



УНИВЕРЗИТЕТ
У НОВОМ САДУ



ФАКУЛТЕТ
ТЕХНИЧКИХ НАУКА

Трг Доситеја Обрадовића 6, 21000 Нови Сад, Југославија
Деканат: 021 350-413; 021 450-810; Централа: 021 350-122
Рачуноводство: 021 58-220; Студентска служба: 021 350-763
Телефакс: 021 58-133; e-mail: ftndean@uns.ns.ac.yu



Сертификован
систем
квалитета



PROJEKAT

IZ SENZORA I AKTUATORA

NAZIV PROJEKTA:

Pozicioniranje autića korišćenjem senzora MPU6050

TEKST ZADATKA:

Pomoću senzora MPU6050 izvršiti grupu mjerenja prilikom kretanja autića. Takođe proračunati preciznost senzora.

MENTOR PROJEKTA:

Prof. dr Jovan Bajić

PROJEKAT IZRADILI:

Drozgović Đorđe, EE43/2018

Popadić Filip, EE45/2019

U Novom Sadu, 01.10.2023.

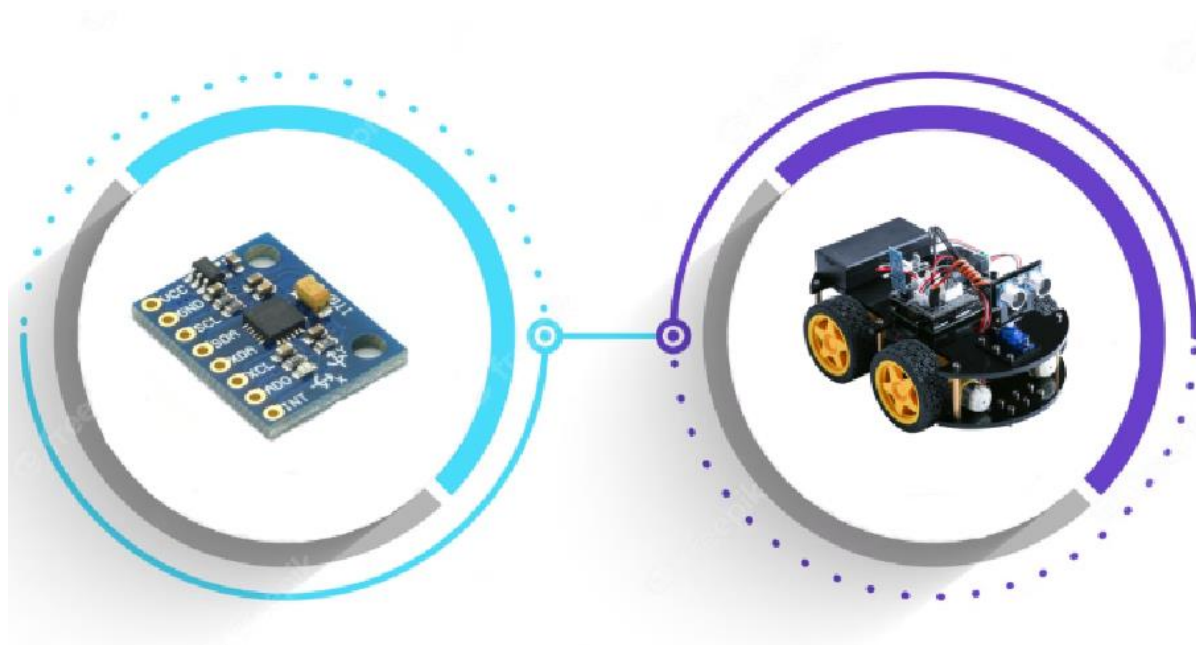
SADRŽAJ

1.UVOD	3
2.ANALIZA PROBLEMA	4
2.1 Ograničenja senzora i složenost fuzije	4
2.2 Tumačenje podataka i kontrolno prevođenje	4
2.3 Preciznost praćenja ivice	5
2.4 Ograničenja u stvarnom svijetu i praktična primjena.....	5
2.5 Integracija i interdisciplinarna priroda	5
3.OSNOVNA IDEJA PROJEKTA	6
3.1 Ključne funkcionalnosti	6
4.PRORAČUNI I SIMULACIONI REZULTATI	7
4.1 Proračun za obrtni moment motora	7
4.2 Proračun ugla.....	8
5.OPIS DETALJA PREDMETA PROJEKTA	9
5.1 MPU6050 žiroskop i akcelerometar	9
5.1.1 Način rada senzora MPU6050.....	10
5.2 DC motor sa četkicama	12
5.3 L298N modul	13
5.4 Ultrasonični senzor HC-SR04	14
5.5 Blutoot modul HC-05	15
5.6 Mikrokontroleri	16
5.7 Kalman filter.....	17
5.8 Slika uređaja u krajnjem stadijumu izrade	18
6.GLAVNA LOGIKA I ALGORITAM RADA	19
6.1 Glavna logika rada.....	19
6.2 Algoritam rada.....	20
7.REZULTATI TESTIRANJA	21
7.1 Mapiranje.....	22
7.2 Rezultati filtriranja signala	23
8.ZAKLJUČAK	24
9. LITERATURA	25

1.UVOD

U okviru projekta iz predmeta "Senzori i Aktuatori," razvijen je autonomni sistem kretanja tenkića koristeći ultrasonične senzore i senzor MPU6050. Glavni cilj projekta je bio istraživanje i implementacija tehnologija senzora i algoritama za precizno mapiranje prostora i upravljanje kretanjem tenkića. MPU6050, kombinuje podatke akcelerometra i žiroskopa kako bi pružio uvid u realnom vremenu u orijentaciju i kretanje objekata u trodimenzionalnom prostoru. Pored navedenog, dodatak dva ultrasonična senzora omogućuje da autić sa povećanom sviješću o svom okruženju, prati ivice i reaguje na prepreke. Osim zamršenosti integracije senzora, naš projekat se bavi sinhronizacijom između različitih sistema. Spoj senzorskih podataka: akcelerometra, žiroskopa i ultrasoničnih senzora, transformiše sirove informacije u kontrolne komande koje se mogu primijeniti.

Pri izradi projekta se susrećemo sa brojnim izazovima. Potrebno je prevazići inherentna ograničenja senzora: od žiroskopskog drifta do nepreciznosti ultrasoničnih senzora, a potom prevesti njihove izlaze u jasna i pouzdana uputstva za motore tenkića. Štaviše, scenario u stvarnom svijetu manevrisanja kroz zamršena okruženja, praćenje ivica i izbjegavanje prepreka zahtijeva zamršenu koreografiju algoritama, logike i donošenja odluka u realnom vremenu. Ovaj dokument opisuje različite faze našeg projekta, od analize problema, osnovnih proračuna za senzor do praktične implementacije testiranja sistema za pozicioniranje autića. Istražujemo teorijske osnove senzora MPU6050, ulazimo u zamršene algoritme spoja senzora sa aktuatorima i predstavljamo naš prilagođeni pristup prevođenju podataka senzora u kontrolne signale koji se mogu primijeniti.



Slika 1 Ideja projekta

2.ANALIZA PROBLEMA

U domenu robotike i autonomnih sistema, precizno pozicioniranje, mapiranje i kontrola pokreta su ključni izazovi koji nastavljaju da podstiču inovacije. Bilo da se radi o navigaciji lavirintom, praćenju određene putanje ili izbjegavanju prepreka, mogućnost precizne kontrole položaja mobilnog uređaja je od suštinskog značaja. Cilj nam je da stvorimo robustan sistem pozicioniranja koji prikazuje potencijal MPU6050 senzora u poboljšanju aplikacija za kontrolu pokreta.

Ideja projekta jeste da se tenkić postavi unutar lavirinta, kvadratnog oblika, poznatih dimenzija i uradi mapiranje samog prostora u kom se kreće. Na kraju treba unijeti dobijene X i Y kordinate u Excel tabelu i pretvoriti ih u tjemena kvadrata (tenkić će „izbaciti“ četiri vrijednosti, odnosno po jednu za svako tjeme) i tako dobijeni kvadrat uporediti sa poznatim.

2.1 Ograničenja senzora i složenost fuzije

Senzor MPU6050 kombinuje mjerenja akcelerometra i žiroskopa kako bi pružio sveobuhvatan pogled na kretanje objekta. Međutim, on nije bez ograničenja. Akcelerometri su osjetljiviji na vibracije i buku, što dovodi do grešaka u mjerenju. Žiroskopi, s druge strane pate od pomjeranja tokom vremena. Izazov leži u ublažavanju ovih inherentnih ograničenja kako bi se stvorio pouzdan i brz sistem pozicioniranja. Tehnike fuzije senzora, kao što su komplementarni filter ili Kalmanov filter, igraju ključnu ulogu u harmonizaciji podataka sa različitih senzora, ali njihova primjena zahtijeva nijansirano razumijevanje matematike i obrade signala.

2.2 Tumačenje podataka i kontrolno prevođenje

Pretvaranje neobrađenih podataka senzora u značajne kontrolne signale za tenkić predstavlja višestruki izazov. Projekat zahtijeva duboko uranjanje u interpretaciju podataka akcelerometra i žiroskopa kako bi se izvukle informacije o orijentaciji automobila, ugaonoj brzini i ubrzanju. Štaviše, prevođenje ovih podataka u operativne kontrolne komande zahtijeva pažljivo razmatranje mehanizama kontrole motora i procedura kalibracije. Podstizanje glatke i precizne kotrnole zavisi od efektivne interakcije između tumačenja podataka i kontrolnog prevođenja.

2.3 Preciznost praćenja ivice

Jedan aspekt našeg projekta uključuje omogućavanje tenkića da prati ivice, zadatak koji može izgledati jednostavan, ali predstavlja inherentnu složenost. Ultrasonični senzor zadužen za detekciju ivica mora da pokaže delikatan balans između osjetljivosti i smanjenja šuma. Kalibracija ovog senzora da dosledno prepozna lokaciju ivice, posebno reflektujućim površinama, dovodi u pitanje našu sposobnost da izvučemo značajne podatke iz okruženja i prevedemo ih u efikasne kontrolne komande.

2.4 Ograničenja u stvarnom svijetu i praktična primjena

Dok teorijski koncepti pružaju osnovu, implementacija u stvarnom svijetu često uvodi neočekivane izazove. Tenkić mora da se kreće kroz fizičke prepreke, različite terene i potencijalne smetnje iz spoljašnjih izvora. Praktična primjena sistema za pozicioniranje zahtijeva robusnost, prilagodljivost i mogućnost prilagođavanja u realnom vremenu. Balansiranje teorijske osnove sa praktičnim zamršenostima je od vitalnog značaja kako bi se osiguralo da se rezultati projekta efikasno pretvore u opipljive rezultate.

2.5 Integracija i interdisciplinarna priroda

Pozicioniranje tenkića pomoću senzora MPU6050 je inherentno interdisciplinarno i zahtijeva stručnost u elektronici, mehanici, razvoju softvera i teoriji upravljanja. Projekat naglašava potrebu za efikasnom saradnjom između ovih domena kako bi se razvilo dobro rješenje. Integracija senzorskih podataka sa kontrolom motora, dizajniranje algoritma upravljanja koji reaguje i optimizacija performansi sistema zahtijevaju kohezivan pristup koji premošćuje praznine između različitih polja.

3.OSNOVNA IDEJA PROJEKTA

Ovaj projekat ima za cilj razvoj automobila sa sposobnošću samonavigacije korišćenjem senzora i aktuatora i pravilnog mapiranja u toku samog kretanja. Automobil koristi različite senzore za prikupljanje podataka o okolini, uključujući senzor MPU6050 za mjerenje ubrzanja i brzine okretanja, kao i ultrazvučne senzore za mjerenje udaljenosti od prepreka. Na osnovu prikupljenih podataka, automobil donosi odluke o svom kretanju, izbjegavajući prepreke i održavajući pravac i pritom konstantno mapira pređeni put.

3.1 Ključne funkcionalnosti

U nastavku su opisane neke od ključnih funkcionalnosti projekta:

Mjerenje Udaljenosti: Projekat koristi ultrazvučne senzore (HC-SR04) za tačno mjerenje udaljenosti od prepreka. Ovi senzori omogućavaju tenkiću da detektuje prepreke u svom okruženju. Iskorišćena su dva takva senzora, jedan za detekciju prepreka ispred sebe, dok se drugi nalazi sa strane i koristi se radi detekcije odnosno praćenja ivice.

Senzor MPU6050: Senzor MPU6050 se koristi za mjerenje ubrzanja i brzine okretanja tenkića. Ovi podaci mu pomažu da odluči kako da se kreće i održava svoju putanju. Konkretno u našem slučaju koristimo akcelerometar i žiroskop sa senzora i to tako što se žiroskop primarno koristi za skretanje dok se akcelerometar koristi radi pravilnog računanja pređenog puta.

Bluetooth Upravljanje: Automobil može da se upravlja daljinski putem, putem Bluetooth komunikacije. Korisnik može slati komande za upravljanje automobilom sa mobilnog uređaja ili u našem slučaju laptopa.

Praćenje Putanje: Projekat prati putanju kretanja automobila i čuva informacije o pređenom putu i orijentaciji. Ovo je korisno za praćenje kretanja automobila i analizu performansi.

Kontrola Motora: Automobil koristi fizičke motore za kretanje. Projektom se upravlja putem digitalnih signala za kontrolu pravca i analognih PWM signala za kontrolu brzine. Koristimo dva aktuatora radi realizacije kretanja. Iskorišćen je tenkić sa gusjenicama radi lakšeg okretanja u mjestu.

Efikasna komunikacija između ovih komponenti omogućava automobilu da donosi odluke o kretanju na osnovu informacija koje prikuplja od senzora i komandi koje prima preko Bluetooth-a. Ova koordinacija je ključna za funkcionalnost projekta i omogućava tenkiću da se autonomno kreće i izbjegava prepreke.

4.PRORAČUNI I SIMULACIONI REZULTATI

4.1 Proračun za obrtni moment motora

Specifikacije motora potrebnog za kretanje možemo da proračunamo tako što ćemo uzeti najzahtjevniji položaj u kom se motor može naći, po pitanju opterećenja koje mu se nalazi na vratilu. Razmotrićemo primjer gdje se robot kreće uz strimu padinu koja je pod nagibom od 30 stepeni. Masa našeg robota je 0.812 kilograma. Poluprečnik točkova gusjenice je 0.02 m. Da bi se kretao, robot mora da savlada gravitacionu silu i silu trenja. Na osnovu njih možemo da odredimo potrebni obrtni moment za motore. Proračun za gravitacionu silu je dat u jednačini 1.

$$F_{gr} = m \cdot g \cdot \sin(\alpha) \quad (1)$$

Gdje je F_{gr} gravitaciona sila, m ukupna masa robota, g gravitaciono ubrzanje koje je približno 9.81 m/s^2 , a α nagib uspona u stepenima. Dobijamo da je

$$F_{gr} = 0.812 \text{ kg} \cdot 9.81 \text{ m/s}^2 \cdot \sin(30^\circ) = 3.98 \text{ N}$$

Koeficijent trenja između gusjenica robota i površne uzbrdice zavisi od materijala gusjenica i karakteristike površine. Naš robot se kreće po betonskoj ravni, gusjenice su od plastike pa za koeficijent trenja, između ta dva materijala, uzimamo da iznosi 0.3. Silu trenja pronalazimo iz jednačine 2.

$$F_{tr} = \mu \cdot m \cdot g \cdot \cos(\alpha) \quad (2)$$

Ovde F_{tr} predstavlja silu trenja, sa μ je označen dinamički koeficijent trenja. Proračun se sada svodi na sledeće:

$$F_{tr} = 0.3 \cdot 0.812 \text{ kg} \cdot 9.81 \text{ m/s}^2 \cdot \cos(30^\circ) = 2.07 \text{ N}$$

Sada možemo pronaći potrebni obrtni moment koji je jednak proizvodu ukupne sile koju robot mora da savlada i poluprečnika točkova.

$$T = (F_{tr} + F_{gr}) \cdot r = (3.98 \text{ N} + 2.07 \text{ N}) \cdot 0.02 \text{ m} = 0.121 \text{ Nm}$$

U ovom primjeru uzmimo da nam robot treba preći dva metra u vremenu jednakom četiri sekunde. To nam daje brzinu od 0.5 m/s . Na osnovu ovih podataka možemo odabrati koji bi bio najbolji izbor motora za naš projekat. Motor koji najviše zadovoljava naše uslove bio bi Maxon katalogskog broja 283872. Ovaj motor ima nominalni obrtni moment od 0.203 Nm . Masa motora je 720 grama, radi pri naponu od 48 V i ima strujnu potrošnju od 2.19 A . Velika mana je njegova cijena koja iznosi 563 evra.



Slika 2 DC Motor Maxon katalogskog broja 283872

Za upravljanje motorima iskorišćen je L298N modul za motore koji daje maksimalnu struju od 2A. Motor koji smo mi iskoristili za projekat je jednostavni DC motor sa četkicama koji ima nominalni obrtni momenat od 0.3 Nm. Vidimo da je naš motor u stanju da pokrene našeg robota u ekstremnijim uslovima, ali je naravno bolje izabrati, u slučaju da je to moguće, tehnički optimalniju opciju koji smo proračunali. Bitno je naglasiti da je za naš proračun izabran motor koji će sigurno zadovoljiti sve potrebne karakteristike i obezbediti pouzdanost sistema.

4.2 Proračun ugla

U kodu se koristi yaw vrednost, odnosno ugao oko vertikalne (Z-ose) koji se čita pomoću senzora MPU6050 i koristi za kontrolu kretanja. Konkretno koristimo `Wire.read()`; funkciju, koja se koristi za čitanje bajtova (osam bitova) sa žičane veze (I2C komunikacija). MPU6050 senzor komunicira preko I2C protokola i ovi bajtovi predstavljaju podatke o rotaciji senzora u odnosu na Z-osu.

$$\text{GyroZ} = (\text{Wire.read()} \ll 8 | \text{Wire.read()}) / 131.0;$$

„ $\ll 8$ “ se koristi za pomeranje prvog bajta (starijeg bajta) u levo za 8 pozicija. Ovo je potrebno jer senzor šalje podatke u dva bajta, stariji i mlađi bajt, gde stariji bajt sadrži starije bitove podataka, a mlađi bajt sadrži mlađe bitove. Pomeranjem starijeg bajta za 8 mesta u levo, dobijamo njegovu odgovarajuću vrednost. Potom se koristi „ $|$ “ operator za spajanje ova dva bajta kako bismo dobili potpun broj. Na kraju, podela rezultata se koristi za skaliranje podataka, tj. za pretvaranje sirovih podataka iz senzora u jedinice stepeni u sekuni ($^{\circ}/s$).

Potom filtriranu vrednost (upotrebom Kalman filtra) smeštamo u novu promenljivu `estimated_value` i kao takvu koristimo za proračun „yaw“, odnosno željenog ugla:

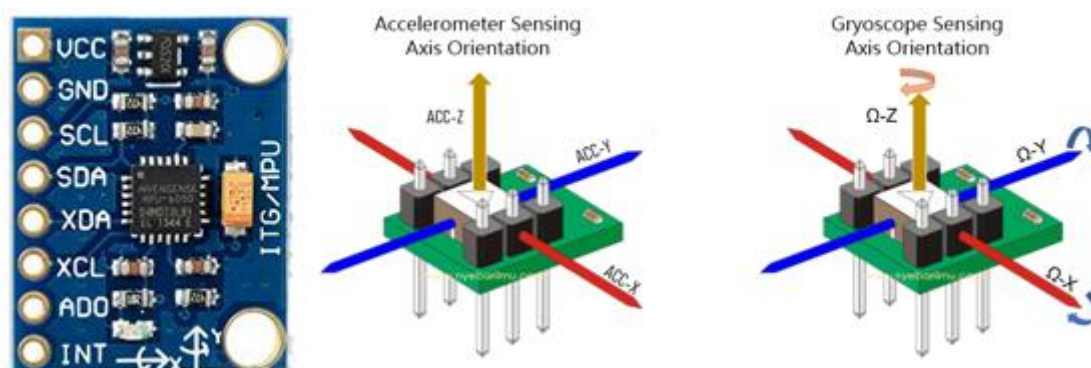
$$\text{yaw} = \text{yaw} + \text{estimated_value} * \text{elapsedTime}$$

5.OPIS DETALJA PREDMETA PROJEKTA

5.1 MPU6050 žiroskop i akcelerometar

MPU6050 je veoma popularan žiroskopski čip sa akcelerometrom koji ima šest osa sa 16-bitnom rezolucijom mjerenja. Mnogi komercijalni proizvodi su opremljeni sa MPU6050. Kombinacija žiroskopa i akcelerometra se obično naziva inercijalna mjerna jedinica ili IMU(Inertial-Measurement-Unit).

IMU senzori se koriste u širokom spektru aplikacija kao što su mobilni telefoni, tableti, sateliti, svemirske letjelice, bespilotne letjelice, robotika i još mnogo toga. Koriste se za praćenje kretanja, detekciju orijentacije i položaja...



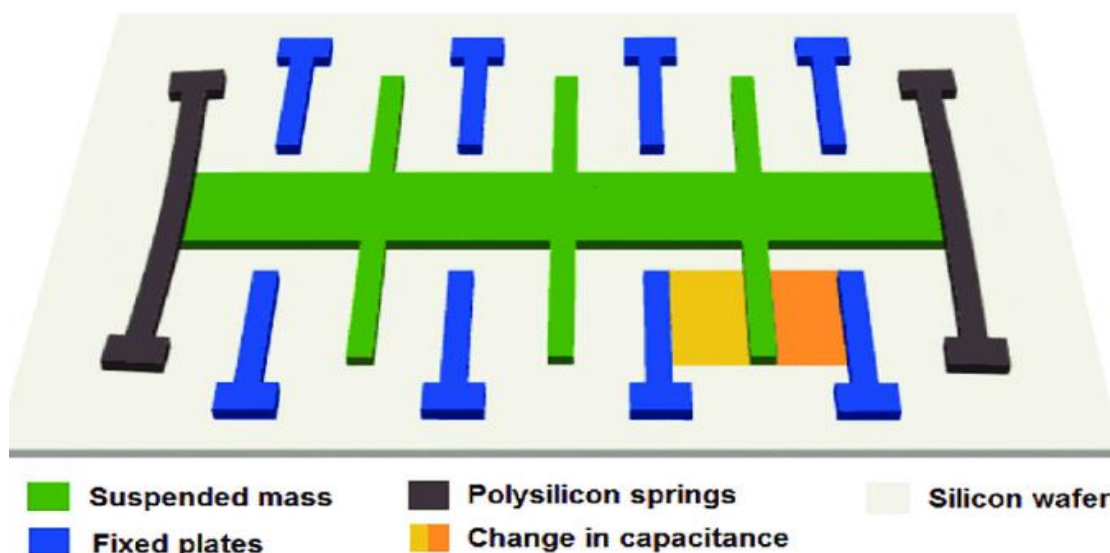
Slika 3 MPU6050 modul

Za komunikaciju sa senzorom MPU6050, obično se koristi I2C(Inter-Integrated Circuit) protokol. I2C je sinhrona komunikacija koja se koristi za mala rastojanja, najčešće za povezivanje integrisanih kola na štampanoj ploči. Za ovaj protokol su potrebne dvije fizičke linije open drejn tipa. Ove linije se nazivaju SCL(Serial Clock) i SDA(Serial Data).

5.1.1 Način rada senzora MPU6050

MPU6050 se zasniva na MEMS(Micro-Elektro-Mechanical-System) sistemu koji se sastoji od 3-osnog akcelerometra i 3-osnog žiroskopa unutar njega. Ovo nam pomaže da mjerimo ubrzanje, brzinu, orijentaciju, pomjeranje i mnoge druge parametre sistema ili objekta. Ovaj modul takođe ima DMP(Digital-Motion-Processor) digitalni procesor pokreta unutar sebe koji je dovoljno moćan da sam izvrši složene proračune i tako rastereti mikrokontroler.

MEMS akcelerometri se koriste svuda gdje postoji potreba za mjerenjem linearnog kretanja, bilo kretanja, udara ili vibracija, ali bez fiksne reference. Oni mjere linearno ubrzanje svega za šta su vezani. Svi akcelerometri rade na principu mase na opruzi, kada se stvar za koji su pričvršćeni ubrzava, tada masa želi da ostane nepokretna zbog svoje inercije i stoga se opruga rasteže ili sabija stvarajući silu koja se detektuje i odgovara na primjenjeno ubrzanje.

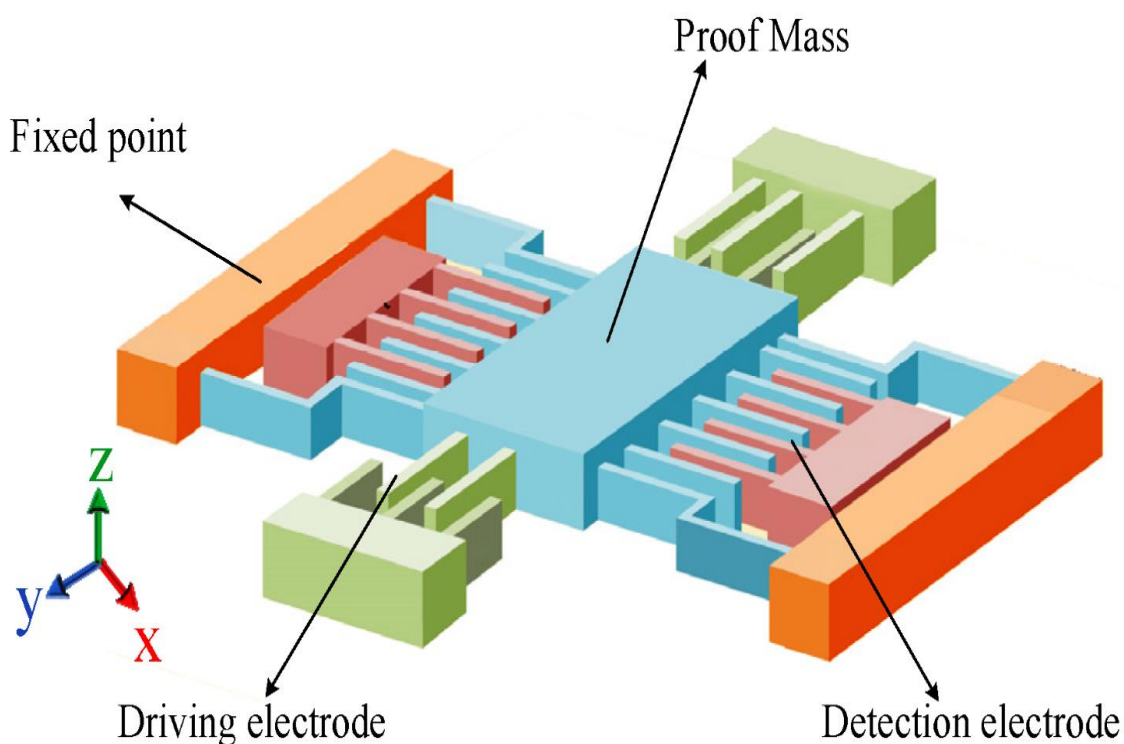


Slika 4 Interna struktura MPU6050 akcelerometra

Precizna detekcija linearnog ubrzanja u dvije ortogonalne ose postiže se parom silicijumskih MEMS detektora formiranih od opružnih „otpornih masa“. Sva masa obezbijeduje pokretnu ploču promjenljive kapacitivnosti formiranu nizom isprepletanih struktura u obliku prstiju. Kada je senzor podvrgnut linearnom ubrzanju duž svoje osjetljive ose, masa za provjeru teži da se odupre kretanju zbog svoje inercije, pa se masa i njeni prsti pomjeraju u odnosu na prste fiksne elektrode. Ovo pomjeranje indukuje diferencijalni kapacitet proporcionalan primjenjenom ubrzanju. Ti podaci se vode na AD konvertor visoke rezolucije, pretvaraju se u čitljivu vrijednost i dolaze do glavnog I2C uređaja.

Rad MEMS žiroskopa je zasnovan na Koriolisovom efektu. Koriolisov efekat kaže da kada se masa kreće u određenom pravcu sa nekom brzinom i na nju se primjeni spoljno ugaono kretanje, stvara se sila koja izaziva okomito pomjeranje mase. Sila koje se generiše naziva se Koriolisova sila. Brzina pomjeranja će biti direktno povezana sa primjenjenim ugaonim kretanjem.

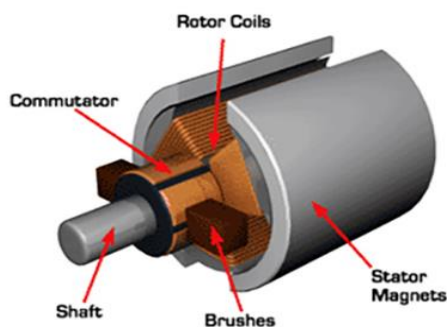
Ovakav žiroskop sadrži set od četiri otporne mase i održava se u neprekidnom oscilirajućem pokretu. Kada se primjeni ugaono kretanje, Koriolisov efekat izaziva promjenu kapacitivnosti između masa u zavisnosti od ose ugaonog kretanja. Efekat uzrokuje promjenu kapacitivnosti koja se očitava.



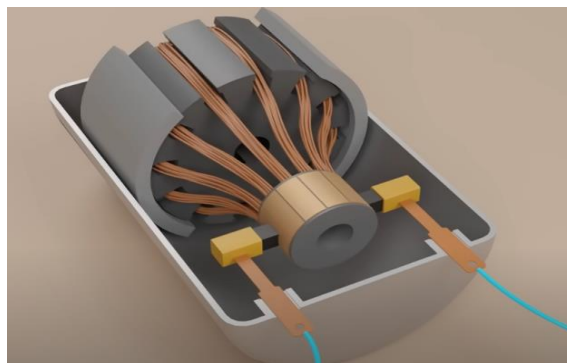
Slika 5 Interna struktura MPU6050 žiroskopa

5.2 DC motor sa četkicama

Jednostavan jednosmjerni ili DC(direct current) motor se sastoji od šest osnovnih dijelova: statora, rotora, komutatora, četkica, osovine i priključaka za jednosmjerno napajanje.

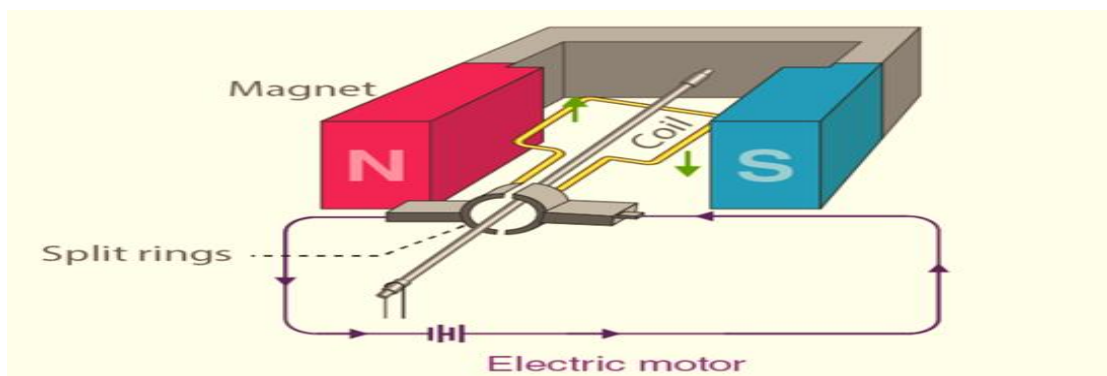


Slika 6 DC motor sa četkicama



Slika 7 DC motor uz priključke za napajanje

Spoljašnja strana DC motora je stator: trajni magnet koji se ne kreće. Stator je predstavljen sa dva zakrivljena trajna magneta. Unutrašnji dio je rotor koji se obrće. Rotor zajedno sa komutatorom čini armaturu pričvršćenu za osovinu. U osnovi, rotor predstavlja jednu zatvorenu petlju odnosno zavojnicu povezanu sa armaturom. Najčešće radi postizanja boljeg efekta rotor se sastoji od više petlji koje su gusto namotane žicom. Ovakav motor je pogodan zbog niske cijene, velike pouzdanosti i jednostavne kontrole brzine obrtanja.

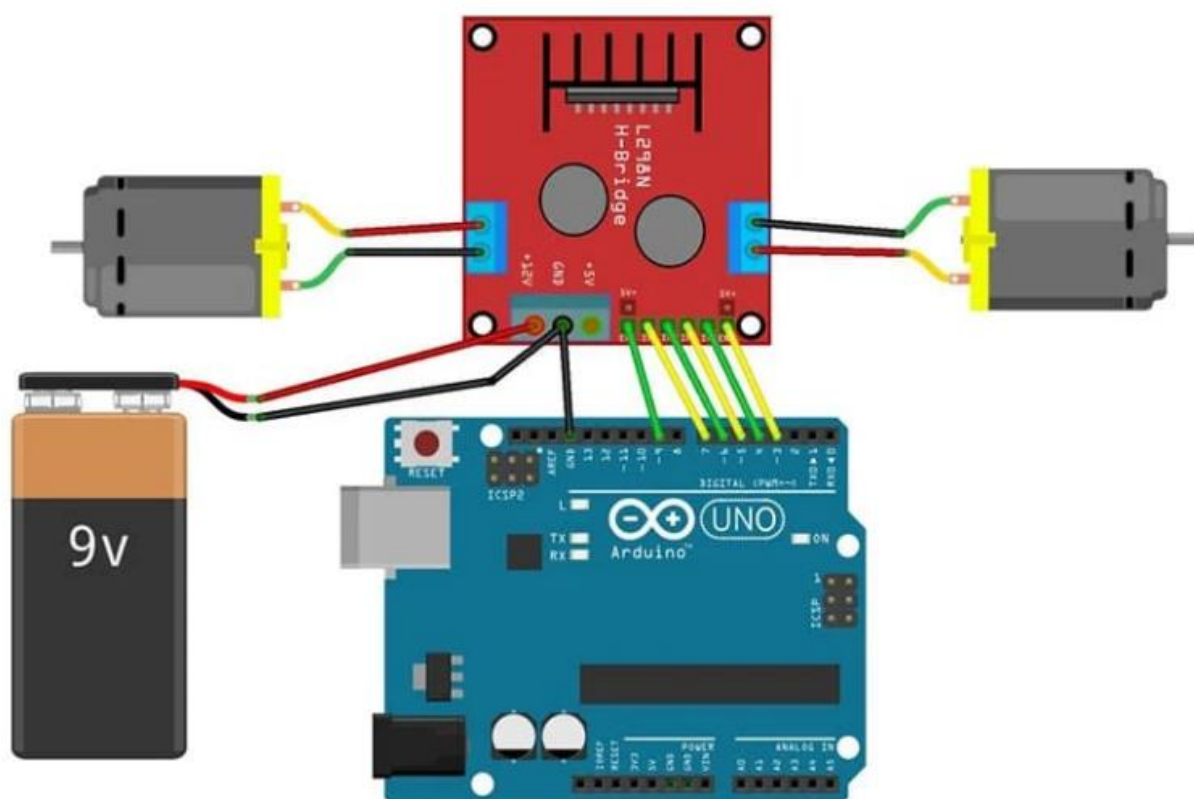


Slika 8 Princip rada DC motora

Osnovni princip rada DC motora zasniva se na korišćenju elektromagneta(u ovom slučaju zavojnica) koji kada se priključi napajanje, radi kao konvencionalni trajni magnet i proizvodi magnetsko polje koje izaziva okretanje. Bitna komponenta je komutator povezan na krajeve zavojnice, koji predstavlja metalni prsten podeljen na dvije odvojene polovine. Njegov zadatak je da preokrene električnu struju u kalemu svaki put kada se kalem okrene za pola okreta da ne bi došlo do narušavanja smjera obrtanja. Jedan kraj zavojnice je pričvršćen za svaku polovinu komutatora. Električna struja iz baterije povezuje se sa električnim terminalima motora. Oni dovode električnu energiju u komutator kroz par labavih konektora zvanih četkice, napravljenih od komada grafita ili tankih dužina elastičnog metala. Sa komutatorom na mjestu, kada struja teče kroz kolo, zavojnica će se stalno rotirati u istom pravcu. Smjer okretanja mijenjamo obrtanjem polariteta baterije.

5.3 L298N modul

Kao drajver za motore, potreban da bi njima upravljali, iskorišćen je drajver L298N, sa duplim H-mostom, koji je u sastavu istoimenog modula, pogodnim za kontrolu motora snage do 2A. Ovaj modul koristi PWM za kontrolu brzine i H-most za kontrolu smjera rotacije. Pogodan je jer može istovremeno kontrolisati dva DC motora, svestran je i lak za upotrebu i jednostavno se povezuje na bilo koji mikrokontroler ili Arduino ploču.



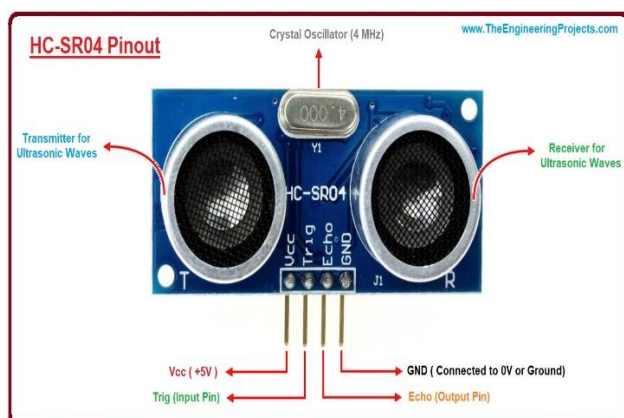
Slika 9 L298N modul povezan sa kontrolerom i dva DC motora na napajanje

5.4 Ultrasonični senzor HC-SR04

Ultrasonični senzor HC-SR04 je popularan i jeftin senzor koji se često koristi za mjerenje udaljenosti u mnogim projektima u elektronici. Koristi se za ultrazvučne talase za mjerenje udaljenosti. Senzor šalje kratke ultrazvučne impulse i mjeri vrijeme koje je potrebno da se ti impulsi odraze od objekta i vrate nazad do senzora. Senzor može mjeriti udaljenosti u opsegu od 2 cm do 400 cm (0.02 m - 4 m).

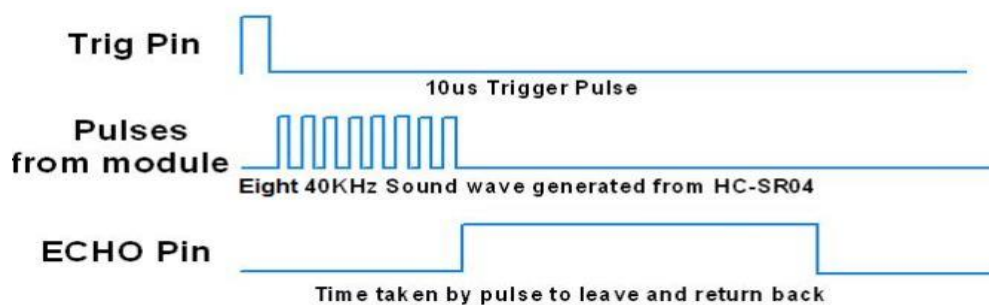
Kako bi se izmjerila udaljenost, prvo se na *Trigger* pinu šalje kratki visoki impuls (puls) koji traje najmanje 10 mikrosekundi. Ovaj impuls pokreće senzor da šalje ultrazvučne talase. Kada ultrazvučni talasi pogode objekat i odbiju se nazad, senzor na *Echo* pinu generiše visoki nivo signala. Vrijeme koje je proteklo između slanja impulsa i njegovog prijema (mjereno u mikrosekundama) se koristi za izračunavanje udaljenosti. Udaljenost se može izračunati koristeći formulu: Udaljenost (cm) = Vreme (μs) / 58, gde je 58 konstanta koja se koristi za konverziju vremena u udaljenost u centimetrima.

HC-SR04 senzor se često koristi u projektima autonomnih vozila (kao što je naš tenkić), robotima, sistemima za parkiranje vozila, mjerenju nivoa tečnosti, i drugim projektima koji zahtijevaju mjerenje udaljenosti.



Slika 10 Fizički izgled Ultrasoničnog senzora HC SR04

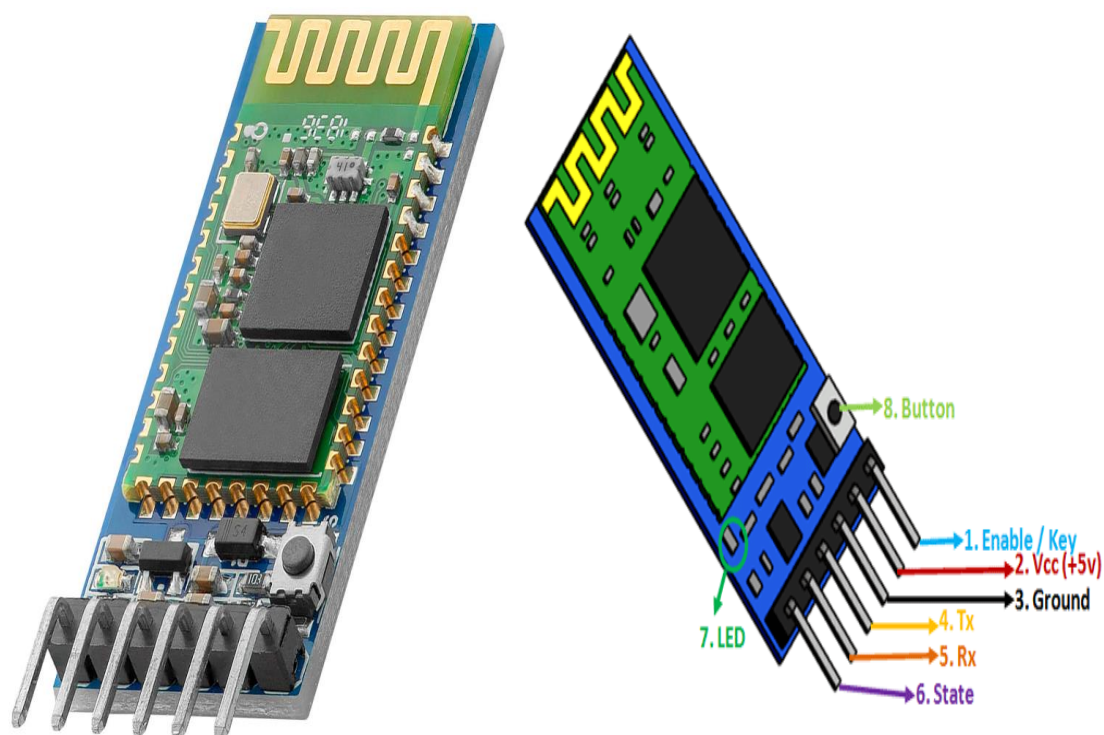
Ultrasonic HC-SR04 module Timing Diagram



Slika 11 Vremenski Dijagram Ultrasoničnog Senzora

5.5 Blutut modul HC-05

U cilju lakšeg očitavanja vrijednosti mjerenja sa MPU6050 žiroskopa i akcelermotra upotrebljen je HC-05 blutut modul. Pomoću ovog modula lako pratimo vrijednosti sa senzora tokom kretanja autića. Koristi bežičnu serijsku RS232 komunikaciju, posjeduje integrisani master i slave model, integrisanu antenu na čipu, 8 MB fleš memorije, a maksimalna brzina prenosa podataka kreće se u opsegu 2-3 Mbps. Radi u opsegu frekvencija 2.4 GHz – 2.48 GHz. Za naš projekat omogućeno je pokretanje sistema slanjem poruke preko module.



Slika 12 Blutut modul HC-05

5.6 Mikrokontroleri

Odabrani kontroler za naš projekat je Arduino UNO, mikrokontrolerska ploča zasnovana na ATmega328P kontroleru. Ima 14 digitalnih ulazno/izlaznih pinova(od kojih se 6 može koristiti kao PWM izlazi nama bitni za generisanje takvog signala), 6 analognih ulaza, keramički oscilator od 16MHz, priključke za serijsku konekciju i napajanje kao i dugme za resetovanje. Njegova velika prednost je pristupačna cijena i jednostavna upotreba zbog već postojećih gotovih biblioteka za senzore i motore koji smo koristili.



Slika 13 Arduino UNO ekosistem (ATmega328P)

5.7 Kalman filter

Kalman filter je matematički algoritam koji se koristi za procijenu i filtriranje podataka u realnom vremenu. Glavna uloga Kalman filtra je da kombinuje informacije iz različitih izvora kako bi se bolje odredila nepotpuna i šumom zagađena mjerenja, čime se dobija tačnija procijena trenutnog stanja nekog sistema ili objekta. U nastavku su neki od osnovnih elemenata uloge Kalman filtra:

Predviđanje stanja: Filter koristi model sistema da predvidi sledeće stanje na osnovu prethodnog stanja i kontrolnih ulaza (ako postoje). Ova predikcija uzima u obzir dinamiku sistema i šum koji se očekuje.

Mjerenje: Filter prima mjerenja iz stvarnog svijeta. Ta mjerenja su često neprecizna i sadrže šum, pa je potrebno procijeniti tačno stanje sistema na osnovu ovih mjerenja.

Korekcija: Kalman filter kombinuje informacije iz predviđanja i mjerenja kako bi se dobio najbolji mogući procenat trenutnog stanja sistema. Ovaj korak je od suštinske važnosti za smanjenje uticaja šuma i poboljšanje preciznosti procjene.

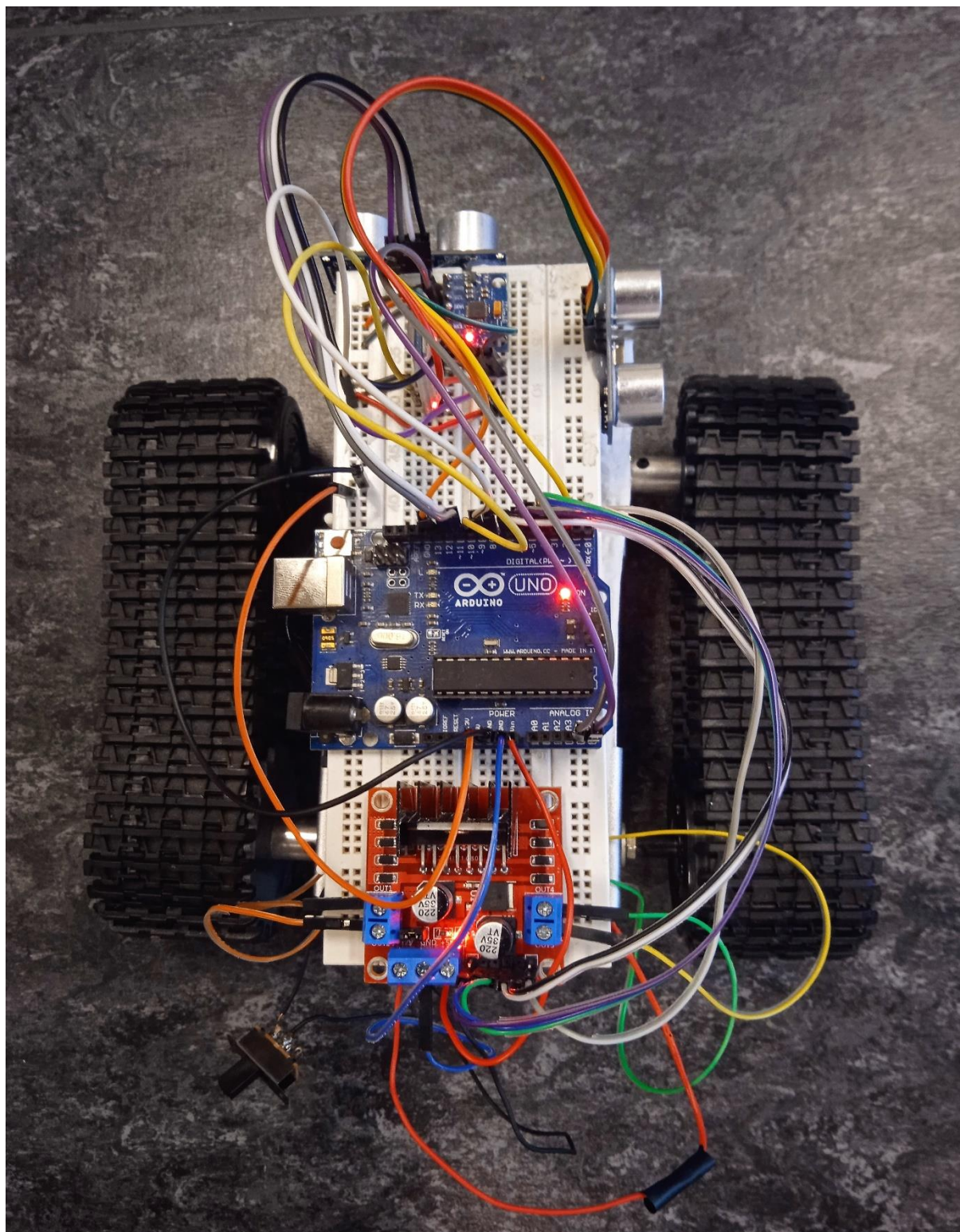
Iteracija: Proces se iterira kroz vrijeme, kontinuirano ažurirajući procjene kako bi se održala tačnost u promenljivim uslovima.

```
SimpleKalmanFilter simpleKalmanFilter(0.5, 0.5, 0.03);  
estimated_value1 = simpleKalmanFilter.updateEstimate(GyroZ);
```

Slika 14 Kreiranje objekta filtra sa specifikiranim parametrima

U ovom projektu iskoristili smo <SimpleKalmanFilter.h> biblioteku. Nakon njenog pozivanja potrebno je kreirati objekat i specificirati parametre samoga filtra. Prvi parametar u zagradi je početna vrijednost procjene, sledeći je početna vrijednost varijanse mjerenja i treći varijansa procesa (modela). U toku rada potrebno je ažurirati filter sa svakim novim mjerenjem putem metode update(). U našem slučaju, empirijskim postupkom smo došli do vrijednosti parametara koji daju najpreciznije podatke prilikom kretanja tenkića.

5.8 Slika uređaja u krajnjem stadijumu izrade



Slika 15 Izgled uređaja

6. GLAVNA LOGIKA I ALGORITAM RADA

6.1 Glavna logika rada

Princip rada "right-edge following" ili "praćenje desne ivice" je strategija kretanja autonomnih robota ili vozila tako da održavaju konstantan kontakt ili udaljenost sa desnom ivicom nekog objekta ili površine. Ovaj princip se često koristi u navigaciji i robotici kako bi se izbegli sudari i kako bi se pratila unapred definisana putanja koristeći desnu stranu kao referencu.

```
if(zastava4 < 5){
  if (distance2 > 5 && distance2 < 30 && distance1 > 31 && zastava3 != 1 && zastava4 != 4) {
    idi_pravo1();
  }
  else if (distance1 > 10 && distance1 < 30 && distance2 > 5) {
    zastava3 = 1;
  }else if (zastava3 == 1) {
    if (yaw < targetGyro) {
      levo();
    }
  }else{
    stani();
    zastava4++;
    zastava3 = 0;
  }
}
else {
  pravo();
}
}
```

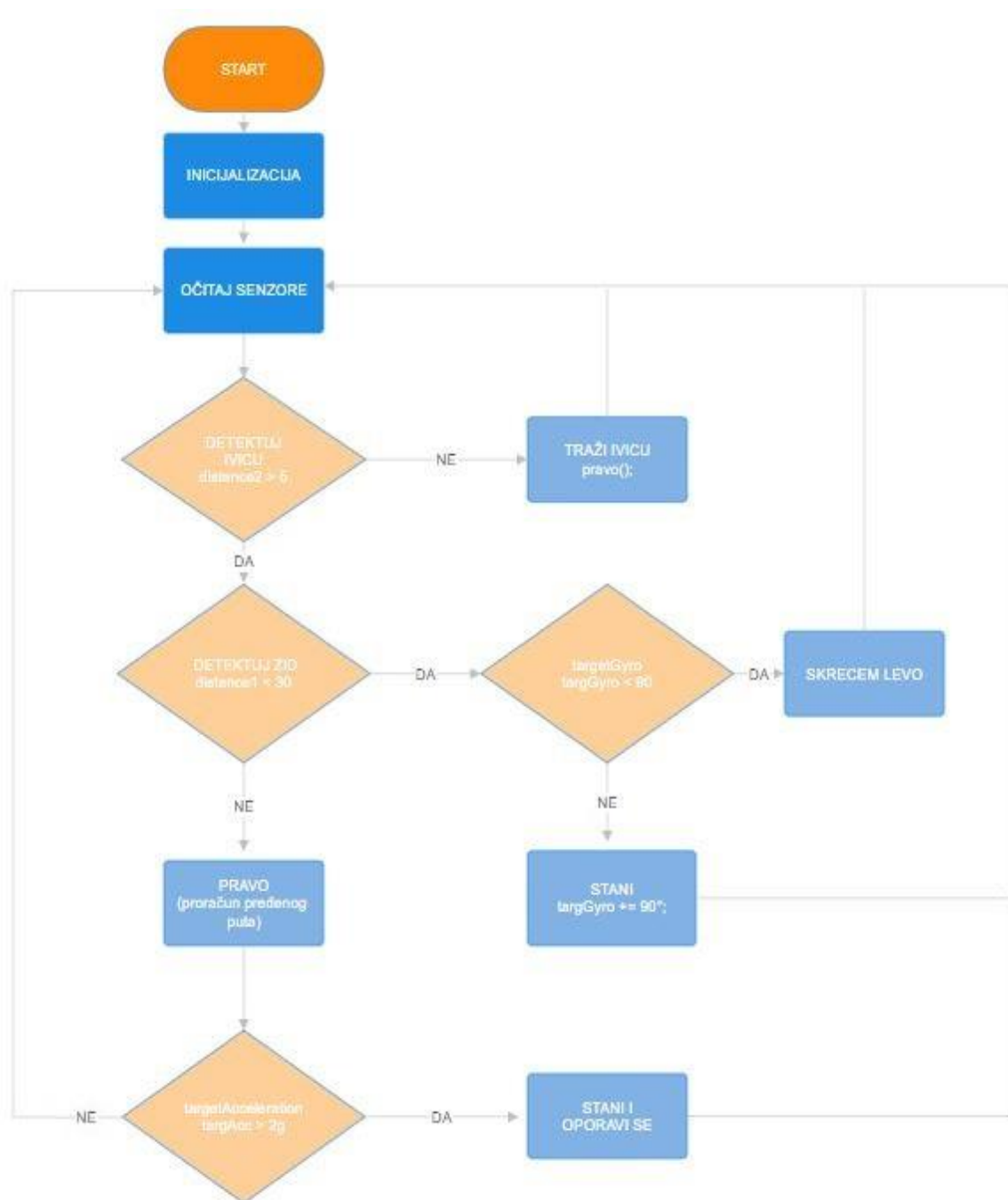
Slika 16 Algoritam kretanja robota u kodu

Senzor MPU6050 je sposoban da meri ubrzanje i brzinu rotacije u tri osnovne dimenzije, što ga čini izuzetno korisnim alatom za sprečavanje sudara i očuvanje bezbednosti tenkića. Za zaštitu tenkića od nepredviđenih situacija prilikom kretanja, konstantno se proverava da li je došlo do nagle promene ubrzanja. Softverska implementacija je jako jednostavna i predstavlja jednostavan uslov koji se konstantno proverava.

```
void idi_pravo1() {
  if (ubrzanje < targetAcceleration) {
    // dio koda unutar funkcije
  }
  else {
    stani();
    delay(600);
  }
}
```

Slika 17 Zaštita u slučaju velikog ubrzanja

6.2 Algoritam rada



Slika 18 Ilustracija algoritma rada

7.REZULTATI TESTIRANJA

Kako bi dobili adekvatne rezultate testiranja potrebno je izvršiti odgovarajuće proračune u samom kodu. Prilikom pokretanja mikrokontrolera potrebno je pravilno izvršiti kalibraciju što je odrađeno u *setup()* dijelu pozivom funkcije *calculate_IMU_error()*. Sama funkcija prvo izvrši 200 očitavanja senzora i uzme poslednju vrijednost kao validnu. Ideja je da sistem sačeka da se senzor umiri i onda krene sa daljim radom.

Takođe bitan parametar prilikom računanja pređenog puta jeste vrijeme. U tu svrhu iskorišćena je gotova arduino funkcija *millis()* koja odbrojava u milisekundama, od trenutka kada je kontroler dobio napajanje ili od momenta pritiska restart tastera. Potrebno je čuvati i poslednju vrijednost proteklog vremena, koju smiještamo u posebnu promjenljivu i konstantno je oduzimati od trenutnog vremena, radi pravilnog računanja proteklog vremena.

```
previousTime1 = currentTime1;  
currentTime1 = millis();  
elapsedTime1 = (currentTime1 - previousTime1) / 1000;
```

Slika 19 Dio koda za računanje vremena

Razlog za postojanje razlike ova dva vremena, odnosno neka vrijednost Δt , je zbog samog izvršavanja ostakta koda i ponovnog nailaska na ovu liniju koda (ova vrijednost je jako mala, ali opet postoji). Kao što je ranije navedeno, potrebno je konstantno pratiti vrijednost ubrzanja po Y osi (*AccY u kodu*) za računanje trenutne brzine, a vrijednost trenutne brzine je potrebno pomnožiti sa vrijednosti proteklog vremena Δt kako bi smo dobili vrijednost trenutnog puta.

```
trenutnaBrzina += AccY * elapsedTime1;  
trenutniPut += trenutnaBrzina * elapsedTime1;  
trPut += trenutniPut;
```

Slika 20 Računanje pređenog puta u kodu

Poznavajući vrijednost trenutnog puta, koji se prilikom kretanja povećava i smiješta u istu promjenljivu, i vrijednosti ugla po z osi možemo pravilno proračunati X i Y kordinate na sledeći način:

```
yaw2 = radians(yaw);  
deltaY = distanca * cos(yaw2);  
deltaX = distanca * sin(yaw2);  
novaX += deltaX;  
novaY += deltaY;
```

Slika 21 Dio koda za računanje koordinata x i y

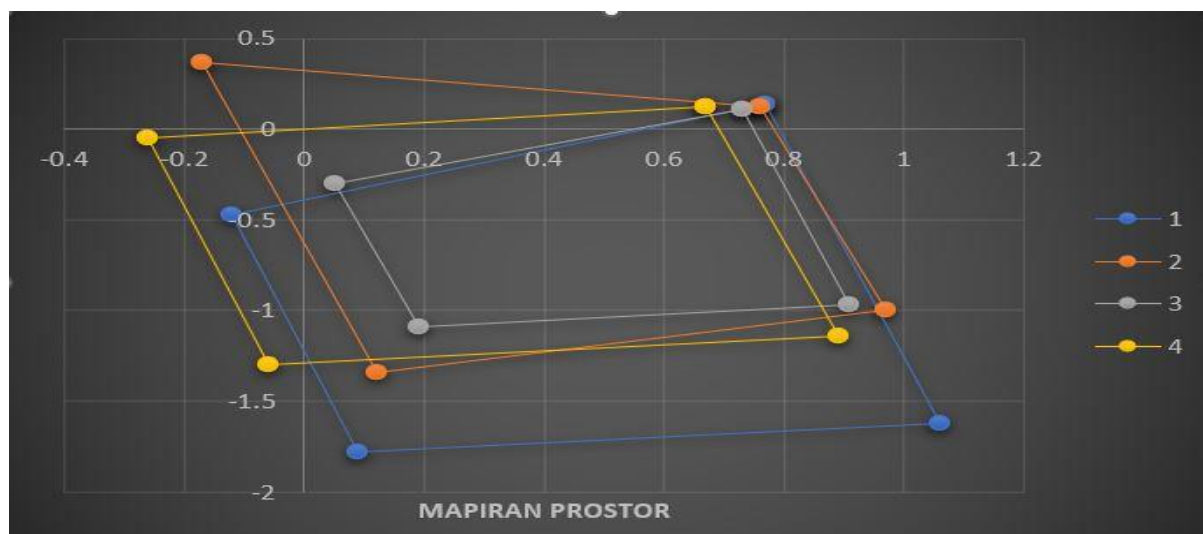
7.1 Mapiranje

Nakon svakog skretanja lijevo ispisujemo vrijednosti preko blutut modula kao što je prikazano na sledećoj slici. Ovdje vidimo vrijednosti za koordinate X i Y, pređeni put u toku vremena, distanca pređena između dva skretanja kao i ugao koji se nakon svakog skretanja uvećava. Za potrebe testiranja ovaj ispis je urađen za četiri različita obilaženja terena tako da bi uvidjeli kako se greška sa senzora postepeno nagomilava i odstupa od očekivanih rezultata.

```
Kord X: 0.67  
PrPut: 0.68  
dst: 0.68  
yaw:80.01  
Kord Y: -0.93  
Kord X: 0.84  
PrPut: 1.74  
dst: 1.06  
yaw:170.99  
Kord Y: -1.04  
Kord X: 0.14  
PrPut: 2.44  
dst: 0.70  
yaw:261.31  
Kord Y: 0.08  
Kord X: -0.04  
PrPut: 3.58  
dst: 1.13  
yaw:350.86
```

Slika 22 Rezultati testiranja ispisani na terminalu

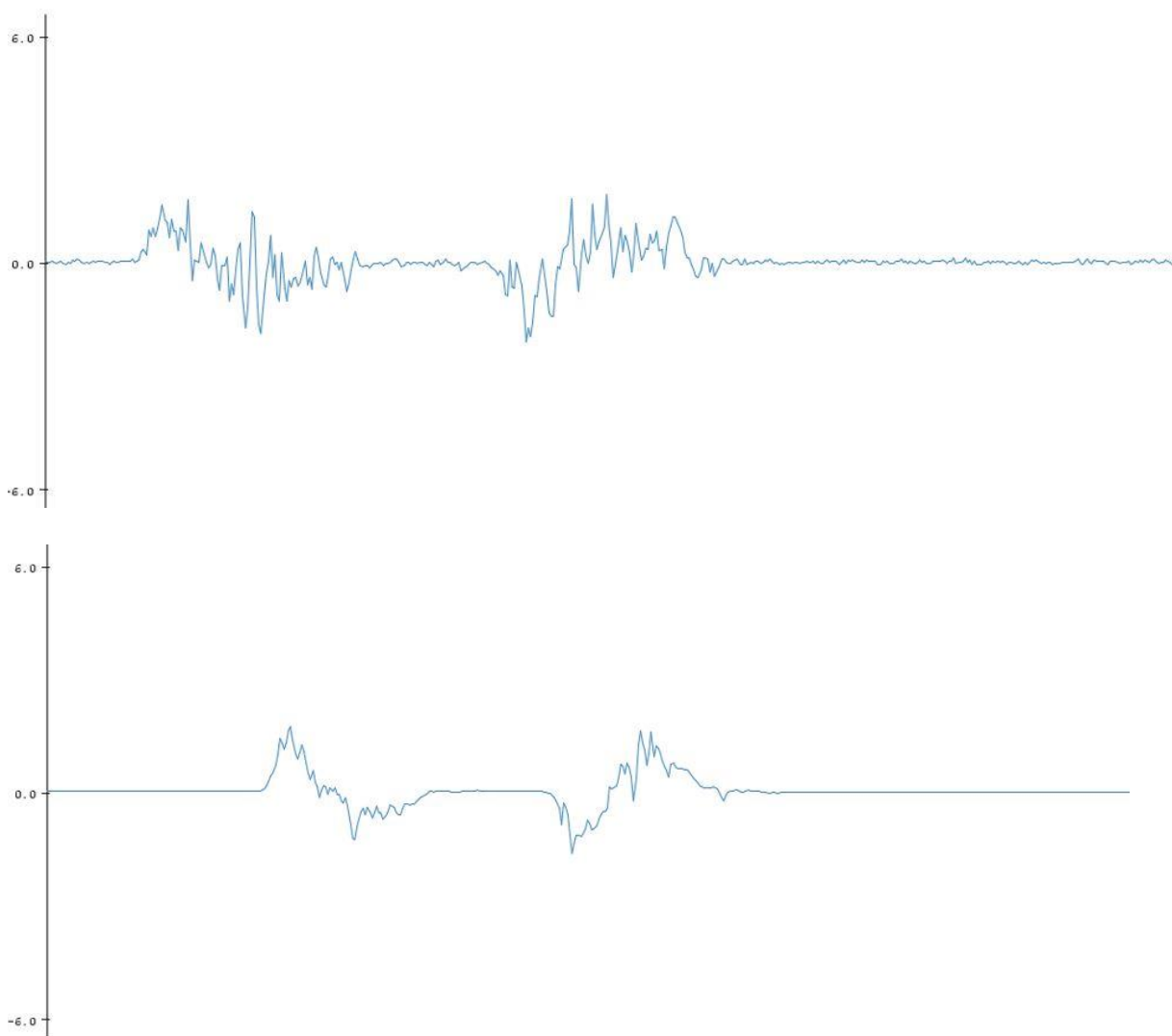
Kao što je navedeno u poglavlju 3, potrebno je mapirati odnosno pravilno izračunati četiri vrijednosti koje mogu poslužiti za crtanje kvadrata. Na sledećoj slici je prikazano poređenje proračunatih puteva za četiri kompletirane putanje, pritom tenkić započinje narednu putanju sa poslednje proračunate pozicije. Radi jednostavnosti upotrebljen je lavirnit u obliku kvadrata dimenzija 1x1, odnosno 1m².



Slika 23 Rezultati mapiranja dobijeni u Excel-u

7.2 Rezultati filtriranja signala

Prilikom kretanja robota u fizičkom svijetu nastaje dosta vibracija zbog same podloge po kojoj se kreće. Te vibracije se prenose i na sam senzor MPU6050 koji je jako osjetljiv. Na sledećoj slici može se vidjeti razlika između nefiltrirane i filtrirane vrijednosti ubrzanja dobijenih na ploteru. Ovako filtrirane vrijednosti se koriste i dosta poboljšavaju proračun. Treba biti oprezan i ne pretjerivati sa filtriranjem, jer može doći do gubitka podataka.



Slika 24 Rezultati ubrzanja akcelerometra bez Kalman filtra (slika gore) i rezultati nakon primjene Kalman filtra (slika dole)

8.ZAKLJUČAK

Projekat je uspješno realizovan, odnosno izvršava zahtijevano mapiranje prostora u kom se tenkić kreće. Projekat je započet kao skup raznih ideja, ali je na kraju odlučeno da se senzor MPU6050 upotrebi radi poboljšanja autonomnog kretanja tenkića nakon čega je odlučeno da primarni zadatak ovog senzora bude mapiranje prilikom kretanja. Zbog dosta hardverskih problema, u toku razvoja, odlučili smo se za mapiranje jednostavnog lavirinta u obliku kvadrata čime je primarni zadatak projekta fokusiran na očitavanja senzora MPU6050. Naravno, moramo uzeti u obzir da sam senzor nije najprecizniji i „gomila“ grešku u toku samog rada pa je poželjno izvršiti veći broj mjerenja i uzeti srednju vrijednost. Ovo je jedna od glavnih mana ovog senzora, ali je se pokazao dosta dobar s obzirom na nisku cijenu. Za ovakvu vrstu mapiranja najbolje je koristiti enkodere, lidar senzor ili precizniji i skuplji žiroskop.

Na početku izrade projekta najveći zadatak je predstavljala kalibracija senzora. Bilo je potrebno naći, empirijskim putem, koeficijent koji će se konstantno oduzimati od mjerene vrijednosti. Problem je nastao jer je ovo odrađeno dok se senzor nalazio u stacionarnom stanju i nismo uzeli u obzir da će se na sam senzor prenositi vibracije tokom kretanja. Kao rješenje novonastalog problema je iskorišćen Kalman filter koji je dosta poboljšao očitavanja, ali je zaključeno da očitane vrijednosti ipak ne treba previše izfiltrirati jer gubimo preciznost prilikom računanja pređenog puta.

Za skretanje je isključivo upotrebljeno očitavanje ugla po Z osi i to tako da kada tenkić detektuje zid započne skretanje lijevo dok ne dostigne threshold vrijednost od 90 stepeni. Zbog samih vibracija tokom skretanja zaključeno je da će threshold vrijednost ustvari biti 80 stepeni u kodu, što predstavlja 90 stepeni u fizičkom svijetu. Prilikom povratka autića na početnu poziciju očitavanja ovog ugla su najpreciznija, konstantno dobijamo vrijednost od 350 stepeni, što je jako blizu realne vrijednosti. Ovo je postignuto dobrim koeficijentima Kalman filtera.

Ultrasonični senzori takođe nisu najprecizniji jer njihov pravilan rad zavisi od reflektovane komponente zvuka, ali su upotrebljeni zbog svoje niske cijene. Potrudili smo se da se tenkić kreće uz samu ivicu, ali zbog nepreciznosti senzora daljine, povremeno se desi da gusenica zakači zid i u račun se doda nagla promjena ubrzanja što na kraju rezultuje u većim X i Y komponentama. Tokom pravolinijskog kretanja tenkića konstantno se provjerava ubrzanje, odnosno podešen je još jedan threshold. Prilikom nepredviđenih situacija u toku kretanja kao što su sudar ili rupa možemo očitati naglu promjenu ubrzanja i zaštititi tenkić tako što ga privremeno stopiramo. Ovo je takođe uspješno implementirano. Veza između mikrokontrolera i senzora ostvarena je pomoću Protoboard-a i tako je ostavljeno zbog vremenskog ograničenja, što predstavlja i jednu od najvećih mana ovog projekta. Parazitne vrijednosti su velike i tokom samog kretanja može doći do gubitka kontakta. Takođe iskoristili smo Arduino UNO pločicu i bili smo ograničeni sa 13 digitalnih pinova, od kojih su svi iskorišćeni. Preporučljivo bi bilo da smo upotrebili i enkodere radi poređenja pređenog puta što bi dodatno pomoglo prilikom kalibracije senzora. Takođe bilo bi poželjno dodati još dva ultrasonična senzora čime bi imali pokriven čitav krug i poboljšali preciznost autonomnog kretanja. To bi naravno značilo da bi se tenkić mogao snaći u dosta kompleksnijem lavirintu. Preporučuje se izrada posebnog PCB-a na kom će se nalaziti i biti povezani svi senzori

9. LITERATURA

<https://www.alldatasheet.com/datasheet-pdf/pdf/1132807/TDK/MPU-6050.html>

[1] datum pristupa sajtu: 08. Avgust 2023.

<https://www.optolab.ftn.uns.ac.rs/index.php/education/course-notes/senzori-i-aktuatori-sia/> [2] datum pristupa sajtu: 30. Septembar 2023.

<https://www.electronicwings.com/sensors-modules/mpu6050-gyroscope-accelerometer-temperature-sensor-module> [3] datum pristupa sajtu: 11. Avgust 2023.

<https://circuitdigest.com/microcontroller-projects/interfacing-mpu6050-module-with-arduino#:~:text=How%20does%20MPU6050%20Module%20Work,of%20a%20system%20or%20object.> [4] datum pristupa sajtu: 08. Avgust 2023.

<https://www.watelectronics.com/ir-sensor/> [5] datum pristupa sajtu: 05. Avgust 2023.

<https://electronics.howstuffworks.com/motor.htm> [6] datum pristupa sajtu: 11. Avgust 2023.

<https://www.hackster.io/lakshyajhalani56/l298n-motor-driver-arduino-motors-motor-driver-l298n-0f2cfd> [7] datum pristupa sajtu: 10. Avgust 2023.

<https://www.bzarg.com/p/how-a-kalman-filter-works-in-pictures/> [8] datum pristupa sajtu: 10. Avgust 2023.

<https://www.electronicwings.com/arduino/mpu6050-interfacing-with-arduino-uno> [9] datum pristupa sajtu: 10. Avgust 2023.

<https://www.instructables.com/Self-Path-Correcting-Robot/> [10] datum pristupa sajtu: 30. Septembar 2023.

<https://www.instructables.com/Arduino-AND-Bluetooth-HC-05-Connecting-easily/> [11] datum pristupa sajtu: 30. Septembar 2023.

<https://howtomechatronics.com/tutorials/arduino/ultrasonic-sensor-hc-sr04/> [12] datum pristupa sajtu: 30. Septembar 2023.

<https://howtomechatronics.com/tutorials/arduino/arduino-and-mpu6050-accelerometer-and-gyroscope-tutorial/> [13] datum pristupa sajtu: 30. Septembar 2023.

<https://mae.ufl.edu/designlab/Class%20Projects/Background%20Information/Friction%20Coefficients.htm> [14] datum pristupa sajtu: 15. Septembar 2023.

<https://www.maxongroup.com/maxon/view/msp/> [15] datum pristupa sajtu: 30. Septembar 2023.