislosti na programu. Servomotor je uchycen na **Stand** a z je boku připevněn **Servo Holder**, a vše je sešroubováno. Na servomotor je připevněn **Plug**, který pohybuje s konstrukcí sestavenou ze **Slideru** a **Marker holderu**.

Díl 15 – END switch socket



Tento díl byl poměrně dost náročný pro návrh, za pomoci několika úhlů jsem byl schopen navrhnout tuto součástku. Bylo nezbytně nutné ji navrhnout co nej přesněji pro přesnou kalibraci robotického ramene. Pro návrh bylo také velice podstatné zjistit při jaké délce koncový spínač snímá.

9 Programování

Programování bylo poměrně složité, protože jsem se učil programovat za chodu. K programování jsem používal upravený jazyk C++ pro ESP32. Při programování jsem musel myslet na spoustu faktorů jako například: druhy pinů z mikropočítače, práce s knihovnami, matematické propočítání úhlů a stupňů s délkami ramen a jednotlivých částí, nastavení minimálního limitu, abych zabránil robotickému ramenu v pohybu samo do sebe, kalibrace a přesné určení nulového bodu, práce s vypočítanými hodnoty a následné využití v dalších vzorcích, práce s drivery pro krokové motory, nastavení driverů krokových motorů.

```
D1_AB = sqrt(((0.5 * a) * (0.5 * a)) + (D2_AB * D2_AB));

alpha_AB = atan((0.5 * a) / (D2_AB)) * (180 / PI);
beta_AB = acos(((D1_AB * D1_AB) + (length_A1 * length_A1) - (length_A2 * length_A2)) / (2 * D1_AB * length_A1)) * (180 / PI);
delta_AB = acos(((length_A1 * length_A1) + (length_A2 * length_A2) - (D1_AB * D1_AB)) / (2 * length_A1 * length_A2)) * (180 / PI);

Serial.print(" alpha_AB = "), Serial.print(alpha_AB), Serial.print("      "), Serial.print(" beta_AB = "), Serial.print(beta_AB), Serial.print("      "), Serial.print(" delta_AB = ");

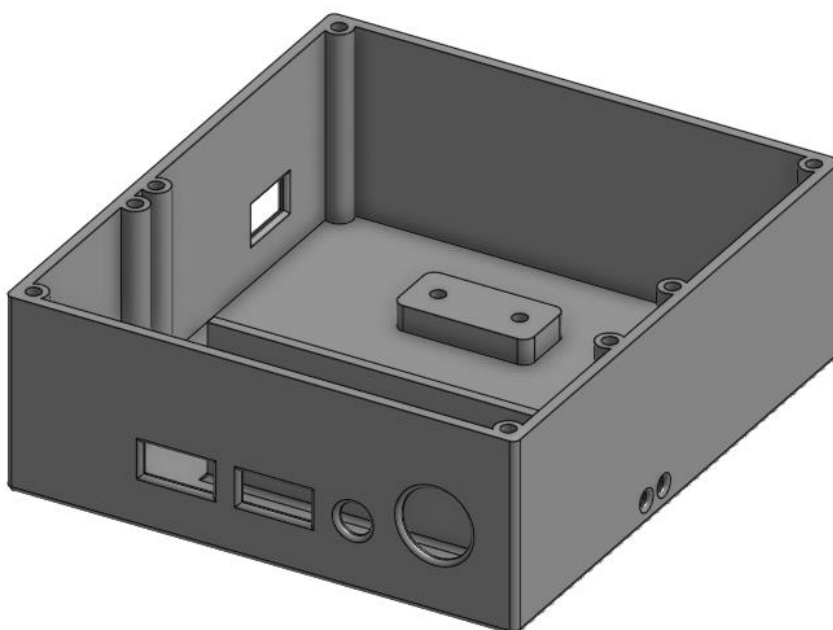
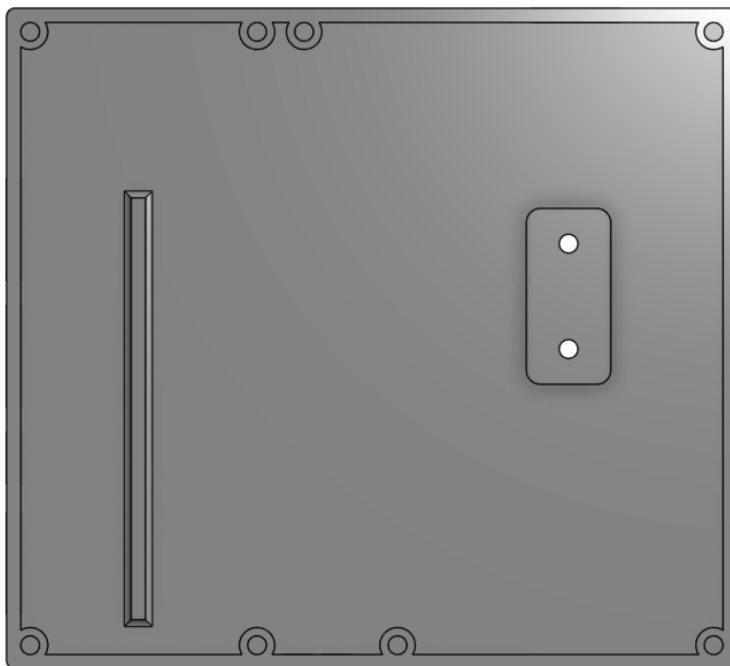
stepper_alpha_angle.moveTo((alpha_AB + beta_AB) * (transmitter_ratio_J1 / 360));
stepper_beta_angle.moveTo((-180 + delta_AB) * (transmitter_ratio_J2 / 360));

while (stepper_alpha_angle.distanceToGo() != 0 || stepper_beta_angle.distanceToGo() != 0) {
  stepper_alpha_angle.run(); // Spustí krok motoru J1
  stepper_beta_angle.run();  // Spustí krok motoru J2
}
```

V tomto kódu je nejdůležitější část, a to výpočet trigonometrických funkcí pro komplexní pohyb robota.

10 Závěrečné úpravy

Ke konci projektu jsem navrhl krabičku pro elektroniku, kam lze schovat přebytečné kabely. Krabička chrání elektroniku také před mechanickým poškozením. Do krabičky budu muset zavést chlazení, protože jsem zjistil, že zejména drivery se poměrně dost ohřívají.



11 Představa o budoucím vylepšení

Do budoucna bych chtěl udělat robota více robustního, schopného kreslit pomocí G-kódu. Konstrukce by také měla být schopna udržet laser pro gravírování například do dřeva. Dále bych chtěl navrhnout a naprogramovat systém pro měnění nástrojů, popřípadě barev fix. Samozřejmě by také stálo zvážit rychlost mikropočítače ESP32 a jeho vlastnostmi s například Raspberry pi pro lepší optimalizaci a programování přímo v jazyce C++ a přímou komunikací s počítačem, kde bych mohl v reálném čase za pomoci výstupu z magnetických senzorů pozorovat naprogramovanou animaci, která by se zobrazovala na monitoru. Co se týče převodovek, chtěl bych použít spíše řemeny pro nulový backlash, který je klíčový v robotice a přesnosti vůbec. Určitě by bylo dobré předělat konstrukci os v převodovkách a použít ložiska pro konce os pro plynulejší rotaci koleček s minimálními vibracemi a odporem. Nejvíce ambiciózní představa by byla schopnost robota soustružit dřevo nebo hliník pro výrobu pevnějších součástí.

12 Seznam použité literatury

<https://drotek.cz/arduino/1025-ramps1.4-drv8825-driver-pro-krokovy-motory.html>

<https://drotek.cz/arduino/51547-esp32-devkitc-development-board-38pin.html>

<https://drotek.cz/arduino/48391-krokovy-motor-nema17-33mm-42hd2037-01.html>

<https://drotek.cz/arduino/897-eses-servo-motor-9g.html>