

# 3D rotacijski skener

---

**Brzica, Karlo**

**Undergraduate thesis / Završni rad**

**2019**

*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://um.nsk.hr/um:nbn:hr:235:118790>

*Rights / Prava:* [In copyright](#)/[Zaštićeno autorskim pravom.](#)

*Download date / Datum preuzimanja:* **2024-07-24**

*Repository / Repozitorij:*

[Repository of Faculty of Mechanical Engineering and Naval Architecture University of Zagreb](#)



SVEUČILIŠTE U ZAGREBU  
FAKULTET STROJARSTVA I BRODOGRADNJE

# ZAVRŠNI RAD

**Karlo Brzica**

Zagreb, 2019.

SVEUČILIŠTE U ZAGREBU  
FAKULTET STROJARSTVA I BRODOGRADNJE

# ZAVRŠNI RAD

Mentori:

Prof. dr. sc. Mladen Crneković, dipl. ing.

Student:

Karlo Brzica

Zagreb, 2019.

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

Zahvaljujem se profesoru Mladenu Crnekoviću za pruženo povjerenje i praćenje izrade ovog rada, te kolegama i prijateljima na korisnim savjetima.

Na kraju, zahvaljujem se svojoj obitelji na pruženoj ljubavi, povjerenju i podršci tokom cijelog studija.

Karlo Brzica



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	Prilog
Klasa:	
Ur.broj:	

## ZAVRŠNI ZADATAK

Student: **KARLO BRZICA** Mat. br.: 0035206500

Naslov rada na hrvatskom jeziku: **3D ROTACIJSKI SCANER**

Naslov rada na engleskom jeziku: **3D ROTATING SCANNER**

Opis zadatka:

Izrada 3D modela nekog složenog predmeta u CAD alatima može zahtijevati iskusnog konstruktora i biti dugotrajna. Umjesto izrade modela CAD alatima predmet možemo skenirati te pomoću dobivenog oblaka točaka rekonstruirati njegov oblik.

Potrebno je konstruirati uređaj koji bi radio kao scanner za predmete veličine do 30 cm, a imao bi dva stupnja slobode gibanja (rotacija predmeta i translacija senzora).

U radu je potrebno:

- konstruirati mehanički sustav uređaja,
- odabrati motore i njihove upravljačke uređaje,
- odabrati senzor za scanner i centralni procesor,
- napisati korisničko sučelje i procijeniti vrijednost uređaja.

Potrebno je navesti korištenu literaturu i ostale izvore informacija, te eventualno dobivenu pomoć.

Zadatak zadan:

29. studenog 2018.

Zadatak zadao:

Prof.dr.sc. Mladen Crneković

Rok predaje rada:

1. rok: 22. veljače 2019.

2. rok (izvanredni): 28. lipnja 2019.

3. rok: 20. rujna 2019.

Predviđeni datumi obrane:

1. rok: 25.2. - 1.3. 2019.

2. rok (izvanredni): 2.7. 2019.

3. rok: 23.9. - 27.9. 2019.

Predsjednik Povjerenstva:

Prof. dr. sc. Branko Bauer

## SADRŽAJ

SADRŽAJ .....	I
POPIS SLIKA .....	III
POPIS TABLICA.....	IV
POPIS TEHNIČKE DOKUMENTACIJE .....	V
POPIS OZNAKA .....	VI
SAŽETAK.....	VII
SUMMARY .....	VIII
1. UVOD.....	1
2. OPĆENITO O 3D SKENERIMA .....	2
2.1. Područje primjene 3D skenera .....	2
2.2. Tehnologije i metode 3D skeniranja .....	2
2.2.1. Kontaktna metoda .....	2
2.2.2. Time of flight (TOF) metoda .....	3
2.2.3. Triangulacijska metoda .....	4
2.2.4. Metoda strukturiranog svjetla .....	4
3. IZRADA 3D ROTACIJSKOG SKENERA .....	5
3.1. Slični tipovi skenera.....	5
3.1.1. Ciclop 3D scanner - BQ.....	5
3.1.2. V2 3D scanner - Matter & Form.....	5
3.1.3. Usporedba skenera i cilj novog skenera.....	6
3.2. Princip rada .....	7
3.3. Opis korištenih komponenti .....	9
3.3.1. Mikrokontroler .....	9
3.3.2. VL6180X senzor .....	9
3.3.2.1. Ispitivanje točnosti senzora.....	11
3.3.3. Koračni motor .....	15
3.3.4. Driver .....	15
3.3.5. Memorijski modul.....	16
3.3.6. Korisničko sučelje.....	17
3.4. Konstrukcijska izvedba .....	18
3.4.1. Kućište .....	18
3.4.2. Postolje.....	18
3.4.3. Nosač postolja .....	19
3.4.4. Prijenosnik .....	19
3.4.5. Nosač senzora .....	20
3.4.6. Nosač motora .....	20
3.5. Program.....	21
3.5.1. Inicijalizacija .....	21
3.5.2. Glavna petlja .....	22
3.5.2.1. Referenciranje .....	22
3.5.2.2. Nakon pritiska tipkala start .....	24

---

3.5.3. Okretanje postolja .....	24
3.5.4. Okretanje vretena .....	25
3.5.5. Funkcija čitanje .....	26
4. PROCJENA VRIJEDNOSTI .....	28
5. PROVEDENA SKENIRANJA I REZULTATI .....	29
5.1. Stožac – manji, jednostavan model .....	29
5.2. Tenisica- veći, složeniji model.....	29
5.3. Karton za jaja – veliki složen model .....	31
5.4. Skulptura dupina – manji, složen model .....	32
6. EVALUACIJA REZULTATA SKENIRANJA .....	33
6.1. Karakteristike skenera.....	33
7. ZAKLJUČAK.....	34
LITERATURA.....	35
PRILOZI.....	36

**POPIS SLIKA**

Slika 1.	View from the Window at Le Gras, Joseph Nicéphore Niépce 1826 .....	1
Slika 2.	Taktilne metode skeniranja .....	3
Slika 3.	Putanja laserske zrake .....	3
Slika 4.	Prikaz triangulacijske metode skeniranja .....	4
Slika 5.	Prikaz strukturiranog skeniranja.....	4
Slika 6.	Ciclop .....	5
Slika 7.	V2 .....	6
Slika 8.	3D skener i njegove komponente .....	7
Slika 9.	Mjerenje objekta.....	8
Slika 10.	Arduino Uno.....	9
Slika 11.	Adafruit VL6180X .....	9
Slika 12.	Usporedba rezultata mjerenja udaljenosti tijela različite refleksije.....	10
Slika 13.	Prikaz vidnog područja senzora i dodanog cilindra .....	11
Slika 14.	Mjerenje 200 uzoraka .....	11
Slika 15.	Mjerenje 100 uzoraka .....	12
Slika 16.	Mjerenje 50 uzoraka .....	12
Slika 17.	Mjerenje 25 uzoraka .....	13
Slika 18.	Mjerenje 10 uzoraka .....	13
Slika 19.	Mjerenje 5 uzoraka .....	14
Slika 20.	Dodatna mjerenja 10 uzoraka .....	14
Slika 21.	Koračni motor.....	15
Slika 22.	L298N.....	16
Slika 23.	Memorijski modul .....	16
Slika 24.	Prikaz podataka .....	17
Slika 25.	Upravljačko sučelje .....	17
Slika 26.	Kučište.....	18
Slika 27.	Postolje .....	18
Slika 28.	Nosač postolja .....	19
Slika 29.	Prijenosnik.....	19
Slika 30.	Nosač senzora.....	20
Slika 31.	Nosač motora.....	20
Slika 32.	Definiranje varijabli .....	21
Slika 33.	Glavna petlja 1.dio .....	22
Slika 34.	Referenciranje .....	23
Slika 35.	Glavna petlja 2.dio .....	24
Slika 36.	Okretanje postolja.....	25
Slika 37.	Okretanje vretena .....	26
Slika 38.	Funkcija citanje .....	27
Slika 39.	Stožac – manji, jednostavniji model.....	29
Slika 40.	Tenisica, veći složeniji model .....	29
Slika 41.	Tenisica drugo skeniranje 1.....	30
Slika 42.	Tenisica drugo skeniranje 2.....	30
Slika 43.	Karton za jaja – veliki složen model .....	31
Slika 44.	Karton za jaja dobiveno skeniranje .....	31
Slika 45.	Karton za jaja dobiveno skeniranje drugi pogled.....	31



---

**POPIS TABLICA**

Tablica 1. Usporedba i cilj skenera .....	6
Tablica 2. Troškovi izrade 3D skenera.....	28
Tablica 3. Karakteristike 3D rotacijskog skenera .....	33

---

**POPIS TEHNIČKE DOKUMENTACIJE**

BROJ CRTEŽA	Naziv iz sastavnice
RS-00-00	Rotacijski skener
RS-00-01	Nosač postolja
RS-00-02	Prijenosnik
RS-00-03	Postolje
RS-00-04	Razmaknica
RS-00-05	Nosač senzora
RS-00-06	Nosač motora
RS-00-07	Shema spajanja
RS-00-08	Tiskana pločica
RS-00-09	Vodilica
RS-01-00	Sklop kućišta
RS-01-01	Poklopac
RS-01-02	Bočna stranica
RS-01-03	Podnica
RS-01-04	Leđna ploča
RS-01-05	Bočna gornja ploča
RS-01-06	Prednja ploča

---

**POPIS OZNAKA**

<b>Oznaka</b>	<b>Jedinica</b>	<b>Opis</b>
$c$	m/s	Brzina svjetlosti
$t$	s	Vrijeme
$d$	m	Udaljenost
$d_c$	m	Udaljenost centra rotacije postolja od senzora
$U$	V	Napon
$h$	m	Visina senzora
$P_h$	mm	Korak navojnog vretena
$\alpha$	°	stupanj zakreta koračnog motora postolja
$\beta$	°	stupanj zakreta koračnog motora vretena

---

**SAŽETAK**

Rad ukratko opisuje najpoznatije tehnologije i metode skeniranja kao što su: kontaktna metoda, triangulacijska metoda, metoda strukturiranog svjetla i time of flight metoda. Zatim se opisuje izrada i ispitivanje 3D rotacijskog skenera koji sadrži dva stupnja slobode gibanja. Stupnjevi slobode su rotacija objekta skeniranja te translacija senzora. Senzor koji se koristi je VL5180X i radi na principu time of flight metode skeniranja te mu je deklarirana preciznost od  $\pm 1$  mm. Maksimalni radni prostor 3D rotacijskog skenera je promjera 300 mm i visine 270 mm koja se može dodatno povećati, a brzina skeniranja je ovisna o veličini skeniranog objekta.

Ključne riječi: 3D skener, time of flight, oblak točaka

---

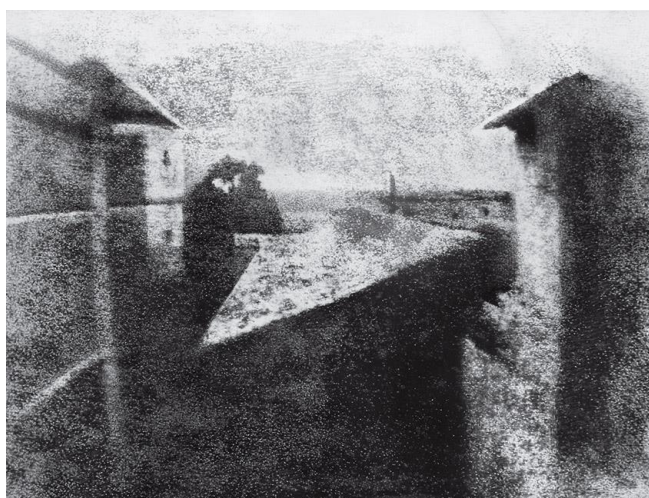
**SUMMARY**

The paper briefly describes best-known scanning methods like contact method, triangulation method, a method of structured light and time of flight method. Afterward is described the production and testing of 3D rotational scanner that consists of two degrees of freedom. One degree of freedom is the rotation of the scanning object and the second one is the translation of the sensor. A sensor used is VL5180X, which works on a time of flight method of scanning, and its stated accuracy is  $\pm 1$  mm. The maximum working area of a 3D rotational scanner is a diameter of 300mm and height of 270 mm, which can be additionally increased. The speed of the scanning process depends on the size of scanning object.

Keywords: 3D scanning, time of flight, cloud of points

## 1. UVOD

Prije otprilike 30 000 godina naši su preci crtali po zidovima pećina i izrađivali figurice i skulpture. Nepoznato je iz kojeg razloga su to radili, no oduvijek je postojala želja odnosno potreba zabilježiti svijet oko sebe. Kroz veći dio povijesti crtanje je bio jedini način kako bi se prikazala naša okolina. To se promijenilo prvi put u prvoj polovici 19. stoljeća, kad je 1826.godine Joseph Nicéphore Niépce, fotografirao prvu fotografiju [Slika 1.].



Slika 1. View from the Window at Le Gras, Joseph Nicéphore Niépce 1826

Time su otvorena vrata razdoblja novih i objektivnijih mogućnosti arhiviranja i dokumentiranja. Naime, fotografiranje i kasnije digitalna obrada slike kao i svaka 2D tehnika dokumentiranja je ograničena prilikom prikazivanja trodimenzionalnog svijeta, a što se svijet više razvijao i tehnika napredovala to je potreba za prikazom trodimenzionalnih objekata i 3D skeniranjem bivala sve veća. [1]

## 2. OPĆENITO O 3D SKENERIMA

Skeniranje je proces mjerenja i snimanja površine postojećeg objekta kako bi se dobio njegov oblik u digitalnom obliku koji je pogodan za kasniju obradu. Samim time 3D skener je uređaj koji različitim metodama snima stvarne objekte i prikuplja podatke o njihovom obliku. U nastavku ovog seminarskog rada su spomenute najčešće metode i načini primjene 3D skeniranja.

### 2.1. Područje primjene 3D skenera

Mnoge grane znanosti, umjetnosti i tehnologije koriste tehnologiju 3D skeniranja kako bi olakšali odnosno uveličali svoj rad. Primjer su arhitektura (prikaz i mjerenje postojećih objekata velikom preciznošću i velikim brojem detalja), povijest i umjetnost (prikaz mjerenje i uspoređivali neke forme kulturološkog nasljeđa), medicina, strojarstvo (mjerenje, kontrola). [2]

### 2.2. Tehnologije i metode 3D skeniranja

#### 2.2.1. Kontaktna metoda

Kontaktna metoda je jedna od prvih metoda 3D skeniranja, a ujedno i najjednostavnija. Korištenjem senzora dodira (sonde), te njegove pozicije i orijentacije u prostoru dobiva se oblak točaka neke skenirane površine. Senzor je moguće postaviti na jednu od tri vrste mehanizma.

Prva vrsta mehanizma [Slika 2. lijevo] je sastavljena od okomito postavljenih vodilica po kojima klizi nosač koji nosi senzor. Ovakav tip konstrukcije je pogodan za skeniranje ravnih profila ili površina sa jednostavnim konveksnim krivuljama.

Druga vrsta mehanizma [Slika 2. desno] je sastavljena od krutih zglobno povezanih segmenata sa visoko preciznim sensorima zakreta u zglobovima. Ovakav tip konstrukcije je idealan za skeniranje rupa i unutrašnjih površina do kojih je teško doprijeti.

Treća vrsta mehanizma je kombinacija prva dva mehanizma. Takvim spajanjem se može postići veće radno područje skenera. [3]



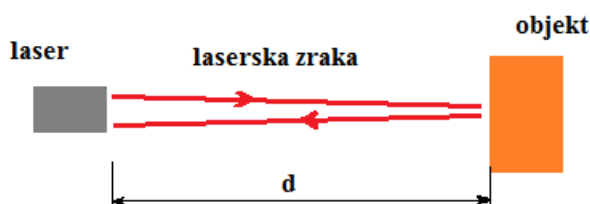
Slika 2. Taktilne metode skeniranja

### 2.2.2. *Time of flight (TOF) metoda*

Ovakva vrsta skenera koristi laserski mjerac udaljenosti kako bi skenirao objekt. Pomoću lasera emitira se svjetlosni impuls, te se mjeri vrijeme  $t$  koje je potrebno da se emitirani impuls odbije od objekta skeniranja i vrati nazad [Slika 3.]. Pomoću dobivenog vremena i poznate brzine svjetlosti  $c$  izračunava se udaljenost  $d$  objekta od skenera.

$$d = \frac{t \cdot c}{2} \quad (1)$$

Laserski mjerac udaljenosti mjeri samo udaljenost točke u smjeru pogleda, prema tome kako bi skener skenirao cijeli objekt mora ga mjeriti iz različitih pogleda. To se može postići rotacijom laserskog daljinomjera, objekta skeniranja ili korištenjem sustava rotirajućih zrcala. Točnost ovakvih skenera ovisi o preciznosti mjerenja vremena  $t$  što je ujedno i nedostatak jer je teško mjeriti male udaljenosti. Prednost ovakvih skenera je što mogu mjeriti velike udaljenosti te se takvi skeneri mogu koristiti za skeniranje velikih objekata poput zgrada ili geografskih objekata. [3]

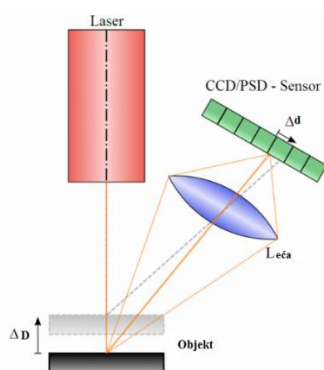


Slika 3. Putanja laserske zrake



### 2.2.3. Triangulacijska metoda

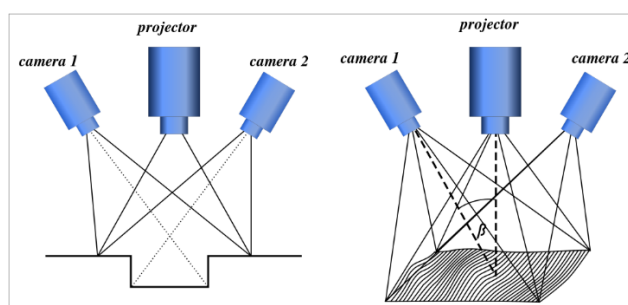
Triangulacijska metoda 3D skeniranja također se koristi laser kako bi skenirali objekt, no za razliku od TOF metode skeniranja ova metoda je pogodnija za skeniranje manjih objekata. Preciznost ovakve metode je u redovima desetinki mikrometara. Kod ovakve metode laserska zraka obasja objekt skeniranja te pomoću kamere ispituje lokaciju laserske točke na objektu. Ovisno o tome koliko daleko laser obasja površinu, laserska točka se pojavi na različitom mjestu u vidnom polju kamere. Laserska točka na objektu i kamera tvore trokut po čemu je metoda i dobila naziv triangulacijska [Slika 4.]. Pošto je poznata udaljenost između lasera i kamere kao i kut emitiranja lasera, te kut upada laserske zrake u vidno polje kamere moguće je odrediti udaljenost objekta skeniranja. Kako bi se ubrzao proces skeniranja najčešće se kod ove metode koristi laserska pruga koja obasja objekt skeniranja. [3]



Slika 4. Prikaz triangulacijske metode skeniranja

### 2.2.4. Metoda strukturiranog svjetla

Skeneri koji rade na principu strukturiranog svjetla projekiraju uzorak svjetla na objekt skeniranja. Pomoću kamere koja promatra oblik i deformacije projekcionog uzorka svjetla izračunavaju se udaljenosti točaka u vidnom polju [Slika 5.]. Prednost ovakve metode skeniranja je velika brzina i preciznost razlog tome je skeniranje više točaka istovremeno ili cijelog vidnog polja odjedanput. [3]



Slika 5. Prikaz strukturiranog skeniranja

### 3. IZRADA 3D ROTACIJSKOG SKENERA

U nastavku je prikazana princip rada i izvedba 3D skenera koji se sastoji od dva stupnja slobode gibanja, rotacije predmeta i translacije senzora te usporedba sa 3D skenerima sličnog principa rada

#### 3.1. Slični tipovi skenera

##### 3.1.1. Ciclop 3D scanner - BQ



Slika 6. Ciclop

Ciclop [Slika 6.] je jedan od najmanjih 3D scannera na tržištu. Radi na principu triangulacijske metode sa 2 lasera, te ima jedan stupanj slobode - rotacija objekta skeniranja. Maksimalna masa objekta skeniranja je 2,25 kg, a maksimalna veličina objekta skeniranja je 200 x 200 x 205 mm, te mu je vrijeme skeniranja 8 minuta. [4]

##### 3.1.2. V2 3D scanner - Matter & Form

V2 [Slika 7.] isto kao i Ciclop radi na principu triangulacijske metode pomoću dva lasera, ali za razliku od Ciclopa V2 ima 2 stupnja slobode gibanja – rotacija objekta i translacija nosača senzora. Maksimalna masa objekta skeniranja je 3 kg, a maksimalna veličina objekta skeniranja je 180 x 180 x 250 mm. [4]



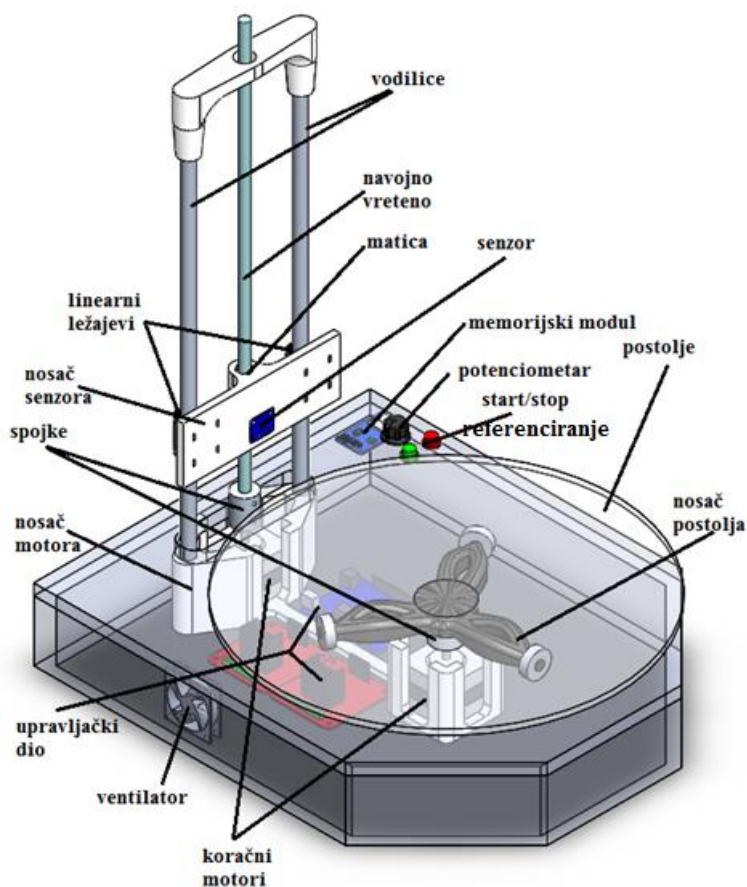
Slika 7. V2

### 3.1.3. Usporedba skenera i cilj novog skenera

	Ciclop	V2	Cilj
Metoda rada	triangulacijska	triangulacijska	TOF (time of flight)
Stupnjevi slobode	1	2	2
Preciznost [mm]	$\pm 0,2$	$\pm 0,1$	$\pm 1$
Maksimalna veličina objekta skeniranja [mm]	$\varnothing 200 \times 205$	$\varnothing 180 \times 250$	$\varnothing 300 \times 270$
Maksimalna masa objekta skeniranja [kg]	2,25	3	4
Cijena [HRK]	2500	4480	< 2000

Tablica 1. Usporedba i cilj skenera

### 3.2. Princip rada



Slika 8. 3D skener i njegove komponente

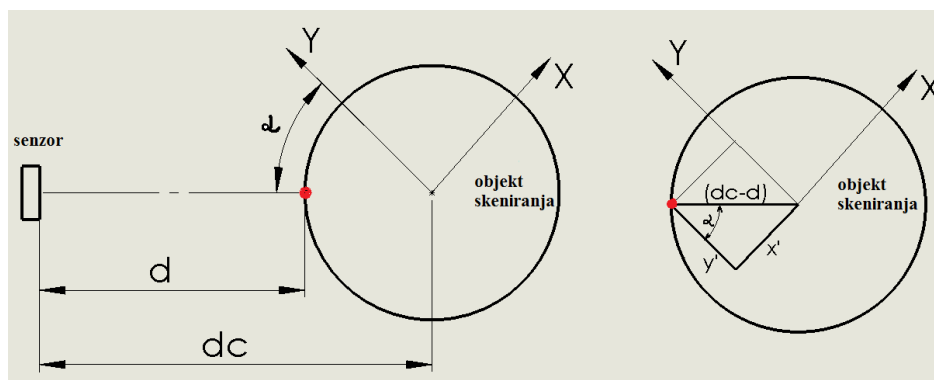
Skeniranje se izvodi tako da se objekt skeniranja okreće oko svoje za neki korak, te se prilikom svakog koraka mjeri njegova udaljenost od senzora i izračunava se koordinata te točke u prostoru. Nakon punog kruga dobiven je oblak točaka jedne ravnine objekta. Zatim se senzor podiže na drugu visinu te se ponavlja postupak okretanja i mjerenja objekta. Cijeli postupak se ponavlja dok se ne dobije oblak točaka cijelog objekta. Oblak točaka odnosno koordinate se spremaju na memorijsku karticu u memorijskom modulu.

Okretanje objekta je izvedeno tako da je postolje, na kojem leži objekt, pomoću spojke povezano na vratilo koračnog motora. Kako vratilo koračnog motora ne bi nosilo cijelu težinu objekta skeniranja postolje leži na nosaču postolja koje rotira po kućištu 3D skenera.

Mjerenje [Slika 9]. se izvodi pomoću senzora VL6180X koji mjeri udaljenost  $d$ . Pomoću te udaljenosti i poznate udaljenosti  $d_c$ , odnosno udaljenosti senzora od centra rotacije postolja te poznatog kuta zakreta koračnog motora  $\alpha$  moguće je izračunati koordinate točke u prostoru pomoću sljedećeg izraza:

$$X = \sin(\alpha) \cdot (d_c - d) \quad (2)$$

$$Y = \cos(\alpha) \cdot (d_c - d) \quad (3)$$



Slika 9. Mjerenje objekta

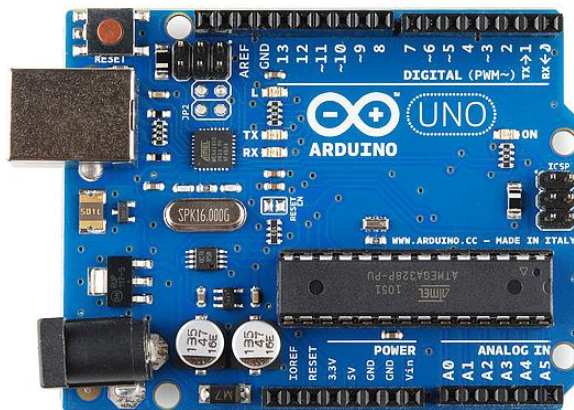
Senzor je postavljen na nosaču senzora koji sadrži maticu i dva linearna ležaja. Prilikom okretanja navojnog vretena, pomoću koračnog motora, podižemo senzor. Pomoću poznatog kuta zakreta motora  $\beta$  i koraka navojnog vretena  $Ph$  izračunava se visina mjerenja senzora po sljedećem izrazu:

$$h = \frac{360}{\beta} \cdot Ph \quad (4)$$

Koračnim motorima se upravlja pomoću upravljačkog dijela odnosno preko mikrokontrolera koji šalje informaciju driverima koji pokreću koračne motore.

### 3.3. Opis korištenih komponenti

#### 3.3.1. Mikrokontroler



Slika 10. Arduino Uno

Arduino Uno je mikrokontroler odnosno prototipna platforma koji ima 14 digitalnih ulazno izlaznih pinova, 6 analognih ulaza, 16MHz kvarcni kristal, USB konektor, konektor za napajanje, ICSP, reset i namijenjena je kreiranju elektroničkih projekata. [5][6]. Zbog svoje jednostavnosti programiranja i upotrebe, te dovoljnog broja ulaza odnosno izlaza izabran je kao upravljačka jedinica ovog 3D skenera.

#### 3.3.2. VL6180X senzor

Adafruit VL6180X je TOF (time of flight) senzor udaljenosti. Sastoji se od izvora laserske svjetlosti, pripadajućeg senzora za očitavanje svjetlosti, regulatora napona i ostalih komponenti koje sudjeluju u procesuiranju podataka.[Slika 11.] Kao što je ranije spomenuto TOF senzori mjere vrijeme potrebno emitiranoj laserskoj zruci da se odbije od predmeta i vrati u senzor, te pomoću tog vremena i brzine svjetlosti izračunavaju udaljenost predmeta od senzora.



Slika 11. Adafruit VL6180X

Senzor se sastoji od 7 pinova :

VIN- služi za napajanje senzora. Pošto čip koristi 2,8 V na tiskanu pločicu senzora je postavljen regulator napona tako da na pin VIN je potrebno dovesti napon od 3-5 V

2v8- izlaz od 2,8 V

GND- pin za uzemljenje

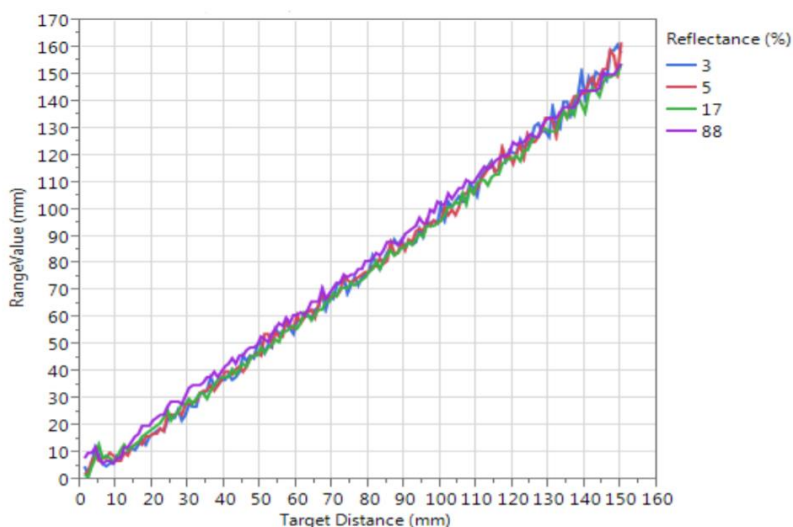
SCL- clock pin koji koristi za IC2 komunikaciju

SDA- data pin koji se koristi za IC2 komunikaciju

GPIO- pin koji se koristi kako bi senzor poslao informaciju da je podatak sa senzora spreman za slanje

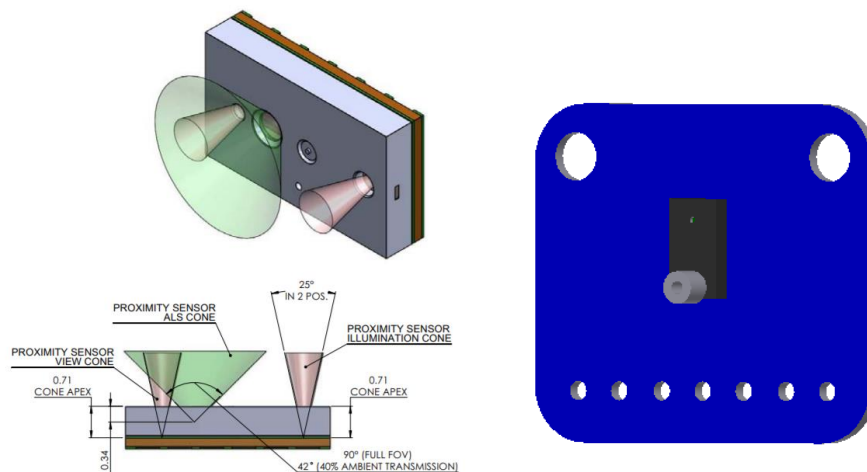
SHDN- pin koji služi za gašenje senzora [7]

Mjerni raspon ovog senzora je 5-160mm, no preciznost na udaljenostima većim od 100mm uvelike ovisi o vanjskim uvjetima kao što temperatura, osvjetljenje i karakteristike objekta skeniranja odnosno njegovoj svojstvu reflektiranja laserskih zraka. Na slici 12. je prikazan graf uspoređivanja rezultata mjerenih vrijednosti na tijelima različite refleksije. [8]



Slika 12. Usporedba rezultata mjerenja udaljenosti tijela različite refleksije

Na slici 13.- lijevo je prikazan konus osvjetljenja laserske zrake koju emitira senzor. Izlazni kut laserske zrake je  $25^\circ$ , a da bi se dobilo što preciznije mjerenje poželjno je da je zraka što uža odnosno usmjerenija na manju površinu. Kako bi se suzio kut izlazne zrake na senzor odnosno izlaz laserske zrake je ugrađen cilindar. [Slika 13 - desno]

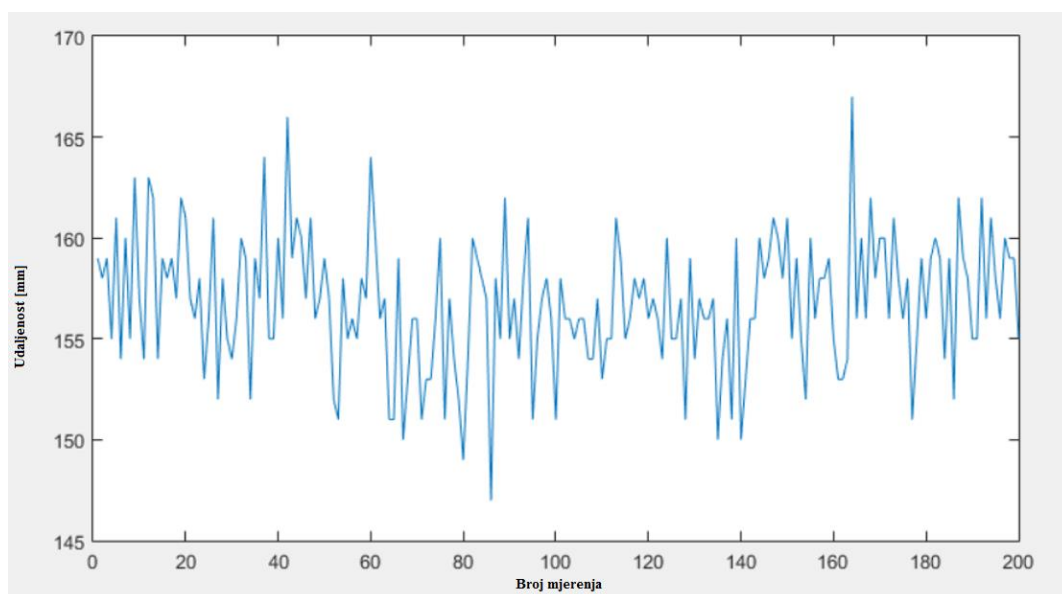


Slika 13. Prikaz vidnog područja senzora i dodanog cilindra

### 3.3.2.1. Ispitivanje točnosti senzora

Prema specifikacijama proizvođača preciznost ovog senzora je  $\pm 1$  mm. U nastavku su provedena ispitivanja točnosti senzora sa različitim brojem ponavljanja mjerenja. Ispitivanje je provedeno kako bi se utvrdilo sa kojim najmanjim brojem uzoraka mjerenja i njihovom srednjom vrijednošću dobivamo prihvatljivu točnost. Objekt mjerenja se nalazi na udaljenosti od 157 mm.

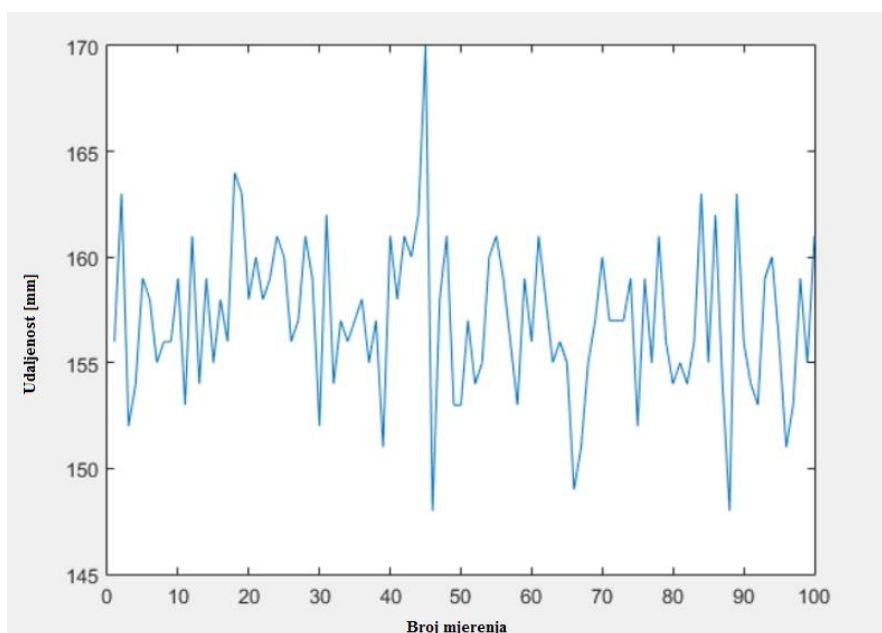
Prvo ispitivanje je provedeno sa 200 mjerenja te je dobivena srednja vrijednost 156,91 mm, što je prihvatljivo, ali zahtjeva previše vremena za potrebe 3D skenera.



Slika 14. Mjerenje 200 uzoraka

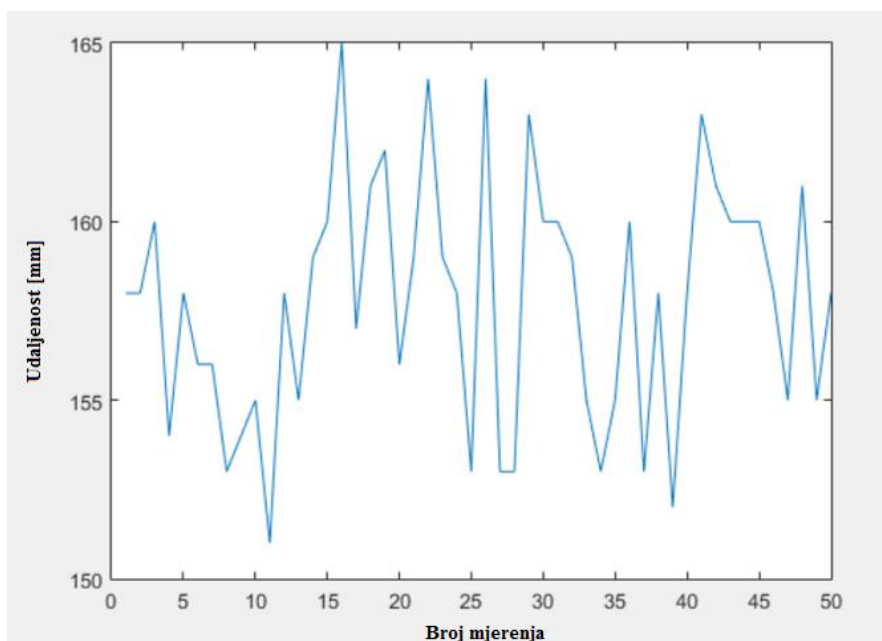


Sljedeće ispitivanje je provedeno sa 100 uzoraka [Slika 15.] prilikom kojeg je dobivena vrijednost od 157,20 mm, što je također prihvatljivo, ali zahtjeva previše vremena za potrebe 3D skenera.



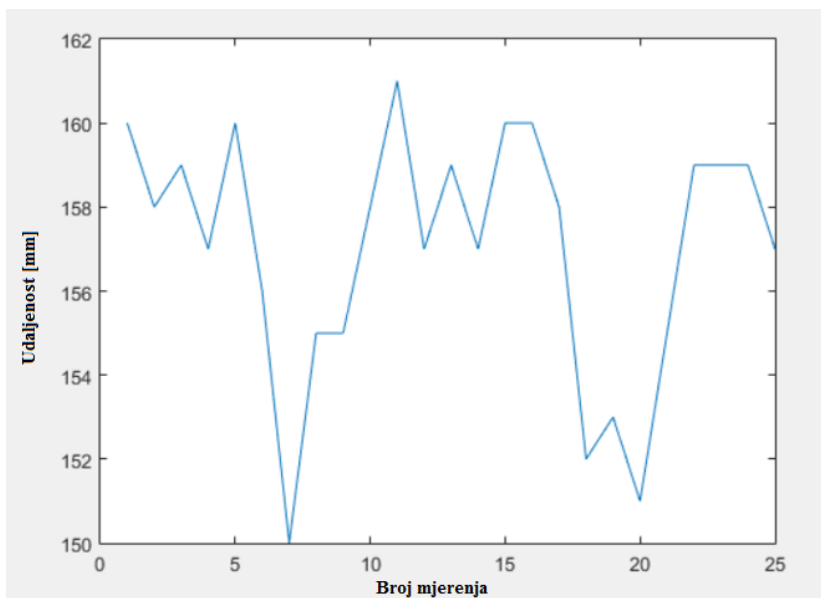
Slika 15. Mjerenje 100 uzoraka

Sljedeće ispitivanje je provedeno sa 50 uzoraka [Slika 16.] prilikom kojeg je dobivena vrijednost od 157,52 mm, što je također prihvatljivo, ali zahtjeva previše vremena za potrebe 3D skenera.



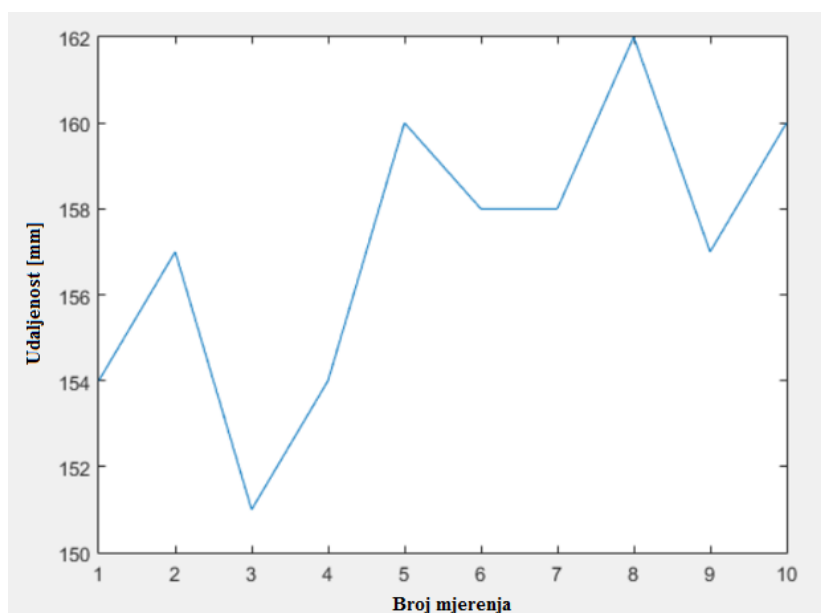
Slika 16. Mjerenje 50 uzoraka

Sljedeće ispitivanje je provedeno sa 25 uzoraka [Slika 17.] prilikom kojeg je dobivena vrijednost od 157,00 mm, što je također prihvatljivo, ali zahtjeva previše vremena za potrebe 3D skenera.



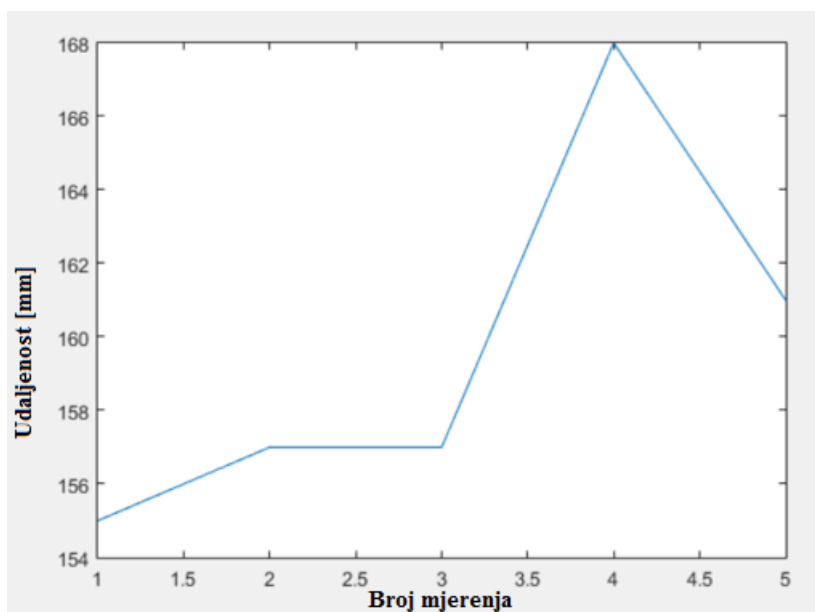
Slika 17. Mjerenje 25 uzoraka

Sljedeće ispitivanje je provedeno sa 10 uzoraka [Slika 18.] prilikom kojeg je dobivena vrijednost od 157,1mm, što je također prihvatljivo.



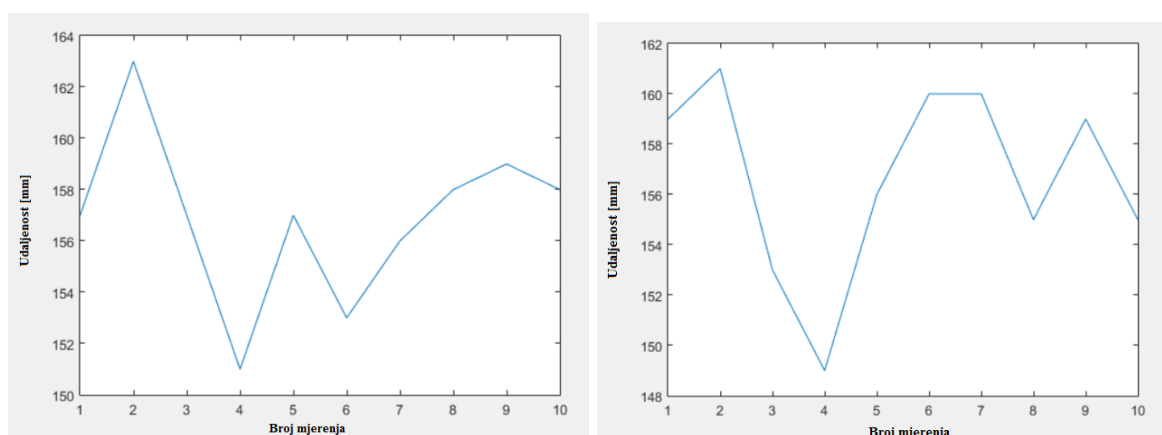
Slika 18. Mjerenje 10 uzoraka

Sljedeće ispitivanje je provedeno sa 5 uzoraka [Slika 19.] prilikom kojeg je dobivena vrijednost od 159,1 mm, što nije prihvatljivo.



Slika 19. Mjerenje 5 uzoraka

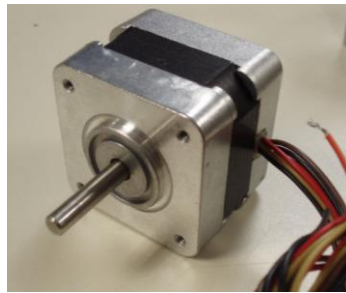
Iz grafova dobivenih rezultata mjerenja vidljivo je da postoje velike oscilacije u rezultatima mjerenja, te da je 10 najmanji broj mjerenja udaljenosti za koji je dobiven prihvatljiv rezultat. U nastavku su provedena još 2 ispitivanja mjerenjem 10 uzoraka.



Slika 20. Dodatna mjerenja 10 uzoraka

Prilikom dodatnih ispitivanja mjerenja 10 uzoraka ponovno su dobiveni prihvatljivi rezultati od 156,7 mm i 156,9 mm.

### 3.3.3. Koračni motor



Slika 21. Koračni motor

Skener koristi unipolarne Minebea hybrid 17PM-M012-13 koračne motore:

- ulaznog napona 6-24V
- struja 0,67 A
- korak  $1,8^\circ$
- točnosti +/- 5%
- maksimalni moment držanja 0,0833 Nm.

Motor sadrži 6 žica od kojih su 4 namotaji motora te 2 žice su na istom potencijalu. [9] Pošto znamo da je korak motora  $1,8^\circ$ , a potpuna rotacija iznosi  $360^\circ$ , može se izračunati da je potrebno 200 koraka kako bi motor napravio cijeli krug. Taj podatak je potrebno poznavati za programiranje 3D skenera.

### 3.3.4. Driver

Driver koji se koristi u ovom 3D skeneru je L298N [Slika 22]. L298N ima mogućnost upravljanja naponima od 4 V do 35 V te strujama do 4 A. [10] Dizajniran je kako bi upravljao dva istosmjerna motora ali zbog svoje mogućnosti upravljanja velikim strujama je pogodan za upravljanje koračnim motorom ovog 3D skenera. Kako bi se osiguralo da se driveri ne pregrijevaju ugrađen je ventilator.



Slika 22. L298N

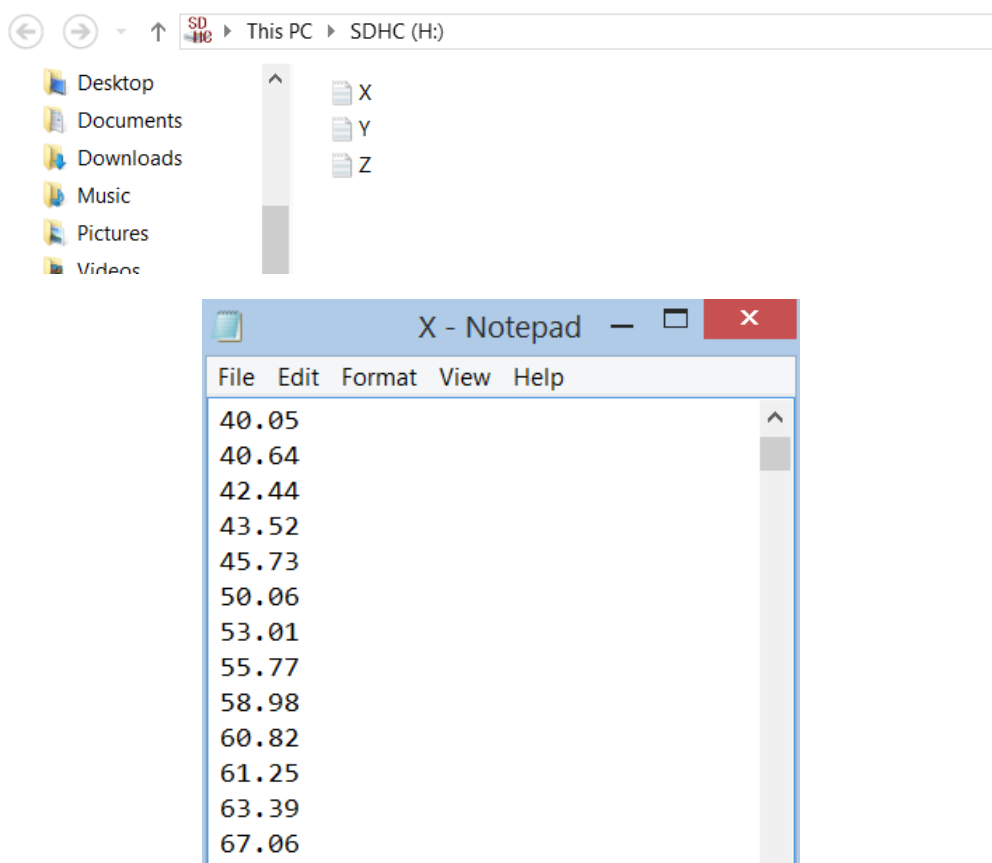
### 3.3.5. Memorijski modul

Kako bi se omogućilo spremanje podataka očitanih 3D skenerom koristi se memorijski modul koji sprema podatke na microSD karticu.[11] Modul komunicira sa mikrokontrolerom putem SPI (Serial Peripheral Interface), odnosno serijskim perifernim sučeljem koje mikrokontroleri koriste za brzu komunikaciju sa jednim ili više uređaja na kratkim udaljenostima.[12] Ovakav modul može koristiti microSD kartice veličine do 32GB te radi na naponu od 4,5 V do 5 V



Slika 23. Memorijski modul

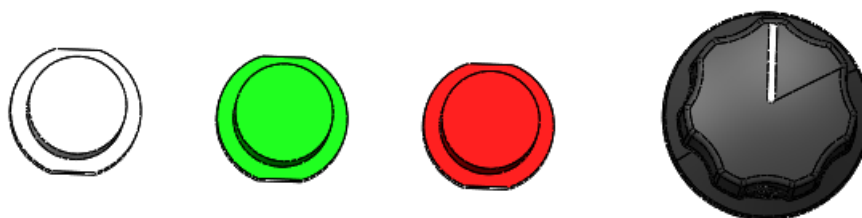
Podatci se spremaju u obliku .txt datoteka. [Slika 24.]



Slika 24. Prikaz podataka

### 3.3.6. Korisničko sučelje

Korisničko sučelje se sastoji od tri tipkala i jednog potenciometra. Pomoću potenciometra od 10 k $\Omega$  se definira visina skeniranja, odnosno visina objekta skeniranja. Pomoću bijelog tipkala se započinje proces referenciranja dok se pomoću zelenog započinje skeniranje, a pomoću crvenog tipkala se resetira sustav.

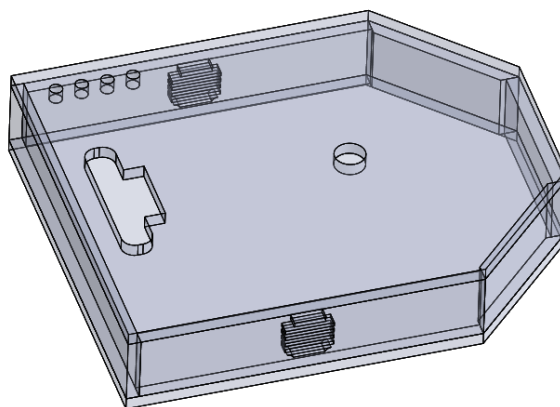


Slika 25. Upravljačko sučelje

### 3.4. Konstrukcijska izvedba

#### 3.4.1. Kućište

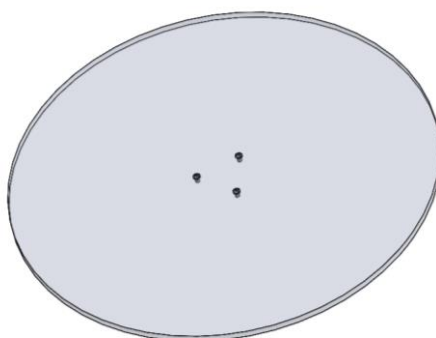
Kućiče [Slika 26.] je sastavljeno od 8 međusobno vijcima vezanih komada PMMA, odnosno Akrila. Komadi kućišta su dobiveni laserskim rezanjem Akrila prema nacrtima. Debljina svakog komada je 10 mm što osigurava potrebnu čvrstoću za ciljanu nosivost 3D rotacijskog skenera.



Slika 26. Kućište

#### 3.4.2. Postolje

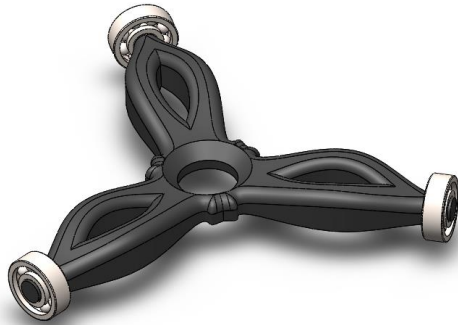
Postolje [Slika 27.] je izrađeno laserskim rezanjem 5 mm debelog PMMA, odnosno Akrila.



Slika 27. Postolje

### 3.4.3. Nosač postolja

Nosač postolja [Slika 28.] je izrađen 3D printanjem PETG materijala prema zadanom modelu. Na njega su taljenjem materijala postavljeni kuglični ležajevi koji prenose težinu objekta skeniranja sa postolja na kućište 3D skenera.



Slika 28. Nosač postolja

### 3.4.4. Prijenosnik

Zadaća prijenosnika [Slika 29.] kao što mu i samo ime kaže je da prenosi pokrete koračnog motora na postolje. Sa koračnim motorom je spojen preko aluminijske spojnice, a sa postoljem pomoću tri vijka. Prijenosnik je izrađen 3D printanjem PETG materijala prema zadanom modelu.

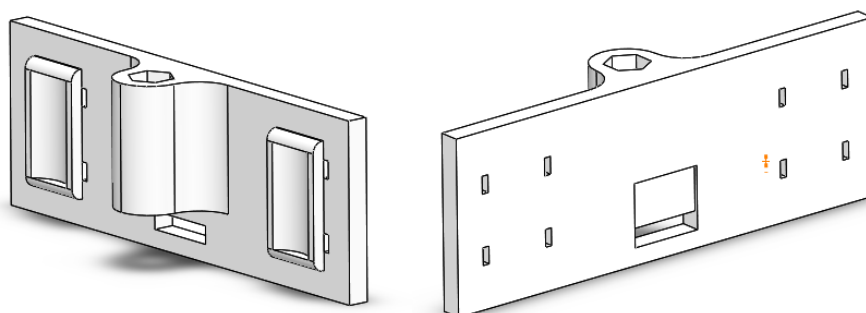


Slika 29. Prijenosnik



### 3.4.5. Nosač senzora

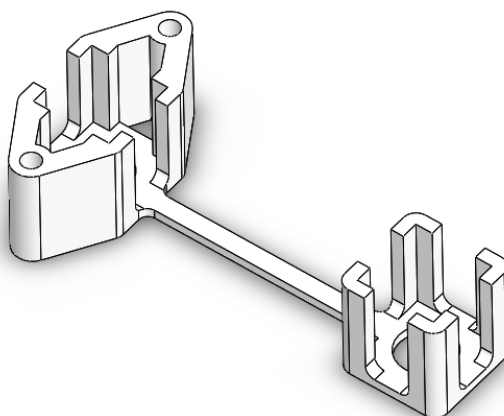
Nosač senzora [Slika 30.] pomoću matice pretvara rotacijsko gibanje motora odnosno vretena u translacijsko gibanje senzora. Nosač također sadrži utore u kojima leže linearni ležajevi koji osiguravaju linearno kretanje nosača senzora i samim time senzora. Nosač je izrađen 3D printanjem PLA materijala prema zadanom modelu.



Slika 30. Nosač senzora

### 3.4.6. Nosač motora

Nosač motora [Slika 31.] je izrađen 3D printanjem PLA materijala prema zadanom modelu i služi kako bi osigurao da se koračni motori i vodilice nalaze na svome mjestu. Povezan je sa kućištem pomoću četiri vijka. Bitno je naglasiti da je ovo prototipno rješenje te bi pravi nosači motora trebali služiti ujedno kao i hladnjaci te bi morali biti dovoljno kruti kako bi onemogućili neželjene pomake motora.



Slika 31. Nosač motora

### 3.5. Program

U nastavku je objašnjen program napisan u Arduino programskom jeziku. Kod je podijeljen u cjeline i objašnjen kroz način rada skenera radi lakšeg shvaćanja. Kompletan program se nalazi u prilogu.

#### 3.5.1. Inicijalizacija

```

///ZAVRŠNI RAD, Karlo Brzica, 2019, Zagreb
///3D rotacijski skener
///Fakultet strojarstva i brodogradnje
///Sveučilište u Zagrebu

#include <SPI.h>
#include <SD.h>
#include <Wire.h>
#include "Adafruit_VL6180X.h"

Adafruit_VL6180X vl = Adafruit_VL6180X();

//definiranje varijabli
File myFile;
int pinCS = 10;           //pin 10 za komunikaciju sa memorijskim modulom
int postoljePin1 = 5;     //pin koračnog motora postolja int 4
int postoljePin2 = 4;     //pin koračnog motora postolja 3
int postoljePin3 = 3;     //pin koračnog motora postolja 2
int postoljePin4 = 2;     //pin koračnog motora postolja int 1
int vretenoPin1 = 9;      // pin koračnog motora vretena int4
int vretenoPin2 = 8;      // pin koračnog motora vretena 3
int vretenoPin3 = 7;      // pin koračnog motora vretena 2
int vretenoPin4 = 6;      // pin koračnog motora vretena 1

int i=0;                  | //pomoćna varijabla
int del=50;               //delay postolja
int delv=25;              //delay postolja

float visina=0;           //pomoćna varijabla visine

int analogPin3=A3;        // definiranje naziva analognih pinova
int analogPin2=A2;
int analogPin1=A1;
int analogPin0=A0;
int potenciometar=0;     //definiranje vrijednosti analognih pinova
int start=0;
int ref=0;
int granicnik=0;

int korak=5;              //željeni pomaka z osi [mm]

float visina_predmeta=0;  //visina objekta skeniranja [mm]

float udaljenost=0;       //udaljenost dobivena mjerenjem senzora [mm]
float stupanj=0;          //pomoćna varijabla zakreta postolja

float x=0;                 //X koordinata
float y=0;                 //Y koordinata
float z=0.0;

//postavke
void setup()
{
  pinMode(postoljePin1,OUTPUT);
  pinMode(postoljePin2,OUTPUT);
  pinMode(postoljePin3,OUTPUT);
  pinMode(postoljePin4,OUTPUT);           //definiranje ulaza / izlaza
  pinMode(vretenoPin1,OUTPUT);
  pinMode(vretenoPin2,OUTPUT);
  pinMode(vretenoPin3,OUTPUT);
  pinMode(vretenoPin4,OUTPUT);
  pinMode(pinCS, OUTPUT);

```

Slika 32. Definiranje varijabli

Prilikom uključivanja upravljačke jedinice odnosno mikroupravljača pozivaju se datoteke koje sadrže dodatne funkcije (SPI.h, SD.h, Wire.h, Adafruit\_VL6180.h) zatim se inicijaliziraju pinovi i varijable koje će se koristiti u nastavku programa. Zatim se definira kakvu funkciju će izvršavati koji pin odnosno hoće li biti ulazni ili izlazni. [Slika 32.]

### 3.5.2. Glavna petlja

Nakon inicijalizacije mikrokontroler čita glavnu petlju programa. [Slika 33.] Glavna petlja se sastoji od početnog izbornika u kojem mikrokontroler čita stanja ulaza tipkala za start i tipkala za referenciranje. Ukoliko korisnik pritisne tipkalo mikrokontroler to registrira te odlazi u funkciju naziva "referenciranje".

```
void loop()
{
  //Serial.print("Pritisnite start ili započnite referenciranje");

  start = analogRead(analogPin2);      // čitanje stanja analognih ulaza (tipkala start i ref)
  ref = analogRead(analogPin1);

  while(start<999)                     // provjera stanja tipkala start
  {

    start = analogRead(analogPin2);
    ref = analogRead(analogPin1);
    if(ref>999)                         //provjera stanja tipkala ref
    {
      //Serial.print("Proces referenciranja započinje");
      delay(del);
      referenciranje();                 //pozivanje funkcije referenciranje
    }
  }
}
```

Slika 33. Glavna petlja 1.dio

#### 3.5.2.1. Referenciranje

Pomoću funkcije referenciranje [Slika 34.] nosač senzora se spušta do krajnje donje pozicije koju mu određuje graničnik. Proces se odvija tako da mikrokontroler prvo provjerava stanje pina na kojem se nalazi granični prekidač. Ukoliko je očitano da granični prekidač nije okinut pokreće se beskonačna petlja u kojoj se koračni motor pokreće korak po korak. Između svakog koraka mikrokontroler provjerava stanje pina graničnog prekidača. Onog trenutka kada mikrokontroler očita da je granični prekidač okinut prebacuje se nazad u glavnu petlju. Idealnije rješenje bi bilo uvođenje funkcije interrupt u program no zbog popunjenosti pinova to nije moguće izvesti sa ovim mikrokontrolerom. Sljedeći korak nakon referenciranja je pritiskivanje tipkala start.

```

void referenciranje()                                //funkcija referenciranja
{
  //Serial.print("Referenciranje");
  granicnik = analogRead(analogPin3);               //čitanje stanja graničnika
  if(granicnik>999)                                  //provjera stanja graničnika
  {
    loop();                                          //povratak u funkciju loop (ako je graničnik uključen)
  }
  while(1)                                           //beskonačna petlja
  {
    digitalWrite(vretenoPin1, LOW);
    digitalWrite(vretenoPin2, LOW);                 //pokretanje koračnog motora
    digitalWrite(vretenoPin3, LOW);
    digitalWrite(vretenoPin4, HIGH);
    granicnik = analogRead(analogPin3);
    Serial.print(granicnik);
    Serial.print("\n");
    if(granicnik>999)                                //provjera stanja graničnika
    {
      loop();
    }
    delay(delv);                                     //čekanje
    digitalWrite(vretenoPin1, LOW);
    digitalWrite(vretenoPin2, HIGH);
    digitalWrite(vretenoPin3, LOW);
    digitalWrite(vretenoPin4, LOW);
    granicnik = analogRead(analogPin3);
    if(granicnik>999)
    {
      loop();
    }
    delay(delv);
    digitalWrite(vretenoPin1, LOW);
    digitalWrite(vretenoPin2, LOW);
    digitalWrite(vretenoPin3, HIGH);
    digitalWrite(vretenoPin4, LOW);
    granicnik = analogRead(analogPin3);
    if(granicnik>999)
    {
      loop();
    }
    delay(delv);
    digitalWrite(vretenoPin1, HIGH);
    digitalWrite(vretenoPin2, LOW);
    digitalWrite(vretenoPin3, LOW);
    digitalWrite(vretenoPin4, LOW);
    granicnik = analogRead(analogPin3);
    if(granicnik>999)
    {
      loop();
    }
    delay(delv);
  }
}

```

Slika 34. Referenciranje

### 3.5.2.2. Nakon pritiska tipkala start

Nakon pritiska tipkala start mikrokontroler izlazi iz petlje za provjeru stanja tipkala te čita stanje analognog ulaza na kojem se nalazi potenciometar. Stanje na analognom ulazu može biti od 0 do 1023. Preračunavanjem stanja analognog ulaza se dobiva zadana visina objekta skeniranja. Zatim mikrokontroler ulazi u drugu uvjetnu petlju u kojoj se vrti dok senzor ne postigne visinu skeniranog objekta. [Slika 35.]

```
potenciometar = analogRead(analogPin0); //čitanja stanja analognog ulaza potenciometra
Serial.print(potenciometar);
visina_predmeta=potenciometar/4.092; // 1023/250=4,092 --> radno područje do 250mm

while(z<visina_predmeta) //ponavljanje radnje sve dok visina senzora nije veća od visine objekta
{
    okretanjeistolja(z); //pozivanje funkcije okretanja istolja
    okretanjevretena(); //pozivanje funkcije okretanja vretena
    z=z+korak; //povećavanje varijable visine senzora za korak
}
```

Slika 35. Glavna petlja 2.dio

### 3.5.3. Okretanje istolja

Funkcija "okretanjeistolja" [Slika 36.] je zadužena za okretanje istolja, a to radi na način da šalje informaciju pinovima koji su spojeni na drivere koračnog motora istolja. Prilikom postavljanja pina u stanje 'HIGH' on je aktivan odnosno daje napon na svoj izlaz. Izmjenjujući stanje izlaza pinova mikrokontroler upravlja driverom odnosno koračnim motorom. Nakon napravljenog koraka koračnog motora odnosno nakon zakreta objekta skeniranja, mikrokontroler računa za koji se stupanj zakrenuo objekt skeniranja te sa tom informacijom odlazi u funkciju 'čitanje'. Nakon funkcije čitanje poslije se zakreće za jedan korak te opet odlazi u funkciju čitanje sa novim informacijama. Postupak se ponavlja sve dok nije ispunjen uvjet petlje odnosno sve dok objekt skeniranja ne napravi puni krug. Nakon punog kruga mikrokontroler odlazi u funkciju "okretanjevretena".

```

void okretanjePostolja(float z) // funkcija okretanja postolja
{
    while(stupanj < 359) // ponavljajuća petlja sve dok nije ispunjen uvjet
    {
        digitalWrite(postoljaPin1, HIGH);
        digitalWrite(postoljaPin2, LOW);
        digitalWrite(postoljaPin3, LOW); // pokretanje koračnog motora
        digitalWrite(postoljaPin4, LOW);
        delay(del); // čekanje
        stupanj=stupanj+1.8; // povećavanje pomoćne varijable stupnja za kut zakreta motora
        citanje(stupanj,z); // pozivanje funkcije citanje sa varijablama stupanj i visine senzora
        digitalWrite(postoljaPin1, LOW);
        digitalWrite(postoljaPin2, LOW);
        digitalWrite(postoljaPin3, HIGH);
        digitalWrite(postoljaPin4, LOW);
        delay(del);
        stupanj=stupanj+1.8;
        citanje(stupanj,z);
        digitalWrite(postoljaPin1, LOW);
        digitalWrite(postoljaPin2, HIGH);
        digitalWrite(postoljaPin3, LOW);
        digitalWrite(postoljaPin4, LOW);
        delay(del);
        stupanj=stupanj+1.8;
        citanje(stupanj,z);
        digitalWrite(postoljaPin1, LOW);
        digitalWrite(postoljaPin2, LOW);
        digitalWrite(postoljaPin3, LOW);
        digitalWrite(postoljaPin4, HIGH);
        delay(del);
        stupanj=stupanj+1.8;
        citanje(stupanj,z);
    }
    stupanj=0; // resetiranje varijable stupanj
}

```

Slika 36. Okretanje postolja

### 3.5.4. Okretanje vretena

U funkciji okretanje vretena [Slika 37.] mikrokontroler ulazi u uvjetnu petlju u kojoj okreće vreteno pomoću koračnog motora kojim upravlja na istom principu kao i kod okretanja postolja. Vreteno se okreće toliko puta dok ne zadovolji uvjet koraka senzora. Nakon funkcije okretanja vretena mikrokontroler odlazi u uvjetnu petlju u kojoj povećava varijablu visine senzora.

```
void okretanjevretena() // funkcija okretanje vretena
{
  while(visina < korak) // ponavljajuća petlja dok nije ispunjen uvijet
  {
    while(stupanj < 360) // ponavljajuća petlja dok nije ispunjen uvijet
    {
      digitalWrite(vretenoPin1, HIGH);
      digitalWrite(vretenoPin2, LOW);
      digitalWrite(vretenoPin3, LOW);
      digitalWrite(vretenoPin4, LOW);
      delay(delv);
      digitalWrite(vretenoPin1, LOW);
      digitalWrite(vretenoPin2, LOW);
      digitalWrite(vretenoPin3, HIGH);
      digitalWrite(vretenoPin4, LOW);
      delay(delv);
      digitalWrite(vretenoPin1, LOW);
      digitalWrite(vretenoPin2, HIGH);
      digitalWrite(vretenoPin3, LOW);
      digitalWrite(vretenoPin4, LOW);
      delay(delv);
      digitalWrite(vretenoPin1, LOW);
      digitalWrite(vretenoPin2, LOW);
      digitalWrite(vretenoPin3, LOW);
      digitalWrite(vretenoPin4, HIGH);
      delay(delv);
      stupanj=stupanj+7.2; // povećavanje pomoćne varijable stupnja za kut zakreta motora
    }
    stupanj=0; // resetiranje pomoćne varijable
    visina=visina+2; // povećavanje pomoćne varijable visina za korak navojnog vretena
  }
  visina=0; // resetiranje pomoćne varijable
}
```

Slika 37. Okretanje vretena

### 3.5.5. Funkcija čitanje

Funkcija čitanje [Slika 38.] se sastoji od ponavljajuće petlje za mjerenje udaljenosti objekta skeniranja te sumiranja i dijeljenja te udaljenosti kako bi se smanjila greška mjerenja. Zatim se ta udaljenost prema prije objašnjenom načinu pretvara u koordinate točke u prostoru. Te koordinate se spremaju u posebne datoteke na mikroSD karticu.

```

int citanje(float stupanj,float z) // funkcija citanje sa ulaznim varijablama stupanj i z
{
    while(i < 25) // ponavljajuća petlja dok nije ispunjen uvijet mjerenja 25 uzoraka
    {
        uint8_t range = vl.readRange(); // mjerenje udaljenosti
        udaljenost=udaljenost+range; // sumiranje vrijednosti
        i++;
    }
    i=0; // resetiranje vrijednosti
    udaljenost=udaljenost/25;
    uint8_t status = vl.readRangeStatus(); // provjera statusa senzora
    x=cos(stupanj/57.29578)*(160-udaljenost); // računanje X koordinate
    y=sin(stupanj/57.29578)*(160-udaljenost); // računanje Y koordinate
    udaljenost=0; // resetiranje vrijednosti

    if (SD.begin()) // provjera modula
    {
        Serial.println("SD card is ready to use.");
    }
    else
    {
        Serial.println("SD card initialization failed");
        return;
    }

    myFile = SD.open("x.txt", FILE_WRITE); // otvaranje/kreiranje datoteke x.txt:
    if (myFile) // provjera uspješnosti otvaranja
    {
        myFile.print(x);myFile.println(""); // upisivanje x koordinate u datoteku
        myFile.close(); // zatvaranje datoteke
    }

    myFile = SD.open("y.txt", FILE_WRITE);
    if (myFile)
    {
        myFile.print(y);myFile.println("");
        myFile.close();
    }

    myFile = SD.open("z.txt", FILE_WRITE);
    if (myFile)
    {
        myFile.print(z);myFile.println("");
        myFile.close();
    }
}

```

Slika 38. Funkcija citanje



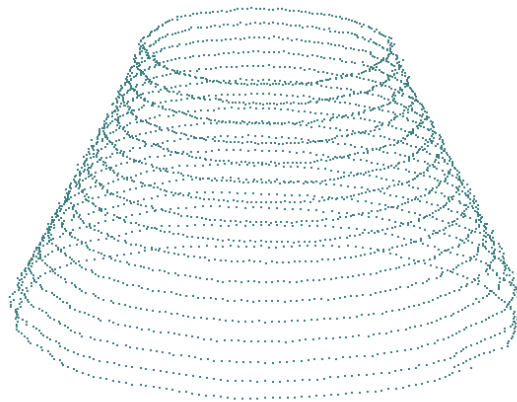
**4. PROCJENA VRIJEDNOSTI**

Naziv	Broj komada	Jedinična cijena [HRK]	Ukupna cijena [HRK]
17-PM-M012-13 koračni motor	2	170,00	340,00
Modul s čitačem microSD kartice	1	35,00	35,00
MicroSD kartica 16GB	1	47,50	47,50
Arduino UNO	1	250,00	250,00
L298N driver	2	75,00	150,00
LM10UU linearni ležaj	2	12,50	25,00
Aluminijska spojnica 5 x 8 mm	2	27,50	55,00
Adafruit VL6180X	1	155,00	155,00
Ventilator 12 V 0,1 A 40 x 40 mm	1	38,50	38,50
Nosač senzora	1	19,00	19,00
Nosač motora	1	25,00	25,00
Odstojnik	1	11,00	11,00
Nosač postolja	1	20,00	20,00
Postolje	1	65,00	65,00
Kućište	1	380,00	380,00
Potenciometar 10 k $\Omega$	1	15,00	15,00
Tipkalo	3	5,00	5,00
Navojno vreteno M8	1	28,00	28,00
Matica M8 x 30	1	3,20	3,20
Tiskana pločica	1	30,00	30,00
Eloksirana aluminijska šipka 10	2	25,50	51,00
<b>UKUPNO:</b>			<b>1748,20</b>

Tablica 2. Troškovi izrade 3D skenera

## 5. PROVEDENA SKENIRANJA I REZULTATI

### 5.1. Stožac – manji, jednostavan model

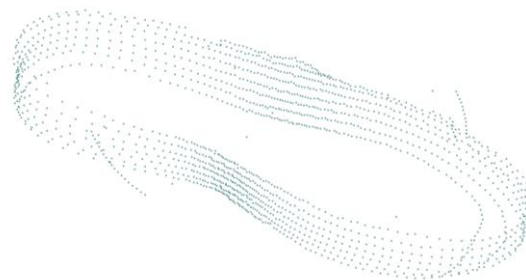


Slika 39. Stožac – manji, jednostavniji model

Vrijeme skeniranja: 24 minuta

Broj točaka: 3400

### 5.2. Tenisica- veći, složeniji model

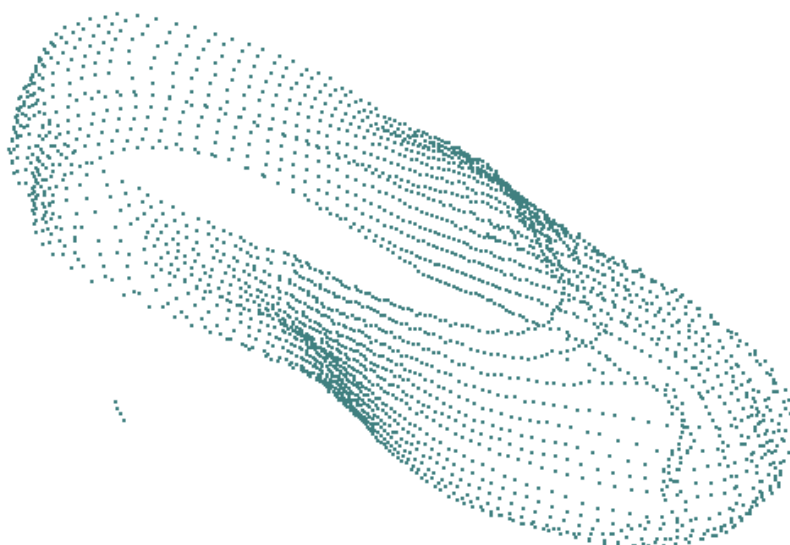


Slika 40. Tenisica, veći složeniji model

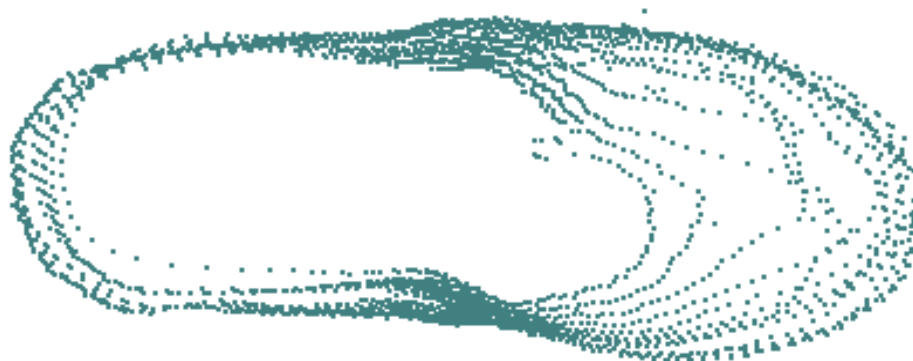
Vrijeme skeniranja: 8 minuta

Broj točaka: 1528

Prilikom skeniranja ovog objekta primijećeno je da senzor očitava loše rezultate prilikom skeniranja u području crne boje. Iz tog razloga je objekt obučen u bijelu boju te se skeniranje ponovilo.



Slika 41. Tenisica drugo skeniranje 1



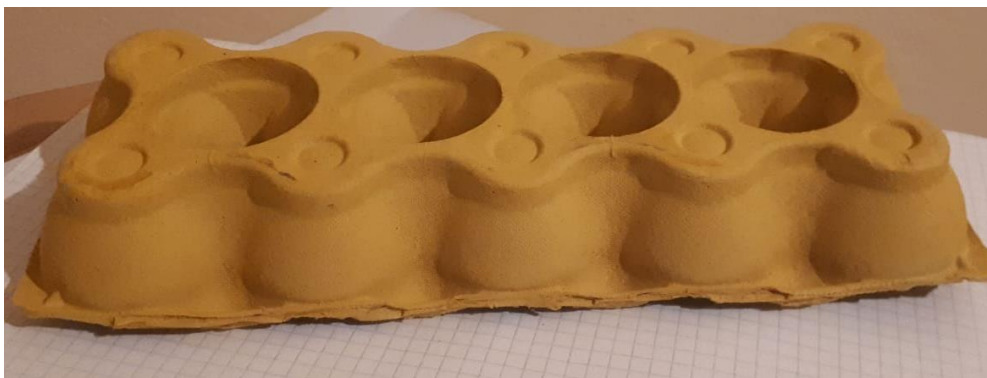
Slika 42. Tenisica drugo skeniranje 2

Vrijeme skeniranja: 16 minuta

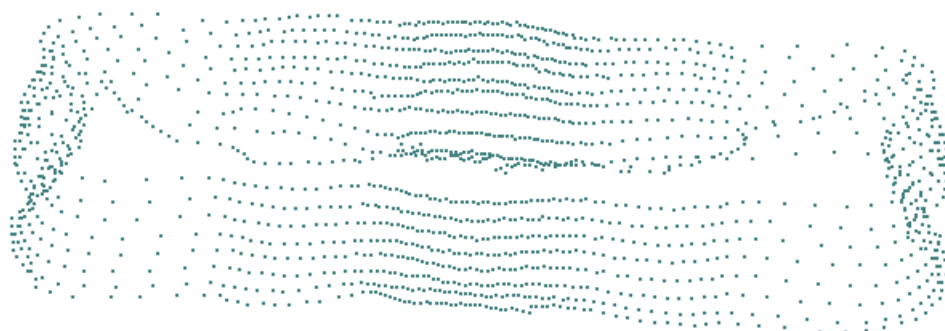
Broj točaka: 2836

Drugo skeniranje je dalo nešto bolje rezultate. No vidljivo je da se objekt skeniranja zamaknuo u odnosu na početni položaj.

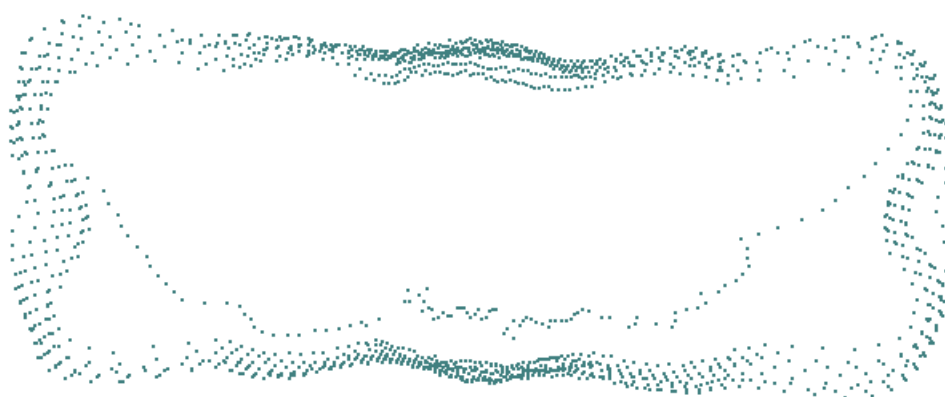
### 5.3. Karton za jaja – veliki složen model



Slika 43. Karton za jaja – veliki složen model



Slika 44. Karton za jaja dobiveno skeniranje

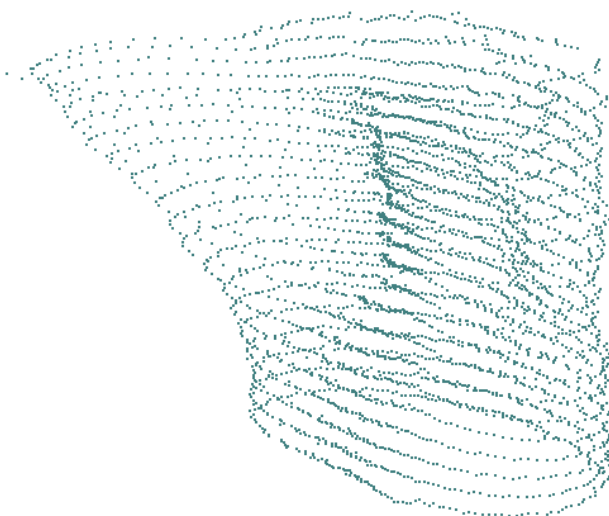


Slika 45. Karton za jaja dobiveno skeniranje drugi pogled

Vrijeme skeniranja: 10 minuta

Broj točaka: 1801

#### 5.4. Skulptura dupina – manji, složen model



Vrijeme skeniranja: 28 minute

Broj točaka: 4000

Iz dobivenih rezultata skeniranja nije moguće razaznati stvarni oblik objekta skeniranja.

## 6. EVALUACIJA REZULTATA SKENIRANJA

Nakon razmatranja rezultata skeniranja može se procijeniti odnos vremena i brzine skeniranja. Sa trenutnom konfiguracijom skenera omogućen je korak visine senzora od 5 mm. Uzimajući to u obzir procijenjeno vrijeme za skeniranje predmeta visine 1cm je 150 sekundi, odnosno procijenjena brzina skeniranja je 2,5 minute po centimetru visine predmeta. Ta brzina uvelike ovisi o dimenzijama objekta skeniranja, što je objekt širi to će zbog TOF senzora skeniranje biti brže i obrnuto.

Kroz provedena skeniranja se može zaključiti da je problem ovog skenera manjak senzora. Odnosno problem skeniranja udaljenosti samo sa jednog mjesta nas ograničava na očitavanje udaljenosti na samo jednoj liniji što onemogućava prikaz složenih geometrija kao je vidljivo na skeniranju dupina.

Još jedan problem ovakvog senzora odnosno načina skeniranja je to što će nam točnost ovisiti o boji objekta skeniranja kao što je vidljivo na skeniranju tenisice. Prilikom promjene njene boje u bijelu skeniranje je solidno obavljeno.

Nadalje kako bi skeniranje bilo kvalitetno potrebno je osigurati dobro prianjanje objekta skeniranja na postolje kako ne bi došlo do zakretanja objekta od početnog položaja.

### 6.1. Karakteristike skenera

		3D skener
Točnost	[mm]	±4
Nosivost	[kg]	3
Brzina	[min/cm]	3
Broj stupnjeva slobode		2
Maksimalne dimenzije objekta skeniranja [mm]		Ø 300 x 270
Cijena	[HRK]	1748,20

Tablica 3. Karakteristike 3D rotacijskog skenera

---

## 7. ZAKLJUČAK

U ovom radu prikazana je izrada, način rada, testiranje i evaluacija 3D rotacijskog skenera. Kroz provedena skeniranja je uočeno da je mana ovakvog načina skeniranja to što je mjerenje udaljenosti ograničeno samo na jednu liniju i samim time je nemoguće vjerno prikazati objekte složene geometrije. Zatim uočeno je da brzina skeniranja uvelike ovisi o veličini objekta skeniranja te preciznost ovisi o boji skeniranog objekta.

Jedno od mogućih rješenja kako poboljšati ovaj 3D rotacijski skener je postaviti još jedan dodatan senzor udaljenosti pod drugim kutom kako bi primali više informacija o objektu skeniranja te njegovoj površini. Nadalje moguće je poboljšati nosivost skenera korištenjem jačih koračnih motora i / ili zupčanika za prijenos snage. Također korištenje potencijometra za odabir visine skeniranja objekta nije idealno rješenje, prihvatljivije i elegantnije rješenje bi bilo korištenje tipkovnice sa LCD zaslonom za prikaz procesa skeniranja i povratne informacije.

Zaključeno je da ova izvedba 3D rotacijskog skenera nije prikladna za korištenje u profesionalne svrhe ali je svakako dobra osnova za učenje i uvod u svijet 3D skeniranja.

---

**LITERATURA**

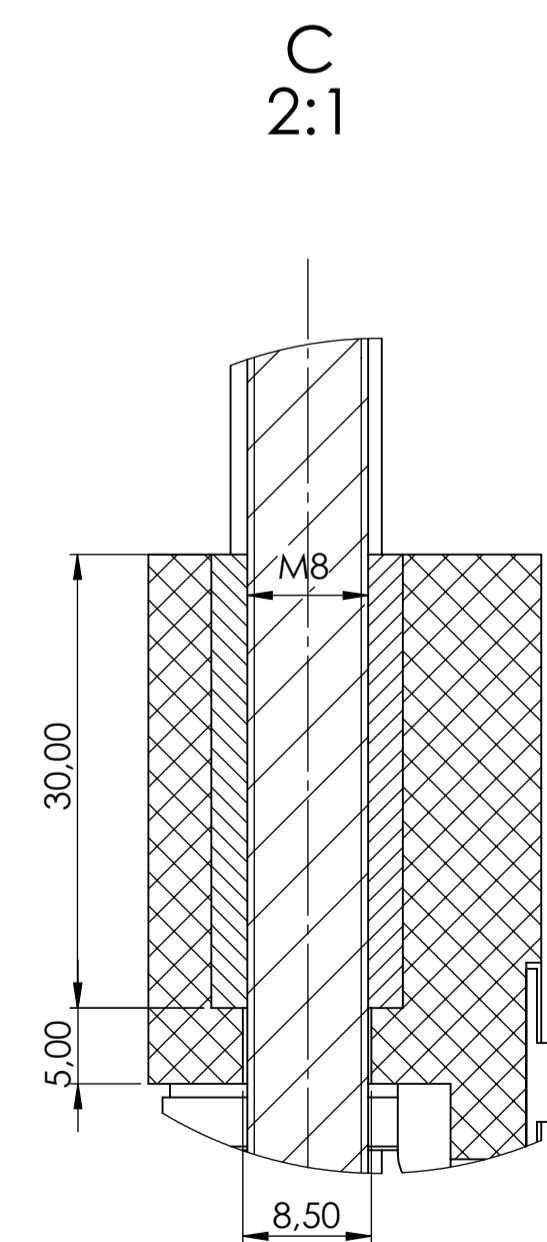
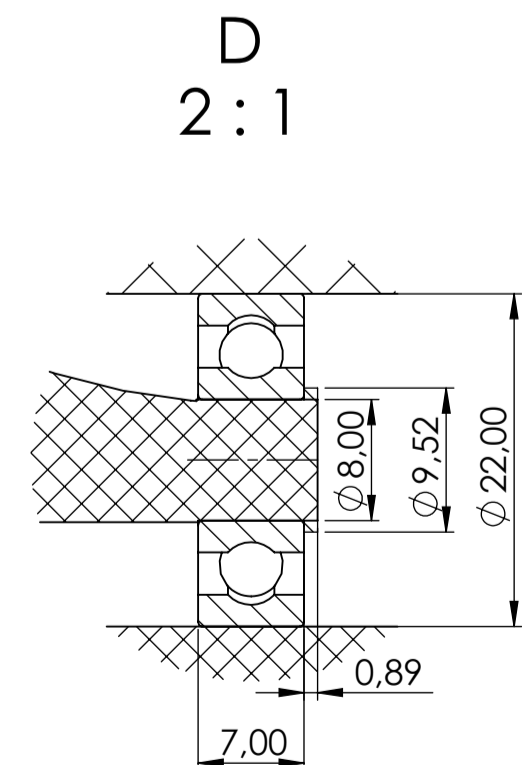
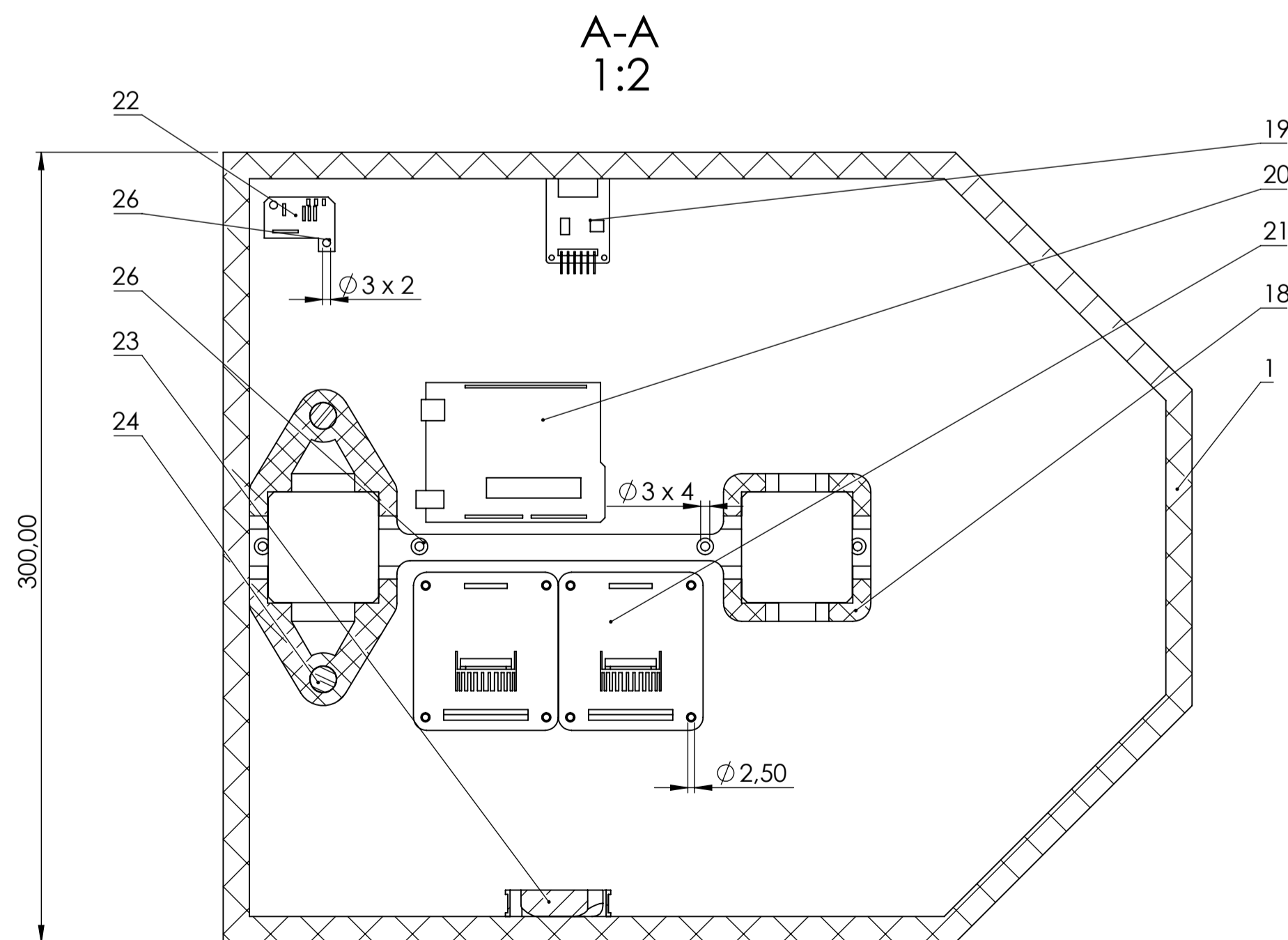
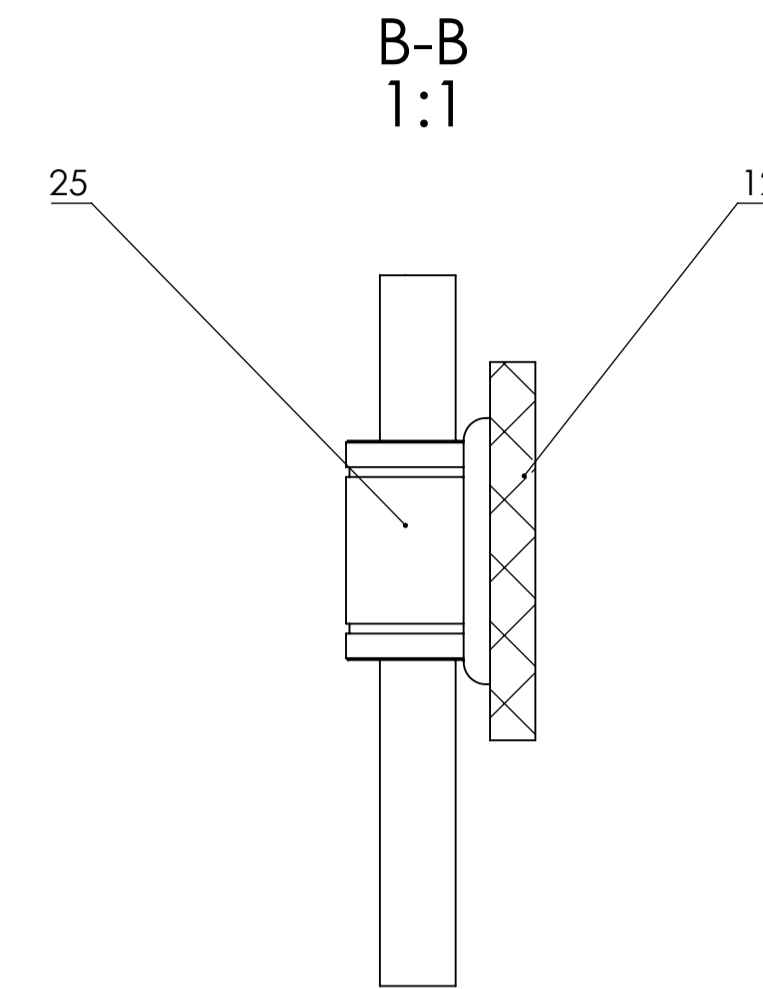
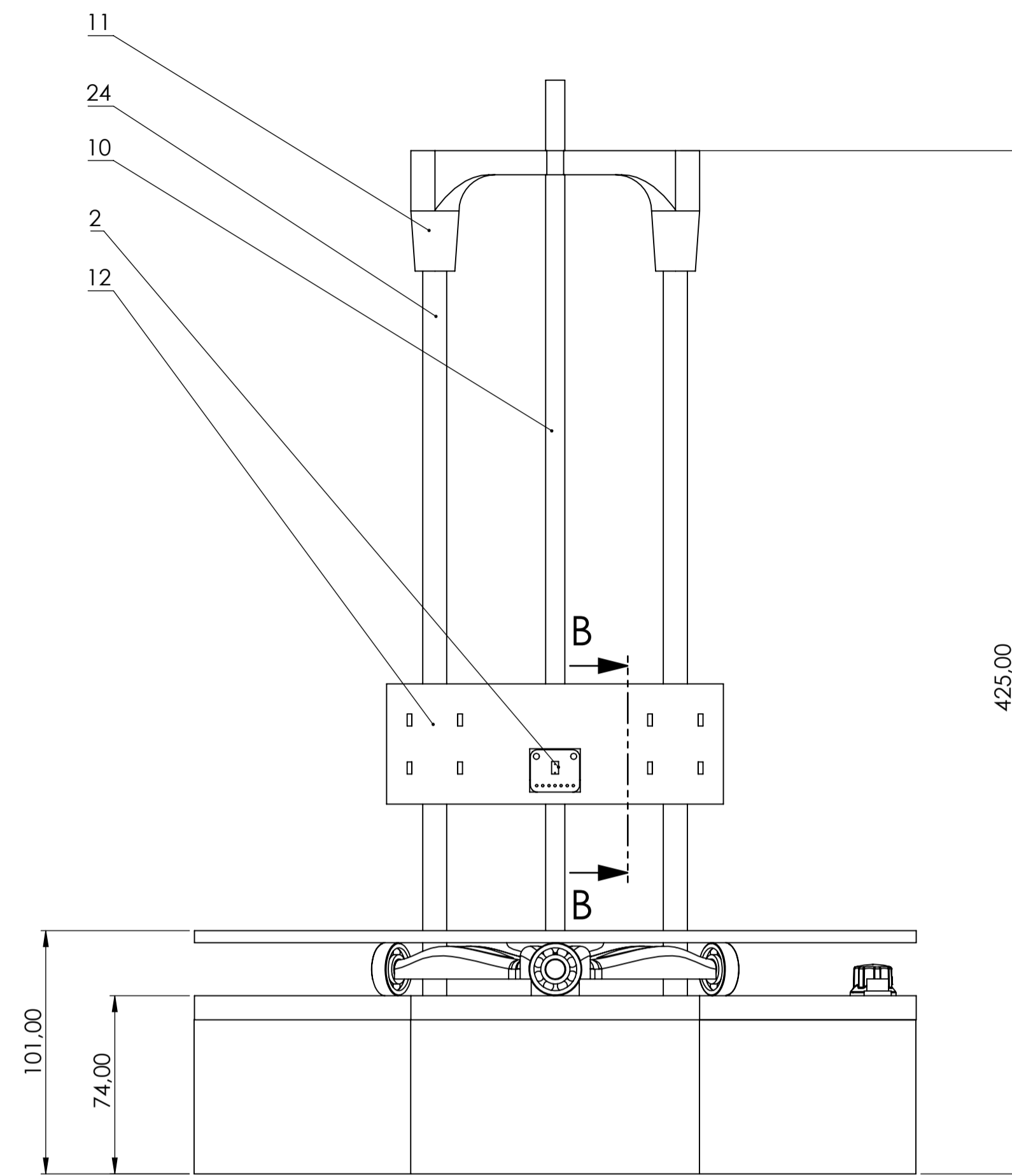
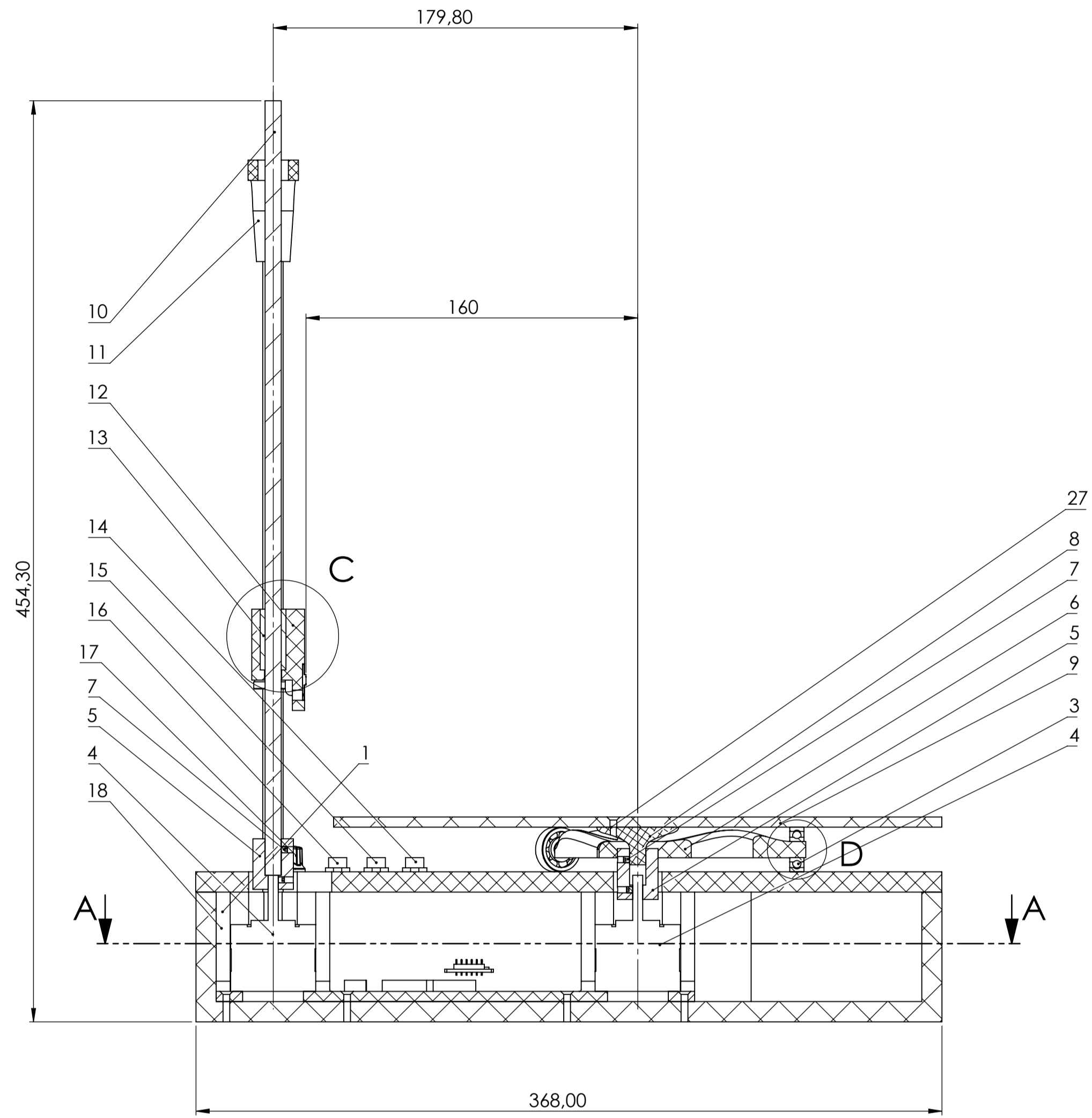
- [1] [https://www.bcs.org/upload/pdf/ewic\\_ev14\\_s19paper3.pdf](https://www.bcs.org/upload/pdf/ewic_ev14_s19paper3.pdf)
- [2] <http://weareprintlab.com/blog/the-real-value-of-3d-scanning-and-its-applications>
- [3] [https://en.wikipedia.org/wiki/3D\\_scanning#Contact](https://en.wikipedia.org/wiki/3D_scanning#Contact)
- [4] <https://www.3dnatives.com/en/top-10-low-cost-3d-scanners280320174/>
- [5] <https://www.arduino.cc/>
- [6] <https://e-radionica.com/hr/blog/2015/10/08/sto-je-arduino-i-croduino/>
- [7] <https://www.adafruit.com/product/3316>
- [8] <https://www.st.com/resource/en/datasheet/vl6180x.pdf>
- [9] <https://bit.ly/2Ndu6hF>
- [10] <https://www.st.com/resource/en/datasheet/l298.pdf>
- [11] [https://www.diykits.eu/shophr/products/p\\_11246](https://www.diykits.eu/shophr/products/p_11246)
- [12] <https://www.arduino.cc/en/reference/SPI>



---

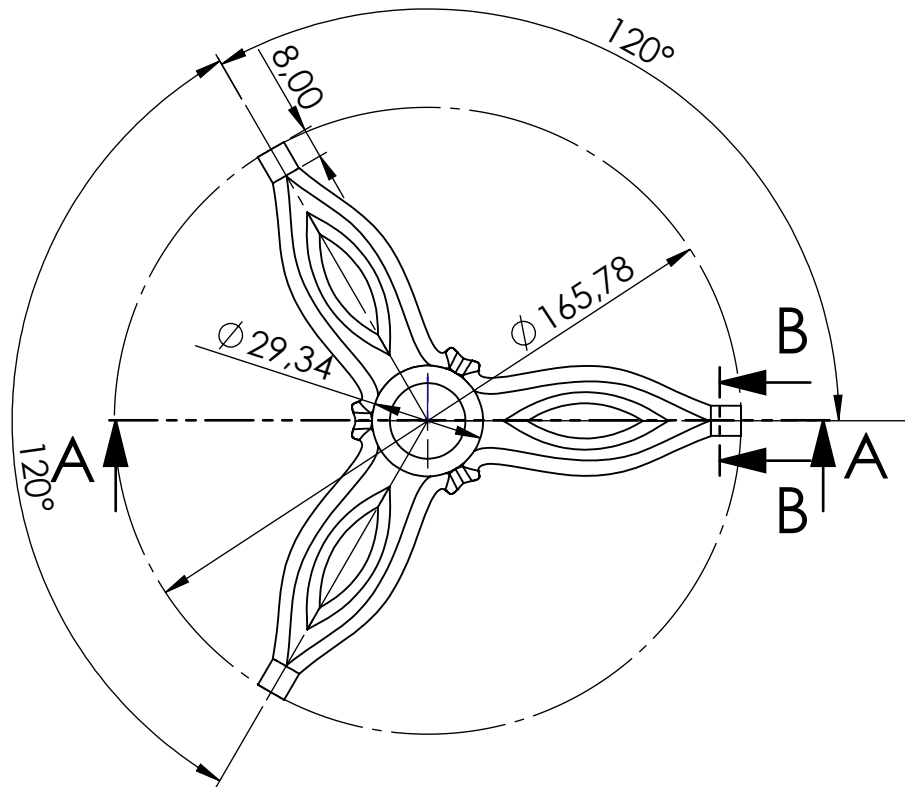
## **PRILOZI**

- I. CD-R disc
- II. Tehnička dokumentacija
- III. Arduino kod

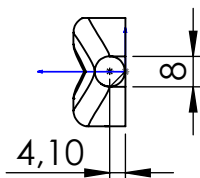


27	M 3 x 10	3	ISO: 10642		
26	M 3 x 15	6	ISO: 10642		
25	LM10UU	2			
24	Vodilica 10 mm	2	RS-00-09	Aluminij	
23	Ventilator 12 V	1			
22	Tiskana pločica	2	RS-00-08		
21	L298N	2			
20	Arduino UNO	1			
19	Modul microSD	1			
18	Nosač motora	1	RS-00-06	PLA	
17	Potenciometar	1			
16	Tipka reset	1			
15	Tipka start	1			
14	Tipka referenciranje	1			
13	Matica M8 x 30	1	DIN: 6334		
12	Nosač senzora	1	RS-00-05	PLA	
11	Razmaknica	1	RS-00-04	PLA	
10	Navojno vreteno	1			
9	Postolje	1	RS-00-03	PMMA	
8	Prijenosnik	1	RS-00-02	PETG	
7	M3 x 3	8	ISO: 4029		
6	Nosač postolja	1	RS-00-01	PETG	
5	Aluminijska spojnica	2		Aluminij	
4	17PM-M012	2			
3	Ležaj 608	3	FAG		
2	VL6180X	1			
1	Sklop kućišta	1	RS-01-00		

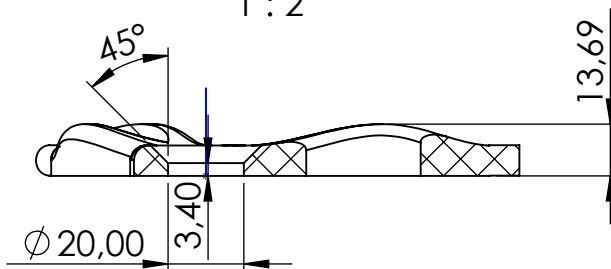
Poz.	Naziv dijela	Kom.	Crtež broj Norma	Materijal	Sirove dimenzije Proizvođač	Masa
Broj naziva - code		Datum		Ime i prezime		Potpis
Projektirao		Karlo Brzica		Kopija		FSB Zagreb
Razradio		Karlo Brzica				
Crtao		Karlo Brzica				
Pregledao		Karlo Brzica				
ISO - tolerancije		Objekt:		Objekt broj:		Smjer: Mehatronika i robotika
Napomena:						
Materijal:		Masa:				
Mjerilo originala		Naziv:		Pozicija:		Format: A1
1:2		3D ROTACIJSKI SKENER		Listova: 1		Lis: 1
		Crtež broj:		RS-00-00		




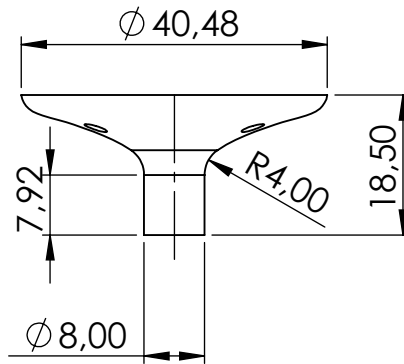
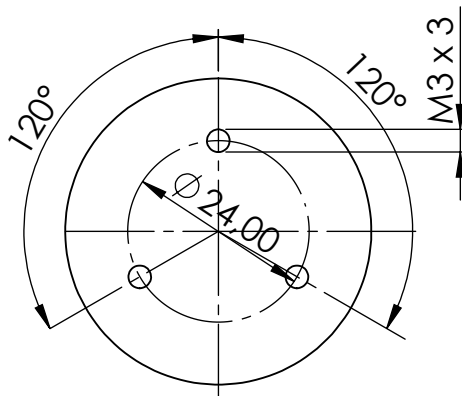
B-B  
1 : 2




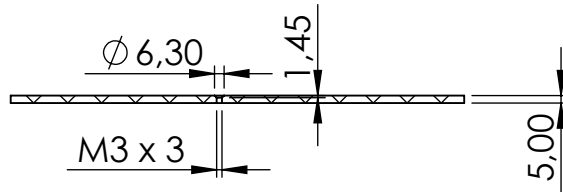
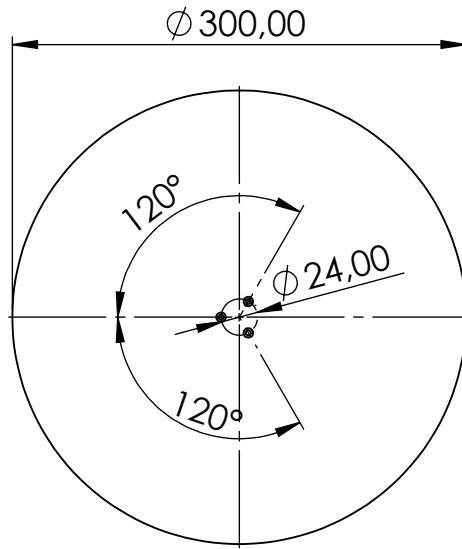
A-A  
1 : 2


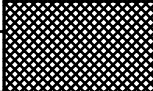
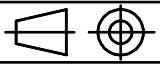


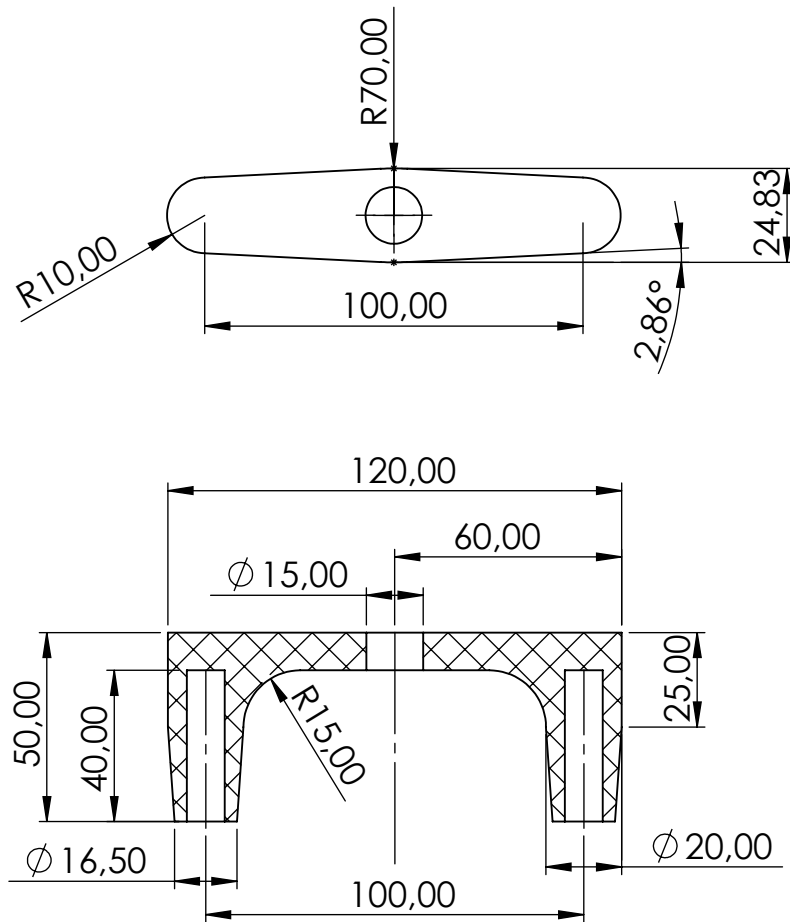
	Datum	Ime i prezime	Potpis				
Projektirao		Karlo Brzica					
Razradio		Karlo Brzica					
Crtao		Karlo Brzica					
Pregledao							
Objekt:			Objekt broj:				
			R. N. broj:				
Napomena: izrađeno 3D printanjem			Mehatronika i robotika		Kopija		
Materijal: PETG		Masa:					
 Naziv:			Pozicija:				
Mjerilo originala			Nosač postolja		6		
1:2			Crtež broj: RS-00-01		Format: A4		
					Listova: 1		
					List: 1		





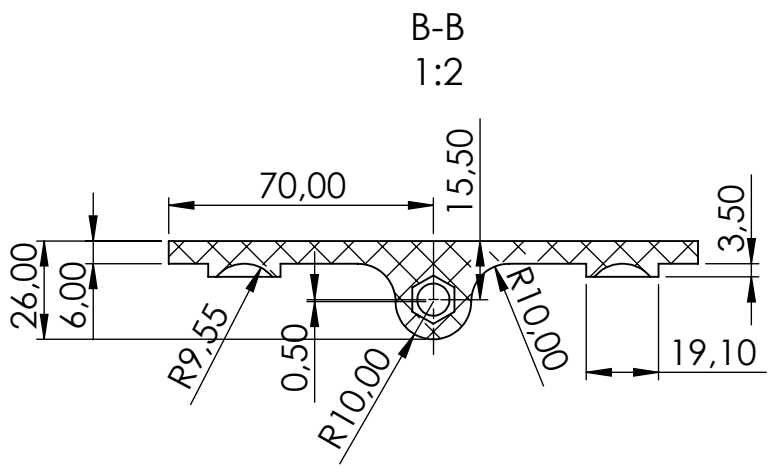
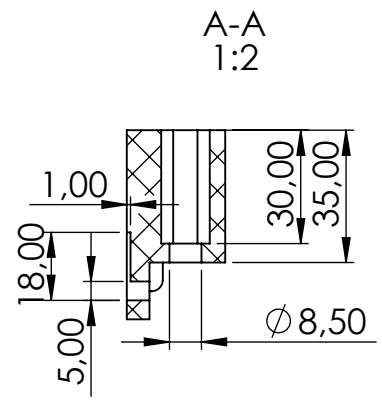
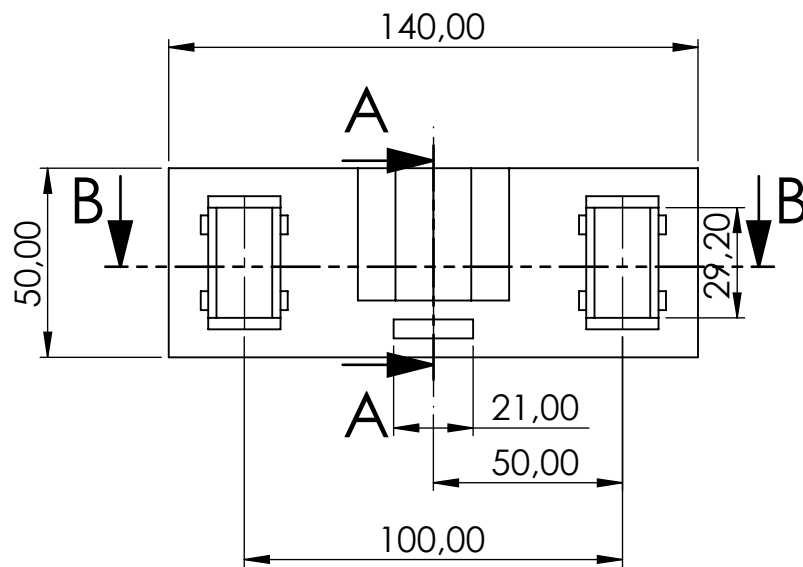
	Datum	Ime i prezime	Potpis	 <b>FSB Zagreb</b>
Projektirao		Karlo Brzica		
Razradio		Karlo Brzica		
Crtao		Karlo Brzica		
Pregledao				
Objekt:			Objekt broj:	
			R. N. broj:	
Napomena:	izrađeno 3D printanjem		Mehatronika i robotika	Kopija
Materijal:	PETG	Masa:		
 Mjerilo originala	Naziv:		Pozicija:	
1:1	Prijenosnik		8	
Crtež broj:			RS-00-02	Format: A4
				Listova: 1
				List: 1


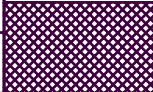



	Datum	Ime i prezime	Potpis	 <b>FSB Zagreb</b>
Projektirao		Karlo Brzica		
Razradio		Karlo Brzica		
Crtao		Karlo Brzica		
Pregledao				
Objekt:			Objekt broj:	
			R. N. broj:	
Napomena:			Mehatronika i robotika	Kopija
Materijal:	PMMA	Masa:		
	Naziv:		Pozicija:	
Mjerilo originala	Postolje		9	
1:5	Crtež broj: RS-00-03			Format: A4
				Listova: 1
				List: 1

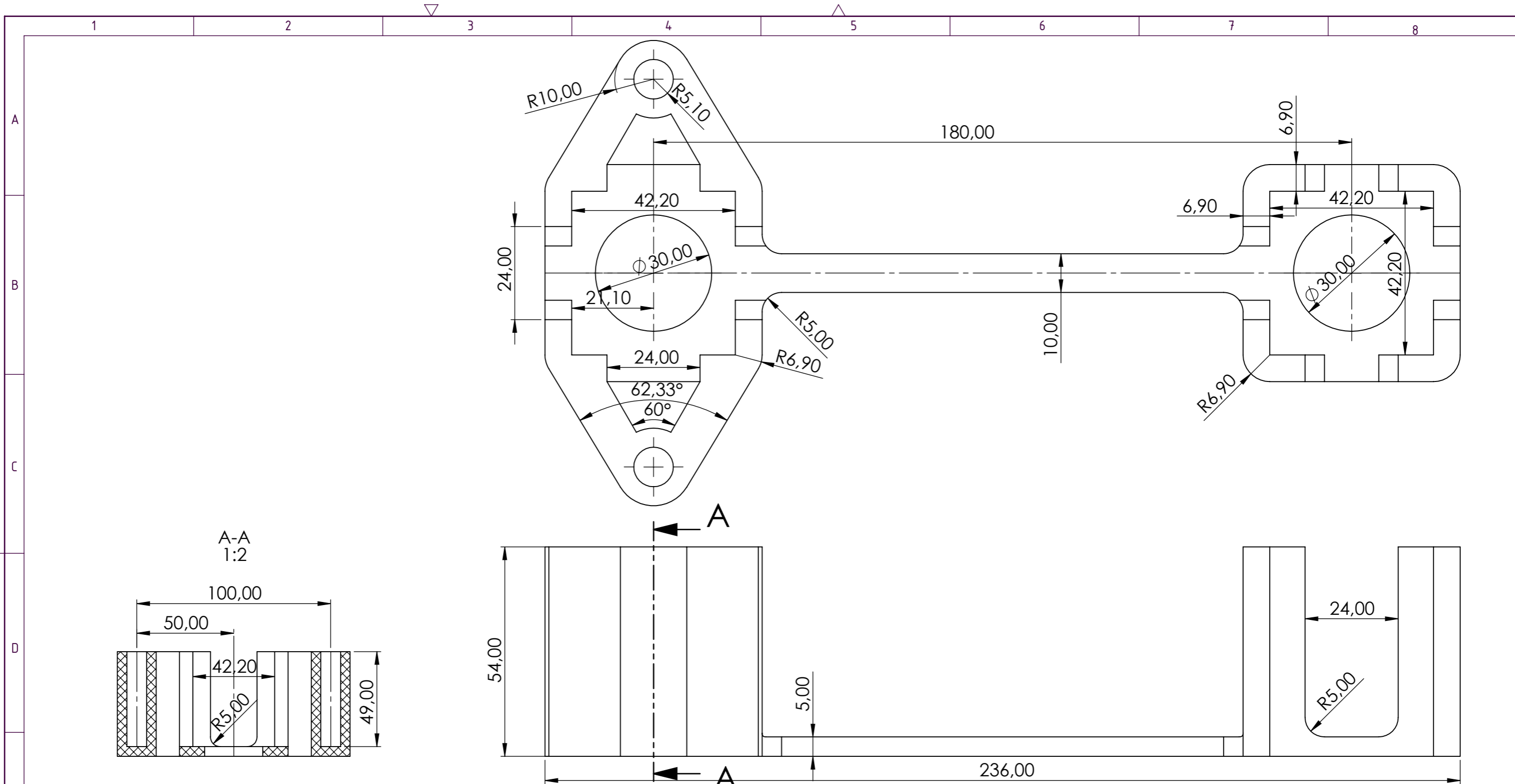


	Datum	Ime i prezime	Potpis	 <b>FSB Zagreb</b>
Projektirao		Karlo Brzica		
Razradio		Karlo Brzica		
Crtao		Karlo Brzica		
Pregledao				
Objekt:			Objekt broj:	
			R. N. broj:	
Napomena: izrađeno 3D printanjem			Mehatronika i robotika	Kopija
Materijal: PLA		Masa:		
 Mjerilo originala	Naziv:		Pozicija:	
	Razmaknica		11	Format: A4
1:2	Crtež broj: RS-00-04			Listova: 1
				List: 1

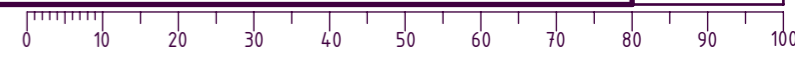


	Datum	Ime i prezime	Potpis	 <b>FSB Zagreb</b>	
Projektirao		Karlo Brzica			
Razradio		Karlo Brzica			
Crtao		Karlo Brzica			
Pregledao					
Objekt:		Objekt broj:			
		R. N. broj:			
Napomena: izrađeno 3D printanjem		Mehatronika i robotika		Kopija	
Materijal: PLA		Masa:			
		Naziv:			Pozicija:
Mjerilo originala		Nosač senzora			12
1:2		Crtež broj: RS-00-05		Format: A4	
				Listova: 1	
				List: 1	

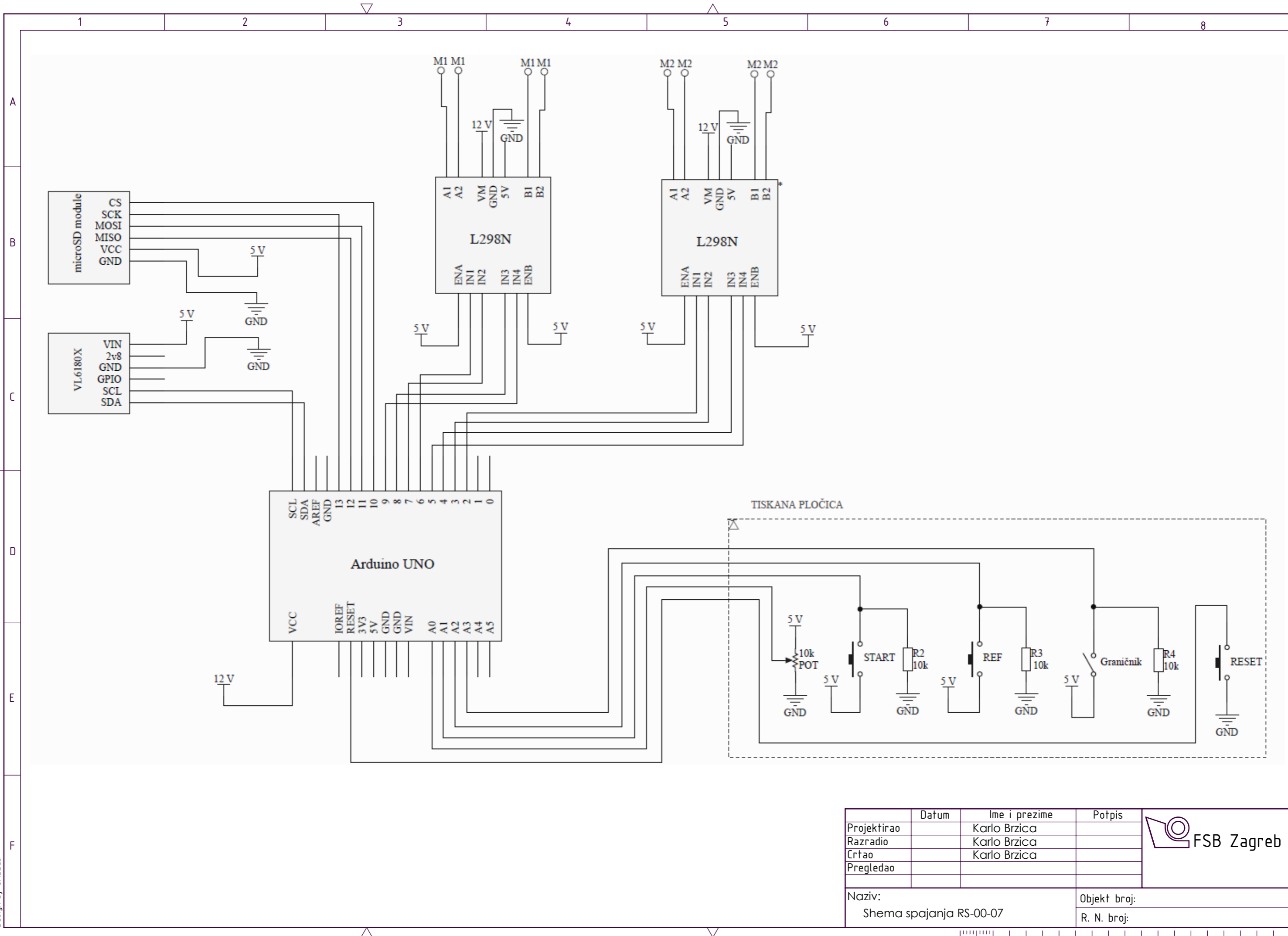
Design by CADLab



Broj naziva - code	Projektirao	Datum	Ime i prezime	Potpis	
	Razradio		Karlo Brzica		
	Crtao		Karlo Brzica		
	Pregledao		Karlo Brzica		
	Mentor				
ISO - tolerancije	Objekt:			Objekt broj:	
				R. N. broj:	
	Napomena:			Mehatronika i robotika	Kopija
	Materijal:	PLA	Masa:		
		Naziv:		Pozicija:	Format: A3
	Mjerilo originala	Nosač motora		18	Listova: 1
	1:1	Crtež broj: RS-00-06		List: 1	



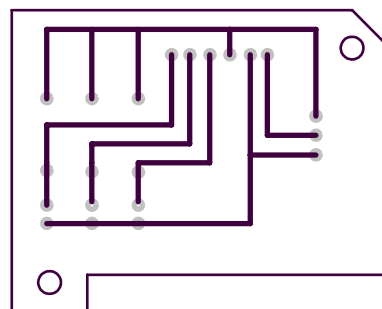
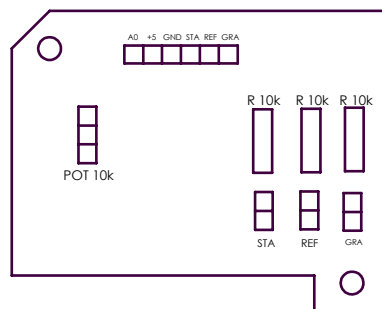




Design by CADlab

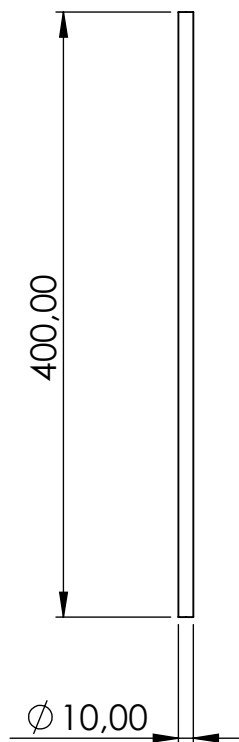
	Datum	Ime i prezime	Potpis
Projektirao		Karlo Brzica	
Razradio		Karlo Brzica	
Crtao		Karlo Brzica	
Pregledao			
Naziv: Shema spajanja RS-00-07		Objekt broj: R. N. broj:	


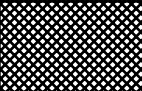





	Datum	Ime i prezime	Potpis
Projektirao		Karlo Brzica	
Razradio		Karlo Brzica	
Crtao		Karlo Brzica	
Pregledao			
Naziv: Tiskana pločica RS-00-08			Objekt broj: R. N. broj:





	Datum	Ime i prezime	Potpis	 <b>FSB Zagreb</b>
Projektirao		Karlo Brzica		
Razradio		Karlo Brzica		
Crtao		Karlo Brzica		
Pregledao				
Objekt:			Objekt broj:	
			R. N. broj:	
Napomena:			Mehatronika i robotika	Kopija
Materijal:	Aluminij	Masa:		
 Mjerilo originala	Naziv:		Pozicija:	
1:5	Vodilica		Format: A4	
	Crtež broj: RS-00-09		Listova: 1	
			List: 1	

1 2 3 4 5 6 7 8

A

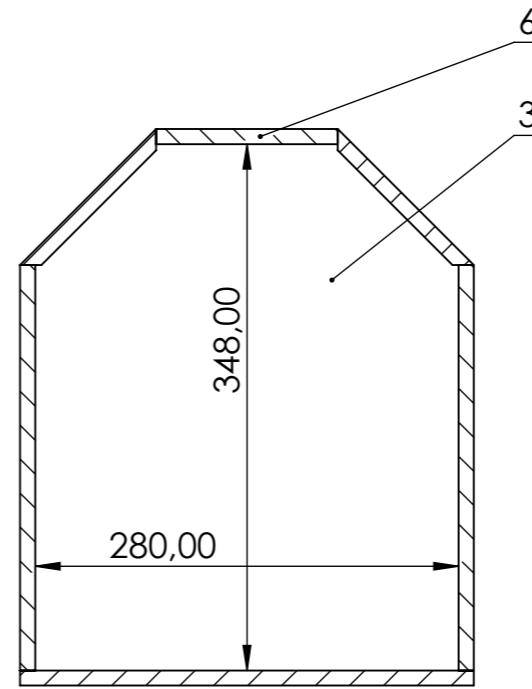
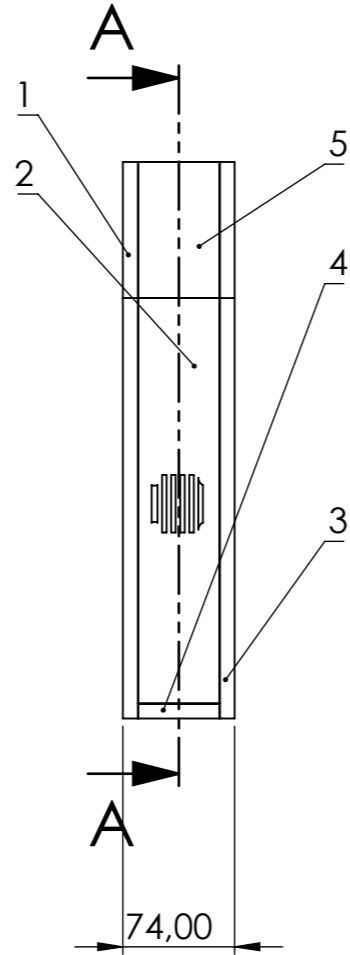
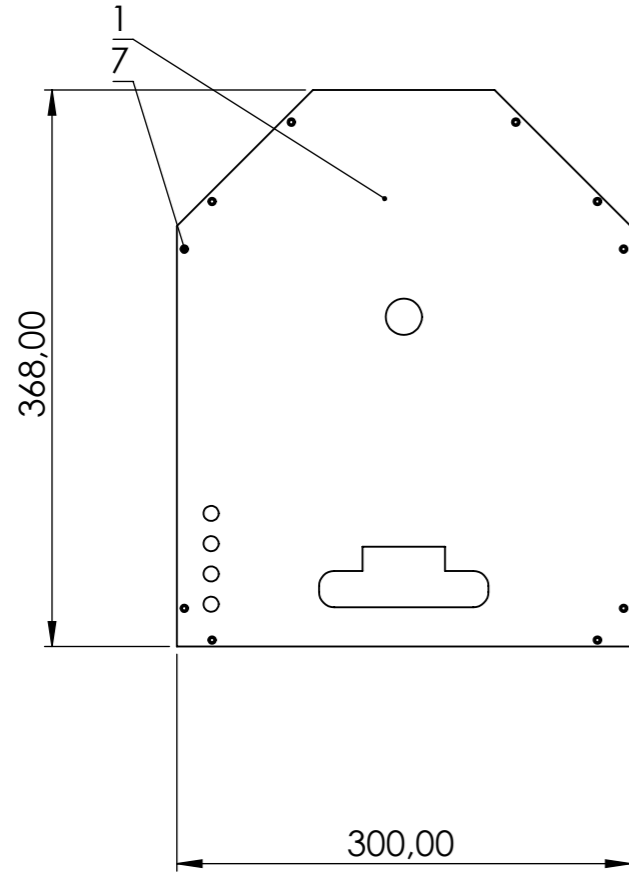
B

C

D

E

F

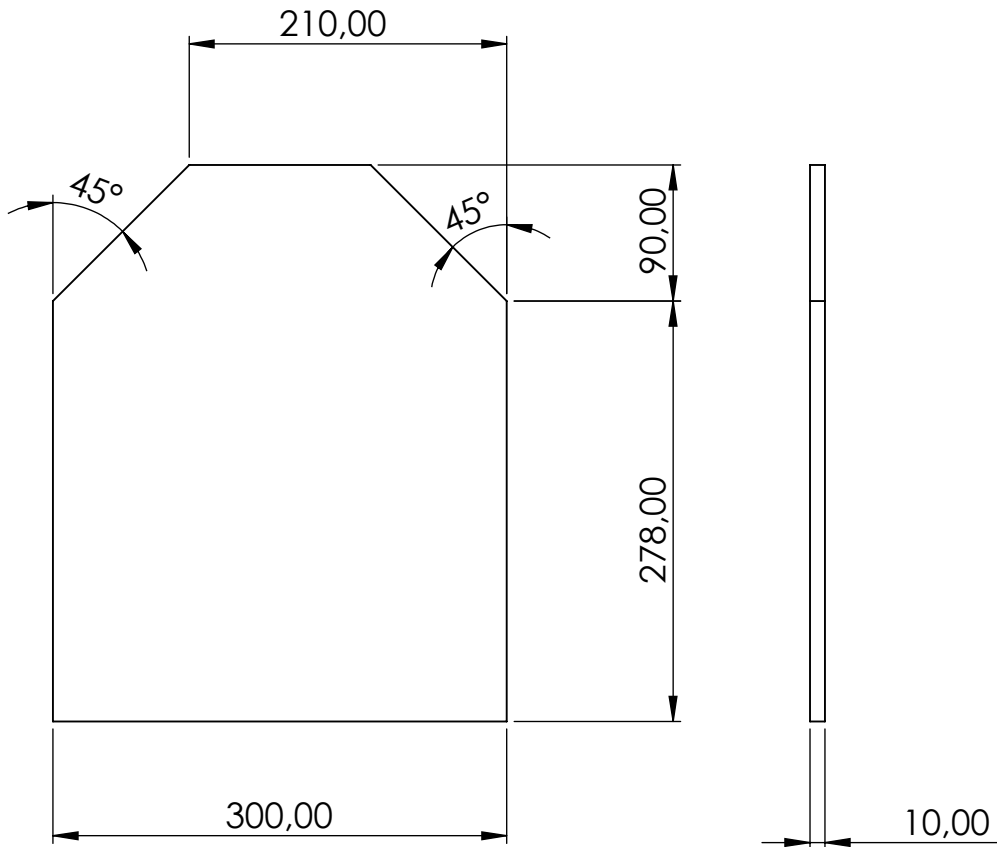



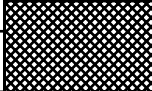
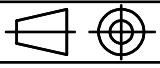
A-A  
1:5

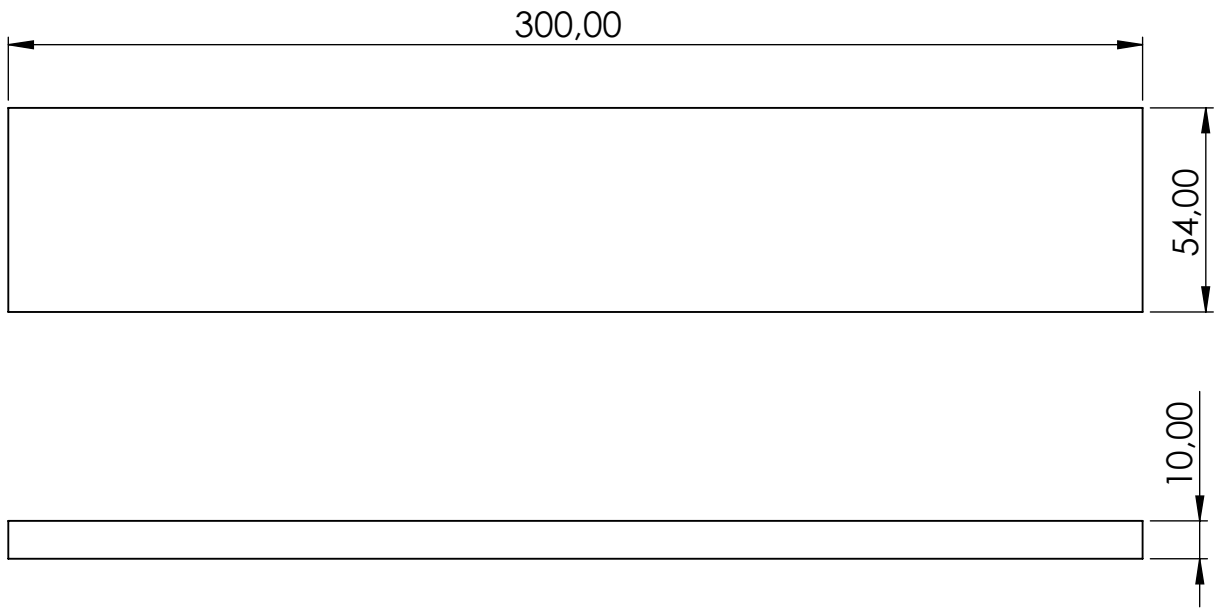
7	Vijak 3,5 x 25	22	DIN: MAC 11			
6	Prednja	1	RS-01-06	PMMA		
5	Bočna gornja	1	RS-01-05	PMMA		
4	Leđna stranica	1	RS-01-04	PMMA		
3	Podnica	1	RS-01-03	PMMA		
2	Bočna stranica	2	RS-01-02	PMMA		
1	Poklopac	1	RS-01-01	PMMA		
Poz.	Naziv dijela	Kom.	Crtež broj Norma	Materijal	Sirove dimenzije Proizvođač	Masa
Broj naziva - code		Datum	Ime i prezime	Potpis		
Projektirao			Karlo Brzica			
Razradio			Karlo Brzica			
Crtao			Karlo Brzica			
Pregledao						
Mentor						
ISO - tolerancije		Objekt:		Objekt broj:		
				R. N. broj:		
		Napomena:		Mehatronika i robotika		Kopija
		Materijal:		Masa:		
				Naziv:		Pozicija:
		Mjerilo originala		Sklop kućišta		1
		1:5		Crtež broj: RS-01-00		Listova: 1
						List: 1





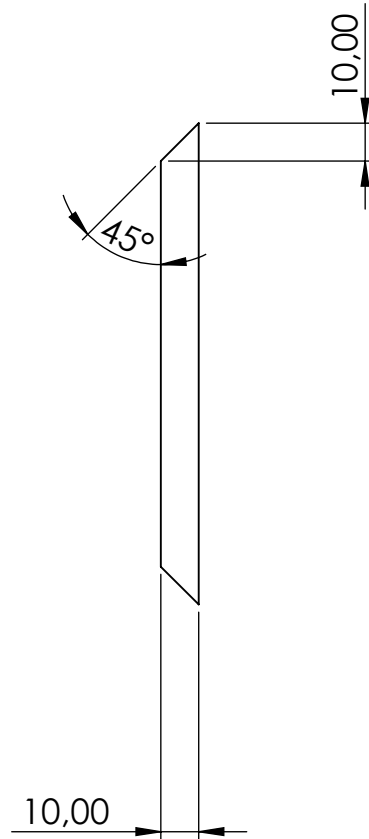
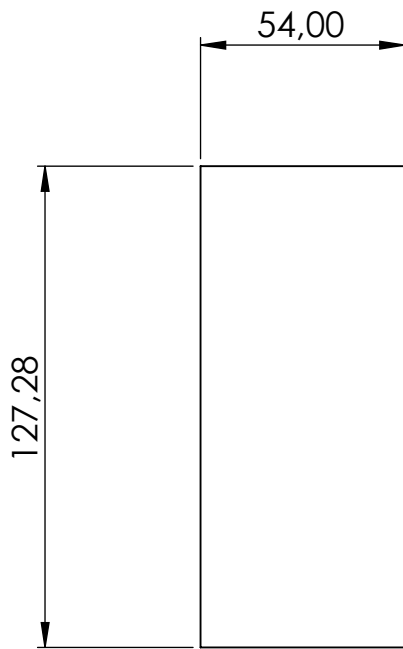


	Datum	Ime i prezime	Potpis	 <b>FSB Zagreb</b>
Projektirao		Karlo Brzica		
Razradio		Karlo Brzica		
Crtao		Karlo Brzica		
Pregledao				
Objekt:			Objekt broj:	
			R. N. broj:	
Napomena:			Mehatronika i robotika	Kopija
Materijal:	PMMA	Masa:		
	Naziv:		Pozicija:	Format: A4
Mjerilo originala	Podnica		3	Listova: 1
1 : 5	Crtež broj: RS-01-03			List: 1

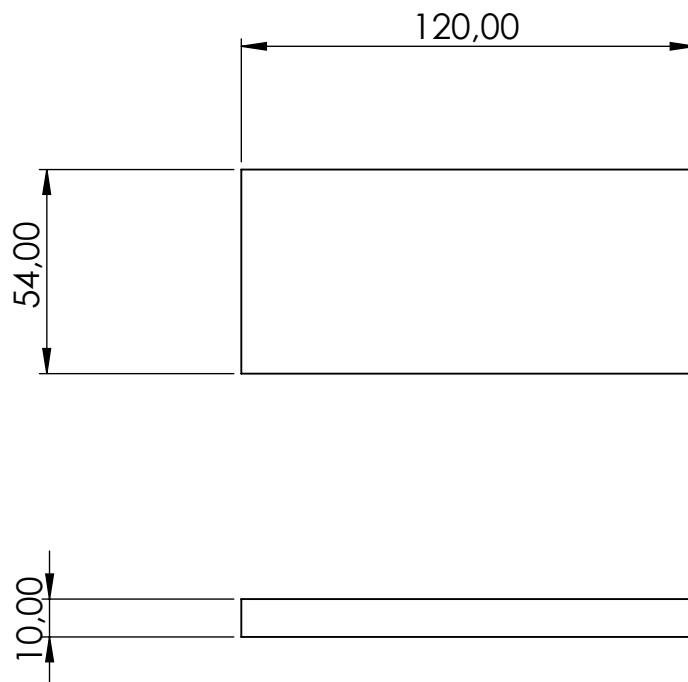



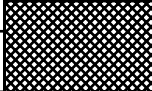
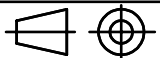
	Datum	Ime i prezime	Potpis	 <b>FSB Zagreb</b>
Projektirao		Karlo Brzica		
Razradio		Karlo Brzica		
Crtao		Karlo Brzica		
Pregledao				
Objekt:			Objekt broj:	
			R. N. broj:	
Napomena:			Mehatronika i robotika	Kopija
Materijal:	PMMA	Masa:		
	Naziv:		Pozicija:	Format: A4
Mjerilo originala	Leđna ploča		4	Listova: 1
1 : 2	Crtež broj: RS-01-04			List: 1





	Datum	Ime i prezime	Potpis	 <b>FSB Zagreb</b>
Projektirao		Karlo Brzica		
Razradio		Karlo Brzica		
Crtao		Karlo Brzica		
Pregledao				
Objekt:		Objekt broj:		
		R. N. broj:		
Napomena:		Meatronika i robotika		Kopija
Materijal:	PMMA	Masa:		
	Naziv:		Pozicija:	Format: A4
Mjerilo originala	Bočna gornja ploča		5	Listova: 1
1 : 2	Crtež broj: RS-01-05			List: 1



	Datum	Ime i prezime	Potpis	 <b>FSB Zagreb</b>
Projektirao		Karlo Brzica		
Razradio		Karlo Brzica		
Crtao		Karlo Brzica		
Pregledao				
Objekt:			Objekt broj:	
			R. N. broj:	
Napomena:			Mehatronika i robotika	Kopija
Materijal:	PMMA	Masa:		
	Naziv:		Pozicija:	
Mjerilo originala	Prednja ploča		6	
1 : 2	Crtež broj: RS-01-06			Format: A4
				Listova: 1
				List: 1

## PRILOG: Arduino kod

```
///ZAVRŠNI RAD, Karlo Brzica, 2019, Zagreb
```

```
///3D rotacijski skener
```

```
///Fakultet strojarstva i brodogradnje
```

```
///Sveučilište u Zagrebu
```

```
#include <SPI.h>
```

```
#include <SD.h>
```

```
#include <Wire.h>
```

```
#include "Adafruit_VL6180X.h"
```

```
Adafruit_VL6180X vl = Adafruit_VL6180X();
```

```
//definiranje varijabli
```

```
File myFile;
```

```
int pinCS = 10;      //pin 10 za komunikaciju sa memorijskim modulom
```

```
int postoljePin1 = 5; //pin koračnog motora postolja int 4
```

```
int postoljePin2 = 4; //pin koračnog motora postolja 3
```

```
int postoljePin3 = 3; //pin koračnog motora postolja 2
```

```
int postoljePin4 = 2; //pin koračnog motora postolja int 1
```

```
int vretenoPin1 = 9; // pin koračnog motora vretena int4
```

```
int vretenoPin2 = 8; // pin koračnog motora vretena 3
```

```
int vretenoPin3 = 7; // pin koračnog motora vretena 2
```

```
int vretenoPin4 = 6; // pin koračnog motora vretena 1
```

```
int i=0;           //pomoćna varijabla
```

```
int del=50;        //delay postolja
```

```
int delv=25;       //delay postolja
```

```
float visina=0;    //pomoćna varijabla visine
```

```
int analogPin3=A3;    // definiranje naziva analognih pinova
int analogPin2=A2;
int analogPin1=A1;
int analogPin0=A0;
int potencijetar=0;  //definiranje vrijednosti analognih pinova
int start=0;
int ref=0;
int granicnik=0;

int korak=5;        //željeni pomaka z osi [mm]

float visina_predmeta=0; //visina objekta skeniranja [mm]

float udaljenost=0;   //udaljenost dobivena mjerenjem senzora [mm]
float stupanj=0;     //pomoćna varijabla zakreta postolja

float x=0;          //X koordinata
float y=0;          //Y koordinata
float z=0.0;

//postavke
void setup()
{
  pinMode(postoljePin1,OUTPUT);
  pinMode(postoljePin2,OUTPUT);
  pinMode(postoljePin3,OUTPUT);
  pinMode(postoljePin4,OUTPUT);    //definiranje ulaza / izlaza
  pinMode(vretenoPin1,OUTPUT);
  pinMode(vretenoPin2,OUTPUT);
  pinMode(vretenoPin3,OUTPUT);
  pinMode(vretenoPin4,OUTPUT);
  pinMode(pinCS, OUTPUT);
```

```
Serial.begin(9600);          //uključivanje serijskog slanja podataka

while (!Serial)
{
  delay(1);
}

if (! vl.begin())          //čekanje za uspostavu komunikacije sa senzorom udaljenosti
{
  while (1);
}

}

///
///glavna petlja programa
///

void loop()
{
  //Serial.print("Pritisnite start ili započnite referenciranje");

  start = analogRead(analogPin2);    // čitanje stanja analognih ulaza (tipkala start i ref)
  ref = analogRead(analogPin1);

  while(start<999)          // provjera stanja tipkala start
  {
```

```

start = analogRead(analogPin2);
ref = analogRead(analogPin1);
if(ref>999)          //provjera stanja tipkala ref
{
  //Serial.print("Proces referenciranja započinje");
  delay(del);
  referenciranje(); //pozivanje funkcije referenciranje
}

}

potenciometar = analogRead(analogPin0); //čitanja stanja analognog ulaza potenciometra
Serial.print(potenciometar);
visina_predmeta=potenciometar/4.092; // 1023/250=4,092 --> radno područje do 250mm

while(z<visina_predmeta) //ponavljanje radnje sve dok visina senzora nije veća od visine
objekta
{
  okretanjepostolja(z); //pozivanje funkcije okretanja postolja
  okretanjevretena(); //pozivanje funkcije okretanja vretena
  z=z+korak; //povećavanje varijable visine senzora za korak
}
}

//
// pozicioniranje nosača senzora u nultu poziciju
//

void referenciranje() //funkcija referenciranja
{
  //Serial.print("Referenciranje");

```

```

granicnik = analogRead(analogPin3);    //čitanje stanja graničnika
if(granicnik>999)                       //provjera stanja graničnika
{
    loop();                               //povratak u funkciju loop (ako je graničnik uključen)
}
while(1)                                 //beskonačna petlja
{
    digitalWrite(vretenoPin1, LOW);
    digitalWrite(vretenoPin2, LOW);      //pokretanje koračnog motora
    digitalWrite(vretenoPin3, LOW);
    digitalWrite(vretenoPin4, HIGH);
    granicnik = analogRead(analogPin3);
    Serial.print(granicnik);
    Serial.print("\n");
    if(granicnik>999)                   //provjera stanja graničnika
    {
        loop();
    }
    delay(delv);                        //čekanje
    digitalWrite(vretenoPin1, LOW);
    digitalWrite(vretenoPin2, HIGH);
    digitalWrite(vretenoPin3, LOW);
    digitalWrite(vretenoPin4, LOW);
    granicnik = analogRead(analogPin3);
    if(granicnik>999)
    {
        loop();
    }
    delay(delv);
    digitalWrite(vretenoPin1, LOW);
    digitalWrite(vretenoPin2, LOW);
    digitalWrite(vretenoPin3, HIGH);
    digitalWrite(vretenoPin4, LOW);
}

```

```

    granicnik = analogRead(analogPin3);
    if(granicnik>999)
    {
        loop();
    }
    delay(delv);
    digitalWrite(vretenoPin1, HIGH);
    digitalWrite(vretenoPin2, LOW);
    digitalWrite(vretenoPin3, LOW);
    digitalWrite(vretenoPin4, LOW);
    granicnik = analogRead(analogPin3);
    if(granicnik>999)
    {
        loop();
    }
    delay(delv);
}
}

///
///okretanje postolja
///

void okretanjepostolja(float z)    // funkcija okretanja postolja
{
    while(stupanj < 359)           // ponavljajuća petlja sve dok nije ispunjen uvijet
    {
        digitalWrite(postoljePin1, HIGH);
        digitalWrite(postoljePin2, LOW);
        digitalWrite(postoljePin3, LOW);    // pokretanje koračnog motora
        digitalWrite(postoljePin4, LOW);
    }
}

```



```

delay(del);           // čekanje
stupanj=stupanj+1.8; // povećavanje pomoćne varijable stupnja za kut zakreta motora
citanje(stupanj,z);  // pozivanje funkcije citanje sa varijablama stupanj i visine senzora
digitalWrite(postoljePin1, LOW);
digitalWrite(postoljePin2, LOW);
digitalWrite(postoljePin3, HIGH);
digitalWrite(postoljePin4, LOW);
delay(del);
stupanj=stupanj+1.8;
citanje(stupanj,z);
digitalWrite(postoljePin1, LOW);
digitalWrite(postoljePin2, HIGH);
digitalWrite(postoljePin3, LOW);
digitalWrite(postoljePin4, LOW);
delay(del);
stupanj=stupanj+1.8;
citanje(stupanj,z);
digitalWrite(postoljePin1, LOW);
digitalWrite(postoljePin2, LOW);
digitalWrite(postoljePin3, LOW);
digitalWrite(postoljePin4, HIGH);
delay(del);
stupanj=stupanj+1.8;
citanje(stupanj,z);

}

stupanj=0;           // resetiranje varijable stupanj
}

///  

///  

///  


```

```

void okretanjevretena()           // funkcija okretanje vretena
{
  while(visina < korak)           // ponavljajuća petlja dok nije ispunjen uvijet
  {

    while(stupanj < 360)           // ponavljajuća petlja dok nije ispunjen uvijet
    {
      digitalWrite(vretenoPin1, HIGH);
      digitalWrite(vretenoPin2, LOW);
      digitalWrite(vretenoPin3, LOW);
      digitalWrite(vretenoPin4, LOW);
      delay(delv);
      digitalWrite(vretenoPin1, LOW);
      digitalWrite(vretenoPin2, LOW);
      digitalWrite(vretenoPin3, HIGH);
      digitalWrite(vretenoPin4, LOW);
      delay(delv);
      digitalWrite(vretenoPin1, LOW);
      digitalWrite(vretenoPin2, HIGH);
      digitalWrite(vretenoPin3, LOW);
      digitalWrite(vretenoPin4, LOW);
      delay(delv);
      digitalWrite(vretenoPin1, LOW);
      digitalWrite(vretenoPin2, LOW);
      digitalWrite(vretenoPin3, LOW);
      digitalWrite(vretenoPin4, HIGH);
      delay(delv);
      stupanj=stupanj+7.2;         // povećavanje pomoćne varijable stupnja za kut zakreta
motora
    }
    stupanj=0;                     // resetiranje pomoćne varijable
    visina=visina+1.25;           // povećavanje pomoćne varijable visina za korak navojnog
vretena
  }
}

```

```

}
visina=0;           // resetiranje pomoćne varijable
}

///
///analiziraje stanja senzora i spremanje na micro SD karticu
///

int citanje(float stupanj,float z)      // funkcija citanje sa ulaznim varijablama stupanj i z
{

while(i < 25)           // ponavljajuća petlja dok nije ispunjen uvijet mjerenja 25 uzoraka
{
uint8_t range = vl.readRange();      // mjerenje udaljenosti
udaljenost=udaljenost+range;        // sumiranje vrijednosti
i++;
}
i=0;           // resetiranje vrijednosti
udaljenost=udaljenost/25;
uint8_t status = vl.readRangeStatus(); // provjera statusa senzora
x=cos(stupanj/57.29578)*(160-udaljenost); // računanje X koordinate
y=sin(stupanj/57.29578)*(160-udaljenost); // računanje Y koordinate
udaljenost=0;           // resetiranje vrijednosti

if (SD.begin())           // provjera modula
{
Serial.println("SD card is ready to use.");
}
else
{
Serial.println("SD card initialization failed");
}
}

```

```
return;
}

myFile = SD.open("x.txt", FILE_WRITE); // otvaranje/kreiranje datoteke x.txt:
if (myFile) // provjera uspješnosti otvaranja
{
myFile.print(x);myFile.println(""); // upisivanje x koordinate u datoteku
myFile.close(); // zatvaranje datoteke
}

myFile = SD.open("y.txt", FILE_WRITE);
if (myFile)
{
myFile.print(y);myFile.println("");
myFile.close();
}

myFile = SD.open("z.txt", FILE_WRITE);
if (myFile)
{
myFile.print(z);myFile.println("");
myFile.close();
}
}
```