

Projektiranje sustava upravljanja dinamikom formula student vozila i eksperimentalna provjera na umanjenoj maketi vozila

Pečanić, Matko

Master's thesis / Diplomski rad

2019

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Zagreb, Faculty of Mechanical Engineering and Naval Architecture / Sveučilište u Zagrebu, Fakultet strojarstva i brodogradnje**

Permanent link / Trajna poveznica: <https://urn.nsk.hr/urn:nbn:hr:235:629220>

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

Download date / Datum preuzimanja: **2024-12-19**

Repository / Repozitorij:

[Repository of Faculty of Mechanical Engineering and Naval Architecture University of Zagreb](#)



SVEUČILIŠTE U ZAGREBU
FAKULTET STROJARSTVA I BRODOGRADNJE

DIPLOMSKI RAD

Matko Pečanić

Zagreb, 2019.

SVEUČILIŠTE U ZAGREBU
FAKULTET STROJARSTVA I BRODOGRADNJE

DIPLOMSKI RAD

Mentori:

Prof.dr. sc. Danijel Pavković, dipl. ing.

Student:

Matko Pečanić

Zagreb, 2019.

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

Zahvaljujem se mentoru Danijelu Pavkoviću na podršci, FSB Racing Team-u na financijskim sredstvima i pruženoj prilici za razvijanje i testiranje diplomskog rada. Zahvaljujem se i kolegama Kruni Hrvatinić i Annie Kovač iz Rimac Automobili d.o.o. na korisnim savjetima.

Zahvaljujem se roditeljima na podršci tokom mog studiranja i ulaganju u moje znanje. Na kraju najveća hvala supruzi Martini i kćeri Loreni na podršci i strpljenju.

Matko Pečanić



SVEUČILIŠTE U ZAGREBU
FAKULTET STROJARSTVA I BRODOGRADNJE



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

Sveučilište u Zagrebu Fakultet strojarstva i brodogradnje	
Datum: 09. 05. 2019	Prilog:
Klasa: 602-04/19-G/3	
Ur. broj: 15-1703-19-207	

DIPLOMSKI ZADATAK

Student: **MATKO PEČANIĆ** Mat. br.: 0035194680

Naslov rada na hrvatskom jeziku: **Projektiranje sustava upravljanja dinamikom formula student vozila eksperimentalna provjera na umanjenoj maketi vozila**

Naslov rada na engleskom jeziku: **Design of a formula student vehicle dynamics control system and experimental verification on a down-scaled vehicle model**

Opis zadatka:

Sustavi regulacije dinamike vozila kao što su sustavi regulacije vučne sile (engl. Traction Control Systems) i sustavi za koordinirano upravljanje okretnim momentom na pogonskim kotačima (engl. Torque Vectoring Systems) ključni su za sigurnost vožnje vozila u uvjetima smanjene adhezije između automobilske gume i podloge te za performanse vozila u trkaćim uvjetima. Stoga se kontinuirano radi na unaprjeđenju ovih sustava. U radu je potrebno napraviti sljedeće:

1. Dati kratki pregled sustava za upravljanje dinamikom cestovnih vozila i formulirati matematički model vozila koji uključuje uzdužnu i poprečnu dinamiku gibanja vozila, dinamiku rotacije oko centra mase vozila i model adhezije između automobilske gume i podloge.
2. Po potrebi pojednostavniti predloženi model vozila za karakteristične scenarije (uzdužno i poprečno gibanje te rotacija oko centra mase) te na temelju pojednostavljenih modela treba predložiti odgovarajuće algoritme upravljanja (regulatore) dinamičkih veličina vozila u gibanju.
3. Na temelju potpunog matematičkog modela treba implementirati simulacijski model dinamike gibanja vozila pogodan za ispitivanje upravljačkih algoritama, a koje treba ispitati i vrednovati za odabrane tipične situacije u vožnji.
4. Izraditi umanjeni model cestovnog vozila i opisati njegove karakteristične dijelove i sklopovske komponente. Primjenom razvijenog umanjenog modela vozila potrebno je eksperimentalno ispitati pojedine sustave regulacije dinamike vozila.

U radu je također potrebno navesti korištenu literaturu i eventualno dobivenu pomoć.

Zadatak zadan:
07. ožujka 2019.

Rok predaje rada:
09. svibnja 2019.

Predviđeni datum obrane:
15. svibnja 2019.
16. svibnja 2019.
17. svibnja 2019.

Zadatak zadao: *Danijel Pavković*
prof. dr. sc. Danijel Pavković

Predsjednica Povjerenstva:
Biserka Runje
prof. dr. sc. Biserka Runje

SADRŽAJ

SADRŽAJ	I
POPIS SLIKA	II
POPIS OZNAKA	IV
SAŽETAK.....	V
SUMMARY	VI
1. UVOD.....	1
1.1. TCS - Traction Control System	1
1.2. TV – Torque Vectoing	1
2. O VOZILU I PRIMJENI RAZVIJENIH SUSTAVA	2
2.1. Primjena razvijenih sustava	2
2.1.1. O FSB Racing Team-u.....	2
2.1.2. O vozilu.....	2
3. TCS – Traction Control System	4
3.1. Model sustava	4
3.1.1. Osnovna dinamika	4
3.1.2. Prijenos težine.....	5
3.1.3. Električni motor	6
3.1.4. Model gume	7
3.2. Projektiranje regulatora na razini pogonskih kotača.....	10
3.2.1. Regulator struje/okretnog momenta.....	10
3.2.2. Regulator brzine vrtnje	14
4. Torque Vectoring.....	17
4.1. Model sustava	17
4.2. Projektiranje regulatora.....	25
5. Rezultati simulacija	27
6. Izrada makete.....	31
6.1. Kupovne komponente	31
6.2. Izrađene komponente	36
6.3. Električna shema	39
6.4. Programiranje.....	41
7. Rezultati eksperimentalnih mjerenja	47
8. ZAKLJUČAK.....	52
LITERATURA.....	53
PROGRAMSKI KOD	54

POPIS SLIKA

Slika 1.	Električna formula	2
Slika 2.	Model longitudinalne dinamike vozila.....	4
Slika 3.	Model osnovne longitudinalne dinamike vozila	5
Slika 4.	Model prijenosa težine	6
Slika 5.	Model elektromotora	7
Slika 6.	Model gume (F_z =konst., karakteristika podloge=konst.)	8
Slika 7.	Prošireni model gume.....	8
Slika 8.	$g(v_x, F_z)$	9
Slika 9.	Ovisnost sile trenja o postotku klizanja gume za razne vertikalne sile na kotač.....	9
Slika 10.	Ovisnost longitudinalnih i lateralnih sila trenja o postotku klizanja	10
Slika 11.	Blokovski dijagram podređenog regulacijskog kruga struje armature s PI regulatorom i kompenzatorom utjecaja elektromotorne sile	11
Slika 12.	Blokovski dijagram regulacijskog kruga struje na kojem se zasniva sinteza PI regulatora struje	12
Slika 13.	Odziv regulacijskog kruga struje na skokovitu pobudu	13
Slika 14.	Blokovski dijagram regulacijskog kruga brzine vrtnje s P regulatorom	14
Slika 15.	Blokovski dijagram regulacijskog kruga brzine vrtnje s podređenim regulatorom struje	15
Slika 16.	Regulator broja okretaja s prefiltrom	16
Slika 17.	Odziv regulacijskog kruga broja okretaja na skokovitu pobudu i poremećaj (moment tereta)	17
Slika 18.	Prikaz vozila sa označenim veličinama	18
Slika 19.	Pacejka model gume.....	19
Slika 20.	Karakteristike gume	20
Slika 21.	Ilustracija kuteva skretanja - Ackermanovo skretanje.....	20
Slika 22.	Upravljački kut skretanja.....	21
Slika 23.	Trajektorija pasivnog ponašanja vozila	23
Slika 24.	Horizontalne sile svakog kotača.....	24
Slika 25.	Kut skretanja β i brzina zakreta vozila	24
Slika 26.	Kut zakreta prednjih kotača.....	25
Slika 27.	Regulator brzine zakreta vozila	25
Slika 28.	Pojednostavljeni sustav s regulatorom	25
Slika 29.	Ubrzanje sa i bez TCS-a.....	27
Slika 30.	Robusnost pri nailasku na lošiju podlogu.....	28
Slika 31.	Usporedba trajektorije vozila sa i bez TV-a.....	29
Slika 32.	Kut zakreta kotača	29
Slika 33.	Brzine zakreta i aktivni M_z	30
Slika 34.	Horizontalne sile svakog kotača.....	30
Slika 35.	CAD model makete i izrađena maketa	31
Slika 36.	TI Launchpad	32
Slika 37.	Arduino Nano	32
Slika 38.	DC motor s enkoderom	33
Slika 39.	L298N PWM Modul	33
Slika 40.	Senzor struje	33
Slika 41.	Optički enkoder	34
Slika 42.	Li-ion ćelija Sony VTC6	34
Slika 43.	Battery Management System.....	35
Slika 44.	RC servo motor	35

Slika 45.	LM2595	35
Slika 46.	NRF24L01	36
Slika 47.	MPU-6050	36
Slika 48.	CAD podvozja	36
Slika 49.	Podvozje	37
Slika 50.	CAD prednjeg ovjesa (sakriven lijevi kotač)	37
Slika 51.	Zakreti kotača - Ackermanovo skretanje	38
Slika 52.	Baterija	38
Slika 53.	Stari i nove verzije diska enkodera	38
Slika 54.	Očitanje pomoću osciloskopa - Usporedba signala starog (lijevo) i novog (desno) enkodera pri vrtnji	39
Slika 55.	Električna shema	40
Slika 56.	Maketa i daljinski upravljač	41
Slika 57.	Značajan šum senzora struje (0-100% struje)	42
Slika 58.	Struktura PI regulatora TCS-a na mikrokontroleru	43
Slika 59.	Odziv TCS-a na maketi	44
Slika 60.	Zahtijevani moment kao poremećaj	45
Slika 61.	Feedforward programski kod	46
Slika 62.	Feedforward mjerenje	46
Slika 63.	Ubrzanje bez TCS-a	47
Slika 64.	Ubrzanje sa TCS-om	48
Slika 65.	Prolazak makete kroz zavoj TV=off	49
Slika 66.	Snimka prolaza zavoja TV=off	50
Slika 67.	Prolazak makete kroz zavoj TV=on	50
Slika 68.	Snimka prolaza zavoja TV=on	51

POPIS TABLICA

Tablica 1. Specifikacije bolida	3
---------------------------------------	---

POPIS OZNAKA

Oznaka	Jedinica	Opis
a	m/s^2	akceleracija
b	m	udaljenost od težišta do prednje osovine vozila
c	m	udaljenost od težišta do stražnje osovine vozila
F	N	sila
g	m/s^2	ubrzanje sile teže
h_g	m	visina težišta
I	A	jakost struje
I_{zz}	$kg\ m^2$	inercija vozila oko Z-osi
i		prijenosni omjer reduktora
J	$kg\ m^2$	inercija
K		pojačanje
K_e	V/(rad/s)	konstanta elektromotorne sile
K_m	Nm/A	momentna konstanta elektromotora
L	H	induktivitet
l	m	međuosovinski razmak
M	Nm	moment
m	kg	masa vozila
m_L	Nm	moment tereta
R	m	radijus skretanja
R_a	Ω	otpor armature motora
r	m	radijus kotača
s_x, s_y	m	put
T	s	vremenska konstanta
t	m	trag kotača
U	V	napon
v	m/s	brzina
α	deg	kut klizanja kotača
β	deg	kut između usmjerenja vozila i vektora brzine vozila
β_i	deg	kut između usmjerenja vozila u poziciji kotača i vektora brzine kotača
δ	deg	kut zakreta kotača
ε	s	ekvivalentna vremenska konstanta
κ		longitudinalno klizanje gume
ψ	deg	zakret vozila u globalnom koordinatnom sustavu
ω	rad/s	kutna brzina vrtnje

SAŽETAK

Vozila pogonjena električnim motorima kod kojih svaki motor pogoni zasebni kotač dobra su platforma za razvoj sustava kontrole stabilnosti vozila zbog brzog odziva motora, lake upravljivosti i velike koristi takvih sustava. Cilj ovog rada je razviti i testirati takve sustave za trkaći bolid formule student. Jedan od sustava ima zadaću spriječiti suvišno proklizavanje pogonskih kotača preciznim i pravovremenim smanjivanjem momenta . Drugi sustav ima zadaću stabilizirati brzinu zakretanja vozila ostvarivanjem različitih momenta na lijevim i desnim pogonskim kotačima. Performanse ovih sustava testirane su u simulacijama za koje su razvijeni matematički modeli. Testovi su također provedeni i na umanjenoj maketi vozila. U oba slučaja ostvarena je zadaća sustava i uočeno znatno poboljšanje dinamike vozila.

Ključne riječi: formula student, kontrola dinamike vozila, kontrola vuče, kontrola bočne dinamike, umanjena maketa, sustav regulacije vuče, sustav stabilizacije vozila pomoću kontrolirane raspodjele momenata

SUMMARY

Vehicles powered by electric motors where each motor drives a separate wheel are a good platform for development of vehicle stability control systems due to the quick motor response, easy control and the great benefits of such systems. The aim of this paper is to develop and test such systems for the formula student racing car. One of the systems has the task of preventing excessive slip of the drive wheels by precise reduction in torque at right time. The second system has the task of stabilizing the speed of rotation of the vehicle by achieving different torque on the left and right driven wheels. The performance of these systems has been tested in simulations for which mathematical models have been developed. Tests were also performed on the scaled-down model of the vehicle. In both cases, the systems successfully accomplished the tasks and a significant improvement in the vehicle dynamics was observed.

Key words: formula student, vehicle dynamics control, traction control, lateral dynamics control, scaled-down version, Traction Control System, Torque Vectoring

1. UVOD

Pasivno ponašanje vozila u stacionarnim režimima vožnje je stabilno i prihvatljivo, no u ekstremnim uvjetima, kao što su utrke ili kod velikih snaga pogonskih sustava, ponašanje vozila postaje nelinearno tj. nestabilnije i nepredvidivije za vozača, a također i same promjene događaju se brže nego ih čovjek može pratiti i kontrolirati. Stoga je korisno projektirati sustave kontrole vozila koji će reagirati brže, preciznije i ponovljivije. Takvi sustavi mogu imati autoritet nad komandama kojima vozač nema pristupa i stabilizirati vozilo bolje od vozača kako bi se on mogao usredotočiti na svoju ulogu. Sustavi koji će se projektirati u ovom radu su sustav protiv proklizavanja kotača pri ubrzanju tj. kontrola vuče, u daljnjem tekstu TCS (*eng. traction control system*) i sustav stabilizacije zakreta vozila pomoću kontrolirane raspodjele momenata na pogonske kotače, u daljnjem tekstu TV (*eng. Torque Vectoring*).

1.1. TCS - Traction Control System

Ovaj sustav brine se o sprječavanju proklizavanja pogonskih kotača pri velikim momentima na kotačima. Točnije, automobilska guma uvijek malo proklizava u odnosu na podlogu pri primijenjenom momentu, ono što je nepoželjno tj. što uzrokuje bočnu nestabilnost vozila i lošiju akceleraciju je prekomjerno klizanje. Stoga je zadaća sustava držati klizanje u optimalnom području kako bi se ostvarila maksimalna sila tj. akceleracija vozila i zadržala bočna stabilnost vozila.

1.2. TV – Torque Vectoring

Ovaj sustav nadzire brzinu zakretanja vozila i uspoređuje ju sa očekivanom/predviđenom. Pri pojavi odstupanja tih dviju veličina sustav određuje potrebni moment oko vertikalne osi vozila koji će korigirati odstupanje i vozilu omogućiti predviđenu tj. očekivanu brzinu zakretanja. Moment oko vertikalne osi vozila ostvaruje se zadavanjem različitih pogonskih momenata na lijevim i desnim kotačima vozila.

2. O VOZILU I PRIMJENI RAZVIJENIH SUSTAVA

2.1. Primjena razvijenih sustava

Sustavi razvijeni i ispitani u sklopu ovog rada bit će implementirani u pravo trkaće vozilo. Radi se o formula student vozilu kojeg razvija studentska udruga HSA-SF tj. FSB Racing Team na Fakultetu strojarstva i brodogradnje. Vozilo će se utrkiivati na brojim svjetskim Formula Student natjecanjima. Svrha tih natjecanja je osmisliti i izraditi trkaći jednosjed čija će se tehnička rješenja braniti pred sucima iz vrha autoindustrije. Performanse vozila bit će bodovane u raznim dinamičkim disciplinama poput testa ubrzanja, bočnog držanja, najbržeg kruga, i utrke izdržljivosti.

2.1.1. O FSB Racing Team-u

FSB Racing Team osnovan je 2004. godine i okuplja studente Sveučilišta u Zagrebu, primarno studente Fakulteta strojarstva i brodogradnje i Fakulteta elektrotehnike i računarstva ali i mnoge druge budući da je domena udruge također i ekonomija, marketing i novinarstvo.

2.1.2. O vozilu

Do sada je izrađeno pet bolida, a u izradi je novi bolid po prvi puta sa električnim pogonom za kojeg se razvijaju sustavi obrađeni u ovom radu. Po prvi puta se također koristi kompozitno podvozje od ugljičnih vlakana.



Slika 1. Električna formula

Tablica 1. Specifikacije bolida

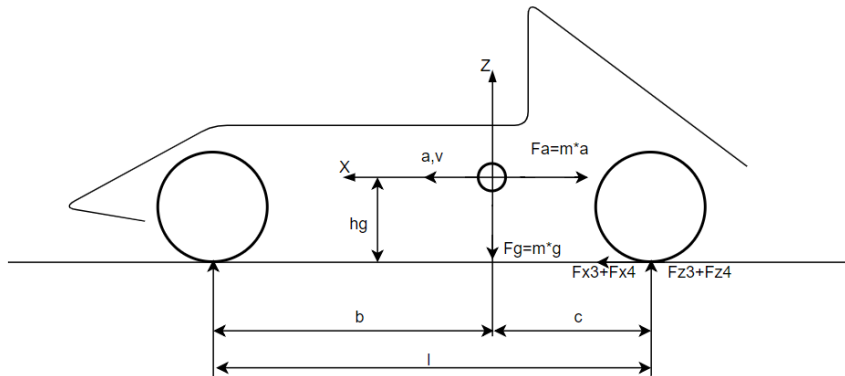
Snaga	80 kW
Masa	215 kg
Moment na kotačima	2x 580Nm
Šasija	Monocoque – ugljična vlakna
Motori	2x zasebno upravljani sinkroni elektromotori sa permanentnim magnetima <ul style="list-style-type: none">- 7 kg- 40 kW- 14 000 rpm- 45 Nm- Vodeno hlađenje
Baterija	7.5 kWh 45 kg Sony VTC6 ćelija Hlađenje isparavanjem polimera
Reduktor	Jednibrzinski dvostepeni

3. TCS – Traction Control System

3.1. Model sustava

3.1.1. Osnovna dinamika

Za razvoj TCS-a dovoljan nam je model longitudinalne dinamike vozila.



Slika 2. Model longitudinalne dinamike vozila.

Na slici su prikazane sile koje djeluju na vozilo i dimenzije potrebne za osnovni proračun. Ovaj model predstavlja koncentriranu masu na definiranoj udaljenosti između prednje i stražnje osovine, te na definiranoj visini od tla. Pogon se nalazi samo na stražnjim kotačima.

Težina formule je:

$$F_g = m \cdot g \quad [N] \quad (1)$$

F_g – težina formule [N]

m – masa formule [kg]

g – ubrzanje sile teže 9.81 [m/s²]

Akceleracija formule proporcionalna je rezultantnoj sili koja na nju djeluje i obrnuto je proporcionalna masi:

$$a_x = \frac{F_{rx}}{m} \quad \left[\frac{m}{s^2} \right] \quad (2)$$

a_x – ubrzanje vozila u x-osi [m/s²]

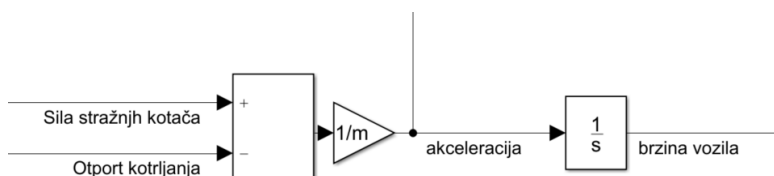
F_{rx} – suma svih sile na vozilo u x-osi

Brzina formule je integral akceleracije u vremenu:

$$v_x = \int a_x dt \left[\frac{m}{s} \right] \quad (3)$$

Prijeđeni put formule je integral brzine u vremenu:

$$s_x = \int v_x dt [m] \quad (4)$$



Slika 3. Model osnovne longitudinalne dinamike vozila

3.1.2. Prijenos težine

Na svaku masu koja ubrzava djeluje prividna d'Alambertova sila u suprotnom smjeru od akceleracije sa hvatištem u težištu tijela [1]:

$$F_{ax} = m \cdot a_x [N] \quad (5)$$

F_a – d'Alambert-ova sila [N]

Tako se uslijed akceleracije, težina bolida pomiče ka stražnjoj osovini. Kako bi opisali ovu pojavu i izračunali vertikalne sile na pogonskim kotačima primijenit ćemo metodu sume momenata oko neke točke. Za izračun vertikalne sile stražnjih kotača računamo sumu momenata oko prednje kontaktne točke:

$$\sum M_p = 0 \quad (6)$$

$$mgb + a_x m h_g - (F_{z3} + F_{z4}) \cdot l = 0 \quad (7)$$

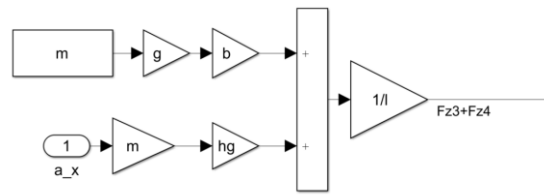
$$F_{z3} + F_{z4} = \frac{mgb + a_x m h_g}{l} \quad (8)$$

$F_{z3}+F_{z4}$ – vertikalna sila stražnjih kotača [N]

l – međuosovinski razmak [m]

h_g – visina težišta formule [m]

b, c – udaljenost od težišta do prednje tj. stražnje osovine [m]



Slika 4. Model prijenosa težine

3.1.3. Električni motor

Kod sinkronog motora s permanentnim magnetima moment koji motor razvija jednak je struji koja protječe kroz faze motora pomnoženoj sa momentnom konstantom motora „ K_m “ [2]:

$$M = I_{faz} * K_m \text{ [Nm]} \quad (9)$$

M – moment motora [Nm]

I_{faz} – fazna struja motora [A]

K_m – momentna konstanta elektromotora [Nm/A]

Inducirani (generirani) napon u motoru uslijed vrtnje je:

$$EMS = \omega * K_e \text{ [V]} \quad (10)$$

EMS – generirani napon [V]

K_e – konstanta elektromotorne sile [V/(rad/s)]

ω – kutna brzina motora [rad/s]

Struja koja protječe motorom uzrokovana je razlikom između napona narinutog na faze motora i generiranog napona:

$$I_{faz} = \frac{U - EMS}{R_a} \text{ [A]} \quad (11)$$

I_{faz} – fazna struja [A]

U – napon na motoru [V]

R_a – otpor armature [Ω]

Izraz za moment motora tako glasi:

$$M = \frac{U - \omega * K_e}{R_a} * K_m \text{ [Nm]} \quad (12)$$

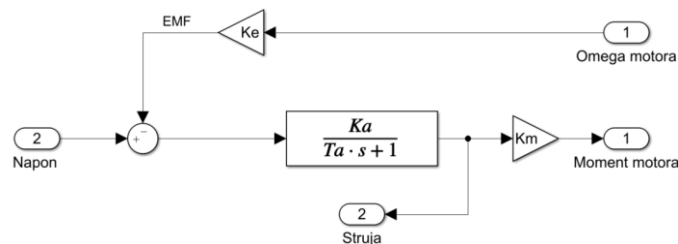
Induktivitet elektromotora definira dinamiku prijelazne pojave struje, a ta prijelazna pojava može se opisati P1 članom gdje je vremenska konstanta:

$$T_a = \frac{L_a}{R_a} \text{ [s]} \quad (13)$$

T_a – vremenska konstanta armature [s]

L_a – Induktivitet armature [H]

Izraz $1/R_a$ zamijenit ćemo sa K_a [A/V] te je konačni model elektromotora:



Slika 5. Model elektromotora

3.1.4. Model gume

Za potrebe projektiranja TCS-a smatrat ćemo da ne postoji bočno klizanje i bočne sile budući da se fokusiramo na longitudinalnu dinamiku. Glavni ulazni podatak u model gume za izračun sile je uzdužno klizanje κ :

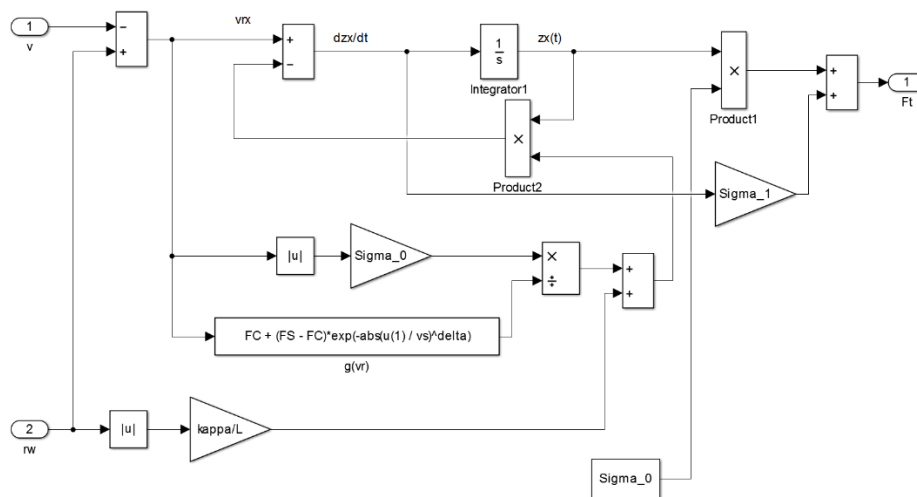
$$\kappa = \frac{r\omega_k - v_x}{v_x} \quad (14)$$

v_x – longitudinalna brzina vozila [m/s]

r – radijus kotača [m]

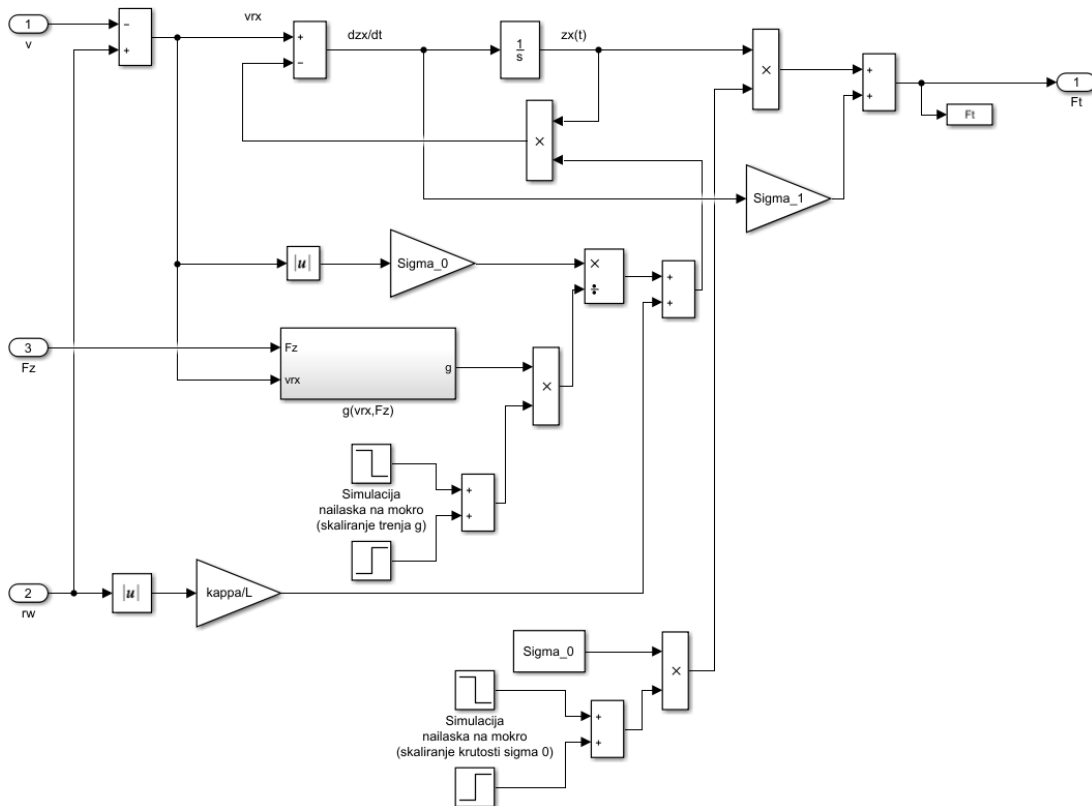
ω_k – kutna brzina kotača [rad/s]

Za izračun sile trenja iz uzdužnog klizanja i vertikalne sile koristit ćemo LuGre model [3]:



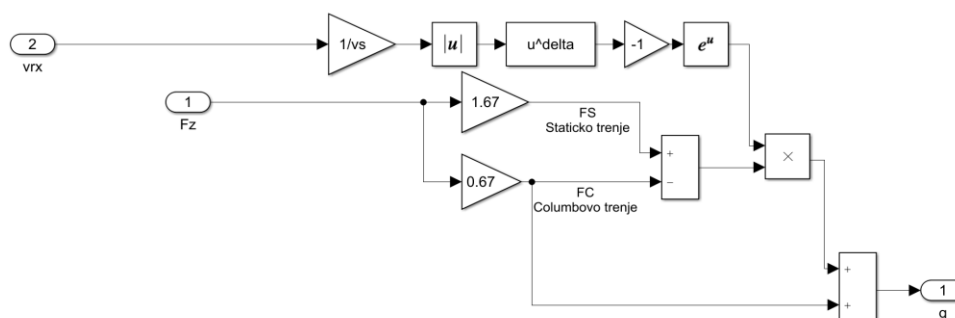
Slika 6. Model gume ($F_z = \text{konst.}$, karakteristika podloge = konst.)

Kako se pri ubrzanju, zbog prijenosa težine, mijenja vertikalna sila F_z i kako postoji mogućnost nailaska na različite podloge (kiša, istrošen dio asfalta), model ćemo proširiti kako bi opisivao i te pojave.



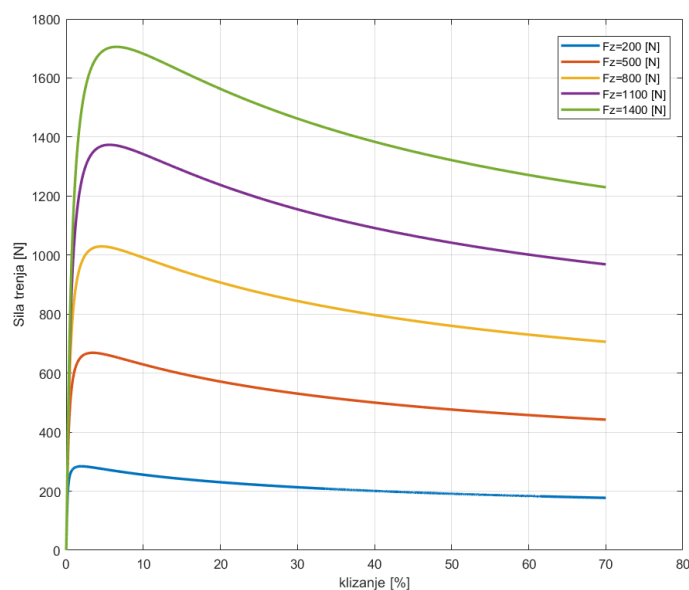
Slika 7. Prošireni model gume

Gdje je $g(v_x, F_z)$:



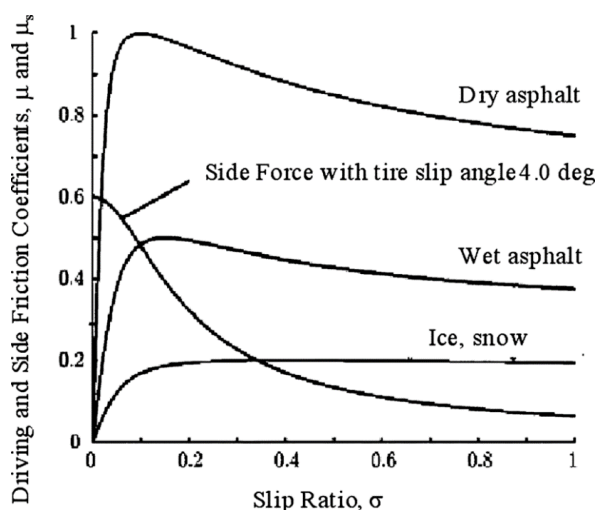
Slika 8. $g(v_x, F_z)$

Na slici 9. prikazana je ovisnost uzdužne sile trenja u ovisnosti o postotku klizanja za različite vertikalne sile.



Slika 9. Ovisnost sile trenja o postotku klizanja gume za razne vertikalne sile na kotač

Vidimo da je potencijal sile trenja najveći u području malih klizanja $\sim 5\%$. Iako nam se može učiniti da nakon maksimuma sile trenja ne gubimo znatnu silu trenja sa povećanjem klizanja, ono što se događa je znatan gubitak moguće lateralne sile trenja što uzrokuje lateralnu nestabilnost vozila, vidjeti sliku 10. Stoga su koristi TCS-a i povećanje bočne stabilnosti vozila.



Slika 10. Ovisnost longitudinalnih i lateralnih sila trenja o postotku klizanja

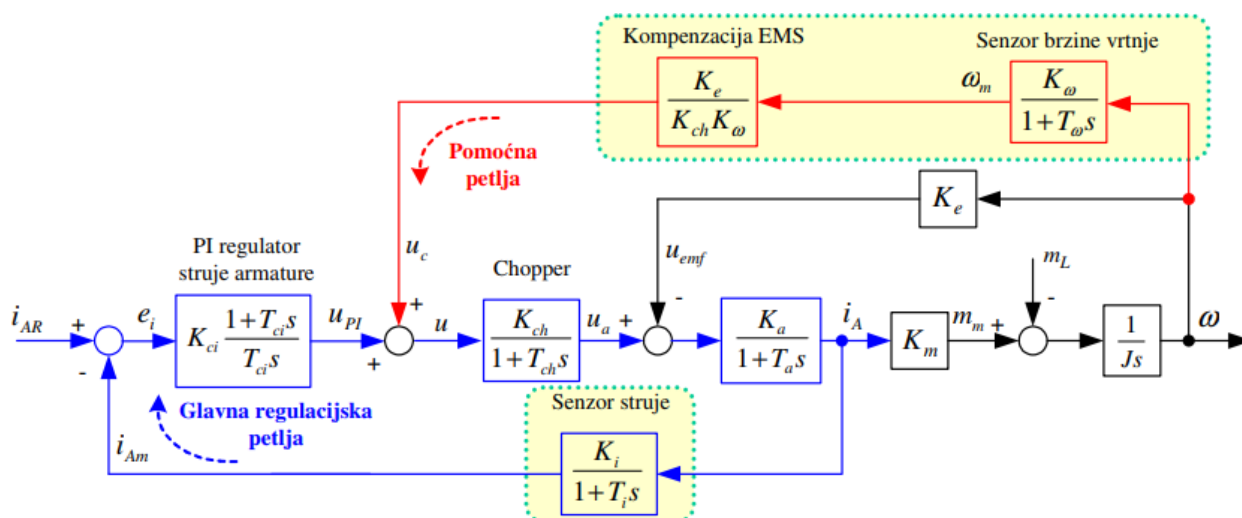
Budući da razmatrano vozilo ima precizno izračunati kapacitet baterije za završetak utrke radi smanjenja mase, još jedan od benefita smanjenja klizanja je smanjenje gubitaka u pogonu koji bi se morali nadoknaditi iz baterije.

3.2. Projektiranje regulatora na razini pogonskih kotača

U slučaju znatnog proklizavanja kotača (npr. primjena velikih momenta, ili skliska podloga) regulatorom broja okretaja zadržat ćemo broj okretaja pogonskih kotača samo nekoliko postotaka većim od brzine slobodno kotrljajućih kotača.

3.2.1. Regulator struje/okretnog momenta

Aktuator, tj. izlaz regulatora broja okretaja je okretni moment na kotaču/motoru te stoga prvo valja projektirati regulator koji će moći ostvariti taj zahtijevani moment. Budući da je moment direktno proporcionalan struji dovoljno je regulirati tu veličinu. Izlaz regulatora struje je napon na motoru. Na stvarnoj formuli regulator struje tj. momenta izveden je u samom frekvencijskom pretvaraču.



Slika 11. Blokovski dijagram podređenog regulacijskog kruga struje armature s PI regulatorom i kompenzatorom utjecaja elektromotorne sile

K_{ch} – pojačanje pretvarača napona $2U_b/U_{cpu}$

T_{ch} – vremenska konstnta frekvencijskog pretvarača [s]

U_b – napon baterije [V]

U_{cpu} – radni napon mikrokontrolera [V]

m_L – moment tereta [Nm]

K_i – pojačanje senzora struje [V/A]

T_i – vremenska konstanta senzora struje [s]

Moment inercije kotača J zbroj je momenta inercije motora i momenta inercije kotača reduciranog na vratilo motora:

$$J = J_{motor} + J_{w_{red}} \quad [kg \ m^2] \quad (15)$$

J – regulirana inercija [kg m^2]

J_{motor} – inercija rotora motora [kg m^2]

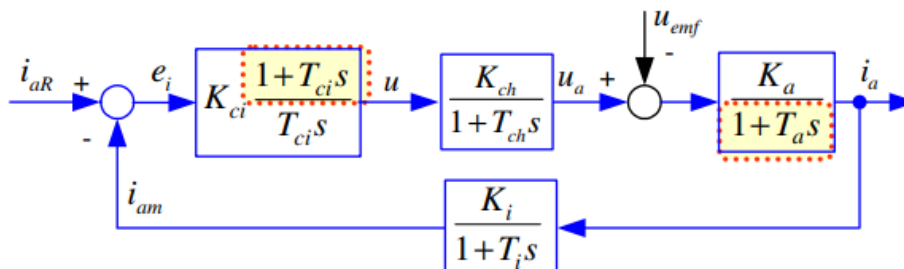
$J_{w_{red}}$ – inercija kotača reducirana na vratilo motora [kg m^2]

$$J_{w_{red}} = \frac{J_w}{i^2} \quad [kg \ m^2] \quad (16)$$

J_w – inercija kotača [kg m^2]

i – prijenosni omjer reduktora

„PI regulator se projektira uz pretpostavku da se elektromotorna sila (EMS) može smatrati vanjskim poremećajem (kojeg će kompenzirati kompenzator EMS-a ako se radi o brzim promjenama EMS ili samo PI regulator ako se radi o sporim promjenama“ [4]. Stoga u svrhu projektiranja regulatora možemo pojednostaviti regulacijski krug struje .



Slika 12. Blokovski dijagram regulacijskog kruga struje na kojem se zasniva sinteza PI regulatora struje

„Sinteza regulatora struje započinje podešavanjem vremenske konstante regulatora T_{ci} . Kako je vremenska konstanta armature T_a uobičajeno dominantna vremenska konstanta ($T_a \gg T_i, T_{ch}$), odziv zatvorenog regulacijskog kruga struje može se značajno ubrzati ukoliko se nulom regulatora $(1 + T_{ci}s)$ pokradi dominantna dinamika (pol) objekta upravljanja $(1 + T_a s)$, odnosno ako se odabere $T_{ci} = T_a$. Nakon toga korištenjem optimuma dvostrukog odnosa (ODO) dobivamo pojačanje regulatora „ K_{ci} “ i parametre za nadomjesni P1 član koji zamjenjuje cijeli zatvoreni regulacijski krug struje. Postupak se zasniva na izjednačavanju karakterističnog polinoma prijenosne funkcije regulacijskog kruga s karakterističnim polinomom optimuma dvostrukog odnosa“ [4].

$$A(s) = D_n D_{n-1}^2 \dots D_2^{n-1} T_e^n s^n + D_{n-1} D_{n-2}^2 \dots D_2^{n-2} T_e^{n-1} s^{n-1} + \dots \quad (17)$$

$$+ D_2 T_e^2 s^2 + T_e s + 1$$

T_e – ekvivalentna vremenska konstanta (daje informaciju o brzini odziva)

D_i – karakteristični odnosi ($i=2,3\dots n$), koji određuju prigušenje sustava

Postavljanjem svih karakterističnih odnosa na optimalni iznos 0.5 postiže se kvazi-aperiodski odziv regulacijskog kruga sa 6% nadvišenja u odzivu. U svrhu pojednostavljenja sinteze parazitska dinamika regulacijskog kruga (naponski pretvarač T_{ch} i senzor struje T_i) može se apriskimirati PT₁ članom:

$$G_{par}(s) = \frac{K_{ch}}{1 + T_{ch}s} \frac{1}{1 + T_i s} = \frac{K_{ch}}{1 + (T_{ch} + T_i)s} = \frac{K_{ch}}{1 + T_{\Sigma i} s} \quad (18)$$

$T_{\Sigma i}$ – parazitska vremenska konstnta regzulacijskog kruga struje [s]

Tako prijenosna funkcija regulacijskog kruga struje armature poprima slijedeći oblik:

$$G_{ci}(s) = \frac{i_s(s)}{i_{aR}(s)} = \frac{1}{1 + \frac{T_{ci}}{K_{ci}K_{ch}K_a} s^2} \quad \overleftrightarrow{ODO} \quad G_{ei}(s) = \frac{1}{1 + T_{ei}s + D_{2i}T_{ei}^2s^2} \quad (19)$$

Odavde slijedi da je ekvivalentna vremenska konstanta zatvorenog regulacijskog kruga struje aramture $T_{ei} = T_{\Sigma i}/D_{2i}$, što za $D_2=0.5$ iznosi $T_{ei}=2T_{\Sigma i}$. Nakon sređivanja imamo izraz za optimalno pojačanje regulatora:

$$K_{ci} = \frac{T_{ci}}{T_{\Sigma i}} \frac{D_2}{K_{ch}K_a} \rightarrow D_2 = 0.5 \rightarrow K_{ci} = \frac{T_{ci}}{T_{\Sigma i}} \frac{1}{2K_{ch}K_a} \quad (20)$$

Nadomjesna vremenska konstanta parazitske dinamike:

$$T_{\Sigma i} = T_i + T_{ch} \quad [s] \quad (21)$$

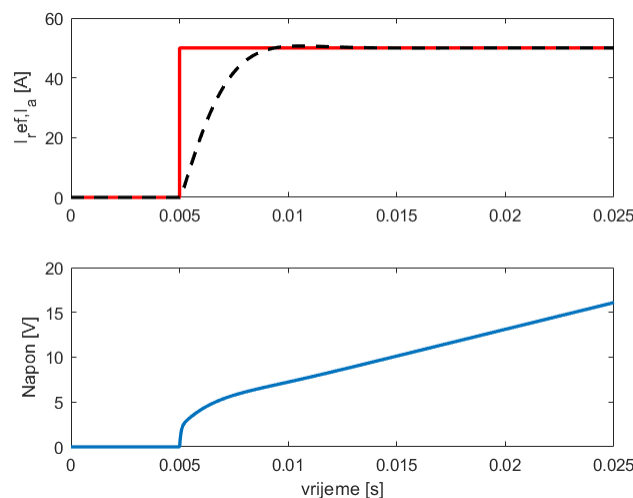
Integralna vremenska konstanta regulatora:

$$T_{ci} = T_a \quad [s] \quad (22)$$

Nadomjesna vremenska konstanta regulacijskog kruga struje:

$$T_{ei} = 2T_{\Sigma i} = 2(T_i + T_{ch}) \quad [s] \quad (23)$$

Prikazat ćemo odziv sustava na skokovitu pobudu reference struje sa regulatorom izračunatih parametara:



Slika 13. Odziv regulacijskog kruga struje na skokovitu pobudu

Za test regulatora kotač nije bio u kontaktu s podlogom te stoga napon brzo raste jer prati brzi porast broja okretaja kotača. Odziv struje je zadovoljavajuće brz, stabilan i točan.

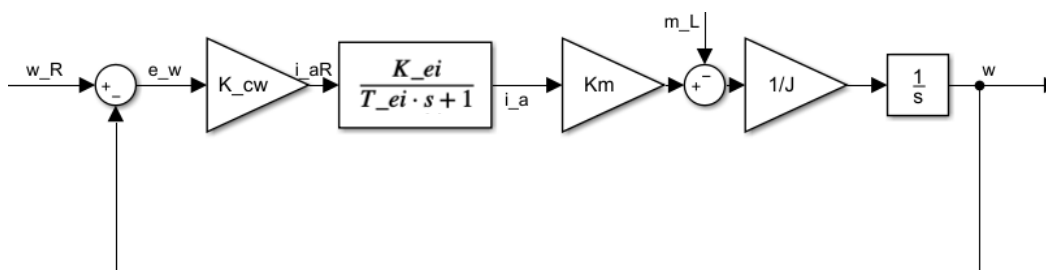
3.2.2. Regulator brzine vrtnje

Za potrebe sinteze regulacijskog kruga brzine vrtnje, regulacijski krug struje armature od referentne vrijednosti struje i_{a_ref} do stvarne struje armature i_a opsat ćemo pojednostavljeno PT1 članom sa slijedećom prijenosnom funkcijom:

$$G_{ei(s)} = \frac{i_a(s)}{i_{aR}(s)} \approx \frac{1}{1 + T_{ei}s} \quad (24)$$

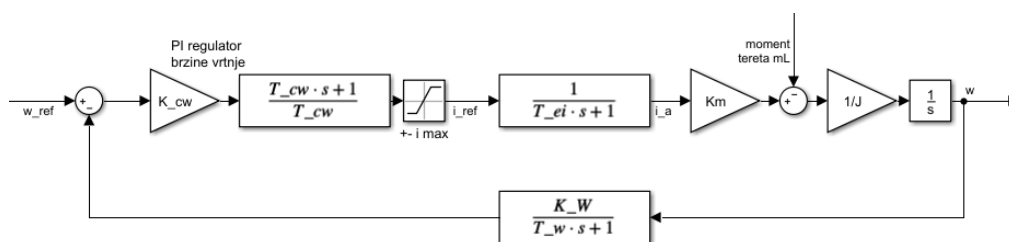
T_{ei} – nadomjesna vremenska konstanta regulatora struje [s]

Regulacijski krug brzine vrtnje sada možemo prikazati slijedećim blokovskim dijagramom:



Slika 14. Blokovski dijagram regulacijskog kruga brzine vrtnje s P regulatorom

Za ostvarivanje statičke točnosti s obzirom na skokovite promjene referentne vrijednosti brzine ω_{ref} bio bi dovoljan i P regulator (integriranje kutne akceleracije, uslijed momenta, u kutnu brzinu unosi integracijsko djelovanje u regulacijsku petlju). Međutim, kako bi se postigla stacionarna točnost regulacije brzine vrtnje u prisustvu skokovitih promjena momenata tereta m_L potrebno je dodati i I regulator. Kako bi se izbjegao efekt prenapinjanja integratora u regulatoru čiji je izlaz limitiran (tipično na +/- dvostruku nazivnu struju), regulator također uključuje i reset-antiwindup intervenciju). Na slici 15 ova funkcionalnost je predstavljena samo simbolom limita radi lakše predodžbe ukupne funkcionalnosti.



Slika 15. Blokovski dijagram regulacijskog kruga brzine vrtnje s podređenim regulatorom struje

U svrhu pojednostavljenja sinteze PI regulatora brzine vrtnje parazitska dinamika regulacijskog kruga struje i vremena uzorkovanja aproksimira se slijedećim nadomjesnim članom:

$$G_{par}(s) = \frac{1}{1 + T_{ei}s} \cdot \frac{1}{1 + T_us} \approx \frac{1}{1 + (T_{ei} + T_u)s} = \frac{1}{1 + T_{\Sigma\omega}s} \quad (25)$$

$T_{\Sigma\omega}$ – parazitska vremenska konstanta regulacijskog kruga brzine vrtnje [s]

T_u – vrijeme uzorkovanja mikroprocesora [s]

Parametri PI regulatora brzine vrtnje određuju se primjenom kriterija optimuma dvostrukog odnosa na prijenosnu funkciju zatvorenog regulacijskog kruga brzine.

$$G_{c\omega}(s) = \frac{\omega_m(s)}{\omega_R(s)} = \frac{1 + T_{c\omega}s}{1 + T_{c\omega}s + \frac{T_{c\omega}}{K_{c\omega}} \frac{J}{K_{c\omega}K_m} s^2 + \frac{T_{c\omega}JT_{\Sigma\omega}}{K_{c\omega}K_m}} \quad (26)$$

Karakteristični polinom prijenosne funkcije zatvorenog regulacijskog kruga brzine vrtnje izjednačava se s karakterističnim polinomom optimuma dvostrukog odnosa:

$$A_{odo}(s) = 1 + T_{e\omega}s + D_{2\omega}T_{e\omega}^2s^2 + D_{3\omega}D_{2\omega}^2T_{e\omega}^3s^3 \quad (27)$$

Odavde se izravno dobije da je ekvivalentna vremenska konstanta zatvorenog kruga jednaka integralnoj vremenskoj konstanti PI regulatora:

$$T_{e\omega} = T_{c\omega} \quad (28)$$

Rješavanjem sustava jednažbi:

$$\left. \begin{aligned} \frac{D_{3\omega}D_{2\omega}^2T_{e\omega}^3}{D_{2\omega}T_{e\omega}^2} = D_{3\omega}D_{2\omega}T_{e\omega} = T_{\Sigma\omega} \\ \frac{D_{2\omega}T_{e\omega}^2}{T_{e\omega}} = D_{2\omega}T_{e\omega} = \frac{J}{K_{c\omega}K_m} \end{aligned} \right\} \begin{aligned} T_{c\omega} = T_{e\omega} = \frac{T_{\Sigma\omega}}{D_{3\omega}D_{2\omega}} \\ K_{c\omega} = \frac{J}{D_{2\omega}T_{e\omega}K_m} = \frac{D_{3\omega}J}{T_{\Sigma\omega}K_m} \end{aligned} \quad (29)$$

$$\left. \begin{aligned} \frac{D_{2\omega}T_{e\omega}^2}{T_{e\omega}} = D_{2\omega}T_{e\omega} = \frac{J}{K_{c\omega}K_m} \end{aligned} \right\} \begin{aligned} T_{c\omega} = T_{e\omega} = \frac{T_{\Sigma\omega}}{D_{3\omega}D_{2\omega}} \\ K_{c\omega} = \frac{J}{D_{2\omega}T_{e\omega}K_m} = \frac{D_{3\omega}J}{T_{\Sigma\omega}K_m} \end{aligned} \quad (30)$$

Za $D_{2\omega} = D_{3\omega} = 0.5$

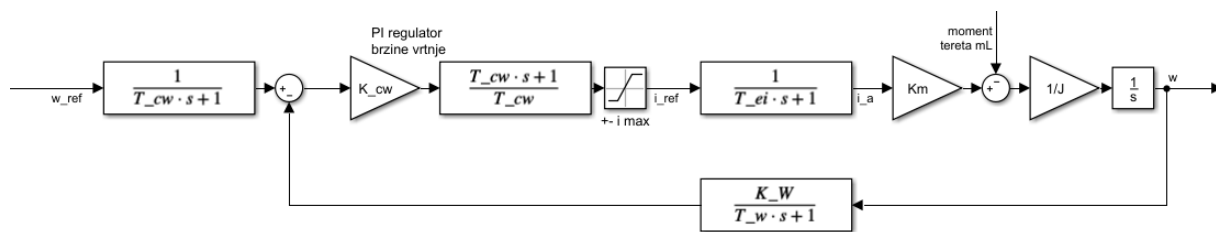
$$T_{c\omega} = T_{e\omega} = 4T_{\Sigma\omega} \text{ [s]} \quad (31)$$

$$K_{c\omega} = \frac{J}{2T_{\Sigma\omega}K_m} \text{ [Nm/(rad/s)]} \quad (32)$$

Prijenosna funkcija zatvorenog kruga je:

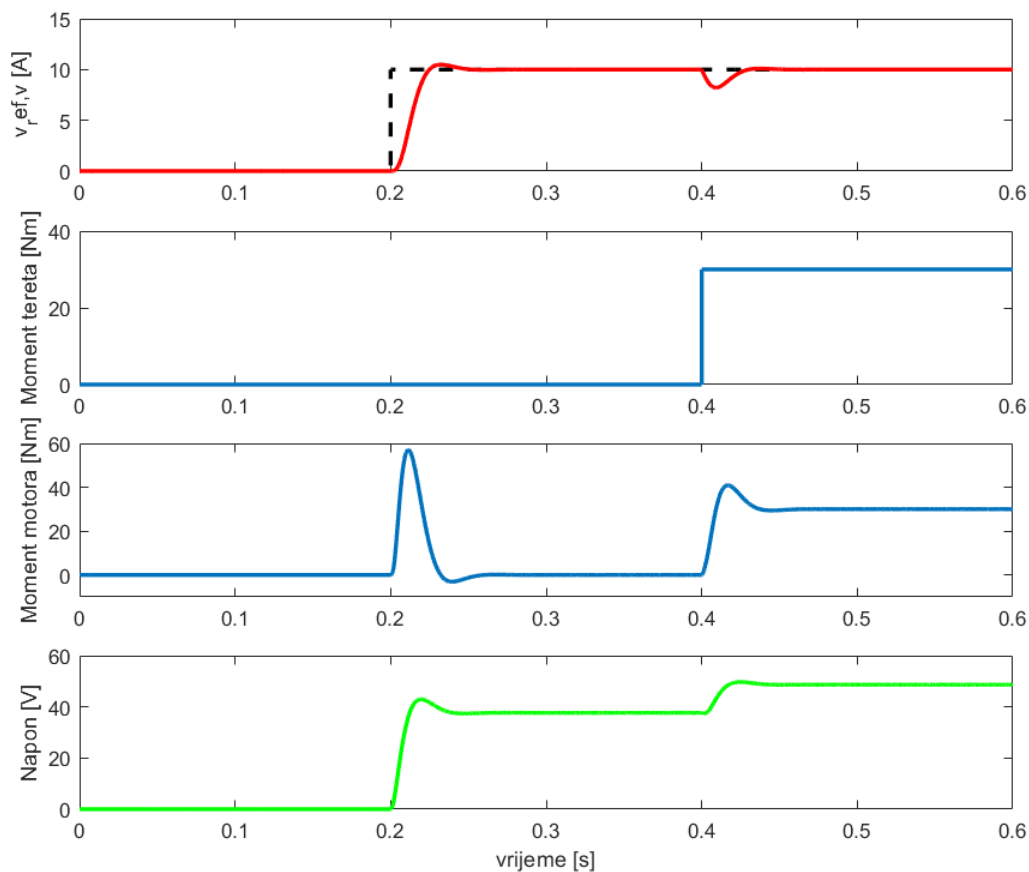
$$G_{c\omega} = \frac{w_m(s)}{w_R(s)} = \frac{1 + T_{e\omega}s}{1 + T_{e\omega}s + D_{2w}T_{e\omega}^2s^2 + D_{3w}D_{2w}^2T_{e\omega}^3s^3} \quad (33)$$

Kako bi poništili derivirajuće djelovanje nule u prijenosnoj funkciji zatvorenog kruga ($T_{e\omega}s$ u brojniku) na referencu ćemo dodati prefiltar sa vremenskom konstantom $T_f=T_{c\omega}$ koji će „smiriti“ sustav pri skokovima reference, ali neće umanjiti performanse uklanjanja poremećaja.



Slika 16. Regulator broja okretaja s prefiltrom

Za parametre realne formule dobivamo $K_{c\omega}=3.66$ i $T_{c\omega}=0.0125$, te odziv regulatora na skokovitu pobudu i poremećaj skokovitog momenta tereta izgleda kako je dano na slici 17.:



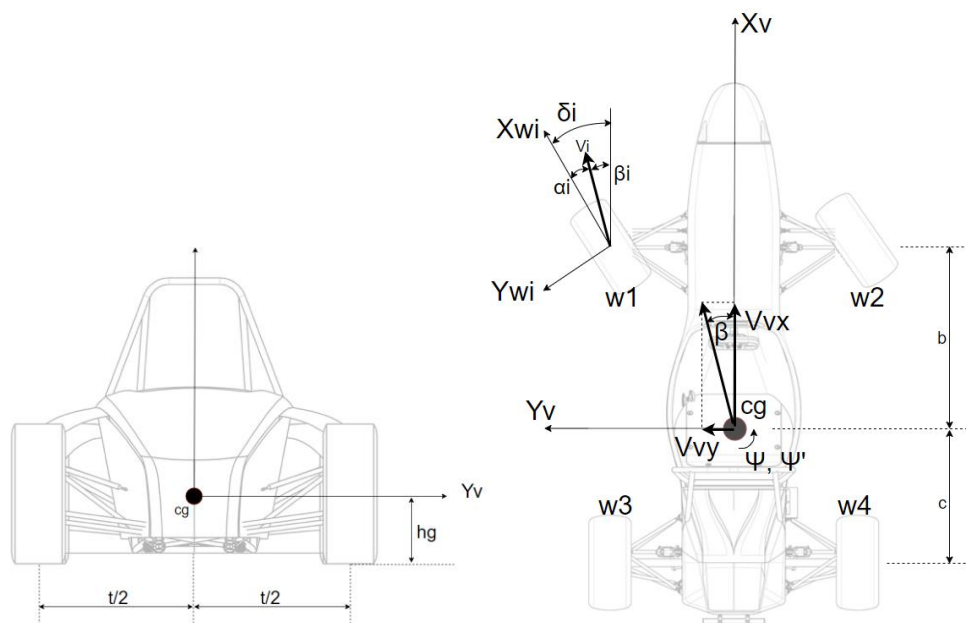
Slika 17. Odziv regulacijskog kruga broja okretaja na skokovitu pobudu i poremećaj (moment tereta)

Možemo uočiti poprilično brz odziv sa samo 2-3% prebačaja (prebačaj je puno veći bez prefiltra) i dobro prigušeno ponašanje zatvorenog regulacijskog kruga brzine vrtnje. Vidimo i brzu korekciju broja okretaja pri pojavi poremećaja tj. momenta tereta u 0.4 s.

4. Torque Vectoring

4.1. Model sustava

Za potrebe simulacije rada TCS-a bio je dovoljan samo model longitudinalne dinamike vozila, no za Torque Vectoring (skraćeno TV) potreban je dvotračni model vozila kako bi se dovoljno dobro opisale sile na kotačima koje zakreću vozilo. Za to je potrebno je uzeti u obzir lateralni transfer težine, i izračunati kuteve klizanja svakog kotača.



Slika 18. Prikaz vozila sa označenim veličinama

Za početak ćemo izračunati statičke sile na kotačima te promjenu tih sila uslijed longitudinalnog tj. lateralnog ubrzanja te izraziti rezultatnu F_z za svaki kotač:

$$F_{z1\ stat}, F_{z2\ stat} = \frac{mg}{l} * c \quad [N] \quad (34)$$

$$F_{z3\ stat}, F_{z4\ stat} = \frac{mg}{l} * b \quad [N] \quad (35)$$

$F_{zi\ stat}$ = vertikalna sila kotača za statični slučaj [N]

$$F_{z1\ long}, F_{z2\ long} = -\frac{a_{xv}mh_g}{l} \quad [N] \quad (36)$$

$$F_{z3\ long}, F_{z4\ long} = \frac{a_{xv}mh_g}{l} \quad [N] \quad (37)$$

$F_{zi\ long}$ = promjena sile uslijed longitudinalne akceleracije [N]

a_{xv} – longitudinalna akceleracija vozila [m/s^2]

$$F_{z1\ lat}, F_{z3\ lat} = -\frac{a_{yv}mh_g}{t} \quad [N] \quad (38)$$

$$F_{z2\ lat}, F_{z4\ lat} = \frac{a_{yv} m h_g}{t} \quad [N] \quad (39)$$

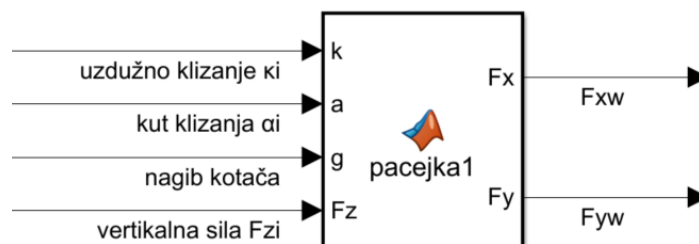
$$F_{zi} = F_{zi\ stat} + F_{zi\ long} + F_{zi\ lat} \quad [N] \quad (40)$$

$F_{zi\ lat}$ – promjena sile uslijed lateralne akceleracije [N]

a_{yv} – lateralna akceleracija vozila [m/s^2]

F_{zi} – vertikalna sila kotača i [N]

Horizontalne sile koje se javljaju između vozila i podloge opisujemo modelom gume. Ulazi u model gume su: vertikalna sila F_{zi} , kut klizanja α i uzdužno klizanje κ . Pri pogonu uzdužno klizanje javljat će se samo na pogonskim kotačima tj. stražnjima. Uz pretpostavku da TCS ispunjava svoju ulogu, pri ubrzanju umjesto momenta na kotačima možemo definirati uzdužno klizanje koje će tako definirati uzdužnu silu koja ubrzava vozilo kako bi pojednostavili model. Stoga nam preostaje definirati kuteve klizanja kako bi imali sve ulaze u model gume. Model gume definiran je pomoću [5].

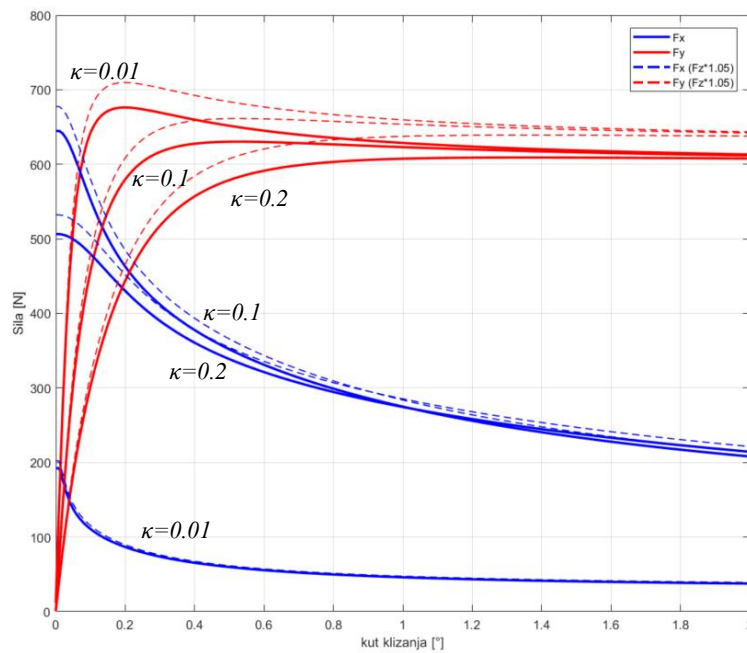


Slika 19. Pacejka model gume

F_{xw} – sila na kotač u smjeru osi x kotača [N]

F_{yw} – sila na kotač u smjeru osi y kotača [N]

Prikazat ćemo iznose sile za različite iznose kuteva klizanja pri različitim iznosima uzdužnih klizanja za dva iznosa vertikalne sile: $F_z = \frac{1}{4}G$ i 5% povećanja F_z .



Slika 20. Karakteristike gume

Kuteve klizanja za svaki kotač definiramo kao kut između vektora brzine vozila u toj točki i kuta zakreta kotača.

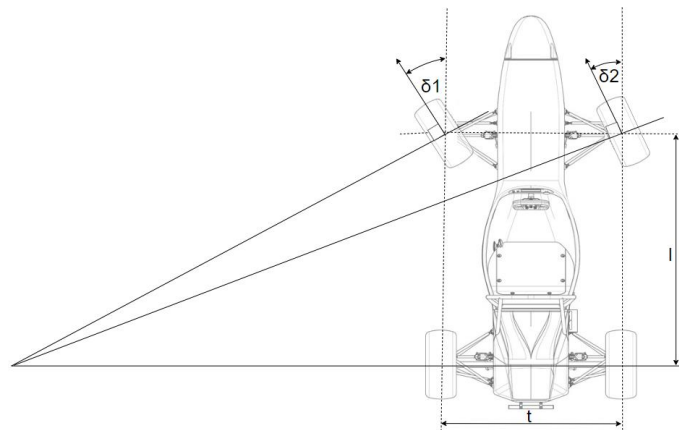
$$\alpha_i = -(\delta_i - \beta_i) \quad (41)$$

α_i – kut klizanja kotača i [rad]

δ_i – kut zakreta kotača i [rad]

β – kut između brzine vozila u kontaktnoj površini i uzdužne osi vozila [rad]

Stražnji kotači se ne zakreću te su stoga $\delta_3 = \delta_4 = 0$,



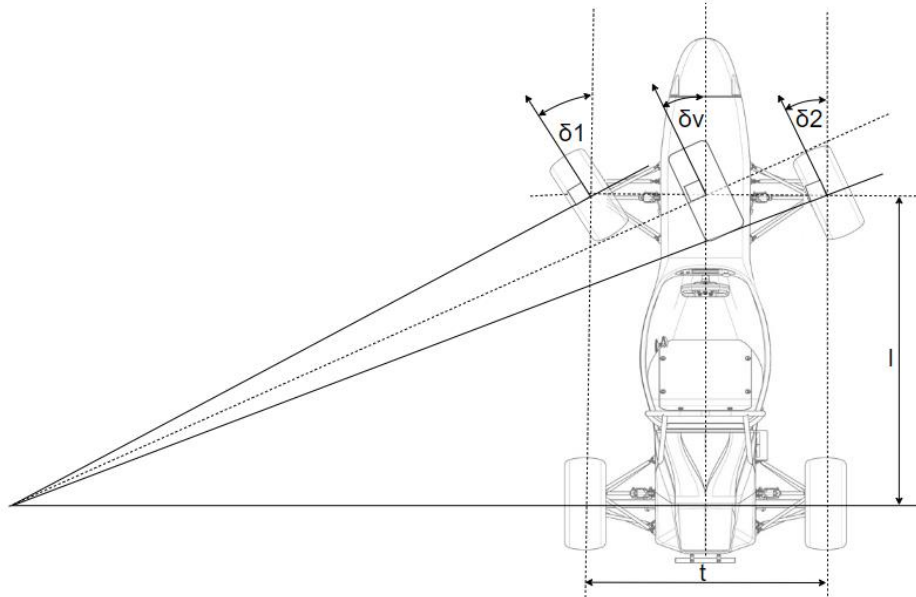
Slika 21. Ilustracija kuteva skretanja - Ackermanovo skretanje

Iz zadane geometrije možemo izvesti:

$$l * ctg(\delta_2) - l * ctg(\delta_1) = t \quad (42)$$

t -trag kotača [m]

Možemo definirati kut skretanja za središnji virtualni kotač δ_v koji će služiti kao upravljačka veličina iz kojeg ćemo definirati kut pojedinog kotača.



Slika 22. Upravljački kut skretanja

$$l * ctg(\delta_v) - l * ctg(\delta_1) = \frac{t}{2} \quad (43)$$

$$l * ctg(\delta_2) - l * ctg(\delta_v) = \frac{t}{2} \quad (44)$$

δ_v – kut imaginarnog središnjeg kotača [rad]

Iz čega slijedi:

$$\delta_1 = \text{atan} \left(\frac{1}{\frac{1}{\tan(\delta_v)} - \frac{t}{2l}} \right) \text{ [rad]} \quad (45)$$

$$\delta_2 = \text{atan} \left(\frac{1}{\frac{1}{\tan(\delta_v)} + \frac{t}{2l}} \right) \text{ [rad]} \quad (46)$$

Preostaje nam odrediti kuteve β_i :

$$\beta_i = \text{atan}\left(\frac{V_{xvi}}{V_{yvi}}\right) \quad [\text{rad}] \quad (47)$$

V_{xvi} – longitudinalna brzina vozila u kontaktnoj točki kotača i

V_{yvi} – lateralna brzina vozila u kontaktnoj točki kotača i

$$V_{xvi} = V_{xv} + \psi' * \frac{t}{2} * (-1)^i \quad i = 1,2,3,4 \quad \left[\frac{m}{s}\right] \quad (48)$$

$$V_{yv1}, V_{yv2} = V_{yv} + \psi' * b \quad \left[\frac{m}{s}\right] \quad (49)$$

$$V_{yv3}, V_{yv4} = V_{yv} - \psi' * c \quad \left[\frac{m}{s}\right] \quad (50)$$

ψ' – brzina zakreta vozila [rad/s]

Jedine nepoznanice sada su horizontalne brzine vozila i brzina zakreta vozila:

$$v_{xv} = \int a_{xv} dt \quad \left[\frac{m}{s}\right] \quad (51)$$

$$v_{yv} = \int a_{yv} dt \quad \left[\frac{m}{s}\right] \quad (52)$$

$$\psi' = \int \psi'' dt \quad \left[\frac{\text{rad}}{s}\right] \quad (53)$$

ψ'' – kutna akceleracija vozila [rad/s²]

Akceleracije i kutna akceleracija su:

$$a_{xv} = \frac{\Sigma F_{xv}}{m} + a_c \sin(\beta) \quad \left[\frac{m}{s^2}\right] \quad (54)$$

$$a_{yv} = \frac{\Sigma F_{yv}}{m} - a_c \cos(\beta) \quad \left[\frac{m}{s^2}\right] \quad (55)$$

$$\psi'' = \frac{M_{zp}}{I_{zz}} \quad \left[\frac{\text{rad}}{s^2}\right] \quad (56)$$

a_c – centripetalna akceleracija vozila [m/s²]

M_{zp} – Moment vozila oko Z-osi uslijed pasivne dinamike vozila [Nm]

I_{zz} – Moment inercije vozila oko Z-osi [kg m²]

$$M_z = -\left(F_{x1}\frac{t}{2} + F_{x3}\frac{t}{2} + F_{y3}c + F_{y4}c\right) + F_{x2}\frac{t}{2} + F_{x4}\frac{t}{2} + F_{y1}b + F_{y2}b \quad [\text{Nm}] \quad (57)$$

F_{xi} – sila na kotač i u smjeru x -osi vozila $[\text{N}]$

F_{yi} – sila na kotač i u smjeru y -osi vozila $[\text{N}]$

Radijus zakretanja je:

$$R = V_v / (\beta + \psi)' \quad [\text{m}] \quad (58)$$

R - radijus zakretanja vozila $[\text{m}]$

V_v – brzina vozila $\sqrt{V_{xv}^2 + V_{yv}^2}$ $[\text{m/s}]$

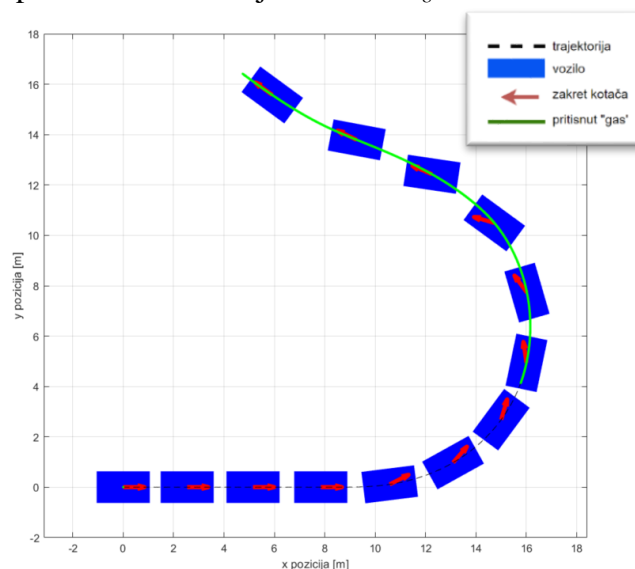
Te je centripetalna akceleracija:

$$a_c = \frac{V_v^2}{R} \quad \left[\frac{\text{m}}{\text{s}^2}\right] \quad (59)$$

2D transformacija sila kotača iz koordinatnog sustava kotača u koordinatni sustav vozila:

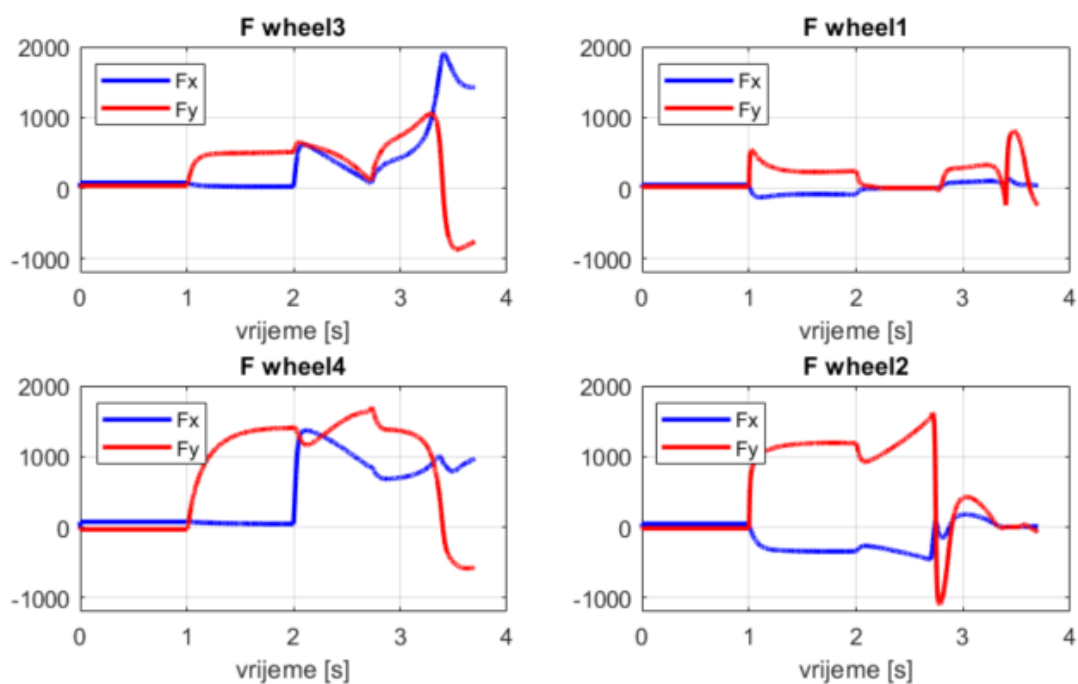
$$\begin{bmatrix} F_{xi} \\ F_{yi} \end{bmatrix} = \begin{bmatrix} \cos(\delta_i) & -\sin(\delta_i) \\ \sin(\delta_i) & \cos(\delta_i) \end{bmatrix} \begin{bmatrix} F_{xwi} \\ F_{ywi} \end{bmatrix} \quad [\text{N}] \quad (60)$$

Pokazat ćemo pasivnu dinamiku vozila u scenariju gdje vozač prolazi kroz kratki oštri zavoј te pritisće „gas“ u trenutku prolaska kroz zavoј u trenutku $t_{gas} = 2.00$ s.

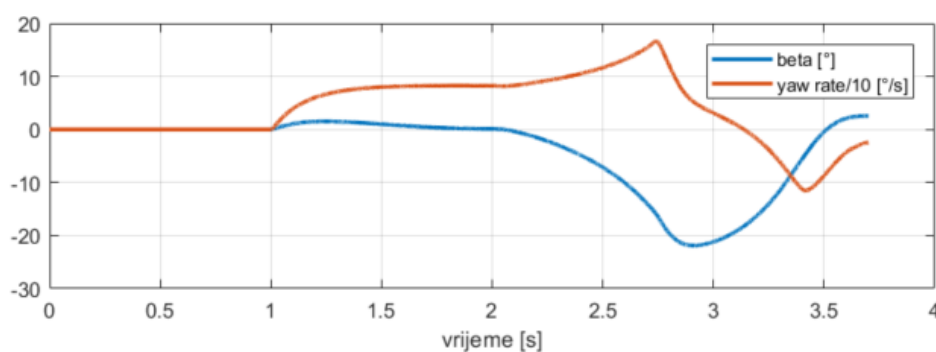


Slika 23. Trajektorija pasivnog ponašanja vozila

Vidimo da se ubrzo nakon primjene momenta odnosno pritiskanja papučice „gasa“ kod vozila javlja značajno preupravljanje (oversteer), tj. prekomjerno zanošenje/zakretanje vozila. Ovakvo ponašanje je nestabilno te ga je teško kontrolirati, a također rezultira i sporijim prolaskom kroz zavoj.

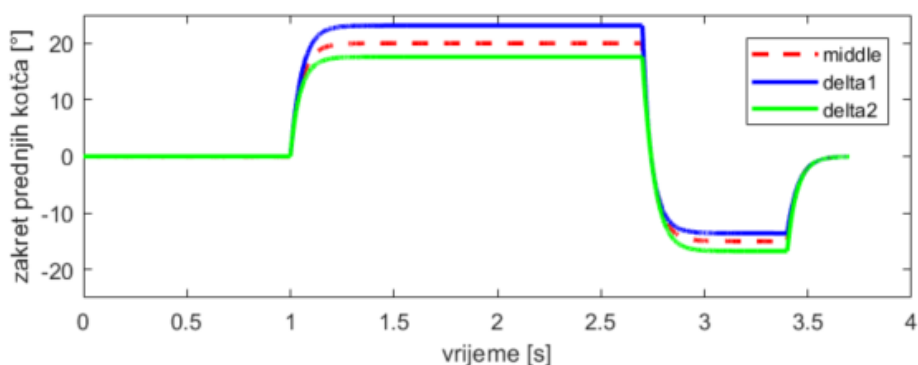


Slika 24. Horizontalne sile svakog kotača



Slika 25. Kut skretanja β i brzina zakreta vozila

Negativni kut β u lijevom zavoju ukazuje na preupravljanje (eng. *oversteer*) vozila, što je vidljivo i prikazanoj trajektoriji. Vozač ispravlja ponašanje vozila kratkim i naglim okretanjem volana u suprotnom smjeru.

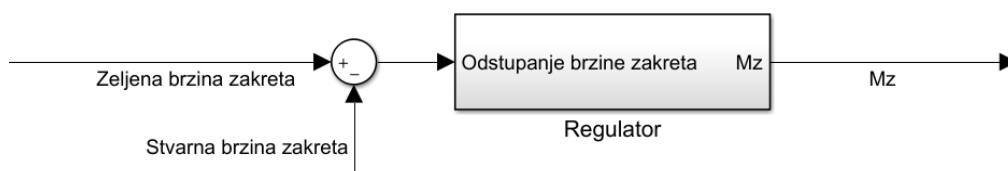


Slika 26. Kut zakreta prednjih kotača

Uočeno je da male razlike u vremenu reagiranja preko kuta zakreta uvelike utječu na ishod prolaska zavoja, točnije na okretanje/izlijetanje. Brzine reagiranja su na rubu ljudskog vremena reagiranja.

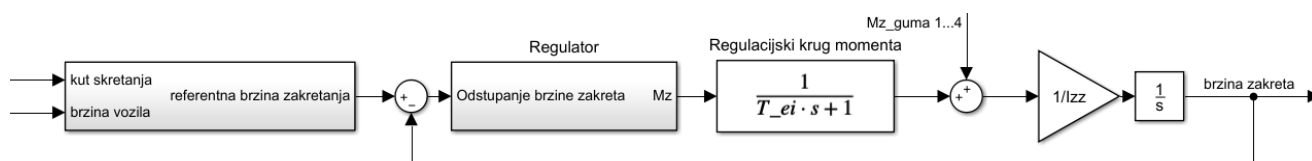
4.2. Projektiranje regulatora

Kako smo ranije spomenuli TV određuje željenu brzinu zakreta vozila, uspoređuje ju sa stvarnom, te na tu razliku reagira momentom oko Z-osi vozila.



Slika 27. Regulator brzine zakreta vozila

Kako bi mogli odrediti vrstu i pojačanja regulatora, vozilo ćemo reprezentirati inercijom na koje djeluju dva momenta oko Z-osi. Jedan je aktivni moment koji je izlaz iz regulatora i pasivni kojeg uzrokuju sile guma kojeg ćemo smatrati poremećajem. Odstupanje, tj. poremećaj na koji reagiramo će se javljati tak kada se vozilo ne ponaša kako smo predvidjeli, odnosno ne slijedi referentnu trajektoriju brzine skretanja.



Slika 28. Pojednostavljeni sustav s regulatorom

Ranije smo za regulator brzine utvrdili da je dovoljan P regulator, ali smo dodali i I za stacionarnu točnost. Ponašanje P regulatora je predvidivo sa stajališta vozača i neovisno o vremenu, dok bi I djelovanje moglo uzrokovati neželjenu dinamiku i nepredvidivo/neponovljivo ponašanje vozila što je vozačima strogo nepoželjno. Stoga ćemo zadržati samo P djelovanje. U svrhu pojednostavljenja sinteze regulatora brzine zakreta parazitska dinamika regulacijskog kruga momenta i vremena uzorkovanja aproksimira se slijedećim nadomjesnim članom:

$$G_{par}(s) = \frac{1}{1 + T_{ei}s} \cdot \frac{1}{1 + T_us} \approx \frac{1}{1 + (T_{ei} + T_u)s} = \frac{1}{1 + T_{\Sigma yaw}s} \quad (61)$$

$T_{\Sigma yaw}$ – parazitska vremenska konstanta regulacijskog kruga brzine zakreta [s]

Prijenosna funkcija zatvorenog regulacijskog kruga tada je:

$$G_{yaw}(s) = \frac{\omega_m}{\omega_R} = \frac{1}{1 + \frac{I_{zz}}{K_{yaw}}s + \frac{T_{\Sigma yaw}I_{zz}}{P}s^2} \quad (62)$$

Izjednačavanjem karakterističnog polinoma s karakterističnim polinomom optimuma dvostrukog odnosa:

$$A_{odo}(s) = 1 + T_{eyaw}s + D_{2yaw}T_{eyaw}^2s^2 \quad (63)$$

Dobivamo sustav jednadžbi:

$$T_{eyaw} = \frac{I_{zz}}{K_{yaw}} \quad (64)$$

$$D_2T_{eyaw} = \frac{T_{\Sigma yaw}I_{zz}}{K_{yaw}} \quad (65)$$

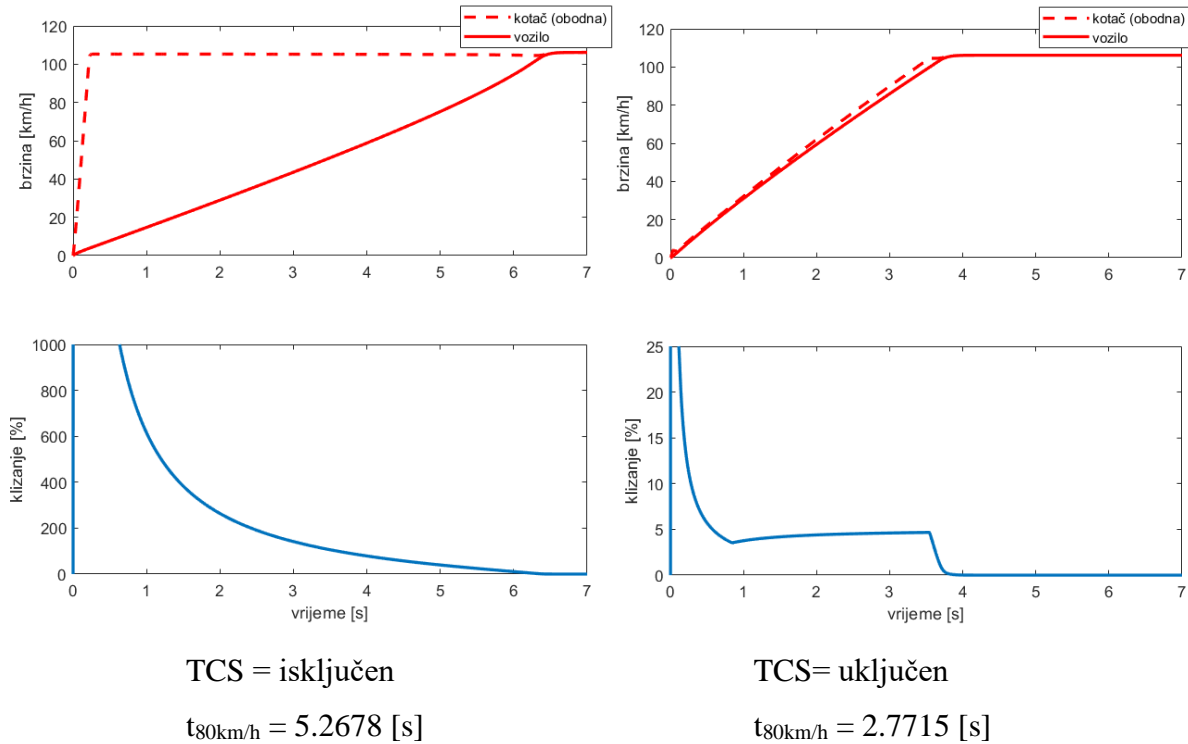
Sređivanjem i uvrštavanjem $D_2 = 0.5$ dobivamo pojačanje regulatora:

$$K_{yaw} = \frac{0.5}{T_{\Sigma yaw}} I_{zz} \quad (66)$$

Za realnu formulu $K_{yawf}=45 \cdot I_{zz_formula}$, a za maketu je $K_{yawm}=33 \cdot I_{zz_maketa}$

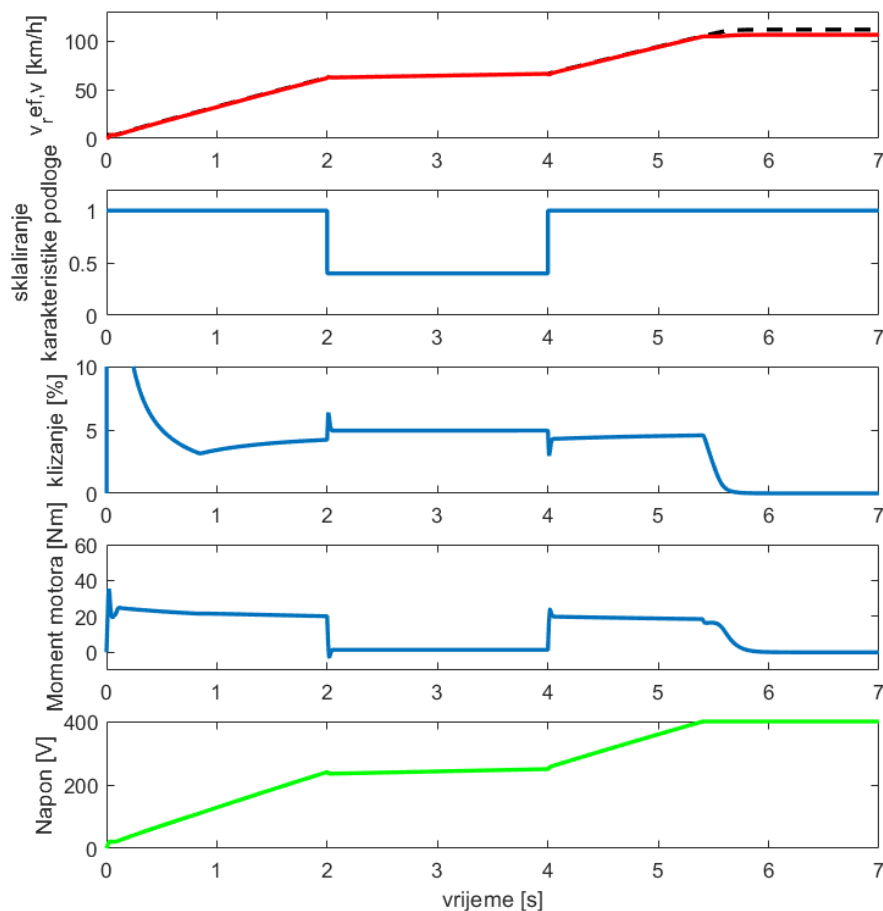
5. Rezultati simulacija

Prikazat ćemo ubrzanje formule sa i bez TCS-a



Slika 29. Ubrzanje sa i bez TCS-a

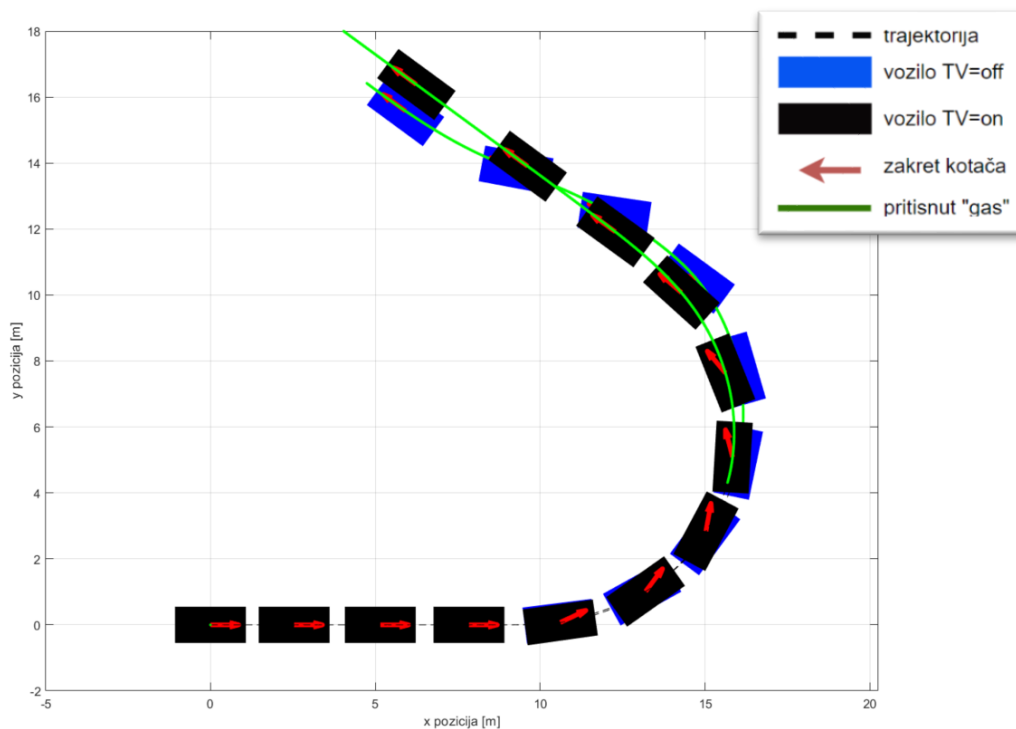
Kako se natjecanja organiziraju na raznim lokacijama često staze nisu zasebne odvojene površine asfalta već prelaze preko raznih površina kao što su parking, prijelaz preko ceste, mini rampe između različitih visina staze. Takve prijelaze predstaviti ćemo naglom promjenom kvalitete podloge, tj. trenja i provjeriti ponašanje sustava u takvim uvjetima.



Slika 30. Robusnost pri nailasku na lošiju podlogu

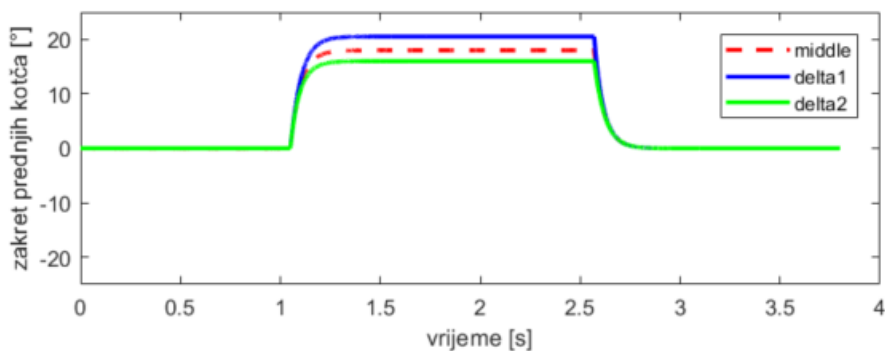
Vidimo da TCS regulator naglo reagira na promjenu karakteristike podloge u 2. sekundi kako bi klizanje imalo minimalno i što kraće odstupanje od zadanih 5%. Reagira sa naglim smanjenjem momenta kotača, a kada se karakteristika podloge ponovno promjeni u 4. sekundi reagira jednako dobro, ovaj put sa naglim povećanjem momenta.

Pokazat ćemo pasivnu dinamiku vozila u istom scenariju kao i prije gdje vozač prolazi kroz oštri zavoj te pritišće „gas“ za vrijeme prolaska zavoja, ali ovaj put sa TV-om uključenim. Ponašanje ćemo usporediti sa ponašanjem bez TV-a tako preklopimo trajektorije i orijentacije vozila za oba slučaja, sa i bez TV-a.



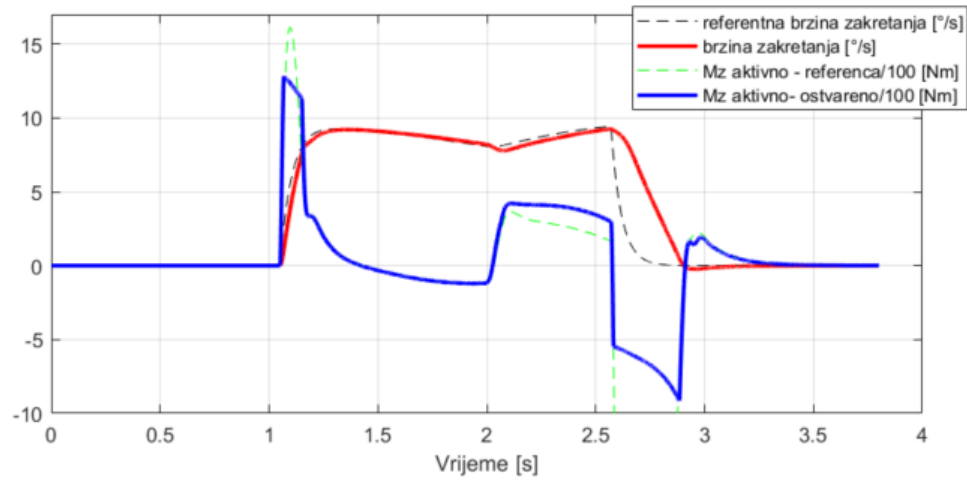
Slika 31. Usporedba trajektorije vozila sa i bez TV-a

Možemo vidjeti da vozilo prolazi zavoj kontrolirano i bez preupravljanja ili podupravljanja. Također vidimo da vozilo sa TV-om zauzima manje bočnog prostora, i ima predvidljivije bočno ponašanje što omogućuje vozaču da bolje iskoristi punu širinu staze pri prolasku kroz zavoj. Također vozač nije morao vršiti korekcije zakretom volana.

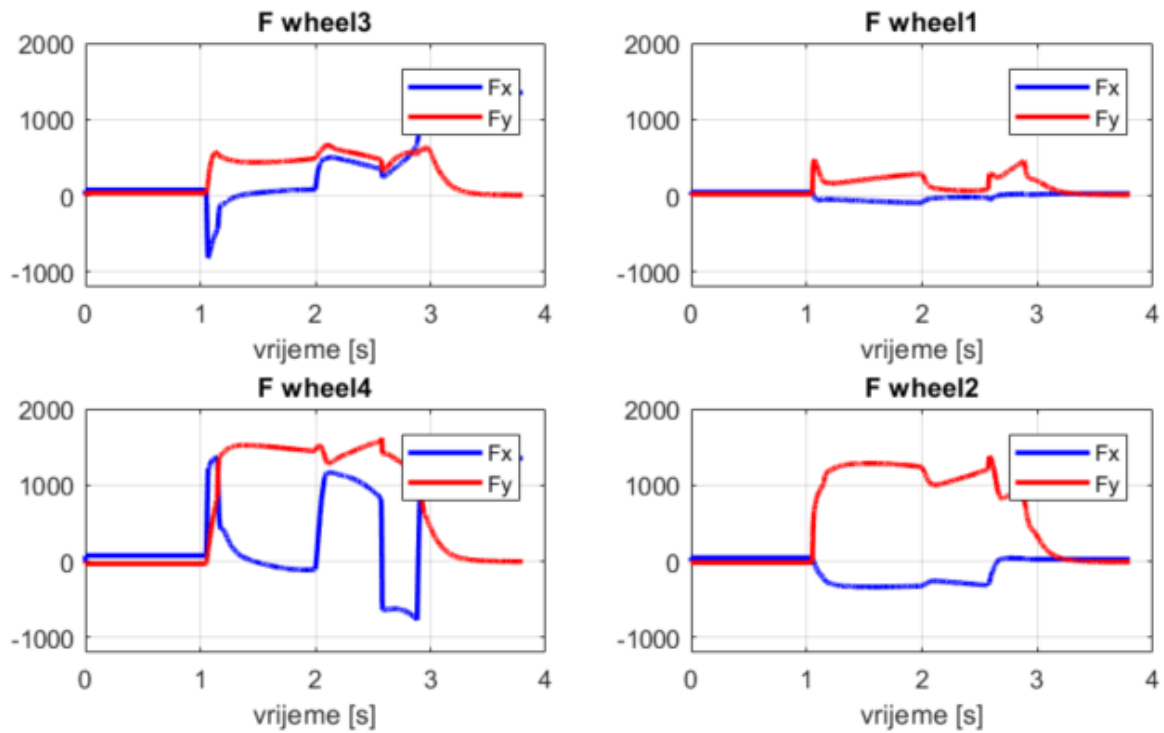


Slika 32. Kut zakreta kotača

Prikazat ćemo i dinamiku zakretanja vozila i djelovanje momenta oko Z-osi.



Slika 33. Brzine zakreta i aktivni Mz

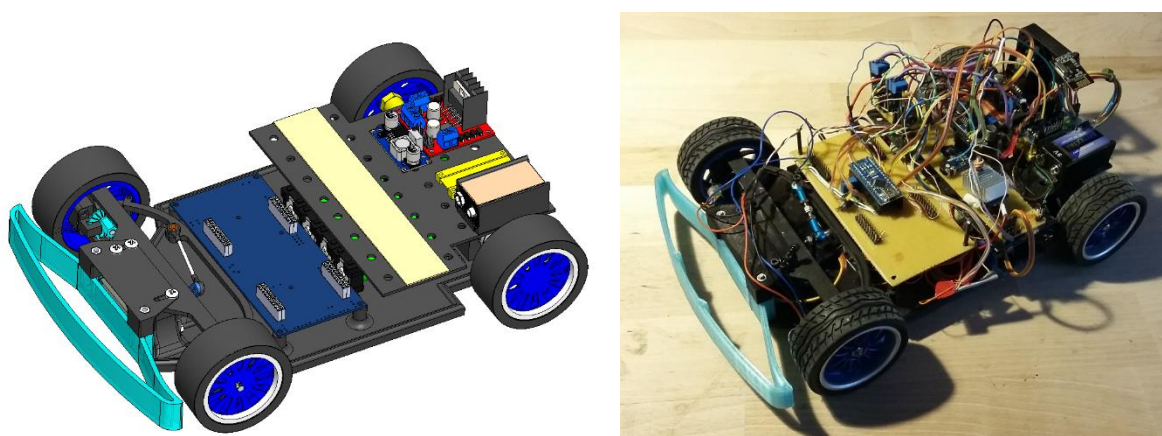


Slika 34. Horizontalne sile svakog kotača

Prema simulacijama vidimo značajno poboljšanje dinamike vozila sa razvijenim sustavima te ćemo u svrhu realnijih testova i boljeg razumijevanja konstruirati umanjenu maketu vozila sa svim funkcionalnim sustavima koji su bitni za testiranje prethodno opisanih sustava.

6. Izrada makete

U svrhu testiranja razvijenih algoritama za stabilizaciju vozila konstruirana je i proizvedena umanjena maketa bolida. Zadržani su odnosi između veličina kao što su trag kotača i međuosovinski razmak kao kod realne formule. Maketa također ima dva motora koji pogone zasebno svaki stražnji kotač. Brzine svakog kotača mjerene su enkoderima. Brza i laka proizvodnja i izmjena raznih dijelova moguća je zbog tehnologije izrade dijelova 3D printanjem.



Slika 35. CAD model makete i izrađena maketa

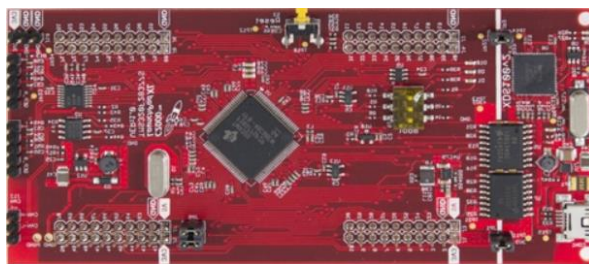
6.1. Kupovne komponente

Mikrokontroler

Inicijalno je odabran je Texas Instruments Launchpad C2000 228379D zbog mogućnosti programiranja iz Matlab Simulinka, CAN linije i jakog procesora [6]. Ideja je bila razviti upravljački kod na maketi te približno isti takav koristiti na bolidu.

Specifikacije:

- dvojezgreni 32-bitni procesor 200 MHz
- 1024 KB Flash memorije
- 204 KB RAM memorije
- 24x 16-bit analogni ulaz
- 24x PWM izlaz
- komunikacije: CAN, SPI, UART, I2C



Slika 36. TI Launchpad

Unatoč brojim poteškoćama pri početnom programiranju i lošoj on-line podršci mikrokontroler je uspješno programiran. No ubrzo se sa rastom kompleksnosti programa mikrokontroler pokazao nerobusnim na male izmjene u kodu. Rješavanje sve češćih problema postalo je dugotrajno, uz već nestabilnu UART komunikaciju prema računalu tijekom spremanja podataka za vrijeme rada, i ne funkcionalnu I2C komunikaciju sa senzorom zakreta. Zbog navedenih problema odabran je dobro poznat i pouzdan mikrokontroler sa obilnom i dostupnom on-line podrškom: Arduino Nano.



Slika 37. Arduino Nano

Specifikacije [7]:

- ATmega328 Procesor
- 16 MHz
- 32 KB Flash memorije
- 2 KB RAM
- 1KB EEPROM
- 8x 12-bit ulaz
- 14x digitalni ulaz/izlaz (6 pwm)
- 2x interrupt
- komunikacije: I2C, UART, SPI

Motori

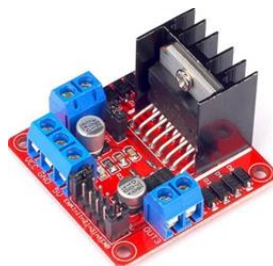
Motori su klasični DC motori sa ugrađenim planetarnim reduktorom i enkoderom. Nominalni napon motora je 12V. Reduktor ima prijenosni omjer 4,4 : 1. Enkoder je izveden pomoću magneta na vratilu motora i Hallovog senzora koji daje 58,6 impulsa pri jednom okretaju izlaznog vratila. Nazivni moment na izlaznom vratilu je 0,16 Nm. Definirana je struja kratkog spoja 5.6A



Slika 38. DC motor s enkoderom

PWM modul

12V PWM modul je izveden pomoću L298N dvostrukog integriranog H-mosta, što omogućuje spajanje dva DC motora na jedan modul. Maksimalni radni napon je 35 V, a nazivna struja je 4A kontinuirano.



Slika 39. L298N PWM Modul

Senzor struje

Modul senzora izveden je pomoću ACS712 integriranog kruga za mjerenje struje zasnovanom na Hallovom efektu. Osjetljivost je 185 mV/A. Senzor će biti korišten za informaciju o momentu motora.



Slika 40. Senzor struje

Optički enkoder

Služi za mjerenje brzine kotača, a sastoji se od optičkih vrata koja detektiraju prisutnost objekta između izvora svjetlosti i optičkog senzora, odnosno prekid svjetlosne zrake, i diska sa radijalnim prorezima koji se montira na kotač. Radijalni prorezi, ovisno o kutu zakreta diska prekidaju ili ne prekidaju zraku svjetlosti te tako na izlazu senzora, uslijed kontinuirane vrtnje kotača, dobivamo pravokutni signal frekvencije proporcionalne brzini vrtnje kotača.



Slika 41. Optički enkoder

Baterija

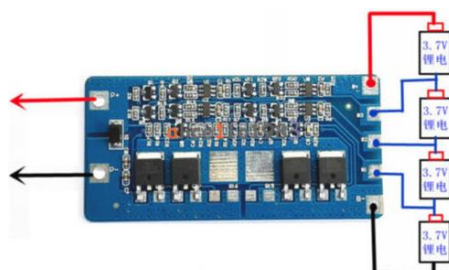
Korištene su Litij-ionske Sony VTC6 ćelije. Nazivni napon ćelije je 3.6V ima kapacitet 3000 mAh, i maksimalnu kontinuiranu struju 15A. Baterijski paket sastoji se od četiri ovakve ćelije spojene u seriju što daje radni napon u rasponu od 11,2 do 16,8V.



Slika 42. Li-ion ćelija Sony VTC6

Battery Management System

BMS je električni sklop kojem je uloga spriječiti kratki spoj baterije, prepunjavanje ćelija, preveliko ispražnjivanje, te balansirati napone ćelija. Maksimalna struja je 20A.



Slika 43. Battery Management System

Servo motor za skretanje

Za zakretanje kotača odabran je klasični modelarski servo motor. Napaja se sa 5V i može ostvariti ~1Nm momenta. Kut zakreta zadaje se duljinom impulsa raspona 1-2ms svakih 20ms što odgovara kutevima 0-90° [8]



Slika 44. RC servo motor

5V napajanja

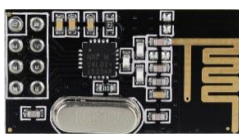
Za napajanje logičke elektronike korišten je LM7805 integriran na pločici H-mosta L298N zbog relativno niskih struja. Za potrebe servo motora koji potrebno je jače napajanje napona 5V te je stoga korišten DC-DC step-down modul LM2595 koji ujedno odvađa napajanje logičke elektronike od servomotora i smanjuje utjecaj šuma kojeg stvara rad snažnog servo motora



Slika 45. LM2595

Daljinsko upravljanje

Za bežično upravljanje vozilom iskorišten je stari RC daljinski kojem je uklonjena unutrašnja elektronika osim potencijometara. Ugrađen je Arduino Nano koji očitava pozicije upravljačkih potencijometara te ih zatim šalje mikroprocesoru na maketi. Posrednici između ta dva mikroprocesora su moduli za bežičnu komunikaciju NRF24L01. Zbog osjetljivosti bežične komunikacije na maketi se za to koristi zasebno 9V napajanje kako smetnje DC motora ne bi utjecale na bežični modul. Mikroprocesor za bežičnu komunikaciju na maketi je također Arduino Nano koji predstavlja vozača. Njegovi izlazi su analogni signali pozicije papučice „gasa“ i signal za zakret servo motora.



Slika 46. NRF24L01

Senzor brzine zakreta

Za senzor brzine zakreta vozila korišten je jednostavni i vrlo popularni modul MPU-6050.

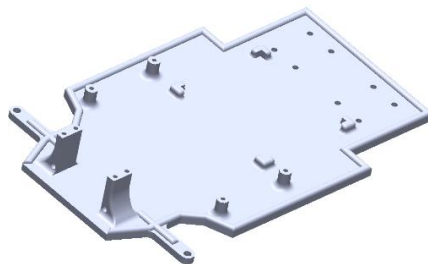


Slika 47. MPU-6050

6.2. Izrađene komponente

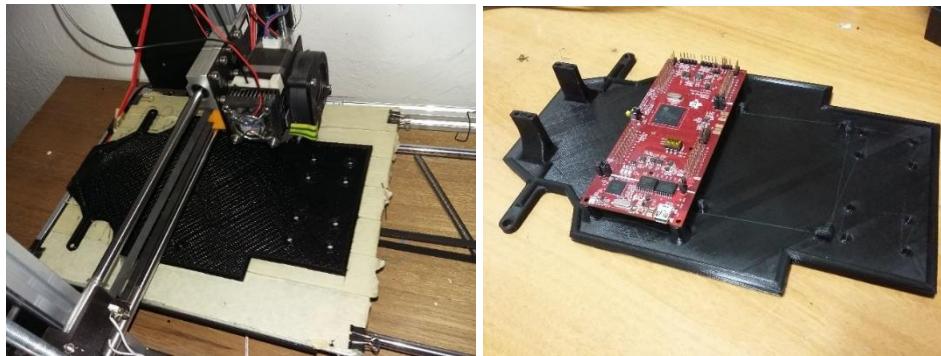
Osnovne dimenzije kao što su međuosovinski razmak i trag kotača skalirane su naspram realnog vozila. Modeli su izrađeni pomoću CAD programa Solidworks..

Podvozje



Slika 48. CAD podvozja

Na podvozju su konstruirani prihvati za većinu komponenata kao što su, nosači motora, servo motor, prednji ovjes i baterija. Debljina podvozja je 5mm i printana je sa 50% ispune iz materijala PLA.

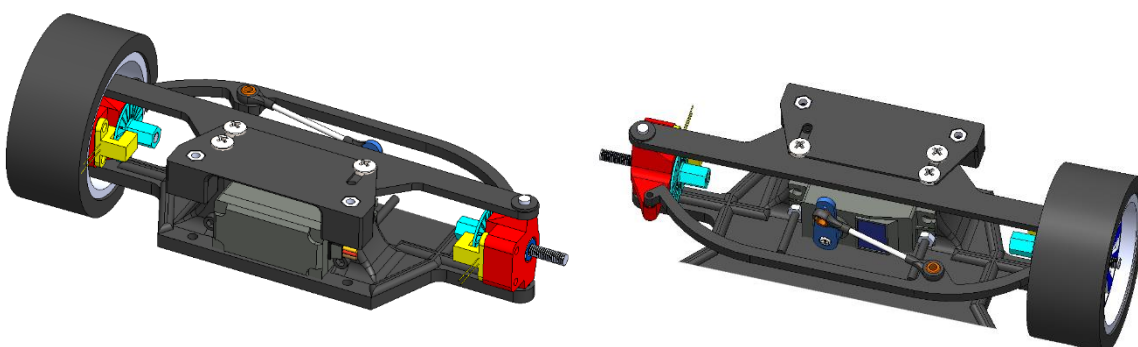


Slika 49. Podvozje

Skretanje

Sa konstrukcijskog stajališta skretanje je jedno od najkompleksnijih sustava ove makete. Kompleksnosti pridnose zahtjevi na nosač kotača tj *Upright*, a to su:

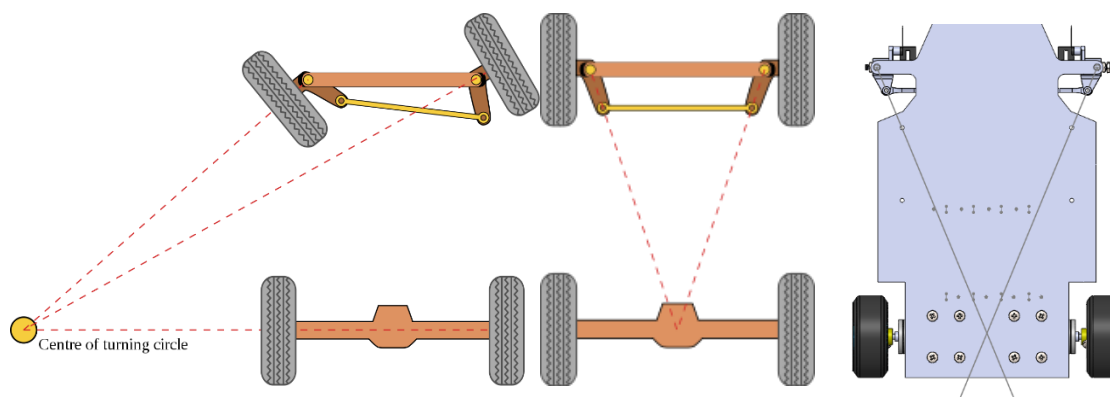
- nošenje kotača koji se slobodno vrti oko svoje osi simetrije
- zakretanje oko Z osi koja prolazi kroz centar kotača
- pomak po Z osi
- prihvat enkodera
- Ackermanovo skretanje
- ostaviti dovoljno prostora za disk enkodera



Slika 50. CAD prednjeg ovjesa (sakriven lijevi kotač)

Ackermanovo skretanje, ukratko, nalaže da je pri skretanju unutarnji kotač zakrenut više od vanjskoga. Pri skretanju vozilo se zakreće oko neke točke u ravnini vozila. Za ravnomjernu raspodjelu sila po kotačima ta točka predstavlja sjecište osi svih kotača kao što je prikazano na

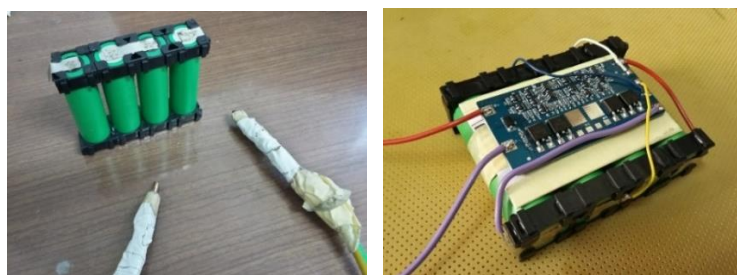
slici niže. Ostvaruje se tako da, kada su kotači izravnati, pravci koji prolaze kroz osi zakreta svakog kotača i kroz zglob letve za skretanje sijeku u polovištu stražnje osovine [9].



Slika 51. Zakreti kotača - Ackermanovo skretanje

Baterija

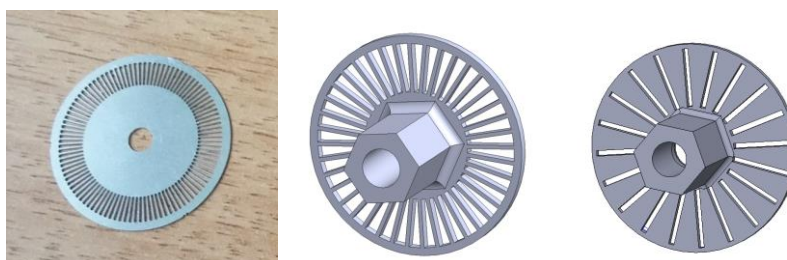
Baterija je sastavljena od četiri Sony VTC6 ćelije spojene u seriju točkastim zavarivanjem. Napon baterijskog paketa je tako 16,8 V za pune tj 12V za prazne ćelije.



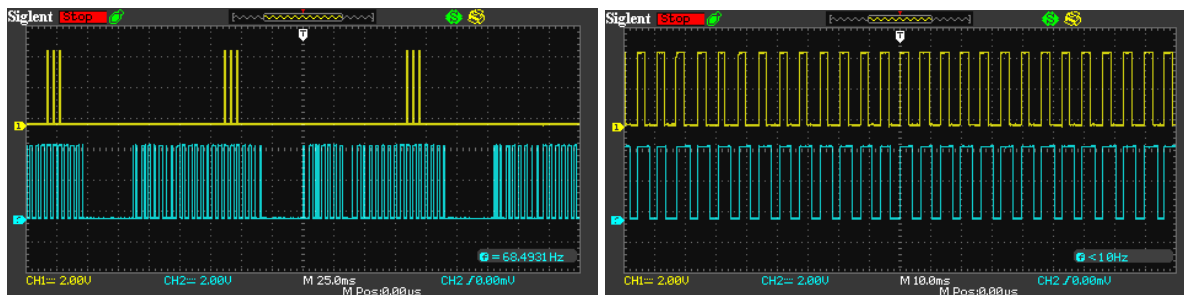
Slika 52. Baterija

Disk enkodera

Originalni disk enkodera ima vrlo uske proreze na malom rasponu radijusa što dovodi do puno loših očitavanja prilikom vibracija i zračnosti sklopa. Također je nepraktičan za montažu. Izrađen je novi disk koji ima manji, ali i dalje dovoljan, broj proreza, a koji daju pouzdaniji signal. Disk sa užim prorezima ima pouzdaniji signal.



Slika 53. Stari i nove verzije diska enkodera



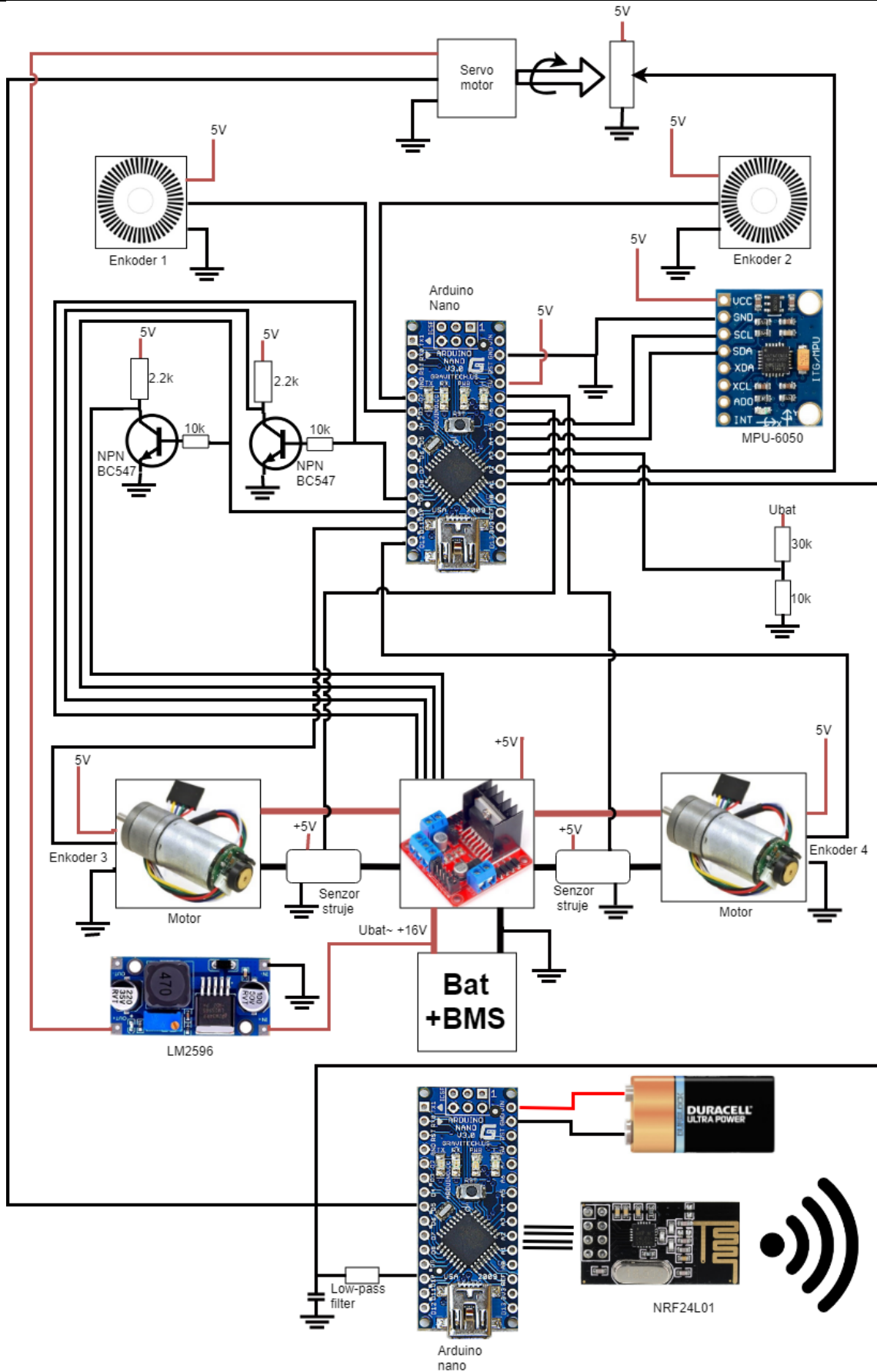
Slika 54. Očitanje pomoću osciloskopa - Usporedba signala starog (lijevo) i novog (desno) enkodera pri vrtnji

Pomoću osciloskopa vidljivo je da stari enkoder ima isprekidani signal, dok kod novoga nema niti jednog prekida u signalu.

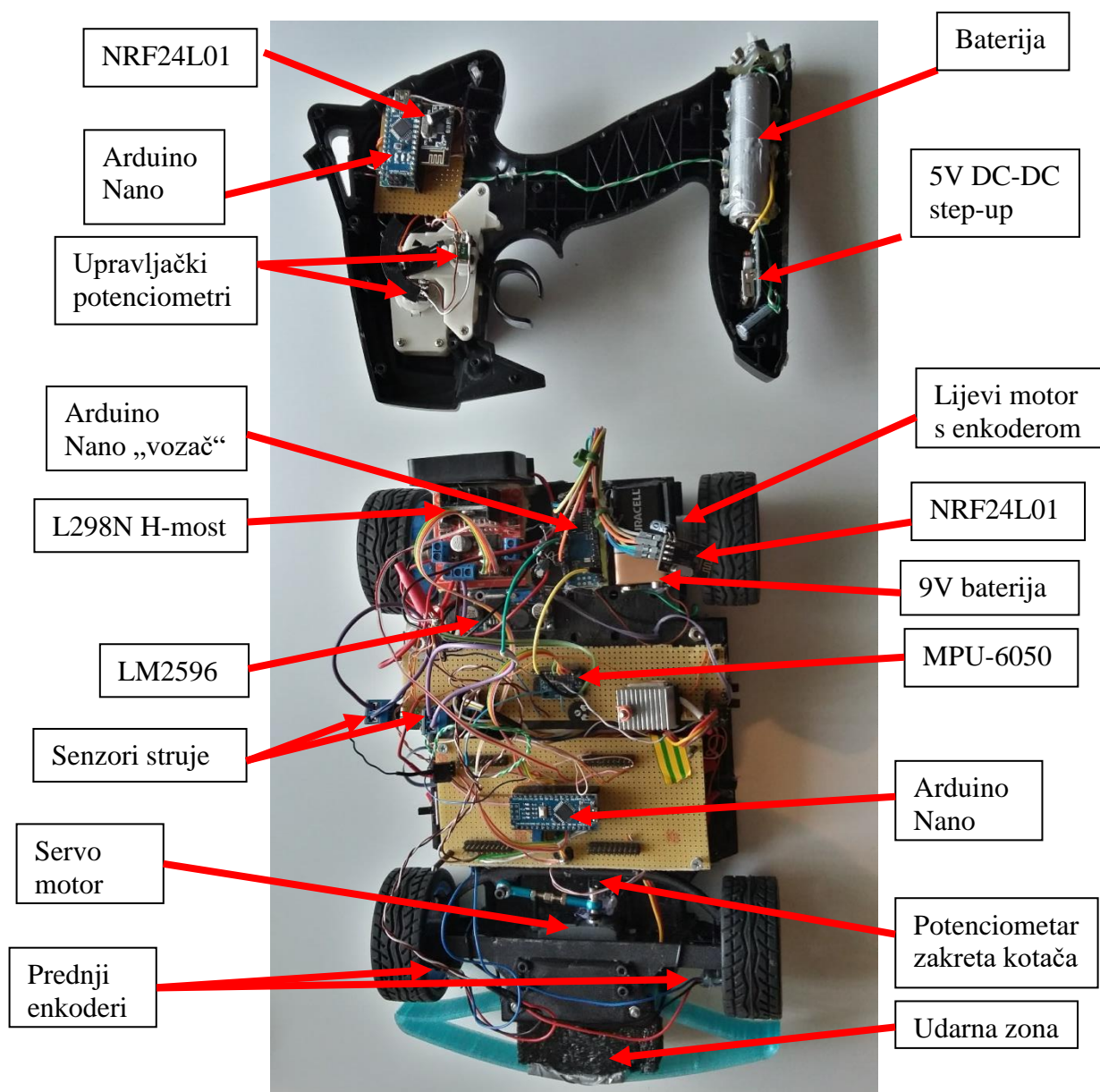
6.3. Električna shema

Na slici niže prikazana je električna shema vozila. Shema je pojednostavljena u svrhu lakšeg razumijevanja ali su prikazani svi ključni dijelovi.

Iz sheme su izostavljene komponente koje nisu ključne za razumijevanje funkcionalnosti, kao što su: prekidači kojima se uključuju i isključuju TCS i TV, potencijometar za fino podešavanje nekog proizvoljnog parametra i konektor punjača.



Slika 55. Električna shema

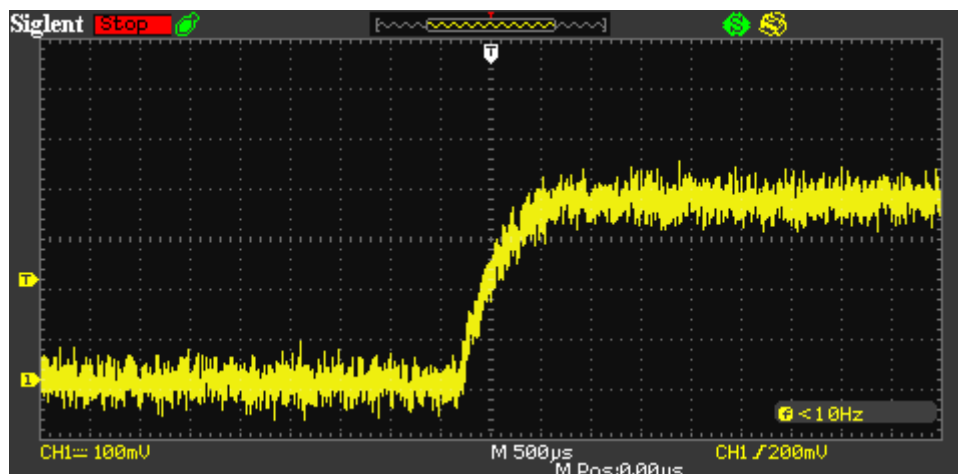


Slika 56. Maketa i daljinski upravljač

6.4. Programiranje

Prikazat će se ključni dijelovi upravljačkog algoritma. Iako je program pisan u razvojnom sučelju za Arduino mikrokontrolere, radi lakšeg razumijevanja njegova funkcionalnost bit će pilustrirana pomoću blokovskih dijagrama.

Zbog znatnog šuma senzora struje regulacija momenta na elektromotorima odrađena je preko poznatih odnosa napona motora, brzine vrtnje i momenta.



Slika 57. Značajan šum senzora struje (0-100% struje)

Iz (12) zaključujemo da ukoliko znamo napon motora i njegovu brzinu vrtnje možemo odrediti moment. Budući da napon motora sami zadajemo, a signal brzine vrtnje je puno manje zašumljen nego signal struje zaključujemo da je ovo kvalitetnija metoda određivanja momenta. Ukoliko konstante za otpor motora i moment zamijenimo jednom nepoznanicom, moment motora je (uz pretpostavku zanemarive dinamike struje armature, vidi poglavlje 3):

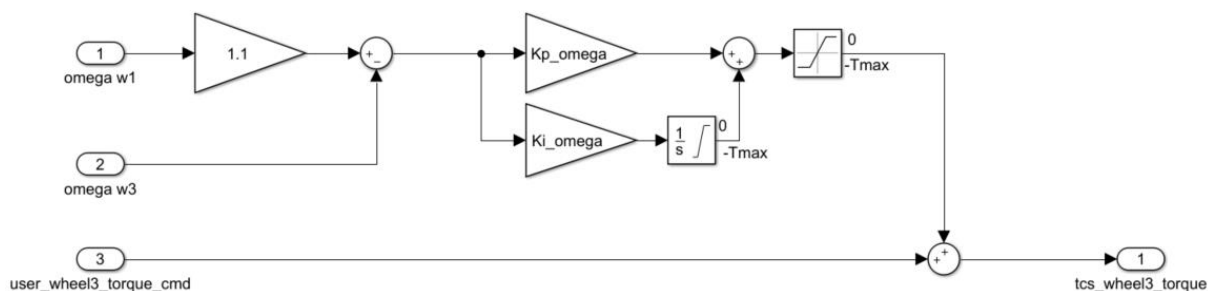
$$M = (U - \omega * K_e) * K_m' \text{ [Nm]} \quad (67)$$

K_m' – zamjenska momentna konstanta $K_m' = K_m / R_a$

Ovu jednu konstantu lako određujemo mjerenjem sile koju motor proizvodi za neku razliku između narinutog i generiranog napona. Određivanje generiranog napona tj. konstante K_e lako je izvedivo mjerenjem ostvarenih broja okretaja za neki napon na motoru kada on nije opterećen. Stoga je za ostvarivanje momenta M potrebno na motoru zadati napon:

$$U = \omega * K_e + \frac{1}{K_m'} * M \quad (68)$$

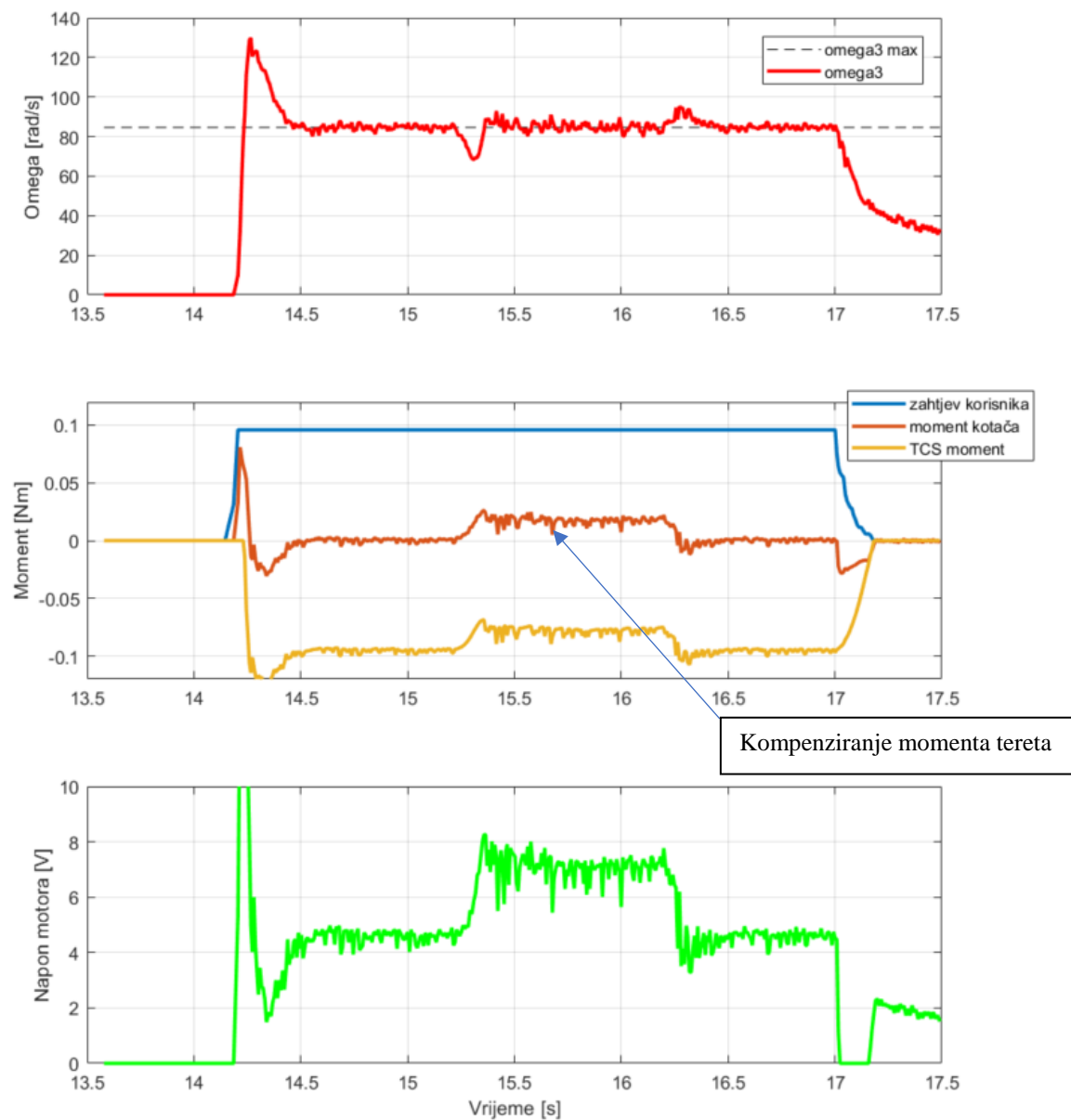
Regulatori za TCS su projektirani u situaciji gdje se konstantno zadaje referenca brzine stražnjih kotača kako bi se ostvarilo maksimalno ubrzanje. Za normalnu svakidašnju primjenu ovo nije način rada TCS-a, već se na kotače zadaje moment koji određuje vozač, a koji se umanjuje u slučaju prevelikog klizanja. Kako bi ostvarili realnu funkcionalnost sustava i zadržali izračunata pojačanja strukturu regulatora smo izmijenili na slijedeće.



Slika 58. Struktura PI regulatora TCS-a na mikrokontroleru

Na ovaj način, kada nema znatnog klizanja korisnikov zahtjev za moment se neometano proslijeđuje kotačima. Ukoliko se brzina stražnjih kotača poveća iznad npr. 10% prednjih PI regulator izračunava moment kojim će umanjiti korisnikov zahtjev.

Ako u prijašnje izračune za Traction control uvrstimo parametre umanjene makete dobivamo $K_{c\omega}=0.000339$ i $T_{c\omega}=0.0643$. Pojačanja na mikrokontroleru izvedena su u drugoj strukturi i drugim jedinicama te ih konvertiramo u $K_p=0.0011$ i $K_i=0.0245$. Parametri regulatora su dodatno fino podešeni. K_i vrijednost je zadovoljavajuća, no K_p je pojačan na $K_p=0.002$. Kako točnost parametara direktno ovisi o točnosti parametara opisanog sustava očekivano je odstupanje iz više razloga. Za kupovne komponente poput motora i kotača informacija o inerciji nije deklarirana budući da su komponente hobi razine gdje te informacije nisu bitne. Stoga su inercije određene iz poznatih ili pretpostavljenih dimenzija i gustoća. Također odstupanja postoje i u mjerenom momentu kotača. Za realnu formulu sve ove veličine su precizno određene. Dobiven je odziv prikazan na slici 59:

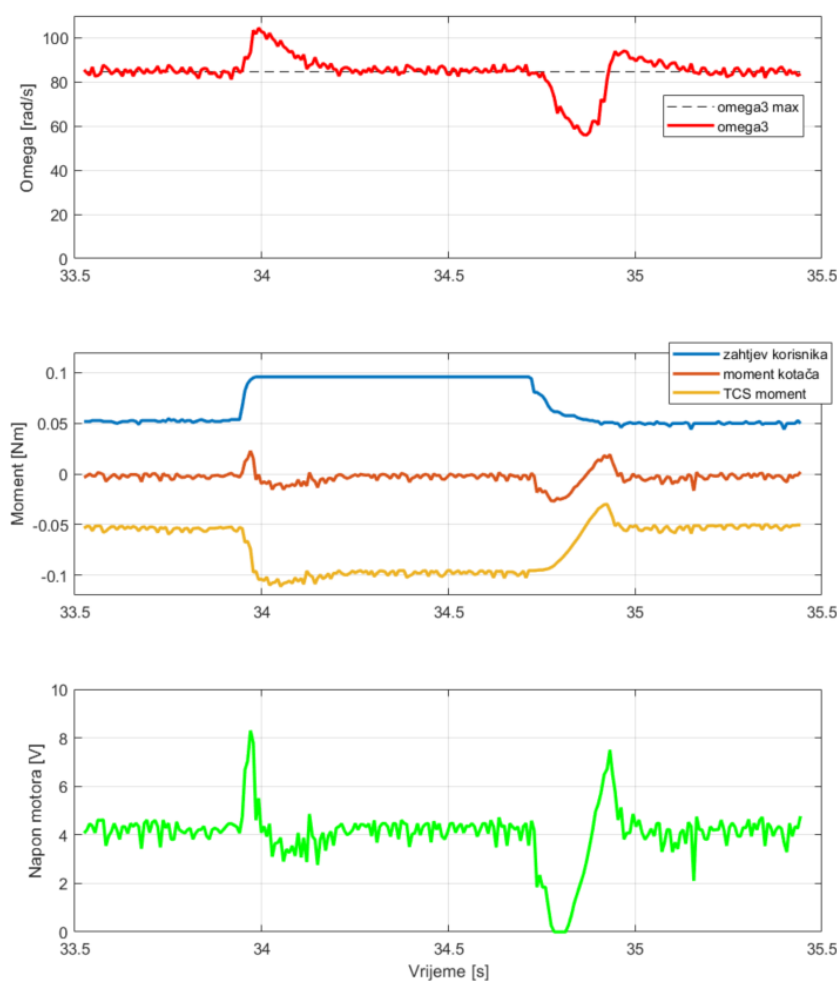


Slika 59. Odziv TCS-a na maketi

Vidimo da regulator solidno održava brzinu kotača i kompenzira moment tereta koji se pojavi nakon prijelazne pojave, no sama prijelazna pojava ima popriličan prebačaj. Razlog je taj što nije dodan prefiltar na referencu te se pojavljuje ranije spomenuti znatni prebačaj. Dodavanjem prefiltara koji bi smanjio prebačaj znatno se produžuje vrijeme uzorkovanja sa 12 ms na 20~30 ms zbog brojnih prebacivanja unutar memorije mikroprocesora i brojnih operacija dijeljenja. Ovakvo usporenje mikroprocesora jednako šteti koliko i sam prefiltar doprinosi prijelaznoj pojavi, a kompenziranje poremećaja lošije. Stoga je odlučeno ne koristiti prefiltar jer promjene

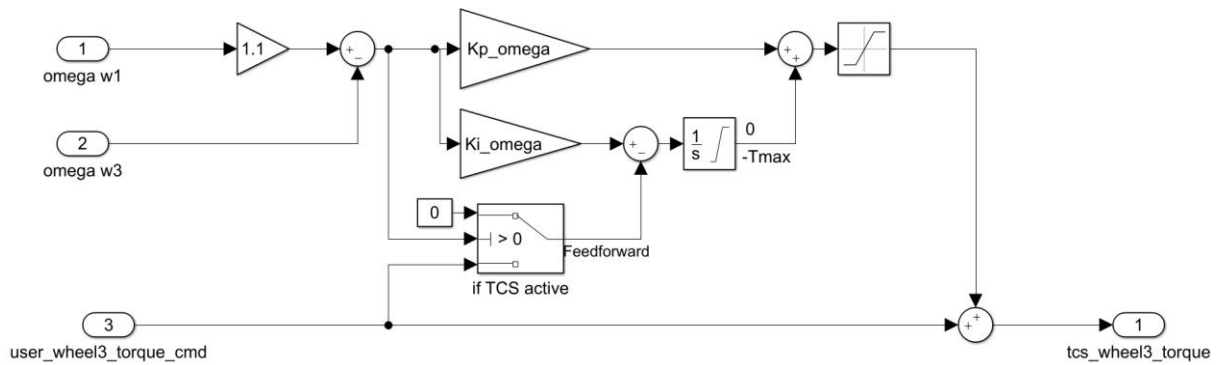
reference, tj brzine prednjih (ne pogonjenih) kotača su ionako relativno spore, odnosno imaju manje utjecaja nego poremećaji, na koje je bitnije brže reagirati.

Promjene korisnikovog zahtjeva prosljeđuju se direktno na izlaz u oba slučaja. Ako je TCS aktivan i korisnik mijenja zahtjev, tj. pritisće ili otpušta „gas“ te promjene regulator mora „ponovno“ kompenzirati. To znači da se korisnikov zahtjev tretira kao poremećaj koji remeti ostvarenu brzinu kotača. Ovo uzrokuje da se npr. za neke dvije konstantne pozicije „gasa“ kada je TCS aktivan i potrebno je smanjivati moment, pri prelasku između tih dvaju stanja dogodi privremeni propad ili porast momenta i brzine.

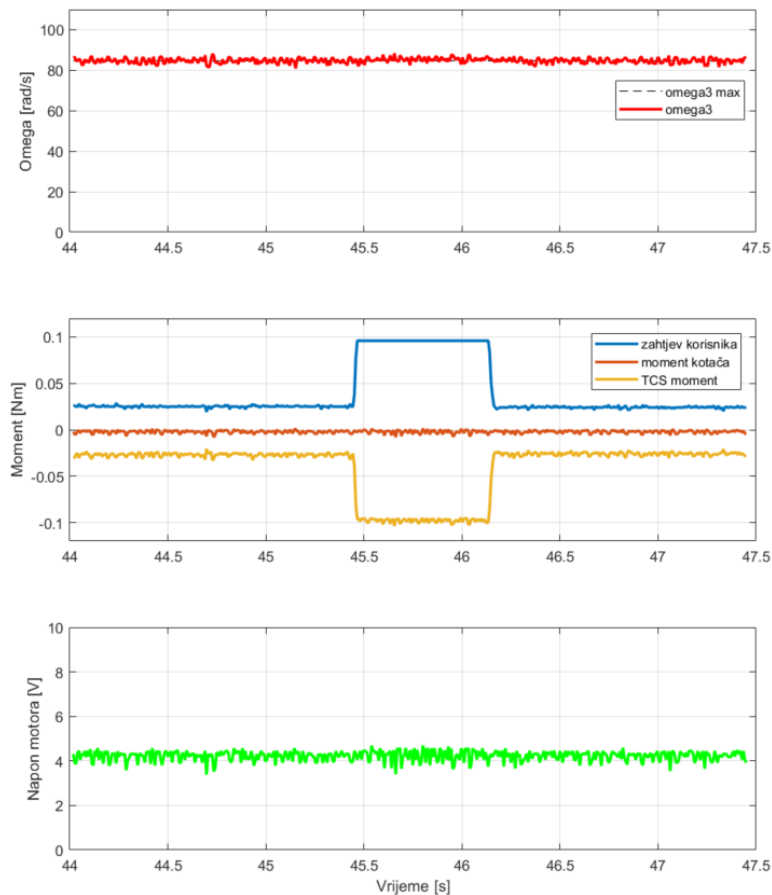


Slika 60. Zahtijevani moment kao poremećaj

Budući da je korisnikov zahtjev poznata informacija, pa tako i njene promjene, poželjno ih je uzeti u obzir u regulatoru kako se ona ne bi morala kompenzirati kao nepoznanica. Ovakav način rada zove se *Feedforward*.



Slika 61. Feedforward programski kod

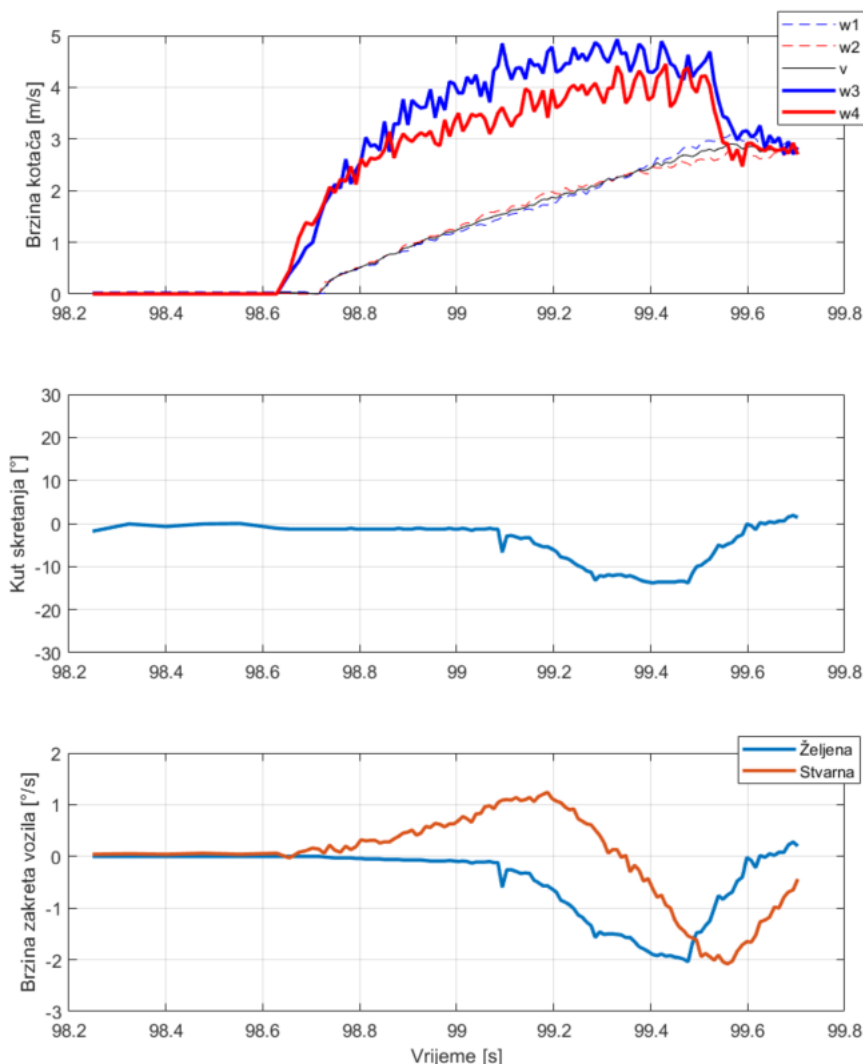


Slika 62. Feedforward mjerenje

Možemo uočiti da pri promjenama zahtjeva (dok smo u području rada TCS-a) ne dolazi do promjena u brzini kotača, a moment TCS-a, kao i izlazni moment na kotač, je stabilan bez oscilacija. *Feedforward* bi se mogao napraviti i za neke ostale poznate promjene, kao npr. transfer težine, čime bi se dodatno poboljšala regulacija.

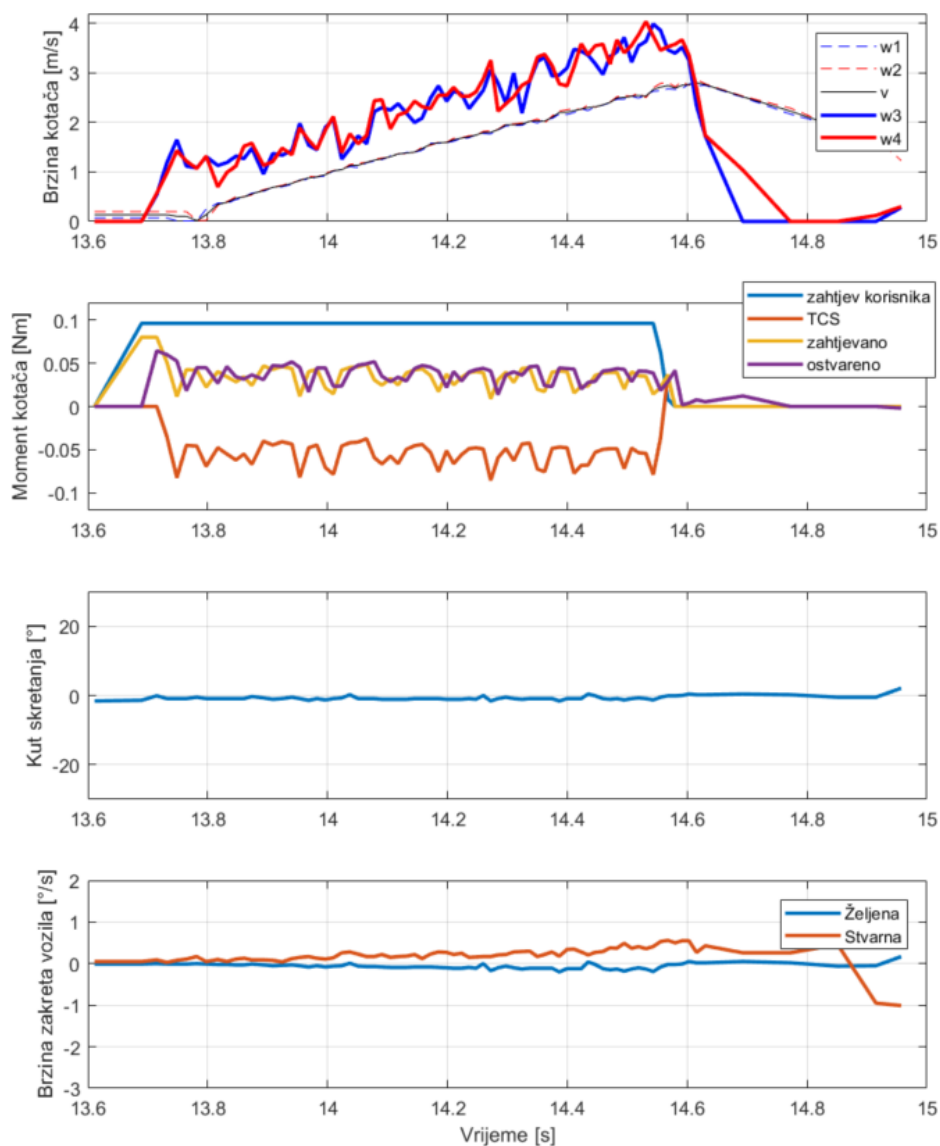
7. Rezultati eksperimentalnih mjerenja

Ubrzanje bez TCS-a:



Slika 63. Ubrzanje bez TCS-a

Vidimo da je brzina stražnjih kotača znatno veća od prednjih. Prosječno vrijeme ubrzanja 0-2.5 m/s je 0.74 s. Uz to, od početka ubrzanja vidimo povećanje brzine zakreta vozila, na koje vozač mora reagirati zakretanjem volana u drugu stranu. Uzrok je, kako je ranije objašnjeno, smanjeno bočno trenje gume kada su uzdužna klizanja velika. Također brzine lijevog i desnog kotača se razlikuju što dodatno pogoršava situaciju. Razlog tome je što motori nemaju identične karakteristike iako bi trebali biti identični. Ovakvo lateralno ponašanje vozila nikako nije poželjno u zavoju jer uzrokuje instantno okretanje/izlijetanje ukoliko za volanom nije iskusni vozač koji dobro poznaje vozilo i može kompenzirati ovakve pojave.



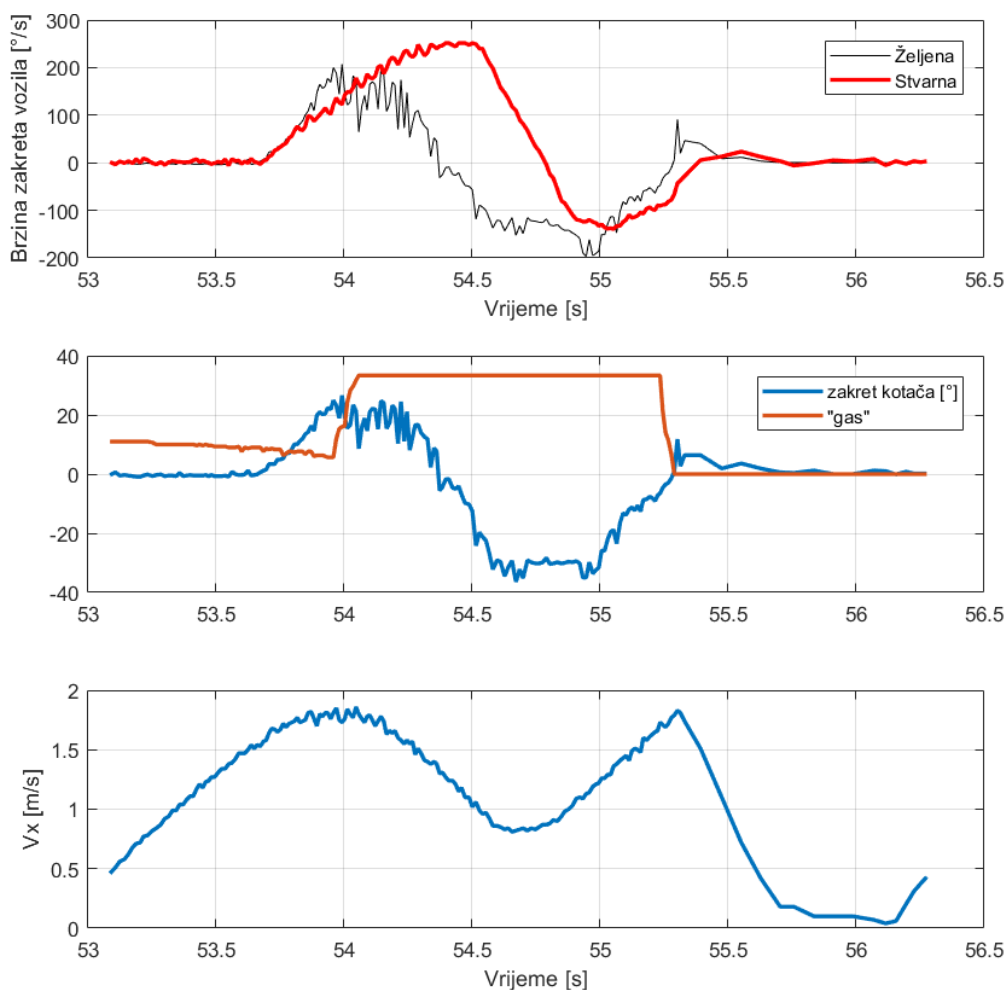
Slika 64. Ubrzanje sa TCS-om

Uočeno je da se sa povećanjem dopuštenog klizanja sa 10 na 30% vremena ubrzanja poboljšavaju. Prosječno vrijeme ubrzanja 0-2.5 m/s je 0.79 s. Dakle nešto lošije nego bez TCS-a. Pretpostavka je da guma na maketi u dodiru s podlogom od kućnog laminata nema istu karakteristiku kao i automobilska guma u dodiru sa asfaltom budući vidimo da se brzina održava na željenoj, a ubrzanje je lošije nego kada klizanje puno veće.

Iako sustav u načelu funkcionira, brzina stražnjih kotača ima popriličan šum i odstupanja. Prvi razlog je inherentni šum enkodera stražnjih kotača. Stražnji enkoder nije disk sa prorezima za optički senzor već je disk magnetni kojem se prisustvo polova očitava hallovim senzorom. Pretpostavka je da ti polovi nisu precizni kao prorezi na optičkom enkoderu, što se da zaključiti

po puno većem šumu signala stražnjih naspram prednjih brzina i u normalnoj vožnji bez ikakvih regulatora. Drugi razlog je sporo vrijeme uzorkovanja zbog korištenja „Pulsein“ funkcije na arduinu za mjerenje trajanja impulsa stražnjih enkodera. Funkcija mora biti korištena na bar dva enkodera jer arduino nano ima samo dva „interrupt“ pina. Ukratko interrupt pinovi omogućuju mjerenje frekvencije dugotrajnih impulsa bez usporavanja rada procesora (nekoliko milisekundi su veoma dugi impulsi za mikroprocesore. *Interrupt* je iskorišten za prednje kotače budući da imaju manji broj impulsa po okretaju kotača tj. duže impulse, i uz to se stražnji vrte nešto brže nego prednji, stoga je zadržavanje procesora puno kraće nego u obrantom slučaju. Funkcija „Pulsein“ zadržava rad procesora onoliko koliko traje jedan period signala. To znači da je pri manjim brzinama to zadržavanje poprilično dugo i smanjuje se sa povećanjem brzine. Kod normalne vožnje pri aktiviranju TCS-a poželjno je jednako smanjiti momente na lijevom i desnom kotaču kako se nebi stvorio Mz koji zakreće vozilo.

Prolazak zavoja bez TV-a:

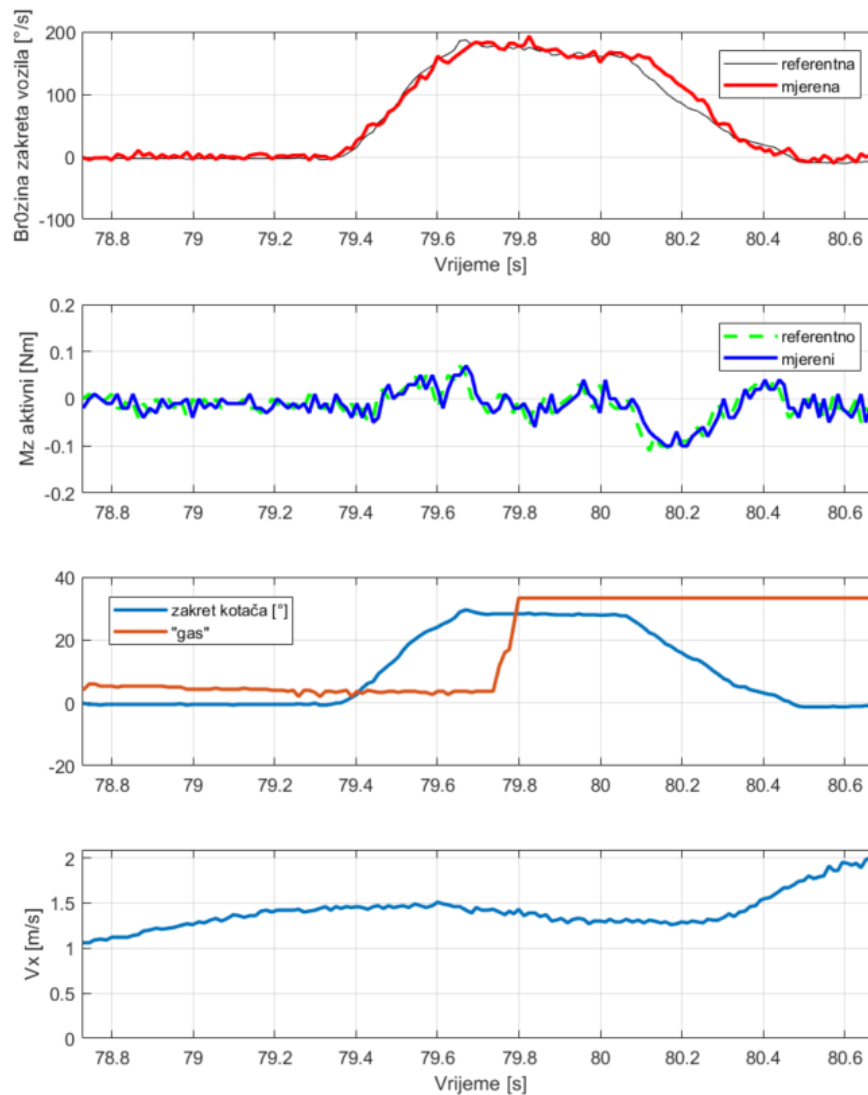


Slika 65. Prolazak makete kroz zavoje TV=off



Slika 66. Snimka prolaza zavoja TV=off

Možemo uočiti znatno odstupanje referentne i stvarne brzine zakreta vozila, što u ovom slučaju ukazuje na znatno preupravljanje.



Slika 67. Prolazak makete kroz zavoje TV=on

Kada je TV uključen možemo uočiti bolje ostvarivanje referentne brzine zakretanja vozila, pa tako i stabilnost vozila tokom prolaska kroz zavoj. Vidimo da se znatni moment oko Z-osi pojavljuje tek pri prelaznim pojavama. Također vidimo da je brzina zakretanja poprilično konstantna i stabilna dok se zakret kotača ne mijenja što je poželjno.



Slika 68. Snimka prolaza zavoja TV=on

Kao i na grafovima, ovdje uočavamo da sa TV-om uključenim nema prekomjernog zanošenja vozila pri kraju zavoja te zaključujemo da svi sustavi rade kako je predviđeno i omogućuju predvidivo i stabilno ponašanje vozila.

8. ZAKLJUČAK

U radu je predstavljen dizajn sustava upravljanja uzdužnom dinamikom vozila i dinamikom skretanja koji se u prvom slučaju sastoje od sustava kontrole vuče (engl. Traction Control System, TCS), odnosno u potonjem sustava razdiobe okretnih momenata na kotačima (engl. Torque Vectoring, TV). Za potrebe ispitivanja navedenih sustava provedeno je detaljno matematičko modeliranje na temelju kojeg su izrađeni simulacijski modeli dinamike vozila. Nadalje, za potrebe eksperimentalnog ispitivanja funkcionalnosti navedenih sustava izrađeno je umanjeno daljinski upravljano eksperimentalno vozilo. Na njemu su savladani problemi koji ne postoje u simulacijama poput diskretnog vremena uzorkovanja, šuma senzora, kašnjenja, te naravno programiranja samog mikrokontrolera i usklađivanja rada svih sustava. Umanjeno vozilo pokazalo je slična poboljšanja u dinamici vozila tako što je sa stajališta vozača vozilo puno upravljivije i smirenije što omogućava lakše usavršavanje putanje kroz zavoje i smanjenje vremena prolaska staze.

Budući rad na ovoj problematici uključit će eksperimentalno ispitivanje navedenih sustava na realnom vozilu Formula Student koje izrađuje FSB Racing Team.

LITERATURA

- [1] <https://www.britannica.com/science/dAlemberts-principle>
- [2] Skalicki, B., Grilec, J., 2011. Električni strojevi i pogoni
- [3] FSB kolegij: „Automobilski mehatronički sustavi“, auditorne vježbe:
- [4] FSB kolegij: „Elektromotorni servo pogoni“, auditorne vježbe 8
- [5] FSB Racing Team, odjel ovjesa + suradnja sa Rimac Automobili d.o.o
- [6] <http://www.ti.com/tool/LAUNCHXL-F28379D>
- [7] <https://store.arduino.cc/arduino-nano>
- [8] <https://electronics.stackexchange.com/questions/52845/generating-servo-signal-in-atmega2560>
- [9] https://en.wikipedia.org/wiki/Ackermann_steering_geometry

POGRAMSKI KOD

```
float max_F=2.5;           //maksimalna sila kotača
float wsf_min = 0.25;     //minimalna brzina prednjih kotača
float allowed_slip = 1.15; //relativno dozvoljeno klizanje
float fixed_slip=0.1;     //apsolutni dodatak klizanja

float Kp_omega = 0.002;   // 0.002 in air eksperimentalno
float Ki_omega = 0.0245;  // 0.01 eksperimentalno

float Mz_Kp=0;
boolean tcs_on = 0 ;
boolean TV_on = 0 ;
float Ki_tL = 5.6*1.25;   //Km' + kompenzacija razlike u motorima
float Ki_tR = 5.6;
float Ke_left = 1.47;     //Ke
float Ke_right= 1.7;

#include <Wire.h>          // I2C komunikacija
long accelX, accelY, accelZ; //varijable giroskopa
float gForceX, gForceY, gForceZ;
long gyroX, gyroY, gyroZ;
float rotX, rotY, rotZ;

int e1p = 6;
int e2p = 5;
int e3p = 11;
int e4p = 12;
```

```
unsigned long e1r = 0;
unsigned long e2r = 0;
unsigned long e3r = 0;
unsigned long e4r = 0;

int MLp = 10;           //pin lijevi motor
int MRp = 9;           //pin desni motor
int THRp = A1;         //pin "gasa"
int STRp = A2;         //pin senzora skretanja
int POTp = A0;         //pin potenciometra za fino podešavanje
int THRr = 0;
float THR = 0;
float BRK = 0;
int POTr = 0;
int Ntf = 20;          //broj zubi prednjeg enkodera
float Ntr = 52.8;      //broj zubi stražnjeg enkodera
float rw = 0.063 / 2;  //radijus kotača
float speed_factor_front=0;
float speed_factor_rear=0;
float ws1 = 0;         //brzina kotača 1
float ws2 = 0;         //brzina kotača 2
float ws3 = 0;         //brzina kotača 3
float ws4 = 0;         //brzina kotača 4
int MLpwm = 128;       //0V na motoru
int MRpwm = 128;       //0V na motoru
int IL = 0;            //struja sa senzora lijevi
int IR = 0;            //struja sa senzora desni
```

```
float TL = 0;           //okretni moment lijevog kotača
float TR = 0;           //okretni moment desnog kotača
float TL_tcs = 0;
float TR_tcs = 0;
float Ubat = 0;         //napon baterije
float UL = 0;           //napon na lijevom motoru
float UR = 0;           //napon na desnom motoru
float EMFL = 0;         //EMF lijevog motora
float EMFR = 0;         //EMF desnog motora
float FL = 0;           //sila lijevog kotača
float FR = 0;           //sila desnog kotača
float FL_tcs = 0;
float FR_tcs = 0;

float deltaUL = 0;
float deltaUR = 0;

float Vx = 0;           //Brzina vozila
float yaw_rate = 0;     //Brzina zakreta
float yaw_rate_target = 0;
float Rt = 0;           //radijus skretanja
float l_c = 0.196;      //trag kotača
float Izz = 0.007;      //Inercija vozila
float steering_angle = 0;
float Mz = 0;
float Mz_feedback=0;

float F_user_cmd=0;
```

float FL_user_cmd = 0;

float FR_user_cmd = 0;

float eFL = 0;

float eFR = 0;

float UL_cmd = 0.0;

float UR_cmd = 0.0;

float ws1_fake = wsf_min;

float ws2_fake = wsf_min;

float omega3_ref = 0;

float omega4_ref = 0;

float omega1 = 0;

float omega2 = 0;

float omega3 = 0;

float omega4 = 0;

float e_omega3 = 0;

float e_omega4 = 0;

float FL_cmd = 0;

float FR_cmd = 0;

float TL_p = 0;

float TL_i = 0;

float TR_p = 0;

float TR_i = 0;

```
int i = 0;
int j = 0;

boolean tcsL = 0;      //TCS aktivan
boolean tcsR = 0;

float tcsL_force_reduction = 0;
float tcsR_force_reduction = 0;
float F_reduction=0;
float FL_user_Feedforward=0;
float FR_user_Feedforward=0;
float FL_user_cmd_z1=0;
float FR_user_cmd_z1=0;
float v_avg=0;

volatile unsigned long int left_time=0;
volatile unsigned long int right_time=0;
volatile unsigned long int left_time_memory=0;
volatile unsigned long int right_time_memory=0;
volatile unsigned long int left_time_period=0;
volatile unsigned long int right_time_period=0;

float timestep = 0;

unsigned long int t = 0;
unsigned long int t1=0;
```

```
void setup() {
  TCCR1B = (TCCR1B & 0b11111000) | 0x01; //31.37255 [kHz] //promjena frekvencije
  PWM-a
  Serial.begin(115200); //UART komunikacija
  // definiranje funkcije ulaza/izlaza
  pinMode(e1p, INPUT);
  pinMode(e2p, INPUT);
  pinMode(e3p, INPUT);
  pinMode(e4p, INPUT);
  pinMode(MLp, OUTPUT);
  pinMode(MRp, OUTPUT);
  pinMode(THRp, INPUT);
  pinMode(STRp, INPUT);
  pinMode(POTp, INPUT);
  pinMode(7, INPUT_PULLUP); // tcs on
  pinMode(8, INPUT_PULLUP); // TV on

  Wire.begin(); //setupMPU(); //giroskop
  Wire.beginTransmission(0b1101000); //This is the I2C address of the MPU
  (b1101000/b1101001 for AC0 low/high datasheet sec. 9.2)
  Wire.write(0x6B); //Accessing the register 6B - Power Management (Sec. 4.28)
  Wire.write(0b00000000); //Setting SLEEP register to 0. (Required; see Note on p. 9)
  Wire.endTransmission();

  pinMode(2, INPUT); //interupti premdjih enkodera
  pinMode(3, INPUT);
  attachInterrupt(digitalPinToInterrupt(3), left_event, RISING);
  attachInterrupt(digitalPinToInterrupt(2), right_event, RISING);
}
```

```
if(digitalRead(7)==LOW)          //TCS ukljucen
{
    tcs_on=1;
    digitalWrite(13, HIGH);
    delay(1000);
    digitalWrite(13, LOW);
    delay(200);
}
if(digitalRead(8)==LOW)          //TV ukljucen
{
    TV_on=1;
    digitalWrite(13, HIGH);
    delay(200);
    digitalWrite(13, LOW);
    delay(200);
    digitalWrite(13, HIGH);
    delay(200);
    digitalWrite(13, LOW);
    delay(200);
}

speed_factor_front=1000000*(1./Ntf)*2.*rw*3.14; //faktor za izračun brzine stražnjih kotača
speed_factor_rear=1000000*(1./Ntr)*2.*rw*3.14;
Ki_omega = Ki_omega/1000000.;
}

//===== MAIN LOOP =====

void loop()
{
```

```
read_speed();          //očitanje brzine vozila
timestep = micros() - t;
t = micros();          //vrijeme uzorkovanja
read_driver_input();   //korisnikovi ulazi
read_yaw();            //očitanje brzine zakreta
calc_yaw_rate_desired(); //izračun referentne brzine zakreta
calc_Mz_cmd();         //izračun potrebnog Mz
read_torque();         //očitanje momenta kotača
tcs();
POTr = analogRead(POTp); //potencijometar ta fino podešavanje
Mz_Kp=(POTr/1024.)*50.;
i++;
}
//=====

void read_speed()
{
  unsigned long timeout = 50000;
  //izmjena redosljeda čitanja brzine stražnjih kotača
  //radi eliminacije kašnjenja stalno istoga signala
  if(i%2==0)
  {
    e3r = pulseIn(e3p, HIGH, timeout);
    e4r = pulseIn(e4p, HIGH, timeout);
  }
  else
  {
    e4r = pulseIn(e4p, HIGH, timeout);
```



```
e3r = pulseIn(e3p, HIGH, timeout);
}

e1r=left_time_period;
e2r=right_time_period;

//izračun brzine iz duljine impulsa
if (e1r == 0) ws1 = 0;
else if (e1r > 600) ws1 =speed_factor_front/e1r ;
ws1_fake = constrain(ws1, wsf_min, 100);

if (e2r == 0) ws2 = 0;
else if (e2r > 600) ws2 =speed_factor_front/e2r;
ws2_fake = constrain(ws2, wsf_min, 100);

if (e3r == 0) ws3 = 0;
else ws3 = speed_factor_rear/(e3r*2);

if (e4r == 0) ws4 = 0;
else ws4 = speed_factor_rear/(e4r*2);

omega1 = ws1 / rw;
omega2 = ws2 / rw;
omega3 = ws3 / rw;
omega4 = ws4 / rw;
v_avg=(ws1+ws2)/2.;
}
```

```
void read_driver_input()
{
  THRr = analogRead(THRp);
  steering_angle = analogRead(STRp)*10.;
  steering_angle = map(steering_angle, 3900, 6900, 280, -280)/10.; //mapiranje u kut zakreta
kotača
  THR = map(THRr, 440, 770, 0, 100);    //mapiranje gasa
  THR = constrain(THR, 0, 100);
  BRK = map(THRr, 250, 380, 100, 0);
  BRK = constrain(BRK, 0, 100);
}

void read_yaw()
{
  Wire.beginTransmission(0b1101000); //I2C address of the MPU
  Wire.write(0x43); //Starting register for Gyro Readings
  Wire.endTransmission();
  Wire.requestFrom(0b1101000, 6); //Request Gyro Registers (43 - 48)
  while (Wire.available() < 6);
  gyroX = Wire.read() << 8 | Wire.read(); //Store first two bytes into accelX
  gyroY = Wire.read() << 8 | Wire.read(); //Store middle two bytes into accelY
  gyroZ = Wire.read() << 8 | Wire.read(); //Store last two bytes into accelZ
  rotX = gyroX / 131.0;
  rotY = gyroY / 131.0;
  rotZ = gyroZ / 131.0 + 1.3;
}

void read_torque()
```

```
{
  Ubat = analogRead(A3) / 1023.*5.*4.;
  Ubat = constrain(Ubat, 12.1, 18.1);
  UL = (MLpwm-135)/110. *(Ubat-2); //za pwm 135 napon na motoru je 0V
  UR = (MRpwm-135)/110. *(Ubat-2);
  EMFL = ws3 * Ke_left;
  EMFR = ws4 * Ke_right;
  deltaUL = UL - EMFL;
  FL = deltaUL * (1.25/7)/1.25 ;
  TL = FL * rw;
  deltaUR = UR - EMFR;
  FR = deltaUR * (1.25/7);
  TR = FR * rw;
  Mz_feedback=(-FL+FR)*(track/2);
}

void read_current()
{
  IL = analogRead(A7);
  IR = analogRead(A6);
}

void calc_yaw_rate_desired()
{
  Vx = (ws1 + ws2) / 2.;
  if (tcs == 1) //računanje brzine pomoću svih kotača ako je TCS uključen - manji šum
  {
    Vx = (ws1 + ws2 + ws3 + ws4) / 4.;
```

```

}
Rt = tan((-steering_angle - 90) * ((2 * 3.14) / 360)) * l_c; //izračun radijusa skretanja
yaw_rate_target = (Vx / Rt)*0.9 *1; // 0.5 is neutral when last factor i *1 car is neutral
yaw_rate_target = yaw_rate_target*(1-(Vx/12. )); //3 is 0, 6 is 0.5, 12 is 0.75
}

```

```
void calc_Mz_cmd()
```

```

{
yaw_rate = (rotZ / 360.) * 2 * 3.14;
Mz = (yaw_rate_target - yaw_rate) * Izz * Mz_Kp; //POTr/20.;
Mz = constrain(Mz, -0.3, 0.3);
}

```

```
void tcs()
```

```

{
F_user_cmd= THR / 100 * max_F;
FL_user_cmd = THR / 100 * max_F; //- BRK / 2.;
FR_user_cmd = THR / 100 * max_F; //- BRK / 2.;
if(TV_on==1) //dodavanje Mz na korisnikov ulaz
{
FL_user_cmd=FL_user_cmd-Mz/track;
FR_user_cmd=FR_user_cmd+Mz/track;
}
}

```

```
if (tcs_on == 1)
```

```

{
omega3_ref = (ws1_fake * speed_ref_for_tcs +fixed_slip) / rw ;
e_omega3 = omega3_ref - omega3; //računanje odstupanja
}

```

```
TL_p = e_omega3 * Kp_omega;          // P regulator
TL_p = constrain(TL_p, -max_F*rw,0); // P limit
TL_i = e_omega3 * Ki_omega*timestep + TL_i; // I regulator
TL_i = constrain(TL_i, -max_F*rw,0); // I limit
TL_tcs = TL_p + TL_i;

FL_user_Feedforward=FL_user_cmd-FL_user_cmd_z1; //derivacija korinikovog ulaza za
feedforward

FL_user_cmd_z1=FL_user_cmd;
if (TL_tcs<0) tcsL = 1;
else tcsL=0;

TL_i=TL_i-FL_user_Feedforward*rw*tcsL; //feedforward ako je tcs aktivan
TL_tcs= TL_p + TL_i;
FL_tcs = TL_tcs/rw;
}
else FL_tcs=0;

if (tcs_on == 1)
{
omega4_ref = (ws2_fake * speed_ref_for_tcs +fixed_slip) / rw ;
e_omega4 = omega4_ref - omega4;
TR_p = e_omega4 * Kp_omega;
TR_p = constrain(TR_p, -max_F*rw,0);
TR_i = e_omega4 * Ki_omega*timestep + TR_i;
TR_i = constrain(TR_i, -max_F*rw,0);
TR_tcs = TR_p + TR_i;
FR_user_Feedforward=FR_user_cmd-FR_user_cmd_z1;
FR_user_cmd_z1=FR_user_cmd;
if (TR_tcs<0) tcsR = 1;
```

```
else tcsR=0;
TR_i=TR_i-FR_user_Feedforward*rw*tcsR;
TR_tcs = TR_p + TR_i;
FR_tcs = TR_tcs / rw;
}
else FR_tcs=0;
```

// eliminiranje stvaranja dodatnog Mz zbog tcs-a jednakim reduciranjem momenta na oba kotača

```
F_reduction=min(FL_tcs, FR_tcs);
FL_cmd = FL_user_cmd+F_reduction;
FR_cmd = FR_user_cmd+F_reduction;

eFL = FL_cmd - FL;
eFR = FR_cmd - FR;
UL_cmd = UL + Ki_tL * eFL; //određivanje potrebnog napona za zadani moment
UR_cmd = UR + Ki_tR * eFR;

// mapiranje napona za pwm naredbu
MLpwm = map(UL_cmd * 100, 0, (Ubat-2)* 100, 13500, 25500) / 100.;
MLpwm = constrain(MLpwm, 135, 255);
MRpwm = map(UR_cmd * 100, 0, (Ubat-2)* 100, 13500, 25500) / 100;
MRpwm = constrain(MRpwm, 135, 255);
analogWrite(MLp, MLpwm); // zadavanje napona na motore
analogWrite(MRp, MRpwm);
if (BRK > 25) //kočenje i nazad
{
analogWrite(MLp, 128 - BRK/1.5);
analogWrite(MRp, 128 - BRK/1.5);
```

```
}  
}  
  
void left_event() // interupt lijevog enkodera  
{  
    left_time=micros();  
    left_time_period=left_time-left_time_memory;  
    left_time_memory=left_time;  
}  
  
void right_event() //interrupt desnog enkodera  
{  
    right_time=micros();  
    right_time_period=right_time-right_time_memory;  
    right_time_memory=right_time;  
}
```