



IRIM - Institut za razvoj
i inovativnost mladih

Generacija
NOW
2021



Robotsko vozilo s daljinskim upravljanjem

Tehnički opis rada

Đurđevac

svibanj 2021., Đurđevac

SADRŽAJ

| | |
|---|--------------|
| 1. PROJEKTNI TIM..... | 3 |
| Mentor..... | 3 |
| Učenici..... | 4 |
| Informacije o školi..... | 5 |
| 2. KOMPONENTE KORIŠTENE ZA IZRADU..... | 5 |
| Arduino Nano LAFVIN..... | 5 |
| Robotska šasija..... | 6 |
| HC-12..... | 7 |
| Li-Po baterije..... | 8 |
| Upravljač..... | 9 |
| Upravljač istosmjernih motora | 10 |
| 3. Sheme spajanja..... | 11 |
| Robotsko vozilo----- | 11 |
| Daljinski upravljač-- | 12 |
| 4. Faze izrade..... | 13 |
| 5. Programski kod..... | 14 |
| Robotsko vozilo----- | 14-18 |
| Daljinski upravljač-- | 19 |
| 6. Fotografije gotovog uređaja..... | 20 |
| Robotsko vozilo----- | 20 |
| Daljinski upravljač-- | 20 |
| 7. ODRŽAVANJE..... | 21 |
| 8. FOTO ALBUM RADA NA PROJEKTU..... | 22-25 |

1. PROJEKтни TIM

Mentor :

Željko Brček mr.ing.el.



Učenci:

Borna Carek



Edi Lazar



Paola Posavec



Tomislav Čukljaš



Informacije o školi:

Strukovna škola Đurđevac

Dr. Ivana Kranjčeva 5

<http://ss-strukovna-djurdjevac.skole.hr/>

2. KOMPONENTE KORIŠTENE ZA IZRADU

LAFVIN Nano CH340

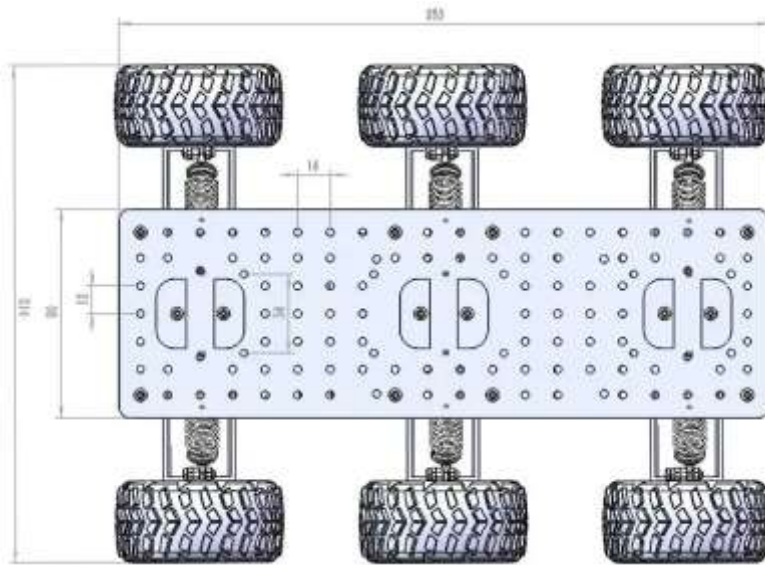


LAFVIN Nano CH340 je mala, cijela ploča je temeljena na ATmega328. Najmanja pločica, može se koristiti na razvojnoj ploči. LAFVIN Nano je komatibilan sa Windows, MacOS i Linux operacijskim sustavima. Sličan Arduino UNO-u, ali je manji i kompatibilniji. Vrlo pristupačan cijenom i vrlo jednostavan za korištenje.

Specifikacije LAFVIN Nano-a:

- 5V operacijsko napajanje
- 8 analognih portova: A0 – A7
- 14 digitalnih I/O portova: TX, RX, D2 – D13
- 6 PWM portova: D3, D5, D6, D9, D10, D11
- 1 par TTL serijskih porova za primanje i slanje signala: RX / TX
- USB podrška i napajanje
- Podrška za vanjsko napajanje: 5V – 12V DC
- Dodatno napajanje: 9V baterija
- Podržava ISP preuzimanja

Robotska šasija



Robotska šasija (6WD platforma) je cijelo tijelo je sastavljeno od 2mm aluminijske legure i 6 komada DC motora visoke brzine(17000RPM).

Koristi 6 hidrauličkih opruga da sigurno poveže motor za gornje tijelo.

Specifikacije:

- Veličina: 28x21x11.5 cm
- Težina: 1.4kg
- Maksimalni teret: 6kg
- Radni napon: 12V
- Označena struja: 350mA*6
- Brzina motora: 17000RPM
- Boja: Crna

Đurđevac

HC-12 bežični modul



HC-12 bežični modul za komunikaciju je nova generacija od više kanala stavljenih bežičnih modula za prenošenje podataka. Radio frekvencija od 433.4 – 473.0MHz, može postaviti kanal za razmjenu informacija. Maksimalna jačina koju model može dati je 100mW. Maksimalna duljina za komunikaciju je oko 500m. Dimenzije modula su 27.4mm*13.2mm*4mm (uključujući mjesto za antenu, ali ne i antenu). Modul ima PBC mjesto za antenu ANT1, korisnici mogu koristiti koaksijalni kabel.

Specifikacije:

- Frekvencija rada: 433.4MHz do 473.0MHz
- Napon: 3.2V to 5.5VDC
- Komunikacijska metraža: 1000m u otvorenom prostoru
- Serijski brzina prijenosa : 1.2Kbps do 115.2Kbps
- Osjetljivost: -117dBm do -100dBm
- Izlazna snage: -1dBm do 20dBm
- Protokoli sučelja: UART/TTL
- Temperatura rada: -40°C do +85°C

Li-Po baterija

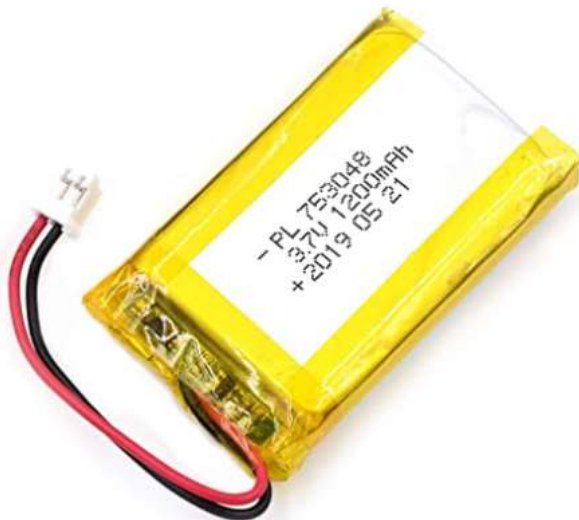
Eachine 1500mAh 3S 11.1V 25C



Eachine 1500mAh 3S 11.1V 25C Li-Po baterija:

- Dimenzije su 88mm*18mm*35mm
- Težina je 119g
- Napon je 11.1V

3.7 V 1200 mAh 4.44Wh

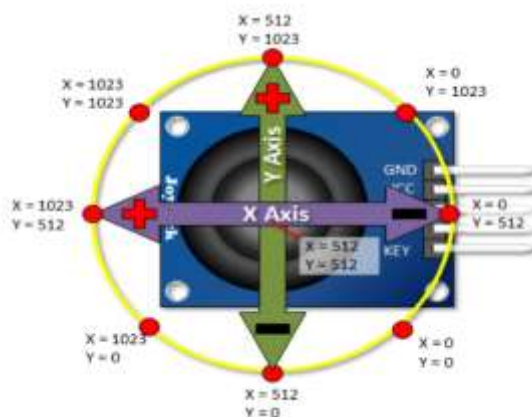
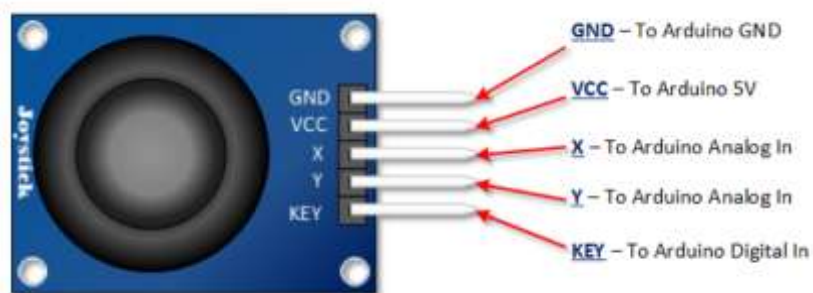


- 3.7V 1200mAh 753048 Lipo Battery Rechargeable Lithium Polymer ion Battery Pack with JST Connector

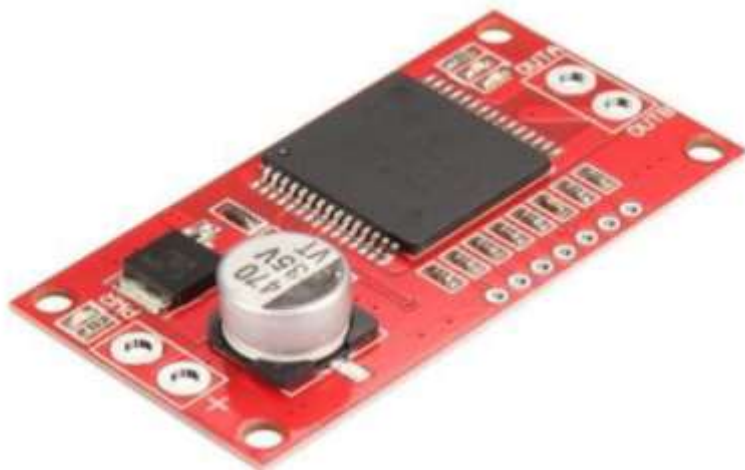
KY-023 Arduino upravljáč



KY-023 Arduino upravljáč koristi dvoosni potenciometar za upravljanje X i Y osi. Kada se pritisne prema dolje, aktivira prekidač. Na temelju upravljáča PlayStation2 kontrolera, koristi se za upravljanje širokim spektrom projekata od RC vozila do LED dioda u boji. Radni napon je 3.3V do 5V. Dimenzije 2.6 cm* 3.4 cm.



Upravljač istosmjernih motora VNH2SP30



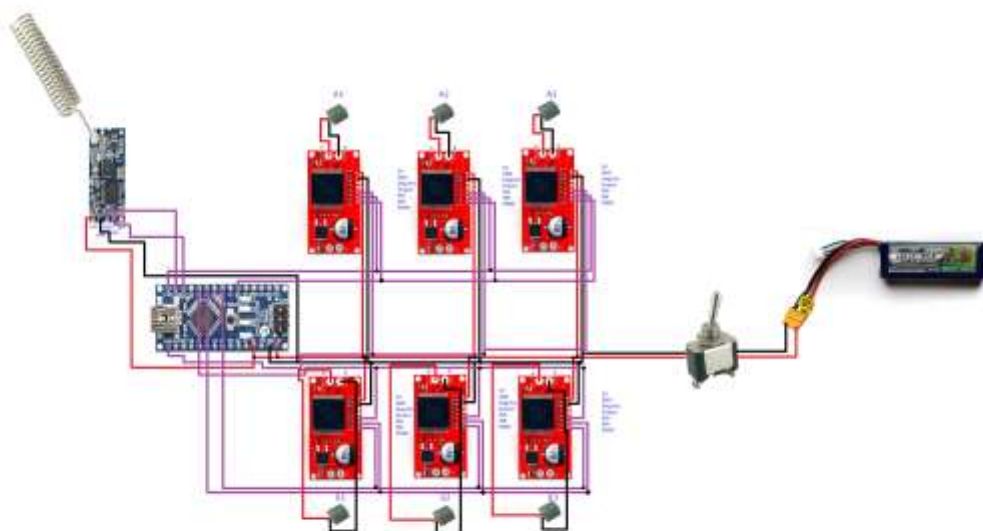
Visokonaponski upravljač istosmjernih motora baziran na VNH2SP30 čipu. Dizajniran za aplikacije za visokonaponske motore istosmjerne struje s najvišom jačinom struje 30A i stalnom strujom od 15A. Ovaj model je lagano koristiti s arduinom ili nekim drugim mikroupravljačem. Ima mogućnost toplinskog isključivanja za zaštitu od pregrijavanja, također ima senzore struje i zaštitu od prenapona.

Specifikacije:

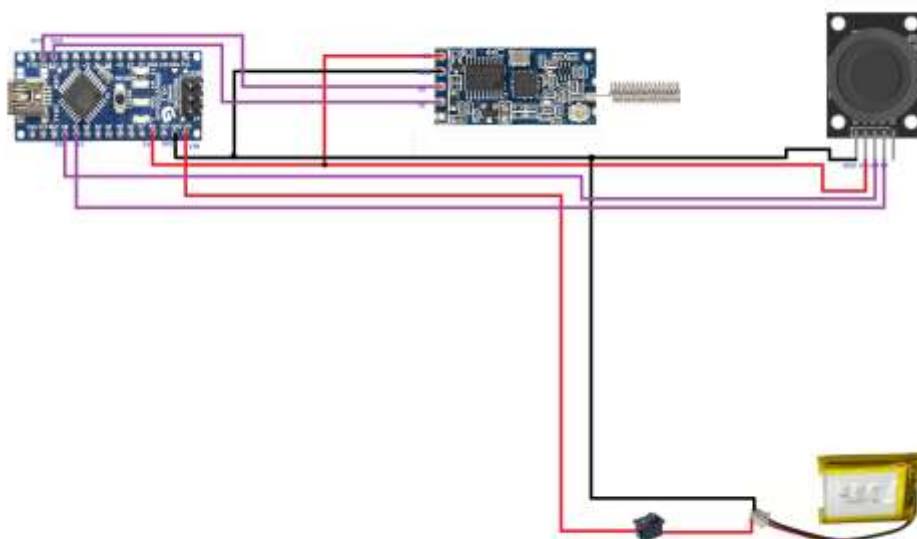
- Maksimalna struja: 30A
- Praktična stalna struja: 15A
- Maksimalni napon: 16V
- Maksimalna PWM frekvencija: 20kHz
- Toplinsko isključivanje
- Podnaponsko i prenaponsko isključivanje
- Senzori struje omogućeni za Arduino analogne pinove
- Masa: 25g
- Veličina: 53 x 22mm

3. SCHEM SPAJANJA

Robotsko vozilo

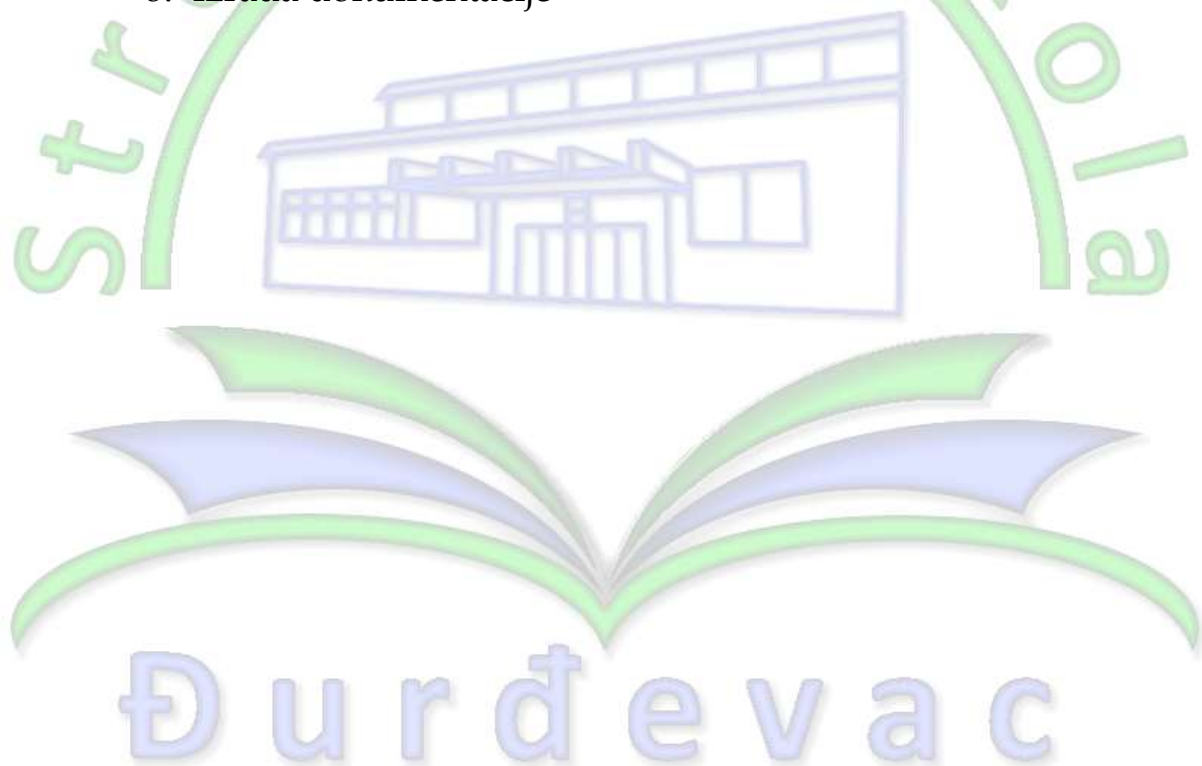


Daljinski upravljač



4. FAZE IZRADE

1. Sastavljanje robotske šasije
2. Spajanje komponenti na razvojnim pločicama
3. Razvoj i testiranje programskog koda
4. Izrada 3d modela kućišta za elektroniku robotskog vozila i daljinskog upravljača
5. Ispis 3d modela na 3d pisaču
6. Montaža i spajanje komponenti u kućišta
7. Testiranje funkcionalnosti
8. Izrada dokumentacije



5. PROGRAMSKI KOD

Robotsko vozilo

```
//Prijamnik
#include <SoftwareSerial.h> //uključuje vanjsku biblioteku "SoftwareSerial"
#include <Wire.h> //uključuje vanjsku biblioteku "Wire", u prijevodu "žica"
// Definicija naziva pinova za spajanje motora
//Pinovi za motore s lijeve strane (A1,A2,A3)
#define MOTOR_A1_INA_PIN 13 //definira motor A1 INA na pin 13
#define MOTOR_A1_INB_PIN 12 //definira motor A1 INB na pin 12
#define MOTOR_A1_PW 3 // definira motor A1 na PW pin 3
//Pinovi za motore s desne strane (B1,B2,B3)
#define MOTOR_B1_INA_PIN 8 //definira motor B1 INA na pin 8
#define MOTOR_B1_INB_PIN 7 //definira motor B1 INB na pin 7
#define MOTOR_B1_PW 5 // definira motor B1 na PW pin 5

SoftwareSerial HC12(10, 11); // Postavljanje pinova HC-12 wifi modula, 10=TX Pin, 11=HC-12 RX
Pin

int ld,nn, motorA_brzina, motorB_brzina; //varijable za upravljanje smjerom kretanja i brzinom
kretanja

String input; //string varijabla za prijem komandi s daljinskog upravljača
int niskoVezano; // pomoćna varijabla za čitanje stringa
int visokoVezano; // pomoćna varijabla za čitanje stringa
const char granicnik = ','; // pomoćna varijabla za čitanje stringa

void setup() { //glavna naredba koja se pokreće pri pokretanju programa

//Definica načina rada pinova za motore
pinMode(MOTOR_A1_INA_PIN, OUTPUT); //postavlja motor A1 INA kao izlaz
pinMode(MOTOR_A1_INB_PIN, OUTPUT); //postavlja motor A1 INB kao izlaz
pinMode(MOTOR_A1_PW, OUTPUT); //postavlja motor A1 PW kao izlaz
pinMode(MOTOR_B1_INA_PIN, OUTPUT); //postavlja motor B1 INA kao izlaz
pinMode(MOTOR_B1_INB_PIN, OUTPUT); //postavlja motor B1 INB kao izlaz
pinMode(MOTOR_B1_PW, OUTPUT); //postavlja motor B1 PW kao izlaz

Serial.begin(9600); //postavljanje brzine serijskog prijenosa
HC12.begin(9600); //postavljanje brzine serijske komunikacije wifi modula

}

void loop() { //glavna naredba koja se pokreće nakon "void setup" i traje do isključivanja koda

if(HC12.available()) //naredba if, služi na neki uvjet
{
input = HC12.readStringUntil('\n'); //Očitavanje prijema sa wifi modula
if (input.length() > 0) //naredba if, služi za neki uvjet
{
```

```

Serial.println(input); // upis primljenih komandi u input

visokoVezano = input.indexOf(granicnik, niskoVezano+1); //postavlja "visokoVezano" u
unos "indexOf"
Serial.print("visokovezano = "); //ispisuje podatke u serijal port
Serial.println(visokoVezano); //upis primljenih komandi u visokoVezano
ld = input.substring(niskoVezano, visokoVezano).toInt(); // komanda za kretanje lijevo-
desno
nn = input.substring(visokoVezano+1).toInt(); // komanda za kretanje naprijed-natrag

// poseban prikaz vrijednosti x i napona na palici
Serial.print("ld = "); //ispisuje podatke u serijal port
Serial.print(ld); //ispisuje podatke u serijal port
Serial.print(" ldX_Voltage = "); //ispisuje podatke u serijal port
Serial.println(float(ld)*5.0/1024.0); //upis primljenih podataka u ID

Serial.print("nn = "); //ispisuje podatke u serijal port
Serial.print(nn); //ispisuje podatke u serijal port
Serial.print(" nnY_Voltage = "); //ispisuje podatke u serijal port
Serial.println(float(nn)*5.0/1024.0); //upis primljenih podataka u "nn"

//delay(10); //pričekaj 10 milisekundi
}
//Kretanje robota
Kretanje(); //Funkcija za odabir smjera i brzine kretanja
}
}

void lijevo() //funkcija lijevo
{
motorA_brzina=50; // brzina motora s lijeve strane (A1,A2,A3)
motorB_brzina=100; // brzina motora s lijeve strane (A1,A2,A3)
analogWrite(MOTOR_A1_PW, motorA_brzina); // upisuje analogni podatak u pin na koji je
definiran motor A1 PW
analogWrite(MOTOR_B1_PW, motorB_brzina); // upisuje analogni podatak u pin na koji je
definiran motor B1 PW
digitalWrite(MOTOR_A1_INA_PIN, HIGH); // postavlja pin u HIGH (1)
digitalWrite(MOTOR_A1_INB_PIN, LOW); // postavlja pin u LOW (0)
digitalWrite(MOTOR_B1_INA_PIN, HIGH); // postavlja pin u HIGH (1)
digitalWrite(MOTOR_B1_INB_PIN, LOW); // postavlja pin u LOW(0)
Serial.println("Lijevo"); // upis primljenih podataka "Lijevo"
}

void desno() //funkcija desno
{
motorA_brzina=100; //brzina motora s desne strane (A4, A5, A6)
motorB_brzina=50; //brzina motora s desne strane (A4, A5, A6)
analogWrite(MOTOR_A1_PW, motorA_brzina); // upisuje analogni podatak u pin na koji je
definiran motor A1 PW
analogWrite(MOTOR_B1_PW, motorB_brzina); // upisuje analogni podatak u pin na koji je
definiran motor B1 PW

```

```

digitalWrite(MOTOR_A1_INA_PIN, LOW); //postavlja pin u LOW (0)
digitalWrite(MOTOR_A1_INB_PIN, HIGH); //postavlja pin u HIGH (1)
digitalWrite(MOTOR_B1_INA_PIN, LOW); //postavlja pin u LOW (0)
digitalWrite(MOTOR_B1_INB_PIN, HIGH); //postavlja pin u HIGH (1)
Serial.println("Desno"); //upis primljenih podataka "Desno"
}

```

```

void stop() //funkcija stop

```

```

{
motorA_brzina=0; //brzina motora (A1,A2,A3)
motorB_brzina=0; //brzina motora (A1,A2,A3)
analogWrite(MOTOR_A1_PW, motorA_brzina); //upisuje analogni podatak u pin na koji je
definiran motor A1 PW
analogWrite(MOTOR_B1_PW, motorB_brzina); //upisuje analogni podatak u pin na koji je
definiran motor B1 PW
digitalWrite(MOTOR_A1_INA_PIN, LOW); //postavlja pin u LOW (0)
digitalWrite(MOTOR_A1_INB_PIN, LOW); //postavlja pin u LOW (0)
digitalWrite(MOTOR_B1_INA_PIN, LOW); //postavlja pin u LOW (0)
digitalWrite(MOTOR_B1_INB_PIN, LOW); //postavlja pin u LOW (0)
Serial.println("Stop"); //upis primljenih podataka "stop"
}

```

```

void naprijedLijevo() //funkcija naprijedLijevo

```

```

{
motorA_brzina=25; //brzina motora (A1,A2,A3)
motorB_brzina=100; //brzina motora (A1,A2,A3)
analogWrite(MOTOR_A1_PW, motorA_brzina); //upisuje analogni podatak u pin na koji je
definiran motor A1 PW
analogWrite(MOTOR_B1_PW, motorB_brzina); //upisuje analogni podatak u pin na koji je
definiran motor B1 PW
digitalWrite(MOTOR_A1_INA_PIN, LOW); //postavlja pin u LOW (0)
digitalWrite(MOTOR_A1_INB_PIN, HIGH); //postavlja pin u HIGH (1)
digitalWrite(MOTOR_B1_INA_PIN, HIGH); //postavlja pin u HIGH (1)
digitalWrite(MOTOR_B1_INB_PIN, LOW); //postavlja pin u LOW (0)
Serial.println("Naprijed lijevo"); //upis primljenih podataka "Naprijed lijevo"
}

```

```

void naprijedDesno() //funkcija naprijedDesno

```

```

{
motorA_brzina=100; //brzina motora (A4,A5,A6)
motorB_brzina=25; //brzina motora (A4,A5,A6)
analogWrite(MOTOR_A1_PW, motorA_brzina); //upisuje analogni podatak u pin na koji je
definiran motor A1 PW
analogWrite(MOTOR_B1_PW, motorB_brzina); //upisuje analogni podatak u pin na koji je
definiran motor B1 PW
digitalWrite(MOTOR_A1_INA_PIN, LOW); //postavlja pin u LOW (0)
digitalWrite(MOTOR_A1_INB_PIN, HIGH); //postavlja pin u HIGH (1)
digitalWrite(MOTOR_B1_INA_PIN, HIGH); //postavlja pin u HIGH (1)
digitalWrite(MOTOR_B1_INB_PIN, LOW); //postavlja pin u LOW (0)
Serial.println("Naprijed desno"); //upis primljenih podataka "Naprijed desno"
}

```



```

}

void natrag() //funkcija natrag
{
motorA_brzina=50; //brzina motora (A1,A2,A3)
motorB_brzina=50; //brzina motora (A4,A5,A6)
analogWrite(MOTOR_A1_PW, motorA_brzina); //upisuje analogni podatak u pin na koji je
definiran motor A1 PW
analogWrite(MOTOR_B1_PW, motorB_brzina); //upisuje analogni podatak u pin na koji je
definiran motor B1 PW
digitalWrite(MOTOR_A1_INA_PIN, HIGH); //postavlja pin u HIGH (1)
digitalWrite(MOTOR_A1_INB_PIN, LOW); //postavlja pin u LOW (0)
digitalWrite(MOTOR_B1_INA_PIN, LOW); //postavlja pin u LOW (0)
digitalWrite(MOTOR_B1_INB_PIN, HIGH); //postavlja pin u HIGH (1)
Serial.println("Natrag"); //upis primljenih podataka "Natrag"
}

void naprijed() //funkcija naprijed
{
motorA_brzina=50; //brzina motora (A1,A2,A3)
motorB_brzina=50; //brzina motora (A4,A5,A6)
analogWrite(MOTOR_A1_PW, motorA_brzina); //upisuje analogni podatak u pin na koji je
definiran motor A1 PW
analogWrite(MOTOR_B1_PW, motorB_brzina); //upisuje analogni podatak u pin na koji je
definiran motor B1 PW
digitalWrite(MOTOR_A1_INA_PIN, LOW); //postavlja pin u LOW (0)
digitalWrite(MOTOR_A1_INB_PIN, HIGH); //postavlja pin u HIGH (1)
digitalWrite(MOTOR_B1_INA_PIN, HIGH); //postavlja pin u HIGH (1)
digitalWrite(MOTOR_B1_INB_PIN, LOW); //postavlja pin u LOW (0)
Serial.println("Naprijed"); //upis primljenih podataka "Naprijed"
}

void Kretanje() //funkcija Kretanje
{
if ((nn <= 100) && (ld > 100) && (ld < 700)) //naredba if, služi na neki uvjet
{
lijevo(); //poziva se funkcija lijevo
}

else if ((nn >= 700) && (ld > 100) && (ld < 700)) //nareba else if, služi za neki uvjet
{
desno(); //poziva se funkcija lijevo
}

else if ((ld <= 100) && (nn > 100) && (nn < 700)) //nareba else if, služi za neki uvjet
{
natrag();
}
}

```

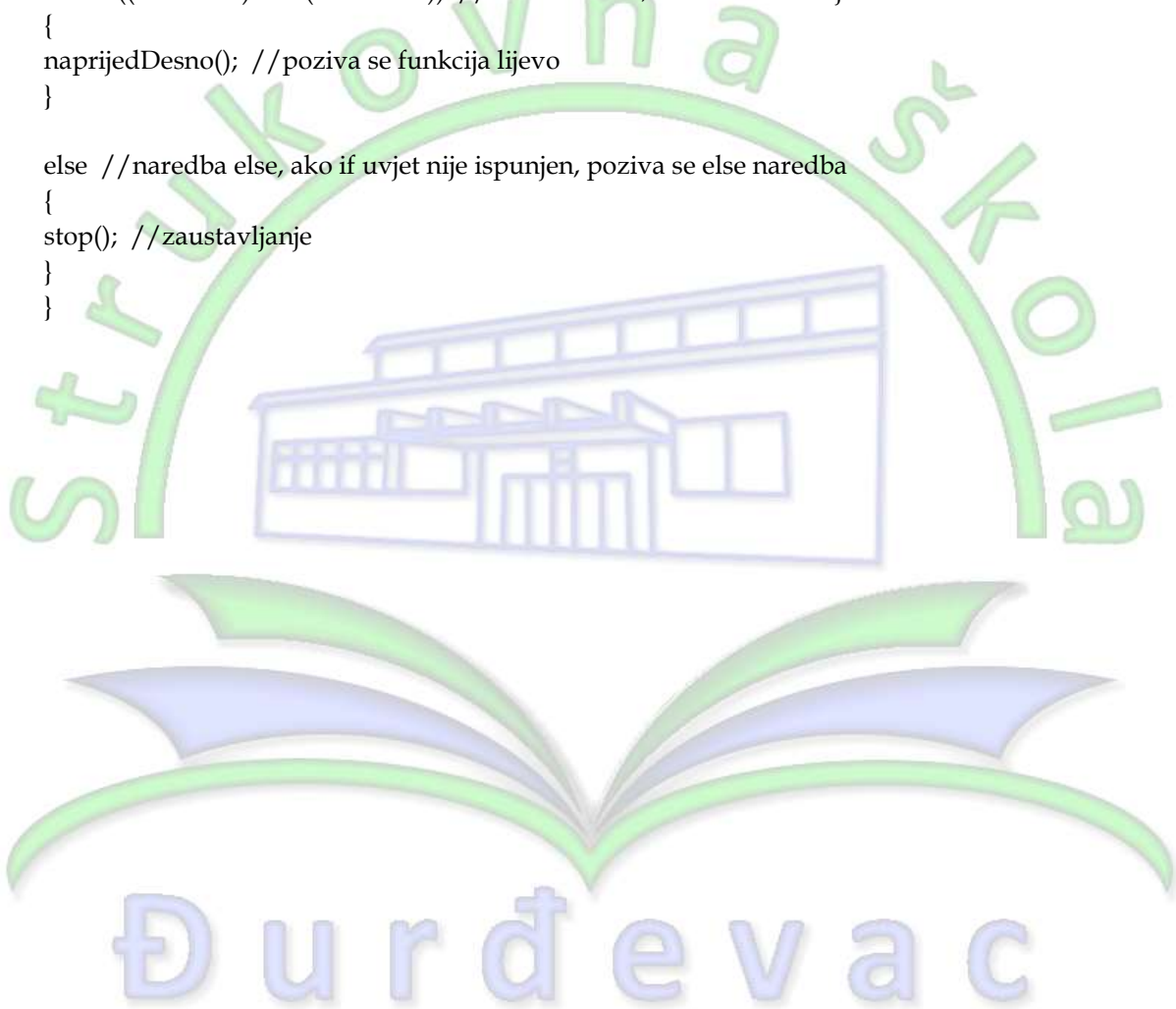


```
else if ((ld >= 700) && (nn > 100) && (nn < 700)) //nareba else if, služi za neki uvjet
{
naprijed(); //poziva se funkcija lijevo
}
```

```
else if ((ld >= 700) && (nn <= 100)) //nareba else if, služi za neki uvjet
{
naprijedLijevo(); //poziva se funkcija lijevo
}
```

```
else if ((ld >= 700) && (nn >= 700)) //nareba else if, služi za neki uvjet
{
naprijedDesno(); //poziva se funkcija lijevo
}
```

```
else //naredba else, ako if uvjet nije ispunjen, poziva se else naredba
{
stop(); //zaustavljanje
}
}
```



Daljinski upravljač

```
//Odašiljač
#include <SoftwareSerial.h> //uključuje vanjsku biblioteku "SoftwareSerial"
#include <Wire.h> //uključuje vanjsku biblioteku "Wire", u prijevodu "žica"
SoftwareSerial HC12(10, 11); // Postavljanje pinova HC-12 wifi modula, 10=TX Pin, 11=HC-12
RX Pin

int ld=0; // Varijabla za x os palice za upravljanje - lijevo/desno kretanje
int nn=0; // Varijabla za y os palice za upravljanje - naprijed/natrag kretanje
int skl=0; // Varijabla za stanje sklopke palice za upravljanje - promjena brzine kretanja
brže/sporije

void setup() //glavna funkcija setup, pokreće se kada se program pokrene
{
  HC12.begin(9600); //postavljanje brzine serijskog prijenosa između arduinoa i wifi kartice
  Serial.begin(9600); // početak prijenosa podataka prijenosa između arduinoa i wifi kartice
  // postavljanje analognih pinova na ulazne
  pinMode(A0, INPUT); //x -ky-23, postavlja pin A0 u INPUT, ulaz
  pinMode(A1, INPUT); //y- ky-023, postavlja pin A1 u INPUT, ulaz
}
void loop() //glavna funkcija loop, pokreće se nakon funkcije setup i traje do kada se program
ne isključi
{

  ld = analogRead(A0); // očitavanje x osi Ky-023
  nn = analogRead(A1); // očitavanje y osi Ky-023

  // kreiranje podataka za prijenos

  HC12.print(ld); //ispisuje podatke na HC12 Serial
  HC12.print(","); //ispisuje podatke na HC12 Serial
  HC12.print(nn); //ispisuje podatke na HC12 Serial
  HC12.println(""); //ispisuje podatke na HC12 Serial

  delay(1000); //pričekaj 1 sekundu (1000 milisekundi)
}
```

6. Fotografije gotovog uređaja Robotsko vozilo



Daljinski upravljač



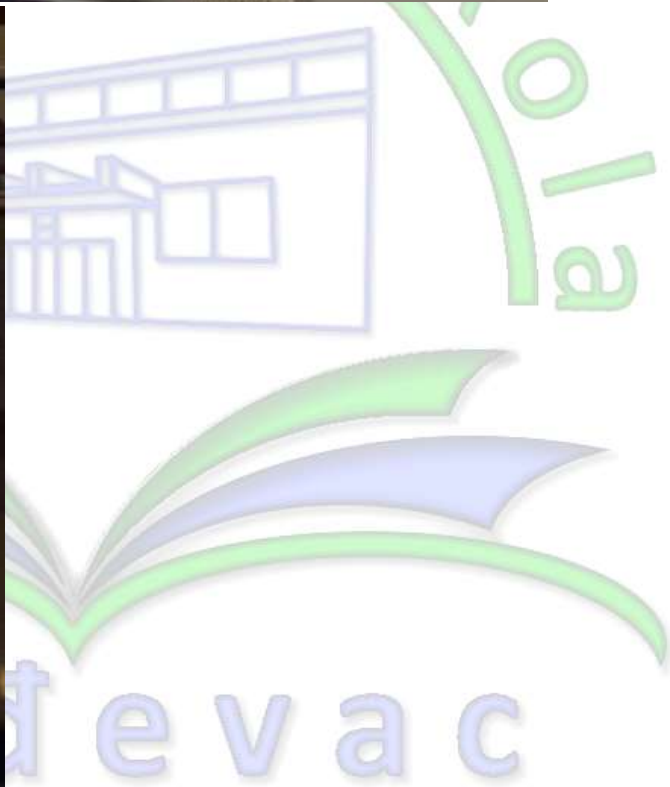
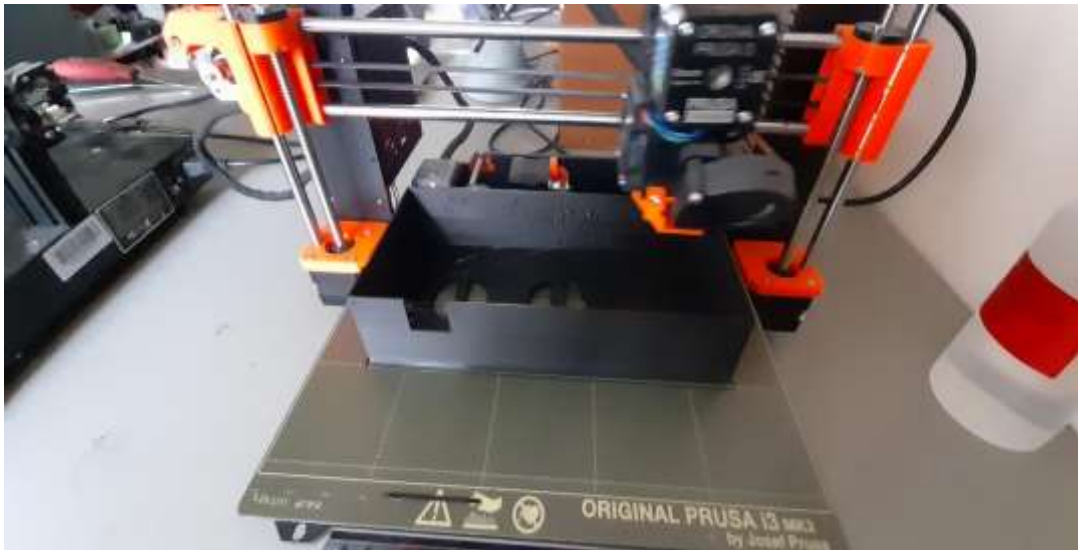
Đurđevac

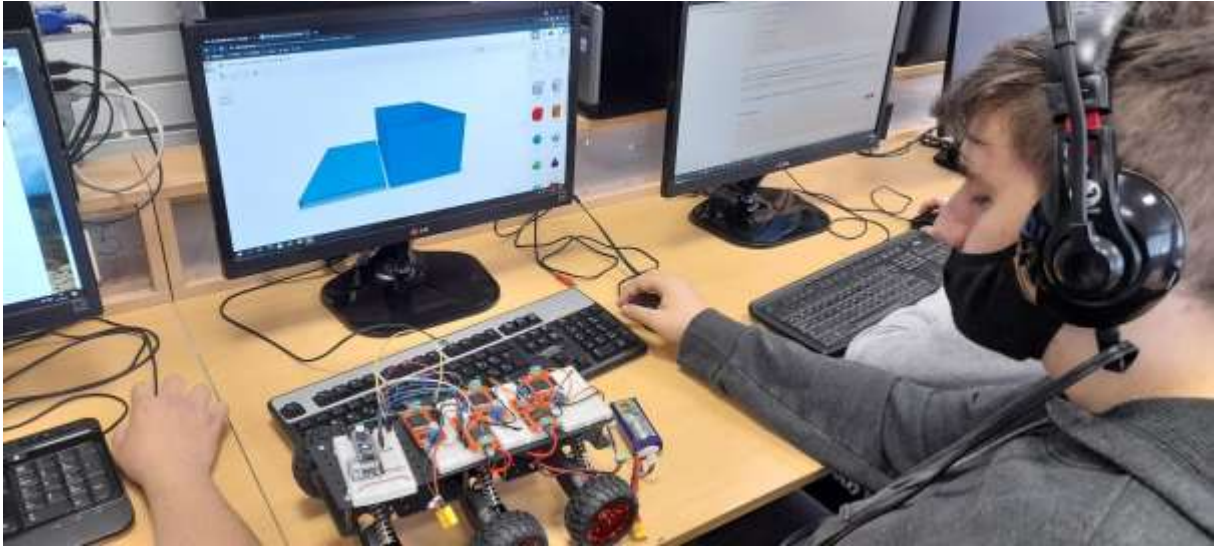
7. ODRŽAVANJE

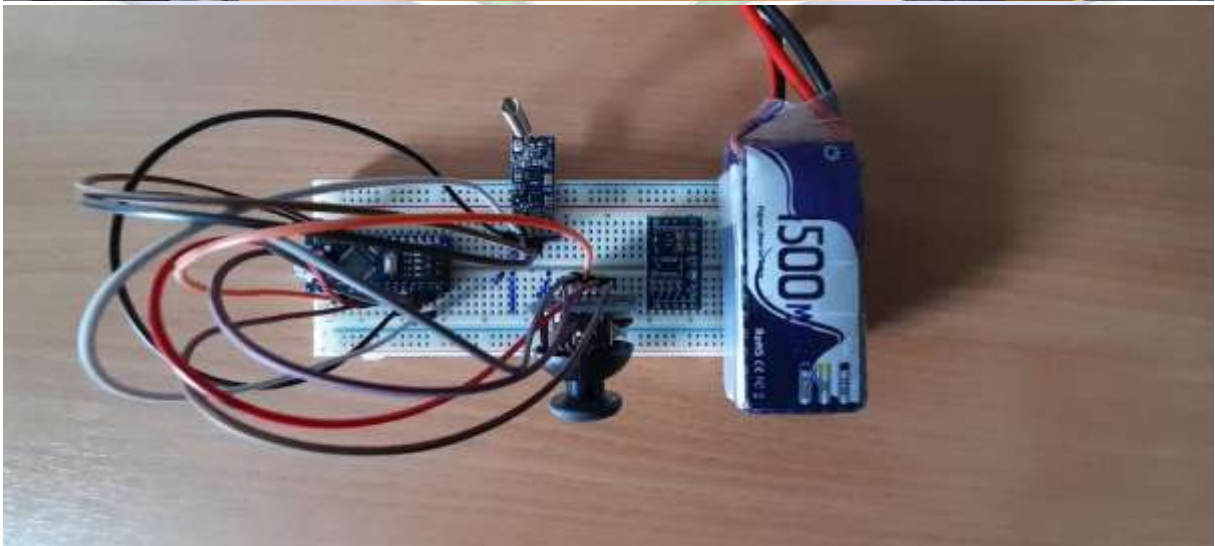
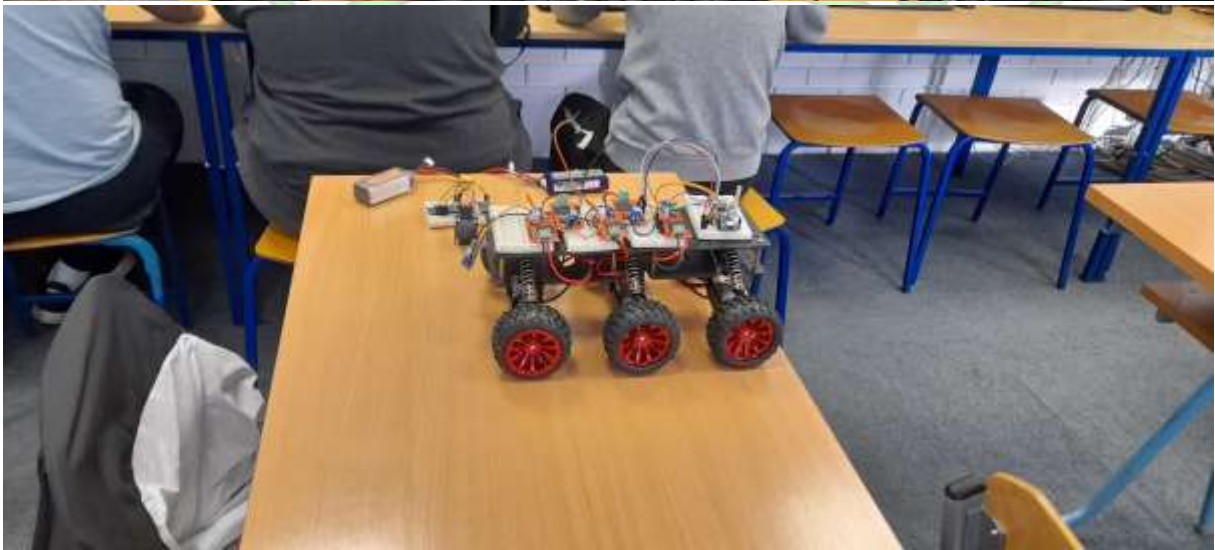
Sve elemente i dijelove auta potrebno je redovito kontrolirati i održavati. U slučaju oštećenja ili nekog njegovog dijela (kućišta, konstrukcije, podloge ili drugog elementa opreme), popravak se može izvršiti zamjenom oštećenog elementa isključivo istim elementom.



8. FOTO ALBUM RADA NA PROJEKTU









Đurđevac