

ŠOLSKI CENTER CELJE



Srednja šola za strojništvo, mehatroniko in medije

Raziskovalna naloga

RAZVOJ ROBOTSKEGA MEHANIZMA IN IZDELAVA S TEHNOLOGIJO 3D PROSTORSKEGA TISKA

Avtorji:

Uroš KUKOVIČ, M-4. c
Žan CMOK, M-4. c
Gašper DRAKSLER, M-4. c

Mentor:

mag. Andro GLAMNIK, univ. dipl. inž. el
mag. Matej VEBER, univ. dipl. inž. el

Celje, marec 2020

Mentorja Andro Glamnik in Matej Veber v skladu z 20. členom Pravilnika o organizaciji mladinske raziskovalne dejavnosti »Mladi za Celje« Mestne občine Celje, zagotavlja, da je v raziskovalni nalogi z naslovom Razvoj robotskega mehanizma in izdelava s tehnologijo 3D prostorskega tiska, katere avtorji so Uroš Kukovič, Žan Cmok in Gašper Draksler:

- besedilo v tiskani in elektronski obliki istovetno,
- pri raziskovanju uporabljeno gradivo navedeno v seznamu uporabljene literature,
- da je za objavo fotografij v nalogi pridobljeno avtorjevo dovoljenje in je hranjeno v šolskem arhivu,
- da sme Osrednja knjižnica Celje objaviti raziskovalno nalogo v polnem besedilu na knjižničnih portalih z navedbo, da je raziskovalna naloga nastala v okviru projekta Mladi za Celje,
- da je raziskovalno nalogo dovoljeno uporabiti za izobraževalne in raziskovalne namene s povzemanjem misli, idej, konceptov oziroma besedil iz naloge ob upoštevanju avtorstva in korektnem citiranju,
- da smo seznanjeni z razpisni pogoji projekta Mladi za Celje.

Celje, _____

žig šole

Podpis mentorja

Podpis odgovorne osebe

*

POJASNILO

V skladu z 20. členom Pravilnika raziskovalne dejavnosti »Mladi za Celje« Mestne občine Celje je potrebno podpisano izjavo mentorja (-ice) in odgovorne osebe šole vključiti v izvod za knjižnico, dovoljenje za objavo avtorja (-ice) fotografskega gradiva, katerega ni avtor (-ica) raziskovalne naloge, pa hrani šola v svojem arhivu.

ZAHVALA

Zahvaljujemo se mentorjema mag. Andru Glamniku, univ. dipl. inž., in mag. Mateju Vebru, univ. dipl. inž., za usmerjanje in strokovno svetovanje.

Zahvalili bi se tudi podjetju Mobilis, d. o. o., za sponzorstvo pri projektu.

Na koncu gre zahvala še Brigiti Renner, prof. in Simoni Tadeji Ribič, prof., za lektoriranje naše raziskovalne naloge.

POVZETEK

V raziskovalni nalogi je predstavljena robotska roka, ki smo jo izdelali s tehnologijo 3D prostorskega tiska. Opisali smo celoten postopek koncipiranja, priprave tiskanja in sestavljanja, električne vezave in programiranja ter raziskali možnosti izvedbe. V nadaljevanju smo predstavili, na kakšne težave smo naleteli, ko smo želeli robotsko roko uporabiti za strego 3D-tiskalniku in kako smo jih na koncu rešili. Pojasnili smo potrjevanje in zavračanje hipotez ter na koncu predstavili nadaljnje izboljšave.

Ključne besede: robotska roka, strega 3D-tiskalniku, avtomatiziran sistem, 3D-tisk, umetna masa

SUMMARY

In this research assignment a robotic arm is presented. We designed it with the help of a 3D printer. The entire printing and assembly procedure, as well as electrical wiring and programming, is described. In addition, we present the problems we encountered when we wanted the robotic arm to serve a 3D printer, and how we solved them. Furthermore, we give reasons for confirming and rejecting the hypotheses, and present further potential improvements that would make the robotic mechanism even better.

Keywords: robotic arm, 3D printer serving, automated system, plastic, 3D printing

KAZALO VSEBINE

1 UVOD	1
1.1 PREDSTAVITEV PROBLEMATIKE.....	1
1.2 HIPOTEZE	1
1.3 METODE RAZISKOVANJA	2
2 UVOD V RAZISKOVANJE	3
2.1 3D-TISKALNIK.....	3
2.2 IDEJNA ZASNOVA	4
2.3 ROBOTSKI MEHANIZEM.....	5
2.4 SPREMEMBE NA ROBOTU	6
2.5 TISKANJE ROBOTSKO ROKE.....	7
2.5.1 Izbira materiala	7
2.5.2 Nastavitve tiskanja	8
2.6 SESTAVA ROBOTA.....	10
2.6.1 Sestava prijemala	10
2.6.2 Sestava prve osi.....	11
2.6.3 Sestava druge osi.....	14
2.6.4 Sestava tretje osi	15
2.6.5 Sestava četrte osi.....	16
2.6.6 Sestava pete osi	17
3 ELEKTRIČNA VEZAVA	19
3.1 IZBIRA MOTORJEV.....	19
3.1.1 Servo motor.....	19
3.1.2 Koračni motorji.....	19
3.2 IZBIRA KRMILNIKA IN NADZORNEGA VEZJA	20
3.3 RAMPS 1.4 in PROGRAM.....	21
4 PROGRAMIRANJE	25

4.1 PROGRAMIRANJE TISKALNIKA	26
4.2 PROGRAMIRANJE ROBOTSKE ROKE.....	26
5 MERJENJE NATANČNOSTI ROBOTSKE ROKE.....	32
6 MERJENJE NOSILNOSTI ROBOTSKE ROKE	34
7 MERJENJE HITROSTI ROBOTSKE ROKE	35
8 PREDSTAVITEV PROBLEMATIKE STREGE 3D-TISKALNIKA	37
9 POTRDITVE HIPOTEZ	38
10 NADALJNJE IZBOLJŠAVE.....	41
11 ZAKLJUČEK.....	42
11 VIRI.....	43

KAZALO SLIK

Slika 1: Prikaz posameznih osi na 3D-tiskalniku.....	3
Slika 2: 3D-tiskalnik.....	4
Slika 3: Skica robota	5
Slika 4: Industrijski robot KUKA	5
Slika 5: Ime našega robotskega mehanizma	6
Slika 6: 3D-tiskalnik.....	7
Slika 7: Kolut črne PLA-plastike	8
Slika 8: Prerez valja.....	9
Slika 9: Začetki sestavljanja	10
Slika 10: Sestavljeno prijemalo.....	11
Slika 11: Sestava prve osi robotske roke.....	12
Slika 12: Ohišje za motor prve osi	13
Slika 13: Druga os robotske roke	15
Slika 14: Tretja os robotske roke.....	16
Slika 15: Četrta os robotske roke	17
Slika 16: Sestavljena peta os robotske roke	18
Slika 17: Ramps 1.4	20

Slika 18: Arduino Mega	20
Slika 19: Število ekstrudorjev in tip matične plošče	20
Slika 20: Potrebne spremembe parametrov za našo robotsko roko.....	21
Slika 21: Koračni motorji	22
Slika 22: Skica vezave za gonilnike	23
Slika 23: Vezava iz gonilnikov na Ramps priključke	23
Slika 24: Skica definiranja motorjev na posamezno os.....	24
Slika 25: Program Pronterface	25
Slika 26: Končni del G-kode tiskalnika	26
Slika 27: Postopki programiranja robotske roke	27
Slika 28: Premikanje vsake osi posebej	27
Slika 29: Shranjevanje koordinat osi.....	27
Slika 30: Razdalja na korak za posamezne osi.....	28
Slika 31: G-koda robotske roke.....	29
Slika 32: Vijachenje robota na ploščo.....	29
Slika 33: Napajalnika na spodnji strani lesene plošče.....	30
Slika 34: Neurejeni kabli motorjev in gonilnikov	30
Slika 35: Urejeni kabli z vezicami	30
Slika 36: Nameščen pokrov.....	31
Slika 37: Celoten robot z zaščitnimi kabli.....	31
Slika 38: G-koda za merjenje natančnosti	32
Slika 39: Rezultati merjenja natančnosti roke.....	33
Slika 40: Rezultati nosilnosti robotske roke.....	34
Slika 41: Merjenje nosilnosti robotske roke.....	34
Slika 42: Krožni lok	35
Slika 43: Rezultati hitrosti posameznih osi	36
Slika 44: 6-osna robotska roka	39
Slika 45: Hitrosti posameznih osi pri 180°	40
Slika 46: Strega 3D-tiskalniku za robotsko roko	40

KAZALO TABEL

Tabela 1: Elementi za sestavo 1. osi	12
Tabela 2: Elementi za sestavo 1. osi	13
Tabela 3: Elementi za sestavo 2. osi	14
Tabela 4: Elementi za sestavo 3. osi	15
Tabela 5: Elementi za sestavo 4. osi	17
Tabela 6: Elementi za sestavo 5. osi	18
Tabela 7: Podatki koračnih motorjev	19

KAZALO GRAFOV

Graf 1: Merjenje natančnosti robotske roke	33
Graf 2: Hitrosti posameznih osi	36

1 UVOD

Dandanes je čedalje več poznavalcev 3D-tiska in vse več ljudi se ljubiteljsko ukvarja s 3D-tiskalniki. Pred približno deset leti so se tiskalniki masovno uporabljali le v industriji, danes pa so zelo popularni med potrošniki, saj so se tehnologije razvile, cene pa zelo znižale. Vendar je potek tiskanja izdelka počasen in lahko traja tudi več deset ur. Ideja te raziskovalne naloge je koncipirati in izdelati 3D natisnjeno robotsko roko in jo kot referenčni primer uporabiti za strego 3D-tiskalniku. Podane in predstavljene so tudi vse skice ter napredek projekta. Na drugi strani najdemo robotske roke, naprave, ki so ključnega pomena v sodobni industriji. Več bomo predstavili v nadaljevanju.

1.1 PREDSTAVITEV PROBLEMATIKE

Pri izdelavi 3D natisnjene robotske roke se lahko pojavi veliko problemov, kot sta npr. natančnost in ponovljivost, saj se odstopanja pojavijo že pri tiskanju posameznih komponent. To lahko pri končno izdelani robotski roki privede do velikih odstopanj in nenatančnosti. Problem se pojavi tudi pri bremenu, ki ga robot lahko dvigne, saj motorji nimajo enake moči in navora. Robot bo lahko dvignil samo toliko, kot bo to omogočal najšibkejši motor. Zelo pomembna je tudi hitrost, vendar pri tem nastopi težava, katere motorje uporabiti. Zgradba motorja mora biti dovolj močna, da prenese vse obremenitve jermenic in jermenov. Bistven del robotske roke je prijemalo. Brez njega robotska roka izgubi funkcionalnost, saj bi jo mi uporabljali za prijemanje in premikanje izdelkov. Zaradi tega smo morali dobro premisliti, kakšno prijemalo bomo uporabili in kakšen motor bo nameščen nanj.

1.2 HIPOTEZE

V okviru raziskovalne naloge smo postavili sledeče hipoteze:

H1 – S tehnologijo 3D prostorskega tiska je možno izdelati 6-osni robotski mehanizem.

H2 – Robotska roka bo dosegla delovno hitrost 0.15 m/s.

H3 – Robotska roka bo dosegla nosilnost 1 kg.

H4 – Robota lahko uporabljamo za strego 3D-tiskalniku.

1.3 METODE RAZISKOVANJA

Pri raziskovalnem delu smo uporabili različne metode raziskovanja. Najprej smo morali poiskati in pridobiti čim več informacij o robotskih rokah. Pri odločitvi, da bomo robotsko roko uporabili za strego 3D-tiskalniku, smo morali obstoječe znanje še nadgraditi. Pri tem smo spoznali prednosti in slabosti 3D-tiskalnikov ter odkrili težave, ki bi nam otežile delo. Pozanimali smo se tudi, kateri material bi bil najprimernejši za tiskanje robotske roke, poleg tega pa smo morali preučiti še motorje in njihove moči.

Naslednja raziskovalna metoda, ki smo jo uporabili, je konstruiranje delov robotske roke v modelirnem programu SolidWorks. S pomočjo programa smo spremenili jermenico prve osi, narisali smo ščita za motor in za krmilje robotske roke, spremenili smo šesto os in na trup dodali ime skupine.

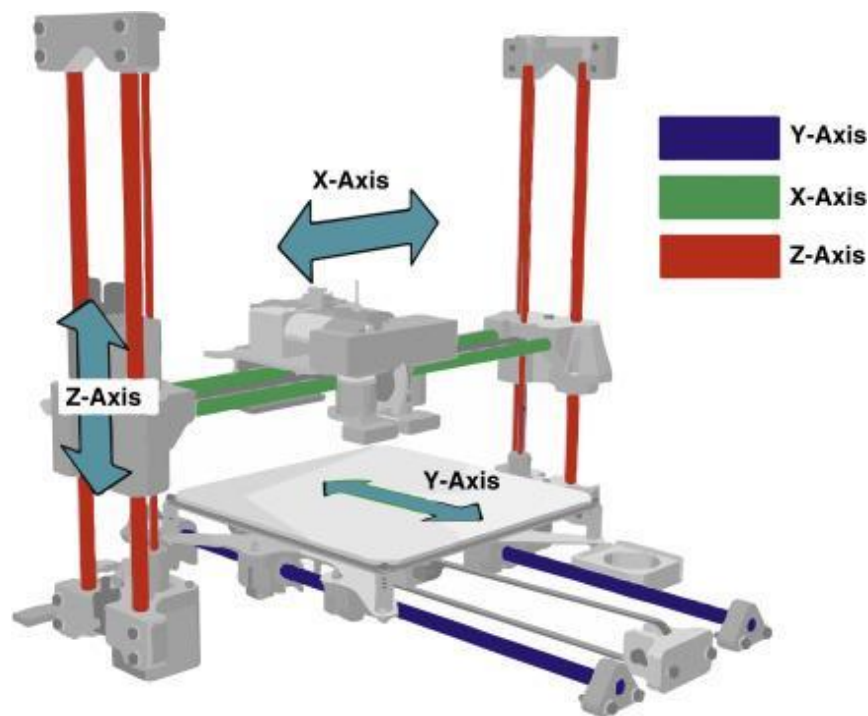
Ko smo robotsko roko izdelali, smo jo testirali. Najprej smo izmerili natančnost in nosilnost, v nadaljevanju pa še hitrost gibanja. Po testiranjih smo uporabili še četrto raziskovalno metodo, to je analiza rezultatov. Z analizo smo lahko kasneje narisali grafe.

2 UVOD V RAZISKOVANJE

2.1 3D-TISKALNIK

3D-tiskalnik je naprava, s katero lahko ustvarimo poljubni izdelek. Če se nam doma nekaj zlomi, pokvari ali če kaj izgubimo, lahko to stvar enostavno natisnemo in jo zamenjamo. Natisnemo lahko različne pripomočke, igrače za otroke ...

Tovrstni tiskalniki delujejo na principu tridimenzionalnega nanašanja materiala na delovno površino. Tridimenzionalno delovanje drugače opišemo kot gibanje v treh oseh za izdelovanje predmetov. Pri večini tiskalnikov se delovna površina/miza giblje v smeri Y-osi, konica, ki nanaša material, pa v smeri X- in Z-osi. Osi so prikazane na naslednji sliki.



Slika 1: Prikaz posameznih osi na 3D-tiskalniku

(Vir: <https://prototypehubs.com/wp-content/uploads/2019/09/3D-Printer-Axis.png>)

Material za 3D-tiskanje je navit na kolutih, ponavadi je to umetna masa, uporabimo pa lahko tudi lesene, kovinske, steklene, glinene materiale ipd. Motorček, ki je vgrajen na strani tiskalnika, potiska ta material po cevki vse do konice tiskalnika, kjer se segreje do tekočega stanja. Temperatura konice se giblje od 180 °C do 240 °C, odvisno od materiala. Tiskalnik z gibanjem po zgoraj omenjenih oseh nanaša staljeni material na delovno površino in tako gradi izdelek. Ko se material nanese, se ta ohladi nazaj v trdno obliko in ostane na mestu nanosa. Največkrat se plasti nanašajo na 0.2 mm debelo, kar privede do velikih časov tiskanja. Z nadgradnjami tiskalnika lahko

izboljšamo vse dejavnike, ki porabljajo čas med njegovim obratovanjem. Mi smo se osredotočili na problem, ki se pojavi pred začetkom tiskanja in po njem. Pred tiskanjem se mora tiskalnik zagreti, naložiti moramo program in ga pognati, po koncu pa moramo izdelek odstraniti in na novo zagnati program, če želimo ustvariti enak izdelek. To porabi veliko našega časa. Primer klasičnega tiskalnika je prikazan na spodnji sliki.

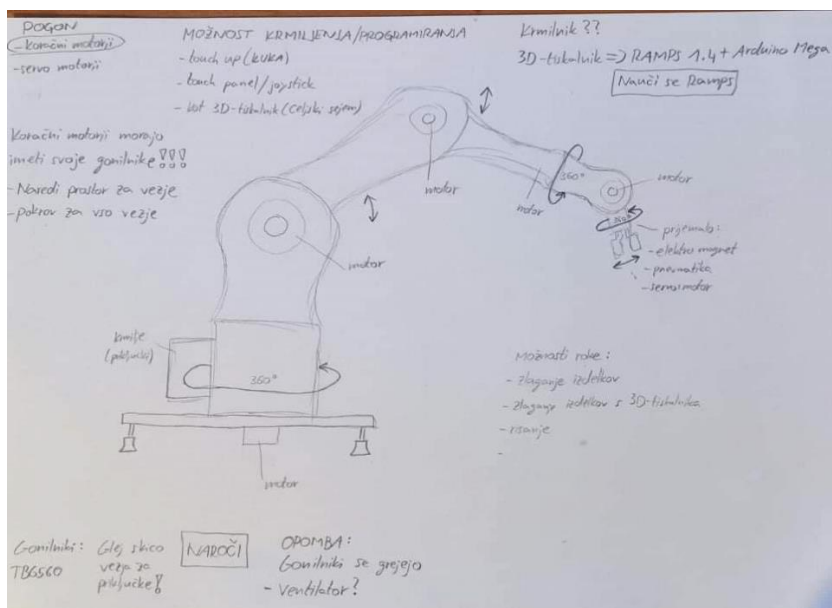


Slika 2: 3D-tiskalnik

(Vir: https://commonslib.gr/wp-content/uploads/2016/08/TAZ_6_Angle_PressRelease_1500px.jpg)

2.2 IDEJNA ZASNOVA

Prvotna ideja je bila, da izdelamo robotsko roko, ki bi jo uporabljali za različne namene. Med njeno izdelavo smo se odločili, da jo bomo kot primer uporabili za aplikacijo strege 3D-tiskalniku. Po pregledu vseh 3D natisnjenih robotskih rok, ki smo jih našli na spletu, smo si izbrali tisto, ki se nam je zdela najprimernejša in jo izdelali s tehnologijo 3D prostorskega tiska



Slika 3: Skica robota

(Vir: osebni arhiv)

2.3 ROBOTSKI MEHANIZEM

Robotska roka je vrsta mehanske roke, ki jo običajno programiramo, in ima podobne funkcije kot človeška roka [13]. Veliki proizvajalci robotskih rok so KUKA, Yaskawa, ABB, poznamo pa še veliko drugih podjetij. Tako kot vsaka naprava, ima tudi robotska roka osi, ki so definirane z X, Y in Z. Poznamo večosne robotske roke, najbolj pa je znana 6-osna. V proizvodnji 4.0 se veliko uporabljajo npr. pri proizvodnji avtomobilov. Sprva smo nameravali izdelati 5-osno robotsko roko, a smo se kasneje odločili, da potrebujemo šesto os zaradi rotacije prijemale.



Slika 4: Industrijski robot KUKA

(Vir: <https://www.roboteh.si/>)

2.4 SPREMEMBE NA ROBOTU

Ker robot ni bil izpopolnjen in dovršen, kot smo si želeli, smo nekaj stvari spremenili. Prva os ni več omejena. To smo dosegli tako, da smo spremenili jermenico, ki je bila zmodelirana tako, da je bil jermen fiksno vpet. Ko smo to spremenili, smo omogočili, da se robot zavrti za več kot 360° oz. se lahko zavrti toliko, kolikor to dopušča dolžina kablov, da se le-ti ne pretrgajo. Spremenili smo tudi šesto os in s tem omogočili možnost vgradnje močnejšega koračnega motorja. Na izdelek smo dodali tudi svoje ime: ARM3D (glej sliko 5). Nazadnje pa smo zmodelirali še dva ščita. Prvega smo zasnovali za koračni motor prve osi, saj je motor skupaj z jermenico odprt in nevaren tako za človeka kot tudi za kable na robotu, ki se lahko zapletejo v jermenico in jermen ter se pri tem pretrgajo. Drugi ščit oziroma pokrov pa ščiti krmilni sistem robota.

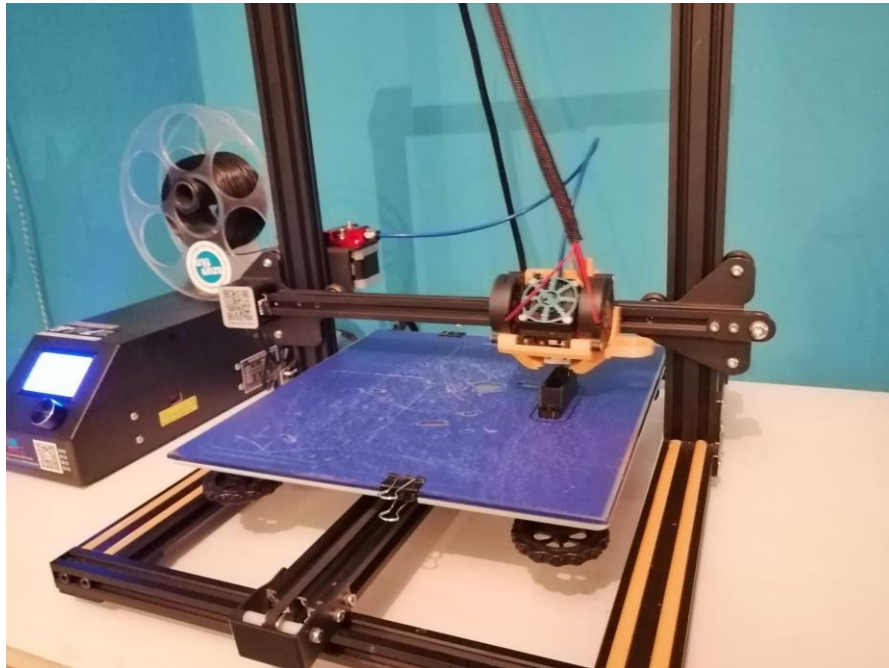


Slika 5: Ime našega robotskega mehanizma

(Vir: osebni arhiv)

2.5 TISKANJE ROBOTSKE ROKE

Tiskanje smo izvajali doma na lastnem tiskalniku, za katerega tudi izdelujemo robotsko roko. Uporabljali smo PLA-plastiko v črni in zeleni barvi, porabili smo jo približno 4 kg. Potrebovali smo približno 400 ur, da smo natisnili vse sestavne dele za robotsko roko.



Slika 6: 3D-tiskalnik

(Vir: osebni arhiv)

2.5.1 Izbira materiala

Pri izbiri materiala smo morali upoštevati trdoto, žilavost in obstojnost tiskalnih materialov. Za robotsko roko te velikosti potrebujemo dokaj trd in žilav material, saj je vse zelo težko. Pri tem odpadejo vsi fleksibilni materiali. Naslednji problem je toplota, ki jo oddajajo motorji. Če se plastika stali, se lahko robot zlomi, pokvari pa se lahko tudi motor. S tem iz ožjega izbora odstranimo lesene materiale in nekatere podporne vrste materialov. Materiali, kot so ABS, PETG in ASA, so najodpornejši proti toploti in vremenskim pojavom, a so najtežji za tiskanje. Na toploto se malo slabše odziva tudi PLA-plastika, a se nam je zdela najprimernejša, saj je najlažja za tiskanje, poleg tega pa je tudi najcenejša. Prav tako je dovolj trdna in ni fleksibilna. Polilaktid ali krajše PLA-plastika je navita na kolutih.



Slika 7: Kolut črne PLA-plastike

(Vir: osebni arhiv)

2.5.2 Nastavitve tiskanja

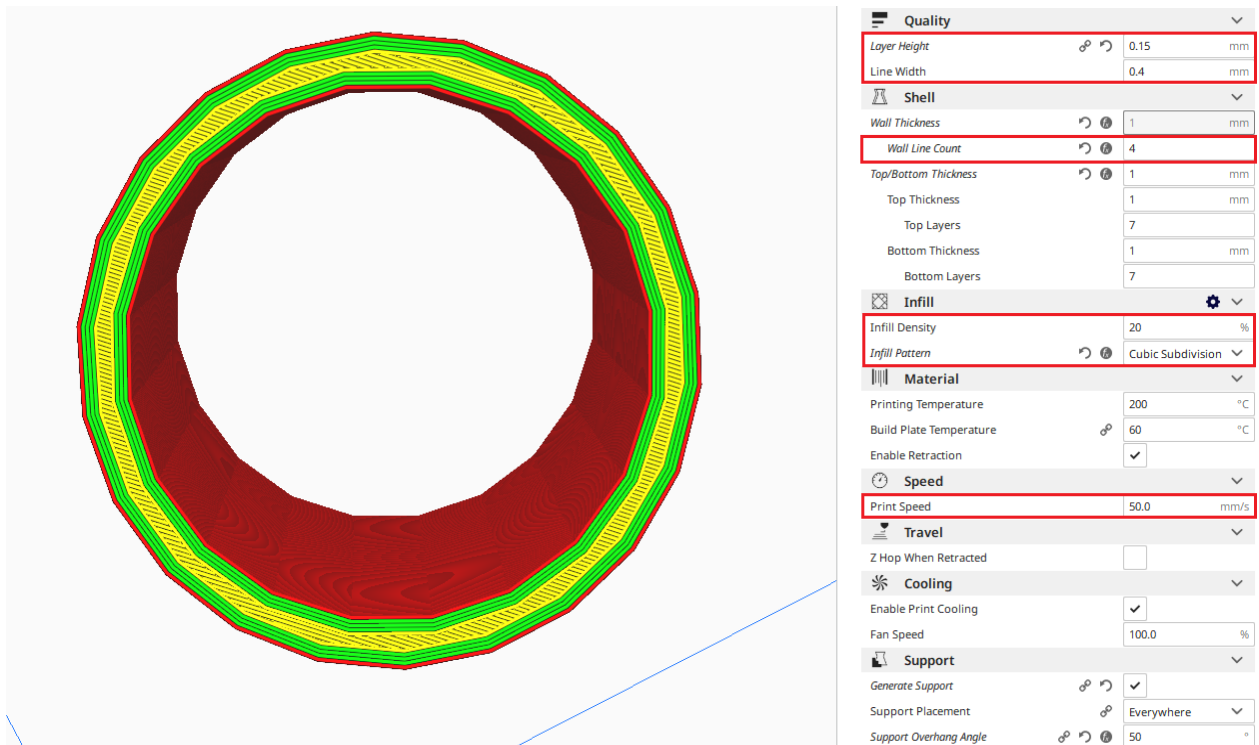
Pri tiskanju smo bili pozorni na:

- debelino sten,
- vzorec polnila,
- gostoto polnila in
- hitrost ter debelino nanašanja.

Debelina sten je pomembna za trdnost robota. Če bi izbrali premalo sten, bi se lahko element med delovanjem zlomil. Vzorec polnila je prav tako pomemben za trdnost izdelka. Po daljšem preizkušanju tudi z drugimi izdelki smo ugotovili, da se najbolje obnese ta polnila Grid in Cubic Subdivision, prvi kvadratne oblike in drugi trikotne. Odločili smo se za Cubic Subdivision, ker se je izkazalo, da se lažje tiska oz. ga tiskalnik lepše natisne, zato je močnejši od Grida. Gostoto polnila, ki je tudi pomembna za trdnost, smo nastavili na 20 %. To pomeni, da ima izdelek v sredini 20 % materiala in 80 % zraka. Ta vrednost se je izkazala za dokaj primerno, čeprav se zdi majhna.

Po testiranju tiskalnika smo ugotovili, da tiskalnik najlepše tiska na 50 mm/s. Debelino tiska smo nastavili na 0.15 mm (glede na Z-os). To je sredina med kakovostjo izdelka in trdnostjo.

Na sliki spodaj vidimo, katere črte so stene. Zunanja stena je obarvana rdeče, notranje stene pa zeleno.

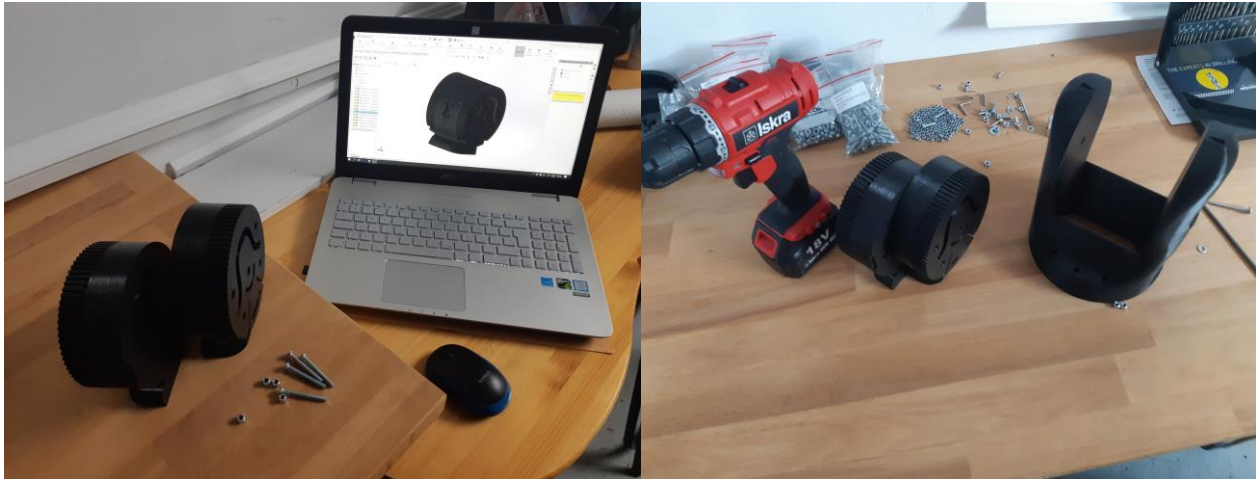


Slika 8: Prerez valja

(Vir: osebni arhiv)

2.6 SESTAVA ROBOTA

Robota smo prav tako 3D-Cad prilagodili s pomočjo programa SolidWorks, na katerem smo imeli načrte za sestavljanje. Najprej smo morali očistiti vse sestavne dele oz. odstraniti odvečno plastiko, ki nastane pri 3D-tiskanju kot opora posameznemu sestavnemu delu med tiskanjem. Tudi vse luknje smo morali očistiti in povrtati za lepše prileganje vijakov, ležajev in puš. Za sestavljanje smo porabili precej manj časa kot za tiskanje, približno 8–10 ur.

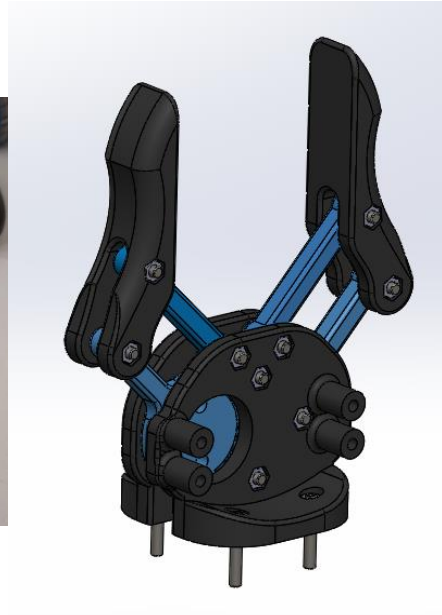


Slika 9: Začetki sestavljanja

(Vir: osebni arhiv)

2.6.1 Sestava prijemala

Sledilo je tiskanje prijemala. Natisnili smo obe strani prijemala in obe plošči, na kateri smo ga pritrdili. Prijemalo deluje na principu zobnikov, ki jih poganja servomotor. Nanj smo namestili tudi gumo, in sicer na del prijemala, ki prijema izdelke. Lahko bi jim rekli prsti. Zaradi boljšega oprijema smo uporabili gumo, saj se nam ta zdi najprimernejša. V mislih smo imeli tudi natisnjeno fleksibilno plastiko, a ima guma več trenja in zato manj drsi. To pride v ospredje pri držanju izdelka, saj nočemo, da izdelek po nesreči pade iz prijema roke.



Slika 10: Sestavljeno prijemalo

(Vir: osebni arhiv)

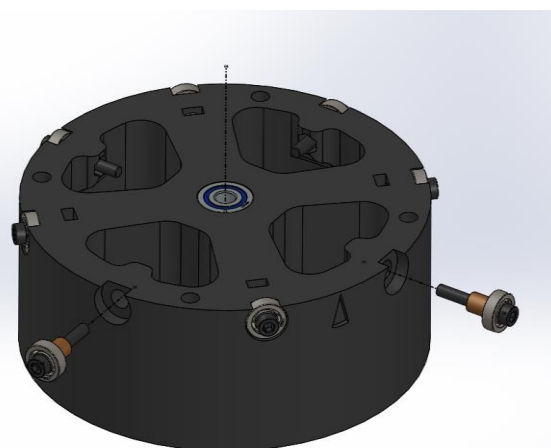
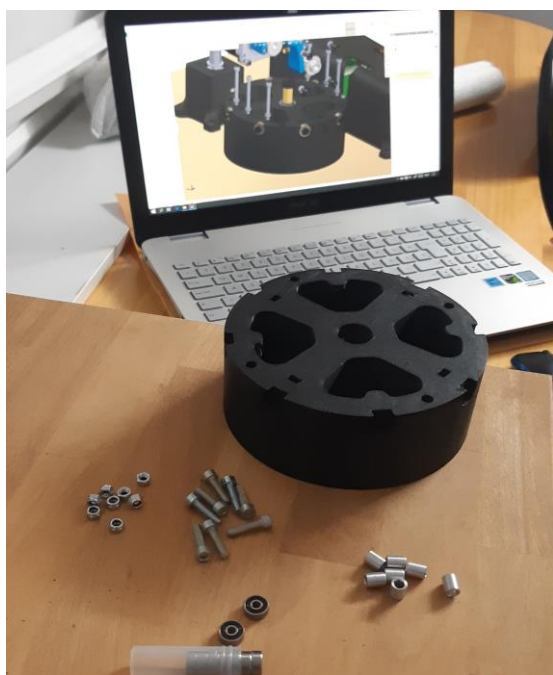
2.6.2 Sestava prve osi

Nadaljevali smo s sestavo prve osi. Priskrbeli smo vse potrebne komponente, kot so: vijaki, matice, podložke, ležaji, jermenice in puše. Pomembno je, da smo delali previdno in korak za korakom ter upoštevali načrte za sestavo robotske roke. Pri vstavljanju puš v jermenico smo si morali pomagati z ročno hidravlično stiskalnico. Ta postopek se nam je zdel najprimernejši, saj si nismo smeli dovoliti, da bi izdelek poškodovali. Prva os je navpična in se premika s pomočjo ležajev, nameščenih na zunanji strani ogrodja. Z nameščanjem ležajev smo imeli nekaj težav, saj so imeli preveliko zračnost, kar je slabo vplivalo na natančnost robota. Zaradi tega smo spodnji del robota obtežili in ležaje enakomerno privijačili. Tako smo dosegli večjo natančnost. Ležaji delujejo kot kolesa, nad njimi pa je natisnjena jermenica za pritrditev jermena. Na vse omenjeno bo nameščena druga os.

Tabela 1: Elementi za sestavo 1. osi

(Vir: osebni arhiv)

KOMPONENTE	ŠT. KOMPONENT
podstavek robota	1
natisnjena jermenica	1
kroglični ležaj (SKF-625)	8
kroglični ležaj (SKF-608)	1
vijak (M8 x 65 mm)	1
vijak (M5 x 20 mm)	8
puša (5 x 8 x 10 mm)	8
puša (8 x 12 x 20 mm)	1
jermen (T5 x 16 mm)	1
samovarovalna matica (M8)	1



Slika 11: Sestava prve osi robotske roke

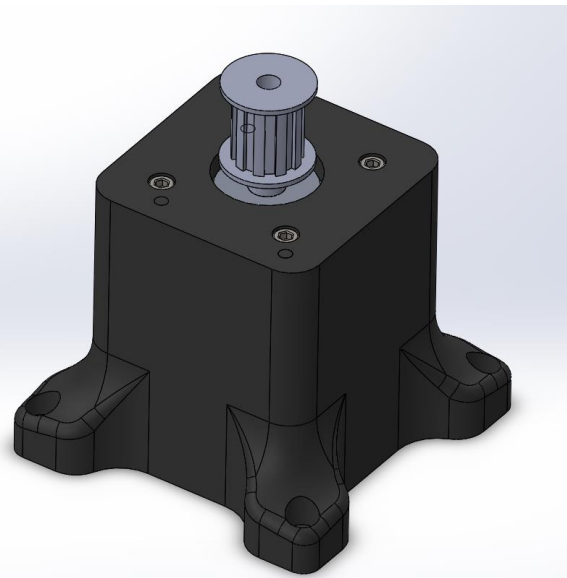
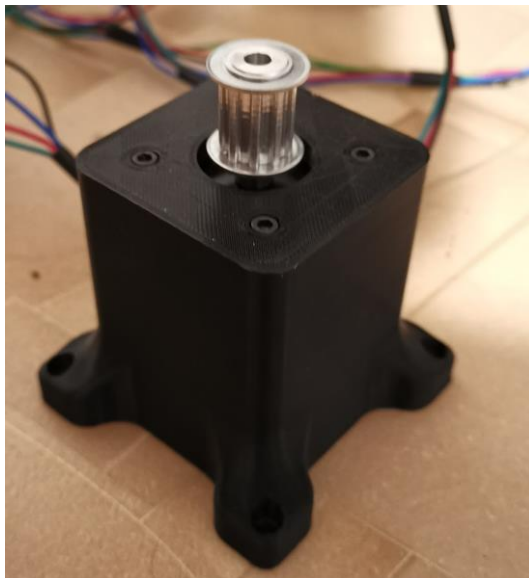
(Vir: osebni arhiv)

Na drugi strani podlage stoji motor za vrtenje prve osi. Povezava je omogočena z jermenom in jermenico na motorju, vse skupaj pa je zaprto z zaščitnim pokrovom, ki smo ga naknadno izdelali. Za izdelavo pokrova smo se odločili, saj želimo zaščititi motor in s tem preprečiti možnost nesreče. Prav tako preprečimo zapletanje kablov z motorjem in človekovo poseganje med procesom delovanja.

Tabela 2: Elementi za sestavo 1. osi

(Vir: osebni arhiv)

KOMPONENTE	št. KOMPONENT
ohišje motorja	1
motor (Nema 17)	1
vijaki (M3 x 8 mm)	4
jermenica (T5, 10 zob, Φ izvrtine 5 mm)	1
vijak (M x 10 mm)	4
samovarovalne matice (M3)	4



Slika 12: Ohišje za motor prve osi

(Vir: osebni arhiv)

2.6.3 Sestava druge osi

Druga os poteka vodoravno glede na podstavek robotske roke. Tiskanje ohišja prve osi je potekalo najdlje, kar 60 ur, saj je to največji 3D natisnjen del na robotu. Dolžina osi, ki je bila vstavljena v to ohišje, je 141 mm. Poleg osi smo za sestavo tega dela robotske roke potrebovali še dva ležaja, jermenico, vse potrebne vijake za pritrditev motorja in jermenice ter dva pokrova, ki zavarujeta ležaje in preprečujeta, da bi se na njih nabirala umazanija. Za premikanje te osi uporabljamo dva koračna motorja tipa Nema 23, ki sta najmočnejša med vsemi motorji, uporabljenimi na našem robotu. Nameščena sta pri vznožju robotske roke, zato premikata celotno roko in sta ključnega pomena za dvigovanje predmetov.

Tabela 3: Elementi za sestavo 2. osi

(Vir: osebni arhiv)

KOMPONENTE	ŠT. KOMPONENT
ohišje druge osi	1
motor (Nema 23)	2
vijak (M5 x 15 mm) - za pritrditev motorjev	8
jermenica (T5, 14 zob, Φ izvrtine je 8 mm)	2
kroglični ležaj (SKF-608)	2
os (Φ 8 x 141 mm)	1
navojni zatič (M3 x 5 mm)	2
navojni vložek (M4)	2



Slika 13: Druga os robotske roke

(Vir: osebni arhiv)

2.6.4 Sestava tretje osi

Sestavljanje tretje osi je potekalo podobno kot sestavljanje prejšnje, vendar smo za razliko od druge osi uporabili samo en motor, ki ima vgrajen reduktor v razmerju 5 : 1. S tem smo zmanjšali težo robota, ker ni potrebna vgradnja dveh motorjev, saj z reduktorjem, ki je na motorju, pridobimo veliko moči.

Tabela 4: Elementi za sestavo 3. osi

(Vir: osebni arhiv)

KOMPONENTE	ŠT. KOMPONENT
motor (Nema 17 1:5)	1
jermenica (T5, 14 zob, Φ izvrtine je 8 mm)	1
navojni vložek (M4)	1
navojni zatič (M3 x 5 mm)	1
vijaki (M3 x 12 mm) - za pritrditev motorja	4
navojni vložek (M3)	6
vijak (M3 x 10 mm)	6
kroglični ležaj (SKF-608)	2
os (Φ 8 x 121 mm)	1



Slika 14: Tretja os robotske roke

(Vir: osebni arhiv)

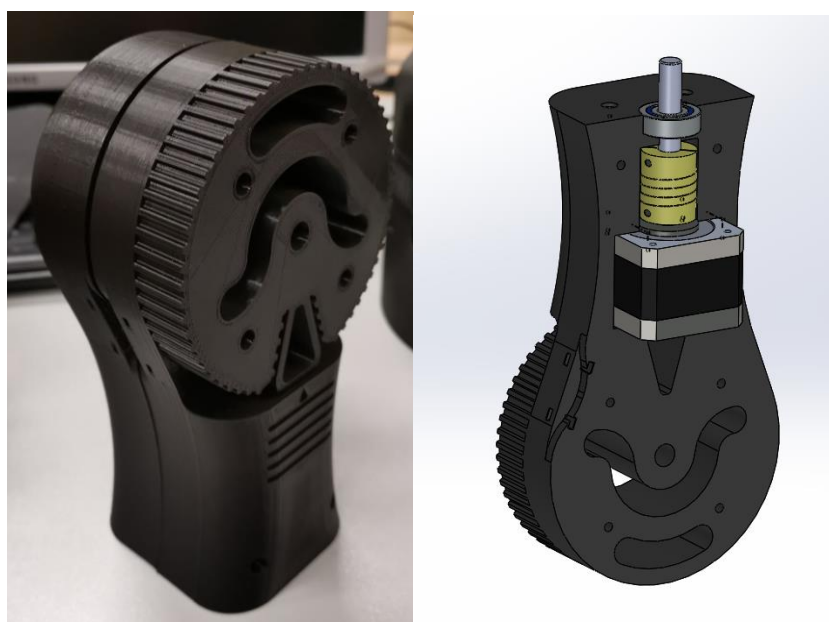
2.6.5 Sestava četrte osi

Sestavljanje četrte osi je bilo nekoliko drugačno in bolj zapleteno kot sestavljanje ostalih osi, saj je motor v tem primeru vgrajen v notranjosti same robotske roke. Zapletlo se je že pri vgradnji motorja, ki se ni povsem prilegal v notranjost natisnjene delu. Zaradi tega smo morali odstraniti nekaj odvečne plastike, tako da se je motor lepo prilegal v notranjost. Kot os je v tem delu robota vstavljena navojna palica M8, ki skrbi, da se lahko peta os in prijemalo premikata za 360° okoli svoje osi. Za povezovanje motorja z navojno palico smo uporabili gibljivo sklopko, ki ima na eni strani luknjo $\Phi 5$ mm za prileganje na motor, na drugi strani pa luknjo $\Phi 8$ mm za privijanje navojne palice.

Tabela 5: Elementi za sestavo 4. osi

(Vir: osebni arhiv)

KOMPONENTE	ŠT. KOMPONENT
motor Nema 17	1
kroglični ležaj (SKF-608)	1
navojna palica (M8 x 50 mm)	1
vijak (M3 x 40)	6
vijak (M3 x 10)	2
matica (M3)	8
gibljava sklopka	1



Slika 15: Četrta os robotske roke

(Vir: osebni arhiv)

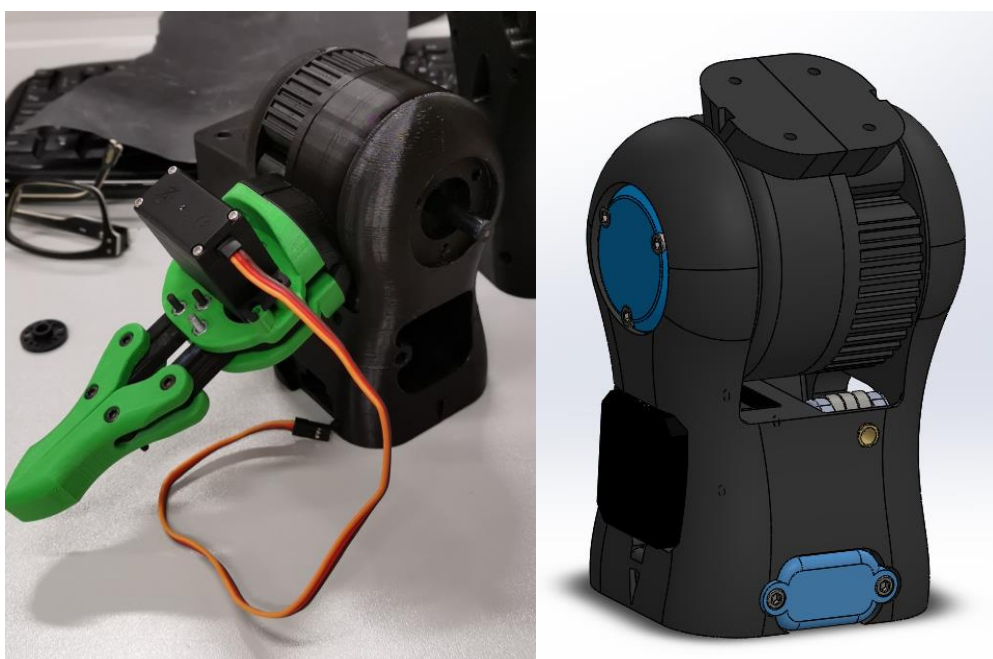
2.6.6 Sestava pete osi

Na peto os je pritrjeno prijemalo; vanjo je vgrajen najmanjši motor na robotu. Os je vodoravna glede na podstavek in ima vgrajen motor NEMA 14, os, dolžine 81 mm, ležaje, dva pokrova, vse potrebne vijake in maticice. Premikanje prijemala okoli svoje osi omogoča jermen, ki ga poganja motor s pomočjo jermenice.

Tabela 6: Elementi za sestavo 5. osi

(Vir: osebni arhiv)

KOMPONENTE	ŠT. KOMPONENT
jermenica	1
motor (NEMA 14)	1
kroglični ležaj (SKF-608)	2
os ($\Phi 8 \times 81 \text{ mm}$)	1
vijaki (M3 x 8 mm)	4
vijak (M3 x 10 mm)	6
navojni vložek (M3)	6
pokrov	2



Slika 16: Sestavljena peta os robotske roke

(Vir: osebni arhiv)

3 ELEKTRIČNA VEZAVA

3.1 IZBIRA MOTORJEV

Medtem ko so se tiskali deli robota, smo na spletu poiskali potrebne motorje in jih naročili. Izbrati smo morali motorje, ki bodo prenašali celotno maso robota in zagotovili potrebno delovno silo. Odločali smo se med servomotorji in koračnimi motorji.

3.1.1 Servo motor

Servomotor je motor, ki deluje na principu določanja stopinj. Servomotorji so lahko 360° ali 180°. Na spletu smo našli veliko robotskih rok, ki imajo nameščene servomotorje, a težava je v tem, da pri njih ne moremo poljubno nastavljati hitrosti. Hitrost lahko le definiramo v konfiguraciji servomotorja. Zanje se nismo odločili prav zaradi tega, ker imajo ostale naprave, kot so 3D-tiskalniki in manjši CNC-stroji, nameščene koračne motorje.

3.1.2 Koračni motorji

Koračni motorji delujejo na principu branja in izvajanja korakov. Zaradi večjih navorov in tokov potrebujejo svoje gonilnike, ki jim zagotavljajo ravno dovolj električne napetosti in toka. Gonilniki potrebujejo krmilnik, ki jim pošilja točno toliko korakov, kot jih prejme iz računalnika. Poznamo bipolarne in unipolarne koračne motorje. Razlika med njimi je v številu priključkov na tuljave oz. številu žic.

Odločili smo se za bipolarne motorje, saj imajo le 4 žice za krmiljenje in lahko proizvedejo večji navor kot unipolarni. Krmiliti jih je težje, saj potrebujejo tok, ki teče v nasprotno smer (reverse current). Po natančnem premisleku in primerjanju tipov motorja s tistimi, ki so bili podani v originalni datoteki, smo se odločili za naslednje motorje:

Tabela 7: Podatki koračnih motorjev

(Vir: osebni arhiv)

Tip motorja	Tok	Obratovalna napetost	Dimenzije [mm]	Št.kosov
NEMA 23	3.0A	24V	57x57x113	2
NEMA 17	2.0A	24V	48x42x42	1
NEMA 17 1:5	1.6A	24V	42.3x42.3x67.3	1
NEMA 14	0.8A	24V	35.2x35.2x34	1
NEMA 17	2.1A	24V	42x42x60	1

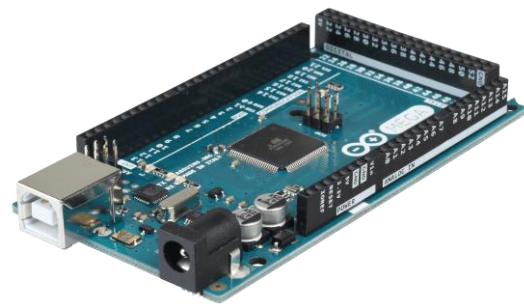
3.2 IZBIRA KRMILNIKA IN NADZORNEGA VEZJA

Ker smo si zastavili cilj, da naša robotska roka deluje kot 3D-tiskalnik, smo se morali odločiti za primeren krmilnik. Po premisleku in po posvetovanju s profesorji smo se odločili za mikrokrmilnik RAMPS 1.4, saj ga pogosto uporabljajo v doma izdelanih 3D-tiskalnikih, prav tako je zelo dobro ocenjen in omogoča krmiljenje s pomočjo Arduino MEGA, na katerega je nameščen RAMPS 1.4. Arduino MEGA nam služi kot vmesnik, na katerem so določeni vsi parametri, da lahko Ramps 1.4 prepozna vse koračne motorje. Definirani so vsi priključki za step, dir, enable in hitrosti. Na Arduino je praktično nameščen program MARLIN, vendar smo morali spremeniti število "ekstrudorjev" iz 1 na 2, saj bomo imeli četrto in peto os priključeno na osi za ekstrudorje. To smo morali v programu tudi definirati. Spremeniti smo morali tudi tip matične plošče, in sicer iz številke 33 v 34.



Slika 17: Ramps 1.4

(Vir: https://reprap.org/wiki/RAMPS_1.4)



Slika 18: Arduino Mega

(Vir: <https://www.arduino.cc/en/Guide/ArduinoMega2560>)

```
// The following define selects which electronics board you have.
// Please choose the name from boards.h that matches your setup
#ifdef MOTHERBOARD
  #define MOTHERBOARD 33
#endif

// Define this to set a custom name for your generic Mendel,
// #define CUSTOM_MENDEL_NAME "This Mendel"

// Define this to set a unique identifier for this printer, (Used by some programs to differentiate between machines)
// You can use an online service to generate a random UUID. (eg http://www.uuidgenerator.net/version4)
// #define MACHINE_UUID "00000000-0000-0000-0000-000000000000"

// This defines the number of extruders
#define EXTRUDERS 1
```

Slika 19: Število ekstrudorjev in tip matične plošče

(Vir: osebni arhiv)

```

// The following define selects which electronics board you have.
// Please choose the name from boards.h that matches your setup
#ifdef MOTHERBOARD
  #define MOTHERBOARD 34
#endif

// Define this to set a custom name for your generic Mendel,
// #define CUSTOM_MENDEL_NAME "This Mendel"

// Define this to set a unique identifier for this printer, (Used by some programs to differentiate between machines)
// You can use an online service to generate a random UUID. (eg http://www.uuidgenerator.net/version4)
// #define MACHINE_UUID "00000000-0000-0000-0000-000000000000"

// This defines the number of extruders
#define EXTRUDERS 2

```

Slika 20: Potrebne spremembe parametrov za našo robotsko roko

(Vir: osebni arhiv)

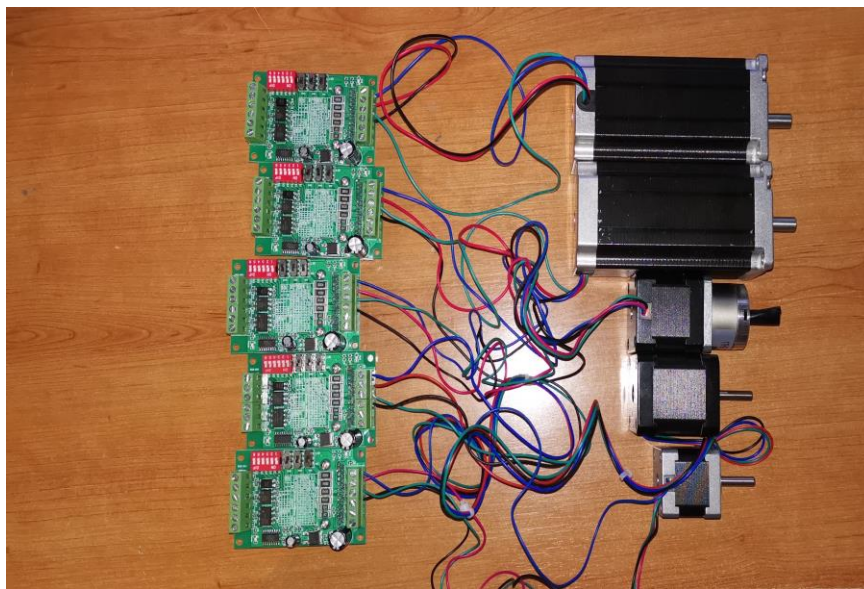
Ko smo se odločili za mikrokrmilnik in motorje, smo jih naročili preko spleta in jih po manjših zapletih tudi dobili. Nato smo lahko začeli izdelovati prva električna vezja.

3.3 RAMPS 1.4 in PROGRAM

Ramps 1.4 je mikrokrmilnik, ki skrbi, da koračni motorji delujejo po oseh, ki smo jih definirali z mestom priključitve gonilnikov. Namesti se na Arduino MEGA, saj ima le-ta vse potrebne priključke za Ramps. Ramps nato organizira podatke in jih usmeri v določene osi, na katere so priključeni motorji. MEGA je samo zalogovnik parametrov, ki povedo Rampsu, kaj narediti s podatki, ki jih dobi iz računalnika, npr. kako hitro naj se nekaj premakne, kam naj se premakne, kako naj se premakne; določene so vse spremenljivke. Vendar MEGA nima dovolj moči za obdelavo vseh podatkov, ki jih dobi iz računalnika. Z združitvijo obeh dobimo sistem, ki nam omogoča premikanje osi. Nanj lahko priklopimo 5 osi, tj. X, Y, Z, E0, in E1. E0 in E1 sta osi za ekstruderje, ki so člani v 3D-tiskalnikih. Uporabljajo se za podajanje materiala do konice, ki ta material stali in nanese na delovno površino. E0-os imajo definirano vsi 3D-tiskalniki, E1 pa samo tisti tiskalniki, ki imajo možnost tiskanja z dvema različnima materialoma. Največkrat se dva ekstruderja uporabljata za večbarvne izdelke. Ramps se uporablja večinoma v doma izdelanih 3D-tiskalnikih, saj omogoča enostavno krmiljenje in je dokaj poceni. Nanj smo lahko kasneje tudi priključili servomotor. Za vsako os ima razpisane priključke – kam priključiti enable, step, direction in ground iz gonilnika. Ko smo Ramps namestili na Arduino, je bil naslednji korak izbrati programsko opremo za našo robotsko roko. Odločili smo se za MARLIN, odprtokodno programsko opremo, ki izvaja kodirane ukaze in upravlja z vsemi dejavnostmi robotske roke, vključno s pošiljanjem koordinat gibanja koračnim motorjem. MARLIN podpira številne krmilne plošče, med

drugimi tudi Ramps 1.4. Na voljo ima tudi svoje G-kode, s katerimi premikamo in programiramo robotsko roko. Ko smo spremenili vse potrebno na programu, smo ga namestili na Arduino Mega z računalniškim programom Arduino IDE. Na spletni strani, kjer je predstavljen Ramps 1.4, je objavljena skica, ki kaže vse definirane priključke. S pomočjo te skice smo priključili step, direction in enable priključke iz gonilnikov.

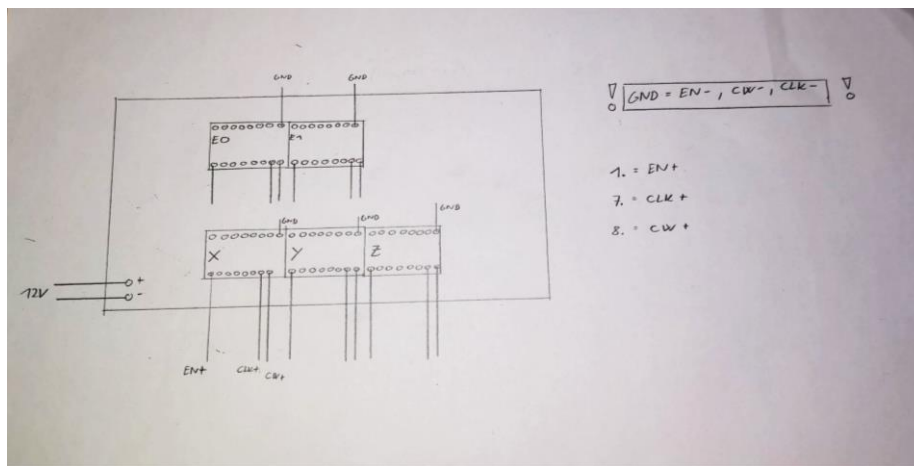
Sledi logično zaporedje. Koračni motorji so priključeni vsak na svoj gonilnik, kar je pogoj, da lahko obratujejo. Gonilniki so priključeni na 24 V DC. Mikrokrmilnik Ramps 1.4 je priključen na 12 V DC. Uporabljamo gonilnike TB6560, ki imajo obratovalno napetost 0–32 V, prenašajo pa tokove do 3.0 A, kar je dovolj za našo robotsko roko, saj največji motorji potrebujejo 3.0 A. Nastaviti smo morali vse potrebne tokove po diagramu, ki so narisani na gonilnikih. Če na gonilnikih ne bi nastavili maksimalnih tokov, le-ti ne bi prenesli tokov in bi se uničili. Morali smo biti pozorni, saj sta dva motorja, ki sta odgovorna za Z-os, obrnjena drug proti drugemu in se mora eden izmed njiju vrteti v nasprotno smer. To smo preprosto rešili z zamenjavo dveh žic tuljav. Servomotorju, ki je odgovoren za prijemalo, smo morali zagotoviti zunanji vir napajanja 6 V in ga nato povezati na Ramps priključke.



Slika 21: Koračni motorji

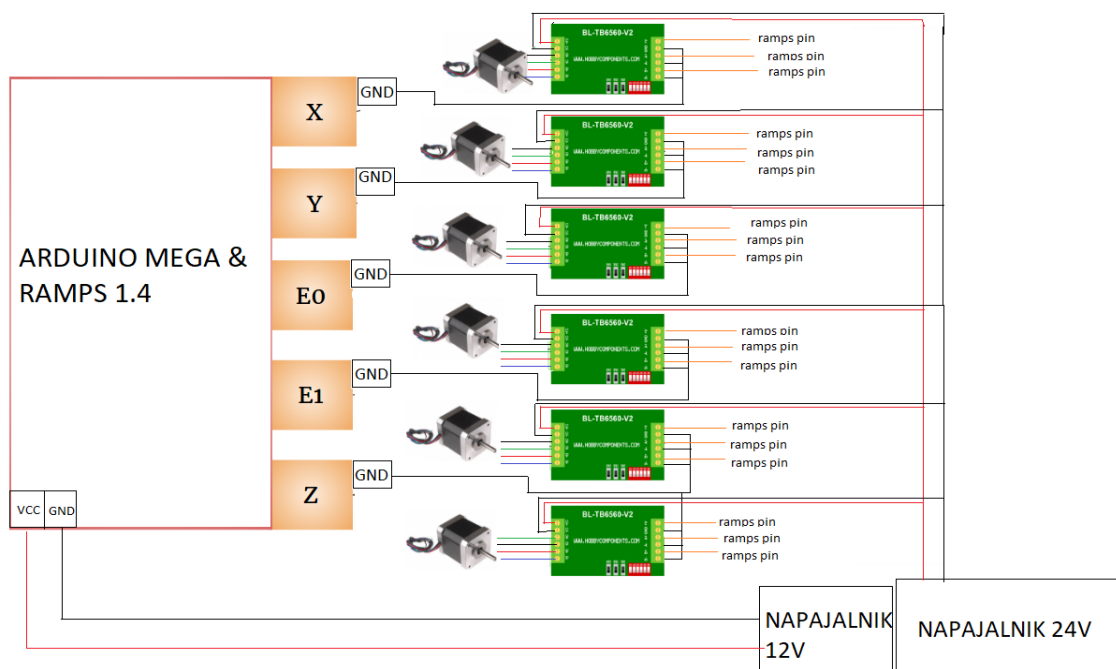
(Vir: osebni arhiv)

Vezava, ki smo jo sestavili za priključitev gonilnikov in motorjev na RAMPS, izgleda tako:



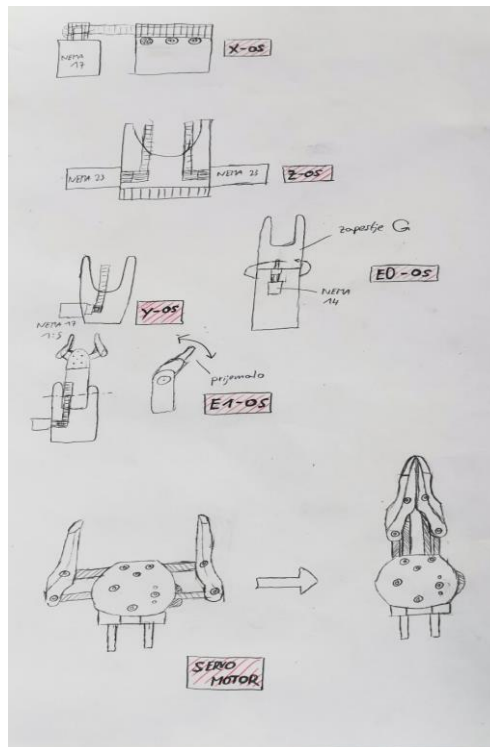
Slika 22: Skica vezave za gonilnike

(Vir: osebni arhiv)



Slika 23: Vezava iz gonilnikov na Ramps priključke

(Vir: osebni arhiv)



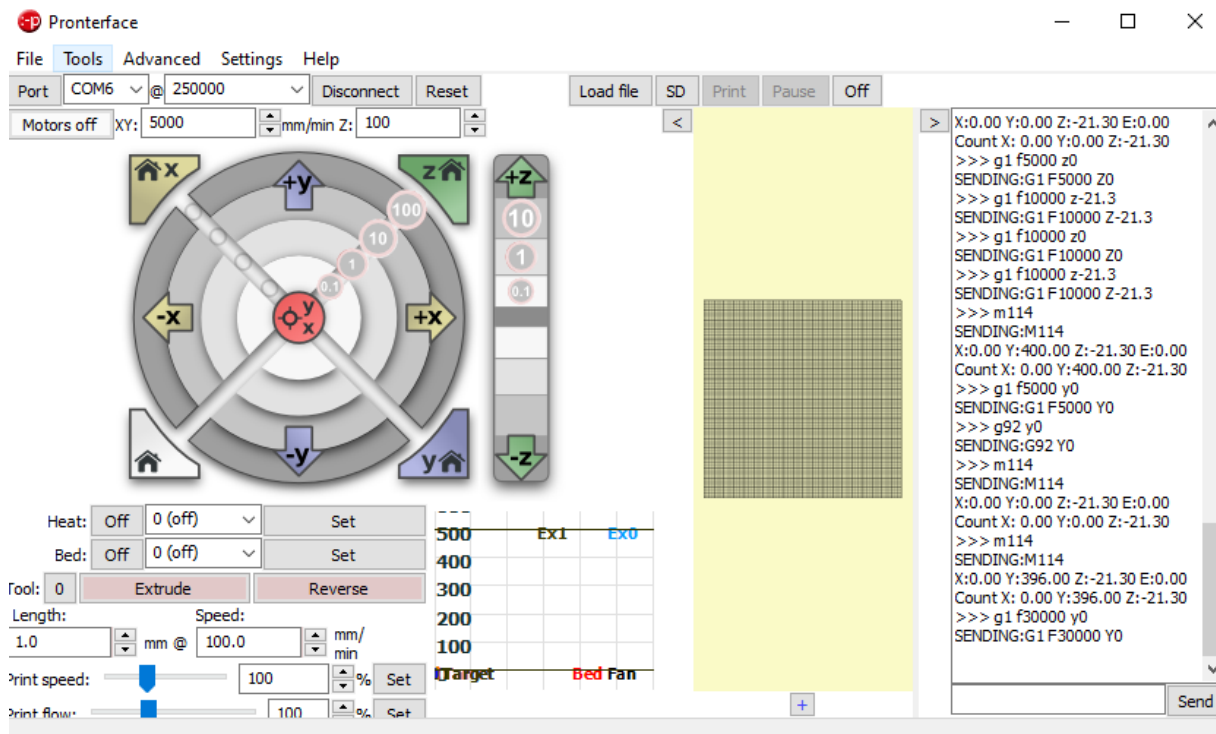
Slika 24: Skica definiranja motorjev na posamezno os

(Vir: osebni arhiv)

Uporabljali smo program Pronterface, s katerim lahko krmilimo 3D-tiskalnik. Ker imamo krmilnik za 3D-tiskalnik in Arduino MEGA, lahko robotsko roko krmilimo in premikamo s pomočjo tega Softwarea. Na ta program je mogoče naložiti tudi datoteko, ki vsebuje G-kodo. Sprva smo robota premikali s premikanjem posameznih osi, da smo videli, kako hitro se odziva, kako natančna je posamezna os in kakšne hitrosti dosega. Nato pa smo sestavili svojo G-kodo in jo naložili na program Pronterface. Koda je napisana tako, da se robotska roka premika relativno. Torej, vsaka točka je relativno od zadnje točke. Odločili smo se lahko med absolutnim in relativnim načinom krmiljenja. Kasneje smo G-kodo spremenili, da se robotska roka premika absolutno, ker lahko njeno ničlo (izhodišče koordinatnega sistema robotske roke) določimo sami. Uporabili smo ukaz G1, ki nam omogoča, da motorjem dodelimo hitrost, s katero se bodo gibali. Ugotovili smo, da se nekateri motorji gibljejo hitreje od drugih pri isti definirani hitrosti. Vzrok tega so različne stopinje na korak motorja, a jih kljub temu lahko premikamo skupaj, saj vsaka os posebej izračuna potrebno hitrost. Tako je motor, ki ima vgrajen zobniški prenos, počasnejši, a je zato mnogo močnejši. Podali smo optimalne hitrosti in nato izmerili natančnost robotske roke.

4 PROGRAMIRANJE

Ko 3D-tiskalnik preneha tiskati, se za določen čas ustavi. Medtem robotska roka seže na mizo 3D-tiskalnika, prime izdelek in ga odloži v zabojnik. Da to dosežemo, mora biti 3D-tiskalnik povezan z robotsko roko, program Pronterface pa mora točno vedeti, kdaj naj se program robota začne. Sprva smo nameravali za krmilnik uporabiti še en Arduino UNO, saj bi nanj lahko naložili G-kodo in izvedli program, ko bo čas za to. Vendar smo se vprašali, kako bi lahko Arduino UNO bral "surovo" G-kodo brez pomoči določenega programa na računalniku. Rešitve nismo našli, saj je nastopil še dodatni problem. Programska oprema Marlin ima v Arduino programu zapisane že vse potrebne parametre za komunikacijo s programom Pronterface. Ker je ta program zelo velik – vsebuje preko 20 knjižnic in ogromno kod – smo to možnost zavrgli in se odločili, da bomo imeli poleg računalnik, na katerem bo odprt Pronterface, ki bo bral napisano G-kodo in jo posredoval robotski roki.



Slika 25: Program Pronterface

(Vir: osebni arhiv)

4.1 PROGRAMIRANJE TISKALNIKA

Naša prva ideja je bila, da na začetku podlage robotske roke namestimo končno stikalo, ki bi bilo v višini mize 3D-tiskalnika. Tiskalniku bi spremenili G-kodo, tako da bi se po končanem tiskanju njegova miza prestavila vse do končnega stikala na robotu. Tako bi sprožila stikalo, ki bi dal signal robotski roki, da začne opravljati svojo nalogo. Medtem bi v G-kodi dodali še ukaz, da tiskalnik počaka za določen čas, da robotska roka opravi svojo nalogo. Ko jo bo končala, se bo začelo tiskanje novega izdelka. Na koncu smo se odločili za časovno zakasnitev. Glede na čas tiskanja izdelka bi spremenili, kako dolgo robotska roka čaka. Tako se tiskalniku ni potrebno dotikati podlage robota, poleg tega pa je lahko na katerikoli strani robota.

```
G28 X0 Y0 ; home x and y axis
G1 Y300 ; move bed forward
M106 S0 ; turn off cooling fan
G4 S300 ; wait for 5 minutes
M23 CCR-10_valj_v1.gcode|
M24
;End of Gcode
```

Slika 26: Končni del G-kode tiskalnika

(Vir: osebni arhiv)

Ukaz G28 nam X- in Y-os postavi v začetno pozicijo. Z G1 Y300 postavimo mizo za 300 mm naprej, s čimer približamo izdelek robotski roki. M106 S0 pomeni, da se ventilatorji izključijo, saj mora tiskalnik ostati vroč, da lahko takoj nadaljuje z delom. G4 S300 je ukaz, ki ustavi tiskalnik za 5 minut. M23 in M24 pa izbereta enak izdelek, ki ga hočemo ponovno natisniti. Program za izdelek je shranjen na SD-kartici, ki je v tiskalniku, s temi ukazi pa izberemo ime programa. V našem primeru je to CCR-10_valj_v1.gcode.

4.2 PROGRAMIRANJE ROBOTSKÉ ROKE

Programiranje robotske roke bo delovalo na principu premikanja motorjev in branja koordinat. Sprva bomo nastavili novo ničelno točko vseh osi na željnem mestu in to potrdimo z ukazom G92 X0 Y0 E0 Z0.

```

>>> M114
SENDING:M114
X:0.00 Y:-140.00 Z:-
3.30 E:0.00 Count X:
0.00 Y:-140.00 Z:-3.30
>>> G92 X0 Y0 E0 Z0
SENDING:G92 X0 E0 Y0
Z0
>>> M114
SENDING:M114
X:0.00 Y:0.00 Z:0.00
E:0.00 Count X: 0.00
Y:0.00 Z:0.00

```

Slika 27: Postopki programiranja robotske roke

(Vir: osebni arhiv)

Slika zgoraj kaže, kako smo robotsko roko pripeljali do željene pozicije ničelne točke (1). Nato smo z ukazom G92 X0 Y0 E0 Z0 potrdili, da so zdaj koordinate vseh osi 0. (2). Sledilo je ponovno preverjanje, ali so se osi res nastavile na 0. (3) Potem s te točke ročno premaknemo določeno os in z ukazom M114 izpišemo trenutne koordinate vseh osi ter si jih shranimo. Nato po istem principu nadaljujemo z vsemi osmi, dokler ni prijemalo tam, kjer želimo. Ko smo si vse koordinate zapisali, lahko sestavimo G-kodo.


```

>>> m114
SENDING:M114
X:0.00 Y:-200.00 Z:-5.20 E:-9.50 Count X: 0.00 Y:-200.00 Z:-5.20

```

Slika 28: Premikanje vsake osi posebej

(Vir: osebni arhiv)

 *Koordinate za programiranje.txt – Beležnica

Datoteka Uredi Oblika Pogled Pomoč

```

Y: -200
Z: -5.20
E1: -5
X: 0
E0: -4.5

```

Slika 29: Shranjevanje koordinat osi

(Vir: osebni arhiv)

Celotnega postopka pisanja kode ne bomo avtomatizirali, saj bomo imeli en program za veliko izdelkov, saj je to serijska izdelava določenih modelov. Ko napišemo G-kodo (lahko kar v navadni beležnici), datoteko naložimo v Pronterface kot format .gcode. Nato izvedemo samo ukaz print, ki zažene celoten program. Naleteli smo na težavo, saj ne moremo krmiliti več osi skupaj. Če se osredotočimo na G-kodo, lahko skupaj premaknemo le Z- in Y-os, ker so njune hitrosti približno enake. Če imamo hitrost Y-osi, ki ima motor z reduktorjem 5000, se bo Z-os še vedno premikala z optimalno hitrostjo. Torej se bosta premikali podobno. Če pa hočemo sočasno premakniti Z-, Y- in X-os, pa nastopi težava, saj X-os potrebuje manjše hitrosti, da obratuje normalno. Hitrost X-osi je za Y- in Z-os veliko premajhne, zato se bosta premikali izredno počasi. Težava je v tem, da so v Marlinu v zavihku configuration.h določeni različni parametri za razdaljo na korak. Torej Z-os premaga manj razdalje na korak kot npr. E0-os. Potrebno bi bilo vse na novo preračunati, zato tega nismo počeli in obdržali naše parametre. Tako lahko hkrati premikamo le Z- in Y-os.

```

//// MOVEMENT SETTINGS
#define NUM_AXIS 4 // The axis order in all axis related arrays is X, Y, Z, E
#define HOMING_FEEDRATE {50*60, 50*60, 4*60, 0} // set the homing speeds (mm/min)

// default settings

#define DEFAULT_AXIS_STEPS_PER_UNIT {128.5,45.5,206.8019,760*1.1} // default steps per mm
#define DEFAULT_MAX_FEEDRATE {500, 500, 500, 25} // (mm/sec)
#define DEFAULT_MAX_ACCELERATION {60, 100, 5, 10000} // X, Y, Z, E maximum start speed for accelerated moves.

```

Slika 30: Razdalja na korak za posamezne osi

(Vir: osebni arhiv)

G-koda robotske roke bo imela na začetku ukaz G4 P (čas v milisekundah) za zakasnitev delovanja robotske roke. Ker v Marlin in Ramps ne moremo pisati pogojev, kot so If stavek pri programiranju, nimamo druge možnosti. 3D-tiskalnik pred začetkom tiskanja izračuna približen čas tiskanja. Ti časi so precej natančni, zato se lahko tudi orientiramo po njih. Zakasnitev bo trajala nekaj časa več kot predviden čas tiskanja. G-koda bi izgledala približno tako:

```

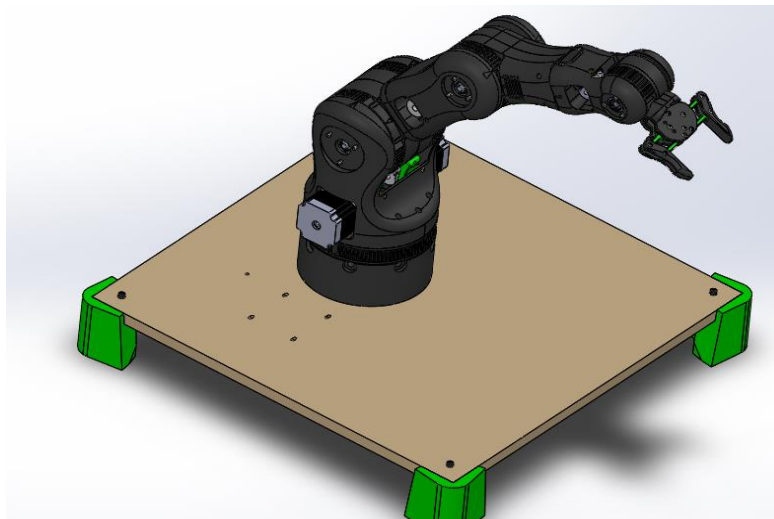
strega 3D-tiskalnika.gcode - Beležnica
Datoteka Uredi Oblika Pogled Pomoč
G90; Absolutni način
G4 P600000; Robotska roka čaka toliko, kolikor traja tiskanje
G1 F5000 Z10
G1 F10000 Y150
M280 P4 S45; Zgrabi izdelek
G1 F10000 Y-120
G1 F500 X-60
G1 F10000 Y120
M280 P4 S0; Odloži izdelek
G1 F5000 Z0; Z-os gre v začetno pozicijo
G1 F10000 Y0; Y-os gre v začetno pozicijo
G1 F500 X0; X-os gre v začetno pozicijo

```

Slika 31: G-koda robotske roke

(Vir: osebni arhiv)

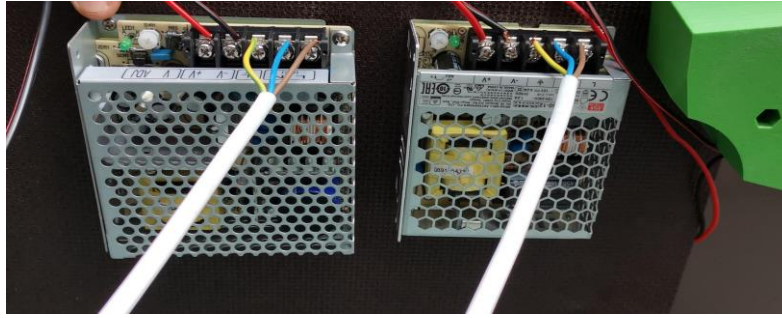
Ko smo vse elemente zvezali in preizkusili njihovo delovanje, je sledila montaža na leseno ploščo. Nanjo smo izvrtali luknje ter z vijaki in maticami pritrdili prvi sklop robotske roke. Nato smo roko namestili še na ta sklop.



Slika 32: Vijachenje robota na ploščo

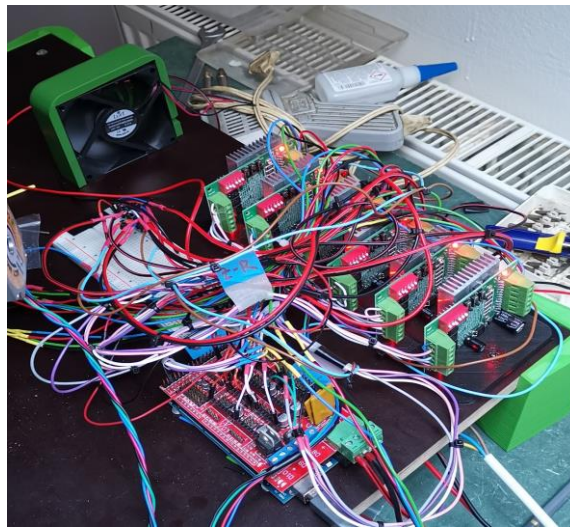
(Vir: osebni arhiv)

Potem smo izvrtali luknje, da bomo lahko motor, ki je odgovoren za vrtenje celotnega robota, premikali in »zategovali« jermen, ki se bo čez čas raztegnil. Lahko bi dodali zategovalnik jermena, ki bi ga zmodelirali sami, ampak smo se odločili za to rešitev, saj je veliko hitrejša in enostavnejša. Natisnili smo pokrov, ki pokrije vse gonilnike, in jih tudi privijačili. Nato smo vse potrebne žice iz gonilnikov usmerili iz pokrova in jih pritrdili z vezicami. V nadaljevanju smo vse povezali na mikrokrmilnik Ramps 1.4. Napajalnike smo privijačili na spodnjo stran lesene plošče, saj tako zavzamejo čim manj prostora.



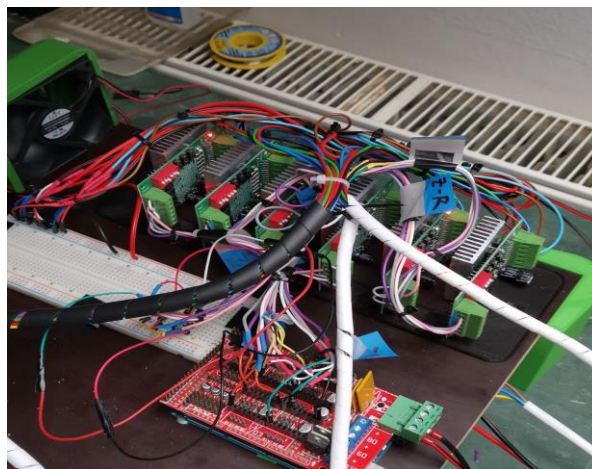
Slika 33: Napajalnika na spodnji strani lesene plošče

(Vir: osebni arhiv)



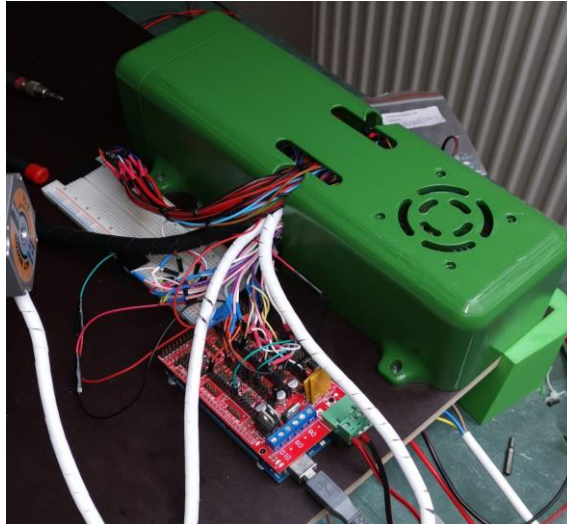
Slika 34: Neurejeni kabli motorjev in gonilnikov

(Vir: osebni arhiv)



Slika 35: Urejeni kabli z vezicami

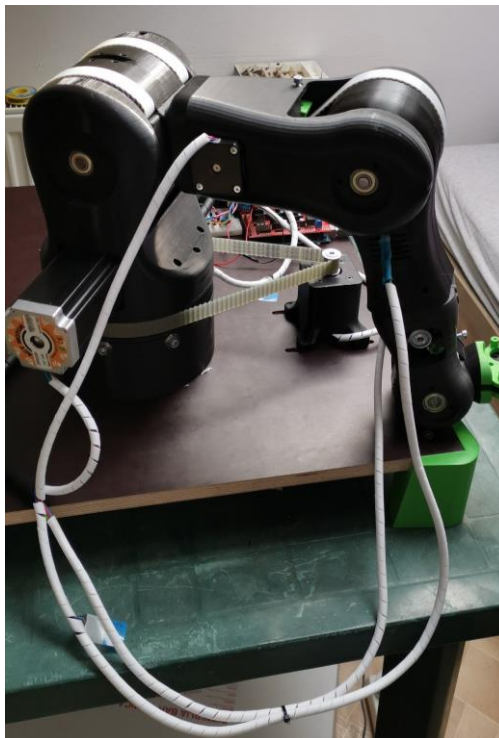
(Vir: osebni arhiv)



Slika 36: Nameščen pokrov

(Vir: osebni arhiv)

V pokrov smo namestili še 24 V ventilator, ki skrbi za hlajenje gonilnikov v ohišju. Vse kable motorjev smo pospravili v svedrasto kabelsko zaščito. Nato smo motor in jermen X-osi pokrili, da slednji ne pride v kontakt s kablji ali s človekom.

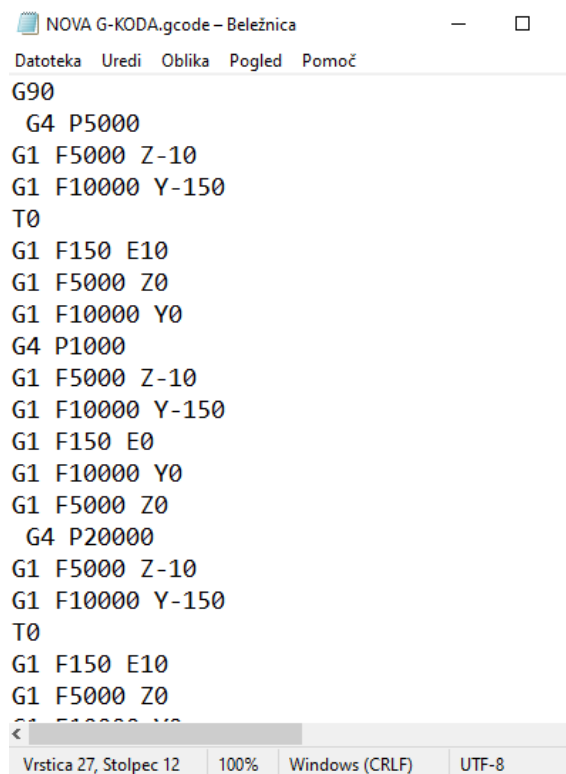


Slika 37: Celoten robot z zaščitnimi kablji

(Vir: osebni arhiv)

5 MERJENJE NATANČNOSTI ROBSKE ROKE

Merjenje natančnosti smo izvedli tako, da smo robotsko roko prestavili v neko izhodišče in pod prijemalo postavili predmet. Roko smo nastavili tako, da je bilo med njo in predmetom točno 0.5 mm zraka. Nato smo napisali kodo, da je robotska roka premikala vse osi v različne smeri in po nekaj gibih prišla nazaj v začetno izhodišče. Ko se je program zaključil, smo izmerili razdaljo med predmetom in robotsko roko ter jo primerjali z začetno razdaljo. Kodo smo sestavili tako, da smo napisali, kaj bo robotska roka delala in nato vse kopirali 14 x, da smo dobili eno celotno G-kodo, po kateri se je robotska roka ravnila. Tako smo lahko videli ponovljivost robotske roke. G-koda je izgledala tako:



```
NOVA G-KODA.gcode - Beležnica
Datoteka  Uredi  Oblika  Pogled  Pomoč
G90
  G4 P5000
G1 F5000 Z-10
G1 F10000 Y-150
T0
G1 F150 E10
G1 F5000 Z0
G1 F10000 Y0
G4 P1000
G1 F5000 Z-10
G1 F10000 Y-150
G1 F150 E0
G1 F10000 Y0
G1 F5000 Z0
  G4 P20000
G1 F5000 Z-10
G1 F10000 Y-150
T0
G1 F150 E10
G1 F5000 Z0
G1 F10000 Y0
<
Vrstica 27, Stolpec 12    100%    Windows (CRLF)    UTF-8
```

Slika 38: G-koda za merjenje natančnosti

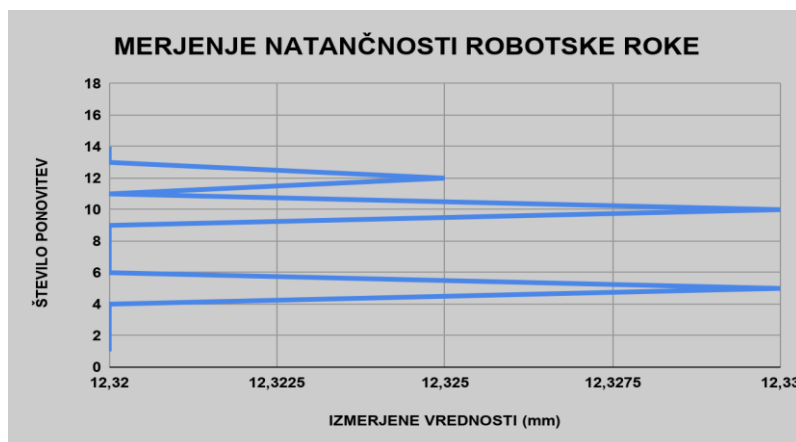
(Vir: osebni arhiv)

Rezultati so nas presenetili, saj nismo predvidevali, da bo roka tako natančna. Opazili smo, da je bil vmesni rezultat dvakrat 12.33 mm in enkrat 12.325 mm, a verjetno je to zgolj napaka merjenja, pri katerem se zlahka zmotimo za 0.01 mm. Zaradi tega smo opravili 15 meritev.

Natančnost
12.32mm ORIGINAL
1. meritev = 12.32mm
2. meritev = 12.32mm
3. meritev = 12.32mm
4. meritev = 12.32mm
5. meritev = 12.33mm
6. meritev = 12.32mm
7. meritev = 12.32mm
8. meritev = 12.32mm
9. meritev = 12.32mm
10. meritev = 12.33mm
11. meritev = 12.32mm
12. meritev = 12.325mm
13. meritev = 12.32mm
14. meritev = 12.32mm

Slika 39: Rezultati merjenja natančnosti roke

(Vir: osebni arhiv)



Graf 1: Merjenje natančnosti robotske roke

Zgornji graf prikazuje rezultate 15 meritev natančnosti robotske roke. Na njem lahko vidimo, da je odstopanje pri meritvah le 0,1 mm.

6 MERJENJE NOSILNOSTI ROBSKE ROKE

Merjenja nosilnosti robotske roke smo se lotili tako, da smo si priskrbeli uteži posameznih mas. Uporabili smo 100 g, 145 g, 400 g in 1 kg uteži. Potem smo s prijemalom prijeli za utež in jo počasi začeli dvigovati. Naš cilj je doseči čim večjo nosilnost, pri kateri robot še lahko normalno obratuje. Izmerili smo tudi hitrost pri določeni masi in to zapisali. Rezultati so bili pričakovani, saj smo predvidevali, da najšibkejši motor, tj. 0.8 A, ki skrbi za dvigovanje zapestja, ne bo veliko zdržal. Kasneje smo na tej osi poiskali točko, na kateri zavora motorja drži in nadaljevali s preizkušanjem ostalih osi. Ugotovili smo, da vse osi, razen osi E1, delujejo normalno. Osi niso pokazale nobene spremembe pri hitrosti, zvoku in stabilnosti robota. Kot je razvidno na spodnji sliki, os E1 lahko dvigne le 145 g, ostale osi pa lahko dvignejo 1000 g.

Nosilnost				
100g	E1 OK, Y OK, Z OK, X OK, E0 OK			
145g	- E1 OK, Y OK, Z OK, X OK, E0 OK			
200g	- E1 Ne dvigne in ne obdrži bremena, Y OK, Z OK, X OK, E0 OK			
400g	- E1 Ne dvigne in ne obdrži bremena, Y OK, Z OK, X OK, E0 OK			
1000g	- E1 Ne dvigne in ne obdrži bremena, Y OK, Z OK, X OK, E0 OK			

Slika 40: Rezultati nosilnosti robotske roke

(Vir: osebni arhiv)

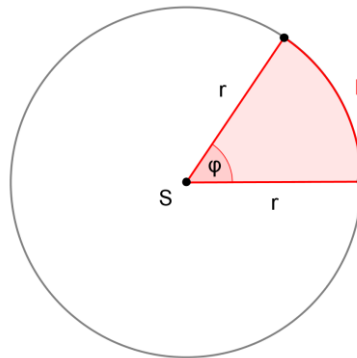


Slika 41: Merjenje nosilnosti robotske roke

(Vir: osebni arhiv)

7 MERJENJE HITROSTI ROBSKE ROKE

Zanima nas tudi, s kakšno največjo hitrostjo lahko roka izvaja svojo nalogo. Naša ideja je bila, da na list papirja narišemo dve točki, med katerima je določena razdalja. Nato bomo robotsko roko približali točki in napisali program, da roka doseže drugo točko. Ta proces bomo merili s štoparico in s preprostim računom izračunali hitrost robotske roke. Kot smo že omenili, robotske roke ne moremo premikati z isto hitrostjo, zato bomo morali hitrost delovanja izračunati za vsako os posebej. Ker roka potuje v smeri krožnice, se bomo morali dela lotiti nekoliko drugače. Vsi poznamo krožni lok. To je razdalja, ki jo premaga neka točka pri krožnem gibanju.



Slika 42: Krožni lok

(Vir: https://si.openprof.com/wb/krog_in_kro%C5%BEnica?ch=131)

Če imamo osrednji kot in polmer od središča, lahko po enačbi $2\pi r \frac{\alpha}{360^\circ}$ izračunamo dolžino krožnega loka oz. kolikšno razdaljo premaga končna točka. Robot smo postavili v poljubni položaj in nato izmerili razdaljo od središča tiste osi, ki jo želimo meriti, do končne točke (točke, katere hitrost želimo izračunati). V našem primeru bo to središče osi prijemala. Ker se prijemalo lahko dvigne oz. spusti, os pa ostane fiksna, bo tudi hitrost konstantna. Torej smo za vse osi merili hitrosti pri največjem polmeru. Začeli smo z X-osjo. Poiskali smo središče X-osi in izmerili razdaljo do središča željene osi. Razdalja je ista kot pri Z-osi, zato bo polmer pri Z- in X-osi isti. Rezultati so sledeči:

Hitrost

MAX hitrost X-osi

Krožni lok = 696mm

polmer = 443mm

čas = 1.96s

kot = 90°

Hitrost = 0.355 m/s

MAX hitrost Z-osi

krožni lok = 696mm

kot 90°

polmer = 443mm

čas = 3.84s

Hitrost = 0.18 m/s

MAX hitrost Y-osi

kot 90°

polmer = 223mm

krožni lok = 350mm

čas = 3.80s

Hitrost = 0.09 m/s

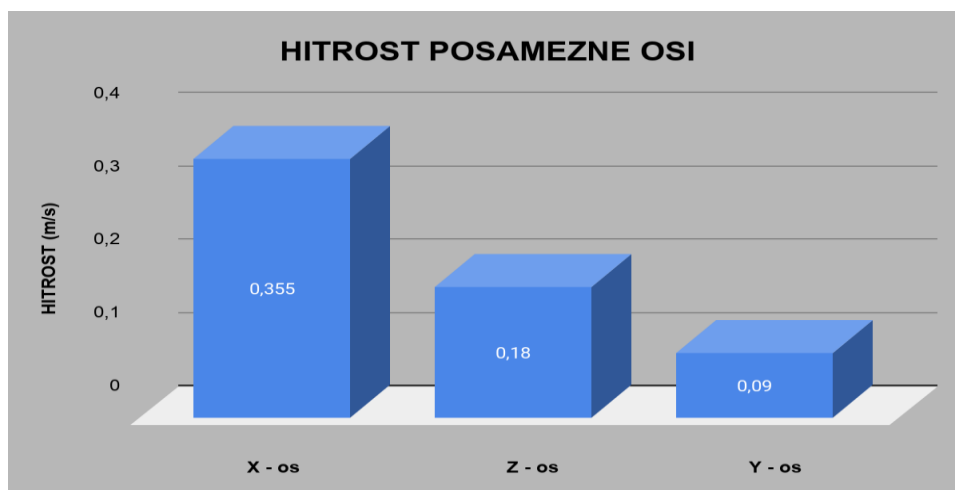
Slika 43: Rezultati hitrosti posameznih osi

(Vir: osebni arhiv)

Iz analize podatkov lahko v naslednjem grafu razberemo, da je X-os mnogo hitrejša od ostalih osi, saj je v konfiguraciji programske opreme definirana večja razdalja na korak kot na drugih oseh.

Graf 2: Hitrosti posameznih osi

(Vir: osebni arhiv)







8 PREDSTAVITEV PROBLEMATIKE STREGE 3D-TISKALNIKA

Pri stregi 3D-tiskalniku se pojavi kar nekaj težav. Najprej bomo opisali težavo s povezovanjem 3D-tiskalnika z robotsko roko. Naša prva rešitev je bilo končno stikalo, ki bi pognalo robotsko roko. Miza tiskalnika bi se po končanem tiskanju pomaknila do senzorja in tako aktivirala robota. Če bi stikalo definirali kot končno stikalo ene izmed osi, bi lahko za pogoj aktivacije enostavno uporabili ukaz HOME določene osi v G-kodi. To pomeni, da se mora os postaviti v začetno pozicijo. Če vzamemo za primer X-os, torej uporabimo HOME X, bi se eden izmed koračnih motorjev vrtel, dokler ne bi pritisnili na končno stikalo. Če se izdelek tiska pol ure, bi se motor vrtel pol ure. Zaradi omejenega vrtenja pa bi se robot zaletel sam vase že po 10 sekundah, zato to ni izvedljivo. Če bi lahko v MARLIN-ovi G-kodi uporabljali pogoje, kot je npr. if pogoj, bi robotska roka čakala na izpolnjen pogoj, podobno kot ukaz HOME.

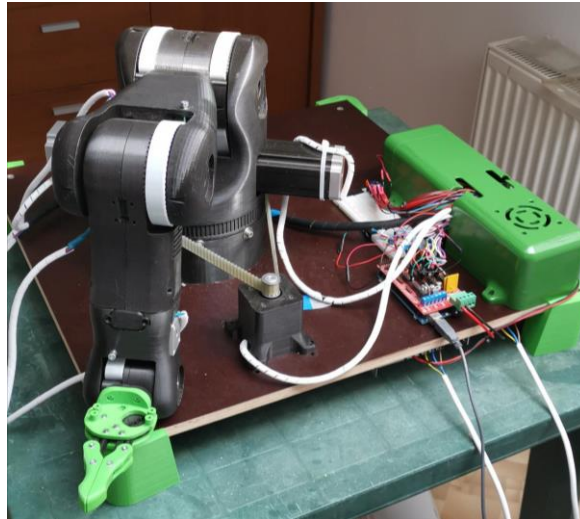
Naslednja težava se pojavi pri odstranjevanju izdelkov. Iz lastnih izkušenj in testiranj smo opazili, da večje predmete težje odstranimo s tiskalne podlage kot manjše. Za primer smo uporabili valj, ki je enostavne oblike in ima majhno površino tiskanja. Tiskali smo na steklo, ki je gladko in se zato izdelek manj prime. Če je steklo preveč ali premalo odmaknjeno od konice tiskalnika, je velika verjetnost, da se material ne bo prijel podloge in bo zato izdelek neuspešen. Robot je dovolj močan, da valj zagradi in ga odstrani. Če pa bi tiskali večje in bolj zapletene oblike, bi imeli težave.

9 POTRDITVE HIPOTEZ

1.	S tehnologijo 3D prostorskega tiska je možno izdelati 6-osno robotsko roko.	
2.	Robotska roka ima nosilnost 1 kg.	
3.	Robotska roka lahko doseže delovno hitrost 0.15 m/s.	
4.	Robota lahko uporabljamo za strego 3D-tiskalniku.	

1. S tehnologijo 3D prostorskega tiska je možno izdelati 6-osno robotsko roko.

To hipotezo lahko potrdimo. Izkazala se je za pravilno, saj smo uspeli izdelati 6-osno robotsko roko s tehnologijo 3D prostorskega tiska. Robotska roka deluje in je funkcionalna. Ponavadi so robotske roke narejene iz kovin, a se je plastika izkazala za konkurenčno, seveda v tej velikosti robotske roke. Pri težjih, npr. 2 toni težkih robotih plastika ne pride v poštev, ker ne bi zdržala tako velikih obremenitev.



Slika 44: 6-osna robotska roka

(Vir: osebni arhiv)

2. Robotska roka ima nosilnost 1 kg.

Hipotezo moramo ovreči, saj je 1 kg preveč za najmanjši motor, ki je odgovoren za dvig zapestja. Vsi motorji so samodržni, kar pomeni, da sami držijo pozicijo, kjer se ustavijo. Vendar je ta "zavora" odvisna od velikosti toka in moči motorja. Vsi ostali motorji brez težav držijo bremena, saj so veliko močnejši od ostalih. Vendar je prenašanje brez četrte osi oz. osi zapestja nemogoče, ker moramo sami nadeti breme tja, kjer motor še zdrži.

3. Robotska roka lahko doseže delovno hitrost 0.15 m/s.

To hipotezo lahko v celoti potrdimo, saj smo vse preizkusili in naredili tudi vse potrebne meritve. Vendar je vprašljivo, kako lahko potrdimo hipotezo, če so meritve manjše od predpostavljene. Vzrok je v tem, da je za Z-, Y- in X-osi v konfiguraciji programske opreme MARLIN definiran mehek pospešek in mehko zaviranje. Hitrosti smo merili na razdalji osrednjega kota, ki je pri vseh

meril 90°. Ta razdalja je bila za vse osi prekratka, da bi razvile večje hitrosti. Torej lahko potrdimo, ker vemo, da lahko razvijejo višje hitrosti, če bi bila razdalja večja. 90° smo vzeli zato, ker bo to povprečna razdalja, ki jo bo robot dosegal pri obratovanju. Pot zaviranja in pot pospeševanja pri Y-osi in Z-osi se s hitrostjo večata, saj sta veliki osi in potrebujeta več časa, da se zaustavita. To še posebej velja za Z-os. Rezultati za 180° so sledeči:

180°		180°		180°	
X-os		Y-os		Z-os	
Krožni lok = 139.17cm polmer = 443mm		Krožni lok = 70cm polmer = 223mm		Krožni lok = 139.17cm polmer = 443mm	
čas = 2.65s		čas = 4.0		čas = 5.61s	
Hitrost = 0.525 m/s		Hitrost = 0.155 m/s		Hitrost = 0.24808 m/s	

Slika 45: Hitrosti posameznih osi pri 180°

(Vir: osebni arhiv)

4. Robota lahko uporabljamo za strego 3D-tiskalniku.

Robotsko roko je mogoče programirati, kakor želimo. Ima tudi vse potrebne osi za pobiranje izdelkov z mize 3D-tiskalnika. Torej je hipoteza potrjena, vendar so se pri tem pojavile težave, ki smo jih predstavili pri izboljšavah.



Slika 46: Strega 3D-tiskalniku za robotsko roko

(Vir: osebni arhiv)

10 NADALJNJE IZBOLJŠAVE

Robotsko roko lahko seveda še izboljšamo. Našteli bomo nekaj idej, s katerimi jo lahko nadgradimo:

- Koračni motorji na robotski roki so precej glasni. To se dogaja zaradi gonilnikov, saj niso najbolj kakovostni in imajo možnost le 1/16 mikrokoraka, kar zaradi kakovosti gonilnikov ni dovolj dobro. Ker so sodobni gonilniki, ki so veliko kakovostnejši, zelo dragi, se nismo odločili zanje. Imajo pa možnost 1/32 mikrokoraka, kar je mnogo bolje od sedanjih.
- Kot smo že omenili, je G-koda sestavljena tako, da bo robotska roka čakala, dokler izdelek ne bo dokončan. Nato pa se G-koda zažene. Poskusili bomo odkriti način, kako ustvariti zanko, kjer bo pogoj pritisnjeno končno stikalo. Ko bo tiskalna miza sprožila stikalo, bo robotska roka začela svoj proces.
- Naslednja izboljšava, ki jo nameravamo izvesti, je, da bi imeli posebej krmilnik za G-kodo in ne računalnik. Ker moramo imeti računalnik vedno pri roki, to ni praktično, včasih pa to tudi ni izvedljivo. Tako bi lahko robot nemoteno deloval brez prisotnosti računalnika.
- Razmišljali smo, da bi dodali še kamero za lažje nadzorovanje procesa.
- V našem primeru z delovne površine tiskalnika odstranjujemo enostavne valje. V primeru, da izdelek ni tako enostavne oblike, pride do problema z njegovim odstranjevanjem.
- Ker je motor za dvig prijemala zelo šibek, razmišljamo, da bi vgradili močnejši motor, kar bi odprlo možnost dvigovanja težjih bremen.
- Pri robotski roki smo odkrili še en problem. Ko jo izklopimo, se ničelna točka izbriše oz. se premakne toliko, kolikor mi premaknemo robota, saj ob izklopu motorji izgubijo moč. Razmišljamo o vgradnji končnih stikal, ki bodo robotu omogočala vedno isto postavitvev v ničelno pozicijo. Razmišljali smo tudi o namestitvi enkoderjev na koračne motorje, ki bodo obdržali število korakov kljub izklopu robotske roke.

11 ZAKLJUČEK

Pri izdelavi robotske roke smo morali vložiti veliko truda in znanja, ki smo ga pridobili v času srednješolskega izobraževanja. Med projektiranjem se je pojavilo precej težav, ki smo jih rešili s pomočjo spleta in priporočene literature s strani mentorjev. Naučili smo se konstruiranja v programu za 3D-modeliranje z imenom SolidWorks. Prav tako smo se naučili sestave in izdelave električnega vezja in programiranja. Težave so nam povzročali 3D natisnjeni elementi, ki včasih niso bili dovolj natančni za precizno sestavo. Soočili smo se z urejanjem žic in kablov, saj jih je zaradi velikega števila motorjev in ostale opreme, veliko. Zanimivo je bilo iskanje rešitev, ki smo se jih morali domisliti sami. Kljub vsem oviram in težavam nas je izdelek navdušil, saj je boljši, kot smo si ga zamislili na začetku. Pri raziskovalni nalogi smo dosegli naš cilj, raziskati izdelavo in meje 3D natisnjene robotske roke, ki jo bomo v prihodnosti izboljšali. Pridobili smo izkušnje, ki nam bodo koristile kasneje v življenju pri opravljanju našega poklica.

11 VIRI

- [1] Arduino MEGA (online). (citirano 15. 11. 2019). Dostopno na naslovu: <https://www.arduino.cc/>
- [2] BCN3D - all data (online). (citirano 20. 12. 2019). Dostopno na naslovu: <https://github.com/BCN3D/BCN3D-Moveo>
- [3] BCN3D - assembly (online). (citirano 20. 12. 2019). Dostopno na naslovu: <https://www.instructables.com/id/Build-a-Giant-3D-Printed-Robot-Arm/>
- [4] BCN3D – forum (online). (citirano 22. 12. 2019). Dostopno na naslovu: <https://forum.arduino.cc/>
- [5] BCN3D - moveo (online). (citirano 20. 12. 2019). Dostopno na naslovu: <https://www.bcn3d.com/bcn3d-moveo-the-future-of-learning/>
- [6] 3D – tiskalnik (online). (citirano 22. 1. 2020). Dostopno na naslovu: https://en.wikipedia.org/wiki/3D_printing
- [7] Koračni motor (online). (citirano 1. 3. 2020). Dostopno na naslovu: https://en.wikipedia.org/wiki/Stepper_motor
- [8] Programska oprema Marlin (online). (citirano 7. 2. 2020). Dostopno na naslovu: <https://marlinfw.org/>
- [9] Pronterface (online). (citirano 22. 2. 2020). Dostopno na naslovu: <https://www.pronterface.com/>
- [10] RAMPS 1.4 (online). (citirano 10. 11. 2019). Dostopno na naslovu: https://reprap.org/wiki/RAMPS_1.4
- [11] Servo motor (online). (citirano 1. 3. 2020). Dostopno na naslovu: <https://en.wikipedia.org/wiki/Servomotor>
- [12] Stepper – online (online). (citirano 21. 12. 2019). Dostopno na naslovu: <https://www.omc-stepperonline.com/>
- [13] Wikipedia (online). (citirano 15. 2. 2020). Dostopno na naslovu: https://en.wikipedia.org/wiki/Robotic_arm