Korištenje senzora za praćenje linije u nastavi programiranja

Petričević, Luka

Master's thesis / Diplomski rad

2022

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Zagreb, Faculty of Science / Sveučilište u Zagrebu, Prirodoslovno-matematički fakultet**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:217:438044

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-03-29**

Repository / Repozitorij:

Repository of the Faculty of Science - University of Zagreb

SVEUČILIŠTE U ZAGREBU
PRIRODOSLOVNO-MATEMATIČKI FAKULTET
MATEMATIČKI ODSJEK

Luka Petričević

KORIŠTENJE SENZORA ZA PRAĆENJE LINIJE U NASTAVI PROGRAMIRANJA

Diplomski rad

Voditelj rada: v. pred. dr. sc. Goran Igaly

Zagreb, 2022.

Ovaj diplomski rad obranjen je dana _____ pred ispitnim povjerenstvom u sastavu:

1. _____, predsjednik
2. _____, član
3. _____, član

Povjerenstvo je rad ocijenilo ocjenom _____.

Potpisi članova povjerenstva:

1. _____
2. _____
3. _____

Zahvaljujem svom mentoru v. pred. dr. sc. Goranu Igalyju na korisnim savjetima, puno razumijevanja i ogromnom strpljenju tijekom izrade ovog diplomskog rada. Hvala svim prijateljima koje sam upoznao na ovom obrazovnom putu. koji su me uveseljavali i motivirali da idem k cilju. Posebno hvala cimerima koji su me trpjeli u napornim satima učenja i mučenja, Sabo majstore hvala ti! Ipak. najveća podrška i vjera u mene i moje mogućnosti dolazila je od šire i uže obitelji te obitelji Znaor, kojima se ovim putem neizmjerno zahvaljujem. Na kraju, moram spomenuti najbitniju osobu u svim danima moga studiranja, djevojku Anu, koja je puna razumijevanja svih ovih godina, bila moj oslonac u dobrim i manje dobrim trenucima. Hvala ti do neba!

Sadržaj

1. Uvod **1**

2. Senzori **2**

2.1. Povijest senzora: 2

2.2. Općenito o senzorima i vrste senzora 4

2.3. Upotreba senzora u nastavi 9

3. Robot mBot **11**

4. Senzori za praćenje linije **19**

4.1. Line Follower 19

4.2. RGB Line Follower 25

5. Provjera učinkovitosti senzora **32**

6. Kalibracija **40**

7. Popis preuzetih slika **42**

8. Bibliografija **44**

9. Sažetak **45**

10. Summary **46**

11. Životopis **47**

1. Uvod

Cilj ovog rada je istražiti mogućnosti korištenja senzora za praćenje linija u nastavi programiranja na način da će se izložiti i analizirati različiti algoritmi za praćenje jednostavne linije. Pri tome će se koristiti model robota mBot te dva senzora, Line Follower i RGB Line Follower. Analizirat će se prednosti i poteškoće pojedinih rješenja s posebnim naglaskom na mogućnostima korištenja ovih senzora u nastavi programiranja u školama. Prije predstavljanja robota mBot i dvaju senzora u drugom poglavlju, osvrnuti ćemo se na povijest senzora i njihove specifikacije. Nakon vrsta senzora spomenuti ćemo potrebu za upotrebom u nastavi informatike koja se sve više modernizira i traži načine usvajanja praktičnih vještina učenika. U trećem i četvrtom poglavlju pobliže ćemo upoznati robot mBot te senzore koje skupa s njim u ostatku rada, kroz razne primjere i situacije, proučavamo i bilježimo zapažanja dobrih i manje dobrih strana njihove upotrebe.

2. Senzori

2.1. Povijest senzora:

Iako ne možemo odrediti točno vrijeme i izumitelja prvog senzora, možemo proći kroz povijest pojavljivanja različitih tipova senzora. Namjena prvog modernog senzora koji je korišten u svrhu zaštite protiv provalnika u obliku alarma pojavio se sredinom prošlog stoljeća, kada ga je izumio Samuel Bagno. Za vrijeme Drugog svjetskog rata povećanjem potražnje i razvojem tehnologije za nadzor i praćenje unapređuje se upotreba senzora za pokret. Kao primjer možemo navesti radar, izumljen godinama ranije, koji se usavršava u svrhu ostvarivanja prednosti nad neprijateljskim zračnim snagama. Unatoč poboljšanju, tehnologija je i dalje bila prikladna samo za vojne svrhe. Sve do početka četrdesetih godina dvadesetog stoljeća, kada Samuel Bagno koristeći znanje o radarima i tek razvijenih elektroničkih komponenata, pokreće projekt istraživanja o ultrazvučnim alarmima, po prvi put bez vojne namjene. Njegov izum funkcionirao je tako što se iz modema širom prostorije slala mreža ultrazvučnih valova, koja se zatim vraćala nazad u modem odbijanjem od statičkih objekata. Tada, u slučaju pokreta, dolazi do interferencije valova što modem detektira čime se pokreće alarm. Početkom 21. stoljeća upotreba senzora je ostvarila ogroman napredak te se počinje primjenjivati i u svijetu video igrica.

Slika 1. Samuel Bagno (lijevo na slici)

Tako je, 2006. godine Nintendo wii, kombinirano primao kontrole od joysticka i daljinskog upravljača, koja su oba sadržavala infracrveno orijentiranje ovisno o svjetlosti te senzore ubrzanja koje je emitirala igraća konzola. Pored navedenog primjera, možemo spomenuti i upotrebu senzora radi detekcije slobodnih mjesta na parkingu, pregleda okolnog prostora automobila (Tesla S) što vozačima pomaže u sprječavanju prelaska u drugi trak te izazivanja prometnih nesreća. Iako je Andrew Ura patentirao prvotni oblika termostata 1830. godine, napredniji oblik jednostavnih termostata izumio je 1885. godine Švicarac Albert Butz. Temperatura je kontrolirana automatskom polugom za otvaranje i zatvaranje vrata peći. Nakon osnutka „Minneapolis Heat Regulator Company" doživio je uspjeh što možemo vidjeti po tome da je tvrtka jedna od najpoznatijih proizvođača termostata kroz dosadašnju povijest. Prvi senzor kisika (*lambda* sonde) izumila je 1970. godine tvrtka Robert Bosch, koristeći primarne sonde od cirkonija u obliku cilindra. Zbog problema sa zagrijavanjem auspuha, osamdesetih godina se dodaje grijač u središte keramičkog senzora kisika, što je dovelo i do smanjenja opasnosti od pregrijavanja. U SAD-u se devedesetih godina prošlog stoljeća ugrađuju „On-Board Diagnostics Generation Two" kontrole pokrenuvši razvoj te postavljanje novih, preciznijih senzora s mogućnošću povrata većeg broja informacija. Dodatan razvoj doveo je do današnjih automobila u kojima je postavljen senzor kisika kao jedan od ukupno desetak senzora. Posljednji primjer razvitka senzora koji ćemo spomenuti je senzor za digitalne fotografije. Ovi senzori započinju svoj razvitak u periodu svemirske utrke između SAD-a i SSSR-a nakon što su George Smith i Willard Boyle razvili i demonstrirali CCD (*charge-coupled-device*) senzor koji radi putem pretvaranja svjetlosti u električne signale. Za razliku od tadašnjih 100x100 piksela, današnji CCD senzori imaju čak dvanaest do četrnaest megapiksela. 1975. godine napravljen je prototip digitalne kamere koja je bila veličine tostera, težila tri kilograma te koristeći CCD senzor snimala crno bijele slike koje su zatim spremane na kazete. U usporedbi s današnjim mogućnostima, slika dobivena ovom kamerom bila je razlučivosti 0.01 megapiksela, a za snimanje slike na kazetu bile su potrebne 23 sekunde. Iste godine izumljen je filtar boja koji je omogućio CCD senzorima snimanje u boji. Skoro pedeset godina poslije, ljudi su u svakodnevnom doticaju s digitalnim kamerama koristeći mobilne uređaje, posebice pametne telefone koji sadrže prednju i stražnju kameru.

Slika 2. George Smith i Willard Boyle

2.2. Općenito o senzorima i vrste senzora

Iako nailazimo na razne definicije o tome što je senzor ili pretvornik, najjednostavnije bismo ga mogli definirati kao ulazni uređaj koji proizvodi izlazni signal u odnosu na određenu fizikalnu veličinu (npr. tlak, svjetlost, temperaturu, vlažnost zraka). Senzor spada u ulazne uređaje jer je dio većeg sustava koji daje ulaz procesoru, odnosno glavnom upravljačkom sustavu. Zbog toga se može proširiti ova definicija tako da se senzor definira kao uređaj koji detektira promjene u svom okruženju te ih šalje u obliku informacija nekom drugom sustavu, primjerice procesu računala. Posljednju definiciju senzora koju ćemo spomenuti u ovom poglavlju ilustrirat ćemo na jednom primjeru. Definicija opisuje senzor

kao uređaj koji pretvara signale iz jedne energetske domene u električnu domenu. Jedan od najjednostavnijih primjera je LDR tj. otpornik ovisan o svjetlosti (*light dependent resistors*). Njegov otpor se mijenja ovisno o intenzitetu svjetlosti prostora u kojemu se nalazi. Ako je manje svjetlosti, tada otpor postaje visok, a što je svjetlosti više, otpor LRD-a postaje sve manji. Zbog mogućih problema zbog previše svjetlosti, što smeta mnogim senzorima u detekciji određenih fizičkih pojava, LDR se može spojiti u djelitelj napona te se napon tada kalibrira prema količini svjetlosti koja pada na svjetlosni senzor. Više o tome ćemo vidjeti u poglavlju u kojem će biti prikazan primjer kalibracije.

Slika 3. Senzori

Postoji mnogo različitih klasifikacija senzora, no najvažnija je podjela s obzirom na primjenu. Ostale podjele možemo vršiti prema:

- uvjetima rada,
- principu rada gabarita,
- vrsti izlaznog signala,
- prirodi izlazne veličine,
- prirodi mjerne veličine,
- načinu upotrebe,
- unutarnjoj strukturi, itd.

Sada ćemo vidjeti nekoliko klasifikacija:

Klasifikacija prema uvjetima rada:

- prenosivi: mogućnost prenošenja na razna mjesta
- stacionarni: fiksirani na mjestu

Klasifikacija prema prirodi ulazne veličine:

- toplinski: mjerenje temperature, toplinskog kapaciteta, entalpije, topline izgaranja,
- mehanički: mjerenje sile i momenta, tlaka, vakuuma, mehaničkog naprezanja,
- kinematički: mjerenje linearnog i kutnog ubrzanja i brzine, protoka,
- geometrijski: mjerenje položaja (koordinata) tijela razmjere, nivoa,
- radijacijski: mjerenje intenziteta toplinskog, nuklearnog, akustičnog i elektromagnetskog zračenja, boje, parametara valnog procesa,
- vremenski: mjerenje vremenskog perioda i frekvencije,
- električni: mjerenje elektromotorne sile, struje, otpora, induktiviteta, kapaciteta, provodljivosti,
- kemijski: mjerenje kemijskog sastava,
- fizikalni: mjerenje mase, gustoće, vlažnosti, tvrdoće, plastičnosti, hrapavosti.

Klasifikacija senzora prema vrsti izlaznog signala:

- analogni: imaju normiran dinamički opseg izlaznog signala i osjetljivost na šumove,
- digitalni: imaju veliku mogućnost interakcije sa mikroprocesorskim upravljačkim uređajima.

Klasifikacija prema prirodi izlazne veličine:

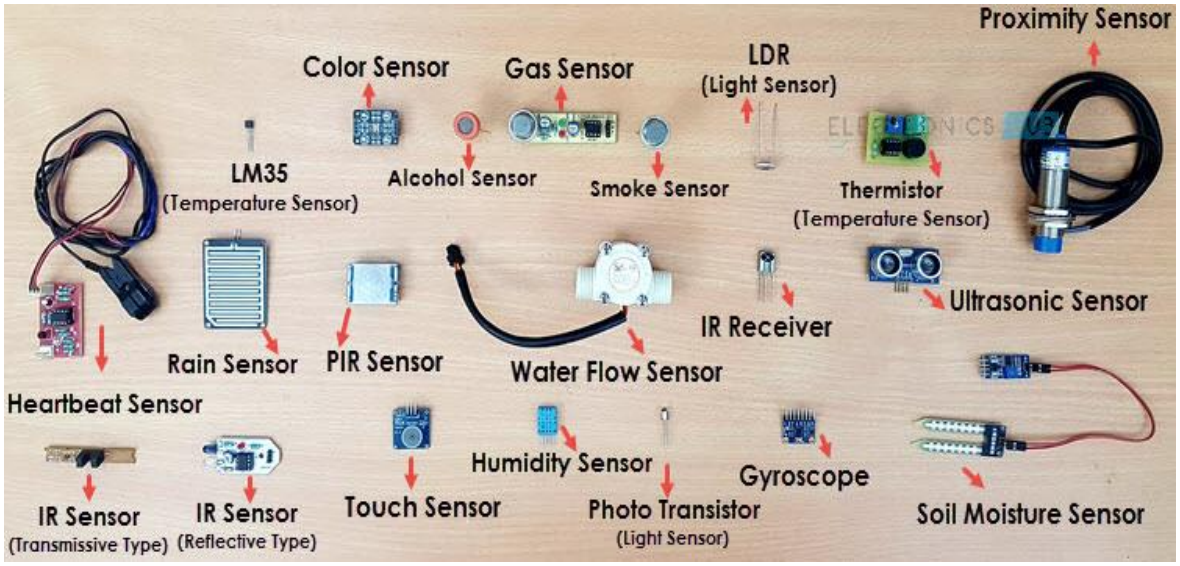
- mehanički: izlazni signal je sila ili pomak,
- vremenski: izlazni signal je vremenski ili frekvencijski kod,
- električni: izlazni signali mjerene veličine odražavaju se kroz promjenu R (*right*), L (*left*), C (*center*) parametara izlaza pa se zovu i parametarski.

Klasifikacija prema gabaritima:

- normalni: standardne izvedbe,
- malogabaritni: za primjenu u ograničenom prostoru,
- minijaturni: za specijalne namjene

Klasifikacija prema načinu upotrebe:

- operativni: za neposredno davanje informacija,
- pomoćni: za dobivanje pomoćnih informacija.



Slika 4. Primjeri nekih senzora

Popis različitih vrsta senzora s obzirom na primjenu (uz hrvatske nazive je i engleski prijevod zbog boljeg razumijevanja gornje slike)**:**

1. Senzor temperature (Temperature Sensor)
2. Senzor blizine (Proximity Sensor)
3. Brzinomjer (Accelerometer)
4. IR senzor (IR Sensor)
5. Senzor tlaka (Pressure Sensor)
6. Senzor svjetlosti (Light Sensor)
7. Ultrazvučni senzor (Ultrasonic Sensor)
8. Senzor dima (Smoke Sensor)
9. Dodirni senzor (Touch Sensor)
10. Senzor boje (Color Sensor)
11. Senzor vlažnosti (Humidity Sensor)
12. Senzor položaja (Position Sensor)
13. Magnetski senzor (Magnetic Sensor)
14. Mikrofon (Sound Sensor)
15. Senzor nagiba (Tilt Sensor)
16. Senzor protoka i razine tekućine (Flow and Level Sensor)
17. PIR senzor (pasivni infracrveni) (PIR Sensor)
18. Senzor naprezanja i težine (Strain and Weight Sensor)

2.3. Upotreba senzora u nastavi

U novije se vrijeme nastava informatike koristi sve više suvremenim tehnikama kojima se modernizira poučavanje djece programiranju. Roboti postupno ulaze u redovnu nastavu u školama koje posjeduju odgovarajuću opremu. Za očekivati je da će u budućnosti u Hrvatskoj još veći broj škola imati mogućnost korištenja robota u nastavi informatike što bi pomoglo učenicima mlađe i starije dobi pri usvajanju praktičnih znanja iz programiranja. Otvorimo li kurikulum za nastavni predmet informatike možemo pronaći preporuku korištenja robota u različitim razredima te u raznim očekivanim ishodima učenja od kojih ću neke kasnije navesti. Senzor za praćenje linije koji se nadogradi na robota može koristiti npr. kod uvođenja naredbi grananja (naredba *if-then-else*), te petlje (*while* petlja). Između ostalog, naredbe grananja mogu se obraditi na način da, na temelju informacije dobivenih iz okoline putem senzora, određeni sustav može izvesti neku radnju poput uključivanja i isključivanja svjetla, otvaranja i zatvaranja ventila, uključivanja i isključivanja grijanja i hlađenja, kretanja po liniji itd. Tako ćemo i mi, koristeći edukacijski robot mBot te dva senzora kojima ćemo se baviti, pokušati izvesti određeni zadatak, tj. praćenje linije te steći znanje koje se praktično može primijeniti pri realizaciji autonomnih vozila gdje je jedan od temeljnih zadataka upravo to.

Ishodi iz kurikuluma u kojima se preporuča korištenje robota:

- **B.2.2** Nakon druge godine učenja predmeta Informatika u domeni Računalno razmišljanje i programiranje učenik stvara niz uputa u kojemu upotrebljava ponavljanje. (2. razred osnovne škole)
- **B.4.1** Nakon četvrte godine učenja predmeta Informatika u domeni Računalno razmišljanje i programiranje učenik stvara program korištenjem vizualnim okruženjem u kojemu se koristi slijedom, ponavljanjem, odlukom i ulaznim vrijednostima. (4. razred osnovne škole)
- **B.5.1** Nakon pete godine učenja predmeta Informatika u domeni Računalno razmišljanje i programiranje učenik se koristi programskim alatom za stvaranje

programa u kojemu se koristi ulaznim i izlaznim vrijednostima te ponavljanjem. (5. razred osnovne škole)

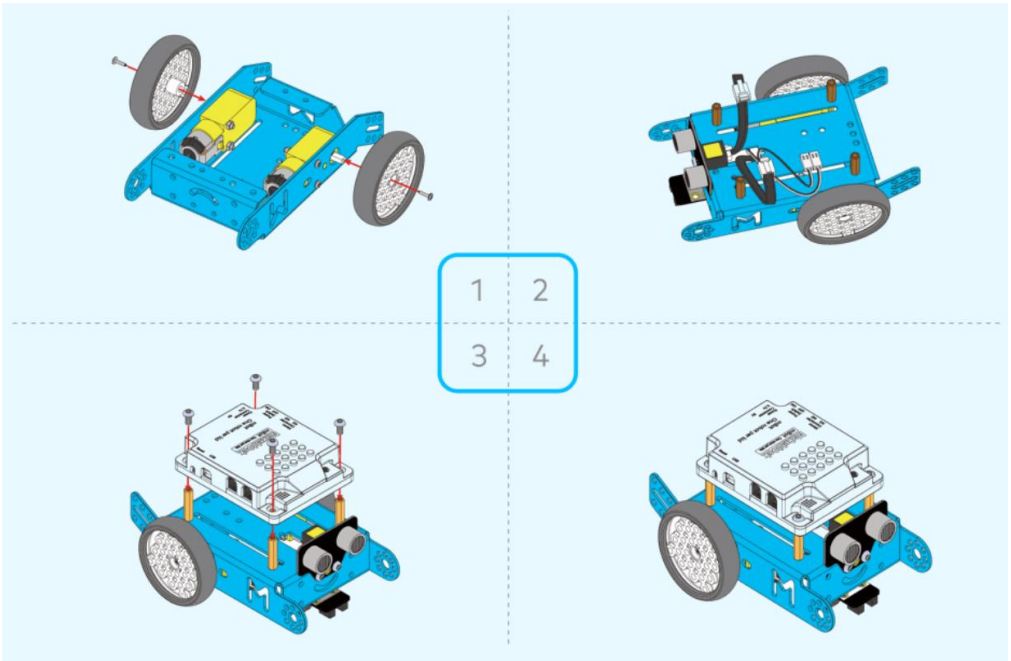
- **B.6.1** Nakon šeste godine učenja predmeta Informatika u domeni Računalno razmišljanje i programiranje učenik stvara, prati i preuređuje programe koji sadrže strukture grananja i uvjetnoga ponavljanja te predviđa ponašanje jednostavnih algoritama koji mogu biti prikazani dijagramom, riječima govornoga jezika ili programskim jezikom. (6. razred osnovne škole)
- **C.1.3** Nakon prve godine učenja predmeta Informatika u srednjoj školi u domeni Digitalna pismenost i komunikacija učenik u *online* okruženju surađuje i radi na projektu. (1. razred srednje škole)

3. Robot mBot

mBot je STEM edukacijski robot koji omogućuje početnicima, najčešće djeci, jednostavan ulazak u svijet programiranja, elektronike i robotike. Upravljanje mBotom se može odvijati putem mobilnog uređaja ili računala koji se povezuje s robotom pomoću *Bluetootha*.

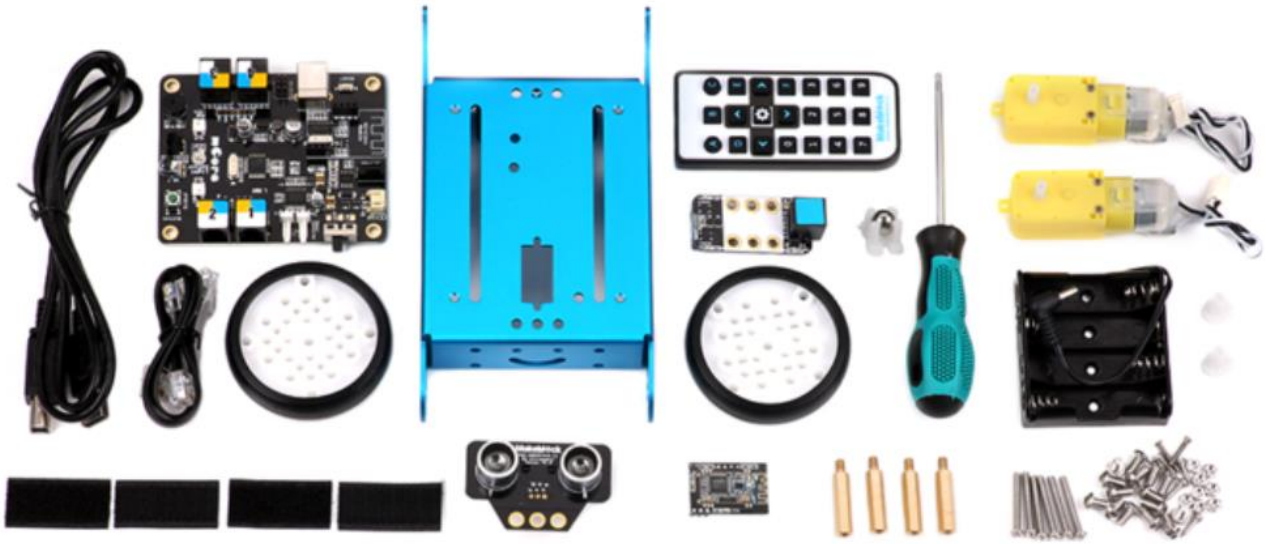
Slika 5. Prikaz mBota sa svih strana

Ovaj pokretni robot s kotačima jednostavan je za sastavljanje stoga ga djeca, služeći se uputama i odvijačem, mogu složiti u kratkom vremenskom roku, te se tako upoznati s određenim principima mehanike, elektronike i načinom funkcioniranja robota.



Slika 6. Prikaz 4 osnovna koraka u sastavljanju robota

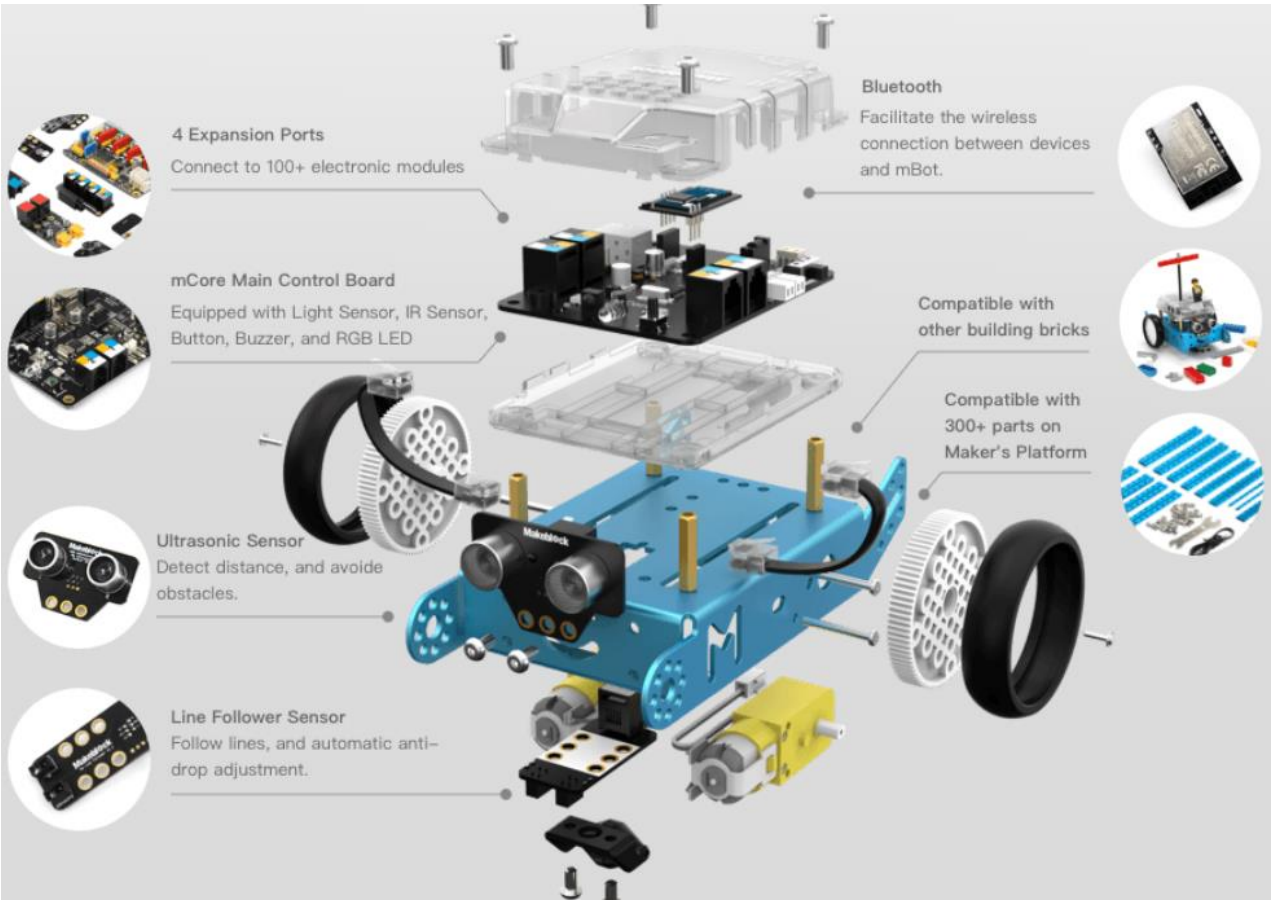
Robot se sastoji od 38 komponenti koji se moraju po uputama sastaviti kako bi mBot funkcionirao. Među njima se nalaze i obojeni žičani konektori RJ25 koji povezivanje čine jednostavnim. Slaganjem robota djeca stječu iskustvo rada s hardverom (proučavanje uputa, spajanje senzora i žica, pričvršćivanje dijelova).



Slika 7. Komponente nesloženog modela mBota

mBot ima četiri priključka za proširenje te se može spojiti na preko 100 vrsta elektroničkih modula. Postoji mogućnost korištenja raznih dodatnih paketa i više od 300 dijelova na platformi *Maker* za izgradnju raznih maštovitih oblika. Dodatna stvar koja omogućuje olakšano slaganje ovog robota je modularni dizajn. Njegovo sučelje matične ploče i elektronički moduli i označeni su odgovarajućim bojama kako bi ih se moglo što jednostavnije povezati. Osim korisnih informacija o osnovama robotskih strojeva i elektroničkim dijelovima stroja te iskustva koje djeca stječu slažući ovaj robot od najjednostavnijih komponenti fundamentalni zadatak korištenja i igranja s ovim robotom

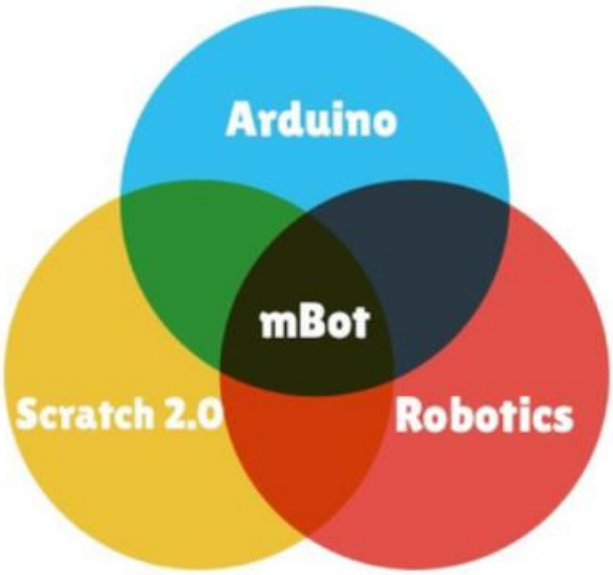
je upoznavanje s osnovama blokovskog programiranja i razvijanje logičkog razmišljanja koje će biti prijeko potrebno u budućem obrazovanju učenika, posebno onih koji će se nastaviti baviti programiranje, robotikom ili nekim srodnim područjem.



Slika 8. Neki od dijelova mBota

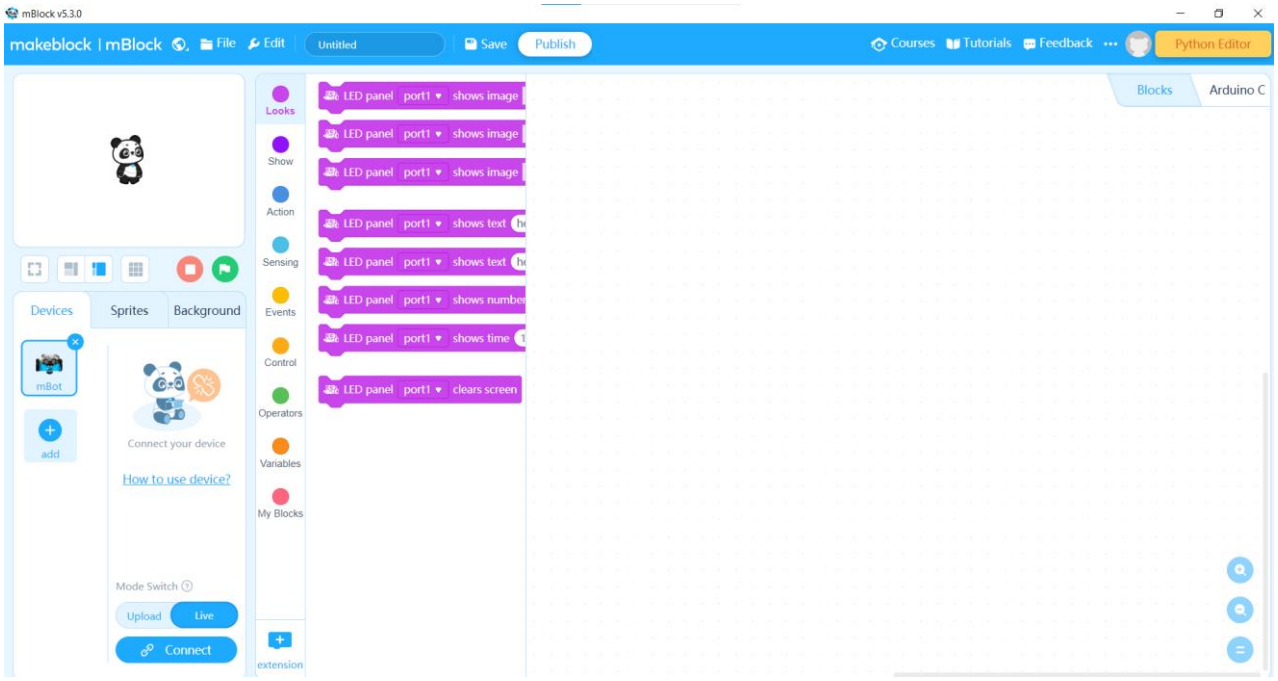
Platforma za programiranje robota mBot zove se mBlock i moguće ju je preuzeti na osobno računalo ili pametni uređaj. mBlock je grafički softver za programiranje u kojem se program izrađuje povlačenjem i spajanjem blokova koda. Upravo zato, riječ je o blokovskom programskom jeziku. Tvrtka MakeBlock razvila je mBlock, program temeljen na blokovskom vizualnom programskom jeziku Scratch 2.0, platformi otvorenog koda Arduinu i robotici. U program mBlock koji nasljeđuje sve karakteristike Scratcha,

nadograđene su komponente za programiranje robota mBot uz pomoć Arduina, platforme otvorenog koda koja se koristi za izradu projekata elektronike. Arduino koristi pojednostavljenu verziju programskog jezika C++ i olakšava djeci napredovanje u programiranju.

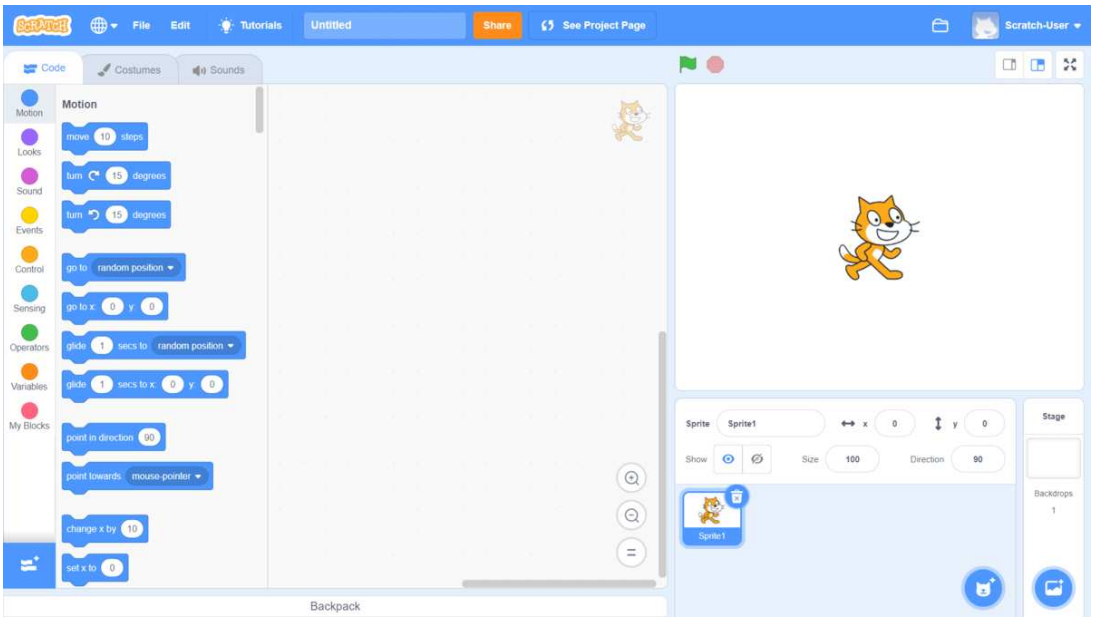


Slika 9. Nastanak mBlocka

Programirati se može kretanje robota, rad različitih motora, komunikacija među robotima, uključivanje i isključivanje svjetlećih dioda i zujalice, izvršavanje određenih radnji temeljem očitavanja senzora itd.

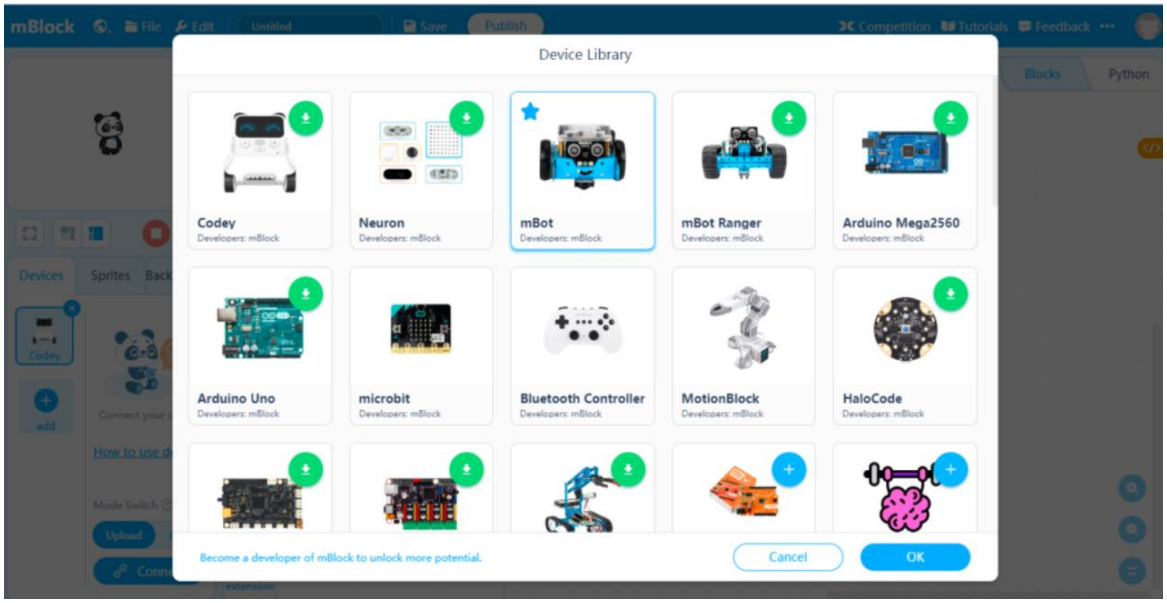


Slika 10. Program mBlock



Slika 11. Program Scratch

Program mBlock ima mogućnost povezivanja s raznim uređajima među kojima su Codey, Neuron, Arduino Uno, micro:bit i mnogi drugi. Za dodavanje određenog uređaja, potrebno je odabrati opciju „*Device*s“ klikom na znak plus. Nakon toga se otvara novi prozor „Device Library“ gdje se nalazi niz ponuđenih uređaja, među kojima je i mBot. Ukoliko se u desnom kutu ikone željenog uređaja nalazi zelena oznaka „Device update“ potrebno je pritiskom na tu oznaku ažurirati verziju uređaja prije dodavanja u program mBlock.

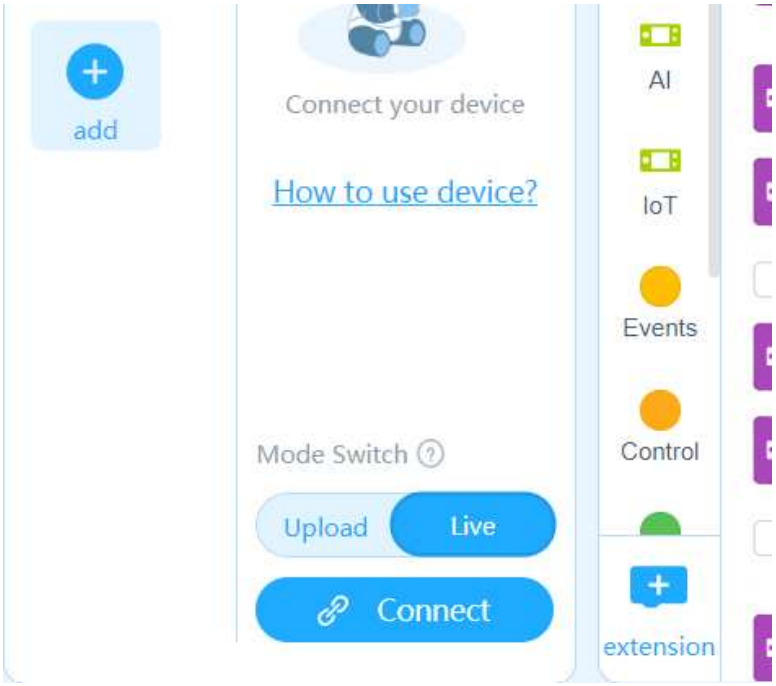


Slika 12. Odabir uređaja

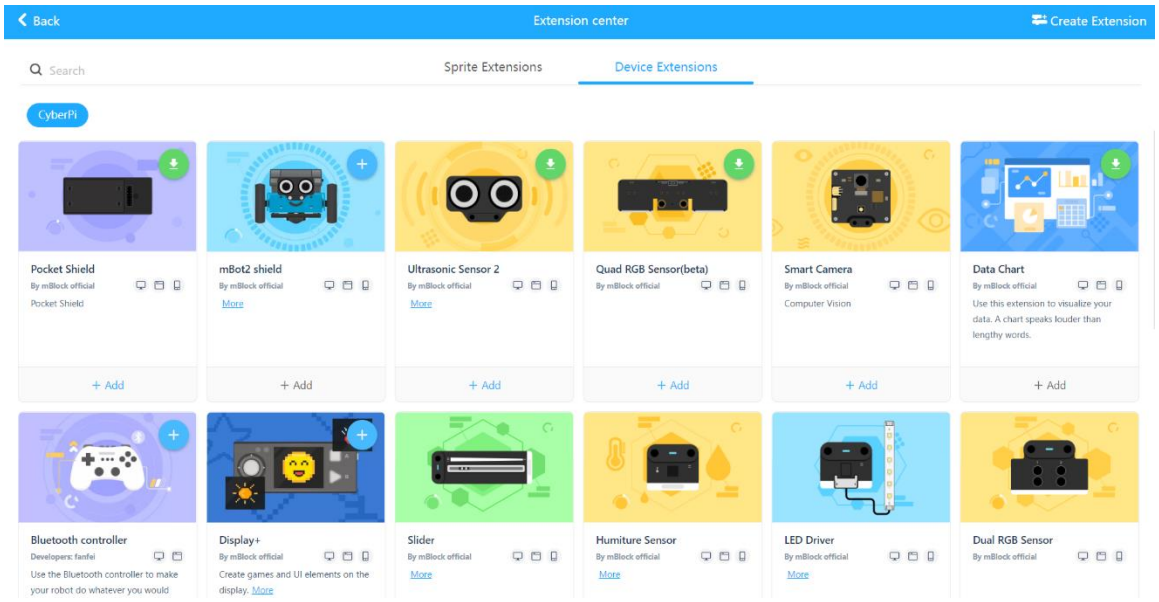
Povezivanje mBota s računalom odvija se putem USB kabela. Na robotu postoji sklopka za uključivanje i isključivanje robota koja mora biti namještena tako da je robot uključen kako bi mu se moglo uopće pristupiti preko računala ili mobilnog uređaja.

Kada je mBot povezan s računalom, u programu mBlock odabirom na opciju *Connect* i odabirom odgovarajućeg porta završavamo postupak povezivanja. Postoje dva načina rada robota – da radi kao samostalan robot (*Upload*) ili da radi uz pomoć računala (*Live*).

Ovisno o potrebi, u programu mBlock treba odabrati *Live* ili *Upload* način rada.



Slika 13. Opcije povezivanja i dva načina rada



Slika 14. Ekstenzije koje je moguće dodati u mBlocku

Product	mBot
Product properties	Entry-level educational robot kit
Assembly needed?	Yes
Main Control Board/ Chip	ATmega328
Sensors	Light Sensor, Button, IR Receiver, Ultrasonic Sensor, Line Follower Sensor
Other Programmable Modules	Buzzer, RGB LED ×2, IR Transmitter, Motor ×2
Connectivity	Bluetooth
Compatibility	Compatible with Makeblock DIY Platform and other building bricks
Add-on Pack	Six-legged Robot Add-on Pack, Interactive Light & Sound Add-on Pack, Servo Pack, Perception Gizmos Add-on Pack, Variety Gizmos Add-on Pack
Power Supply	3.7V Lithium Battery
Dimensions	190×130×130mm (L×W×H) assembled
Weight	500g

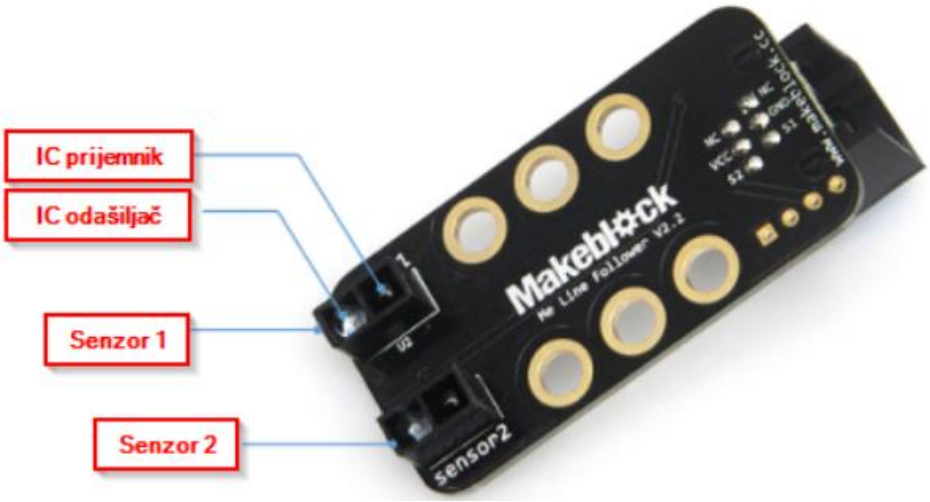
Slika 15. Specifikacije uređaja (mBot)

Na slici broj 15. vidimo specifikacije robota mBota među kojima su i senzori koji dolaze s njim. U nastavku diplomskog rada će fokus biti na dva senzora za praćenje linije, a to su Line Follower i RGB Line Follower čije ćemo načine rada - detaljnije opisati u idućem poglavlju.

4. Senzori za praćenje linije

4.1. Line Follower

Line follower, koji se nalazi na donjoj strani mBota, sastoji se od senzora 1 i senzora 2. Svaki od senzora ima infracrveni odašiljač i infracrveni prijemnik. Odašiljač radi tako da neprekidno emitira infracrvenu svjetlost, koja se reflektira kada robot prepozna bijelu ili neku drugu svijetlu površinu te tada prijemnik prima infracrveni signal i šalje upravljačkoj pločici vrijednost 1. Ako robot prepozna crnu ili tamnu površinu, infracrveno svjetlo se apsorbira odnosno ne može se odraziti i tada prijemnik ne prima infracrveni signal pa šalje vrijednost 0. Senzor ima raspon detekcije od 1 do 2 centimetra. Ako je mBot udaljen više od 2 centimetra od površine, infracrveno svjetlo se ne reflektira pa senzor šalje vrijednost 0 kao u slučaju kada robot detektira tamnu površinu.

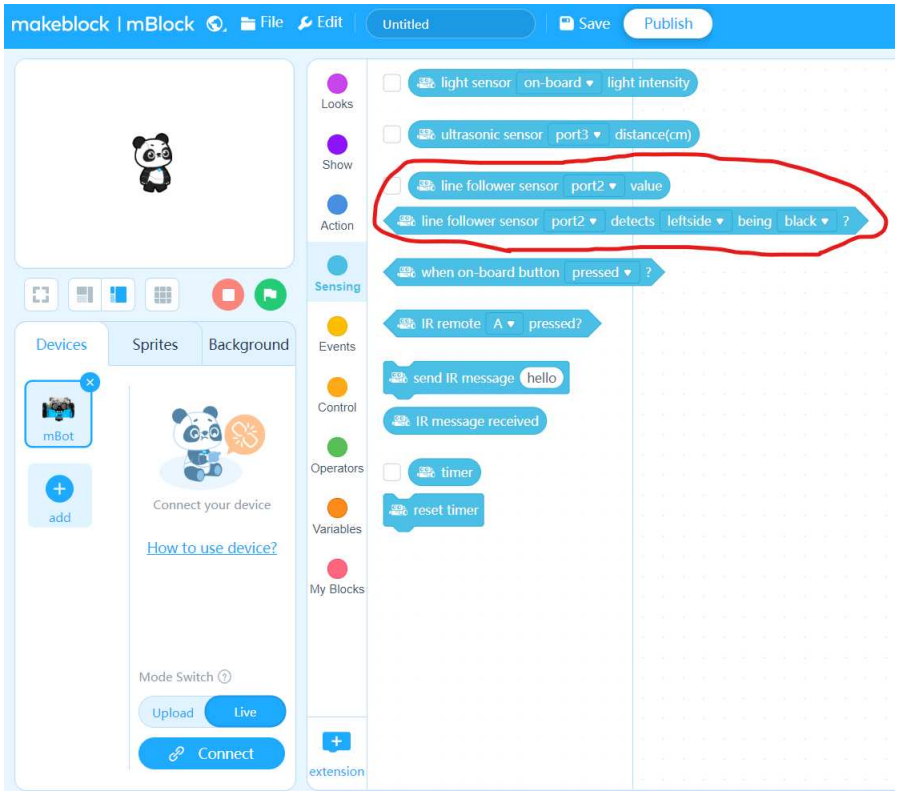


Slika 16. Line Follower

Svaki od senzora detektira nalazi li se na svijetloj (1) ili tamnoj površini (0), stoga su pretvaranjem binarnog u dekadski sustav četiri moguće vrijednosti koje senzor šalje upravljačkoj pločici: 0, 1, 2, 3. U tablici ispod prikazana su sve različite mogućnosti koje senzor može primiti.

Senzor 1	Senzor 2	Vrijednost senzora za praćenje linije
0	0	0
0	1	1
1	0	2
1	1	3

Naredbe za Line Follower nalaze se u mBlock programu pod kategorijom *Sensing*.

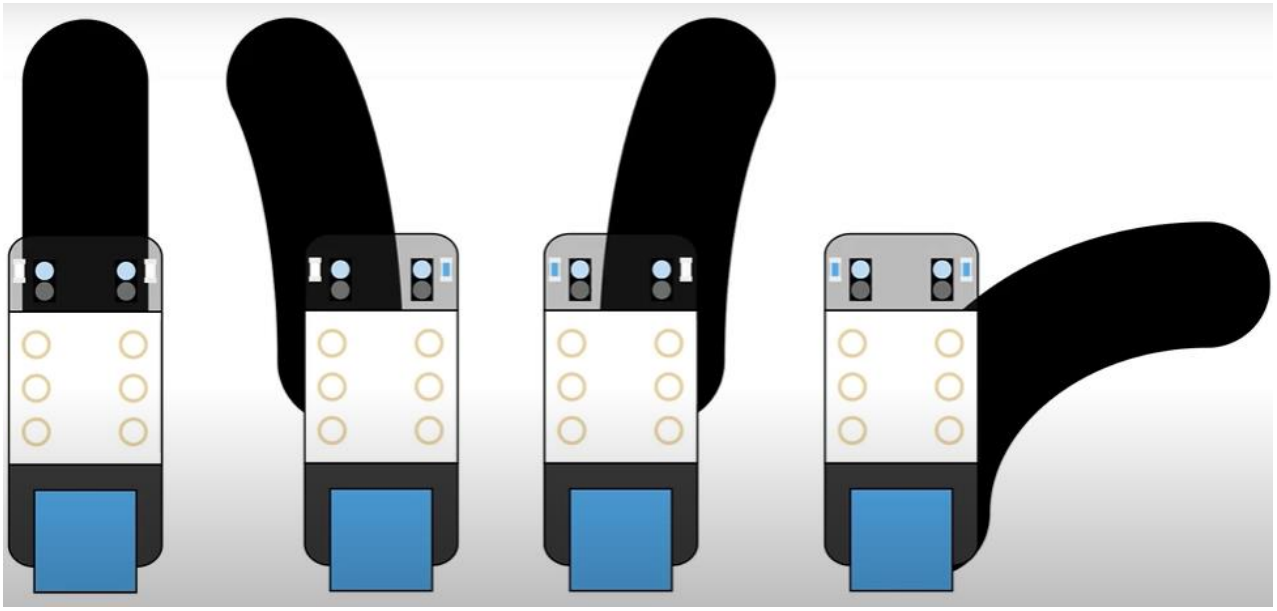


Slika 17. Naredbe za senzor zaokružene su crvenom bojom

Prvom naredbom u programu se provjerava očitanje pojedinog senzora za praćenje linije.

Druga naredba u programu javlja vrijednost senzora za praćenje linije (0, 1, 2 ili 3), ovisno o tome detektiraju li senzori tamnu ili svjetlu podlogu.

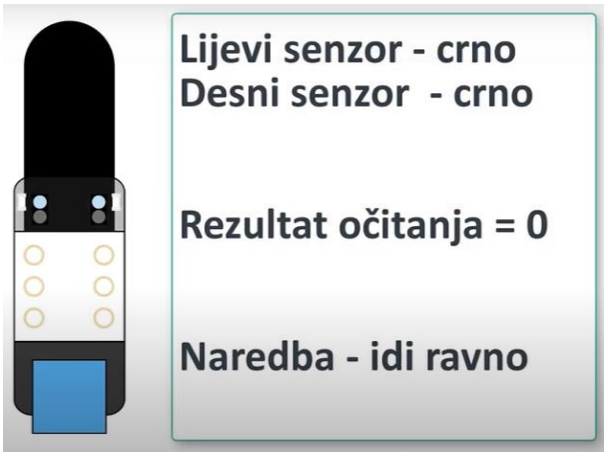
Prilikom unosa programa sa senzorom za praćenje linije treba prvo provjeriti na kojem ulazu (*portu*) je spojen senzor na upravljačkoj pločici te odabrati spomenuti *port* na naredbama koje su prikazane iznad.



Slika 18. Četiri različite mogućnosti kretanja robota

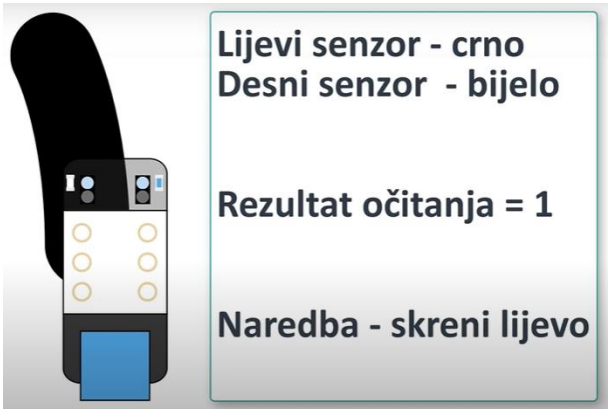
Na prethodnoj slici prikazane su već spomenute četiri različite mogućnosti koje senzori mogu detektirati dok se robot kreće po podlozi prateći liniju. Sada ćemo prvo opisno, zatim primjerom programskog koda u mBlocku prikazati ova četiri slučaja i objediniti ih u program praćenja linije koristeći Line Follower.

Robot kreće s vožnjom pritiskom na tipku (*wait until when on-bord button pressed*). Definirajmo četiri načina kretanja robota s obzirom na četiri moguća očitanja senzora podloge na kojoj se mBot može naći. To je razlog korištenja četiri *if then* naredbe grananja.



Slika 19. Idi ravno

U prvom načinu kretanja su oba senzora na tamnoj podlozi. Senzor očitava vrijednost 0 (*if line follower sensor port2 value = 0*) te se robot kreće ravno dalje po tamnoj crti (*move forward at power 40%*).

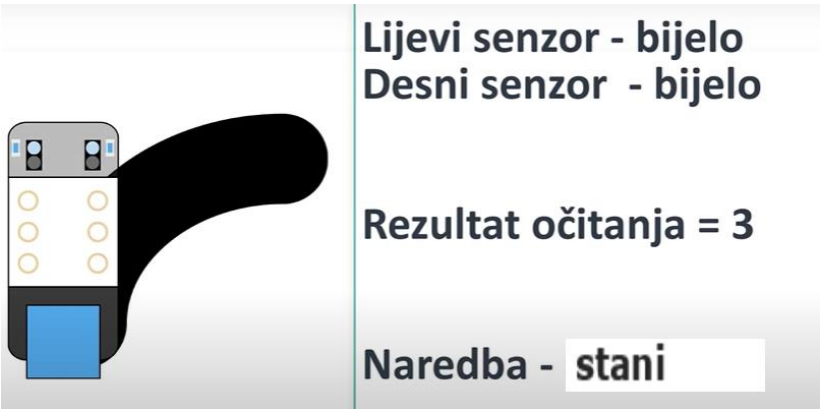


Slika 20. Skreni lijevo

U drugom načinu kretanja u kojem želimo da robot skrene lijevo. Prvi senzor je na tamnoj, a drugi na svijetloj podlozi te senzor očitava vrijednost 1 (*if line follower sensor port2 value = 1*). Lijevi kotač miruje, a desni se kreće (*left wheel turns at power 0%, right wheel at power 40%*) te tako robot skreće lijevo.

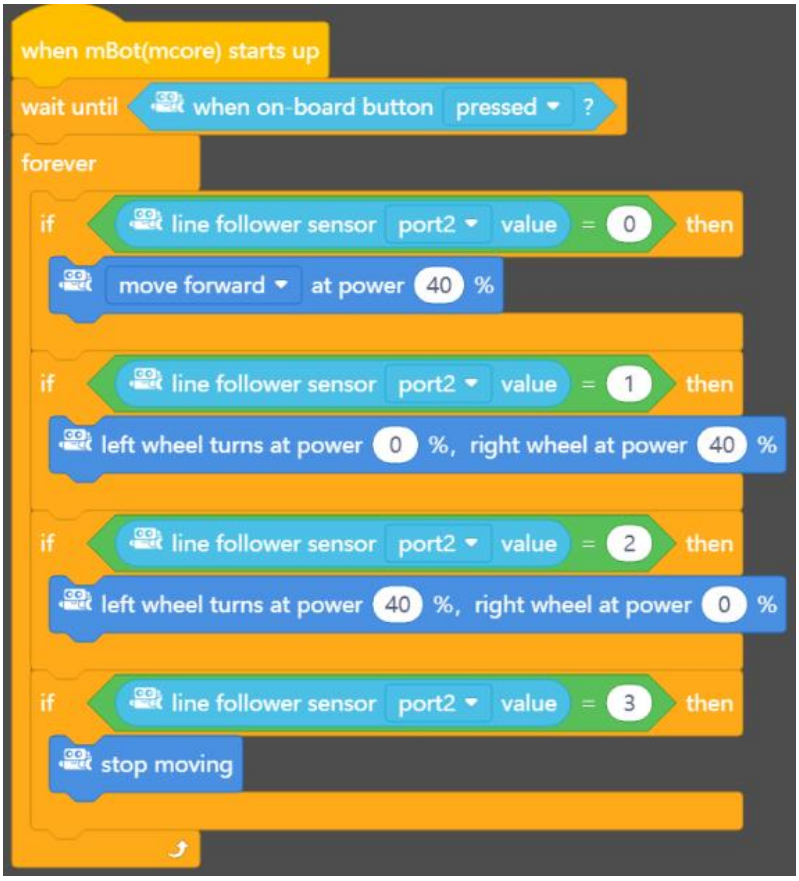
Slika 21. Skreni desno

U trećem načinu kretanja želimo da robot skrene desno. Prvi senzor detektira svijetlu podlogu, a drugi tamnu te senzor očitava vrijednost 2 (*if line follower sensor port2 value = 2*). Desni kotač miruje, a lijevi se kreće (*left wheel turns at power 40%, right wheel at power 0%)* stoga robot skreće desno.

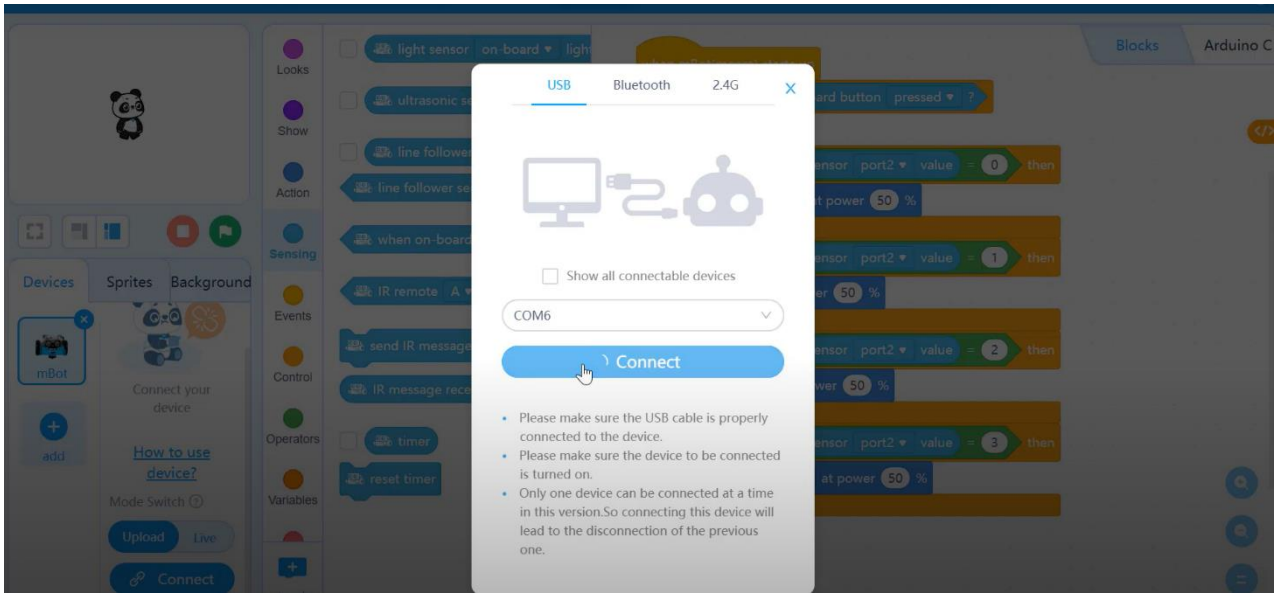


Slika 22. Robot staje

U posljednjem načinu kretanja robota oba su senzora na svijetloj podlozi. Senzor očitava vrijednost 3 (*if line follower sensor port2 value = 3*) te robot staje naredbom *stop moving*. Sve spomenute naredbe smještene su u *forever* petlju pa se program izvršava dokle god se robot nalazi po tamnoj podlozi.



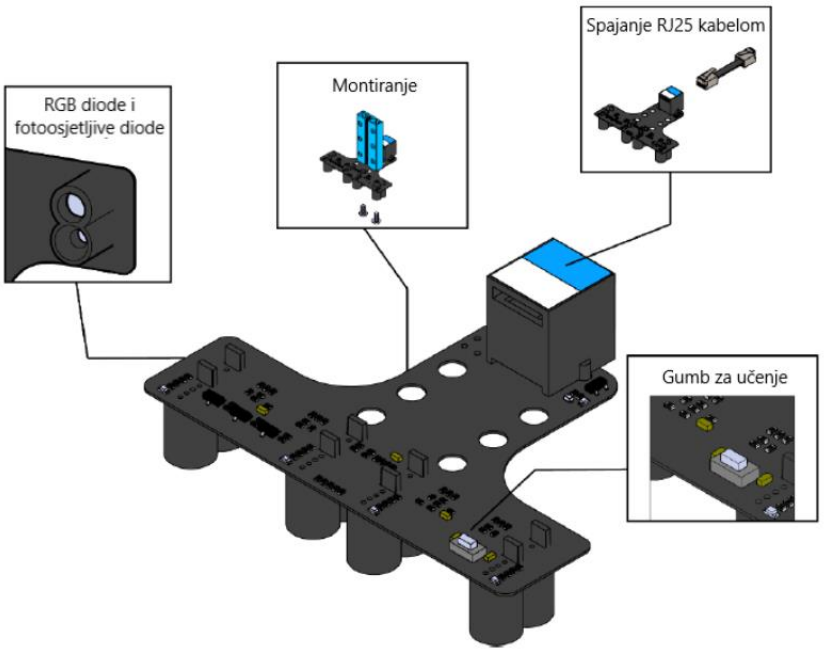
Slika 23. Programski kod praćenja linije Line Followerom



Slika 24. Prije samog kretanja robota po napisanom programu, potrebno je "prenijeti" program do mBota pritiskom na tipku Connect

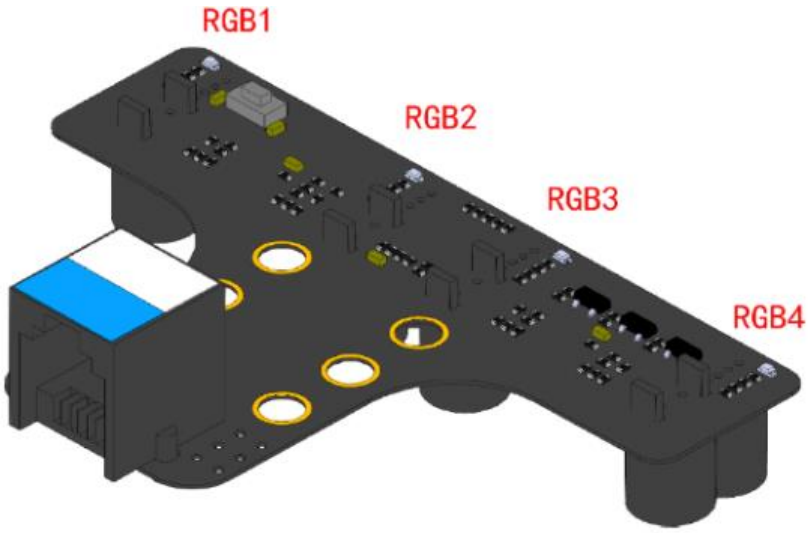
4.2. RGB Line Follower

RGB Line Follower sadrži četiri RGB diode i 4 fotoosjetljive diode u pripadajućim cijevima. S prednje strane senzora nalaze se LED indikator i gumb za učenje, koji služi za određivanje boje linije za praćenje na stazi i boje pozadine, što omogućuje da s ovim senzorom pratimo svijetle linije na tamnoj podlozi i tamne linije na svijetloj podlozi. RGB senzor gumbom za učenje mora „naučiti“ izgled staze kako bi se prilagodio vanjskim faktorima, tako što gumb prepoznaje i pamti boju pozadine i linije. Senzor se povezuje na mBotov *port* kabelom RJ25.



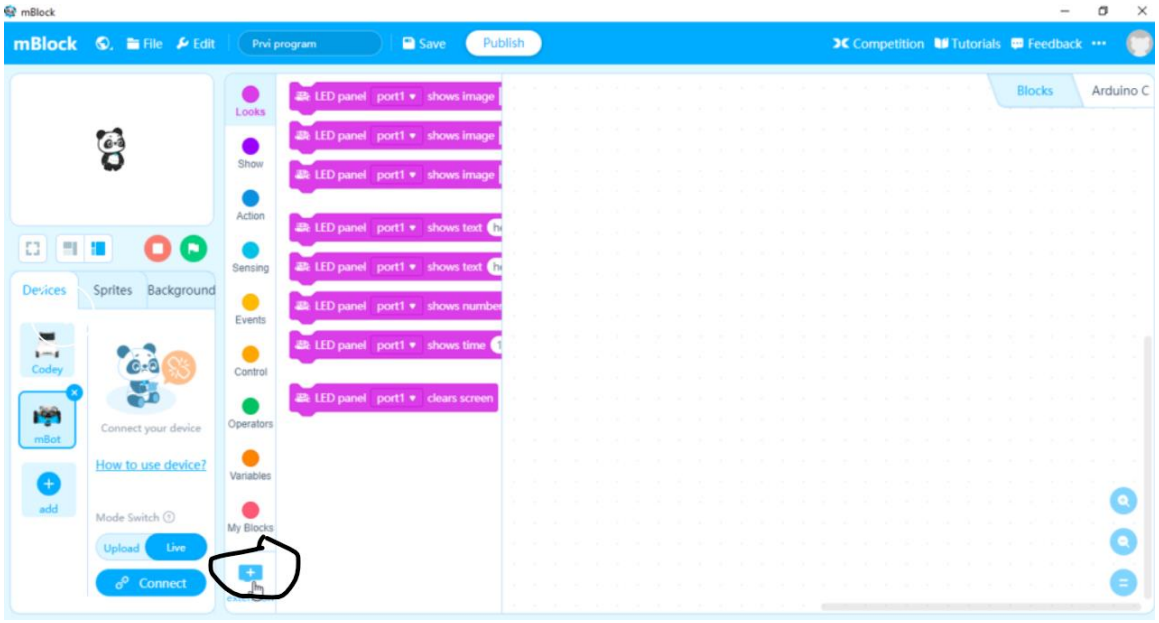
Slika 25. RGB senzor i njegovi dijelovi

RGB senzor ima četiri LED indikatora koji daju povratne informacije o praćenju linije i za učenje nove okoline te on osvjetljava stazu crvenom, plavom ili zelenom bojom. Držanjem gumba za učenje izabire se jedna od tih boja koja najbolje odgovara stazi. Ovisno o materijalu staze i svjetlosnim uvjetima RGB se postavlja na visini od 5 do 15 milimetara.



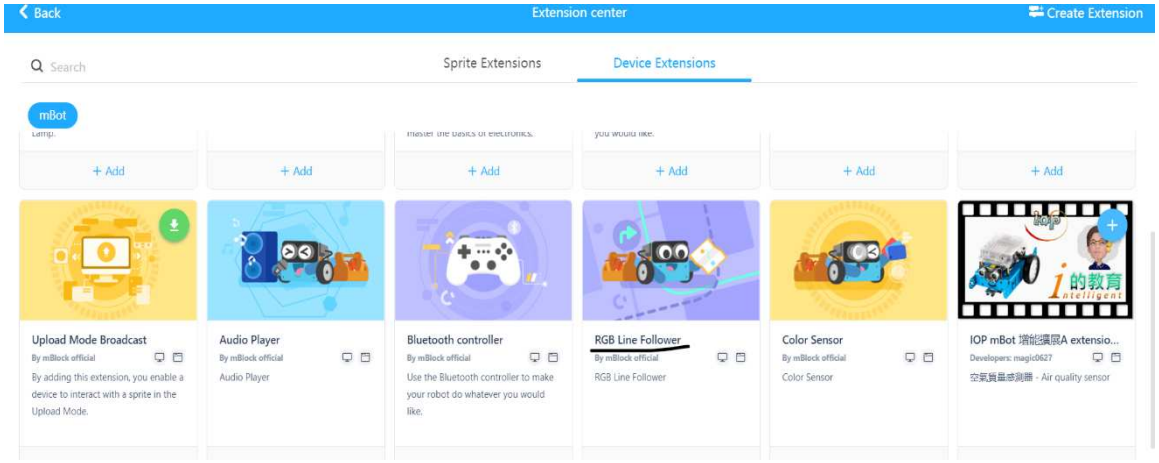
Slika 26. 4 diode na RGB senzoru

Na gornjoj slici vidimo raspored fotoosjetljivih dioda na senzoru. One prelaze preko različitih boja te pretvaraju te optičke informacije u elektroničke signale i određene izlazne vrijednosti. Softver tada analogne vrijednosti tih dioda koristi i stvara zadani algoritam za praćenje linije. Da bi se mBot kretao po stazi praćenjem linije, potrebno je da upozna i nauči boju linije i pozadine na sljedeći način. Podesimo boju svjetla senzora koje će osvjetljavati stazu tako da dugo pritisnemo na gumb za učenje čime se mijenjaju boje. Zatim postavimo RGB senzor na robota tako da sva četiri svjetla obasjavaju pozadinu staze te pritisnemo gumb za učenje. Kada indikatori počnu treperiti, proces učenja je ispravan, a nakon par sekundi prestaju treperiti što označava kraj procesa učenja boje pozadine. Da bi naučio boju linije, postavi se RGB senzor na robota tako da svjetla obasjavaju liniju te ponovimo postupak treperenja i čekanja par sekundi kada proces učenja boje linije završava. Provjera naučenog se izvodi tako što gledamo kako se senzor ponaša, to jest upale li se LED indikatori uz cijev iznad koje je pozadina i gase li se oni uz cijev koja je iznad linije. Ako su sva četiri svjetla iznad linije za praćenje, LED indikatori bi trebali biti ugašeni. Ako se to ne dogodi, potrebno je ponoviti postupak učenja pozadine i linije. Za razliku od Line Followera, naredbe za programiranje RGB senzora potrebno je odabrati. Nakon što u programu mBlock odaberete mBot kao uređaj za koji izrađujemo programski kod, ispod kategorija naredbi, dodajte opciju *extension* koja se ne nalazi u osnovnom skupu naredbi.

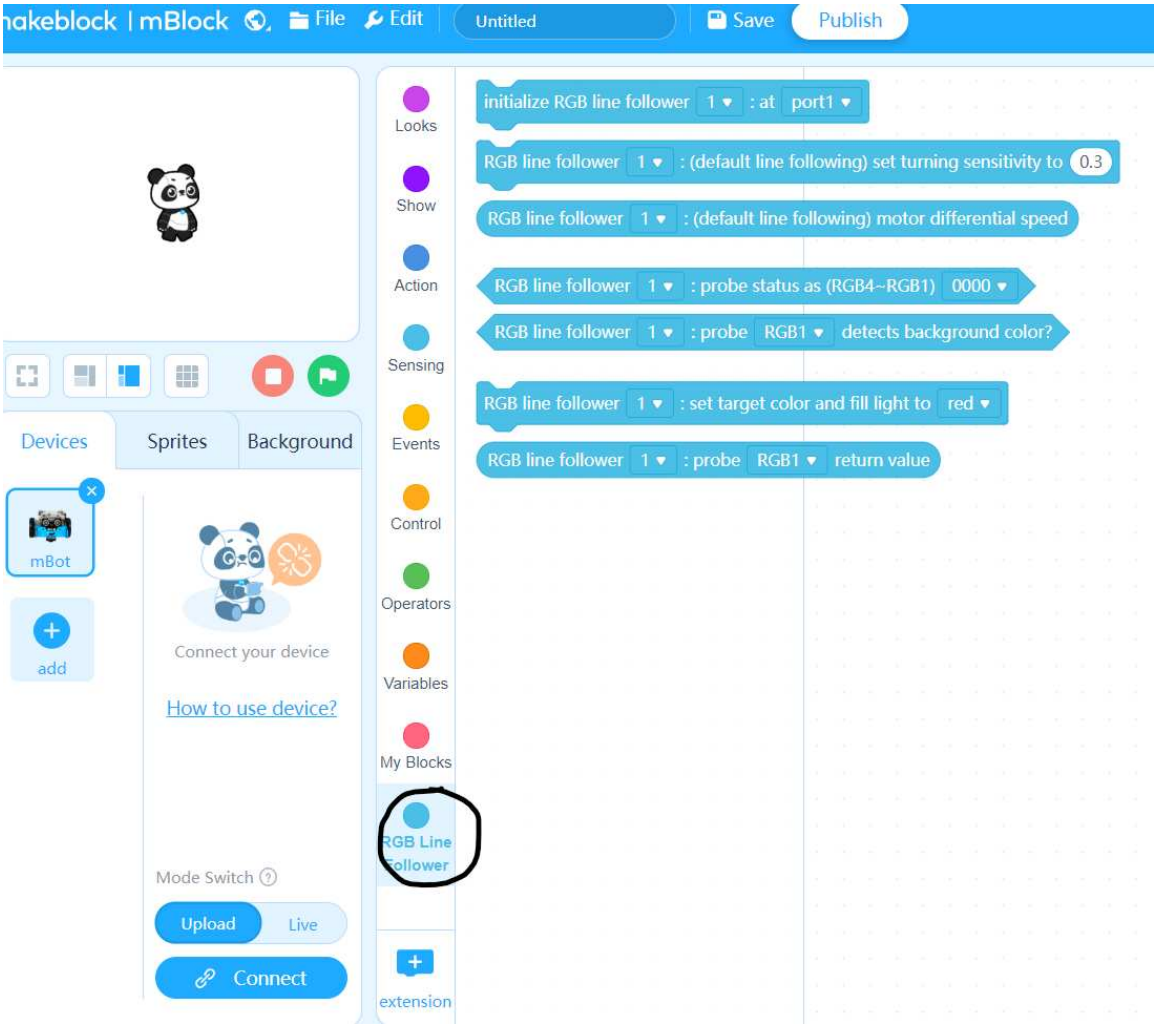


Slika 27. Pritisak tipke extension

Nakon pritiska tipke *extension*, otvara se prozor u kojem se odabire set naredbi za dodatne senzore za mBot. Nakon pronalaska kategorije RGB Line Follower potrebno je pritisnuti *Add* čime se završava preuzimanje naredbi.

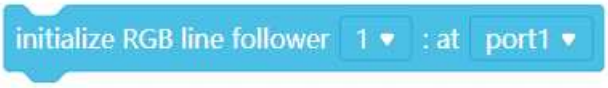


Slika 28. Biranje dodatnog paketa za RGB Line Follower

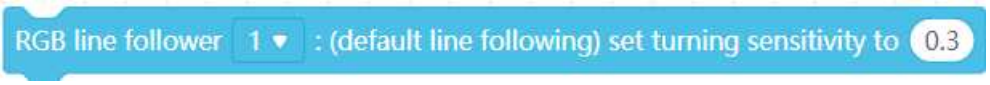


Slika 29. Popis naredbi kategorije RGB Line Follower

U popisu kategorija naredbi RGB Line Follower nalazi se na posljednjem mjestu što vidimo na slici iznad. Kategorija RGB sastoji se od sedam različitih naredbi koje ćemo sada jednu po jednu objasniti. Primijetimo da u ovoj kategoriji ima pet naredbi više nego što ih ima obični Line Follower što daje dosta dodatnih mogućnosti u radu s RGB senzorom koji je zato i zbog dva senzora više dosta precizniji i uporabljiviji u različitim uvjetima u odnosu na Line Follower.



Inicijalizira RGB senzor za praćenje linije u programu.



Određuje osjetljivost senzora. Što je veća vrijednost, veća je osjetljivost te senzor brže reagira.

Dobivanje diferencijalne brzine. Odmak prema RGB4 vraća pozitivnu vrijednost. Odmak prema RGB1 vraća negativnu vrijednost.

Provjerava status senzora (0 ili 1). 0 – senzor je na liniji. 1 – senzor je na pozadini. Primijetite da redoslijed ide od RGB4 do RGB1.

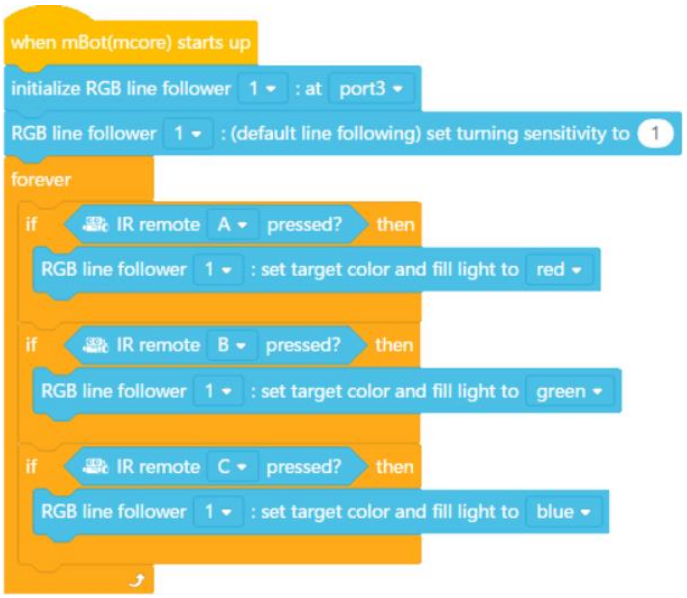
Provjerava detektira li senzor pozadinu.

Određuje boju kojom senzor osvjetljava površinu staze.

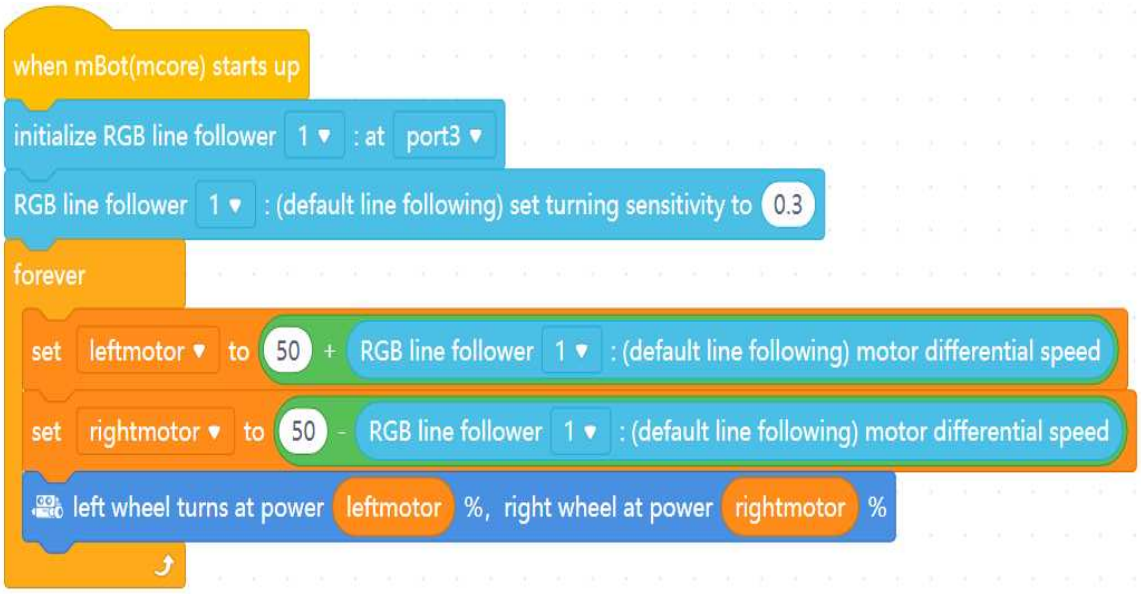
Vraća vrijednost boje koju senzor očitava.

Kao i za Line Follower, tako i za RGB senzor prilikom programiranja senzora za praćenje linije provjerite na kojem ulazu (*portu*) je spojen senzor na upravljačkoj pločici te u skladu s time izradite program.

Sada ćemo napisati blokovski kod za određivanje boje za osvjetljenje staze. Na početku programa moramo inicijalizirati RGB senzor na određeni ulaz i njegovu osjetljivost. U ovom primjeru je bio spojen na treći *port*, a osjetljivost je postavljena na maksimalnu vrijednost. Kao što smo već spomenuli, pritiskom na gumb se mijenja boja kojom senzor osvjetljava podlogu i liniju. No postoji i način preko daljinskog upravljača koji ima tri tipke: A, B i C. Pritiskom na tipku A senzor stazu osvjetljava crvenom bojom. Pritiskom na tipku B senzor stazu osvjetljava zelenom bojom. Pritiskom na tipku C senzor stazu osvjetljava plavom bojom.



Slika 30. Napisani kod u mBlocku zadatka iznad

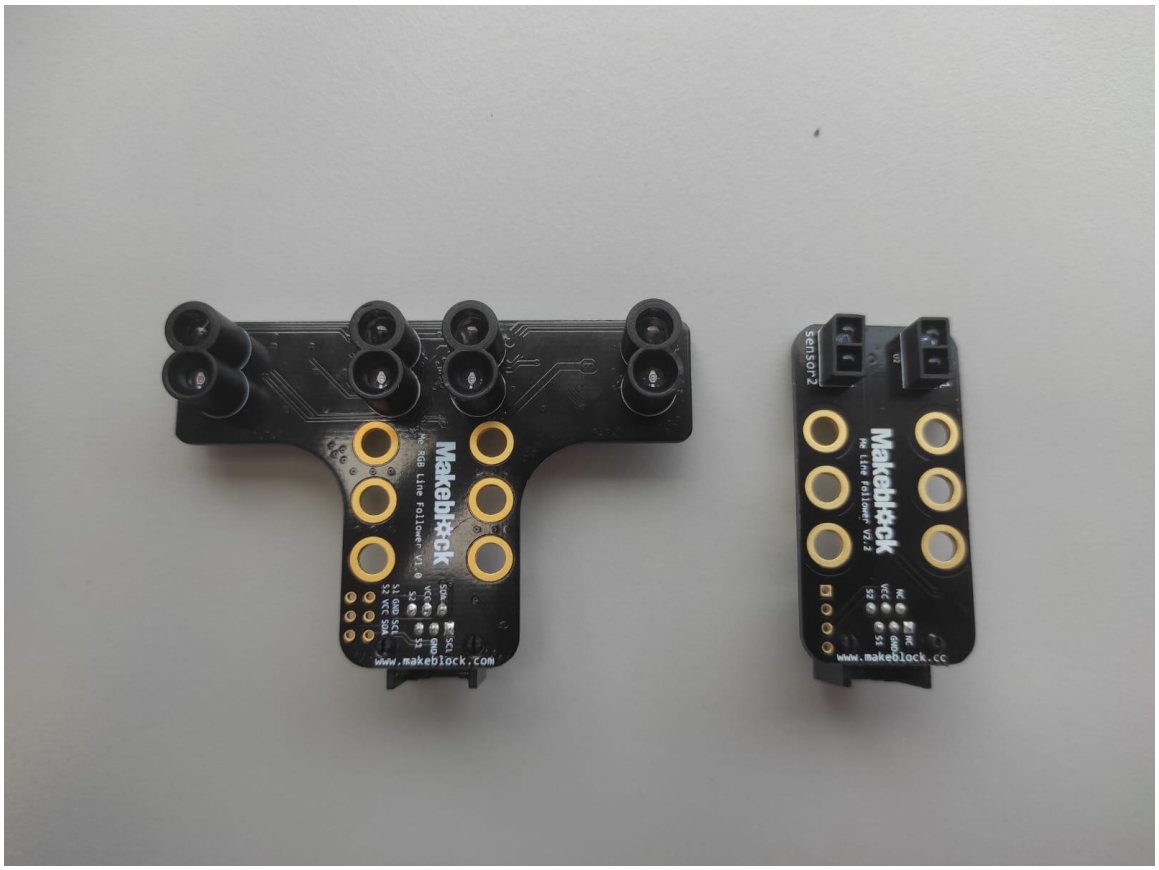


Slika 31. Programski kod praćenja linije RGB Line Followerom

5. Provjera učinkovitosti senzora

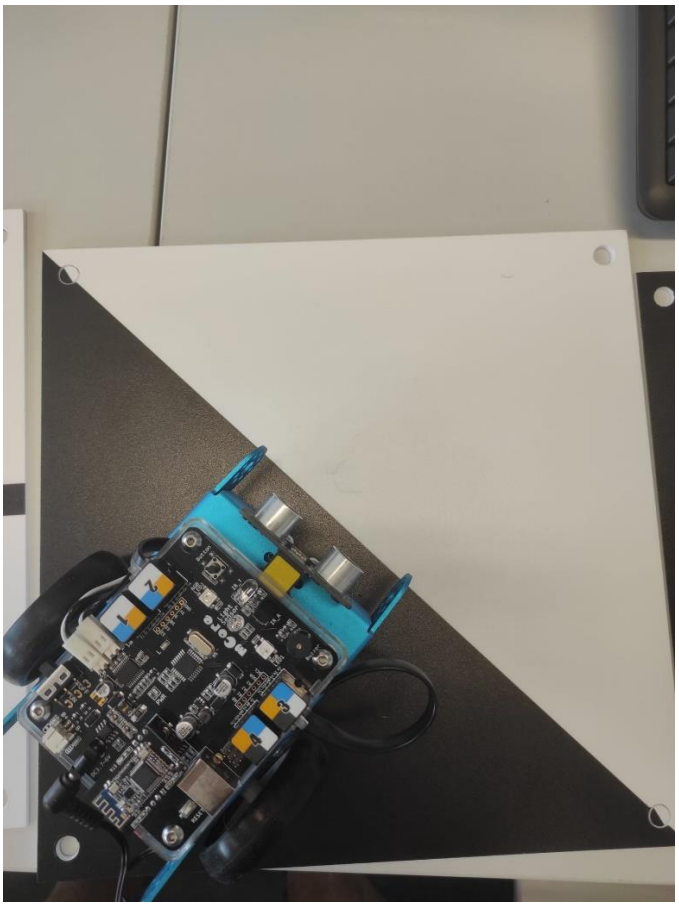
Nakon predstavljanja oba senzora te njihovih dijelova mogućnosti i načina rada, ovo poglavlje će biti posvećeno prikazivanjem prednosti i nedostataka ovih dvaju senzora te ideje za njihovo praktično korištenje u nastavi programiranja.

Radeći neke testove te provjeravajući rad senzora na različitim podlogama i uvjetima ustanovio sam najpogodnije uvjete i teren za rad sa senzorima za praćenje linije u nastavi programiranja. Paralelno proučavajući s oba senzora provjeravao sam iste situacije s upotrebom oba senzora jednim za drugim.

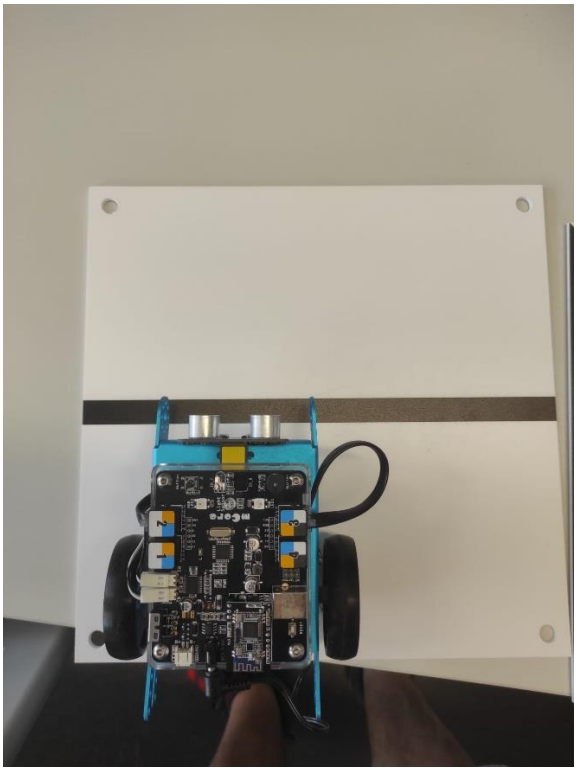


Slika 32. Proučavani senzori

Svoje istraživanje započeo sam osnovnim prepoznavanjem tamne i svijetle podloge uspoređujući preciznost i brzinu jednog po jednog senzora. Line Follower, zbog samo dva senzora koja detektiraju podlogu te šalju vrijednosti upravljačkoj pločici, pokazao je probleme s tanjim linijama sporim prepoznavanjem koje se događalo samo pod nekim kutovima i udaljenostima senzora od podloge što je potencijalno problematično za nastavu za vrijeme koje bi se u tom slučaju dogodio kaos u razredu, kada bi većina djece javila isti problem. Također misleći da su nešto pogriješili pri kodiranju, mogu posumnjati u svoje sposobnosti ili izgubiti volju za daljnjim nastavkom sata, ne znajući da je senzor nepouzdan u određenim situacijama.



Slika 33. Prelazak s tamnog na svjetli dio podloge (i obrnuto)

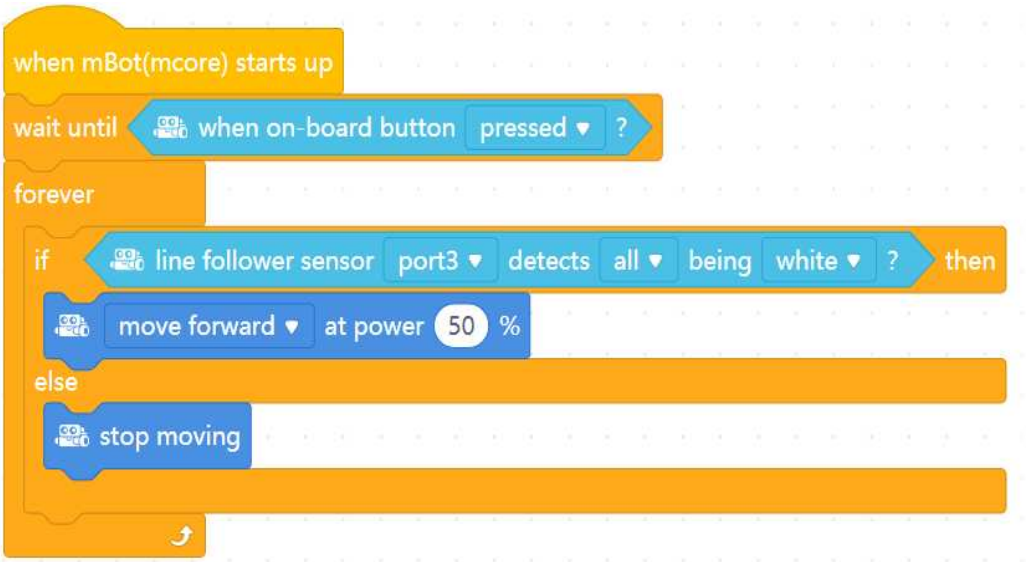


Slika 34. Uočavanje tanke crne linije

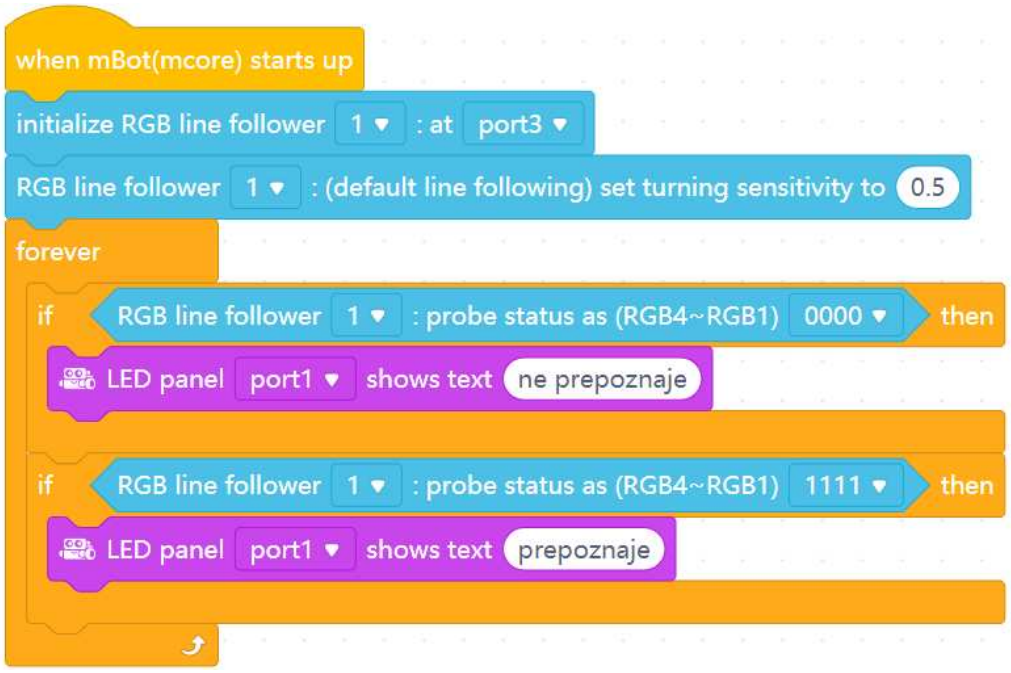
RGB senzor, zahvaljujući svoje četiri diode, nije imao jednake probleme s debljinom kao i Line Follower.

Sljedeća provjera je bila test raspoznavanja različitih podloga u boji, ne samo bijele i crne. Dok su tamnije nijanse prepoznala oba senzora, Line Follower je imao određenih problema s raspoznavanjem svjetlijih podloga kao što je žuta podloga.

Slika 35. Podloge različitih boja

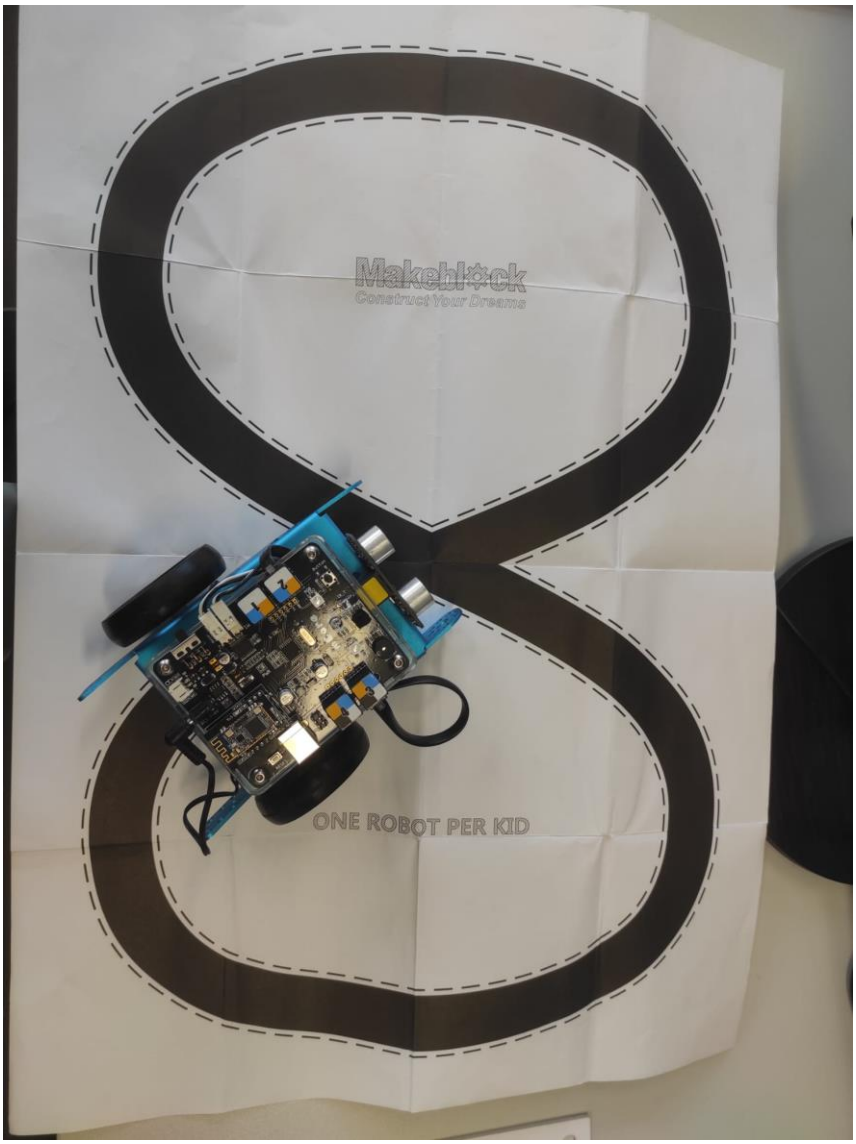


Slika 36. Program za prepoznavanje tamnih i svijetlih podloga pomoću Line Followera



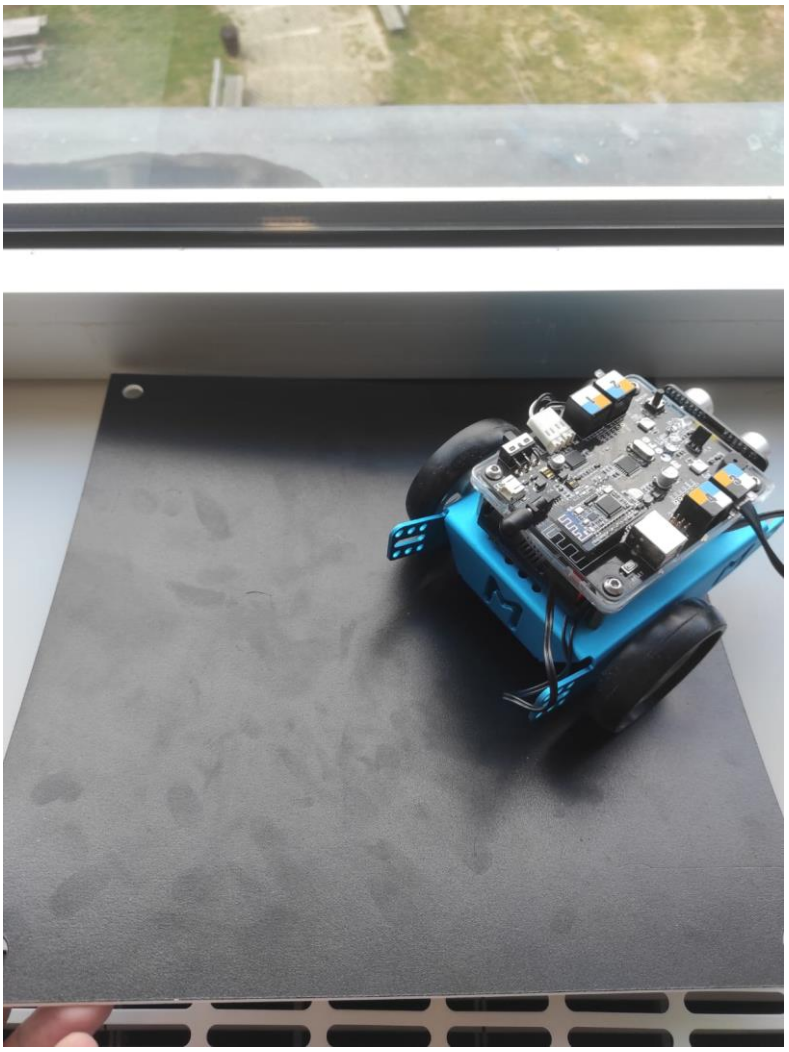
Slika 37. Program za prepoznavanje tamnih i svijetlih podloga pomoću RGB Line Followera

Sljedeća stavka je bila provjera kretanja robota s oba senzora po liniji u obliku broja osam. Razlog važnosti ovog testa bio je dio te linije koja se presijeca jedna preko druge u sredini. Zanimalo me je hoće li to sjecište „zbuniti“ senzore te iznjedriti pogrešku. S oba senzora (koristeći isti kod za kretanje po liniji koji su već prikazani u prošlom poglavlju) prošao sam par puta po „osmici“ i iako je većina krugova kod oba senzora prošla bez problema i po pravoj putanji, ipak je mBot dva puta otišao u krivom pravcu i to opet s jednostavnijim senzorom, no s obzirom da je uzorak napravljen na temelju više ponavljanja, možemo zaključiti da je i Line Follower dovoljno dobar za ovakav oblik aktivnosti.



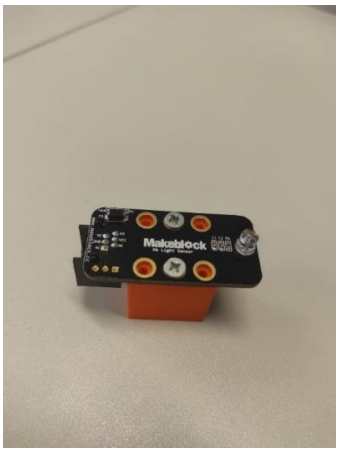
Slika 38. Kretanje mBota po obliku broja 8

Najveći problem za rad oba senzora predstavljala je svjetlost. Ovisno o osvjetljenju prostorije i vanjskom prirodnom svjetlu senzori će se ponašati drugačije zbog podloge koja reflektira svjetlost pa onemogućuje pravilnu detekciju senzora. Zbog nemogućnosti (za sada, nadajmo se da će se to unaprijediti u godinama koje dolaze ispred nas s daljnjim razvitkom robotike) kalibriranja senzora, tj. promjene osjetljivosti senzora te omogućavanja rada na bilo kojoj količini svjetlosti (postoji robot Zagria kojeg je moguće kalibrirati pa ćemo izvođenje kalibracije pokazati u posljednjem poglavlju).



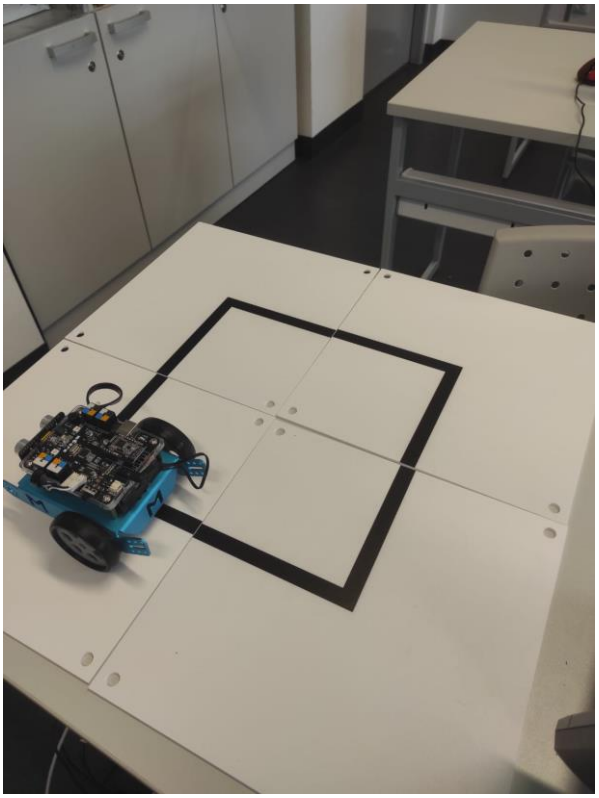
Slika 39. Vidimo crnu podlogu od koje se reflektira svjetlost što onemogućuje normalan rad senzora

Ovaj problem stavlja na veliku kušnju održavanje nastave programiranja s robotima, kako u učionicama tako i u prirodi, tj. na otvorenom. Nažalost zbog već spomenute nemogućnosti kalibracije, dva proučavana senzora funkcioniraju u određenim uvjetima na koje nastavnici ne mogu uvijek djelovati. Stoga bi bilo dobro prije izvođenja satova s robotima i pripadajućim senzorima, provjeriti uvjete i odlučiti je li moguće održati sat kakav smo zamislili. Stoga jedna od zamisli koja bi mogla pomoći nastavnicima u osmišljavanju sata te potvrde mogućnosti održavanja zamišljenog sata je upotreba svjetlosti (Light Sensor) koji je također kao i naši senzori napravljen za program mBlock i rad s mBotovima. Ideja je postavljanje svjetlosnog senzora prije sata te dobivanja informacije o količini svjetlosti koja stiže u prostoriju (ili na otvorenom). Light Sensor detektira svjetlost te šalje upravljačkoj pločici vrijednost u rasponu od 0 do 1023. Ako je vrijednost veća od 500, svjetlost je najčešće prevelika za izvođenje svih zamišljenih komponenti sata, ako je vrijednost manja od 100 tada su noćni uvjeti te je problem nedostatak svjetlosti što također umanjuje mogućnost uspjeha. Optimalna količina svjetlosti je između 100 i 500 što predstavlja otprilike vrijednost sličnu unutrašnjim uvjetima u prostoriji s normalnom količinom svjetlosti. Izvodeći testove sa senzorima i provjeravanjem količine svjetlosti pomoću Light Sensora primijetio sam da iako pomaže u slučaju da je svjetlosti previše ili premalo (što uzrokuje nepouzdani rad senzora), čak i kada je svjetlost u granici od 100 do 500 znalo je biti problema, no ipak jako rijetko. Stoga ova provjera može biti od velike pomoći nastavnicima koji koriste robote u nastavi programiranja.



Slika 40. Light Sensor

Posljednji problem na koji sam naišao su oštri zavoji. Za razliku od blagih zavoja kao što je bilo u testu s „osmicom“ gdje su oba senzora bez prevelikih problema uspijevala pratiti liniju, u ovom slučaju to je malo teže. Prvo sam napravio podlogu sa 4 oštra zavoja kao na slici.



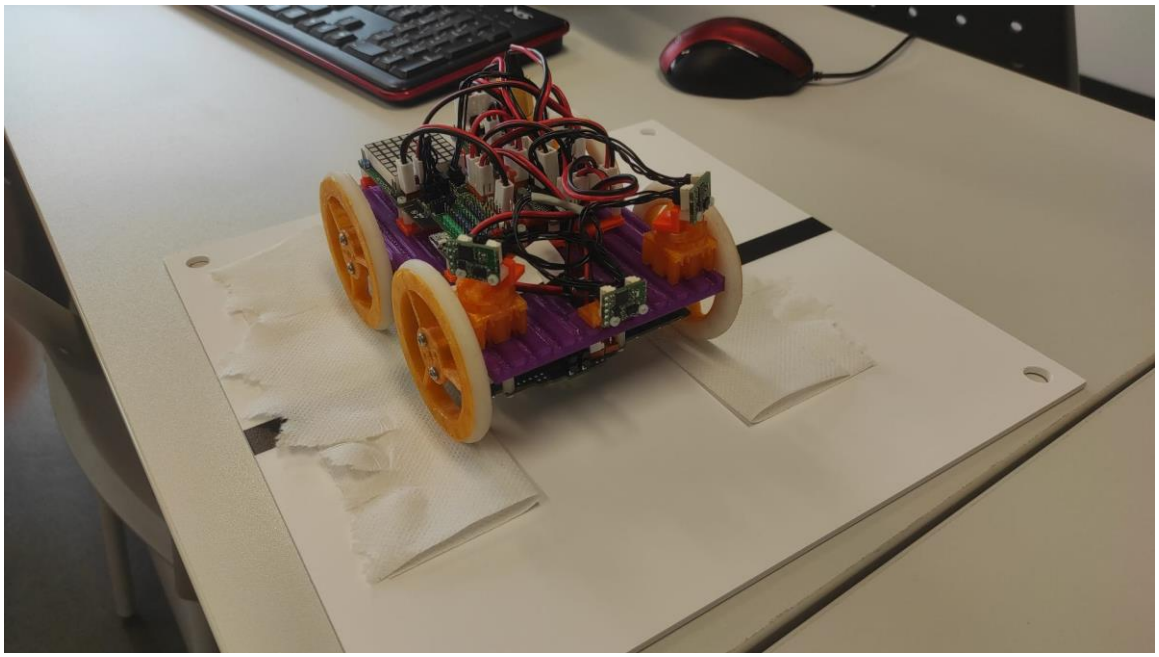
Slika 41. Kretanje mBota po oštrim zavojima

Zatim napravivši kodove (programski kod je također ostao isti kao i za kretanje po „osmici“) za kretanje po liniji za oba senzora pustio sam mBot te primijetio probleme. Praćenje crte pomoću dva senzora radilo je većinski dobro, sve dok se nisu pojavili oštri zavoji. Problem je taj što, za praćenje blagih zavoja, razlika u brzinama lijevih i desnih kotača ne smije biti prevelika. Za oštre zavoje treba biti vrlo velika. U suprotnom će robot na zavoju produžiti ravno. Stoga se senzor s četiri diode ponovo pokazao kao učinkovitiji i točniji. Naravno kod oba senzora možemo primijetiti pomicanje robota dio po dio sve dok se ne okrene za 90 stupnjeva, nažalost automatsko okretanje za 90 stupnjeva nije moguće trenutno bez obzira koji senzor koristimo.

6. Kalibracija

Zbog mogućih problema zbog previše svjetlosti, što smeta mnogim senzorima u detekciji određenih fizičkih pojava, nekalibrirane senzore je teško koristiti, u našim slučajevima u aktivnostima s djecom izvan školske ustanove čak i nemoguće. Tijekom rukovanja sa senzorima kojima se bavi ovaj rad, najviše problema uzrokuje kontrola količine svjetlosti na mjestu testiranja robota. Kako Line Follower i RGB Line Follower nemaju mogućnost kalibracije, pokazati ćemo primjer kalibracije na robotu Zagria.

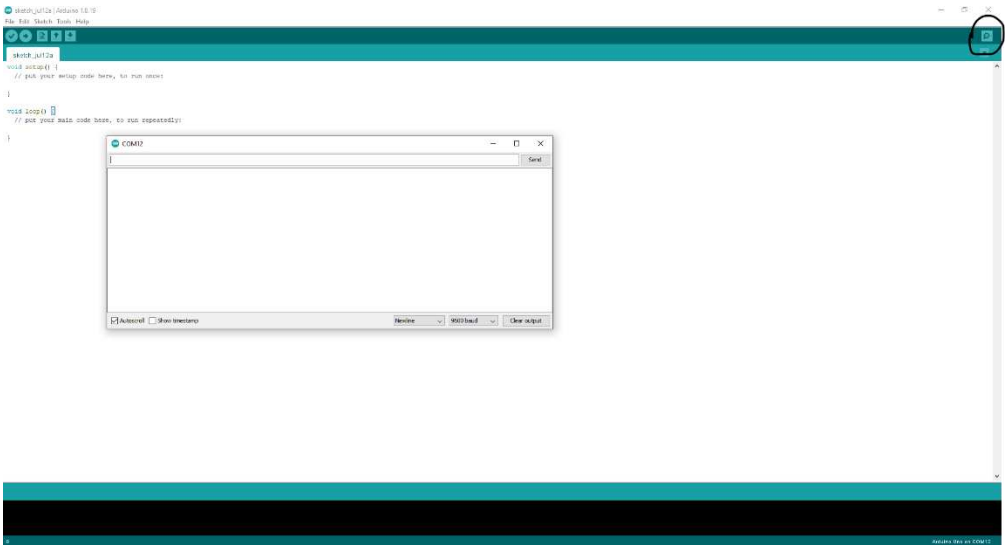
Provedimo sada kalibraciju. Za početak stavimo robota na bijelu podlogu, ispred crne linije. Radi problema s trenjem između silikonskih kotača i podloge, stavio sam dva komada papirnatih ubrusa da bi se lakše provela kalibracija robota.



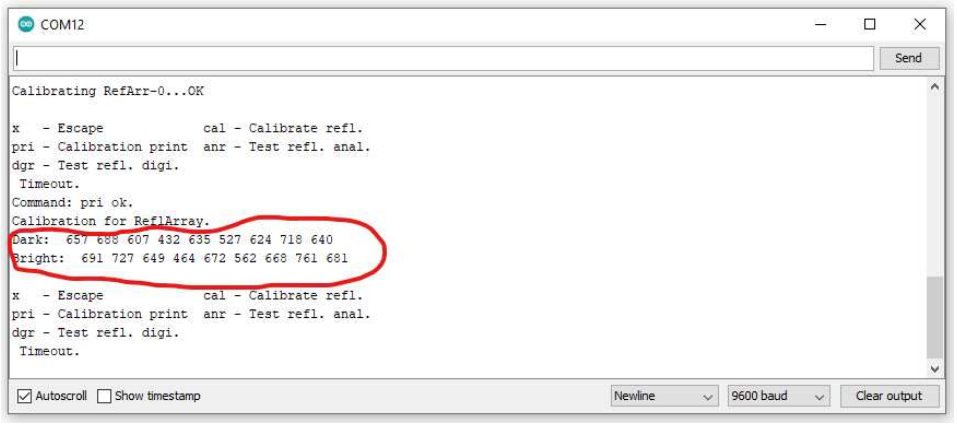
Slika 42. Zagria na podlozi

Zatim otvorimo Arduino i Serial Monitor u koji tipkamo naredbu "*ref*", kako bismo došli do podmenija za reflektivni senzor. Iz podmenija, odaberemo naredbu "*cal*" (engl. "*calibration*"). Ispisat će se tekst. Tada počnemo pomicati robota naprijed-natrag, tako da svi senzori vide liniju i podlogu više puta. Nakon par sekundi će se ispisati tekst "*OK*", što

znači da je postupak kalibracije gotov. Za vrijeme kalibracije će LED reflektivnog senzora brzo treptati pa je i to način da se vidi kad je kalibracija okončana.



Slika 43. Sučelje Arduina i otvoreni Serial Monitor



Slika 44. Brojčane vrijednosti nakon kalibriranja zaokružene crvenom bojom

Nakon kalibracije ispišu se brojčane vrijednosti svjetline i mraka koje je Zagria sad primila kao posljedicu kalibriranja. Ovim kratkim procesom nastavnici mogu brzo riješiti probleme viška svjetlosti, stoga se nadamo da će uskoro i dva gore predstavljena senzora na mBotu imati mogućnost kalibracije.

7. Popis izrađenih slika i slika preuzetih iz literature

Slika 1. Samuel Bagno (lijevo na slici) 2
Slika 2. George Smith i Willard Boyle 4
Slika 3. Senzori 5
Slika 4. Primjeri nekih senzora 7
Slika 5. Prikaz mBota sa svih strana 11
Slika 6. Prikaz 4 osnovna koraka u sastavljanju robota 11
Slika 7. Komponente nesloženog modela mBota 12
Slika 8. Neki od dijelova mBota 13
Slika 9. Nastanak mBlocka 14
Slika 10. Program mBlock 15
Slika 11. Program Scratch 15
Slika 12. Odabir uređaja 16
Slika 13. Opcije povezivanja i dva načina rada 17
Slika 14. Ekstenzije koje je moguće dodati u mBlocku 17
Slika 15. Specifikacije uređaja (mBot) 18
Slika 16. Line Follower 19
Slika 17. Naredbe za senzor zaokružene su crvenom bojom 20
Slika 18. Četiri različite mogućnosti kretanja robota 21
Slika 19. Idi ravno 22
Slika 20. Skreni lijevo 22
Slika 21. Skreni desno 23
Slika 22. Robot staje 23
Slika 23. Programski kod praćenja linije Line Followerom 24
Slika 24. Prije samog kretanja robota po napisanom programu, potrebno je "prenijeti" program do mBota pritiskom na tipku Connect 24
Slika 25. RGB senzor i njegovi dijelovi 25
Slika 26. 4 diode na RGB senzoru 26
Slika 27. Pritisak tipke extension 27
Slika 28. Biranje dodatnog paketa za RGB Line Follower 27
Slika 29. Popis naredbi kategorije RGB Line Follower 28
Slika 30. Napisani kod u mBlocku zadatka iznad 30
Slika 31. Programski kod praćenja linije RGB Line Followerom 31
Slika 32. Proučavani senzori 32
Slika 33. Prelazak s tamnog na svjetli dio podloge (i obrnuto) 33
Slika 34. Uočavanje tanke crne linije 34
Slika 35. Podloge različitih boja 34
Slika 36. Program za prepoznavanje tamnih i svijetlih podloga pomoću Line Followera 35
Slika 37. Program za prepoznavanje tamnih i svijetlih podloga pomoću RGB Line Followera 35
Slika 38. Kretanje mBota po obliku broja 8 36
Slika 39. Vidimo crnu podlogu od koje se reflektira svjetlost što onemogućuje normalan rad senzora 37

Slika 40. Light Sensor 38
Slika 41. Kretanje mBota po oštrim zavojima 39
Slika 42. Zagria na podlozi 40
Slika 43. Sučelje Arduina i otvoreni Serial Monitor 41
Slika 44. Brojčane vrijednosti nakon kalibriranja zaokružene crvenom bojom 41

8. Bibliografija

[1.] Edukacijski web portal Croatian Makersa, dostupno na https://izradi.croatianmakers.hr/, pristupljeno 25.7.2022.

[2.] Hrvatsko društvo za robotiku, Uvod u robotiku, dostupno na https://top.in.hr/Page.ashx?id=457, pristupljeno 15.8.2022.

[3.] Jelena Pisarov, Programming the mBot Robot in School, dostupno na https://www.researchgate.net/profile/GyulaMester/publication/337926235_Programming_the_mBot_Robot_in_School/links/5df661dea6fdcc28372455e3/Programming-the-mBot-Robot-in-School.pdf, pristupljeno 15.8.2022.

[4.] Kurikulum nastavnoga predmeta informatika za osnovne i srednje škole, dostupno na https://narodne-novine.nn.hr/clanci/sluzbeni/2018_03_22_436.html, pristupljeno 20.9.2021., pristupljeno 25.7.2022.

[5.] Line follower sensor's review, dostupno na https://www.youtube.com/watch?v=xysz2xh69rQ, pristupljeno 25.7.2022.

[6.] Line sensors and how to use them, dostupno na https://www.futurelearn.com/info/courses/robotics-with-raspberry-pi/0/steps/75899, pristupljeno 15.8.2022.

[7.] Merlin 20/21 Priručnik za nastavnike, dostupno na https://wiki.srce.hr/pages/viewpage. action?pageId=53379097, pristupljeno 1.9.2022.

[8.] Nermina Zainović-Uzunović, Mjerna ternika, dostupno na https://unze.ba/download/SkriptaMetrologija.pdf, pristupljeno 25.7.2022.

[9.] Robotics in education—Advantages, dostupno na benefits and importance for kids, https: //www.idtech.com/blog/educational-benefits-robotics, pristupljeno 15.8.2022.

9. Sažetak

Tema kojom smo se bavili u ovom diplomskom radu je korištenje senzora za praćenje linije u nastavi programiranja. Podijeljen je u pet središnjih poglavlja. Prvo od središnjih poglavlja rada proučava nastanak i razvoj senzora, predstavljajući kategorije i vrste senzora te motivaciji za upotrebom istih u nastavi informatike. U drugom poglavlju predstavljamo robot mBot, njegove specifikacije način korištenja njega i programa mBlock u kojem blokovskim programiranjem upisujemo kodove za pokretanje robota sa senzorima. Treće je podijeljeno na dva dijela u kojima se navode i analiziraju dva senzora Line Follower i RGB Line Follower. Četvrto poglavlje sadrži praktični dio ovog rada koji se sastoji od raznih problema i zadataka koji nam daju precizniji uvid u prednosti i mane pojedinog seminara testiranih u istim uvjetima. U petom i posljednjem poglavlju glavnog dijela ovog diplomskog rada definiramo pojam kalibracije robota te prikazujemo postupak kalibracije na jednom primjeru. Kalibracija je nužna zbog otežavajućih okolnosti pri upotrebi robota i senzora u učionicama i na otvorenom prostoru zbog danje svjetlosti.

10. Summary

The topic we dealt with in this thesis is the use of sensors for line tracking in programming classes. It is divided into five central chapters. The first of the central chapters of the work studies the origin and development of sensors, presenting the categories and types of sensors and the motivation for using them in teaching computer science. In the second chapter, we present the mBot robot, its specifications, the way to use it and the mBlock program, in which we use block programming to write the codes to start the robot with sensors. The third is divided into two parts in which the two sensors Line Follower and RGB Line Follower are listed and analyzed. The fourth chapter contains the practical part of this work, which consists of various problems and tasks that give us a more precise insight into the advantages and disadvantages of an individual seminar tested under the same conditions. In the fifth and last chapter of the main part of this thesis, we define the concept of robot calibration and show the calibration procedure on one example. Calibration is necessary due to aggravating circumstances when using robots and sensors in classrooms and in open spaces due to daylight.

11. Životopis

Moje ime je Luka Petričević i rođen sam 16. prosinca 1995. godine u Splitu. Pohađao sam osnovnu školu „Skalice“, a srednjoškolsko školovanje završio sam u splitskoj matematičkoj gimnaziji „MIOC“. Preddiplomski sveučilišni studij upisao sam na Prirodoslovno-matematičkom fakultetu, smjer Matematika upisao sam 2014., a 2016. sam se prebacio na preddiplomski sveučilišni studij Matematika; smjer: nastavnički. Diplomski sveučilišni studij Matematika i informatika; smjer: nastavnički, upisao sam 2018. godine na Prirodoslovno-matematičkom fakultetu u Zagrebu.