

Sveučilište u Zagrebu  
Fakultet strojarstva i brodogradnje

# **ZAVRŠNI RAD**

Dominik Sremić

Zagreb, 2015.

Sveučilište u Zagrebu  
Fakultet strojarstva i brodogradnje

# **ZAVRŠNI RAD**

## **KONSTRUKCIJA I UPRAVLJANJE ELKTRIČNIM ROMOBILOM**

Mentor:

Prof. dr. sc. Mladen Crneković, dipl. ing.

Student:

Dominik Sremić

Zagreb, 2015.

Izjavljujem da sam ovaj rad izradio samostalno koristeći stečena znanja tijekom studija i navedenu literaturu.

Zahvaljujem se profesoru, mentoru dr.sc. Mladenu Crnekoviću na odličnoj suradnji i bezrezervnoj pomći, kao i gospodinu Zvonku Grgecu na stručnoj pomoći. Još bi se htio zahvaliti djelatnicima Laboratorija za alatne strojeve na izradi i tokarenju pojedinih dijelova za električni romobil.



SVEUČILIŠTE U ZAGREBU  
FAKULTET STROJARSTVA I BRODOGRADNJE



Središnje povjerenstvo za završne i diplomske ispite  
Povjerenstvo za završne ispite studija strojarstva za smjerove:  
proizvodno inženjerstvo, računalno inženjerstvo, industrijsko inženjerstvo i menadžment, inženjerstvo  
materijala i mehatronika i robotika

Sveučilište u Zagrebu Fakultet strojarstva i brodogradnje	
Datum: 16-02-2015	Prilog
Klasa: 602-04/15-6/3	
Ur.broj: 15-1803-15-137	

## ZAVRŠNI ZADATAK

Student: **DOMINIK SREMIĆ** Mat. br.: 0035182919

Naslov rada na hrvatskom jeziku: **KONSTRUKCIJA I UPRAVLJANJE ELEKTRIČNIM ROMOBILOM**

Naslov rada na engleskom jeziku: **DESIGN AND CONTROL OF ELECTRIC SCOOTER**

Opis zadatka:

Romobil je jedno od jednostavnijih i efikasnijih prijevoznih sredstava. Dodatkom električnog pogona romobil postaje pristupačan i osobama koje ga ne mogu voziti snagom svojih mišića.

Potrebno je projektirati romobil koji bi imao sve funkcije transportnog sredstva s idejom komercijalne proizvodnje.

Tražena rješenja:

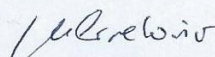
- projektirati robusnu mehaničku konstrukciju romobila koja može nositi odraslog čovjeka,
- odabrati potrebne motore, senzore i kontroler,
- napisati upravljački program za vožnju,
- sastaviti troškovnik te procijeniti isplativost proizvodnje.


Zadatak zadan:  
25. studenog 2014.

Rok predaje rada:  
1. rok: 26. veljače 2015.  
2. rok: 17. rujna 2015.

Predviđeni datumi obrane:  
1. rok: 2., 3., i 4. ožujka 2015.  
2. rok: 21., 22., i 23. rujna 2015.

Zadatak zadao:

  
Prof. dr. sc. Mladen Crneković

Predsjednik Povjerenstva:  
  
Prof. dr. sc. Zoran Kunica



# Sadržaj

Sažetak .....	1
Summary .....	2
1. Uvod .....	3
2. Električni pogon – jeftin, a ne zagađuje okoliš .....	4
3. Razvoj proizvoda.....	5
3.1. Tijek izrade proizvoda:.....	5
4. Proračun konstrukcije.....	6
5. Projektiranje 3D modela.....	8
5.1. Dizajniranje konstrukcije romobila .....	8
5.2. Dimenzije romobila.....	10
6. Izrada konstrukcije .....	11
6.1. Tijek izrade konstrukcije električnog romobila.....	11
7. Pogonski dio električnog romobila.....	13
7.1. Elektromotor.....	13
7.2. Prednosti i mane BLDC motora .....	14
7.3. Specifikacije elektromotora.....	15
7.4. Specifikacije kontrolera (ESC-a).....	16
8. Elektronika .....	17
8.1. Svrha i namjena tiskanih pločica za električni romobil.....	17
8.2. Električne sheme pločica.....	18
8.3. PCB pločice.....	21
9. Opis elektroničkih komponenti .....	23
9.1. Atmega328p – mikrokontroler .....	23
9.2. Podešavanje brzine romobila.....	24
9.3. Mehaničke kočnice.....	24
9.4. Mjerenje napona baterija .....	25
9.5. Mjerenje struje motora .....	26
9.6. Mjerenje brzine romobila .....	26
9.7. LED osvjetljenje.....	27
10. Softverski dio i programiranje.....	29
10.1. Glavni izbornik.....	30
10.2. Režimi vožnje.....	32
11. Testiranja i troškovi.....	35
11.1. Ovisnost struje elektromotora o njegovoj brzini .....	35

---

11.2.	Troškovi vožnje električnim vozilom.....	36
11.3.	Troškovi izrade električnog vozila .....	37
12.	Daljnji razvoj.....	38
12.1.	Specifikacije malog električnog romobila (mES) .....	38
12.2.	Usporedba malog i velikog električnog romobila .....	38
12.3.	Fotografije mES-a .....	39
13.	Zaključak.....	41
	Programski kod u mikrokontroleru .....	42
	Literatura .....	50

## Popis slika

Slika 1. Električni romobil .....	3
Slika 2. „Benzinske“ postaje u budućnosti [1] .....	4
Slika 3. "Slaganje" proizvoda [2] .....	5
Slika 4. Osnovni pravokutni profil .....	6
Slika 5. Pravokutna metalna ploča .....	6
Slika 6. Raspodjela naprezanja.....	6
Slika 7. Raspodjela progiba.....	7
Slika 8. Razvoj konstrukcije (1/4).....	8
Slika 9. Razvoj konstrukcije (2/4).....	8
Slika 10. Razvoj konstrukcije (3/4).....	9
Slika 11. Razvoj konstrukcije (4/4).....	9
Slika 12. Završna 3D verzija romobila.....	9
Slika 13. Nacrt romobila .....	10
Slika 14. Tlocrt romobila.....	10
Slika 15. Konstrukcija (1/6) .....	11
Slika 16. Konstrukcija (2/6) .....	11
Slika 17. Konstrukcija (3/6) .....	12
Slika 18. Konstrukcija (4/6) .....	12
Slika 19. Konstrukcija (5/6) .....	12
Slika 20. Konstrukcija (6/6) .....	12
Slika 21. Valni oblik signala BLDC motora [3].....	13
Slika 22. Stator (lijevo) i rotor (desno) BLDC motora [4] .....	13
Slika 23. Karakteristike BLDC motora [5].....	14
Slika 24. Dimenzije BLDC motora [6].....	15
Slika 25. BLDC motor na romobilu [7].....	15
Slika 26. Kontroler za motor (ESC) [8].....	16
Slika 27. Shema donje pločice .....	18
Slika 28. FTDI adapter [9] .....	18
Slika 29. Shema donje pločice .....	19
Slika 30. 4S LiPo baterija [10] .....	20
Slika 31. Gornja PCB pločica.....	21
Slika 32. Donja PCB pločica.....	22
Slika 33. Atmega328p - raspored pinova [11].....	23
Slika 34. Ručica za gas [12] .....	24
Slika 35. Logička tablica vrijednosti za ILI sklop [13] .....	24
Slika 36. Shema spajanja pull down otpornika [14].....	25
Slika 37. Naponsko djelilo [15].....	25
Slika 38. Senzor struje [16] .....	26
Slika 39. Nacin spajanja senzora struje .....	26
Slika 40. REED relay [17].....	26
Slika 41. 40% snage (normalna vožnja).....	27
Slika 42. 100% snage (kočenje) .....	27
Slika 43. Prednje LED svjetlo ukupne snage 12W.....	28
Slika 44. Boot screen.....	29
Slika 45. Unlock screen.....	29



Slika 46. Main menu .....	30
Slika 47. Speed menu .....	31
Slika 48. Power menu.....	31
Slika 49. Driving modes (1/4) .....	32
Slika 50. Driving modes (2/4) .....	33
Slika 51. Driving modes (3/4) .....	33
Slika 52. Driving modes (4/4) .....	34
Slika 53. Dijagram brzina-struja.....	35
Slika 54. SolidWorks model malog romobila .....	39
Slika 55. Električna shema pločice.....	39
Slika 56. PCB pločica      Slika 57. Programiranje mikrokontrolera.....	40
Slika 58. Završena mehanička konstrukcija .....	40
Slika 59. Električni romobil .....	41

# Sažetak

**Dominik Sremić**

## **Električni romobil**

Izrada vozila na električni pogon od samih početaka činila se kao jako zanimljiv projekt. Konačni cilj izrade električnog romobila bio je taj da, osim sigurne i udobne vožnje, njegova maksimalna brzina bude barem 30km/h, a autonomija vožnje približno 10 kilometara. U konačnici, rezultati su bolji od predviđanja, pa se tako najveća brzina penje čak do 43km/h, a autonomija do oko 15 kilometara. U ovom radu detaljno je opisana mehanička konstrukcija i njeno modeliranje u *CAD* programu, izvršeni su proračuni naprezanja i maksimalne moguće opteretivosti konstrukcije te je izabran osnovni metalni profil za konstrukciju. Motor koji pogoni romobil teži samo 1.2kg, a maksimalna snaga koju daje je 2400W što je daleko iznad većine kineskih vozila na električni pogon ovoga tipa. Tiskane pločice crtane su na računalu, a cjelokupnim sustavom upravlja slični mikrokontroler pod nazivom ATmega328P. On je programiran tako da ima nekoliko izbornika koji prikazuju različite podatke, a moguće je mijenjati čak i režim vožnje električnog romobila. Romobil je opremljen prednjim i stražnjim osvjetljenjem potrebnim za noćnu vožnju. Vozilo ovoga tipa moguće je voziti bez registriranja, ako se vozi isključivo u „economy“ režimu vožnje, a vožnja kao takva je jednostavna i sigurna zbog robusne konstrukcije, amortizera, disk kočnica i dobro programiranog sustava za upravljanje motorom.

**Ključne riječi:** vozilo na električni pogon, mehanička konstrukcija, elektronički upravljani sustav, korisničko sučelje

# Summary

**Dominik Sremić**

## **Electric scooter**

From the beginning, the idea of making an electrically propelled vehicle seemed to be a very interesting project. The ultimate goal was to make a safe and comfortable vehicle, with a maximum speed of at least 30km/h and with range of 12 kilometers. Ultimately, the results were even better than predictions so the top speed climbs up to 43 km/h and the autonomy is about 15 kilometers. In particular, this seminar contains detailed descriptions of the mechanical design and it's modeling in a *CAD* program. Also, the calculations of strainings and maximum possible load capacity of the structure were carried out and the basic metal profile for the construction was chosen. The engine of the scooter weighs only 1.2kg and the maximum power that it gives is 2400W, which is far above all Chinese electric-powered vehicles of this type. Printed circuit boards were drawn on the computer and the entire system is controlled by a tiny microcontroller called the Atmega328. It is programmed in such a way that there are several menus that display different data and it is even possible to change the driving mode of electric scooter. Scooter is also equipped with front and rear lighting needed for night driving. This vehicle is drivable without registering if you drive exclusively in "economy" mode driving. Also, driving is easy and safe due to robust construction, shock absorbers, disc brakes and well programmed engine management system.

**Key words:** electric vehicle, mechanical construction, electronic control system, user interface

## 1. Uvod

U današnje, suvremeno doba, veliki problem u gradovima postaju gužve na prometnicama i općenito u bilo kojem obliku prometa kao takvog. U teoriji, taj problem je rješiv izbjegavanjem korištenja automobila ili bilo kojeg oblika javnog prijevoza, a pritom nam je dodatna pogodnost ušteda na gorivu i skupim voznim kartama. Najčešće rješenje takvog problema je upravo svima nam znano pješčenje. Međutim, želimo li naše kretanje ubrzati, možemo se voziti i na biciklu, što se doima kao vrlo pogodno rješenje zbog sve većeg broja biciklističkih staza. Ako se ipak ne želimo umoriti, a opet uštedjeti i stići na odredište relativno brzo, tada bi primjer kompatibilnog rješenja bilo praktično električno vozilo, u ovom slučaju električni romobil.



*Slika 1. Električni romobil*



## 2. Električni pogon – jeftin, a ne zagađuje okoliš

Danas je razvoj vozila na električni pogon uvelike uznapredovao. Električni pogon donosi brojne pogodnosti, pa tako ne zagađuje okoliš što je velika i jedna od glavnih prednosti u odnosu na današnja vozila koja svakodnevno emitiraju štetne ispušne plinove te time zagađuju zrak, okoliš, a i štetno utječu na ljude. Vožnja je također jeftinija te tako pristupačnija široj populaciji. Uz još jedno zapažanje - fosilna goriva će u roku od nekoliko desetaka godina biti iscrpljena te će tako znatno porasti potraga za vozilima na alternativnu vrstu pogona. Jedini problem je što su vozila, pogotovo automobili na električni pogon, još uvijek vrlo skupa, a i za razliku od drugih, jako malo zastupljena na tržištu zbog pojave konkurentnosti s onima koje pokreću fosilna goriva. Tako se, bez obzira na vrlo nisku cijenu električne energije kao pokretača ovakvih vozila i svih ostalih prednosti, ona jako rijetko mogu vidjeti na ulicama.



Slika 2. „Benzinske“ postaje u budućnosti [1]

## 3. Razvoj proizvoda

Izrada električnog vozila ili bilo kojeg drugog složenijeg projekta nije jednostavan zadatak. Ona zahtjeva mnoštvo novih ideja, računanja, odabira valjanih dijelova, skiciranja na papiru, izrade 3D modela, proračunavanja čvrstoće i dimenzioniranja dijelova, piljenja, rezanja i ostalog potrebnog za konačnu realizaciju projekta.

Neke osnovne smjernice kako bi trebao izgledati tijek izrade električnog romobila navedene su u donjem tekstu:

### 3.1. Tijek izrade proizvoda:

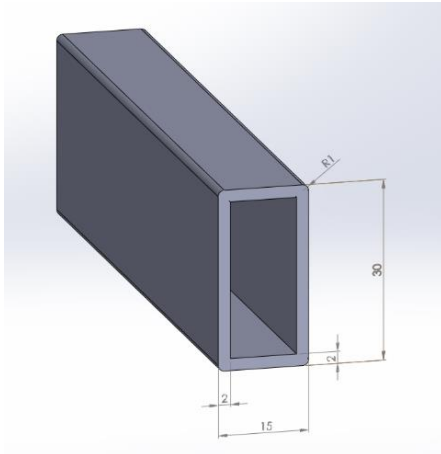
1. Osmišljavanje okvirnog izgleda električnog romobila
2. Odabir glavnog profila za konstrukciju
3. Odabir ključnih dijelova na vozilu (kotači, amortizeri, motor)
4. Izrada 3D modela u *CAD* programu
5. Rezanje dijelova konstrukcije i priprema za zavarivanje
6. Zavarivanje svih dijelova konstrukcije
7. Detaljno brušenje svih metalnih površina
8. Bojanje konstrukcije temeljnom bojom („primer“)
9. Bojanje dijelova konstrukcije odabranom bojom
10. Montaža
11. Testiranje



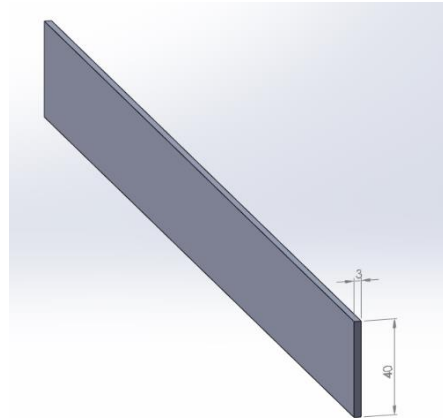
*Slika 3. "Slaganje" proizvoda [2]*

## 4. Proračun konstrukcije

Osnovni dio konstrukcije je čelična cijev pravokutnog profila dimenzija  $30 \times 15 \text{ mm}$ , sa stijenkom  $2 \text{ mm}$  debljine ( $30 \times 15 \times 2 \text{ mm}$ ). U izbor su bili uključeni i čelični profili  $20 \times 20 \times 2 \text{ mm}$  te  $30 \times 15 \text{ mm}$  sa  $1 \text{ mm}$  ili  $1.5 \text{ mm}$  debljinom stijenke, ali profil  $30 \times 15 \times 2 \text{ mm}$  se pokazao kao najbolji i najsigurniji izbor. Uz takvu cijev pravokutnog presjeka, za konstrukciju su još upotrebljene okrugle cijevi različitih dimenzija, te puni profil  $40 \times 3 \text{ mm}$ .

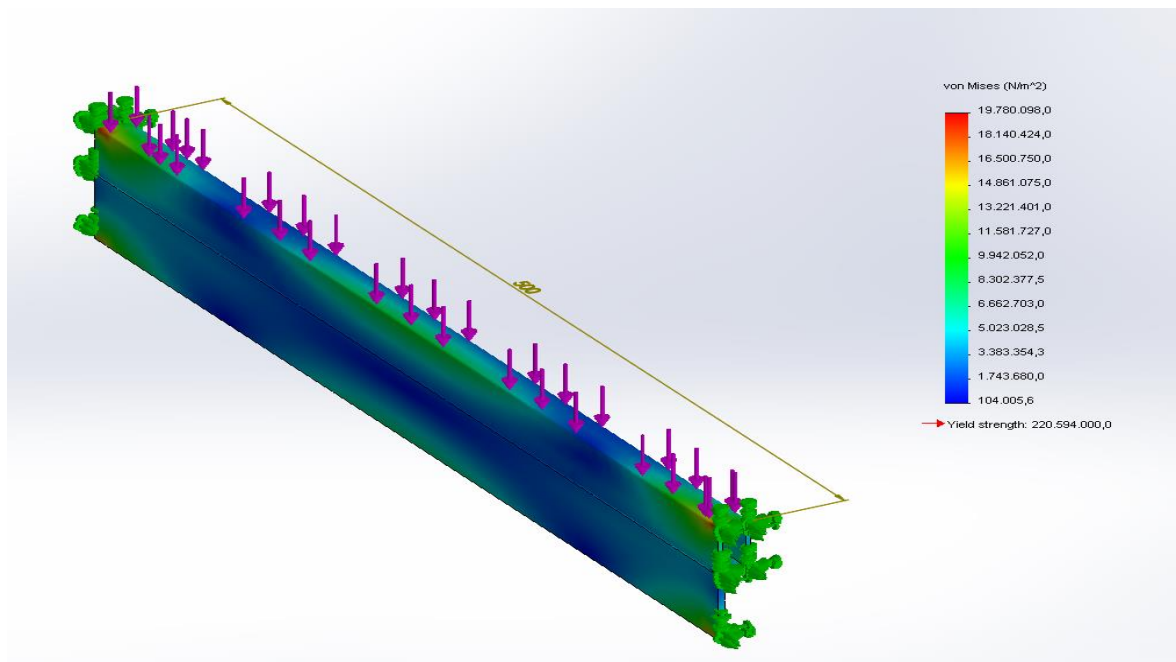


Slika 4. Osnovni pravokutni profil



Slika 5. Pravokutna metalna ploča

Najopterećeniji dio romobila čine dvije horizontalne cijevi na kojima je s donje strane zavarena metalna ploča dimenzije  $40 \times 3 \text{ mm}$ . Na njima se nalazi aluminijska ploča koju s gornje strane opterećuje vozač. Zbog težine vozača, koja je veća od sveukupne težine romobila, taj dio je nedvojbeno najopterećeniji dio konstrukcije, a opterećen je na savijanje.



Slika 6. Raspodjela naprezanja

Slika 6. prikazuje raspodjelu napreznja duž najopterećenijeg dijela konstrukcije. Zelene strjelice prikazuju uklještenje, ti dijelovi zavareni su na ostatak konstrukcije. Crvene strelice označavaju silu od  $1000N$  koja djeluje prema dolje. Ako pretpostavimo da je prosječna težina vozača  $80kg$ , računica bi bila slijedeća:

$$F_{vozač} = mg = 80 * 9.81 \approx 800N$$

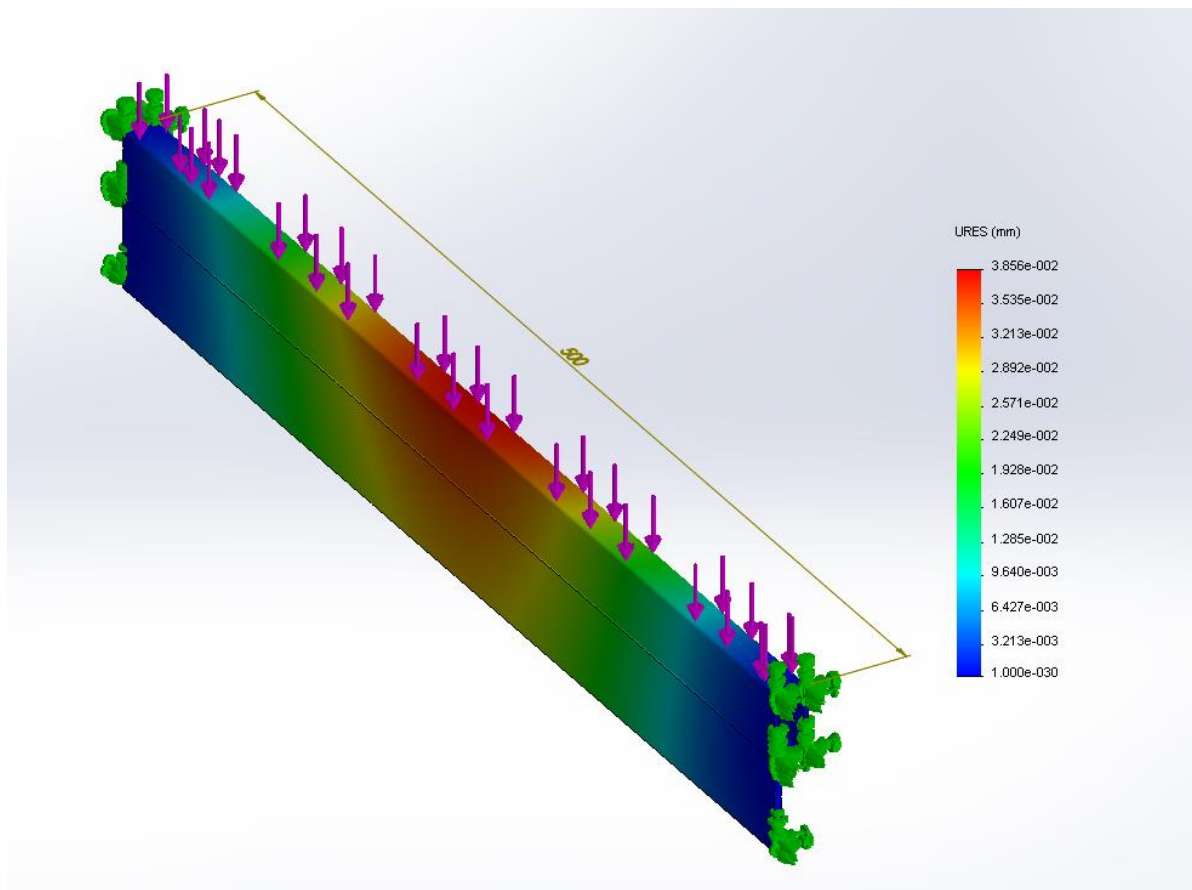
$$F_{proračunato} = 1000 * 2 = 2000N \text{ (sila na 2 paralelne cijevi)}$$

$$S = \frac{F_{proračunato}}{F_{vozač}} = \frac{2000}{800} = 2.5$$

Potrebno je napomenuti da ovdje  $F_{proračunato} \neq F_{max}$  što nas navodi na zaključak da bi konstrukcija izdržala težinu veću i od  $200kg$ !

Prilikom statičkog opterećenja tog dijela konstrukcije, vidljivo je kako je najveće napreznje nešto manje od  $\sigma_{max} = 20\,000 \frac{N}{m^2}$ , a s obzirom da je  $\sigma_{dop} = 220\,000 \frac{N}{m^2}$ , time opasnost od loma ne postoji te je konstrukcija sigurna i u slučaju djelovanja mnogo većih sila koje mogu biti uzrokovane neravnim podlogama po kojima se vožnja vrši, prijelaz s nogostupa na cestu pri velikoj brzini ili pri sudaru s određenim objektom.

Na slici 7 prikazana je raspodjela pomaka na tom istom elementu. Maksimalni progib je  $0.038\,mm$  te ako uzmemo u obzir duljinu elementa od  $500mm$ , dobijemo da je progib  $0.00007712\%$  što je u slučaju ovakve konstrukcije zaista zanemarivo.



Slika 7. Raspodjela progiba



## 5. Projektiranje 3D modela

3D modeliranje je proces kreiranja matematičke reprezentacije nekog trodimenzionalnog objekta. 3D model je u svojoj srži skup podataka o točkama u prostoru i drugih informacija koje računalo interpretira u virtualni objekt koji se iscertava na zaslonu.

Objekti se modeliraju na način da se na nekoj proizvoljnoj plohi nacrtaju glavni oblik objekta (krug, kvadrat, pravokutnik itd.) te mu se nakon toga dodaje treća dimenzija. Time dobivamo trodimenzionalno tijelo. Ako u/na tom tijelu želimo napraviti određenu promjenu kao što je na primjer rupa, potrebno je odabrati površinu („plane“) na kojoj ćemo nacrtati proizvoljni manji krug te umjesto dodavanja treće dimenzije („extrude boss/base“) izabrati opciju brisanja dijela objekta („extrude cut“)

Izrada 3D modela vrlo je bitna zbog sljedećeg:

- Moguće je prenijeti ideje o željenom izgledu projekta na računalo
- Jednostavan proračun konstrukcije
- Na temelju konstruiranog 3D modela moguće je vidjeti koje dijelove moramo/možemo naknadno modelirati
- Mogućnost printanja 3D modela smanjenih dimenzija (za prezentaciju projekta)
- Jednostavna promjena dimenzija pojedinih dijelova konstrukcije
- Jednostavna izrada ravnih ploča pomoću CNC rezača, bušilica i glodalica

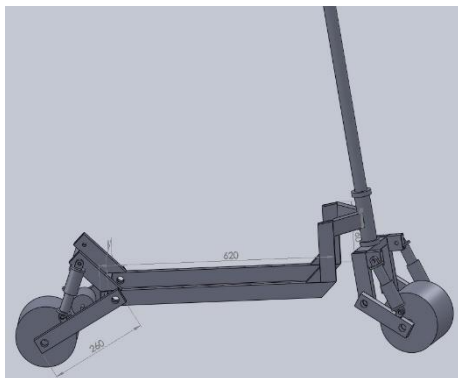
### 5.1. Dizajniranje konstrukcije romobila

U samom začetku projekta romobil je bio zamišljen kao jednostavno električno vozilo s kojim je moguće prevesti se od točke *A* do točke *B* ugodnom vožnjom bez umaranja s minimalnim utroškom novca.

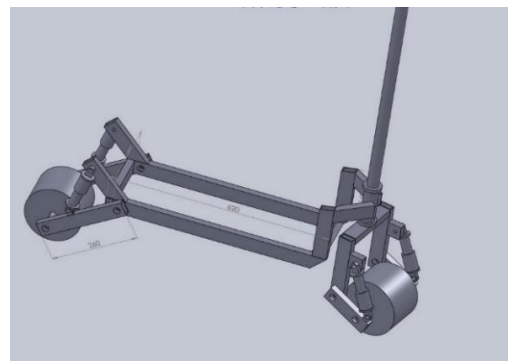
Iz spomenutog razloga, na jednom od prvih izrađenih 3D modela, romobil ima dva amortizera s prednje i također dva sa stražnje strane. To se pokazalo lošim izborom jer su amortizeri kao takvi bili prekruti za takvo postavljanje.

Također, kotače sa slike početnog 3D modela nije bilo moguće kupiti (širina i promjer), pa su stoga i kotači promijenjeni.

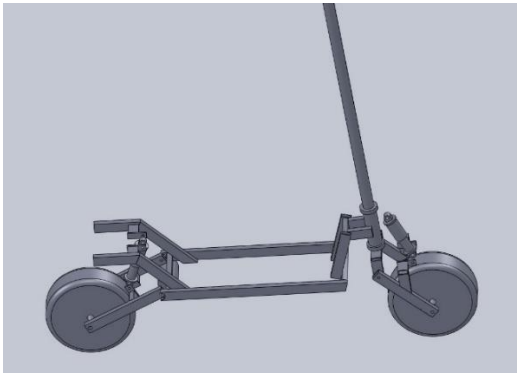
Nakon toga, konstrukcija romobila je više puta mijenjana i doradivana dok se nije postigao najbolji mogući izgled, čvrstoća, jednostavnost obrade i montaža dijelova. Odabir kotača predstavljao je velik problem, ali naposljetku su izabrani oni koji su obično namijenjeni za tačke (260mm promjera i 80mm širine). Na sljedećim slikama prikazan je daljnji razvoj konstrukcije:



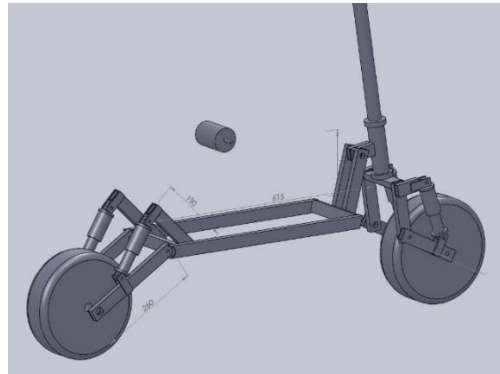
Slika 8. Razvoj konstrukcije (1/4)



Slika 9. Razvoj konstrukcije (2/4)

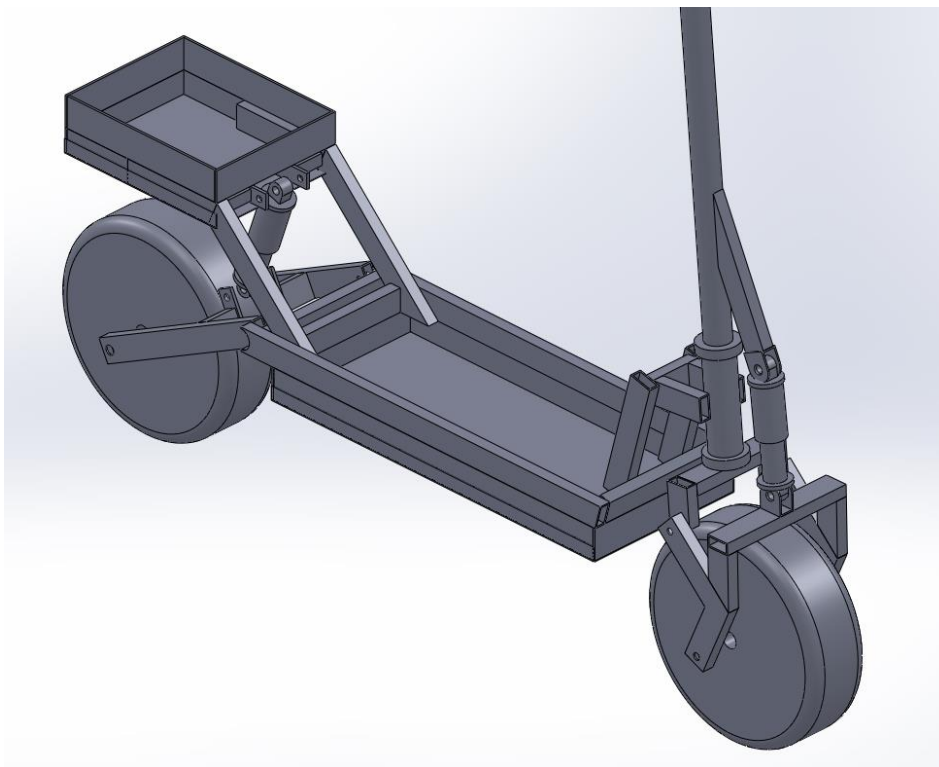


Slika 10. Razvoj konstrukcije (3/4)



Slika 11. Razvoj konstrukcije (4/4)

Konačna verzija električnog romobila:



Slika 12. Završna 3D verzija romobila

Na završnom 3D modelu poboljšana je i ojačana konstrukcija, dodan je stražnji pretinac, dok je u središnjem prostoru, koji je dubine 7cm, stvoreno više prostora za baterije i elektroniku. Broj amortizera sveo se na jedan sa svake strane romobila te su kotači prilično vjerno prikazani u odnosu na promjer i širinu.

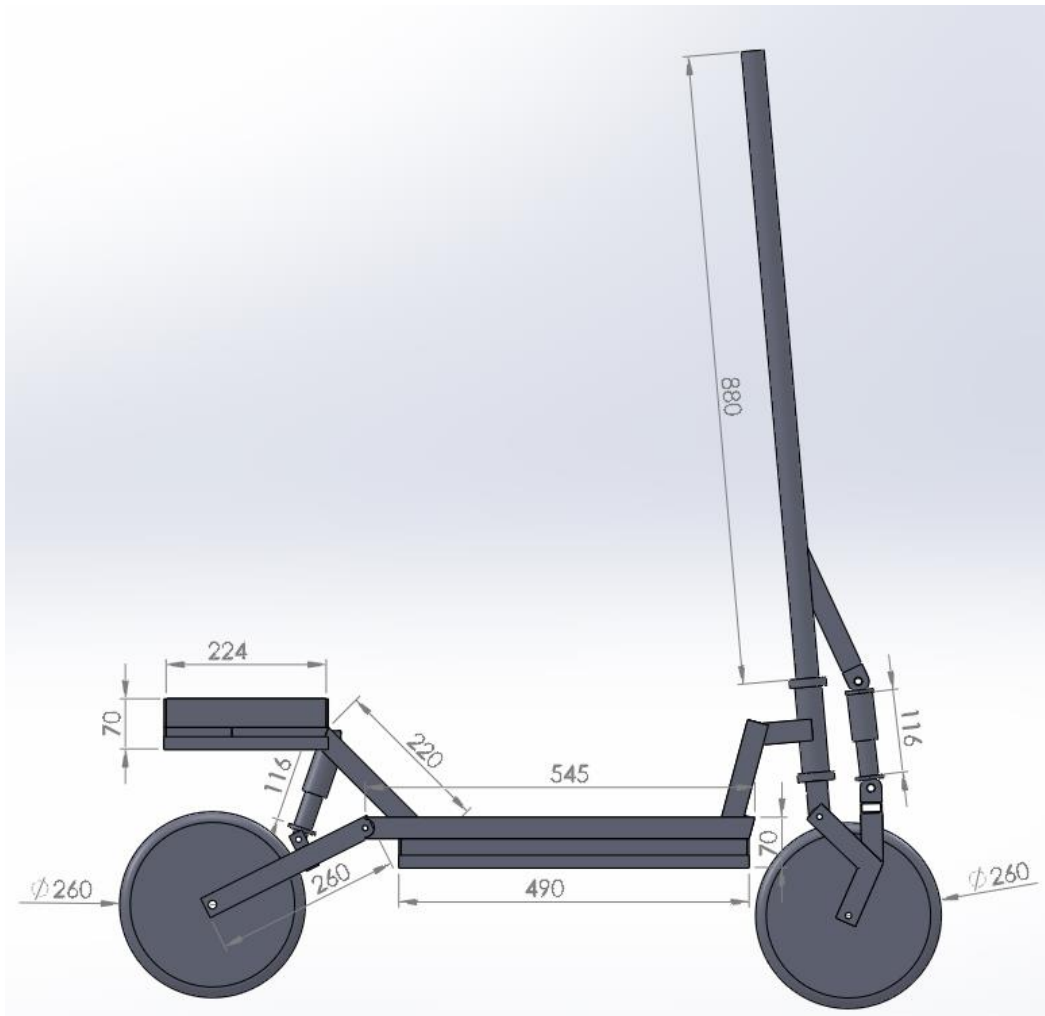
Jedini nedostatak na slici 12 je podnica i poklopac za stražnji pretinac te volan koji je identičan biciklističkom zajedno s kočnicama i ručicama.

## 5.2. Dimenzije romobila

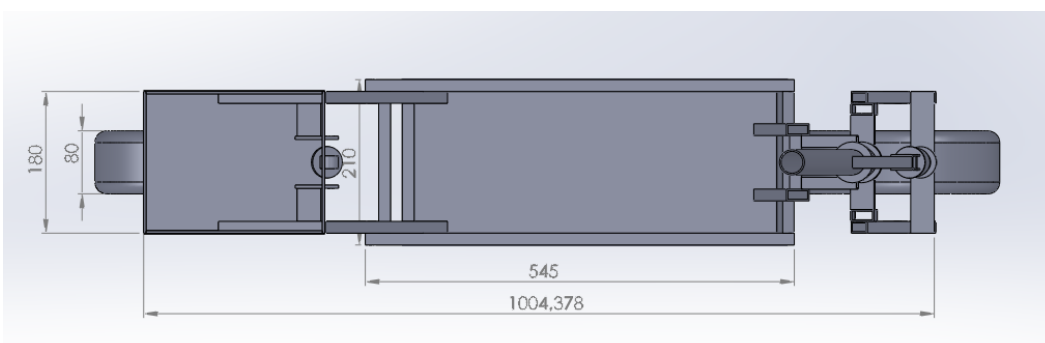
Dimenzije svakog objekta vrlo su važne, pogotovo u današnje vrijeme kada se teži smanjenju dimenzija samog vozila kao takvog što je više moguće. Što je vozilo manjih dimenzija, to je mobilnije i lakše za rukovanje ili pospremanje u stan, auto ili garažu.

Ovaj romobil je po dimenzijama veći od dječjih romobila, ali je zato mnogo manji i lakši od skutera, a kraći je i od bicikla.

Udaljenost najniže točke od poda je  $19\text{cm}$  što predstavlja veliku udobnost prilikom prelaska s nogostupa na ulicu, čak bez potrebe smanjivanja brzine.



Slika 13. Nacrt romobila



Slika 14. Tlocrt romobila

## 6. Izrada konstrukcije

Uz pomno izrađen plan i 3D model bez grešaka, konstrukcija koju je potrebno izraditi ne predstavlja prevelik problem. Kada je u pitanju izrada prototipa, rijetko kada je gotov proizvod identičan prvotnoj verziji 3D modela. Problemi dolaze „u hodu“ zbog nedovoljne preciznosti strojeva (bušilica, brusilica) ili samog radnika koji izvodi tehnološke procese. Oni, također, mogu nastati i zbog oskudne ponude potrebnih dijelova na tržištu: na primjer nedostatak manje krutih amortizera, kotača zamišljenih dimenzija, lančanika, ležajeva i ostalih dijelova koji su projektirani u CAD software-u. Iako bi se neke od navedenih stvari mogle pronaći na tržištu ili čak izraditi uz dovoljno znanja i predanosti, upitna je cijena takvog izratka u odnosu na neki jednostavniji koji se isto tako može primijeniti uz neke preinake.

Izrada konstrukcije sastoji se od nekoliko tehnoloških procesa:

- Piljenje cijevi i metalnih profila
- Tokarenje cijevi
- Brušenje pojedinih dijelova, a na kraju i cjelokupne zavarene konstrukcije
- Bušenje raznih provrta
- Zavarivanje konstrukcije
- Montaža

### 6.1. Tijek izrade konstrukcije električnog romobila

Izrada same konstrukcije, koja uključuje sve navedene tehnološke postupke, zauzela je najduži vremenski period.

Kod ručne izrade takvog projekta potrebno je puno strpljenja i preciznosti kako bi u konačnici sve pravilno funkcioniralo. Potrebno je vrlo precizno izbušiti svaku rupu, izrezati svaku cijev i profil, i ono najkompliciranije, točno namjestiti određeni dio konstrukcije prije zavarivanja i spriječiti pomake nekog elementa uzrokovane jakim zagrijavanjem konstrukcije za vrijeme zavarivanja.

Na sljedećim slikama prikazan je redoslijed slaganja konstrukcije:



Slika 15. Konstrukcija (1/6)



Slika 16. Konstrukcija (2/6)



*Slika 17. Konstrukcija (3/6)*



*Slika 18. Konstrukcija (4/6)*



*Slika 19. Konstrukcija (5/6)*



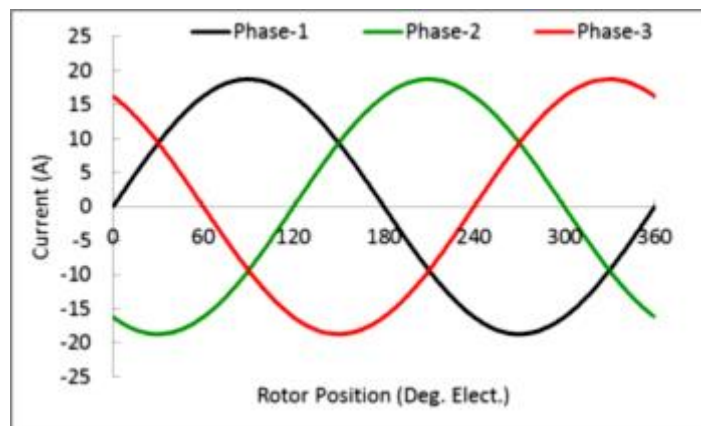
*Slika 20. Konstrukcija (6/6)*



## 7. Pogonski dio električnog romobila

### 7.1. Elektromotor

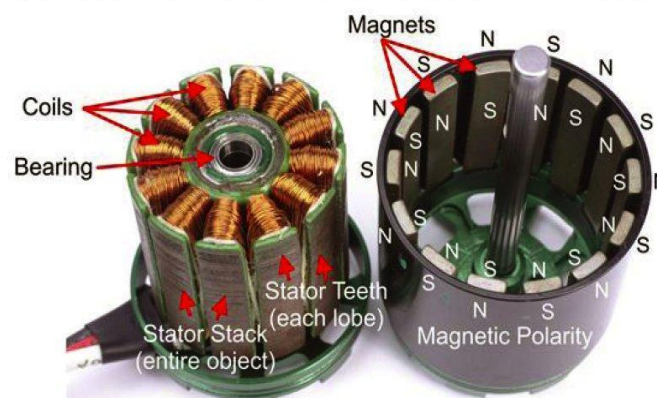
Pogonski dio romobila čini takozvani *brushless* elektromotor, točnije *Brushless DC* elektromotor. Iz samog imena doznajemo da je motor napajan DC-om, to jest, istosmjernom strujom te da je to tip motora bez četkica. To je sinkroni motor koji preko uređaja zvanog ESC (*Electric speed controller*) pretvara istosmjernu struju u izmjeničnu koja pogoni motor. Ta izmjenična struja nema klasičan sinusoidni valni oblik već je to dvosmjerna struja bez ograničenja na valnom obliku. Spomenuti valni oblik struje izgleda ovako:



Slika 21. Valni oblik signala BLDC motora [3]

*Brushless* motor sastoji se od kućišta na kojem se nalaze jaki permanentni magneti i statora na kojem se nalaze zavojnice. Postoje takozvani *Inrunner* i *Outrunner* motori, pri čemu su *Outrunner* daleko popularniji *brushless* motori zbog boljeg hlađenja namotaja. Kod navedenih motora, cijelo kućište motora spojeno je s osovinom te sve zajedno rotira oko statora. Vrlo mali dio motora je statičan, gledano izvana, i upravo taj dio se pričvršćuje za metalni dio konstrukcije romobila.

### OUTRUNNER COMPONENTS

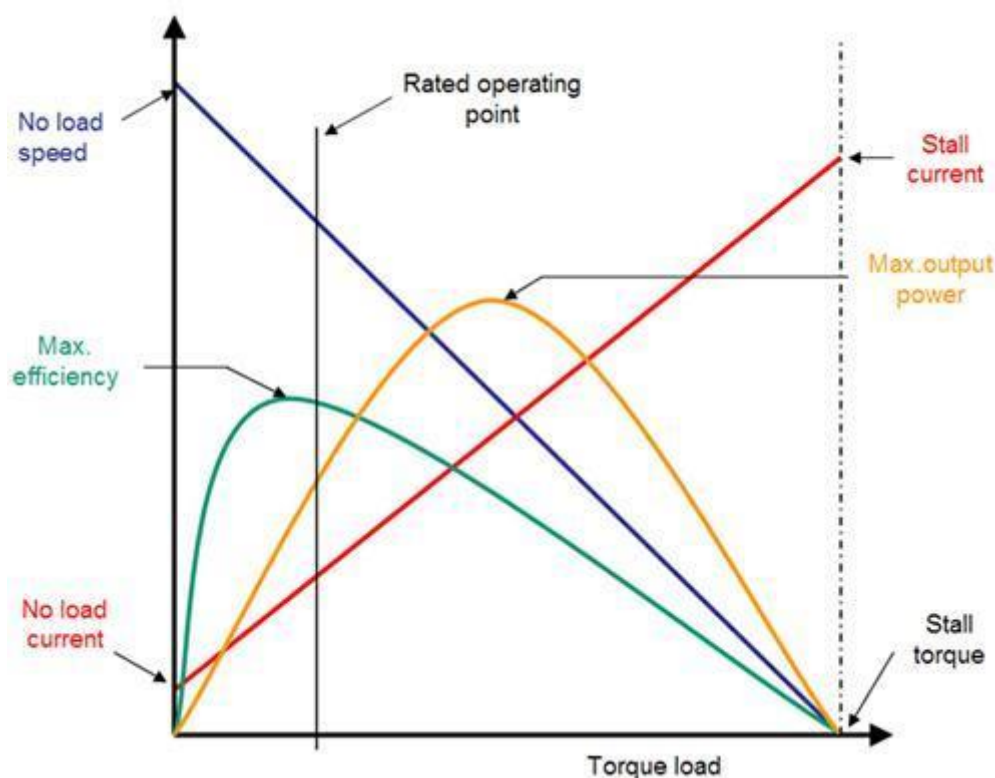


Slika 22. Stator (lijevo) i rotor (desno) BLDC motora [4]

## 7.2. Prednosti i mane BLDC motora

**Prednosti** *brushless* motora nad običnim *brushed* (četkastim) motorima su višestruke: *Brushless DC* (BLDC) motori imaju veću gustoću snage, to jest, više momenta po jedinici težine motora kao i više momenta po jedinici snage što ujedno predstavlja i veću efikasnost takvog motora. Spomenuti motori proizvode manju buku, duži im je vijek trajanja što je posljedica izostavljanja četkica i mehaničkog trošenja na njima. Također, nema iskrenja na četkicama, a smanjene su i elektromagnetske smetnje. Maksimalna snaga BLDC motora je ograničena jedino toplinom koju takav motor može izdržati, tj. toplina koju mogu izdržati namotaju, a toplina razvijena u motoru proporcionalna je kvadratu struje motora:  $W_{\text{topline}} = I^2 * R$

**Mane** su mu relativno visoka cijena u usporedbi s DC motorima i nužnost uporabe električnih kontrolera brzine (ESC – electric speed controller) bez kojih BLDC motor ne može biti u funkciji. Također je mana ovakvog motora nedostatak startnog momenta, to jest, ako je opterećen određenom silom pri pokretanju, on se neće moći pokrenuti. Ukoliko se počne okretati minimalnom brzinom, već tada može početi savladavati određeni moment, to jest, kod električnog romobila nakon postignute brzine od oko 5km/h moguće je početi ubrzavati.



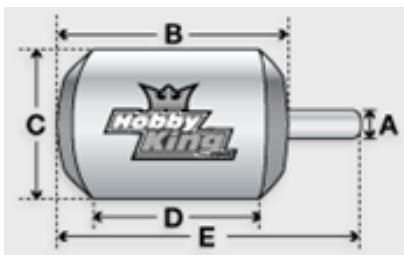
Slika 23. Karakteristike BLDC motora [5]

### 7.3. Specifikacije elektromotora

Za električni romobil potreban je relativno snažan BLDC motor koji je u praksi namijenjen samo za model RC aviona, ali prednost je što je BLDC motor s tolikom snagom daleko manji od bilo kojeg DC motora tolike ili čak manje snage. Snaga motora ovisi o naponu na koji je spojen i o struji koju u određenom trenutku povlači iz baterija. Struja je proporcionalna opterećenju motora, što znači da će struja kod pokretanja, to jest, ubrzanja biti vrlo velika, a kod kontinuiranog gibanja ili usporavanja znatno manja. Maksimalna snaga koju ovaj motor daje za pogon romobila je 2240W. Za navedenu snagu motor povlači 70A struje uz približno konstantan napon baterije od 32V:

$$P = UI, \quad 2240W = 32V * 70A$$

U dolje prikazanoj tablici navedene su dimenzije i ostale specifikacije motora:



Slika 24. Dimenzije BLDC motora [6]

Max Current (A):	70
Resistance (mh):	19
Max Voltage (V):	44
Power(W):	2400
Shaft A (mm):	8
Length B (mm):	86
Diameter C (mm):	59
Can Length D (mm):	62
Total Length E (mm):	113

Kv (rpm/v):  
168  
Weight (g):  
840



Slika 25. BLDC motor na romobilu [7]



## 7.4. Specifikacije kontrolera (ESC-a)

### Specifikacije:

- Max Cont Current: 120A
- Max Burst Current: 140A for 10 seconds
- Input Voltage: 4-14 cells li-XX or 12-42 Ni-MH/Ni-Cd battery
- BEC: None (OPTO)
- PWM: 8~16 KHz
- Max RPM: 240,000rpm for 2 Poles Brushless Motor
- PCB Size: 68x53mm
- Weight: 164g (including wires)



Slika 26. Kontroler za motor (ESC) [8]

### Upravljanje kontrolerom:

ESC kontrolerima namijenjenim za Brushless motore koji najčešće pogone daljinski upravljane letjelice ili druga vozila upravlja se na sljedeći način: preko mikrokontrolera potrebno je poslati kratki impuls, i to ponavljati u periodima od 20ms (20 000 $\mu$ s). Ovisno o dužini trajanja tog kratkog impulsa, motor će se vrtiti brže ili sporije. Pa tako impuls u trajanju od 500 $\mu$ s označuje mirovanje, a impuls koji traje 1500 $\mu$ s označuje maksimalnu brzinu vrtnje motora. Tako se u razmaku od 500-1500 $\mu$ s može vrlo precizno regulirati željena brzina vrtnje motora.

## 8. Elektronika

Svakom vozilu, ili općenito, bilo kojem električnom uređaju, potrebne su elektroničke komponente za izvršavanje raznih funkcija. Čak i kod najjednostavnijih električnih uređaja možemo pronaći poneku sklopku ili lampicu koja nešto označava na spomenutom uređaju. U električnim vozilima taj sustav elektroničkih komponenti znatno je složeniji.

Takav sustav u električnim vozilima ima nebrojeno mnogo zadataka od kojih su samo neke: mjerenje i prikazivanje napona baterija sumirano ili svake ćelije baterija zasebno, mjerenje struje, snage i temperature motora, kontroliranje i modificiranje kontrole za ubrzavanje (temeljeno na trenutnoj brzini i/ili struji motora) itd.

### 8.1. Svrha i namjena tiskanih pločica za električni romobil

Tiskane pločice u romobilu izrađene su vrlo popularnim i jednostavnim postupkom izrade, tako zvanim foto-postupkom. U romobilu se nalaze dvije tiskane pločice - jedna koja je zadužena za upravljanje motorom i druga koja služi za prikaz podataka, mjerenje raznih fizikalnih veličina i mnoštva drugih priključaka o kojima će biti više riječi u nastavku.

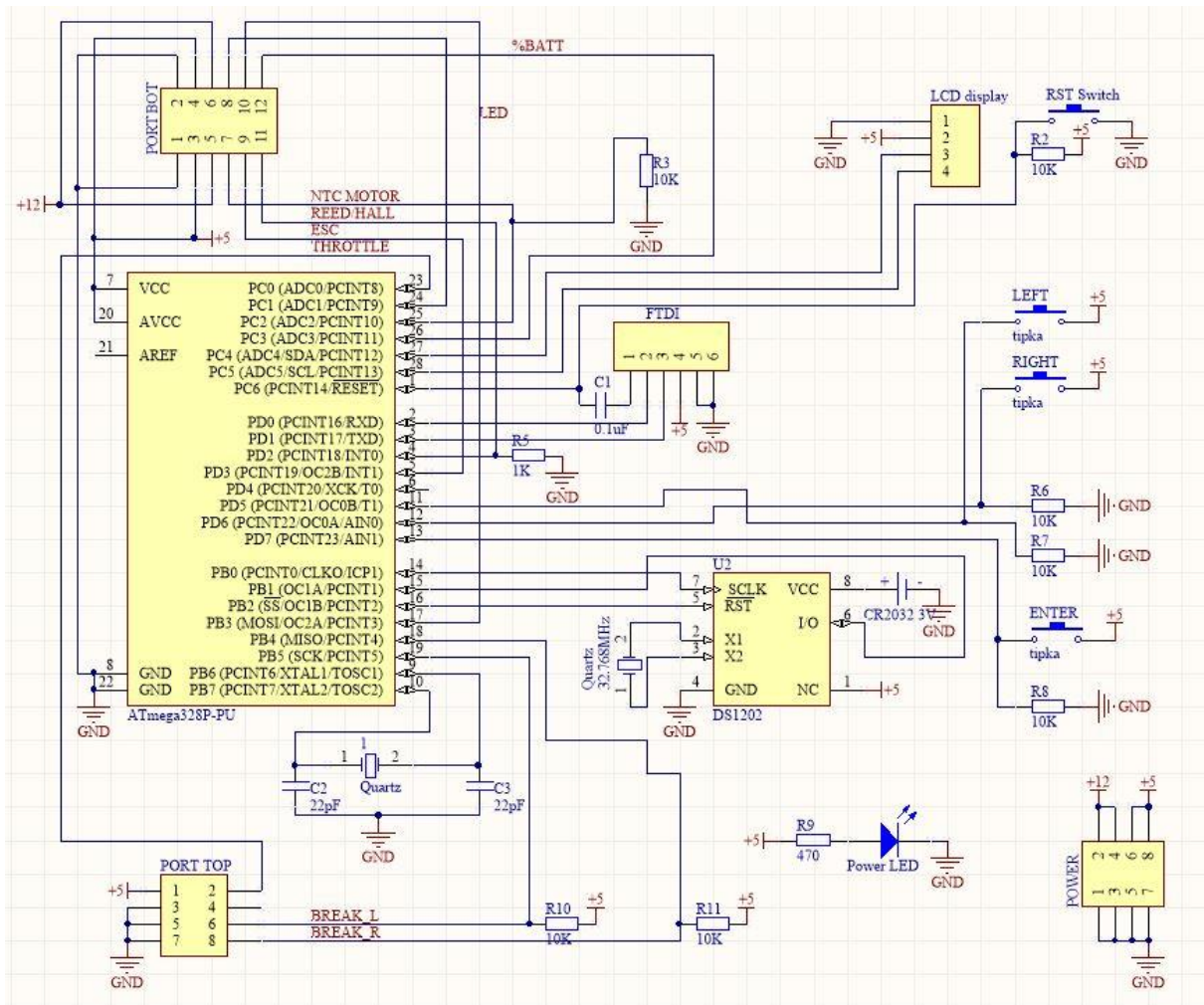
Spomenuta druga pločica pričvršćena je za volan romobila, a iznad nje se nalazi veliki 20x4 ekran (20 znakova u 4 reda) na kojem se prikazuju razni podatci tokom vožnje. Kako ekran ima samo 4 retka, sučelje je napravljeno tako da nam pritisak na gumb za lijevo ili desno omogućuje mijenjanje željenih podataka koji se prikazuju. Neki od podataka koji se prikazuju su sljedeći:

- Trenutna i maksimalna brzina
- Trenutna i maksimalna potrošnja struje
- Trenutna i maksimalna snaga na motoru
- Temperatura motora
- Vrijeme vožnje
- Prijedni put
- Predviđanje puta koji je moguće prijeći s obzirom na stanje ostalih podataka
- Trenutna i prosječna brzina vožnje
- Stanje baterije u postocima (0-100%)
- Trenutno vrijeme (MM:HH:SS)

U donjem pretincu romobila (ispod podnice) nalazi se „power“ pločica. Na nju su spojene 4 baterije koje pogone motor te jedna koja napaja elektroniku i *LED* svjetla. Pločica je dizajnirana na način da kada se baterije jednom spoje one se više ne odspajaju utoliko jer postoji poseban priključak koji se spaja na punjač i puni sve baterije motora istovremeno.

Gornja i donja pločica povezane su 12-pinskim, takozvanim *flat* kabelom koji prenosi električne signale iz gornje u donju pločicu i obrnuto.

## 8.2. Električne sheme pločica



Slika 27. Shema donje pločice



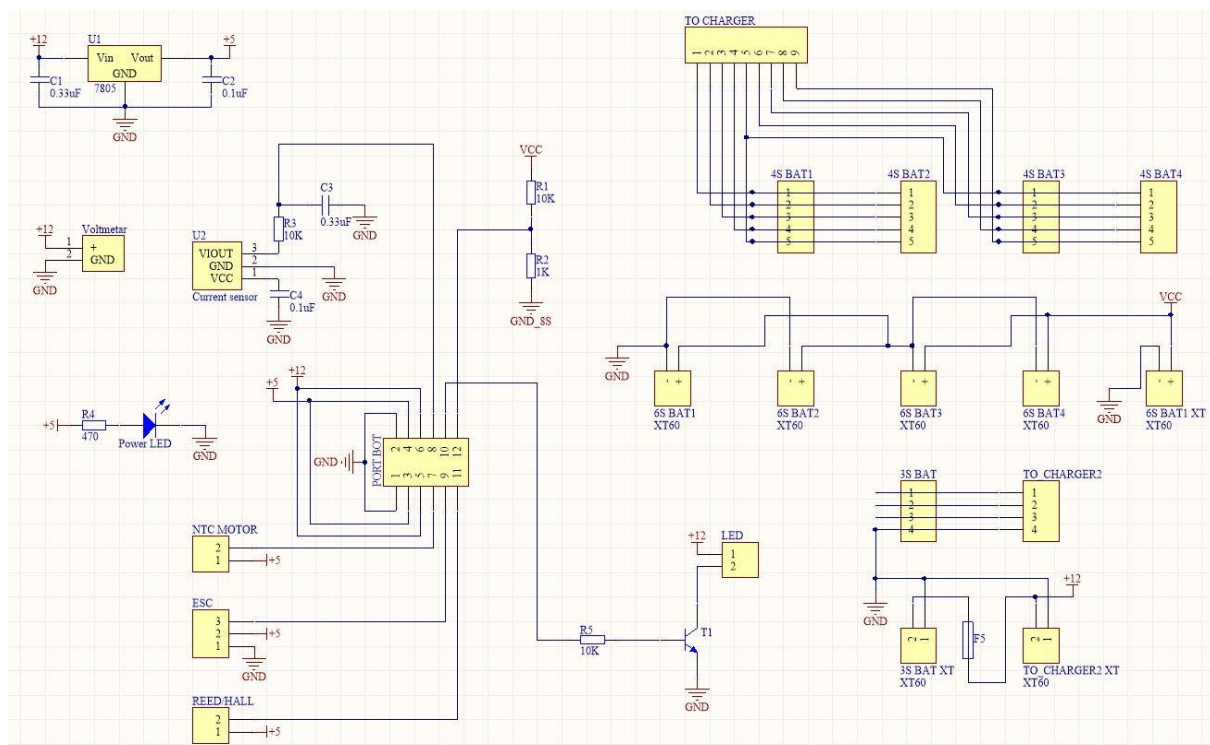
Slika 28. FTDI adapter [9]

Na prikazanoj pločici, kao što je već rečeno, nalazi se mikrokontroler. On se programira pomoću malog adaptera (FTDI adapter na slici lijevo) koji povezuje računalo i mikroprocesor. Na vrhu sheme nalazi se konektor za povezivanje gornje i donje pločice. Preko spomenutog konektora gornja pločica dobiva napajanje (+12V, +5V, 0V). Ostale žice su signalne, a služe za slanje i primanje različitih podataka na električnom romobilu.

Mjerenje temperature motora vrši se pomoću NTC termistora koji je spojen s dodatnim otpornikom u naponskom dijelilu te na izlazu daje napon ovisan o temperaturi. REED/HALL žica spojena je na prednjoj vilici električnog romobila na REED relay koji, uz magnet pričvršćen na prednjem kotaču, služi za mjerenje brzine. Žica s oznakom ESC spojena je na kontroler za motor, a žica throttle spojena je s ručicom gasa na volanu.

Kako bi saznali i točno vrijeme, potreban nam je čip *DS1202* koji, spojen na zasebnu bateriju, uvijek pamti koliko je sati, čak i kada je glavno napajanje ugašeno. Na konektoru pod nazivom „PORT TOP“, osim ručice za gas, spojene su i kočnice koje, kada su u funkciji, šalju mikrokontroleru informaciju da upali stop svjetlo. Power priključak, osim za prednja LED svjetla, služi i kao napajanje nekog dodatnog uređaja na romobilu (na primjer punjač za mobitel). Tipke LEFT i RIGHT služe za mijenjanje „stranica“ na ekranu, to jest, prikaz različitih podataka. Tipka ENTER služi za namještanje pojedinih parametara zajedno s LEFT i RIGHT tipkama.

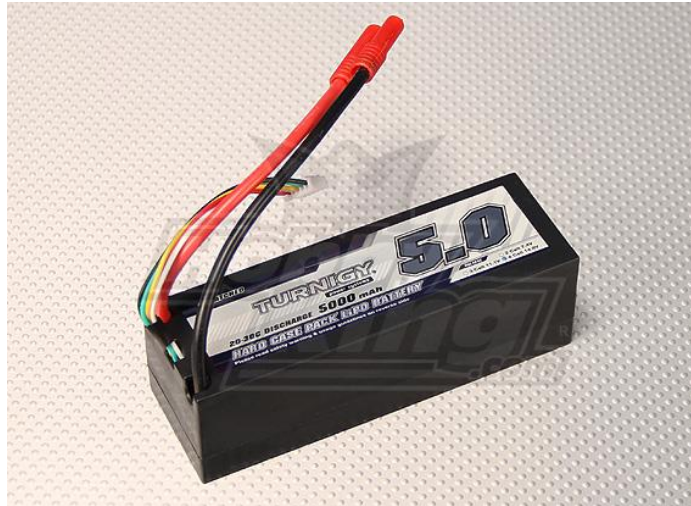
Na sljedećoj slici je prikazan shematski prikaz donje pločice:



Slika 29. Shema donje pločice

Dok je gornja pločica zadužena za upravljanje, mjerenje i prikaz podataka, donja pločica služi za prijenos signala do gornje gdje se oni obrađuju te za spajanje svih baterija potrebnih za napajanje motora i elektronike.

Baterije koje služe za napajanje motora su LiPo (Lithium Polymer) baterije koje mogu predati vrlo veliku struju za razliku od NiMh ili nekih drugih baterija. To je kod električnog romobila poželjno svojstvo utoliko jer motor ponekad povuče i preko 70A struje. One su i poprilično opasne zbog spomenute činjenice što mogu predati veliku struju. Kada su u kratkom spoju, njihova struja je veća i od 100A te iako su žice dovoljno debele da podnose takvu struju, postoji opasnost od zapaljenja baterija. Naglašeni problem u romobilu riješen je mjerenjem jakosti struje u svakom trenutku (i prikaz istoga na ekranu) te u slučaju kada njena vrijednost prijeđe 90A, motor se odmah ugasi.

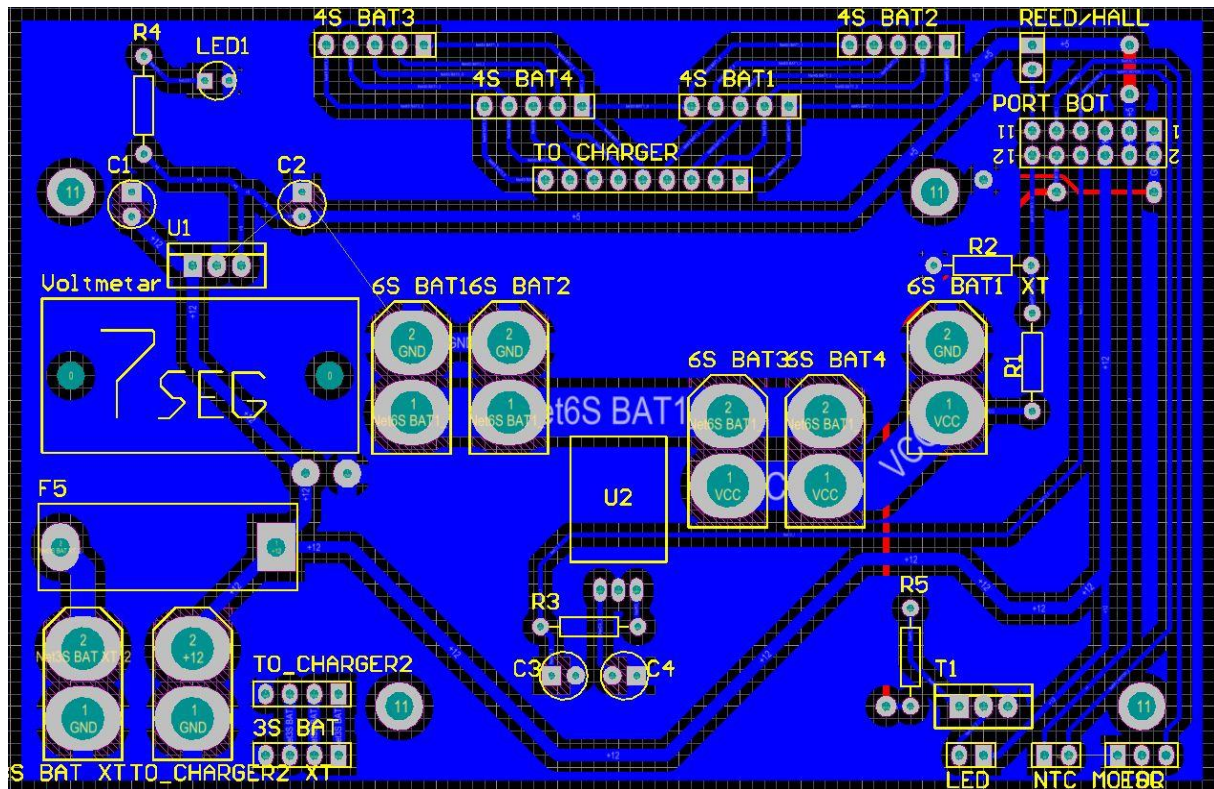


Slika 30. 4S LiPo baterija [10]

U donjem pretincu romobila nalaze se 4 takve baterije za pogon motora i jedna manja za elektroniku i osvjetljenje. Po dvije baterije spojene su u paralelu (što povećava njihov kapacitet), a zatim ta ista dva para baterija u serijski spoj (što povećava ukupni napon). Kako je jedna baterija kapaciteta 5000mAh i nazivnog napona 14.8V, motor ukupno na raspolaganju ima 10Ah (10 000mAh) i 29.6V napona. Spomenute baterije interno su spojene kao 4 ćelije od 5000mAh spojene u seriju (4S), a nakon spajanja tih dodatnih baterija u seriju dobivamo 8S, to jest, gore navedeni napon od 29.6V. Baterije su neprestano spojene na pločicu, a mijenja se samo izlazni konektor koji ide, ili prema kontroleru motora, ili u punjač za punjenje baterija. Na slici 29, u gornjem desnom uglu, prikazani su konektori za takozvane balans kabele koji služe tome da svaka ćelija u baterijama poprma isti napon (4.20V kada je puna).







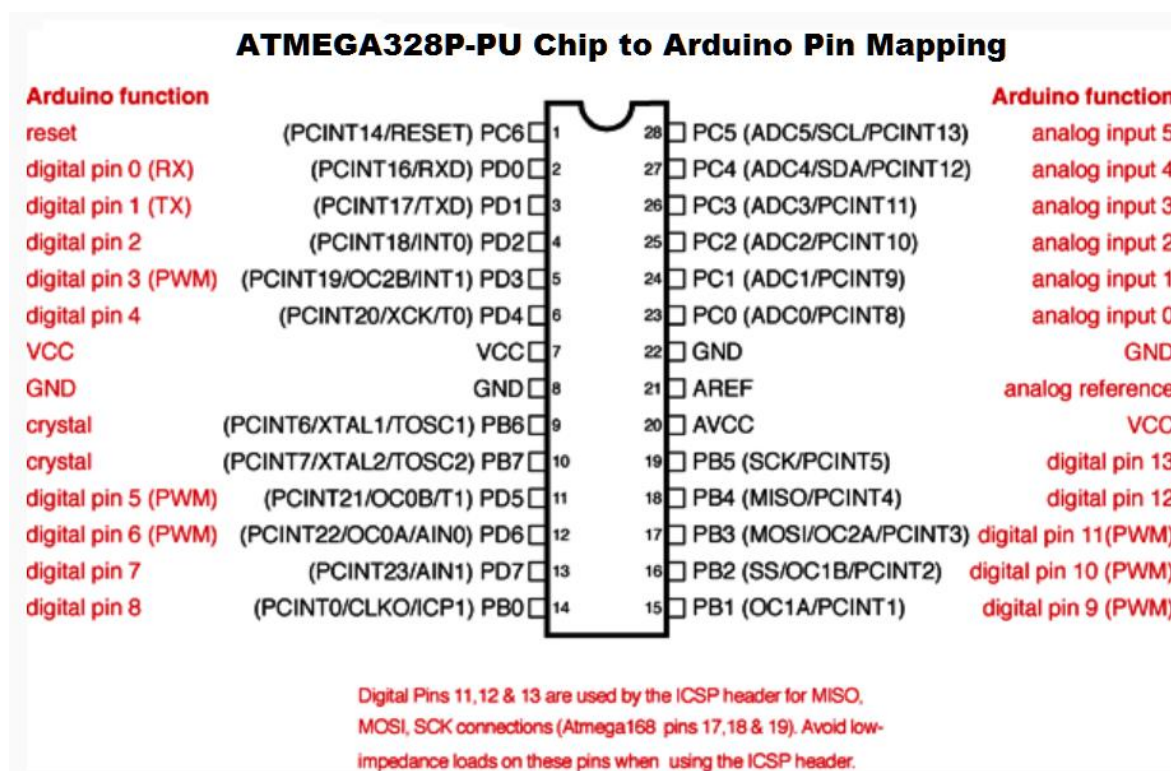
Slika 32. Donja PCB pločica

Donja PCB pločica bitno je drugačija od gornje utoliko što je ona spojena na baterije motora te mora podnositi visoki napon (32V) i veliku struju (do 100A). Zbog visokog napona, potrebno je razmaknuti sve vodove, u ovom slučaju su oni razmaknuti 1mm. Zbog velike struje vodovi moraju biti (jako) debeli. Na mjestima gdje se nalaze debeli vodovi (200milsa tj. 5mm) zalemljene su još i prave žice koje mogu prenositi struju do 100A. Napon baterija motora mjeri se pomoću otporničkog dijelila, a postotak napunjenosti baterije prikazuje se na ekranu iznad gornje pločice. Napon baterije za elektroniku prikazuje se direktno na pločici pomoću digitalnog voltmetra (na slici 7seg), no ta baterija, iako mnogo manja od baterija za motor, ima autonomiju približno 50km vožnje ponajprije zbog male potrošnje struje elektroničkih komponenti i LED svjetala.

## 9. Opis elektroničkih komponenti

### 9.1. Atmega328p – mikrokontroler

Atmega 328p vrlo je popularan mikrokontroler, a sastavni je dio Arduino UNO-a. U električnom romobilu on se programira i ponaša upravo kao da je Arduino UNO. Ima 28 pinova od kojih je 12 digitalnih ulaza ili izlaza, a 5 analognih ulaza. Na romobilu se koriste svi navedeni pinovi što ukazuje na činjenicu kako je mikroprocesor u potpunosti iskorišten.



Slika 33. Atmega328p - raspored pinova [11]

Popularnost je stekao zbog vrlo jednostavnog i intuitivnog programiranja. Čak bi i „programerski laici“ mogli, uz određeni priručnik, naučiti paliti i gasiti različite digitalne izlaze ili upravljati određenim jednostavnim uređajem. Ipak, u ovom projektu izrade električnog romobila pokazalo se kako navedeni mikrokontroler može upravljati znatno složenijim uređajima i na temelju mjerenja različitih ulaznih signala, upravljati motorom, svjetlima i LCD zaslonom na volanu romobila.



## 9.2. Podešavanje brzine romobila

Brzinu je moguće podešavati preko takozvane gas-ručice. U njoj su smještena 2 magneta te između njih spomenuti *hallov* senzor. Ručica ima tri žice - dvije za napajanje i jednu za signal. Spaja se na „*PORT TOP*“ koji se nalazi na gornjoj tiskanoj pločici. Hallov senzor napajan je s 5V DC strujom, a na svom izlazu daje struju u rasponu od 1 do 4 V. Taj napon daje analogni signal i spaja se na analogni ulaz koji ulazi u mikrokontroler. Dalje se taj podatak od 1-4V uz rezoluciju od 1024 dijela u rasponu od 0-5V ( $1V=204$ ,  $4V=819$ ) obrađuje i pretvara u upravljački signal za motor. Signal se najprije šalje u ESC (*Electric Speed Controler*). Brzinom motora se upravlja preko ESC-a slanjem PWM signala. PWM signal je pulsno-širinska modulacija signala kojom nekom uređaju dajemo uvijek isti napon, ali zbog razlike u vremenima kada je vrijednost napona maksimalna i onda kada je 0, uređaj dobiva neku srednju vrijednost napona i u ovisnosti o tome se različito ponaša s rezolucijom od 256 bita. Zaključujemo da 0 označava 0V i motor miruje, a 256 označava 5V što šalje motoru signal da se vrti najvećom brzinom. Sve ostale vrijednosti su srednja vrijednost napona dobivena pulsno-širinskom modulacijom i u ovisnosti o spomenutoj srednjoj vrijednosti napona u signalnomvodu, motor se vrti brže ili sporije.

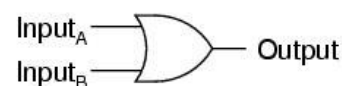


Slika 34. Ručica za gas [12]

## 9.3. Mehaničke kočnice

Kočnice na romobilu su mehaničke, uz dodatno postavljene prekidače. Oni se ponašaju kao NC (normally closed) prekidači, to jest, ukoliko ne pritisćemo kočnicu, prekidači će cijelo vrijeme biti zatvoreni. Kako su prekidači većinu vremena „pritisnuti“, na pločici su spojeni tako da je signal koji oni daju dok se ne koči romobilom (dok su oni pritisnuti), logička „0“. Preko takozvanog ILI sklopa, u mikrokontroleru je programirano da, ukoliko se bilo kojom (prednjom ili stražnjom) kočnicom zakoči, signal bude logička „1“. Ovako izgleda takozvani „truth table“ spomenutog signala:

2-input OR gate

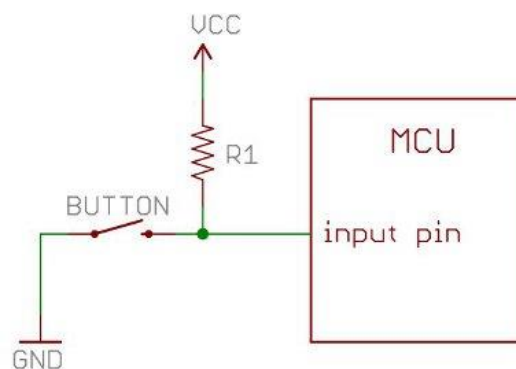


A	B	Output
0	0	0
0	1	1
1	0	1
1	1	1

Slika 35. Logička tablica vrijednosti za ILI sklop [13]

Kada je prekidač pritisnut (kočnica nije pritisnuta), signal je direktno spojen na *GND*, to jest, na  $0V$  iz razloga što, činjenično, struja uvijek teče tamo gdje je otpor manji. Čim se kočnica pritisne, signal putuje preko otpornika na *VCC*, to jest, na  $+5V$  napon i daje logičku jedinicu mikrokontroleru.

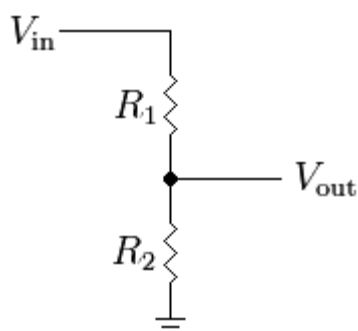
S tim signalom kočenja dalje se može lako upravljati. Na električnom romobilu, kada je bilo koja ručica kočnice pritisnuta, stražnja *LED* svjetla svijetlit će punom snagom čime će se signalizirati kočenje.



Slika 36. Shema spajanja pull down otpornika [14]

## 9.4. Mjerenje napona baterija

Napon na baterijama najlakše se može mjeriti pomoću naponskog, odnosno otporničkog dijelila. Ako je napon baterije  $30V$ , a analogni ulaz mikrokontrolera može mjeriti napone samo od  $0$  do  $5V$ , u svrhu smanjenja napona, taj posao odrađuje naponsko dijelilo.



$$U_{izl,max} = \frac{R_2}{R_1 + R_2} * U_{ul,max}$$

$$U_{ul,max} = 33.6V$$

$$U_{ul,min} = 29.6$$

$$U_{izl,max} = 5V$$

$$R_2 = 1k\Omega, R_2 = ?$$

Slika 37. Naponsko djelilo [15]

$$R_2 = \frac{R_1}{\left(\frac{U_{ul,max}}{U_{izl,max}} - 1\right)} = \frac{10000}{\left(\frac{33.6}{5} - 1\right)} = 1748$$

Potrebno je uzeti manju vrijednost otpora od izračunate, a to je  $1500\Omega$ . S danim vrijednostima otpora, dobili bi sljedeći napon na ulazu mikrokontrolera za punu i praznu bateriju:

$$U_{izl,max} = \frac{1.5}{1.5 + 10} * 33.6 = 4.38V;$$

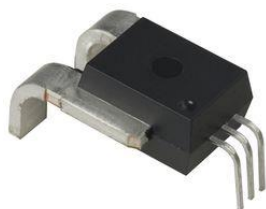
$$U_{izl,min} = \frac{1.5}{1.5 + 10} * 29.6 = 3.86V$$

Mikrokontroler će razliku tih napona pretvoriti u postotke napunjenosti baterije, to jest,  $0\%$  za  $3.86V$ , a  $4.38V$  za  $100\%$ . Razlika tih napona je:  $4.38V - 3.86V = 0.52V$ , a uz rezoluciju mikrokontrolera od  $0.004V$  to je  $\frac{0.52V}{0.004V} = 130$ . Dakle, svaki postotak baterije moći će se točno prikazati.

## 9.5. Mjerenje struje motora

Struju je vrlo korisno mjeriti kako bi znali koliko je opterećenje na motoru te da se možemo zaustaviti kada i ako je motor preopterećen. Također je moguće preko mikrokontrolera isprogramirati gašenje motora kada iznos struje prijede neki određeni iznos.

Mjerenje struje može se izvršiti na više načina od kojih je jedan od najkompatibilnijih preko Hallovog senzora za mjerenje struje. Takav uređaj daje analogni signal koji je ovisan o struji koja prolazi kroz žicu u njegovoj blizini.

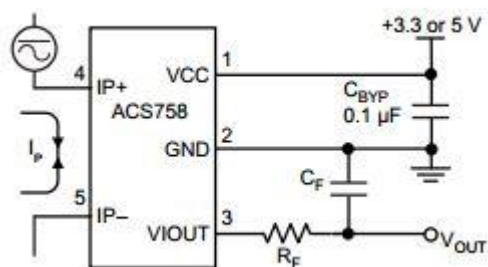


Slika 38. Senzor struje [16]

Kako je snaga umnožak struje i napona, i ona se može prikazivati na ekranu ako se pomnože te dvije izmjerene vrijednosti.

### Karakteristike senzora:

Senzor struje korišten u električnom romobilu ima oznaku ACS758ECB-200U. Zadnja brojka u njegovoj oznaci označuje maksimalnu vrijednost struje koju je moguće izmjeriti, to je u ovom slučaju 200A, i to samo u jednom smjeru (U-unidirectional). Senzor na svojem izlazu daje analogni napon od približno 0 do 5 volti, a taj napon linearno je ovisan o struji koja prolazi sensorom. Na slici 39. prikazan je način spajanja senzora.



Slika 39. Način spajanja senzora struje

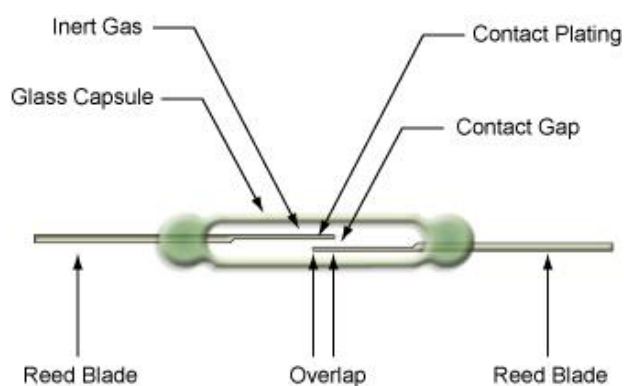
## 9.6. Mjerenje brzine romobila

Jedan od najjednostavnijih načina mjerenje brzine je uz pomoć *REED relay*. Ta elektromehanička komponenta ponaša se kao prekidač koji je normalno otvoren, a kada se pojavi u nekom magnetskom polju, zatvori se. Kako je to mehanička komponenta (mehaničko otvaranje i zatvaranje), on kao takav nije najkompatibilnije rješenje za takvu primjenu, prvobitno zbog svoje brzine. No iz razloga što brzina romobila ne prelazi 50km/h, on je sasvim dovoljan i vrlo pouzdano može računati brzinu.

*REED relay* smješten je na vilici romobila, a

magnet je pričvršćen za kotač. Svakim okretom kotača on šalje impuls mikrokontroleru koji tada obrađuje dobiveni podatak. On mjeri vrijeme od jednog do drugog impulsa i na temelju toga izračunava brzinu romobila koja se prikazuje na LCD ekranu.

Sa spomenutim sensorom, uz dobro programiranje, moguće je složiti mjerenje prijednog puta, prosječnu brzinu, maksimalnu brzinu, trenutnu brzinu i ostalo.



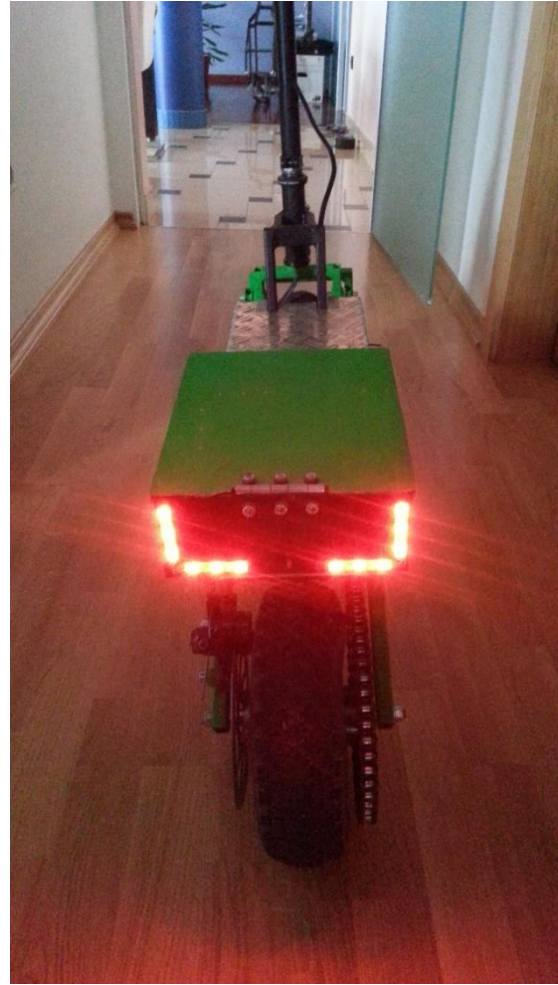
Slika 40. REED relay [17]

## 9.7. LED osvjetljenje

Na električnom romobilu postoje prednja i stražnja LED svjetla. Sa stražnje strane nalazi se 12 LED dioda koje svjetle punom jačinom u slučaju kočenja, dok u normalnoj vožnji svijetle znatno slabije.



*Slika 41. 40% snage (normalna vožnja)*



*Slika 42. 100% snage (kočenje)*

S prednje strane dvije su LED diode, svaka jačine 6W koje služe za osvjetljenje puta ili ceste po kojoj se vožnja vrši. Sva svjetla pale se prilikom paljenja romobila, iako je prednje svjetlo moguće ugastiti u slučaju kada ono nije potrebno.





*Slika 43. Prednje LED svjetlo ukupne snage 12W*



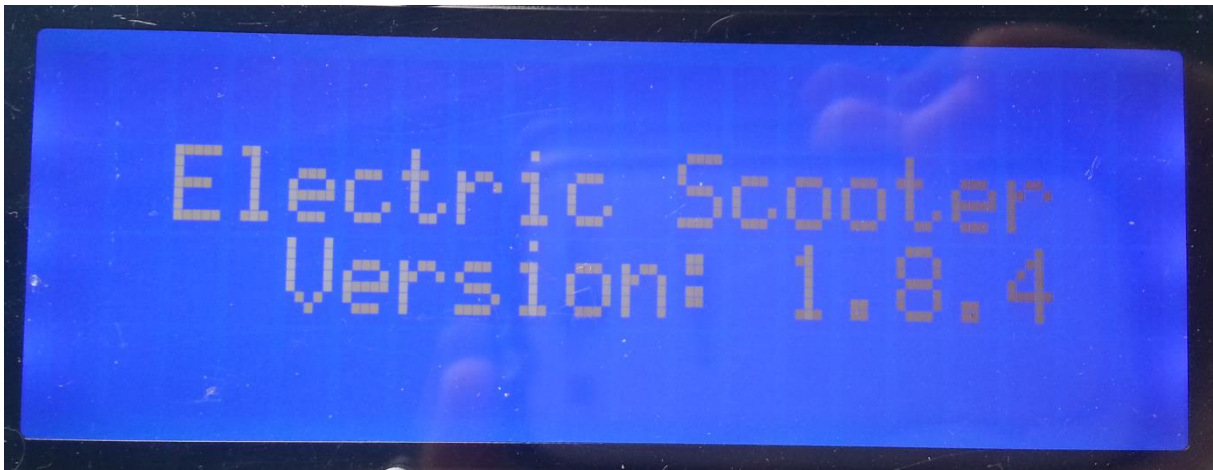
## 10. Softverski dio i programiranje

Nakon izrađenog hardverskog dijela romobila i spajanja svih elektroničkih komponenti, na red je došlo programiranje mikrokontrolera koji će upravljati zadanim procesima.

Kao što je već rečeno, „mozak“ romobila nalazi se u sićušnom Atmelovom mikrokontroleru ATmega328p. Na njegove analogne ulaze spojeni su senzori za struju, napon i temperaturu, dok su na digitalne ulaze spojene tipke, kočnice i senzor za brzinu, a paljenje stražnjeg stop-svjetla vrši se pomoću digitalnog izlaza. Na mikrokontroler još je spojen drugi čip koji je zadužen za pamćenje i računanje vremena.

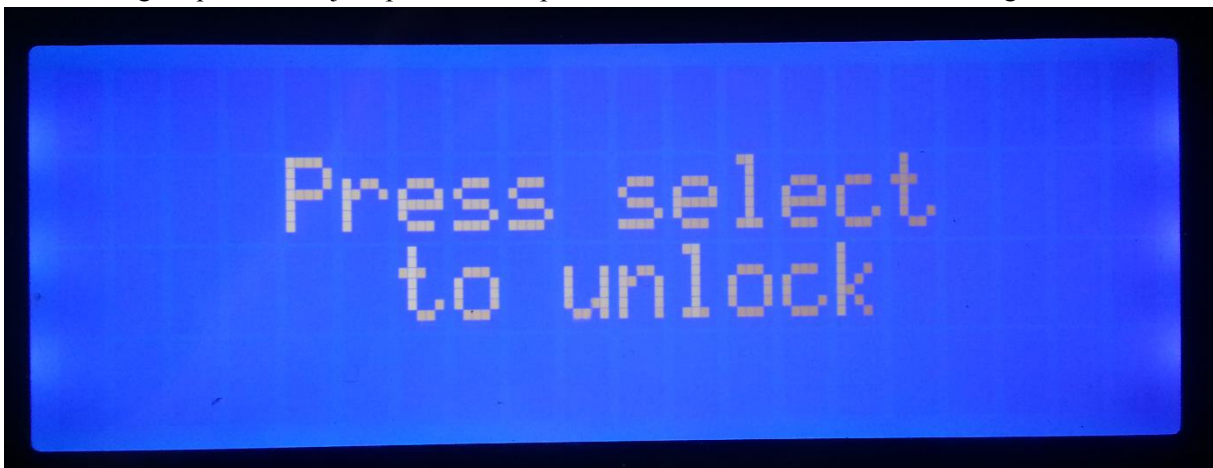
Pomoću tri tipke na prednjoj strani pločice moguće je upravljati izbornicima na ekranu električnog romobila.

Kod paljenja električnog romobila, najprije se prikazuje ekran s trenutnom verzijom softvera (koja je još u fazi razvoja). Spomenuti ekran na slici prikazuje se dvije sekunde te nakon toga prelazi na „Press select to unlock“ ekran.



Slika 44. Boot screen

Nakon toga potrebno je pritisnuti tipku „select“ kako bi ušli u glavni izbornik.



Slika 45. Unlock screen

## 10.1. Glavni izbornik

Na sljedećim slikama prikazano je što se u pojedinom izborniku prikazuje na ekranu:

Na donjoj slici prikazan je izgled glavnog izbornika („main menu“). To je ekran koji je najaktivniji za vrijeme vožnje, a programiran je za prikazivanje raznovrsnih korisnih podataka. U gornjem lijevom uglu ispisuje se trenutna temperatura motora u celzijevim stupnjevima (°C). Na sredini prvog retka prikazano je trenutno vrijeme (koje se odbrojava i kada je romobil ugašen), a u desnom dijelu prvog retka prikazana je napunjenost baterija za motor u postocima. Drugi redak prikazuje trenutnu brzinu i prijeđeni put u metrima. U trećem retku prikazuju se dvije vrijednosti gasa. Drugi broj prikazuje željenu to jest zadanu vrijednost gasa (to jest jačina pritiska ručice gasa od 0-100), a prva vrijednost se dalje prosljeđuje u kontroler pa nadalje u električni motor. Prva vrijednost rijetko kada se poklapa s drugom jer ovisi o brzini, struji, pa čak i temperaturi motora. Ukoliko je struja ili temperatura motora visoka, gas neće moći postići željenu vrijednost nego će biti onoliko manji koliko su struja i temperatura daleko izvan dozvoljenog područja. U zadnjem retku prikazuje se trenutna potrošnja snage na motoru.



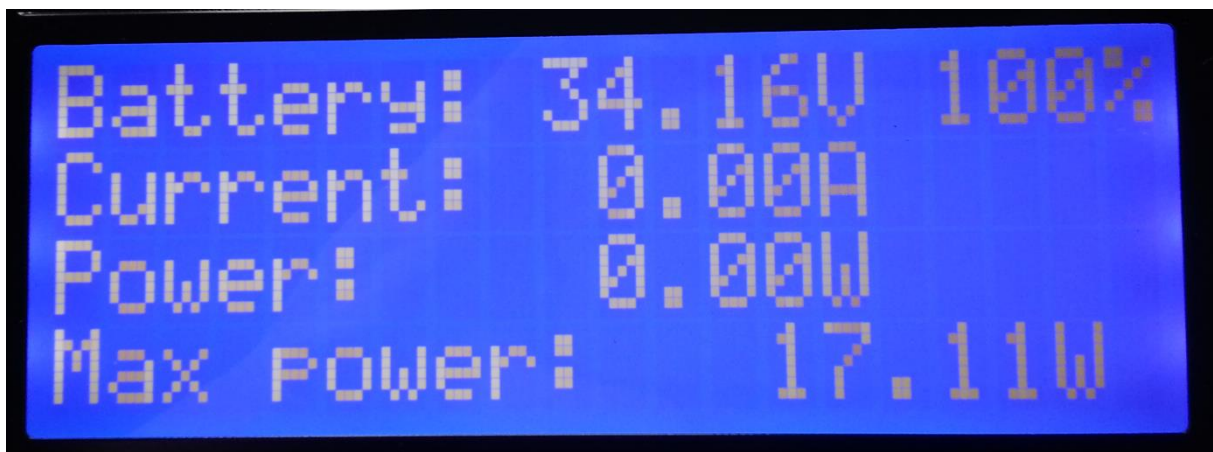
Slika 46. Main menu

U izborniku „speed menu“ prikazuje se sve vezano uz senzor za mjerenje brzine. Pa tako na temelju poznatog opsega kotača (tj. poznatog i uvijek istog prijeđenog puta) i varijabilnog vremena potrebnog za puni okret kotača mikrokontroler izračunava brzinu električnog romobila. Maksimalna brzina se sprema u zasebnu varijablu i pohranjuje samo maksimalnu vrijednost brzine. Zbrajanjem okretaja i množenjem s opsegom kotača dobiva se pređeni put („distance“). Vrlo je korisno znati koliko još kilometara možemo prijeći s odredbenim stilom vožnje (tj. odredbenom brzinom). Ta informacija prikazuje se u zadnjem retku, a izračunava se na temelju trenutne brzine i trenutne struje potrebne za kontinuirano gibanje pri toj brzini. Ta vrijednost obično varira od 5-20km jer ovisi o akceleraciji (pri kojoj je potrošnja struje veća pa stoga i mogući put manji)



Slika 47. Speed menu

Prilikom testiranja ili kada želimo vidjeti ponašanje motora u pojedinom trenutku u vožnji, bitan je i izbornik prikazan slikom 47. Na njemu su grupirani podatci o struji, naponu i snazi motora. Napon baterije osim u voltima, prikazuje se i u postotcima (26.4V je 0%). Struja se konstantno mjeri i obnavlja na ekranu, kao i napon baterije, a pomoću te dvije varijable dobiva se snaga. Maksimalna snaga, kao i maksimalna brzina u prethodnom izborniku dobiva se spremanjem maksimalne vrijednosti snage.



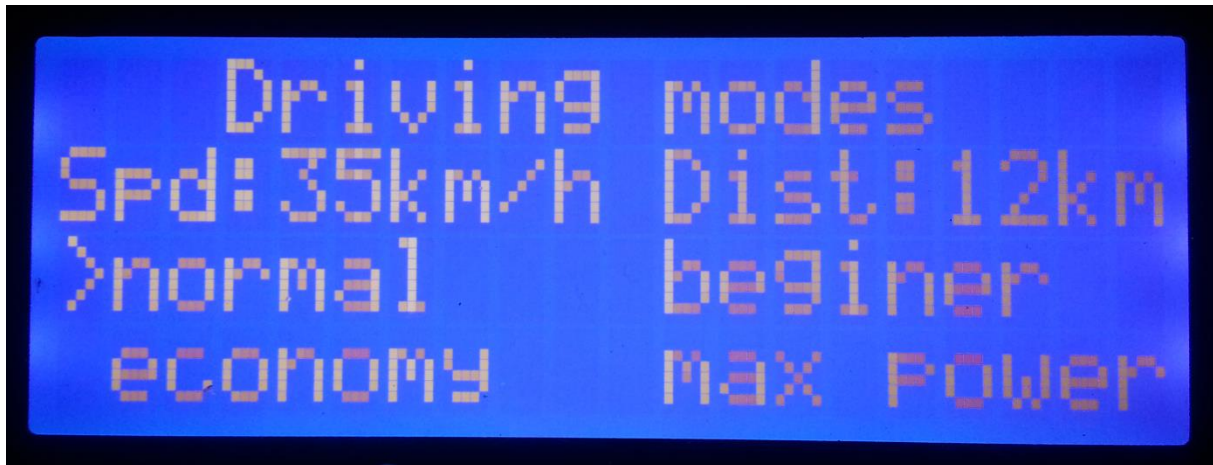
Slika 48. Power menu

Promjena trenutno prikazivanog izbornika vrši se pritiskom na tipke za lijevo ili desno.

## 10.2. Režimi vožnje

Na električnom romobilu programirana su 4 režima vožnje. Željeni režim vožnje moguće je izabrati u izborniku „driving modes“. U pojedinom režimu razlike se temelje na tome koliko je moguće dati gasa motoru, kolika je najveća brzina i koliko najviše struje može motor vući iz baterija. Tako se razlikuju „normal“, „beginer“, „economy“ i „max power“ mačini rada.

Oni se izabiru pritiskom na „select“ tipku, a trenutni režim koji smo odabrali označen je strelicom. U drugom retku prikazana je najveća brzina i predviđena autonomija u kilometrima za zadani režim vožnje.

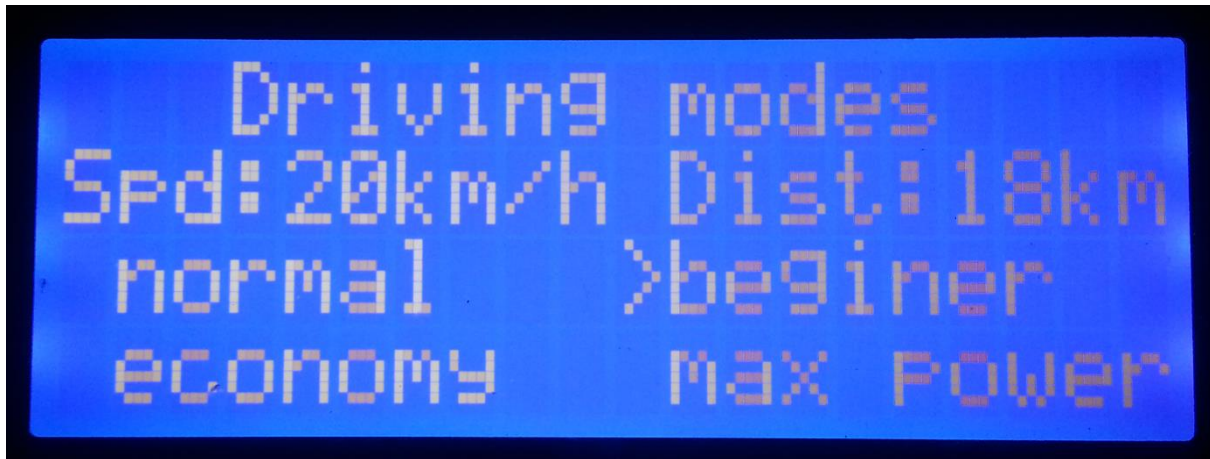


Slika 49. Driving modes (1/4)

Pod režimom normal najveća moguća brzina je 35km/h, a autonomija 12km. Struja do koje nema ograničenja je 50A, a iznad toga se linearno smanjuje vrijednost ubrzanja koje se šalje motoru. Također, ta vrijednost se linearno smanjuje na manjim brzinama. Na primjer, na brzini od 10km/h uz vrijednost na ručici gasa od 100%, motor dobiva 30% energije. Time je moguće ručicu odmah dovesti u krajnji položaj, a mikrokontroler će na temelju brzine i struje izračunati koji će signal motor dobiti u odnosu na ono što korisnik želi.

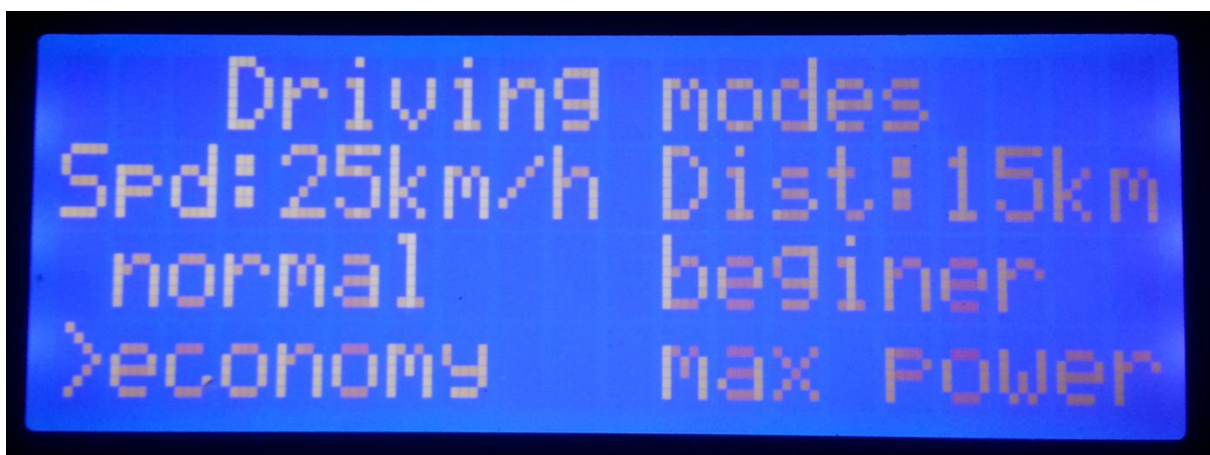
Režim „beginer“ (u prijevodu početnik), kao što ime kaže, osmišljen je za korisnike koji po prvi puta žele iskusiti vožnju na električnom romobilu. U njemu je maksimalna brzina vožnje programirana na 20km/h, a autonomija 18km. Ručica gasa, iako korisnik zada da je 100%, nikada neće poprimiti tu vrijednost, nego će isto kao i kod normalnog režima rada, na temelju struje i brzine davati najveću moguću vrijednost. Struja iznad 20A značajno smanjuje vrijednost ručice gasa, a veliko ubrzanje nije moguće zbog ograničenja s mjerenjem brzine utoliko jer je do 10km/h maksimalna vrijednost ubrzanja 20%.





Slika 50. Driving modes (2/4)

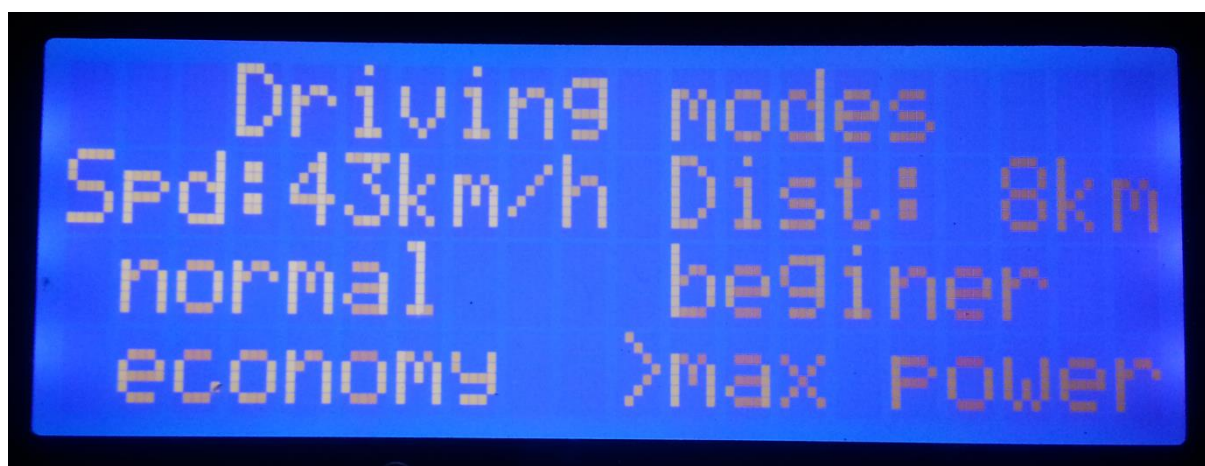
Economy režim je najštedljiviji režim sa zadovoljavajućom maksimalnom brzinom od 25km/h. To je najkompatibilniji režim utoliko što je brzina od 25km/h sasvim dovoljna za gradsku vožnju, a struja motora nikada ne prelazi vrijednost od 40A čime je postignuta i štedljivost. Kontinuirano gibanje od 25km/h zahtjeva konstantnu struju od 25A. Ograničenje je postavljeno na 40A jer samo ubrzavanje uvijek traži veću struju od one pri kontinuiranom gibanju. Predviđena autonomija je 18km (uz napomenu da struja duž cijelog puta neće biti 25A, nego će srednja vrijednost biti manja).



Slika 51. Driving modes (3/4)

U max power mode-u ograničenja nema, ni po struji, ni po brzini. To nam ukazuje na mogućnost kvara motora, kontrolera ili nekog drugog elektroničkog dijela ukoliko naglo ubrzavamo ili dajemo gas dok je romobil na mjestu. Zbog navedenog, spomenuti režim nije pogodan za gradsku vožnju već samo za testiranja. Maksimalna brzina u tom režimu jednaka je maksimalnoj brzini električnog romobila i iznosi 43km/h. Kako pri toj brzini motor vuče preko 50A struje, baterija će se isprazniti vrlo brzo, pretpostavljajući za manje od 8km vožnje pri maksimalnoj brzini. Bez obzira na vlastito i prisilno hlađenje motora, zbog relativno velikih struja, sve komponente se jako zagrijavaju što je još jedno od nepoželjnih svojstava koje je na manjim strujama (tj. brzinama) neprimjetno.





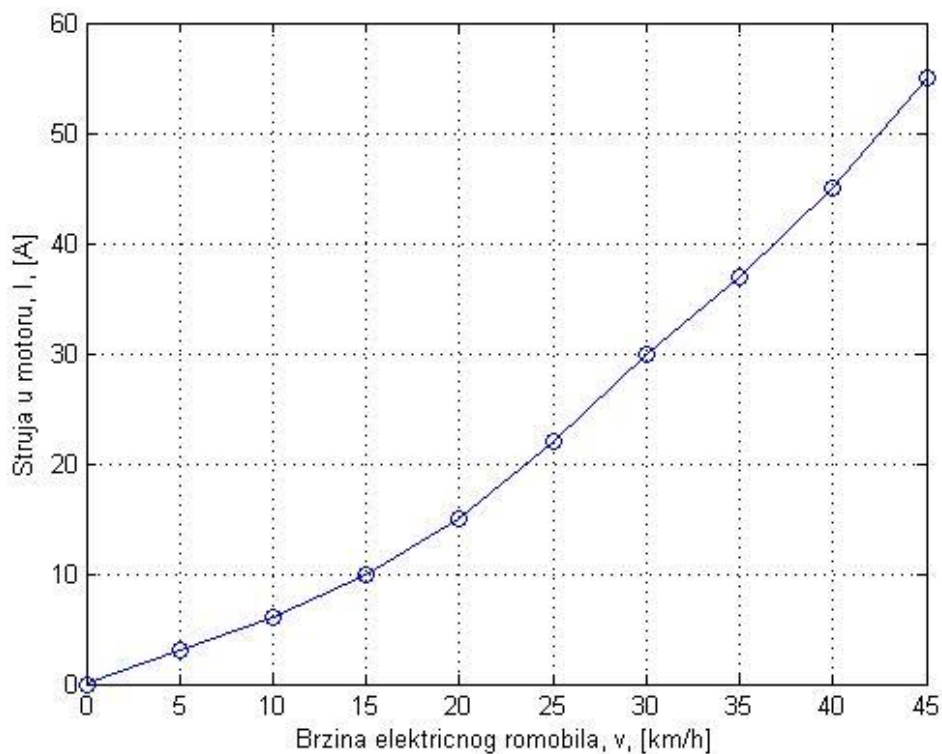
*Slika 52. Driving modes (4/4)*

## 11. Testiranja i troškovi

### 11.1. Ovisnost struje elektromotora o njegovoj brzini

Dijagram potrošnje struje u ovisnosti o brzini je vrlo koristan po pitanju optimiranja brzine vožnje. Autonomija romobila ovisi o jačini struje koju baterije predaju motoru. Kapacitet baterije od  $5Ah$  označava da takva baterija može davati  $5A$  konstantne struje punih sat vremena to jest  $1A$  struje punih 5 sati. Ako motor konstantno vuče primjerice  $50A$ , baterija će trajati samo 6 minuta vožnje. O tome koliko će motor vući struje ovisi o opterećenju elektromotora u vožnji. Njegovo opterećenje pak ovisi o brzini romobila, otporu zraka, trenju, to jest, podlozi po kojoj se vožnja odvija, napumpanosti guma i ostalom. Kod najsporije vožnje motor će vući vrlo malo struje iz baterija što nam daje mogućnost prelaženja velikih udaljenosti. Kod brze vožnje, motor troši puno više struje i mogući prijeđeni put se drastično smanjuje. Iz tog razloga vrlo je korisno izraditi krivulju struje motora u ovisnosti o brzini te tako možemo vidjeti na kojoj brzini ćemo moći prijeći relativno dug put s dovoljno velikom brzinom.

Mjerenje ovisnosti struje elektromotora o brzini romobila imalo je diskretan oblik. Svakih  $5km/h$  očitavala se struja motora. Vožnja na određenoj brzini trajala je neko vrijeme kako bi gibanje bilo jednoliko kontinuirano, a ne ubrzano (kada bi struja bila veća).



Slika 53. Dijagram brzina-struja

Usporedimo sada mogući prijeđeni put s **najmanjom** i **najvećom** brzinom:

$$v_{\min} = 5 \frac{\text{km}}{\text{h}}, I = 3A$$

kapacitet baterija: 10 Ah

$$\frac{10\text{Ah}}{3\text{A}} = 3.33 \text{ h} = 200 \text{ min}$$

$$3.33\text{h} * 5 \frac{\text{km}}{\text{h}} = \mathbf{16.6 \text{ km}}$$

$$v_{\text{max}} = 45 \frac{\text{km}}{\text{h}}, I = 55\text{A}$$

kapacitet baterija: 10 Ah

$$\frac{10\text{Ah}}{55\text{A}} = \frac{2}{11} \text{ h} = 0.1818 \text{ h} = 10.9 \text{ min}$$

$$0.1818\text{h} * 45 \frac{\text{km}}{\text{h}} = \mathbf{8.2 \text{ km}}$$

Uzevši u obzir odrađena mjerenja, s najmanjom brzinom možemo prevaliti dva puta veći put nego onaj s najvećom brzinom. To je, dakle, velika prednost male brzine u odnosu na veliku, ali pogledamo li minimalnu brzinu od  $5\text{km/h}$ , što je brzina normalnog hoda, i brzinu od  $45\text{km/h}$ , jasno je da je potrebno pronaći optimalnu brzinu pri kojoj potrošnja struje ne bi bila prevelika te bi utoliko i autonomija bila nešto veća od  $10\text{km}$ . Iz prikazanog dijagrama može se zaključiti da je brzina od  $25\text{km/h}$  najekonomičnija sukladno s brzinom vožnje i autonomijom. Sada bismo lako mogli izračunati koliko bismo dugo mogli voziti te koliki bi bio domet romobila s jednim punjenjem baterija kada bi konstantno vozili navedenom brzinom (ukoliko ne računamo ubrzanja romobila koja troše veću struju):

$$v = 25 \frac{\text{km}}{\text{h}}, I = 22\text{A}$$

kapacitet baterija: 10Ah

$$\frac{10\text{Ah}}{22\text{A}} = \frac{5}{11} \text{ h} = 0.4545\text{h} = 28\text{min}$$

$$0.4545\text{h} * 25 = \mathbf{11.4\text{km}}$$

Iz gornjeg izračuna je jasno vidljivo da za 28 minuta vožnje možemo prijeći put od  $11.4\text{km}$  pri brzini od  $25\text{km/h}$  što je za kretanje po gradu na kraćim relacijama sasvim dovoljna brzina.

## 11.2. Troškovi vožnje električnim vozilom

Dolazimo do vrlo zanimljivog dijela – cijena električne energije koja se utroši za vožnju romobilom. Trenutačne vrijednosti struje koju HEP ispostavlja potrošačima su slijedeće:

**Cijena za višu tarifu:** 1.14kn za kWh električne energije

**Cijena za nižu tarifu:** 0.56kn za kWh električne energije

U proračunu će biti uvrštene cijene punjenja baterija u noćnoj tarifi koja je duplo jeftinija od one dnevne tarife.

Kilovat-sat je količina energije koju troši uređaj od jednog kilovata ( $1\text{kW}$ ) u vremenu od jednog sata. 1 kilovat-sat energije odgovara iznosu od 3.6 mega džula:  $1\text{kWh} = 3.6\text{MJ}$ .

Utrošak energije jednak je umnošku snage i vremena:

$$[\text{kWh}] = P * t = U * I * t = [\text{V}][\text{A}][\text{t}]$$

Iz prethodne jednadžbe vidljivo je kako je potrošena električna energija jednaka umnošku napona, struje i vremena. Ako je na primjer uređaj spojen na napon od  $10V$  i vuče  $10A$  struje punih 10 sati, to će iznositi  $1kWh$  potrošene energije.

Kada bi pretpostavili da je napon baterija električnog romobila konstantan ( $25 V$ ) i da je brzina konstantna ( $20km/h$ ) pri kojoj motor troši struju od  $15A$ , lako se može izračunati koliko se dugo možemo voziti za  $1kWh$  električne energije bez punjenja baterija:

### Izračun utroška energije pri vožnji električnog romobila:

#### Poznato:

- Napon baterije:  $32V$
- Brzina vožnje:  $25km/h$
- Struja motora:  $22A$
- $1kWh$  energije (noćni režim):  $0.56kn$

Prijedeni put za 20 minuta vožnje s maksimalnom brzinom ( $25km/h$ ):

$$s = v * t = 25 \frac{km}{h} * \frac{1}{3} h = \mathbf{8.33km}$$

Električna energija od  $1kWh$  se stoga potroši u sljedećem vremenu:

$$\text{Snaga koja se troši na elektromotru: } P_m = U * I = 32V * 22A = \mathbf{704W}$$

$$P * t = 1000Wh = 0.56kn \rightarrow t_m = \frac{P}{P_m} = \frac{1000W}{704W} = 1.42h = \mathbf{85min}$$

Dakle, električna energija od jednog kilovat sata se vožnjom uz maksimalnu brzinu potroši za 85min.

$$\text{Pri tome je prijedeni put jednak: } s = v * t = 25 \frac{km}{h} * 1.42h = \mathbf{35.5km}$$

Time dolazimo do zaključka da  $1km$  vožnje električnim romobilom košta:

$$\mathbf{1km: } \frac{0.56kn}{35.5km} = \mathbf{0.01577kn}$$

$$\text{Zaključujemo da } \mathbf{100km} \text{ košta: } 0.01577kn * 100 = \mathbf{1.58kn}$$

Poželite li takvim vozilom otputovati do Dubrovnika, troškovi puta iz Zagreba ( $600km$ ), ne računajući cestarine, bili bi  $9.46kn$ ; naravno uz pretpostavku kada bi imali baterije dovoljno velikog kapaciteta za takvo putovanje.

## 11.3. Troškovi izrade električnog vozila

Okvirne cijene dijelova potrebnih za izradu električnog romobila su sljedeće:

Metalna konstrukcija:	400kn
Ležajevi, kočnice, i amortizeri:	500kn
Motor i kontroler za motor:	1500kn
Baterije i ostala elektronika:	800kn
Piljenje, tokarenje, zavarivanje	200kn
Brušenje, bojanje	400kn
<b>UKUPNO:</b>	<b>3800kn</b>

## 12. Daljnji razvoj

Razvoj malih električnih vozila nije stao na ovom romobilu. Preko ljeta, za vrijeme studentske prakse, počela je izrada potpuno drugačijih, malih romobila, naravno na električni pogon.

### 12.1. Specifikacije malog električnog romobila (mES)

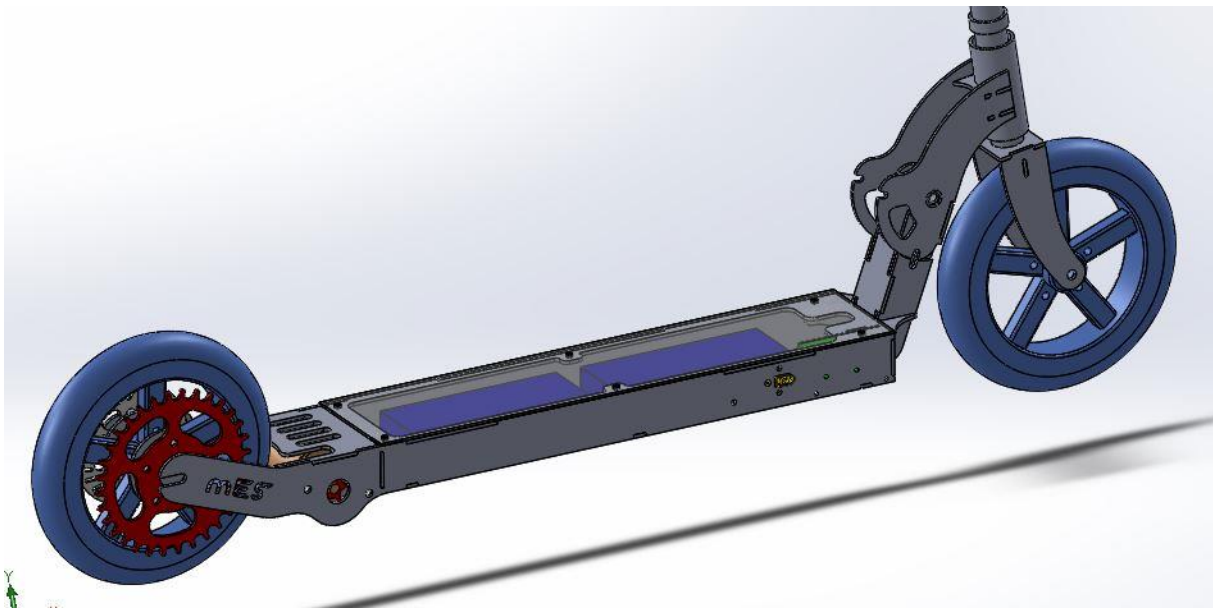
Mali električni romobil (mES – mini electric scooter) namijenjen je za gradsku vožnju na kraćim relacijama. Mehanička konstrukcija potpuno je drugačija od one na velikom romobilu a sastoji se od laserski izrezanih čeličnih ploča debljine 3mm. Ovaj romobil neusporedivo je lakši od njegovog prethodnika te sa svom elektronikom, baterijama, kotačima i motorom teži nešto više od 6kg. Elektronika je vrlo jednostavna, čip koji upravlja procesima u romobilu je AtTiny45 i ima samo 8 nožica, a njegova zadaća je da uspostavi vezu između ručice za gas i brzine motora, mjerenje napona baterije te simbolični prikaz napunjenosti baterije pomoću dvije LED diode. Mana ovog romobila je to što su kotači tvrdi, silikonski, i što je vožnja manje udobna od one na velikom romobilu. Ali zato je ovaj romobil vrlo jednostavno preklopiv, te zbog svoje male težine moguće ga je bez problema staviti u auto ili neki mali prostor. U situacijama kada želimo na primjer otići do nekog trgovačkog centra sa ovim vozilom, moguće ga je preklopiti, unijeti unutra, te ga vući sa sobom za vrijeme shoppinga.

### 12.2. Usporedba malog i velikog električnog romobila

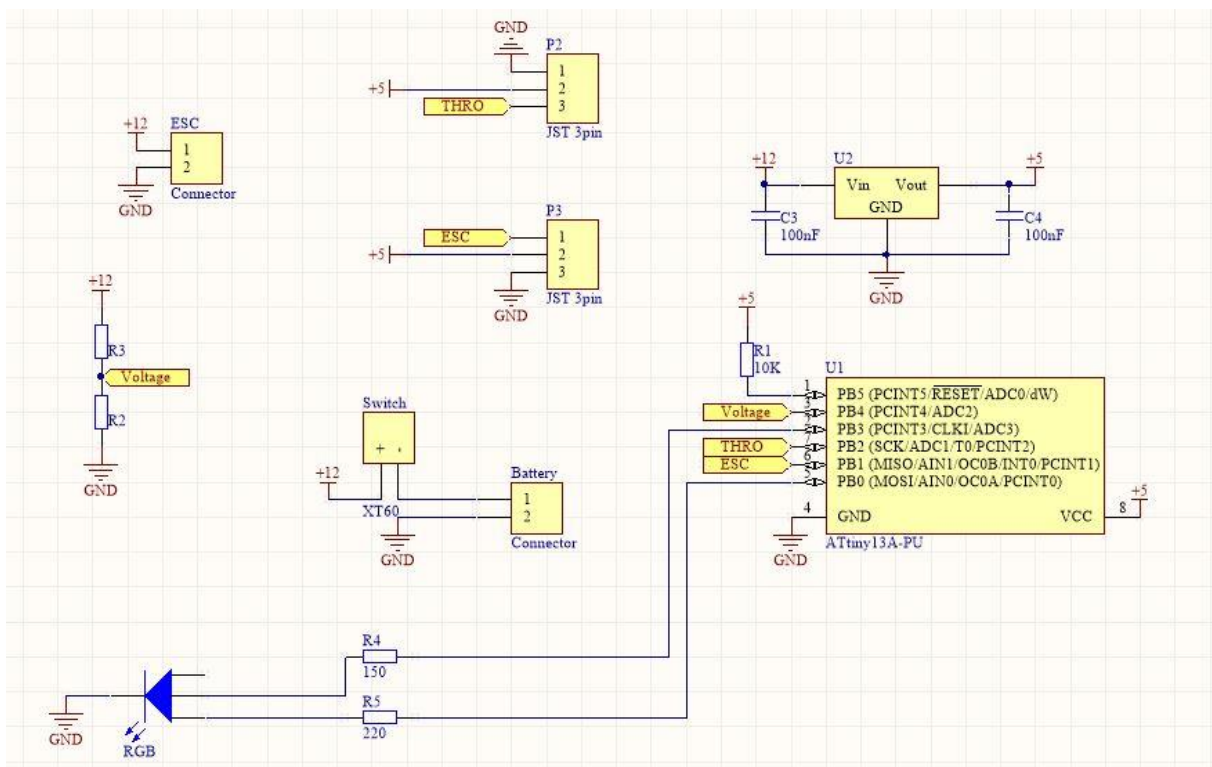
- |                         |         |                         |         |
|-------------------------|---------|-------------------------|---------|
| • Maksimalna brzina:    | 20km/h  | • Maksimalna brzina:    | 45km/h  |
| • Autonomija:           | 8km     | • Autonomija:           | 15km    |
| • Napon baterije:       | 12V     | • Napon baterije:       | 30V     |
| • Najveća snaga motora: | 1kW     | • Najveća snaga motora: | 2.1kW   |
| • Težina vozila:        | 6.8kg   | • Težina vozila:        | 28kg    |
| • Suspenzije:           | Nema    | • Suspenzije:           | Ima     |
| • Veličina kotaca:      | D=205mm | • Veličina kotaca:      | D=260mm |



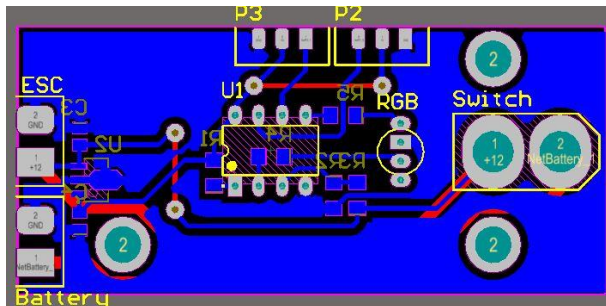
### 12.3. Fotografije mES-a



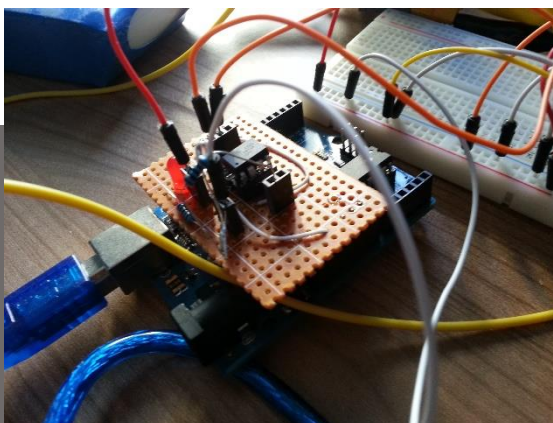
Slika 54. SolidWorks model malog romobila



Slika 55. Električna shema pločice



Slika 56. PCB pločica



Slika 57. Programiranje mikrokontrolera



Slika 58. Završena mehanička konstrukcija



## 13. Zaključak

Električni romobil moguće je voziti svakodnevno i s njime izbjeći gužvu na gradskim ulicama. Vožnja je predviđena po biciklističkoj stazi tamo gdje je ima ili po nogostupu tamo gdje je nema zbog još uvijek nedovoljne snage i brzine romobila kako bi se kretao po prometnicama.



*Slika 59. Električni romobil*

Zakon u današnje vrijeme ne ide u prilog ubrzanom razvoju tehnologije i načina transporta te je bez potrebe registriranja vozila nužno da vozilo ne prelazi brzinu od 25km/h te da snaga motora ne premašuje 300W. U „economy“ režimu vožnje električnog romobila „unutar“ zakona smo te ne kršimo niti jedno pravilo. Kada bi ovakvo vozilo ušlo u serijsku proizvodnju, moguće bi bilo postaviti i slabiji motor ili jednostavno elektronički onemogućiti brzinu iznad 25km/h i snagu motora iznad 300W.

Dodatne preinake ovakvom vozilu su postavljanje brava s ključem kojima bi onemogućili otvaranje pretinaca na romobilu gdje je u jednom od njih smješten pogon te bi utoliko bilo moguće zaključati i onesposobiti pokretanje romobila neovisno gdje se nalazili.

## Programski kod u mikrokontroleru

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <Servo.h>
#include <math.h>
#include <DS1302.h>
#include <EEPROM.h>

Servo esc;
int throttlePin = 0;    // Analog Pin 0
int currentPin = 1;    // Analog Pin 1
int thermistorPin = 2; // Analog Pin 2
int batteryPin = 3;    // Analog Pin 3
// DS1302: CE pin    -> Arduino Digital 10
//                I/O pin -> Arduino Digital 9
//                SCLK pin -> Arduino Digital 8

int leftKey = 5;
int leftStatus = 0;
int rightKey = 6;
int rightStatus = 0;
int keyCount = 1;
int enterKey = 7;
int enterStatus = 0;
int enterCount = 0;
int reed = 2;
int led = 11;
int leftBrake = 12;
int leftBrakeStatus = 0;
int rightBrake = 13;
int rightBrakeStatus = 0;

float maxCurrent = 0;
float maxVoltage = 0;
float maxPower = 0;
float maxSpeedkmph = 0;
int revolutions = 0;
float distance = 0;
float revo = 0;
float partialCurrentConsumption = 0;

long previousMillis = 0;    // will store last time LED was updated
long intervalDisplay = 500; // interval at which to blink
(millisecond)
long intervalKeys = 100;    // interval at which to blink
(millisecond)
long intervalMotor = 1;    // interval at which to blink
(millisecond)

long debouncing_time = 20; //Debouncing Time in Milliseconds
volatile unsigned long last_micros;

DS1302 rtc(10, 9, 8);

unsigned int rpmilli;
float speedmps;
```

```

float speedkmph;
double Thermistor(int RawADC) {
    long Resistance; double Temp;
    Resistance=((10240000/(RawADC)) - 10000); // 10k Thermistor, in lower
part
    Temp = log(Resistance);
    Temp = 1 / (0.001129148 + (0.000234125 * Temp) + (0.0000000876741 * Temp
* Temp * Temp));
    Temp = Temp - 273.15;
    return Temp;
}

volatile float time = 0;
volatile float time_last = 0;
volatile float period = 0;

LiquidCrystal_I2C lcd(0x20, 2, 1, 0, 4, 5, 6, 7,
3, POSITIVE); // Set the LCD I2C address

void setup()
{
    pinMode(leftKey, OUTPUT);
    pinMode(rightKey, OUTPUT);
    pinMode(enterKey, OUTPUT);
    pinMode(reed, OUTPUT);
    pinMode(led, OUTPUT);
    pinMode(leftBrake, INPUT);
    pinMode(rightBrake, INPUT);

    // Set the clock to run-mode, and disable the write protection
    rtc.halt(false);
    rtc.writeProtect(false);

    // The following lines can be commented out to use the values already
stored in the DS1302
    //rtc.setDOW(MONDAY); // Set Day-of-Week to FRIDAY
    //rtc.setTime(15, 24, 00); // Set the time to 12:00:00 (24hr format)
    //rtc.setDate(31, 3, 2014); // Set the date to August 6th, 2010

    esc.attach(3);
    Serial.begin(9600); // Used to type in characters
    attachInterrupt(0, debounceInterrupt, RISING);

    double temp;
    lcd.begin(20,4);
    lcd.backlight();
    lcd.setCursor(2,1); lcd.print("Electric Scooter");
    lcd.setCursor(4,2); lcd.print("Version: 1.6");
    esc.write(0);
    delay(200);
    lcd.clear();
}

void debounceInterrupt() {
    if((long)(micros() - last_micros) >= debouncing_time * 1000) {
        Interrupt();
        last_micros = micros();
    }
}

void Interrupt() {

```



```
time = (millis() - time_last);
time_last = millis();
revolutions = revolutions+1;
float currentSensor = analogRead(currentPin);
float currentSensorInverted = 1024-currentSensor;
float current = map(currentSensorInverted, 512, 870, 0, 900);
partialCurrentConsumption = ((time*1000)/3600)*(current/10);

}

void loop()
{

  lcd.home ();
  unsigned long currentMillis = millis();

  //LED BREAK*****

  leftBrakeStatus = digitalRead(leftBrake);
  rightBrakeStatus = digitalRead(rightBrake);
  if ((leftBrakeStatus == HIGH) && (rightBrakeStatus == HIGH)) {
    analogWrite(led, 45);}
  else {digitalWrite(led, HIGH);}

  //SPEED*****

  if (time > 1) {
    period = time/1000;
    speedmps = 0.78/period;
    speedkmph = speedmps*3.6; }
  if (speedkmph - maxSpeedkmph > 0) {
    maxSpeedkmph = speedkmph;}

  //TEMPERATURE*****

  int temp=Thermistor(analogRead(thermistorPin));

  //BATTERY*****

  int battery = analogRead(batteryPin);
  int voltagePercent = map(battery, 490, 630, 0, 100);
  float voltage = (battery/204.8)*11;
  if (voltage - maxVoltage > 0) {
    maxVoltage = voltage;}

  //CURRENT*****

  float currentSensor = analogRead(currentPin);
  float currentSensorInverted = 1024-currentSensor;
  float current = map(currentSensorInverted, 512, 870, 0, 900);
  if (current - maxCurrent > 0) {
    maxCurrent = current;}

  //POWER*****

  float power = (current/10)*voltage;
  if (power - maxPower > 0) {
    maxPower = power;}

  //DISTANCE*****
```

```

    if (time > 10) {
    distance = revolutions*0.78;
    float revo = distance;
    }
    float batPerCur = 10/(current/10);
    float estimatorDistance = batPerCur*speedkmph;

//THROTTLE*****

    int throttle = analogRead(throttlePin);
    int fullThrottle = analogRead(throttlePin);
    int maxThrottle = 127;

    switch (enterCount) {
        case 1:
            if (speedkmph >= 0 && speedkmph <=10) {
                maxThrottle = map(speedkmph, 0, 10, 0, 45);}
            if (speedkmph > 10 && speedkmph <=20) {
                maxThrottle = map(speedkmph, 10, 20, 40, 70);}
            if (speedkmph > 20 && speedkmph <=30) {
                maxThrottle = map(speedkmph, 20, 30, 60, 95);}
            if (speedkmph > 30 && speedkmph <=50) {
                maxThrottle = map(speedkmph, 30, 45, 80, 125);}
            break;
        case 2:
            if (speedkmph >= 0 && speedkmph <=10) {
                maxThrottle = map(speedkmph, 0, 10, 0, 30);}
            if (speedkmph > 10 && speedkmph <=20) {
                maxThrottle = map(speedkmph, 10, 20, 20, 50);}
            if (speedkmph > 20 && speedkmph <=30) {
                maxThrottle = map(speedkmph, 20, 30, 40, 70);}
            if (speedkmph > 30 && speedkmph <=50) {
                maxThrottle = map(speedkmph, 30, 45, 60, 100);}
            break;
        case 3:
            maxThrottle = maxThrottle;
            break;
        case 4:
            maxThrottle = map(speedkmph, 0, 45, 0, 125);
            break;
    }

    throttle = map(throttle, 176, 867, 0, maxThrottle); // maxThrottle = 127
    is full throttle range
    int throttlePercent = analogRead(throttlePin);
    throttlePercent = map(throttle, 0, 126, 0, 100);
    fullThrottle = map(fullThrottle, 180, 863, 0, 100); //fullThrottle =
    map(fullThrottle, 180, 863, 0, 100);

    esc.write(throttle);

//BUTTONS

    leftStatus = digitalRead(leftKey);
    rightStatus = digitalRead(rightKey);

```

```

enterStatus = digitalRead(enterKey);
if (leftStatus == HIGH) {
    if (keyCount == 4) {
        keyCount = 1;} else {
        keyCount = keyCount+1;
    }
} else if (rightStatus == HIGH) {
    if (keyCount == 1) {
        keyCount = 4;} else {
        keyCount = keyCount-1;
    }
}
if (enterStatus == HIGH) {
    if (enterCount == 4) {
        enterCount = 1;} else {
        enterCount = enterCount+1;
    }
}

//DISPLAY

switch (keyCount) {

//BOOT MENU*****

case 0:
    lcd.setCursor(0,1); lcd.print ("    Press select");
    lcd.setCursor(0,2); lcd.print ("        to unlock");
    if (enterCount > 0) {keyCount = keyCount+1;}

    break;

//DRIVING MODES*****

case 1:
    switch (enterCount) {
        case 1:
            lcd.setCursor(0,0); lcd.print ("    Driving modes    ");
            lcd.setCursor(0,1); lcd.print ("Spd:35km/h Dist:12km");
            lcd.setCursor(0,2); lcd.print ("*normal    beginner ");
            lcd.setCursor(0,3); lcd.print (" economy    max power");
            break;
        case 2:
            lcd.setCursor(0,0); lcd.print ("    Driving modes    ");
            lcd.setCursor(0,1); lcd.print ("Spd:20km/h Dist:18km");
            lcd.setCursor(0,2); lcd.print (" normal    *beginner ");
            lcd.setCursor(0,3); lcd.print (" economy    max power");
            break;
        case 3:
            lcd.setCursor(0,0); lcd.print ("    Driving modes    ");
            lcd.setCursor(0,1); lcd.print ("Spd:25km/h Dist:15km");
            lcd.setCursor(0,2); lcd.print (" normal    beginner ");
            lcd.setCursor(0,3); lcd.print ("*economy    max power");
            break;
        case 4:
            lcd.setCursor(0,0); lcd.print ("    Driving modes    ");
            lcd.setCursor(0,1); lcd.print ("Spd:43km/h Dist: 8km");
            lcd.setCursor(0,2); lcd.print (" normal    beginner ");

```

```

        lcd.setCursor(0,3); lcd.print (" economy *max power");
        break;
    }

    break;

//MAIN MENU*****

    case 2:

        if(currentMillis - previousMillis > intervalDisplay) {
            // save the last time you blinked the LED
            previousMillis = currentMillis;
            lcd.setCursor(0, 0); //Display the motor temperature
            lcd.print(temp); lcd.print ("C ");
            lcd.setCursor(6, 0); //Display time centered on the upper line
            lcd.print(rtc.getTimeStr()); lcd.print (" ");
            lcd.setCursor(16,0);
            if (voltage < 1) {lcd.print (" --");}
            else {
                if (voltagePercent > 100) {
                    lcd.print ("100%"); }
                else {
                    if (voltagePercent < 10) lcd.print (" ");
                    if (voltagePercent < 100) lcd.print (" ");
                    lcd.print (voltagePercent); lcd.print ("%"); }
            }
        }

        lcd.setCursor(0,1); //Display speed in the second line
        if (speedkmph > 1) {
            if (speedkmph < 10) lcd.print (" ");
            lcd.print (speedkmph); lcd.print ("km/h "); }
        else lcd.print (" 0.00km/h ");

        lcd.setCursor(11,1); //Display distance traveled
        if (distance < 10) lcd.print (" ");
        if (distance < 100) lcd.print (" ");
        if (distance < 1000) lcd.print (" ");
        if (distance < 10000) lcd.print (" ");
        lcd.print (distance); lcd.print ("m");

        lcd.setCursor(0,2);
        lcd.print ("Throttle: ");
        if (throttlePercent < 10) lcd.print (" ");
        if (throttlePercent < 100) lcd.print (" ");
        lcd.print (throttlePercent);
        lcd.print ("/");
        lcd.print (fullThrottle);
        lcd.print (" ");

        lcd.setCursor(0,3);
        if (current > 10) {
            lcd.print ("Power: ");
            if (power < 10) lcd.print (" ");
            if (power < 100) lcd.print (" ");
            if (power < 1000) lcd.print (" ");
            lcd.print (power); lcd.print ("W "); }
            else {lcd.print ("Power: 0.00W "); }

```

```

    }
    break;

//POWER MENU*****

case 3:
    lcd.setCursor(0,0);
    if (voltage < 1) {lcd.print ("Battery:  0.00V  --");}
    else {
    lcd.print ("Battery: "); lcd.print (voltage); lcd.print ("V ");
    lcd.setCursor(16,0);
    if (voltagePercent > 100) {
        lcd.print ("100%"); }
    else {
        if (voltagePercent < 10) lcd.print (" ");
        if (voltagePercent < 100) lcd.print (" ");
        lcd.print (voltagePercent); lcd.print ("%"); }
    }
    lcd.setCursor(0,1);
    if (current > 5) {
    lcd.print ("Current:");
        if (current < 100) lcd.print (" ");
        if (current < 1000) lcd.print (" ");
        lcd.print (current/10); lcd.print ("A      "); }
    else {lcd.print ("Current:  0.00A      "); }

    lcd.setCursor(0,2);
    if (current > 5) {
    lcd.print ("Power: ");
        if (power < 10) lcd.print (" ");
        if (power < 100) lcd.print (" ");
        if (power < 1000) lcd.print (" ");
        lcd.print (power); lcd.print ("W      "); }
    else {lcd.print ("Power:  0.00W      "); }

    lcd.setCursor(0,3);
    lcd.print ("Max power: ");
        if (maxPower < 10) lcd.print (" ");
        if (maxPower < 100) lcd.print (" ");
        if (maxPower < 1000) lcd.print (" ");
        lcd.print (maxPower); lcd.print ("W");
    break;

//SPEED MENU*****

case 4:
    lcd.setCursor(0,0);
    if (speedkmph > 1) {
    lcd.print ("Speed:      ");
        if (speedkmph < 10) lcd.print (" ");
    lcd.print (speedkmph); lcd.print ("km/h"); }
    else lcd.print ("Speed:  0.00km/h");

    lcd.setCursor(0,1);
    lcd.print ("Max speed: ");
        if (maxSpeedkmph < 10) lcd.print (" ");
    lcd.print (maxSpeedkmph); lcd.print ("km/h");

    lcd.setCursor(0,2);
    lcd.print ("Distance: ");
        if (distance < 10) lcd.print (" ");

```



```
        if (distance < 100) lcd.print (" ");
        if (distance < 1000) lcd.print (" ");
        if (distance < 10000) lcd.print (" ");
        lcd.print (distance); lcd.print ("m ");

        lcd.setCursor(0,3);
        if (current > 10) {
            lcd.print ("Range: ");
            if (estimatorDistance < 10) lcd.print (" ");
            lcd.print (estimatorDistance); lcd.print ("km"); }
        else {lcd.print ("Range:  --          "); }

    break;
}

}
```

## Literatura

- [1] [http://www.transportation.anl.gov/smart\\_grid/ev\\_pilot.html](http://www.transportation.anl.gov/smart_grid/ev_pilot.html), 13.4.2014
- [2] <http://acuitybusiness.com/blog/bid/123246/Five-Disciplines-to-Ensure-Project-Success>, 13.4.2014
- [3] <http://www.mqitechnology.com/in-focus.jsp>, 13.4.2014
- [4] [http://www.rcuniverse.com/magazine/article\\_display.cfm?article\\_id=1344](http://www.rcuniverse.com/magazine/article_display.cfm?article_id=1344), 13.4.2014
- [5] <http://www.johnsonelectric.com/en/resources-for-engineers/automotive-applications/motion-technology/ec-motor-brushless.html>, 13.4.2014
- [6] [http://www.hobbyking.com/hobbyking/store/\\_18182\\_Turnigy\\_Aerodrive\\_SK3\\_6374\\_168kv\\_Brushless\\_Outrunner\\_Motor.html](http://www.hobbyking.com/hobbyking/store/_18182_Turnigy_Aerodrive_SK3_6374_168kv_Brushless_Outrunner_Motor.html), 13.4.2014
- [7] <http://www.hobbyking.com/hobbyking/store/catalog/18182.jpg>, 13.4.2014
- [8] [http://www.hobbyking.com/hobbyking/store/\\_40277\\_Hobbyking\\_YEP\\_120A\\_HV\\_4\\_14S\\_Marine\\_Brushless\\_Speed\\_Controller\\_Opto\\_.html](http://www.hobbyking.com/hobbyking/store/_40277_Hobbyking_YEP_120A_HV_4_14S_Marine_Brushless_Speed_Controller_Opto_.html), 13.4.2014
- [9] <http://www.inmotion.pt/store/ftdi-basic-breakout-5v>, 13.4.2014
- [10] [http://www.hobbyking.com/hobbyking/store/\\_15521\\_Turnigy\\_5000mAh\\_4S1P\\_14\\_8v\\_20C\\_hardcase\\_pack.html](http://www.hobbyking.com/hobbyking/store/_15521_Turnigy_5000mAh_4S1P_14_8v_20C_hardcase_pack.html), 13.4.2014
- [11] <http://www.jameco.com/Jameco/workshop/JamecoBuilds/arduinocircuit.html>, 13.4.2014
- [12] [http://www.artas.hr/vozila/elektricni\\_bicikli\\_dijelovi.htm](http://www.artas.hr/vozila/elektricni_bicikli_dijelovi.htm), 13.4.2014
- [13] [http://www.allaboutcircuits.com/vol\\_4/chpt\\_3/4.html](http://www.allaboutcircuits.com/vol_4/chpt_3/4.html), 13.4.2014
- [14] <https://learn.sparkfun.com/tutorials/pull-up-resistors/what-is-a-pull-up-resistor>, 13.4.2014
- [15] [http://cnmat.berkeley.edu/user/andy\\_schmeder/blog/2008/08/08/optimizing\\_voltage\\_divider\\_circuit\\_resistive\\_analog\\_sensing](http://cnmat.berkeley.edu/user/andy_schmeder/blog/2008/08/08/optimizing_voltage_divider_circuit_resistive_analog_sensing), 13.4.2014
- [16] <http://www.digikey.com/product-detail/en/ACS758LCB-100B-PFF-T/620-1321-ND/2042746>, 13.4.2014
- [17] <http://wiki.pickeringtest.net/Choose+a+Reliable+Reed+Relay+Construction>, 13.4.2014