

Mobilni robot za savladavanje labirinta slijedemnjem linije

Jarić, Karlo

Master's thesis / Diplomski rad

2023

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Rijeka, Faculty of Maritime Studies, Rijeka / Sveučilište u Rijeci, Pomorski fakultet**

Permanent link / Trajna poveznica: <https://um.nsk.hr/um:nbn:hr:187:817793>

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

Download date / Datum preuzimanja: **2025-03-27**



Sveučilište u Rijeci, Pomorski fakultet
University of Rijeka, Faculty of Maritime Studies

Repository / Repozitorij:

[Repository of the University of Rijeka, Faculty of Maritime Studies - FMSRI Repository](#)



**SVEUČILIŠTE U RIJECI
POMORSKI FAKULTET**

KARLO JARIĆ

**MOBILNI ROBOT ZA SAVLADAVANJE LABIRINATA
SLIJEĐENJEM LINIJE**

DIPLOMSKI RAD

Rijeka, 2023. godina.

**SVEUČILIŠTE U RIJECI
POMORSKI FAKULTET**

**MOBILNI ROBOT ZA SVLADAVANJE LABIRINATA
SLIJEĐENJEM LINIJE**

MOBILE MAZE SOLVING ROBOT BY FOLLOWING LINE

DIPLOMSKI RAD

Kolegij: Robotika

Mentor: Izv. prof. dr. sc. Jasmin Čelić

Student/studentica: Karlo Jarić

Studijski smjer: Elektroničke i informatičke tehnologije u pomorstvu

JMBAG: 0112070177

Rijeka, svibanj 2023. godine.

Student/studentica: Karlo Jarić

Studijski program: Elektroničke I informatičke tehnologije u pomorstvu

JMBAG: 0112070177

IZJAVA O SAMOSTALNOJ IZRADI DIPLOMSKOG RADA

Kojom izjavljujem da sam diplomski rad s naslovom

MOBILNI ROBOT ZA SVLADAVANJE LABIRINATA SLIJEĐENJEM LINIJE

(naslov diplomskog rada)

izradio/la samostalno pod mentorstvom

Izv. prof. dr. sc. Jasmin Čelić _____

(prof. dr. sc. / izv. prof. dr. sc. / doc. dr. sc Ime i Prezime)

U radu sam primijenio metodologiju izrade stručnog/znanstvenog rada i koristio/la literaturu koja je navedena na kraju diplomskog rada. Tuđe spoznaje, stavove, zaključke, teorije i zakonitosti koje sam izravno ili parafrazirajući naveo u diplomskom radu na uobičajen, standardan način citirao sam i povezao s fusnotama i korištenim bibliografskim jedinicama, te nijedan dio rada ne krši bilo čija autorska prava. Rad je pisan u duhu hrvatskoga jezika.

Student/studentica

Jarić
(potpis)

Karlo Jarić

Student/studentica: Karlo Jarić

Studijski program: Elektroničke I informatičke tehnologije u pomorstvu

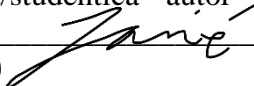
JMBAG: 0112070177

IZJAVA STUDENTA – AUTORA
O JAVNOJ OBJAVI OBRANJENOG DIPLOMSKOG RADA

Izjavljujem da kao student – autor diplomskog rada dozvoljavam Pomorskom fakultetu Sveučilišta u Rijeci da ga trajno javno objavi i besplatno učini dostupnim javnosti u cjelovitom tekstu u mrežnom digitalnom repozitoriju Pomorskog fakulteta.

U svrhu podržavanja otvorenog pristupa diplomskim radovima trajno objavljenim u javno dostupnom digitalnom repozitoriju Pomorskog fakulteta, ovom izjavom dajem neisključivo imovinsko pravo iskorištavanja bez sadržajnog, vremenskog i prostornog ograničenja mog diplomskog rada kao autorskog djela pod uvjetima *Creative Commons* licencije CC BY Imenovanje, prema opisu dostupnom na <http://creativecommons.org/licenses/>

Student/studentica - autor

(potpis) 

SAŽETAK

Definiranjem pojma labirinta postignuta je osnova rada, koja se nadograđuje teorijskim dijelom gdje su obuhvaćene tehnologije i algoritmi primijenjeni u izradi robota koji rješavaju labirinte kao i senzori koji se koriste u aplikacijama te vrste, zatim slijedi primjena u industriji i kao kraj praktična izvedba robota koji rješava labirint primjenom pravila lijeve ruke. Vrlo je važno naglasiti važnost prilagodljivosti robota različitim područjima primjene što otvara mnoge mogućnosti razvoja tehnologije u različitim smjerovima.

Ključne riječi: mobilni robot, algoritmi savladavanja labirinta, labirint, slijeđenje linije, vizualni i nevizualni senzori

SUMMARY

By defining the concept of a maze, the basis of the work has been achieved, which is upgraded with a theoretical part where the technologies and algorithms applied in the creation of robots that solve labyrinths as well as sensors used in applications of this type are covered, followed by application in industry and, finally, the practical performance of a robot that solves maze using the left hand rule. It is very important to emphasize the importance of adaptability of robots to different areas of application, which opens many opportunities for technology development in different directions.

Keywords: mobile robot, maze solving algorithms, maze, line following, visual and non-visual sensors

SADRŽAJ

1. UVOD	1
2. OPĆENITO O LABIRINTIMA	3
2.1. Pojam i teorija labirinta	3
2.2. Razvoj sustava autonomnog rješavanja labirinta kroz povijest.....	4
2.2.1. Mikro miš razvijen na sveučilištu Istočni London	5
2.2.2. Mikro miš razvijen na sveučilištu tehnologije u Maleziji	6
3. ALGORITMI SAVLADAVANJA LABIRINTA	7
3.2. Algoritmi temeljeni na nepoznatim uvjetima labirinta.....	8
3.2.1. Pratitelj zida.....	8
3.2.2. Tremaux-ov algoritam	8
3.2.3. Algoritam slučajnog odabira	8
3.2.4. Založni algoritam.....	8
3.3. Usporedba algoritama korištenih u savladavanju labirinta.....	9
3.3.1. Prednosti i nedostaci algoritma slijeđenja zida.....	9
3.3.2. Prednosti i nedostaci Lee – ovog algoritma	10
3.3.3. Prednosti i nedostaci algoritam preplavlivanja	11
3.3.4. Komparativna analiza.....	12
4. SENZORIKA	14
4.1. Općenito o senzorima	14
4.1.1. Karakteristike senzora	15
4.1.2. Vrste senzora.....	18
4.2. Primjenjivani senzori i njihov princip rada	18
4.2.1. Princip rada ultrazvučnog senzora.....	19
4.2.2. Princip rada infracrvenog senzora	19
4.2.3. Princip rada kamere	20
5. PRIMJENA U INDUSTRIJI	21
6. IZRADA ROBOTA KOJI RJEŠAVA LABIRINT.....	23
6.1. Plan izrade	23
6.2. Komponente robota	25
6.2.1. Kompletan robot	47

6.3. Programski kod.....	49
6.3.1. Opis programa.....	49
6.3.2. Tijek programa.....	49
6.4. Rješavanje labirinta	53
7. ZAKLJUČAK.....	55
LITERATURA	56
KAZALO KRATICA.....	58
POPIS TABLICA	59
POPIS GRAFIKONA	59
POPIS SHEMA.....	59
PRILOG 1	60

1. UVOD

Razvoj tehnologije u današnje vrijeme je veći nego ikad, a zajedno s njim rastu i mogućnosti robotike i automatizacije raznih procesa. Roboti su prvenstveno zamišljeni kao pomoć ljudima ili kao svojevrsna zamjena ljudi na pojedinim pozicijama kako bi se razina kvalitete života mogla podići na višu razinu, zajedno s efikasnošću i preciznošću proizvodnih procesa.

Autonomni mobilni roboti (engl. *Autonomous mobile robot*) prate tu zamisao i u ovom trenutku imaju veliku ulogu u razvoju kako industrije, tako i novih načina mobilnosti te su se pokazali kao vrlo dobra, ako ne i najbolja opcija za pojedine specifične zadatke, primjerice u industriji za točan i brz transport dijelova i ostalih sirovina s jedne pozicije na drugu. Također su se pokazali dobrim pomagačima u traganju i spašavanju ljudi iz opasnih ili teško dostupnih mjesta bez ugrožavanja drugih života. Nadalje, mobilni roboti također pronalaze svoju primjenu i u automatizaciji kućanstava pri čemu izvode razne poslove poput usisavanja, pranja podova i sličnog te na taj način štede vrijeme ljudima i omogućavaju im preraspodjelu slobodnog vremena u njihovu korist [1,2,3].

Autonomni mobilni roboti su inteligentna vozila koja imaju mogućnost putovanja kroz više željenih pozicija, prateći pred definiranu trajektoriju ili navigirajući se u prostoru uz pomoć neke vrste senzora. U svakom slučaju osnovna zadaća robota jest izbjegavanje prepreka i kretanje unaprijed kako bi došao do točke odredišta [1].

Primjer takvog robota jest autonomni mobilni robot koji rješava labirint (engl. *Autonomous Maze Solving Robot*), a posebno je zanimljiv jer prikazuje potencijal robota ovakve vrste. Naime, kao što mu i sam naziv kaže, to je robot kojem je zadaća da u što kraćem mogućem vremenu, odnosno što efikasnije riješi labirint.

Svaka struktura sačinjena od više različitih mogućih puteva se može nazvati labirintom. Put između dvije određene pozicije koje postoje u labirintu mora biti dobro planiran, planirana putanja mora biti bez sudara s drugim predmetima koji bi mogli rezultirati oštećenjem robota, a nakon optimizacije putanje poželjno je i da ruta bude najkraća moguća. U većini aplikacija je najvažniji zadatak robota, pored planiranja puta, takozvano mapiranje. Mapiranje je postupak u kojem robot pamti pojedine dijelove rute, sprema ih i stvara najoptimalniju putanju [1,2].

Optimizacija putanje svoju najveću prednost pokazuje u takozvanom inteligentnom upravljanju prometom koji pomaže vozilima prve pomoći, vatrogasnoj službi ili robotima koji vrše traganje i spašavanje, a pronalazak najoptimalnijeg puta u tom trenutku može biti od presudne važnosti [2].

Tradicionalne tehnike robota koji savladava labirint se uglavnom temelje na pokušaju i pogreški, pri čemu robot pokušava proći svaku dostupnu putanju dok u konačnici ne pronađe izlaz ili odredišnu poziciju. Spomenute tehnike su izrazito vremenski zahtjevne i pri kompleksnijim labirintima mogu uzrokovati, odnosno uhvatiti robota u zatvorenu petlju iz koje ne "vidi" izlaz. Uz ovu činjenicu često ne garantiraju najkraću putanju te je iz tog razloga potrebno birati neki od algoritama rješavanja labirinta kako bi se doskočilo tom problemu [1,3].

Pored toga, izuzetno veliku važnost ima i vrsta senzora, odnosno osjetnika uz čiju pomoć robot ima kontakt s radnom okolinom. Uglavnom se dijele na robote bazirane na kamerama, odnosno obradi slike te robote bazirane na sensorima koji mogu biti ultrazvučni, infracrveni itd.

Rad se sastoji od ukupno pet dijelova, a to su redom općenito o labirintu, algoritmi savladavanja labirinta, sensorika, primjena u industriji te kao kraj rada izrada robota koji rješava labirint, pri čemu se u poglavlju općenito o labirintu prvo definira labirint pa se saznaje o povijesti labirinta i nastanku robota koji rješavaju labirint. Zatim slijede algoritmi savladavanja labirinta gdje se definiraju pojedine metode, objašnjava logika koja stoji iza metoda i na kraju usporedbom pojedinih metoda ukazuje na prednosti i mane kompariranih metoda. U poglavlju sensorika definiraju se vrste senzora prema principu rada, njihove karakteristike te se navode senzori korišteni u konkretnom primjeru, odnosno kod robota namijenjenih rješavanju labirinta. Primjena u industriji objašnjava sveprisutnost i primjenjivost robota u današnjem društvu, dok izrada robota koji rješava labirint zaokružuje cijeli rad iz razloga što su u tom poglavlju primijenjeni svi prethodno definirani pojmovi i činjenice te je isto prikazano na konkretnom primjeru čime se potvrđuje primjenjivost i modularnost robota.

2. OPĆENITO O LABIRINTIMA

2.1. Pojam i teorija labirinta

Labirint se smatra putujućom slagalicom koja se sastoji do zbunjujuće mreže međusobno povezanih putanja i prepreka, a cilj je pronaći putanju od početne do završne točke.

Prvi puta u povijesti labirinti se spominju već u grčkoj mitologiji, gdje je opisano kako je prvi labirint izgrađen u grčkom gradu Knosu po nalogu kralja Minosa kako bi se u njemu zadržalo mitsko biće Minotaur. Kroz povijest se uloga i namjena labirinta mijenjala, od objekta koji je namijenjen da zadrži nešto unutar sebe, preko ritualnih namjena gdje su korišteni kao simbol životnog putovanja, izrastaju u objekt koji je korišten u znanstvene svrhe testiranja inteligencije i orijentacije ljudi. Zahvaljujući labirintima, znanstvenici su otkrili protein čiji nedostatak uzrokuje nastanak Alzheimerove bolesti [14]. Danas su izvor zabave, raznih natjecanja i izvor rješenja logičkih i drugih problema vezanih uz matematiku i računalno programiranje.

U engleskom jeziku postoje dva različita pojma koja se u hrvatskom jeziku jednostavno nazivaju labirint a to su engl. *maze* i engl. *labyrinth* pri čemu je *labyrinth* struktura koja se sastoji od samo jednog ulaza koji je ujedno i izlaz te od samo jednog hodnika. Drugim riječima, ukoliko nakon ulaska u labirint osoba ili u ovom slučaju robot "hoda" dovoljno dugo, vratit će se na isto mjesto odnosno izlaz. S druge strane, *maze* označava strukturu koja se može sastojati od više ulaza, izlaza i hodnika koji su međusobno povezani isprepletenom putanjom, a sam put od ulaza do izlaza nije jednostavan [14].

Općenito se labirinte može podijeliti na dvije vrste: jednostavno povezani labirint i višestruko povezani labirint. Koliko god kompleksno izgledali, većina labirinta je formirana kao jedan neprekinuti zid s puno ogranaka i skretanja [2,5]. Jednostavno povezani labirint (slika 1.) ima konstrukciju putanja koje se nikada međusobno ne spajaju skretanjem, pa je svaki odabrani put ili drugi put ili mrtva točka. Rješenje za jednostavno povezani labirint uvijek se može pronaći takozvanim pravilom lijeve ili desne ruke. Princip ovog pravila je vrlo jednostavan; potrebno je neprekidno držati bilo lijevu ili desnu ruku uza zid i hodati prema naprijed [5].

Višestruko povezani labirint (slika 1.) može imati više od jednog ulaza koji kruže u razne odjeljke, dok jednostavni ima isključivo jedan ulaz i jedan izlaz. Općenito su mnogo kompleksniji za riješiti, stoga je potrebno koristiti i kompleksnije metode.



Slika 1. – prikaz jednostavno povezanog i višestruko povezanog labirinta [5]

2.2. Razvoj sustava autonomnog rješavanja labirinta kroz povijest

Prvu na svijetu poznatu inačicu tzv. mikro miša imena *Theseus* napravio je 1952. matematičar Claude Shannon koja je u osnovi bila magnet unutar igračke miša koji je bio upravljan telefonskim elektromagnetskim relajima koji su svojim paljenjem i gašenjem upravljali smjerom kretanja miša. Ovo postignuće se smatra prvim primjerom strojnog učenja, a direktor *google*-a je jednom prilikom izjavio kako je ono pokrenulo cijelo područje umjetne inteligencije u smjeru koji je danas poznat [15].

Potaknut *Theseusom* sredinom 20. stoljeća problem rješavanja labirinta postao je vrlo važno područje interesa robotike. Urednici časopisa IEEE spektar (engl. *IEEE Spectrum*) prvi puta 1972. godine izlažu koncept takozvanog mikro miša (engl. *Micro mouse*) koji je predstavljen kao malo, mikroprocesorom upravljano autonomno i inteligentno vozilo koje može savladavati labirint. Nadalje, u svibnju 1977. u istom tom časopisu predstavljena je brža verzija robota te popularnost ideje raste i razvija se iz godine u godinu [7].

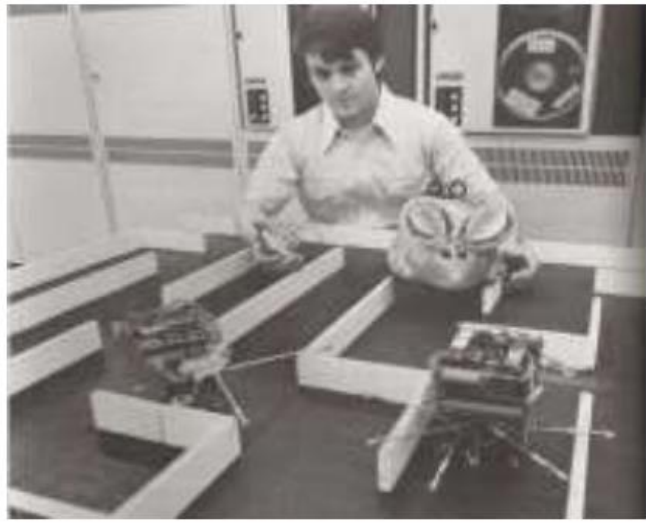
U kasnim 70. godinama 19. stoljeća dizajn robota bio je vrlo velikih dimenzija i sastojao se od mnogo logičkih sklopova. Postupno skaliranje samih sklopova je uslijedilo daljnjim razvojem

robotike i elektroničkih komponenti do dimenzija koje su nam danas poznate. Iz slika 2 i 3 moguće je vidjeti dimenzije i izgled prvih autonomnih robota [7].

Sva događaji popraćeni su brojnim natjecanjima u savladavanju labirinta u kojima su natjecatelji izlagali svoja nadasve zanimljiva rješenja. Natjecanja se provode i danas, a trenutni rekord je ispod 10 sekundi. Kako bi se postigao ovaj zavidan rezultat roboti su opterećeni silama ekvivalentnim formuli F1, a neke od tehnologija su korištenje ventilatora kako bi se poboljšao kontakt s podlogom [15].



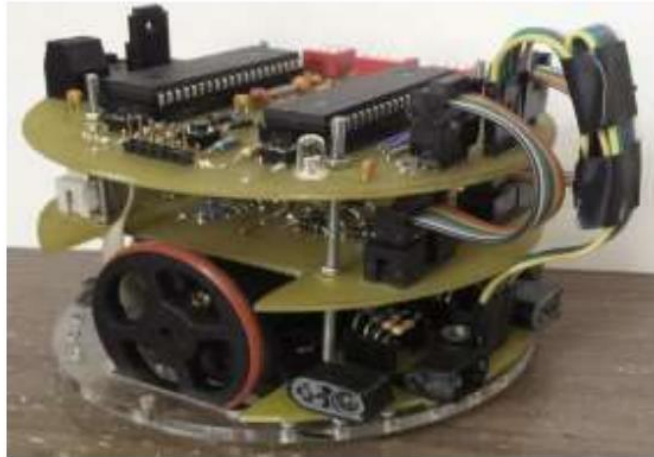
Slika 2. – prototip mikro miša [7]



Slika 3. – usporedba dimenzija robota s čovjekom [7]

2.2.1. Mikro miš razvijen na sveučilištu Istočni London

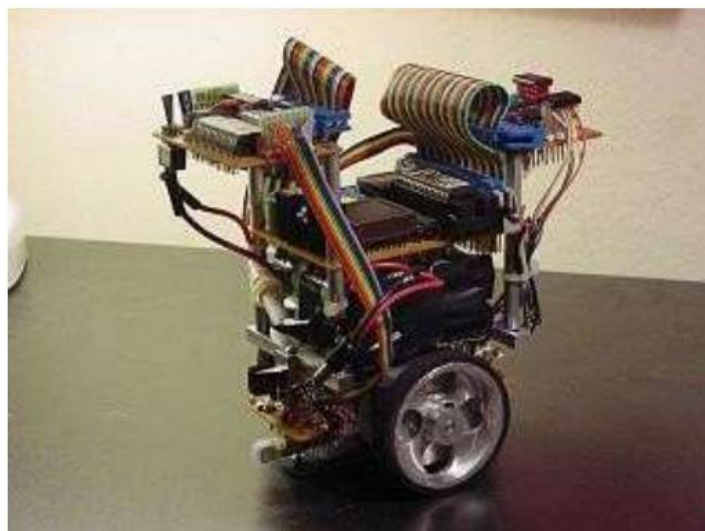
Jedan od prvih modernijih autonomnih robota razvio je Michael Gims na sveučilištu Istočni London 1999. godine. Razvio ga je u suradnji s Sonjom Lenz i Dirkom Beckerom. Koristili su najjednostavniju metodu za rješavanje takozvanih transverzalnih labirinta, a to je metoda slijeđenja zida. Njihov robot nije imao mogućnost inteligentnog kretanja po mapi te nije mogao spremati rute ni pamti petlje kako bi ih kasnije u daljnjim prolazima kroz mapu mogao optimizirati putanju. Koncept njihovog robota se može vidjeti na slici 4, a iz nje je moguće vidjeti veliki napredak u dimenzijama robota kao i konstrukcijskom rješenju [7].



Slika 4. – Gims – ov mikro miš [7]

2.2.2. Mikro miš razvijen na sveučilištu tehnologije u Maleziji

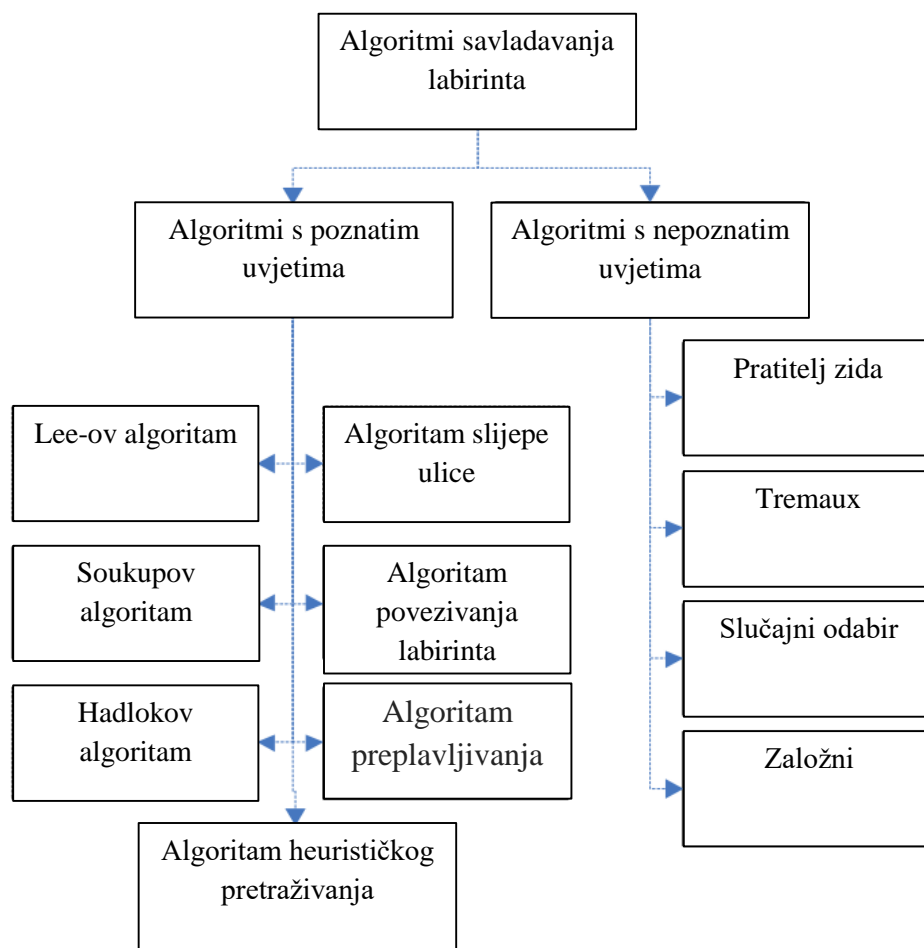
Chan Yuen Chung je student na Sveučilištu tehnologije iz Malezije koji je 2008. godine prvi razvio robota koristeći algoritam preplavlivanja. Ova metoda se pokazala kao superiorna u donosu na prije korištene metode ponajprije zbog brzine rješavanja labirinta. Robot je dizajniran u tri sloja te je mnogo kompaktniji, ali mu je mana smanjena mogućnost detekcije kvara. Algoritam preplavlivanja omogućuje robotu pronalazak najkraće putanje i njezino pamćenje, ali je potrebno više memorije [7]. Prikaz robota može se vidjeti na slici 5.



Slika 5. – Yuen Chang – ov mikro miš [7]

3. ALGORITMI SAVLADAVANJA LABIRINTA

Konfiguriranje algoritma primijenjenog pri svakom pojedinom sustavu savladavanja labirinta predstavlja prvi korak. Postoje razni algoritmi čiji je zajednički cilj pronalazak putanje od početne do završne točke, a međusobno se razlikuju prema svojim karakteristikama. Neki su u mogućnosti pronaći samo jedno rješenje, a neki sva te pojedini algoritmi mogu riješiti sve vrste labirinta, dok neki samo uobičajene vrste. Također ih se razlikuje po brzini rješavanja labirinta, kao i po količini zahtijevane memorije. Opća klasifikacija algoritama se odrađuje u dvije kategorije, koje su algoritmi temeljeni na poznatim i algoritmi temeljeni na nepoznatim uvjetima. Kod algoritama temeljenih na poznatim uvjetima labirinta, robot ima predznanje o uvjetima labirinta, a kod algoritama temeljenih na nepoznatim uvjetima robot nema apsolutno nikakvo predznanje o labirintu kojeg treba savladati [4,5]. Dijagram 1 prikazuje klasifikaciju algoritama.



Dijagram 1. – klasifikacija algoritama [4]

3.2. Algoritmi temeljeni na nepoznatim uvjetima labirinta

3.2.1. Pratitelj zida

Algoritam pratitelj zida (engl. *Wall-follower algorithm*) najčešće je korišten algoritam gdje je glavna ideja pratiti zid labirinta. Robot se drži bilo lijevog ili desnog zida i kreće se prema naprijed kroz labirint dok ne pronađe izlaz. S obzirom na to koristi li se pravilo lijevog ili desnog zida, robot se kreće kroz labirint ili samo ravno i desno ili samo ravno i lijevo [5,6].

3.2.2. Tremaux-ov algoritam

Algoritam zahtijeva označavanje linija na podu kako bi se označila putanja te zasigurno funkcionira u svim labirintima s dobro definiranim putanjama slijedeći ova pravila: (1) ako križanje nema oznake koristi algoritam neoznačenih putanja i na kraju ga označi, (2) ukoliko križanje ima jednu oznaku, okreni se i vrati tom putanjom i označi ju dva puta, (3) ukoliko križanje ima više od jedne oznake koristi onu putanju s najmanje oznaka, prati ju i označi drugi put. U konačnici finalni put ostaje označen samo jednom i označava najkraći put do starta [5,6].

3.2.3. Algoritam slučajnog odabira

Algoritam slučajnog odabira (engl. *Random mouse algorithm*) može biti primijenjen kod bilo kojeg robota i radi po principu kako bi miš riješio labirint. Robot putuje labirintom pokušavajući naći određeno mjesto sve dok ne dođe do križanja gdje mora donijeti odluku koju donosi nasumično. Jedini uvjet je da se nikada ne vraća nazad istim putem nego uvijek ide prema naprijed sve dok ne dođe do odredišta. Ova metoda je primjenjivana od strane neinteligentnih robota te se smatra najjednostavnijom metodom [5,6].

3.2.4. Založni algoritam

Založni algoritam (engl. *Pledge algorithm*) se djelomično oslanja na algoritam praćenja zida. Algoritam služi za rješavanje transverzalnih labirinta, a koristi se u slučaju kada postoje prekidi u zidu s preprekama. Robot drži prepreku bilo na svojoj lijevoj ili desnoj strani, ovisno o primijenjenom pravilu i broji okrete uz pomoć brojača, pri čemu desni zaokret predstavlja povećanje broja stanja brojača, a lijevi zaokret predstavlja smanjenje broja stanja brojača. Kada brojač dostigne nulu, robot je uspješno savladao prepreku i nastavlja dalje savladavati labirint na temelju pravila desne ruke. Robot uvijek započinje savladavanje lijevim zaokretom pa zatim

u ovisnosti o stanju u labirintu primjenjuje slijedeća pravila: (1) ukoliko nema nikakvih objekata na desno robot se kreće desno, (2) ukoliko postoji objekt na desnoj strani robota ali nema objekata ispred robota, robot se kreće unaprijed, (3) ukoliko postoje objekti ispred i nadesno robota, robo se kreće ulijevo. Robot nastavlja put brojeći skretanja sve dok ponovno ne postigne nulu brojača te nastavlja put unaprijed [5,6].

3.3. Usporedba algoritama korištenih u savladavanju labirinta

Uzevši u obzir sve iznad spomenute vrste algoritama, svaka s određenim prednostima i nedostacima, u ovom poglavlju usporedit će se tri algoritma savladavanja labirinta - algoritam praćenja zida, Lee-ov algoritam i algoritam preplavlivanja, koji najbolje odražavaju razlike i mogućnosti istih. Kako bi se algoritmi mogli kvalitetno usporediti potrebno je prvo ispisati njihove prednosti i mane te na temelju toga izvršiti usporedbu i dobiti zaključak.

3.3.1. Prednosti i nedostaci algoritma slijeđenja zida

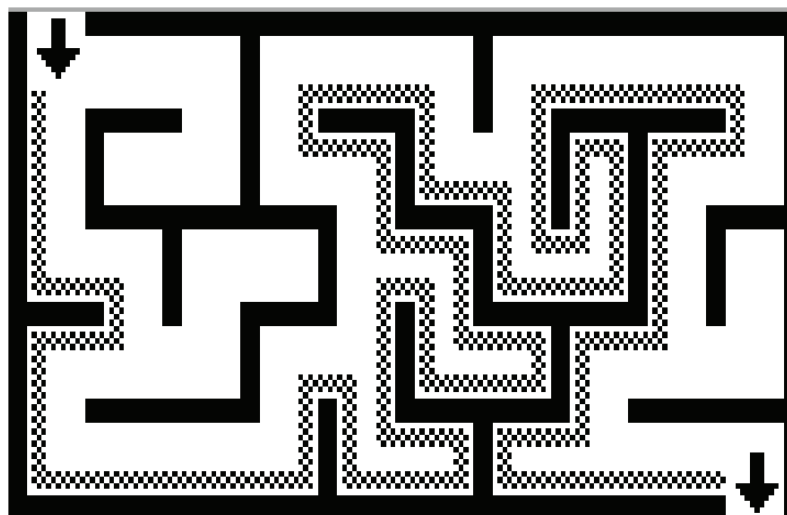
Prednosti:

- (1) Jednostavan princip metode – jedino što je potrebno je da robot „postavi“ svoju lijevu ili desnu ruku uza zid i konstantno prati taj zid bez obzira na raskrižja
- (2) Zajamčeno rješenje – algoritam jamči rješenje labirinta pod uvjetom da je riječ o jednostavno povezanom labirintu
- (3) Jednostavnost algoritma – struktura algoritma je vrlo jednostavna i može se postići bez previše uloženog truda [5]

Nedostaci:

- (1) Uvjet jednostavne povezanosti labirinta – algoritam ne može riješiti labirint koji nije jednostavno povezan iz razloga što algoritam ne vrijedi ukoliko vanjski zid koji ograđuje labirint nije iz jednog komada, već je isprekidan.
- (2) Ograničena memorija – algoritmu je potrebno mnogo vremena kako bi riješio labirint, odnosno došao do određene točke prilikom čega je potrebno u memoriju upisati svako rješenje pojedinog križanja, što zahtjeva mnogo memorijskog prostora koji je u aplikacijama ovakve vrste vrlo limitiran. Stvarni primjeri labirinta su mnogo kompleksniji od testnih pa samim time vrijeme rješavanja još više dolazi do izražaja, kao i ograničenje dostupne memorije namijenjene za pamćenje putanje.

Slika 6, ispod zorno prikazuje primjer rješavanja jednostavnog labirinta metodom slijeđenja zida te je iz nje vidljiva kompleksnost putanje, kao i broj čvorišta koji je potrebno upisati u memoriju. [5]



Slika 6. – primjer rješavanja labirinta algoritmom praćenja zida [5]

- (3) Ograničenost snage – povećanjem kompleksnosti labirinta potrebna je i veća autonomija robota iz razloga povećanja vremena rješavanja labirinta, gdje kao rezultat dobivamo određeno vrijeme autonomije robota u kojem on može rješavati labirint, a koji direktno ovisi o jačini baterije koju robot nosi.
- (4) Sporost – u načelu robot prolazi gotovo čitav labirint prije nego nađe rješenje, što je vrlo vremenski iscrpno odnosno sporo. [5]

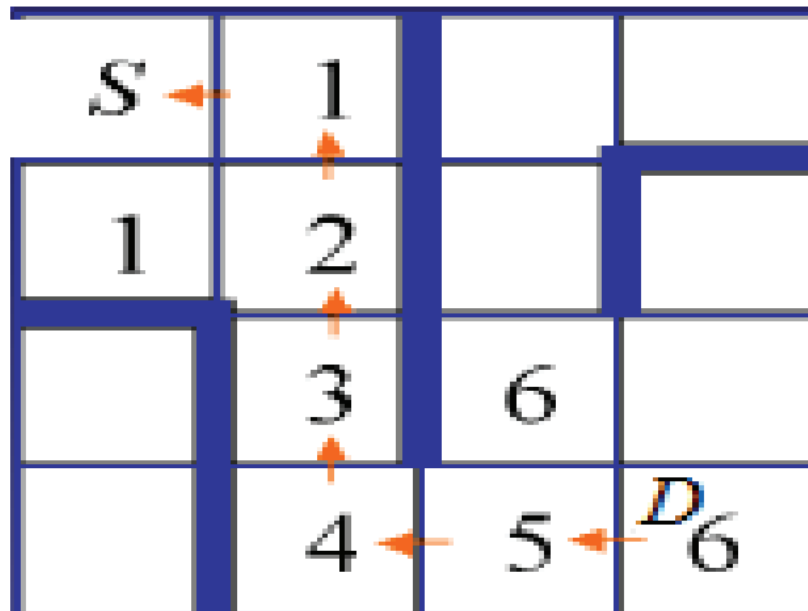
3.3.2. Prednosti i nedostaci Lee – ovog algoritma

Prednosti:

- (1) Algoritam traži najkraću moguću putanju između dvije točke – analizom algoritma može se zaključiti kako se primjenjuje takozvana širinska metoda koja u kombinaciji s Dijkstrovim algoritmom pronalazi najkraću moguću putanju između dvije točke u labirintu.
- (2) Zajamčeno rješenje – algoritam se općenito smatra prikladnijim za rješavanje labirinta u usporedbi s algoritmima koji vrše dubinsku pretragu labirinta koji kao krajnji rezultat ne daju ni približno idealnu ni najkraću putanju. [5]

Nedostaci:

- (1) Potreba za velikom količinom memorije – algoritam ima jako velike potrebe za memorijom, što smatramo velikom manom naročito kod robota čija je izvedba vrlo zbijena, odnosno kompaktnih dimenzija.
- (2) Predznanje o labirintu – glavni nedostatak Lee - ovog algoritma jest da se očekuje određeno predznanje o strukturi labirinta kojeg robot treba savladati te na temelju njega stvara strategiju savladavanja.
- (3) Mala brzina savladavanja labirinta – algoritam jamči pronalazak najoptimalnije putanje, ali uz određeno duže vrijeme. [5]



Slika 7. – princip generiranja putanje Lee – ovim algoritmom [5]

3.3.3. Prednosti i nedostaci algoritam preplavlivanja

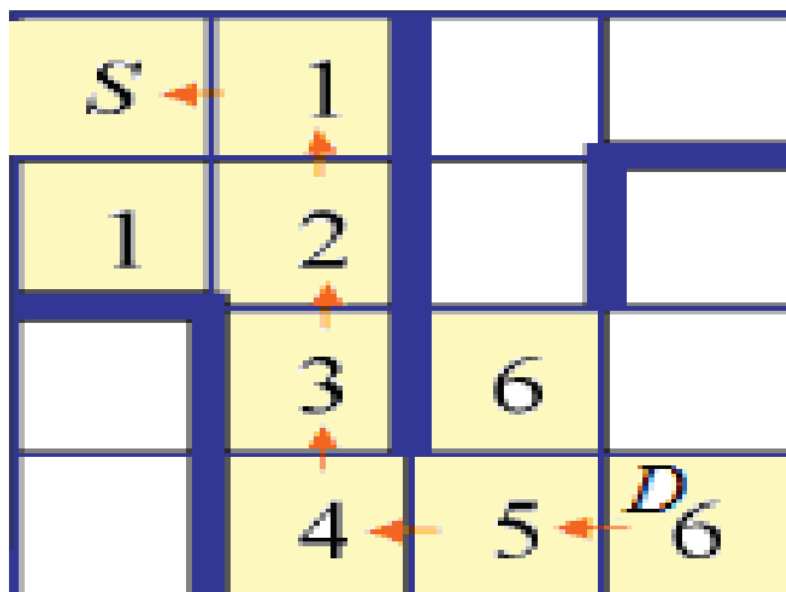
Prednosti:

- (1) Bez predznanja – za razliku od Lee-ovog algoritma nije potrebno predznanje o strukturi labirinta. Algoritam smatra labirint praznim prostorom na kojeg primjenjuje Lee-ov algoritam pomoću kojeg određuje putanju.
- (2) Jednostavnost – algoritam kao i njegova implementacija su vrlo jednostavni.

- (3) Efikasnost – vrlo visoka efikasnost algoritma koja garantira pronalazak rješenja ukoliko ono postoji.
- (4) Najkraća putanja – algoritam daje kao rezultat najkraću moguću putanju između dvije točke bez dodatne optimizacije. [5]

Nedostaci:

- (1) Visoka rekurzivnost – rekurzivnost algoritma odlazi u dubinu vrlo visokim brojem poziva dijelova algoritma, što može dovesti do takozvanog rekurzivnog stoga ili preljevanja.
- (2) Bez provjere – algoritam nema komponentu koja provjerava posjećenost neke putanje.
- (3) Složenost prostora – ima vrlo visoku složenost prostora, što naročito dolazi do izražaja kod velikih labirinta gdje stog može biti vrlo velik. [5]



Slika 8. – determiniranje putanje algoritam preplavlivanja

3.3.4. Komparativna analiza

Za jednostavno povezane labirinte, algoritam slijeđenja zida najjednostavnija je moguća primjenjiva metoda, no u današnje vrijeme labirinti koje možemo pronaći u praksi vrlo često nisu jednostavno povezani, već su vrlo kompleksni što predstavlja problem ovoj metodi

rješavanja koja će možda čak i pronaći rješenje, ali uz vrlo upitno vrijeme trajanja pronalaska, kao i vrlo složenu putanju [5].

Uzimajući ove stavke u obzir, jasno je kako je vrlo vjerojatno potrebno potražiti neki drugi algoritam koji će mnogo optimalnije doći do rješenja glede vremena, utrošene memorije, potrebne energije ili složenosti rezultatne putanje, stoga se Lee-ov algoritam nalaže kao mnogo bolje rješenje uz uvjet da je robotu prije početka rješavanja labirinta vidljiv cijeli labirint. Algoritam jamči optimalnu i najkraću putanju od početka do cilja, no mana mu je kompleksnost prostora i činjenica da zauzima mnogo memorije za pohranjivanje nepotrebnih podataka kako bi pronašao rješenje labirinta [5].

Konačno se može zaključiti kako je algoritam preplavlivanja najbolji izbor za rješavanje labirinta bilo koje vrste bez potrebe da robot prije rješavanja ima uvid u cijeli labirint. Razlog je tome što algoritam u početku smatra labirint praznim prostorom te se krećući labirintom otkrivaju prepreke koje se pohranjuju u memoriju. Algoritam kao rezultat daje najbolju moguću i najoptimalniju putanju odnosno rješenje labirinta, ukoliko ono postoji. Vrlo je jednostavna i pouzdana metoda koja je zasnovana na nizu pokreta koji su definirani logički [5].

Za lakši i pregledniji uvid presjeka algoritama niže je priložena tablica usporedbe.

Parametri	Sljeditelj zida	Lee – ov algoritam	Algoritam preplavlivanja
Princip rada	Pravilo lijeve ili desne ruke	Širinska metoda	Bellman Ford algoritam
Efikasnost	Mala	Srednja	Najveća
Složenost prostora	Srednja	Velika	Velika
Vremenska učinkovitost	Srednja	Velika	Velika
Predznanje	Nije potrebno	Potrebno	Nije potrebno

Tablica 1. – algoritmi i njihove karakteristike [5]

4. SENZORIKA

4.1. Općenito o sensorima

Senzor je po definiciji uređaj koji prima signal ili neki podražaj te ga u ovisnosti o njegovom intenzitetu pretvara u električni impuls - strujni ili naponski. Ulazni podražaj može biti fizički, kemijski ili biološki. Općenita klasifikacija se najčešće vrši prema: aplikaciji, ulaznom signalu, pretvaračkom mehanizmu, korištenim materijalima za izradu, karakteristikama rada i točnosti. Možemo ih pronaći gotovo svugdje, primjerice u uredima, vrtovima, trgovinama, kućama, autima, igračkama itd. Osnovna zadaća im je činiti život ljudi jednostavnijim na način da odrađuju neke naizgled jednostavne zadatke, poput mijenjanja kanala na televiziji, automatskog podešavanja temperature sobe, detekcije predmeta prilikom kretanja vozila unazad umjesto njih. Također se može reći kako je senzor svojevrsni prevoditelj neelektričnih veličina u električne.

Osnovna podjela senzora je na aktivne i pasivne. Pasivni senzori ne zahtijevaju odvojeni sustav napajanja kako bi izvršavali svoju zadaću, već izlazni signal generiraju direktno iz vanjskog podražaja. Ovime se misli kako pretvaraju ulaznu energiju u izlaznu. Primjeri pasivnih senzora su fotografski, termalni, električni, kemijski infracrveni, seizmički. S druge strane, aktivni senzori zahtijevaju odvojeni sustav napajanja kako bi generirali izlazni signal. Aktivne senzore se često naziva i parametričkim, što proizlazi iz mogućnosti podešavanja njihovih unutarnjih postavki i na taj način podešavanja izlaznog signala. Najčešće aplikacije aktivnih senzora vezane su za meteorologiju i prikupljanje podataka o stanju površine zemlje ili atmosfere. Tablica 2 niže prikazuje razlike između pasivnih i aktivnih senzora iz koje se može zaključiti kako svaka vrsta ima prednosti i mane pa je prema potrebi eksploatacije važno izabrati prikladnu vrstu senzora.

Uz ovu postoje mnoge podjele kao što su prema: parametrima detekcije, mehanizmu pretvorbe, analogni i digitalni. Parametri detekcije uključuju električne, magnetske, fizičke, kemijske.

Mehanizmom pretvorbe smatra se sklop unutar senzora dogovorani za pretvorbu ulaznog u izlazni signal čiji su primjeri: fotoelektrični, termo električni, elektromehanički, elektromagnetski. Analogni senzori na izlazu daju analognu veličinu, odnosno izlazni signal proporcionalan je promjeni mjerene veličine, dok kod digitalnih senzora postoje dvije diskretne razine odnosno nula ili jedan [8,9,10,11].

Pasivni senzori	Aktivni senzori
Nema potrebe za vanjskim izvorom energije	Potreban je vanjski izvor energije
Detektiraju zadanu veličinu jedino uz uvjet da ona postoji na području detekcije senzora	Koriste energiju dobivenu iz napajanja te traže zadanu veličinu
Nemaju problem s interferencijom	Imaju manje probleme s interferencijom
Mogu raditi samo u uvjetima za koje su dimenzionirani	Mogu raditi u raznim uvjetima prema potrebi eksploatacije
Osjetljivi na vremenske uvjete rada	Nisu osjetljivi na vremenske uvjete rada
Imaju poteškoće s interpretacijom izlaznog signala	Jednostavno interpretiraju izlazni signal
Manja mogućnost kontrole šuma	Bolja mogućnost kontrole šuma
Niža cijena	Viša cijena
Primjeri: kamera, sonar	Primjeri: laser, radar

Tablica 2. – usporedba pasivnih i aktivnih senzora [8]

4.1.1. Karakteristike senzora

Kako bi se željena veličina uspješno pretvorila u električni signal, potrebno je pomno izabrati vrstu senzora kojeg će se primijeniti u pojedinoj aplikaciji. Navedenu obradu ponajprije vršimo prema karakteristikama senzora koje predstavljaju ključne činjenice i uvelike pomažu pri odabiru senzora. Važno je napomenuti kako različite vrste senzora kao primjerice temperaturni i tlačni senzori pokreta, mogu imati dodatne, specifične karakteristike povezane s veličinom koju mjere ili aplikacijom u kojoj su primijenjeni.

Karakteristike senzora su:

1. Prijenosna funkcija (engl. *Transfer function*) – prikazuje odnos između fizičkog ulaznog signala i izlaznog signala, a može biti linearna i nelinearna.

2. Razlučivost ulaznog signala (engl. *Full scale input* - FSI) – definirana je razlika između minimuma i maksimuma vrijednosti ulaznog signala, a izražena je u decibelima.
3. Razlučivost izlaznog signala (engl. *Full scale output* - FSO) – je razlika između maksimuma i minimuma vrijednosti izlaznog elektroničkog signal, a predstavlja ekvivalent ulaznog minimuma i maksimuma. Uključuje i devijacije od idealne prijenosne funkcije.
4. Točnost (engl. *Accuracy*) – vrlo važna karakteristika senzora koja označava odstupanje izmjerene veličine od stvarne veličine. Može se izražavati u postotnoj vrijednosti u odnosu na izmjerenu veličinu ili u odnosu na raspon.
5. Kalibracija (engl. *Calibration*) – skup podešavanja senzora kako bi na izlazu davao što točniju vrijednost.
6. Pogreška histereze (engl. *Hysteresis*)– vrlo čest fenomen uzrokovan promjenom parametara materijala kao što su trenje ili neke strukturne promjene. Najlakše ju je objasniti kao razliku između dvije izlazne veličine dobivene mjerenjem iste ulazne veličine istim senzorom.
7. Nelinearnost (engl. *Non linearity*) – linearnost funkcije senzora računa se maksimalnom devijacijom ravne linije dobivene jednadžbom linearnosti u određenom dinamičkom pojasu. Nelinearnost senzora računa se kao mjera razlike y osi i dvije kose linije od kojih jedna prolazi minimalnim, a druga maksimalnim vrijednostima izlazne karakteristike senzora.
8. Rezolucija (engl. *Resolution*) – definirana je kao minimalna fluktuacija signala kojeg je moguće izmjeriti pri čitanju ili mjerenju neke vrijednosti upotrebom senzora. Također se može definirati i kao sposobnost zapažanja senzora da detektira i najmanju moguću promjenu mjerene veličine.
9. Zasićenje (engl. *Saturation*) – svaki senzor ima svojevrsna ograničenja, odnosno granicu do koje može mjeriti neku veličinu u željenoj točnosti, a zasićenje je pojava pri kojoj senzor više nema željenu točnost mjerenja.
10. Ponovljivost (engl. *Repeatability*) – smatra se pojavom kada senzor za istu mjerenu veličinu na izlazu daje istu vrijednost.
11. Mrtvo područje (engl. *Dead band*) – područje u kojem senzor praktično ne može izmjeriti mjerenu veličinu, odnosno u tom području bilo koja promjena mjerene veličine na izlazu senzora daje nulu.

12. Pouzdanost (engl. *Reliability*) – sposobnost senzora da izvrši željenu funkciju u željenim uvjetima rada u željenom periodu rada. U okvirima statistike ovu karakteristiku se može protumačiti kao vjerojatnost da će senzor odraditi željeni posao bez greške kvara.
13. Izlazne karakteristike (engl. *Output characteristics*) – kao što ulazne karakteristike ovise o vrsti ulazne veličine, tako i izlazne ovise o vrste izlaznog električnog signala generiranog iz senzora. Izlazni signal može biti strujni ili naponski pa prema tim karakteristikama treba i birati senzor kako bi bio prikladan aplikaciji, odnosno kompatibilan s ostalim komponentama.
14. Impedancija (engl. *Impedance*) – kao i sve ostale komponente nekog sustava, senzori imaju svoju impedanciju koju je potrebno uračunati u daljnju obradu signala. Najlakše ju je saznati proučavanjem karakteristika senzora izdanim od strane proizvođača.
15. Uzbuda (engl. *Excitation*) – signal kojeg je potrebno dovesti senzoru kako bi on počeo mjeriti željenu veličinu.
16. Dinamičke karakteristike (engl. *Dynamical characteristics*) – može ih se otkriti analizom odaziva prijenosne funkcije, kalibracije i dovođenjem raznih oblika funkcije na ulaz senzora. Očituju se kao odstupanja odaziva od ulaznog signala zbog prijenosnih karakteristika senzora.
17. Preciznost (engl. *Precision*) – za razliku od točnosti, preciznost ne mora nužno značiti da su izlazne veličine točne, već je potrebno samo da izlazne veličine budu u određenom području karakteristike senzora kako bi ih se smatralo preciznima.
18. Uvjeti okoline (engl. *Environmental factors*) – vrlo su važni i potrebno ih je uzeti u kalkuciju senzora jer mogu uvelike utjecati na izlazni signal. Postoje posebne izvedbe senzora namijenjene za ekstremne uvjete rada koji su mnogo otporniji na vanjske uvjete. Uz uvjete okoline, važno je i obratiti pažnju na buku i starenje senzora koje degradira sposobnost normalne funkcije senzora.
19. Nesigurnost (engl. *Uncertainty*) – svako mjerenje ima neku nesigurnost, što je općenito podatak koji sadrži određene vrijednosti varijabli koje uzrokuju devijaciju mjerene veličine od ispravne ili stvarne veličine, a mjeri se količinom pogreški kao prosječne veličine skupa podataka.
20. Svojstva aplikacije (engl. *Application characteristics*) - senzori nisu opće dizajnirani, nego ih je potrebno pažljivo birati za pojedine aplikacije kako bi se postigla željena točnost i preciznost izlaznih podataka [8,9,10,11].

4.1.2. Vrste senzora

Postoji gotovo nebrojeno mnogo vrsta senzora, međutim ovdje će biti nabrojani najčešći, odnosno oni koje je danas moguće vidjeti u većini aplikacija.

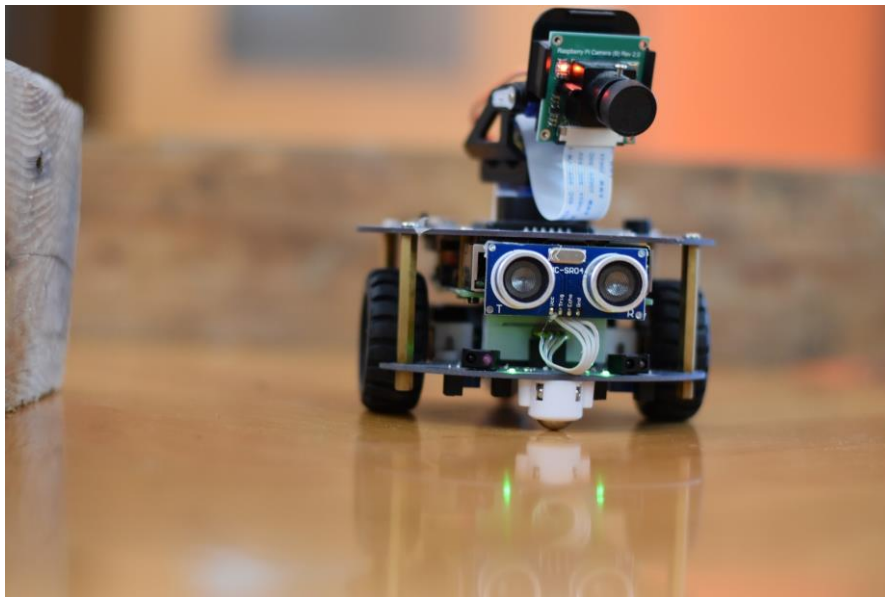
1. Temperaturni – mjere temperaturu
2. Pozicijski – mjere poziciju
3. Svjetlosni – mjere jačinu svjetlosti
4. Zvučni – mjere glasnoću zvuka
5. Senzori za mjerenje udaljenosti – mjere udaljenost
6. Senzori za mjerenje ubrzanja – mjere ubrzanje
7. Infracrveni – detektiraju prisutnost objekata
8. Senzori za mjerenje tlaka – mjere tlak
9. Ultrazvučni – detektiraju prisutnost objekata
10. Senzori osjetljivi na dodir – detektiraju dodir
11. Senzori za mjerenje vlage – mjere postotak vlage
12. Senzori boje – detektiraju boje
13. Kemijski – detektiraju kemijske reakcije
14. Seizmički - mjere gibanja zemlje
15. Magnetski – mjere intenzitet magnetskog polja

4.2. Primjenjivani senzori i njihov princip rada

Kako bi se robot koji rješava labirint mogao kretati kroz njega potrebni su mu različiti senzori. Senzori koriste robotu kako bi otkrio zidove, granice i prepreke u labirintu te kako bi ga mogao uspješno riješiti, odnosno doći do cilja. Postoje razne vrste senzora koje se mogu primijeniti za rješavanje ove problematike, a najčešći su:

1. Infracrveni – senzori emitiraju infracrveno zračenje i mjere njegovu refleksiju od zidova ili podloge.
2. Ultrazvučni – senzori emitiraju visokofrekventne zvučne valove i mjere vrijeme koje je potrebno valovima da se odbiju o tvrdu podlogu, u ovom slučaju od zidova labirinta ili prepreka.
3. Kamere – snimaju slike labirinta koje se potom uz pomoć raznih algoritama digitalno obrađuju te se na taj način pozicioniraju objekti unutar labirinta.

Važno je napomenuti kako je u modernim rješenjima sve češća kombinacija triju navedenih vrsta senzora kako bi se postigla željena točnost i preciznost robota te uz navedene prednosti takvi roboti imaju potencijalno šire područje primjene, kao i veći potencijal u vidu kompleksnosti labirinta koje su sposobni rješavati [8,9,10,11]. Jednu takvu implementaciju možemo vidjeti na slici 9 na kojoj se može vidjeti robot koji rješava labirint koristeći kombinaciju ultrazvučnog senzora i kamere.



Slika 9. – prikaz modernijeg rješenja robota koji savladava labirint [13]

4.2.1. Princip rada ultrazvučnog senzora

Ultrazvučni senzor koristi ultrazvučne valove kako bi detektirao prisutnost ili izmjerio udaljenost od nekog objekta. Sastoji se od prijemnika i predajnika smještenih jedan pored drugog. Predajnik odašilje valove na frekvenciji od 40 kHz u prostor koji se nailaskom na objekt odbijaju od istog i vraćaju nazad u prijemnik. Zatim se mjeri vrijeme koje je bilo potrebno da se val vrati do prijemnika te se na temelju tog vremena mjeri udaljenost od detektiranog objekta [8,9,10,11].

4.2.2. Princip rada infracrvenog senzora

Senzor se sastoji od dvije diode od kojih je jedna prijemna, a druga predajna. Predajnik koristi infracrveno zračenje, a prijemnik ga prima. Princip rada jest da se infracrvena zraka odbija od

objekta i dolazi do prijemnika, koji zatim u ovisnosti od intenziteta na izlazu senzora generira signal. Zraka se bolje odbija od svijetlih predmeta, dok ju tamni upijaju, stoga je vrlo važno obratiti pažnju na okolinu u kojoj se senzor namjerava primjenjivati [8,9,10,11].

4.2.3. Princip rada kamere

Kamere se koriste kod robota koji imaju uvid u cijeli labirint prije samog rješavanja istog, kao i kod labirinta koji nema prethodno znanje. Bez obzira na primjenjivu metodu, načelo rada kamere kao senzora je isto. Robot bilježi sliku kamerom, koja potom odlazi na obradu u kojoj se slika segmentira, odnosno podjeljuje se na objekte. Obrada slike se sastoji od faze učenja, istraživanja, pozicioniranja i integracije informacija. U fazi učenja se može reći da robot uči područje rada, odnosno filtrira važno od nevažnog. Prilikom faze istraživanja, robot bilježi pojedine objekte u prostoru, u ovom slučaju zidove koje potom u fazi pozicioniranja pozicionira u odnosu na sebe, a u zadnjoj fazi, fazi integracije, zabilježeno plasira u kod koji zatim djeluje na motore koji pokreću robota [8,12].

5. PRIMJENA U INDUSTRIJI

Mogućnosti primjene sežu od jednostavnih zadataka poput transporta sirovina u tvornicama, zgradama, uredima, učionama te ostalim radnim prostorima do opasnih zadataka poput dohvata teško dostupnih područja tijekom akcija spašavanja, traženja bombi itd. [4]

Neki primjeri upotrebe robota:

1. Proizvodnja i logistika - roboti se mogu vrlo učinkovito kretati složenim mrežama polica u skladištima, identificirajući najkraće moguće putanje za preuzimanje i isporuku artikala. Mogu optimizirati kretanje robe te time smanjiti ljudski rad i poboljšati produktivnost postrojenja. Također mogu pomoći pri upravljanju zaliha iz razloga što je svaki puta kada robot uzme neki predmet s police zapisan pa je tu informaciju lako dostaviti u program koji upravlja zalihama.
2. Poljoprivreda – u poljoprivrednim djelatnostima roboti se mogu kretati usjevima, voćnjacima ili vinogradima te obavljati zadatke kao što su sadnja, nadzor i berba plodova. Mogu se samostalno kretati strukturama nalik labirintu optimizirajući aktivnosti upravljanja usjevima i tako smanjiti ljudski rad. Roboti mogu biti opremljeni sensorima za analizu strukture tla te tako prikupljati podatke o tlu, zdravlju usjeva i prinosu, pomažući poljoprivrednicima u donošenju odluka o daljnjim postupcima.
3. Potraga i spašavanje – kao što je već nekoliko puta kroz rad spomenuto ovo je vrlo bitna djelatnost koju mogu odrađivati roboti, a njihova prednost posebno dolazi do isticanja u područjima pogođenim katastrofom ili složenim strukturama kao što su ruševine, a sve kako bi pronašli unesrećene. Spasiteljima pružaju povratne informacije koje im pomažu u efikasnosti i planiranju spašavanja.
4. Rudarstvo – u rudarskoj industriji roboti se mogu koristiti za istraživanje i mapiranje podzemnih područja rudnika. Mogu se kretati kroz složene mreže rudarskih okna i tunela kako bi prikupili podatke o geološkim strukturama, identificirali potencijalne opasnosti i pomogli u planiranju radova u rudniku, kao i sigurnosnim protokolima.
5. Obrazovanje i istraživanje – mogu služiti kao obrazovni alati za podučavanje programiranja, robotike i drugih vještina temeljenih na rješavanju problema. Učenici mogu dizajnirati labirinte i programirati robote da se kreću kroz njih i na taj način im pomoći u savladavanju logike algoritama i principa robotike. Također se mogu koristiti u istraživačkim programima za proučavanje navigacijskih algoritama, integracije senzora itd.

6. Zabava i natjecanja – često se koriste u robotskim natjecanjima ili događanjima koja se fokusiraju na autonomnu navigaciju i rješavanje problema takve vrste. Takvi događaji izazivaju sudionike da razvijaju robote koji mogu učinkovito rješavati labirinte te na taj način ukazivati na napredak u robotskoj tehnologiji i algoritmima. Isto tako se mogu koristiti kao uslužni roboti na događajima bilo kakve vrste, pri čemu mogu dostavljati hranu, piće i ostale moguće potrepštine.

Roboti koji rješavaju labirint imaju potencijal revolucionirati različite grane industrije poboljšavanjem učinkovitosti, smanjenjem troškova rada i omogućavanjem autonomnih operacija u složenim okruženjima. Kako tehnologija napreduje, može se očekivati daljnji razvoj ove grane kao i njihova primjena u raznim granama [4].

6. IZRADA ROBOTA KOJI RJEŠAVA LABIRINT

Kako i kod svakog projekta, osnovno je izraditi dobar plan koji se temelji na prethodnom detaljnom istraživanju materije. Projekt je sam po sebi relativno zahtjevan i nikako nije preporučljivo preskakati ovaj dio jer će u protivnom nastati mnogi problemi, a koje se moglo izbjeći u samom početku. Zatim je prema planu izrade potrebno odabrati komponente, kao i dijelove šasije, koje će biti osnova robota. Pri ovom koraku preporučljivo je koristiti pouzdane komponente i dobavljače kako bi se mogućnost kvara ili bilo kakvih drugih problema vezanih uz tu materiju smanjila na minimum. Najbolje je konzultirati se s nekim iskusnijim u tom polju ili pronaći trgovinu u kojoj su prodavači susretljivi i žele pomoći. Nadalje, slijedi programski kod gdje se bira željena metoda rješavanja labirinta, a potom se prema istoj programira robot. Pri ovom koraku vrlo je važno obratiti pozornost na izabrane komponente pa prema njima učitati sve potrebne biblioteke i upravljačke programe potrebne za uspostavu veze mikro upravljača i računala. Rješavanje zadanog problema, u ovom slučaju labirinta, sastoji se od izrade programskog koda, testiranja programskog koda, prilagođavanja istog konkretnom robotu kako bi robot, na primjer, prilikom poziva naredbe okretanja na mjestu pod kutom od 180° istu naredbu i odradio do dobivanja željenog rješenja, što je u ovom slučaju dolazak robota na kraj labirinta. Završni korak projekta je optimizacija rješenja, odnosno postupak u kojem se putanja i vrijeme savladavanja labirinta svode na minimum.

6.1. Plan izrade

Plan izrade ovog projekta započeo je istraživanjem interneta i ostalih dostupnih resursa o zadanoj materiji. Iz istraživanja se može zaključiti kako postoji mnogo mogućih pristupa rješavanju problema, kao i samih fizičkih rješenja robota. Navedeno bi se moglo nazvati „projektom prije projekta“, prvenstveno iz razloga preskakanja bitnih koraka u procesu planiranja, a koje se ne smije izostaviti.

Neki od ključnih koraka uzetih u obzir u procesu planiranja projekta su:

1. Definiranje cilja - Potrebno je jasno definirati cilj robota, stoga se postavlja pitanje želi li se samo uspješno proći kroz labirint ili je cilj projekta pronalazak najkraće putanje. Prema navedenim pitanjima usmjerava se na odgovarajuće algoritme i senzore koji će se koristiti za rješavanje problema.

2. Odabir platforme – Potrebno je razmotriti koja će se platforma koristiti za izradu robota, pri čemu se mogu koristiti mikro kontroleri poput Arduina ili Raspberry Pi-a, robotski setovi koji su prethodno definirani ili se može razviti vlastiti hardver. Odabir je potrebno pomno odraditi, vodeći se vještinama i iskustvom osobe koja izrađuje robota. Najbolje je odabrati mikro kontroler s kojim je osoba koja izrađuje projekt familijarizirana te razumije njegovu osnovu i princip rada.
3. Senzori i aktuatori – Identificiranje senzora i aktuatora koji će se koristiti za navigaciju robota kroz labirint. Kako bi se ovaj dio uspješno odradio potrebno je znati vrstu labirinta koji se želi riješiti, kao i okolinu u kojoj će se labirint rješavati. Neke od mogućnosti su: ultrazvučni senzori za otkrivanje prepreka ili zidova, infracrveni senzori za otkrivanje linije ili zidova, senzori udaljenosti, kamera za vizualnu percepciju itd. Aktuatori, odnosno u ovom slučaju motori, koriste se za kretanje robota, a kontroliraju se mikro kontrolerom. Motori mogu biti bez ili sa četkicama, dc motori, koračni motori ili servo motori.
4. Algoritam za rješavanje labirinta – Potrebno je izabrati odgovarajući algoritam, a može se koristiti jednostavnije algoritme poput onog desne ruke ili lijeve ruke, kao i kompleksnije algoritme poput Tremaux – ovog. Ovisno o željenom cilju, postoji potreba za prilagođavanjem algoritma kako bi se robot znao suočiti s mogućim problemom koji može nastati prilikom rješavanja zadatka.
5. Razvoj programskog koda - Ova komponenta predstavlja inteligenciju robota jer bez nje je robot sam po sebi neupotrebljiv i ne može izvršavati nikakvu zadaću. Ovdje spada i odabir željenog radnog okruženja npr. Visual studio ili Arduino IDE, kao i odabir željenog programskog jezika npr. Python ili C++. Odabir se vrši prema želji i iskustvu korisnika s pojedinim sučeljem. Implementacija samih algoritama i logike omogućit će robotu da donosi odluke na temelju podataka dobivenih od senzora.
6. Testiranje, iteracija i optimizacija – Potrebno je testirati robota i njegovu funkcionalnost na različitim uzorcima labirinta kako bi se provjerila i potvrdila sposobnost rješavanja problema. Promatrajući njegovo kretanje, analizirajući podatke dobivene od senzora i provjeravajući parametre, može se otkriti slabosti robota te ih unaprijediti izmjenom koda ili fizičke komponente koja radi probleme.
7. Zaključak – Konačnim korakom u projektu izrade robota koji rješava labirint smatra se zaključak. U zaključku je potrebno napraviti procjenu uspješnosti i primjenjivost u stvarnom svijetu te navesti probleme s kojima se susretalo tijekom izrade projekta kao i eventualna rješenja kako bi se isti mogli izbjeći.

Iz gore navedenih koraka razvidno je kako je planiranje vrlo važan, a i kompleksan dio izrade robota, stoga mu treba posvetiti mnogo pažnje kako bi se izbjegli eventualni problemi koji mogu nastati prilikom izrade i programiranja robota. U slučaju da dođe do problema, uputno je ponovno proći kroz sve gore navedene korake i na taj način pokušati otkloniti određenu manjkavost.

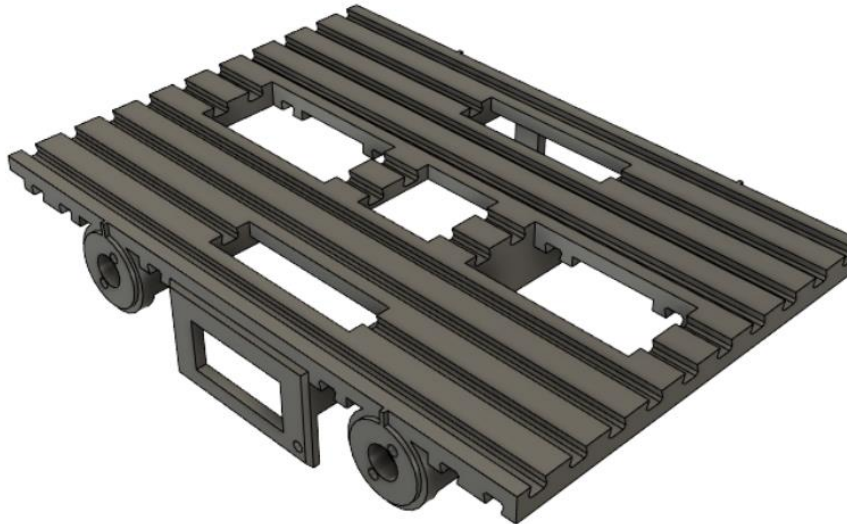
6.2. Komponente robota

Po završetku izrade plana projekta, potrebno je napraviti popis potrebnih komponenti za izradu robota s pripadajućim cijenama u svrhu planiranja troškova. Neki od dijelova robota su izrađeni tehnikom 3D ispisa te im stoga nije moguće procijeniti vrijednost. Tablica 3. prikazuje popis dijelova potrebnih za izradu robota iz koje je vidljivo kako se isti sastoji od ukupno šesnaest dijelova. Dijelovi i njihovo međusobno povezivanje će biti detaljno obrađeni dalje u radu.

Naziv Komponente	Broj potrebnih komada	Cijena (po kom. u €)
1. Šasija (osnova robota)	1	
2. Motor	4	5.56
3. Kotač	4	
4. Spojnice kotača	4	1.25
5. 8x8 led	1	14.84
6. Napajanje	1	20.06
7. Upravljač motora	1	22.95
8. Distribucijska pločica	1	4.74
9. MRMS ESP32 Mikro kontroler	1	35.04
10. CAN bus napajanje	1	2.95
11. 9x infracrveni senzor	1	14.39
12. Baterija	1	6.80
13. CAN bus kablovi	4	1.80
14. 12v kabel	6	1.80
15. 5v kabel	4	1.80
16. 3.3v kabel	2	1.80

Tablica 3. Popis dijelova [Izradio autor]

1. Šasija (osnova robota – slika 10.) – Izrađena tehnikom 3D ispisa se u osnovi sastoji od glavnog dijela, nosača motora, poklopca baterije, kopče baterije i nosača komponenti robota. Uz navedeno, koriste se spojni vijci, matice te odstoynici (engl. *spacer*) kako bi se komponente povezale u cjelinu.



Slika 10. Prikaz šasije robota [16]

Šasija robota prikazana na slici iznad koncipirana je na način da s gornje i donje strane ploče ima male kanaliće koji omogućuju njezinu modularnost, odnosno mogućnost postavljanja raznih modula i na taj način nadogradnje robota. Dimenzije su 100 X 143 mm, a za 3D prikaz moguće je otvoriti pripadajuću STEP datoteku. Na šasiju robota uz pomoć plastičnih vijaka i matica pričvršćuje se poklopac baterije (Slika 11.) kako bi se baterija držala na mjestu, a jednostavnu zamjenu baterije omogućuje njezina kopča (Slika 12.). Navedene komponente su također izrađene 3D pisačem i priložene su uz rad u STEP formatu. STEP format je skraćenica od engl. *Standard for the Exchange of Product Data - STEP* također poznat i kao ISO 10303 norma. STEP je najčešći oblik datoteke korišten u tehnici 3D ispisivanja i razmjene 3D modela u digitalnom obliku, a moguće ju je otvoriti raznim besplatnim, ali i profesionalnim alatima za modeliranje ili mijenjanje postojećih modela, koji se potom šalju 3D pisaču koji ih ispisa. Standardizacija ove vrste datoteka doprinijela je sveprisutnosti tehnike 3D ispisivanja, koja je danas poznata kao tehnika koja omogućuje izradu raznih dijelova kako u vidu hobija, tako i u profesionalne svrhe [18].

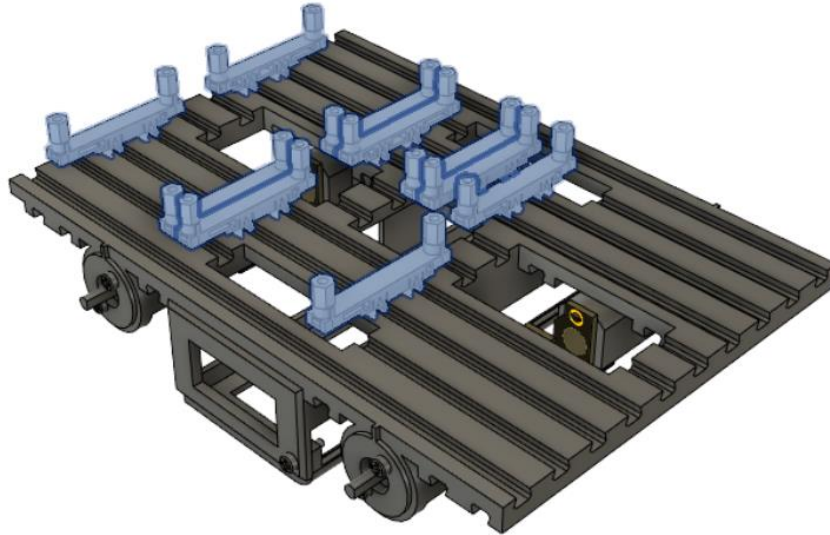


Slika 11. Prikaz poklopca baterije robota [16]



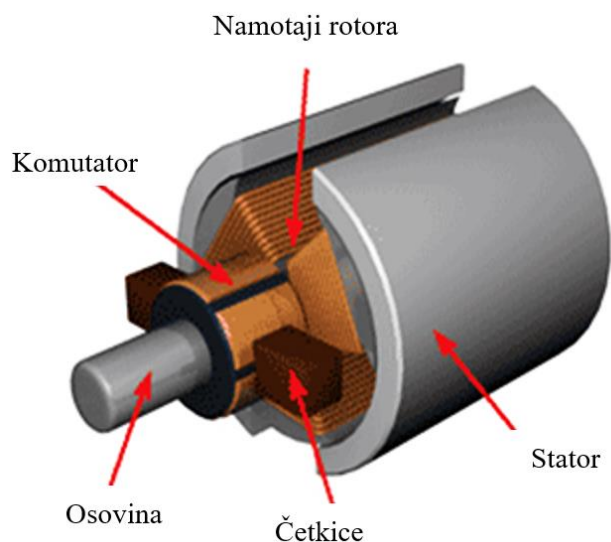
Slika 12. Prikaz kopče poklopca baterije [16]

Sve elektroničke pločice ugrađene na robota su izrađene u standardnim dimenzijama 35 mm širine i duljine, koja je višekratnik broja 5, što omogućuje izradu univerzalnih držača pločica koji su prikazani na slici 13. Na sam držač se postavljaju odstoynici dimenzija 20 mm s standardnim M2 profilom navoja te se na njih uz pomoć plastičnih vijaka postavljaju elektroničke pločice koje će biti obrađene u nastavku.



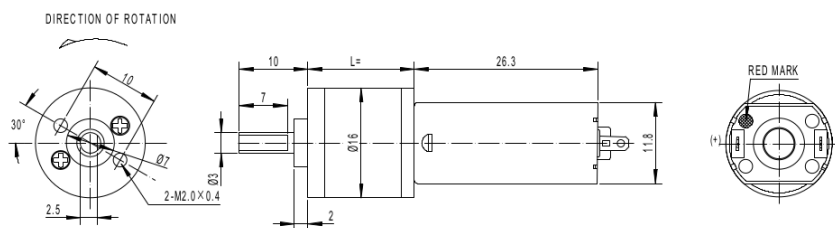
Slika 13. Prikaz držača elektroničkih pločica [16]

2. Motor (Slika 15.) – Pokretačka jedinica, odnosno pogon ovog sustava. Ima ih ukupno četiri, rade na 12 V napona i 150 mA struje (vršna struja 0.55 A), vrsta motora je istosmjerni motor s četkicama, a upravljanje se vrši upravljačem motora. Istosmjerni motori s četkicama koriste princip elektromagnetske indukcije te komutator i četkice kako bi distribuirali struju na rotor, koji potom inducira magnetsko polje koje se opire magnetskom polju statora te se rotor zakreće. Komutator je cilindrična struktura čvrsto vezana na rotor, sastavljena od bakrenih dijelova koji su međusobno izolirani, a svaki bakreni dio predstavlja jedan kraj namotaja rotora. Četkice su uglavnom izrađene od grafita ili karbona te uz pomoć opruge prijanjaju na komutator i na taj način prenose struju na namotaje rotora. Kako se rotor okreće, četkice i komutator zajedno mijenjaju smjer protoka struje kroz rotor, čime se postiže konstantna promjena magnetskog polja i rotacija rotora. Glavne prednosti istosmjernih motora s četkicama su jednostavnost, niska cijena i velik okretni moment pri pokretanju, dok su mane kompliciranije održavanje u odnosu na motore bez četkica. Razlog potonjeg su ugljične četkice koje se troše i stvaraju prašinu na komutatoru i rotoru, a uz četkice se s vremenom troši i komutator, stoga motori ove vrste imaju ograničen rok trajanja [17]. Dijelove istosmjernih motora s četkicama vidi se na slici 14., odnosno pojednostavljenom modelu motora. Navedeni motori su se pokazali kao dobar izbor za ovu aplikaciju zbog omjera cijene, izdržljivosti i upravljivosti, premda bi potencijalno bolji izbor bio istosmjerni motor bez četkica, međutim on bi podigao cijenu motora, a posljedično i cijenu cjelokupnog robota.



Slika 14. Model istosmjernog motora s četkicama [18]

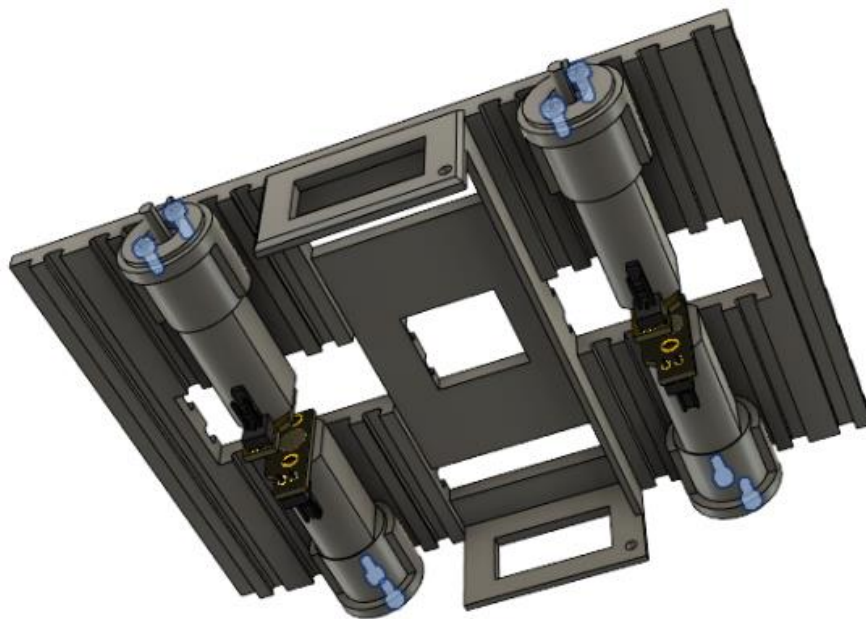
U sklopu motora imamo kompaktan prijenos 1:100, što znači da se osovina rotora motora okrene jednom kako bi se izlazna osovina motora okrenula sto puta. Ovaj prijenosni omjer omogućuje robotu dobar okretni moment, kao i mogućnost savladavanja svih prepreka koje mu se mogu naći na putu. Dimenzije motora mogu se vidjeti iz njegove skice na slici 16., dok je za ovaj projekt najvažnija dimenzija ukupnog promjera na najširem dijelu zbog postavljanja motora u ležište, a koja iznosi 16 mm. Sastavni dio motora, uz prijenos, je adapter konektora (slika 17.), koji omogućuje postavljanje priključnog kabla pod kutom od 90°, a spaja se s motorom tehnikom mekog lemljenja. Motori se pričvršćuju za šasiju robota pomoću dva metalna vijka profila M2 i duljine 6 mm. Način postavljanja, kao i mjesto postavljanja, može se vidjeti na slici 18.



Slika 15. Prikaz motora [19] Slika 16. Prikaz skice motora s dimenzijama [19]



Slika 17. Prikaz adaptera konektora [16]



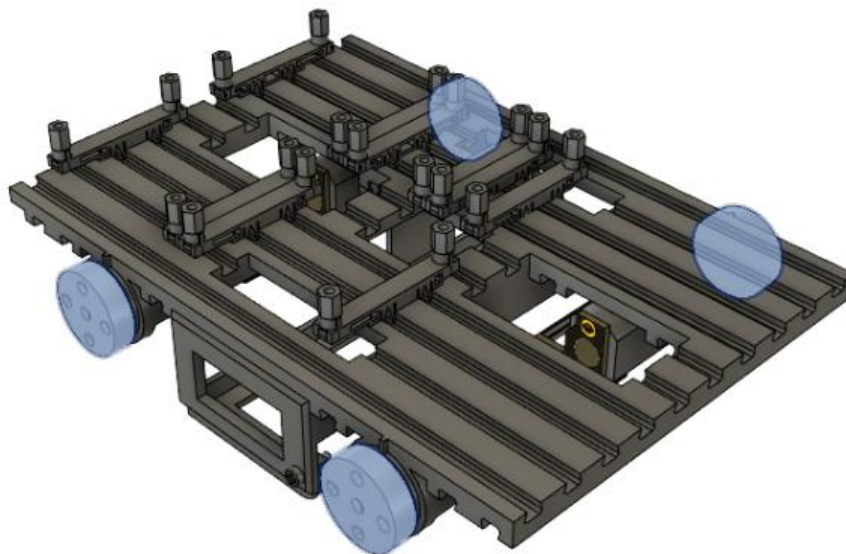
Slika 18. Prikaz načina postavljanja motora i pripadajućih vijaka [16]

3. Spojnice kotača – Kako bi se okretni moment prenio s osovine motora na kotač, potrebno je imati kvalitetan prihvata, koji je u ovom slučaju izveden uz pomoć metalnog cilindra izrađenog od aluminija tehnikom CNC glodanja s provrtom 3mm na sredini vrlo male razlike u promjeru u odnosu na promjer osovine motora, a kako bi se postiglo još bolje prianjanje, ugrađen je i maleni imbus vijak M3 profila koji se steže direktno na osovinu. Nakon dobro uspostavljenog kontakta s osovinom, potrebno je ostvariti i dobar kontakt s kotačem, što je izvedeno uz pomoć četiri dijagonalna provrta profila M3 u koje je moguće

stegnuti maksimalno četiri vijka s vanjske strane kotača, no provjereno je dovoljno koristiti dva od četiri provrta za kvalitetan kontakt. Osim svega navedenog, ovaj dio robota namijenjen je i centriranju kotača pa ukoliko postoje neke nepravilnosti na osovini robota, iste se može u nekoj mjeri ispraviti. Dosjed spojnice u kotaču je prethodno predviđen tako da metalni dio upada u plastični okvir kotača. Na robotu koristimo ukupno četiri ovakva komada, za svaki kotač po jedan, a model je vidljiv na slici 19. Slika 20. prikazuje način postavljanja dijela na osovinu motora.



Slika 19. Prikaz spojnice kotača [20]



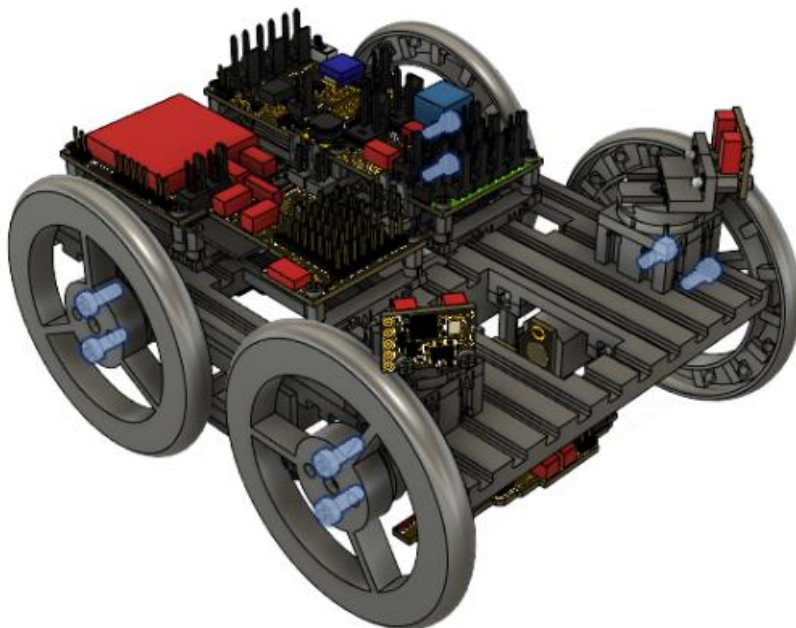
Slika 20. Prikaz načina postavljanja na modelu robota [16]

4. Kotač – Uz motor, kotač predstavlja jednu od glavnih komponenti pri prijenosu okretnog momenta na podlogu pa ga je stoga vrlo važno dobro dimenzionirati kako bi se izbjegli problemi s upravljivošću robota. Ovdje su korišteni kotači izrađeni tehnologijom 3D pisača u kombinaciji s lijevanjem tekućeg silikona. Silikon je korišten kako bi se povećao kontakt

s podlogom, a po kotaču je potrebno upotrijebiti točno 12g silikona. Robot ukupno ima 4 kotača promjera 70 mm pričvršćenih spojnicom kotača s dva metalna vijka profila M3 duljine 5 mm. Prikaz kotača nalazi se na slici 21., dok je na slici 22. prikazan način postavljanja kotača na robota. Navedene modele moguće je otvoriti u 3D prikazu otvaranjem priložene STEP datoteke.

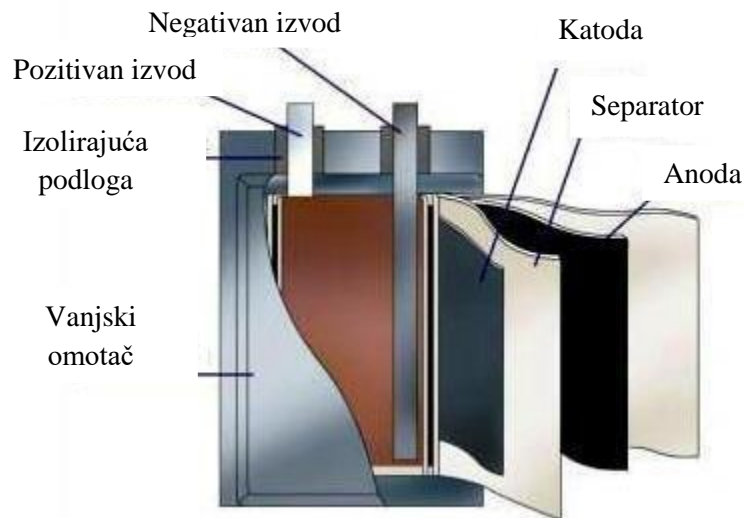


Slika 21. Prikaz kotača [16]



Slika 22. Prikaz postavljanja kotača uz pomoć dva vijka [16]

5. Baterija – Kako bi robot mogao imati određenu autonomiju, potrebno je u prethodno spomenuto kućište ugraditi bateriju. U ovom slučaju ugrađena je Li-Po baterija napona 11.1 V i kapaciteta 1.3 Ah koja se sastoji od tri članka (3S). Litiji – polimer (Li-Po) baterije su vrsta punjivih baterija koje koriste Litij – ion tehnologiju. Poznate su zbog takozvane velike gustoće energije, lagane konstrukcije i mogućnosti da isporuče velike struje. Konstrukcijski se sastoje od nekoliko ćelija povezanih u seriju. Svaka ćelija se sastoji od pozitivne elektrode (katode) napravljene od litiji-kobalt-oksida ili litij-magnezij-oksida, negativne elektrode (anode) uobičajeno napravljene od grafita, separatora koji sprječava direktan kontakt između elektroda te vanjskog fleksibilnog omotača sačinjenog od polimera. Gustoća energije, odnosno prethodno spomenuta velika gustoća energije kod ove vrste baterije, znači da su takve baterije sposobne spremi značajne količine energije u relativno male dimenzije, što ih čini pogodnima za razne aplikacije u kojima je odnos utrošenog prostora i mase baterije bitan, kao što su potrošačka elektronika, dronovi i prijenosni uređaji. Normalan napon pojedine ćelije iznosi 3.7 V, a baterije se najčešće definiraju prema broju članaka baterije, stoga postoje npr. 2S i 3S baterije gdje brojčana oznaka označava broj ćelija koje čine bateriju. Kapacitet je izražen u mili Amper satima (mAh) ili Amper satima (Ah) što označava koliko su energije sposobne spremi. Li-Po baterije zahtijevaju posebne procedure prilikom punjenja i pražnjenja kako bi im se osigurala dugovječnost i sigurnost upotrebe. Potrebno ih je puniti specijaliziranim punjačima koji nadziru napon pojedine ćelije i na taj način preveniraju prepunjivanje, dok s druge strane suviše duboko pražnjenje pojedine ćelije uzrokuje oštećenje koje je trajno, čime baterija postaje praktično neupotrebljiva. Pri rukovanju Li-po baterijama vrlo je važno pročitati preporuke proizvođača kako bi se spriječila eventualna samozapaljenja, dimljenje ili neko drugo trajno oštećenje [21]. Struktura baterije prikazana je na slici broj 23., dok slika 24. prikazuje moguće oštećenje prilikom nepravilnog punjenja/pražnjenja. Baterija ima dva konektora, pri čemu je jedan za punjenje, a drugi za napajanje robota. Konektor punjenja je oznake JST-XH, koji je standard u robotici kao i u tehnici dronova i ostaloj hobi elektronici, konektor napajanja robota je oznake XT60, koji je također standard u ovom području. Prilikom odabira baterije bitno je znati autonomiju robota koju se želi postići, dok radni napon najčešće diktira odabrana elektronika. Navedena baterija je izabrana zbog dobrih svojstava, cijene i kompaktnosti.



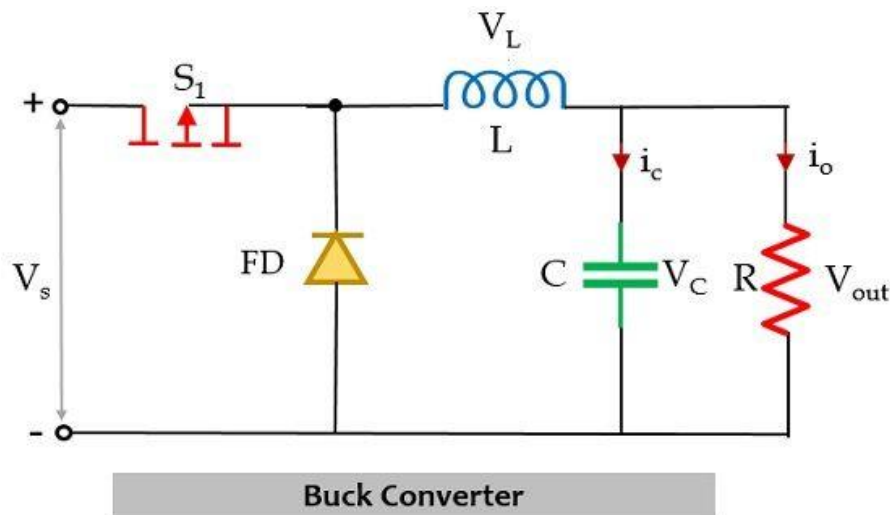
Slika 23. Prikaz strukture baterije [22]



Slika 24. Prikaz oštećene baterije [23]

6. Napajanje – Napajanje robota se u osnovi sastoji od tiskane pločice dimenzija 35 X 35 mm ulaznog napona 6-16 V, sastavljene od jednog ulaznog konektora standarda Molex KK396, tri izlazna konektora, pokazivača napona napajanja te prekidača za paljenje i gašenje napajanja (Slika 27.). Kao i sve ostale elektroničke komponente, na robota se postavlja uz pomoć odstojnika profila M2 i plastičnih vijaka istog profila. Prvi izlazni konektor istog standarda, kao i ulazni, daje ulazni napon s baterije preko prekidača i njega se najčešće spaja na motore i druge potrošače kojima je potrebna veća količina energije, a druga dva predstavljaju dvije standardne naponske veličine 3.3 V (narančasti konektor) i 5 V (crveni konektor), koji su izvedeni standardnim konektorom Molex KK254 (slika 25.). Navedene

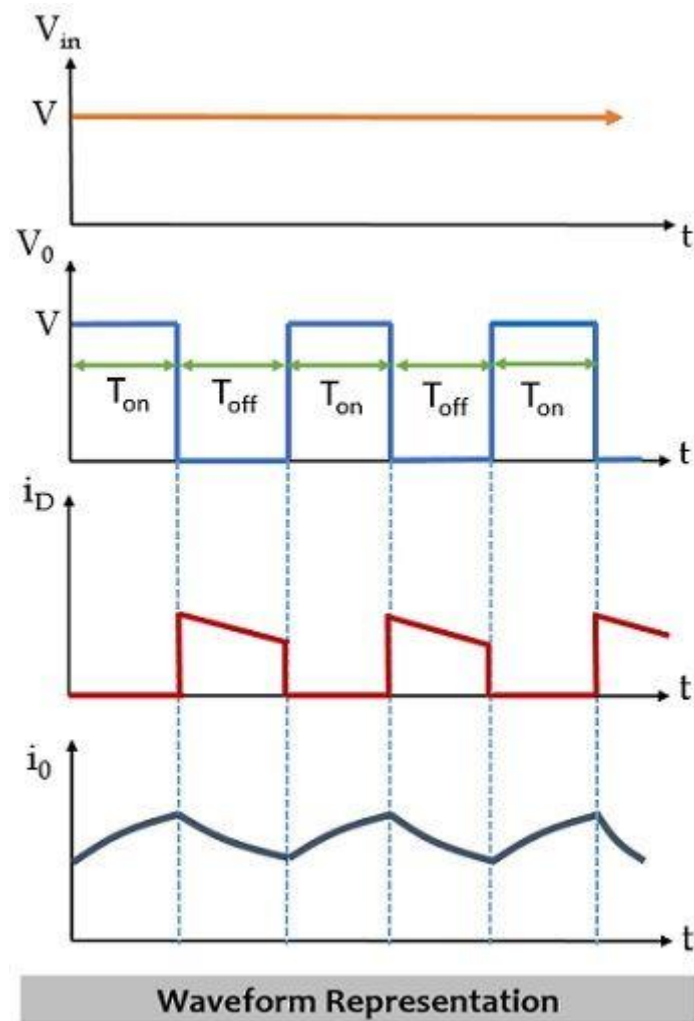
standardne konektore se koristi iz razloga modularnosti i mogućnosti jednostavne zamjene. Tipizirani su i na sebi imaju utore koji onemogućuju korisniku spajanje krivih polariteta, osim ako se ne koristi sila. Kako bi se postigli željene naponske razine, napajanje ima ugrađen prekidački Buck pretvarač (engl. *Switching step-down (Buck) converter*). Buck pretvarač je elektronički krug korišten za učinkovito smanjenje ulaznog napona na željene naponske nivoe, a općenito je vrlo široko korišten u aplikacijama regulacije napona i napajanja raznih elektroničkih uređaja. Isti koristi poluvodičke prekidačke elemente (Slika 25. oznaka S_1), najčešće MOSFET (engl. *Metal oxide semi-conductor field effect transistor*), kako bi vrlo velikom brzinom uključivao i isključivao ulazni napon te na taj način izlazni napon činio manjim. Slika 25. prikazuje pojednostavljenu shemu Buck pretvarača.



Slika 25. Prikaz pojednostavljene sheme Buck pretvarača [24]

Kao upravljački signal poluvodičke komponente koristi se pulsno - širinska modulacija (engl. *puls width modulation - PWM*) generirana od strane upravljačke jedinice. Kako bi se smanjili udari, odnosno kako ne bi postojali nagli prijelazi iz uključenog i isključenog stanja, ugrađuju se zavojnice (Slika 25. oznaka L) koje tijekom uključenog stanja skladište energiju, a potom tijekom isključenog stanja istu propuštaju kroz krug. Uz zavojnice, ključne komponente Buck pretvarača su i kondenzatori (Slika 25. oznaka C), kojima je zadaća zaglađivanje i stabilizacija izlaznog napona. Upravljačka jedinica Buck pretvarača upravlja tzv. radnim ciklusom (engl. *Duty cycle*) PWM signala koji određuje omjer uključenog i isključenog stanja, a podešavanjem radnog ciklusa pretvarač regulira napon. Radni ciklus i odziv na izlazu kod određenog radnog ciklusa može se vidjeti na prijemu

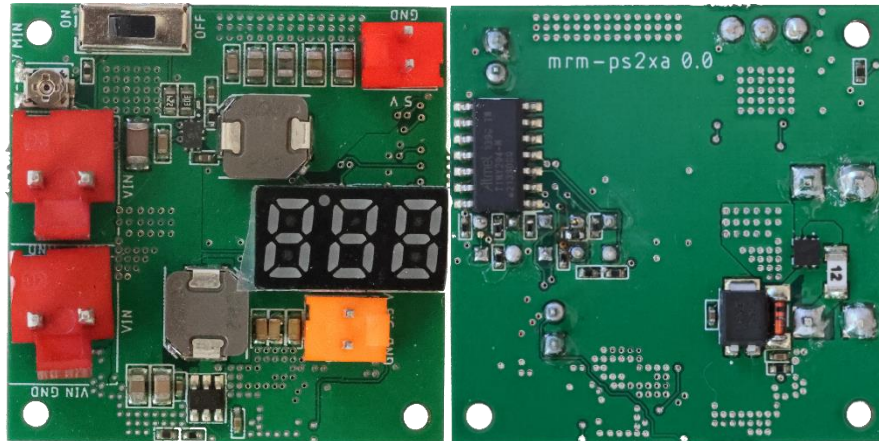
slike 26., pri čemu se na prvom grafu vidi ulazni napon, na drugom grafu radni ciklus 50%, a na grafovima 3 i 4 se vidi resultantna izlazna struja i izlazni napon. Ukoliko se obrati pažnja na graf 4, vidljiva je i funkcija kondenzatora i zavojnica u krugu koji zaglađuju izlazni napon. Upravljačka jedinica također kontinuirano nadzire izlazni napon te prema dobivenim parametrima dodatno prilagođava radni ciklus, kako bi izlazni napon regulatora bio što je moguće više stabilan.



Slika 26. Prikaz radnog ciklusa 50% i njegovog odaziva na izlazu [24]

Isti se smatraju vrlo efikasnim pretvaračima u odnosu na linearne pretvarače, prvenstveno iz razloga metode upravljanja izlaznim naponom koja uzrokuje manje gubitke i manje rasipanje energije, a učinkovitost može sezati i preko 90%, ovisno o dizajnu i vrsti opterećenja. Dobre izlazne karakteristike rezultirale su vrlo širokom primjenom gdje god je potreban stabilan i regulirani napon na izlazu pretvarača, a neki od primjera su računala, automobili, mobilni uređaji, audio oprema, kao i uređaji napajani nekom vrstom baterije

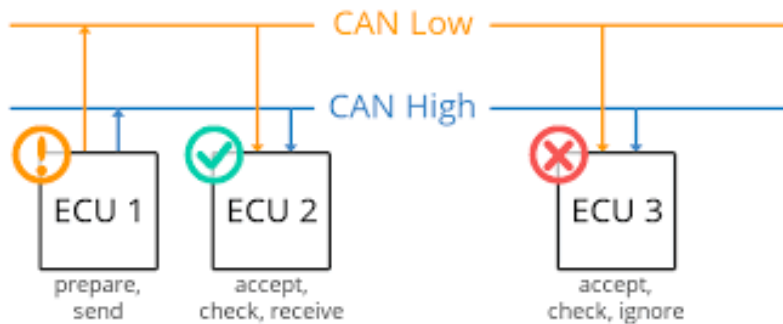
gdje je potrebno ulazni napon prilagoditi trošilu [24,25]. Na tiskanu pločicu su, uz Buck pretvarač, ugrađene zaštite od prevelike struje (ograničeno na 6.5A), SMD osigurač te diode koje sprječavaju propuštanje napona u krivom smjeru. Za upravljanje PWM signalom, s donje strane pločice ugrađen je Atmelov čip, koji je u osnovi mikro kontroler s upisanim programom za održavanje izlaznog napona na 3.3 V i 5 V. Navedene komponente, raspored konektora i fizički izgled pločice može se vidjeti na slici 27.



Slika 27. Prikaz pločice napajanja s prednje i stražnje strane [26]

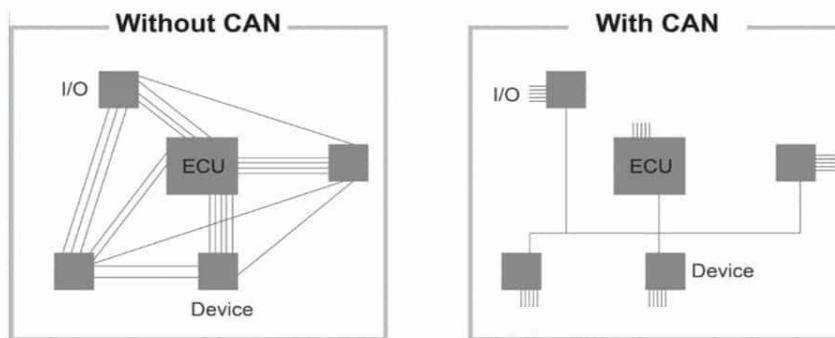
7. Mikro upravljač – Točan naziv mikro upravljačke pločice je Micro line ESP32, koji dolazi od naziva tvrtke koja je razvila pločicu i pripadajućeg fizičkog mikro upravljača. Predstavlja tzv. inteligenciju robota jer se putem sučelja u pločicu upisuje programski kod, prema kojem upravljačka pločica obrađuje podatke dobivene od senzora te ih pretvara u akciju nad aktuatorima. Važno je naglasiti da se radi o Arduino kompatibilnoj pločici. ESP32 je vrlo snažna mikro upravljačka pločica opremljena mnogim sensorima, WiFi i Bluetooth čipom namijenjenim za bežičnu komunikaciju, CAN bus prijemnikom/predajnikom (engl. *Trasceiver*) koji omogućuje CAN bus komunikaciju, Bosch BN0055 multifunkcijskim senzorom, ukupno 33 GPIO ulazno izlaznih (engl. *General purpose input/output*) pinova i micro USB konektorom za povezivanje mikro upravljača na računalo. ESP32 mikro upravljač konkretnog naziva ESP32-WROOM-32 opremljen je Xtensa dvojezgrenim 32 bitnim LX6 procesorom, napajan s 3.3 V, radnog takta 2 X 240 Mhz, memorije 448 KB ROM, 520 KB SRAM za podatke i instrukcije, eFuse zaštitom od obrnutog napona, prenapona, preniskog napona, obrnute struje i pregrijavanja, analogno - digitalnim pretvaračem kao i digitalno - analognim pretvaračem [28]. CAN bus (engl. *Conotroller Area Network bus*) ili CAN sabirnica je komunikacijski protokol koji omogućuje mikro

upravljačima i drugim uređajima jednostavnu međusobnu komunikaciju po principu mreže. Sustav je originalno razvijen od strane Roberta Boscha 1980. godine u svrhe proizvodnje automobila kako bi smanjio broj kabela i masu ožičenja u vozilima, međutim do danas je ova tehnologija pronašla svoju primjenu u mnogim drugim granama zbog svoje pouzdanosti, robusnosti kao i generalnih mogućnosti. CAN sabirnica koristi serijski komunikacijski protokol diferencijalne signalne sheme, pri čemu se za komunikaciju koriste dvije žice CANH (engl. *CAN High*) i CANL (engl. *CAN Low*) odnosno tzv. visoka i niska te postupak dekodiranja informacije bez povratka na nulu (engl. *Non-return-to-zero - NRZ*), pri čemu više čvorišta (engl. *Node*) može putem iste sabirnice istovremeno komunicirati sa upravljačkom jedinicom. Slika 28. prikazuje način vezanja pojedinih jedinica na sabirnicu te se mogu vidjeti prethodno spomenute dvije žice CANH i CANL [27].

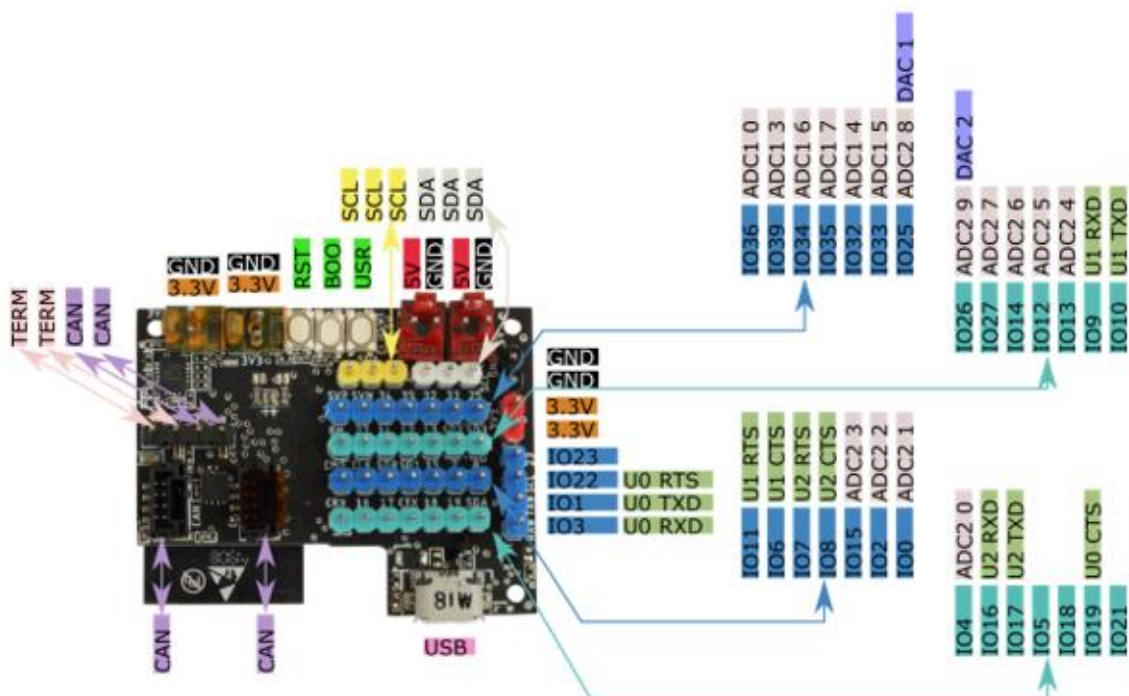


Slika 28. Prikaz CAN sabirnice [26]

Koristi se tzv. multi-drop topologija, koju je najjednostavnije objasniti kao serijsku vezu svih komponenti u sustavu koje mogu poslati informaciju istovremeno, a informacija je adresirana jedinstvenom oznakom - identifikacijom pojedine komponente. Identifikacija komponente, uz slanje informacije, omogućuje i primanje informacije poslana od mikro upravljača. Svaka poruka se, osim identifikacije, sastoji i od prioriteta i stvarne informacije. Uz identifikatore, protokol koristi i razne zaštite poruka kako se iste ne bi pomiješale, kao i sustave detektiranja pogreške. CAN sabirnica u ovom slučaju omogućuje pouzdanu i brzu vezu svih komponenti s mikro upravljačkom jedinicom uz maksimalno reduciranje ožičenja [27]. Usporedbu količine utrošenih žica za ožičenje nekog sustava jasno prikazuje slika 29., na kojoj se vidi znatno reduciranje količine žica u desnom sustavu koji je opremljen CAN sabirnicom.

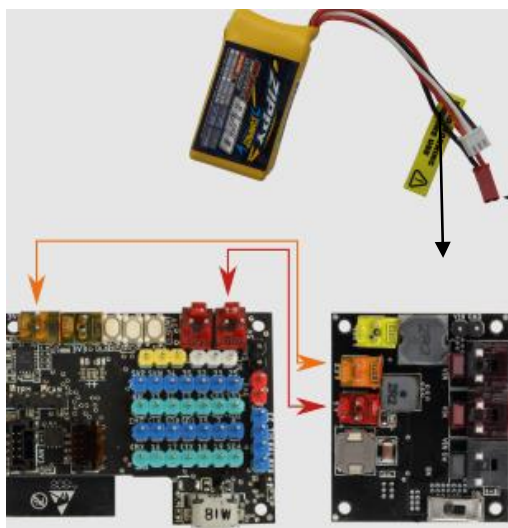


Slika 29. Prikaz sustava sa i bez CAN sabirnice [27]



Slika 30. Prikaz pinova mikro upravljača [28]

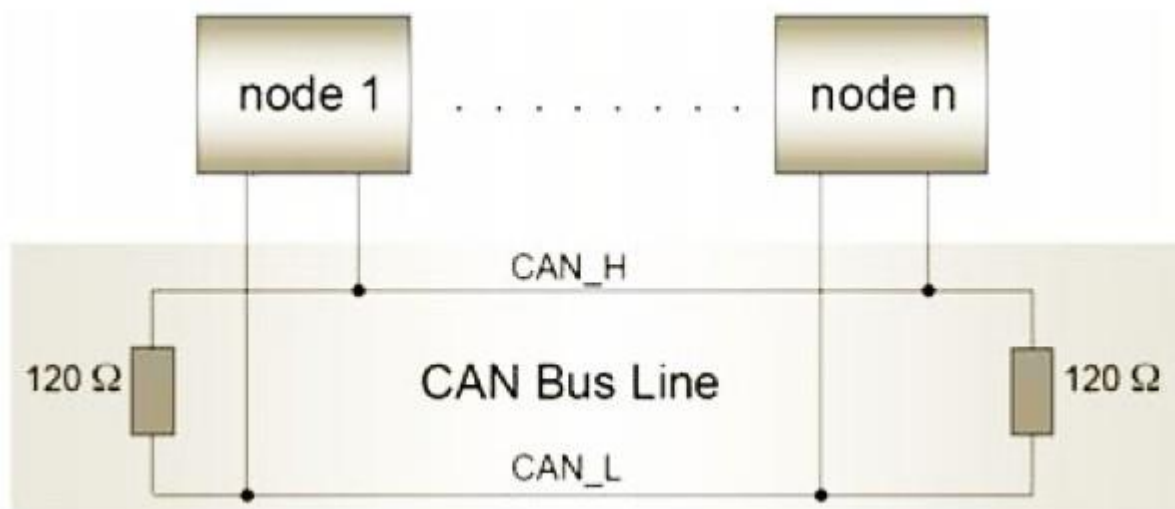
Pinovi i konektori su označeni na slici 30., pri čemu su glavne grupe konektori ulaznog napona 3.3 V (dva MOLEX KK254 standardna priključka) i 5 V, CAN sabirnički konektori (dva 4-pin JST CAN konektora), USB konektor, tri tipkala od kojih RST resetira pločicu, BOOT uzemljuje GPIO 0 pin kada je pritisnuto, a treće tipkalo je slobodno i može se programirati prema potrebi korisnika. Preostali Dupont standardni pinovi su spojeni na pripadajuće ulazno izlazne pinove ESP 32 mikro upravljačke jedinice, a pločica također ima ugrađenu led diodu koja se pali zeleno priključenjem pločice na napon [28].



Slika 31. Prikaz osnovnog spoja mikro upravljačke pločice [28]

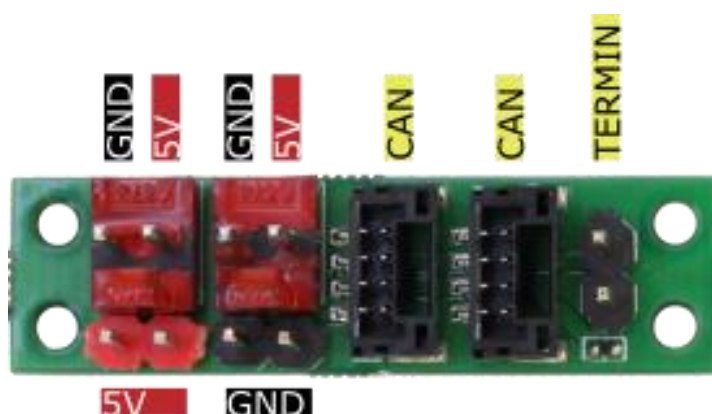
Za sam rad pločice potrebno je spojiti samo 3.3 V na jedan od narančastih konektora i pločica će "pokazati znakove života", dok se 5 V napajanje (crveni konektor) koristi ukoliko se želi koristiti CAN sabirnica [28]. Ispravan način za napajanje pločice je onaj prikazan slikom 31., no prikazani način nije jedini način za isto. Važno je napomenuti kako pločica ne radi ako je uključen samo USB konektor na računalo, kao što je to uobičajeno kod ostalih mikro upravljačkih jedinica, već je potrebno dovesti napajanje na 3.3 V konektor. Svi konektori kako napajanja, tako i CAN sabirnice, dolaze u paru, što je nastalo iz ideje serijskog povezivanja komponenti u cjelinu pa jedan konektor predstavlja ulaz, a drugi izlaz [28].

8. CAN bus napajanje (Slika 33.) – Služi za napajanje i terminiranje CAN sabirnice, dimenzije pločice su 35 X 10 mm, a na robota se pričvršćuje držačem, odstojećima i četiri plastična vijka standardnog profila M2. Uz napajanje sabirnice, stabilnim naponom od 5 V i strujom 1 A omogućuje se terminiranje sabirnice kratkim spajanjem nožica terminatora (Slika 33.) na pločici, čime se u krug spaja 120 Ohm otpornik. Terminiranje je vrlo bitno odraditi kako bi se ostvarila kvalitetna i pouzdana CAN sabirnica [30]. Prikaz kvalitetno odradenog terminiranja moguće je vidjeti na slici 32.



Slika 32. Terminiranje sabirnice na njezinom početku i kraju [29]

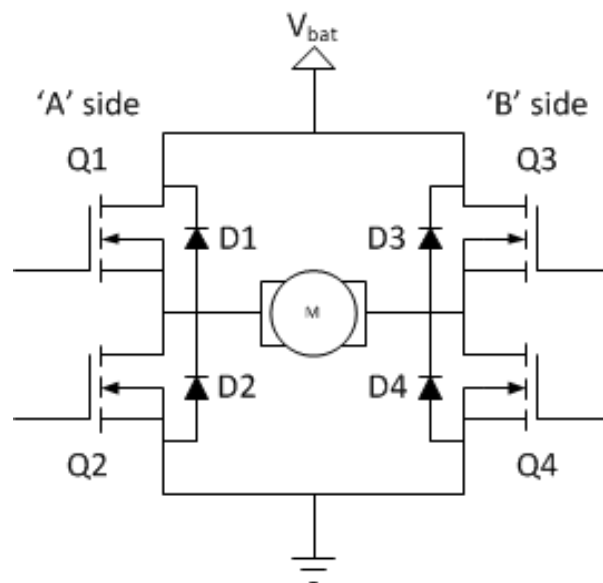
Ukoliko se preskoči korak terminiranja sabirnice, ubrzo bi se pojavili problemi reflektiranja valova, kao i vrlo loših performansi sabirnice, što bi posljedično uzrokovalo postupno otkazivanje pojedinih senzora kako bi se sabirnica rasteretila. Korištenjem napajanja CAN sabirnice moguće je u mrežu povezati do 120 čvorišta, odnosno senzora ili drugih uređaja, dok je bez napajanja moguće spojiti samo 12. U teoriji je u sabirnicu moguće povezati gotovo neograničen broj uređaja, no u praksi je priča drugačija jer s duljinom sabirnice pada brzina prijenosa podataka, a problem nastaje i s opskrbljivanjem sabirnice naponom. Stoga je uputno koristiti CAN napajanje kako bi se izbjegli navedeni problemi [30].



Slika 33. Prikaz pločice napajanja CAN sabirnice [30]

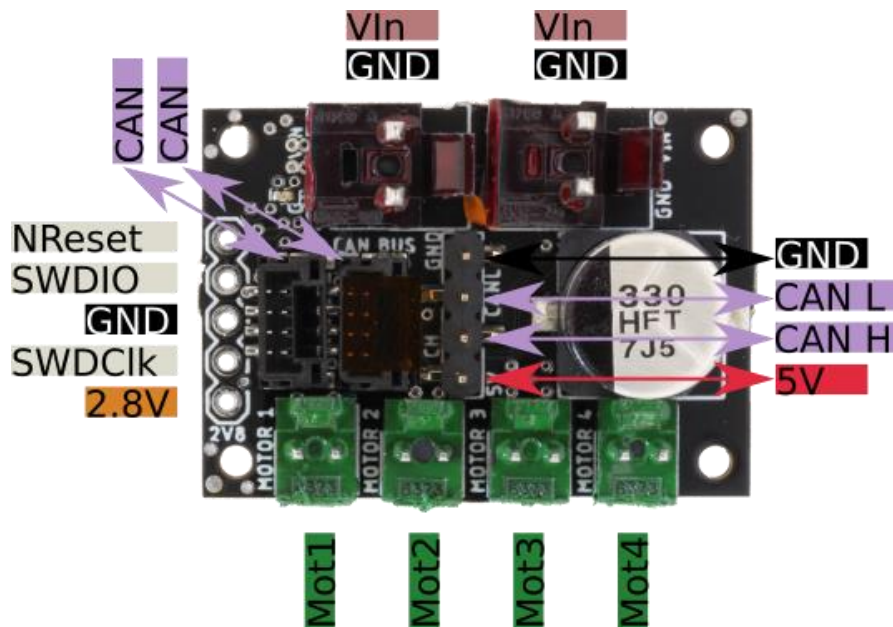
Na slici 33. prikazana je pločica napajanja CAN sabirnice s pripadajućim opisima pinova. Pločica se također može koristiti u svrhu repliciranja signala ukoliko se koristi više od 120 čvorišta, što je specifično za jako velike sustave.

9. Upravljač motora – Korišteni upravljač je namijenjen za pogon četiri mala motora s četkicama vršne izlazne struje 4 X 3.6 A, dok je napon motora 6.5 – 45 V. Upravljan je mikro kontrolerom CAN sabirnicom, a sastoji se od čipova TI DRV8871, ARM M0 mikro upravljačkog čipa, kao i od zaštite od prenapona, prevelike struje, obrnutog napona i pregrijavanja. ESP 32 upravlja ARM M0 čipom koji potom upravlja radom TI DRV8871 čipom slanjem PWM signala na njegove ulazne pinove [32]. TI DRV8871 je čip upravljača motora s ugrađenim H mostom, namijenjen velikoj skupini uređaja koji koriste istosmjerne motore s četkicama. H most je električni krug koji omogućuje upravljanje smjerom vrtnje motora. Sastoji se od četiri poluvodičke sklopke (tranzistora ili MOSFET-a), povezane na način da strujni krug nalikuje slovu H, od kuda potječe i naziv sklopa. Naizmjeničnim uključivanjem i isključivanjem sklopki određuje se smjer protoka struje kroz strujni krug, a time i posljedično smjer rotacije motora što se može vidjeti na slici 34. Primjerice, ukoliko se uključi gornju desnu (Q1) i donju lijevu (Q4) sklopku, a druge dvije su isključene, postiže se jedan smjer vrtnje, dok s druge dvije sklopke postizemo drugi smjer vrtnje. Osim mogućnosti promjene smjera vrtnje, H most omogućuje i regulaciju brzine vrtnje motora primjenom već spomenute PWM regulacije te na taj način mijenjanja vremena uključenosti pojedine sklopke. Pri korištenju H mosta, potrebno je pripaziti na eventualne krive smjerove protoka struje, stoga su iz tog razloga u većini modernih čipova unaprijed ugrađene razne zaštite, poput dioda prikazanih na slici koje služe za blokiranje protoka struje u pogrešnom smjeru, zaštite od prevelike struje i preopterećenja kruga [31].



Slika 34. Prikaz sklopa H mosta [31]

Dimenzije pločice su 35 X 25 mm, a na robot se pričvršćuje uz pomoć četiri M2 vijka i četiri odstojnika. Prikaz pločice s pripadajućim pinovima vidi se na slici 35. Ponovo su korišteni standardni konektori KK396, KK254 i JST za priključenje motora, napajanja i CAN sabirnice.



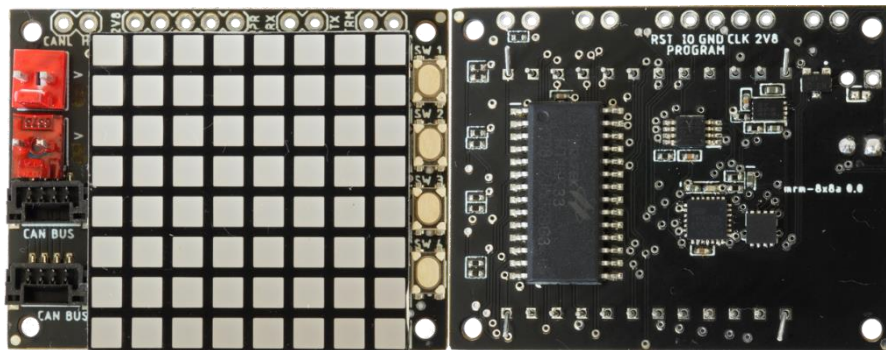
Slika 35. Prikaz pločice upravljača motora [32]

10. Distribucijska pločica – Predstavlja najjednostavniju elektroničku pločicu na cijelom robotu, a namjena joj je umnožavanje broja mogućih naponskih konektora. U ovom slučaju se radi o pet konektora 3.3 V i pet konektora 5 V, što znači da ukoliko se spoji jedan konektor kao ulaz, preostaju još četiri konektora na koja se mogu spojiti sve potrebne komponente kojima treba dati navedene naponske nivoe. Izrađena je s Molex KK254 standardnim konektorima, a na osnovu robota se montira pomoću četiri odstojnika i četiri plastična vijka profila M2, pri čemu su dimenzije pločice 35 X 10 mm [33]. Prikaz distribucijske pločice može se vidjeti na slici 36.



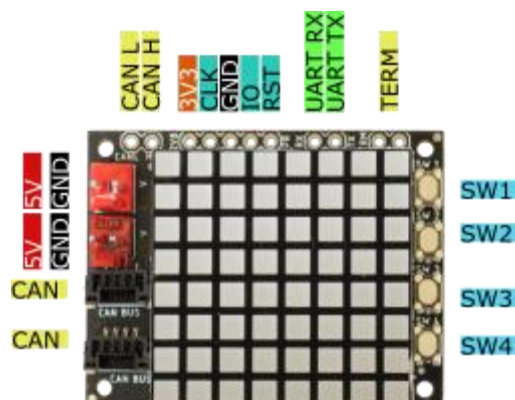
Slika 36. Prikaz distribucijske pločice [33]

11. 8 X 8 LED matrica – Pločica dimenzija 35 X 45 mm koja se u osnovi sastoji od polja LED dioda u dvije boje (crvena i zelena), koje omogućuju i treću boju (narančasta) ukoliko se upale obje LED diode istovremeno. Za komunikaciju s mikro upravljačem koristi CAN sabirnicu, a za napajanje 5V preko standardnog KK254 konektora. Pločica ima ugrađene i četiri sklopke koje se mogu programirati prema potrebama korisnika. S stražnje strane nalazi se upravljač koji upravlja radom ukupno 64 LED diode, a ugrađene su i zaštite od obrnutog napona i elektrostatskog elektriciteta [34].



Slika 37. Prikaz pločice led matrice [34]

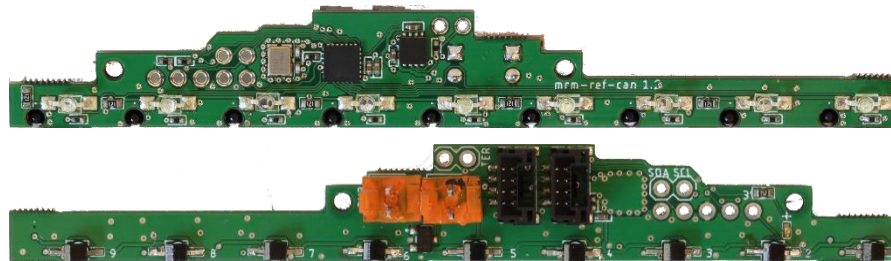
Naizmjeničnim paljenjem i gašenjem određenih skupina LED dioda omogućeno je prikazivanje raznih simbola, teksta te gotovo bilo čega kako bi se uspostavila svojevrsna komunikacija s robotom, što može uvelike olakšati razvoj robota. Fizički izgled pločice s prednje i stražnje strane može se vidjeti na slici 37., na kojoj se vide svi dostupni konektori, tipkala kao i sama LED matrica [34].



Slika 38. Prikaz pozicija pinova na pločici [34]

Ukoliko se obrati pažnja na sliku 38., može se vidjeti kako postoje i dodatni pinovi koji su namijenjeni direktnom lemljenu žica na pločicu te je na taj način je omogućen alternativni pristup pločici ukoliko se iz nekog razloga želi ili treba izbjeći standardne konektore, odnosno KK254 za napajanje i JST konektor za CAN sabirnicu .

12. 9X infracrveni senzor – Infracrveni senzor se koristi za detekciju linije, a sastoji se od 7 digitalnih i dva analogna infracrvena senzora smještena na krajevima pločice. Osim detekcije svijetlih i tamnih površina, konkretan senzor je sposoban prepoznati i srebrnu boju. Preporučena udaljenost od podloge za ispravan rad senzora je 1 – 3 cm, dok su rezultati u rangu od 3 – 4 cm još uvijek provjereno iskoristivi [39]. Fizički izgled, konstrukcija, konektori i pinovi prikazani su na slici 39.

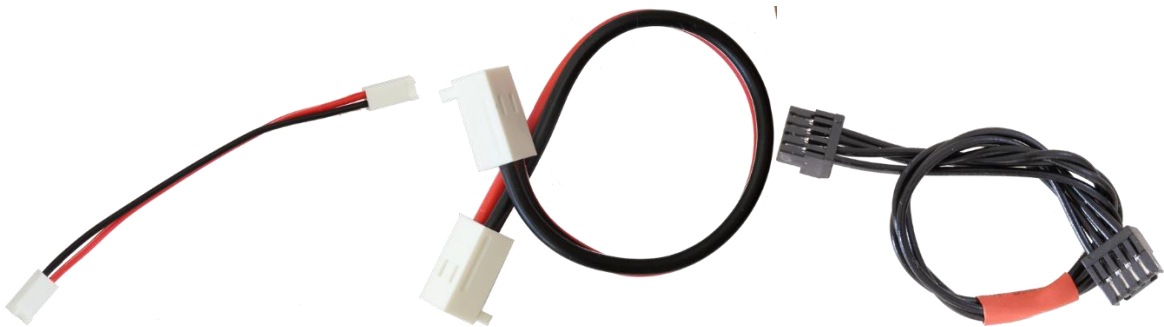


Slika 39. Prikaz infracrvenog senzora [35]

Senzor koristi 3.3V napajanje i CAN sabirnicu za komunikaciju s mikro upravljačem, dok se na robot postavlja upotrebom držača, odstojnika i plastičnih vijaka profila M2. Za upravljanje radom senzora zadužen je ARM mikro upravljač koji uspostavlja komunikaciju s glavnim mikro upravljačem i usklađuje pojedine senzore u grupu. Za razliku od udruživanja pojedinih infracrvenih senzora, ovaj senzor je optimiran tako da radi kao grupa što omogućuje ujednačenje jakosti snopa svjetlosti i zajedničku kalibraciju pa posredno i točnija i preciznija očitavanja, kao i veću pouzdanost. Promatranjem slike 39. također je moguće uočiti primjenu standardnih priključaka za CAN sabirnicu i napajanje, a uz to postoje i izvodi za terminiranje CAN sabirnice preko ugrađenog otpornika otpora 120 Oma, koji se koriste ukoliko senzor predstavlja kraj sabirnice.

13. Ožičenje – Jednako važno kao i svi prethodno navedeni elementi robota jest ožičenje. Igra ključnu ulogu jer opskrbljuje sve komponente robota naponom, kao i upravljačkim signalima. Za izradu robota potrebna su ukupno četiri standardna konektora, a to su redom

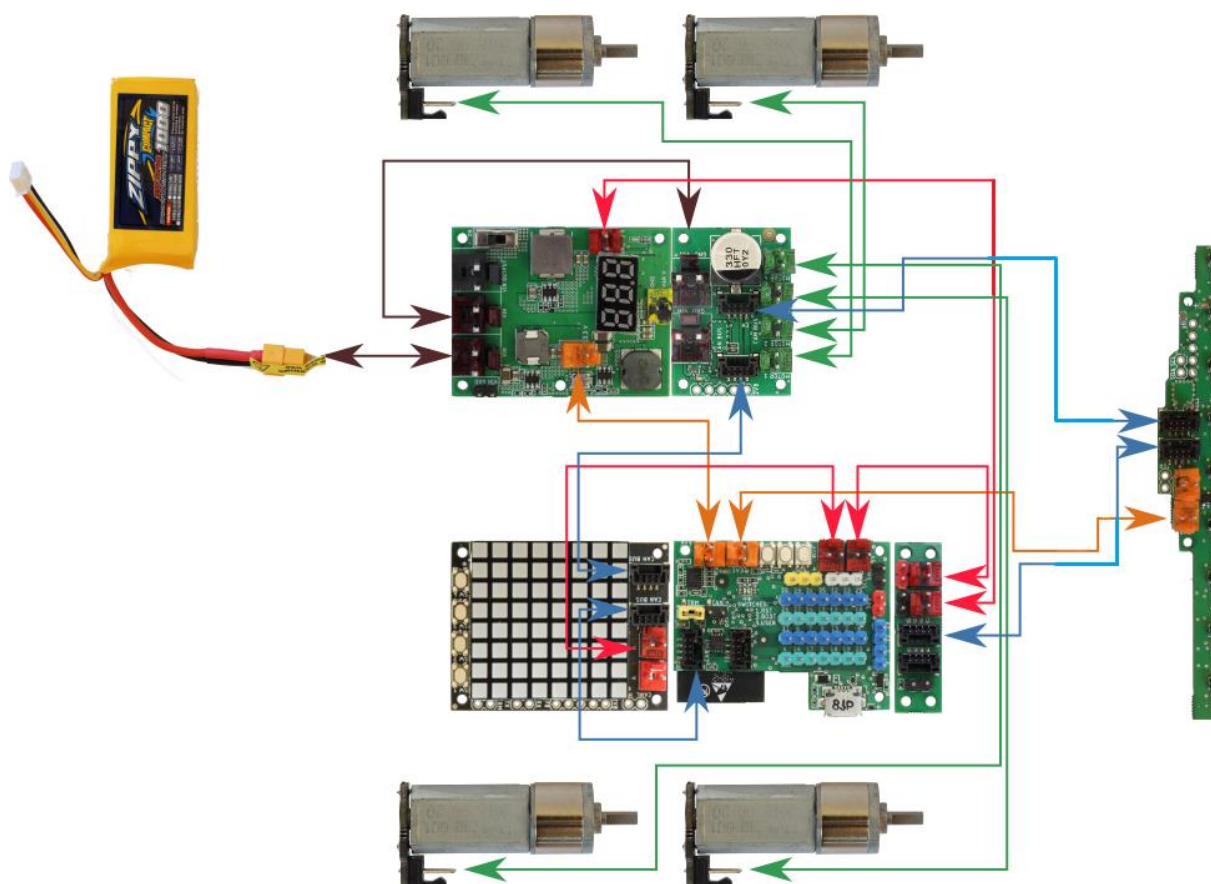
od baterije prema zadnjoj komponenti, odnosno senzoru - XT60, Molex KK254, Molex KK396 i JST konektor. Uz navedene konektore, na robotu postoje dva različita naponska (3.3 V i 5 V) i jedan komunikacijski vod (CAN sabirnica). Slika 40. prikazuje konektore pri čemu je vidljivo kako su KK254 i KK396 vrlo slični, a razlika je u veličini konektora, pri čemu je KK254 manji i namijenjen je za manja opterećenja (uglavnom 3.5 V i 5 V), dok je KK396 namijenjen 12V napajanju komponenata. CAN sabirnica koristi JST konektor s četiri žice, pri čemu su dvije namijenjene visokom i niskom vodu, a druge dvije s vanjskih strana napajanju CAN sabirnice.



Slika 40. Prikaz korištenih standardnih konektora korištenih u izradi robota [36]

Detaljan prikaz svih spojeva robota vidljiv je na slici 41. gdje su različiti vodovi označeni različitim bojama: smeđa – 12V napajanje, crvena – 5V napajanje, narančasta – 3.3V napajanje, zelena – vodovi napajanja motora i plava – vodovi CAN sabirnice.

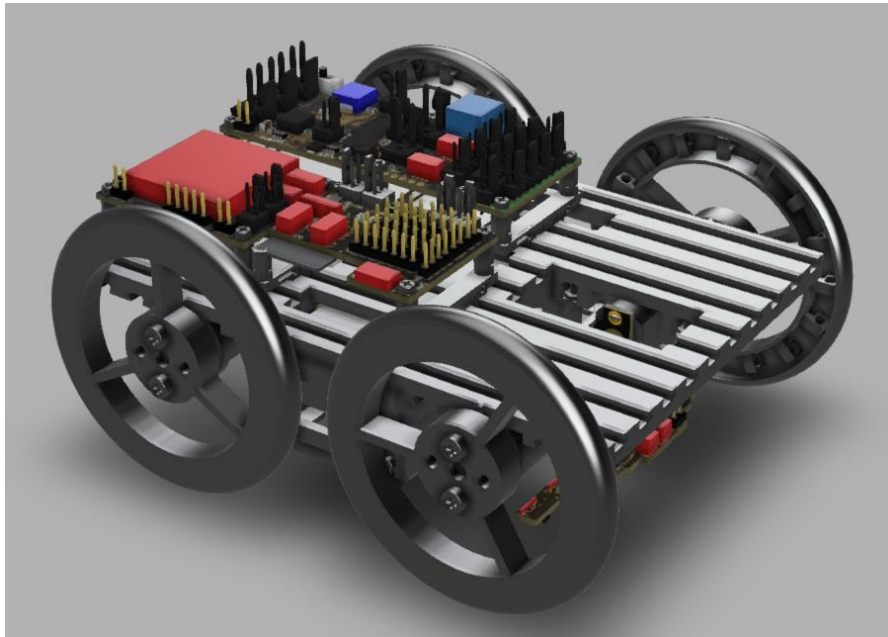
Prikazan način spajanja nije jedini način, međutim potrebno je obratiti pažnju na naponske nivoe kako bi se izbjegla trajna oštećenja pojedinih komponenata. Vidljivo je kako svaka pločica, osim napajanja, ima par korištenih konektora, pri čemu je ideja da je jedan ulazni, a jedan izlazni što omogućuje nizanje komponenti i jednostavniju organizaciju ožičenja.



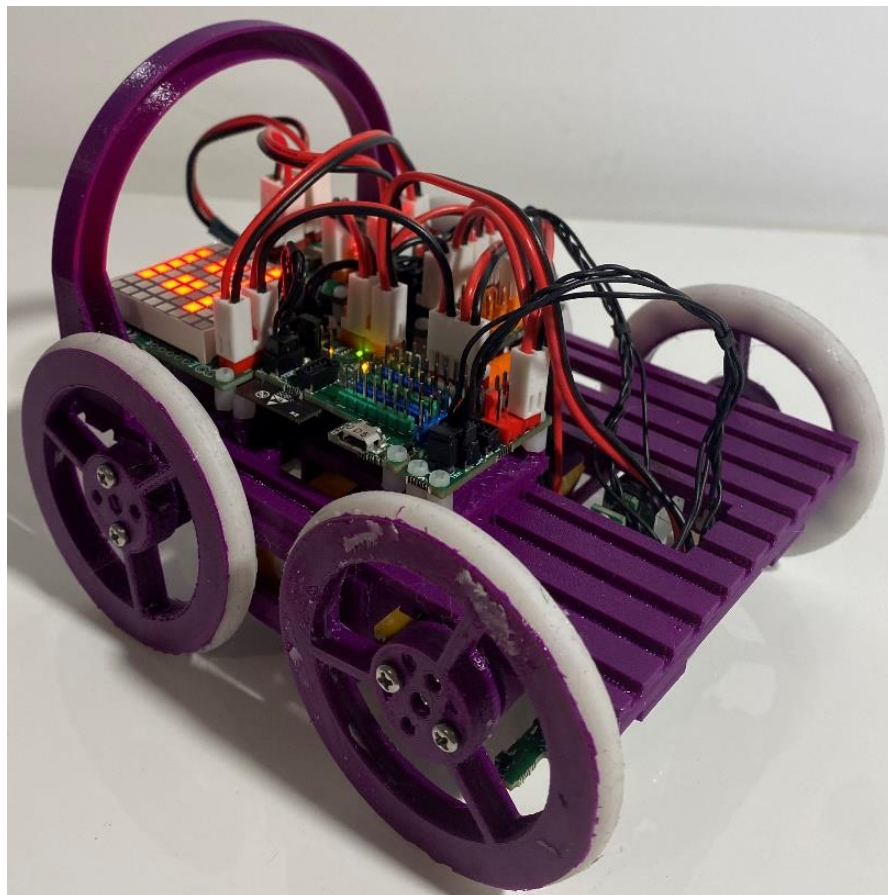
Slika 41. Prikaz sheme ožičenja robota [37]

6.2.1. Kompletan robot

Izradom šasijske nosače komponenti, postavljanjem motora na opisani način, kotača, baterije, komponenti i međusobnim ožičenjem, postiže se konačni cilj izrade fizičkog robota, kojeg je potom uz pomoć računalnog sučelja i programiranja potrebno "naučiti" da radi ono za što je namijenjen. Sve komponente imaju ugrađene zaštite, ali svejedno je prije puštanja robota u rad potrebno dodatno provjeriti sve veze kako bi se uklonile eventualne pogreške nastale brzopletošću ili bilo kojom drugom ljudskom pogreškom te kako bi se izbjegla mogućnost eventualnog kvara komponenti. Ukoliko je sve spojeno prema shemi, robot treba davati znakove "života", odnosno signalne lampice napajanja komponenti i LED matrica bi se trebali upaliti. Potpuno sastavljenog robota može se vidjeti na 3D prikazu modela robota (Slika 42.) i slici fizičkog robota (Slika 43.)



Slika 42. Prikaz 3D modela robota [38]



Slika 43. Sastavljeni robot [Izradio autor]

6.3. Programski kod

Za programiranje robota korišteno je Arduino IDE (engl. *Integrated Development Environment* - IDE), razvojno sučelje koje je bazirano na C++ programskom jeziku. Vrlo je jednostavno za korištenje i prilagođeno jednostavnom snalaženju korisnika u programu. Kako bi bilo izvedivo programirati ESP32 mikro upravljač, potrebno je učitati njegove inicijalne naredbe tzv. knjižnice (engl. *library*), a osim za mikro upravljač potrebno je također i učitati knjižnice za CAN sučelje, kao i sve ostale komponente korištene na robotu čije knjižnice su dostupne. Po završetku pravilne instalacije potrebno je pokušati uspostaviti vezu s mikro upravljačem, što je najjednostavnije priključenjem istog putem USB sučelja na računalo, a ako je veza uspostavljena zadovoljeni su svi preduvjeti te se može početi s pisanjem koda.

6.3.1. Opis programa

Kod konkretnog primjera rješavanja labirinta, većina koda se bazira na kodu za praćenje linije, odnosno robota se uči da prati liniju na način da srednji senzor konstantno vidi liniju, dok ostali služe poravnanju robota, odnosno vraćanju robota na sredinu linije. Pomoću ostalih senzora određuje se greška, koja se potom ispravlja bržim ili sporijim zakretanjem pojedinih kotača robota. Potom je potrebno izabrati željenu metodu rješavanja labirinta, koja može biti pravilo lijeve ili desne ruke, širinska metoda, dubinska metoda, Tremauxova metoda, kao i mnoge druge. Za rješavanje labirinta u ovom slučaju je odabrano pravilo lijeve ruke, čije načelo je da se robot stalno drži lijeve strane labirinta dok ne dođe do rješenja te staje kada dođe do rješenja.

6.3.2. Tijek programa

Program započinje pozivanjem potprograma i učitavanjem tzv. biblioteka.

```
#include <mrm-8x8a.h>
#include <mrm-mot4x3.6can.h>
#include "mrm-robot-line.h"
```

Zatim slijedi definiranje motora i definiranje funkcije 'go', koja naređuje robotu kretanje.

```
motorGroup = new MotorGroupDifferential(this, mrm_mot4x3_6can, 0, mrm_mot4x3_6can,
1, mrm_mot4x3_6can, 2, mrm_mot4x3_6can, 3);
```

```
mrm_mot4x3_6can->directionChange(0);
```

```

mrm_mot4x3_6can->directionChange(1);
mrm_mot4x3_6can->directionChange(2);
mrm_mot4x3_6can->directionChange(3);
void RobotLine::go(int16_t leftSpeed, int16_t rightSpeed) {motorGroup->go(leftSpeed,
rightSpeed);}

```

Definiranje tipkala broj tri da pokreće željeni program.

```
mrm_8x8a->actionSet(_actionLoop, 2);
```

Čitanje tipkala

```
bool RobotLine::button(uint8_t number) {return mrm_8x8a->switchRead(number);}
```

Definiranje i provjera rada LED matrice – pri definiranju LED matrice potrebno je isprogramirati željene bit mape tj. simbole koji će se prikazivati na LED matrici.

```

void RobotLine::bitmapsSet() {
mrm_8x8a->alive(0, true);
uint8_t red[8];
uint8_t green[8];

```

Definiranje infracrvenog senzora i kalibriranje u odnosu na svjetlinu podloge – navedena funkcija omogućuje ujednačeno očitavanje svih senzora.

```
uint16_t RobotLine::brightness(uint8_t transistorNumber) {return mrm_ref_can->reading(transistorNumber);}
```

Programiranje senzora da traži liniju – ova funkcija provjerava prisutnost linije na bilo kojem od senzora, a potom se idućom linijom upisuje broj senzora koji je detektirao liniju i pokušava pronaći centar, kao i centrirati robota na liniji.

```

bool RobotLine::lineAny(uint8_t firstTransistor, uint8_t lastTransistor) { return mrm_ref_can->any(true, 0, firstTransistor, lastTransistor);}
bool RobotLine::line(uint8_t transistorNumber) {return mrm_ref_can->dark(transistorNumber);}
float RobotLine::lineCenter() {
return (mrm_ref_can->center() - 5000) / 80.0;}
void RobotLine::lineFollow() {

```

```

static uint32_t ms = 0;
if (setup()){
    ms = millis();
    armUp();
}
if (lineAny())
    ms = millis();
if (line(8))
    go(-90, 90);
else if (line(0))
    go(90, -90);
else if (line(7))
    go(0, 100);
else if (line(1))
    go(100, 0);
else if (line(6))
    go(30, 80);
else if (line(2))
    go(80, 30);
else if (line(5))
    go(40, 70);
else if (line(3))
    go(70, 40);
else if (line(4))
    go(70, 70);
else
    if (millis() - ms > 100)
        stop();
    else
        go(70, 70);

```

Pronalaskom linije i centriranjem robota prema liniji, preostalo je isprogramirati pravilo lijeve ruke, što rješavaju navedene linije.

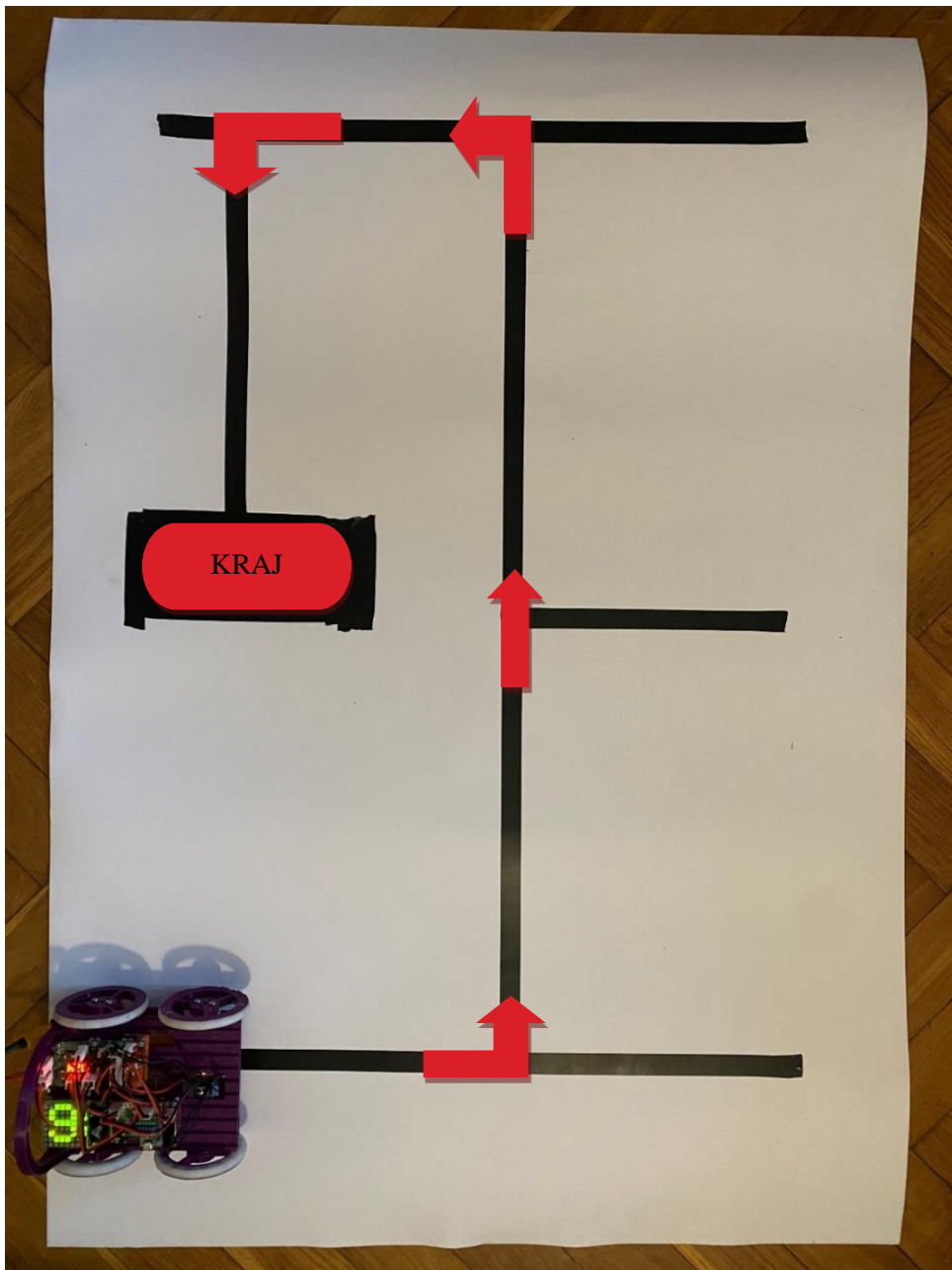
```

void RobotLine::loop() {
  mrm_8x8a->rotationSet(LED_8X8_BY_90_DEGREES);
  bitmapsSet();
  if(lineAny(true)){
    if(line(0) && line(LAST_TRANSISTOR / 2) && line(LAST_TRANSISTOR)){
      go(0, 0);
      display(LED_EVACUATION_ZONE);
      delayMs(20000); → kraj programa
    }
    if (line(0) || line(LAST_TRANSISTOR)){
      go(40,40);
      delayMs(300);
      go(-60,60);
      display(LED_LINE_FULL_MARK_LEFT);
      delayMs(500); → skretanje po pravilu lijeve ruke
    }
    float lineCenterNow = lineCenter();
    go(lineCenterNow < 0 ? TOP_SPEED : TOP_SPEED - lineCenterNow * 8, lineCenterNow
    < 0 ? TOP_SPEED + lineCenterNow * 8 : TOP_SPEED);
    display(LED_LINE_FULL); → kretanje po liniji
  }
  else{
    go(50, -50);
    display(LED_OBSTACLE);
    delayMs(300); → slijepa ulica
  }
}

```

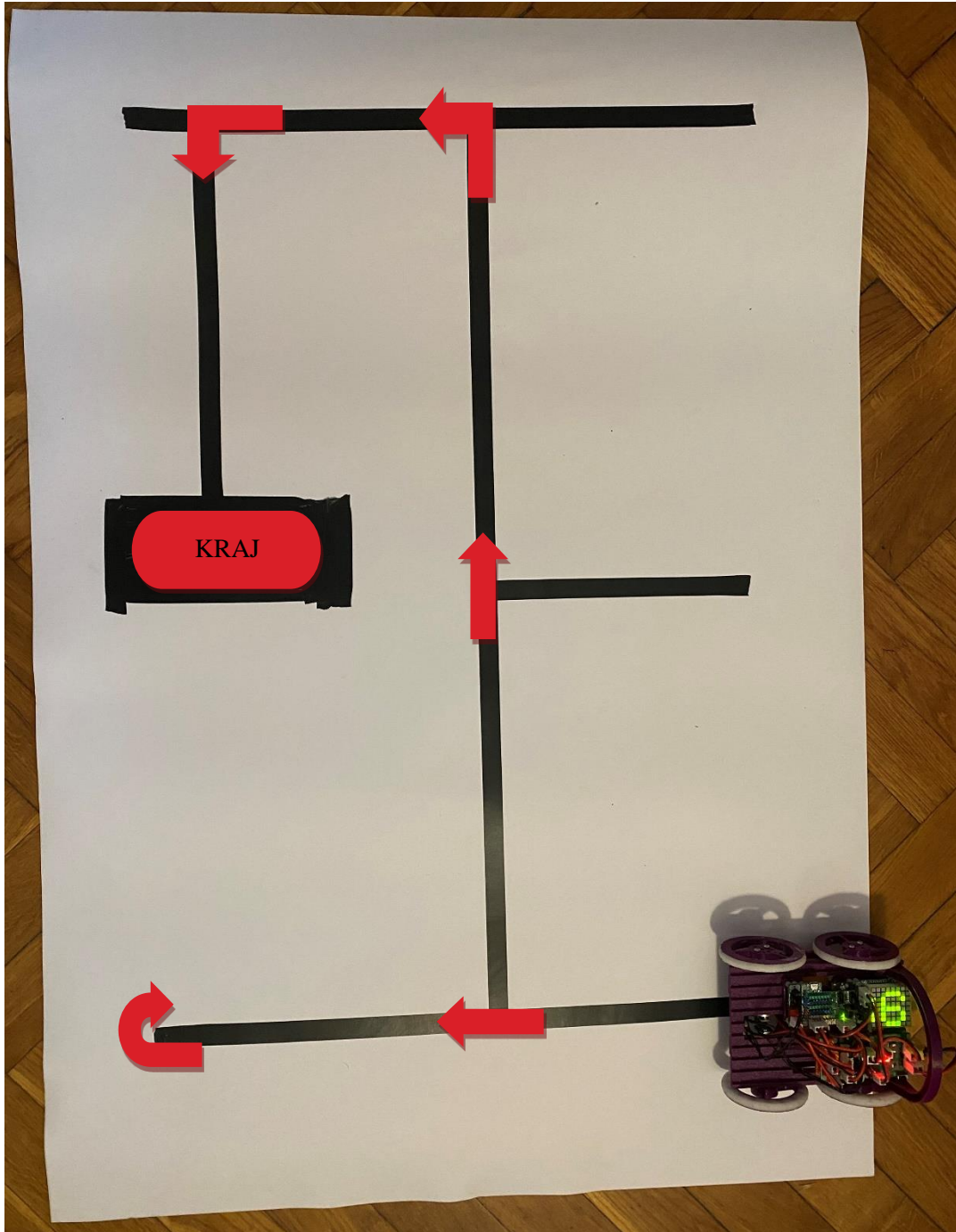
6.4. Rješavanje labirinta

Rješavanju labirinta se pristupa primjenom pravila lijeve ruke, na način da robot uvijek prati zid lijevom rukom te primjenjujući to pravilo bira skretanje na raskrižju. Ukoliko skretanje lijevo nije moguće, odnosno na raskrižju nema lijeve putanje, robot nastavlja ravno, a kao zadnja mogućnost ostaje skretanje desno. Jedan primjer primijenjenog pravila na konkretnom labirintu prikazan je na slici 44.



Slika 44. Prikaz toka rješavanja labirinta [izradio autor]

Kako bi se navedeno pravilo, kao i sposobnost robota da riješi labirint potvrdili, potrebno je robota pustiti kroz još minimalno jedan primjer labirinta, što je moguće vidjeti na slici 45.



Slika 45. Prikaz toka rješavanja labirinta [izradió autor]

7. ZAKLJUČAK

Obrada teme robota koji rješava labirint pruža uvid u fascinantne mogućnosti robotike i primjene algoritama u rješavanju kompleksnih problema poput ovog. Kroz rad je detaljno obrađena tehnologija, kao i algoritmi korišteni u izradi robota za navigaciju kroz labirinte različitih vrsta.

Razumijevanje i rješavanje labirinta zahtjeva kombinaciju senzora, algoritama za navigaciju i sposobnosti prilagodbe različitim situacijama u kojima se robot može naći prilikom rješavanja zadanog problema. Obrađeni su različiti pristupi rješavanju labirinta, uključujući i Lee-ov algoritam, metodu preplavlivanja, kao i metodu lijeve i desne ruke koja predstavlja najčešće korištenu metodu rješavanja jednostavnih labirinta.

Nakon detaljnog istraživanja, razvoja i testiranja, uspješno je razvijen robot koji je sposoban riješiti labirint. Robot je opremljen sensorima za otkrivanje okoline, motorima za kretanje i algoritmom za navigaciju. Tijekom testiranja, robot je pokazivao visoku uspješnost u pronalaženju izlaza iz labirinta. Sensori su omogućili precizno očitavanje linije, a algoritam sposobnost prilagodbe različitim vrstama labirinta kako bi robot pronašao put do rješenja.

Kroz praktični primjer i demonstraciju može se vidjeti sposobnost robota da detektira crtu i na temelju njezina uzorka donosi odluku o idućem koraku pri rješavanju labirinta. Ključni faktor uspjeha u izradi robota jest bio pažljivo planiranje, a također je i primijenjen princip iterativnog razvoja, što je podrazumijevalo brojna testiranja i precizno podešavanje robota.

Također se kroz poglavlje primjene u industriji može prepoznati prostor za napredak i primjenu predmetnih robota.

Sveukupno, rad naglašava važnost interdisciplinarnog pristupa pri obradi ove teme kako bi se uspješno izradio projekt ove vrste. Timski rad stručnjaka za robotiku, računalnu viziju, algoritme i strojno učenje ključan je za uspješan razvoj i napredak ove grane industrije.

Radovi poput ovog važni su za širenje znanja i promicanje interakcije među istraživačima, stručnjacima, entuzijastima i učenicima u području robotike.

LITERATURA

- [1] Intelligent Maze Solving Robot Based on Image Processing and Graph Theory Algorithms - Mohammad O.A. Aqel, Ahmed Issa, Mohammed Khdair, Majde ElHabbash, Mohammed AbuBaker, Mohammed Massoud, 2017.
- [2] An Autonomous Maze-Solving Robotic System Based on an Enhanced Wall-Follower Approach - Shatha Alamri, Hadeel Alamri, Wejdan Alshehri, 2022.
- [3] Maze solving robot with automated obstacle avoidance - Rahul Kumar, Peni Jitoko, Sumeet Kumar, Krishneel Pillay, Pratish Prakash, Asneet Sagar, Ram Singh, Utkal Mehta, 2016.
- [4] Autonomous Maze Solving Robotics: Algorithms and Systems - Shatha Alamri, Shuruq Alshehri, Wejdan Alshehri, Hadeel Alamri, Ahad Alaklabi, and Tareq Alhmiedat, 2021.
- [5] Survey on Techniques used in Autonomous Maze Solving Robot - Bhawna Gupta and Smriti Sehgal, 2020.
- [6] https://en.wikipedia.org/wiki/Maze-solving_algorithm, Maze-solving algorithm, Wikipedia
- [7] A Review on: Maze Solving Robot Using Arduino Uno - Khan Aqsa Sajid, Gawali Ashwini Sanjay, Patil Prajakta Shivraj, 2019.
- [8] Introduction to sensors - Bhagwati Charan Patel, G R Sinha and Naveen Goel, 2020.
- [9] <https://www.electronicshub.org/different-types-sensors/>, What is a Sensor? Different Types of Sensors and their Applications
- [10] <https://hr.wikipedia.org/wiki/Senzori>, Senzori, Wikipedia
- [11] <https://en.wikipedia.org/wiki/Sensor>, Sensors, Wikipedia
- [12] Image processing methodes for interactive robot control, Christoph Theis, Ioannis Iossifidis and Axel Steinhage, 2001
- [13] https://github.com/filippitlovic/Maze_Solver---AlphaBot2_Raspberry-Pi, Github
- [14] MATEMATIKA LABIRINATA, Marko Knežević, 2015 Zagreb
- [15] The Fastest Maze-Solving Competition On Earth, Veritasium, YouTube 2023.
- [16] <https://top.in.hr/Page.ashx?id=458>
- [17] <https://robu.in/brushed-dc-motor-working-principle-construction-applications/>
- [18] <https://www.adobe.com/creativecloud/file-types/image/vector/step-file.html>
- [19] https://www.microline.hr/HTML.ashx?_command=showItem&itemKey=mrm-bdc16150-100
- [20] https://www.microline.hr/HTML.ashx?_command=showItem&itemKey=mrm-hub3

- [21] https://en.wikipedia.org/wiki/Lithium_polymer_battery
- [22] https://www.researchgate.net/publication/318827556_Design_and_Control_of_Compact_legged-wheeled_robot_SPICAR/figures?lo=1
- [23] https://hobbyking.com/en_us/blog/how-to-tell-if-lipo-is-safe-or-not/?__store=en_us
- [24] <https://electronicscoach.com/buck-converter.html>
- [25] <https://www.fluke.com/en-us/learn/blog/electrical/what-is-duty-cycle>
- [26] <https://www.csselectronics.com/pages/can-bus-simple-intro-tutorial>
- [27] <https://911electronic.com/what-is-can-bus-protocol/>
- [28] https://www.microline.hr/HTML.ashx?_command=showItem&itemKey=mrp-esp32
- [29] <https://support.maxongroup.com/hc/en-us/articles/360009241840-CAN-bus-topology-and-bus-termination>
- [30] https://www.microline.hr/HTML.ashx?_command=showItem&itemKey=mrp-can-pow
- [31] <https://www.modularcircuits.com/blog/articles/h-bridge-secrets/h-bridges-the-basics/>
- [32] https://www.microline.hr/HTML.ashx?_command=showItem&itemKey=mrp-mot4x3.6can
- [33] https://www.microline.hr/HTML.ashx?_command=showItem&itemKey=mrp-distrib-f
- [34] https://www.microline.hr/HTML.ashx?_command=showItem&itemKey=mrp-8x8a
- [35] https://www.microline.hr/HTML.ashx?_command=showItem&itemKey=mrp-ref-can
- [36] <https://www.microline.hr/HTML.ashx?name=PriceList&key=kab>
- [37] <https://top.in.hr/Page.ashx?id=461>
- [38] <https://myhub.autodesk360.com/ue2d14232/g/shares/SH56a43QTfd62c1cd968ccc2b2d43416f48a>

KAZALO KRATICA

IEEE – engl. *Institute of Electrical and Electronics Engineers* – međunarodna organizacija za napredovanje tehnologije vezane s električnom energijom.

IDE – engl. *Integrated Development Environment* – razvojno sučelje

STEP - engl. *Standard for the Exchange of Product Data* – vrsta datoteke

CNC – engl. *Computer Numerical Control* – strojna obrada materijala

Li-Po - Litij – Polimer – vrsta baterije

PWM – engl. *Puls Width Modulation* – pulsno širinska modulacija

SMD – engl. *Surface Mount Device* - elektroničke komponente koje se ne postavljaju na klasičan način nego direktno na podlogu

CAN – engl. *Controller Area Network* – standard komunikacije među komponentama u elektronici

WiFi – engl. *Wireless networking technology* – standard za bežičnu komunikaciju

ROM – engl. *Random Access Memory* – radna memorija

SRAM – engl. *Static Random Access Memory* – statička radna memorija

RST – engl. *Restart* – Ponovno pokretanje

GPIO – engl. *General Purpose Input/Output* – ulazno/izlazna sučelja

USB – engl. *Universal Serial Bus* – standard za komunikaciju računala s periferijom

LED – engl. *Light Emitting Diode* – Dioda koja emitira svjetlo

POPIS TABLICA

Tablica 1 – algoritmi i njihove karakteristike [5]

Tablica 2 – usporedba pasivnih i aktivnih senzora [8]

Tablica 3. Popis dijelova [Izradio autor]

POPIS GRAFIKONA

Dijagram 1 – klasifikacija algoritama [4]

POPIS SHEMA

Slika 41. Prikaz sheme ožičenja robota [37]

PRILOG 1

Zbog specifičnosti projekta svi prilozi su objedinjeni u ovom linku i može ih se preuzeti

<https://filetransfer.io/data-package/gZxfcMYl#link>