IMPLEMENTACIJA UPRAVLJAČKOG MODULA ZA IZBJEGAVANJE PREPREKA SVESMJERNIM MOBILNIM ROBOTOM

Lesičar, Alen

IMPLEMENTACIJA UPRAVLJAČKOG MODULA ZA IZBJEGAVANJE PREPREKA SVESMJERNIM MOBILNIM ROBOTOM

Lesičar, Alen

VELEUČILIŠTE U KARLOVCU

STROJARSKI ODJEL

Stručni studij Mehatronike

ALEN LESIČAR

Implementacija upravljačkog modula za izbjegavanje prepreka svesmjernim mobilnim robotom

Implementation of an obstacle avoidance control module for the omnidirectional mobile robot

Završni rad

Karlovac, 2020. godine.

VELEUČILIŠTE U KARLOVCU

STROJARSKI ODJEL

Stručni studij Mehatronike

ALEN LESIČAR

Implementacija upravljačkog modula za izbjegavanje prepreka svesmjernim mobilnim robotom

Implementation of an obstacle avoidance control module for the omnidirectional mobile robot

Završni rad

mentor: Denis Kotarski, mag.ing.mech

Karlovac, 2020. godine.

PREDGOVOR

Izjavljujem da sam ovaj rad izradio samostalno koristeći znanja stečena u dosadašnjem dijelu studija te uz navedenu literaturu.

Zahvaljujem se svojoj obitelji koja mi je omogućila ovo školovanje te mi je bila potpora kroz studij, te prijateljima koji su mi uljepšali studiranje.

Posebno se zahvaljujem profesoru i mentoru Denisu Kotarskom, na savjetima i uputstvima koje mi je davao kroz ovaj rad, te na strpljenju.

Alen Lesičar

Sadržaj:

Sažetak

U ovome završnome radu opisani su svesmjerni mobilni roboti. Takvi roboti su vrlo korisni za obavljanje raznih zadataka te imaju širok spektar mogućnosti. Građa samog kotača omogućava gibanje u svim smjerovima, te rotaciju tijekom određenog gibanja. Za programiranje robota potreban nam je Arduino IDE preko kojeg unosimo kod. Gibanje nebi bilo moguće bez hardverskih komponenti kao što su: servo motori, Arduino Nano pločica, HC-SR04 ultrazvučni senzor. Glavni zadatak robota je izbjegavanje prepreka pomoću ultrazvučnog senzora. Prije svega moramo biti upoznati s kinematikom samog robota. Sve ove značajke predstavljene su u daljnjem tekstu.

Summary

This final paper describes omnidirectional mobile robots. Such robots are very useful for performing a variety of tasks and have a wide range of options. The structure of the wheel itself allows motion in all directions and rotation during a certain motion. To program the robot, we need an Arduino IDE through which we enter the code. The motion would not be possible without hardware components such as servo motors, Arduino Nano, HC-SR04 ultrasonic sensor. The main task of the robot is to avoid obstacles with the help of an ultrasonic sensor. First of all, we need to be familiar with the kinematics of the robot itself. All of these features are presented in the text below.

Popis slika

Popis tablica

1. UVOD

Robotika je višedisciplinarna grana tehnike koja objedinjuje mehaniku, fiziku, matematiku, automatsko upravljanje, elektroniku, umjetnu inteligenciju i drugo. Razvoj robota pokrenut je zbog želje čovjeka da pronađe zamjenu za sebe koja bi imala mogućnost oponašanja njegovih svojstava u različitim primjenama, uzimajući u obzir i međudjelovanje sa okolinom koja ga okružuje.

Roboti su automatizirani strojevi višestruke namjene koji se sastoje od konstrukcije s pripadajućim pogonskim uređajima, senzora i upravljačkog uređaja. Dijele se po stupnju pokretljivosti (statički i mobilni roboti), strukturi konstrukcije (mehatronički, biotronički i bioroboti), namjeni (industrijski, medicinski, edukacijski, podvodni, roboti za istraživanje svemira, vojni roboti, osobni roboti), veličini (makroroboti, mikroroboti i nanoroboti).

Inteligentni roboti posjeduju sposobnost učenja, rasuđivanja i donošenja zaključaka te imaju visok stupanj funkcionalne, organizacijske i mobilne autonomnosti. Roboti te skupine razvijaju se ubrzano, usporedno s razvojem naprednih informacijskih tehnologija i umjetne inteligencije. Očekuje se kako će se inteligencija robota općenito, a posebice biorobota ostvarenih genetičkim inženjerstvom, još više približiti inteligenciji čovjeka.

Danas se najviše primjenjuju industrijski roboti, npr. u automobilskoj industriji za sastavljanje dijelova vozila, bojenje ili zavarivanje karoserije i dr. Roboti se osobito rabe u okruženju opasnom za čovjeka (podvodni roboti, svemirske robotske letjelice, roboti za razminiranje i dr.), kao sredstva unutarnjeg transporta, ali i za zabavu, natjecanja te kao dječje igračke. [1]

Slika 1: Robotika u budućnosti.

2. KOPNENI MOBILNI ROBOTI

Mobilni roboti moraju posjedovati svojstva mobilnosti, autonomnosti i “inteligencije”. Mobilnost podrazumijeva gibanje robota kroz okolinu. Autonomnost je također ograničena ljudskom interakcijom. Mobilni roboti se dijele na više načina. Najpopularnije podjele su prema vrsti medija po kojem se gibaju (kopno, voda, zrak) i po načinu na koji ostvaruju svoje gibanje (kotači, noge, zmijoliki, itd).

2.1. Tipovi i namjene

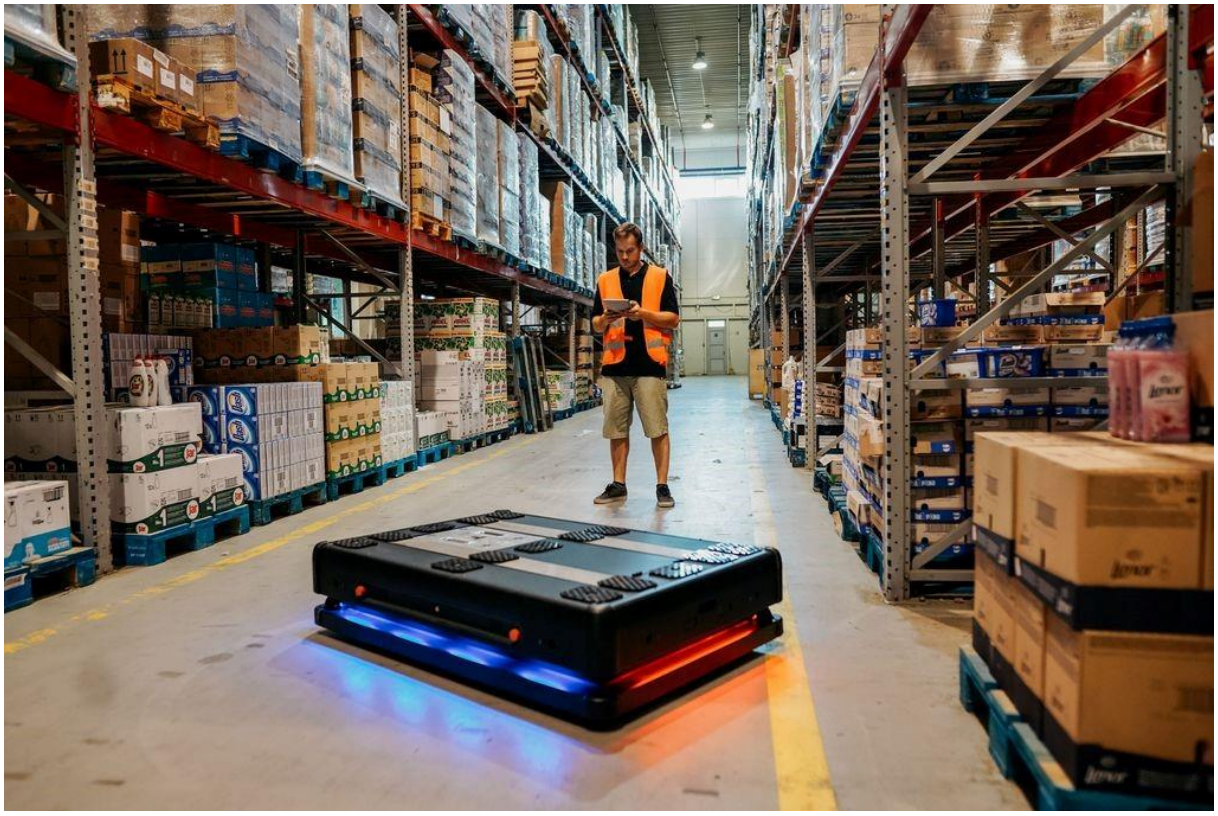
Kopneni mobilni roboti se u današnje vrijeme sve više koriste u tvornicama, skladištima i kućanstvima. Ovakvi tipovi robota olakšavaju ljudima poslove. Što se tiče tvornica za dobar primjer imamo tvornicu automobila marke Audi. U takvom okruženju roboti rade poslove gdje su u blizini visokih temperatura i tlakova bez ikakvih problema, dok bi ljudima to bio puno veći problem.

Ljudi su oduvijek imali potrebu za skladištenjem stvari. U današnje vrijeme to je lako izvesti pomoću robota. Postoje i pametna skladišta koja funkcioniranju na principu automatiziranog skladištenja stvari. Skladišta su većinom u obliku matrice koja se sastoji od željenog broja redaka i stupaca u kojima se spremaju stvari. Kod ovakvih skladišta čovjeku bi bilo prenaporno pamtiti te konstantno premještati kutije sa stvarima, što nam olakšava robot. Svoju primjenu roboti nalaze i u kućanstvu, kod poslova kao što su: čišćenje, usisavanje, itd. Ovo su neke od marki spomenutih robota: Fanuc, ABB, Gideon Brothers, Tonor robot usisavač.

2.2. Autonomni mobilni roboti

Autonomni roboti su skupina robota koji donose odluke o svojoj brzini i smjeru kretanja ovisno o njihovom cilju. Ovakvi roboti se koriste u prostorima koji prethodno nisu poznati ili se mijenjaju tijekom rada. Zbog ovakve okoline rada robota potrebna je mogućnost prilagođavanja, tj. navigacije robota. Sposobnost prepoznavanja prepreke te zaobilaženje navedenog je ključna operacija autonomnog mobilnog robota. Kako bi robot bio u mogućnosti prepoznavanja, te izbjegavanja prepreka potrebni su mu razni senzori, npr. ultrazvučni senzor. Uz senzore koriste se i razni algoritmi. Pored senzora i algoritama od

velike važnosti je i struktura robota. Način na koji je robot građen od velike je važnosti što se tiče mobilnosti i mogućnosti promjene smjera. Roboti koji imaju mogućnost kretanja u svim smjerovima imaju veliku prednost nad robotima koji tu mogućnost nemaju. Gibanje u uskim prostorima s velikim brojem prepreka zahtijeva precizno pozicioniranje robota što je najlakše izvesti pomoću robota koji se može gibati u svim smjerovima, tj. svesmjernog robota.



Slika 2: Autonomni mobilni robot Gideon Brothersa.

2.2.1.Komercijalni autonomni roboti

Na slici 2 prikazan je autonomni robot tvrtke Gideon Brtothers. Tvrtka Gideon Brothers osnovana je 2017. godine i od samog početka fokusirana je na razvoj proizvoda baziranih na sustavima za robotsku autonomiju temeljenu na vizualnoj percepciji koja kombinira stereoskopske kamere i duboko učenje. Zbog tih značajki su ljudi, oprema i ostali strojevi u pokretu sigurni od gibanja robota kako nebi došlo do sudara ili nečega sličnog. Roboti imaju nosivost oko 800kg. Autonomni mobilni roboti Gideon Brothersa trenutno u

Hrvatskoj u skladištima koriste tvrtke kao što su: Atlantic grupa, Tokić, Hrvatska pošta, te od nedavno vodeća njemačka logistička tvrtka DB Schenker.

Robota odlikuje potpuna autonomija te mu nije potreban nikakav vanjski sustav navođenja kao npr. markeri ili magnetske vrpce. Sam percipira svoju okolinu i izrađuje mapu prostora, te se na istoimenoj mapi lokalizira. Mapu prostora robot mijenja prema registriranim promjenama u okolini.

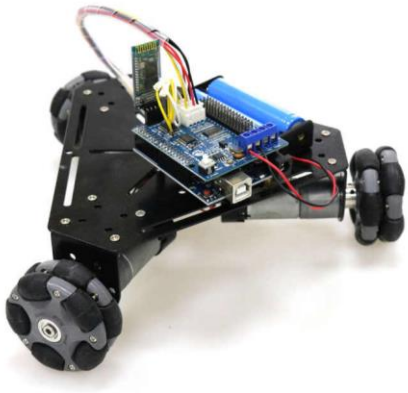
Robot zahtjeva vrlo malo fizičkih elemenata. Potreban mu je okvir za palete gdje robot pokupi ili dostavi teret te stanica za punjenje baterija. Jedinstvena značajka je mogućnost zamjene baterije robota tijekom njegovog rada bez gašenja što smanjuje vrijeme punjenja na minimum.

Gideonova autonomija “Advanced Visual Perception” nudi puno širi spektar scenarija kod korištenja od konkurencije zbog toga jer je tehnologija kod lokalizacije i prepoznavanja objekata neusporedivo bolja nego standardna rješenja koja se baziraju samo na LiDAR senzorima (laserski "radari"). Primjerice, LiDAR senzor ne vidi predmete koji su nisko na tlu kao što su recimo vilice praznog viličara koje moraju biti spuštene zbog sigurnosnih propisa.

Ovi roboti koriste dvije stereoskopske kamere gdje svaka od njih snima sliku iz svoje perspektive. Kombinacijom te dvije slike dobivamo 3D sliku preko koje robot svoje okruženje vidi kao i čovjek svojim okom. Kratko rečeno robot može vidjeti dubinu i širinu prostora. [2]

2.3. Svesmjerni kopneni mobilni roboti

Veliki korak naprijed što se tiče visoke razine i sigurnosti manevriranja je svesmjerni robot koji se koristi umjesto korištenja konvencionalne platforme robota s dva kotača koji ima dodatni prednji kotač za funkciju skretanja. Svesmjerni roboti razlikuju se i po broju kotača. Postoji izvedba svesmjernog robota koji koristi četiri kotača za gibanje, te izvedba s tri kotača. Svesmjerni roboti imaju prednost što mogu promijeniti smjer gibanja bez poduzimanja međukoraka rotacije kao npr. mobilni robot s diferencijalnim pogonom. Ovakva vrsta robota može se gibati od jedne točke prema drugoj te se istovremeno rotirati.



Slika 3: Izvedba svesmjernog robota s tri kotača.

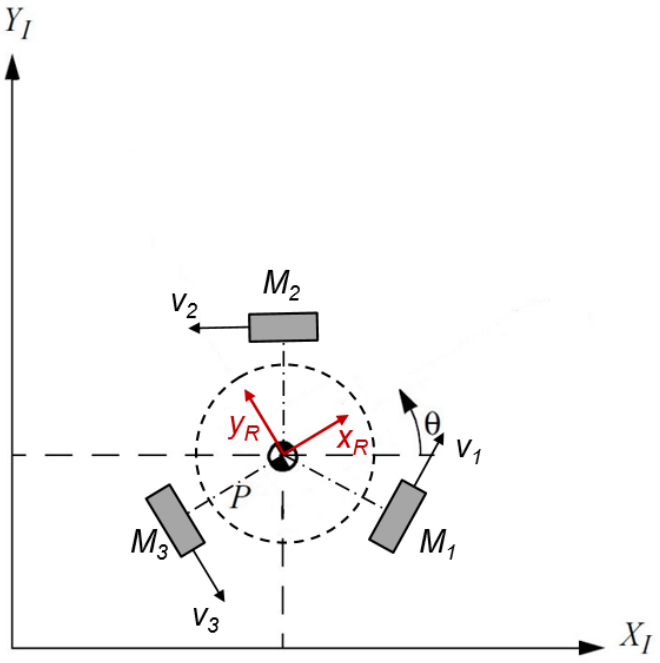
Slika 4: Izvedba svesmjernog robota sa četiri kotača.

3. IZVEDBA SVESMJERNOG KOPNENOG MOBILNOG ROBOTA

Postoji nekoliko tipova svesmjernih kopnenih mobilnih robota, a karakteristični dio koji ih određuje je izvedba kotača. U osnovi postoje dva osnovna tipa kotača svesmjernih robota koji određuju model robota. Prvi tip je tzv. omni ili švedski kotač koji se koriste kod robota s tri (četiri) pogonska aktuatora te tzv. mecanum kotač koji se koristi kod robota sa četiri pogonska aktuatora.

3.1. Matematički opis

Svesmjerni kopneni mobilni roboti egzistiraju u dvodimenzionalnom prostoru te imaju tri stupnja slobode gibanja (engl. *Degrees of Freedom - DOF*). Za opisivanje kinematike robota, potrebno je definirati dva Kartezijeva koordinatna sustava: inercijski koordinatni sustav($\mathcal{F}^I$) i koordinatni sustav mobilnog robota ($\mathcal{F}^R$).

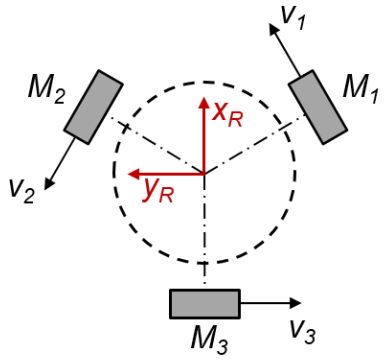


Slika 5: Prikaz robota u kartezijevom koordinatnom sustavu [3].

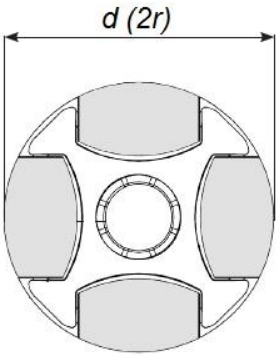
Stupnjevi slobode predstavljeni su pozicijom (X,) i orijentacijom (θ) robota koje definiraju stanje (**ε**)

$$\boldsymbol{\varepsilon} = \begin{bmatrix} X \\ Y \\ \theta \end{bmatrix}. \tag{1}$$

Ulazi modela jesu kutne (rotacijske) brzine pojedinog kotača ω_i, a izlazi modela pozicija i orijentacija mobilnog robota.



Slika 6: Geometrijski raspored kotača [3].

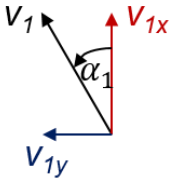


Slika 7: Dimenzije kotača.

Translacijske brzine pojedinih kotača ovise o kutnim (rotacijskim) brzinama kotača te o njihovom promjeru (radijusu).

$$\begin{aligned} v_1 &= r\omega_1, \\ v_2 &= r\omega_2, \\ v_3 &= r\omega_3 \end{aligned} \tag{2}$$

Sljedeće je potrebno definirati kutove kotača u odnosu na koordinatni sustav mobilnog robota ($\mathcal{F}^R$).



$$\begin{aligned} \alpha_1 &= 30° \\ \alpha_2 &= 150° \\ \alpha_3 &= 270° \end{aligned}$$

Sada se mogu definirati brzine za pojedini kotač u ovisnosti o geometrijskom rasporedu, za prvi kotač brzine su jednake:

$$\begin{aligned} v_{1x} &= \cos(\alpha_1)v_1, \\ v_{1y} &= \sin(\alpha_1)v_1. \end{aligned} \tag{3}$$

Na temelju brzina svih kotača, dobivaju se translacijske i rotacijska brzina robota u $\mathcal{F}^R$.

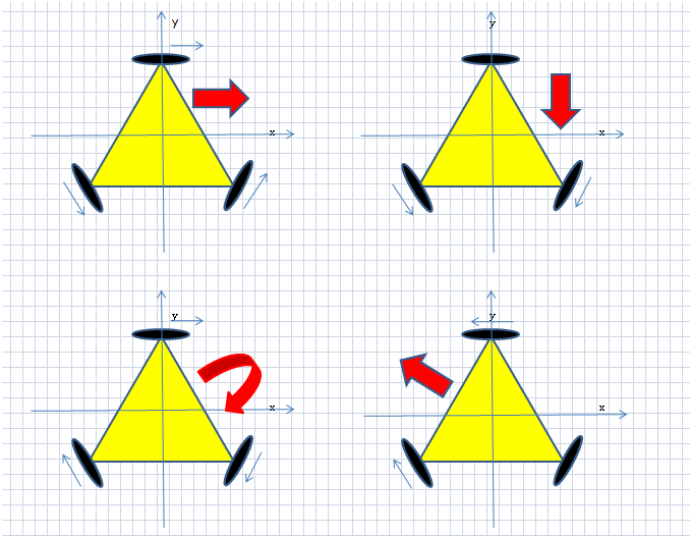
$$\begin{aligned} v_x &= v_{1x} + v_{2x} + v_{3x}, \\ v_y &= v_{1y} + v_{2y} + v_{3y} \\ \omega &= \frac{v_1}{R} + \frac{v_2}{R} + \frac{v_3}{R} \end{aligned} \tag{4}$$

Može se zapisati matrično

$$\begin{bmatrix} v_x \\ v_y \\ \omega \end{bmatrix} = \begin{bmatrix} \cos(\alpha_1) & \cos(\alpha_2) & \cos(\alpha_3) \\ \sin(\alpha_1) & \sin(\alpha_2) & \sin(\alpha_3) \\ 1/R & 1/R & 1/R \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} \tag{5}$$

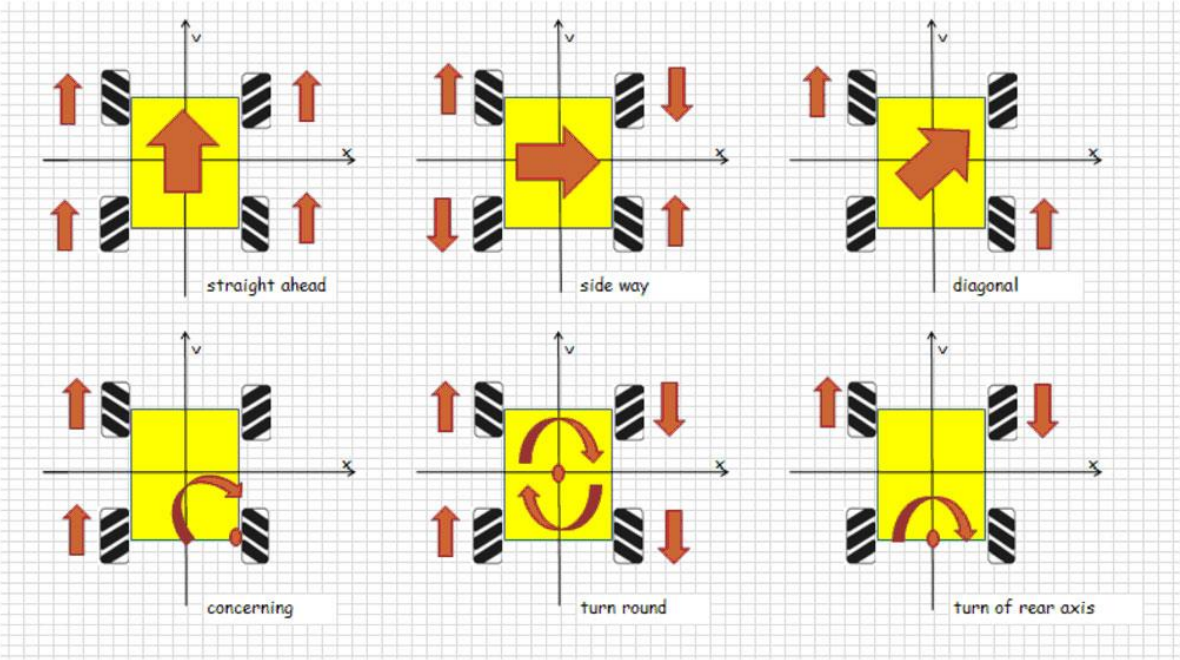
3.1.1.Primjeri gibanja svesmjernih robota

Shematski prikaz gibanja svesmjernog robota s tri omni kotača prikazan je na slici 8.



Slika 8: Shematski prikaz gibanja svesmjernog robota s tri omni kotača [4].

Shematski prikaz gibanja svesmjernog robota sa četiri omni kotača prikazan je na slici 9.



Slika 9: Shematski prikaz gibanja svesmjernog robota sa četiri mecanum kotača [5].

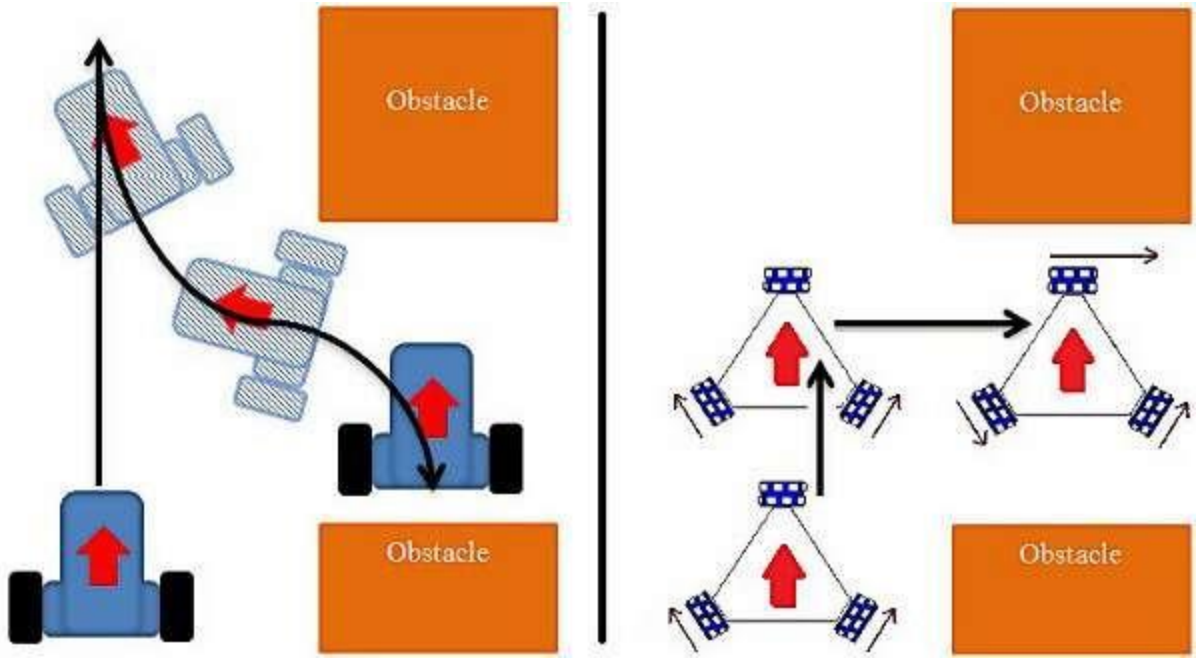
3.2. Komponente robota

3.2.1.Kotači

Svesmjerni[1] robot može imati, kao što je prikazano u prethodnom tekstu tri kotača, no postoje i izvedbe s više kotača. Takve izvedbe robota osiguravaju puno bolju stabilnost i nosivost.

Robot s diferencijalnim pogonom zahtijeva dodatne korake skretanja za promjenu smjera dok svesmjernom robotu nisu potrebni ti međukoraci. Sve to zahvaljujući građi kotača. To znači i puno bolji rad svesmjernog robota u okolini ili poslu koji zahtijevaju puno skretanja, dok je robot s diferencijalnim pogonom puno sporiji u tome zbog dodatnih koraka kretanja. Slika 10 prikazuje usporedbu svesmjernog robota i robota s diferencijalnim pogonom.

[1] Omnidirectional



Slika 10: Usporedba robota s diferencijalnim pogonom i svesmjernog robota [4].

Omni (švedski) kotač

Omni kotač je sastavljen od više valjaka koji su smješteni po obodu konstrukcije kotača. Valjci kotača su u neposrednom kontaktu s tlom. Postoji više tipova omni kotača, a u ovome tekstu pisat će se o standardnoj izvedbi. Takva izvedba kotača ima valjke koji su smješteni tako da mogu rotirati u ortogonalnom smjeru s obzirom na smjer rotacije kotača. Postoje i izvedbe omni kotača koji imaju valjke smještene tako da valjci rotiraju u ne – ortogonalnom smjeru. Pod ne - ortogonalnim smjerom mislimo na kut od 45° između kotača i valjaka. [6]

Slika 11: Omni (švedski) kotač [3].

Omni kotači se ugrađuju na jednakoj udaljenosti od središta robota tako da je kut između svih kotača jednak. Ako oko središta kotača zamislimo kružnicu, tangenta na kružnicu nam pokazuje vektor brzine gibanja kotača, a ujedno i smjer.

Slika 12: Svesmjerni kotač druge izvedbe (okomiti).

Ovaj kotač ima jednaku učinkovitost kao i standardni omni kotač, kao i veoma sličnu građu. Okomiti kotač ima puno više valjaka, tj. kotačića, po obodu kotača koji omogućavaju ortogonalno gibanje u odnosu na smjer rotacije samoga kotača. Kontakt s

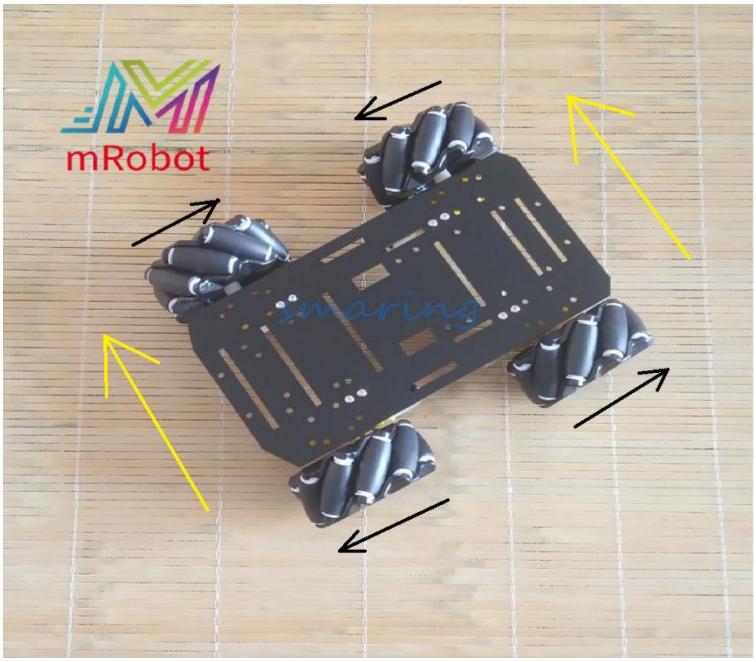
tlom je osiguran najviše s dva kotačića, ako govorimo o ravnoj površini. Ovakvi kotači se često koriste na natjecanjima s robotima.

Mecanum kotač

Za mecanum kotače preporuča se korištenje najmanje četiri kotača kod pokretanja robota. Kotači se ugrađuju u dvije vrste, svaka vrsta ima dva kotača. To znači npr. ako je platforma robota pravokutna, na svaki pravokutni kut ide po jedan kotač, no to nije jedini način pozicioniranja kotača. Kod kretanja robota prema naprijed kotači se okreću također prema naprijed, prema nazad se kotači rotiraju unazad. Ako je potrebno da robot ide u stranu to se ostvaruje tako da se svaki motor kotača rotira u suprotnom smjeru od nasuprotnog kotača. [7]

Slika 13: Mecanum kotač [4].

Slika 14 prikazuje gibanje robota u smjeru žutih strelica, dok je smjer vrtnje kotača označen crnim strelicama.

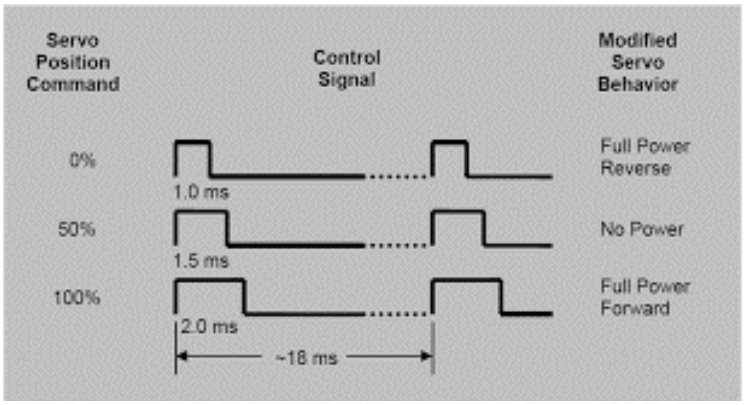


Slika 14: Gibanje svesmjernog robota u stranu pomoću ne - ortogonalnih univerzalnih kotača.

3.2.2.Aktuatori

Aktuatori su naprave koje se na pobudu nekog upravljačkog signala dovode u željani položaj, ostvaruje se gibanje, razvija sila kojom djeluju na okolinu. U osnovi aktuator pretvara neke električne ulaze u mehaničke izlaze. Izlazni signal je puno veći nego ulazni, tj. njihova energija. Servomotor je elektromotor koji prema primljenom upravljačkom signalu zauzima određeni zakretni položaj (zakretni ili rotacijski servomotor) ili mjesto na nekoj putanji (pravocrtni ili linearni servomotor), odnosno razvija odgovarajući zakretni moment ili silu. U ovom poglavlju pisat će se o servo motorima koji imaju mogućnost kontinuiranog kretanja u oba smjera.

Kontinuirani servo motor ima kut rotacije od 360 stupnjeva. Kada se šalje kontrolni impuls kontinuirano rotacijskom servo motoru, impuls kontrolira snagu, točnije okretni moment i pravac vratila. Zato, kada se naredi servo motoru 0% pozicija sa impulsom od 1 milisekunde to stvarno rezultira u servo motoru primjenu minimalne snage u povratnom pravcu. Slično kada se naredi servo motoru 100% pozicija to rezultira maksimalnom snagom prema naprijed. Ako se servu naredi 50% pozicija to rezultira nultim okretnim momentom. Za kontrolu je moguće koristiti bilo koji hardver i bilo koju biblioteku za servo motore, driveri nisu potrebni. [8]



Slika 15: Kontrola servo motora pomoću PWM signala [5].

Svaki servomotor ima 3 žice, najčešće smeđu (impuls), crvenu (plus), žutu (minus). Za pogon robota odabrani su servo motori kontinuirajućeg načina rada, oznake FS5103R. Njegove specifikacije su prikazane u tablici 1.

Tablica 1: Specifikacije servo motora FS5103R [9]

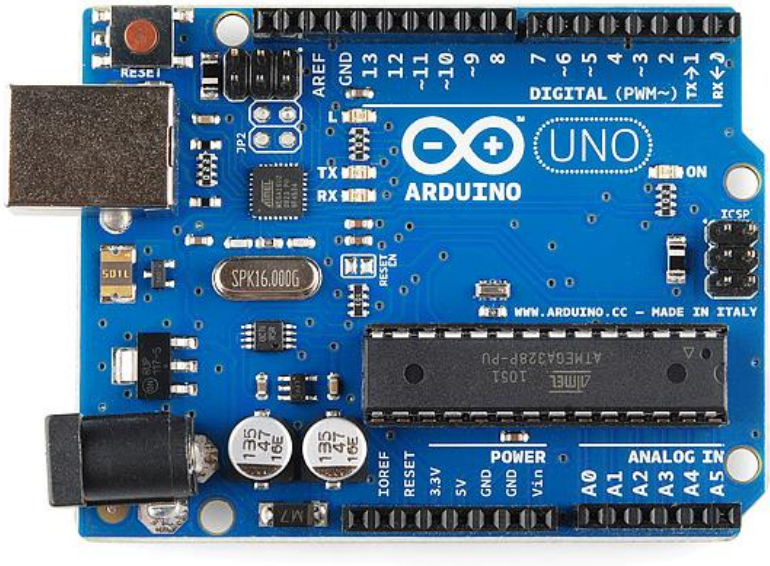
Radni napon	4.8 V – 6.0 V	
Dimenzija	40.8 x 20.1 x 38 mm	
Masa	40 g	
Brzina zakreta (4.8 V)	0.18sec/60°	
Brzina zakreta (6.0 V)	0.16sec/60°	
Zakretni moment (4.8 V)	3.0kg-cm	
Zakretni moment (6.0V)	3.2 kg-cm	

3.2.3.Upravljačka jedinica

Arduino je elektronička platforma namjenjana kreiranju elektroničkih projekata. Sastoji se od hardware i software dijela. Hardverski dio je zapravo fizički elektronički programibilni strujni krug koji se zove mikrokontroler. Softverski dio se naziva IDE[2], koji se pokreće na samome računalu i preko njega se programira i upravlja pločicom. Pločica je nastala u Italiji 2005. godine. Arduino pločicu je osmislila jedna grupa studenata. Postala je brzo

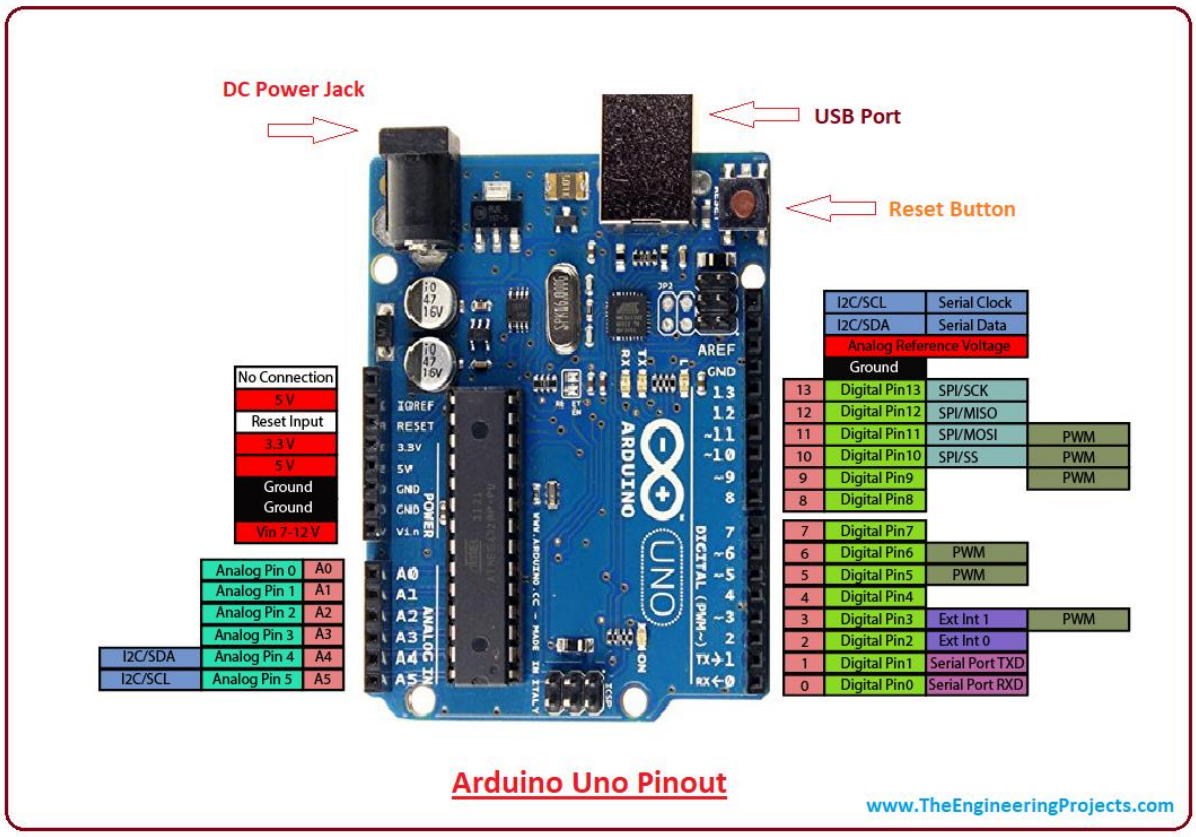
[2] Integrated Development Environment

popularna zbog svoje jednostavnosti. Za programiranje pločice nije potreban programator, već je potreban samo USB kabel koji omogućava spajanje na svako računalo bez obzira na operacijski sustav računala. Slika 16 prikazuje najpoznatiju i najkorišteniju Arduino pločicu.



Slika 16: Arduino Uno upravljačka jedinica.

Arduino Uno pločica ima više I/O (INPUT / OUTPUT) digitalnih i analognih pinova koji funkcioniraju na napajanju od 5V. Pločica se može napajati na tri načina: USB, Vin pin i pomoću istosmjernog priključka (DC Power Jack). USB podržava napon oko 5V dok Vin pin i DC priključak podržavaju napon u rasponu od 7 – 20V, preporučava se napajanje od 5V. Ploča je opskrbljena s više priključaka za uzemljenje (GND). Reset input pin daje mogućnost resetiranja pločice preko napisanog programa, dok postoji i mogućnost fizičkog resetiranja preko gumba za reset. Ploča ima 14 digitalnih pinova, od kojih 3, 5, 6, 9, 10 i 11 daju 8 – bitni PWM (Pulse Width Modulation) signal. Postoji ugrađeno LED svjetlo u ploči koja je spojena na pin 13. Kad je pin HIGH LED-ica je upaljena a kad je LOW onda je ugašena. Na slici 17 je vidljivo da postoji 6 analognih pinova. Analogni pinovi osiguravaju pretvorbu 10 – bitnog analognog u digitalni signal. Većina analognih pinova može se koristiti i kao digitalni. [10]



Slika 17: Popis pinova Arduino Uno upravljačke jedinice [7].

ATmega 328 mikrokontroler je središnji dio Arduino Uno upravljačke jedinice. ATmega je 8 – bitni mikrokontroler koji ima 28 pinova. Ima flash memoriju veličine 32KB. Napaja se naponom u rasponu od 3.3 V – 5.5 V, no koristimo 5V kao standardno. Ima 1KB EEPROM[3] memorije što ukazuje na mogućnost pohranjivanja podataka i prikazivanja rezultata i nakon odstranjivanja napajanja. Ugrađena su i tri 8 – bitna timer-a i jedan 16 – bitni.

Slika 18: ATmega 328 mikrokontroler.

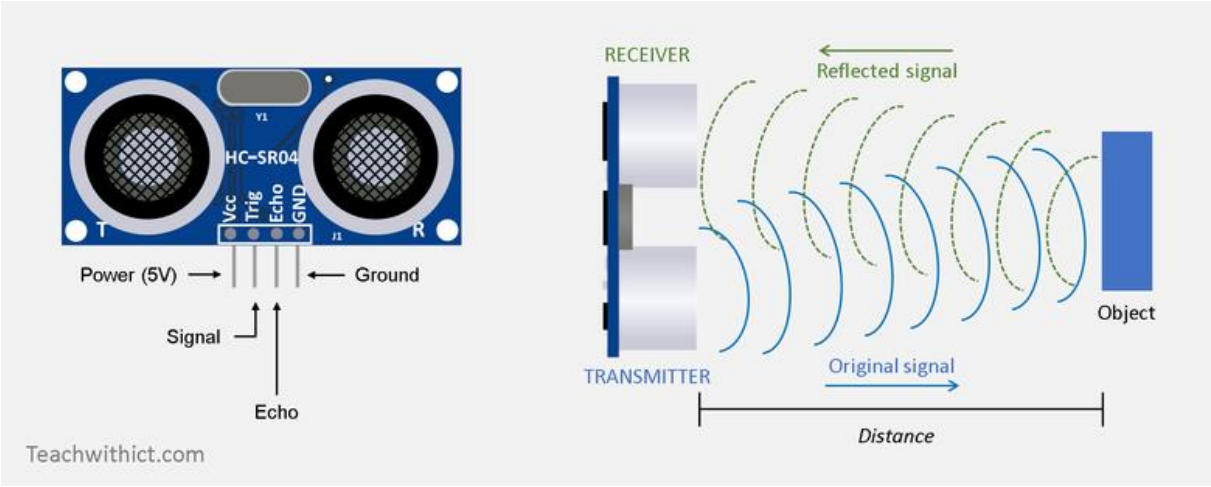
[3] Electrically Erasable Programmable Read Only Memory

3.2.4.Senzori udaljenosti

Senzori su uređaji koji pretvaraju neku fizikalnu veličinu (temperaturu, tlak, broj okretaja motora, itd.) u signal koji je pogodan za daljnju obradu. Senzori se koriste u svakodnevnom životu, a da ljudi toga nisu ni svjesni, npr. u automobilima, robotici, itd. U ovome poglavlju pojasnit će se princip rada senzora udaljenosti: ultrazvučnog senzora (HC-SR04) i infracrvenog senzora.

Ultrazvučni senzor, kao što mu i samo ime govori, koristi ultrazvuk za izračunavanje udaljenosti od nekog objekta. Ultrazvuk je zvuk na frekvenciji većoj od gornje granice ljudskog sluha. Ljudsko uho ima mogućnost prepoznavanja zvukova od 20Hz do 20KHz (20 000 Hz). Stoga zvuk iznad 20 KHz zove se ultrazvuk.

Ultrazvučni senzor mjeri vrijeme odbijanja zvuka, tj. šalje impuls u objekt i mjeri vrijeme odbijanja impulsa (ultrazvuka) i time izračunava udaljenost od objekta. Dakle ultrazvučni senzor proizvodi kratki zvučni signal (impuls) i šalje ga ispred sebe, zvuk se odbija od najbliže prepreke i vraća se natrag do senzora koji ga pokupi. [11]



Slika 19: Princip rada ultrazvučnog senzora.

Brzina zvuka je $= 343\ m/s$. Pretvaranjem u željene mjerne jedinice dobije se

$$v = 0.0343\ cm/\mu s.$$

Vrijeme putovanja zvuka mora se podijeliti s 2 zbog toga jer zvuk putuje od senzora do objekta i nazad, te prevaljuje duplu udaljenost. Formula po kojoj senzor mjeri udaljenost u centimetrima glasi:

$$Udaljenost = vrijeme \cdot ½ \cdot 0.0343$$

Ultrazvučni senzor HC-SR04 ima četiri pina: Vcc – napajanje, Trig – odašiljač ultrazvuka, Echo – prijamnik ultrazvuka, GND – uzemljenje. Napajanje i uzemljenje spajaju se na 5V pin i na pin za uzemljenje na arduino pločici. Trig i Echo pinovi spajaju se na bilo koji digitalni I/O (INPUT / OUTOUT) pin na Arduino pločici.

Infracrveni senzor također možemo koristiti za detekciju objekata, tj. zapreka. To je elektronička uređaj koji može prepoznavati svoju okolinu emitiranjem i detektiranjem infracrvenog zračenja. Može se koristiti u različite svrhe kao što su npr. komunikacija, mjerenje kuta ili udaljenosti, detekcija objekata. Mala cijena i točnost mjerenje čine ih vrlo pogodnima za mjerenje udaljenosti. Senzor se sastoji od infracrvene LED diode i fotodiode. LED dioda emitira infracrveno zračenje koje nije vidljivo ljudskom oku, a fotodioda detektira to zračenje.

Svaki put kad se ispred IR senzora pojavi prepreka tada se emitirana infracrvena svjetlost odbija od toga objekta nazad u fotodiodu. Fotodioda se tada aktivira i zbog detektirane svjetlosti šalje signal kroz output u arduino pločicu. Dok ispred IR senzora nema prepreke infracrvena svjetlost ne dolazi do fotodiode. IR senzor ima pinove kao što su: Vcc (napajanje), GND (uzemljenje), OUT (izlazni signal koji se šalje putem fotodiode). Vcc se spaja na 5V pin na arduino pločici, GND na označeni GND pin, te OUT koji se spaja na bilo koji digitalni pin na arduino pločici. Izlaz (OUT) IR senzora šalje digitalni oblik signala na arduino pločicu koja ga obrađuje. Ovaj senzor ima na sebi potenciometar preko kojega se okretanjem podešava udaljenost na kojoj senzor može detektirati zapreku, tj. objekt. [12]

3.3. Programsko okruženje

IDE je open-source software i radi na svim operativnim sustavima kao što su Windows, Linux i Mac. Programiranje u softweru je neka mješavina C++-a i C-a. Nakon spajanja hardware-a poznati su pinovi koji se koriste, te je sve spremno za pisanje programa, tj. koda. Preko software-a i napisanog koda upravlja se svim komponentama koje se koriste. [13]

Slika 20: Primjer programiranja Arduina [11].

Na slici 20 je primjer programa blinkanja LED-ice. Na početku programa definira se digitalni pin 13 na arduino pločici kao konstanta, što znači da je ta vrijednost nepromjenjiva dalje u programu. LED-ica je spojena na pin 13. Void setup je funkcija kroz koju softver prođe samo jedanput nakon svakog pokretanja ili resetiranja arduino pločice kako bi se definirali ulazi i izlazi, pokrenule varijable itd. Void loop je funkcija koja program vrti u krug do beskonačnosti. Kao što je prikazano na slici 20, u setup-u je preko PinMode

funkcije pin 13 na arduino pločici definiran kao izlaz, tj. output. Nakon definiranja varijabli i ulaza i izlaza potrebno je napisati glavni dio programa koji se nalazi u funkciji void loop. Preko funkcije digitalWrite zadaje se koji će se pin na arduino pločici uključiti (HIGH) ili isključiti (LOW). Postoji i funkcija analogWrite koja služi za upravljanje analognim pinovima pločice, no u ovome primjeru se koriste digitalni pa se zbog toga koristi digitalWrite. Funkcija delay služi za vremensko ograničavanje duljine trajanja radnje. Broj koji se nalazi u oblim zagradama funkcije delay izražen je u milisekundama. Nakon pojašnjenja programa može se uočiti da će se LED-ica upaliti i biti upaljena jednu sekundu, te će se nakon toga ugasiti i biti ugašena jednu sekundu. Ova radnja će se stalno vrtjeti u krug što će rezultirati konstantnim blinkanjem LED-ice.

Pojašnjenjem ovog primjera može se predočiti za što služi arduino softver i kako se koristi. Treba napomenuti da su ovo osnove programiranja u arduinu. Softver nudi razne mogućnosti. Otvaranjem službene Arduino stranice na internetu mogu se vidjeti razne funkcije i mogućnosti koje ovaj softver nudi.

4. UPRAVLJAČKI MODUL ZA IZBJEGAVANJE PREPREKA

4.1. Testiranje pogonskog sustava

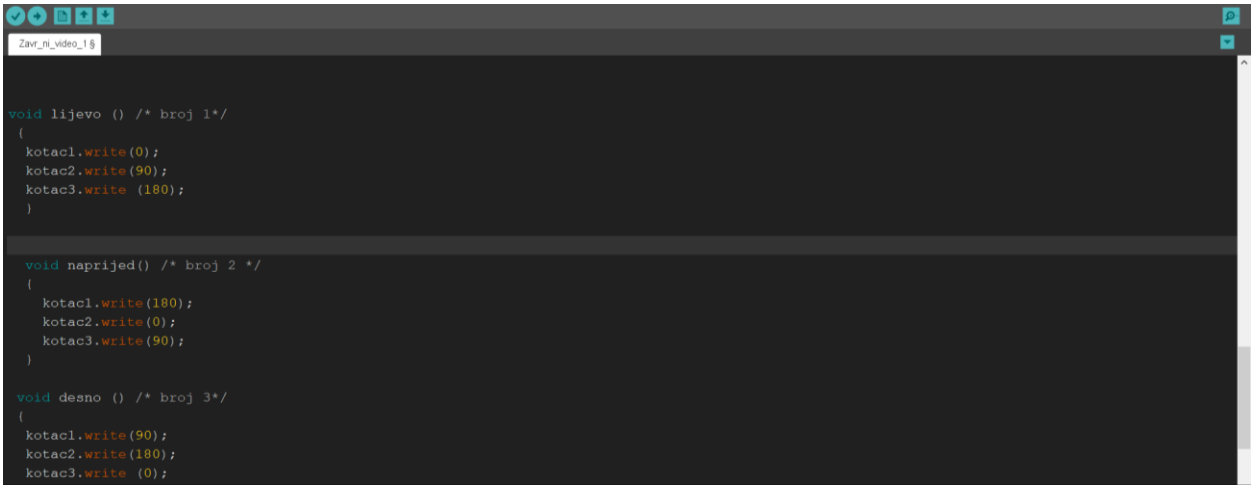
Kako bi se robot gibao potrebno mu je definirati osnovna gibanja u programskom okruženju, tj. Arduinu. Prije samoga programiranja potrebno je odrediti smjerove na samome robotu, tj. definirati koji je smjer naprijed, koji nazad itd. Taj problem riješen je označavanjem smjerova, a osnovni su označeni sljedećim redoslijedom:

- lijevo – broj 1
- naprijed – broj 2
- desno – broj 3

Za početak je potrebno u programu pozvati servo library kako bi se uopće moglo upravljati servo motorima i ostvariti određeno gibanje. Sljedeći korak je kreirati tri servo motora i definirati njihove pinove na arduino pločici.

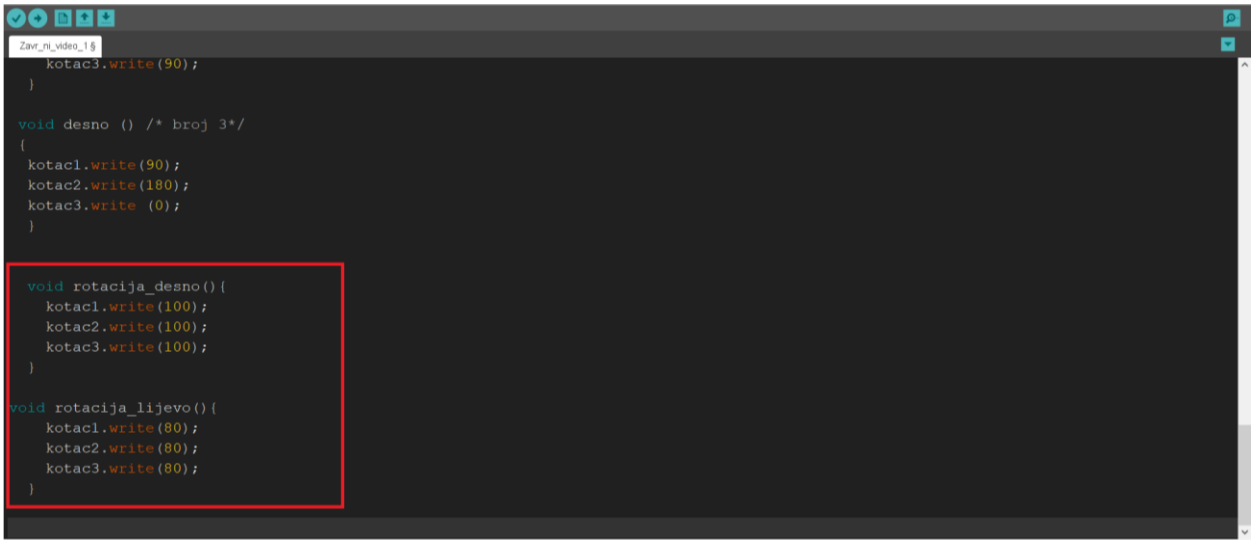
Slika 21: Definiranje servo motora te njihovog spajanja

Gibanje u smjeru natrag suprotno je od gibanja broj 1. Rotacija desno gibanje je u smjeru kazaljke na satu u kojem sudjeluju svi motori, dok je rotacija lijevo suprotno gibanje. U programu je svako gibanje kreirano kao funkcija koja se poziva u glavnome programu. Na taj način u glavnome programu izbjegavaju se bespotrebne linije koda.



Slika 22: Definiranje funkcija za glavne smjerove gibanja.

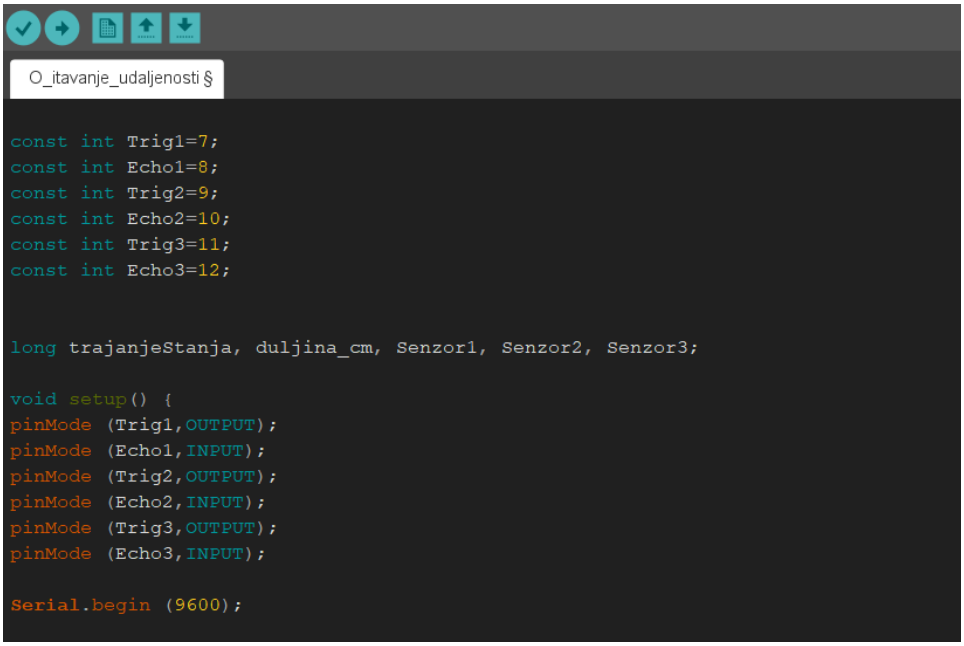
U programu su definirana tri glavna gibanja i dva pomoćna gibanja. Tri glavna gibanja prikazana su na slici 22. To su gibanja u kojima sudjeluju po dva servo motora dok treći miruje. Kod pomoćnih gibanja imamo rotaciju lijevo i rotaciju desno gdje sudjeluju po tri servo motora. Pomoćna gibanja prikazana su na slici 23.



Slika 23: Definiranje funkcija za pomoćne smjerove gibanja.

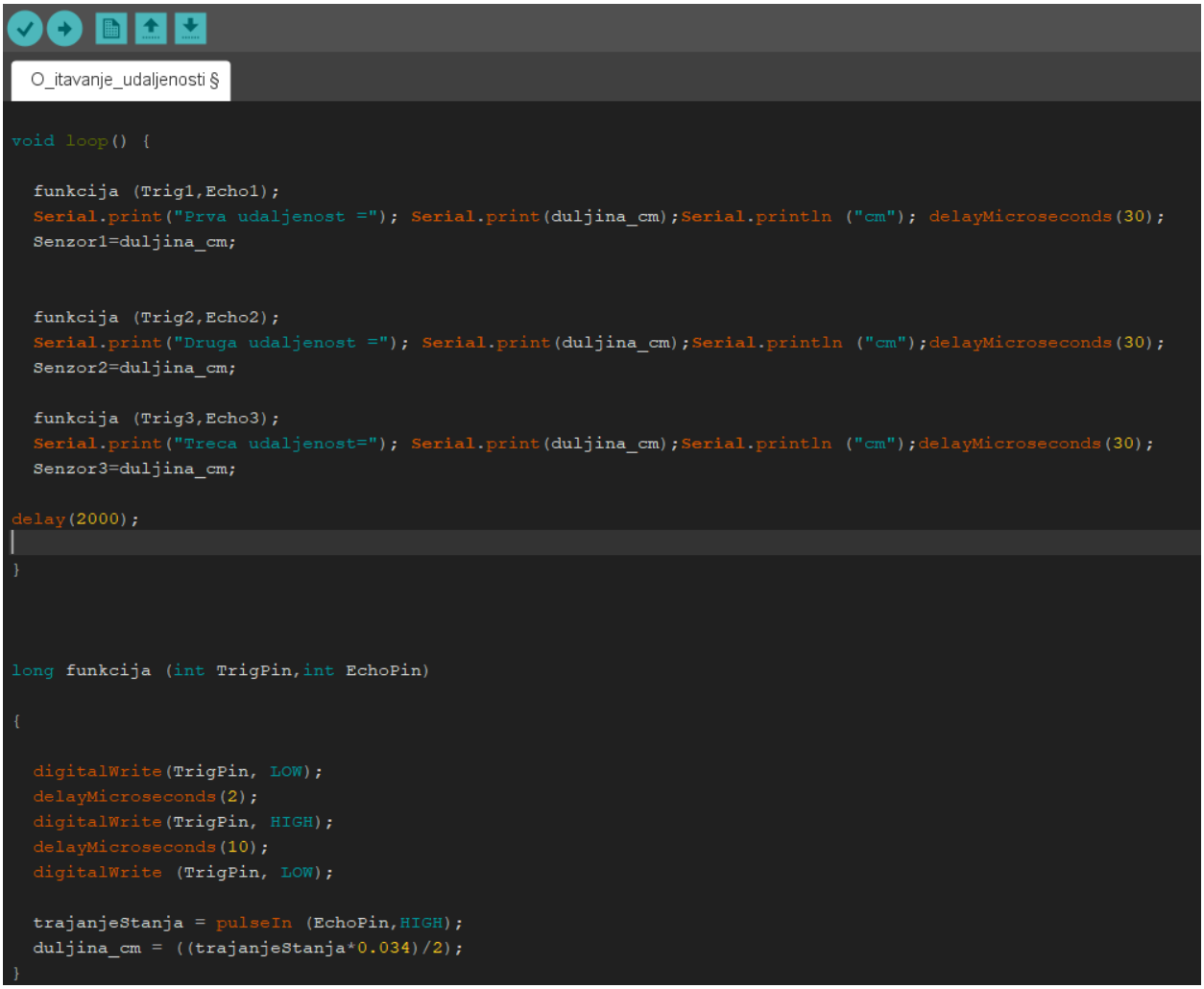
4.2. Testiranje senzorskog sustava

Robot koristi tri HC-SR04 ultrazvučna senzora za mjerenje udaljenosti. Senzori su razmaknuti 120° što čini 360° ukupno. U programu je potrebno definirati pinove na koje je svaki senzor priključen. Nakon toga potrebno je odrediti pinove koji se ponašaju kao OUTPUT i pinove koji se koriste kao INPUT.



Slika 24: Definiranje Echo i Trig pinova, mjesta spajanja, pinModa.

Prva verzija programa za očitavanje udaljenosti napravljena je tako da je kreirana funkcija za izračunavanje udaljenosti u samome programu. Ta funkcija daje vrijednost udaljenosti u centimetrima. Funkcija radi na principu da TrigPin šalje ultrazvučni signal prema objektu i mjeri vrijeme potrebno da se signal vrati prema EchoPin-u. Funkcija se poziva u glavnome programu, tj. loop(), gdje se uvrštava svaki od tri senzora u funkciju. Na kraju se dobije ispis očitanja udaljenosti svakog senzora u Serial Monitor-u.



Slika 25: Prva verzija programa za očitanje udaljenosti.

Druga verzija programa za očitanje udaljenosti koristi NewPing library. Taj library pomaže kod izbjegavanja nepotrebnih linija koda. Naime, pohranjuje senzore u array. U array-u za svaki senzor pohranjujemo tri stvari. Primjer: NewPing(7,8,400), gdje je TrigPin broj 7, EchoPin broj 8 i 400 označava maksimalnu udaljenost u centimetrima. Kako bi se došlo do očitane vrijednosti određenog senzora, senzor pozivamo iz array-a pomoću njegovog indeksa u array-u, npr. prvi senzor ima indeks 0, itd. Program možemo vidjeti na slici 26.

Očitane vrijednosti mogu se vidjeti pomoću Serial Monitor-a. Obje verzije programa za očitanje udaljenosti daju jednak rezultat u Serial Monitor-u.

sketch_jul07b §

```
#include <NewPing.h>
NewPing senzor [3] = {
  NewPing (7,8,400),
  NewPing (9,10,400),
  NewPing (11,12,400)
};
void setup() {
  Serial.begin(9600);
}
void loop() {
  Serial.print("Prva udaljenost ="); Serial.print(senzor [0].ping_cm());Serial.println ("cm"); delayMicroseconds(30);


 Serial.print("Druga udaljenost ="); Serial.print(senzor [1].ping_cm());Serial.println ("cm"); delayMicroseconds(30);


  Serial.print("Treca udaljenost ="); Serial.print(senzor [2].ping_cm());Serial.println ("cm"); delayMicroseconds(30);

  delay(2000);
}
```

Slika 26: Druga verzija programa za očitanje udaljenosti.

COM5

```
Prva udaljenost =120cm
Druga udaljenost =331cm
Treca udaljenost=96cm
Prva udaljenost =119cm
Druga udaljenost =331cm
Treca udaljenost=95cm
Prva udaljenost =120cm
Druga udaljenost =331cm
Treca udaljenost=96cm
Prva udaljenost =119cm
Druga udaljenost =331cm
Treca udaljenost=96cm
Prva udaljenost =119cm
Druga udaljenost =331cm
Treca udaljenost=96cm
Prva udaljenost =120cm
Druga udaljenost =330cm
Treca udaljenost=97cm
```

Autoskrol Show timestamp

Slika 27: Očitavanje udaljenosti.

4.3. Modul za izbjegavanje prepreka

Ovaj modul sastoji se od pogonskog i senzorskog sustava koji međusobno komuniciraju po napisanom programu za izbjegavanje prepreka i tako odrađuju zadatak izbjegavanja prepreka. Napisane su dvije verzije programa za izbjegavanje prepreka. Prva verzija izbjegava prepreke na način da odmah promijeni smjer, ta verzija programa nalazi se na slici 28 i 29.

Slika 28: Program za izbjegavanje prepreka - prva verzija (prvi dio).

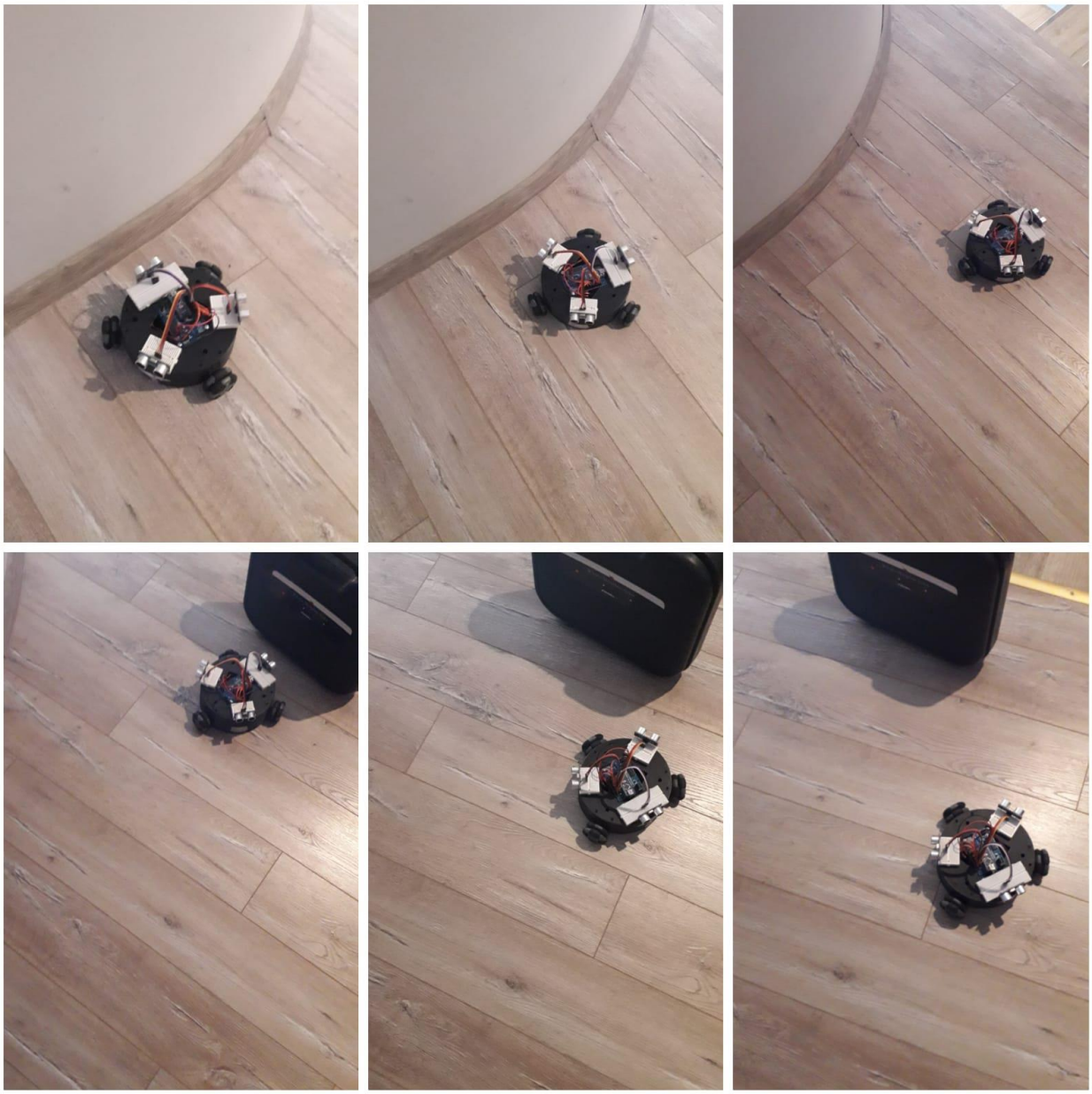
```
else if(senzor [2].ping_cm() <= granica_min && senzor [2].ping_cm() > 0){
  do{
   lijevo();   }
while(senzor[0].ping_cm() >= granica_min && senzor [0].ping_cm()<600);
  }

}

void lijevo () /* broj 1*/
 {
  kotac1.write(0);
  kotac2.write(90);
  kotac3.write (180);
  }



  void naprijed() /* broj 2 */
  {
    kotac1.write(180);
    kotac2.write(0);
    kotac3.write(90);
  }

 void desno () /* broj 3*/
 {
  kotac1.write(90);
  kotac2.write(180);
  kotac3.write (0);
  }
```

Slika 29: Program za izbjegavanje prepreka - prva verzija (drugi dio).

Druga verzija programa za izbjegavanje prepreka koristi jedan senzor kao glavni senzor, a ostale senzore kao pomoćne. Kada glavni senzor dođe do prepreke robot se rotira tako da ga pomoćni senzori navigiraju. Svi senzori izbjegavaju prepreke ali glavni senzor ujedno služi i za gibanje robota prema naprijed.

Slika 30: Program za izbjegavanje prepreka - druga verzija (prvi dio).

Slika 31: Program za izbjegavanje prepreka - druga verzija (drugi dio).



Slika 32: Primjeri izbjegavanja prepreka svesmjernog robota

5. ZAKLJUČAK

Zadatak završnog rada bio je implementacija upravljačkog modula za izbjegavanje prepreka svesmjernim mobilnim robotom. Tijekom programiranja senzorskog sustava dolazi do problema kad senzor kao očitanu vrijednost daje vrijednost 0 što upućuje na preveliku udaljenost prepreke ispred senzora, a premaleni mjerni opseg senzora. Taj problem rješen je definiranjem tog uvjeta u određenoj if petlji.

Problem kod izbjegavanja prepreka dolazi kad senzor naiđe na neravnu površinu što znači da dolazi do nepravilnih odbijanja ultrazvuka od objekta, no ovaj problem djelomično se može rješiti implementacijom boljeg senzora.

Kod testiranja senzorskog i pogonskog sustava ima malih nedostataka koji bi se riješili nadogradnjom sustava. Nedostaci su nedovoljan broj senzora. Naime tri senzora ne pokrivaju svih 360° potrebna za kretanje robota, pa dolazi do sudaranja u prepreku kad prepreka nije u vidljivom području senzora. Taj nedostatak može se riješiti ugradnjom još tri senzora ili ugradnjom senzora s većim kutom otvora. Što se tiče vizualnog dojma kao poboljašanje treba spomenuti mogućnost ugradnje kučišta za bolji pregled žica i cijelog modula.

Literatura

[1] https://hr.wikipedia.org/wiki/Robotika

Datum pristupa: 12.06.2020.

[2] https://www.bug.hr/startup/autonomni-skladisni-roboti-gideon-brothersa-u-pilot-projektu-s-orbicom-11318

Datum pristupa: 12.06.2020.

[3] Osnove robotike: podloge za predavanje, VUKA

[4] https://www.robotshop.com/media/files/pdf/omni-wheel-introduction-10013.pdf

Datum pristupa: 18.06.2020.

[5] https://www.elecfreaks.com/mecanum-wheel-64mm-x-4.html

Datum pristupa: 25.06.2020.

[6] https://en.wikipedia.org/wiki/Omni_wheel

Datum pristupa: 26.06.2020.

[7] https://en.wikipedia.org/wiki/Mecanum_wheel

Datum pristupa: 26.06.2020.

[8] http://www.gimnazija-velika-gorica.skole.hr/upload/gimnazija-velika-gorica/images/static3/1003/File/03%20Programiranje%20manevara.pdf

Datum pristupa: 29.06.2020.

[9] https://servodatabase.com/servo/feetech/fs5103r

Datum pristupa: 03.07.2020.

[10] http://plotdiagram.francescachillemi.it/diagram/wiring-diagram-arduino

Datum pristupa: 05.07.2020.

[11] https://e-radionica.com/hr/blog/2015/08/19/kkm-ultrazvucni-modul-hc-sr04/

Datum pristupa: 05.07.2020.

[12] https://www.instructables.com/id/Using-Infrared-Sensor-With-Arduino/

Datum pristupa: 08.07.2020.

[13] https://www.arduino.cc/

Datum pristupa: 08.07.2020.

Prilozi

1. CD-R