Razvoj upravljačkog sustava za niskobudžetne 3D pisače

Vadlja, Hrvoje

Undergraduate thesis / Završni rad

2020

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Zagreb, Faculty of Mechanical Engineering and Naval Architecture / Sveučilište u Zagrebu, Fakultet strojarstva i brodogradnje**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:235:391737

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-03-28**

Repository / Repozitorij:

Repository of Faculty of Mechanical Engineering and Naval Architecture University of Zagreb

SVEUČILIŠTE U ZAGREBU
FAKULTET STROJARSTVA I BRODOGRADNJE

ZAVRŠNI RAD

Hrvoje Vadlja

Zagreb, 2020. godina.

SVEUČILIŠTE U ZAGREBU
FAKULTET STROJARSTVA I BRODOGRADNJE

ZAVRŠNI RAD

Mentor: Student:

Doc. dr. sc. Tomislav Staroveški, dipl. ing. Hrvoje Vadlja

Zagreb, 2020.godina.

Izjavljujem da sam ovaj rad izradio samostalno koristeći znanja stečena tijekom studija i navedenu literaturu.

Zahvaljujem se mentoru Doc.dr.sc. Tomislavu Staroveškom na pruženoj stručnoj pomoći i savjetima prilikom izrade završnog rada.

Zahvaljujem se roditeljima i sestri na pruženoj podršci i razumijevanju tijekom studiranja.

Hrvoje Vadlja

SVEUČILIŠTE U ZAGREBU
FAKULTET STROJARSTVA I BRODOGRADNJE
Središnje povjerenstvo za završne i diplomske ispite
Povjerenstvo za završne ispite studija strojarstva za smjerove:
proizvodno inženjerstvo, računalno inženjerstvo, industrijsko inženjerstvo i menadžment, inženjerstvo materijala i mehatronika i robotika

Sveučilište u Zagrebu Fakultet strojarstva i brodogradnje	
Datum 21 -02- 2020	Prilog
Klasa: 602-04/20-6/3	
Ur.broj: 15-1207-20-926	

ZAVRŠNI ZADATAK

Student: **Hrvoje Vadlja** Mat. br.: **0035201273**

Naslov rada na hrvatskom jeziku: **Razvoj upravljačkog sustava za niskobudžetne 3D pisače**

Naslov rada na engleskom jeziku: **Control system for low budget 3D printers**

Opis zadatka:

U Laboratoriju za alatne strojeve dostupan je djelomično izrađen niskobudžetni 3D pisač. Pisač je opremljen odgovarajućim posmičnim motorima i glavnim prigonom, međutim, nedostaje odgovarajući upravljački sustav. U cilju cjelovite realizacije predmetnog pisača, u radu je potrebno:

1. Opisati princip rada 3D pisača zasnovanih na postupku taložnog očvršćivanja (FDM).
2. Dati pregled postojećih otvorenih upravljačkih sustava prikladnih za primjenu na 3D pisačima.
3. Odabrati odgovarajući upravljački sustav i parametrirati ga.
4. Dati zaključke rada.

Zadatak zadan:
28. studenog 2019.

Datum predaje rada:
1. rok: 21. veljače 2020.
2. rok (izvanredni): 1. srpnja 2020.
3. rok: 17. rujna 2020.

Predviđeni datumi obrane:
1. rok: 24.2. – 28.2.2020.
2. rok (izvanredni): 3.7.2020.
3. rok: 21.9. - 25.9.2020.

Zadatak zadao:

Doc. dr. sc. Tomislav Staroveški

Predsjednik Povjerenstva:

Prof. dr. sc. Branko Bauer

SADRŽAJ

SADRŽAJ I
POPIS SLIKA II
SAŽETAK III
SUMMARY IV
1. UVOD 1
2. FDM tehnologija 2
2.1. Povijest 3D pisača 2
2.2. Princip rada FDM tehnologije 2
2.3. Materijali 5
2.4. Prednosti i nedostaci 3D pisača (FDM) 5
2.4.1. Prednosti 3D pisača (FDM) 6
2.4.2. Nedostaci 3D pisača (FDM) 6
3. Izrada modela i priprema za tisak 9
3.1. Izrada CAD modela 9
3.2. STL format 10
3.3. Programi za analizu STL datoteka i generiranje putanje alata 12
3.4. Princip rada slicer programa 12
3.4.1. Cura (Verzija „Ultimaker Cura 4.4.1“) 12
3.4.1.1. Quality 16
3.4.1.2. Shell 17
3.4.1.3. Infill 17
3.4.1.4. Material 18
3.4.1.5. Ispis G-koda 19
4. Simulacija 3D pisača pomoću LinuxCNC programa 21
4.1. LinuxCNC 21
4.2. Simulacija rada 3D pisača 23
4.2.1. Model 3D pisača 23
4.2.2. Modul za vizualizaciju stroja u LinuxCNC-u s pripadajućim izvodnicama (engl. Pins) 26
4.2.3. INI datoteka 30
4.2.4. HAL skripta 34
5. Simulacija 3D pisača 38
6. Zaključak 39
7. LITERATURA 40

POPIS SLIKA

Slika 1. Nanošenje slojeva ... 3
Slika 2. Primjer žice za 3D pisač ... 3
Slika 3. Glava ekstrudera s dvije sapnice ... 4
Slika 4. Primjer slomljenog modela pod opterećenjem ... 7
Slika 5. Usporedba troškova ovisno o veličini serije (FDM i injekcijsko prešanje) ... 8
Slika 6. Narušena kvaliteta uslijed nepravilnog skupljanja materijala ... 8
Slika 7. FreeCAD korisničko sučelje ... 9
Slika 8 *Export* 3D CAD modela u STL format ... 10
Slika 9 3D CAD model spremljen u .stl formatu ... 11
Slika 10 STEP i STL datoteka ... 11
Slika 11. Početno sučelje programa *Cura* ... 12
Slika 12. Unos vrijednosti radnog prostora (*Cura)* ... 13
Slika 13. Unos karakteristike mlaznice pisača, program *(Cura)* ... 13
Slika 14. Učitani model (*Cura)* ... 14
Slika 15. Grupe značajki (*Cura)* ... 15
Slika 16. Odabir visine sloja *(Cura)* ... 16
Slika 17. Odabir debljine stijenke *(Cura)* ... 17
Slika 18. Odabir gustoće i vrste ispune *(Cura)* ... 17
Slika 19. Odabir temperature printanja*(Cura)* ... 18
Slika 20. Testni model ... 18
Slika 21. Prikaz očekivanog trajanja ispisa *(Cura)* ... 19
Slika 22. Ručni pregled procesa tiskanja *(Cura)* ... 19
Slika 23. Početak generiranog G-koda *(Cura)* ... 20
Slika 24. Grafički prikaz arhitekture programa LiunxCNC [6] ... 22
Slika 25. Simulacija rada 3D pisača ... 23
Slika 26. STEP model 3D pisača ... 24
Slika 27. Učitavanje 3D modela i generiranje .stl datoteke *(Autodesk 123D Design)* ... 25
Slika 28. Odabir parametara za pohranu STL datoteke ... 25
Slika 29. *Backplot* ... 28
Slika 30. Dozvola izvršavanja ... 29
Slika 31. Pokretanje skripte ... 29
Slika 32 3D tiskač ... 30
Slika 33. Vizualizacija *Limit 3* modula ... 35
Slika 34. Vizualizacija *wcomp* modula ... 36
Slika 35. Vizualizacija *not* modula ... 36
Slika 36 Naredba za pokretanje simulacije 3D tiskača ... 38
Slika 37 Simulacija 3D tiskača ... 38

SAŽETAK

Ovaj rad se bavi razvojem upravljačkog sustava za niskobudžetne 3D pisače. Prvi dio rada opisuje tehnologiju taložnog očvršćivanja (engl. *Fused Deposition Modeling*) koju predmetni 3D pisač koristi.U drugom dijelu rada prikazan je način generiranja G-koda za izradu proizvoljnog modela te su opisani koraci projektiranja upravljačkog sustava u programskom paketu LinuxCNC. Upravljački sustav testiran je pomoću simulacijskog modela koji je također razvijen u okviru ovog završnog rada.

Ključne riječi: 3D pisač, FDM, LinuxCNC, upravljački sustav, simulacija, otvoreni kod

SUMMARY

This thesis describes the development of a control system for low budget 3D printers. The first part of the paper describes the Fused Deposition Modeling technology used by the 3D printer in question. The second part of the paper describes methods and tools for G-code generation, and describes the steps for designing a control system in a LinuxCNC software package that can interpret and execute it. The control system was tested using suitable model, which was also developed as part of this paper.

Key words: 3D printer, FDM, LinuxCNC, control system, simulation, open source

1. UVOD

Upotreba 3D pisača sve je raširenija u području industrije. Broj kompanija koje ih koriste rapidno raste. Tehnologija 3D tiska nadišla je početnu, iako još uvijek raširenu, primjenu za brzu izradu prototipa. Danas se koristi u medicini, zrakoplovnoj, automobilskoj i mnogim drugim industrijama u svrhu izrade funkcionalnih dijelova, a ne samo prototipa. Mnogi proizvođači plasirali su na tržište serije niskobudžetnih pisača koji nisu namijenjeni za upotrebu u komercijalne svrhe.Takvi strojevi najčešće koriste upravljačke sustave i programske pakete zatvorenog koda koji se ne mogu u potpunosti parametrizirati. Stoga je u ovom radu razmatran razvoj 3D pisača temeljen na otvorenom i javno dostupnom LinuxCNC sustavu.

2. FDM tehnologija

Izraz „3D tisak“ objedinjuje mnogo tehnologija koje se koriste za izradu trodimenzionalnih dijelova aditivnim postupkom. Tehnologija 3D pisača koji je predmet ovog završnog rada zasniva se na postupku taložnog očvršćivanja (FDM) engl. *Fused Deposition Modeling*.

2.1. Povijest 3D pisača

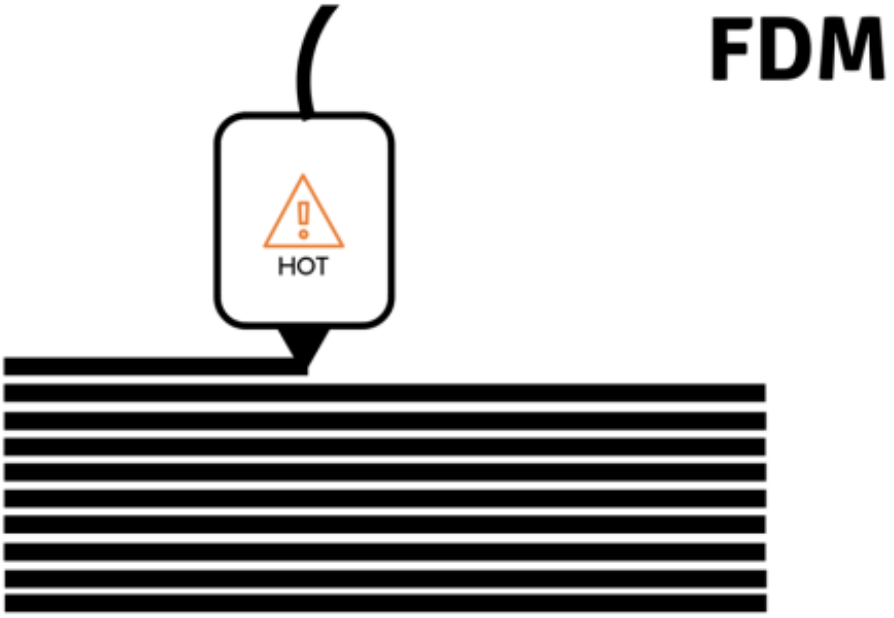
Već 1981., Hideo Kodama izdao je rad o izradi prototipova koristeći foto polimere čiji se model izrađivao dodavanjem slojeva jedan na drugi.

1986. Charles Hull je patentirao tu metodu i nazvao je stereolitografija (STL), ona je preteča svim ostalim aditivnim tehnologijama. Koristi tekući materijal (foto polimer) koji se pomoću izuzetno preciznog ultraljubičastog lasera očvršćuje i omogućuje visoku točnost izrade (0.02 mm).

Danas najraširenij u i najdostupniju tehnologiju 3D tiska izumio je Steven Scott Crump. Već 1989. zaštitio ju je patentom pod imenom „*Apparatus and Method for Creating Three-Dimensional Objects“.* Patent je istekao 2009. godine što je uvelike pridonijelo dostupnosti 3D pisača koji koriste FDM tehnologiju. Steven Scott Crump osniva tvrtku Stratasys koja je i danas vodeća u području 3D tiska.[1]

2.2. Princip rada FDM tehnologije

Izrada dijelova postupkom taložnog očvršćivanja (FDM) zasniva se na nanošenju materijala sloj po sloj (engl. *layer by layer*).



Slika 1. Nanošenje slojeva

Materijal koji se koristi je u obliku dugačke žice omotane na kolut [Slika 2.], on se kod strojeva manjih dimenzija najčešće nalazi na vanjskom dijelu kućišta, a kod većih modela je ukomponiran unutar stroja.

Slika 2. Primjer žice za 3D pisač

Pogonski kotačići uvlače žicu kroz cijevi i kontroliraju brzinu kojom ona dolazi do ekstrudera, u njemu se ona zagrijava do tekućeg stanja te se kroz vrh ekstrudera (sapnica/mlaznica) nanosi na podlogu ili prethodno nanesen sloj [Slika 1].

Nakon svakog sloja podloga se pomiče za korak (visinu sljedećeg sloja) po z osi. Na svaki sloj se nanosi idući i postupak se ponavlja dok se ne izradi cijeli model.

Postoje dvije vrste materijala; gradivni i potporni. Gradivni je onaj materijal od kojeg se naš model izrađuje, a izrada kompliciranije geometrije ponekad zahtjeva korištenje potpornog materijala. U takvom slučaju koristi se glava ekstrudera s dvije mlaznice, kroz jednu izlazi gradivni materijal , a kroz drugu potporni.

Slika 3. Glava ekstrudera s dvije sapnice

2.3. Materijali

Može se koristiti više vrsta gradivnih materijala, ovisno o namjeni proizvoda. Materijali moraju imati određena svojstva poput odgovarajuće čvrstoće, gustoće i modula elastičnosti kako bi se materijal pravilno rasporedio. Najčešći gradivni materijali koji se koriste kod FDM tehnologije su ABS i PLA, a postoje još; PA, PETG, TPU, PEI, PPSF/PPSU, PC, PC/ABS.[2]

ABS (akrilonitril-butadienstiren) se najčešće polimerizira kroz proces emulzije. Dobiva se polimerizacijom stirena, akrilonitrila uz prisutnost polibutadiena. Omjeri mogu varirati od 15% do 35% akrilonitrila (C3H3N), 5% do 30% butadiena (C4H6) i 40% do 60 % stirena (C8H8).[2] Često se koristi zbog visoke tvrdoće i otpornosti na udarce. Izrazito je otporan i na korozivne kemikalije. Niska temperatura taljenja olakšava proces 3D tiskanja. Raznim udjelima u sastavu može se modificirati otpornost na udarce, žilavost i otpornost na toplinu.

Uz ABS kao potporni materijal najčešće se koristi HIPS (engl. *High Impact Polystyrene*). HIPS je topljiv materijal koji ne ostavlja nikakve tragove potpornog materijala na modelu, jeftin je i lagan. Otporan je na vodu i udarce, također, ima poprilično slična svojstva ispisa kao ABS što ga čini odličnim „partnerom“ u izradi modela. Uz to, može se koristiti i kao gradivni materijal za potrošne dijelove.

Drugi najčešći gradivni materijal je PLA (polilaktidna kiselina). PLA je biorazgradiva plastika koje se dobiva iz obnovljivih organskih izvora; kukuruzni škrob i šećerna trska. PLA filamenti su kompatibilni s gotovo svim 3D pisačima i ne zahtijevaju grijanu podlogu pa su zbog toga najprodavaniji materijal za 3D tisak. Postiže se sjajni završni sloj pa se često koristi za izradu dekorativnih predmeta, maketa i umjetnina.

Kao potporni materijal uz PLA najčešće se koristi PVA (engl. *Polyvinyl Alcohol*). Zato što je najpraktičnije rješenje za potporni materijal kod izrade kompleksne unutarnje geometrije jer se izrazito lako uklanja razgrađivanjem u toploj vodi budući da je jako osjetljiv na vlagu.

2.4. Prednosti i nedostaci 3D pisača (FDM)

Iako je još uvijek teško odrediti koliko su općenito FDM 3D pisači ekološki prihvatljivi smanjena količina otpadnog materijala i energetska učinkovitost sugeriraju da ima potencijala da bude ekološki iznimno pogodan. Također, FDM tehnologija i dalje svakodnevno napreduje, proces se poboljšava i pojavljuju se novi materijali. S druge strane i dalje imamo relativno limitiran spektar materijala koji se mogu koristiti. Mnoge vrste plastike ne mogu se još uvijek dovoljno precizno temperaturno kontrolirati da bi bile pogodne za tisak.[3]

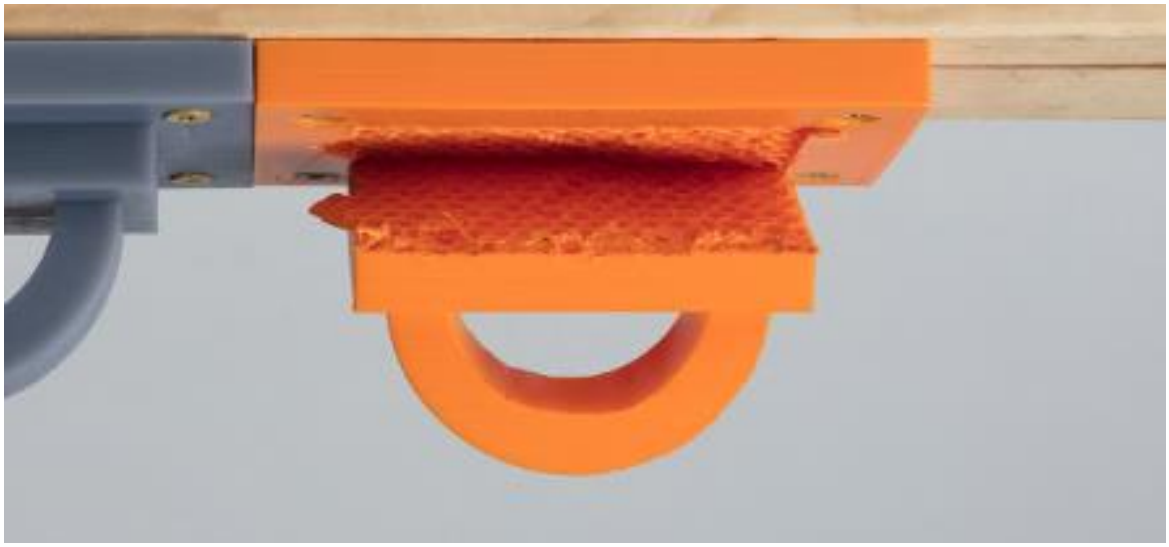
2.4.1. Prednosti 3D pisača (FDM)

Za razliku od obrade odvajanjem čestica nema škarta, materijala se koristi točno onoliko koliko i treba, a otpadni materijal je samo potporni. Moguća je izrada veoma kompleksne unutarnje geometrije koja bi ponekad bila nemoguća za izradu klasičnim postupcima odvajanjem. To ga čini odličnim za dijelove koji služe za hlađenje jer nemamo limita kod konstruiranja unutarnje mreže kanala. Često se koristi za izradu prototipova. Kod njih je najbitnije što brže doći od CAD modela do stvarnog proizvoda da bi se uočile eventualne konstrukcijske pogreške. 3D tisak je „singlestep manufacturing process" što znači da se štedi puno vremena koje bi se potrošilo kod samog transporta komada između obradnih strojeva. Sam proces dizajna sastoji se od iteracija koje su kod korištenja 3D pisača sada mnogo „bliže" i za svaku promjenu moguće je napraviti novi prototip u razumnom i prihvatljivom vremenu. Nije nužno ni da tvrtka posjeduje 3D pisač, „*outsource*" takve usluge je izrazito brz, dostupan i jednostavan. Sve više poduzeća se odlučuje na kupnju 3D pisača jer početno ulaganje nije veliko, a materijali su jeftini. Uz efikasniju izradu prototipova postoji i mogućnost uštede skladišnog prostora, izrađuje se virtualna baza dijelova koji se onda izrađuju po „baš na vrijeme" (engl. *just in time*) modelu poslovanja. Takav model omogućuje brze promjene na postojećim modelima bez velikih izvanrednih troškova.

2.4.2. Nedostaci 3D pisača (FDM)

Nemoguće je uvijek egzaktno predvidjeti ponašanje materijala, može se pojaviti nepredvidljivo skupljanje ili raslojavanje. Kada se raslojavanje dogodi ono je između dva sloja modela, teško se može kontrolirati jer se sama tehnologija svodi na izradu sloj po sloj.

Proces takve izrade uzrokuje i smanjena mehanička svojstva okomito na smjer tiska. Modeli su jako osjetljivi na smično opterećenje pa se kod izrade mora voditi računa o pozicioniranju modela kako bi se zadovoljila zahtijevana mehanička svojstva.

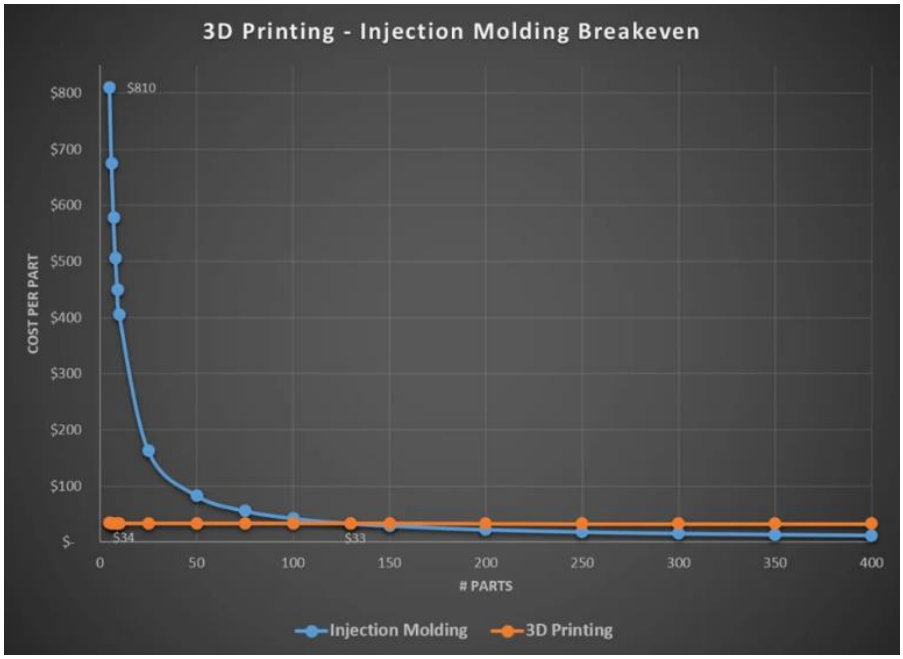


Slika 4. Primjer slomljenog modela pod opterećenjem

Slika 4. prikazuje problem raslojavanja i popuštanja veza između slojeva uslijed opterećenja. Iako samo opterećenje nije bilo smično već vlačno ono se moglo izbjeći. Rotiranjem modela za devedeset stupnjeva oko horizontalne osi kod same izrade izbjegla bi se ovakva deformacija, ali bi to ujedno značila veće troškove. Utrošak potpornog materija bio bi mnogo veći te bi vrijeme printanja bilo duže zbog većeg broja slojeva.

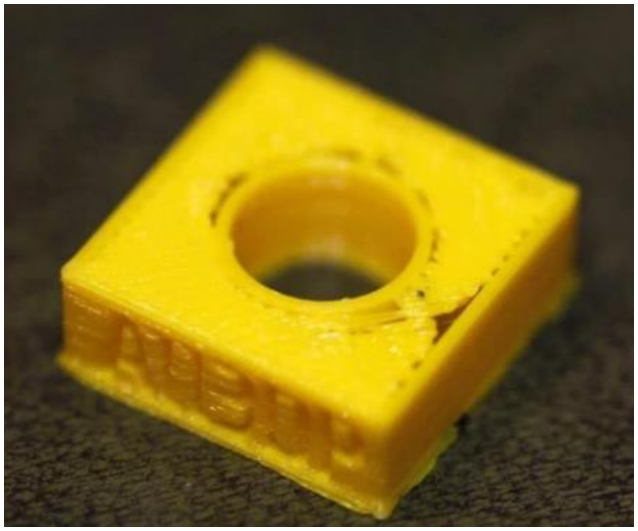
FDM ima najmanju dimenzijsku točnost od svih tehnologija 3D tiska. Mlaznica je okrugla te je nemoguće izraditi oštre rubove. Iz tih razloga je nepogodan za sitne dijelove.

Najjeftiniji je za proizvodnju pojedinačnih dijelova, ali u slučaju serijske proizvodnje koristi se samo kad je nemoguće izraditi unutarnju geometriju drugim postupcima. Kod velikoserijske proizvodnje i dalje su konvencionalni postupci često jeftiniji i isplativiji kroz duža razdoblja. Graf na slici 5. prikazuje troškove izrade ovisno o veličini serije.



Slika 5. Usporedba troškova ovisno o veličini serije (FDM i injekcijsko prešanje)

Drugi problem kod velikoserijske proizvodnje je velik broj komada koji se moraju otpisati zbog nekonzistentnosti tiska. Komadi zbog raznih utjecaja imaju prevelika odstupanja od dimenzije ili sama kvaliteta strukture odstupa od propisane. Također, brzina 3D tiska kod zahtjeva za velikoserijsku proizvodnju nije dovoljna.



Slika 6. Narušena kvaliteta uslijed nepravilnog skupljanja materijala

Zbog dimenzija radnog prostora veličina izradaka je ograničena. Modeli većih dimenzija moraju se tiskati iz više pojedinačnih dijelova koji se naknadno spajaju.

3. Izrada modela i priprema za tisak

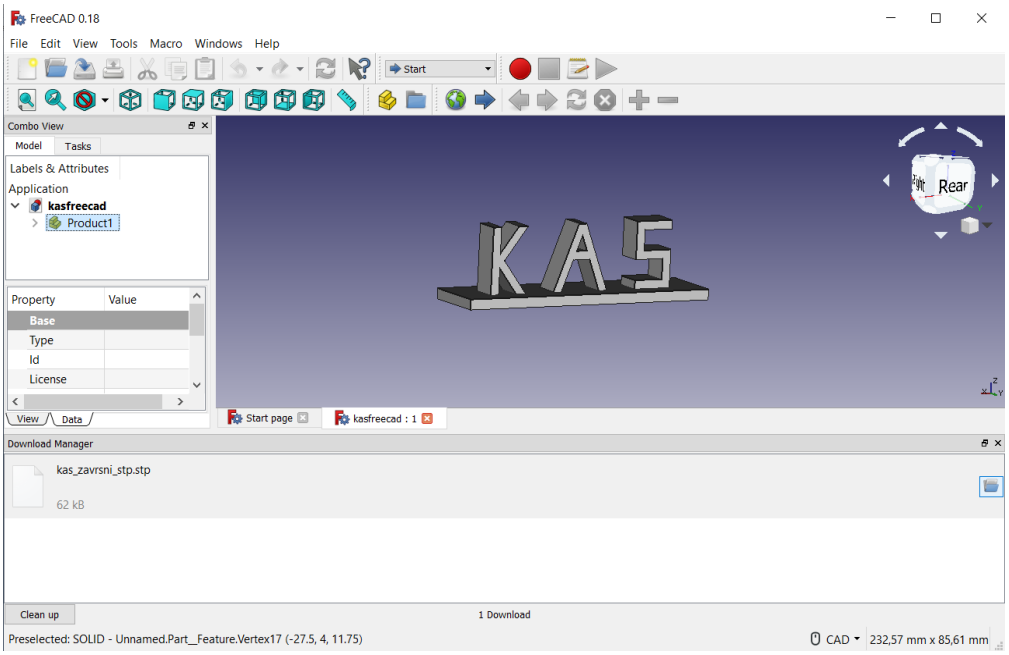
Za uspješno 3D tiskanje potrebno je pratiti određene korake, od izrade 3D modela sve do pripreme istoga za tisak.

3.1. Izrada CAD modela

Proces tiska započinje izradom CAD (skraćenica engl. *Computer-aided design*) modela. On se mora spremiti u format koji aplikacija za izradu G-koda može razumjeti. Najčešće je to STL datoteka, iako ima i drugih formata koje pojedine aplikacije mogu koristiti (OBJ, VRML, PLY itd.).

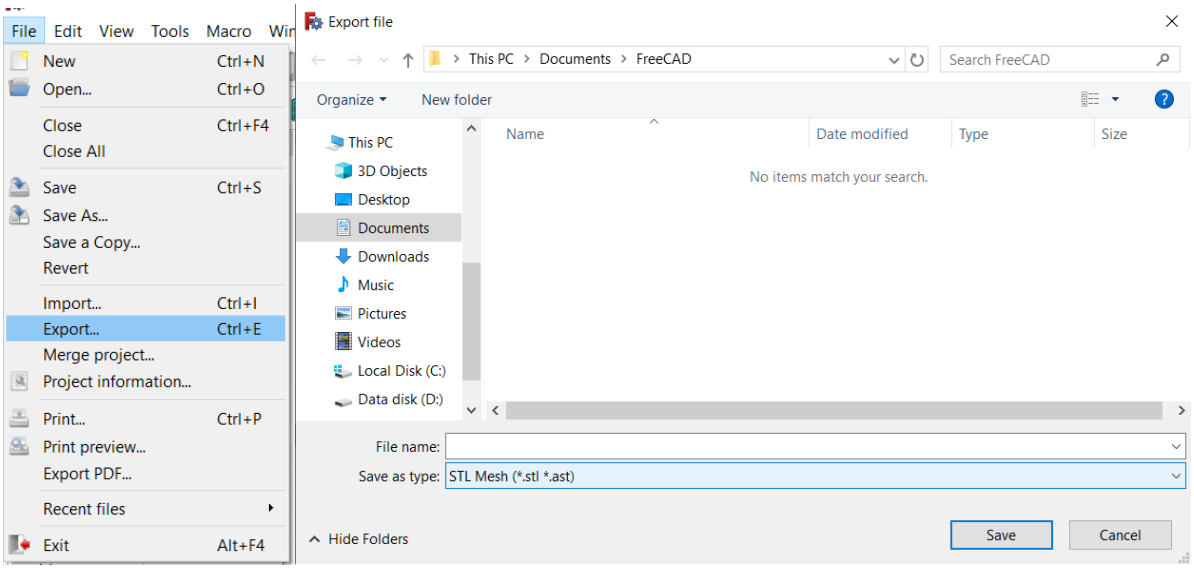
Postoje komercijalni alati koje se mogu koristiti (Catia, Solidworks, Siemens NX), ali budući da je ideja ovog završnog rada pokrenuti 3D pisač pomoću besplatnih programa koristiti će se FreeCAD koji može ispuniti zadaću izrade modela.

Najbitnije je da CAD sustav podržava spremanje datoteke u STL formatu. FreeCAD je jednostavan alat za modeliranje vrlo sličan Solidworksu. Ako korisnik ima iskustva u modeliranju ne iziskuje puno vremena da savlada osnove koje su mu dovoljne za izradu modela koji će se koristiti u 3D tisku.



Slika 7. FreeCAD korisničko sučelje

Za spremanje 3D modela u STL datoteku najprije je potrebno u padajućem izborniku kartice „*File*“ odabrati opciju „*Export*“ a zatim odabrati lokaciju gdje će se spremiti novonastala STL datoteka.



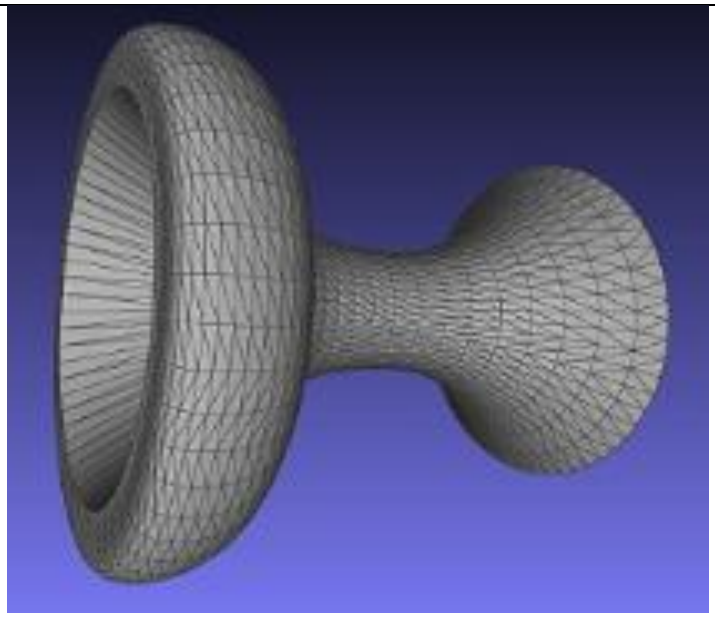
Slika 8. *Export* 3D CAD modela u STL format

3.2. STL format

Skraćenica „STL“ dolazi od engleske riječi „*stereolithography*“ koja označava tehnologiju koja je opisana u prvom dijelu. Postoji i druga teorija koja kaže da je „STL“ akronim engleskog izraza „*Standard Tessellation Language*“.

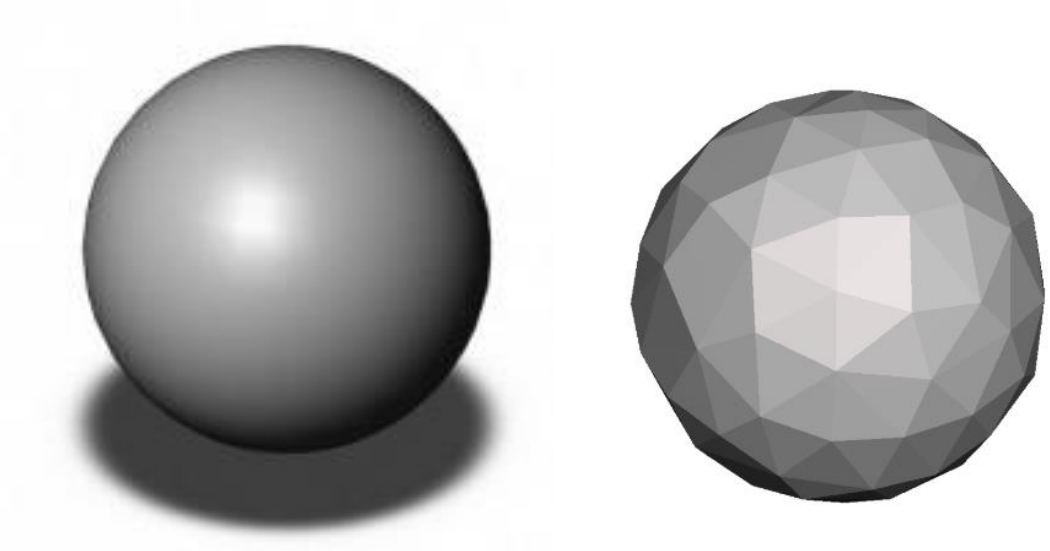
80-ih godina prošlog stoljeća stvoren je upravo za potrebe 3D tiska. Služi kao poveznica između CAD modela i *slicer* programa koji su objašnjeni u nastavku.

Model spremljen u STL formatu sastoji se od serije povezanih trokuta koju opisuju njegovu površinu, veći broj trokuta znači točniju površinu, odnosno, višu rezoluciju. Ne sadrži informacije o boji modela, materijalu, hrapavosti i sl. Ako su potrebni oni se spremaju u odvojenu datoteku.



Slika 9. 3D CAD model spremljen u .stl formatu

Treba imati na umu da proces pretvorbe datoteke u STL nije reverzibilan. Ako se u pretvorbi površine modela u seriju povezanih trokuta izgube neki oblici koji su važni za model njih je nemoguće „vratiti“ iz STL formata bez dodatnog modeliranja.



Slika 10. STEP i STL datoteka

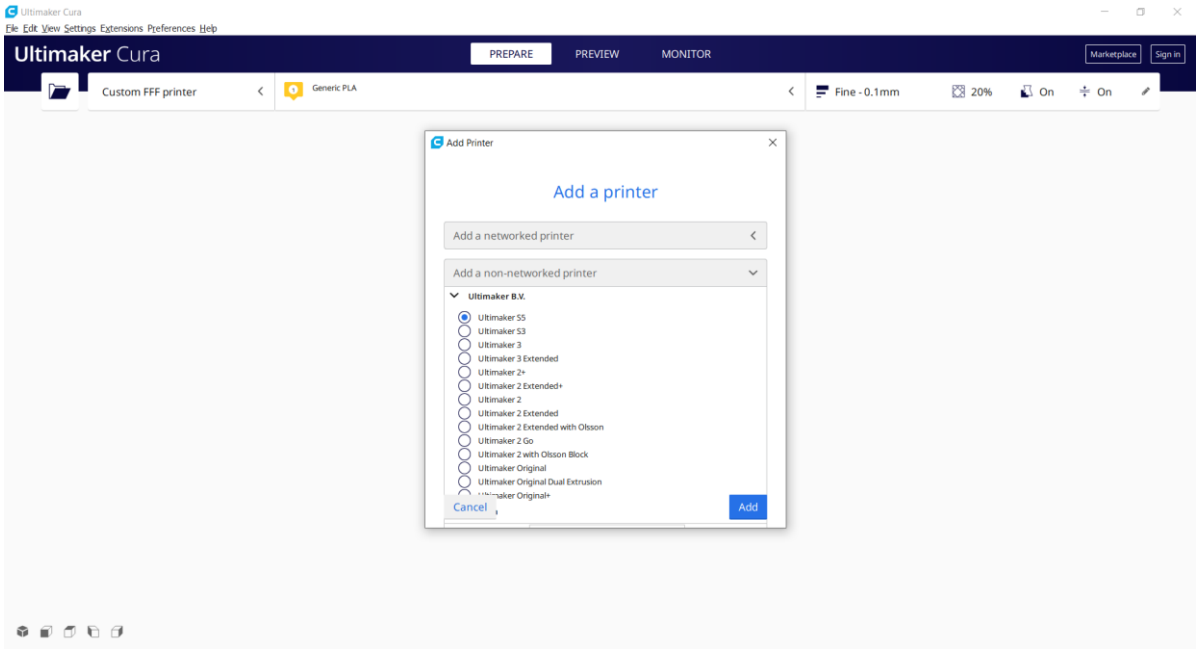
3.3. Programi za analizu STL datoteka i generiranje putanje alata

Slicer je alat koji pomoću STL datoteke iz prethodnog poglavlja generira kôd kojeg 3D pisač razumije i pomoću njega izrađuje model.

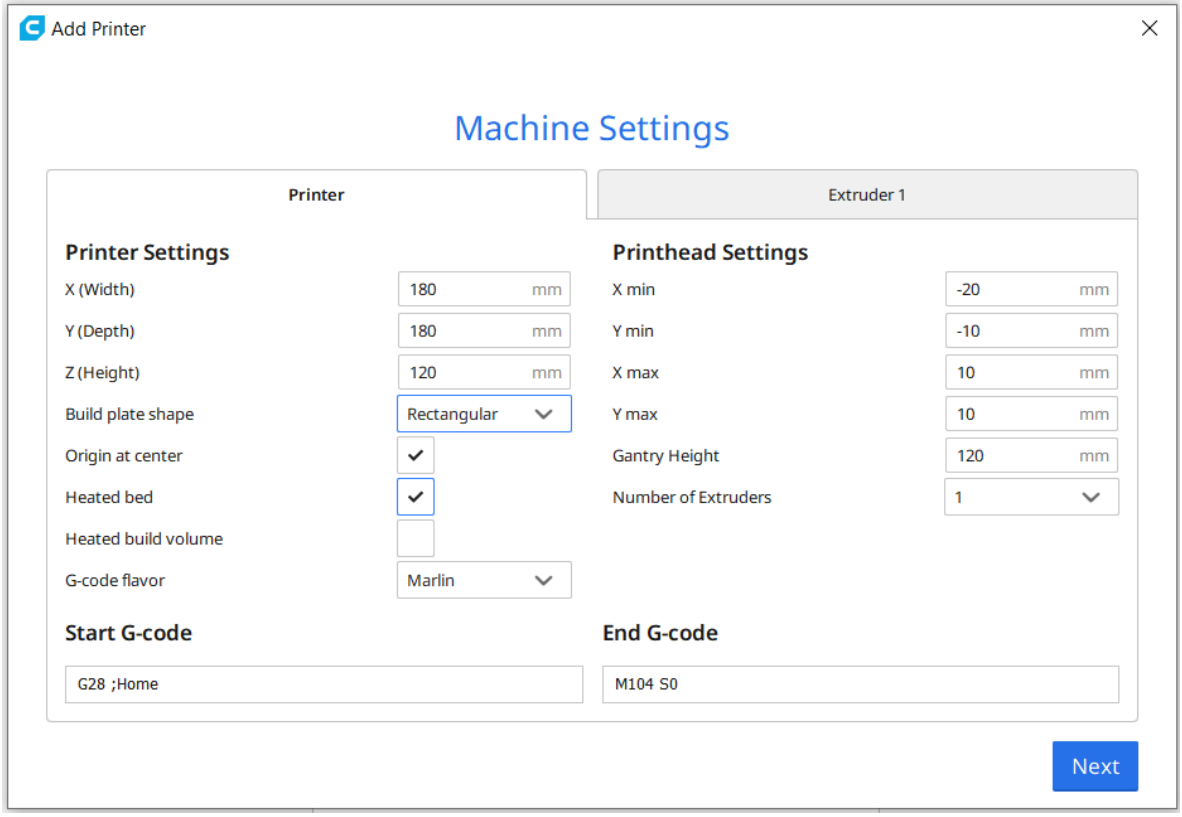
3.4. Princip rada slicer programa

Slicer najprije dijeli model na niz slojeva. Zatim stvara numeričku datoteku (G-kôd) koja opisuje putanju glave ekstrudera te količinu i temperaturu materijala potrebnu za proces tiska. Ta datoteku može se prenijeti na 3D pisač i započeti tiskanje komada.
Postoji mnogo popularnih alata koji se mogu koristiti. U ovom završnom radu koristiti će se Cura programski paket. Generirat će G-kôd koji pokreće i upravlja strojem u procesu izrade modela. Parametre koji se mogu odabrati i na što utječu objasnit će se u nastavku.

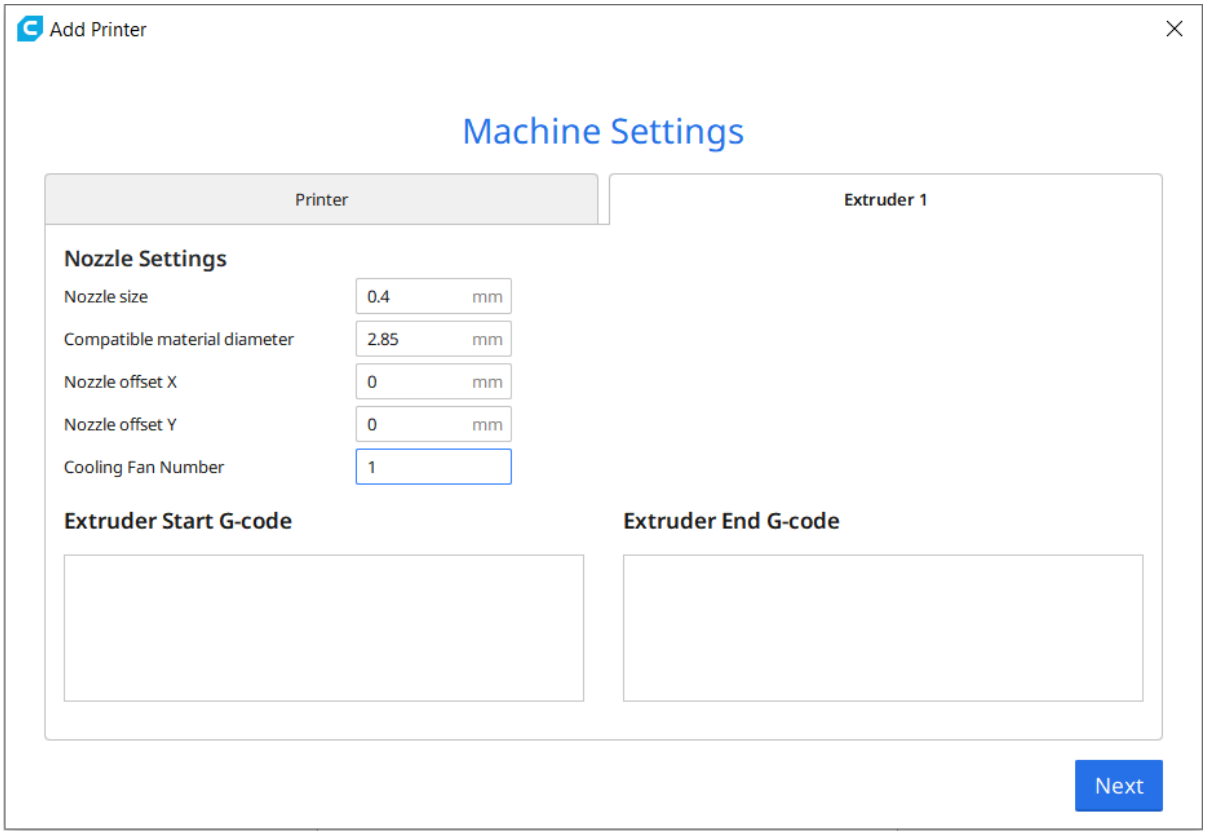
3.4.1. Cura (Verzija „Ultimaker Cura 4.4.1“)



Slika 11. Početno sučelje programa *Cura*



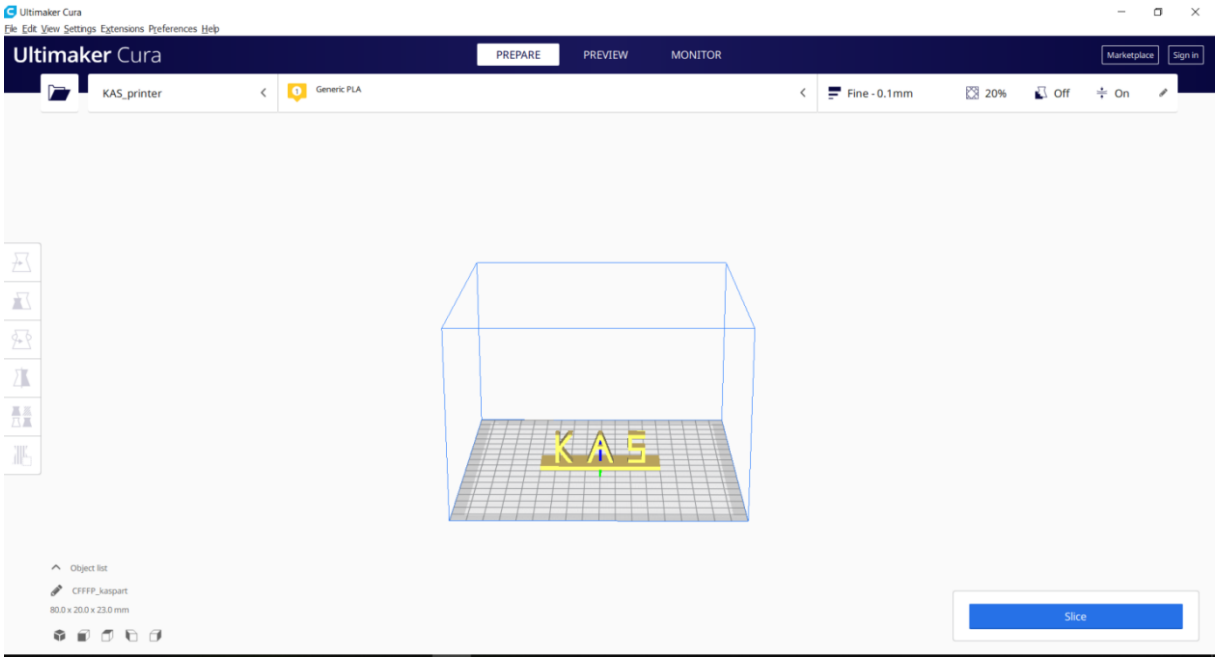
Slika 12. Unos vrijednosti radnog prostora (*Cura)*



Slika 13. Unos karakteristike mlaznice pisača, program *(Cura)*

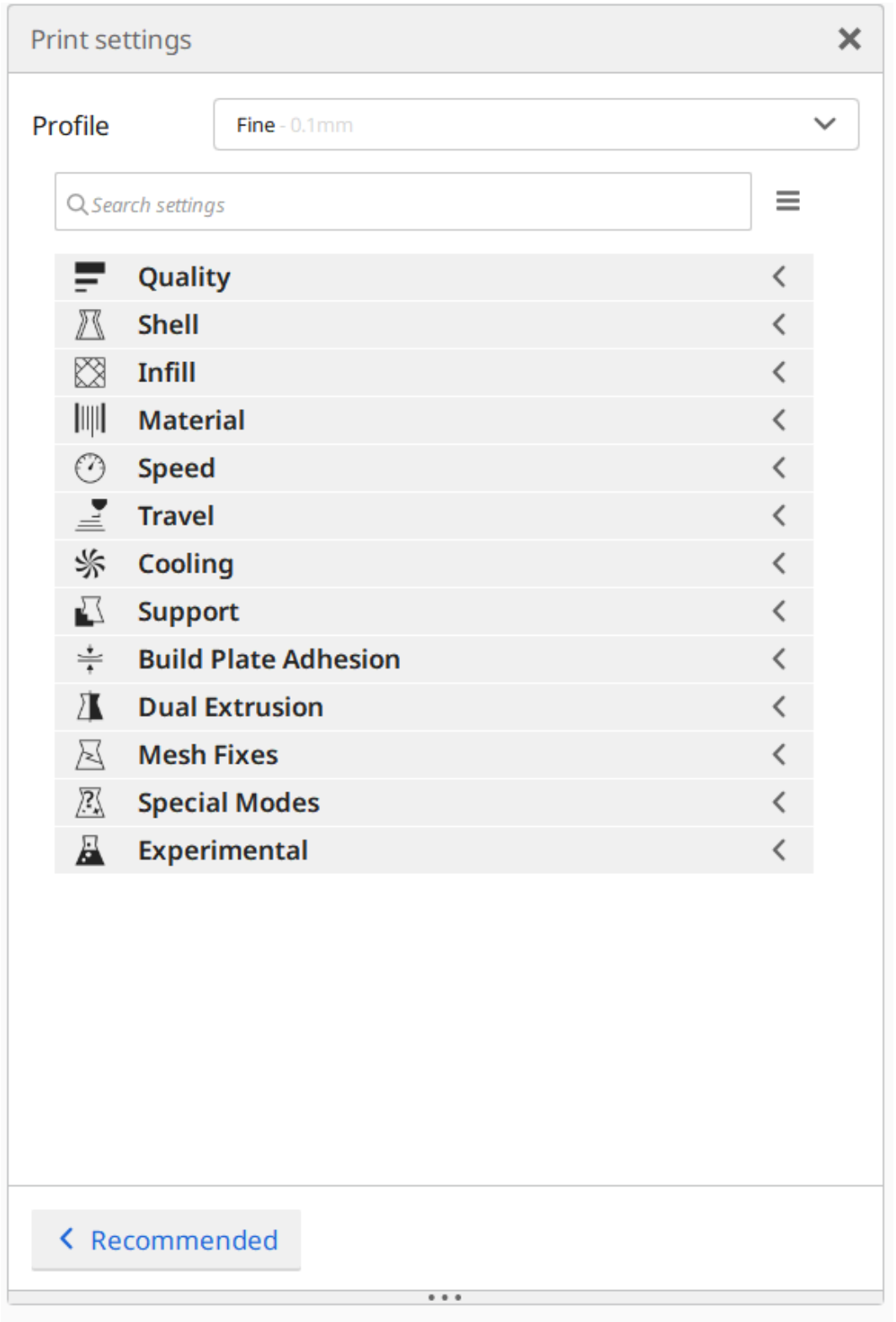
Potrebno je unijeti podatke o stroju koji se koristi. Slika 12. prikazuje prozor u kojem se definiraju vrijednosti radnog prostora kojim raspolažemo, mora se označiti upotreba grijane podloge, grijane komore i slično. U slučaju da se koristi 3D pisač kupljen od nekog proizvođača kao npr. Ultimaker, Cura nudi bazu podataka s već upisanim specifikacijama takvih strojeva i nije potrebno ručno unositi iste. Na slici 13. je drugi prozor u koji se unose karakteristike mlaznice pisača.

Nakon unošenja karakteristika stroja na ekranu se pojavljuje radni prostor 3D pisača u koji se sada može učitati bilo koja STL datoteka („*drag and drop*“) [Slika 14.]



Slika 14. Učitani model (*Cura)*

Potom se odabiru vrijednosti koje utječu na tisak. Na slici 15. prikazane su grupe značajki koje se mogu mijenjati. U nastavku će biti objašnjene one najbitnije i kako njihova promjena utječe na kvalitetu, te mehanička i vizualna svojstva izratka. Inicijalno su automatski unesene vrijednosti koje se najčešće koriste.



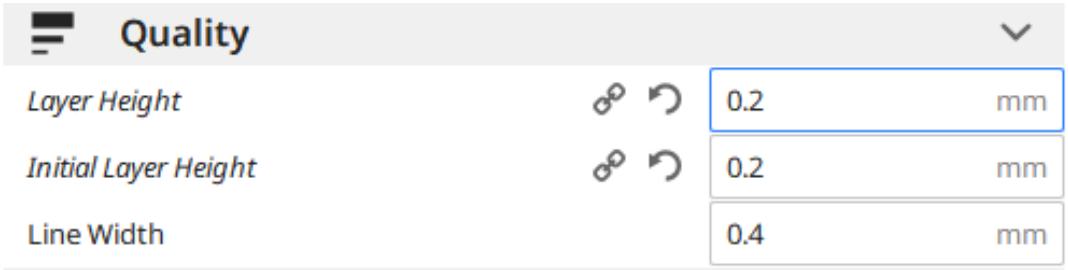
Slika 15. Grupe značajki (*Cura)*

3.4.1.1. Quality

Visina sloja (engl. *Layer height*) utječe na vrijeme potrebno za izradu, kvalitetu površine i čvrstoću izradaka. Većina jeftinijih tiskača zbog većeg promjera mlaznice ne podržava tisak sloja manjeg od 0.1 mm. Veća visina sloja znači da tiskač mora napraviti manje slojeva za isti komad te je vrijeme potrebnu za izradu kraće. Visina sloja utječe i na kvalitetu površine, tisak koji koristi manju visinu sloja imat će manja odstupanja od pravca ili neke druge krivulje na skošenim, odnosno zakrivljenim površinama. Također, neka istraživanja su pokazala da korištenje visine sloja u iznosu od 0.3 mm daje 24% veću tvrdoću od komada izrađenog od visine sloja 0.1 mm.[4]

Početna visina sloja (engl. *Initial layer height*) se odnosi na prvi sloj, poželjno je da on bude manji on visine sloja izrade da bi se omogućilo što bolje prijanjanje materijala na podlogu. Manja visina sloja daje glađu i popunjeniju površinu.

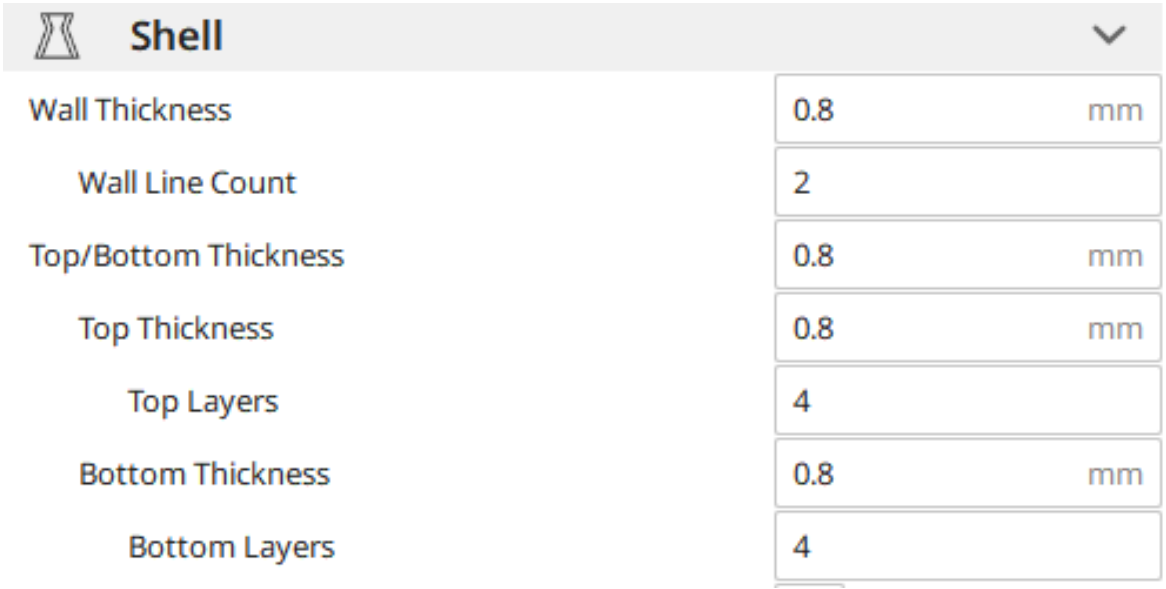
U slučaju da 3D pisač ima grijanu podlogu potrebno je povećati širinu sloja (engl. *Line width*) da bi se poboljšala adhezija [Slika 16.]. [9]



Slika 16. Odabir visine sloja *(Cura)*

3.4.1.2. Shell

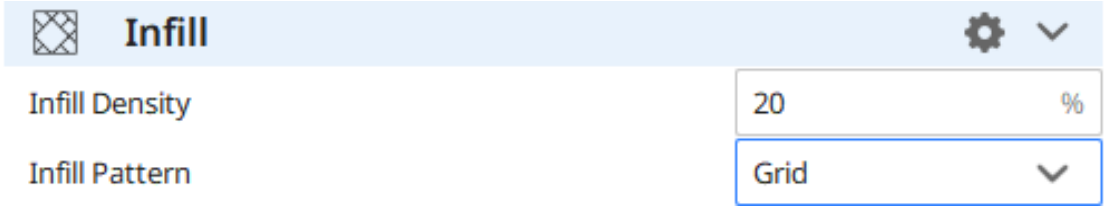
U ovoj rubrici se upisuju vrijednosti vezane uz debljinu stijenke (engl. *Shell*). Najčešće je to dvostruka vrijednost promjera mlaznice. Ona utječe na čvrstoću izradaka, a pretanka stijenka može biti uzrok pucanja komada kod opterećenja ili čak raspadanja kod procesa izrade. [Slika 17,]



Slika 17. Odabir debljine stijenke *(Cura)*

3.4.1.3. Infill

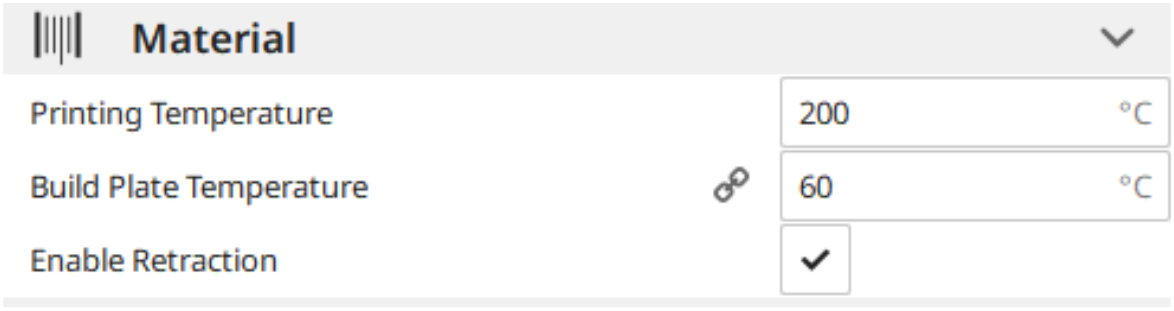
Ispuna (engl. *Infill*) također utječe na brzinu tiska, masu i čvrstoću. U suštini odabire se popunjenost modela. Broj u izborniku na slici 23. označava postotak. „Grid“ označava je geometriju unutrašnjosti modela. [Slika 18.]



Slika 18. Odabir gustoće i vrste ispune *(Cura)*

3.4.1.4. Material

Svaki materijal ima preporučenu temperaturu tiskanja. Usprkos tome, potrebno je provesti nekoliko testnih printanja kako bi se iteracijom došlo do najbolje temperature jer ona često ovisi o uvjetima na koje se u nekontroliranoj okolini ne može precizno utjecati kao npr. vlažnost i temperatura zraka. U sekciji za materijal bira se i temperatura grijane podloge [Slika 19.] (eng. *Build plate temperature*). Ona mora biti dovoljno visoka da omogući adheziju između materijala i ploče.



Slika 19. Odabir temperature printanja *(Cura)*

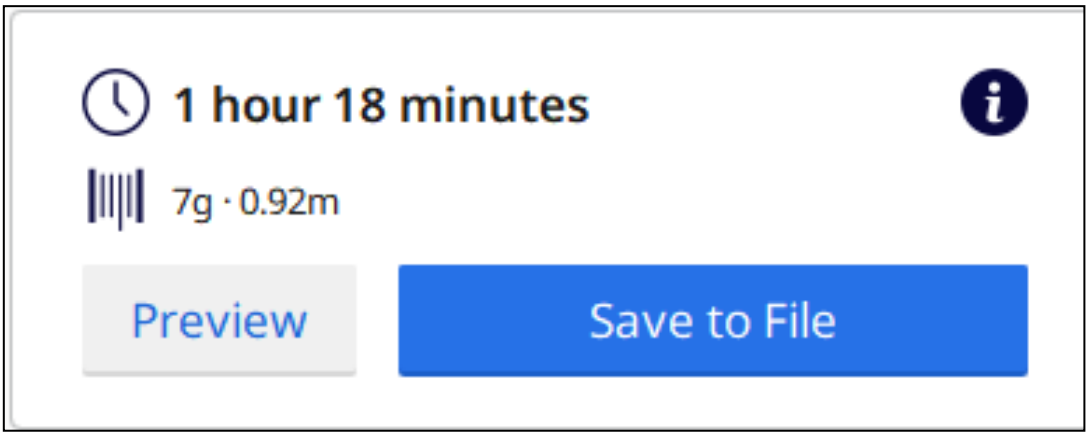
Kako bi se utvrdilo da su svi uneseni parametri odgovarajući, izrada testnog modela [Slika 20.] najbolji je pokazatelj toga. Ako se materijal ne lijepi za podlogu to je znak preniske temperature mlaznice. Ako je temperatura previsoka izrađeni model će biti neuredan i deformiran.

Slika 20. Testni model

3.4.1.5. Ispis G-koda

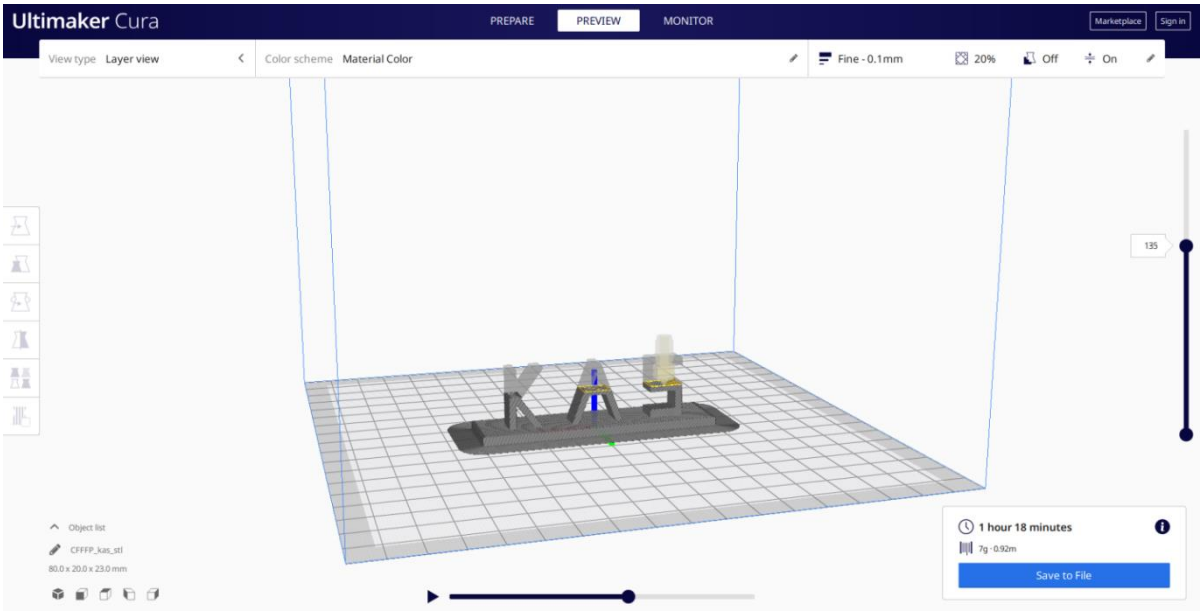
Nakon što su odabrani svi parametri tiska može se pokrenuti rezanje (engl. *Slicing*).

„Cura“ pruža povratnu informaciju o predviđenom vremenu potrebnom za izradu modela [Slika 21.].



Slika 21. Prikaz očekivanog trajanja ispisa *(Cura)*

Postoji i mogućnost ručnog pregleda procesa tiskanja [Slika 22.].



Slika 22. Ručni pregled procesa tiskanja *(Cura)*

Programski paket *Cura* sada ispisuje G-kod na temelju ulaznog 3D modela i podešenih parametara unutar samog programskog paketa.

```
;FLAVOR:Marlin
;TIME:4704
;Filament used: 0.924332m
;Layer height: 0.1
;MINX:-47.8
;MINY:-17.8
;MINZ:0.3
;MAXX:47.8
;MAXY:17.8
;MAXZ:23
;Generated with Cura_SteamEngine 4.4.1
M140 S60
M105
M190 S60
M104 S200
M105
M109 S200
M82 ;absolute extrusion mode
G28 ;Home
G1 Z15.0 F6000 ;Move the platform down 15mm
;Prime the extruder
G92 E0
G1 F200 E3
G92 E0
G92 E0
G92 E0
G1 F1500 E-6.5
;LAYER_COUNT:228
;LAYER:0
M107
G0 F3600 X40 Y17.8 Z0.3
;TYPE:SKIRT
G1 F1500 E0
G1 F1800 X-40 Y17.8 E1.50484
G1 X-40.789 Y17.76 E1.51971
G1 X-41.569 Y17.641 E1.53455
G1 X-42.334 Y17.443 E1.54941
G1 X-43.074 Y17.169 E1.56425
G1 X-43.783 Y16.821 E1.57911
G1 X-44.454 Y16.403 E1.59398
G1 X-45.078 Y15.92 E1.60883
G1 X-45.651 Y15.377 E1.62367
```

Slika 23. Početak generiranog G-koda *(Cura)*

4. Simulacija 3D pisača pomoću LinuxCNC programa

4.1. LinuxCNC

LinuxCNC (prijašnjeg naziva *Enhanced Machine Controller*) je programski paket koji je javno dostupan i pisan u otvorenom kodu (engl. *Open Source*). Predmetno je okruženje prilagođeno za operacijske sustave sa Linux jezgrom i predviđeno je za realizaciju numeričkog upravljanja različitih vrsta strojeva, poput; glodalica, tokarilica, plazma rezača, robota i sl.

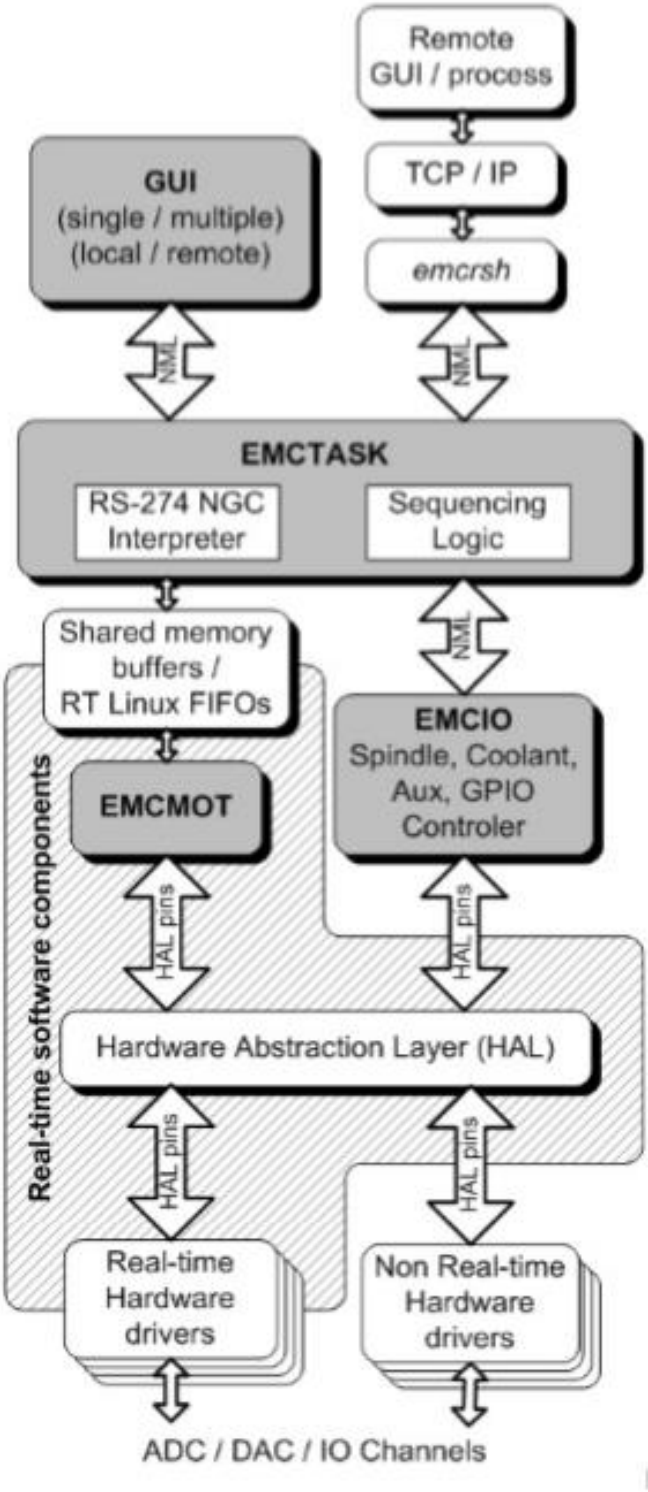
Može kontrolirati strojeve koji imaju do 9 stupnjeva slobode gibanja upravljanih G-kodom (RS-274NGC) koji se pogone servo ili koračnim motorima.

Okosnicu LinuxCNC sustava čini skupina modula koji se, zavisno o konfiguraciji stroja, međusobno povezuju u tzv. hardverskom apstrakcijskom sloju (engl. *Hardware Abstraction Layer*, HAL). Moduli predstavljaju funkcijske cjeline upravljačkog sustava, a najosnovniji mogu biti pisani u više programskih jezika, te se mogu izvršavati u stvarnom vremenu.

LinuxCNC sastavljen je od četiri glavne komponente:

1. Kontroler gibanja (engl. *Motion controller*, EMCMOT)
2. Diskretni I/O kontroler (engl. *Discrete I/O controller*, EMCIO)
3. Koordinirajući modul (engl. *Task coordinating module*, EMCTASK)
4. Grafičko sučelje (engl. *Graphical user interfaces*, GUI)

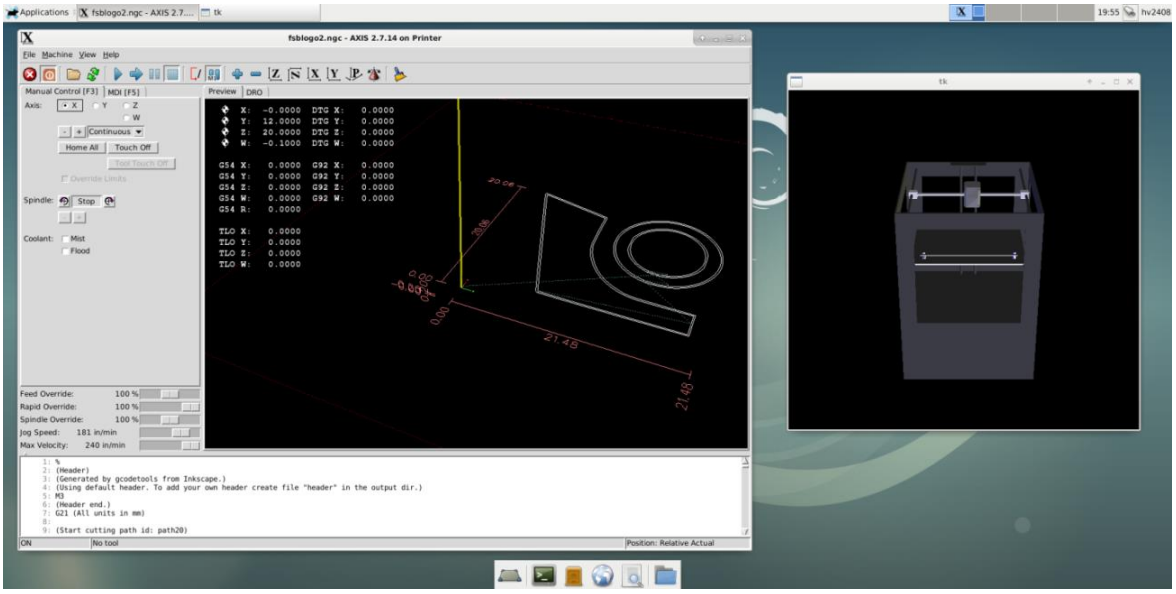
Od četiri navedena modula samo se EMCMOT izvršava u stvarnom vremenu. Komunikacija između modula koji se izvršava u stvarnom vremenu i modula koji se ne izvršavaju u stvarnom vremenu provodi se kroz dijeljenu memoriju. Komunikacija između modula koji se ne izvršavaju u stvarnom vremenu provodi se kroz NML (kratica, engl. *Neutral Messaging Language*) kanale [6].



Slika 24. Grafički prikaz arhitekture programa LiunxCNC [6]

4.2. Simulacija rada 3D pisača

U ovom poglavlju je detaljno opisan način na koji je realizirana simulacija 3D pisača. Za potrebe testiranja korišten je 3D model logotipa FSB-a, dok je putanja alata generirana interpretacijom, te izvršavanjem G-koda skripte.



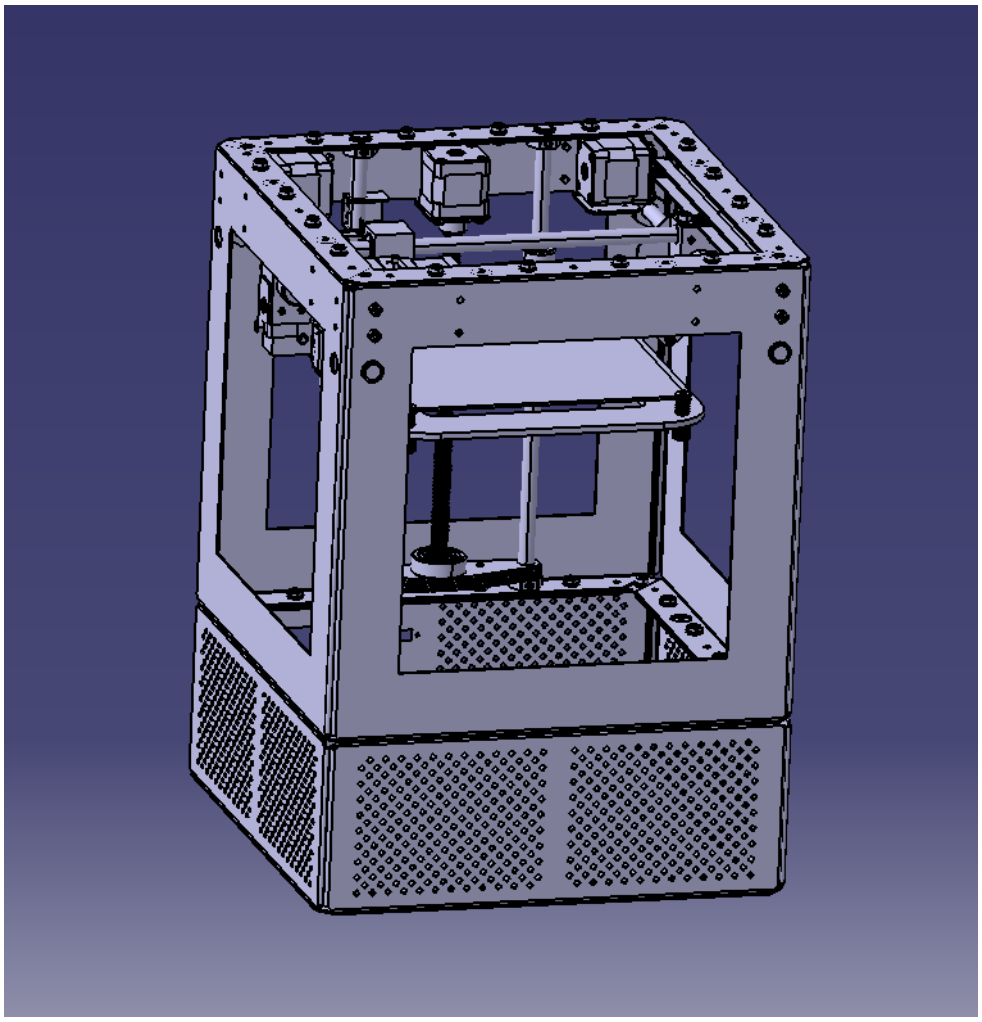
Slika 25. Simulacija rada 3D pisača

U cilju realizacije predmetnog pisača bilo je potrebno:

1. Izraditi 3D model stroja i potom model izvesti u STL formatu
2. Napisati LinuxCNC modul za trodimenzionalnu vizualizaciju 3D pisača
3. Napisati odgovarajuću konfiguracijsku datoteku s parametrima koji odgovaraju konfiguraciji
4. Napisati odgovarajuće HAL konfiguracijske datoteke koje odgovaraju konfiguraciji razmatranog pisača

4.2.1. Model 3D pisača

Na slici 26. prikazan je detaljan 3D model stroja koji je korišten u simulaciji. Dostupne biblioteke LinuxCNC-a za izradu modula za vizualizaciju stroja podržavaju isključivo uvoz STL datoteka u ASCII formatu. Imajući u vidu kako bi detaljan prikaz stroja u takvom formatu bio računski intenzivan, u radu su korišteni pojednostavljeni modeli.

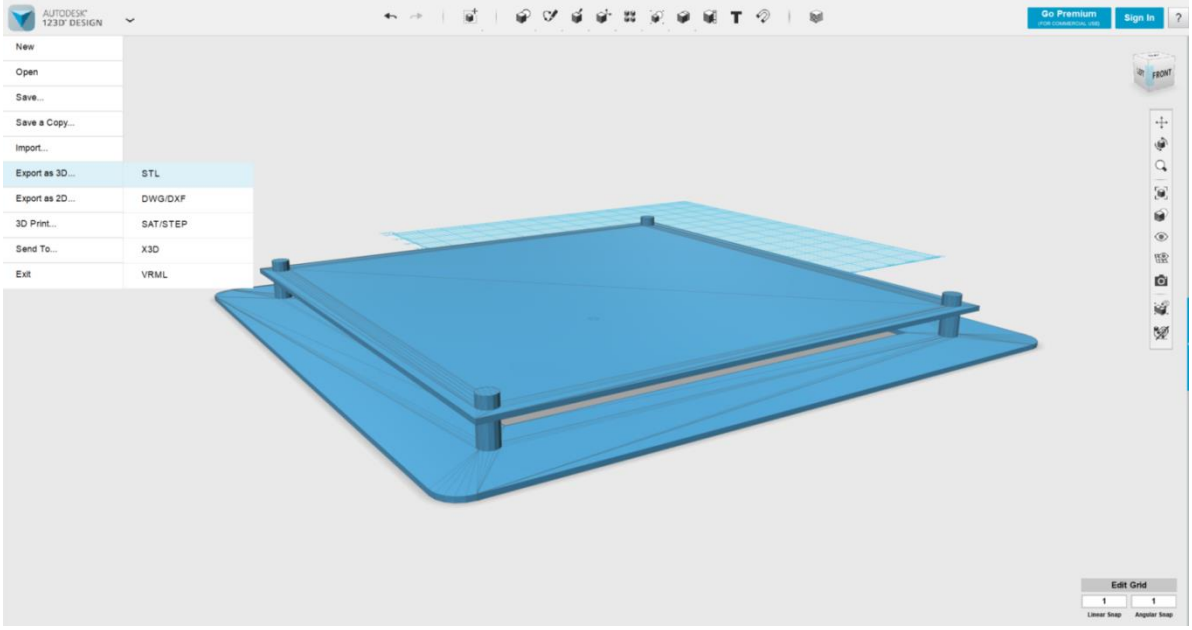


Slika 26. STEP model 3D pisača

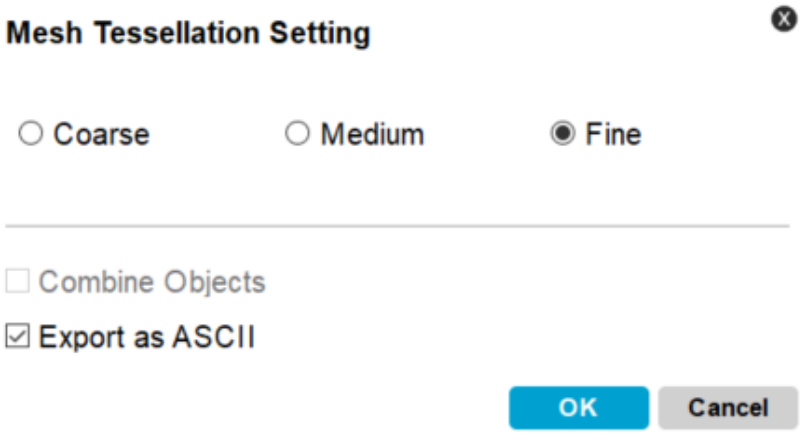
FreeCAD ne podržava spremanje modela u ASCII STL format pa su oni spremljeni u binary STL kojeg LinuxCNC ne zna interpretirati. Postoji više opcija prebacivanja takve datoteke u ASCII. Kao što je već spomenuto koristiti će se samo besplatni programi pa je odabran Autodesk 123D Design.

STL datoteke se učitaju u program i prema slici 38. se prebacuju u ASCII STL format.

U padajućem izborniku desnom tipkom miša odabere se „*Export as 3D*“, zatim „STL“ i u prozoru koji se tada otvara potrebno je označiti „*Export as ASCII*“ kućicu [Slika 28.].



Slika 27. Učitavanje 3D modela i generiranje .stl datoteke *(Autodesk 123D Design)*



Slika 28. Odabir parametara za pohranu STL datoteke

4.2.2. Modul za vizualizaciju stroja u LinuxCNC-u s pripadajućim izvodnicama (engl. Pins)

U sklopu rada razvijen je modul za vizualizaciju procesa 3D tiska prikazan i objašnjen u ovom poglavlju. Koristiti će se simulacija kako bi se eventualne greške u upravljačkom sustavu prepoznale prije spajanja stvarnog hardvera.

```
#!/usr/bin/python

from vismach import *
import hal

c = hal.component("printergui")

c.newpin("y-os", hal.HAL_FLOAT, hal.HAL_IN)
c.newpin("z-os", hal.HAL_FLOAT, hal.HAL_IN)
c.newpin("x-os", hal.HAL_FLOAT, hal.HAL_IN)
c.newpin("ekstruder", hal.HAL_FLOAT, hal.HAL_IN)
c.ready()

kuciste = AsciiSTL(filename="/home/hv2408/Desktop/kuciste.stl")
kuciste = Color([0.3,0.3,0.3,1],[kuciste])
kuciste = Translate([kuciste],0,0,-300)
kuciste = Rotate([kuciste],270,0,1,0)

ploca = AsciiSTL(filename="/home/hv2408/Desktop/ploca2ascii.stl")
ploca = Color([0.3,0.3,0.3,1],[ploca])
ploca = Rotate([ploca],-90,0,1,0)
ploca = Rotate([ploca],-90,1,0,0)
ploca = Translate([ploca],-200,0,0)

ploca = HalTranslate([ploca],c,"z-os",1,0,0)

vodilica = AsciiSTL(filename="/home/hv2408/Desktop/vodilica.stl")

vodilica = Rotate([vodilica],90,0,0,1)
vodilica = Rotate([vodilica],90,0,1,0)
vodilica = Translate([vodilica],-370,2,0)
vodilica = Rotate([vodilica],90,0,0,1)
vodilica = Translate([vodilica],1.85,0,14.25)
vodilica = Rotate([vodilica],-90,0,0,1)

vodilica = HalTranslate([vodilica],c,"y-os",0,0,1)

glava = AsciiSTL(filename="/home/hv2408/Desktop/ekstruder2ascii.stl")
glava = Color([0.6,0.6,0.6,1],[glava])
glava = Rotate([glava],90,0,0,1)
glava = Rotate([glava],90,0,1,0)
glava = Translate([glava],-370,0,15)

glava = HalTranslate([glava],c,"x-os",0,1,0)
glava = HalTranslate([glava],c,"y-os",0,0,1)
```

```
46
47 work = Capture()
48
49 tooltip = Capture()
50
51 vrh = AsciiSTL(filename="/home/hv2408/Desktop/vrhASCII.stl")
52 vrh = Collection([tooltip,vrh])
53 vrh = Color([0.5,0.5,0.5,1],[vrh])
54 vrh = Rotate([vrh],90,0,1,0)
55 vrh = Translate([vrh],-320,0,15)
56
57 vrh = HalTranslate([vrh],c,"x-os",0,1,0)
58 vrh = HalTranslate([vrh],c,"y-os",0,0,1)
59
60 printer = Collection([vodilica,glava,ploca,kuciste,vrh])
61 model = Collection([tooltip, printer, work])
62 model = Rotate([model],270,0,0,1)
63
64 main(model, tooltip, work, 1350)
```

Prvi redak definira interpreter skripte (modula), koji je u ovom slučaj Python verzije 2.7.

Četvrti red skripte služi za inicijalizaciju sučelja prema HAL-u. Drugim riječima, deklariraju se objekti koji će nakon inicijalizacije biti vidljivi u HAL sloju kao izvodnice, odnosno nožice modula za vizualizaciju. Tijekom inicijalizacije potrebno je obratiti pozornost na vrstu izvodnice, tj. na tip i smjer podataka koji će se izmjenjivati u HAL prostoru. U ovom slučaju, sve su izvodnice ulazne (modul samo prihvaća podatke) i deklarirani su kao realni brojevi (engl. *float*).

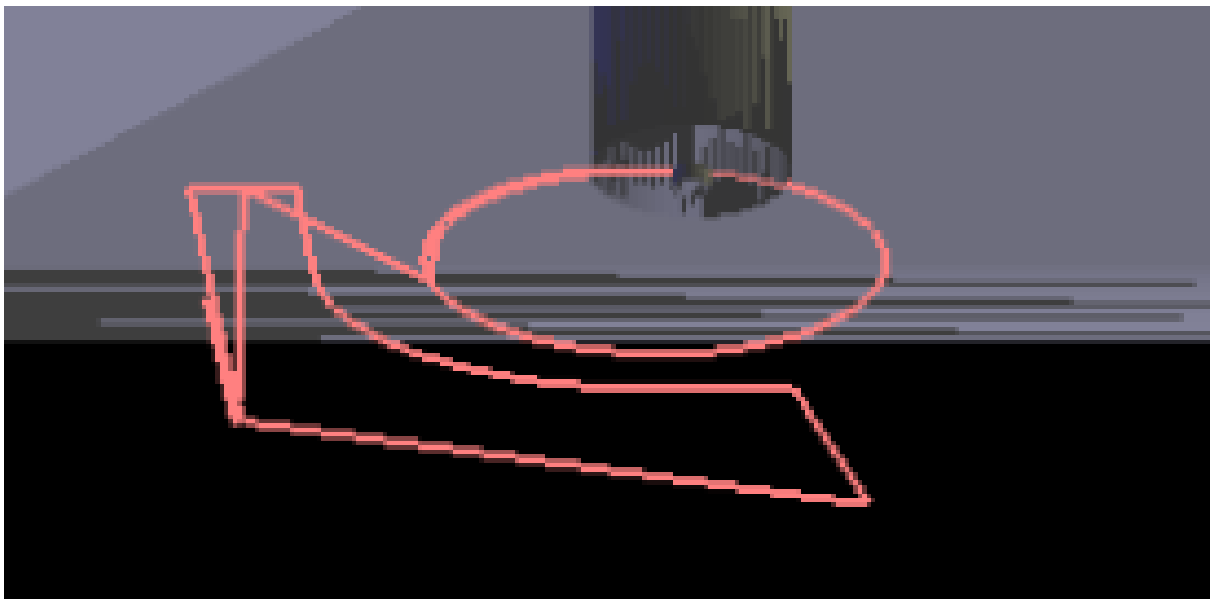
Inicijalizacija modula u HAL prostoru započinje naredbom „hal.component“ (6. redak) gdje se jedinim argumentom definira naziv modula u HAL prostoru, a završava naredbom „ready()“ (12. red)

Dio skripte od 14. do 17. retka prikazuje kako se učitava model, u ovom slučaju kućište. Za tu svrhu se koristi funkcija „AsciiSTL=()“ čiji je argument putanja (engl. *Path*) do datoteke. Idući red (15.) definira boju modela i pridružena su mu četiri argumenta. Prva tri broja definiraju boju prema „RGB“ sustavu, a zadnji broj je postotak boje, „1“ znači udio od 100%, a npr. broj „0.4“ bi označavao 40%.

Funkcija iz 16. retka „Translate“ translatira komad unutar vizualnog prostora u smjeru zadanih stupnjeva slobode. Pridružena su joj tri argumenta koji predstavljaju iznose pomaka u smjeru tih stupnjeva slobode.

Funkcija „Rotate“ služi za rotiranje modela oko željene osi. Sadrži četiri argumenta, prvi definira broj stupnjeva za koji se model zakreće, a iduća tri predstavljaju jedinične vektore u smjerovima slobode gibanja i pomoću njih se izabire os oko koje će se ta rotacija izvršiti.

Redosljed upisivanja tih funkcija je bitan. Referentni koordinatni sustav za rotaciju modela kreće iz točke u koju je model inicijalno učitan pa se može dogoditi u slučaju da se prvo model translatira pa tek onda rotira da ga se rotira na nekoj udaljenosti oko te osi. U tom slučaju problem najčešće rješava pomicanje funkcije „Rotate“ ispred funkcije „Translate“.
Dio skripte od 19. do 25. retka služi za učitavanje modela ploče.
Redak 26. Kod predmetnog pisača pomak po z-osi ostvaruje se pomicanjem podloge (ploče) prema gore ili prema dolje. Da bi se takvo gibanje u simulaciji prikazalo koristi se naredba „HalTranslate“. Ona također dolazi uz četiri argumenta. „z-os“ označava HAL izvodnicu iz koje se učitava željeni položaj z-osi. Preostala tri argumenta označavaju jedinične vektore osi X, Y i Z, odnosno smjerove u kojima će se translacija izvršiti.
Na isti način izvedeno je pozicioniranje i kontrola prigona vodilice (od 27. do 36. redka).
Dio skripte između 38. i 42. retka učitava i pozicionira ekstruder (glavu).
Ekstruder ima dva stupnja slobode gibanja. Postoji više načina kako prikazati takvo gibanje. Budući da se mora ostvariti zajedničko gibanje ekstrudera i vodilice u jednom smjeru slobode gibanja na ekstruder je spojen isti signal koji dobiva i vodilica. Na taj način je osigurano da će se vodilica i ekstruder kretati u ravnini sukladno konstrukciji stvarnog stroja (45. redak).
Za razliku od vodilice, glava se samostalno kreće uzduž iste. Zato je potrebno iskoristiti još jednom „HalTranslate“ naredbu da bi se ostvario dodatni stupanj slobode gibanja (44. redak).
U dijelu skripte od 47. do 58. redka cilj je vizualno prikazati obris alata kao na slici 29.



Slika 29. ***Backplot***

U FreeCAD programskom paketu je izrađen model zanemarivih dimenzija imena „vrh“ te je on pozicioniran na vrh mlaznice ekstrudera. Redak „tooltip = Capture()“ koristi se za prikaz putanje alata. Nakon pozicioniranja na željeno mjesto na njega su spojeni isti signali kao i na ekstruder jer je ideja ovog dijela skripte upravo da se prikaže gibanje vrha mlaznice ekstrudera kroz prostor.

„vrh = Collection([tooltip,vrh])“; povezuje „tooltip“ s vrhom mlaznice kako bi se gibali kao jedan komad.

Redak 60. definira koji sve komadi ulaze u model koji će se učitati u virtualni prostor.

Idući redak (61.) taj model spaja sa slobodnim varijablama koje su potrebne da bi se model mogao učitati.

64. linija skripte služi da se prikaže model printera u prozoru, argument „1350“ definira samo udaljenost inicijalnog pogleda od centra modela kod prvog pokretanja.

Da bi se skripta mogla učitati kao model, nužno je postaviti atribut datoteke za dozvolu izvršavanja [slika 30.].

```
hv2408@linuxcnc-hv:~$ sudo su
[sudo] password for hv2408:
root@linuxcnc-hv:/home/hv2408# chmod +x /home/hv2408/printergui
```

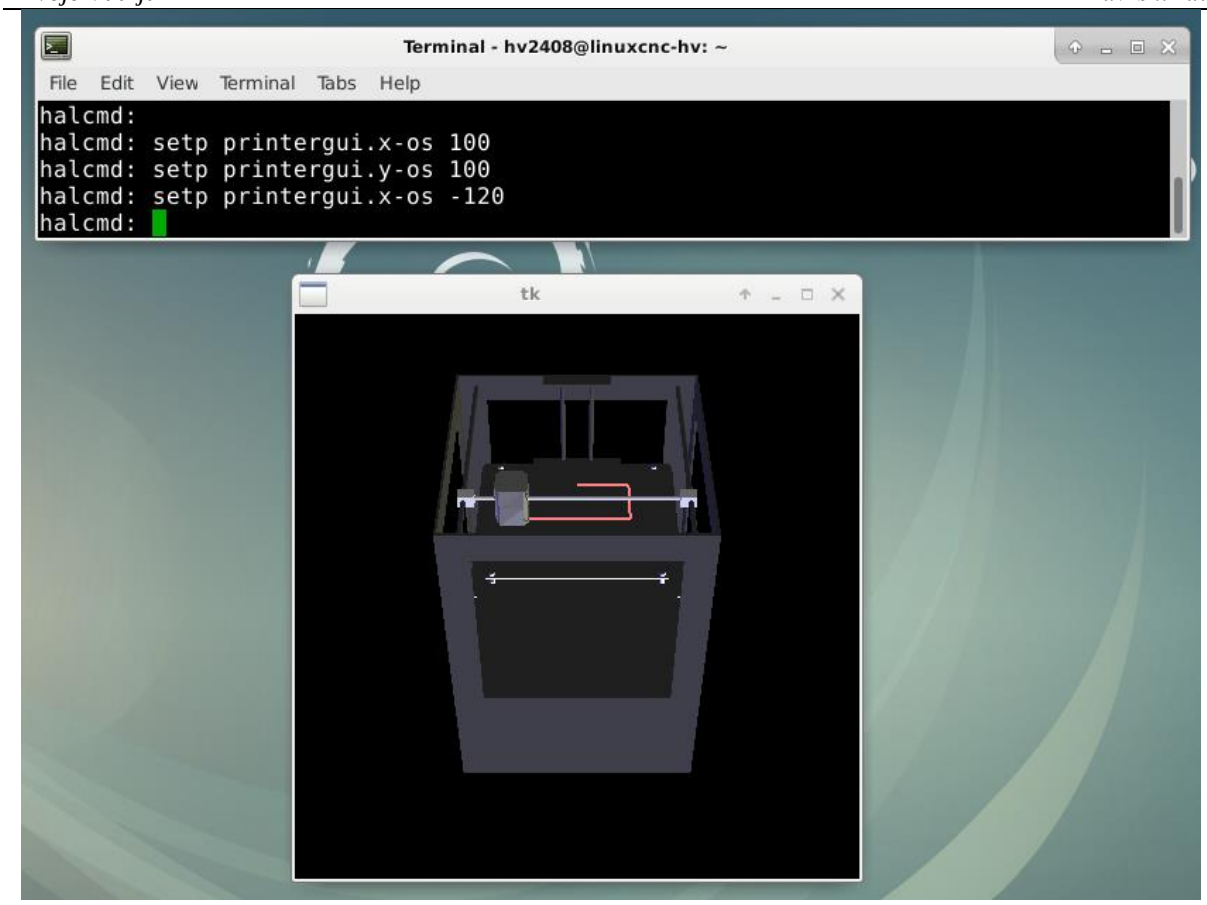
Slika 30. Dozvola izvršavanja

Razvijeni je modul moguće testirati neovisno o ostatku LinuxCNC sustava unutar HAL okruženja. Za tu je svrhu dovoljno pokrenuti naredbu „halrun“ i potom učitati modul naredbom „loadusr“. [Slika 31.]

```
hv2408@linuxcnc-hv:~$ halrun
halcmd: loadusr /home/hv2408/printergui
```

Slika 31. Pokretanje skripte

Rezultat će biti prozor s prikazanim 3d modelom printera. Pojedine prigone je potom moguće pomicati postavljanjem vrijednosti odgovarajućih HAL izvodnica učitanog modula na željene vrijednosti naredbom „setp“ [Slika 32.].



Slika 32 3D tiskač

4.2.3. INI datoteka

INI datoteka sastoji se od sekcija koje sadrže različite varijable.

INI datoteka je skup varijabli bitnih za konfiguraciju sustava, te varijable se nalaze na jednom mjestu i sortirane su u različite funkcionalne sekcije. U uglatim zagradama nalazi se “glava” sekcije, a unutar nje različite varijable koje su joj pridružene.

```
[EMC]
VERSION =               $Revision$
MACHINE =               Printer
DEBUG =                 0

[DISPLAY]
DISPLAY =              axis
CYCLE_TIME =            0.200
POSITION_OFFSET =       RELATIVE
POSITION_FEEDBACK =     ACTUAL
MAX_FEED_OVERRIDE =     2.0
PROGRAM_PREFIX = /home/hv2408/linuxcnc/nc_files
INTRO_GRAPHIC = linuxcnc.gif
INTRO_TIME = 5
EDITOR = gedit

[TASK]
TASK =                  milltask
CYCLE_TIME =            0.010

[RS274NGC]
PARAMETER_FILE =        printer.var

[EMCMOT]
EMCMOT =               motmod
COMM_TIMEOUT =          1.0
COMM_WAIT =             0.010
SERVO_PERIOD =               1000000

[HAL]
HALFILE =                    printer.hal


```

```
[TRAJ]
AXES =                  3
COORDINATES =           X Y Z
HOME =          0 0 0 0
LINEAR_UNITS =          mm
ANGULAR_UNITS =         degree
CYCLE_TIME =            0.010
DEFAULT_VELOCITY =      3.0
MAX_VELOCITY =          4.0
MAX_ACCELERATION =      8.0
DEFAULT_ACCELERATION =  8.0

[AXIS_0]
#VODILICA
TYPE =                          LINEAR
HOME =                          0.000
MAX_VELOCITY =                  4.0
MAX_ACCELERATION =              12.0
MIN_LIMIT =                     -130
MAX_LIMIT =                     130
HOME_SEARCH_VEL =               -3.0

[AXIS_1]
#EKSTRUDER
TYPE =                          LINEAR
HOME =                          0
MAX_VELOCITY =                  4.0
MAX_ACCELERATION =              12.0
MIN_LIMIT =                     -130
MAX_LIMIT =                     130
HOME_SEARCH_VEL =               -3.0

[AXIS_2]
#PLOCA
TYPE =                          LINEAR
HOME =                          100
MAX_VELOCITY =                  4.0
MAX_ACCELERATION =              8.0
MIN_LIMIT =                     -130
MAX_LIMIT =                     200
HOME_SEARCH_VEL =               3.0

[EMCIO]
EMCIO =                 io
CYCLE_TIME =                    0.100
```

EMC sekcija sadrži općenite informacije o konfiguraciji. U prvom retku navodi se ime verzije, a u trećem retku definira se ime kontrolera. Varijabla iz petog retka, „DEBUG“, kojoj je pridružena vrijednost „0“ isključuje prikazivanje upozorenja kada se konfiguracija uključuje iz terminala.

Između šestog i petnaestog retka upisana je sekcija prikaza (engl. Display) koja sadrži postavke povezane s grafičkim sučeljem. Sedma linija skripte definira ime grafičkog sučelja

koje će se koristiti. Iduća linija (8.) definira brzinu osvježavanja prikaza. Deveti i deseti redak utječu na udaljenosti položaja alata. U dvanaestom retku određena je inicijalna lokacija G-kodova. Trinaesti i četrnaesti redak učitavaju neki grafički prikaz kod pokretanja konfiguracije i određuju vrijeme trajanja tog prikaza. Petnaesti redak određuje koji se uređivač teksta koristiti kod uvođenja ili mijenjanja varijabli u INI datoteku.

TASK sekcija. „milltask“ pridružen u osamnaestom retku je uvijek isti i ne utječe na konfiguraciju, ali ga je potrebno upisati (ostalo iz prošlih verzijaLinuxCNC-a). U devetnaestom retku se u sekundama definira period čekanja između dva koraka.

RS274NGC je interpreter koji se koristi za izvršavanje G-koda i u dvadeset drugom retku definira se ime datoteke u koju će se spremati varijable direktno iz G-koda.

U EMCMOT definiraju se parametri kontrolnog modula koji se izvršava u stvarnom vremenu. Dvadeset peti redak navodi naziv kontrolera koji se koristi. Iduća tri retka definiraju periode izvršavanja u nanosekundama.

HAL sekcija sadrži ime HAL skripte koja se koristi u konfiguraciji (31. redak).

TRAJ sekcija sadrži dodatne parametre potrebne za kontrolu gibanja. U trideset petom retku navodi se broj stupnjeva slobode gibanja te u idućem retku koordinatne osi koje se poklapaju s tim stupnjevima slobode gibanja. Nakon toga odabiru se mjerne jedinice (redak 38. i 39.), period izvršavanja (redak 40.), inicijalna i maksimalna brzina (redci 41. i 42.), maksimalna akceleracija (redak 43.) i inicijalno ubrzanje (redak 44.).

Slijede tri sekcije imena AXIS_N od 46. do 95. retka koje definiraju parametre vezane uz svaki stupanj slobode gibanja pojedinačno. Opisat će se prva sekcija AXIS_0. Definira se tip osi koja može biti linearna (engl. *linear*) ili kutna (engl. *angular*) u retku 48. Zatim se definira vrijednosti nultočke za taj stupanj slobode gibanja (redak 49.) Maksimalna brzina (redak 50.), maksimalno ubrzanje (redak 51.), veličina radnog prostora (redci 52. i 53.), brzina gibanja u slučaju traženja nultočke (redak 54.),

EMCIO sekcija navodi naziv programa IO kontrolera u sedamdeset sedmom retku. Zadnji redak skripte (78.) definira period u kojem će se IO kontroler izvršavati. U slučaju unošenja negativnog broja kontroler neće uopće mirovati.

4.2.4. HAL skripta

LinuxCNC sadrži neke primjere takvih simulacija. „Horizontal Boring Mill Simulation“ je isto prikaz stroja s tri stupnja slobode gibanja i koristi prikladan kontroler. Po uzoru na HAL skriptu te simulacije napisane su konfiguracijske datoteke za razmatrani pisač.

```
loadrt [EMCMOT]EMCMOT servo_period_nsec=[EMCMOT]SERVO_PERIOD num_joints=[TRAJ]AXES

loadrt trivkins

loadrt limit3 count=4

loadrt wcomp count=4
loadrt not count=4

loadusr -W printergui

addf limit3.0 servo-thread
addf limit3.1 servo-thread
addf limit3.2 servo-thread
addf limit3.3 servo-thread
addf wcomp.0 servo-thread
addf wcomp.1 servo-thread
addf wcomp.2 servo-thread
addf wcomp.3 servo-thread
addf not.0 servo-thread
addf not.1 servo-thread
addf not.2 servo-thread
addf not.3 servo-thread

setp limit3.0.maxv [AXIS_0]MAX_VELOCITY
setp limit3.0.maxa [AXIS_0]MAX_ACCELERATION
setp limit3.1.maxv [AXIS_1]MAX_VELOCITY
setp limit3.1.maxa [AXIS_1]MAX_ACCELERATION
setp limit3.2.maxv [AXIS_2]MAX_VELOCITY
setp limit3.2.maxa [AXIS_2]MAX_ACCELERATION

setp wcomp.0.min -140
setp wcomp.0.max  140
setp wcomp.1.min -140
setp wcomp.1.max  140
setp wcomp.2.min -120
setp wcomp.2.max  240

net vodilica-pos-cmd  axis.0.motor-pos-cmd => limit3.0.in
net glava-pos-cmd   axis.1.motor-pos-cmd => limit3.1.in
net ploca-pos-cmd axis.2.motor-pos-cmd => limit3.2.in

net vodilica-pos-fb  limit3.0.out => axis.0.motor-pos-fb printergui.y-os  wcomp.0.in
net glava-pos-fb   limit3.1.out => axis.1.motor-pos-fb printergui.x-os   wcomp.1.in
net ploca-pos-fb limit3.2.out => axis.2.motor-pos-fb printergui.z-os wcomp.2.in

net vodilica-limits-ok wcomp.0.out => not.0.in
net vodilica-limits-tripped not.0.out => axis.0.neg-lim-sw-in axis.0.pos-lim-sw-in axis.0.home-sw-in
net glava-limits-ok wcomp.1.out => not.1.in
net glava-limits-tripped not.1.out => axis.1.neg-lim-sw-in axis.1.pos-lim-sw-in axis.1.home-sw-in
net ploca-limits-ok wcomp.2.out => not.2.in
net ploca-limits-tripped not.2.out => axis.2.neg-lim-sw-in axis.2.pos-lim-sw-in axis.2.home-sw-in
```

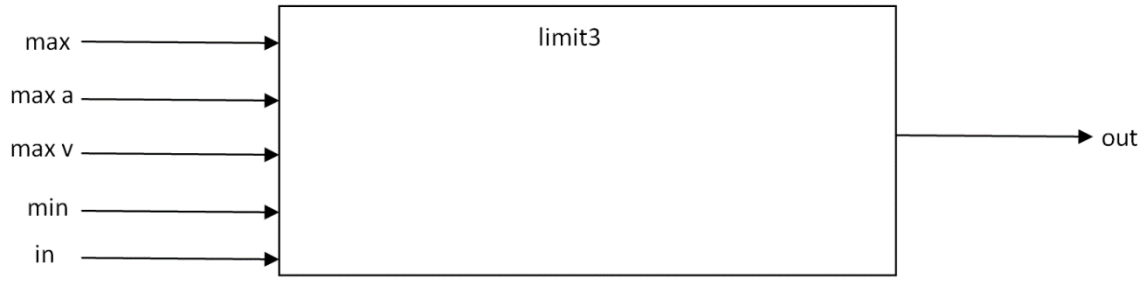
Na početku skripte prvo je potrebno učitati sve potrebne module koji će se izvršavati u stvarnom vremenu (engl. *real time modul*) pomoću naredbe „loadrt“. Grubo objašnjenje real time modula bilo bi da se on izvršava preciznim vremenskim intervalima s minimalnim vremenskim latencijama.

Prva linija skripte učitava modul za kontrolu gibanja, te varijable potrebne za definiranje istog učitava iz INI datoteke. Tim modulom se definiraju periodi osvježavanja svih preostalih modula koji će se izvršavati u stvarnom vremenu.

Treći redak učitava „trivkins“ modul. To je poseban modul, nema ulaze i izlaze, ali je nužan za postavljanje kinematske strukture stroja koja je u ovom slučaju trivijalna. Svaki stroj čiji se stupnjevi slobode gibanja međusobno poklapaju s kartezijskim koordinatnim sustavom predstavljaju primjer trivijalne kinematike.

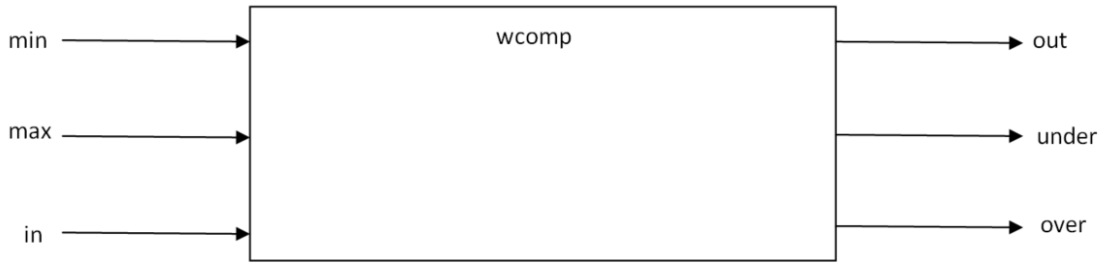
U petom retku se učitava modul „limit3“ [Slika 33.]. Služi za simulaciju dinamike gibanja prigona s mogućnošću definiranja maksimalne akceleracije, brzine i puta.

Argument „count“ govori koliko će se takvih modula učitati.



Slika 33. Vizualizacija *Limit 3* modula

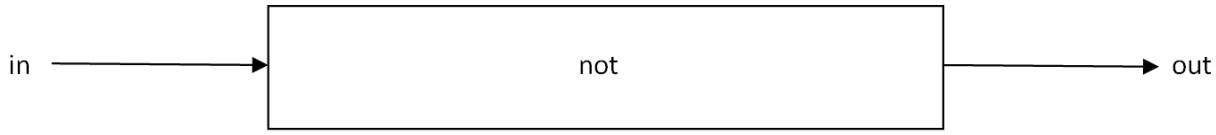
Sedmi redak učitava „wcomp“ [Slika 34.] modul (komparator). Modul sadrži tri izvodnice ulaznog tipa; „in“, „max“ i „min“. „in“ predstavlja vrijednost ulazne veličine koja se želi kontrolirati u nekom trenutku npr. poziciju osi. Sadrži i tri izlazne BIT izvodnice. Komparator uspoređuje vrijednosti ulaza i čitavo vrijeme kada se vrijednost „in“ održava unutar vrijednosti „min“ i „max“ stanje izvodnice „out“ u 1. U trenutku kada vrijednost „in“ bude veća od zadane maksimalne, odnosno manja od minimalne komparator šalje signal na izvodnicu „out“ koji u tom trenu više nije 1 već postaje 0. Šalje i signal na nožicu „under“ ili „over“ ovisno o tome je li vrijednost veća ili manja od granične.



Slika 34. Vizualizacija *wcomp* modula

Smisao uvođenja ovog modula je ograničenje gibanja prema radnom volumenu modela stroja. Ukoliko se iz bilo kojeg razloga bilo koji prigon zamakne izvan granica definiranih „min“ i „max“ izvodnicama, modul će postaviti stanja svojih izlaznih izvodnica „under“ ili „over“ u stanje „1“, a taj se signal može iskoristiti za simulaciju graničnih prekidača.

U osmom retku učitava se „not“ modul [Slika 35.], to je inverter i radi na principu dva BIT signala koji mogu biti vrijednosti „0“ ili „1“. Kada je ulaz jednak „1“ izlaz je jednak „0“ i obrnuto.



Slika 35. Vizualizacija *not* modula

Deseti redak učitava 3D tiskač iz prethodnog poglavlja i njegove izvodnice u HAL prostor.

Od dvanaestog do dvadeset četvrtog retka predmetni moduli pridružuju se „thread“ modulu naziva „servo-“, petlji čiji je period u NC ranije definiran tijekom učitavanja EMCMOT modula.

U idućim koracima definiraju se parametri za učitane module.

Dio skripte napisan između dvadeset petog i tridesetog retka učitava iz INI datoteke parametre u „limit3“ modul. Dodaju se parametri za svaku os; maksimalna brzina i maksimalno ubrzanje. Taj modul će na temelju ulaznih parametara dobro simulirati kinematiku stvarnog hardvera. Nakon toga pridružuju se parametri „wcomp“ modulu.

U dijelu skripte od trideset devetog do četrdeset drugog retka generira se HAL signal koji povezuje NC jezgru i „limit3“ modul, čime se simulira odziv modula na zadanu referencu. Neki učitani G-kod se interpretira i iz NC jezgre dolaze reference pozicija za svaku os pojedinačno koje se zovu „axis.N.motor-pos-cmd“ pri čemu „N“ definira stupanj slobode gibanja. NC jezgra očekuje povratnu vezu naziva „axis.N.motor-pos-fb“ da bi se utvrdilo koja je stvarna pozicija osi.

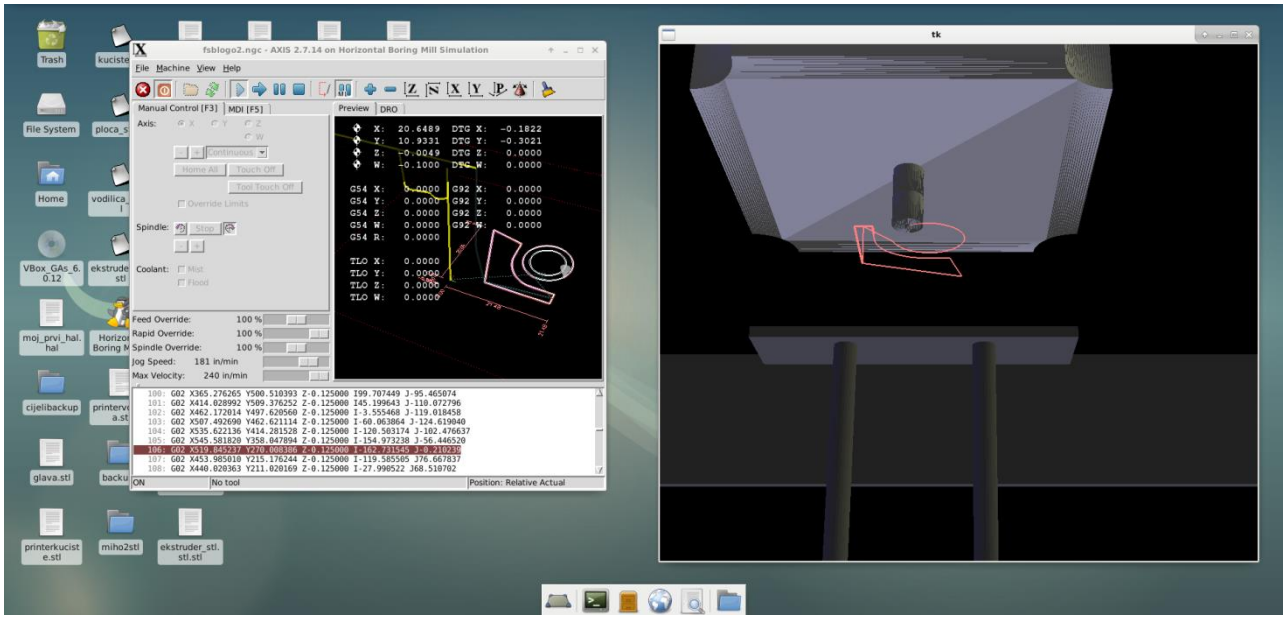
Od četrdeset trećeg do četrdeset šestog redka omogućuje da izlaz iz „limit3“ modula daje povratnu vezu koordinirajućem modulu naziva „axis.N.motor-pos-fb“. Taj isti signal šalje se na izvodnice modula za vizualizaciju „printergui“ kako bi se vizualno prikazalo stanje proces, učitava se i u „wcomp“ modul kojim se simuliraju granični prekidači.

Zadnji dio, od četrdeset sedmog retka do zadnje linije skripte spaja „wcomp“ i „not“ modul, te „not“ modul s koordinirajućim modulom. Kada neka vrijednosti izađe iz prozora „wcomp“ modula (u prozor spadaju sve vrijednosti između „min“ i „max“ u „wcomp“ modulu), „not“ modul invertira vrijednost izlaza i na taj način šalje signal u koordinirajući modul da je aktiviran granični prekidač, odnosno da je neki od prigona došao do kraja svog radnog prostora i da je stroj potrebno zaustaviti.

5. Simulacija 3D pisača

U terminalu se pomoću naredbe “LinuxCNC” pokreće INI datoteka. Učitati će se G-kod za izradu testnog modela.

Slika 36 Naredba za pokretanje simulacije 3D tiskača



Slika 37 Simulacija 3D tiskača

6. Zaključak

U ovom radu opisana je primjena LinuxCNCs ustava u razvoju upravljačkog sustava za 3D pisač. Za tu je svrhu prvo izrađen 3D model pisača, a zatim je na osnovi tog modela izrađen i virtualni simulacijski model kako bi se sustav mogao testirati prije spajanja stvarnog hardvera.

Upravljački sustav je djelomično testiran, a dobiveni rezultati pokazali su da upravljački sustav radi i da je u mogućnosti interpretirati te izvršiti učitani G-kod za sva tri glavna stupnja slobode gibanja.

Svi ključni elementi LinuxCNC sustava, parametri, kao i odgovarajuće konfiguracijske datoteke detaljno su objašnjeni u radu. Pri tome je osvrt dan na parametre koji su u užem smislu vezani za FDM proces.

Iako ovaj rad ne prezentira puni potencijal takvog pristupa, naglašava njegove ključne prednosti. Nastavak ovog rada biti će usmjeren integraciji hardverskih komponenti u upravljački sustav pisača.

7. LITERATURA

[1] www.all3dp.com/history-of-3d-printing

[2] http://www.plastikainfo.com/tehnologija/materijali-abs-akrilonitril-butadien

[3] Pilipović A. *Utjecaj parametara izrade na svojstva polimernoga prototipa* 2012.

[4] https://my3dmatter.com/influence-infill-layer-height-pattern/

[5] http://wiki.linuxcnc.org/cgi-bin/wiki.pl

[6] T. Staroveški, D. Brezak, T. Udiljak, D. Majetić *Implementation of a Linux-based CNC open control system*, 2009.

[7] T. Staroveški, D. Brezak, T. Udiljak *LinuxCNC - the enhanced machine controller: application and an overview*, 2013.

[8] https://opensource.com/alternatives/autocad

[9] https://www.3dhubs.com/knowledge-base/impact-layer-height-3d-print/