

Programiranje Lego Mindstorms Autića

Novačić, Ivan

Undergraduate thesis / Završni rad

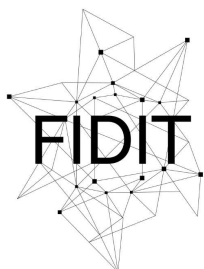
2020

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Rijeka / Sveučilište u Rijeci**

Permanent link / Trajna poveznica: <https://um.nsk.hr/um:nbn:hr:195:163836>

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

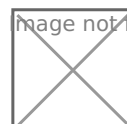
Download date / Datum preuzimanja: **2024-04-25**



Sveučilište u Rijeci
**Fakultet informatike
i digitalnih tehnologija**

Repository / Repozitorij:

[Repository of the University of Rijeka, Faculty of Informatics and Digital Technologies - INFORI Repository](#)



Sveučilište u Rijeci – Odjel za informatiku

Preddiplomski jednopredmetni studij informatike

Ivan Novačić

Programiranje Lego Mindstorms AI autića

Završni rad

Mentor: izv. prof. dr. sc. Marina Ivašić-Kos

Rijeka, 22. rujna 2020.

Sadržaj

1. Zadatak za završni rad	2
2. Sažetak	3
3. Uvod.....	4
4. Općenito o robotici	5
4.1. Kratka povijest	6
4.2. Zašto je robotika važna?	8
4.3. Autonomna vozila	9
4.3.1. Razine autonomije	10
4.3.2. Problemi autonomije	12
5. Lego Mindstorms EV3	13
5.1. Senzori.....	15
5.2. Motori.....	17
5.3. Programski Brick.....	20
5.4. EV3 program.....	22
5.4.1. Autonomni način rada.....	24
5.4.2. Ručno upravljanje autićem	29
5.5. Izrada autonomnog/RC autića	31
6. Zaključak	38
7. Literatura	39
8. Popis slika	40
9. Prilozi	42

1. Zadatak za završni rad



Rijeka, 17. veljače 2020.

Zadatak za završni rad

Pristupnik: **Ivan Novačić**

Naziv završnog rada: **Programiranje Lego Mindstorms autića**

Naziv završnog rada na eng. jeziku: **Programming the Lego Mindstorms car**

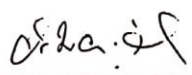
Sadržaj zadatka: Proučiti što je robotika, čime se robotika bavi, kratku povijest i razvoj robotike s naglaskom na autonomna vozila.

Opisati kako funkcioniraju elementi Lego Mindstorms EV3 seta poput senzora, motora i upravljačkog modula te rad s EV3 računalnim programom. Proučiti Lego Mindstorms projekte te mogućnosti i postojeća iskustva vezano za upravljanje Lego Mindstorms robotom koja se mogu primijeniti na upravljanje autonomnim vozilom.

Osmisliti i kreirati Lego autonomni autić te opisati dijelove od kojih je složen i njihovu funkciju. Izraditi i objasniti algoritme za upravljanje Lego autićem koji mu omogućuju autonomnu vožnju te uočavanje i izbjegavanje prepreke. Predložiti daljnja poboljšanja.

Mentor

izv. prof. dr. sc. Marina Ivašić-Kos

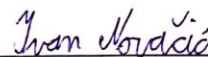


Voditelj za završne radove

doc. dr. sc. Miran Pobar



Zadatak preuzet: 17.6.2020.



(potpis pristupnika)

2. Sažetak

U ovom radu je opisan Lego Mindstorms EV3 set. Objašnjeno je kako svi elementi seta funkcioniraju, poput senzora, motora i ostatka hardvera pa sve do EV3 računalnog programa za kreiranje softvera. Spominje se i robotika, kratka povijest i ponešto o autonomnim vozilima. Opisan je i autonomni Lego autić složen isključivo od dijelova ovog seta. Kako ga napraviti, kako mu funkcionira program i sveukupno iskustvo kako bi ga bilo koja osoba mogla sama složiti i razumjeti kako se njime upravlja.

Ključne riječi: Robotika, Autonomno, Lego, Mindstorms, Senzori, Motori, Brick, EV3

3. Uvod

Glavna tema ovog rada je slaganje Lego Mindstorms EV3 seta (verzija 31313) i programiranje složenog modela. U ovom slučaju odlučio sam umjesto robota složiti Lego autić te ga programirati da se bez ljudske kontrole "autonomno" kreće prostorom.

Opisana je relativno kratka povijest robotike i zašto je važna u današnjem svijetu. Međutim, robotika nije jedina grana tehnologije vezana uz ovaj projekt. Jednako je važno znanje o autonomnim vozilima. Opisano je kako ona generalno funkcioniraju, neki od problema kod autonomije i šest stupnjeva po kojima se takva vozila kategoriziraju.

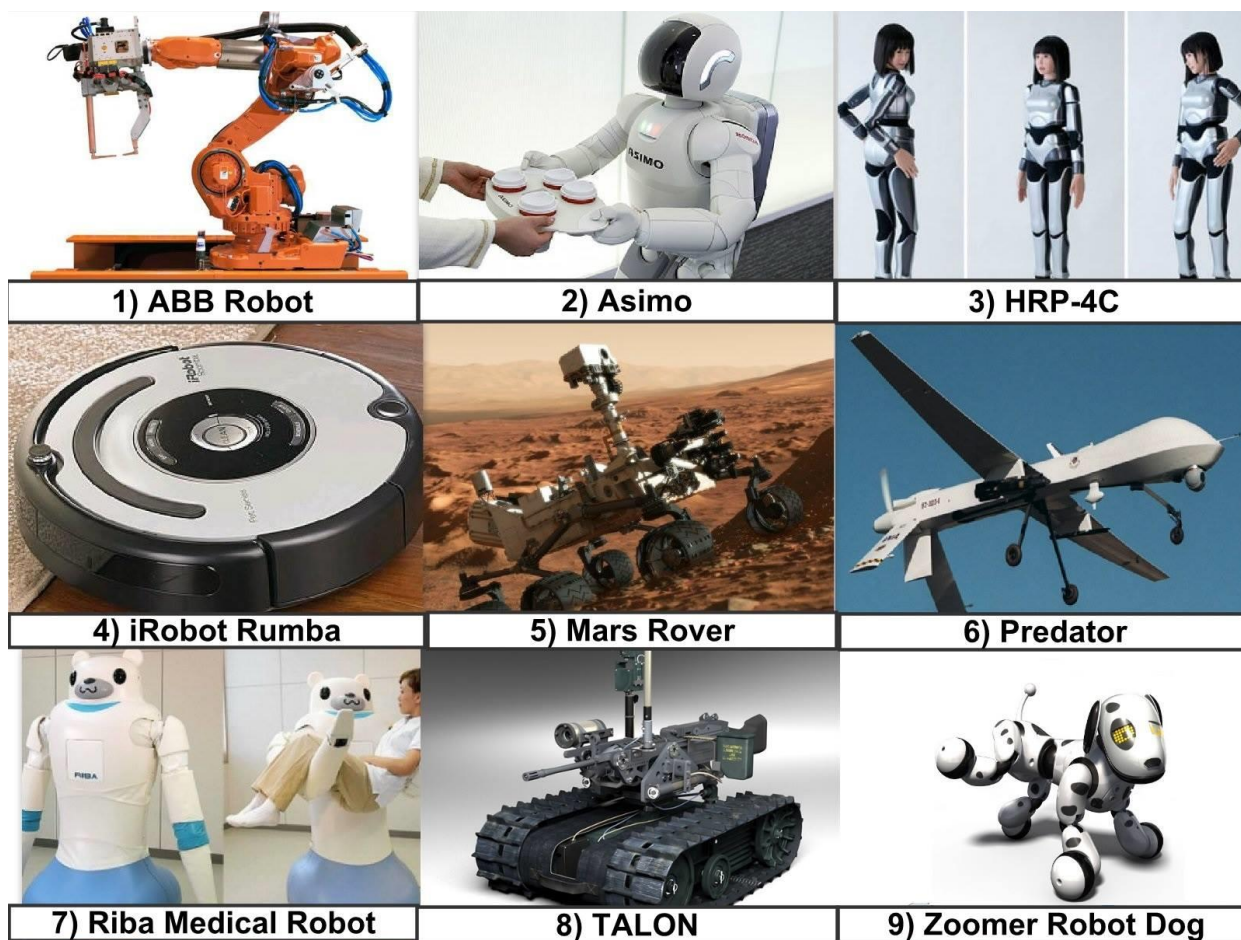
Lego Mindstorms set je idealna podloga za upoznavanje s ovim tehnologijama. Zatim je opisan autić, mehanizmi koji ga pokreću i kako je bilo slagati ga, te izrada koda u Lego EV3 programu i algoritmi koji omogućavaju autiću da se sam kreće prostorom i putem izbjegava prepreke do kojih dolazi.

4. Općenito o robotici

Robotika je interdisciplinarno znanstveno područje koje se bavi projektiranjem, konstruiranjem, upravljanjem i primjenom robota. Glavno područje na kojem se robotika bazira je mehatronika, iako koristi znanja iz mnogih drugih područja poput elektrotehnike, automatike, računarstva, strojarstva, te najzanimljivije od svega koristi znanja i alate umjetne inteligencije [1].

Roboti su definirani kao automatizirani strojevi višestruke namjene, te postoje određene komponente koje svaki robot mora imati. To su uređaji koji pokreću samog robota (motori), uređaji pomoću kojih robot prikuplja informacije iz okoline (senzori) i najbitnija komponenta svakog robota; upravljački uređaj.

Postoji više načina na koje možemo kategorizirati robote. Po stupnju pokretljivosti (mogu biti statički i mobilni), po strukturi konstrukcije (mogu biti biotronički, bioroboti ili mehatronički), po namjeni (od industrijskih, medicinskih, edukacijskih, vojnih, osobnih...), i po veličini (makroroboti, mikroroboti i nanoroboti).



Slika 1: Različite vrste robota

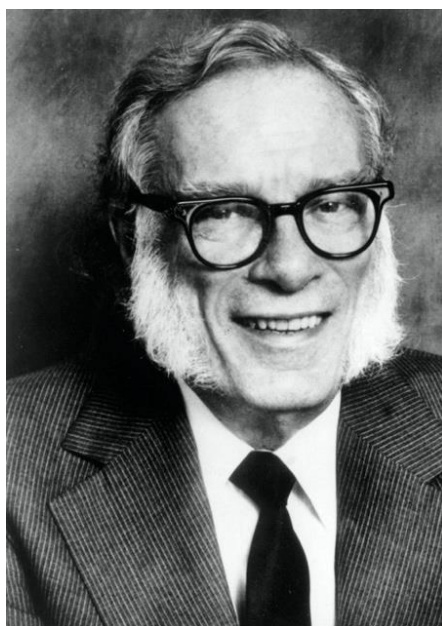
Na slici 1 prikazani su različiti primjeri robota od robotkse ruke, čovjekolikih robota - androida poput Asima do autonomnog tenka TALON i robotskog psa Zoomer. Cilj robotike je napraviti strojeve koji imaju sposobnosti donošenja vlastitih zaključaka baziranih na informacijama dobivenih iz okoline [2]. Da bi u ovome bili uspješni moraju imati visok stupanj mobilne, organizacijske i funkcionalne autonomnosti. Najviši nivo ovoga vidi se u biorobotima koji su razvijani uz genetičko inženjerstvo. Kako tehnologija napreduje i kako se umjetna inteligencija razvija, pretpostavlja se da će ovi roboti (vjerojatno prije nego očekivano) biti jednako inteligentni, ako ne i inteligentniji od ljudi [9].

4.1. Kratka povijest

Prve zabilježene ideje o robotima su se počele pojavljivati još davnog 15. stoljeća kada se razmišljalo o robotu u obliku lava koji bi se samostalno kretao bez ljudske pomoći. Ovu ideju i crteže je među svojim brojnim idejama izmislio veliki Leonardo da Vinci [3].

Riječ "robot" prvi put je iskoristio poznati češki dramatičar Karel Capek. Zapravo je koristio riječ "robotnik", koja se sa Češkog može prevesti kao "radnik" ili "rob" [3]. Njegova ideja robota bila je čovjekoliki stroj koji je mogao razmišljati za sebe i donositi razumne odluke.

Capek je nastupao u Pragu 1921. godine, gdje je prvi put uveo ovu riječ u svojoj reprezentaciji drame Rossum's Universal Robots [3]. Ova predstava je predstavljala njegovu zamišljenu verziju budućnosti u kojoj robotski strojevi rade sve što ljudi pože. Počevši s mnogim prednostima i pogodnostima, ali kasnije isto tako i manama, poput problema nezaposlenosti i filozofskih društvenih pitanja.



Slika 2: Isaac Asimov

Riječ "robotika" se prvi put spominje u kratkoj priči naziva Runaround, koju je objavio Isaac Asimov (slika 2) 1946. godine. Među prvim robotima o kojima je Asimov pisao bio je robotski terapeut. 20 godina kasnije, (1966.) profesor iz Instituta za tehnologiju Massachusettsa, Joseph Weizenbaum napisao je Eliza program kao suvremenu verziju Asimovog izmišljenog karaktera. Weizenbaum je programirao Elizu kao psihoterapeuta koji je na postavljena pitanja odgovarao s više pitanja iz konteksta. Ovo je tada bilo impresivno jer je Eliza u početku imala samo 240 linija koda i mogla je obaviti jednostavne razgovore s pacijentom.

Asimov je osim ovoga napravio nešto mnogo značajnije za današnju robotiku i problem umjetne inteligencije, a to su četiri zakona (pravila) ponašanja robota. Ovo su programski zakoni koje svi roboti moraju slušati, te samim time predstavljaju temeljni dio robotskog inženjerstva. Asimova prva priča koja je uvela prva tri zakona bila je spomenuta drama Runaround, koja se pojavila u jednom od izdanja časopisa Astounding Science Fiction iz ožujka 1942. godine. Uz glavna tri zakona postoji i "nulti" Zeroth zakon. Za razliku od prva tri, Zeroth zakon nije temeljni dio robotskog inženjerstva i nije dio današnjih robota pošto je za njegovu implementaciju potrebna visoka razina umjetne inteligencije i sofisticiranost samog robota [3].

Ovi zakoni su:

1. Zeroth zakon: Robot ne smije naškoditi čovječanstvu ili, zbog neaktivnosti, dopustiti čovječanstvu da sebi nanese štetu.
2. Prvi zakon: Robot ne smije naškoditi ljudskom biću ili, zbog neaktivnosti, dopustiti ljudskom biću da bude oštećeno, osim ako to ne krši zakon višeg reda.
3. Drugi zakon: Robot mora poštivati svaku zapovijed koju mu daju ljudska bića, osim u slučajevima kada bi takva naredba kršila zakon višeg reda.
4. Treći zakon: Robot mora štiti svoje postojanje sve dok takva zaštita ne krši zakon višeg reda.

Godine 1956. dolazi do susreta Georgea Devola i Josepha Engelbergera koji dogovaraju da će zajedno raditi na stvaranju novog robota vodeći računa o postulatima Isaaca Asimova. Tako je došlo do robota naziva Unimate, čija svrha je bila raditi sa strojevima za lijevanje u tvornici General Motorsa. Engelberger je kasnije započeo vlastitu tvrtku za proizvodnju robota naziva Unimation, koja se razvila u prvu komercijalnu tvrtku ove vrste.

4.2. Zašto je robotika važna?

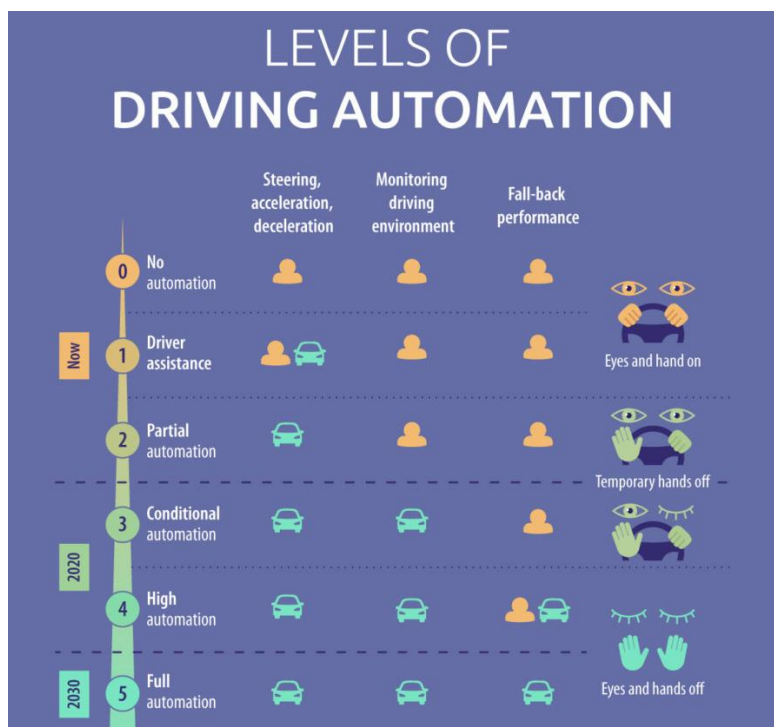
Glavna funkcija robota je raditi razne poslove kako bi ljudima što više olakšali svakodnevicu. Ovo može biti bilo što, od poslova koji su ljudima preteški, opasni, poslova u kojima je potrebna preciznost koju čovjek ne može postići, ili jednostavno poslova koji su ljudima dosadni, te ih roboti rade mnogo efikasnije nego mi [2]. Okoline opasne za ljude mogu biti raznolike, od ekstremnih podmorskih dubina i spuštanja u kratere vulkana, sve do popravljanja svemirskih letjelica i robota za istraživanje površina drugih planeta (npr. Mars Rover Curiosity). Naravno, postoji i mnogo vojnih svrha u kojima su roboti bolja opcija od ljudi, poput robota za razminiranje terena, robotskih tenkova, autonomnih podmornica, letjelica, dronova te mnoštvo drugih primjera [1].

Danas su najzastupljeniji industrijski roboti, što nije iznenađujuće pošto su to zapravo prvi roboti koje je čovječanstvo napravilo. Nakon otkrića parnog stroja i industrijske revolucije, industrija je naglo postala automatizirana jednostavnim strojevima koji znatno ubrzavaju proces u usporedbi s ljudskim radom. Primjerice automobilska industrija, gdje roboti sastavljaju dijelove vozila, bojažu aute, zavaruju karoserije i slično. Zbog svega ovoga dolazi i do raznih problema, najviše što se tiče radnih mjesta. Dok su roboti u većini poslova efikasniji, precizniji, brži i općenito bolji od ljudi, najvažnije od svega je što su jeftiniji. Ulaganje u robota za obavljanje jednostavnih rutinskih poslova je u mnogo slučajeva puno jeftinija opcija nego imati zaposlenog radnika.

Koliko god robotika bila među najzanimljivijim granama današnje tehnologije i prikaz moći modernog čovjeka, ovo stvara sve veći i veći problem. Naime roboti postaju sve napredniji, samostalnjiji i inteligentniji, sve korisniji za upotrebu, i zbog sve većeg broja robota u primjeni i proizvodnji sve jeftiniji. Što znači da kako se tehnologija razvija, roboti će postajati sve pristupačniji raznim manjim poslovanjima i firmama i upotreba će se još više proširiti te će moći obavljati sve više radnih zadataka, pogotovo onih repetitivnih. Kako roboti postaju pametniji, moći će zamijeniti sve više raznih pozicija, ne samo poslove po uzoru na ljudi u pogonima i na pokretnim trakama tako da će i drugi radnici imati razloga za zabrinutost. U današnje vrijeme, postoje roboti koji mogu zamijeniti sve od blagajnika, do velikih robotskih "Roomba" koje zamjenjuju čistače po trgovačkim centrima, njegovatelje u bolnicama, autonomna vozila, vojna vozila na bojištima i slično.

4.3. Autonomna vozila

Potpuno autonomna vozila, točnije vozila koja bi mogla raditi bez vozača, još nisu naša svakidašnjica ali mogla bi postati naša skora budućnost. Iako se danas takva vozila uglavnom koriste u testnoj fazi [10], nije daleko vrijeme kada će takva vozila postojati na našim prometnicama. Google-ov projekt Waymo [11] je najbliži ovom cilju, s time da postoji jedan glavni problem veći od tehnoloških ograničenja. To je etičko i pravno pitanje koji dolazi u slučaju nesreće, a ono glasi "Tko snosi posljedice kada dođe do sudara u kojem je vozilo krivo za sudar?" Unatoč tome, toliko je sigurno njihovo postojanje, da već postoje jasno propisana pravila koja ih klasificiraju u jednu od kategorija uz današnje automobile. Postoje jasno definirani stupnjevi autonomije, te zbog ovoga današnji proizvođači automobila ne nazivaju svoje proizvode "autonomnima", već im određuju kategoriju u koju pripadaju. Ovih šest stupnjeva definirano je Društvo automobilskih inženjera (engl. SAE, Society of Automotive Engineers) [4]. Ovi stupnjevi opisuju koje funkcije autonomni sustav mora preuzeti od vozača, ali ne i kako će vozilo te funkcije ostvariti (slika 3). Ovo znači da korištenje različitih načina prepoznavanja okoline ostaje na izbor proizvođača. Na primjer, neka tvrtka će za detekciju drugih vozila i pješaka koristiti radar, dok će neka druga koristiti lidar, video kamere ili neku drugu tehnologiju. Ovo ne utječe na stupanj autonomije koji će se dodijeliti vozilu već je jedino bitno konačna sposobnost vozila u vožnji. Prema tome, većina današnjih vozila koja dolaze s autopilotom i autonomnim funkcijama su razine 2, u rijetkim slučajevima 3, dok bi potpuna autonomnost pripadala 5. razini.



Slika 3: Šest stupnjeva autonomnosti

4.3.1. Razine autonomije

Postoji šest razina, to jest stupnjeva autonomije [4]. Oni su:

Level 0 (razina 0) - Ova razina znači da vozilo nema nikakvu automatizaciju tijekom vožnje. Ovdje spada većina automobila proizvedena danas. Iako ovi automobili mogu imati funkcije poput ABS-a (assistive breaking system), uphill assist-a, automatske ručne kočnice i slično, ove tehnologije zapravo nisu svjesne okoline nego reagiraju na jednostavne ulazne podatke koji dolaze od vozača ili od samog vozila. Koliko god ove tehnologije pomažu vozaču kod ugodnosti i sigurnosti, i dalje su svi elementi same vožnje kontrolirani od strane vozača. Automatsko kočenje u slučaju nužde je akcija koja se binarno aktivira ako vozilo detektira prepreku, te zbog toga također pripada u ovu kategoriju pošto sustav aktivno ne upravlja vozilom.

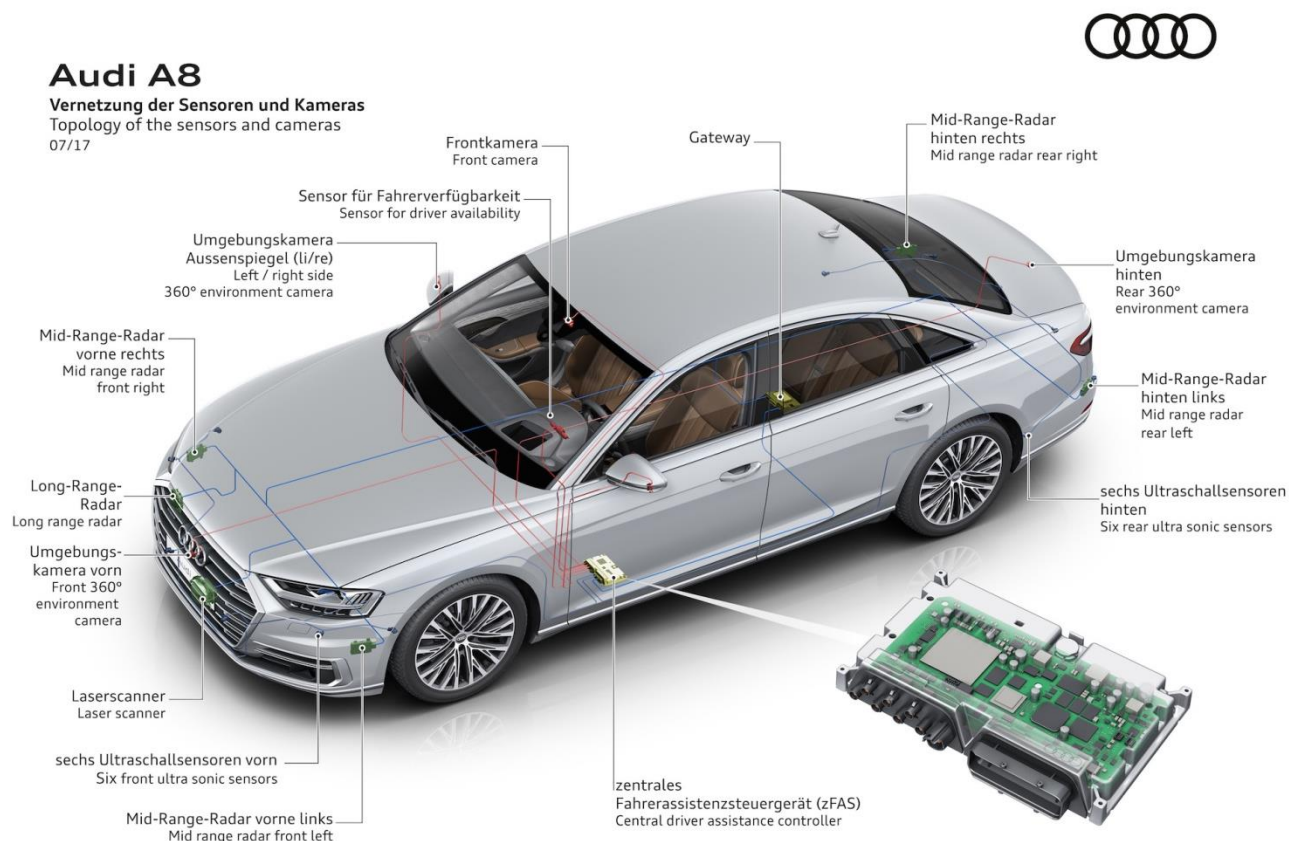
Level 1 (razina 1) - Ovdje dolazi do prvih sustava koji u određenim trenucima mogu vozaču preuzeti ili, ovisno o situaciji, oduzeti kontrolu nad vozilom. Na primjer, u ovo ubrajamo adaptivnu kontrolu brzine (napredniju i pametniju verziju tempomata). Pomoću ovog sustava vozilo je u stanju aktivno prilagođavati brzinu, te samo održavati siguran razmak od vozila ispred sebe. U ovom slučaju vozač i dalje upravlja volanom i svjesno prati okolinu.

Level 2 (razina 2) - Ovo je djelomična automatizacija vožnje. Za razliku od prve razine, u ovom slučaju vozilo je sposobno samo ubrzavati ili usporavati kako bi održalo siguran razmak od vozila ispred sebe, te istovremeno kontrolirati volan i držati se u prometnoj traci u kojoj se nalazi. Vozač u bilo kojem trenutku može preuzeti kontrolu i vozilo će samo isključiti automatizaciju. Teslin Autopilot i General Motorsov Super Cruise oba pripadaju u ovu razinu automatizacije, koju SAE službeno definira kao djelomična automatizacija.

Level 3 (razina 3) - Ova razina se naziva uvjetnom automatizacijom. Ova razina, isto kao sve koje slijede nakon nje, podrazumijeva da automatski sustavi prepoznaju i reagiraju na okolinu u kojoj se vozilo nalazi. Automobili moraju reagirati na prometnice, druga vozila, pješake, znakove, te ostale objekte i prepreke do kojih dolazi u okolini. Iako iz perspektive vozača nema puno razlike između ove i prijašnje razine, s tehnološke strane vozila radi se o mnogo složenijem procesu. Uz sve što mora raditi vozilo druge razine, vozilo treće razine može preuzeti kontrolu i prelaziti druga vozila koja se sporo kreću. Ovo radi na isti način kao i vozač, sposobno je dati žmigavac, provjeriti je li sigurno prestrojiti se u drugu traku, te se prestrojava, ubrzava i istim postupkom vraća u originalnu traku. Prvi automobil ove razine je Audijev A8L sa svojim Traffic Jam Pilot sustavom. Vozač i dalje u svakom trenutku može preuzeti kontrolu.

Level 4 (razina 4) - Ovo je prva visoka razina automatiziranosti, to jest početak "potpune autonomije". Vozila ove razine sadrže sustave koji su u stanju detaljno prepoznavati okolinu, sasvim samostalno voziti bez pomoći vozača te čak intervenirati ako nešto u vožnji pođe po zlu (ako pješak ili neka životinja iskoče na cestu, ili ako nešto u samom automobilu prestane raditi kako treba). Ova vozila i dalje imaju ugrađene ručne kontrole kako bi vozač mogao preuzeti kontrolu u bilo kojem trenutku. Ova vozila su još uvijek u probnoj fazi, ali postoje za kratke rute komercijalnog prijevoza, na primjer ruta od aerodroma do hotela. Zbog neusavršenih sustava ovih vozila, brzina im je ograničena na 40-50km/h, te im je sama autonomija ograničena na male geografske prostore, što se na engleskom zove "geofencing".

Level 5 (razina 5) - Ovo bi u teoriji bila potpuna automatizacija vožnje. Ovakva vozila bi bila sposobna preuzeti sve odgovornosti vozača, te samim time ne bi imala potrebu ni za vozačem, ni za ručnim kontrolama unutar automobila. Moraju biti u sposobna u bilo kojim uvjetima, u bilo koje doba dana ili noći, potpuno samostalno odraditi put od bilo koje zadane točke do druge. Ova vozila se već danas testiraju diljem svijeta, ali ih nije moguće kupiti serijskoj proizvodnji. Ljudi poput Elona Muska kažu da će ova vozila biti dostupna javnosti već za 5 godina, no vjerojatno i više.



Slika 4: Brojni senzori i kamere na automobilu Audi A8

4.3.2. Problemi autonomije

Postoji mnogo problema koji još nisu razriješeni što se tiče automatizacije automobila. Najveći takav problem je činjenica da se čak i ljudi teško snalaze u nepredvidivim situacijama u prometu, a roboti još teže. Autonomna vozila koliko god bila napredna u tehnologiji koju koriste, i koliko god senzora imala, i dalje nisu dovoljno pametna da reagiraju na okolinu s inteligencijom čovjeka. Jedan ovakav primjer se dogodio u Argo [5]. Argo je zajednički projekt Volkswagena i Forda, u kojem je došlo do problema kada je nepredvidivi biciklist vozio slalom između vozila na cesti. Do drugog problema je došlo kada se kamion za odvoz smeća iznenadno i nelogično polukružno okrenuo na raskrižju u Pittsburghu. Autonomno vozilo je sve ove manevre uspješno prepoznalo i uočilo, ali nije moglo predvidjeti sljedeće poteze biciklista i vozača. Iz Arga su rekli kako je 80% softvera i hardvera za autonomna vozila već razvijeno, ali kažu da je zadnjih 20% najteže. Jedan od važnijih događaja koji je inženjere vratio na osnovne koncepte i postavke, bila je nesreća Uberovog autonomnog vozila [5]. U Arizoni, automobil je pogazio biciklisticu koja je sasvim pravilno prelazila cestu; prišla je pješačkom prijelazu, sišla s bicikla i pješice stupila na cestu. Vozačica Ubera nije obraćala pažnju na vožnju, pošto je automobil sam upravljao. Uz ovakve greške na osnovnim stvarima koje automobil mora moći raditi, još je teže programirati ga da se snalazi u nepredvidivim situacijama. Poput na primjer, pješaka koji trči preko semafora dok je upaljeno zeleno svjetlo ili vožnja iza vozila koje se sporo kreće i traži parkirno mjesto. Ako je u pitanju mjesto gdje vozilo ispred planira bočno parkirati, automobil mu neće ostaviti dovoljno mjesta za ići unazad. Ima još mnogo velikih problema kod autonomne vožnje, ali problemi kod ovakvih sitnica nam zapravo govore koliko još posla čeka inženjere, te da je aktivna samostalna vožnja ipak prevelika odgovornost koju osobna vozila ne mogu samo tako preuzeti.

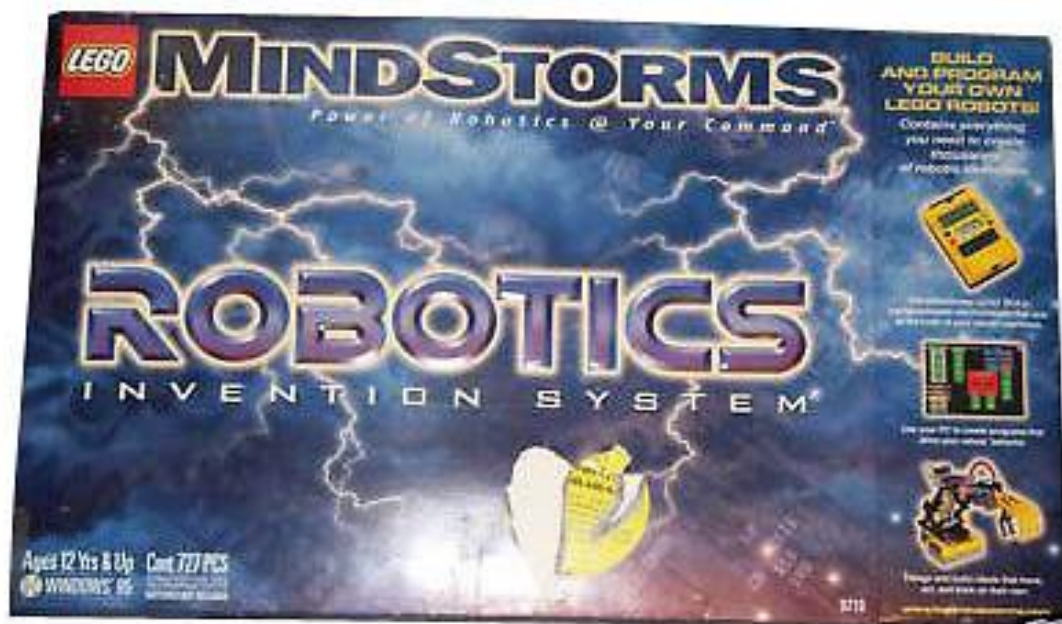
5. Lego Mindstorms EV3

Ovaj Lego Mindstorms EV3 set za učenje robotike sadrži sve potrebne elemente za kreiranje tisuća raznih projekata i robota [6]. Uz sve potrebne dijelove za slaganje, Lego tim je napravio i aplikaciju za pojednostavljeno programiranje napravljenih robota. Postoji računalna i mobilna verzija, te posebna verzija za tablete, koja omogućuje djeci i odraslima učenje osnova robotike. Aplikacije su bazirane na objektno orijentiranom programiranju te funkcioniraju slično programu Scratch, što znači da osoba ne mora znati pisati kod, već je dovoljno naučiti koji blok u programi što radi, i logički ih pravilno povezati.



Slika 5: Lego Mindstorms EV3 set

Ovo je četvrta generacija seta u nizu Lego projekata koji su napravljeni u ovu svrhu. Prva generacija je bio originalni Robotics Invention System, koji je produkt suradnje Lego tima s Massachusetts Institutom za Tehnologiju (Massachusetts Institute of Technology, MIT) [7]. Godine 1985., CEO Lego grupe Kjeld Kirk Kristiansen je stupio u kontakt sa MIT-om nakon što je vidio da imaju veoma slične ideje o učenju kroz izradu robota. Kako bi ovi roboti bili pristupačni djeci, došli su do proizvoda koji je bio kontroliran programskim jezikom LOGO. Nakon što su postigli bežično kontroliranje robota putem računala, Lego je što prije htio komercijalizirati ovu ideju. Kako su osobna računala postajala sve popularnija, Lego tim je tek 1996. godine izdao prvi Robotics Invention System [7]. No, zbog problema kod programskog jezika LOGO i činjenice da je program ipak bio prekompleksan da bi ga savladala ciljana djeca u rasponu od 10 do 14 godina, Lego tim je morao razviti vlastiti programski jezik. Zbog ovoga su propustili originalni datum izlaska i izdali gotov proizvod tek 1998., 2 godine kasnije nego planirano. Nakon ovoga 2006. godine, 8 godina kasnije Lego je kreirao prvi NXT set koji je sadržavao 3 motora, 4 senzora (ultrasonični, zvučni, dodirni i svjetlosni) i NXT programski Intelligent Brick. Ovo je prva verzija koja je napravila veliki korak, te u kojoj vidimo mnoge sličnosti s današnjim EV3 setom. Lego je u ovoj generaciji prvi puta izdao program NXT-G koji je imao grafičko sučelje, također slično današnjem EV3 načinu rada. Sljedeća (treća) generacija u nizu je izašla 2009. godine, pod nazivom NXT 2.0, koji je bio poboljšanje NXT-a u svakom pogledu. Ovdje je glavna razlika ta da su senzori koristili Floating Point operacije za dobivene informacije, za razliku dotadašnjih cijelih brojeva u obliku Integera [7]. Zatim, 2013. Lego izdaje trenutno posljednju generaciju svoje robotske ideje pod nazivom Lego Mindstorms EV3, koji se i danas koristi diljem svijeta.



Slika 6: Lego Robotics Innovation System 2.0 set

5.1. Senzori

Postoji pet različitih vrsta senzora koje je Lego kroz godine stavljao u svoje Mindstorms EV3 setove. Prijašnji setovi su uključivali ultrasonični senzor, koji je koristio odbijanje zvuka visoke frekvencije (toliko visoke da ga ljudi ne mogu čuti) za mjerenje udaljenosti od objekata. Ovaj senzor je s vremenom zamijenjen s infracrvenim senzorom koji je uključen u današnje setove, uz senzor za boje i senzor za dodir. Također je moguće zasebno kupiti Gyro senzor koji mjeri nagib samo po jednoj osi.

- Infracrveni senzor - Ovo je glavni senzor za svakog robota. Ovo je digitalni senzor koji može detektirati infracrvene svjetlosne signale. Ovi signali mogu biti reflektirani od solidnih objekata ili poslani od strane Remote Infrared Beacons (LEGO daljinski upravljač dobiven u setu). Ovaj senzor ima tri načina rada između kojih možemo birati.
 - Proximity mode - Senzor koristi svjetlosne valove reflektirane od nekog objekta za procjenu udaljenosti između senzora i tog objekta. Zatim vraća vrijednost računalu koja može biti od 0 do 100 (ova vrijednost nije u centimetrima već postotak maksimalne udaljenosti). Može detektirati predmete udaljene do 70cm, ovisno o veličini i obliku objekta.
 - Beacon mode - Ako na Remote Infrared Beaconu odaberemo jedan od 4 ponuđena infracrvena kanala, on ih može kontinuirano prenositi dok je upaljen. Senzor može prepoznati ove signale do 200cm udaljenosti u smjeru prema kojem je senzor okrenut. Kada je signal detektiran, senzor može procijeniti iz kojeg smjera signal dolazi i koliko je otprilike udaljenost od senzora do Beaconsa.
 - Remote mode - Ovo robotu omogućuje ručni način rada, to jest kontroliranje robota putem Remote Infrared Beaconsa (daljinskog upravljača). U ovom načinu senzor može detektirati koji gumb ili koja kombinacija gumba je trenutno pritisnuta. Sveukupno postoji 11 ovakvih kombinacija, plus jedna nulta ako se ništa ne prenosi. 4 zasebno za svaki gumb, 6 kombinacija po dva gumba u isto vrijeme i ako je Beacon u načinu rada da odašilje jedan od prijašnje spomenutih infracrvenih kanala.
- Senzor za boje - Ovo je također digitalni senzor koji umjesto infracrvenog svjetla može detektirati boje ili intenzitet svjetla u prostoru. Ovaj senzor također ima tri načina rada:
 - Color Mode - U ovom načinu senzor ima mogućnost prepoznavanja sedam boja: crnu, bijelu, plavu, zelenu, žutu, crvenu i smeđu. Također može prepoznati ako nije nijedna od prepoznatljivih boja (No Color). Koristeći ovaj senzor roboti mogu sortirati kockice raznih boja, pustiti određeni zvuk (ili naziv boje) ovisno koja boja je detektirana ili primjerice zaustaviti neku akciju ako je detektirana crvena boja.

- Reflected Light Intensity Mode - Senzor za boje na sebi ima malu lampu koja odašilje crvenu svjetlost. Ovaj način rada omogućuje robotu mjerenje intenziteta te crvene svjetlosti reflektirane od nekog objekta. Ovo znači da se može napraviti robot koji je programiran kretati se bijelom površinom dok ne dođe do crne linije, ili robot koji prepoznaje bojom kodirane identifikacijske kartice. Senzor u ovom slučaju koristi skalu od 0 do 100, gdje 0 predstavlja jako tamne objekte, a 100 jako svjetle.
- Ambient Light Intensity Mode - Ovaj način rada mjeri snagu svjetlosti koja ulazi u prostoriju od nekog vanjskog izvora. Ovo može biti sunčeva svjetlost, svjetlost ručne baterije i slično. Senzor također koristi skalu od 0 do 100 isto kao u prijašnjem slučaju. Ovo znači da je moguće napraviti robota koji će po svjetlosti u prostoriji prepoznati da je izašlo sunce i ozvučiti alarm, ili zaustaviti rad kada se ugase svjetla u prostoriji.
- Senzor za dodir - Ovo je najjednostavniji senzor u kutiji i jedini koji nije digitalan. Ovaj analogni senzor može detektirati kada je gumb na njemu pritisnut ili pušten. Zbog ovoga može programu vratiti 3 stanja. Ova stanja su: senzor je pritisnut, senzor je pušten, i senzor je "bumped" (na hrvatskom udaren). Bumped zapravo znači da je senzor brzo pritisnut i pušten. Koristeći ovaj senzor može se napraviti robot koji se kreće unaprijed i reagira kada je senzor pritisnut, te se zaustavlja ili kreće unatrag.



Gyro Sensor



Touch Sensor



Infrared Sensor



Infrared Beacon



Colour Sensor



Ultrasonic Sensor

Slika 7: Pet Mindstorms senzora i Infrared Beacon

5.2. Motori

U setu se nalaze 3 motora koji služe za kretanje robota kroz prostor, podizanje, spuštanje stvari, ili bilo koje druge mehaničke pokrete koje robot mora raditi. Postoje dvije vrste motora, Large Motor (veliki motor) i Medium Motor (srednji motor) [7]. Veliki motori, kada ih postavimo na 100% snage, rade otprilike na 160-170 okretaja u minuti, s okretnim momentom od 20 Ncm (Newton centimetara), i zaustavnim momentom od 40 Ncm. Dok srednji motor radi na 240-250 okretaja u minuti, uz okretni momenat od 8 Ncm, i zaustavnim momentom od 12 Ncm. Što znači da su veliki motori jači ali sporiji i manje precizni, dok je srednji motor brži i precizniji, ali dosta slabiji.

Svi motori imaju razne načine rada na koje mogu raditi. Svaki blok koda koji kontrolira motore može se postaviti na jedan od pet načina rada. Dva osnovna su Uključen i Isključen. Uključen ima samo jedan parametar koji kontrolira s kojim postotkom snage će motor raditi dok je u ovom stanju (0-100%). Isključen ima samo jedan parametar koji se zove "Break at end" (Kočenje na kraju), koji je tipa bool. Ako je ovaj parametar postavljen na 1 motor će kočiti, ako je postavljen na 0 motor neće kočiti već će pustiti da se auto nastavi kretati po inerciji dok se sam ne zaustavi.

Ostala tri načina rada su:

1. "On for rotations", kada je potrebno da motor napravi određen broj okretaja.
2. "On for degrees" koji se koristi kada je motor potrebno okrenuti određen broj stupnjeva, može biti do 0 do 360 za jedan okretaj, ili više.
3. "On for seconds", kada je potrebno da motor bude uključen određen broj sekundi. Oni se koriste ovisno o situaciji.

Svaki od ovih načina rada ima parametre za podešavanje snage motora i "Break at end" opciju, koji funkcioniraju isto kao u Uključenom i Isključenom načinu.



Slika 8: Large Motor

U set su uključena dva velika motora koji služe za teže robotske funkcije. Ovi motori se većinom koriste za kretanje robota kroz prostor, to jest prijenos snage na kotače, trake, noge, ili neki drugi mehanizam kretanja. Ovi motori imaju više načina rada, i svaki ima svoj zasebni blok. U EV3 programu postoje 3 tipa blokova za aktivaciju motora, koji određuju kako će se motori međusobno koordinirati.

Prvi tip bloka se zove Large Motor, i koristi se za kontrolu samo jednog motora. Na primjer ako je u pitanju robot koji stoji uspravno i potrebno je da odvojeno pomiče jednu po jednu nogu, stavili bi dva ovakva bloka zasebno za svaki motor. Drugi tip se zove Move Steering i služi za pokretanje oba motora istovremeno. Na primjer ako je u pitanju automobil sa zadnjim pogonom, ovaj blok će osigurati da mu se lijevi i desni zadnji kotač okreću jednakom brzinom. Ovaj blok sadrži i dodatnu varijablu koja se koristi za skretanje auta. Može se podesiti na pozitivnu vrijednost za skretanje udesno ili negativnu za skretanje ulijevo, što automatski programu govori da mora prilagoditi brzine motora. Treći tip se zove Move Tank i služi za pokretanje motora asinkrono ili u suprotnim smjerovima. Ovaj blok ima dvije varijable koje kontroliraju snagu svakog motora, te ih možemo postaviti na različite vrijednosti ako želimo da se trake pokreću različitim brzinama ili u suprotnim smjerovima. Primjerice, ako je u pitanju robotski tenk. Takav tenk bi imao po jedan motor na lijevoj i desnoj traci, te motori ne bi mogli cijelo vrijeme raditi sinkronizirano jer onda tenk ne bi mogao promijeniti smjer kretanja. Ovo je moguće postići s ovim blokom, koji omogućuje odvojeno variranje brzine desne i lijeve trake, ili čak pokretanje motora u suprotnim smjerovima kako bi se tenk rotirao u mjestu.



Slika 9: Lego Mindstorms složen u obliku tenka

Treći motor uključen u set je srednji motor, koji za razliku od velikih motora ima ugrađen senzor za rotaciju koji može detektirati okretaje s preciznošću do jednog stupnja. Osim ovoga, pošto je manji, ovaj motor može brže reagirati na naredbe od velikih motora. Ima istih 5 načina rada kao i veliki motori, ali za razliku od njihova 3 programska bloka, srednji motor ima samo jedan blok za kontrolu koji se zove Medium Motor. Ima iste parametre kao i veliki motori, snaga, koliko dugo će biti uključen (s istom podesivosti po rotacijama, stupnjevima ili sekundama), te "Break at end" opciju.



Slika 10: Medium Motor

5.3. Programski Brick

Programski brick, ili kako ga Lego naziva "Intelligent Brick", je najvažniji dio cijelog seta. Ova velika lego kocka je zapravo mini računalo koje kontrolira cijelog robota. Na vrhu ima mali crno bijeli ekran rezolucije 178x128 piksela koji prikazuje korisničko sučelje. Kroz EV3 program ovdje je moguće dodavati bilo kakve slike, tekst ili grafiku. Na primjer, robot može ovim putem dobiti lice, sretno ili tužno, ili pak prikazati rezultat neke matematičke operacije. Na gornjoj strani Bricka se također nalaze i gumbi koji služe za navigaciju sučeljem. Odmah ispod ekrana se nalazi Back gumb koji služi za vraćanje natrag u izbornicima, prekid programa koji se trenutno izvodi ili gašenje samog Bricka. Ispod njega nalaze se četiri gumba za kretanje izbornicima: gore (Up), dolje (Down), lijevo (Left) i desno (Right). Ovi gumbi se mogu programirati i kao aktivatori određenih funkcija dok je pokrenut neki program, na primjer da robot pritiskom gumba Up podigne ruke, a pritiskom gumba Down ih spusti. Zadnji gumb na Bricku je središnji OK ili Center gumb. On koristi za odgovaranje na pitanja poput "Želite li zaustaviti program?", odabiranje programa za pokretanje, potvrđivanje postavljenih postavka i dok je robot ugašen za paljenje samog Bricka.

Oko ovih gumba se nalazi Brick Status Light (statusno svjetlo) koje korisniku govori trenutno stanje u kojem se Brick nalazi. Ovo svjetlo može prikazivati tri boje i svaka od njih ima dva stanja, upaljeno i pulsirajuće. Crveno svjetlo znači da je Brick u procesu paljenja, gašenja ili ažuriranja sistema. Pulsirajuće crveno znači da je zauzet nekom operacijom. Narančasto znači pozoran i spreman (Alert, Ready), dok pulsirajuće znači pozoran i aktivan (Alert, Running). Zeleno znači spreman (Ready) za primiti naredbu, a pulsirajuće znači da je program u izvedbi (Running program). Brick također sadrži ugrađeni zvučnik kroz koji je moguće puštati bilo koje zvukove koji su preneseni na Brick i programirani za puštanje putem naredbi.

Na vrhu ekrana je mali status-bar koji prikazuje da li je robotu upaljen Bluetooth ili WiFi, da li je povezan na išta putem Bluetootha ili WiFija (ili USB-a), naziv Bricka koji mu je dodijelio korisnik i postotak baterije. Osim bežično, Brick je moguće povezati računalom putem Mini-USB porta i normalnim USB 1.1 ili 2.0. Putem običnog USB-a, osim s računalom Brick je moguće međusobno povezati s još 3 ista takva Bricka i programirati ih da rade zajedno, što Lego naziva Daisy chain. Pokreće ga operacijski sustav LINUX, te ima radnu memoriju od 64 MB i flash memoriju od 16 MB. U slučaju da ovo nije dovoljno memorije, Brick je opremljen i s utorom za Micro SD karticu, te podržava kartice do 32 GB. Na prednjoj strani se nalaze utori A, B, C i D koji služe kao izlazi Bricka, što znači da se ovdje povezuju motori. Na stražnjoj strani se pak nalaze utori 1, 2, 3 i 4 koji služe kao ulazi, pa su ovdje povezani senzori.

Uz sve ovo što Brick može i mora raditi, još jedan posao mu je i napajanje cijelog robota. Na dnu se nalazi poklopac ispod kojeg je smješteno šest AA baterija. Lego preporuča Alkaline ili punjive Lithium Ion baterije. U setu se dobije i originalna Lego punjiva baterija koju je moguće umetnuti umjesto 6 zasebnih AA baterija, ali se sporije puni, ima manje snage i zauzima više mjesta od klasičnih baterija.



Slika 11: Lego EV3 Intelligent Brick

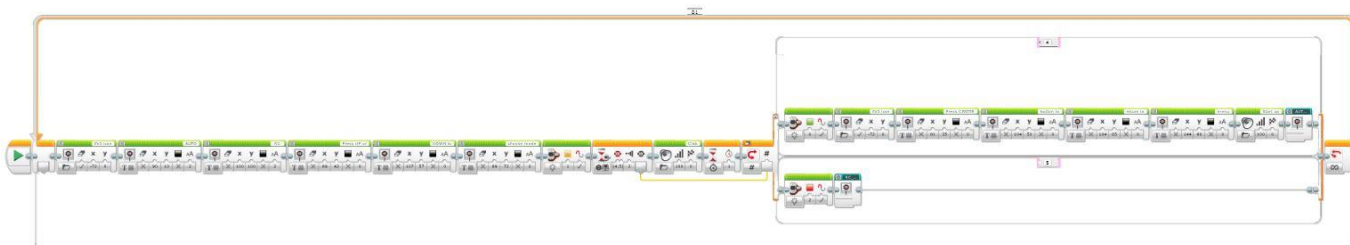
5.4. EV3 program

EV3 program radi na principu blokova (objektno orijentiranog programiranja) tako da nema ručnog pisanja koda. Postoje različite kategorije blokova koji su u programu podjeljni po tabovima. Ovih tabova ima šest i organizirani su po bojama za lakše snalaženje. Po redu kojim su složeni u programu glase:

1. Action (zelena) - Ovaj tab sadrži sve blokove koji pokreću neku akciju. Tu se nalazi blok za puštanje zvukova, sva četiri prijašnje spomenuta bloka za kontroliranje motora (Medium Motor, Large Motor, Move Steering i Move Tank), displaya i za statusno svjetlo oko tipki (Brick Status Light).
2. Flow control (narančasta) - Svi blokovi u ovom tabu kontroliraju tijek programa. Sadrži blokove poput Starta koji označuje početak programa, blok za petlje (Loop), blok za čekanje (Wait), Switch blok i blok za prekid petlje.
3. Sensor (žuta) - Ovaj tab sadrži sve blokove za dobivanje informacija od senzora. Osim kontrola za tri glavna senzora koja imamo u setu ovdje se nalaze i blokovi za detekciju okretaja motora, brojač vremena, blok za prepoznavanje pritiska gumba na Bricku, te blokovi za sve ostale senzore koje je Lego ikad napravio, uključujući i one iz starijih NXT setova.
4. Data operations (crvena) - Ovaj tab sadrži blokove za razne računalne operacije. Neki od primjera su blokovi za varijable i konstante, blok za operacije s poljima, matematički blok (podržava sve osnovne operacije i ima mogućnost upisivanja vlastitih kompleksnijih funkcija), blok za logičke operacije (koji kao rezultat vraća true/false), blok za usporedbu (Compare)...
5. Advanced (plava) - U ovom projektu nema nijednog pojavljivanja bloka iz ovog taba. On sadrži napredne funkcije poput bloka za pristup vanjskoj datoteci, blok za bilježenje podataka, blok za Bluetooth povezanost, blok za inverz okretanja motora i slično...
6. My Blocks (cijan) - Ovo je jedan od najkorisnijih tabova jer omogućava spremanje potprograma kao blok koda. Radi na istom principu kao i vanjske funkcije u bilo kojem programskom jeziku. Ovo drastično smanjuje kompleksnost programa, olakšava snalaženje po projektima i smanjuje redundanciju velikih blokova naredbi.

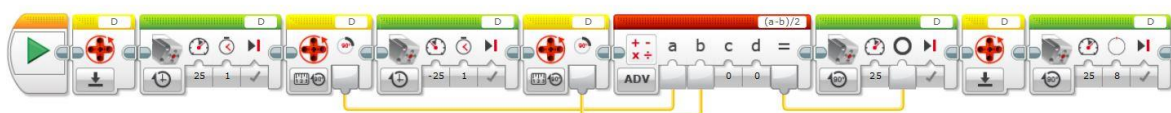
Za ovaj projekt autonomnog autića potrebno je napraviti jedan glavni program, i sedam potprograma. Ovi programi su nazvani INI, Forward, Reverse, Steer, Move, AUTO_MODE, RC_MODE i glavni program koji ih sve obuhvaća naziva AI_Mindstorms_Car. Kada je sve isprogramirano treba povezati Brick sa računalom i na njega skinuti projekt. Ovo se radi klikom na gumb Download u malom EV3 prozorčiću u donjem desnom uglu. Isključimo autić iz računala, stavimo ga na pod i upalimo Brick. Nakon paljenja, gumbima se na Bricku može odabrati i pokrenuti bilo koji od instaliranih programa ili potprograma. Glavni program je onaj koji bi se trebao pokrenuti, u ovom slučaju naziva AI_Mindstorms_Car.

Nakon pokretanja glavnog programa čuje se zvuk koji označuje da je program uspješno pokrenut. Na displayu se prikazuje izbornik na kojem su ponuđene dvije opcije. Tipka gore (Up) služi za biranje automatske vožnje, što pokreće program AUTO_MODE, a tipka dolje (Down) ručno kontroliranje čijim odabirom se pokreće RC_MODE.



Slika 12: AI_Mindstorms_Car program

Slika 12 prikazuje glavni dio programa. Ovaj program na prvu izgleda komplicirano, ali nakon bližeg promatranja zapravo je jako jednostavan. Kao što se vidi na slici 12, većina blokova u ovom programu je zelena, osim velike petlje koja obuhvaća cijeli program i switcha. Treba obratiti pažnju da su na kraju gornjeg i donjeg reda Switcha cijan blokovi koji predstavljaju potprograme AUTO_MODE i RC_MODE. Sve što ovaj program radi je pušta određene zvukove, pali određeni Status Light (narančasti dok se ne odabere opcija) i na displayu prikazuje sve ostale elemente vidljive kod pokretanja programa, uz naravno i sami izbornik za odabir načina rada. Nakon biranja, program prepoznaje koji gumb je stisnut i ulazi u Switch. Ako je odabran automatski, pali se zeleno svjetlo i display ispisuje "Press CENTER button to return to menu.", a ako je odabran ručni rad pali se crveno svjetlo i autić je spreman za akciju.

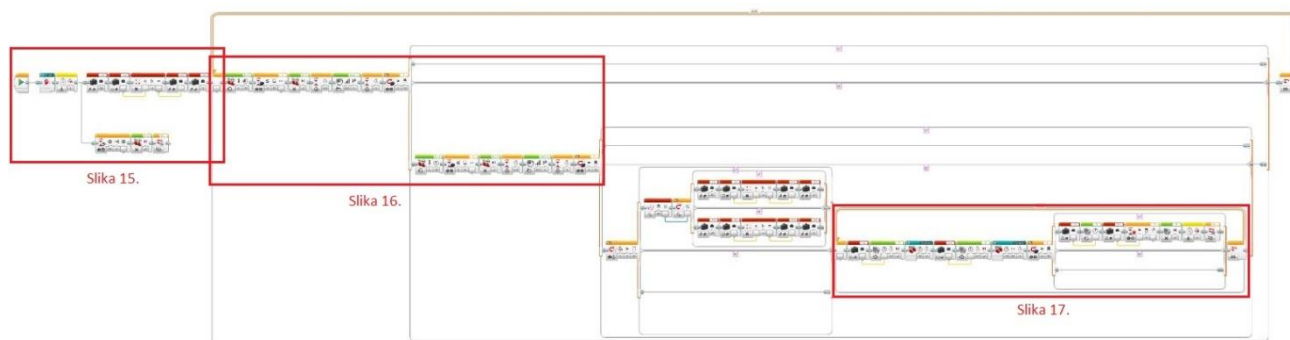


Slika 13: INI program

Slika 13 prikazuje INI program. Ovo je kratica za Initialize (inicijalizirati) i ovaj program se prvi pokreće kod oba načina rada. Sve što on radi je postavlja srednji motor koji služi za skretanje. Ovo čini tako da okrene kotače do kraja u jednom smjeru, zatim u drugom, i na kraju izračuna koliko su okrenuti i u kojem smjeru, te ih pomakne za tu razliku i usmjeri prema naprijed.

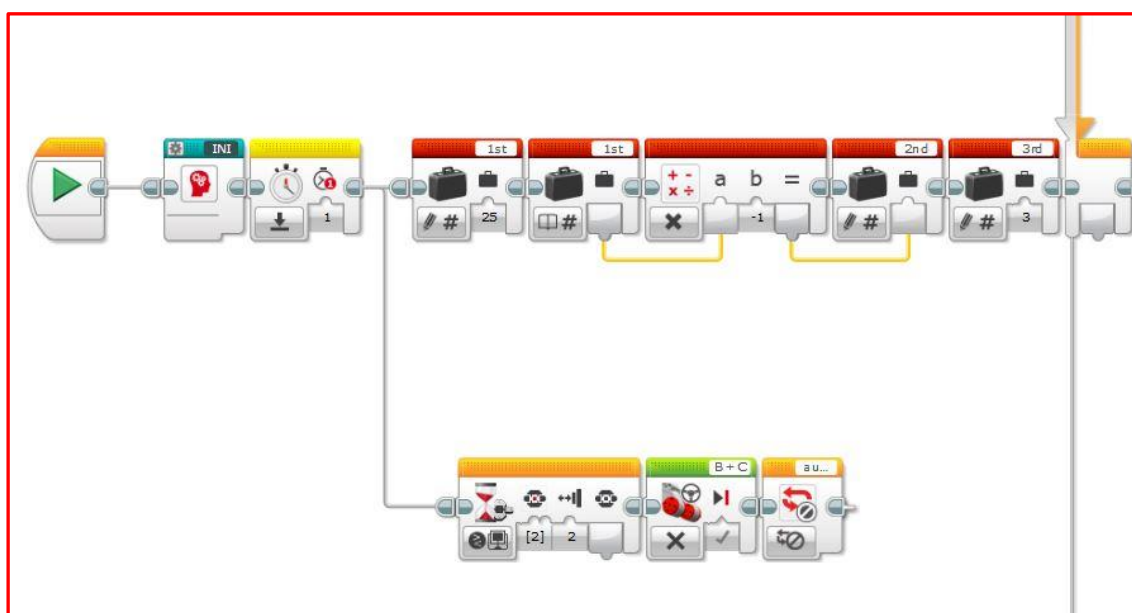
5.4.1. Autonomni način rada

Nakon biranja AUTO_MODE-a autić ulazi u uvjetno rečeno autonomni način rada. Točnije radi se o automatskom načinu koji koristi samo dva potprograma, Forward (za kretanje unaprijed) i Reverse (za kretanje unatrag). Zbog objektnog pristupa programiranju putem blokova, čak i mali programi brzo postaju nepregledni i teško ih je cijele prikazati odjednom na ekranu.

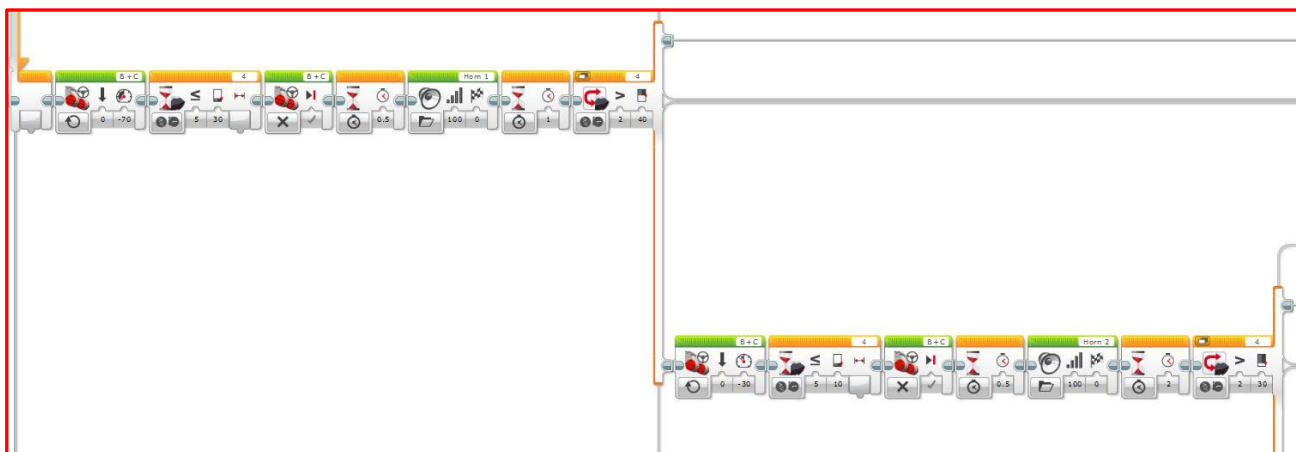


Slika 14: AUTO_MODE program

Glavni dio automatizacije su petlje jer one računalu omogućuju samostalan rad bez vanjskih ulaza. AUTO_MODE se tako sastoji od jedne velike petlje koja obuhvaća sve, 5 switcheva koji odlučuju što auto mora napraviti u određenom trenutku, i jedne manje pomoćne petlje (slika 14). Nakon pokretanja programa poziva se blok INI za inicijalizaciju auta. Zatim program ulazi u glavnu petlju, koja kontrolira kompletnu vožnju i iz koje izlazi tek kada je pritisnut srednji gumb na Bricku za prekid programa (slika 15).

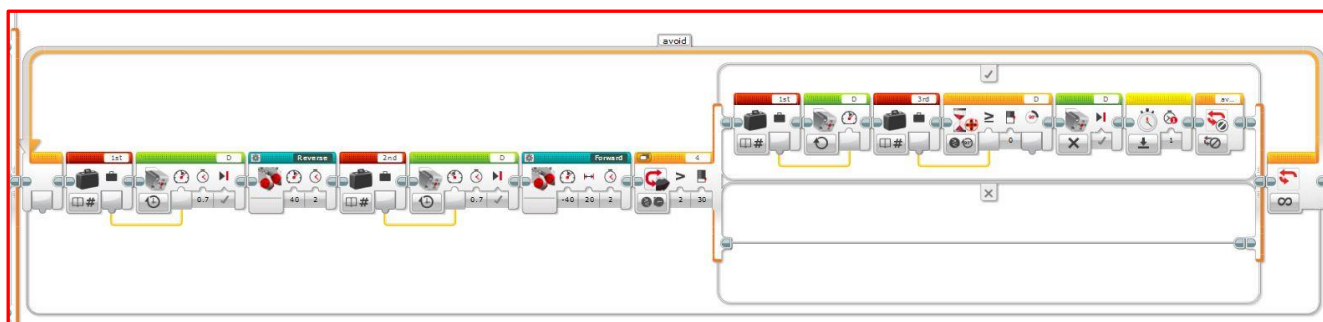


Slika 15: Pozivanje INI i dio programa prije petlje



Slika 16: Prva i druga truba nakon dolaska do prepreke

Samo ponašanje autića u ovom načinu rada je zapravo sličnije robotskom usisavaču Roombi, nego cestovnom autonomnom vozilu. Nakon paljenja program će autić voziti prema naprijed dok ne dođe do neke prepreke. Kada je prepreka prepoznata dolazi do prvog switcha. Autić će "veselo" potrubiti dvaput, signalizirajući prepreci da se makne s puta ako je u pitanju Lego čovječuljak ili drugi auto. Ako se prepreka ne pomakne, autić se polako pomiče još malo unaprijed te "agresivno" kontinuirano trubi 2 sekunde, pokušavajući potjerati prepreku s puta (slika 16).



Slika 17: Manevar okretanja

Ako se prepreka ne makne u iduće 2 sekunde, autić odustaje i pokreće manevar okretanja (slika 17). Ovo radi tako da kotače okrene udesno ili ulijevo i poziva blok Reverse. U Reverse blok šalje parametar koliko sekundi će ići unatrag, u ovom slučaju dvije. Zatim okreće kotače u suprotnom smjeru i poziva blok Forward isto sa dvije sekunde. Tijekom manevra ako senzori u bilo kojem trenutku detektiraju prepreku ispred ili iza autića, on okreće kotače i mijenja smjer kretanja. Ovo osigurava da ne dolazi do sudara tijekom manevra i znači da se autić može izvući iz većine situacija u kojima se nalazi. Nakon uspješnog okretanja, i nakon što je detektirao da više nema prepreke ispred sebe, autić usmjerava kotače prema naprijed i nastavlja svoje putovanje. Postoje još dva slična slučaja kod dolaska do prepreke. Ako se autić zaustavi, potrubiti dvaput i prepreka se makne s puta, autić će nastaviti svojim putem bez manevra za okretanje. Isto vrijedi ako se prepreka ne pomakne prvi put. Autić se približi i "agresivno" potrubiti, te ako se prepreka makne nastavlja svoj put bez manevra.

The image shows a Scratch script for a number guessing game. The script is organized into two main sections: a top section for initialization and a bottom section for the game loop.

Top Section:

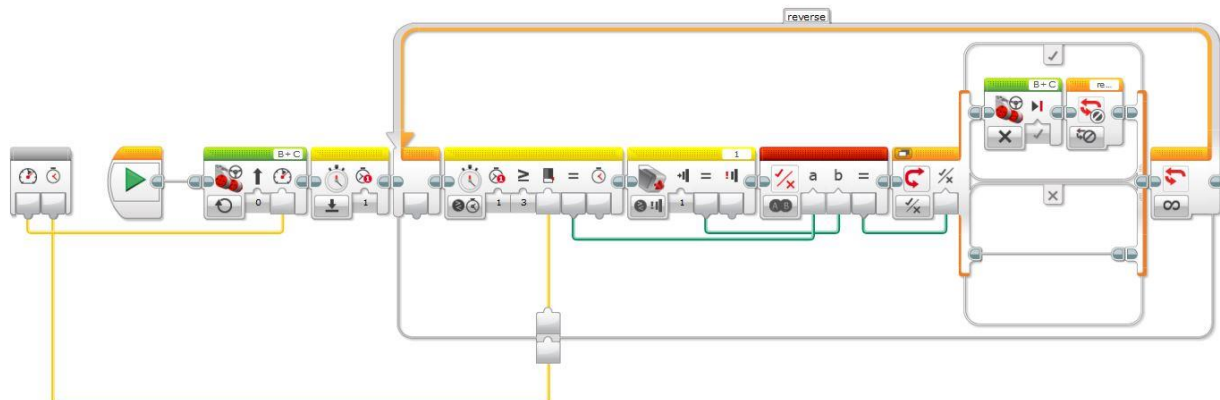
- When Green Flag Clicked:** This event triggers the start of the game.
- Set Score to 0:** A variable block labeled "Score" is set to 0.
- Set Guess to 0:** A variable block labeled "Guess" is set to 0.
- Set Secret Number to 1234:** A variable block labeled "Secret Number" is set to 1234.
- Play Sound:** A sound block labeled "clap" is played.
- Wait 2 seconds:** A wait block labeled "2" is used.
- Speak "Welcome to the game!" for 4 seconds:** A speech bubble block labeled "Welcome to the game!" is spoken for 4 seconds.
- Wait 2 seconds:** Another wait block labeled "2" is used.
- Speak "I'm thinking of a number between 1 and 1000." for 4 seconds:** A speech bubble block labeled "I'm thinking of a number between 1 and 1000." is spoken for 4 seconds.
- Wait 2 seconds:** A final wait block labeled "2" is used.

Bottom Section:

- When Green Flag Clicked:** This event triggers the start of the game loop.
- Set Guess to 0:** A variable block labeled "Guess" is set to 0.
- Set Secret Number to 1234:** A variable block labeled "Secret Number" is set to 1234.
- Play Sound:** A sound block labeled "clap" is played.
- Wait 2 seconds:** A wait block labeled "2" is used.
- Speak "Welcome to the game!" for 4 seconds:** A speech bubble block labeled "Welcome to the game!" is spoken for 4 seconds.
- Wait 2 seconds:** A wait block labeled "2" is used.
- Speak "I'm thinking of a number between 1 and 1000." for 4 seconds:** A speech bubble block labeled "I'm thinking of a number between 1 and 1000." is spoken for 4 seconds.
- Wait 2 seconds:** A final wait block labeled "2" is used.
- Loop:** A "forever" loop containing:
 - Ask "What's your guess?" and wait 2 seconds:** An ask block labeled "What's your guess?" is followed by a wait block labeled "2".
 - Compare Guess and Secret Number:** A "less than or equal to" comparison block labeled "Guess <= Secret Number".
 - If True:** A "say" block labeled "You guessed it!" is spoken for 2 seconds, followed by a "play sound" block labeled "clap".
 - If False:** A "say" block labeled "Wrong guess!" is spoken for 2 seconds, followed by a "play sound" block labeled "clap".
 - Increment Guess:** A "change guess by 1" block is used.
 - Decrement Score:** A "change score by -1" block is used.
 - Check Score:** A "less than or equal to" comparison block labeled "Score <= 0".
 - If True:** A "say" block labeled "Game Over!" is spoken for 2 seconds, followed by a "play sound" block labeled "clap".
 - If False:** A "say" block labeled "Keep guessing!" is spoken for 2 seconds, followed by a "play sound" block labeled "clap".
 - Repeat:** A "repeat" block labeled "8" is used.

Slika 18: Forward program

Reverse program je jako sličan programu Forward, kao što se vidi uspoređivanjem slika 18 i 19. Ovaj program zapravo radi istu stvar samo sa kretanjem unatrag. Glavna razlika je što ne koristi neki od senzora za daljinu već senzor za dodir, koji ako je u pritisnutom stanju gasi motore.



Slika 19: Reverse program

Dolazak do nepomične prepreke (zida), zaustavljanj i obavljanje manevara pomoću infracrvenog senzora i senzora za dodir prikazano je na slici 20.



Slika 20: Izbjegavanje prepreke i pokretanje manevara

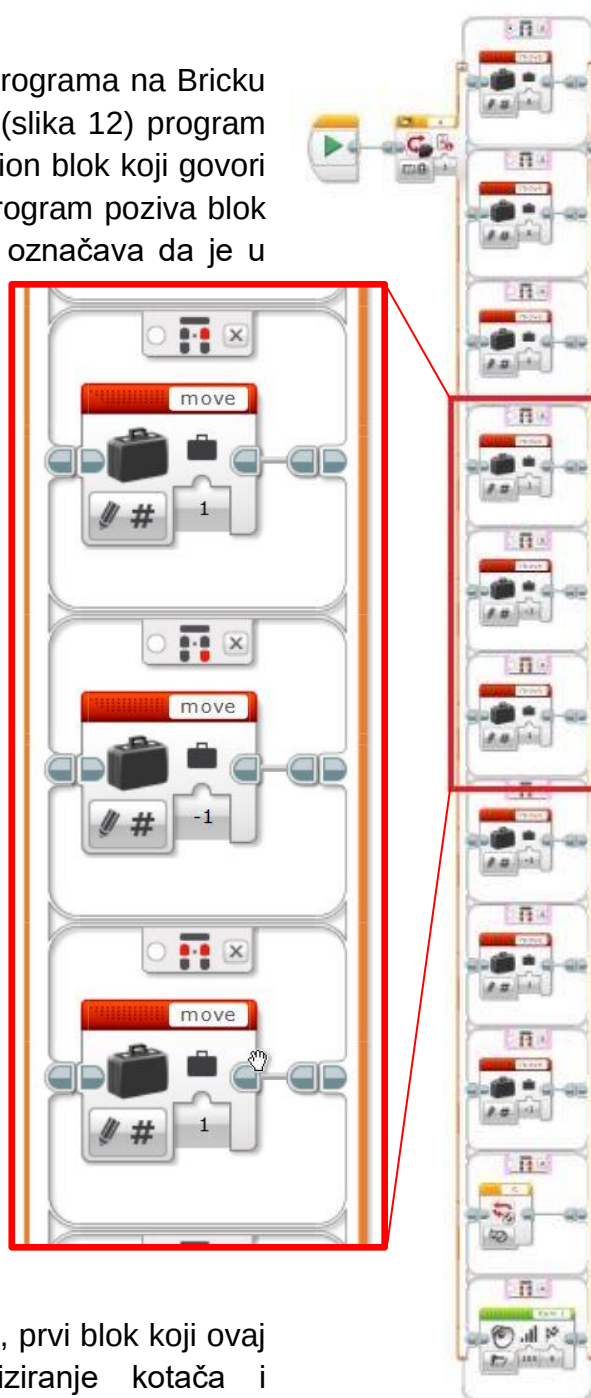
Na slici 21 vidimo autić u akciji trubljenja i približavanja drugom vozilu, koje se nakon prvog "veselog" trubljenja ne makne s puta. Zatim se autić približava i nakon "agresivnog" trubljenja Lamborghini se povlači i autić nastavlja svojom putanjom.



Slika 21: Izbjegavanje vozila

5.4.2. Ručno upravljanje autićem

Ako je nakon pokretanja glavnog programa na Bricku pritisnuta tipka dolje (Down), u switchu (slika 12) program ulazi u drugu opciju. Ovdje se nalazi action blok koji govori Brick Status Lightu da svijetli crveno i program poziva blok RC_MODE. Zatim autić pusti zvuk koji označava da je u ovom načinu rada i da je spreman za primiti naredbe. Za kontrolu autića koristi se prije spomenuti Remote Infrared Beacon. Ovo je mali Lego daljinski upravljač koji na sebi ima četiri gumbića za kontrolu, dva plava i dva crvena. Autić prima signale od upravljača i ovisno koji gumbi su stisnuti radi razne stvari. Program je ponovo kontroliran jednom velikom petljom, koja kontinuirano očitava signale odašiljane od strane daljinskog upravljača. Ova petlja sadrži jedan veliki i dva manja switcha. Oni Sve što ovi switchevi rade je šalju program dalje u tijeku u kojem korisnik upravljača želi da on ide. Primjerice, ako je prepoznato da korisnik drži pritisnutu kombinaciju dva gumba za kretanje naprijed, autić će to napraviti. Ako prima signale koji govore da je stisnuta druga kombinacija autić će ići unatrag, s trećom unatrag i ulijevo, i tako za sve moguće smjerove.

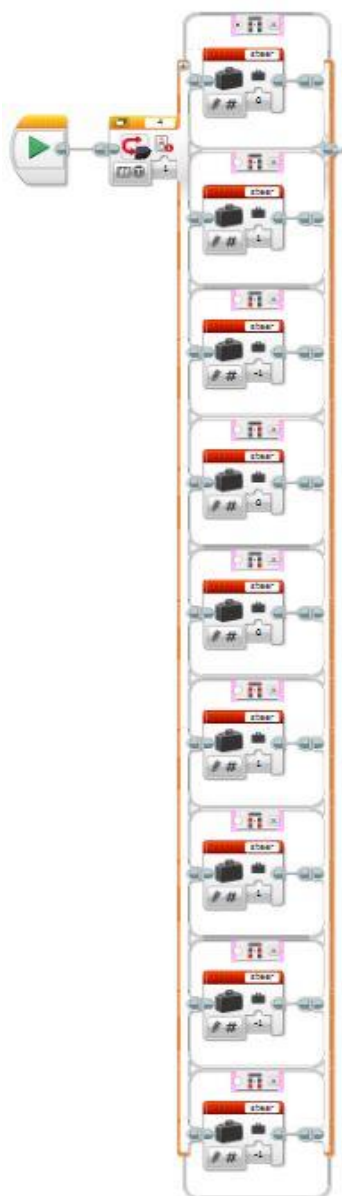


Isto kao u programu AUTO_MODE, prvi blok koji ovaj program poziva je INI, za inicijaliziranje kotača i pripremanje za vožnju. Također slično prijašnjem programu koji je pozivao Forward i Reverse, ovaj blok poziva potprograme Move (slika 22) i Steer (slika 23).

Slika 22: Move program

Move služi kao baza podataka u kojoj su spremljene kontrole za pokretanje motora. Točnije, Move sadrži switch koji ima jedanaest opcija. Svih 11 su kombinacije tipki koje korisnik može stisnuti. U prvih devet su spremljene varijable u kojima je pohranjeno što autić mora napraviti ako dođe do ove kombinacije. U

desetoj je prekid petlje, to jest prekid programa, i jedanaesta kombinacija je naravno truba (slika 22).



Osim Move, drugi potprogram koji RC_MODE poziva je Steer (slika 23). Izgleda jako slično programu Move (slika 22), a još je jednostavniji. Steer također ima switch koji se proteže oko cijelog programa, ali za razliku od programa Move nema zadnje dvije opcije za prekid i trubu. Sadrži istih devet kombinacija tipki u istom redosljedu kao i gore, sa 9 varijabli za skretanje srednjeg motora. Ove kombinacije su sinkronizirane s programom Move tako da kada je pritisnuta neka kombinacija tipka svi motori rade što bi trebali. Na primjer, kada je pritisnuta kombinacija da autić ide naprijed, Move govori da veliki motori počnu raditi, a Steer govori da srednji motor drži autić ravno.

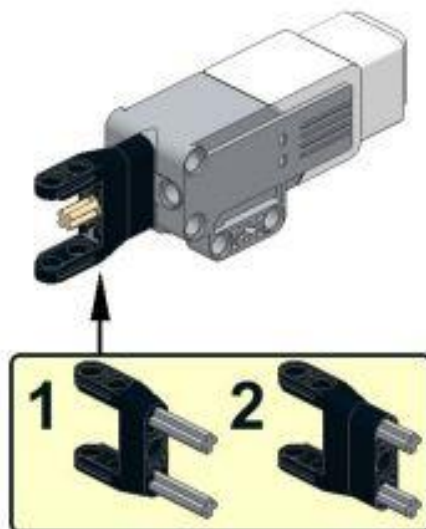
Iako je u ovom načinu autić upravljao ručno, motori nisu jedini koji imaju posla. Autić svedno koristi senzore za prikupljanje informacija te ima "saftey feature" (sigurnosne pogodnosti), poput automatskog kočenja ako dođe do prepreke ili zastavljanja ako je nešto pogodilo zadnji branik. Osim ovoga, autić pusti određene zvukove koji vozaču govore "Uh, oh. Go forward!" ako dođe do sudara tijekom vožnje unatrag, ili "Uh, oh. Go backwards!" ako senzori od naprijed detektiraju prepreku.

Za izlaz iz programa i gašenje autića postoji kombinacija tipki ili opcija je moguće na Bricku pritisnuti Back ili Center gumb.

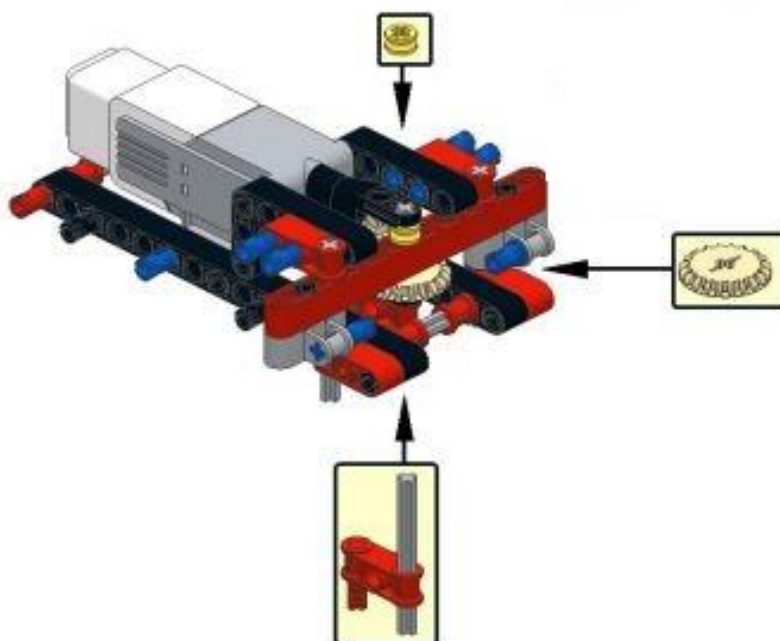
Slika 23: Steer program

5.5. Izrada autonomnog/RC autića

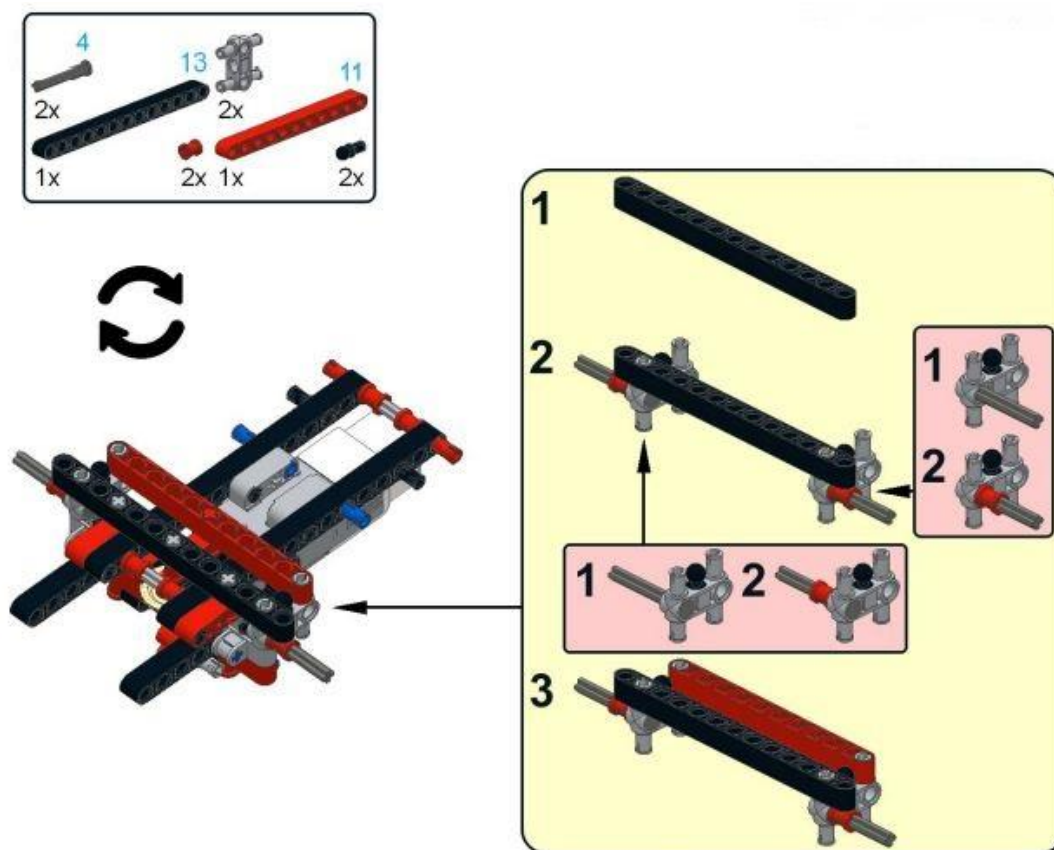
Prvo i osnovno za svaki auto je mehanizam skretanja. U ovom slučaju skretanje je omogućeno pomoću Medium Motora. Motor je postavljen paralelno s podom (slika 18) i koristi 2 zupčanika za promjenu rotacije s vodoravne osi u okomitu (slika 19), otprilike kao i pravi auti. Zbog dobivene rotacije po okomitoj osi moguće je dodati mehanizam koji će skretati oba prednja kotača kao na slici 20.



Slika 24: Postavljanje medium motora

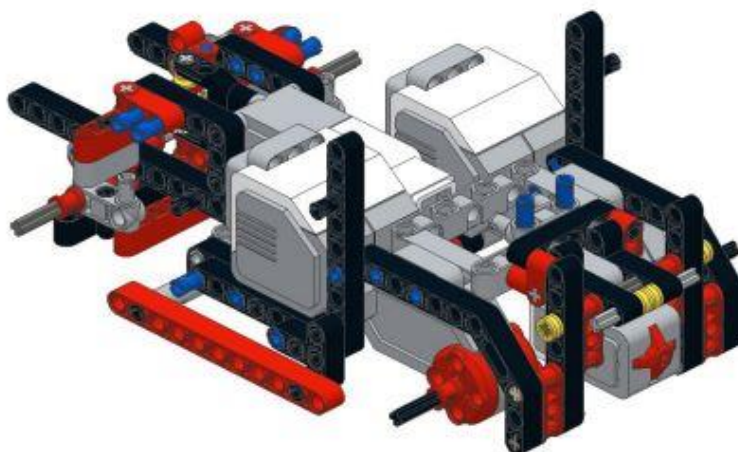


Slika 25: Osnove mehanizma za skretanje



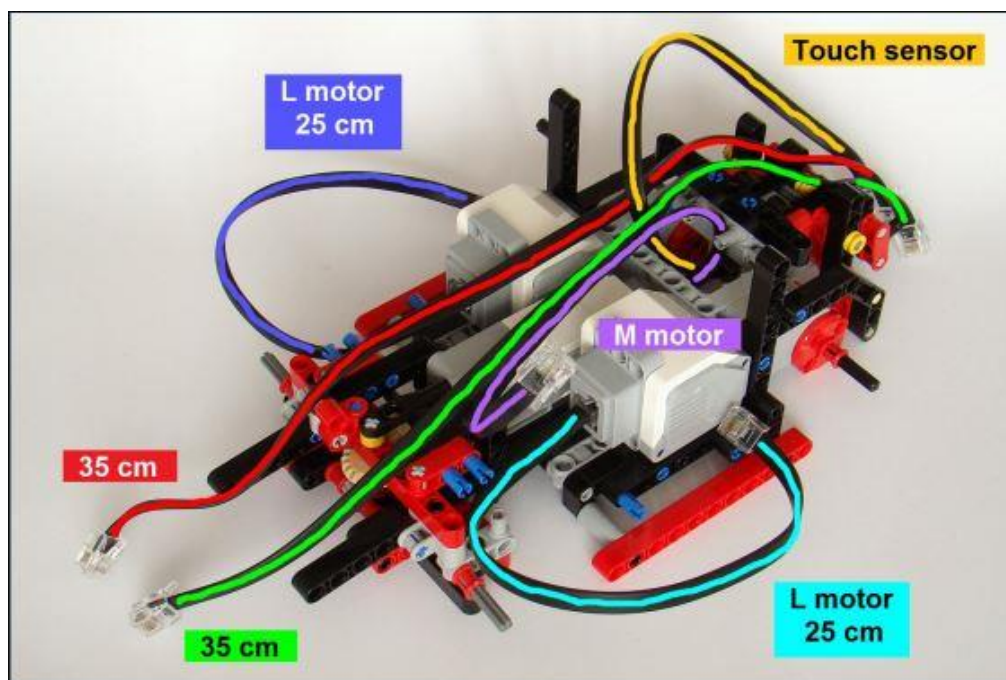
Slika 26: Kompletan mehanizam za skretanje

Nakon ovoga je potrebno dodati nešto što će autiću davati snagu. Tu se koriste 2 Large Motora koji će služiti kao stražnji pogon, po jedan za svaki kotač (slika 21). Nakon dodavanja motora potrebno je na stražnju stranu dodati i senzor za dodir. Nakon slaganja stražnjeg branika, ovaj senzor će autiću reći da je došlo do sudara tijekom vožnje u natrag.

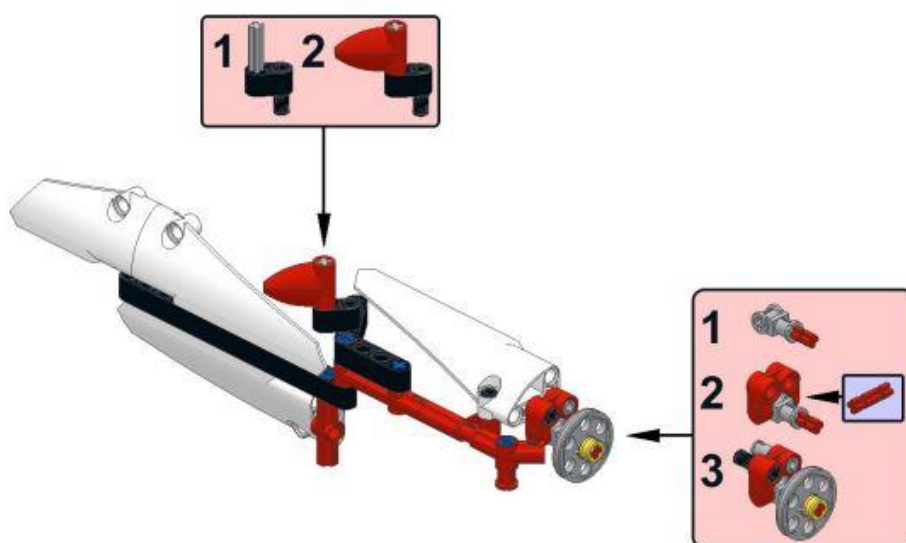


Slika 27: Large Motors i senzor za dodir

Prije slaganja dalje potrebno je početi spajati žice kako bi sve ostalo uredno. Važno je pobrinuti se da žice što manje vise i izviruju van iz autića da ne dolazi do zapinjanja i sličnih problema. Zbog ovoga je poželjno da većina žica ostane unutar karoserije auta. Sveukupno će ova verzija autića imati šest žica, 2 za Large Motore, 1 za Medium Motor i po jedna za svaki od tri senzora. Zasada su žice prikopčane samo na motore i senzor za dodir.

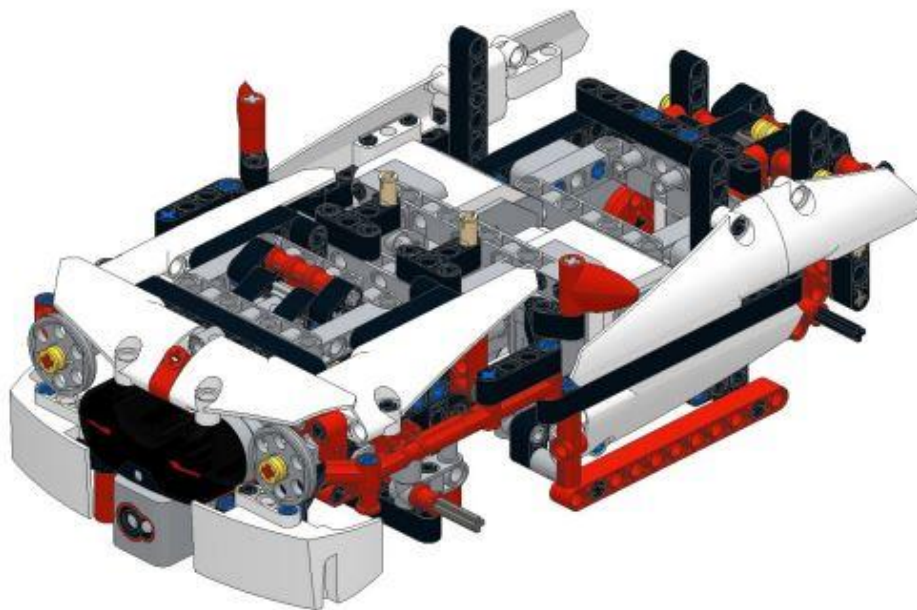


Slika 28: Žice na autiću



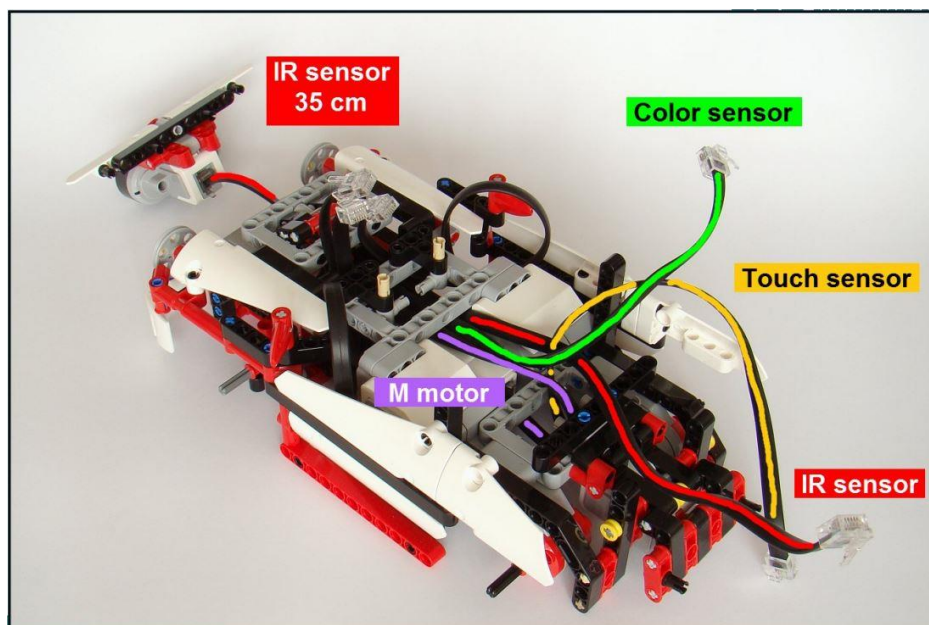
Slika 29: Koncept vrata i prednjeg svjetla

Dakle, za sada autić ima mehanizam za skretanje i pogon na zadnje kotače. Dodali smo žice, dvoja vrata i većinu karoserije (slika 23). Nakon riješenog problema kod zabijanja tijekom kretanja unatrag, vrijeme je isti problem riješiti za vožnju prema naprijed. Idući korak je dodavanje infracrvenog senzora i senzora za boju. Oba ova senzora idu na prednji dio auta (slika 24). Za razliku od kretanja unatrag, u ovom slučaju će autić prepoznati prepreke daleko prije sudara, što je dobro jer nije poželjno da se prednji branik ošteti koliko i zadnji.



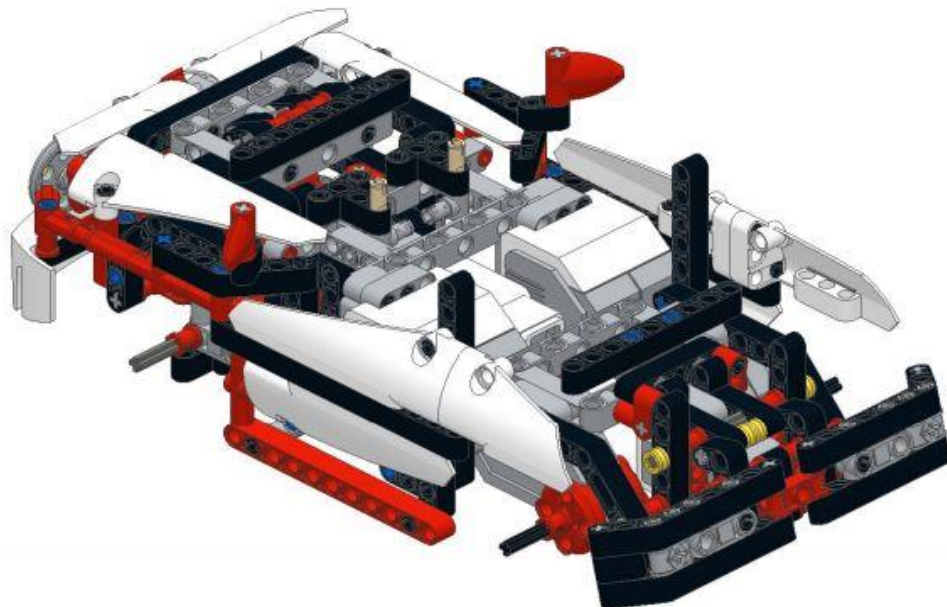
Slika 30: Infracrveni senzor i senzor za boju

Nakon ovoga potrebno je provući žice i spojiti senzore kako prikazuje slika 25.



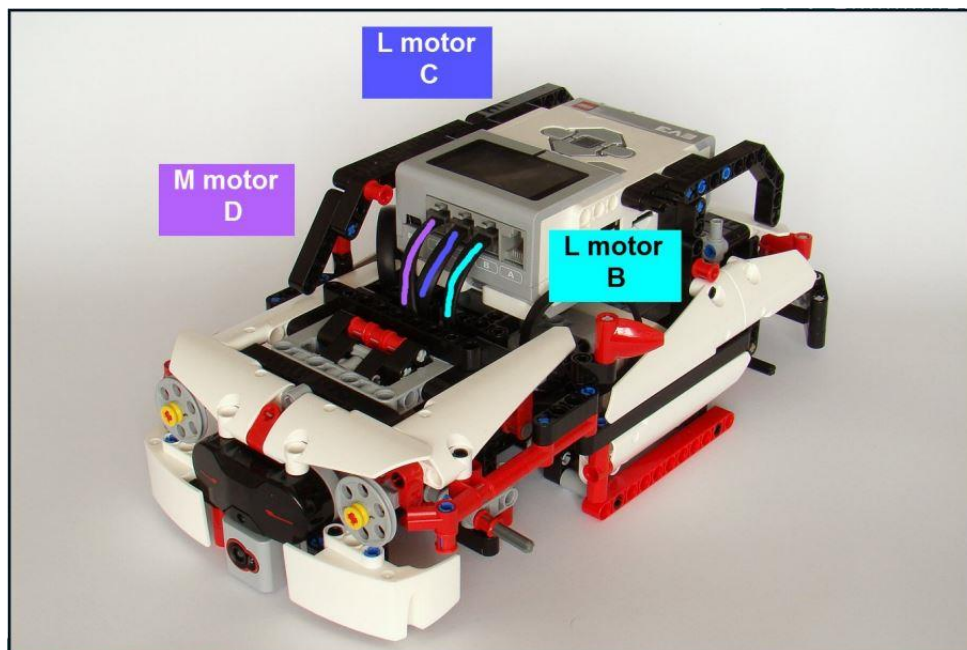
Slika 31: Spajanje infracrvenog senzora i senzora za boju

Idući korak je slaganje i dodavanje stražnjeg branika. Cijeli branik zapravo visi na jednom zglobu malo nagnut prema van. Naslonjen je na senzor za dodir ali ne dovoljno jako da ga pritisne. Ovo omogućava autiću da očitava pritisak na bilo kojem djelu branika i registrira da je došlo do sudara.



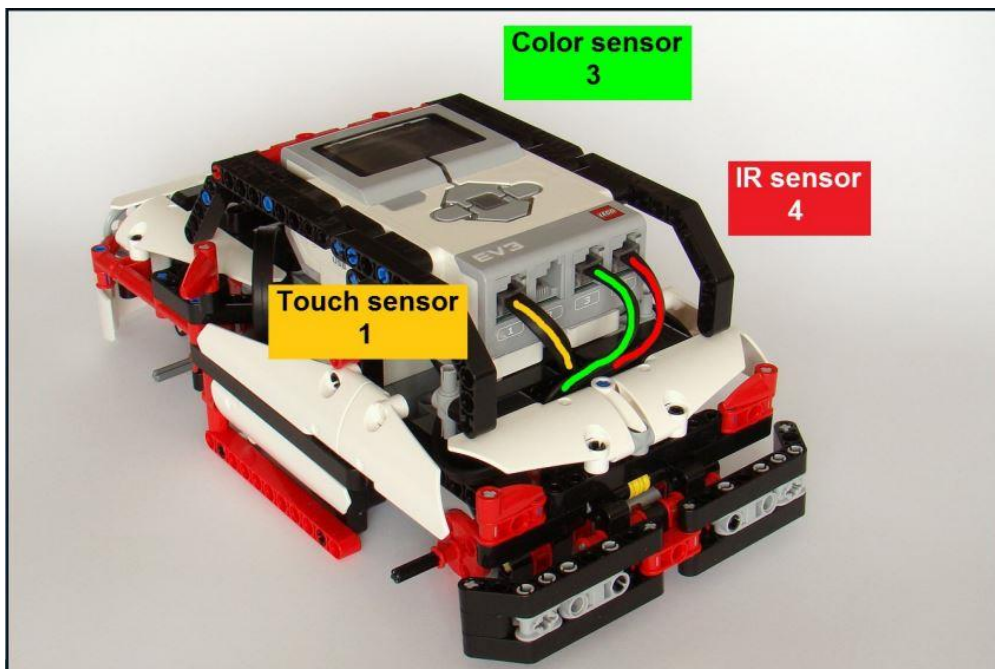
Slika 32: Dodavanje stražnjeg branika

Na kraju se dodaje inteligentni Brick. Sve što je preostalo napraviti je spojiti sve žice u njihove određene portove. Tijekom ovoga je potrebno obratiti pažnju da je Brick dobro okrenut i da su sve žice spojene točno kako prikazuju slike 27 i 28.



Slika 33: Spajanje motora u izlazne portove

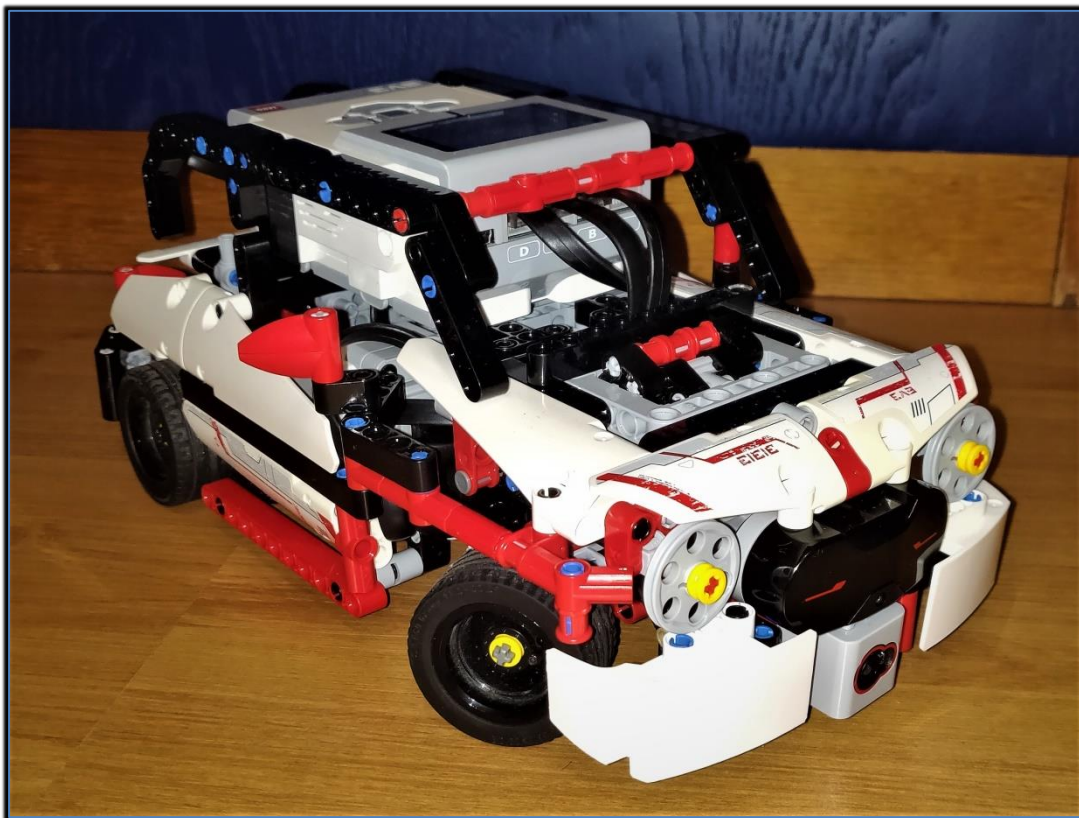
Motori moraju biti spojeni u izlazne portove koji su označeni slovima A, B, C i D. Veliki motori moraju biti spojeni točno u portove B i C, te srednji motor mora biti spojen u port D. Ovako im je određeno u programu i ako je bilo koja žica krivo spojena autić neće raditi.



Slika 34: Spajanje senzora u ulazne portove

Za spajanje senzora vrijede ista pravila. Senzori su spojeni u stražnju stranu Bricka gdje se nalaze ulazni portovi označeni brojevima 1, 2, 3 i 4. Senzor za dodir mora biti spojen točno u port 1, senzor za boje u port 2 i infracrveni senzor u port 4 kao što je prikazano na slici 28. Ako senzori nisu dobro spojeni autić također neće raditi kako bi trebao.

Nakon što je sve dobro spojeno i složeno samo je preostalo dodati kotače, malo podesiti i sakriti žice, pričvrstiti brick s dodatnim dijelovima da sve bude solidno i sigurno i autić je složen!



Slika 35: Završen Lego Mindstorms autić

6. Zaključak

Autonomna vozila i robotika općenito su grane tehnologije koje nas iz godine u godinu sve više približavaju Sci-Fi budućnosti o kojoj maštamo otkako smo bili djeca. Tijekom izrade ovog završnog rada stekao sam nova znanja i mišljenja o robotici, a posebno o autonomnim vozilima. Naučio sam kako koristiti senzore i kamere za kretanje prostorom, kako instalirati na vozilo, te kako koristiti objektno orijentirani EV3 program. Na ovom projektu bi volio nastaviti raditi jer ima mnogo prostora za napredak. Primjerice ako robot dođe do prepreke pod određenim kutem od otprilike 45 stupnjeva može se dogoditi da se zaustavi, i nakon što potrubi ne prepozna da je prepreka i dalje ispred. Zatim dolazi do sudara i autić prestaje raditi kako treba. Bilo bi zanimljivo dodati aktivno prepoznavanje veličine prepreke i može li se zaobići bez manevra okretanja. Ili pak kako bi autić bio sličniji stvarnim autonomnim vozilima, dodati kameru u svrhu prepoznavanja crta na cesti te programiranje neke verzije "lane keeping assista" da se autić drži u svojoj traci tijekom vožnje.

7. Literatura

1. Hrvatska Tehnološka Enciklopedija - Portal Hrvatske tehnološke baštine

<https://tehnika.lzmk.hr/robotika/>

2. PROel - Robotika

<https://www.proel.hr/robotika/>

3. EFerit - Kratka povijest robota

<https://hr.eferit.com/kratka-povijest-robota/>

4. AutoNet - Autonomna vozila

<https://www.autonet.hr teme/predstavljamo/autonomna-vozila-hocemo-li-ikada-sjediti-u-automobilima-bez-volana/>

5. HAK Revija - Autonomni automobili

<https://revijahak.hr/2019/07/30/autonomni-automobili-jos-ih-dugo-necemo-gledati-na-cestama/>

6. Lego Mindstorms EV3 - Official User Guide

https://www.lego.com/cdn/cs/set/assets/bltbef4d6ce0f40363c/LMSUser_Guide_LEG_O_MINDSTORMS_EV3_11_Tablet_ENUS.pdf

7. Lego Mindstorms: A History of Educational Robots

<http://hackededucation.com/2015/04/10/mindstorms>

8. Lego Education YouTube

<https://www.youtube.com/user/LEGOeducationUS>

9. Futurism - Human Level Artificial Intelligence

<https://futurism.com/human-level-artificial-intelligence-agi>

10. mreža.bug.hr - Autonomna vozila nisu tako daleka budućnost

<https://mreza.bug.hr/autonomna-vozila-nisu-tako-daleka-buducnost/>

8. Popis slika

Slika 1: Različite vrste robota

Slika 2: Isaac Asimov

Slika 3: Šest stupnjeva autonomnosti

Slika 4: Brojni senzori i kamere na automobilu Audi A8

Slika 5: Lego Mindstorms EV3 set

Slika 6: Lego Robotics Innovation System 2.0 set

Slika 7: Pet Mindstorms senzora i Infrared Beacon

Slika 8: Large Motor

Slika 9: Lego Mindstorms složen u obliku tenka

Slika 10: Medium Motor

Slika 11: Lego EV3 Intelligent Brick

Slika 12: Glavni dio programa

Slika 13: INI program

Slika 14: AUTO_MODE program

Slika 15: Pozivanje INI i dio programa prije petlje

Slika 16: Prva i druga truba nakon dolaska do prepreke

Slika 17: Manevar okretanja

Slika 18: Forward program

Slika 19: Reverse program

Slika 20: Izbjegavanje prepreke i pokretanje manevra

Slika 21: Izbjegavanje vozila

Slika 22: Move program

Slika 23: Steer program

Slika 24: Postavljanje medium motora

Slika 25: Osnove mehanizma za skretanje

Slika 26: Kompletan mehanizam za skretanje

Slika 27: Large Motors i senzor za dodir

Slika 28: Žice na autiću

Slika 29: Koncept vrata i prednjeg svjetla

Slika 30: Infracrveni senzor i senzor za boju

Slika 31: Spajanje infracrvenog senzora i senzora za boju

Slika 32: Dodavanje stražnjeg branika

Slika 33: Spajanje motora u izlazne portove

Slika 34: Spajanje senzora u ulazne portove

Slika 35: Završen Lego Mindstorms autić

9. Prilozi

Uz rad priložen je projekt za Lego Mindstorms EV3 program u kojem je programiran autić, upute za slaganje ovog specifičnog modela autića i sami Lego Mindstorms EV3 set.