

# Mobilni robot upravlján gestama ruke

---

**Vuković, Matej**

**Master's thesis / Diplomski rad**

**2020**

*Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj:* **Josip Juraj Strossmayer University of Osijek, Faculty of Electrical Engineering, Computer Science and Information Technology Osijek / Sveučilište Josipa Jurja Strossmayera u Osijeku, Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek**

*Permanent link / Trajna poveznica:* <https://um.nsk.hr/um:nbn:hr:200:832514>

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

*Download date / Datum preuzimanja:* **2025-03-12**

*Repository / Repozitorij:*

[Faculty of Electrical Engineering, Computer Science and Information Technology Osijek](#)



**SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU  
FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I  
INFORMACIJSKIH TEHNOLOGIJA**

**Sveučilišni studij**

**MOBILNI ROBOT UPRAVLJAN GESTAMA RUKE**

**Diplomski rad**

**Matej Vuković**

**Osijek, 2020.**

**FERIT**FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA  
I INFORMACIJSKIH TEHNOLOGIJA OSIJEK

Obrazac D1: Obrazac za imenovanje Povjerenstva za diplomski ispit

Osijek, 27.09.2020.

Odboru za završne i diplomske ispite

**Imenovanje Povjerenstva za diplomski ispit**

<b>Ime i prezime studenta:</b>	Matej Vuković
<b>Studij, smjer:</b>	Diplomski sveučilišni studij Računarstvo
<b>Mat. br. studenta, godina upisa:</b>	D-909R, 30.09.2019.
<b>OIB studenta:</b>	78388666182
<b>Mentor:</b>	Izv. prof. dr. sc. Ivan Aleksi
<b>Sumentor:</b>	
<b>Sumentor iz tvrtke:</b>	
<b>Predsjednik Povjerenstva:</b>	Prof.dr.sc. Goran Martinović
<b>Član Povjerenstva 1:</b>	Izv. prof. dr. sc. Ivan Aleksi
<b>Član Povjerenstva 2:</b>	Filip Sušac
<b>Naslov diplomskog rada:</b>	Mobilni robot upravljani gestama ruke
<b>Znanstvena grana rada:</b>	<b>Arhitektura računalnih sustava (zn. polje računarstvo)</b>
<b>Zadatak diplomskog rada:</b>	Tema rezervirana za studenta Matej Vuković U ovom radu treba razviti aplikaciju za daljinsko upravljanje mobilnog robota na temelju pokreta ruke primjenom akcelerometra.
<b>Prijedlog ocjene pismenog dijela ispita (diplomskog rada):</b>	Izvrstan (5)
<b>Kratko obrazloženje ocjene prema Kriterijima za ocjenjivanje završnih i diplomskih radova:</b>	Primjena znanja stečenih na fakultetu: 3 bod/boda Postignuti rezultati u odnosu na složenost zadatka: 3 bod/boda Jasnoća pismenog izražavanja: 3 bod/boda Razina samostalnosti: 3 razina
<b>Datum prijedloga ocjene mentora:</b>	27.09.2020.
Potpis mentora za predaju konačne verzije rada u Studentsku službu pri završetku studija:	Potpis:
	Datum:

**FERIT**FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA  
I INFORMACIJSKIH TEHNOLOGIJA OSIJEK**IZJAVA O ORIGINALNOSTI RADA**

Osijek, 01.10.2020.

**Ime i prezime studenta:**

Matej Vuković

**Studij:**

Diplomski sveučilišni studij Računarstvo

**Mat. br. studenta, godina upisa:**

D-909R, 30.09.2019.

**Turnitin podudaranje [%]:**

4

Ovom izjavom izjavljujem da je rad pod nazivom: **Mobilni robot upravlján gestama ruke**

izrađen pod vodstvom mentora Izv. prof. dr. sc. Ivan Aleksi

i sumentora

moj vlastiti rad i prema mom najboljem znanju ne sadrži prethodno objavljene ili neobjavljene pisane materijale drugih osoba, osim onih koji su izričito priznati navođenjem literature i drugih izvora informacija. Izjavljujem da je intelektualni sadržaj navedenog rada proizvod mog vlastitog rada, osim u onom dijelu za koji mi je bila potrebna pomoć mentora, sumentora i drugih osoba, a što je izričito navedeno u radu.

Potpis studenta:

# SADRŽAJ

1. UVOD .....	1
1.1. Zadatak diplomskog rada.....	1
2. MOBILNI ROBOT .....	2
3. KORIŠTENI ALATI I TEHNOLOGIJE .....	4
3.1. Arduino razvojno okruženje .....	4
3.1.1. Arduino UNO .....	4
3.1.2. Arduino Mega 2560 .....	6
3.2. Akcelerometar .....	7
3.3. Senzor savijanja .....	7
3.4. Istosmjerni motori i kotači.....	8
3.5. L293d upravljač motora.....	9
3.6. Ultrazvučni senzor .....	10
3.7. RF modul .....	11
3.7.1. RF odašiljač.....	11
3.7.2. RF prijemnik .....	12
3.8. LE diode i otpornici .....	13
4. TEHNIČKA REALIZACIJA .....	15
4.1. Izrada mobilnog robota.....	15
4.2. Izrada upravljačke rukavice .....	18
4.3. Programsko rješenje .....	20
4.3.1. Programski kod upravljačke rukavice .....	20
4.3.2. Programski kod mobilnog robota .....	22
4.4. Rad sustava .....	22
5. ZAKLJUČAK .....	24
LITERATURA .....	25
SAŽETAK.....	26
ABSTRACT .....	27
ŽIVOTOPIS .....	28
PRILOG A. Arduino kod upravljačke rukavice.....	29
PRILOG B. Arduino kod mobilnog robota.....	31
OSTALI PRILOZI .....	34

## **1. UVOD**

U ovom diplomskom radu potrebno je izraditi mobilni robot s ugrađenim mikroupravljačkim sustavom. Mobilni robot će biti upravljan gestama ruke putem upravljačke rukavice koja će koristiti akcelerometar i senzor savijanja kao alate za slanje naredbi. Mobilni robot pokretati će četiri istosmjerna motora kojima će upravljati L293d čip. Komunikacija između upravljačke rukavice i mobilnog robota odvijati će se putem RF modula. Oba sklopa biti će bazirana na Arduino razvojnom sustavu.

### **1.1. Zadatak diplomskog rada**

Zadatak diplomskog rada je razviti aplikaciju za daljinsko upravljanje mobilnim robotom na temelju geste ruke primjenom Arduino razvojnog sustava.

## 2. MOBILNI ROBOT

Robot predstavlja automatizirani stroj višestruke namjene koji može obavljati zadaće slične ljudskom djelovanju. Izraz robot nastao je od riječi robota što u prijevodu znači teški rad i prvi puta je spomenut u djelu R.U.R. (engl. *Rossum's Universal Robots*) češkog pisca Karel Čapeka. Najveća primjena robota je u industriji, ponajviše u autoindustriji. Industrijski roboti (Slika 2.1.) sposobni su obavljati razne zadatke s velikom preciznošću kao što je zavarivanje, bojanje ili sastavljanje dijelova automobila. Unatoč svojim sposobnostima industrijski roboti pate od jednog velikog nedostatka, a to je svojstvo kretanja čime im se smanjuje efikasnost [1].



Sl. 2.1. Industrijski robot [2]

Danas se novi razvoj robota uglavnom usmjerava prema umjetnoj inteligenciji, mobilnosti i autonomnosti pa tu na snagu stupaju mobilni roboti (Slika 2.2.). Za razliku od industrijskog robota mobilni robot je robot koji posjeduje svojstvo kretanja u prostoru. Postoje razni oblici kretanja robota kao što su skakanje, plivanje, letenje, hodanje, klizanje i kotrljanje, većina koja ih je inspirirana kretanjama iz prirode. No međutim kretanje robota na kotačima najkorišteniji je i najefikasniji oblik kretanja mobilnih robota i upravo će u ovom radu dati opis izrade jednog takvog mobilnog robota.



Sl. 2.2. Mobilni robot[3]



### 3. KORIŠTENI ALATI I TEHNOLOGIJE

Kroz ovo poglavlje dati će se opis elektroničkih komponenti, programa i alata potrebnih u izradi ovog rada.

#### 3.1. Arduino razvojno okruženje

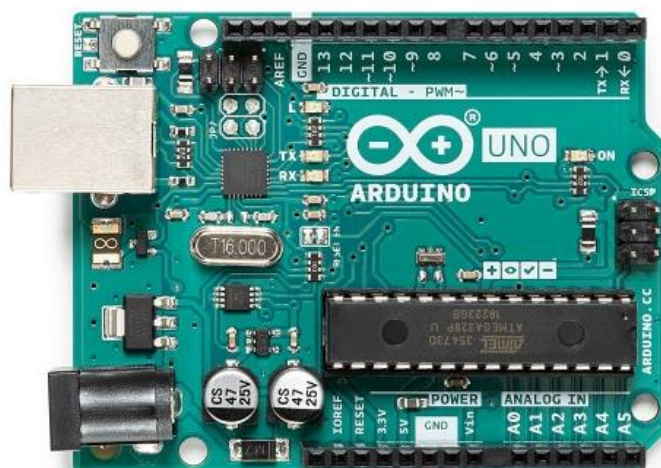
Arduino razvojno okruženje je platforma otvorenog tipa (engl. *open-source*) temeljena na elektroničkom sklopovlju i programskoj podršci koje se mogu koristiti na jednostavan način [4]. Koristi se za kreiranje različitih elektroničkih projekata od onih najjednostavnijih pa sve do vrlo složenih. Nastala je u Italiji 2005. godine i osmišljena kao jednostavan i jeftin alat koji će omogućiti i osobama koje nisu inženjerske struke izradu digitalnih i elektroničkih projekata[5].

Arduino razvojno okruženje sastavljeno je od dva dijela, fizičkog i programskog. Predstavnik fizičkog dijela odnosno sklopovlja je mikroupravljač. Mikroupravljač je integrirani sklop koji je sastavljen od dijelova kao što je procesor, memorija te ulazi i izlazi sustava. Programsku podršku predstavlja Arduino IDE (engl. *Integrated Development Environment*). Arduino IDE je službeno Arduino programsko sučelje koje se koristi za pisanje i prijenos koda na Arduino platforme. IDE čine dva osnovana dijela, urednik koda (engl. *editor*) i prevoditelj koda (engl. *compiler*) gdje prvi služi za pisanje koda, a drugi za prevođenje i učitavanje koda na pločicu. Arduino IDE podržava C i C++ programske jezike.

Arduino danas broji nekoliko desetaka platformi i inačica koje se razlikuju se po tipu procesora, radnoj memoriji, dimenzijama, broju ulaza i izlaza i slično. Zbog svojih karakteristike u ovom radu korištene su Arduino Uno i Arduino Mega platforma.

##### 3.1.1. Arduino UNO

Arduino UNO (Slika 3.1.) najpopularnija je inačica od svih Arduino platformi. Temeljena je na ATmega328p procesoru. Uno na sebi sadrži još 14 digitalnih ulaza/izlaza, šest analognih pinova, ICSP, USB, 16 MHz rezonator te tipkalo za ponovno pokretanje sustava. U tablici 3.1. dane su detaljne karakteristike Uno platforme.



Sl. 3.1. Arduino Uno platforma [6]

Arduino Uno koristiti će se kao razvojni sustav za kontrolu rada upravljačke rukavice odnosno svih komponenti tog sklopa.

<b>Mikroupravljač</b>	ATmega328p
<b>Radni napon</b>	5V
<b>Ulazni napon (preporučeni)</b>	7-12V
<b>Ulazni napon (granični)</b>	6-20V
<b>Digitalni U/I pinovi</b>	14
<b>PWM U/I pinovi</b>	6
<b>Analogni ulazi</b>	6
<b>DC struja po U/I pinu</b>	20 mA
<b>DC struja na 3.3V pinu</b>	50 mA
<b>Flash memorija</b>	32 KB od kojih 0.5 KB zauzima bootloader
<b>SRAM</b>	2 KB
<b>EEPROM</b>	1 KB
<b>Takt procesora</b>	16 MHz
<b>Ugrađena LED</b>	13
<b>Dužina</b>	68.6 mm
<b>Širina</b>	53.4 mm
<b>Težina</b>	25 g

Tab. 3.1. Tablica tehničkih karakteristika Arduino Uno platforme

### 3.1.2. Arduino Mega 2560

Arduino Mega 2560 (Slika 3.2.) predstavlja nešto složeniju inačicu od Arduino Una. Temeljen je na ATmega2560 čipu. Mega platforma na sebi sadrži još 54 digitalna ulaza/izlaza, 16 ulaznih analognih pinova, kristalni oscilator od 16 MHz, USB priključak, 4 UART priključka, ICSP, priključak za napajanje i tipkalo za ponovno pokretanje sustava. U tablici 3.2. prikazane su detaljne tehničke karakteristike Mega platforme. Arduino Mega koristiti će se kao razvojni sustav za kontrolu rada mobilnog robota.



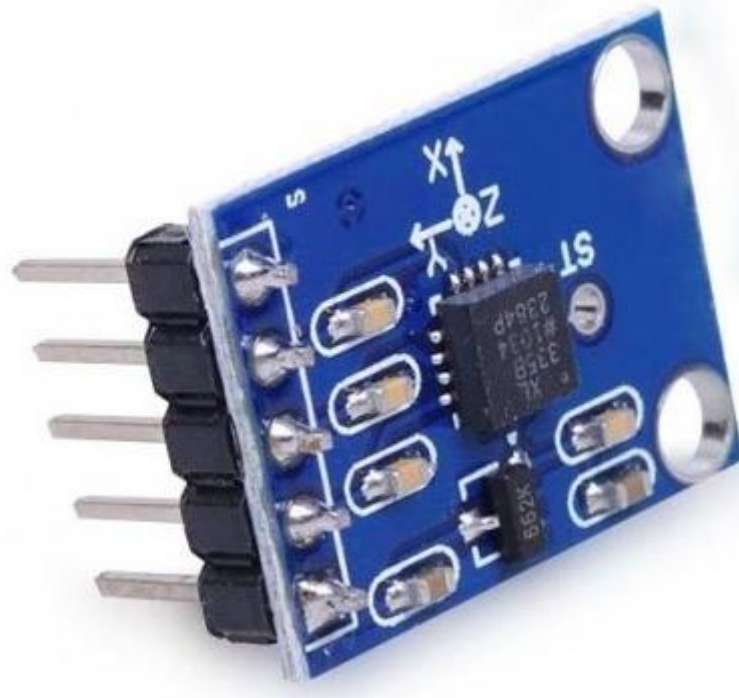
Sl. 3.2. Arduino Mega platforma [7]

<b>Mikroupravljač</b>	ATmega2560
<b>Radni napon</b>	5V
<b>Ulazni napon (preporučeni)</b>	7-12V
<b>Ulazni napon (granični)</b>	6-20V
<b>Digitalni U/I pinovi</b>	54
<b>Analogni ulazi</b>	16
<b>DC struja po U/I pinu</b>	20 mA
<b>DC struja na 3.3V pinu</b>	50 mA
<b>Flash memorija</b>	256 KB od kojih 8KB zauzima bootloader
<b>SRAM</b>	8 KB
<b>EEPROM</b>	4 KB
<b>Takt procesora</b>	16 MHz
<b>Ugrađena LED</b>	13
<b>Dužina</b>	101.52 mm
<b>Širina</b>	53.3 mm
<b>Težina</b>	37 g

Tab. 3.2. Tablica tehničkih karakteristika Arduino Mega platforme

### 3.2. Akcelerometar

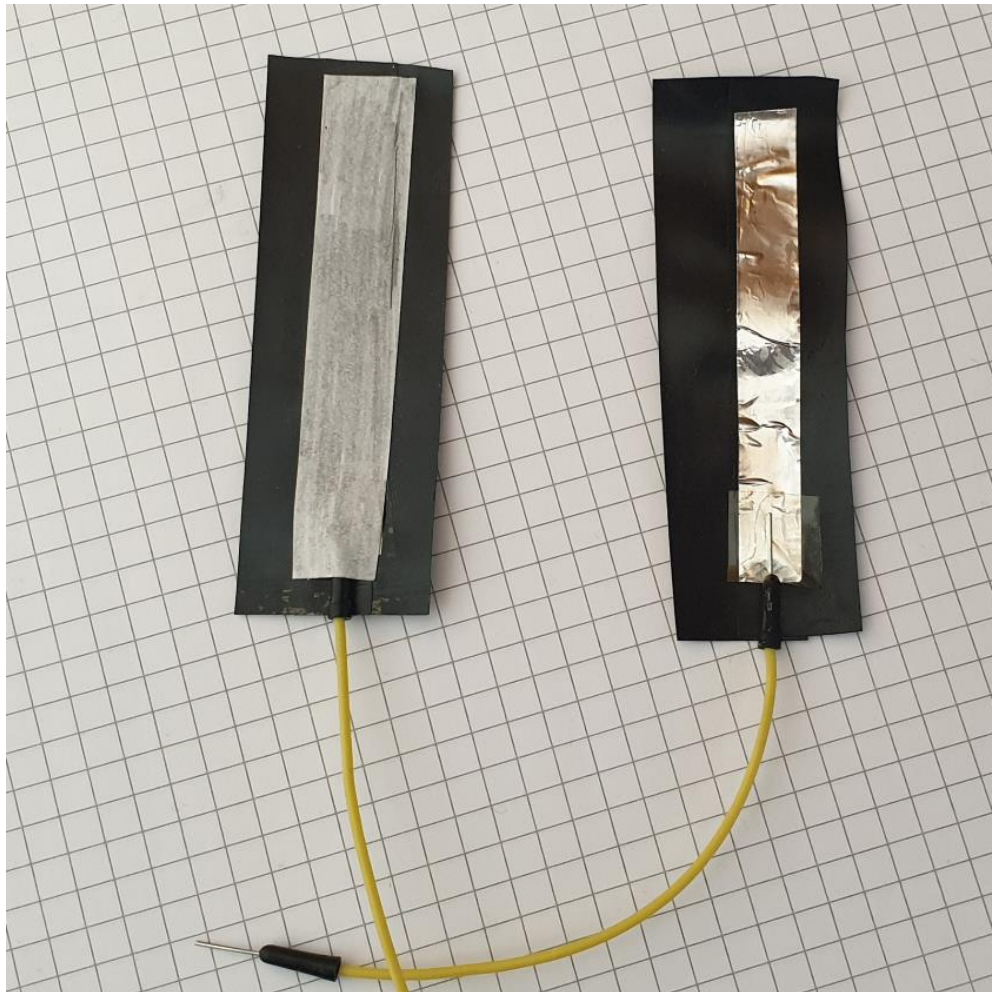
GY-61 DXL335 3-Axis akcelerometar je troosni modul temeljen na ADXL335 integriranom krugu. ADXL335 je troosni akcelerometar s vrlo niskom potrošnjom energije i malom proizvodnjom buke. Senzor posjeduje puni raspon osjetljivosti od +/-3g. Modul može mjeriti statičko ubrzanje gravitacije u aplikacijama osjetljivim na nagib kao i dinamičko ubrzanje koje proizlazi iz gibanja, šoka ili vibracija. Radi na naponu od 3.3V kako bi mogao omogućiti rad ADXL335 čipa stoga mu je za ispravan rad potrebno pružiti napajanje između 3.3V i 6V. Akcelerometar na sebi ima 5 pinova, a to su VCC (napajanje), GND(uzemljenje) i tri pina za tri osi X(duljina), Y(širina), i Z(visina) [6]. U radu će biti korišteni samo X i Y pin jer visina neće biti relevantna za rad sklopa.



Sl. 3.3. Akcelerometar GY-61 DXL335 [8]

### 3.3. Senzor savijanja

Senzor savijanja (engl. *flex sensor*) je senzor koji mjeri iznos pogiba (engl. *deflection*) ili savijanja (engl. *flexion*) na način da daje analogne vrijednosti na temelju savijenosti senzora. Prilikom savijanja senzora mijenja se njegova otpornost te se zbog toga još naziva i fleksibilnim potencijetrom. Na slici 3.4. prikazana je izrada senzora savijanja. Materijali koji su korišteni za izradu su aluminijska vrpca, papir, izolir traka, dvije bakrene žice te komad savitljive plastike.



**Sl. 3.4.** Izrada senzora savijanja

### **3.4. Istosmjerni motori i kotači**

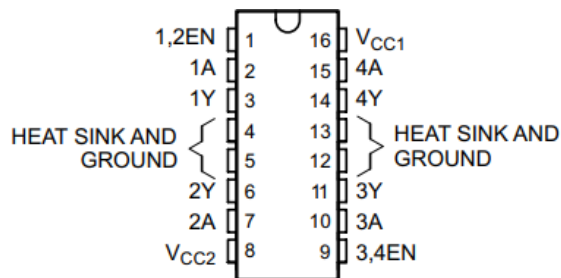
Istosmjerni motor predstavlja uređaj koji pretvara istosmjernu struju u kružno (rotacijsko) gibanje. Istosmjerni motori korišteni su kao pogon mobilnog robota to jest svojstvo pomoću kojega će se robot kretati. Za pogon mobilnog robota korištena su četiri istosmjerna motora koji su povezani u parove (lijevi i desni par). Motori su povezani na kotače (Slika 3.5.) napravljene od gume i plastike kako bi se omogućilo mobilnom robotu kretanje na ravnoj čvrstoj podlozi.



Sl. 3.5. Istosmjerni motori sa kotačima [9]

### 3.5. L293d upravljač motora

Za upravljanje rada istosmjernih motora koristiti će se L293d upravljač motora. Upravljač motora L293d predstavlja integrirani sklop koji se koristi za upravljanje mobilnih robota. Upravljač motora služi kao sučelje između Arduina i istosmjernih motora. Dizajniran je da upravlja radom dva istosmjerna motora istovremeno. Na slici 3.6. nalazi se popis pinova i njihov opis.

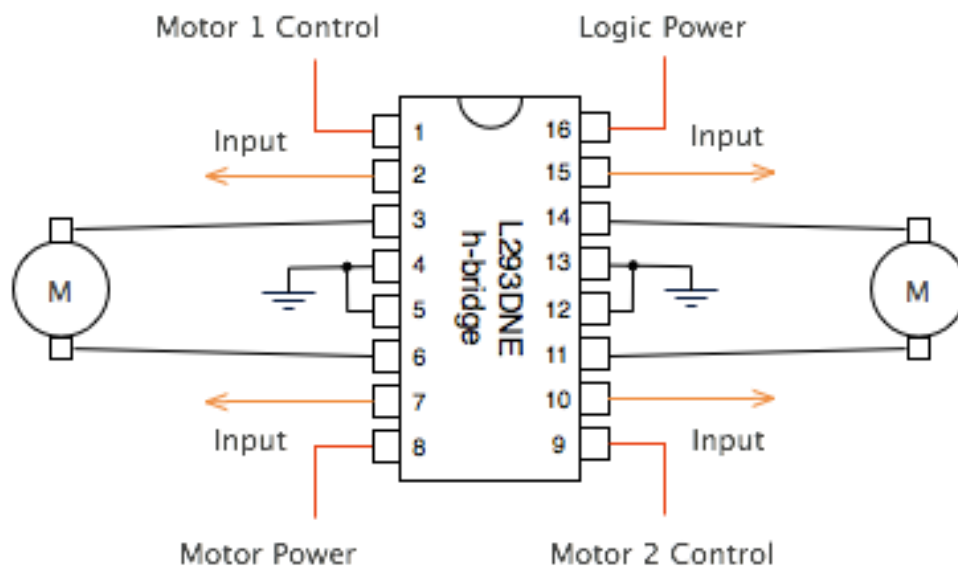


Pin Functions

PIN		TYPE	DESCRIPTION
NAME	NO.		
1,2EN	1	I	Enable driver channels 1 and 2 (active high input)
<1:4>A	2, 7, 10, 15	I	Driver inputs, noninverting
<1:4>Y	3, 6, 11, 14	O	Driver outputs
3,4EN	9	I	Enable driver channels 3 and 4 (active high input)
GROUND	4, 5, 12, 13	—	Device ground and heat sink pin. Connect to printed-circuit-board ground plane with multiple solid vias
V <sub>CC1</sub>	16	—	5-V supply for internal logic translation
V <sub>CC2</sub>	8	—	Power VCC for drivers 4.5 V to 36 V

Sl. 3.6. Opis pinova L293d upravljača motora [10]

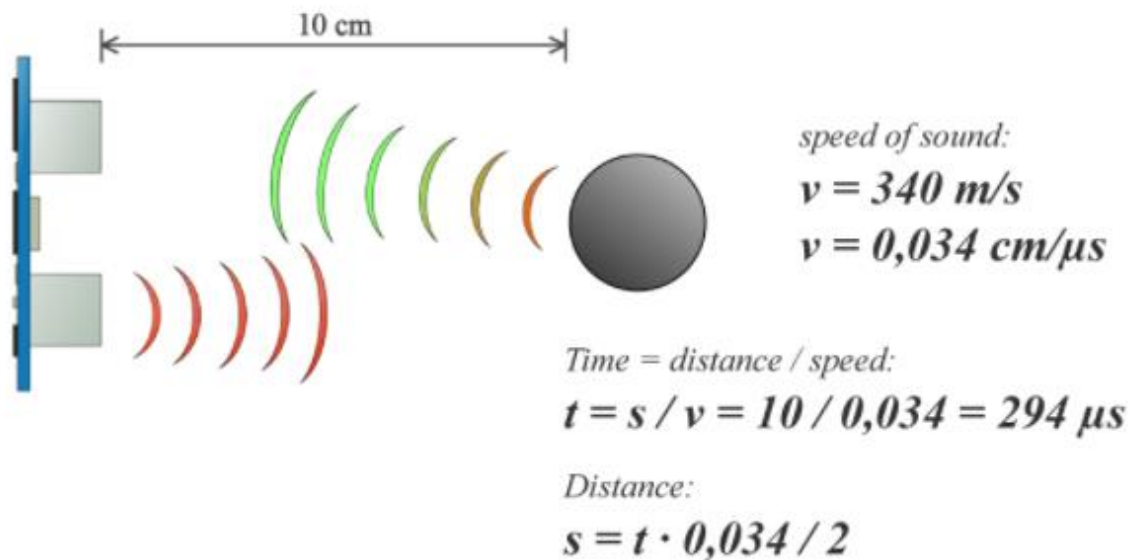
Na slici 3.7. prikazan je način povezivanja istosmjernih motora sa L293d upravljačem.



Sl. 3.7. Shema povezivanja L293d upravljača motora [11]

### 3.6. Ultrazvučni senzor

Ultrazvučni senzori koriste se za dobivanje udaljenosti koja se nalazi između senzora i objekta koji se promatra. Ultrazvučni senzor HC-SR04 može mjeriti udaljenost od 2 centimetra do 400 centimetara. Modul u sebi sadrži odašiljač, prijemnik i kontrolni krug. Senzor radi na način da odašiljač iz modula šalje visoko frekvencijski signal koji putuje kroz zrak. Kada signal dođe nekog objekta, odbija se od njega i vraća nazad do modula gdje ga prijemnik detektira. Senzor tada daje vrijeme koje je bilo potrebna da se signal vrati, a brzinu kretanja zvučnog vala je poznata i ona iznosi 340 m/s. S poznatim vremenom i brzinom udaljenost odnosno distanca od objekta dobiva se njihovim umnoškom. Na slici 3.8. prikazan je postupak dobivanja udaljenosti.

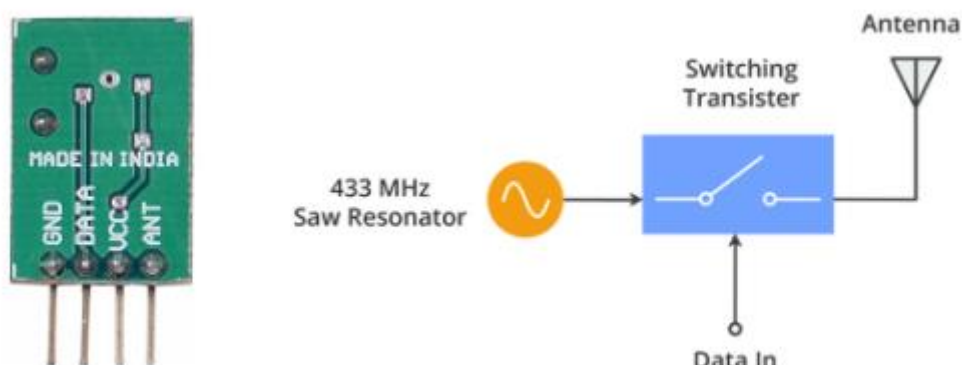


Sl. 3.8. Računanje udaljenosti pomoću HC-SR04 senzora [12]

### 3.7. RF modul

RF (radio frekvencijski) modul je uređaj koji šalje ili prima radio signale kako bi ostvario komunikaciju između dva različita uređaja. RF modul sastoji se od RF odašiljača (engl. *RF receiver*) i RF prijemnika (engl. *RF transmitter*).

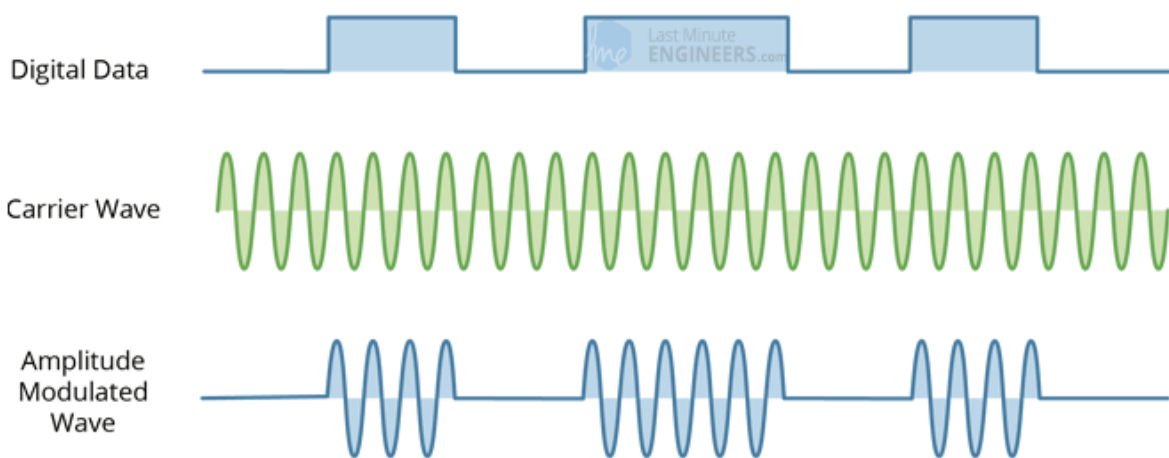
#### 3.7.1. RF odašiljač



Sl. 3.9. RF odašiljač [13]



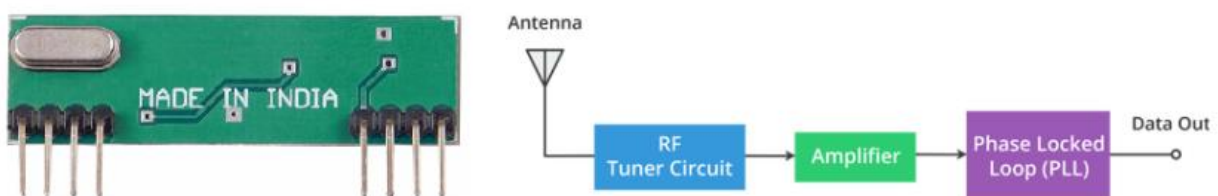
RF odašiljač (Slika 3.9.) dio je RF modula koji je zadužen a slanje signala prema drugom uređaju. Glavna komponenta odašiljača je SAW (engl. *Surface acoustic wave*) rezonator koji je podešen za rad na 433 MHz. RF odašiljač ima 3 pina na sebi, a to su redom GND(uzemljenje), VCC(ulazni napon) i DATA (podatkovni) pin. Kada je logička „1“ poslana na DATA ulaz, oscilator odašiljača počinje proizvoditi konstantni RF izlazni val od 433 MHz sve dok DATA ulaz ne bude postavljen u logičku „0“ gdje oscilator prestaje raditi. Ova tehnika naziva se diskretnom modulacijom amplitude (Slika 3.10.) odnosno ASK(engl. *Amplitude-Shift Keying*).



Sl. 3.10. ASK modulacija [13]

### 3.7.2. RF prijemnik

RF prijemnik predstavlja dio RF modula koji prima signale poslone od strane RF odašiljača. Sastavljen je od nekoliko dijelova od kojih je najvažniji RF podešeni krug te nekoliko pojačala koji pojačavaju dolazne signale. RF prijemnik ima 4 pina, a to su GND,VCC i dva DATA pina.



Sl. 3.11. RF prijemnik [13]

### 3.8. LE diode i otpornici

Svjetleća dioda (Slika 3.12.) odnosno LE (engl. *Light Emitting*) dioda je poluvodički element koji električne signale pretvara u svjetlost. LE dioda biti će korištena kao svjetlosna signalizacija na mobilnom robotu kada je uključen autonoman način rada.



**Sl. 3.12.** LE dioda [14]

Otpornici (Slika 3.13.) su jedna od temeljnih sastavnica većine elektroničkih i električnih sustava. Otpornik je pasivna elektronička komponenta čiji je zadatak ograničavanje jakosti struje u strujnom krugu. Korišteni su gotovo pri svakom spoju u ovom radu radi prevencije od mogućih oštećenja elektroničkih komponenti zbog prevelike jakosti struje.



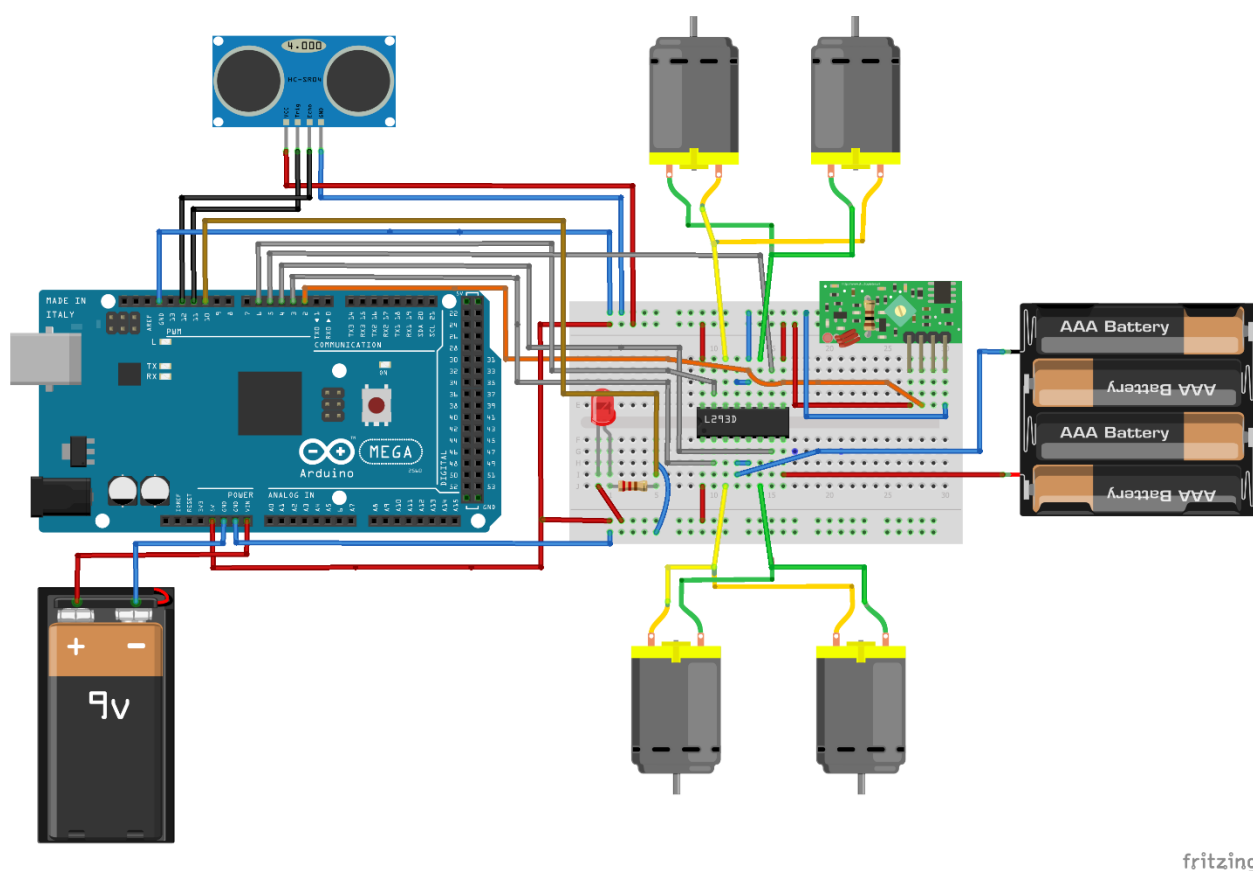
**Sl. 3.13.** Otpornici [15]

## 4. TEHNIČKA REALIZACIJA

Tehnička realizacija ovog rada će se odvijati kroz 3 dijela. Prvi dio je fizička izrada mobilnog robota, drugi je fizička izrada upravljačke rukavice i posljednji dio sastoji se od pisanja programske podrške koja će omogućiti rad prethodno navedenih dijelova. Mobilni robot i upravljačka rukavica zajedno čine mikroupravljački sustav koji će biti baziran na Arduino razvojnom okruženju. U tom sustavu upravljačka rukavica biti će dio sustava koji je zadužen za upravljanje odnosno slanje naredbi prema mobilnom robotu (ima funkciju odašiljača) što znači da će mobilni robot predstavljati dio sustava koji prihvaća naredbu te ju izvršava na način koji je zadan u programskom kodu (imati će funkciju prijemnika).

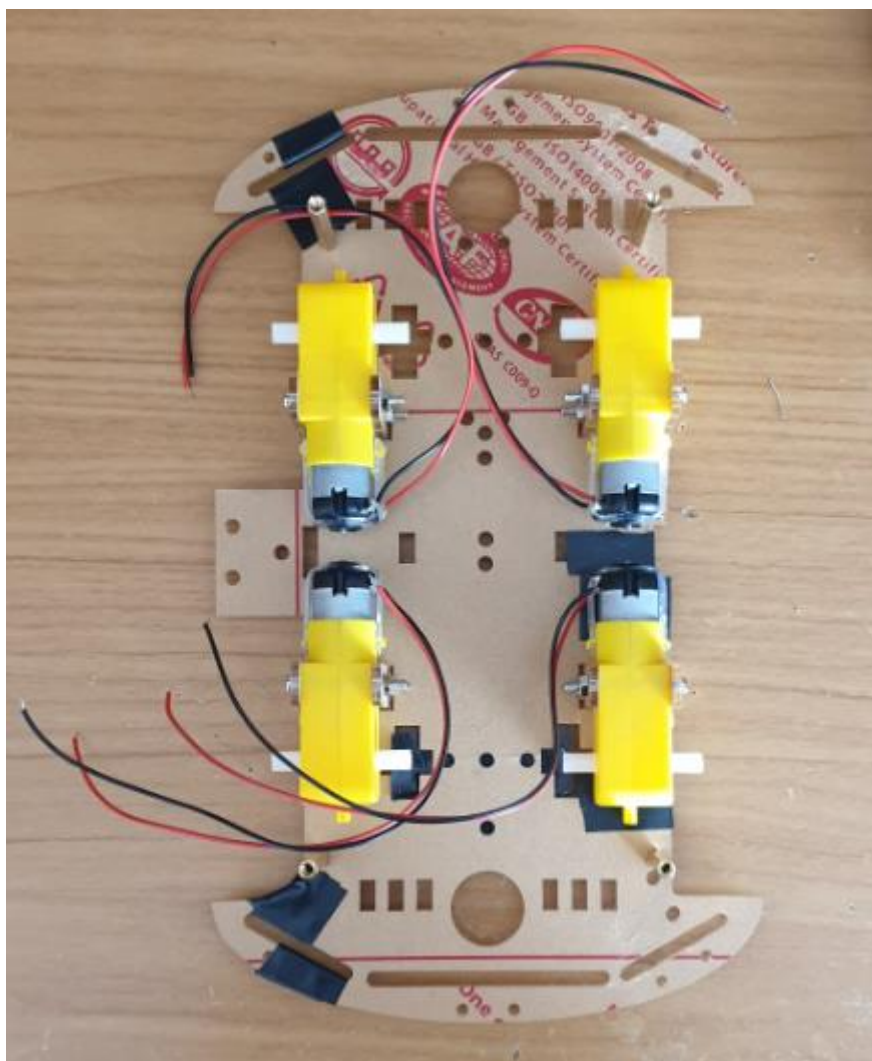
### 4.1. Izrada mobilnog robota

Prije fizičke izrade bilo je potrebno napraviti shemu za spajanje komponenti mobilnog robota. Za izradu sheme korišten je programski alat Fritzing. Fritzing je program otvorenog tipa (engl. *open source*) i spada u EDA (engl. *Electronic Design Automation*) alate koji su namijenjeni za dizajn elektroničkog sklopovlja.



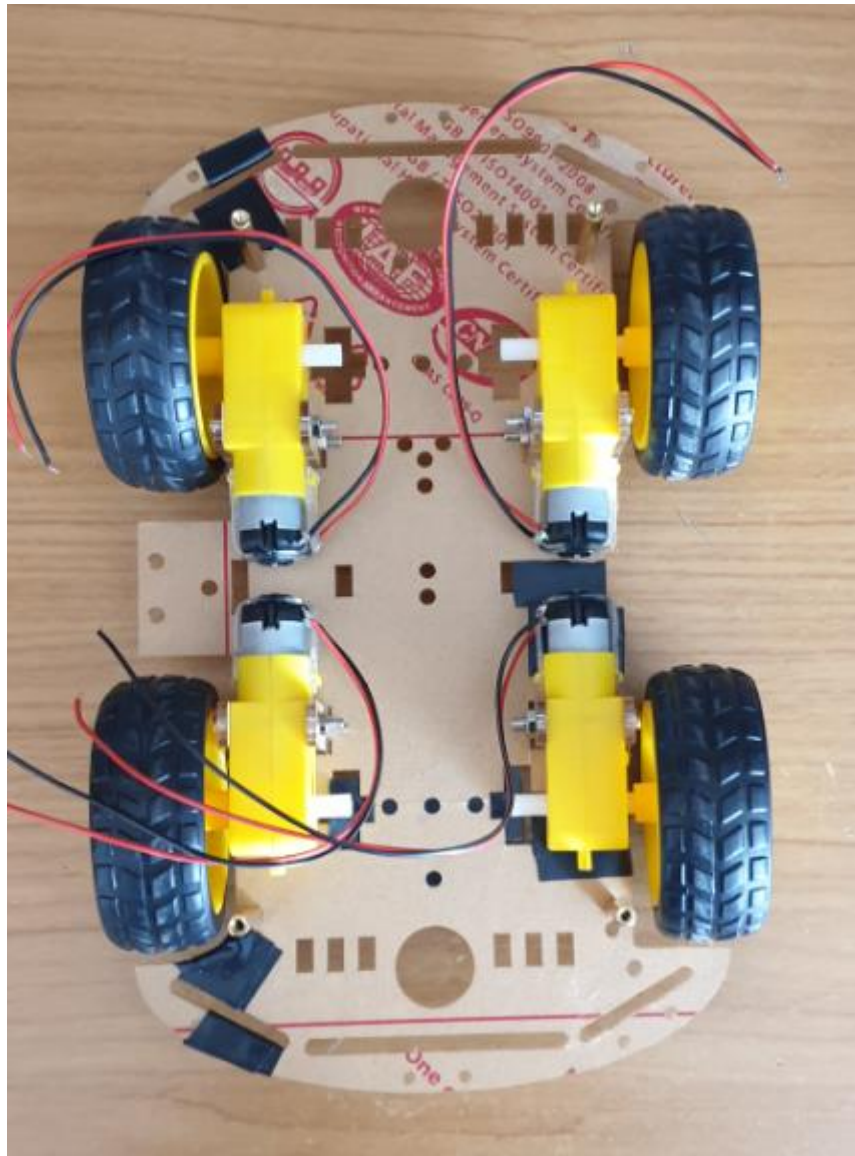
Sl. 4.1. Shema spajanja dijelova mobilnog robota

Nakon izrađene sheme može uslijediti fizička izrada mobilnog robota. Bazu mobilnog robota činiti će dvije plastične ploče na koje će se stavljati ostali dijelovi. Na donju ploču najprije se pričvršćuju istosmjerni motori, dva sa svake strane ploče (Slika 4.2.).



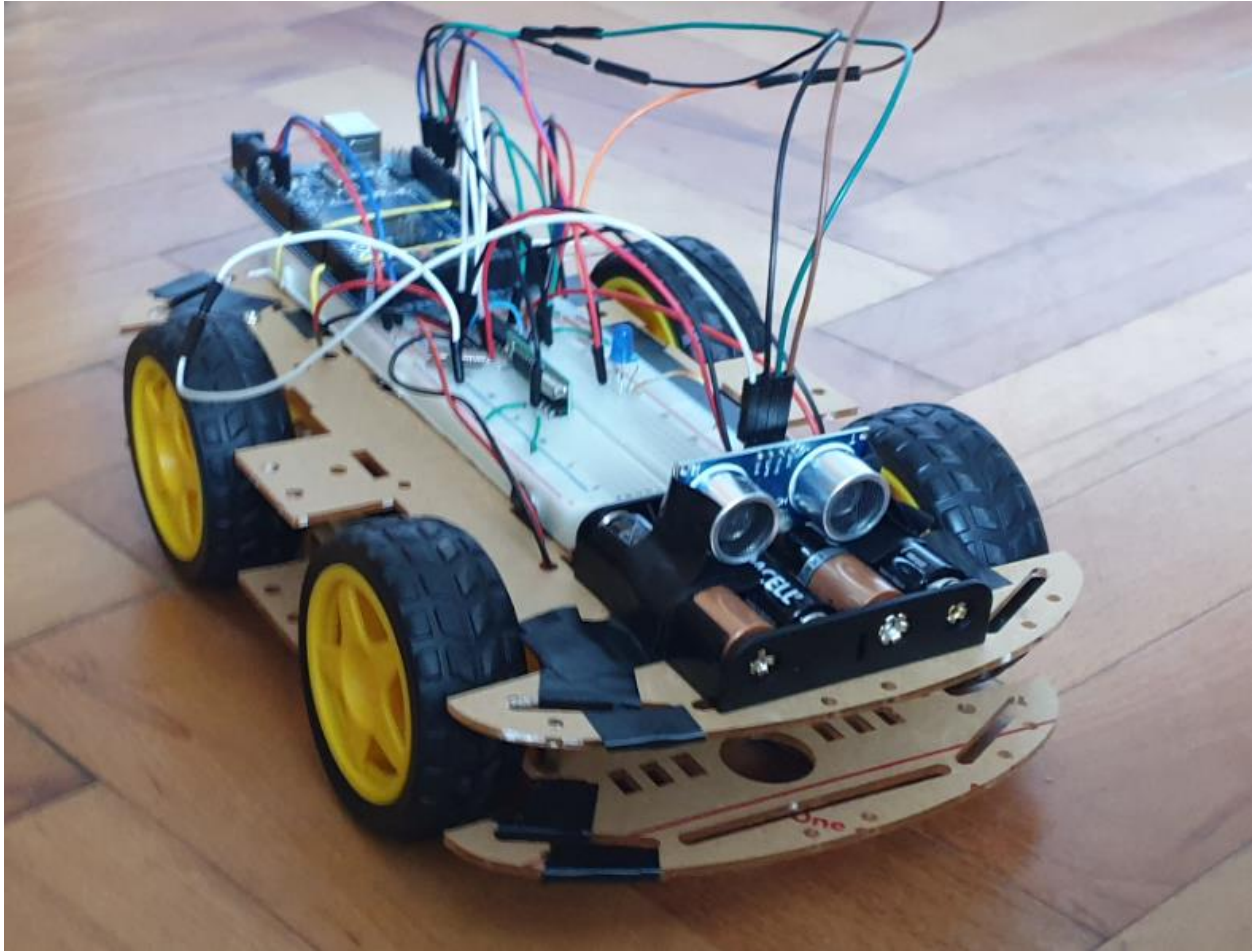
**Sl. 4.2.** Početak spajanja mobilnog robota

Motori su preko bakrene žice povezani i spojeni u parove pa tako ih razlikujemo kao lijeve odnosno desne motore. Na istosmjerne motore dodaju se kotači te se nakon toga gornja ploča može postaviti (Slika 4.3.). Na gornju ploču najprije se postavlja eksperimentalna pločica jer se preko nje povezuju ostale komponente robota.



**Sl. 4.3.** Ugrađeni istosmjerni motori i kotači

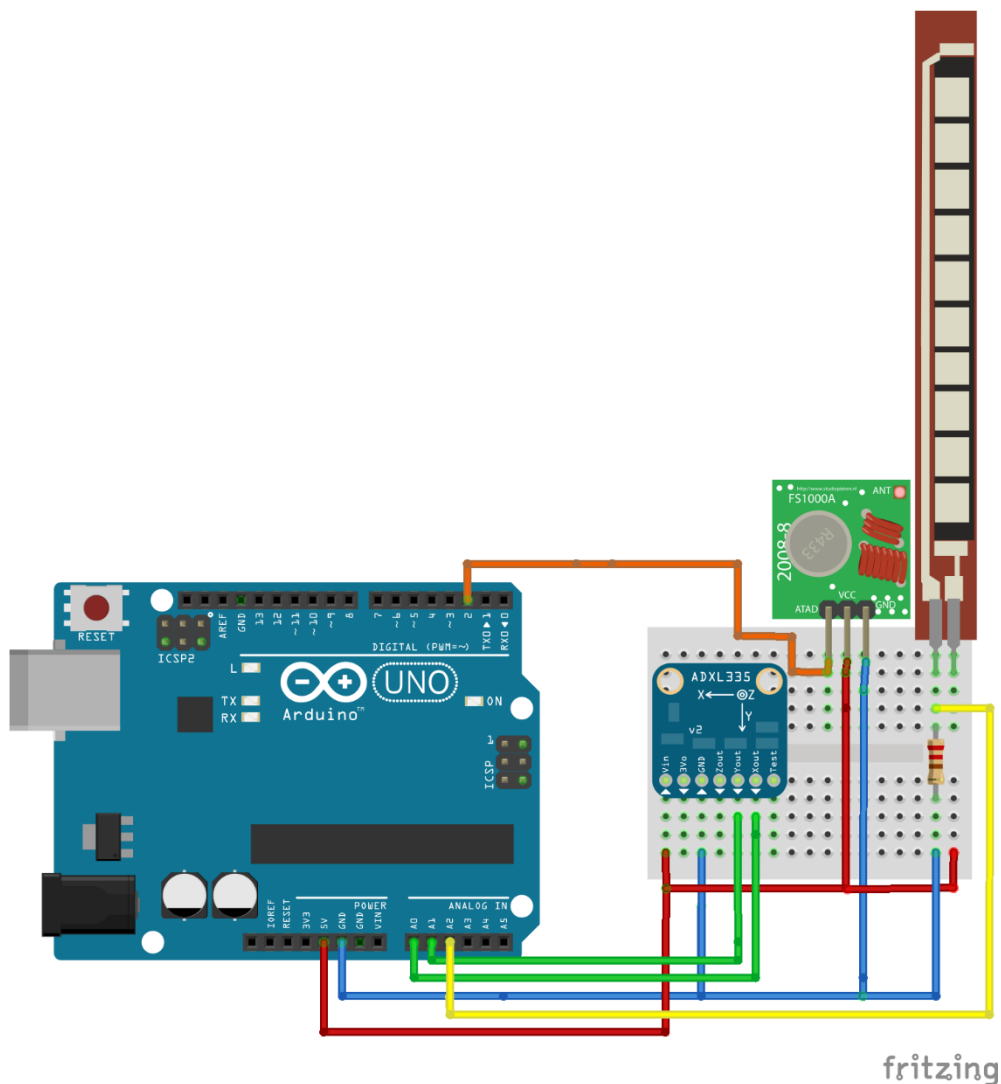
Nakon postavljene eksperimentalne pločice mogu se početi spajati dijelovi mobilnog robota prema izrađenoj shemi. Glavni dio robota je naravno Arduino MEGA. Na Arduino se povezuje ultrazvučni senzor, upravljač motora L293d na kojeg su spojeni lijevi i desni motori, te RF prijemnik koji će primati signale poslane od upravljačke rukavice. Naposljetku su dodani vanjski izvori napajanja odnosno baterija od 9V koja daje ulazni napon Arduino platformi i četiri baterije spojene u seriji koje daju 6V za napajanje L293d čipa koji kontrolira rad istosmjernih motora. Na slici 4.4. prikazan je izgled mobilnog robota spojenog sa svim komponentama.



Sl. 4.4. Izgled kompletno spojenog mobilnog robota

## 4.2. Izrada upravljačke rukavice

Kao i kod mobilnog robota prije same izrade potrebno je napraviti shemu povezivanja dijelova sustava za upravljačku rukavicu. Shema sklopa (Slika 4.5.) upravljačke rukavice dizajnirana je pomoću Fritzing alata.



Sl. 4.5. Shema spajanja dijelova upravljačke rukavice

Završetkom sheme može početi izrada upravljačke rukavice. Iz sheme se vidi da elektronički sklop ovog dijela nema puno komponenata pa je i sama izrada dosta jednostavna. Kao baza za ovaj sustav korištena je biciklistička rukavica za desnu ruku izrađena od poliester materijala. Na rukavicu s gornje strane prvo je postavljena mala eksperimentalna pločica. Na postavljenu eksperimentalnu spaja se Arduino Uno koji će upravljati sustavom. Na Arduino Uno povezani su senzor savijanja i akcelerometar će mjeriti tražene vrijednosti te RF odašiljač putem kojeg će se slati signali prema mobilnom robotu. Na slici 4.6. prikazan je izgled upravljačke rukavice sa svim dijelovima.





Sl. 4.6. Upravljačka rukavica

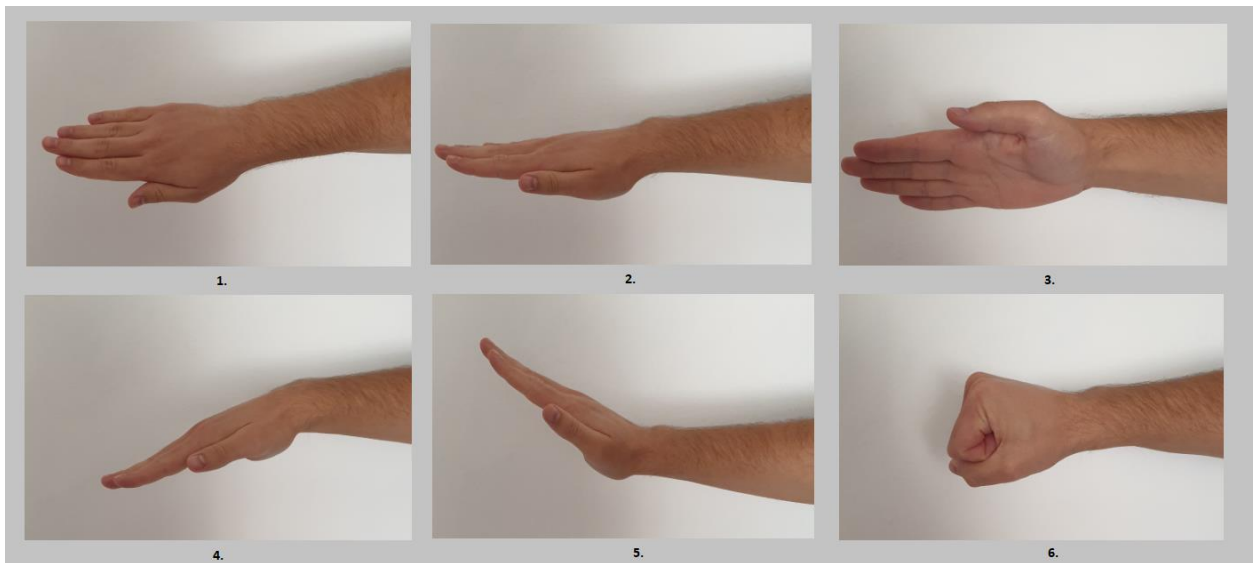
### 4.3. Programsko rješenje

U ovom dijelu opisan je postupak izrade programske podrške za Arduino Uno i Arduino Mega platformu. Programski kod za Arduino Uno platformu omogućiti će rad upravljačke rukavice, a Arduino Mega platforma biti će zadužena za rad mobilnog robota. Oba programska koda pisana su u Arduino IDE sučelju korištenjem C++ programskog jezika. Kroz poglavlje će se također dati pregled korištenih biblioteka te način rada kodova za oba dijela.

#### 4.3.1. Programski kod upravljačke rukavice

Kao što je već spomenuto programska podrška za upravljačku rukavicu biti će rađena na Arduino Uno platformi. Sklop upravljačke rukavice sastoji se od akcelerometra, senzora savijanja i RF

odašiljača te se za njih radi programska podrška u Arduino IDE sučelju. Programski kod započinje dodavanjem RH\_ASK.h biblioteke koja omogućuje rad RF odašiljača te se definiraju ulazni i izlazni pinovi senzora i RF modula tako da odgovaraju načinu na koji su fizički povezani na Arduino platformi. Završetkom definiranja svih potrebnih varijabli i biblioteka napisana je logika programa. Kod radi tako da očitava vrijednosti sa senzora (akcelerometra ili senzora savijanja) u ovisnosti koja gesta ruke je napravljena. Na slici 4.7. prikazane su korištene geste ruku. Akcelerometar prepoznaje prvih pet gesti prikazanih na slici 4.7., a senzor savijanja zadužen je za prepoznavanje šeste geste.



**Sl. 4.7.** Geste ruke korištene za davanje naredbi

Nakon očitanih vrijednosti karakteristični znak za svaku gestu šalje se preko RF odašiljača na RF prijemnik mobilnog robota. U tablici 4.1. dan je popis znakova koji se šalju gestama ruke. Kompletan kod upravljačke rukavice nalazi se u prilogu A.

Poslani znak	Položaj ruke
S	Ruka miruje
F	Ruka je nagnuta prema naprijed
B	Ruka je nagnuta prema natrag
L	Ruka je nagnuta ulijevo
R	Ruka je nagnuta udesno
A	Stisnuta šaka

**Tab. 4.1.** Tablica znakova koje upravljačka rukavica šalje

### 4.3.2. Programski kod mobilnog robota

Programski kod za mobilni robot biti će napravljen pomoću Arduino Mega platforme. Kao i kod prethodnog koda na početku se dodaje biblioteka RH\_ASK.h koja omogućuje rad RF prijemnika i time omogućuje komunikaciju između upravljačke rukavice i mobilnog robota. Nakon učitane biblioteke definirane su potrebne varijable i konstante komponenti. Mobilni robot ima funkciju prijemnika pa se programska logika svodi na čekanje naredbe i u ovisnosti koju je naredbu dobio, izvrši predviđenu funkciju. U tablici 4.2. prikazan je popis funkcija koje mobilni robot izvršava u ovisnosti o primljenom signalu.

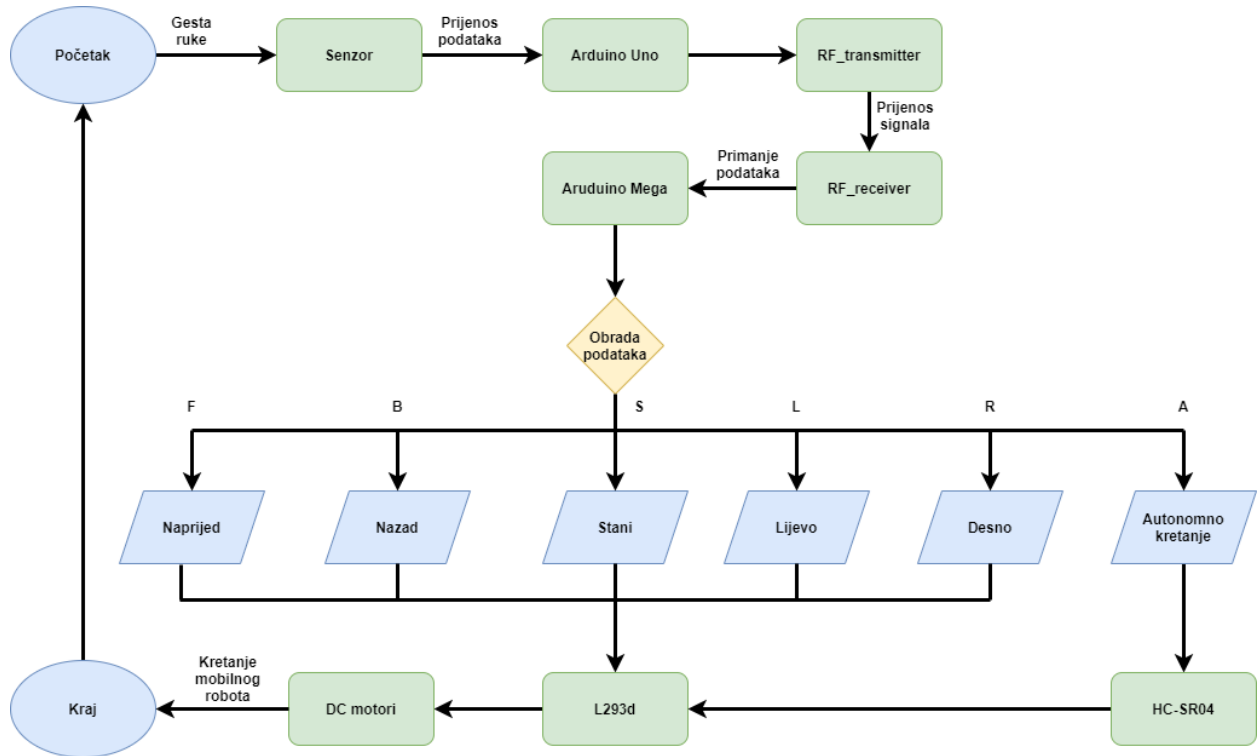
Primljeni znak	Funkcija koju mobilni robot izvršava
S	Mobilni robot se zaustavlja
F	Mobilni robot se kreće prema naprijed
B	Mobilni robot se kreće unatrag
L	Mobilni robot skreće udesno
R	Mobilni robot skreće ulijevo
A	Mobilni robot ulazi u autonomni način rada

**Tab. 4.2.** Tablica funkcija koje mobilni robot izvršava prema primljenom znaku

### 4.4. Rad sustava

U ovom poglavlju dan je opis rada kompletnog sustava mobilnog robota. Rad sustava započinje pokretanjem programa na Arduino Uno i Arduino Mega platformama. Senzori (akcelerometar ili senzor savijanja) spojeni na sklopu kojim upravlja Arduino Uno to jest sklop upravljačke rukavice očitavaju vrijednosti prema trenutnoj gesti ruke. Nadalje, Arduino Uno prema očitanim vrijednostima šalje poruku odnosno specifičan znak putem RF odašiljača. Arduino Mega koji upravlja radom mobilnog robota kroz RF prijemnik čita primljenu poruku te u ovisnosti o sadržaju poruke poduzima sljedeće akcije. Ukoliko je primljeni znak jedan od F, B, S, L ili R Arduino odmah prosljeđuje naredbu na upravljač motora L293d koji pokreće mobilni robot prema naprijed, nazad, lijevo, desno ili ostaje mirovati. Ukoliko je primljeni znak A mobilni robot ulazi u autonoman način rada. Autonomni način započinje tako što L293d upravljač daje znak mobilnom robotu da krene unaprijed te ta funkcija traje sve dok robot ne naiđe na neku prepreku odnosno ne uoči objekt koji se nalazi ispred njega. Prepreka se uočava ultrazvučnim senzorom HC-SR04 koji konstantno mjeri udaljenost. Kada mobilni robot dođe na udaljenost od objekta koja je manja od 20 centimetara vraća se nekoliko centimetara unatrag i okreće udesno za 90 stupnjeva te potom

nastavlja vožnju. Ovakav način rada se nastavlja sve dok mobilni robot ne primi novu naredbu. Na slici 4.8. prikazan je dijagram toka sažima rad čitavog sustava.



SI. 4.8. Dijagram toka sustava

## 5. ZAKLJUČAK

U ovom diplomskom radu izrađen je mobilni robot baziran na Arduino razvojnom sustavu. Sustav ovog projekta sastoji se od upravljačke rukavice i mobilnog robota. Kroz rad dan je pregled svih korištenih komponenti sustava te je opisan način na koji su bile upotrijebljene. Sustav je uspješno realiziran te su sve planirane funkcije mobilnog robota ostvarene. Na osnovu geste ruke upravljačkom rukavicom putem RF komunikacije daju se naredbe za kretanje mobilnog robota. Omogućeno je slanje šest naredbi odnosno senzori upravljačke rukavice prepoznaju šest gesta ruke. Sukladno tome mobilni robot ima šest funkcija koje može izvršiti, a to su kretanje u naprijed i u nazad, skretanje ulijevo i udesno, zaustavljanje te autonomno kretanje.

Ovaj mobilni robot ima i svojih nedostataka kao što je mogućnost kretanja isključivo po ravnoj i čvrstoj podlozi, nedostatak zaštite od vanjskih utjecaja i relativno mala brzina kretanja. No međutim uz svoje nedostatke nudi dosta prostora za proširenje sustava. Neke mogućnosti nadogradnje sustava bile bi izrada zaštitnog kućišta za smještaj elektroničkih komponenti, izrada fleksibilne tiskane pločice za pouzdaniji rad upravljačke rukavice, postavljanje većih elektromotora te amortizera za kretanje po neravnim podlogama ili podlogama sa kosinom, ugrađene sigurnosne kamere tijekom kretanja i slično.

## LITERATURA

- [1] Knjiga: R. Siegwart, I. R. Nourbakhsh, (2004). *Introduction to Autonomus Mobile Robots*
- [2] Slika industrijski robot, <https://medium.com/datadriveninvestor/training-a-robotic-arm-to-do-human-like-tasks-using-rl-8d3106c87aaf>, rujan 2020.
- [3] Slika mobilni robot, <https://www.therobotreport.com/mobile-robots-a3-roundtable/>, rujan 2020.
- [4] Arduino, <https://www.arduino.cc/en/Guide/Introduction>, rujan 2020.
- [5] Internet, <https://www.makerspaces.com/wp-content/uploads/2017/02/Arduino-For-Beginners-REV2.pdf>, rujan 2020.
- [6] Slika Arduino Uno, <https://store.arduino.cc/arduino-uno-rev3>, rujan 2020.
- [7] Slika Arduino Mega, <https://store.arduino.cc/arduino-mega-2560-rev3>, rujan 2020.
- [8] Slika akcelerometar, [http://makeshare.org/bbs/board.php?bo\\_table=Specification&wr\\_id=95](http://makeshare.org/bbs/board.php?bo_table=Specification&wr_id=95), rujan 2020.
- [9] L293d specifikacije, <https://www.ti.com/lit/ds/symlink/l293.pdf>, rujan 2020.
- [10] Slika L293d, <https://developeraspirations.wordpress.com/2010/04/19/l293dne-h-bridge/>, rujan 2020.
- [11] Slika istosmjerni motori sa kotačima, <https://www.nyerekatech.com/shop/dc-gear-motorsupporting-wheels-smart-car-set/>, rujan 2020.
- [12] HC-SR04, <https://howtomechatronics.com/tutorials/arduino/ultrasonic-sensor-hc-sr04/>, rujan 2020.
- [13] RF modul, <https://how2electronics.com/433mhz-rf-module-works-interfacing-arduino/>, rujan 2020.
- [14] Slika LE dioda, [https://www.alibaba.com/product-detail/5mm-LED-Diodes-Flashing-RED-Diffused\\_60675248590.html](https://www.alibaba.com/product-detail/5mm-LED-Diodes-Flashing-RED-Diffused_60675248590.html), rujan 2020.
- [15] Slika otpornik, <https://www.adafruit.com/product/2892>, rujan 2020.

## SAŽETAK

**Naslov:** Mobilni robot upravlján gestama ruke

U ovom radu izrađen je mobilni robot temeljen na Arduino razvojnom sustavu. Sustav se sastoji od upravljačke rukavice i mobilnog robota. Komunikacija između robota i rukavice odvija se preko RF modula. Upravljačka rukavica gestom ruke daje naredbu koja se preko RF odašiljača šalje na RF prijemnik mobilnog robota. Arduino razvojni sustav obrađuje primljeni signal te preko upravljača motora mobilnom robotu daje znak za izvršavanje zadane naredbe.

**Ključne riječi:** Mobilni robot, Arduino, gesta ruke, mikroupravljač, senzor

## **ABSTRACT**

**Title:** Hand gesture controlled mobile robot

In this graduate thesis, a mobile robot based on the Arduino system is made. The system consists of two parts, gesture control glove, and mobile robot. Communication between glove and robot is based on the RF module. Gesture control glove with hand gesture gives an order which is sent from the RF transmitter to the RF receiver of the mobile robot. Arduino process received signal and then through the motor driver gives a sign for a mobile robot to move.

**Key words:** Mobile robot, Arduino, hand gesture, microcontroller, sensor



## **ŽIVOTOPIS**

Matej Vuković rođen je 4. studenoga 1994. godine u Đakovu. Od rođenja živi u mjestu Budrovci gdje pohađa osnovnu školu. Nakon završene osnovne škole upisuje Gimnaziju A. G. Matoša u Đakovu, prirodoslovno – matematički smjer. Srednjoškolsko obrazovanje završava polaganjem državne mature 2013. godine. Iste godine upisuje preddiplomski sveučilišni studij računarstva na Fakultetu elektrotehnike, računarstva i informacijskih tehnologija u Osijeku. Nakon završetka preddiplomskog studija 2017. godine upisuje sveučilišni diplomski studij računarstva, smjer računalno inženjerstvo.

## PRILOG A. Arduino kod upravljačke rukavice

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

int xPin = A0;
int yPin = A1;
int flexPin = A2;

RH_ASK driver(2000, 4, 2, 5);

void setup()
{
  Serial.begin(9600);
  if (!driver.init())
    Serial.println("init failed");

  pinMode(xPin, INPUT);
  pinMode(yPin, INPUT);
  pinMode(flexPin, INPUT);
}

void loop()
{
  int flexdata = analogRead(flexPin);
  int xval = analogRead(xPin);
  int yval = analogRead(yPin);
  Serial.print("x = ");
  Serial.print(xval);
  Serial.print("\t");
  Serial.print("y = ");
  Serial.println(yval);
  Serial.print("\t");
  Serial.print("Flex = ");
  Serial.println(flexdata);
  Serial.print("\t");

  if (flexdata < 200)
  {
    const char *msg = "A";

    driver.send((uint8_t *)msg, strlen(msg));
    driver.waitPacketSent();

  }

  if ((xval > 350 && xval < 390) && (yval > 355 && yval < 390))
  {
    const char *msg = "S";

    driver.send((uint8_t *)msg, strlen(msg));
    driver.waitPacketSent();
  }

  else
  {
    if ((xval > 350 && xval < 390) && (yval > 390 && yval < 450))
    {
      const char *msg = "F";

      driver.send((uint8_t *)msg, strlen(msg));
    }
  }
}
```

```
    driver.waitPacketSent();
}

if ((xval > 350 && xval < 390) && (yval > 280 && yval < 355))
{
    const char *msg = "B";

    driver.send((uint8_t *)msg, strlen(msg));
    driver.waitPacketSent();
}

if ((xval > 285 && xval < 350) && (yval > 355 && yval < 390))
{
    const char *msg = "L";

    driver.send((uint8_t *)msg, strlen(msg));
    driver.waitPacketSent();
}

if ((xval > 390 && xval < 450) && (yval > 355 && yval < 390))
{
    const char *msg = "R";

    driver.send((uint8_t *)msg, strlen(msg));
    driver.waitPacketSent();
}
}
}
```

## PRILOG B. Arduino kod mobilnog robota

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

RH_ASK driver(2000, 2, 4, 5);

#define motor12_en 8
#define motor34_en 9
const int MotorPin1 = 5;
const int MotorPin2 = 6;
const int MotorPin3 = 4;
const int MotorPin4 = 3;
const int ledPin = 10;
const int trigPin = 11;
const int echoPin = 12;
long duration;
int distance;

void setup()
{
  Serial.begin(9600);

  if (!driver.init())
    Serial.println("init failed");

  pinMode(MotorPin1, OUTPUT);
  pinMode(MotorPin2, OUTPUT);
  pinMode(MotorPin3, OUTPUT);
  pinMode(MotorPin4, OUTPUT);
  pinMode(motor12_en, OUTPUT);
  pinMode(motor34_en, OUTPUT);
  pinMode(ledPin, OUTPUT);
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
}

void loop()
{
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);
  duration = pulseIn(echoPin, HIGH);
  distance = duration * 0.034 / 2;
  Serial.print("Distance: ");
  Serial.println(distance);

  uint8_t buf[RH_ASK_MAX_MESSAGE_LEN];
  uint8_t buflen = sizeof(buf);

  if (driver.recv(buf, &buflen))
  {
    driver.printBuffer("Got:", buf, buflen);
    String rcv;

    for (int i = 0; i < buflen; i++)
    {
      rcv += (char)buf[i];
    }
  }
}
```

```

}
Serial.println("Received message");
Serial.println(rcv);

if (rcv == "S")
{
  analogWrite(motor12_en, 0);
  analogWrite(motor34_en, 0);
  digitalWrite(MotorPin1, LOW);
  digitalWrite(MotorPin2, LOW);
  digitalWrite(MotorPin3, LOW);
  digitalWrite(MotorPin4, LOW);
}
else if (rcv == "F")
{
  digitalWrite(ledPin, LOW);
  analogWrite(motor12_en, 255);
  analogWrite(motor34_en, 255);
  digitalWrite(MotorPin1, LOW);
  digitalWrite(MotorPin2, HIGH);
  digitalWrite(MotorPin3, HIGH);
  digitalWrite(MotorPin4, LOW);
}
else if (rcv == "B")
{
  digitalWrite(ledPin, LOW);
  analogWrite(motor12_en, 255);
  analogWrite(motor34_en, 255);
  digitalWrite(MotorPin1, HIGH);
  digitalWrite(MotorPin2, LOW);
  digitalWrite(MotorPin3, LOW);
  digitalWrite(MotorPin4, HIGH);
}
else if (rcv == "R")
{
  digitalWrite(ledPin, LOW);
  analogWrite(motor12_en, 255);
  analogWrite(motor34_en, 255);
  digitalWrite(MotorPin1, LOW);
  digitalWrite(MotorPin2, HIGH);
  digitalWrite(MotorPin3, LOW);
  digitalWrite(MotorPin4, LOW);
}
else if (rcv == "L")
{
  digitalWrite(ledPin, LOW);
  analogWrite(motor12_en, 255);
  analogWrite(motor34_en, 255);
  digitalWrite(MotorPin1, LOW);
  digitalWrite(MotorPin2, LOW);
  digitalWrite(MotorPin3, HIGH);
  digitalWrite(MotorPin4, LOW);
}
else if (rcv == "A")
{
  digitalWrite(ledPin, HIGH);
}

if (digitalRead(ledPin)) {

```

```

analogWrite(motor12_en, 255);
analogWrite(motor34_en, 255);
digitalWrite(MotorPin1, LOW);
digitalWrite(MotorPin2, HIGH);
digitalWrite(MotorPin3, HIGH);
digitalWrite(MotorPin4, LOW);

if (distance < 20)
{
  analogWrite(motor12_en, 255);
  analogWrite(motor34_en, 255);
  digitalWrite(MotorPin1, HIGH);
  digitalWrite(MotorPin2, LOW);
  digitalWrite(MotorPin3, LOW);
  digitalWrite(MotorPin4, HIGH);
  delay(500);
  analogWrite(motor12_en, 255);
  analogWrite(motor34_en, 255);
  digitalWrite(MotorPin1, LOW);
  digitalWrite(MotorPin2, HIGH);
  digitalWrite(MotorPin3, LOW);
  digitalWrite(MotorPin4, LOW);
  delay(1200);
  analogWrite(motor12_en, 0);
  analogWrite(motor34_en, 0);
  digitalWrite(MotorPin1, LOW);
  digitalWrite(MotorPin2, LOW);
  digitalWrite(MotorPin3, LOW);
  digitalWrite(MotorPin4, LOW);
}
}
}

```

## **OSTALI PRILOZI**

### DVD

- Rad u .docx i .pdf formatu
- Arduino programski kod za upravljačku rukavicu u .ino formatu
- Arduino programski kod mobilnog robota u .ino formatu