

# Dizajn mikroupravljačkog sustava mobilnog robota

---

Raguž, Matej

Master's thesis / Diplomski rad

2016

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

*Permanent link / Trajna poveznica:* <https://um.nsk.hr/um:nbn:hr:200:970975>

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

*Download date / Datum preuzimanja:* **2025-02-02**

*Repository / Repozitorij:*

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



**SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU**

**FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I  
INFORMACIJSKIH TEHNOLOGIJA**

**DIZAJN MIKROUPRAVLJAČKOG SUSTAVA  
MOBILNOG ROBOTA**

**Diplomski rad**

**Matej Raguž**

**Osijek, 2016**

# SADRŽAJ

1. UVOD .....	1
2. ROBOTI.....	2
2.1 MOBILNI ROBOTI.....	6
2.1.1 Klasifikacija mobilnih robota.....	7
2.1.2 Komponente mobilnog robota.....	7
3. DIZAJN MOBILNOG ROBOTA .....	9
3.1 ZADATAK MOBILNOG ROBOTA.....	11
3.2 ALGORITAM ZADATKA.....	12
3.2.1 Grafički prikaz algoritma .....	13
3.4 ODABIR MIKROUPRAVLJAČA .....	19
3.5 ODABIR SENZORA .....	22
3.5.1 Princip rada digitalnog infracrvenog senzora s dva stanja.....	22
3.6 SKLOP POKRETAČA MOTORA.....	25
3.6.1 Parametar radni ciklus .....	26
3.6.2 Pulsno – širinska modulacija.....	27
3.7 ODABIR POBUDNOG SKLOPA MOBILNOG ROBOTA.....	31
3.7.1 Vrste baterija i primjena u mobilnom robotu .....	32
3.8 BLOK SHEMA MOBILNOG ROBOTA .....	35
3.9 REALIZACIJA MOBILNOG ROBOTA .....	36
3.9.1 Realizacija sklopovlja mobilnog robota.....	36
3.9.2 Realizacija programske podrške.....	38
4. TESTIRANJE MOBILNOG ROBOTA.....	41
4.1 NEDOSTATCI MOBILNOG ROBOTA I MOGUĆA POBOLJŠANJA.....	43
5. ZAKLJUČAK .....	45
LITERATURA.....	46

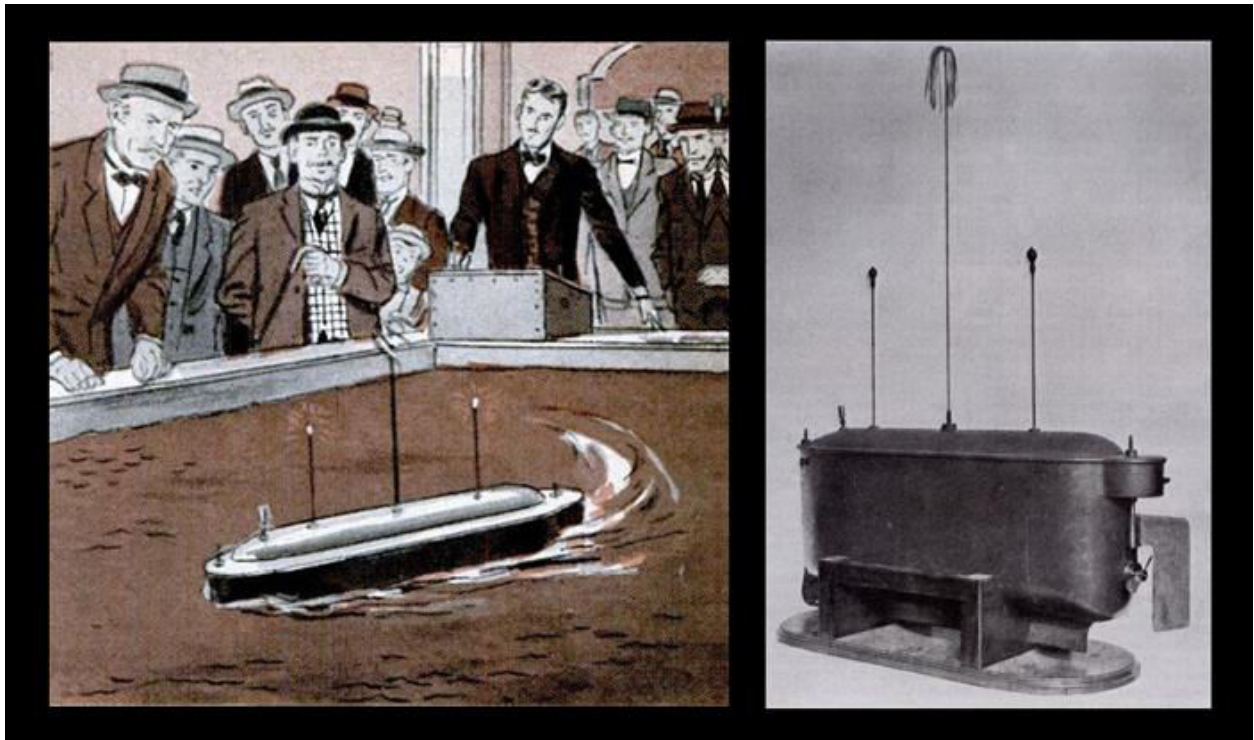
SAŽETAK.....	48
ABSTRACT .....	49
ŽIVOTOPIS .....	50
PRILOZI.....	51

# 1.UVOD

U ovom diplomskom radu će se obrađivati dizajn mikroupravljačkog sustava mobilnog robota. Kao uvod u diplomski rad, prvo se polazi od osnova te se opisuju termini „dizajn“ i „robot“. Prvo se započinje s opisom pojma „robot“ te se nakon toga pristupa pojmu „dizajn“. U trećem poglavlju je opisan proces dizajniranja nekog sustava koji je tu uveden kao obrazac po kojemu se vidi kako od ideje nastaje konačni proizvod (ili sustav). Svaki korak u procesu dizajna je obrađen i dan je primjer kao praktični dio ovog rada kako se postavljenog zadatka kojeg mobilni robot treba rješavati dolazi do konačnog proizvoda, u ovom slučaju mobilnog robota. Budući da je proces koji je dan u poglavlju 3. postavljen kao obrazac koji vrijedi u velikom broju slučajeva koji opisuju proces od ideje do konačnog proizvoda, u ovom diplomskom radu postoji varijacija tog procesa i neki koraci koji su uvedeni u tom obrascu nisu obuhvaćeni prilikom izrade mobilnog robota u sklopu ovog diplomskog rada. U drugom poglavlju je detaljno opisan pojam „robot“, povijest razvoja robotike kao grane znanosti i obrađene su klasifikacije robota. U trećem poglavlju je opisan proces dizajna sustava, gdje je također opisano kako je tekao proces razvoja mikroupravljačkog sustava mobilnog robota. Opisano je koje su komponente korištene kao i razlog njihovog korištenja. Obraden je princip rada svake komponente. Nakon odabira komponenti, pristupilo se razradi blok sheme koja vizualno prikazuje na koji su način povezane pojedine komponente mobilnog robota te je također opisana i interakcija tih komponenti. Nakon toga se pristupilo realizaciji mobilnog robota, a posljedično i testiranju rada. Testiranjem rada mobilnog robota je utvrđeno da mobilni robot posjeduje ograničenja koja su opisana u potpoglavlju 4.1, a u tom potpoglavlju su dani i prijedlozi za otklanjanje tih ograničenja i povećanja autonomnosti i funkcionalnosti mobilnog robota. Nakon toga se daje konačni zaključak rada i u prilogima se može pronaći i programski kod koji predstavlja implementaciju algoritma rada mobilnog robota koji je dan u trećem poglavlju. Programski kod je također dokumentiran i lako pregledan. Cilj diplomskog rada je osmisliti upravljački sustav mobilnog robota temeljen na mikroupravljaču te odabrati senzore i mikroupravljač na temelju postavljenih zahtjeva. Zatim je potrebno sačiniti blokovski dijagram i sheme spajanja i konačno osmisliti programsku podršku za povezivanje sklopovlja i osnovne funkcije.

## 2. ROBOTI

Robot je elektromehanički stroj koji izvodi određene radnje. Robotom može upravljati skup električnih krugova ili računalni program. Grana znanosti koja se bavi dizajnom, konstrukcijom, operacijom i primjenom robota te razvojem programske podrške za kontrolu robota, razvojem povratne veze između robotskih senzora i robota te procesiranjem svih informacija koje se pojavljuju tokom interakcija bilo kojeg entiteta unutar i van robota se naziva robotika. Robotika kao grana znanosti nastoji razviti robote kao autonomne strojeve do te mjere da mogu zamjeniti ljude za rad u okruženjima opasnim po život ili npr. jednostavno zamijeniti ljude unutar industrije tokom proizvodnog procesa. Roboti se proizvode tako da što više sliče ljudima ili određenim dijelovima tijela, kao i ponašanju i kognitivnim sposobnostima ljudi ili životinja. Mnogi od današnjih robota su izrađeni i isprogramirani tako da njihova funkcionalnost i fizički izgled oponašaju životinje u prirodi. Funkcionalnost robota i fizički izgled ovise prvenstveno o namjeni robota, uvjetima u kojima robot radi, a dakako treba napomenuti da tvorci robota uvelike crpe inspiraciju iz prirode. Takvi roboti pripadaju grani robotike koja se naziva biorobotika. Pojam „robot“ prema [1] je prvi put upotrijebljen za prikaz fiktivnog humanoida u drami češkog pisca Karela Čapeka iz 1920. godine pod nazivom *R.U.R.* Smatra se da je zapravo Karelov brat Josef pravi izumitelj tog pojma. Riječ robot na češkom jeziku znači „težak rad“. Prvi automatizirani stroj je sastavio dobro poznati američki znanstvenik s hrvatskim i srpskim podrijetlom Nikola Tesla. Svoj stroj je predstavio javnosti 1898. godine i nazvao ga prema [8] „*Teleautomaton*“. Taj stroj je bio mobilan i plovio je po vodi. Radiofrekvecijskom vezom je riješio problem upravljanja, gdje se sa posebnog sučelja upravljalo strojem koji je plovio po vodi. Taj stroj je prikazan na sljedećoj slici.



**Slika 2.1:** Nikola Tesla predstavlja javnosti prvi automatizirani uređaj 1898. godine koji je plovio po vodi (izvor: [9])

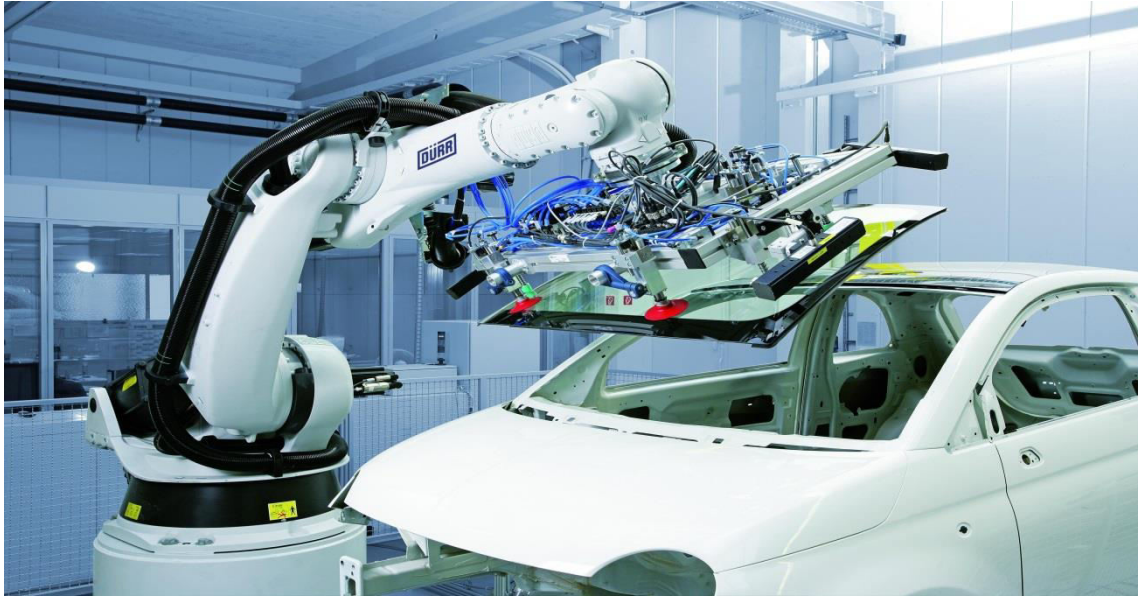
Nikolu Teslu neki znanstvenici smatraju začetnikom robotike. Prema [2] u Americi 1958. godine, a kasnije i u Sovjetskom Savezu su sastavljeni prvi roboti koji su se zvali Scart i Maša. Prema [2] prvog hrvatskog robota sastavio je inženjer Branimir Makanec zajedno s grupom mladih stručnjaka 1966. godine. Robotika se danas najviše primjenjuje u automobilskoj industriji, a njena najveća središta su u Japanu, Kini, SAD - u i u Europi. Roboti su danas ponajviše zamijenili ljude u izvođenju za po život opasnih zadataka za ljude, zatim za zadatke koje ljudi svojim fizičkim sposobnostima ne mogu izvršiti te konačno za po ljudski život opasne okoliše kao što su npr. svemir ili dno oceana. Prema [1], danas postoje brige oko rastuće upotrebe robota u društvu. Robote se krive za porast nezaposlenosti u svijetu budući da zamjenjuju radnike zbog veće brzine izvođenja određenih zadataka i efikasnosti u bilo kojem smislu. Korištenje robota u vojsci kao borbene entitete potiče u mnogim slojevima društva moralna i etička pitanja budući da stvaraju određenu nadmoćnost nad protivnikom koji ne koristi robote kao borbene entitete. Mogućnosti autonomije robota u teoriji su još uvijek neotkrivene i širina njihove primjene raste kako vrijeme teče budući da sve više i više zamjenjuju ljudske resurse kako tehnologija sve više i više napreduje. Smatra se da će u budućnosti postati toliko slični ljudima da se gotovo i neće primijetiti razlika između stvarne osobe i robota. Ta budućnost

će se ostvariti onog dana kada prvi takav robot prođe tzv. Turingov test. Turingov test prema [7] predstavlja test u kojem se provjerava inteligencija računala, odnosno stroja. Test se vrši tako što čovjek u ulozi suca prirodnim govornim jezikom komunicira s drugim čovjekom i strojem, tako da se sucu obojica prikazuju ljudima. Sudac mora otkriti tko je čovjek, a tko je stroj. Ako sudac ne može sa sigurnošću utvrditi koji je sugovornik stroj, a koji je čovjek, smatra se da je stroj prošao Turingov test. Kako bi test bio što jednostavniji, odvija se putem tipkovnice i ekrana. Što se tiče klasifikacije robota, danas nema konsenzusa oko toga koji se strojevi mogu nazivati robotom. Međutim, postoji uvriježeno mišljenje oko stručnjaka i javnosti. To je da su roboti oni strojevi koji posjeduju neke ili sve od sljedećih svojstava i funkcija; moguće ih je programirati, procesiraju podatke električki (digitalno), posjeduju određeni stupanj slobode (u mehanici taj pojam označava slobodu kretanja čvrstog tijela u trodimenzionalnom prostoru, a u robotici označava broj neovisnih pokreta robota u 3D prostoru, npr. kretanje mobilnog robota unaprijed i unazad označavaju dva stupnja slobode), mobilni su, sposobni su percipirati svijet oko sebe pomoću senzora te manipulirati okolišem u kojem se nalaze i konačno, posjeduju inteligentno „ponašanje“ – misli se na ponašanje koje podrazumijeva oponašanje ljudi ili neke od životinja. Neke od najčešćih klasifikacija robota koje su korištene su sljedeće: mobilni robot, industrijski robot, edukacijski robot, vojni robot itd. U ovom diplomskom radu obrađivat će se mobilni robot. Sljedeće slike prikazuju robote ovisno o namjeni i uvjetima za koje su napravljeni.



**Slika 2.2:** Primjer mobilnog robota. Interplanetarni američki mobilni robot Curiosity, koji se trenutno nalazi na Marsu i obavlja istraživačke zadatke (izvor: [10])





**Slika 2.3:** Primjer industrijskog robota koji se koristi u automobilskoj tvornici. Robot postavlja vjetrobransko staklo na automobilu (izvor: [21])



**Slika 2.4:** Primjer edukacijskog robota za djecu (izvor: [19]).



**Slika 2.5:** Vojni mobilni robot koji leti zrakom te obavlja razne vojne zadatke, od snimanja neprijateljskog teritorija do bombardiranja (izvor: [20]).

## 2.1 MOBILNI ROBOTI

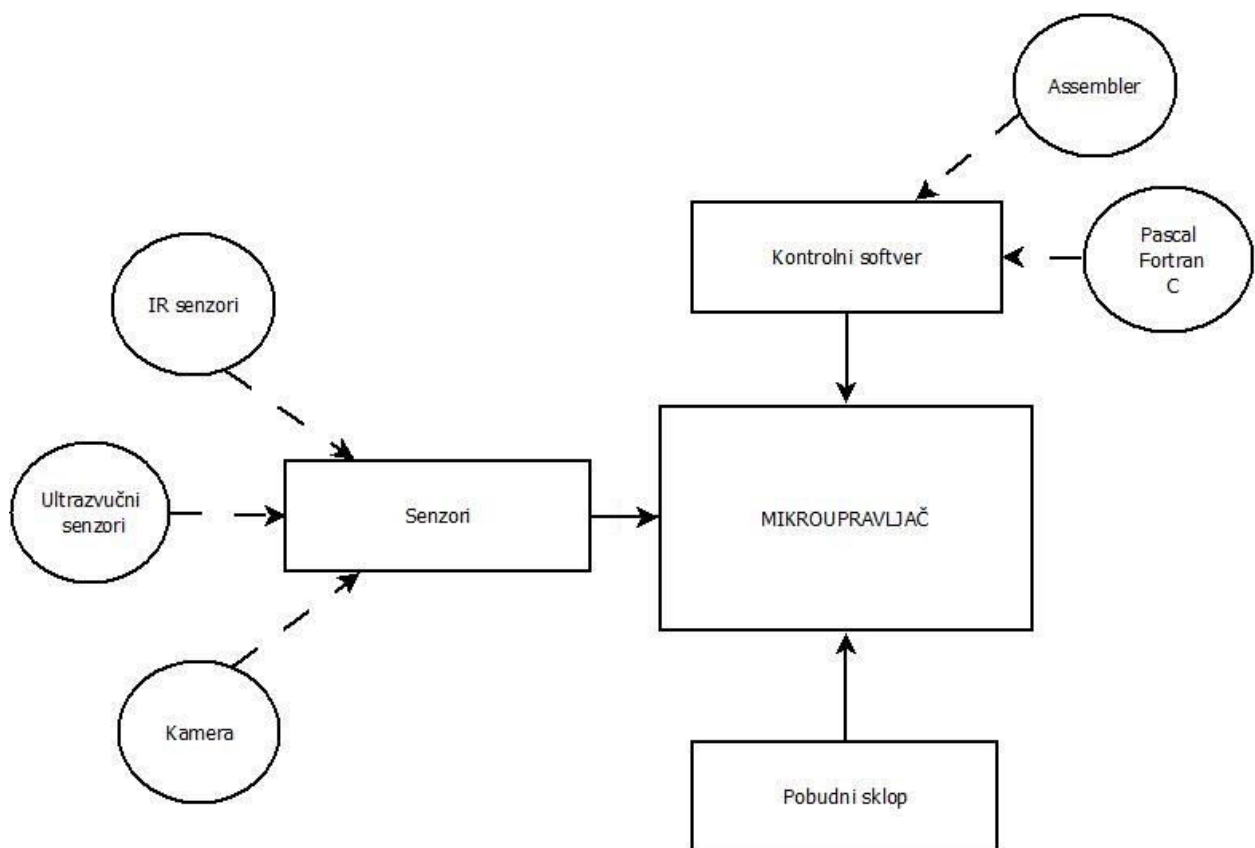
Mobilni robot, kao jedna od klasifikacija robota, prema [2] predstavlja robota koji se može kretati u prostoru i nije vezan za isključivo jednu prostornu lokaciju. Jedan od primjera mobilnih robota je prema [1] automatski vođeni robot (eng. *Automatic Guided Vehicle*). Taj robot prati trag koji može biti linijski (npr. ucrтана linija na podu) ili vizualni (koristi kameru ili lasere) kako bi se snalazio u prostoru i obavljao određene zadatke. Često se sreću u industriji kako prate liniju ucrтанu u podu tvornice da bi dostavili materijal iz točke A u točku B. Mobilne robote se također može sresti i u bolnicama, skladištima, u vojsci i u domaćinstvima. Mobilni roboti se najviše razvijaju u većim sveučilištima i zanimljivo je napomenuti da većina tehničkih fakulteta koja se bavi i robotikom kao znanosti, posjeduje jedan ili više laboratorija koji se bave istraživanjem i unapređivanjem mobilnih robota. Što se tiče mobilnih robota u domaćinstvima, jedan od primjera je robot usisavač koji posjeduje sposobnost orijentacije u prostoru i prepoznavanje neočišćenih površina.

### **2.1.1 Klasifikacija mobilnih robota**

Prema [2], mobilni roboti se klasificiraju prema okolišu za koji su namijenjeni i po uređaju koji koriste za kretanje. Što se tiče okoliša, postoje roboti koji se kreću po zemljanoj površini i površini kakve se pronalaze u domaćinstvima. Ti roboti najčešće koriste kotače ili gusjenice za kretanje po tim površinama, a nazivaju se prema engleskom nazivu UGV (eng. *Unmanned Ground Vehicle*). Zatim postoje transportni roboti ili roboti dostavljači koji prenose materijale u nekakvom radnom ili industrijskom okolišu. Zračni roboti (eng. *Unmanned Aerial Vehicle*) se ne kreću po tlu, već u zraku. Podvodni roboti (eng. *Autonomous Underwater Vehicle*) se kreću u vodi, a postoje i polarni roboti koji su dizajnirani za kretanje u okolišu gdje su prisutne niske temperature, dakle kreću se uglavnom po ledu. Što se tiče uređaja koji koriste za kretanje, mobilni roboti mogu koristiti kotače (eng. *Wheeled Robots*), zatim mogu imati module koji oponašaju ljudske ili životinjske noge (eng. *Legged Robots*) i konačno mogu koristiti gusjenice umjesto kotača.

### **2.1.2 Komponente mobilnog robota**

Komponente mobilnog robota, odnosno sastavni dijelovi mobilnog robota su mikroupravljač, kontrolni softver, senzori te pobudni sklop koji pokreće robota. Mikroupravljač može biti: ulazno/izlazni čip koji sadrži u sebi mikroprocesor, radnu memoriju itd., mikroprocesor ili čak i osobno računalo (PC). Upravljački softver predstavlja program koji upravlja ponašanjem robota. Program može biti pisan u jezicima niske i visoke razine. Za jezik niske razine najčešće se koristi Assembler. Što se tiče programskih jezika visoke razine, najčešće se koriste programski jezici kao što su C, C++, Pascal, Fortran itd. Senzori koji se koriste u robotu ovise o potrebama zadatka kojeg će robot izvršavati. Najčešće se koriste ultrazvučni senzori za određivanje udaljenosti prepreke od robota, zatim infracrveni senzori koji se također koriste za određivanje udaljenosti prepreke, kamera za vizualno prepoznavanje oblika itd. Na sljedećoj slici se nalazi jednostavna opća blok shema mobilnog robota koja sadrži sve prethodno opisane komponente.

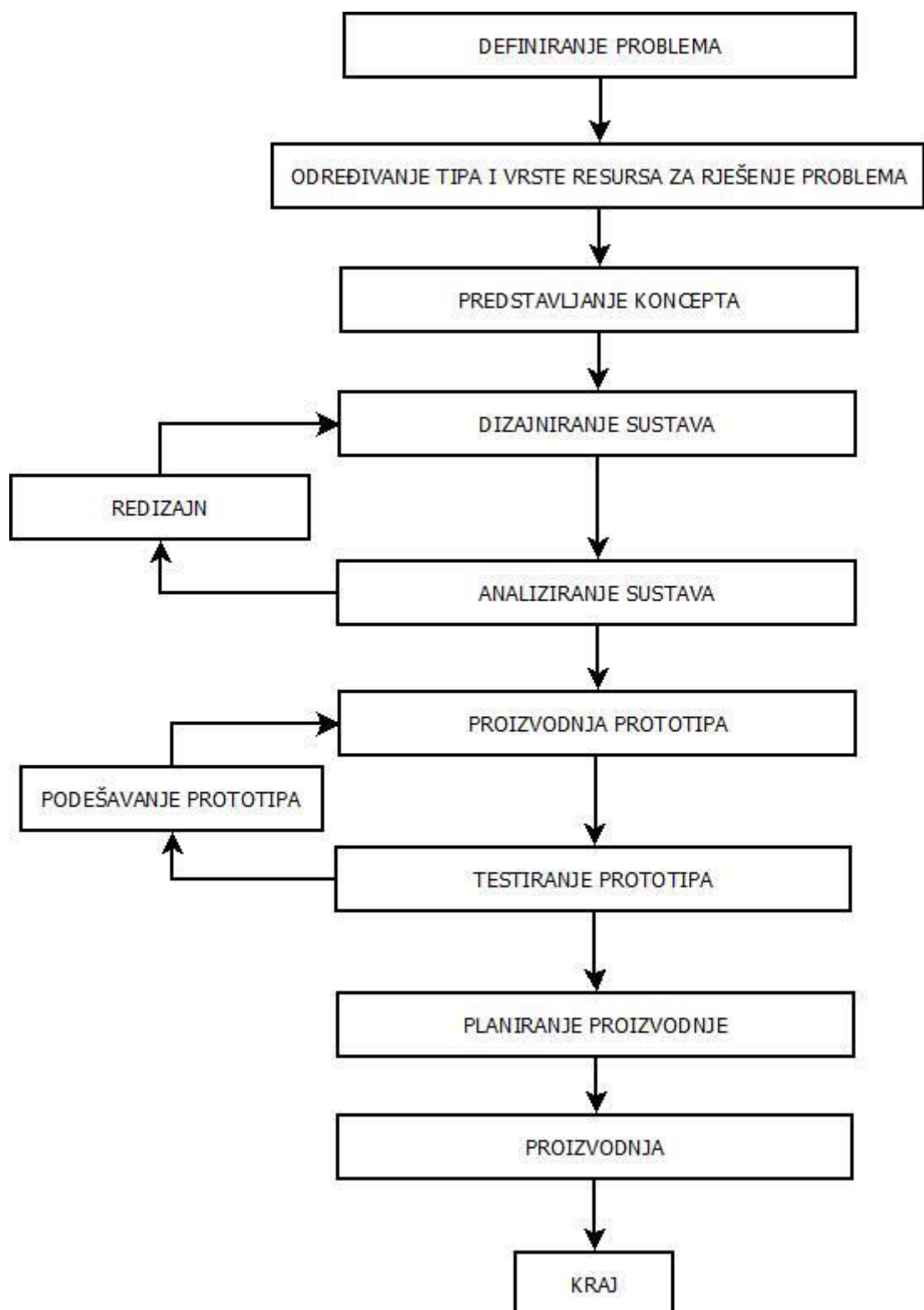


**Slika 2.6:** Jednostavna opća blok shema mobilnog robota

Prema slici 2.6 se vide strelice koje označavaju tip povezanosti komponenata mobilnog robota. Isprekidane strelice predstavljaju vrste pojedinih komponenata, čiji su nazivi navedeni u kružićima. Strelice iscrtane punom linijom predstavljaju vezu između blokova. Veza između senzorskog bloka i bloka mikroupravljača je jednosmjerna veza koja pokazuje da senzori pretvaraju signal bilo koje prirode (npr. mehaničke) u električni signal razumljiv mikroupravljaču. Taj signal mikroupravljač tretira kao podatak, a način tretiranja je opisan blokom kontrolnog softvera. Kontrolni softver predstavlja programski kod pisan u nekom od programskih jezika, zamišljen prema algoritmu rada i predstavlja način na koji mobilni robot „razmišlja“, odnosno na koji način analizira podatke dobivene senzorom te kakvu odluku donosi u skladu s tim podacima. Pobudni sklop je sklop koji napaja mobilnog robota. Za mobilne robote se koriste baterijski sustavi, odnosno baterije. Mikroupravljač je glavni i najvažniji sklop svakog mobilnog robota, to je uređaj koji interpretira podatke na način kako je definirano u algoritmu rada, a napisano u nekom od programskih jezika. U sljedećem poglavlju i potpoglavljima je dan detaljniji opis funkcionalnosti svakih od tih komponenti koje se nalaze na slici 2.6.

### 3. DIZAJN MOBILNOG ROBOTA

Dizajn u robotici podrazumijeva konstruiranje i programiranje robota tako da odgovara postavljenim zahtjevima nekog zadatka. Konstrukcija se vrši s obzirom na vrstu okoliša u kojem će se robot kretati, a programiranje robota se vrši tako da mu se odredi na koji način će interpretirati podatke koje će dobivati od senzora i time učiniti određene radnje. S obzirom na zadatak u ovom diplomskom radu, potrebno je osmisliti upravljački sustav mobilnog robota temeljen na mikroupravljaču te odabrati senzore na temelju postavljenih zahtjeva. Sensori su entiteti koji pretvaraju nekakve informacije iz vanjskog svijeta u električni signal razumljiv robotu. Kod dizajna, prvo je potrebno odrediti namjenu mobilnog robota. Nakon toga se radi algoritam koji će opisivati logiku odlučivanja robota na temelju dobivenih informacija iz senzora. Tokom razrade samog algoritma paralelno se za postavljeni problem odabiru senzori koji će služiti robotu za orijentaciju u prostoru, tip mikroupravljača koji će moći ispuniti zahtjeve zadatka u smislu da ima dovoljno računalnu moć da u zadanim vremenskim okvirima i ostalim uvjetima izvodi program u realnom vremenu bez zastoja. Nakon toga se odabire pobudni sklop (najčešće su to baterijski moduli koji ispunjavaju uvjete napajanja senzora, mikroupravljača i istosmjernih ili koračnih motora), a konačno se odabiru i istosmjerni ili koračni motori prema zadatku koji je postavljen. Kod motora je pored napajanja, nužan i parametar snage zbog postizanja određene brzine, a isto tako i mogućnosti prelaženja preko nepravilnog terena. Kada se taj dio zadatka riješi, stupa se u realizaciju mobilnog robota koji će odrađivati taj zadatak. Prvi korak realizacije dizajna je nabavka ili izrada potrebne opreme za robota. Zatim se po električnoj shemi spajaju određeni dijelovi mobilnog robota, a shema se izrađuje prema zadatku kojeg će mobilni robot odrađivati kao i prema potrebnim uvjetima pojedinog električnog sklopa koji se nalazi u robotu. Nadalje, prema algoritmu se piše programski kod (eng. *firmware*) za mikroupravljač u nekom od programskih jezika koji se prepravlja prema rezultatima testiranja mobilnog robota. Kada rezultati testiranja postanu zadovoljavajući, smatra se da mobilni robot ispunjava zadani zadatak i proces dizajna i realizacije mobilnog robota se smatra završenim. Prema [10], dizajn predstavlja kreativnu aktivnost koja se može predočiti procesom razumnog donošenja odluka. Općenito govoreći, usmjerena je prema zadovoljavanju potrebe zadatka. Nema jedinstvenog pristupa za dizajn bilo kojeg sustava, kao što nema ni jedinstvenog pristupa za postizanje kreativnosti. Svaki dizajner bilo koji zadatak dizajnira na svoj vlastiti jedinstveni način. Međutim, postoji obrazac koji je primjenjiv i koristan za razumijevanje procesa dizajna. Sljedeća slika predstavlja taj obrazac.



Slika 3.1: Prikaz procesa dizajniranja nekog sustava

Gornja slika predstavlja općeniti opis procesa dizajniranja nekog sustava, od početka do kraja. Prilikom dizajniranja bilo kojeg sustava, uvijek se polazi od definiranja problema, a to je prvi i osnovni korak da se uspješno dizajnira neki sustav i eventualno plasira na tržište ukoliko je namijenjen za javnu upotrebu. Korak „analiziranje sustava“ je korak u kojem se detaljno ispituje model nekog sustava te zatim rezultate testiranja se podvrgava analizi i ukoliko se pronade neka nepravilnost ili anomalija, pristupa se redizajnu sustava da bi se ispravila ta anomalija. Korak „testiranje prototipa“ je korak koji je sličan potonjem, s iznimkom da se ne analizira model, bilo teoretski, bilo model izrađen u nekom od programskih paketa, nego se analizira konačan proizvod. Rezultati testiranja otkrivaju pogreške koje se ne mogu obuhvatiti teoretskim modelom, budući da se testira funkcionalnost uređaja u realnom svijetu. Te se pogreške nakon toga ispravljaju, a to je predstavljeno korakom „podešavanje prototipa“.

### **3.1 ZADATAK MOBILNOG ROBOTA**

U ovom diplomskom radu, zadatak mobilnog robota je kretanje po tlu (pod u domaćinstvu) u neodređeno velikom prostoru s preprekama. Robot će koristiti senzore za orijentaciju po prostoru i moći će prepoznavati prepreke i posjedovati funkcionalnost da ih izbjegava. S obzirom na taj zadatak, odabrano je da će mobilni robot koristiti kotače, a za senzore će koristiti digitalne infracrvene senzore s dva stanja. Jedno stanje uključuje odsutnost prepreke, a drugo stanje uključuje prisutnost prepreke. Detaljnije o sensorima je obrađeno u potpoglavlju 3.4. Algoritam po kojem robot funkcionira je opisan u potpoglavlju 3.2. Mikroupravljač kojeg koristi mobilni robot je mikroupravljač tvrtke ATMEL pod nazivom ATMEGA16A. Programski jezik koji će se koristiti za taj mikrontroler je programski jezik C. Detaljnije o odabranom mikroupravljaču se nalazi u potpoglavlju 3.3. Odabir pobudnog sklopa je riješen određivanjem potrebnih napona i struja za mikroupravljač, istosmjerne motore, sklop pokretača motora i senzore. Više detalja o pobudnom sklopu se nalazi u potpoglavlju 3.5. Blok shema mobilnog robota predstavlja grafički prikaz u kojem se nalaze blokovi opisanih entiteta koje se nalaze u robotu, a na tom grafičkom prikazu se nalaze informacije o tome kako su ti blokovi unutar robota povezani. Detaljnije se može vidjeti u potpoglavlju 3.6. Konačno, odabir konstrukcije je odabran rješenjem svih prethodnih problema koji su opisani u odgovarajućim potpoglavljima. Konstrukcija podrazumijeva fizikalno rješenje robota u kojem se rješavaju problemi postavljanja senzora na robotu, određivanje položaja ostalih električnih sklopova kao što su mikroupravljač, pokretač istosmjernih motora i pobudni sklop. Detaljnije o tome se može vidjeti u potpoglavlju 3.7. U

potpoglavlju 3.8. se može vidjeti konačno rješenje mobilnog robota koji ispunjava zahtjeve zadatka. U poglavlju 4. se nalaze informacije o rezultatima testiranja mobilnog robota.

## 3.2 ALGORITAM ZADATKA

Algoritam predstavlja skup koraka koje treba odraditi da bi se izvršio određeni zadatak. Pokazuje redoslijed izvođenja koraka koje treba izvesti da se dođe do konačnog rješenja. Algoritam kao pojam se koristi u raznim granama znanosti, a najčešće u matematici i u računarstvu. U matematici, algoritam predstavlja korake koje treba izvesti da bi se riješio postavljeni matematički zadatak. U računarstvu, algoritam određuje procese unutar nekog entiteta, objekta ili tijela koji se izvode da bi se taj entitet, objekt ili tijelo ponašali onako kako se traži od njih. O tome kako će algoritam izgledati, ovisi o postavljenom zadatku. U ovom diplomskom radu mobilni robot putuje po površini kakva se najčešće pronalazi u domaćinstvima (parket, laminat) i mora obilaziti prepreke kakve se pronalaze u domaćinstvu. Da bi se kreirao algoritam rada robota, prvo se moraju postaviti pitanja i pronaći odgovore na njih. U ovom konkretnom slučaju, na temelju donešenih informacija s obzirom na robotovo okruženje, postavljaju se dva pitanja. Prvo je pitanje, s obzirom na to da je riječ o površinama u domaćinstvu, kakve uređaje treba koristiti da bi se omogućila robotova mobilnost? Drugo je pitanje, kako riješiti problem detektiranja prepreka? Odgovor na prvo pitanje je donešeno na način da se u obzir uzela cijena tog uređaja i vrsta površine po kojoj će se robot kretati. Budući da je riječ o uglavnom ravnoj površini s malom vjerojatnošću pojavljivanja nepravilnog terena, za uređaj su izabrani okrugli kotači umjesto kotača s gusjenicom. Kotači s gusjenicom su pogodni za nepravilne terene i nešto su skuplji od običnih okruglih kotača s gumom te stoga nisu pogodan izbor za ovaj zadatak. Odgovor na drugo pitanje prvo poteže novo pitanje, a to pitanje glasi; s kakvim preprekama robot može imati posla? U domaćinstvima su prepreke česte pojave, a mogu biti dinamičke i statičke. Dinamičke prepreke su pomične prepreke, to npr. može biti i čovjek ili domaća životinja, dok statičke prepreke predstavljaju objekti koji su nepomični i ne mijenjaju položaj u prostoru (npr. regal, stol, stolice itd.). Nakon što je otkrivena priroda prepreka, može se prijeći na odgovaranje drugog pitanja. Odabir vrste senzora koja će služiti za detekciju prepreka također poteže sa sobom nekoliko zahtjeva i postavljaju se nova pitanja. S obzirom na to da je u vidu što jednostavnije riješiti zadatak, prvi zahtjev će biti jednostavnost senzora, a drugi zahtjev će biti cijena. Općenito, senzori mogu biti analogne prirode ili digitalne prirode. Senzori analogne prirode za informacijski signal šalju analogni signal (kontinuirani signal), dok digitalni senzori



šalju digitalni signal (diskretni signal). Za ovaj zadatak se odabiralo između te dvije vrste senzora. Digitalni senzori su jednostavniji za dekodiranje budući da imaju samo dva stanja na izlazu (prepreka je otkrivena i prepreka nije otkrivena, jedno stanje predstavlja logičku jedinicu, a drugo stanje predstavlja logičku nulu). S digitalnim sensorima je lakše raditi jer njima nije potrebna kalibracija i mjerenje udaljenosti da bi se odredila udaljenost na kojemu će senzor na izlazu davati signal koji će označavati stanje da je prepreka pronađena. Digitalni senzori su također podesivi i u primjeru senzora koji je korišten u ovom diplomskom radu može se vidjeti u potpoglavlju 3.4 na koji način se povećava domet senzora. Analogni senzori zahtijevaju dodatna podešavanja i mjerenja izlaznog signala da bi se odredila udaljenost između robota i senzora koja se uzima kao najmanja moguća udaljenost između robota i prepreke. Cjenovno ni jedna ni druga vrsta senzora nemaju značajne razlike. Senzori između kojih se vršio odabir su bili sljedeći; digitalni infracrveni senzor i analogni ultrazvučni senzor. Odabran je digitalni infracrveni senzor zbog jednostavnosti oblika izlaznog signala koji ima samo dva stanja. U poglavlju 4. će se pokazati performanse tog senzora, njegove prednosti i njegovi nedostaci. Nadalje, postavlja se pitanje koliko senzora postaviti na mobilnog robota da bi se optimalno izbjegavale prepreke? Budući da se u ovom diplomskom radu obrađuje dizajn mobilnog robota, prvo se počelo s ugradnjom tri senzora te vrste. U poglavlju 4. će se prikazati posljedice tog izbora. Konačno se postavlja i odabir broja kotača. U ovom konkretnom slučaju, odabrano je da će mobilni robot imati dva pogonska kotača, koje će pogoniti istosmjerni motori pomoću sklopa pokretača motora (da bi se izbjegla povratna veza između mikroupravljača i istosmjernih motora, jer budući da su istosmjerni motori induktivne prirode, može doći do iznenadnog skoka struje uslijed magnetizma unutar motora koja posljedično može uništiti mikroupravljač) i jedan kotač koji će služiti za ravnotežu konstrukcije mobilnog robota. Odabir mikroupravljača trenutno ne igra veliku važnost za izradu algoritma budući da algoritam predstavlja zapravo programski kod, a mikroupravljač je uređaj koji izvršava taj programski kod. Algoritam i logika odlučivanja su prikazani grafički i tekstualno u sljedećem potpoglavlju.

### **3.2.1 Grafički prikaz algoritma**

Nakon što je određen broj senzora i broj kotača, može se pristupiti izradi algoritma. Algoritam se prvo izrađivao tablično gdje su se definirala sva moguća stanja senzora i za svako stanje senzora je opisano što se treba događati s pogonskim kotačima. Taj algoritam s opisom stanja i izvršnih procedura se nalazi u tablici 3.1. Detaljan opis tablice se nalazi ispod tablice.

**Tablica 3.1: Tablični prikaz algoritma**

Redni broj stanja	Lijevi senzor	Prednji senzor	Desni senzor	Odluka
1.	0	0	0	Kreni unazad i okreći se ulijevo dok sva tri senzora ne budu u logičkoj jedinici
2.	0	0	1	Okreći se udesno dok sva tri senzora ne budu u logičkoj jedinici
3.	0	1	0	Kreni unazad i okreći se ulijevo dok sva tri senzora ne budu u logičkoj jedinici
4.	0	1	1	Okreći se udesno dok sva tri senzora ne budu u logičkoj jedinici
5.	1	0	0	Okreći se ulijevo dok sva tri senzora ne budu u logičkoj jedinici
6.	1	0	1	Okreći se ulijevo dok sva tri senzora ne budu u logičkoj jedinici
7.	1	1	0	Okreći se ulijevo dok sva tri senzora ne budu u logičkoj jedinici
8.	1	1	1	Idi ravno dok se ne promijeni stanje senzora

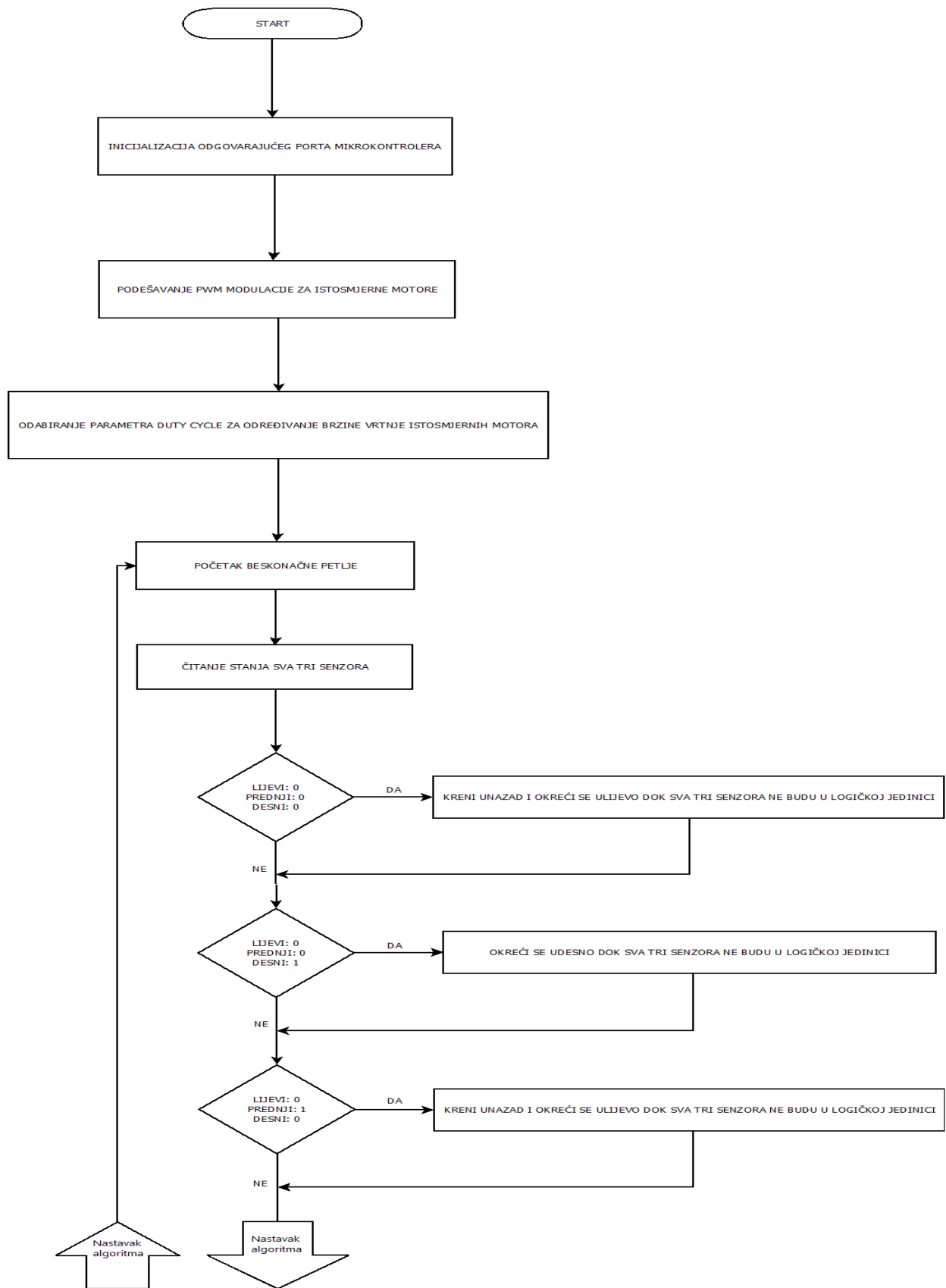
Tablica 3.1 opisuje broj svih mogućih stanja senzora i za svako stanje sva tri senzora definirana je odluka koju robot treba donijeti kada se suoči s takvom situacijom. Broj stanja senzora iznosi osam, a dobiven je na sljedeći način:

$$N = 2^n \quad (3 - 1)$$

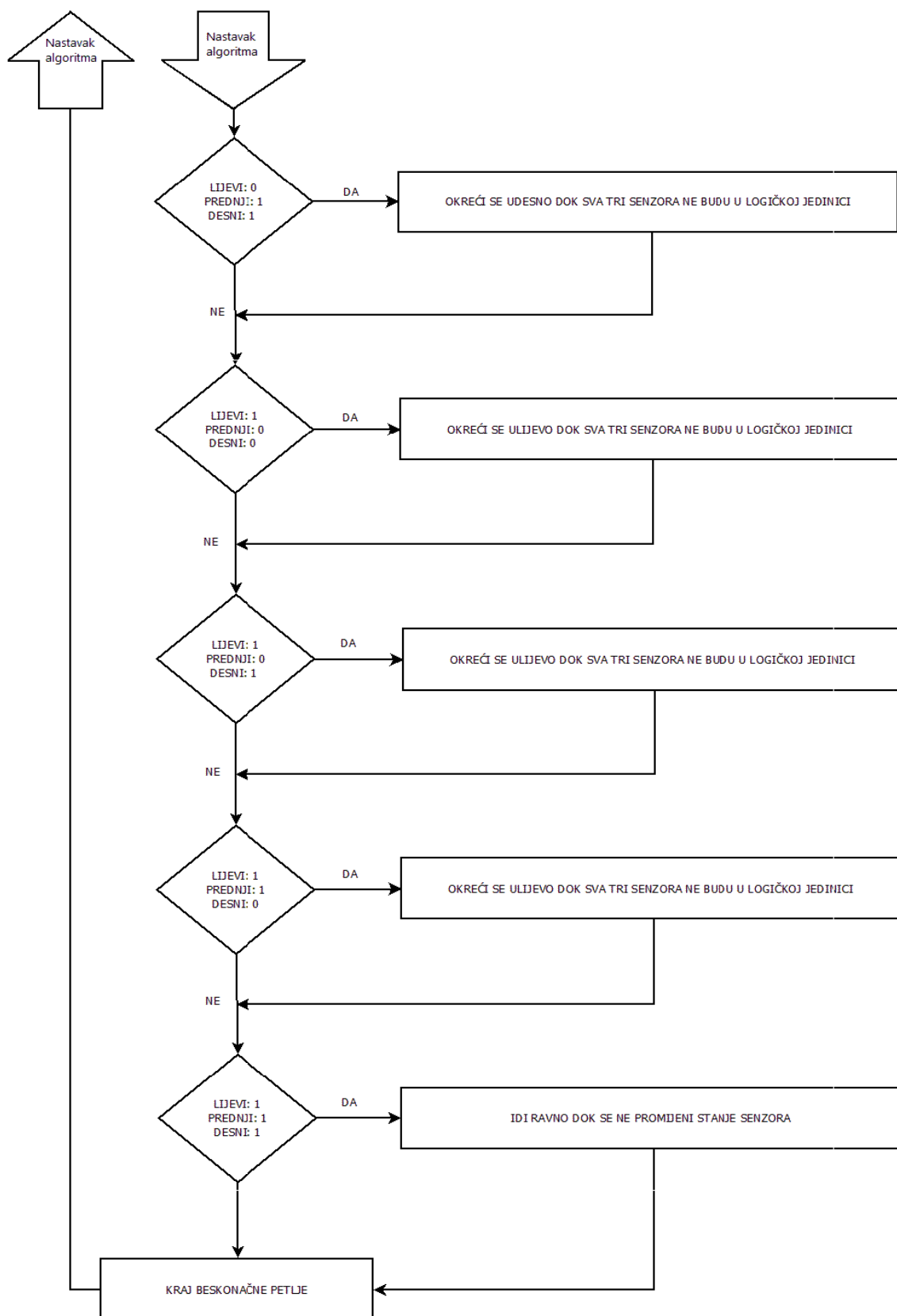
gdje  $N$  predstavlja ukupan broj mogućih stanja sva tri senzora, broj 2 predstavlja broj stanja pojedinog senzora (budući da su senzori digitalni, imaju samo dva stanja), a broj  $n$  predstavlja broj senzora. Uvrštavanjem tih parametara u relaciju (3 - 1) dobije se broj mogućih stanja:

$$N = 2^n = 2^3 = 8 \quad (3 - 2)$$

To je razlog zašto u tablici postoje 8 mogućih stanja. Logičke nule i logičke jedinice predstavljaju ta dva moguća stanja infracrvenih senzora. Logička nula označava stanje u kojem je prepreka detektirana, a logička jedinica označava stanje u kojemu prepreke nema. Razlog tome leži u konstrukciji senzora. Senzor koji je korišten u diplomskom radu ima tako postavljen izlazni digitalni signal. Detaljnije o tome u poglavlju 3.5. Stupci koji označavaju lijevi, desni i prednji senzor na mobilnom robotu, u svakom retku imaju određenu kombinaciju logičkih nula i jedinica. Budući da je relacijom (3 - 2) pokazano da postoji osam mogućih stanja za tri senzora, pokrivene su sve moguće kombinacije logičkih nula i jedinica. Odluke za svako stanje su donešene prema pravilu lijeve ruke. To znači da će robot, ukoliko naiđe na prepreku, izvršiti uvijek skretanje ulijevo. Osim u jednom slučaju kada je desni senzor neaktivan, a tada u tom slučaju robot izvršava skretanje udesno. Razlog tome je efikasnost skretanja, budući da je „neprirodno“ skretati ulijevo ako ima prostora s desne strane. Robot zbog toga radi „prirodnije“ i logično izvodi procedure obilaženja prepreka iako je prednost dana pravilu lijeve ruke. Isto tako se sve može izvesti i obratno, a to je da se uvede pravilo desne ruke. Grafički prikaz algoritma mobilnog robota je prikazan na sljedećim stranicama na slikama 3.1 i 3.2, gdje druga slika prikazuje nastavak algoritma na dio algoritma na prvoj slici. Algoritam je podijeljen na dvije slike zbog preglednosti. Prikazuje način rada mobilnog robota i na vizualni način prikazuje na koji će način mobilni robot detektirati prepreku i koju će odluku donijeti s obzirom na signale koji stižu iz senzora u mikroupravljač.



**Slika 3.1:** Algoritam rada mobilnog robota (prvi dio)

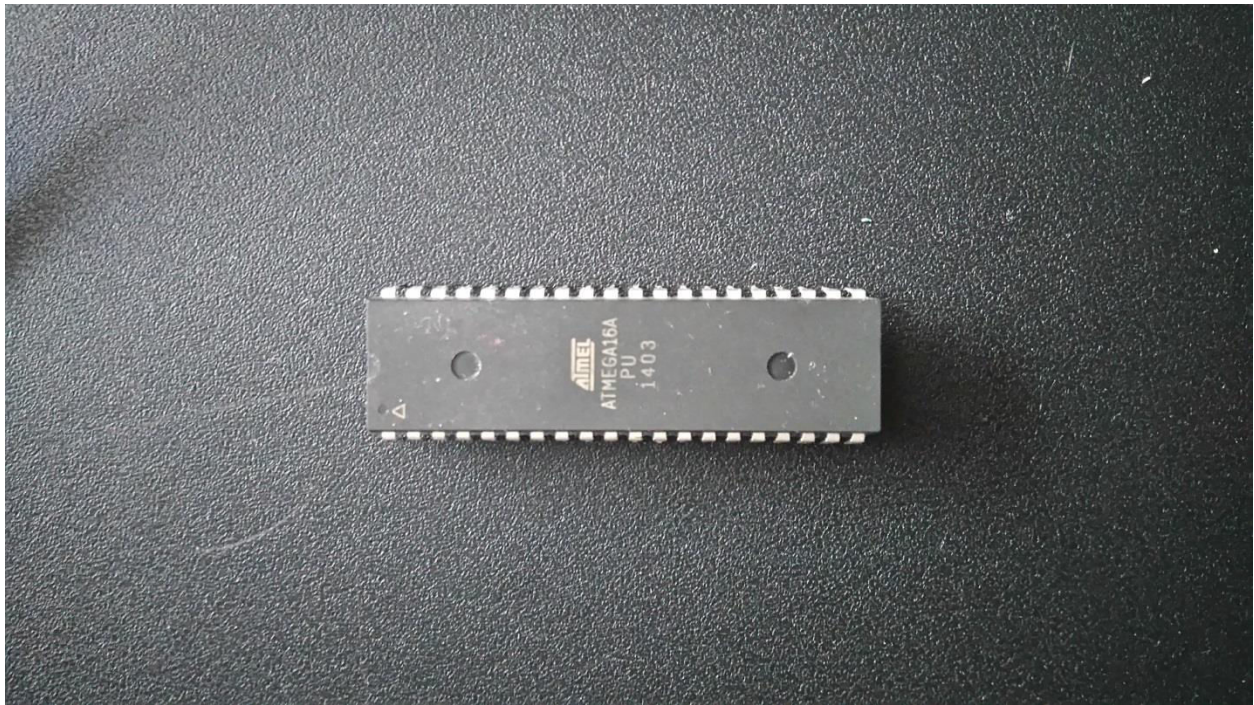


**Slika 3.2:** Algoritam rada mobilnog robota (drugi dio)

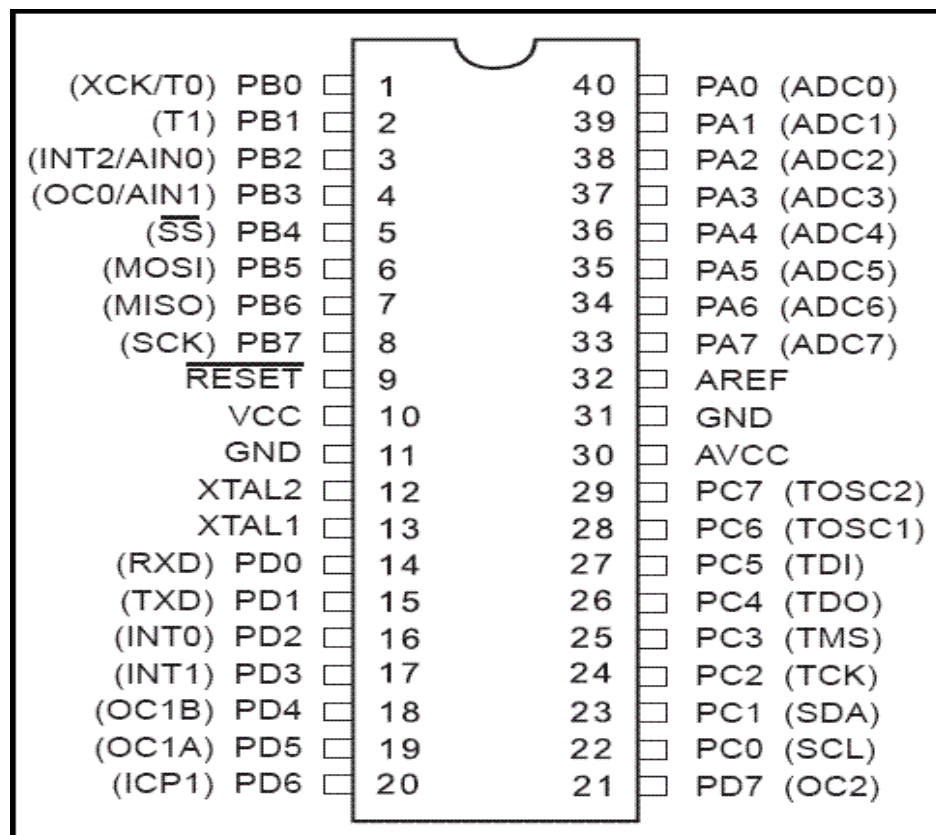
Na slikama 3.1 i 3.2 se mogu vidjeti razni pravokutni i rombovski oblici kao i linije. Tzv. rombovski oblici označavaju uvjet, dok pravokutnici u sebi sadržavaju radnju koju mobilni robot treba izvesti za taj određeni uvjet unutar beskonačne petlje. Na početku algoritma na prvoj slici se nalazi eliptični geometrijski oblik s natpisom *START*, a označava početak izvođenja algoritma. U realnom svijetu se to može promatrati kao kada pobudni sklop mobilnog robota aktivira robota, poput prekidača za svjetlo kada se postavi u aktivni položaj. Algoritam opisuje slijed koraka koje treba izvesti da bi se izveo određeni zadatak. U ovom slučaju ti koraci su pravokutnici i rombovi. *START* označava prvi korak, aktivaciju robota pomoću pobudnog sklopa. Zatim slijedi inicijalizacija odgovarajućeg porta, odnosno portova. To je zbog toga što svaki port sadrži osim općeg ulazno/izlaznog načina rada i svoje svojstvene funkcije koje neki drugi port ne podržava (detaljnije o tome se nalazi u potpoglavlju 3.4). Zbog toga što portovi predstavljaju međusklop između senzora i mikroupravljača, portovi odrađuju dvosmjerne operacije. Jedna od njih je očitavanje stanja senzora, a druga je upravljanje istosmjernim motorima koji su putem osovine spojeni na kotače koji se neovisno okreću. Očitavanje senzora je ulazna operacija, dok je izlazna pokretanje motora. Nakon toga slijedi podešavanje pulsno – širinske modulacije za istosmjerne motore. Pulsno – širinska modulacija je vrsta modulacije električnog signala u kojem se pomoću tog signala upravlja brzinom kretanja istosmjernih motora. Taj signal se šalje na sklop pokretača motora, koji odvaja mikroupravljač i istosmjerne motore (detaljnije o sklopu u potpoglavlju 3.6) zbog toga što se nikada istosmjerni motori ne spajaju izravno na mikroupravljač. Razlog tome je da se spriječi povratna veza od motora prema mikroupravljaču koja je takva da prijeti uništenju mikroupravljača zbog induktivnih svojstava motora. Pohranjivanje magnetske energije unutar bakrenih zavojnica u motoru uzrokuje nagle skokove struje koje mogu biti takve da mogu uništiti mikroupravljač. Nakon toga slijedi odabiranje parametra radnog ciklusa (eng. *duty cycle*) za podešavanje brzine. Detaljnije o tom parametru se nalazi u potpoglavlju 3.6.1. Zatim slijedi ulazak u beskonačnu petlju. Beskonačna petlja predstavlja neprekidno izvođenje određenih radnji na sljedeći način; ulaskom u tu petlju slijedno se izvršavaju operacije i kada se dođe do zadnjeg koraka unutar petlje, sve se počinje ispočetka (unutar beskonačne petlje). Kada se uđe u beskonačnu petlju, dolazi se do prvog uvjeta koji provjerava stanje senzora. Ako stanja sva tri senzora odgovaraju prvom uvjetu unutar petlje, pokreće se rutina na koju upućuje strelica označena s „DA“ unutar algoritma. Nakon što se izvrši ta rutina, kreće se ispočetka i provjerava se bilo koje od mogućih stanja senzora. U bilo kojem slučaju će se uvijek ispuniti jedan uvjet jer ako se pogleda tablica 3.1 vidi se da su sva moguća stanja senzora obuhvaćena.

### 3.4 ODABIR MIKROUPRAVLJAČA

S odabirom mikroupravljača dolazi i odabir programskog jezika u kojemu će se implementirati algoritam. Mikroupravljač koji je odabran je mikroupravljač tvrtke Atmel, naziva ATMEGA16A. Detaljno o mikroupravljaču se nalazi pod [3]. Razlog odabiranja tog mikroupravljača leži u njegovoj jednostavnosti izrade, potrebnim funkcionalnostima koje se nužne za ostvarenje zahtjeva mobilnog robota, njegovoj lakoj dostupnosti i niskoj cijeni. Mikroupravljač raspolaže s četiri ulazno/izlazna porta (port A, port B, port C, port D), od koji svaki od portova ima univerzalnu funkciju i funkciju posebne namjene. Npr. port A raspolaže s analogno - digitalnim pretvaračem, dok su na portu B sadržani vremenski sklopovi, analogni komparator itd. To su funkcije posebne namjene. Svi portovi posjeduju identičnu univerzalnu funkciju, a to je GPIO (eng. *General Purpose Input/Output*) način rada, odnosno opći ulazno - izlazni način rada. To znači da portovi ukoliko nisu inicijalizirani na način da obavljaju neku posebnu funkciju, obavljat će jednostavne ulazno - izlazne radnje kao što su postavljanje izlaza nekog *pina* u logičku jedinicu ili nulu, ili pak učitavanje logičke razine na inicijaliziranom *pinu*. Programski jezik koji će se koristiti s tim mikroupravljačem je programski jezik C. Glavni razlog korištenja tog programskog jezika leži u tome što je to često korišteni programski jezik za mikroupravljač, a sporedni razlog je često korištenje tog programskog jezika na ostalim kolegijima za ostvarivanje drugih projekata na fakultetu u sklopu nekih kolegija. Sljedeća slika prikazuje korišteni mikroupravljač, dok slika ispod te slike shematski prikazuje funkcije svakog *pina* za svaki port u mikroupravljaču.



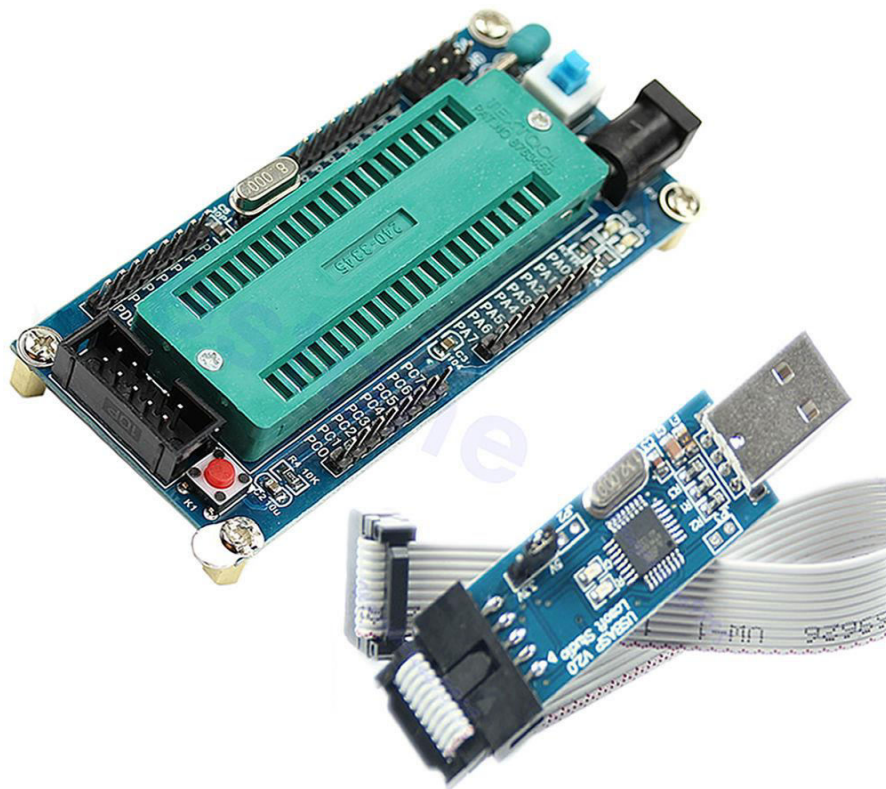
**Slika 3.3:** Mikroupravljač ATMEGA16A koji je korišten u mobilnom robotu



**Slika 3.4:** Prikaz portova A, B, C i D u mikroupravljaču te funkcije svih *pinova* (izvor: [3])



Slika 3.3 prikazuje izvedbu mikroupravljača, dok slika 3.4 prikazuje funkcije *pinova* u mikroupravljaču. Na slici 3.4 se mogu vidjeti sva četiri porta označena na sljedeći način; PAX. X predstavlja broj *pina*, dok oznaka PA predstavlja port A. Pokraj oznake se nalaze opisi, koji opisuju funkciju svakog *pina*. Npr. PA0 s opisom ADC0 predstavlja *pin* čija je funkcija posebne namjene analogno – digitalni pretvarač na *pinu* s oznakom 0. Npr. PB1 predstavlja port B, čiji *pin* pod brojem 1 predstavlja vremenski sklop s oznakom T1. Prema [3] vremenski sklop T1 je 16 – bitni vremenski sklop koji se koristi za mjerenje vremena. Detaljno o načinu rada tog vremenskog sklopa se može vidjeti pod [3]. Na mikroupravljaču se nalaze i *pinovi* koji nisu povezani s portovima. To su *pinovi* bez oznaka portova, npr. *pin* pod rednim brojem 10 s oznakom VCC predstavlja ulaz za napajanje mikroupravljača. Ti *pinovi* moraju nužno biti spojeni s odgovarajućim sklopovljem, inače je mikroupravljač beskoristan. Detaljno o svakom *pinu* i njegovoj posebnoj namjeni se može pronaći pod [3]. Na sljedećoj slici se može vidjeti modul tih sklopovlja koje se lako može nabaviti od proizvođača.



**Slika 3.5:** Prikaz modula koji se koristi za mikroupravljač ATMEGA16A (izvor [22]).

Modul štedi vrijeme pri izradi bilo kojeg sklopa budući da korisnik ne mora spajati oscilator (uređaj koji daje frekvenciju rada mikroupravljaču ako se koristi vanjski oscilator) ili

programatorsko sučelje koje omogućava upis napisanog programa na PC računalu na mikroupravljač, kao i ostale popratne električne komponente po točno određenoj električnoj shemi. Mikroupravljač se stavi u taj modul u zeleno kućište, gdje se pokraj njega nalazi polugica za pričvršćivanje mikroupravljača. S desne strane slike 3.5 se nalazi programatorsko sučelje za upisivanje programa u mikroupravljač s osobnog računala.

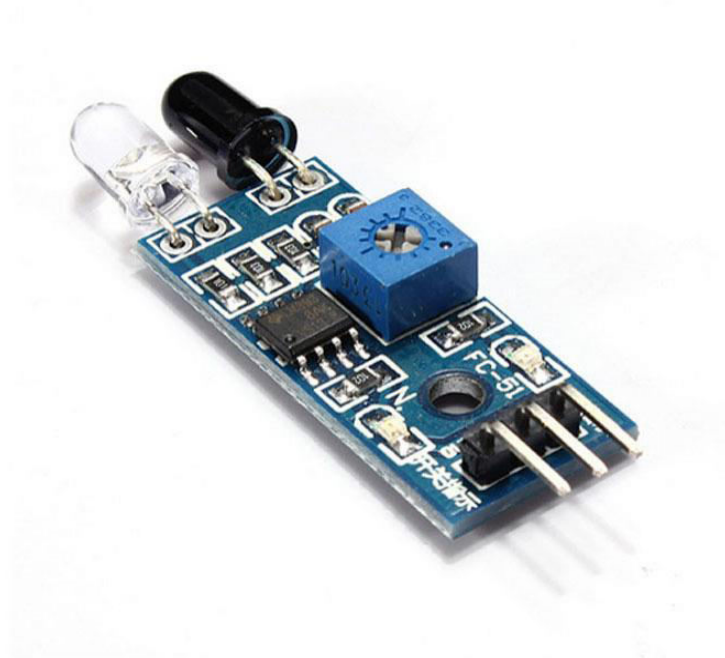
### **3.5 ODABIR SENZORA**

Odabir senzora se vrši prema postavljenim zahtjevima. Budući da mobilni robot treba izbjegavati prepreke kakve se pronalaze u domaćinstvima, najjednostavniji tip senzora koji zadovoljava postavljene zahtjeve je digitalni infracrveni senzor s dva stanja. Jedno stanje uključuje odsutnost prepreke, a drugo stanje uključuje prisutnost prepreke. Jednostavnost je najveća prednost ovoga senzora, budući da nema potrebe za kalibracijom i preciznog određivanja udaljenosti s obzirom na to kada mobilni robot mora započeti proceduru obilaženja prepreke. U poglavlju 4. je pokazano da s tvorničkom kalibracijom mobilni robot izvršava obilaženje prepreka bez problema, jer je prostor između prepreke i senzora dovoljno velik da robot bez udaranja o prepreku izvrši proceduru skretanja. Dakle, iako se taj senzor također može kalibrirati na način da mu se poveća ili smanji udaljenost detekcije prepreke, tvorničke postavke su se pokazale sasvim dovoljne za izvršavanje tih procedura.

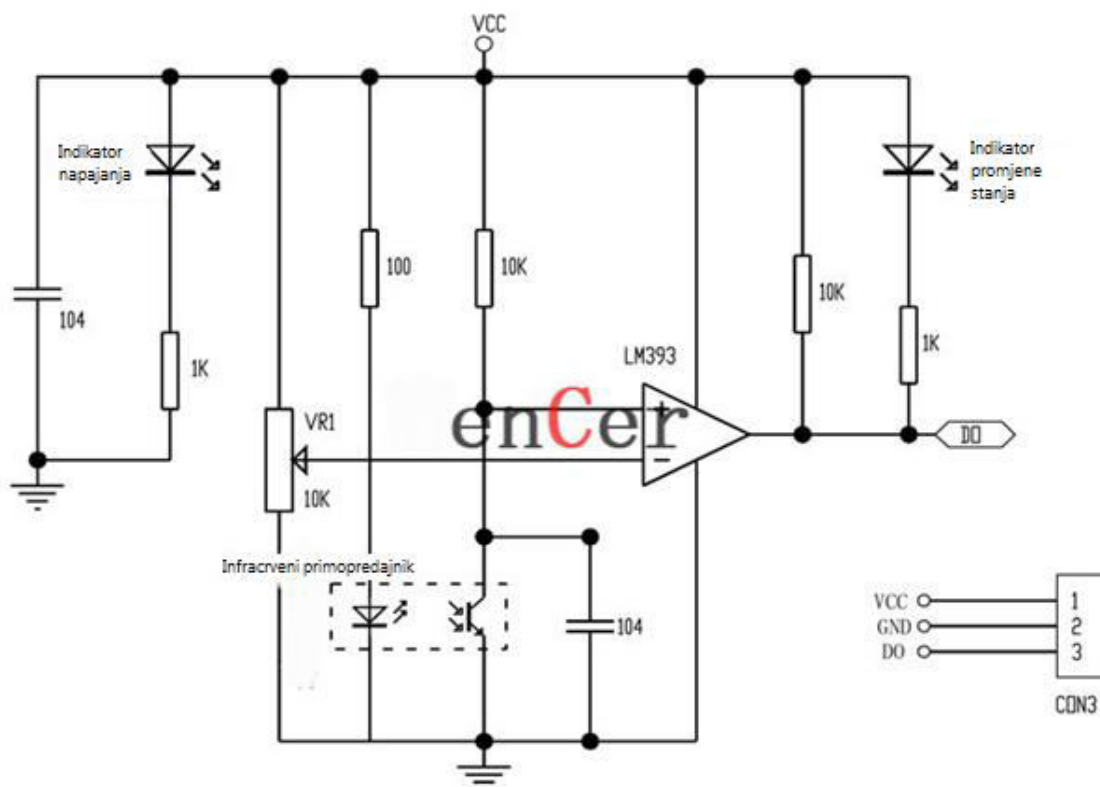
#### **3.5.1 Princip rada digitalnog infracrvenog senzora s dva stanja**

Princip rada senzora je osmišljen na način da se koriste elektromagnetski valovi u infracrvenom spektru. Taj senzor koji je korišten u mobilnom robotu je senzor koji je naručen od proizvođača, nije samostalno dizajniran, zbog jednostavnosti, vremena izrade, a i pouzdanosti rada samog sklopa. Senzor radi na način, da se koristi princip predaja – prijem infracrvenog elektromagnetskog vala (u daljnjem tekstu će se koristiti samo infracrveni val). Infracrveni val se šalje sa predajne komponente senzora, odbija se o prepreku i vrši se detekcija na prijemnoj komponenti senzora. Po prirodi, taj val je usmjeren (uži snop zračenja) budući da je njegova valna duljina veoma mala. Sa smanjenjem valne duljine, odnosno povećanjem frekvencije titranja vala, povećava se usmjerenost tog vala. To je nužan zahtjev za precizno otkrivanje prepreke. Prema [4], senzor otkriva prepreku koja se nalazi ispred putanje tog vala, a kada se prepreka otkrije, uključuje se LE dioda crvene boje, pored jedne koja je već uključena, a prikazuje da je senzor aktivan i da prepreka nije otkrivena. Kada se prepreka otkrije, reflektirani

val se detektira na prijemnoj strani senzora i uzrokuje logičku nulu na njegovom izlazu. To svojstvo se iskorištava na način da mikroupravljač očitava promjenu logičkog stanja i vrši odluku o skretanju. Što se tiče performansi senzora, one su prema [4] sljedeće: senzor očitava prepreke na minimalnoj udaljenosti od dva centimetra, a na maksimalnoj udaljenosti očitava u iznosu od 30 cm. Maksimalni kut usmjerenosti pod kojim se elektromagnetski val može detektirati iznosi  $35^\circ$ . To je raspon udaljenosti unutar kojeg može upasti val u prijemnu komponentu senzora, a da senzor „prepozna“ prepreku. Udaljenost detekcije prepreke se vrši na dva načina: zakretanjem u smjeru kazaljke na satu postiže se povećanje udaljenosti, a u suprotnom smjeru smanjenje udaljenosti detekcije. Nadalje, prema [4], refleksija vala ovisi i o obliku prepreke. Neke prepreke odbijaju val u smjeru u kojemu se ne može postići detekcija. Riječ je uglavnom o preprekama nepravilnog oblika, na kojemu dolazi do raspršenja vala (kada se istovremeno pojavi refleksija, difrakcija i refrakcija vala). Međutim, u domaćinstvima su većina predmeta pravilnog oblika, tako da se jako rijetko događa raspršenje vala. U poglavlju 4. se pokazalo da mobilni robot u domaćinstvu prilikom testiranja niti jednom nije došao pred prepreku da ju nije detektirao. Također, kako je riječ o sensorima elektromagnetske prirode, u nekim slučajevima dolazi i do apsorpcije vala. Apsorpcija elektromagnetskog vala je pojava pretvaranja upadnog vala u toplinu. Dolazi do povećanja stope titranja atoma u objektu, odnosno povećanja temperature objekta. Iako je povećanje temperature zanemarivo malo, budući da snažni komunikacijski odašiljači povećavaju prosječnu temperaturu objekata tek na otprilike trećoj decimali temperature objekta i to predstavlja tek posljednicu ovog fenomena, uzrok je u ovom slučaju boja objekta. Kako je pokazano u poglavlju 4. gdje se vrši testiranje rada mobilnog robota, pokazano je da senzori ostaju neaktivni u udaljenosti detekcije kada robot priđe objektu crne boje. Crna boja objekta ima tendenciju apsorpcije elektromagnetske zrake, dok njena suprotna bijela boja pak ima tendenciju reflektiranja upadne zrake. Na sljedećoj slici je prikazan senzor kao i njegova shema, gdje će se u tekstu ispod slika komentirati rad najvažnijih komponenti senzora.



Slika 3.6: Digitalni infracrveni senzor s dva stanja (izvor: [5])

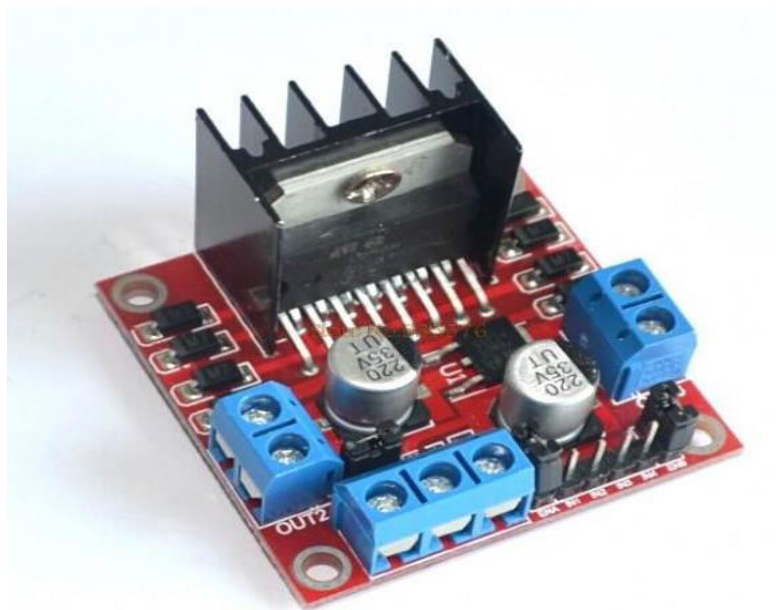


Slika 3.7: Električna shema digitalnog infracrvenog senzora s dva stanja (izvor: [6])

Izlaz senzora se izravno spaja na ulazno – izlazni port mikroupravljača te se koristi napajanje u iznosu od pet volti. Postizanje dva moguća stanja na izlazu se vrši pomoću komparatora LM393 na senzoru. Komparator mijenja stanje s obzirom na referentnu razinu napona i izlaz iz senzora je zapravo izlaz iz komparatora. Senzor je dizajniran da komparator odašilje logičku jedinicu na izlazu dok nema prepreke, a logičku nulu na izlazu kad je prepreka detektirana. Senzor se sastoji od tri LE diode. Jedna LE dioda označava stanje aktivnosti senzora kada je priključena na napajanje, odnosno to je indikator napajanja (eng. *power indicator*). Infracrveni primopredajnik (eng. *infrared tube*) se sastoji od dvije LE diode. LE dioda prema čijem se simbolu vidi da odašilje elektromagnetski val, predstavlja infracrvenu diodu, odnosno predajnu diodu koja odašilje elektromagnetski val u infracrvenom spektru. Druga LE dioda je prijemna fotodioda, odnosno te dvije diode čine infracrveni primopredajnik. Fotodioda provodi struju kada s njom u kontakt dođe upadna infracrvena zraka. Četvrta LE dioda, odnosno indikator promjene stanja, predstavlja diodu koja crvenim svjetlom označava pojavu detekcije prepreke.

### 3.6 SKLOP POKRETAČA MOTORA

Odabir sklopa pokretača motora (eng. *motor driver*) se vrši prema tome koji će motori koristiti za mobilnog robota. U ovom diplomskom radu je odabran sklop pokretača motora naziva LM298N. Taj sklop je prikazan na sljedećoj slici.



**Slika 3.8:** Sklop pokretača motora koji se koristi u radu (izvor: [11])

Iako je ovo zapravo višenamjenski sklop koji se može koristiti i za neke druge svrhe, prema definiciji, ovaj sklop je prema [12] integrirani monolitski krug koji podržava visoke napone i struje te je dizajniran da prihvaća standardne TTL (eng. *transistor-transistor logic*) logičke razine napona i struja. Također je dizajniran da pogoni induktivne terete kao što su releji, solenoidi, istosmjerni motori i koračni motori. Sklop je dizajniran tako da sadrži dva *pin*a koja ovisno o stanju *pinova* omogućava ili ne omogućava rad sklopa. S jednim setom *pinova* se upravlja smjerom vrtnje istosmjernih motora, a postoji i jedan set *pinova* koji primaju *PWM* signal (eng. *pulse – width modulation*) za određivanje brzine vrtnje istosmjernih motora ili nekog drugog uređaja. Detaljnije o *PWM* modulaciji, odnosno o pulsno – širinskoj modulaciji može se vidjeti u potpoglavlju 3.6.2. Ovaj sklop također podržava i izlazni način rada za napajanje ostalih uređaja, gdje baterijski sklop preko ovog sklopa, ukoliko su potrebne razine napajanja od 3.3 V ili 5 V, napaja ostale uređaje u sustavu u kojem se koristi ovaj sklop. Drugim riječima, sklop konkretno služi kao transformator koji prilagođava ulazne vrijednosti napona i struja na te dvije izlazne vrijednosti koje su navedene. Sklop usmjerava napon iz baterijskog modula u motore spojene na ovaj sklop na način na koji mu odredi mikroupravljač. Stoga se motori ponašaju na način da se pokreću i ubrzavaju i mijenjaju smjer na onaj način na koji mu zada kontrolna jedinica, odnosno u ovom slučaju mikroupravljač. Razlog odabira ovog sklopa leži u njegovoj jednostavnosti izvedbe gdje se s lakoćom postiže upravljanje istosmjernim motorima koji se koriste u mobilnom robotu, tj. u praktičnom dijelu ovog diplomskog rada. Ovaj sklop se često koristi u jednostavnim primjerima mobilnih robota jer je pogodan za entuzijaste koji pokazuju interes za izradu bilo kakvih električnih sklopova budući da je izveden tako da značajno pojednostavljuje način na koji se upravlja istosmjernim motorima. Način na koji je sklop povezan u mobilnom robotu s ostalim sklopovljem može se vidjeti u potpoglavlju 3.8.

### 3.6.1 Parametar radni ciklus

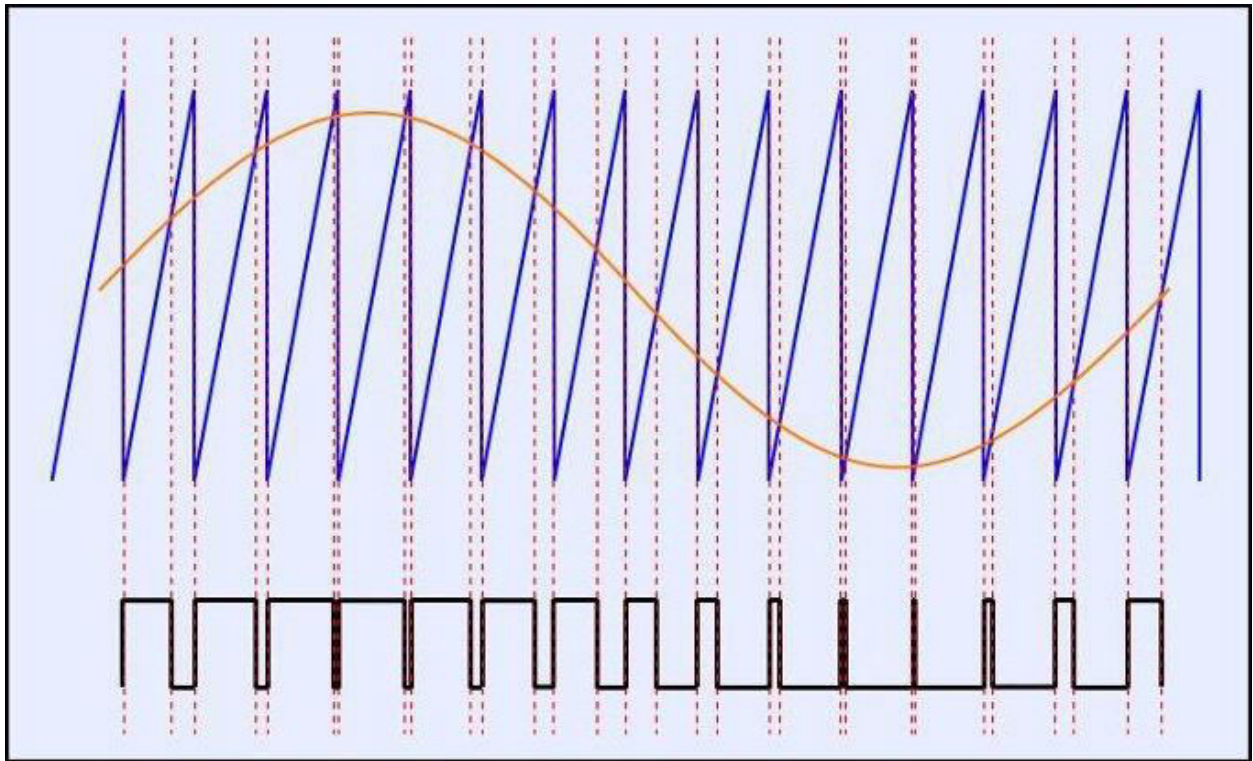
Parametar pod nazivom radni ciklus (eng. *duty cycle*) je prema [13] postotak periode u kojem je signal ili sustav aktivan. Aktivnost se određuje u odnosu na neku referentnu vrijednost, a prema tome i neaktivnost. Perioda je vrijeme kojem treba signalu da završi ciklus aktivnog i neaktivnog stanja. Matematički se radni ciklus izražava na sljedeći način:

$$D = \frac{T}{P} \times 100 \% \quad (3 - 3)$$

gdje parametar  $D$  predstavlja radni ciklus,  $T$  predstavlja vrijeme u kojem je signal u aktivnom stanju, dok  $P$  predstavlja ukupni period signala. Ukoliko npr. parametar radnog ciklusa iznosi  $D = 60 \%$ , utoliko je signal u aktivnom stanju toliki postotak vremena, dok je signal u neaktivnom stanju  $40 \%$  vremena, što predstavlja razliku ukupnog postotka i postotka aktivnosti. U realnom svijetu,  $60 \%$  radnog ciklusa u aktivnom stanju može značiti da je neki sustav ili signal, zbog ukupnog vremena  $T$ , samo djelić sekunde aktivan, a može značiti i da je aktivan i po nekoliko sati, tjedana itd. Sve ovisi o tome o kojem ukupnom vremenu  $T$  se govori. Parametar radni ciklus se može koristiti za opis nekog sustava ili signala u kojem se može vidjeti aktivnost električnog uređaja ili signala unutar nekog vremena. U ovom slučaju, ovaj parametar je nužan u diplomskom radu zbog toga što se pomoću tog parametra određuje brzina vrtnje istosmjernih motora. U sljedećem potpoglavlju o pulsno – širinskoj modulaciji, govorit će se o tome kako taj parametar utječe na promjenu brzine vrtnje motora i zašto se ta modulacija koristi za istosmjerne motore.

### **3.6.2 Pulsno – širinska modulacija**

Pulsno – širinska modulacija (eng. *Pulse – width modulation*) je digitalna modulacijska tehnika koja se koristi za kodiranje neke poruke u digitalni impulsni signal. Iako se ova modulacijska tehnika koristi za kodiranje informacije za prijenos kroz neki medij, glavna mu je namjena kontrola snage koja se daje električnim uređajima kao što su inercijski tereti poput električnih motora. Tom modulacijom se postiže kontrola isporuke snage na način da se može određivati brzina vrtnje električnog motora. Proces pulsno – širinske modulacije ima sljedeći oblik i opisana je na način koji je prikazan na sljedećoj slici.

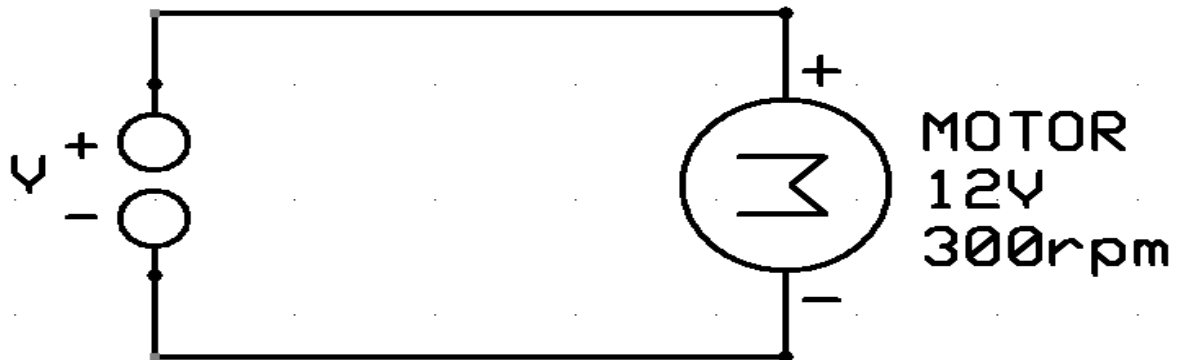


**Slika 3.9:** Prikaz nastajanja pulsno – širinskog signala pomoću pilastog signala i sinusoidalnog signala (izvor: [14])

Na slici se može vidjeti pilasti signal označen plavom bojom, zatim sinusoidalni signal označen narančastom bojom i pravokutni signal označen crnom bojom. Pilasti signal se općenito koristi za generiranje pulsno – širinskog signala i predstavlja se kao pomoćni signal dok se sinusoidalni signal predstavlja kao modulacijski signal. Modulacijski signal je signal koji modulira pomoćni signal i tim postupkom se dobiva modulirani signal. Modulacijski signal je informacijski signal, odnosno signal koji nosi informaciju. Modulirani je dakle, pravokutni signal koji nastaje kombiniranjem tih dvaju signala. Način na koji se ta kombinacija izvršila je sljedeća: sa slike 3.8 se vidi da kada sinusni signal sječe pilasti signal u smjeru iznutra prema van, u tom trenutku započinje generiranje pravokutnog signala u stanju logičke jedinice. Logička jedinica se završava kada sinusni signal presječe pilasti signal u smjeru izvana prema unutra i u tom trenutku započinje logička nula. Logička nula završava u trenutku kada sinusni signal ponovno presječe pilasti signal u smjeru iznutra prema van i ciklus se nastavlja. Trajanja logičke jedinice i nule ovise o obliku modulacijskog signala i na slici se može vidjeti o čemu zapravo ovisi trajanje logičkih razina i zašto je u nekim slučajevima to vrijeme kraće, a u nekim slučajevima to vrijeme duže. Razlog zašto je ova modulacija korisna za električne motore se može najbolje opisati na

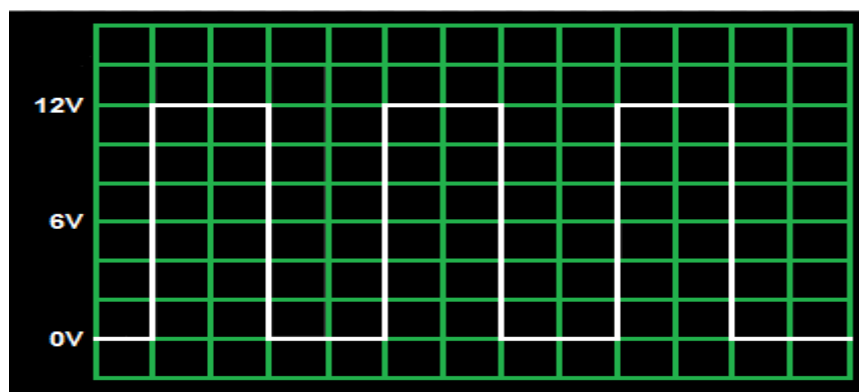


sljedeći način. Neka se pretpostavi da je istosmjerni motor spojen s izvorom na sljedeći način sa podacima o motoru koji su upisani u sljedeću sliku.



**Slika 3.10:** Istosmjerni električni motor u serijskom spoju s istosmjernim izvorom (izvor: [15])

Iz slike 3.10 se vidi da se električni motor može pobuditi s maksimalnim naponom od 12 V, a pri tome naponu razvija brzinu okretanja od 300 okretaja u minuti (eng. *revolutions per minute*, u daljnjem tekstu će se koristiti mjerna jedinica rpm). Ako se na motor primijeni napon od 6 V, razvijat će upola manju brzinu okretanja koja iznosi 150 rpm (pretpostavljajući idealne uvjete rada). Pritom grafički prikaz tog napona je nepromjenjiv u vremenu i u svakoj točki vremena iznosi točno 12 V ili 6 V (pretpostavljajući idealne uvjete). No međutim, ukoliko se na istosmjerni motor primijeni naponski signal sljedećeg oblika, situacija se mijenja. Sljedeća slika opisuje taj naponski signal.



**Slika 3.11:** Prikaz naponskog signala koji se dovodi na istosmjerni motor. Signal je dobiven pulsno – širinskom modulacijom referentnog signala unutar mikroupravljača (izvor: [16])

Na slici 3.11 se može vidjeti pulsno – širinski signal koji značajno mijenja vrijednost iz logičke nule u logičku jedinicu (uz pretpostavku idealnih uvjeta, promjena logičkih stanja se događa beskonačno brzo) i obratno. Kada se ovakav signal dovede na motor, ne događa se da se u periodu logičke jedinice motor vrti, a u periodu logičke nule da se motor zaustavi i da se ta radnja nastavlja kontinuirano dok god je signal narinut na istosmjerni motor. U fizici je poznato da svako tijelo posjeduje nekakvu inerciju. Tako je poznato i da kad je na motor narinuta logička jedinica, osovina motora se ubrzava (ne ubrzava se na krajnju vrijednost vrtnje trenutno) određeno vrijeme, i to vrijeme ubrzavanja je jednako vremenu trajanja logičke jedinice. Kada logička nula nastupi, motor usporava svoju vrtnju, ali ne zaustavlja se trenutno. Prije zaustavljanja motora, ponovno se narine logička jedinica na električni krug motora. Budući da se promjene događaju izuzetno brzo za ljudsko poimanje vremena i da je usporenje vrtnje motora zapravo neznajčajno da bi se išta primijetilo budući da se radi jednostavan primjer mobilnog robota, motor će nastaviti s ubrzavanjem vrtnje, ali nešto sporije nego kad bi bio narinut konstantan napon na krug motora. U kratkom vremenu, električni motor postiže svoju krajnju vrijednost vrtnje. Ukoliko se na motor narine ovakav naponski signal vršne vrijednosti iznosa 12 V, postiže se brzina vrtnje koja je upola manja od maksimalne vrijednosti vrtnje, a iznosi 150 rpm. Dakle, kao da se narine vremenski konstantan napon iznosa 6 V. Ukoliko se mijenja vrijeme trajanja logičke jedinice i logičke nule, neovisno o narinutom naponu na električni motor, utoliko se može utjecati na brzinu vrtnje motora. Parametar koji se koristi u ovom slučaju je vrlo značajan i naziva se radni ciklus, koji je opisan u prethodnom potpoglavlju. Ukoliko je radni ciklus veći, povećava se srednji iznos naponskog signala koji je narinut na motor i posljedično se povećava magnetski tok unutar armaturnih zavoja te se motor posljedično brže vrti. Vrijedi i obratno. Treba napomenuti da se korištenjem ovakve modulacije prema [17] postiže da se amplituda motora ne mijenja i da motor zapravo radi u punoj snazi. Mijenjanjem parametra radnog ciklusa se postiže da se unatoč konstantnoj amplitudi, mijenja brzina vrtnje na potreban način bez pojavljivanja nepravilnosti u radu motora, budući da radi u punoj snazi jer se na njega narine maksimalan napon. Najveća prednost pulsno – širinske modulacije je u tome da je gubitak snage u sklopkama mali. Kada je sklopka otvorena (kada je isključen električni krug), praktički nema struje kroz sklopku, no dok je sklopka zatvorena s električnim krugom (kada je uključen električni krug), snaga se prenosi na teret i gubitak napona na sklopki je veoma mali. Gubitak snage je u oba slučaja veoma mali i po iznosu je blizu nuli. Što se tiče digitalnih sklopki, pomoću njih se može preciznije postići radni ciklus koji je tražen s minimalnim gubitkom snage na sklopkama. Pulsno – širinska modulacija se u mobilnom robotu u sklopu ovog diplomskog

rada postigla pomoću registara posebne namjene unutar mikroupravljača koji je prethodno opisan. Način na koji se to ostvarilo je prikazano u potpoglavlju 3.9.

### **3.7 ODABIR POBUDNOG SKLOPA MOBILNOG ROBOTA**

Pobudni sklop mobilnog robota je veoma važan zbog postizanja njegove autonomnosti u smislu da nije žično povezan s nekim izvorom napajanja, jer se time ograničava njegova mobilnost i sposobnost rada u posebnim okruženjima. Budući da je zadatak mobilnog robota izbjegavanje prepreka u domaćinstvu, važno je da robot postigne što veću autonomnost. To se ostvarilo odabiranjem baterijskih ćelija koje zadovoljavaju potrebe ostalog sklopovlja unutar mobilnog robota za naponom i strujom. Određivanje napona i struje koje baterije moraju osigurati se prema [10] vrše iterativno. Dakle, metodom pokušaja i pogreške. Dimenzije baterijskog sustava, napon koji taj sustav osigurava i mogućnost ponovnog punjenja baterije stvaraju ograničenja na cjelokupan proces dizajniranja mobilnog robota te utječu na autonomnost robota. Neke baterije posjeduju manji kapacitet, pa će robot trebati češće punjenje i češće nadgledanje nadležne osobe tog mobilnog robota, što neće biti toliko potrebno u suprotnom slučaju. Budući da je u ovom diplomskom radu cilj napraviti jednostavan primjer mobilnog robota, bitno je da taj baterijski sustav zadovoljava osnovne zahtjeve svih sklopovlja za strujom i naponom. Svaki sklop unutar mobilnog robota zahtijeva standardnu razinu napajanja od 5 V, a to uključuje i istosmjerni motore. Međutim kako motori koriste veću količinu struje prilikom rada (iako se jednim dijelom ta potrošnja smanjuje korištenjem pulsno – širinske modulacije), potrebno je riješiti problem baterijskog sustava na način da robot bez obzira na potrebne iznose napona i struja za svaki sklop radi besprijekorno i da se osovine električnih motora vrte jednakom brzinom, da robot tokom kretanja zadržava svoj smjer i da ne skreće ulijevo ili udesno. Baterijski sustav mobilnog robota je po dimenzijama najveći od svih sklopova i posjeduje najveću masu. To može utjecati na centar mase mobilnog robota, pa i na kretanje samog robota. Ukoliko istosmjerni motori nisu dovoljno jaki, neće se postići zadovoljavajuće velika brzina kretanja mobilnog robota. Taj problem se u ovom konkretnom slučaju, riješio postavljanjem baterijskog sklopa što više odozad, kako je prikazano u potpoglavlju 3.9. Na taj način se osigurao prostor ostalom sklopovlju i povećala se stabilnost robota na način da je centar mase smješten prema sredini robota. Također, i pozicioniranjem ostalih sklopova se nastojao postići takav centar mase. Električni istosmjerni motori koji se koriste u mobilnom robotu su dovoljno snažni da podnesu baterijski sustav i da se robot kreće zadovoljavajućom brzinom. To je postignuto metodom

pokušaja i pogreške, što je također opisano u potpoglavlju 3.9. Treba napomenuti da kako raste kompleksnot robota, svaki višak mase ili nepravilno smješten pojedini sklop unutar strukture mobilnog robota koji negativno utječe na centar mase robota, odnosno na njegovu ravnotežu, treba obratiti pažnju i na potrošnju sklopova te na masu i pozicioniranje svih sklopova kako bi se postigle optimalne performanse. U ovom slučaju, to nije bilo potrebno zbog tog što su dimenzije robota male i ti parametri zanemarivo utječu na performanse robota koji je ostvaren u sklopu ovog diplomskog rada. Danas je popularno koristiti naponski izvor za mobilnog robota koji ostvaruje 12 V, no za ovaj slučaj odabrala se količina od četiri baterije koje pojedinačno ostvaruju napon od 1.5 V, a u serijskom spoju ostvaruju 6 V. To je dovoljno da se napajaju svi sklopovi unutar mobilnog robota. Struja koju pružaju te baterije je dovoljna da se određeno vrijeme pogoni mobilni robot. To je zaključeno metodom promatranja potrošnje baterija koje se jednostavno provjeravaju ugrađenim mjeračem u baterijama. Baterije koje su korištene su baterije proizvedene u tvrtki Duracell, a prikazane su na sljedećoj slici. Razlog korištenja tih baterija je opisan u sljedećem potpoglavlju.

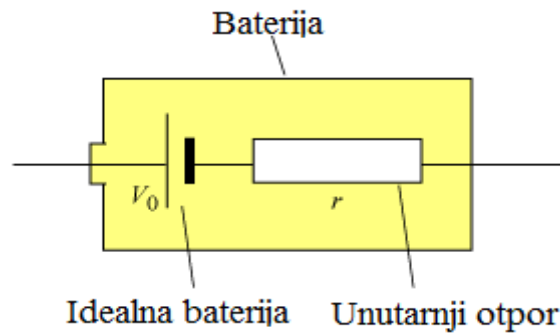


**Slika 3.12:** Jedna od baterija tvrtke Duracell korištene u mobilnom robotu

### **3.7.1 Vrste baterija i primjena u mobilnom robotu**

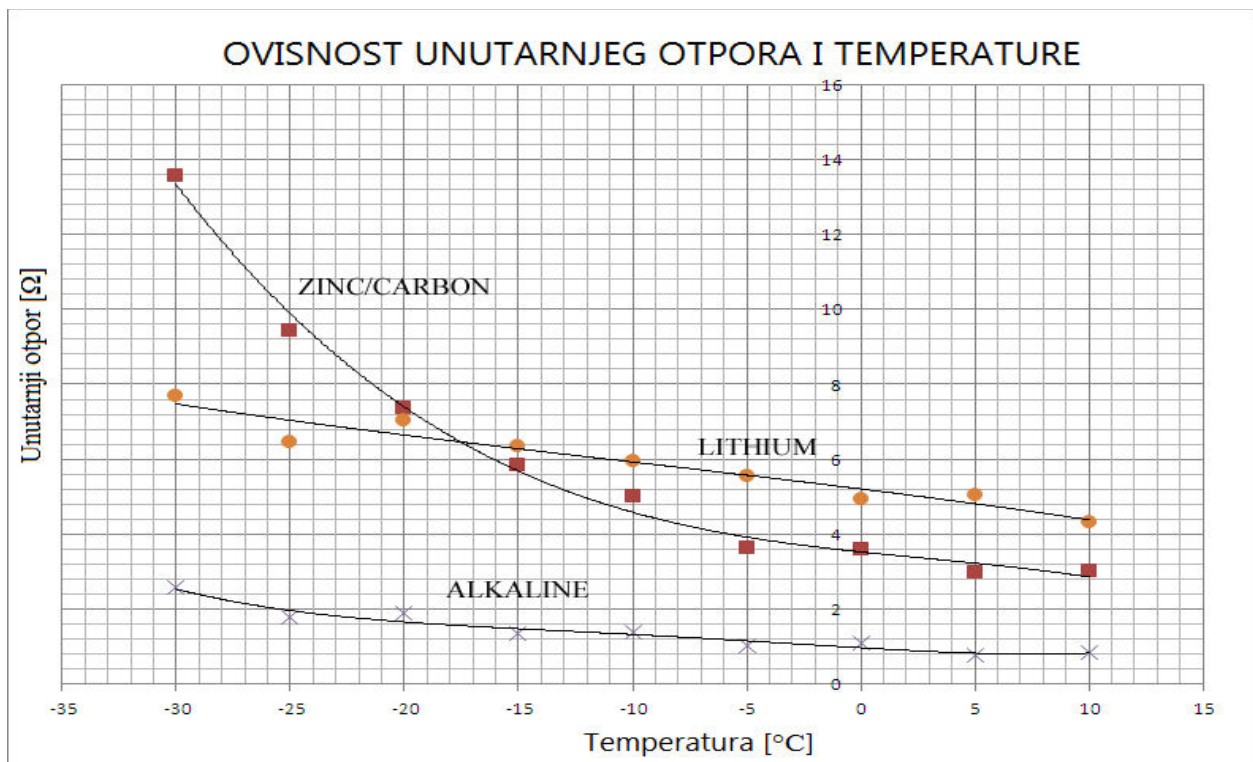
Električna baterija je prema [23] niz spojenih galvanskih članaka, no međutim može obuhvaćati i samo jedan takav članak. Galvanski članci su dobili ime po talijanskom znanstveniku Luigiju Galvaniju, a omogućuju pretvaranje kemijske energije u električnu. Princip rada električne

baterije je takav da se na jednom njenom dijelu gomila pozitivan, a na drugom dijelu negativan naboj. To se npr. može postići fizikalno – kemijskom reakcijom u Leclanche – ovom članku, odnosno u bateriji. Leclanche – ov članak je vrsta galvanskog članka koji je dobio ime po istoimenom francuskom inženjeru. Sadrži elektrodu od cinka u koju se smješta kemijski aktivna tvar, odnosno elektrolit i u njemu uronjeni ugljeni štapić. Zahvaljujući fizikalno – kemijskim procesima u elektrolitu, stvaraju se potencijali u iznosu od 1.5 V na krajevima. Pri tome se negativni naboj gomila na cinčanom plaštu, a na ugljeni štapić pak pozitivan naboj. U principu su te baterije jednokratne. Nakon trošenja se više ne mogu koristiti. Što se tiče kategorizacije baterija, treba napomenuti da svoje nazive dobivaju prema materijalima od kojih su načinjene, a najpoznatije su sljedeće vrste: ugljen – cink (tzv. Leclanche – ov članak ili „obična baterija“), alkalna baterija, živina baterija, srebrna (srebro – oksidna) baterija i litijeva baterija. Postoje i punjive baterije, koje se mogu višekратно puniti nakon pražnjenja. To su tzv. aku – baterije (akumulatorske baterije). Oblici baterija se označavaju standardnim oznakama, a kod rijede korištenih tipova tvorničkim oznakama. U primjeni su američke, europske i druge vrste oznaka. Baterija tvrtke Duracell koja je korištena u mobilnom robotu spada u kategoriju alkalnih baterija. To su primarne baterije, koje mogu, ali i ne moraju biti punjive. Prema [24], razvijene su 1940. godine i bile su otrovne zbog visoke koncentracije žive i kadmija u svojoj strukturi. Bez obzira na to, bile su do osam puta trajnije od baterija s cinkovim i bakrenim elektrodama. Katoda u alkalnim baterijama je od manganovog dioksida, a anoda je od cinka u prahu. Nazivaju se alkalnim jer je elektrolit unutar strukture baterije lužina (npr. kalijev hidroksid). U usporedbi s cink – ugljikovim baterijama koje su razvijene od mokrog Leclanche-ovog članka ili cink – klorid baterij, alkalne baterije su veće energetske gustoće i dužeg roka trajanja od tih baterija uz pretpostavku da oslobađaju na svojim polovima jednak potencijal. Npr. jedna dugmasta baterija koja se temelji na srebrovom oksidu je veće energetske gustoće i kapaciteta od alkalnih baterija, no međutim, znatno je skuplja od alkalnih baterija iste veličine. Alkalne baterije podnose pražnjenje većim strujama, otporne su i posjeduju visoke temperaturne koeficijente. Alkalne baterije se primjenjuju također u kućanstvu za daljinske upravljače, svjetilje, digitalne kamere, igračke itd. Treba napomenuti da se mogu izrađivati u obliku puceta ili cilindra. U ovom slučaju u mobilnom robotu je korištena baterija u obliku cilindra, kako se i vidi na slici 3.12. Na sljedećoj slici je predstavljena električna shema baterije s unutarnjim otporom, a čiji se opis nalazi ispod slike.



**Slika 3.13:** Električna shema baterije s unutarnjim otporom (izvor: [27])

Što se tiče razmatranja unutarnjeg otpora nekog električnog izvora, odnosno u ovom slučaju baterije, prema [25] je izrazito značajan parametar. Prema tome parametru treba donijeti odluku o tome kakvo je opterećenje dopušteno spajati na izvor, a da se osigura optimalno prilagođenje po naponu, struji ili snazi, odnosno da se sačuva izvor od toplinskog uništenja i preopterećenja. Što je neki robot po svojoj strukturi i funkcionalnosti kompleksniji, što je veći broj komponenata od kojih se sastoji i što je kompleksniji algoritam rada, tako proporcionalno raste potreba za napajanjem i nabavkom kvalitetnijih baterijskih sustava. Slika ispod prikazuje ovisnost unutarnjeg otpora različitih vrsta baterija o temperaturi.

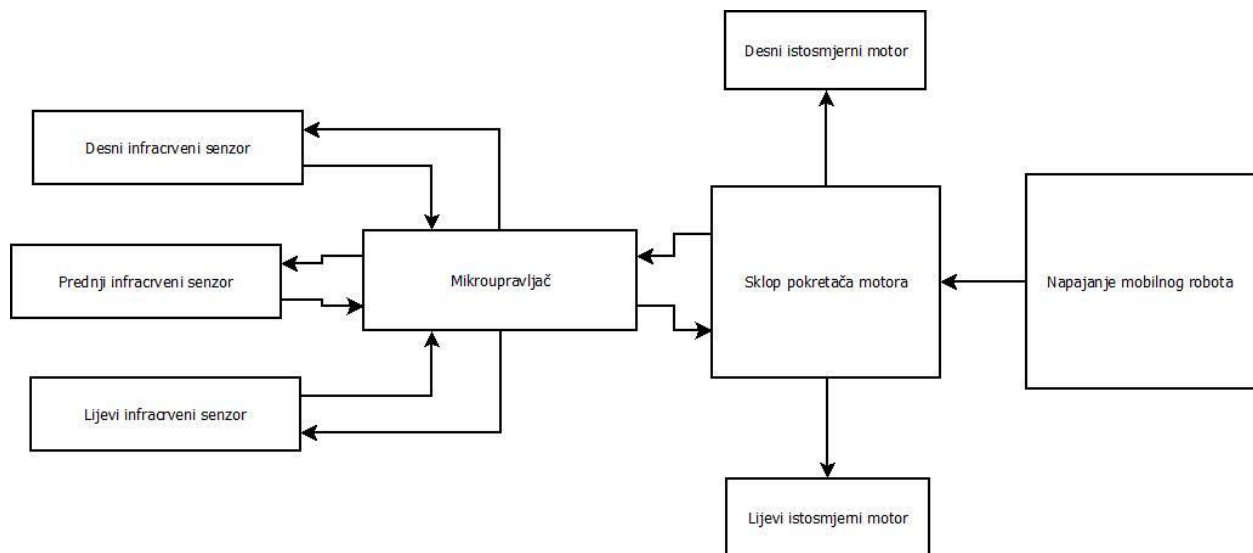


**Slika 3.14:** Ovisnost različitih vrsta baterija o temperaturi (izvor: [26])

Iz slike 3.14 se vidi da alkalinska baterija pokazuje minimalnu promjenu unutarnjeg otpora pri rasponu temperatura od  $-30^{\circ}\text{C}$  do  $10^{\circ}\text{C}$ . Ostalim baterijama ti iznosi variraju i duž cijele definirane domene mjerenja posjeduju nešto više vrijednosti unutarnjeg otpora. Iz tog razloga je odabrana ta vrsta baterija za izradu baterijskog sustava mobilnog robota.

### 3.8 BLOK SHEMA MOBILNOG ROBOTA

Nakon određivanja potrebnih komponenti u mobilnom robotu, pristupa se razradi blok sheme svih sklopovlja unutar mobilnog robota. Blok shema mobilnog robota koji je izrađen u sklopu ovog diplomskog rada je prikazana na sljedećoj slici. Opis blok sheme se nalazi ispod slike.



**Slika 3.15:** Blok shema mobilnog robota

Na slici 3.15 se mogu vidjeti najznačajnije komponente mobilnog robota. Ostale komponente su strukturne prirode i ne igraju značajnu ulogu u funkcionalnosti robota. Tu se radi se o pomoćnom kotaču koji se slobodno rotira oko svoje osi kada robot odrađuje proceduru skretanja kod nailaska na prepreku i kod podvozja robota koje igra ulogu „kostura“ mobilnog robota. Pomoćni kotač je nužan iz razloga ravnoteže, budući da je „kostur“ robota tako dugačak da ovisno o raspodjeli mase na robotu, dolazi do naginjanja na stranu na kojoj je distribuirana veća količina mase, odnosno gdje je centar mase prema jednom od krajeva mobilnog robota. Mobilni robot se ne može se balansirati na dva kotača koje pokreću istosmjerni motori. Pomoćni kotač ne pokreće nijedan motor, već se samo rotira kako robot skreće. Što se tiče značajnih komponenti, može se vidjeti kako su spojeni pojedini entiteti na mobilnom robotu. Entiteti su povezani negdje

jednostrukim, a negdje dvostrukim strelicama s naznačenim smjerom. Smjer strelice određuje komunikaciju i pokazuje gdje je početak, a gdje kraj. Blok naznačen nazivom „Napajanje mobilnog robota“ posjeduje strelicu koja označava smjer u kojem se vrši napajanje. Napajanje dolazi na sklop pokretača motora. Iz sklopa pokretača motora se vidi strelica čiji smjer pokazuje na blok naziva „Mikroupravljač“. Sklop pokretača motora napaja i pokreće istosmjerne motore koristeći pulsno – širinsku modulaciju kako je prikazano strelicama. Mikroupravljač se napaja preko sklopa pokretača motora, budući da taj sklop može funkcionirati kao pristupna točka gdje svi elementi mobilnog robota uzimaju električnu energiju koju pretvaraju u rad čija je priroda rada jednaka namjeni tog elementa. Nadalje, taj mikroupravljač napaja i blokove koje predstavljaju senzore smještene na robotu tako da pokrivaju određeno područje robota u potrazi za preprekama. Međutim, može se vidjeti i kako senzori posjeduju strelice koje pokazuju prema mikroupravljaču. Te strelice predstavljaju informacije koje se šalju mikroupravljaču, koje se zatim analiziraju u algoritmu i donosi se odluka s obzirom za trenutno stanje senzora.

### **3.9 REALIZACIJA MOBILNOG ROBOTA**

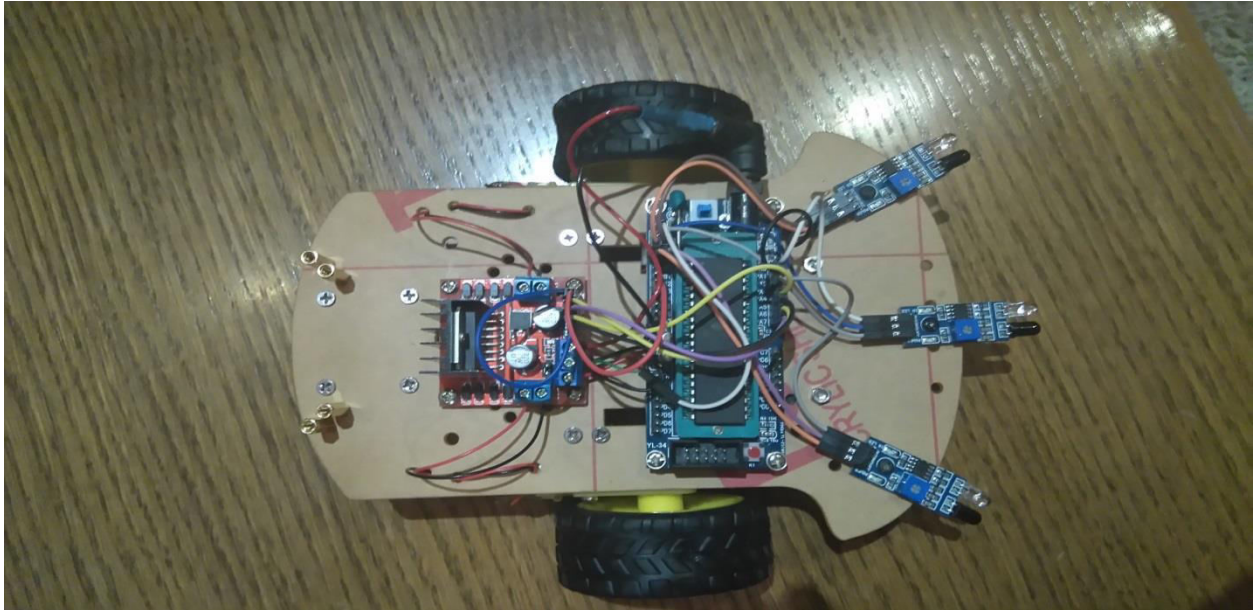
Realizacija mobilnog robota u sklopu ovog diplomskog rada je nastupila kada se odredila njegova svrha, razradila logika donošenja odluka i kada je nacrtana blok shema robota. U procesu realizacije robota, prvi korak je bila nabavka potrebnih komponenti kako bi se konstruirao mobilni robot. Nabavka komponenti mobilnog robota je obavljena na način da su se kupile one komponente koje su opisane u prethodnim poglavljima u za to zaduženim trgovinama. U sljedećem potpoglavlju je opisana konstrukcija mobilnog robota i razmještaj pojedinih električnih sklopova po konstrukciji.

#### **3.9.1 Realizacija sklopovlja mobilnog robota**

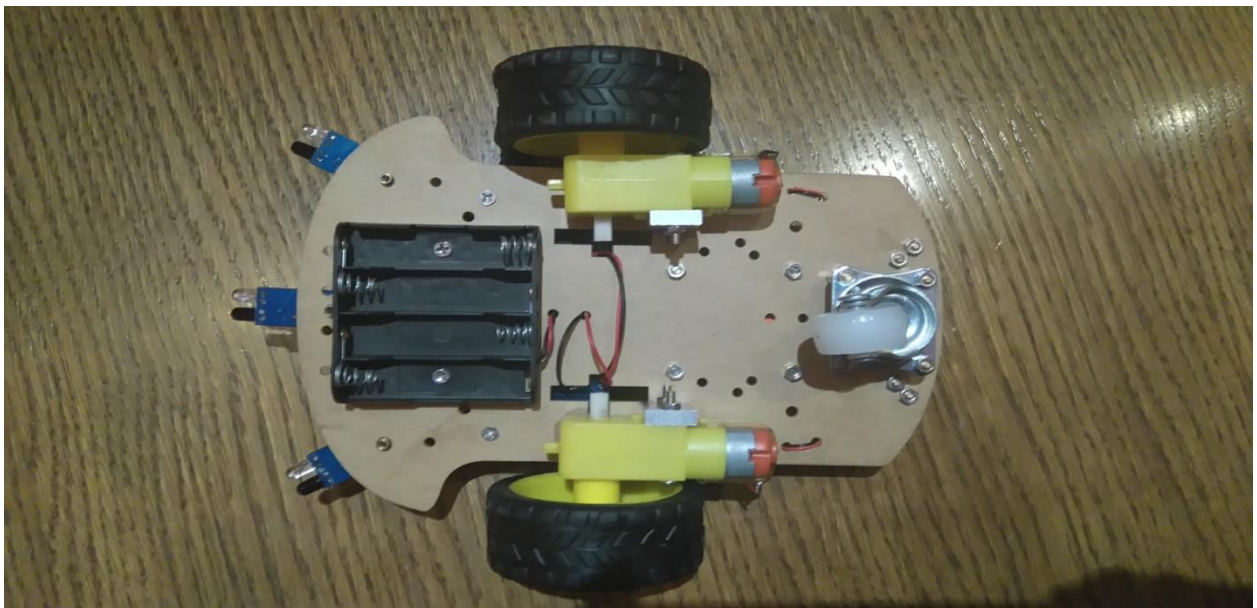
Mobilni robot je konstruiran na način da se sastoji od tankog podvozja na kojem su pričvršćene električne komponente i kotači. Radi uštede prostora, skladnijeg izgleda i praktičnog razmještaja, na gornjem dijelu podvozja se nalaze infracrveni senzori, iza njih mikroupravljač, te sklop pokretača motora. Ispod podvozja se nalazi baterijski sustav, iza njega se nalaze lijevi i desni istosmjerni motor koji su preko reduktora osovinom spojeni na plastične kotače s gumom koji pogone robota. Smještanjem tog sustava ispod podvozja se omogućuje dobivanje na prostoru gornjeg dijela podvozja gdje mogu smjestiti dodatne komponente. Budući da je baterijski sustav smješten više prema prednjem kraju na donjem dijelu podvozja, centar mase je pomaknut prema



naprijed te se stoga robot nagnjao tokom faze realizacije. Naknadno su dodani utezi smješteni tako da djeluju kao protuteža baterijskom sustavu, budući da posjeduje najveću masu. Utezi su smješteni na gornjem dijelu podvozja straga u obliku metalnih stalaka, dok se u donjem dijelu podvozja nalaze između pomoćnog kotača i podvozja. Vizualno gledano, mobilni robot tako izgleda skladnije i u slučaju uvođenja nekog poboljšanja ili otkrivanja kvara, odnosno eventualno nepoželjne radnje, skidanje komponente kao i otkrivanje kvara je pojednostavljeno. Mobilni robot u konačnoj fazi izrade se nalazi na slici ispod s opisom ispod te slike.



**Slika 3.16:** Mobilni robot u konačnoj fazi izrade – gornji dio podvozja



**Slika 3.17:** Mobilni robot u konačnoj fazi izrade – donji dio podvozja

Nakon konstruiranja mobilnog robota, pristupilo se programiranju mobilnog robota tako da napisani program zadovoljava postavljeni algoritam prema kojemu će robot donositi odluke kada naiđe na prepreku. Detaljan opis realizacije programske podrške se nalazi u sljedećem potpoglavlju.

### 3.9.2 Realizacija programske podrške

Problemi koji su se pojavili kod realizacije mobilnog robota, a trebalo je riješiti, su određivanje udaljenosti detekcije prepreke kod senzora i određivanje vrste pulsno – širinske modulacije i određivanje parametra radnog ciklusa kod ostvarenja pokretanja istosmjernih motora te posljedično i cijelog robota. Udaljenost detekcije prepreke je tvornički određena od neke početne do neke krajnje vrijednosti. U tom intervalu se može odrediti udaljenost detekcije prepreke s obzirom na zadatak. U poglavlju 4. je pokazano na koji su način senzori podešeni po pitanju udaljenosti detekcije prepreke. Što se tiče pulsno – širinske modulacije i određivanja parametra radnog ciklusa, treba napomenuti da je vrsta pulsno – širinske modulacije određena na način koji je opisan u potpoglavlju 3.6.2. dok je međutim parametar radni ciklus određen empirijski. U programu za mikroupravljač (eng. *firmware*) za parametar radni ciklus se odabralo više vrijednosti, te su se zatim istosmjerni motori pokretali i prikladna se brzina odredila tim pristupom, budući da je proračunavanje brzine mobilnog robota u ovom slučaju kompleksan i dugotrajan posao, dok se ovim pristupom štedi vrijeme i pojednostavljuje kompleksnost određivanja tog parametra. Podešavanje parametra radnog ciklusa i uključivanje pulsno – širinske modulacije u mikroupravljaču se vrši na način da se podeše registri mikroupravljača koji na izlazu daju pulsno – širinski signal s određenim iznosom radnog ciklusa. Prije podešavanja registara treba se odrediti vrsta pulsno – širinske modulacije koja će se koristiti. Općenito, postoje tri vrste te modulacije: brza pulsno – širinska modulacija, pulsno – širinska modulacija s korekcijom faze i pulsno – širinska modulacija s korekcijom frekvencije i faze. Brza pulsno – širinska modulacija je modulacija koja koristi kao referentni signal pilasti signal, dok pulsno – širinska modulacija s korekcijom faze koristi trokutasti signal. Pulsno – širinska modulacija s korekcijom frekvencije i faze također koristi trokutasti signal. Ono što razlike te tri modulacije je da brza pulsno – širinska modulacija ima dvostruko veću frekvenciju od ove dvije. Za pokretanje istosmjernih motora je u ovom slučaju nevažno koji se tip modulacije koristi, budući da je efekt na kraju jednak i promatrač ne vidi razliku u pokretanju istosmjernih motora. Za ovaj mobilni robot je odabrana brza pulsno – širinska modulacija. U mobilnom robotu, signal koji je moduliran ovom modulacijom se dobije kao izlaz iz vremenskog sklopa mikroupravljača. U

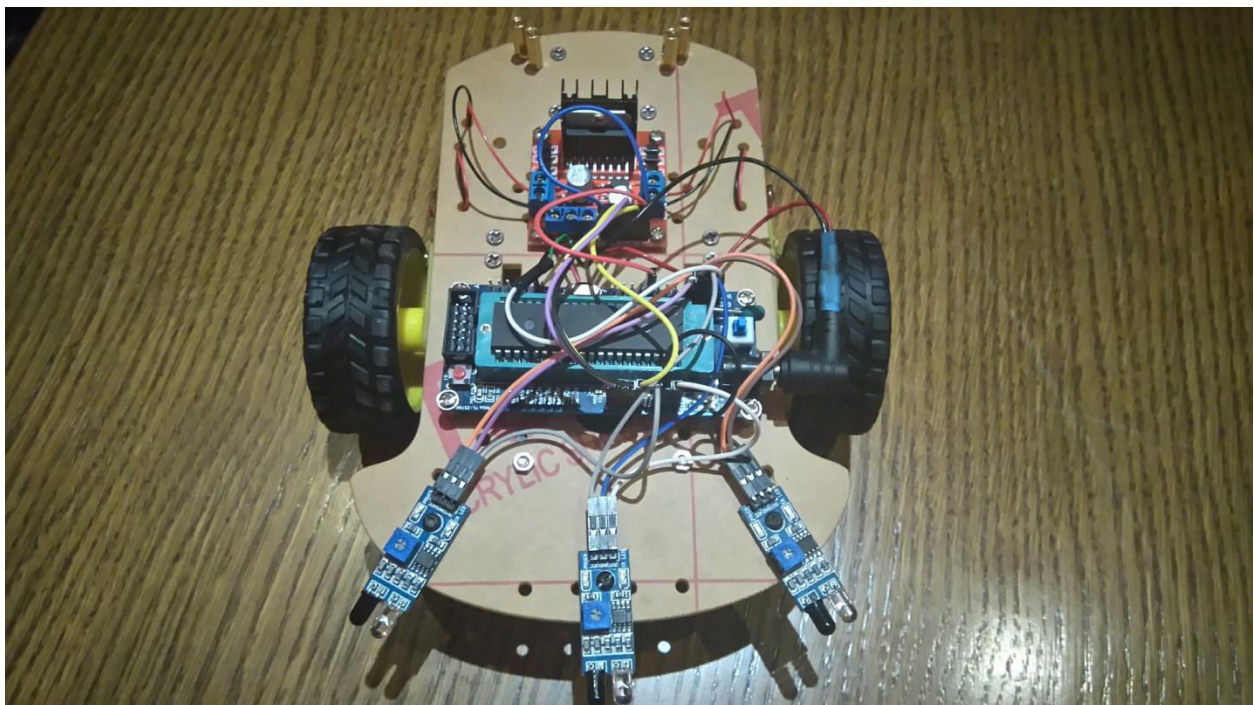
mikroupravljaču samo treba programski podesiti vrstu modulacije i odrediti parametar radni ciklus. Prema dokumentaciji mikroupravljača, odnosno prema [3], vremenski sklopovi TIMER0 i TIMER1 podržavaju generiranje signala moduliranog tom modulacijom. Odabran je 8 – bitni TIMER0 za generiranje tog signala. Razlog tome je što 8 – bitni timer daje  $2^8$  stanja, a što iznosi 256 stanja. Parametar radni ciklus ima maksimum u 256, a minimum u 0. Odabirom bilo kojeg broja unutar tog intervala određuje brinu vrtnje istosmjernih motora. Podešavanje tog vremenskog sklopa je opisano u sljedećem tekstu. Sljedeća slika opisuje registar 8 - bitnog vremenskog sklopa TIMER0 s kojim se podešava modulirani signal opisanom modulacijom za pokretanje istosmjernih motora.

Bit	7	6	5	4	3	2	1	0	
	FOC0	WGM00	COM01	COM00	WGM01	CS02	CS01	CS00	TCCR0
Read/Write	W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

**Slika 3.18:** Prikaz registra 8 – bitnog vremenskog sklopa TIMER0 s kojim se postiže podešavanje izlaznog pulsno – širinskog signala za pokretanje istosmjernih motora (izvor:[18])

Registar se sastoji od 8 različitih stanja, gdje svako stanje na svakom indeksu predstavlja određenu funkcionalnost. Brojevi od 0 – 7 predstavljaju indekse (u daljnjem tekstu koristit će se naziv bit), a kvadratići s upisanim oznakama predstavljaju svako stanje, gdje oznaka predstavlja funkcionalnost. Bit 7 je nevažan za podešavanje registra. Bitovi 6 i 3 predstavljaju vrstu valnog signala. Ti bitovi se mogu postaviti na vrijednosti 00 i 01 ovisno o vrsti pulsno – širinske modulacije koja će se koristiti. U dokumentaciji [3] mikroupravljača postoji tablica stanja po kojoj je za svaku kombinaciju pridijeljena određena vrsta ove modulacije. U ovom slučaju, za brzu pulsno – širinsku modulaciju odabrana je kombinacija 11 (bit 3 i bit 6 postavljeni u logičku jedinicu). Bitovi 5 i 4 predstavljaju mod rada *pina* pod nazivom OC0 (eng. *output compare*), gdje nula predstavlja redni broj, poput TIMER1 i TIMER0. Pod [3] se nalazi tablica stanja s funkcionalnostima za svako stanje. Ovaj registarski sklop unutar glavnog registra određuje kada se događa usporedba i resetiranje brojanja. Usporedba podrazumijeva usporedbu trenutnog stanja vremenskog brojača i zadanog parametra radnog ciklusa, a resetiranje brojanja se vrši kada trenutna vrijednost vremenskog brojača odgovara vrijednosti zadanog parametra radnog ciklusa. Npr. ako se koristi 8 – bitni vremenski sklop, on broji od 0 do 255 s korakom od jedan. 255 predstavlja najviše moguće stanje 8 – bitnog brojila, a 0 najniže. Radni ciklus se bira unutar tog

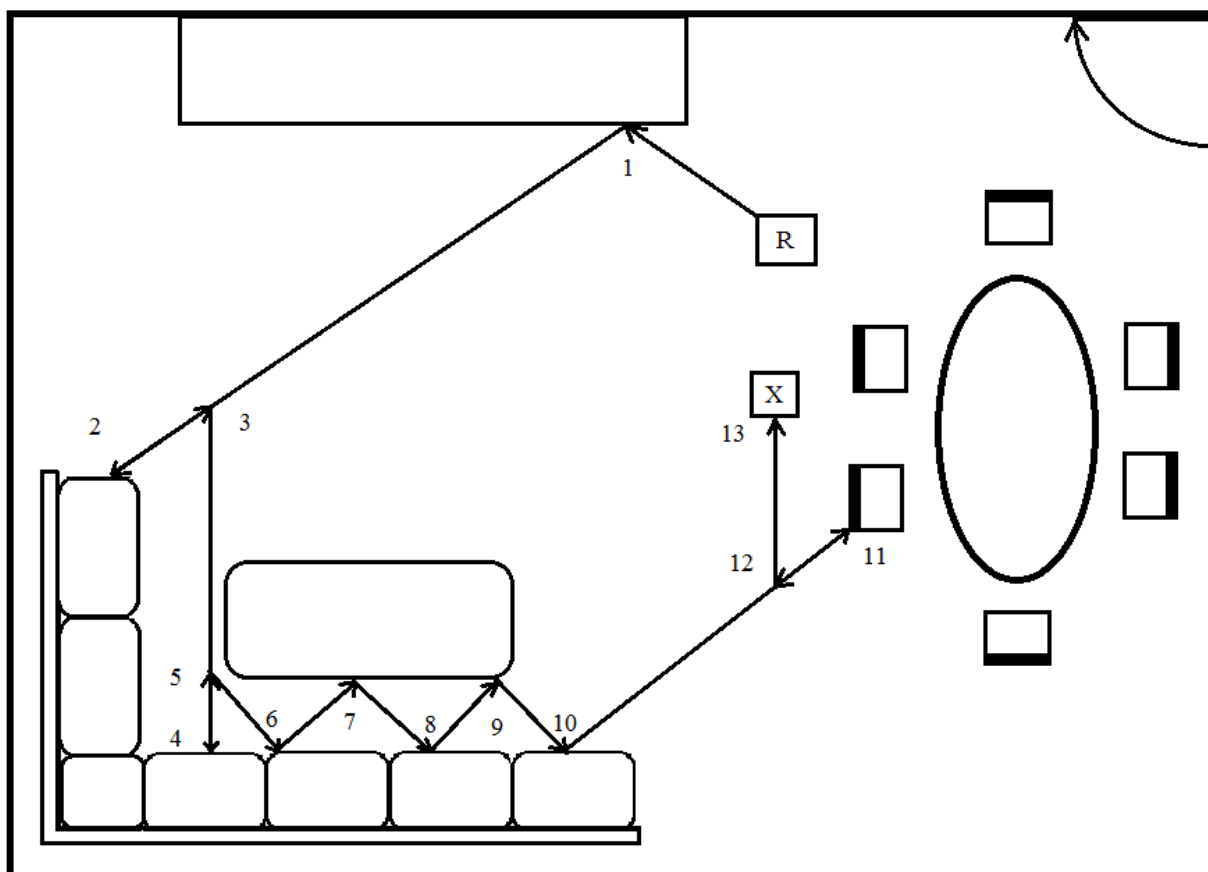
raspona. Kako vrijeme teče, tako se povećava brojilo za jedan korak. Brzina povećanja ovisi o frekvenciji rada mikroupravljača, a odnos je proporcionalan. Ako se npr. za radni ciklus izabere 200, OC0 registar će se ponašati točno onako kako je podešeno. Može se podesiti da se kada se dogodi usporedba (vrijeme brojanja je jednako zadanom radnom ciklusu) resetira brojilo na 0, ili da se npr. broji od maksimalne vrijednosti prema minimalnoj i da se kada se dogodi usporedba, brojač vrati na maksimalnu vrijednost. Mogućnosti su ograničene dokumentacijom. Za ovaj slučaj je odabrana konfiguracija od 00. Registar OCR0 služi za spremanje vrijednosti radnog ciklusa. CS registri predstavljaju modove rada vremenskog sklopa i eventualno korištenje djelitelja frekvencije rada mikroupravljača. Djelitelj dijeli frekvenciju rada mikroupravljača na željenu vrijednost koja se koristi za dobivanje određenog perioda izlaznog signala. U ovom slučaju nije korišten djelitelj, odnosno odabrana je kombinacija CS = 001. U prilogu se može naći programski kod gdje je napisana konfiguracija registara s opisom. Baterijski sustav mobilnog robota zadovoljava postavljene zahtjeve po pitanju napajanja svih električnih komponenti na robotu. Na sljedećoj slici se može vidjeti mobilni robot u konačnoj fazi izrade sa spojenim komponentama. U sljedećem poglavlju je opisano ponašanje tog robota u realnim uvjetima.



**Slika 3.19:** Mobilni robot u konačnoj izvedbi

## 4. TESTIRANJE MOBILNOG ROBOTA

Testiranje mobilnog robota se izvelo nakon realizacije mobilnog robota. Prilikom testiranja mobilnog robota se događalo da je trebalo ugoditi parametar radni ciklus nekoliko puta, zatim prilagođavanje programskog koda s obzirom na podatke dobivenim promatranjem rada mobilnog robota u zadanom okruženju te prilagoditi raspored senzora za dobivanje što preciznije detektiranje prepreka. Najvažnije od svega se pokazalo da je za detekciju prepreke neophodno odrediti vrijeme trajanja pojedinih sekvenci prilikom skretanja mobilnog robota, kako je definirano u algoritmu rada. Npr. za svaku sekvencu je određeno vrijeme trajanja izvođenja, kako bi mobilni robot što „prirodnije“ izvodio izbjegavanje prepreka. Optimizacija mobilnog robota se izvršila pomnim analiziranjem i promatranjem tih podataka dobivenim promatranjem rada mobilnog robota. Nakon nekoliko pokušaja rada, otkriveno je da mobilni robot postiže najbolje rezultate s kašnjenjima definiranim u programskom kodu robota koji je dan u prilogu. Parametar radni ciklus je određen metodom pokušaj – pogreška gdje su se isprobavale nekoliko vrijednosti tog parametra tokom rada robota. Ispostavljeno je da nakon nekoliko pokretanja rada mobilnog robota optimalna vrijednost OCR registra iznosi 255, odnosno nosi njegovu maksimalnu vrijednost, što daje najveću brzinu rada mobilnog robota u smislu brzine kretanja u prostoru. Nakon optimizacije rada, mobilni robot je zadovoljavajuće detektirao i obilazio prepreke i izvršio zadatak, ali treba napomenuti da su primjećene mane rada mobilnog robota i moguća poboljšanja, što je opisano u narednim potpoglavljima. Na sljedećoj slici je prikazan tlocrt prostorijske konfiguracije u kojem se vršilo testiranje rada mobilnog robota, gdje je detaljan opis dan ispod te slike.



**Slika 4.1:** Tlocrt prostorije u kojoj se vršilo testiranje rada mobilnog robota

Na slici 4.1 se vidi tlocrt prostorije u kojoj se vršilo testiranje. Prostorija sadrži regal (označen pravokutnikom na gornjem dijelu slike), zatim stol za blagovaonicu (označen elipsom s desne strane slike) zajedno s pripadajućim stolicama (označenima oko stola za blagovaonicu). S lijeve strane slike se nalazi namještaj zajedno s pripadajućim stolom (označen pravokutnim sa zaobljenim rubovima). Početni položaj mobilnog robota je označen manjim pravokutnikom koji u sebi sadrži slovo „R“. To je položaj u kojemu se uključio mobilni robot i promatrao način rada. Strelice označavaju smjer kretanja mobilnog robota, a brojevima je označen položaj u kojem se uključio neki od senzora mobilnog robota gdje je posljedično robot promijenio smjer kretanja u svrhu zaobilazanja prepreke. Pravokutnik označen slovom „X“ predstavlja završni položaj kretanja mobilnog robota, u kojem se mobilni robot isključio. U položaju 1, mobilni robot je detektirao prepreku s desne strane i promijenio smjer kretanja sve dok nije došao do položaja pod brojem 2. U tom položaju se aktivirao prednji senzor i mobilni robot je tada izveo rutinu kretanja unazad određeni period vremena, do položaja pod brojem 3, gdje je pravilom lijeve ruke otišao u smjeru namještaja do položaja 4. U tom položaju se također aktivirao prednji senzor i

robot se nastavljao kretati unazad do položaja 5, nakon čega se zarotirao ulijevo i krenio naprijed. Došavši do položaja 6, aktivirao se desni senzor gdje je mobilni robot zarotirao ulijevo i krenio ravno prema položaju 7. Došavši do tog položaja, aktivirao se lijevi senzor, zarotirao se udesno i krenuo naprijed prema položaju 8. Rutina se tako ponavljala do položaja 10, kada je mobilni robot krenuo prema položaju 11, u kojem se aktivirao prednji senzor (došavši do stolice), te se vratio nazad do položaja 12, skrenuo ulijevo i nastavio ravno prema položaju 13 u kojem je mobilni robot isključen. S time se završilo testiranje mobilnog robota i zaključeno je da mobilni robot uspješno izbjegava prepreke. Treba napomenuti da se od položaja 4 prema položaju 10 (osim položaja 5) primijetila ograničenja rada mobilnog robota. Također je i osvjetljenje prostorije igralo ulogu. Prepreke su tada bile prilagođene visini senzora (npr. stol označen pravokutnikom sa zaobljenim rubovima je imao relativno nisko podnožje koje je bilo niže od visine senzora, te se stoga detekcija prepreke senzorima bitno otežala). U svrhu potvrde sposobnosti detekcije prepreka, u podnožju tog stola su se dodale dodatne prepreke tako da njihovom visinom omogućuje detekciju prepreka mobilnom robotu (visina podnožja stola povećana za visinu prepreke). U sljedećem potpoglavlju su dodatno opisani nedostaci rada te su dani prijedlozi za poboljšanja rada mobilnog robota.

#### **4.1 NEDOSTATCI MOBILNOG ROBOTA I MOGUĆA POBOLJŠANJA**

Prilikom testiranja mobilnog robota otkriveno je da neki sustavi u mobilnom robotu posjeduju ograničenja. Počevši sa senzorima, otkriveno je da, budući da su senzori na mobilnom robotu infracrveni senzori s dva stanja, sama činjenica da detektiraju isključivo infracrvene elektromagnetske valove predstavlja problem. Problem je osvjetljenje prostora. Ukoliko se mobilni robot nalazi u prostoriji koja je osvjetljena u značajnoj mjeri vanjskim svjetlom, senzori će se aktivirati te će mobilni robot „misliti“ da su svuda oko njega prepreke. Problem također predstavljaju i žarulje, odnosno bilo koji entiteti koji emitiraju takvu svjetlost da u sebi sadrži i infracrveni spektar. Te svjetlosti uzrokuju aktiviranje senzora te daju lažnu informaciju o preprekama u prostoru. Promatranjem rada mobilnog robota je utvrđeno da se mobilni robot ponaša očekivano u slabo osvjetljenoj prostoriji čija je svjetlost takve amplitude da ne utječe na neočekivanu aktivaciju senzora. Druga stvar koja je otkrivena promatranjem je ta da mobilni robot detektira prepreke čije su dimenzije takve da odgovaraju udaljenosti između dva senzora (da su veće ili jednake od te udaljenosti između senzora). Međutim, problem su prepreke čije su dimenzije takve da su manje od udaljenosti između dva infracrvena senzora. Te se prepreke ne

otkrivaju (iako su rijetke u kućanstvu) te ta mana ostavlja prostora optimizaciji detektiranja prepreka. Prijedlog za rješenje ovih dvaju problema je sljedeći: za prvi problem o osvjetljenosti prostorija i žaruljama koji emitiraju infracrvenu svjetlost, problem se može riješiti korištenjem ultrazvučnih senzora koji su neosjetljivi na elektromagnetske valove u kombinaciji s infracrvenim sensorima dok se drugi problem može riješiti ugradnjom većeg broja senzora i povećanjem gustoće detekcije prepreka. Mobilni robot ne posjeduje senzore sa svoje stražnje strane, te se događa (iako rijetko) da udari stražnjim krajem od neku prepreku koju ne može detektirati. Posljedica toga je povećana mogućnost kvara nekog dijela mobilnog robota, no odustalo se od toga zbog toga što se radi o mobilnom robotu koji je sastavljen od izdržljivog materijala, a brzina kretanja mobilnog robota nije dovoljno velika da bi se uzrokovao kvar ukoliko dođe do sudara s preprekom. Prilikom testiranja mobilnog robota je otkriveno da prepreka ne mora nužno biti materijalni objekt, već da bi se na prepreku trebalo gledati na način da to bude entitet koji može nepovoljno utjecati na mobilnog robota ukoliko s njim dođe u interakciju bilo koje vrste (npr. sudar). Jedna od tih vrsta entiteta je mogućnost pojave promjene terena na način da mobilni robot može upasti u neku udubinu ili rupu i ostati nepokretan. Da bi se povećala autonomija robota i riješio ovaj problem, predloženo je korištenje infracrvenih ili ultrazvučnih senzora koji bi se postavili ispod podvozja robota i to tako da „gledaju“ prema podu. Detekcija udubine bi se vršila na način da senzori detektiraju stanje odsutnosti prepreke i izvedu određenu proceduru. Budući da je senzor smješten blizu poda, detektirao bi „prepreku“, no cilj bi bio detektirati udubinu, a udubina nije blizu senzora. Tu bi se dogodilo detektiranje „odsutnosti“ prepreke pri čemu bi tada robot prepoznao da se radi o opasnom dijelu terena i o udubini terena, te bi izveo određenu proceduru definiranu novim poboljšanim algoritmom rada. S povećanjem kompleksnosti robota, mijenja se kompleksnost i algoritma rada te i samog dizajna u cjelokupnom smislu. Budući da je mobilni robot usprkos svojim ograničenjima radio tako da izvršava postavljeni zadatak i budući da je riječ o primjeru na kojemu se treba pokazati kako se dizajnira mikroupravljački sustav mobilnog robota, nije se pristupilo daljnjem poboljšanju mobilnog robota, jer posljedično rastu i troškovi nabavke komponenata.



## 5. ZAKLJUČAK

U ovom diplomskom radu je opisan proces dizajna nekog sustava gdje je taj proces opisan od početne do krajnje točke. Kao završni korak tog procesa, izrađen je mikroupravljački sustav mobilnog robota koji vrši postavljeni zadatak u skladu s očekivanjem, no s primjećanim ograničenjima. Ograničenja su dokumentirana u potpoglavlju 4.1 te su dani prijedlozi za poboljšanje rada mobilnog robota. Prikazan je proces dizajna na praktičnom primjeru gdje je u poglavlju 3. opisano kojim su se redosljedom rješavali problemi postavljeni dizajnom i kako se ostvario mobilni robot. Mobilni robot ima primjenu edukativnog karaktera za demonstraciju dizajna mikroupravljačkog sustava mobilnog robota te također može poslužiti kao osnova za izgradnju kompleksnijih mobilnih robota koji posjeduju kompleksnije algoritme detekcije prepreke u prostoru i vjerno prikazuje način na koji funkcioniraju današnji mobilni roboti. Cilj ovog diplomskog rada je bilo prikazivanje procesa dizajna mikroupravljačkog sustava mobilnog robota, od postavljenog zadatka do konačnog proizvoda.

## LITERATURA

- [1] <https://en.wikipedia.org/wiki/Robot>, 3.3. 2016
- [2] [https://en.wikipedia.org/wiki/Mobile\\_robot](https://en.wikipedia.org/wiki/Mobile_robot), 3.3. 2016
- [3] [http://www.atmel.com/Images/Atmel-8154-8-bit-AVR-ATmega16A\\_Datasheet.pdf](http://www.atmel.com/Images/Atmel-8154-8-bit-AVR-ATmega16A_Datasheet.pdf), 3.3. 2016
- [4] <http://artofcircuits.com/product/infrared-obstacle-avoidance-proximity-sensors-module-fc-51>, 3.3. 2016
- [5] <http://g03.a.alicdn.com/kf/HTB1tHTJIpXXXXasXXXXq6xXFXXU/-IR-Infrared-Obstacle-Avoidance-Sensor-Module-for-Arduino-Smart-Car-Robot-3-wire-Vibration-Sensor.jpg>, 3.3. 2016
- [6] [http://lxxtech.com/upload/images/20131111\\_203754.jpg](http://lxxtech.com/upload/images/20131111_203754.jpg), 10.3. 2016
- [7] [https://hr.wikipedia.org/wiki/Turingov\\_test](https://hr.wikipedia.org/wiki/Turingov_test), 15.4. 2016
- [8] [https://en.wikipedia.org/wiki/Nikola\\_Tesla](https://en.wikipedia.org/wiki/Nikola_Tesla), 15.4. 2016
- [9] [http://i1174.photobucket.com/albums/r617/tonicourt/steampunk/Tesla\\_boat1\\_zps146bd53d.jpg](http://i1174.photobucket.com/albums/r617/tonicourt/steampunk/Tesla_boat1_zps146bd53d.jpg), 15.4. 2016
- [10] [http://mech.vub.ac.be/multibody/final\\_works/ThesisKristofGoris.pdf](http://mech.vub.ac.be/multibody/final_works/ThesisKristofGoris.pdf), 13.5. 2016
- [11] [http://cdn3.bigcommerce.com/s-vt19phz/product\\_images/uploaded\\_images/l298n-motor-driver-from-tronixlabs-australia.png?t=1416543420](http://cdn3.bigcommerce.com/s-vt19phz/product_images/uploaded_images/l298n-motor-driver-from-tronixlabs-australia.png?t=1416543420), 13.5. 2016
- [12] [www.uni-kl.de/elektronik-lager/419183](http://www.uni-kl.de/elektronik-lager/419183), 13.5. 2016
- [13] [https://en.wikipedia.org/wiki/Duty\\_cycle](https://en.wikipedia.org/wiki/Duty_cycle), 13.5. 2016
- [14] <http://i2.wp.com/maxembedded.files.wordpress.com/2011/08/pwmmodulation.jpg>, 13.5. 2016
- [15] <http://i2.wp.com/maxembedded.files.wordpress.com/2011/07/motor-12v-300rpm.png>, 15.5. 2016

- [16] <http://i0.wp.com/maxembedded.files.wordpress.com/2011/07/50perc-duty-cycle-pwm.png>, 15.5. 2016
- [17] <http://www.electronics-tutorials.ws/blog/pulse-width-modulation.html>, 20.6. 2016
- [18] <http://i1.wp.com/maxembedded.files.wordpress.com/2011/12/tccr0.png>, 20.6. 2016
- [19] <http://www.robotshop.com/blog/en/files/iRobi-Q.jpg>, 10.7. 2016
- [20] <http://i2.cdn.turner.com/cnnnext/dam/assets/120928065558-01-drones-dod-horizontal-large-gallery.jpg>, 15.8. 2016
- [21] [http://www.ifr.org/index.php?id=59&df=Duell\\_window\\_glueing\\_with\\_Robot\\_EcoRS\\_Fiat\\_01.jpg](http://www.ifr.org/index.php?id=59&df=Duell_window_glueing_with_Robot_EcoRS_Fiat_01.jpg), 15.8. 2016
- [22] [http://www.icstation.com/images/big/products/5345\\_6191.jpg](http://www.icstation.com/images/big/products/5345_6191.jpg), 15.8. 2016
- [23] <https://hr.wikipedia.org/wiki/Baterija>, 3.9. 2016
- [24] [https://hr.wikipedia.org/wiki/Alkalna\\_baterija](https://hr.wikipedia.org/wiki/Alkalna_baterija), 3.9.2016
- [25] [https://hr.wikipedia.org/wiki/Unutarnji\\_otpor\\_elektri%C4%8Dnog\\_izvora](https://hr.wikipedia.org/wiki/Unutarnji_otpor_elektri%C4%8Dnog_izvora), 3.9. 2016
- [26] <http://drkfs.net/batterytemptestIntR.jpg>, 3.9.2016
- [27] [http://www.hk-phy.org/energy/commercial/act\\_int\\_resist\\_e.html](http://www.hk-phy.org/energy/commercial/act_int_resist_e.html), 3.9. 2016

## SAŽETAK

U ovom diplomskom radu je obrađen proces dizajna mikroupravljačkog sustava mobilnog robota te je za primjer realiziran mobilni robot koji obavlja zadanu radnju. Analizira se pojam „dizajn“ i pojam „robot“. Opisan je proces dizajna nekog sustava od početne do krajnje faze razvoja. Nakon toga se daje teorijska podloga za razumijevanje robota, a zatim se pristupa dizajnu cjelokupnog upravljačkog sustava mobilnog robota kako bi se pokazalo na koji se način dizajnira mikroupravljački sustav mobilnog robota. Tokom procesa dizajna je detaljno opisan proces definiranja algoritma rada s obzirom na postavljeni zadatak, zatim odabira senzora, mikroupravljača, sklopa pokretača istosmjernih motora, pobudnog sklopa i ostalog sklopovlja. Nakon toga je izrađen blokovski dijagram i sheme spajanja komponenata mobilnog robota kako bi se vizualno prikazale najznačanije komponente mobilnog robota kao i njihova interakcija. Mobilni robot je zatim testiran i opisani su rezultati testiranja. Isto tako su dani prijedlozi za daljnja poboljšanja algoritma mobilnog robota kao i primjećeni nedostaci prilikom rada mobilnog robota u zadanom okruženju.

**Ključne riječi:** dizajn, robot, zadatak mobilnog robota, algoritam rada, mikroupravljač, senzori, pobudni sklop, blok shema mikroupravljačkog sustava mobilnog robota, realizacija mobilnog robota

## ABSTRACT

In this master thesis, the process of mobile robot microcontroller design has been described as a practical demonstration of that process and a mobile robot has been designed according to the given task a robot should perform. Firstly, the concept of the term “design“ has been analyzed, and also of the term “robot“. Secondly, the process of a particular system design has been analyzed from the first to the last phase. Thirdly, a theoretical background has been given for the purpose of understanding the concept of robot. Fourthly, a certain approach has been made to show the practical demonstration of mobile robot microcontroller system as a whole in order to show the design process on a real example. During the process of designing, an algorithm has been defined according to the given task a robot should perform when understood, sensors have been chosen, microcontroller has been chosen, motor driver has been chosen and finally both the battery pack and the rest of hardware. Fifthly, the joint schemes of mobile robot hardware block diagram have been made in order to visually represent the most significant components of mobile robot, as well as their interaction. Finally, mobile robot has been tested in given area and both the results of test and noticed limits and so as their recommended improvements have been documented.

**Keywords:** design, robot, mobile robot task, algorithm, microcontroller, sensors, battery pack, mobile robot microcontroller system block scheme, mobile robot realization

## ŽIVOTOPIS

Matej Raguž je rođen 28.9. 1991. u Požegi. U istom gradu završava OŠ Antuna Kanižlića 2006. godine te se iste godine upisuje u srednju Tehničku školu u Požegi. Maturira 2010. godine te se također iste godine upisuje na Elektrotehnički fakultet u Osijeku. U drugoj godini studija na prijedlog profesora sudjeluje na festivalu znanosti u Osijeku na kojem prezentira sklop detektora vlage izrađen za potrebe kolegija Digitalna elektronika na istoj godini. 2013. godine završava preddiplomski studij elektrotehnike na Elektrotehničkom fakultetu u Osijeku. Trenutno radi u tvrtki Gauss Development u Osijeku gdje se bavi izradom mobilnih aplikacija na iOS platformi..

# PRILOZI

Programski kod mobilnog robota u programskom jeziku C:

```
/*
 * Mobile Robot.c
 *
 * Created: 15.5.2016. 19:48:21
 * Author : matej
 */

//Necessary libraries
#include <avr/io.h>
#include <util/delay.h>
#include <stdio.h>

#define F_CPU 4000000UL //Defined frequency of microcontroller
#define setbit(port, bit) (port) |= (1 << (bit)) //Method for setting bit at particular
pin, used for simplifying port operations
#define clearbit(port, bit) (port) &= ~(1 << (bit)) //Method for clearing bit at
particular pin, used for simplifying port operations

volatile unsigned int sensor_read_front; //Volatile integer variable, used for storing
digital state of the front sensor
volatile unsigned int sensor_read_left; //Volatile integer variable, used for storing
digital state of the left sensor
volatile unsigned int sensor_read_right; //Volatile integer variable, used for storing
digital state of the right sensor

//Function used for initialization of pulse - width modulation
void pwm_init()
{
    //Initialization of TCCR0 register for given requirement
    TCCR0 |= (1<<WGM00)|(1<<COM01)|(1<<WGM01)|(1<<CS00);
    //Initializes OC0 pin (pin PB3 for Atmega16A) as output pin
    DDRB |= (1<<PB3);
}
//Function for moving backward
void go_backward()
{
    setbit(PORTA, PA4); //go backward right DC motor
    setbit(PORTA, PA6); //go backward left DC motor
    clearbit(PORTA, PA5); //don't go forward right DC motor
    clearbit(PORTA, PA7); //don't go forward left DC motor
}
//Function for moving forward
void go_forward_initial_moving_procedure()
{
    setbit(PORTA, PA5); //go forward right DC motor
    setbit(PORTA, PA7); //go forward left DC motor
    clearbit(PORTA, PA4); //go backward right DC motor - initially bit not set
    clearbit(PORTA, PA6); //go backward left DC motor - initially bit not set
}
//Function for turning left
void turn_left()
{
    setbit(PORTA, PA6); //go backward left DC motor
    setbit(PORTA, PA5); //go forward right DC motor
    clearbit(PORTA, PA7); //don't go forward left DC motor
    clearbit(PORTA, PA4); //don't go backward right DC motor
}
```

```

}
//Function for turning right
void turn_right()
{
    setbit(PORTA, PA7); //go forward left DC motor
    setbit(PORTA, PA4); //go backward right DC motor
    clearbit(PORTA, PA5); //don't go forward right DC motor
    clearbit(PORTA, PA6); //don't go backward left DC motor
}
//Function used for reading the sensors
void read_the_sensors()
{
    sensor_read_front = PINA & 0x01; //read the front sensor input
    sensor_read_left = PINA & 0x02; // read the left sensor input
    sensor_read_right = PINA & 0x04; //read the right sensor input
}
//Mobile robot initialization of PORT A. This port is connected to sensors and motor
driver
void mobile_robot_init_parameters()
{
    DDRA |= (1<<PA4)|(1<<PA5)|(1<<PA6)|(1<<PA7);
}
//Function used for setting speed of DC motors
void set_speed_of_DC_motors()
{
    uint8_t duty_cycle;
    duty_cycle = 255; //used for setting the speed, 255 is maximum value
    OCR0 = duty_cycle; //tells the robot how fast should it be - set duty_cycle
parameter
}

//Main function
int main(void)
{
    //Initialize mobile robot parameters
    mobile_robot_init_parameters();
    //Initialize pulse - width modulation parameters
    pwm_init();
    //Set speed of DC motors
    set_speed_of_DC_motors();

    //Infinite loop
    while (1)
    {
        //Read the sensor values
        read_the_sensors();
        //if all three have detected obstacle

        if (sensor_read_front == 0 && sensor_read_left == 0 && sensor_read_right ==
0)
        {
            //Go left until all three sensors are equal to 1
            while(sensor_read_front == 0 || sensor_read_left == 0 ||
sensor_read_right == 0)
            {
                go_backward();
                _delay_ms(5000);
                turn_left();
                _delay_ms(1000);
                read_the_sensors();
            }
        }
    }
}

```



```

    }

    //if only front and left have detected obstacle
    if (sensor_read_front == 0 && sensor_read_left == 0 && sensor_read_right ==
4)
    {
        //Go right until all three sensors are equal to 1
        while(sensor_read_front == 0 || sensor_read_left == 0 ||
sensor_read_right == 0)
        {
            go_backward();
            _delay_ms(5000);
            turn_right();
            _delay_ms(1000);
            read_the_sensors();
        }
    }

    //if only left and right have detected obstacle
    if (sensor_read_front == 1 && sensor_read_left == 0 && sensor_read_right ==
0)
    {
        //Go left until all three sensors are equal to 1
        while(sensor_read_front == 0 || sensor_read_left == 0 ||
sensor_read_right == 0)
        {
            go_backward();
            _delay_ms(5000);
            turn_left();
            _delay_ms(1000);
            read_the_sensors();
        }
    }

    //if right has detected obstacle
    if(sensor_read_front == 1 && sensor_read_left == 2 && sensor_read_right ==
0)
    {
        //go left until all three sensors are equal to 1
        while(sensor_read_front == 0 || sensor_read_left == 0 ||
sensor_read_right == 0)
        {
            turn_left();
            _delay_ms(1000);
            read_the_sensors();
        }
    }

    //if front and right have detected obstacle
    if (sensor_read_front == 0 && sensor_read_left == 2 && sensor_read_right ==
0)
    {
        //go left until all three sensors are equal to 1
        while(sensor_read_front == 0 || sensor_read_left == 0 ||
sensor_read_right == 0)
        {
            go_backward();
            _delay_ms(5000);
            turn_left();
            _delay_ms(1000);
            read_the_sensors();
        }
    }

```

```

    }

    //if front only has detected obstacle
    if(sensor_read_front == 0 && sensor_read_left == 2 && sensor_read_right ==
4)
    {
        //go left until all three sensors are equal to 1
        while(sensor_read_front == 0 || sensor_read_left == 0 ||
sensor_read_right == 0)
        {
            go_backward();
            _delay_ms(5000);
            turn_left();
            _delay_ms(1000);
            read_the_sensors();
        }
    }

    //if left only has detected obstacle
    if(sensor_read_front == 1 && sensor_read_left == 0 && sensor_read_right ==
4)
    {
        //Go right until all three sensors are equal to 1
        while(sensor_read_front == 0 || sensor_read_left == 0 ||
sensor_read_right == 0)
        {
            turn_right();
            _delay_ms(1000);
            read_the_sensors();
        }
    }

    if(sensor_read_front == 1 && sensor_read_left == 2 && sensor_read_right ==
4)
    {
        //Go forward
        go_forward_initial_moving_procedure();
    }
}

```