

**Elektrotehnička i ekonomska škola**

**Nova Gradiška**



# **AUTOMATIZIRANA LINIJA ZA PUNJENJE BOČICA**

## **Učenici**

Filip Jažo, Luka Butina, Matko Radošić, Robert Lukić

## **Mentor**

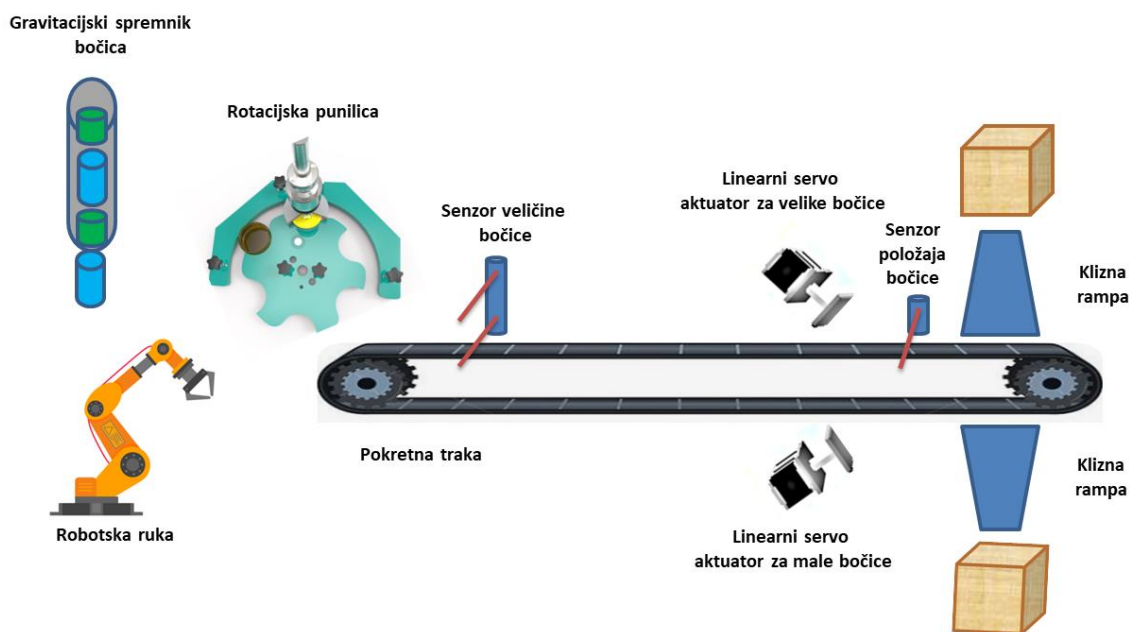
Marijan Vuk

## SADRŽAJ:

1. UVOD.....	3
2. ROBOTSKA RUKA.....	5
3. ROTACIJSKA PUNILICA.....	9
4. POKRETNNA TRAKA.....	12
5. ZAKLJUČAK.....	17
6. LITERATURA.....	18
7. PRILOZI.....	19

# UVOD

Ideja projekta je napraviti modularnu maketu koja bi omogućila i naknadno proširivanje dodatnim modulima. Maketa bi se koristila i u nastavi za izvođenje vježbi iz predmeta Mikroupravljači, PLC-i i mikroupravljači. Projekt je osmišljen u 3 modula. Prvi dio predstavlja robotska ruka koja se koristi za premještanje bočica u dvije veličine iz gravitacijskog spremnika na rotacijsku punilicu. Drugi dio predstavlja rotacijsku punilicu koja puni bočice i ostavlja ih na pokretnu traku. Treći dio predstavlja pokretnu traku koja sortira bočice prema veličini. Slika ispod prikazuje skicu makete po kojoj je rađen projekt.



Slika 1. Skica makete

Linija se sastoji od:

Gravitacijskog spremnika bočica – sadrži nekoliko bočica različitih visina

Robotske ruke – uzima bočica iz spremnika i postavlja ih na rotacijsku punilicu

Rotacijske punilice – puni bočice i ostavlja ih na pokretnu traku

Pokretne trake – prenosi bočice s rotacijske punilice do mjesta sortiranja

Senzor veličine bočice – detektira i šalje informacije o visini bočice linearnim servo aktuatorima

Senzor položaja bočice – detektira bočicu na položaju za sortiranje

Linearni servo aktuatori – sortiraju bočice prema visini

Klizne rampe – služe za spuštanje bočica sa trake



Slika 2. Gotova maketa automatizirane linije za punjenje bočica.

Na slici 2 se vidi gotova maketa projekta. Sva tri modula su povezana u jednu cjelinu koja čini automatiziranu liniju za punjenje bočica. U sljedećim cjelinama bit će opisan pojedini modul.

# ROBOTSKA RUKA

Početni dio automatizirane linije za punjenje bočica sastoji se od robotske ruke i gravitacijskog spremnika bočica. Zadatak tog prvog modula je da robotska ruka uzme bočicu sa gravitacijskog spremnika te ju premjesti na rotacijsku punilicu. Ciklus rada se pokreće pritiskom na gumb.

Robotsku ruku prikazanu na slici 5. pokreću tri servo motora TowerPro MG955H prikazani na slici 3. TowerPro MG955H je tip servo motora koji se obično koristi u vozilima na daljinsko upravljanje, robotici i drugim elektroničkim aplikacijama. MG955H je servo motor s metalnim zupčanicima visokog okretnog momenta koji se može okretati za 360 stupnjeva. Izvršni članak robotske ruke ima hvataljku sa unutrašnjim prihvatom bočice koji je pokretan solenoidom JF0530B prikazan na slici 4.



Slika 3. TowerPro MG955H servo motor



Slika 4. Solenoid





Slika 5. Robotska ruka

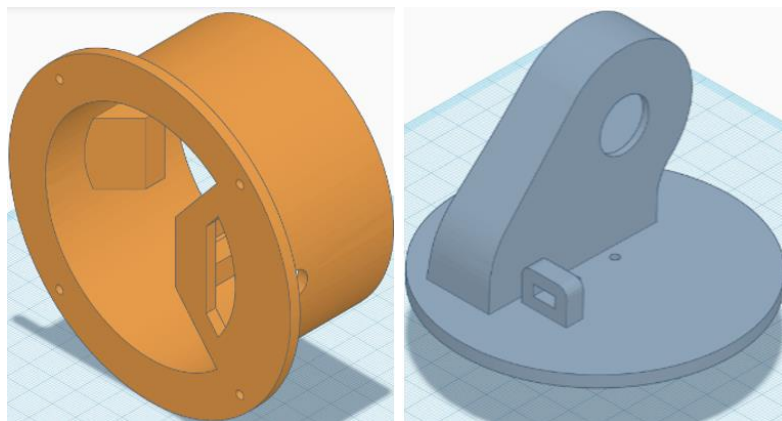
Za programiranje robotske ruke korišten je emoro 2560 kontroler prikazan na slici 6. koji je baziran na arduino ATmega2560 mikroupravljaču. Pločica je odabrana zbog većeg broja servo konektora koji omogućuje lakše povezivanje i upravljanje sa više servo motora.



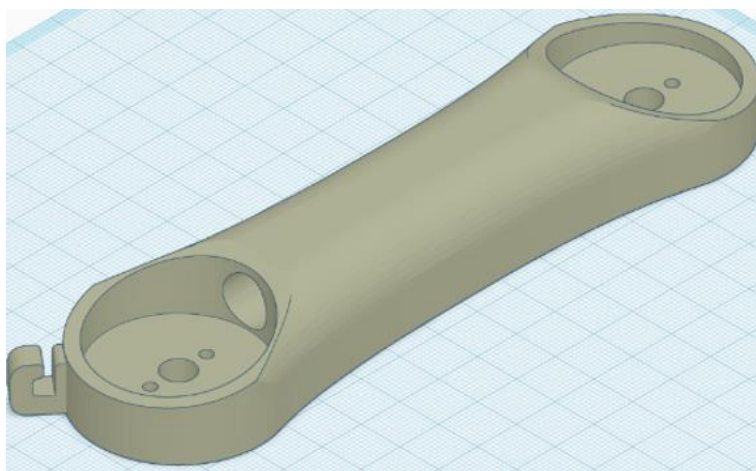
Slika 6. Emoro 2560 kontroler

U prilogu je prikazan kod za upravljanje robotom.

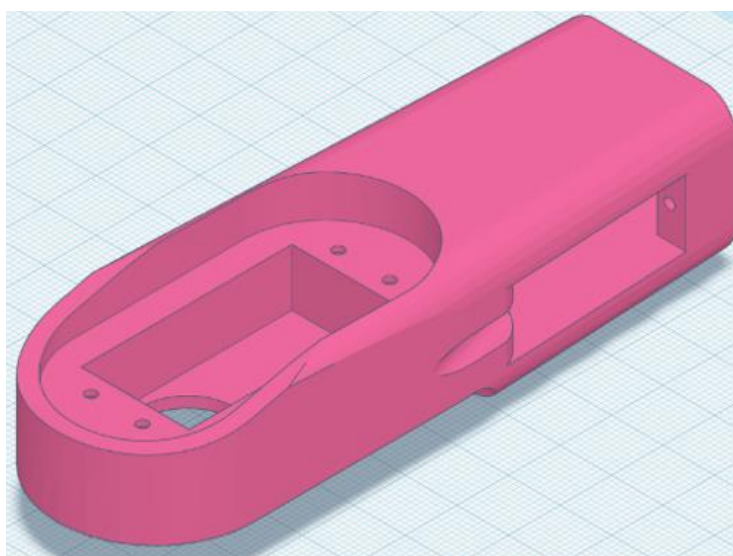
Za izradu robota isprintani su 3d modeli prikazani na slikama 7. do 11.



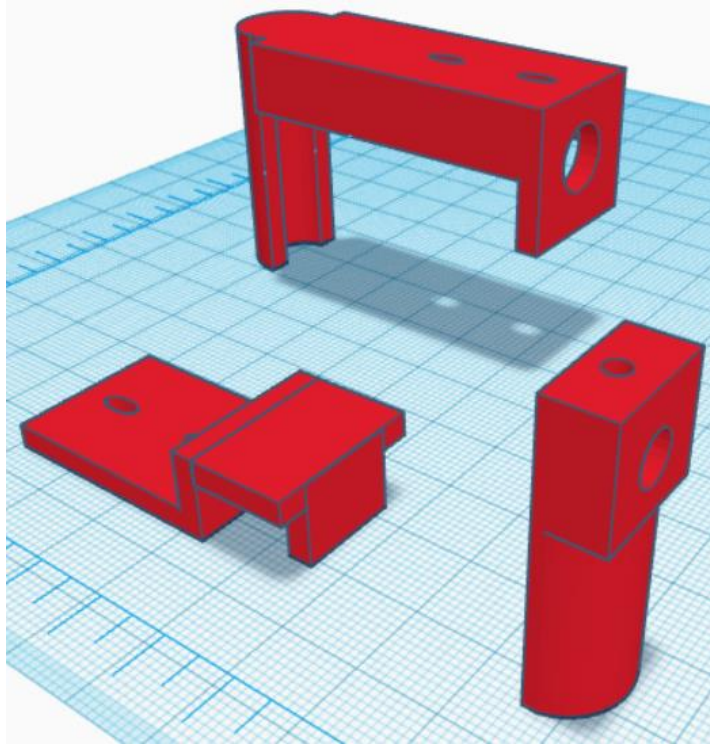
Slika 7. 3D modeli postolja robotske ruke



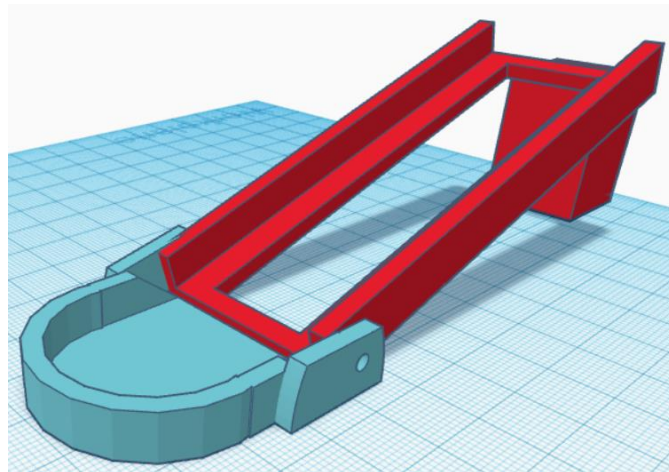
Slika 8. 3D model nadlaktice robotske ruke



Slika 9. 3D model podlaktice robotske ruke



Slika 10. 3D modeli hvataljke



Slika 11. 3D modeli gravitacijskog spremnika bočica



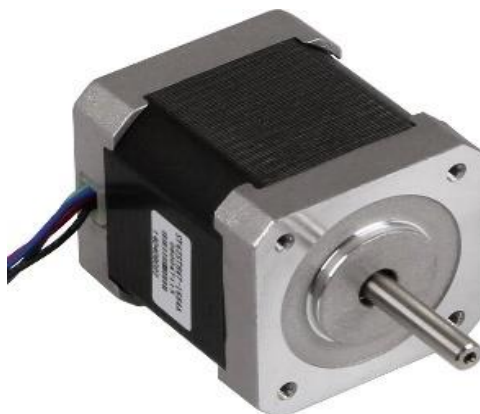
## ROTACIJSKA PUNILICA

Rotacijska punilica prikazana na slici 12. središnji je dio automatizirane linije za punjenje bočica. Sastoji se od dviju ploča: donje nepomične i gornje pomične ploče sa četiri utora za bočice. Služi za prihvat bočica od robotske ruke, punjenje bočica i prenošenje bočica na pokretnu traku. Za rotaciju gornje ploče sa utorama koristi se NEMA 17 koračni motor prikazan na slici 13. NEMA 17 koračni motori obično se koriste u različitim primjenama zbog svoje relativno kompaktne veličine, velikog izlaza okretnog momenta i kompatibilnosti s nizom upravljačkih programa i upravljačkih sustava.

Ciklus rada punilice pokreće se istim gumbom kojim se pokreće i robotska ruka. Nakon što robotska ruka postavi bočicu na donju nepomičnu ploču uključuje se koračni motor koji rotira gornju ploču sa utorama za kut 90 stupnjeva na poziciju za punjenje bočica. Nakon završetka procesa punjenja bočice, ploča se ponovno zakreće za kut 90 stupnjeva na poziciju s otvorom na donjoj nepomičnoj ploči kroz koji bočica spada na pokretnu traku.



Slika 12. Rotacijska punilica

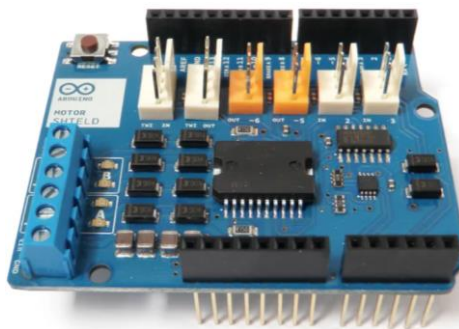


Slika 13. Rotacijska punilica

Za programiranje rotacijske punilice korištena je arduino Uno pločica prikazana na slici 14. te shield pločica za upravljanje koračnim motorom prikazana na slici 15 koji su odabrani radi lakšeg povezivanja koračnog motora sa arduino kontrolerom.



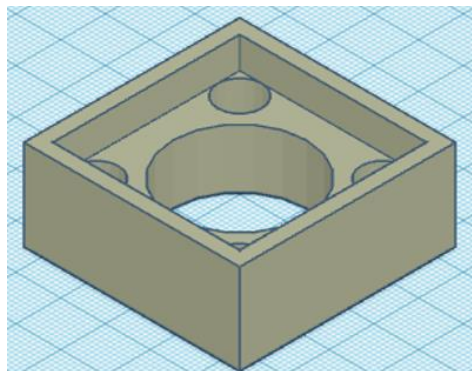
Slika 14. Arduino UNO pločica



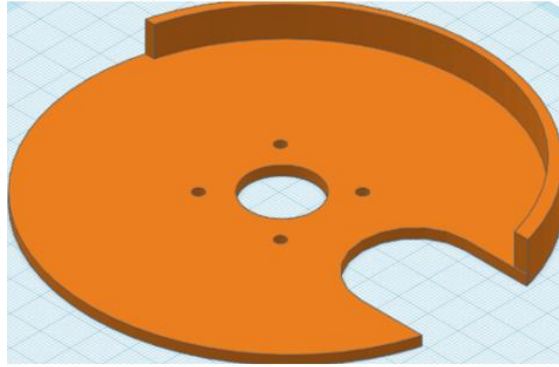
Slika 15. Arduino motor shield pločica

Kod korišten za upravljanje rotacijskom punilicom prikazan je u prilogu.

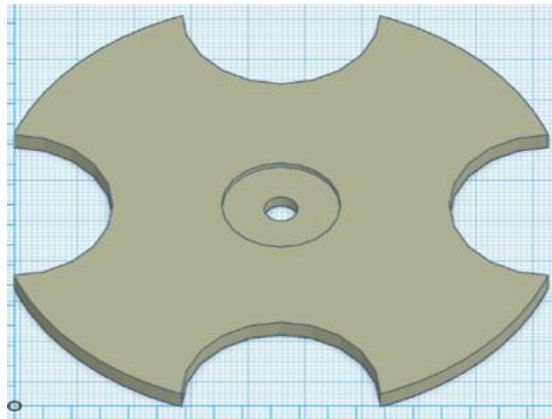
Za izradu rotacijske punilice isprintani su 3d modeli koji su prikazani na slikama 16. do 18.



Slika 16. 3D model postolja za koračni motor NEMA 17



Slika 17. 3D model donje nepomične ploče

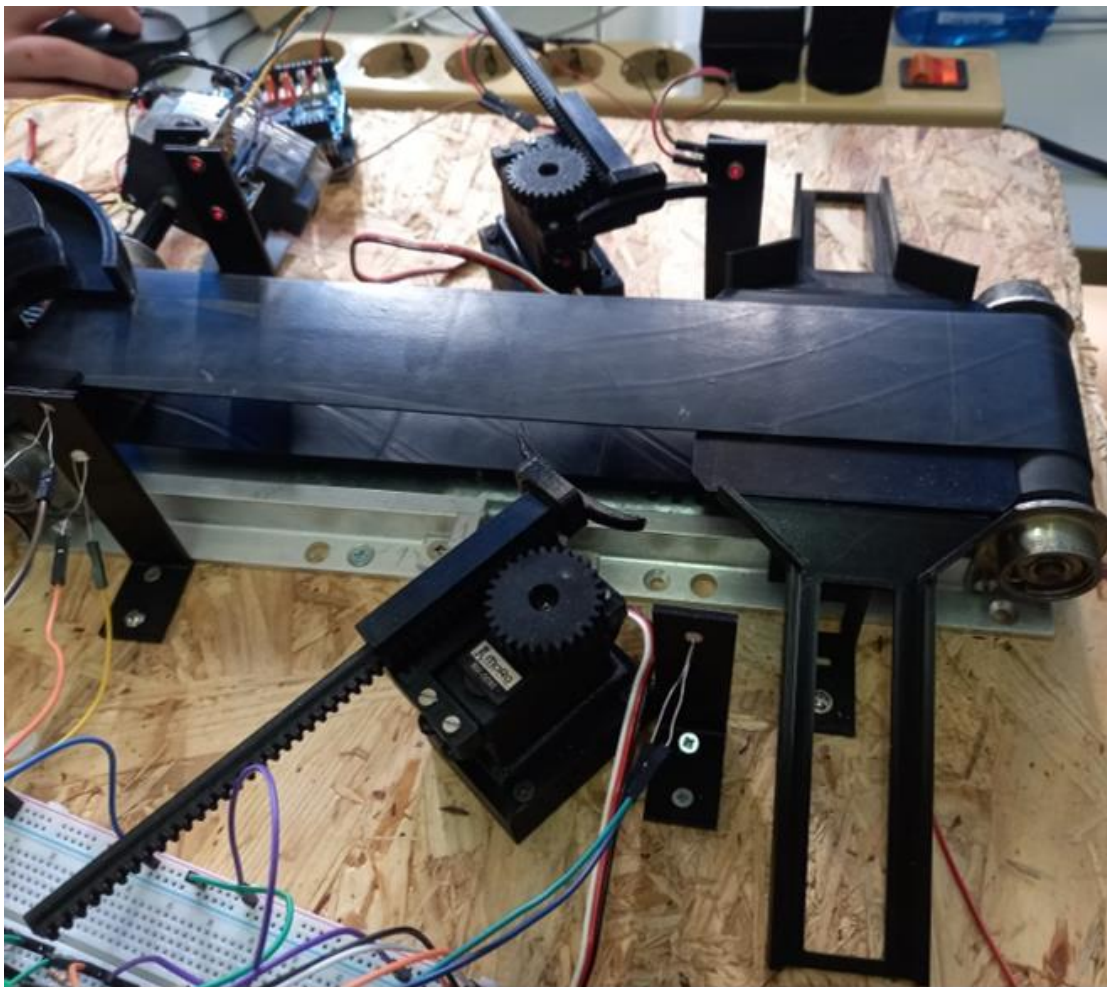


Slika 18. 3D model gornje pomične ploče sa utorima

## POKRETNNA TRAKA

Treći i posljednji dio automatizirane linije za punjenje bočica sastoji se od pokretne trake opremljene sensorima veličine bočica, sensorom položaja bočice, linearnim servo aktuatorima te kliznim rampama.

Pokretna traka prikazana na slici 19. ima funkciju detektirati, a zatim sortirati dvije različite dimenzije bočica. Na početak trake rotacijska punilica postavlja bočicu zatim je traka prenosi do pozicije za detekciju veličine bočice. Sensor veličine bočice sastoji se od dva lasera prikazana na slici 20. postavljena na dvije visine bočica te dva LDR senzora svjetlosti prikazana na slici 21. također postavljena na iste visine kao i laseri. LDR senzori detektiraju prekid laserske zrake uzorkovane bočicom i na taj način detektiraju visinu bočice. Pokretna traka zatim prenosi bočicu do položaja za sortiranje bočica na kojem se nalazi također laser sa LDR sensorom. Nakon prekida laserske zrake na položaju za sortiranje zaustavlja se pokretna traka te pokreće odgovarajući linearni servo aktuator prikazan na slici 22. koji gura bočicu sa pokretne trake niz kliznu rampu prikazanu na slici 23.



Slika 19. Pokretna traka za sortiranje bočica



Slika 20. Laser



Slika 21. LDR senzor



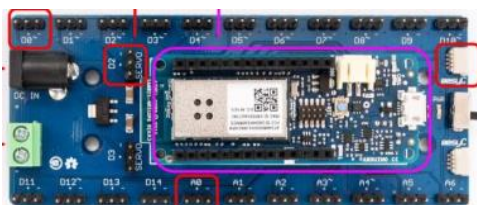
Slika 22. Linearni servo aktuator





Slika 23. Klizna rampa

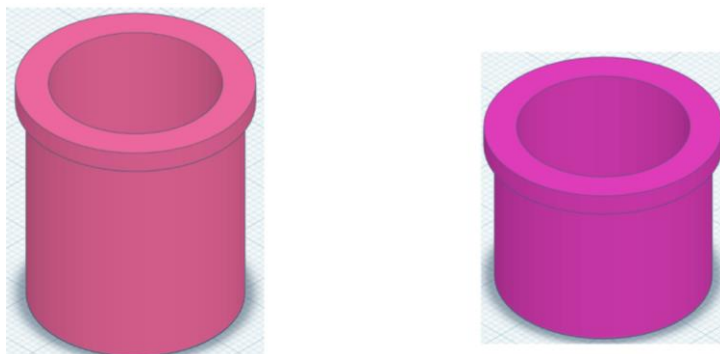
Za programiranje pokretne trake korištena je arduino MKR 1000 pločica sa shieldom prikazana na slici 24.



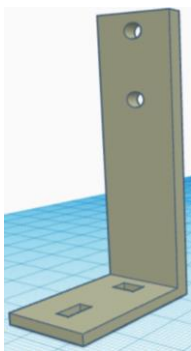
Slika 24. Arduino MKR1000 sa shieldom

Program korišten za upravljanje pokretnom trakom prikazan je u prilogu.

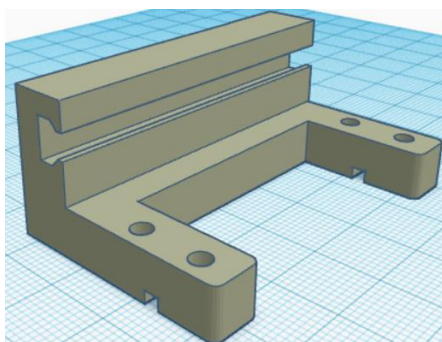
Za treći dio linije za automatizirano punjenje bočica napravljena je pokretna traka od valjaka korištenih kod starih igličnih printera i njihovih ležajeva sa nosačima te postolje pokretne trake od starih LCD monitora. Ostali dijelovi potrebni za rad koji su isprintani prikazani su na slikama 25. do 32.



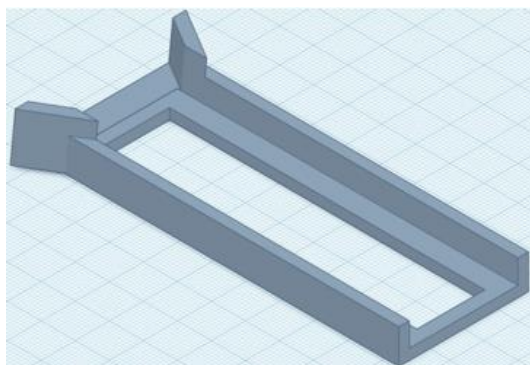
Slika 25. 3D modeli bočica



Slika 26. 3D modeli nosača lasera i LDR-a



Slika 27. 3D model vodilice klizne letve linearnog servo aktuatora



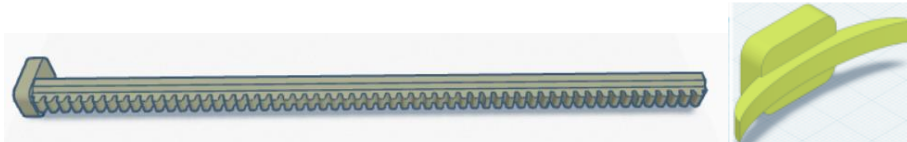
Slika 28. 3D model klizne rampe



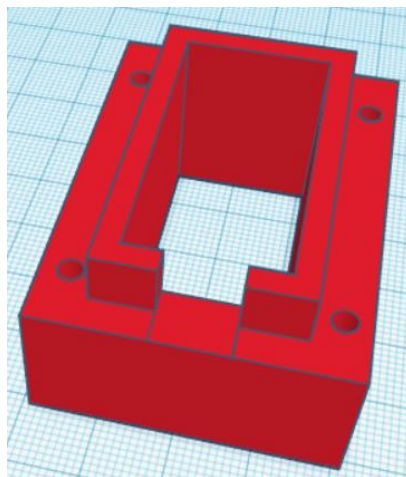
Slika 29. Nosač kliznih rampi



Slika 30. 3D model zupčanika linearnog servo aktuatora



Slika 31. 3D modeli zupčaste letve linearnog servo aktuatora



Slika 32. 3D model nosača linearnog servo aktuatora

## ZAKLJUČAK

Kombinacija robotske ruke, rotacijske punilice i pokretne trake s komponentom za sortiranje bočica nudi brojne prednosti proizvođačima. Sustav je dizajniran za rad velikom brzinom, omogućavajući kontinuirani protok boca kroz proces punjenja. To povećava produktivnost i smanjuje troškove rada, što ga čini privlačnom opcijom za proizvođače koji žele poboljšati svoju krajnju vrijednost.

Zaključno, automatizirane linije za punjenje bočica vrlo su učinkovito rješenje za proizvođače. Integracija robotske tehnologije revolucionirala je proizvodnu industriju, nudeći brojne prednosti uključujući povećanu brzinu, smanjene troškove rada i poboljšanu točnost. Kako potražnja za automatiziranim proizvodnim procesima nastavlja rasti, ova vrsta sustava postat će sve važnija za proizvođače koji žele poboljšati svoje proizvodne procese i povećati svoju konkurentnost na tržištu.

## LITERATURA

<https://www.youtube.com/>

<https://www.w3schools.com/>

<https://forum.arduino.cc/>



## **PRILOZI:**

### **Kod za robotsku ruku:**

```
#include <Servo.h>

Servo servo1;

Servo servo2;

Servo servo3;

Int i = 0;

Int l = 0;

Int k = 0;

Int j = 0;

Int h = 0;

Int g = 0;

    Int a = 89;

    Int b = 90;

Int RobotSpeed = 40;

Int RobotSpeed2 = 200;

Const int buttonPin = 23;

Int solenoid= 22;

Void setup() {

    Servo1.attach(47);

    Servo2.attach(46);

    Servo3.attach(45);

    pinMode(buttonPin,INPUT_PULLUP);

    pinMode(solenoid,OUTPUT);

}
```

```

Void loop() {
    Int digitalVal = digitalRead(buttonPin);
    If(HIGH == digitalVal){
        Pozicija_start();
    }else if(LOW == digitalVal){
        Pozicija_grabljenje();
    }
}

```

```

Void pozicija_start(){
    Servo1.write(100);
    Delay (RobotSpeed2);
    Servo2.write(140);
    Delay (RobotSpeed2);
    Servo3.write(70);
    Delay (RobotSpeed2);
    digitalWrite(solenoid, HIGH);
    delay (RobotSpeed2);
}

```

```

Void pozicija_grabljenje(){
    For( size_t i {140} ; i>=80;--i){
        Servo2.write(i);
        Delay (RobotSpeed);
    }
    For( size_t i {70} ; i>=5;--i){
        Servo3.write(i);
        Delay (RobotSpeed);
    }
}

```

```

}

For( size_t i {80} ; i>=20;--i){
    Servo2.write(i);
    Delay (RobotSpeed);
}

digitalWrite(solenoid, LOW);

////////////////////////////////////

For( size_t i {20} ; i<=40;++i){
    Servo2.write(i);
    Delay (RobotSpeed);
    For( size_t i {5} ; i>=1;--i){
        Servo3.write(i);
        Delay (RobotSpeed);
    }
}

For( size_t i {100} ; i>=73;--i){
    Servo1.write(i);
    Delay (RobotSpeed);
}

For( size_t i {40} ; i>=13;--i){
    Servo2.write(i);
    Delay (RobotSpeed);
}

digitalWrite(solenoid, HIGH);

```

```
////////////////////////////////////
```

```
For( size_t i {15}; i<=80;++i){  
    Servo2.write(i);  
    Delay (RobotSpeed);  
}  
For( size_t i {1}; i<=70;++i){  
    Servo3.write(i);  
    Delay (RobotSpeed);  
}  
digitalWrite(solenoid, HIGH);  
for( size_t i {80}; i<=140;++i){  
    servo2.write(i);  
    delay (RobotSpeed);  
}  
For( size_t i {73}; i<=100;++i){  
    Servo1.write(i);  
    Delay (RobotSpeed);  
}  
}
```

Robotska ruka se pokreće pritiskom na gumb. Ima dvije pozicije, pozicija start i pozicija grabljenje. Pozicija start je pozicija u kojoj robot miruje. Pozicija grabljenje je rad robota u kojem uzima praznu bočicu iz gravitacijske punilice i prenosi ju u rotacijsku punilicu.,

### **Kod za rotacijsku punilicu:**

```
#include <Stepper.h>
```

```
const int stepsPerRevolution = 200;
```

```

#define pwmA 3
#define pwmB 11
#define brakeA 9
#define brakeB 8
#define dirA 12
#define dirB 13

Stepper myStepper = Stepper(stepsPerRevolution, dirA, dirB);

// Pin for the button
const int buttonPin = 7;

void setup() {
  pinMode(pwmA, OUTPUT);
  pinMode(pwmB, OUTPUT);
  pinMode(brakeA, OUTPUT);
  pinMode(brakeB, OUTPUT);
  digitalWrite(pwmA, HIGH);
  digitalWrite(pwmB, HIGH);
  digitalWrite(brakeA, LOW);
  digitalWrite(brakeB, LOW);
  myStepper.setSpeed(30);
  Serial.begin(9600);

  // Set the button pin as input
  pinMode(buttonPin, INPUT_PULLUP);
}

```



```

void loop() {
  Serial.println(digitalRead(buttonPin));
  // Check if the button is pressed
  if (digitalRead(buttonPin) == LOW) {
    delay(13000);
    // Turn the stepper motor
    myStepper.step(50);
    delay(3000);
    myStepper.step(50);
    delay(3000);
    myStepper.step(50);

  }
}

```

**Kod za pokretnu traku:**

```

#include <Servo.h>

Servo myservo;
Servo myservo1;
const int traka=4;

int pos = 0;
int pos1 = 0;
int velikaboca=0;
int malaboca=0;

```

```

int zaustavljanje=A0;
int malisenzor=A1;
int velikisenzor=A2;
int sensorValue1=0;
int sensorValue2=0;
int sensorValue3=0;

void setup() {
  Serial.begin(9600);
  myservo.attach(2);
  myservo1.attach(3);
  pinMode(zaustavljanje,INPUT_PULLUP);
  pinMode(malisenzor,INPUT_PULLUP);
  pinMode(velikisenzor,INPUT_PULLUP);
  pinMode(traka, OUTPUT);
}

void loop() {
  digitalWrite(traka,HIGH);

  sensorValue1=analogRead(malisenzor);
  sensorValue2=analogRead(velikisenzor);
  sensorValue3=analogRead(zaustavljanje);

  if(sensorValue1<400&&sensorValue2>400){
    malaboca=HIGH;
  }
}

```

```
}  
  
if(sensorValue2<400){  
    velikaboca=HIGH;  
}
```

```
Serial.println(sensorValue1);  
//Serial.println(sensorValue2);  
//Serial.println(sensorValue3);
```

```
if(sensorValue3<350&&velikaboca==HIGH){  
    digitalWrite(traka,LOW);  
    myservo1.write(180);  
    delay(2200);  
    myservo1.write(90);  
    delay(15);  
    myservo1.write(0);  
    delay(2200);  
    myservo1.write(90);  
    velikaboca=LOW;  
    malaboca=LOW;  
}
```

```
if(sensorValue3<350&&malaboca==HIGH){  
    digitalWrite(traka,LOW);
```

```
myservo.write(180);  
delay(1800);  
myservo.write(90);  
delay(15);  
myservo.write(0);  
delay(1800);  
myservo.write(90);  
  malaboca=LOW;  
}  
}
```