

Najveća hvala mojim roditeljima, Gordani i Zoranu, na neumornoj podršci duž cijelog mog akademskog puta.

Hvala mentoru, izv. prof. dr. sc. Šandoru Ilešu na njegovom stručnom vodstvu, strpljenju i inspiraciji tijekom diplomskog studija.

Također, hvala dr. sc. Edinu Koči na pruženoj prilici izrade diplomskog rada u Gideonu te ostatku inženjerskog tima, Stojanu, Jerku, Franji, Davoru, Anti i Vedranu, na njihovoj suradnji i savjetovanju.

I na kraju, hvala svim kolegama koji su zadnjih pet godina pretvorili u jedno nezamjepljivo i nezaboravno iskustvo.

SADRŽAJ

1. Uvod	1
2. Model električnog viličara	2
2.1. Upravljive osi viličara	2
2.2. Princip rada električnog hidrauličnog viličara	3
2.3. Tehničke specifikacije viličara	4
2.3.1. Električni motori	4
2.3.2. Frekvencijski pretvarači	5
3. Pregled korištene opreme	6
3.1. Programabilni logički kontroler (PLC)	6
3.1.1. CX5140 središnja procesorska jedinica (CPU)	7
3.1.2. EL6910 sigurnosna procesorska jedinica	7
3.1.3. Moduli	8
3.1.3.1. EL1008 digitalni ulazi	8
3.1.3.2. EL2008 digitalni izlazi	8
3.1.3.3. EL3004 analogni ulazi	9
3.1.3.4. EL3692 mjerjenje otpora	9
3.1.3.5. EL1918 sigurnosni digitalni ulazi	9
3.1.3.6. EL2904 sigurnosni digitalni izlazi	10
3.1.3.7. EL9410 dodatno napajanje za sabirnicu	10
3.1.3.8. EL5021-0090 sučelje za SinCos enkoder	11
3.1.3.9. EL5001-0090 sučelje za SSI enkoder	11
3.1.3.10. EL6751 CANopen komunikacijsko sučelje	12
3.2. Senzori	12
3.2.1. Enkoderi	15
3.2.2. Laserski skeneri	16
4. Pregled korištenih komunikacijskih protokola	17
4.1. CAN protokol	18

4.1.1. CANopen protokol	18
4.2. EtherCAT protokol	19
4.2.1. Sigurnost preko EtherCAT-a (FSoE)	20
5. Proračuni za upravljanje sustavom trakcije viličara	22
5.1. Proračun raspodjele brzine po pogonskim kotačima	22
5.2. Proračun ekvivalentnog kuta virtualnog kotača	26
6. Postavljanje i konfiguracija uređaja	28
6.1. Postavljanje frekvencijskih pretvarača	28
6.2. Postavljanje laserskih skenera	29
7. Implementacija upravljačke logike na PLC-u	31
7.1. Dodavanje uređaja u PLC projekt	31
7.2. Sigurnosni PLC projekt	33
7.2.1. Praćenje stanja sigurnosnih tipkala	33
7.2.2. Praćenje brzine robota	36
7.2.3. Određivanje aktivne sigurnosne zone	39
7.2.4. Sigurnosno zaustavljanje robota	41
7.3. Standardni PLC projekt	43
7.3.1. Upravljanje sustavom trakcije	43
7.3.2. Upravljanje osima za manevriranje vilicama	44
7.3.2.1. Otvorena petlja	44
7.3.2.2. Zatvorena petlja	47
8. Zaključak	53
Literatura	54

1. Uvod

Električni viličari imaju široku primjenu u industrijskim okruženjima već nekoliko desetljeća i predstavljaju neizostavni dio svakog skladišta. Iako su električni viličari napravili svojevrsnu revoluciju u industriji logistike i skladištenja danas se već smatraju starom tehnologijom koja se nije bitno mijenjala iz godine u godinu. Iz tog su razloga napredci u području automatike i robotike potaknuli razna istraživanja i razvojne napore u automatiziranju ovih strojeva kako više nebi bili ograničeni time da ih mora upravljati ljudska ruka.

Cilj ovog diplomskog rada je projektirati i implementirati niskorazinski upravljački sustav integracijom programabilnog logičkog kontrolera na platformu autonomnog električnog viličara tvrtke Gideon. Upravljački sustav mora moći precizno upravljati kretnjama viličara te poslužiti kao osnova za daljnju integraciju visokorazinskog upravljačkog sustava koji čini viličar autonomnim.

Programabilni logički kontroleri su središnja upravljačka jedinica u industrijskoj automatizaciji koji omogućuju upravljanje, nadzor i koordinaciju različitih dijelova sustava. U kontekstu upravljanja električnim viličarom, programabilni logički kontroleri pružaju mogućnost preciznog upravljanja pokretima, nadzor sigurnosnih funkcionalnosti i povezivanje s drugim perifernim uređajima.

U narednim poglavljima opisan je model električnog viličara s pet stupnjeva slobode. Napravljen je pregled korištene opreme za implementaciju upravljačkog sustava. Opisani su korišteni komunikacijski protokoli za povezivanje uređaja. Detaljno je prikazana hardverska arhitektura niskorazinskog upravljačkog sustava. Analizirani su isprogramirani kontrolni algoritmi koji upravljaju osima viličara te su opisane metode praćenja okoline i stanja vozila kako bi se osigurala sigurnost i preciznost kod upravljanja.

Ovim radom nastoji se doprinijeti razumijevanju mogućnosti i prednosti korištenja programabilnih logičkih kontrolera za upravljanje električnim viličarima te pružiti smjernice za daljnji razvoj i primjenu ove tehnologije u području mobilne robotike.

2. Model električnog viličara

U ovom poglavlju analizirane su karakteristike modela električnog viličara koji je predmet ovog rada, uključujući pregled sastavnih dijelova viličara, strukturu upravljivih osi, princip rada ovakvog viličara te njegove tehničke specifikacije.

Na slici 2.1 prikazan je Trey, autonomni viličar tvrtke Gideon.

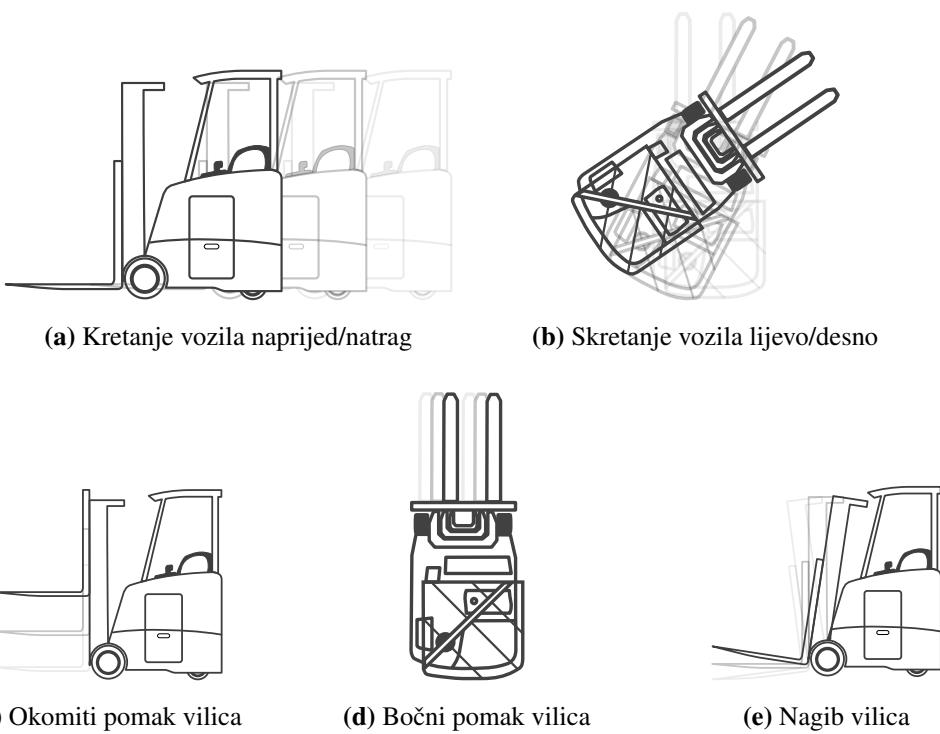


Slika 2.1: Trey - Autonomni viličar tvrtke Gideon

Radi se o standardnoj platformi električnog viličara kojega tvrtka Gideon nastoji nadograditi i po potrebi prilagoditi za autonomni rad.

2.1. Upravljive osi viličara

Pet stupnjeva slobode ili osi električnog viličara odnosi se na pet različitih smjerova u kojima se viličarom može upravljati. Prve dvije osi odnose se na kretanje vozila po podlozi, a preostale tri na pomak vilica. Grafički prikaz upravljivih osi predviđen je slikom 2.2, a one su kretanje vozila naprijed/natrag i skretanje vozila lijevo/desno te podizanje/spuštanje vilica, bočni pomak vilica i nagib vilica.



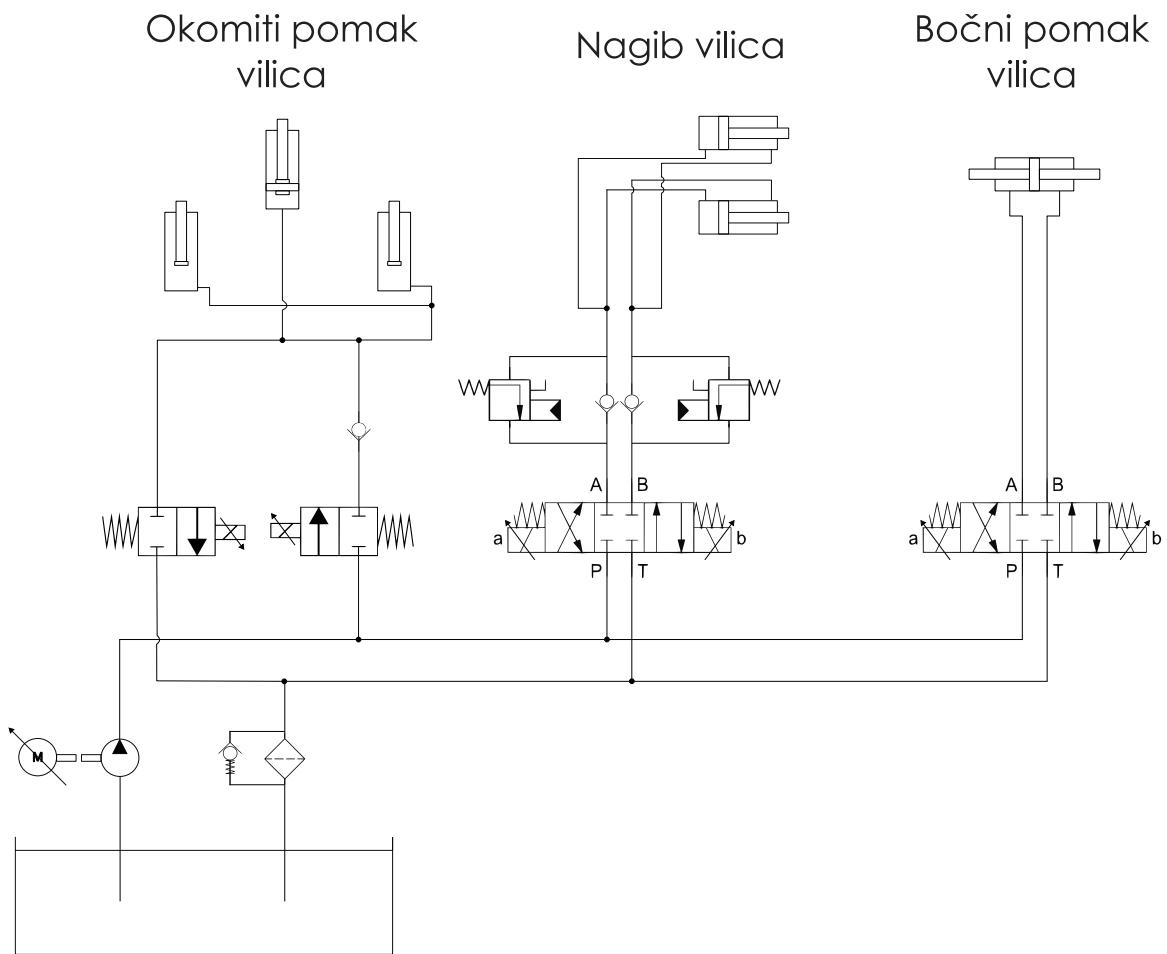
Slika 2.2: Stupnjevi slobode električnog viličara

2.2. Princip rada električnog hidrauličnog viličara

Upravljački sustav viličara može se podijeliti na električni i hidraulični dio. Električni sustav upravlja kretanjem vozila po podlozi, odnosno osima prikazanim slikama 2.2a i 2.2b, dok hidraulični sustav upravlja mehanizmom za manipulaciju vilicama, odnosno osima prikazanim slikama 2.2c, 2.2d i 2.2e..

U središtu hidrauličnog sustava nalazi se hidraulična pumpa koju pogoni električni motor. Pumpa stvara hidraulični tlak guranjem hidraulične tekućine, odnosno ulja u sustav. Ulje se pohranjuje u spremniku i cirkulira kroz sustav te djeluje kao medij za prijenos sile i energije. Hidraulični sustav ima više upravljačkih ventila koji usmjeravaju protok ulja prema određenim aktuatorima, odnosno cilindrima u osima viličara. Hidraulični cilindri pretvaraju hidraulični tlak u mehaničku silu te time pokreću osi viličara u kojima se pomiču vilice.

Slikom 2.3 prikazana je pojednostavljena hidraulična shema. Na dnu sheme nalazi se spremnik tekućine i električni motor, odnosno hidraulična pumpa. Iznad se nalaze upravljački i pomoćni ventili te na samom vrhu hidraulični cilindri za okomiti i bočni pomak te nagib vilica.



Slika 2.3: Pojednostavljenja hidraulična shema Jungheinrich ETG 216 viličara

2.3. Tehničke specifikacije viličara

2.3.1. Električni motori

Na viličaru se nalaze četiri trofazna asinkrona električna motora (jedan po pogonskom kotaču, jedan za okretni kotač te jedan za hidrauličnu pumpu).

Oba pogonska kotača upravlja zasebni električni motor s podacima prikazanim tablicom 2.1. Kotači su na osovinu motora spojeni preko reduktora s omjerom $i = 25.74$, što znači da

Tablica 2.1: Natpisna pločica pogonskih motora

JUNGHEINRICH		
Type AF 4H2-N1	I/N 10110530	
$\Delta 3x25$ V	122 A	2485 min^{-1}
4.0 kW		S2-60 min 85 Hz
IP00/IP30	Ins. Cl. H	EN 1175

je ulazna brzina vrtnje motora približno 26 puta veća od izlazne brzine vrtnje kotača.

Podaci motora koji upravlja okretnim kotačem prikazani su tablicom 2.2.

Tablica 2.2: Natpisna pločica motora okretnog kotača

KORDEL® K		
-		$\cos\varphi 0.8$
$\Delta 3x28 \text{ V}$	29 A	2300 min^{-1}
0.7 kW	S2-15 min 81 Hz	
IP00/IP20		Ins. Cl. F

Okretni kotač također je preko reduktora spojen na električni motor. Prijenosni omjer reduktora iznosi $i = 154.33$, a maksimalni zakret kotača je mehanički limitiran na $\pm 90^\circ$ obzirom na pravocrtno gibanje viličara.

Podaci električnog motora koji pogoni hidrauličnu pumpu prikazani su u tablici 2.3.

Tablica 2.3: Natpisna pločica hidraulične pumpe

JULI		
Type AP 4H6-R1-2		I/N 1012045
$\Delta 3x28 \text{ V}$	353 A	2190 min^{-1}
13 kW	S3-20% 75 Hz	
IP00/IP54	Ins. Cl. H	EN 1175

2.3.2. Frekvencijski pretvarači

Svakim električnim motorom upravlja zasebni frekvencijski pretvarač. Pretvarači su iz porodice frekvencijskih pretvarača tvrtke Curtis Instruments namijenjenim za upravljanjem trofaznim asinkronim električnim motorima.

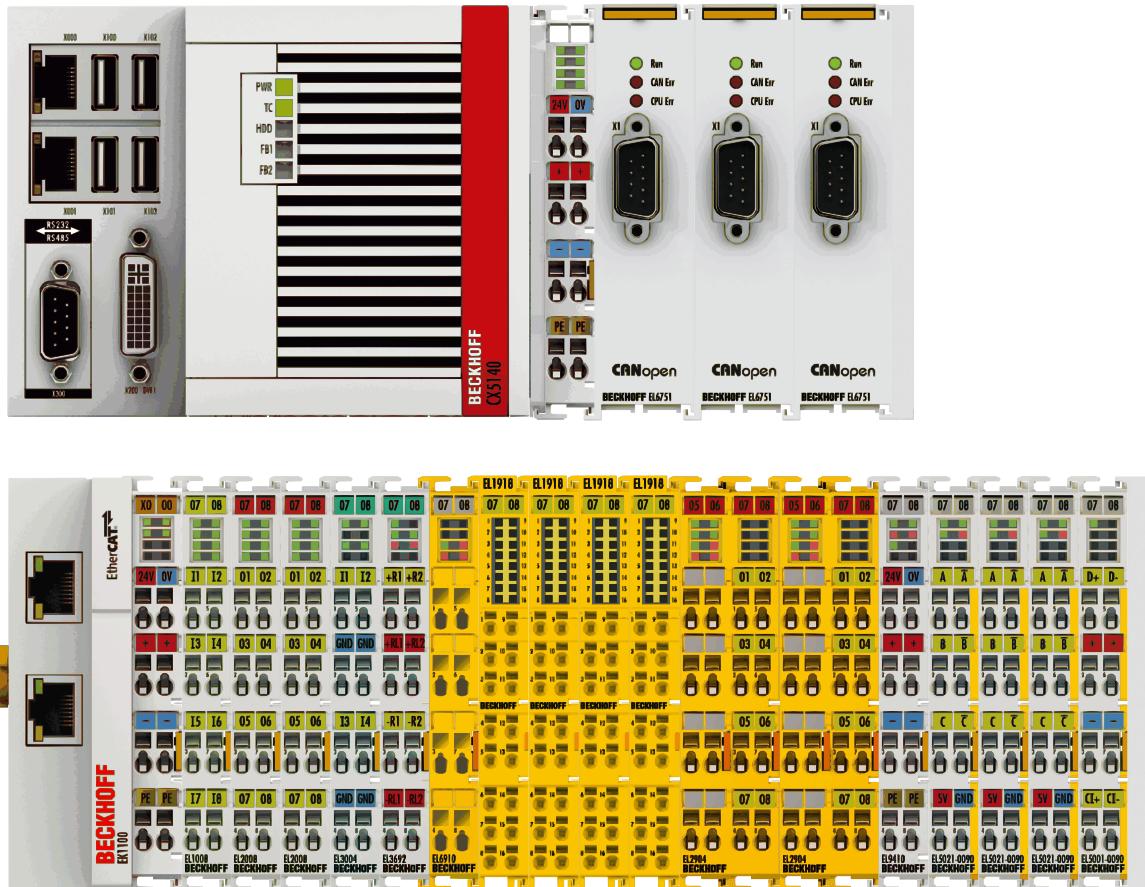
Pogonskim motorima i hidrauličnom pumpom upravlja pretvarač Curtis 1234 SE, a okretnim kotačem upravlja pretvarač Curtis 1222. Uz frekvencijske pretvarače koristi se još Curtis 1353 modul s PWM (eng. *Pulse Width Modulation*) izlazima za regulaciju otvorenosti ventila. Svi pretvarači originalno su povezani na istu komunikacijsku mrežu preko CANopen protokola, zajedno s dijagnostičkim ekranom i komandnom palicom pomoću koje se upravlja vozilom.

3. Pregled korištenе opreme

U ovom poglavlju napravljen je pregled i kratak opis potrebne opreme koja je integrirana na platformu viličara kako bi se mogao implementirati niskorazinski upravljački sustav koji daje služi kao osnova za upravljanje vozilom u autonomnom režimu rada. To uključuje programabilni logički kontroler te razne sigurnosne i ostale senzore koji su potrebni za upravljanje osima viličara u zatvorenoj petlji te stalno praćenje okoline i stanja samog vozila.

3.1. Programabilni logički kontroler (PLC)

Na slici 3.1 prikazan je korišteni programabilni logički kontroler.



Slika 3.1: Korišteni programabilni logički kontroler

Pod pojmom programabilnog logičkog kontrolera (PLC, eng. *Programmable Logic Controller*) podrazumijeva se središnja procesorska jedinica (CPU, eng. *Central Processing Unit*) zadužena za izvršavanje logičkog programa i skup raznovrsnih ulaznih i izlaznih modula koji omogućavaju prikupljanje i obradu signala iz okoline te komunikacijskih modula koji omogućavaju PLC-u da komunicira s drugim uređajima ili sustavima.

Procesorska jedinica i moduli raspodijeljeni su na dvije sabirnice radi ograničenosti prostora unutar upravljačkog ormara na vozilu. Moduli na drugoj sabirnici povezani su s procesorskom jedinicom preko PLC spojnice.

3.1.1. CX5140 središnja procesorska jedinica (CPU)

Za središnju procesorsku jedinicu korišteno je Beckhoff CX5140 ugradbeno računalo koje je zajedno s nekim od važnijih značajki prikazano tablicom 3.1.

Tablica 3.1: Beckhoff CX5140 ugradbeno računalo

Važnije značajke	
<ul style="list-style-type: none">Četverojezreni Intel Atom procesorWindows 10 IoT operacijski sustavEthernet, EtherCAT i USB komunikacijsko sučelje	

3.1.2. EL6910 sigurnosna procesorska jedinica

Sigurnosna procesorska jedinica služi za izvršavanje sigurnosnog logičkog programa koji se izvodi paralelno uz glavni program na središnjoj upravljačkoj jedinici. Potreba za zasebnom jedinicom na kojoj se izvodi sigurnosni program u svijetu automatizacije proizlazi iz raznih sigurnosnih standarda i certifikacija koje sustav mora zadovoljiti. Sigurnosni programi moraju imati kraće trajanje izvođenja jednog programskega ciklusa kako bi se osigurala sigurnost ljudi, opreme i infrastrukture u industrijskim okruženjima. Uz to, odvajanje sigurnosnog programa na odvojenu upravljačku jedinicu olakšava održavanje i dijagnostiku sustava.

Kao sigurnosna procesorska jedinica korišten je Beckhoff EL6910 sigurnosni kontroler čiji su podaci prikazani u tablici 3.2.

Tablica 3.2: Beckhoff EL6910 sigurnosna upravljačka procesna jedinica

Važnije značajke	
<ul style="list-style-type: none">Komunikacija putem TwinSAFE (EtherCAT/FSoE) protokolaVrijeme jednog ciklusa ≈ 1 ms	

3.1.3. Moduli

3.1.3.1. EL1008 digitalni ulazi

Za prikupljanje običnih digitalnih signala poput stanja prekidača, korišten je Beckhoff EL1008 modul s digitalnim ulazima čije su karakteristike prikazane tablicom 3.3.

Tablica 3.3: Beckhoff EL1008 modul s digitalnim ulazima

Važnije značajke	
<ul style="list-style-type: none">24 V DC8 digitalnih ulazaUlagni filter od 3 ms	

3.1.3.2. EL2008 digitalni izlazi

Za upravljanje dijagnostičkim svjetlima na vozilu korišten je Beckhoff EL2008 modul s običnim digitalnim izlazima s podacima prikazanim tablicom 3.4.

Tablica 3.4: Beckhoff EL2008 modul s digitalnim izlazima

Važnije značajke	
<ul style="list-style-type: none">24 V DC, 0.5 A8 digitalnih izlazaZaštita na kratki spoj i obrnuti polaritet	

3.1.3.3. EL3004 analogni ulazi

Za prikupljanje i mjerjenje signala s analognih uređaja poput fotoelektričnih lasera i ultraseničnih senzora korišten je Beckhoff EL3004 modul s naponskim analognim ulazima čiji su podaci prikazani u tablici 3.5.

Tablica 3.5: Beckhoff EL3004 modul s analognim ulazima

Važnije značajke	
<ul style="list-style-type: none">Naponski diferencijalni ulazi, ± 10 V4 analogna ulaza12-bitna rezolucija	

3.1.3.4. EL3692 mjerjenje otpora

Za očitanje pomaka linearnog potenciometra korišten je Beckhoff EL3692 modul za mjerenje otpora čiji su podaci prikazani u tablici 3.6.

Tablica 3.6: Beckhoff EL3692 modul za mjerjenje otpora

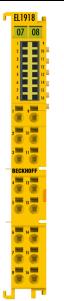
Važnije značajke	
<ul style="list-style-type: none">$2100 \text{ m}\Omega \dots 10 \text{ M}\Omega$2 analogna ulaza za mjerjenje otporaDvožično ili četverožično mjerjenje	

3.1.3.5. EL1918 sigurnosni digitalni ulazi

Za praćenje stanja sigurnosnih tipkala za isklop u nuždi te raznih drugih sigurnosnih tipkala i prekidača korišten je Beckhoff EL1918 modul sa sigurnosnim digitalnim ulazima čije su karakteristike prikazana u tablici 3.7. Razlika između običnih digitalnih ulaza i sigurnosnih je u tome što sigurnosni ulazi imaju dodatne funkcije poput provjere integriteta vodiča, provjere prisutnosti signala i dijagnostičkih mogućnosti koje omogućuju identifikaciju problema ili kvarova u sigurnosnim uređajima. Na EL1918 modulu svaki digitalni ulaz sastoji se od dva dijela, izvora i ponora. Izvor generira tzv. "ispitne impulse" koji se zatim registriraju na ponoru što označava logičku jedinicu.

Najjednostavniji primjer ove funkcionalnosti jest povezivanje sigurnosnih tipkala za isklop u nuždi s NC kontaktom (eng. *Normally Closed*). Sve dok se signal generiran na izvoru prima na ponoru, tipkalo je u normalnom položaju. Pri pritisku na tipkalo signal se prestaje registrirati na ponoru te je potrebno hitno zaustavljanje sustava.

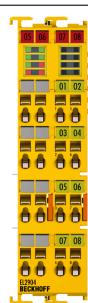
Tablica 3.7: Beckhoff EL1918 modul sa sigurnosnim digitalnim ulazima

Važnije značajke	
<ul style="list-style-type: none"> Komunikacija putem TwinSAFE (EtherCAT/FSoE) protokola 24 V DC 8 sigurnosnih digitalnih ulaza 	

3.1.3.6. EL2904 sigurnosni digitalni izlazi

Sigurnosni digitalni izlazi uglavnom se koriste za uklapanje releja i kontaktora te je za tu svrhu korišten Beckhoff EL2904 modul prikazan tablicom 3.8. Slično kao i kod EL1918 modula sa sigurnosnim digitalnim ulazima, EL2904 podržava testiranje izlaza i mjerjenje struje na izlazima kako bi se provjerio integritet vodiča i otkrili potencijalni neželjeni križni spojevi.

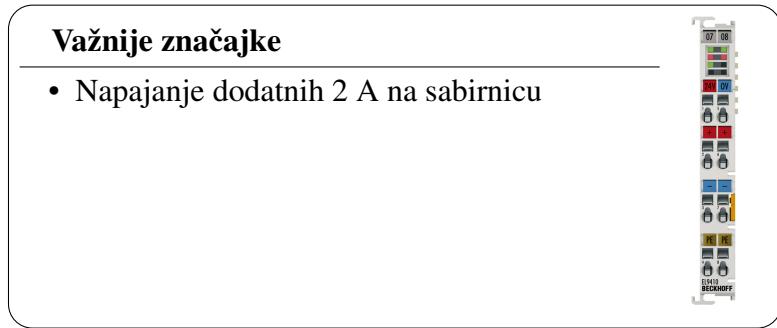
Tablica 3.8: Beckhoff EL2904 modul sa sigurnosnim digitalnim izlazima

Važnije značajke	
<ul style="list-style-type: none"> Komunikacija putem TwinSAFE (EtherCAT/FSoE) protokola 24 V DC, 0.5 A 4 sigurnosna digitalna izlaza 	

3.1.3.7. EL9410 dodatno napajanje za sabirnicu

Svaki modul na sabirnici troši određenu količinu struje, a maksimalna raspoloživa struja na svakoj sabirnici je do 2 A. Zbog velikog broja modula na drugoj sabirnici (slika 3.1) dodan je EL9410 modul prikazan tablicom 3.9, radi dodatnog napajanja sabirnice te je time strujno ograničenje na sabirnici povećano s 2 na 4 A.

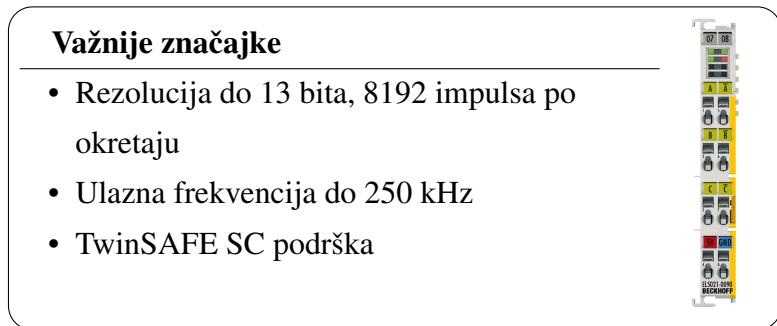
Tablica 3.9: Beckhoff EL9410 modul za dodatno napajanje sabirnice



3.1.3.8. EL5021-0090 sučelje za SinCos enkoder

Za potrebe praćenja brzine vozila korišteni su SinCos enkoderi te time i Beckhoff EL5021-0090 modul sa sučeljem za spajanje SinCos enkodera čiji su podaci prikazani u tablici 3.10. EL5021-0090 modul uz standardno očitanje brzine s enkodera podržava TwinSAFE SC (TwinSAFE Single Channel) funkcionalnost. Time je moguće koristiti standardne signale unutar sigurnosnog programa.

Tablica 3.10: Beckhoff EL5021-0090 modul sa sučeljem za SinCos enkoder



3.1.3.9. EL5001-0090 sučelje za SSI enkoder

Za potrebe spajanja SSI (eng. *Synchronous Serial Interface*) enkodera korišten je Beckhoff EL5001-0090 modul sa sučeljem za SSI enkoder prikazan u tablici 3.11. Isto kao i kod EL5021-0090 modula, EL5001-0090 podržava TwinSAFE SC funkcionalnost.

Tablica 3.11: Beckhoff EL5001-0090 modul sa sučeljem za SSI enkoder

Važnije značajke
<ul style="list-style-type: none">• Rezolucija do 32 bita• Podržano binarno ili gray kodiranje• Ulazna frekvencija do 1 MHz• TwinSAFE SC podrška



3.1.3.10. EL6751 CANopen komunikacijsko sučelje

Za povezivanje originalnog upravljačkog sustava na viličaru s programabilnim logičkim kontrolerom te za daljnje povezivanje raznih CANopen uređaja korišten je Beckhoff EL6751 modul s CANopen komunikacijskim sučeljem prikazan u tablici 3.12.

Tablica 3.12: Beckhoff EL6751 modul s CANopen komunikacijskim sučeljem

Važnije značajke
<ul style="list-style-type: none">• Master/Slave načini rada• Brzina prijenosa podataka od 10, 20, 50, 100, 125, 250, 500, 800 ili 1000 $\frac{\text{kbit}}{\text{s}}$



3.2. Senzori

Na vozilo su postavljeni razni obični i sigurnosni senzori potrebni za nadzor stanja i okoline vozila te za precizno upravljanje osima viličara u zatvorenoj petlji.

Razlika između običnih i sigurnosnih senzora je ponajviše u njihovoј primjeni. Obični senzori uglavnom se koriste za precizno upravljanje, dok se sigurnosni senzori koriste u aplikacijama gdje je potrebno brzo i sigurno zaustavljanje sustava u slučaju u nuždi kako bi se spriječile ozljede čovjeka ili oštećivanje drugih strojeva i infrastrukture u radnoj okolini viličara. Razina sigurnosti senzora definirana je razinama SIL i PL. SIL (eng. *Safety Integrity Level*) i PL (eng. *Performance Level*) su dvije različite metode za klasifikaciju i procjenu sigurnosnih funkcija u industrijskim sustavima. Obje metode koriste se za mjerjenje pouzdanosti i razine sigurnosti pojedinih sigurnosnih funkcija. SIL je mjera sigurnosne razine integriteta sustava, definirana prema standardu IEC 61508 te se koristi za procjenu rizika i

određivanje potrebne razine sigurnosti u sustavima za zaštitu od opasnosti. SIL razina kreće se od SIL 1 (najniža razina sigurnosti) do SIL 4 (najviša razina sigurnosti). PL je mjera razine performansi sigurnosnih funkcija, definirana prema standardu ISO 13849-1, a koristi se za procjenu pouzdanosti i performansi sigurnosnih komponenti i sustava. PL se kreće od PL a (najniža razina performansi) do PL e (najviša razina performansi).

Za mjerjenje visine vilica i za zatvaranje petlje za okomitu os korišten je fotoelektrični senzor prikazan u tablici 3.13.

Tablica 3.13: SICK DT35-B15551 fotoelektrični senzor za mjerjenje udaljenosti

Važnije značajke

- Mjerjenje udaljenosti 50 mm ... 12000 mm
- Preciznost ± 10 mm
- Strujni / naponski analogni izlaz
4 mA ... 20 mA / 0 V ... 10 V
- 12-bitna rezolucija



Za mjerjenje bočnog otklona vilica i za zatvaranje petlje za pomak vilica u bočnoj osi korišten je fotoelektrični senzor prikazan tablicom 3.14.

Tablica 3.14: SICK WTT12L-A2553 fotoelektrični senzor za mjerjenje udaljenosti

Važnije značajke

- Mjerjenje udaljenosti 50 mm ... 1800 mm
- Preciznost ± 15 mm
- Strujni / naponski analogni izlaz
4 mA ... 20 mA / 0 V ... 10 V
- 12-bitna rezolucija



Ultrasonični senzor iz tablice 3.15 korišten je za detekciju prisutnosti palete na vilicama.

Tablica 3.15: SICK UM18-218127111 ultrasonični senzor

Važnije značajke

- Radni raspon 120 mm ... 1000 mm
- Granični raspon 1300 mm
- Naponski analogni izlaz 0 V ... 10 V
- 12-bitna rezolucija



Senzori iz tablica 3.13, 3.14 i 3.15 prebačeni su u režim rada s naponskim izlazima i spojeni na EL3004 modul s naponskim analognim ulazima. Obzirom da EL3004 modul ima analogne ulaze koji mjere napon u rasponu -10 V ... 10 V, a raspon analognih izlaza ova tri senzora je 0 V ... 10 V, negativni dio analognih ulaza na modulu su uzemljeni. Time je umjetno skraćen mjerni raspon analognih ulaza od 0 V ... 10 V.

Za mjerjenje nagiba vilica koristi se linearni potenciometar prikazan u tablici 3.16 koji je spojen na EL3692 modul za mjerjenje otpora.

Tablica 3.16: MEGATRON RC13-50-G linearni potenciometar

Važnije značajke

- Ukupni otpor 2 kΩ
- Mehanički hod 50 mm



Za dodatnu provjeru pozicije vilica u bočnoj osi i osi nagiba, koristi se induktivni senzor prikazan tablicom 3.17. U osi za bočni pomak vilica senzor daje povratnu informaciju o tome nalaze li se vilice u središtu osi, a u osi nagiba nalaze li se vilice u poziciji najvišeg nagiba. Senzori su spojeni na EL1918 modul sa sigurnosnim digitalnim ulazima.

Tablica 3.17: SICK IME2S12-04N4DC0 induktivni senzor

Važnije značajke

- 2 OSSD izlaza
- Mjerna udaljenost ispod 3.2 mm
- SIL2 ; PL d



Za mjerjenje nagiba cijelog vozila korišten je senzor nagiba prikazan u tablici 3.18.

Tablica 3.18: SICK TMM88A-PCI090 senzor za mjerjenja nagiba

Važnije značajke

- Mjerni raspon $\pm 90^\circ$
- Rezolucija 0.01°
- CANopen komunikacijsko sučelje



3.2.1. Enkoderi

Na pogonske kotače viličara postavljena su dva sigurnosna SinCos enkodera iz tablice 3.19 radi sigurnosnog praćenja brzine vozila. Enkoderi su spojeni na EL5021-0090 modul.

Tablica 3.19: SICK sigurnosni SinCos enkoder

Važnije značajke

- Inkrementalni Sin/Cos enkoder
- 1024 Sin/Cos periodi po okretaju
- Raspon mjerjenja brzine do 12000 min^{-1}
- SIL2 ; PL d



Za praćenje visine vilica korišten je absolutni enkoder iz tablice 3.20 koji se sastoji od SinCos i SSI enkodera. U ovakvim senzorima, SinCos enkoder koristi se za određivanje kutnog položaja, a SSI enkoder za praćenje napravljenih punih revolucija te serijski prijenos podataka prema PLC-u.

Tablica 3.20: SICK sigurnosni višeokretni SSI enkoder

Važnije značajke	
<ul style="list-style-type: none">• SSI + Sin/Cos enkoder• 1024 Sin/Cos periodi po okretaju• Maksimalna rezolucija (broj impulsa po okretaju × broj okretaja): 18 bit × 12 bit (262144 × 4096)• SIL3 ; PL e	

3.2.2. Laserski skeneri

Laserski skener je uređaj koji se koristi za skeniranje prostora i prikupljanje informacija o obliku, udaljenosti i položaju objekata u prostoru. Koristeći se LiDAR tehnologijom (eng. *Light Detection and Ranging*) laserski skeneri emitiraju zrake koje se odbijaju od objekata u prostoru i vraćaju u senzor, gdje se zatim obrađuju podaci o udaljenosti i kutu refleksije koji se dalje koriste za stvaranje digitalne slike objekata.

Za praćenje okoline robota korištena su dva jednaka microScan3 sigurnosna laserska skenera tvrtke SICK, namijenjeni za zaštitu ljudi i strojeva od sudara i ozljeda u industrijskim okruženjima. Neke od važnijih značajki sigurnosnog laserskog skenera navedene su u tablici 3.21.

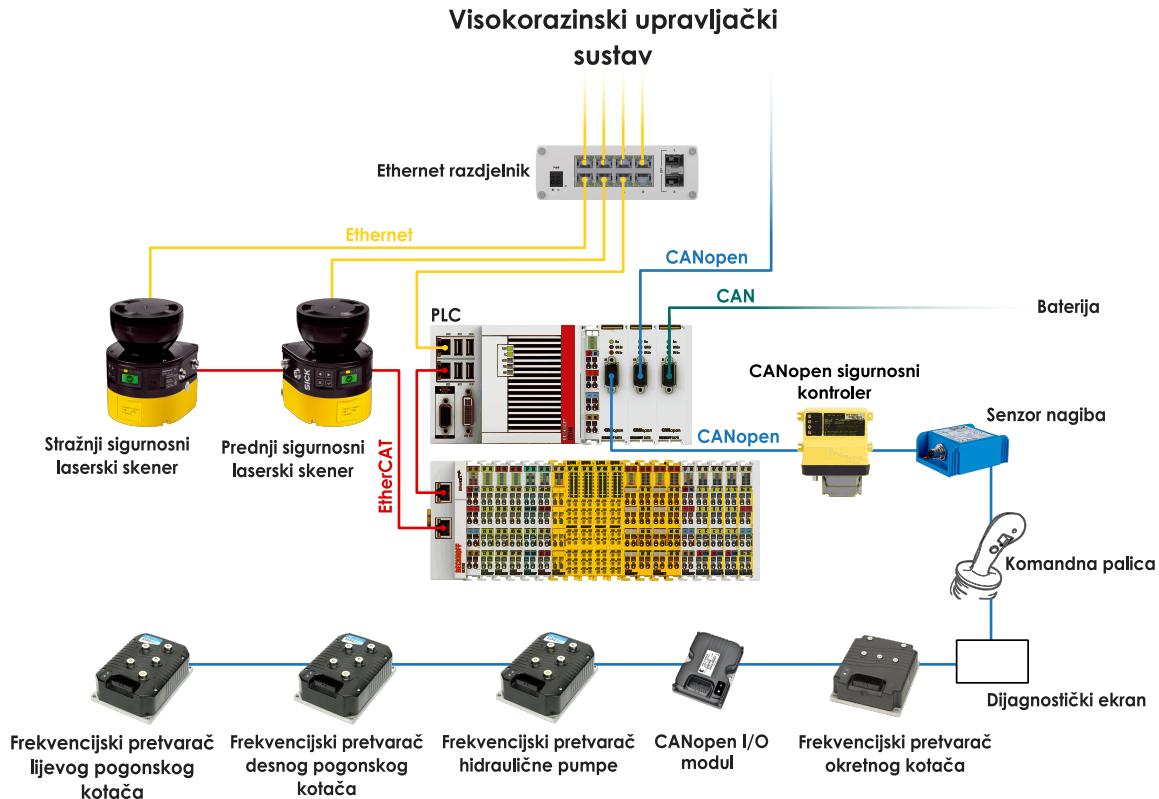
Tablica 3.21: SICK microScan3 sigurnosni laserski skener

Važnije značajke	
<ul style="list-style-type: none">• Raspon zaštitne zone do 9 metara• Raspon zone upozorenja do 64 metara• Dinamičko mijenjanje aktivnih zona• Komunikacija putem EtherCAT/FSoE protokola• Slanje izmjerениh vrijednosti putem Ethernet protokola	

4. Pregled korištenih komunikacijskih protokola

U ovom poglavlju napravljen je kratki pregled korištenih komunikacijskih protokola pri povezivanju uređaja u niskorazinskom upravljačkom sustavu.

Na slici 4.1 nalazi se shema povezanosti uređaja po komunikacijskim mrežama.



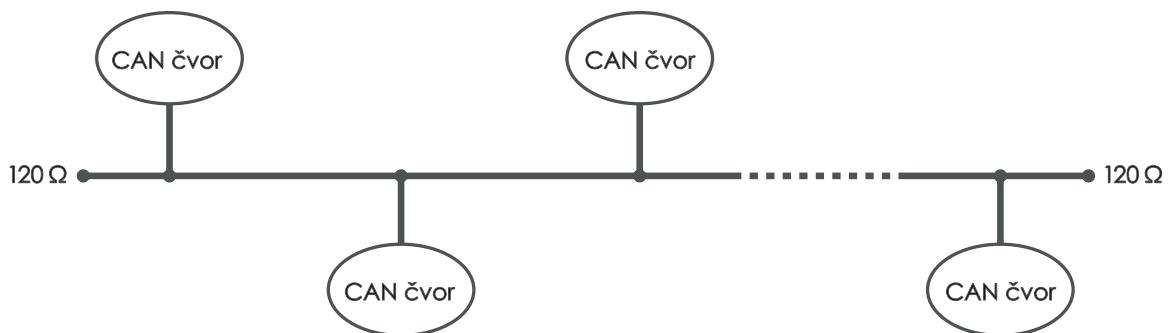
Slika 4.1: Pregled komunikacijskih mreža u niskorazinskom upravljačkom sustavu

U srcu niskorazinskog upravljačkog sustava nalazi se PLC koji je sudionik u svim komunikacijskim mrežama. Za povezivanje uređaja na viličaru s PLC-om korišten je CANopen protokol, za povezivanje baterije s PLC-om korišten je CAN protokol, za povezivanje laserskih skenera s PLC-om te za međusobno povezivanje PLC modula i središnje upravljačke jedinice korišten je EtherCAT protokol, a za povezivanje PLC-a s visokorazinskim upravljačkim sustavom korišteni su Ethernet i CANopen protokoli.

4.1. CAN protokol

CAN (eng. *Controller Area Network*) komunikacijski je protokol razvijen kao standard za serijsku komunikaciju između uređaja u vozilima i kasnije općenito uređajima u industrijskim okruženjima. CAN protokol temelji se na OSI (eng. *Open Systems Interconnection*) modelu i sastoji se od fizičkog i podatkovnog sloja. Fizički sloj definira fizikalne parametre komunikacijske sabirnice poput naponskih razina i brzinu prijenosa signala, a podatkovni sloj definira format poruke te način prijenosa podataka.

Slikom 4.2 prikazana je topologija CAN mreže. Uređaji, odnosno čvorovi na istoj mreži spajaju se serijski s dvije žice, CAN-Hi i CAN-Lo te se krajevi linije zaključavaju s karakterističnom impedancijom od 120Ω radi suzbijanja reflektiranog vala.



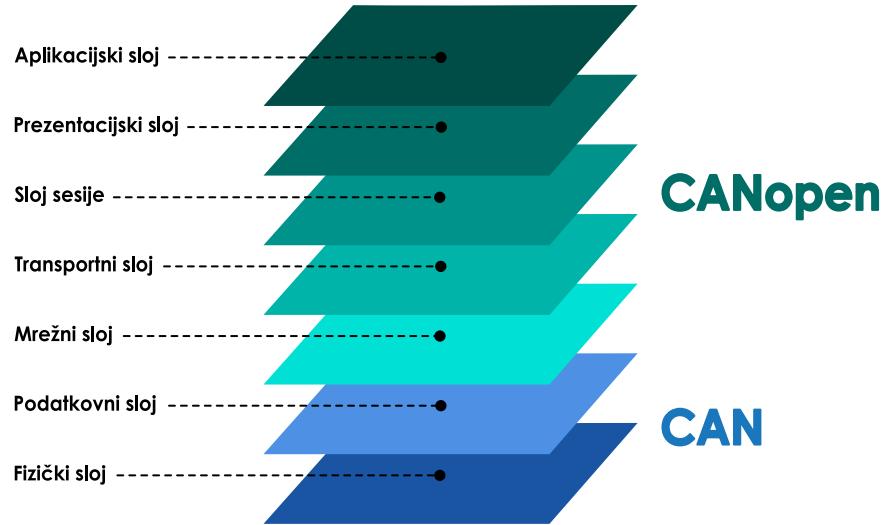
Slika 4.2: Topologija CAN mreže

Svi čvorovi imaju jednak pristup sabirnici te svaki može djelovati kao pošiljatelj ili primatelj poruka. Svaka poruka na mreži sadrži svoj jedinstveni identifikator te postavljanjem filtera na samom uređaju određuje se koje poruke od drugih uređaja taj uređaj čita, a koje ignorira. Poruke s nižim identifikatorom imaju prioritet nad onim s višim.

4.1.1. CANopen protokol

CANopen je komunikacijski protokol koji nadograđuje standardni CAN protokol i pruža dodatne značajke i funkcionalnosti za industrijske aplikacije. Slikom 4.3 prikazani su CAN i CANopen u OSI modelu. CAN i CANopen uređaji mogu se spajati zajedno na istu mrežu jer CANopen koristi isti fizički i podatkovni sloj kao CAN. Za razliku od standardnog CAN protokola, CANopen definira standardizirane komunikacijske profile i protokole za pojedinu vrstu uređaja koji olakšavaju interoperabilnost između različitih uređaja različitih proizvođača. Također, CANopen pruža napredne mehanizme za upravljanje uređajima u mreži, uključujući parametrizaciju, dijagnostiku, konfiguraciju i nadzor, kao i mogućnost ažuriranja parametara tijekom rada sustava.

Struktura okvira CAN poruke prikazana je na slici 4.4.



Slika 4.3: CAN i CANopen u OSI modelu

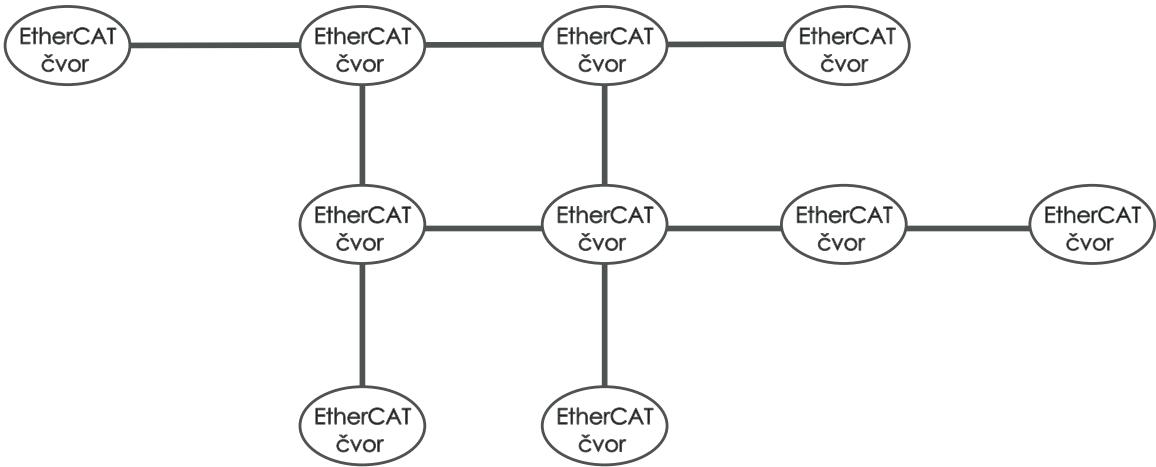


Slika 4.4: Struktura CANopen poruke

Prvi bit, SOF (eng. *Start Of Frame*) označava početak poruke. U sljedećih jedanaest bitova zapisan je COB (eng. *Communication Object Identifier*), odnosno identifikator poruke za standardni CAN protokol ili identifikator uređaja za CANopen protokol. Bit RTR (eng. *Remote Transmission Request*) označava da se radi o zahtjevu za prijenosom podataka. Ako je postavljen u logičku jedinicu, to znači da je poruka zahtjev za podacima, a ako je postavljen u logičku nulu označava prijenos stvarnih podataka. Zatim slijedi polje za upravljačke bitove koji se koriste za slanje naredbi i upravljačkih signala između čvorova te polje za prijenos podataka. CRC (eng. *Cyclic Redundancy Check*) bitovi koriste se za provjeru integriteta poruke, dok se dva bita u ACK (eng. *Acknowledge*) polju koriste za signalizaciju primitka poruke. Na kraju okvira nalazi se polje EOF (eng. *End Of Frame*) koje označava kraj poruke.

4.2. EtherCAT protokol

EtherCAT (eng. *Ethernet for Control Automation Technology*) je protokol za komunikaciju u realnom vremenu dizajniran za primjene u industrijskoj automatizaciji kojeg je razvila tvrtka Beckhoff Automation 2003. godine, a sada ga održava EtherCAT Technology Group (ETG). EtherCAT koristi isti fizički i podatkovni sloj kao Ethernet protokol što znači da može dijeliti istu mrežnu infrastrukturu s drugim protokolima koji se temelje na Ethernet-u, poput TCP/IP ili UDP protokola. Poput Etherneta, uređaji u EtherCAT mreži mogu se povezivati na više načina kako je i prikazano slikom 4.5.



Slika 4.5: Topologija EtherCAT mreže

Dozvoljene su razne topologije poput linijske, prstenaste, zvjezdaste itd.

Na Ethernet mreži svaki uređaj može slati podatke preko mreže i svaki uređaj može primati podatke, dok na EtherCAT mreži jedino glavni uređaj (engl. *Master*) ima dopuštenje za prijenos podataka preko mreže. Podređeni uređaji (eng. *Slave*) mogu dodati podatke i poslati okvir, ali ne mogu sami kreirati nove poruke. EtherCAT okviri ugrađeni su u Ethernet okvir kao što je prikazano na slici 4.6.



Slika 4.6: Struktura EtherCAT poruke

Ove okvire primaju podređeni uređaji, odnosno čvorovi kojima su adresirani. Podređeni uređaji obrađuju podatke i dodaju natrag sve što je zatražio glavni te šalju okvir do sljedećeg čvora. Sljedeći čvor radi upravo istu stvar, uzima podatke koji su mu namijenjeni, vraća tražene podatke natrag u EtherCAT okvir i šalje ih sljedećem čvoru. Brzina prijenosa podataka veća je u odnosu na standardni Ethernet protokol jer postoji samo jedan uređaj koji šalje podatke i zbog tzv. "obrada u hodu" tehnike. U Ethernetu svaki uređaj mora pročitati zaglavlje svake poruke kako bi utvrdio jesu li podaci namijenjeni njemu, zatim unijeti podatke i obraditi ih na neki način. S obradom u hodu, čvor čita zaglavlje i šalje podatke istovremeno.

4.2.1. Sigurnost preko EtherCAT-a (FSoE)

FSoE (eng. *Functional Safety over EtherCAT*) je proširenje EtherCAT protokola koje omogućuje sigurnosno upravljanje i komunikaciju u industrijskim automatizacijskim sustavima.

Prema OSI modelu FSoE definira dodatni najviši sloj, nazvan sigurnosni komunikacijski sloj. Komunikacija između FSoE uređaja vrši se putem tzv. "crnog kanala" (eng. *black*

channel) koji se odnosi na uporabu nižih EtherCAT slojeva za prijenos sigurnosnih podataka zajedno s ostalim kontrolnim podacima. FSoE podaci ugrađuju se u standardni EtherCAT okvir prikazan slikom 4.6. Moguće je povezivati standardne EtherCAT uređaje i FSoE uređaje na istu mrežu.

5. Proračuni za upravljanje sustavom trakcije viličara

U ovom poglavlju napravljena su dva proračuna potrebna za uspješno upravljanje sustavom trakcije. Prvi proračun se odnosi na raspodjelu brzine po pogonskim kotačima obzirom na postavljeni kut okretnog kotača, a drugi se odnosi na određivanje ekvivalentnog kuta virtualnog okretnog kotača koji za razliku od stvarnog nije pomaknut obzirom na središnju uzdužnu os vozila, nego se nalazi u njoj.

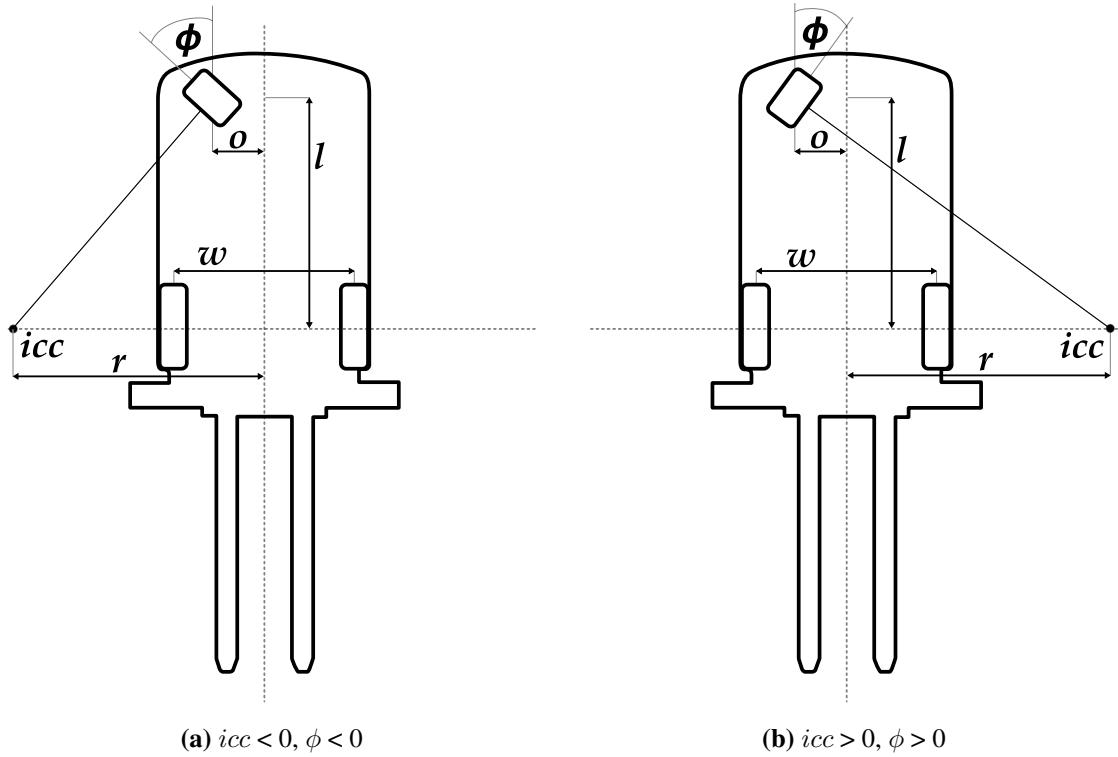
5.1. Proračun raspodjele brzine po pogonskim kotačima

Kako bi se uklonilo proklizavanje kotača, a i kvalitativno upravljalo sustavom trakcije potrebno je obzirom na postavljeni kut okretnog kotača izvesti izraz za raspodjelu brzine po pogonskim kotačima [1]. Izraz se izvodi iz same geometrije kotača na viličaru prema slici 5.1 gdje je:

- ϕ kut koji okretni kotač zatvara sa središnjom uzdužnom osi viličara (za pozitivni predznak kuta viličar skreće udesno, dok za negativni skreće ulijevo)
- o udaljenost okretnog kotača od središnje uzdužne osi
- l osovinski razmak viličara
- w širina traga viličara
- r udaljenost između točke *icc* koja označava trenutno središte zakrivljenosti (eng. *Instantaneous Center of Curvature*) i središta vozila koje se nalazi na sjecištu uzdužne središnje osi i poprečne središnje osi koja prolazi kroz pogonske kotače.

Za izvedbu izraza za raspodjelu referentne vrijednosti linearne brzine po pogonskim kotačima potrebno je prvo odrediti položaj točke *icc*. Pomoću slike 5.1 moguće je izraziti tangens kuta ϕ kao:

$$\tan(\phi) = \frac{l}{r + o} \quad (5.1)$$



Slika 5.1: Geometrija kotača viličara (pogled odozdo)

Točka icc može se nalaziti samo na poprečnoj središnjoj osi i ekvivalentna je po apsolutnom iznosu vrijednosti r te se može izraziti iz jednadžbe 5.1:

$$icc = \frac{l}{\tan(\phi)} - o \quad (5.2)$$

Kako bi se izračunao udio komponenata brzine lijevog i desnog kotača u zadanoj brzini, stavlja se u omjer udaljenost pojedinog pogonskog kotača od točke icc . Izraz za udio brzine lijevog kotača tada glasi:

$$v_{lijevi}^* = \begin{cases} -\frac{icc + \frac{w}{2}}{w} & \phi < 0 \\ \frac{icc + \frac{w}{2}}{w} & \phi > 0 \end{cases} \quad (5.3)$$

a udio brzine za desni kotač:

$$v_{desni}^* = \begin{cases} -\frac{icc - \frac{w}{2}}{w} & \phi < 0 \\ \frac{icc - \frac{w}{2}}{w} & \phi > 0 \end{cases} \quad (5.4)$$

Konačni izrazi za brzinu lijevog i desnog pogonskog kotača mogu se onda zapisati kao:

$$v_{lijevi} = \frac{v_{lijevi}^*}{|v_{lijevi}^*| + |v_{desni}^*|} \cdot v_{linearna} \quad (5.5)$$

$$v_{desni} = \frac{v_{desni}^*}{|v_{lijevi}^*| + |v_{desni}^*|} \cdot v_{linearna} \quad (5.6)$$

Uvrštavanjem izraza 5.3 i 5.4 u izraze 5.5 i 5.6 i uz malo sređivanja, brzine v_{lijevi} i v_{desni} mogu se izraziti direktno kao funkcije točke icc :

$$v_{lijevi} = \begin{cases} -\frac{icc + \frac{w}{2}}{|icc + \frac{w}{2}| + |icc - \frac{w}{2}|} \cdot v_{linearna} & \phi < 0 \\ \frac{icc + \frac{w}{2}}{|icc + \frac{w}{2}| + |icc - \frac{w}{2}|} \cdot v_{linearna} & \phi > 0 \end{cases} \quad (5.7)$$

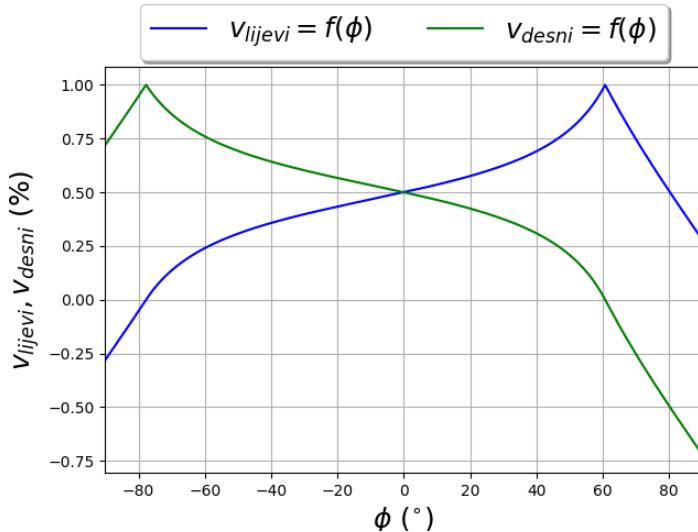
$$v_{desni} = \begin{cases} -\frac{icc - \frac{w}{2}}{|icc + \frac{w}{2}| + |icc - \frac{w}{2}|} \cdot v_{linearna} & \phi < 0 \\ \frac{icc - \frac{w}{2}}{|icc + \frac{w}{2}| + |icc - \frac{w}{2}|} \cdot v_{linearna} & \phi > 0 \end{cases} \quad (5.8)$$

Za kut $\phi = 0^\circ$ izraz 5.2 teži u beskonačno, odnosno točka icc je nedefinirana, što zapravo i odgovara pravocrtnom gibanju vozila gdje se oba pogonska kotača vrte jednakom brzinom u istom smjeru.

Za kut $\phi = \pm 90^\circ$, izraz 5.2 prelazi u $icc = -o$, što odgovara udaljenosti okretnog kotača od središnje uzdužne osi.

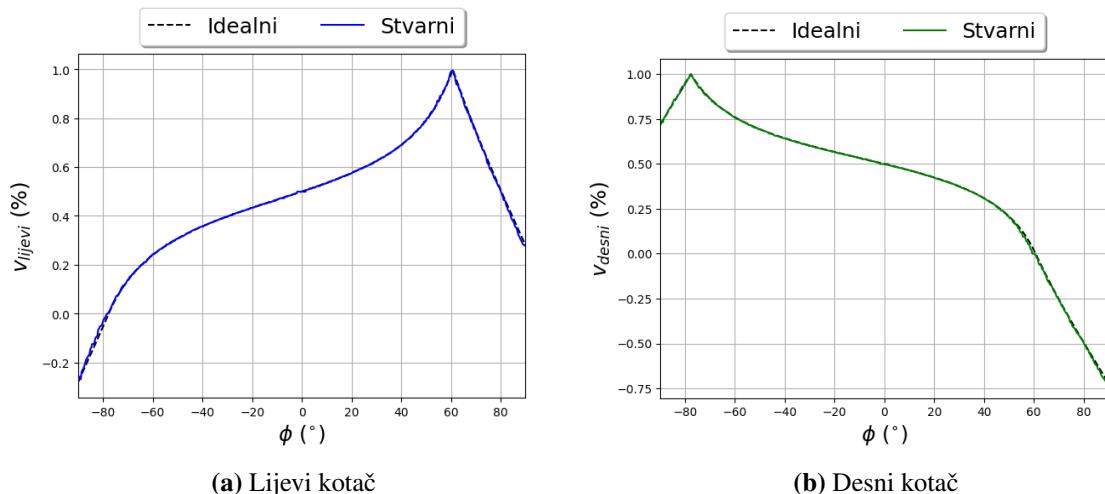
U slučaju $icc = 0$, viličar rotira oko svoje osi, a iz slike 5.1 može se uočiti da je rotacija na mjestu moguća samo za slučaj gdje kut ϕ ima pozitivan predznak. Kod viličara Trey taj kut iznosi $\approx 80.3^\circ$.

Slikom 5.2 prikazana je ovisnost brzina oba pogonska kotača o kutu ϕ za geometriju kotača viličara Trey,



Slika 5.2: Brzine v_{lijevi} i v_{desni} kao funkcije kuta ϕ kod viličara Trey

a na slici 5.3 prikazani su odzivi stvarnih brzina obzirom na zadani kut ϕ .

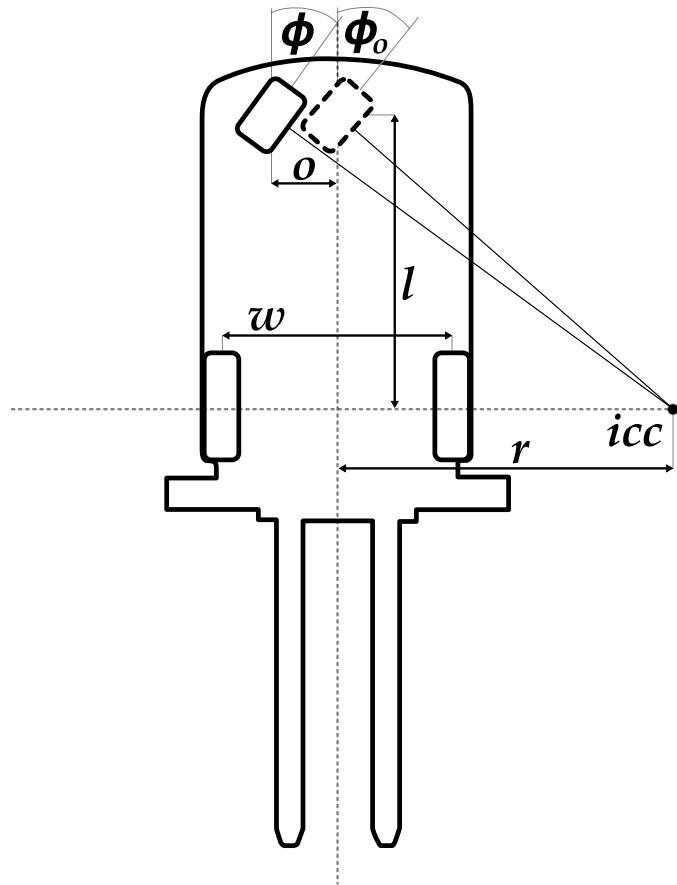


Slika 5.3: Odzivi stvarne brzine vrtnje pogonskih kotača za zadani kut ϕ

Iz slika 5.3a i 5.3b može se uočiti da odzivi stvarne brzine lijevog i desnog kotača vrlo precizno prate zadalu vrijednost referentne brzine.

5.2. Proračun ekvivalentnog kuta virtualnog kotača

Visokorazinski upravljački sustav zadaje referentnu vrijednost linearne brzine i kuta okretog kotača, uz pretpostavku da se okretni kotač nalazi na središnjoj uzdužnoj osi viličara, odnosno na jednakoj udaljenosti od oba pogonska kotača. Zbog toga je potreban proračun kuta ϕ_0 tzv. "virtualnog" okretnog kotača, za koji visokorazinski upravljački sustav zadaje željenu referentnu vrijednost, u kut ϕ stvarnog okretnog kotača kao što je predviđeno slikom 5.4.



Slika 5.4: Virtualni kotač viličara

Pri izračunu korištena su svojstva Ackermanove geometrije koja pruža metodu za određivanje kutova okretnih kotača na osnovu željene putanje vozila. Kada bi viličar imao dva okretna kotača, pri skretanju unutarnji i vanjski okretni kotač trebali bi se postaviti u različite kuteve kako bi oba okretna kotača imali zajedničku točku trenutnog središta zakrivljenosti [2]. Iz ovog se uvjeta izvodi ovisnost izlazne vrijednosti kuta ϕ o ulaznoj vrijednosti kuta ϕ_0 .

Prema slici 5.4 postavljaju se dvije jednadžbe:

$$\tan(\phi) = \frac{l}{r + o} \quad (5.9)$$

$$\tan(\phi_0) = \frac{l}{r} \quad (5.10)$$

te kako bi se izvela ovisnost kuta ϕ o kutu ϕ_0 i ujedno pokratila nepoznаница r , oduzimaju se recipročне vrijednosti izraza 5.9 i 5.10:

$$\frac{1}{\tan(\phi)} - \frac{1}{\tan(\phi_0)} = \text{ctan}(\phi) - \text{ctan}(\phi_0) = \frac{r+o}{l} - \frac{r}{l} = \frac{o}{l} \quad (5.11)$$

$$\text{ctan}(\phi) = \text{ctan}(\phi_0) + \frac{o}{l} \quad (5.12)$$

$$\tan(\phi) = \frac{l \cdot \tan(\phi_0)}{l + o \cdot \tan(\phi_0)} \quad (5.13)$$

iz čega slijedi općeniti izraz za ovisnost kuta ϕ o kutu ϕ_0 :

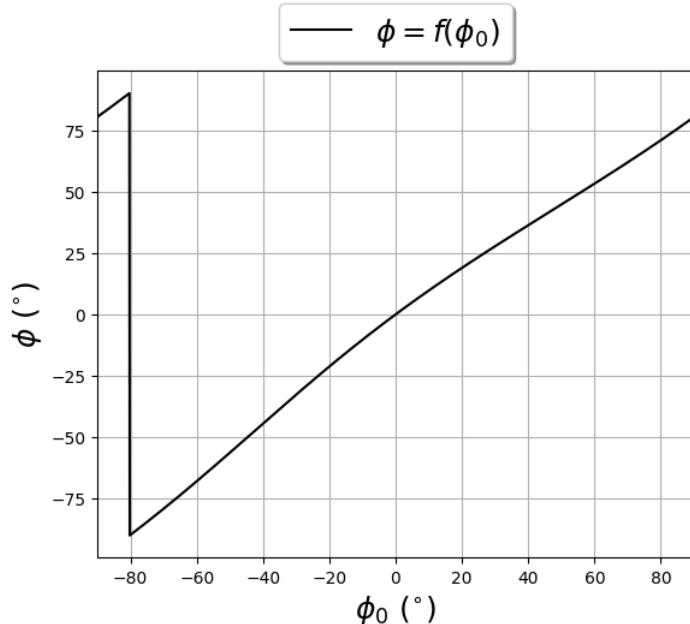
$$\phi = \arctan \left(\frac{l \cdot \tan(\phi_0)}{l + o \cdot \tan(\phi_0)} \right) \quad (5.14)$$

U slučaju kada je $\phi_0 = \pm 90^\circ$, točka *icc* nalazi se u sjecištu središnje uzdužne i središnje poprečne osi te izraz 5.14 prelazi u:

$$\phi = \arctan \left(\frac{l}{o} \right) \quad (5.15)$$

što odgovara okretu vozila oko svoje osi.

Slikom 5.5 prikazana je ovisnost kuta ϕ o kutu ϕ_0 kod viličara Trey.



Slika 5.5: Kut ϕ kao funkcija kuta ϕ_0 kod viličara Trey

6. Postavljanje i konfiguracija uređaja

U ovom poglavlju opisani su koraci i metode pri postavljanju frekvencijskih pretvarača i sigurnosnih laserskih skenera kako bi se mogli povezati s programabilnim logičkim kontrolerom i integrirati unutar razvojnog okruženja TwinCAT 3.

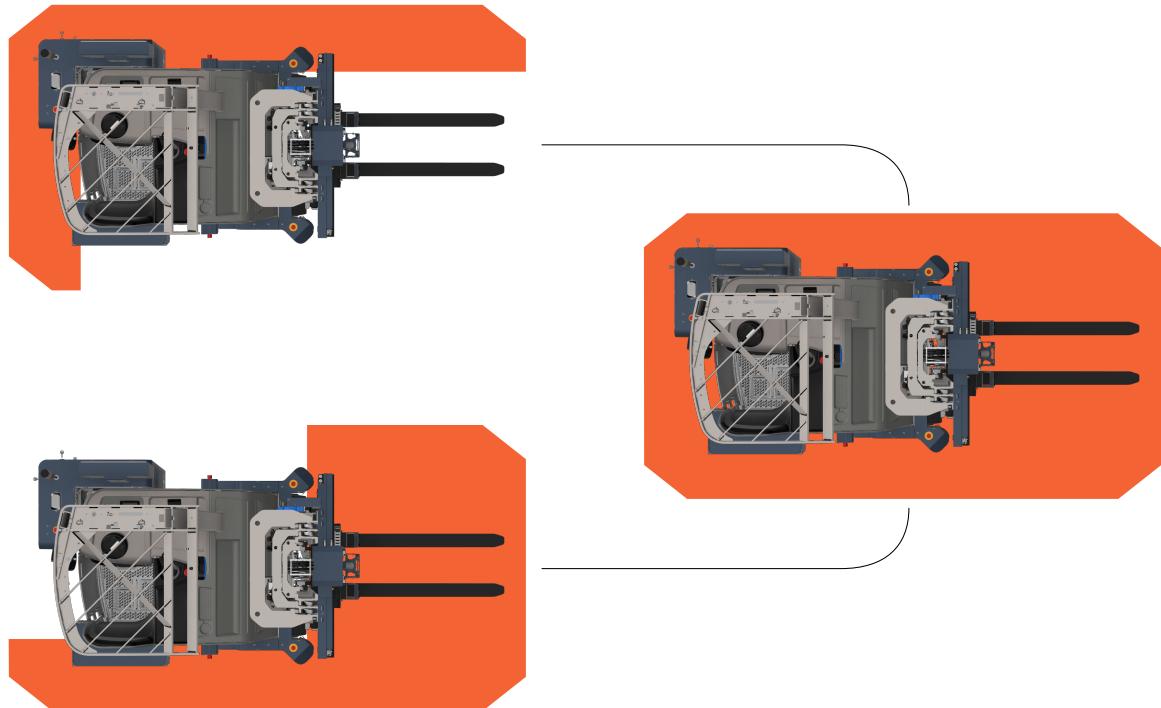
6.1. Postavljanje frekvencijskih pretvarača

Postavke i parametri frekvencijskih pretvarača na vozilu ne razlikuju se puno od onih tvorničkih koje je postavio proizvođač. Postavljeni parametri motora, enkodera i regulatora ostavljeni su istima, a promijenjeni su komunikacijski parametri kako bi se uređaji mogli povezati na CANopen mrežu gdje je PLC glavni uređaj (eng. *Master*) te su dodani STO (eng. *Safe Torque Off*) signali kojima upravlja EL2904 modul sa sigurnosnim digitalnim izlazima.

Frekvencijski pretvarači koji upravljaju pogonskim kotačima i hidrauličnom pumpom imaju dodatnu funkcionalnost izvođenja korisničkog programskog koda. Kod je pisan u VCL (eng. *Vehicle Control Language*) programskom jeziku, čija je sintaksa od zastupljenijih programskih jezika najsličnija onoj C programskog jezika. U VCL skripti koja se izvodi na pretvaračima definirana je razmjena podataka preko CANopen protokola sa višim sustavom upravljanja, odnosno s programabilnim logičkim kontrolerom te je također implementirana logika za obradu grešaka. Na temelju ove logike napisana je EDS (eng. *Electronic Data Sheet*) datoteka u kojoj se navode adresa i tip podataka koje podređeni uređaj šalje prema glavnem uređaju te adresa i tip podataka koje podređeni uređaj prima od glavnog uređaja. Frekvencijski pretvarači od PLC-a primaju referentnu vrijednost brzine vrtnje za električni motor, a pretvarači šalju natrag stvarnu vrijednost brzine vrtnje, temperaturu motora, struju namota statora, stanje elektromagnetske kočnice za onemogućavanje vrtnje motora te informaciju o prisutnosti i vrsti aktivne greške, uz još nekolicinu drugih dijagnostičkih informacija.

6.2. Postavljanje laserskih skenera

Na viličaru se nalaze dva dijametalno postavljena sigurnosna laserska skenera koji zajedno nadziru neposrednu okolinu robota u punom vidokrugu od 360° , kako je to i prikazano slikom 6.1.



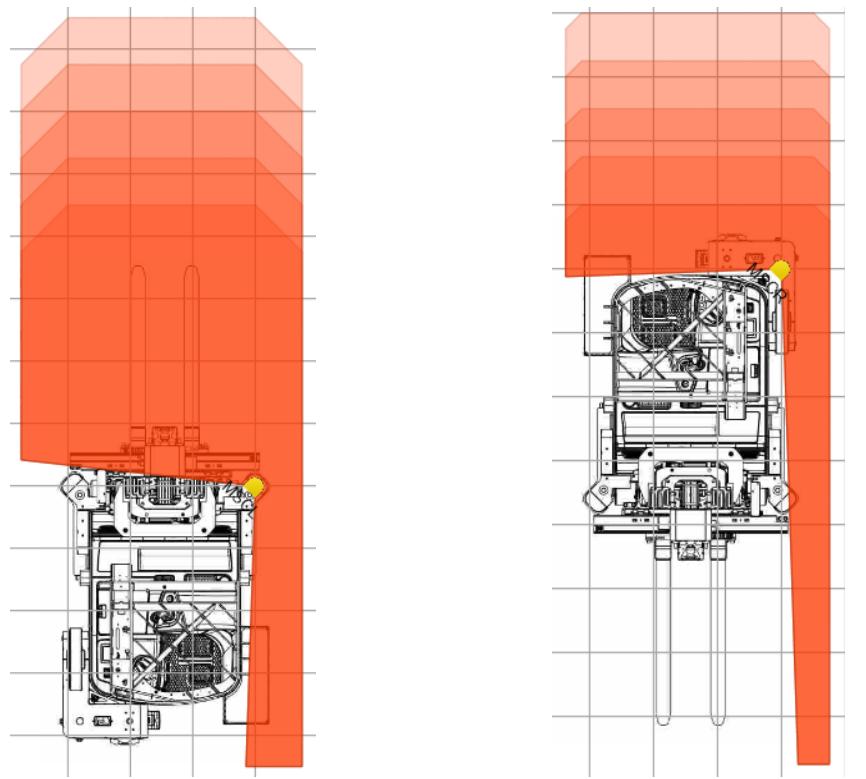
Slika 6.1: Sigurnosne zone oko robota

Praćenje sigurnosnih zona moguće je samo u slučajevima gdje vilice nisu spuštene do kraja, nego malo uzdignute, tako da je u operacijama spuštanja ili podizanja paleta potrebno privremeno isključiti sve sigurnosne zone oko robota.

Laserski skeneri postavljaju se unutar programskog sučelja Safety Designer tvrtke SICK. U Safety Designer-u postavljeni su komunikacijski parametri za Ethernet i EtherCAT/FSoE komunikaciju s drugim uređajima te su definirane dimenzije sigurnosnih zona. Na oba lasera postavljeno je pet sigurnosnih zona iste širine, ali različitih duljina kao što je prikazano slikom 6.2.

Zaštitna zona na laseru odabire se obzirom na brzinu i orijentaciju robota. Na laseru u smjeru kretanja odabire se duža zona, dok zona na laseru obrnutom od smjera kretanja ostaje skraćena. Ova funkcionalnost prikazana je slikom 6.3.

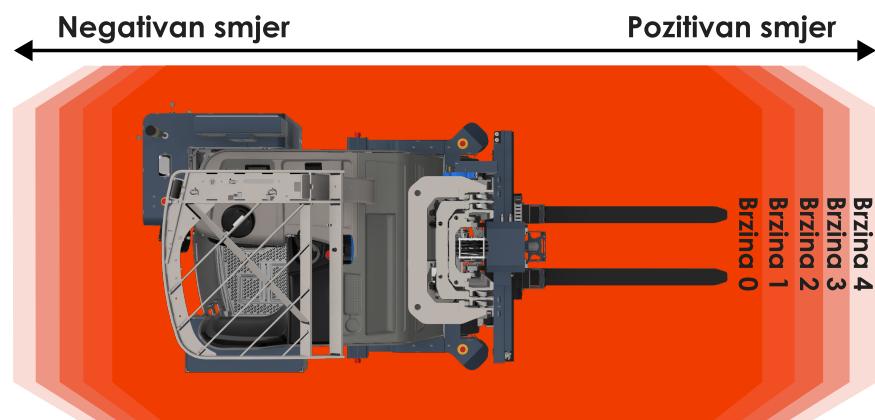
Trenutna aktivna zona na laseru odabire se u PLC sigurnosnoj logici slanjem odgovarajuće vrijednosti u varijabli veličine jednog bajta.



(a) Zaštitne zone prednjeg lasera

(b) Zaštitne zone stražnjeg lasera

Slika 6.2: Postavljene zone unutar programskog sučelja Safety Designer



Slika 6.3: Zaštitne zone oko robota

7. Implementacija upravljačke logike na PLC-u

PLC programiranje u sklopu ovog diplomskog rada temeljeno je na upotrebi naprednog razvojnog okruženja TwinCAT 3 tvrtke Beckhoff Automation koje se koristi za programiranje i implementaciju industrijskih kontrolnih sustava.

Arhitektura TwinCAT 3 sustava raspoređena je u više upravljačkih cjelina koje omogućuju ravnomjerno raspoređivanje zadataka i podataka na različite upravljačke jedinice. Za potrebe ovog rada korištene su dvije takve cjeline, odnosno projekta, tzv. "Standardni PLC projekt" i "Sigurnosni PLC projekt". Unutar standardnog PLC projekta nalazi se glavno sučelje za razvoj i programiranje u TwinCAT 3 okruženju, dok sigurnosni PLC projekt pruža integrirane sigurnosne funkcionalnosti i certificirane komponente koje omogućuju implementaciju sigurnosnih programa. Ove funkcionalnosti osiguravaju siguran rad strojeva i opreme, umanjujući rizik od ozljeda i nesreća.

Zbog provedenih modifikacija nad originalnim sustavom na viličaru, u svim režimima rada PLC je zadužen za upravljanje vozilom. To znači da za razliku od originalne platforme viličara, više nije moguće upravljanje vozilom u ručnom režimu rada bez da se izvodi logički program na PLC-u.

7.1. Dodavanje uređaja u PLC projekt

Za dodavanje CANopen uređaja u PLC projekt učitava se već ranije spomenuta EDS datoteka u PLC u kojoj su zapisani komunikacijski parametri uređaja. U slučaju nepostojanja EDS datoteke potrebno je ručno postaviti komunikacijske parametre unutar PLC sučelja kako bi se uređaji vidjeli na mreži i mogli razmjenjivati podatke. U tablici 7.1 napravljen je pregled CANopen i CAN mreža na Trey viličaru. Uz svaki uređaj u zagradi je naveden njegov *Node ID* (identifikator čvora, odnosno uređaja) koji je jedinstven za svaki čvor na mreži.

Tablica 7.1: Podjela CANopen uređaja

CANopen mreža #1 - 800 kbit/s	
Glavni uređaj	Podređeni uređaji
EL6751 CANopen modul (127)	Dijagnostički ekran (2) Komandna palica (3) Frekvencijski pretvarač okretnog kotača (4) Frekvencijski pretvarač lijevog pogonskog kotača (6) Frekvencijski pretvarač desnog pogonskog kotača (7) Frekvencijski pretvarač hidrauličke pumpe (8) CANopen I/O modul (12) Senzor nagiba (18) CANopen sigurnosni kontroler (20)
CAN mreža #2 - 250 kbit/s	
Glavni uređaj	Podređeni uređaji
EL6751 CANopen modul (127)	Baterija (*)
CANopen mreža #3 - 500 kbit/s	
Glavni uređaj	Podređeni uređaji
Industrijsko računalo (127)	EL6751 CANopen modul (10)

*U CAN mreži svaka poruka ima svoj identifikator, ne i uređaj

Slično kao i za CANopen uređaje, EtherCAT uređaji u PLC projekt dodavaju se učitavanjem XML datoteka koje se istovjetne EDS datotekama, ali za EtherCAT uređaje. Jedna od prednosti koja EtherCAT mreža nudi nad CANopen mrežom jest da se pri identifikaciji uređaja na mreži koju provodi glavni uređaj mogu povući s podređenih uređaja i njihove XML datoteke te je time proces dodavanja i parametriranja svih EtherCAT uređaja na mreži automatiziran. Uz to, EtherCAT uređajima na mreži nije potrebno ručno postavljati parametre jer to već vrši glavni uređaj. Jedini specifični parametar koji je potrebno ručno postaviti jest FSofE adresa za sigurnosne uređaje. U tablici 7.2 napravljen je pregled svih EtherCAT/FSofE uređaja te je u zagradi uz svaki čvor navedena njegova jedinstvena FSofE adresa.

Tablica 7.2: Podjela EtherCAT/FSoE uređaja

EtherCAT/FSoE mreža	
Glavni uređaj	Podređeni uređaji
EL6910 modul (1)	EL1918 modul #1 (20) EL1918 modul #2 (21) EL1918 modul #3 (22) EL1918 modul #4 (23) EL2904 modul #1 (40) EL2904 modul #2 (41) EL5021-0090 modul #1 (60) EL5021-0090 modul #2 (61) EL5021-0090 modul #3 (62) EL5001-0090 modul (63) Laserski skener #1 (80) Laserski skener #2 (81)

7.2. Sigurnosni PLC projekt

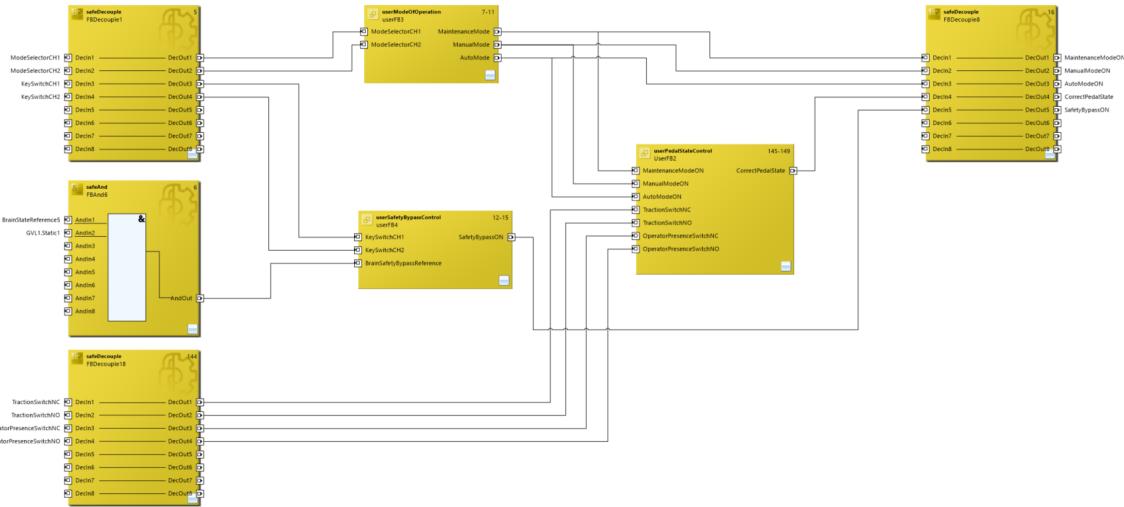
Programiranje unutar sigurnosnog PLC projekta moguće je isključivo korištenjem osnovnih logičkih operacija i funkcijskih blokova koji su već unaprijed definirani unutar TwinCAT 3 sigurnosnog razvojnog okruženja TwinSAFE. Uporabom tih jednostavnih funkcijskih blokova moguće je razviti sigurnosnu aplikaciju koja na temelju ulaznih vrijednosti stanja sigurnosnih tipkala i vrijednosti brojača enkodera postavlja izlaze koji dalje upravljaju relejima, sklopcicima, STO signalima i odabirom trenutne aktivne zone na sigurnosnim laserskim skenerima. Dijelovi sigurnosne logike mogu se grupirati u tzv. "korisničke funkcijске blokove", no logika unutar funkcijskih blokova se i dalje programira isključivo unaprijed definiranim logičkim operacijama koje nudi TwinSAFE razvojno okruženje.

U potpoglavlјima 7.1.1 do 7.1.4 izdvojen je i komentiran dio implementirane logike unutar sigurnosnog PLC projekta koji neposredno pridonosi sigurnosti samog vozila te ljudima i ostalim strojevima u okolini.

7.2.1. Praćenje stanja sigurnosnih tipkala

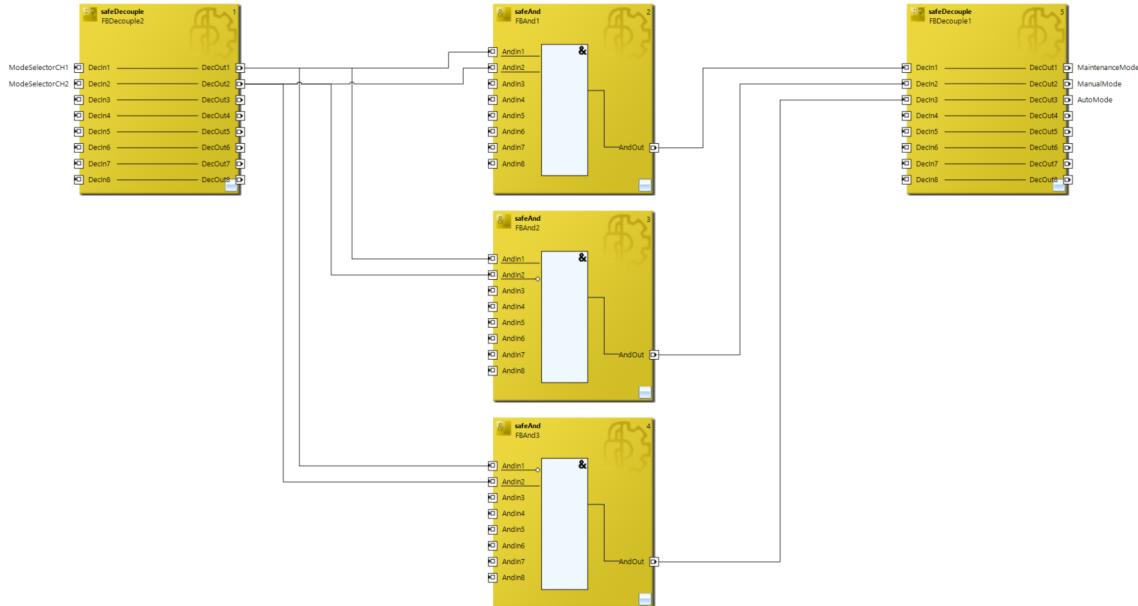
Sigurnosna tipkala na vozilu mogu se podijeliti na ona za odabir trenutnog aktivnog režima rada (ručni, autonomni ili servisni), ključ za privremeno isključivanje sigurnosnih zona, pedale koje ukazuju na prisutnost operatera na vozilu, tipkalo za potvrdu greške te tipkala za isklop u nuždi.

Na slici 7.1 prikazana je stranica gdje je implementirana logika za čitanje stanja sigurnosnih tipkala za odabir režima rada, provjeru stanja pedala na vozilu i ključa za privremeno isključivanje sigurnosnih zona.



Slika 7.1: Odabir režima rada, provjera stanja pedala i isključivanje sigurnosnih zona

Postavljanje trenutnog aktivnog režima rada vrši se u korisničkom funkcijskom bloku *userModeOfOperation* koji je prikazan na slici 7.2.

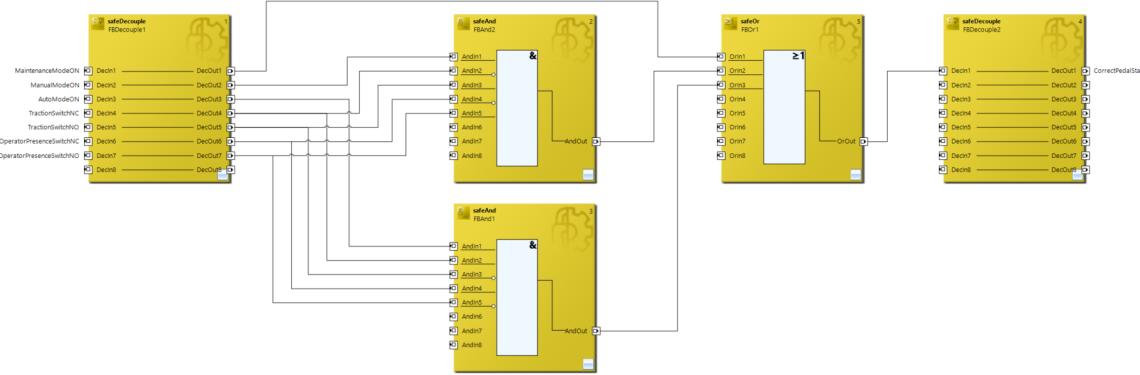


Slika 7.2: Korisnički funkcijski blok za odabir režima rada

Željeni režim rada odabire se dvokanalnom sklopkom koju je moguće postaviti u tri položaja, za tri režima rada.

Na vozilu se nalaze dvije pedale koje upućuju na prisutnost, odnosno odsutnost operatora. Pritiskom na pedalu aktivira se induktivni senzor koji je spojen na sigurnosne ulaze

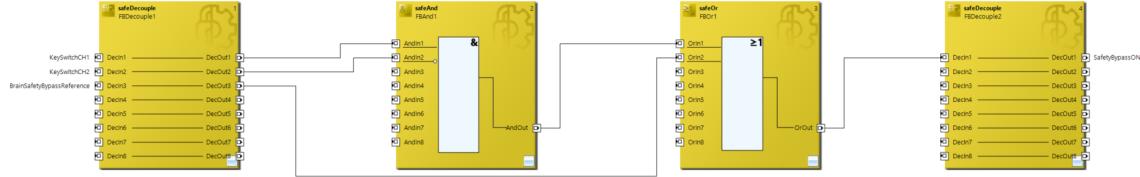
EL1918 modula. Induktivni senzor ima dva komplementarna izlaza, odnosno jedan NO (eng. *Normally Open*) i jedan NC (eng. *Normally Closed*) kontakt. Logika kojom se provjerava ispravno stanje pedala obzirom na odabrani režim rada implementirana je unutar korisničkog funkcijskog bloka *userPedalStateControl* koji je prikazan na slici 7.3.



Slika 7.3: Korisnički funkcijski blok za provjeru stanja pedala na vozilu

U ručnom režimu rada da bi se postavila zastavica *CorrectPedalState* u visoko, obje pedale moraju biti pritisnute, a u autonomnom režimu rada obje pedale moraju biti otpuštene.

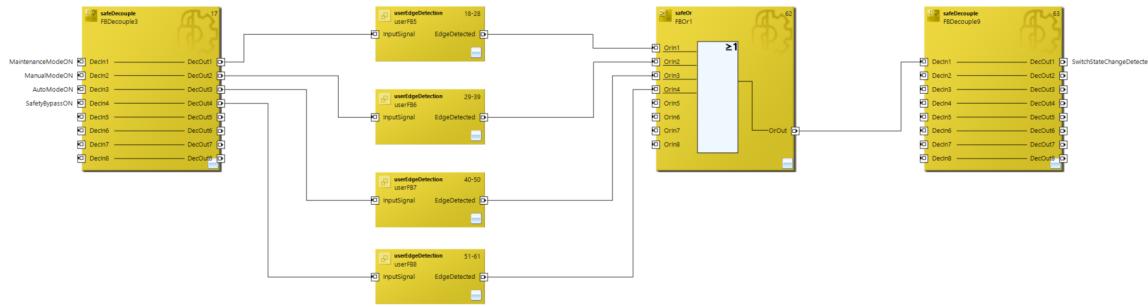
Logika za privremeno isključivanje zona implementirana je unutar korisničkog funkcijskog bloka *userSafetyBypassControl* prikazanog na slici 7.4.



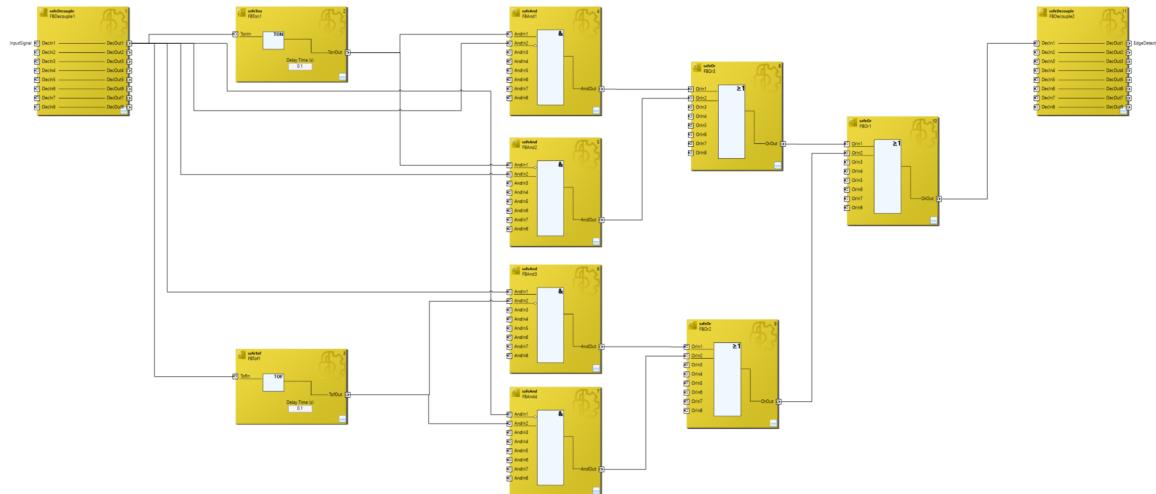
Slika 7.4: Korisnički funkcijski blok za isključivanje sigurnosnih zona

Sigurnosne zone moguće je privremeno isključiti ključem na vozilu ili iz visokorazinskog upravljačkog sustava zastavicom *BrainStateReference5*.

Uz praćenje stanja sigurnosnih tipkala, dodatno se postavlja zastavica *SwitchStateChangeDetected* koja služi za signalizaciju da je detektiran rastući ili padajući brid na tipkalu za odabir režima rada ili na ključu za isključivanje sigurnosnih zona. Ova logika prikazana je na slici 7.5. a u korisničkom funkcijskom bloku *userEdgeDetection* koji je prikazan na slici 7.6 implementirana je logika za detekciju rastućeg ili padajućeg brida.



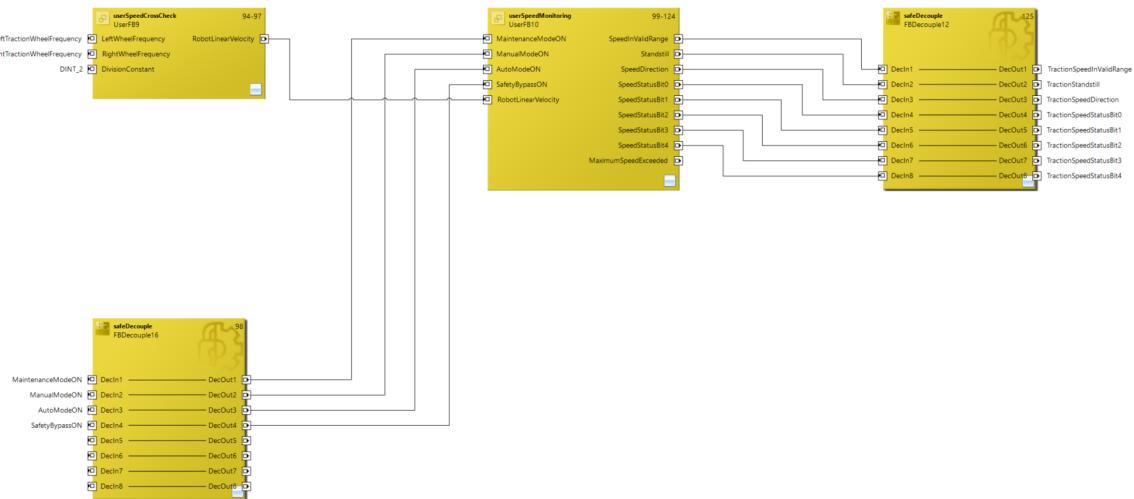
Slika 7.5: Detekcija promjene stanja tipkala za odabir režima rada i ključa za isključivanje sigurnosnih zona



Slika 7.6: Korisnički funkcijski blok za detekciju rastućeg ili padajućeg brida

7.2.2. Praćenje brzine robota

Na slici 7.7 prikazana je stranica gdje je implementirana sigurnosna logika za izračun i nadzor linearne brzine robota.

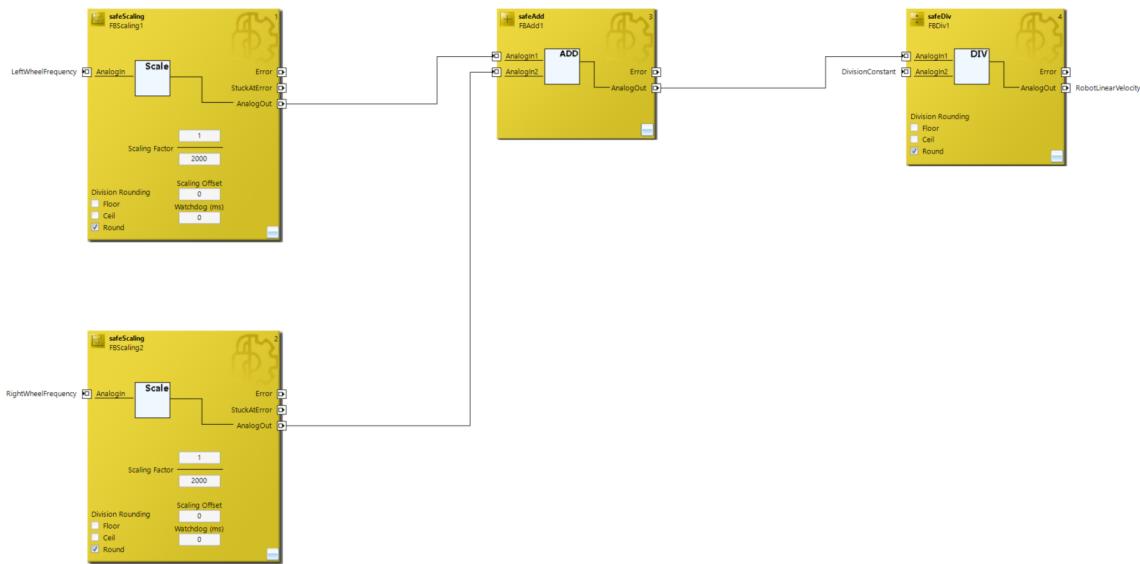


Slika 7.7: Izračun i nadzor linearne brzine robota

Linearna brzina računa se unutar funkcijskog bloka *userSpeedCrossCheck* prema izrazu:

$$v_{linearna} = \frac{v_{lijevi} + v_{desni}}{2} \quad (7.1)$$

a logika kojom je implementiran ovaj izraz prikazana je na slici 7.8.



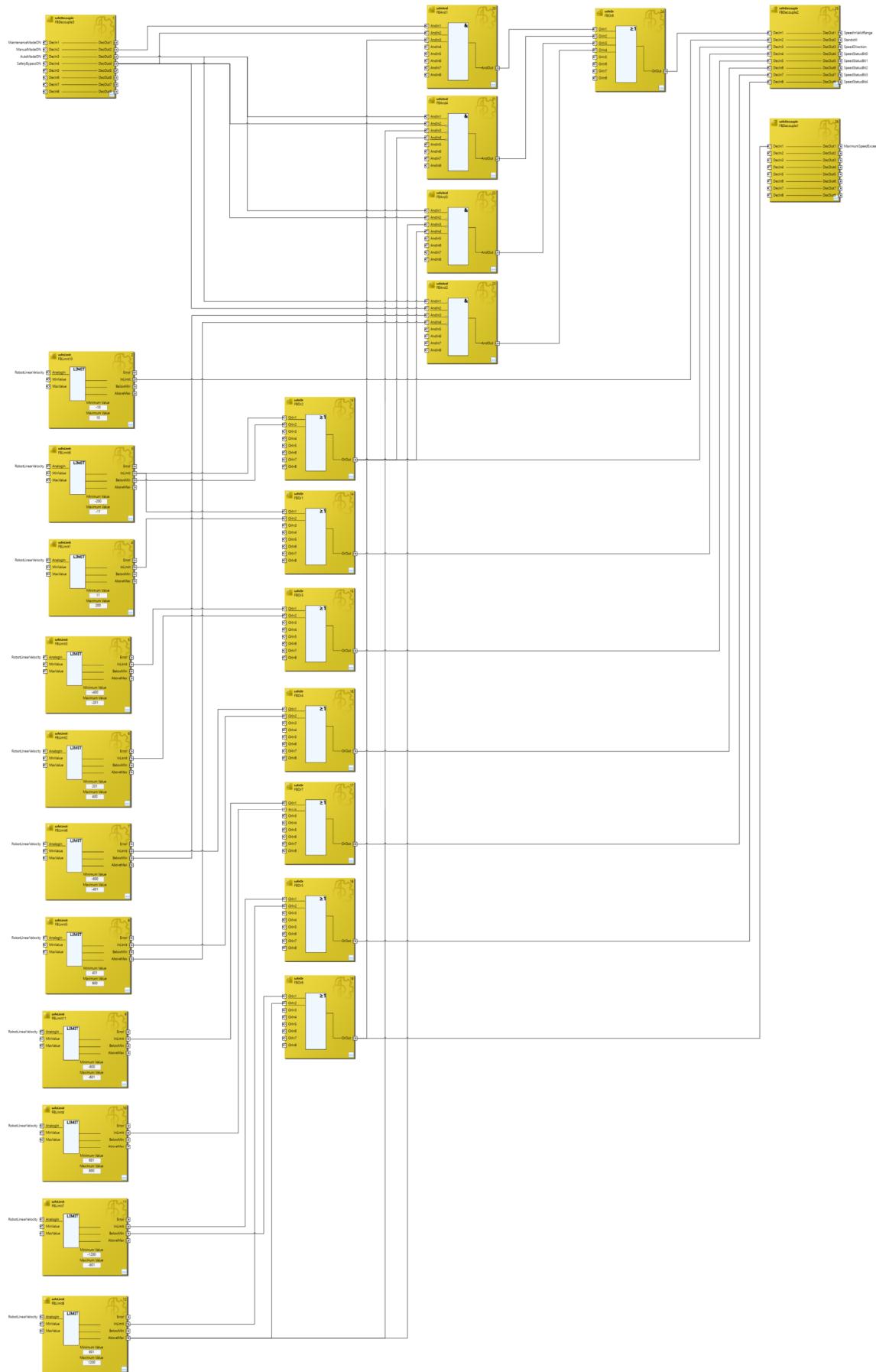
Slika 7.8: Korisnički funkcijski blok za izračun linearne brzine robota

Na ulaz bloka dovode se sirova mjerena sa sigurnosnih enkodera koji mjere brzinu lijevog i desnog pogonskog kotača. Mjerena se zatim skaliraju kako bi se dobila brzina pojedinog kotača u mm/s te zbrajaju i dijele s 2.

Izračunata linearna brzina dalje se proslijeđuje na ulaz korisničkog funkcijskog bloka *userSpeedMonitoring* prikazanog slikom 7.9. Unutar ovog bloka implementirana je logika za postavljanje dijagnostičkih zastavica za praćenje brzine:

- *SpeedInValidRange* - Zastavica za provjeru nalazi li se brzina unutar dozvoljenih granica za razne režime rada
- *SpeedDirection* - Zastavica za predznak brzine (0 - pozitivan smjer, 1 - negativan smjer)
- *Standstill* - Zastavica za slučaj kada je robot u mirovanju
- *SpeedStatusBit0...4* - Zastavice za određivanje u kojem je od postavljenih intervala trenutna brzina
- *MaximumSpeedExceeded* - Zastavica za provjeru je li prekoračena maksimalna dozvoljena brzina, neovisno o režimu rada

Intervali za postavljanje zastavica *SpeedStatusBit0...4* definirani su tablicom 7.3. Zastavica *Standstill* postavlja se uvijek za apsolutnu vrijednost brzine manje od 10 mm/s, dok se



Slika 7.9: Korisnički funkcijski blok za nadzor linearne brzine robota

Tablica 7.3: Uvjeti za postavljanje *SpeedStatusBit0..4* zastavica

Uvjet	Zastavica
$ v_{linearna} \in [10, 200] \frac{\text{mm}}{\text{s}}$	<i>SpeedStatusBit0</i>
$ v_{linearna} \in [200, 400] \frac{\text{mm}}{\text{s}}$	<i>SpeedStatusBit1</i>
$ v_{linearna} \in [400, 600] \frac{\text{mm}}{\text{s}}$	<i>SpeedStatusBit2</i>
$ v_{linearna} \in [600, 800] \frac{\text{mm}}{\text{s}}$	<i>SpeedStatusBit3</i>
$ v_{linearna} \in [800, 1200] \frac{\text{mm}}{\text{s}}$	<i>SpeedStatusBit4</i>

zastavica *MaximumSpeedExceeded* postavlja uvijek za apsolutnu vrijednost brzine veće od 1200 mm/s.

7.2.3. Određivanje aktivne sigurnosne zone

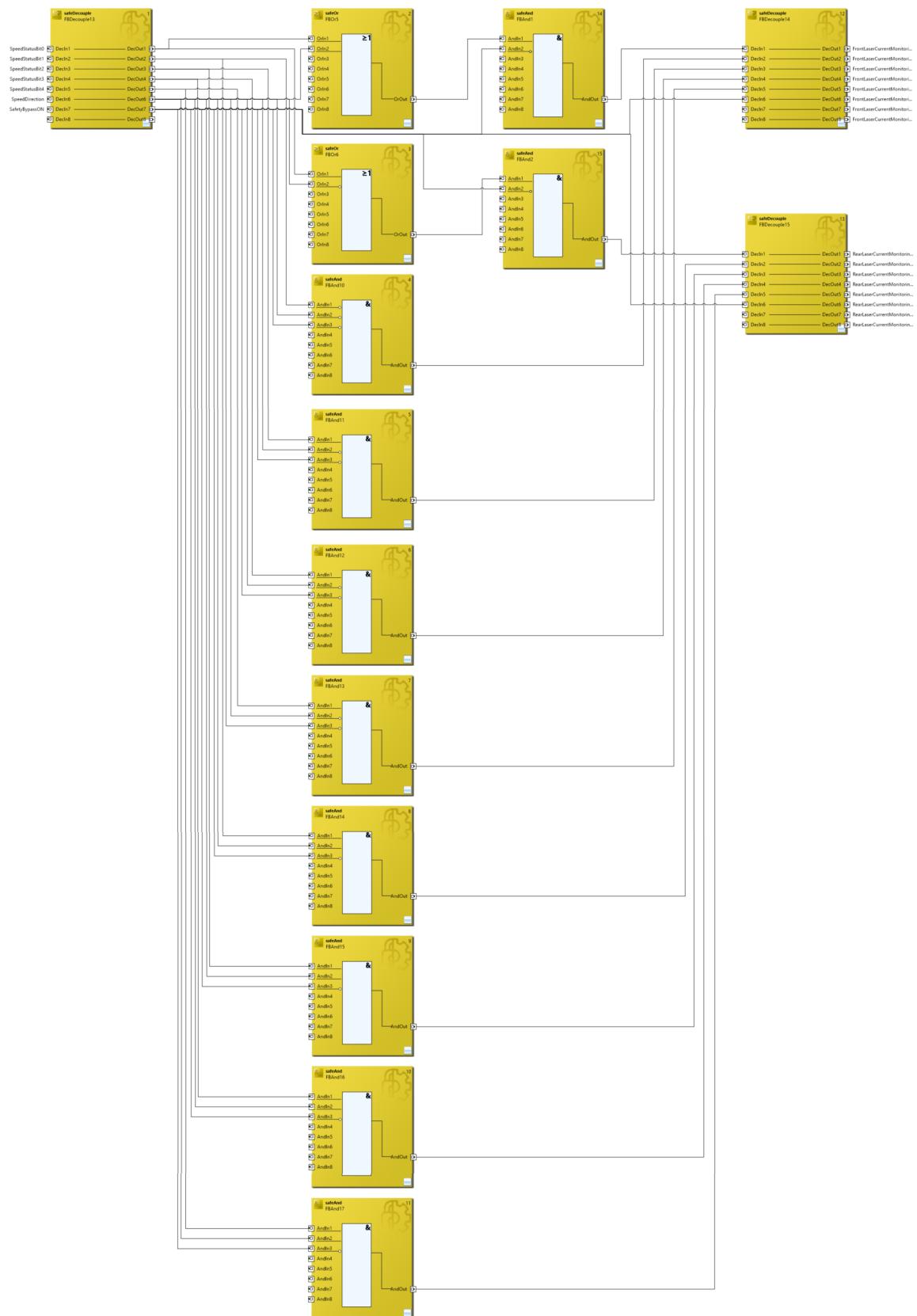
Na slici 7.10 prikazana je stranica gdje je se izvodi logika za određivanje aktivne sigurnosne zone.



Slika 7.10: Odabir aktivne sigurnosne zone

Radi jednostavnosti i zbog relativno malog broja implementiranih zona, odabir je definiran tako da svaka zona odgovara postavljanjem jednog bita u visoko u bajtu namijenjenom za odabir aktivne zone. Bitovi od nula do četiri koriste se za odabir zone, dok peti bit služi za isključivanje svih aktivnih zona oko robota.

Sigurnosna logika kojom se odabire aktivna zona implementirana je u korisničkom funkcijском блоку *userMonitorCaseSet* и приказана је slikом 7.11, а одабир се vrši obzirom на zastavicu *SpeedDirection* и zastavice *SpeedStatusBit0..4* које се постављају у корисниčком функцијском блоку за надзор linearne brzine из poglavља 7.1.2, dok zastavica *SafetyBypassON* služи за isključivanje zona. Logika korisničког функцијског блока *userMonitorCaseSet*



Slika 7.11: Korisnički funkcijski blok za odabir aktive sigurnosne zone

može se sažeti tablicama 7.4 i 7.5.

Tablica 7.4: Uvjeti za postavljanje aktivne zone na prednjem laserskom skeneru

Uvjet	Zona
$(SpeedStatusBit0 \vee SpeedDirection) \wedge \neg SafetyBypassON$	Zona 1
$(SpeedStatusBit1 \wedge \neg SpeedDirection) \wedge \neg SafetyBypassON$	Zona 2
$(SpeedStatusBit2 \wedge \neg SpeedDirection) \wedge \neg SafetyBypassON$	Zona 3
$(SpeedStatusBit3 \wedge \neg SpeedDirection) \wedge \neg SafetyBypassON$	Zona 4
$(SpeedStatusBit4 \wedge \neg SpeedDirection) \wedge \neg SafetyBypassON$	Zona 5
$SafetyBypassON$	-

Tablica 7.5: Uvjeti za postavljanje aktivne zone na stražnjem laserskom skeneru

Uvjet	Zona
$(SpeedStatusBit0 \vee \neg SpeedDirection) \wedge \neg SafetyBypassON$	Zona 1
$(SpeedStatusBit1 \wedge SpeedDirection) \wedge \neg SafetyBypassON$	Zona 2
$(SpeedStatusBit2 \wedge SpeedDirection) \wedge \neg SafetyBypassON$	Zona 3
$(SpeedStatusBit3 \wedge SpeedDirection) \wedge \neg SafetyBypassON$	Zona 4
$(SpeedStatusBit4 \wedge SpeedDirection) \wedge \neg SafetyBypassON$	Zona 5
$SafetyBypassON$	-

Kako su zastavice $SpeedStatusBit0..4$ međusobno isključive, ovom logikom osigurava se da nema konfliktnih slučaja gdje se odjednom odabiru dvije različite zone što bi uslijedilo pojmom greške na sigurnosnom laserskom skeneru i prestanku praćenja zona.

7.2.4. Sigurnosno zaustavljanje robota

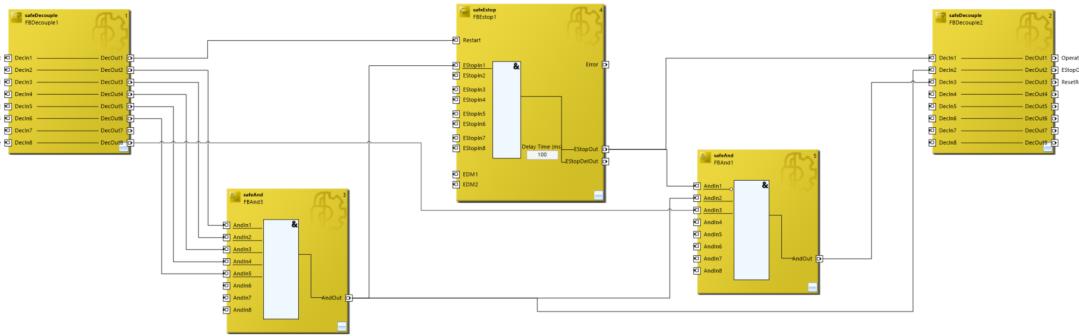
Na slici 7.12 prikazana je logika za dozvoljavanje prelaska robota u operativno stanje.

Na ulaz korisničkog funkcijskog bloka *userReset* dovode se stanja sigurnosnih tipkala za isklop u nuždi te stanja zastavica *SwitchStateChangeDetected*, *TractionSpeedInValidRange* i *CorrectPedalState*. Logika *userReset* funkcijskog bloka prikazana je na slici 7.13. Ako su sve zastavice i tipkala u odgovarajućem stanju, izlaz *OperationEnabled* postavlja se u visoko. Na izlaz funkcijskog bloka dovode se dodatno stanje tipkala za isklop u nuždi te informacija o tome je li potrebno pritisnuti na vozilu tipkalo za ponovno pokretanje.

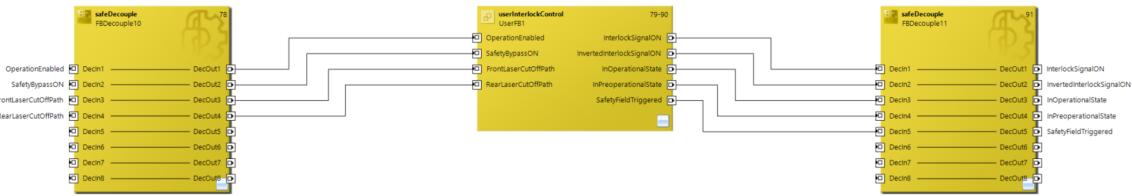
Zastavica *OperationEnabled* dalje se koristi za postavljanje STO signala i uklapanje kontaktora koji napajaju električne motore. Logika za ovu funkcionalnost implementirana je na stranici koja je prikazana na slici 7.14



Slika 7.12: Dozvoljavanje prelaska robota u operativno stanje

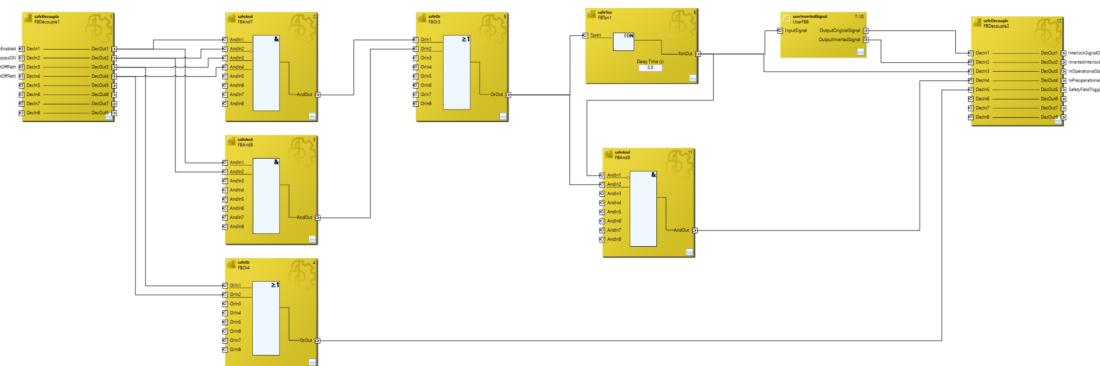


Slika 7.13: Korisnički funkcijski blok za dozvoljavanje prelaska robota u operativno stanje



Slika 7.14: Postavljanje STO signala i uklapanje kontaktora

i u korisničkom funkcijском блоку *userInterlockControl* који је приказан на слици 7.15.



Slika 7.15: Korisnički funkcijski блок за постављање STO сигналса и укљапање контактора

Također se provjerava stanje заштитnih zona, odnosno stanje zastavica *FrontLaserCutOffPath* i *RearLaserCutOffPath* које су постављене у ниско у slučaju ulaska čovjeka u sigurnosnu

zonusu koju nadzire taj laserski skener. STO signali se tada postavljaju u nisko te se robot brzo i sigurno zaustavlja. Nakon što se čovjek više ne nalazi u zoni STO signali se postavljaju u visoko i robot nastavlja sa zadatkom.

7.3. Standardni PLC projekt

Standardni PLC projekt obuhvaća više koraka, uključujući definiranje i organizaciju ulazno-izlaznih modula, programiranje logike upravljanja, konfiguriranje komunikacijskih parametara te testiranje i puštanje u rad.

U narednim odjeljcima ovog poglavlja, bit će opisana struktura standardnog PLC projekta, uključujući organizaciju programa, definiranje varijabli, upravljanje ulazima i izlazima, te obradu logičkih i matematičkih operacija.

Programiranje unutar standardnog PLC projekta moguće je svim jezicima definiranim IEC 61131-3 standardom. Glavni program pisan je u CFC (eng. *Continuous Function Chart*) programskom jeziku s programskim ciklusom izvođenja od 10 ms, a funkcionalni blokovi i funkcije pozivane u glavnom programu pisane su u ST (eng. *Structured Text*) programskom jeziku.

7.3.1. Upravljanje sustavom trakcije

Petlje za pogonske kotače i okretni kotač već su tvornički zatvorene pomoću internih enkoderova na viličaru spojenih na frekvencijske pretvarače preko PI regulatora. Parametri regulatora ostavljeni su na istim vrijednostima kakvima ih je proizvođač i postavio.

Kako su petlje već zatvorene preostaje još isprogramirati upravljački algoritam za raspodjelu brzine po pogonskim kotačima obzirom na kut okretnog kotača prema izrazima 5.5 i 5.6. Ovi izrazi implementirani su pomoću funkcije *Steer2Traction* unutar standardnog PLC projekta čije je zaglavje i programski kod prikazano kodom 7.1 i 7.2.

```
1 FUNCTION Steer2Traction : REAL
2 VAR CONSTANT
3   l           : REAL := 1109.918;    // Wheelbase in millimeters
4   w           : REAL := 864.6;      // Wheel track width in millimeters
5   o           : REAL := 190;        // Steering wheel offset in millimeters
6 END_VAR
7 VAR_INPUT
8   iSteeringWheelDeg   : REAL;          // Steering wheel position in degrees
9   iTractionSpeed      : REAL;          // Traction speed command
10 END_VAR
11 VAR
12   icc           : REAL;          // Instantaneous Center of Curvature
13   freeZone       : REAL := 2;        // Ignore zone for traction speed mapping
14 END_VAR
15 VAR_OUTPUT
16   oThrLeft      : REAL;          // Calculated output speed for left wheel
```

```

17   oThrRight      : REAL;           // Calculated output speed for right wheel
18 END_VAR

```

Kod 7.1: Zaglavje Steer2Traction funkcije

```

1 IF ABS(iSteeringWheelDeg) < freeZone THEN
2   oThrLeft := 0.5 * iTractionSpeed;
3   oThrRight := 0.5 * iTractionSpeed;
4 ELSIF iSteeringWheelDeg < freeZone THEN
5   icc := 1 / TAN(iSteeringWheelDeg * LREAL_TO_REAL(PI) / 180) - o;
6   oThrLeft := (-(icc + w / 2) / w) / (ABS(-(icc + w / 2) / w) + ABS((w / 2 - icc) / w
7     )) * iTractionSpeed;
8   oThrRight := ((w / 2 - icc) / w) / (ABS(-(icc + w / 2) / w) + ABS((w / 2 - icc) / w))
9     * iTractionSpeed;
10  ELSIF iSteeringWheelDeg > freeZone THEN
11    icc := 1 / TAN(iSteeringWheelDeg * LREAL_TO_REAL(PI) / 180) - o;
12    oThrLeft := ((icc + w / 2) / w) / (ABS((icc + w / 2) / w) + ABS(-(w / 2 - icc) / w))
13      * iTractionSpeed;
14    oThrRight := (-(w / 2 - icc) / w) / (ABS((icc + w / 2) / w) + ABS(-(w / 2 - icc) / w
15      )) * iTractionSpeed;
16 END_IF

```

Kod 7.2: Programska kod Steer2Traction funkcije

Ako je apsolutna vrijednost kuta okretnog kotača manja od 2° , tada se postavljaju iste brzine za lijevi $oThrLeft$ i desni $oThrRight$ kotač koja je jednaka vrijednosti $iTractionSpeed$.

U suprotnom se $oThrLeft$ i $oThrRight$ postavljaju prema izrazima 5.5 i 5.6.

7.3.2. Upravljanje osima za manevriranje vilicama

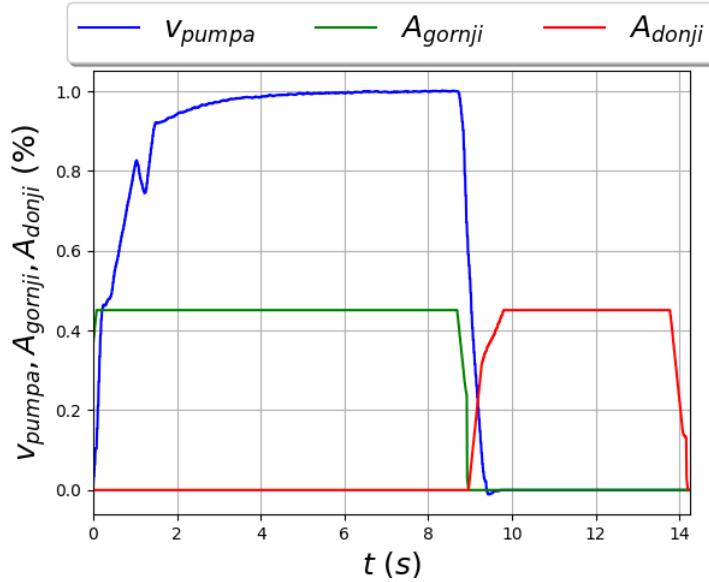
Pojednostavljena shema hidrauličnog sustava koji se koristi za manevriranje vilicama prikazana je na slici 2.3. Upravljačkim algoritmom potrebno je istovremeno upravljati hidrauličnom pumpom i ventilima koji usmjeravaju ulje prema cilindrima u pojedinoj osi. Hidrauličnoj pumpi zadaje se referentna vrijednost brzine čime se diže tlak u sustavu, a otvorenosću ventila upravlja se PWM modulacijom.

7.3.2.1. Otvorena petlja

Upravljanje u otvorenoj petlji osima za okomiti i bočni pomak te nagib vilica ujedno je i upravljanje u ručnom režimu rada gdje se vilicama upravlja pomoću komandne palice na vozilu.

Pri podizanju vilica otklon palice od 0% do 100% linearno odgovara postavljanju referentne vrijednosti brzine vrtnje hidrauličnoj pumpi u rasponu od 0 do 1100 okr./min, a otvorenost odgovarajućeg ventila postavlja se na 45%. Pri spuštanju vilica otvara se odgovarajući ventil za spuštanje vilica u vrijednosti od 20% do 45% obzirom na otklon komandne palice, a referentna vrijednost brzine hidraulične pumpe je naravno postavljena na nulu. Na slici 7.16 prikazani su odzivi za brzinu vrtnje hidraulične pumpe v_{pumpa} i otvorenost ventila

A_{gornji} i A_{donji} po sekvenci gdje se vilice prvo podižu pa zatim spuštaju pomoću komandne palice s maksimalnim otklonom. Brzina vrtnje hidraulične pumpe od 1100 okr./min



Slika 7.16: Odziv brzine vrtnje hidraulične pumpe i otvorenosti ventila za podizanje, odnosno spuštanje vilica u postotnim vrijednostima za upravljanje u otvorenoj petlji

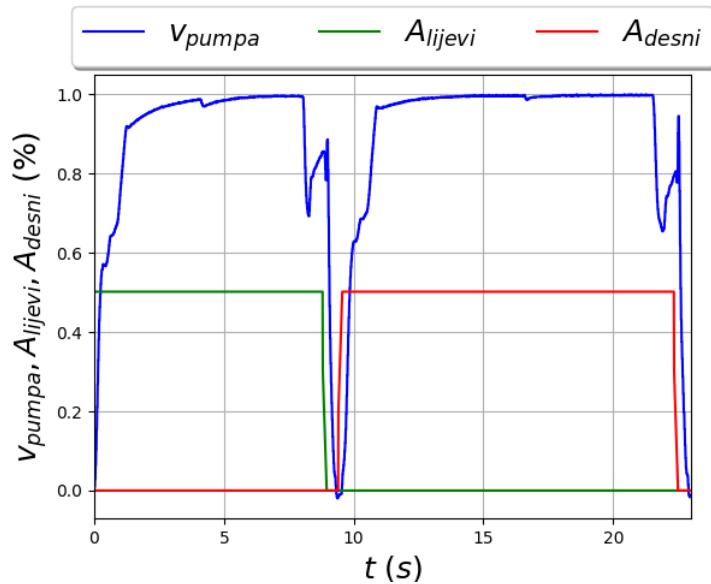
odgovara vrijednosti od 100%.

Za bočni pomak vilica vrijedi isti princip. Otklon palice od 0% do 100% linearno odgovara postavljanju referentne vrijednosti brzine hidrauličnoj pumpi u rasponu od 0 do 660 okr./min, a otvorenost odgovarajućeg ventila za bočni pomak vilica postavlja se na 50%. Na slici 7.17 prikazani su odzivi za brzinu vrtnje hidraulične pumpe v_{pumpa} i otvorenost ventila A_{lijevi} i A_{desni} po sekvenci gdje se vilice prvo pomiču ulijevo pa zatim udesno pomoću komandne palice s maksimalnim otklonom. Brzina vrtnje hidraulične pumpe od 660 okr./min odgovara vrijednosti od 100%.

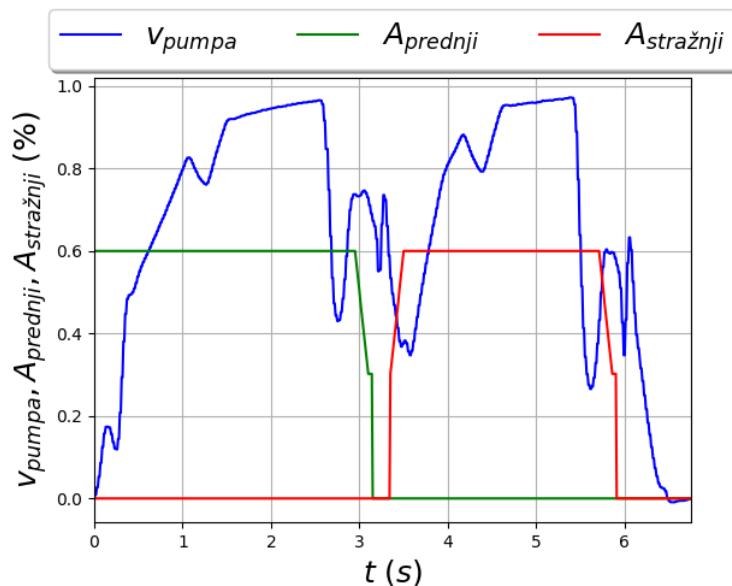
Isto tako i za nagib vilica, otklon palice od 0% do 100% linearno odgovara postavljanju referentne vrijednosti brzine hidrauličnoj pumpi u rasponu od 0 do 1100 okr./min, a otvorenost odgovarajućeg ventila za nagib vilica postavlja se na 60%. Na slici 7.18 prikazani su odzivi za brzinu vrtnje hidraulične pumpe v_{pumpa} i otvorenost ventila $A_{prednji}$ i $A_{stražnji}$ po sekvenci gdje se vilice prvo nagniju prema naprijed pa zatim prema natrag pomoću komandne palice s maksimalnim otklonom. Brzina vrtnje hidraulične pumpe od 1100 okr./min odgovara vrijednosti od 100%.

Zbog kratkog mehaničkog hoda nagiba vilica, pumpa ne dostigne maksimalnu vrijednost prije nego li se vilice nagnu u maksimalni položaj prema naprijed ili natrag.

U slučajevima gdje se istovremeno zadaju komande za pomak vilica u više osi, hidraulika će po principu manjeg otpora prvo vilice pomaknuti u onoj osi u kojoj je trenutno najmanje opterećenje. Iz opažanja, u slučaju gdje se zadaju komande za pomak u sve tri osi, vilice će



Slika 7.17: Odziv brzine vrtnje hidraulične pumpe i otvorenosti ventila za bočni pomak vilica ulijevo, odnosno udesno u postotnim vrijednostima za upravljanje u otvorenoj petlji



Slika 7.18: Odziv brzine vrtnje hidraulične pumpe i otvorenosti ventila za nagib vilica prema naprijed, odnosno prema natrag u postotnim vrijednostima za upravljanje u otvorenoj petlji

se prvo pomaknuti u osi za bočni pomak, zatim u osi nagiba te naposljetku u osi za okomiti pomak vilica.

7.3.2.2. Zatvorena petlja

Petlje za manevriranjem vilicama zatvorene su pomoću senzora spojenih na PLC module i logički program koristeći neku od inačica diskretnog PID regulatora koji je implementiran kao funkcijski blok unutar standardnog PLC projekta. Zaglavje i programske kodove *PIDController* funkcijskog bloka prikazani su kodom 7.3 i 7.4.

```
1 FUNCTION_BLOCK PIDController
2 VAR_INPUT
3   Enable      : BOOL;           // Enables execution of function block
4   ActualValue : REAL;          // Actual value
5   SetPoint    : REAL;          // Desired value
6   SetPointMAX : REAL;          // Maximum desired value
7   SetPointMIN : REAL;          // Minimum desired value
8   Tolerance   : REAL;          // Allowed error
9   Kp          : REAL;          // Proportional gain
10  Ki          : REAL;          // Integral gain
11  Kd          : REAL;          // Derivative gain
12  Tf          : REAL;          // Derivative constant
13  Ts          : REAL;          // Sample time NOTE: mind the PLC task cycle time
14  YMAX        : REAL;          // Maximum value of controller output
15  YMIN        : REAL;          // Minimum value of controller output
16 END_VAR
17 VAR
18   error       : REAL;          // Difference between SetPoint and ActualValue
19   prev_error  : REAL;          // Error in previous cycle
20   measurement : REAL;          // Equivalent to input ActualValue
21   prev_measurement : REAL;    // Measurement value in previous cycle
22   yp          : REAL;          // Proportional output value
23   yi          : REAL;          // Integral output value
24   yd          : REAL;          // Derivative output value
25 END_VAR
26 VAR_OUTPUT
27   Y           : REAL;          // Summed controller output value
28   Positioned  : BOOL;          // True if actual value is in range of given tolerance
29   LimitsActive : BOOL;        // True if output is saturated
30 END_VAR
```

Kod 7.3: Zaglavje *PIDController* funkcijskog bloka

```
1 IF Enable = TRUE THEN
2
3   //Setpoint constraint
4   IF SetPointMIN <> 0 OR SetPointMAX <> 0 THEN
5     IF SetPoint > SetPointMAX THEN
6       SetPoint := SetPointMAX;
7     ELSIF SetPoint < SetPointMIN THEN
8       SetPoint := SetPointMIN;
9     END_IF
10    END_IF
11
12  // Error calculation
13  error := SetPoint - ActualValue;
14
15  // Proportional output value calculation
```

```

16     yp := Kp * error;
17
18 // Integral output value calculation with conditional integration AW
19 IF LimitsActive = TRUE AND error * Y > 0 THEN
20     yi := 0;
21 ELSE
22     yi := yi + 0.5 * Ki * Ts * (error + prev_error);
23 END_IF
24
25
26 // Derivative output value calculation
27 yd := - (2 * Kd * (measurement - prev_measurement) +
28           (2 * Tf - Ts) * yd) / (2 * Tf + Ts);
29
30 // Controller output value calculation
31 IF ABS(Setpoint - ActualValue) < Tolerance THEN
32     Y := 0;
33     yp := 0;
34     yi := 0;
35     yd := 0;
36     Positioned := TRUE;
37 ELSE
38     Y := yp + yi + yd;
39     Positioned := FALSE;
40 END_IF
41
42 // Controller output constraintment
43 IF YMIN <> 0 OR YMAX <> 0 THEN
44     IF Y > YMAX THEN
45         Y := YMAX;
46         LimitsActive := TRUE;
47     ELSIF Y < YMIN THEN
48         Y := YMIN;
49         LimitsActive := TRUE;
50     ELSE
51         LimitsActive := FALSE;
52     END_IF
53 END_IF
54
55 // Storing current error and measurements values
56 prev_error := error;
57 prev_measurement := measurement;
58
59 ELSE
60
61 // Resetting controller parameters when function block is disabled
62 error := 0;
63 prev_error := 0;
64 measurement := 0;
65 prev_measurement := 0;
66 yp := 0;
67 yi := 0;
68 yd := 0;
69 Y := 0;
70 Positioned := FALSE;
71 LimitsActive := FALSE;

```

Kod 7.4: Programski kod *PIDController* funkcijskog bloka

U prvom djelu koda provjerava se je li željena referentna vrijednost u dozvoljenim granicama između minimalne i maksimalne dozvoljene vrijednosti te ako nije, vrijednost se zasićuje na granični iznos.

Zatim se računa greška između referentne vrijednosti i stvarne vrijednosti.

Proporcionalni izlazni dio y_p računa se kao umnožak proporcionalnog pojačanja K_p i greške.

Integralna izlazna vrijednost y_i izračunava se uz uvjetnu integraciju na temelju zastavice *LimitsActive* kako bi se spriječilo namatanje integratora. Ako su granice aktivne, a pogreška i izlaz imaju isti predznak, integralni član postavlja se na nulu [3]. U suprotnom, integralni član se izračunava korištenjem trapezoidnog pravila i pridodaje prethodnoj vrijednosti integrala.

Vrijednost izlazne derivacije y_d izračunava se na temelju trenutnog mjerjenja, prethodnog mjerjenja, pojačanja derivativnog djelovanja K_d i vremenske konstante T_f koja određuje niskopropusni filter prvog reda.

Izlaz regulatora Y izračunava se zbrajanjem proporcionalnog, integralnog i derivativnog djelovanja. Također se provjerava je li izlaz regulatora unutar navedenih minimalnih i maksimalnih vrijednosti te ako nije, zasićuje se i zastavica *LimitsActive* postavlja se u visoko. Uz to, dozvoljava se određeno odstupanje stvarne vrijednosti od referentne vrijednosti. Ako je to odstupanje manje od vrijednosti zapisane u varijable *Tolerance*, izlaz Y , zajedno s varijablama y_p , y_i i y_d , postavlja se u nulu.

Na kraju koda trenutna greška i trenutno mjerjenje pohranjuju se kao prethodna greška i prethodno mjerjenje za korištenje u sljedećoj iteraciji.

Za zatvaranje petlje za okomiti pomak vilica postavljeni su parametri regulatora prema tablici 7.6 do kojih se došlo eksperimentalnom metodom.

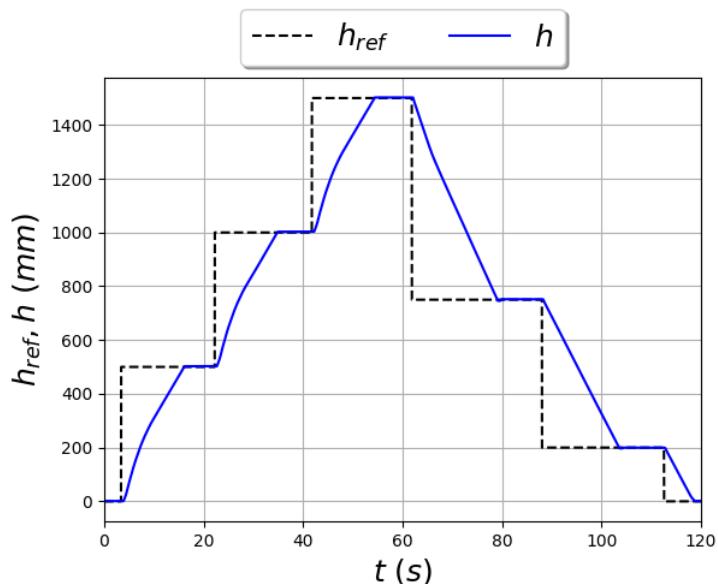
Tablica 7.6: Parametri PID regulatora za okomiti pomak vilica

T_s	0.01 s
K_p	0.05
K_i	1e-8
K_d	0
T_f	0
<i>Tolerance</i>	3 mm

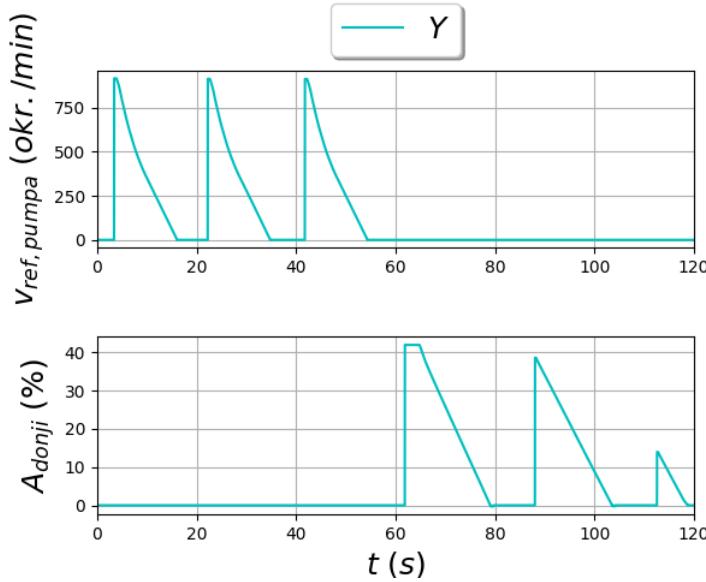
Povratna informacija o visini vilica dobivena je preko senzora iz tablica 3.13 i 3.20. Na

ulaz regulatora dovodi se samo vrijednost s enkodera jer je uočena puno manja zašumljenost mjernog signala, nego kod fotoelektričnog senzora koji se onda koristi kao dodatna provjera visine vilica. U slučaju odstupanja mjerena između ova dva senzora većeg od 2 centimetra, regulator se isključuje postavljanjem ulaza *Enable* u nisko.

Pri podizanju vilica odgovarajući ventil postavljen je na otvorenost od 45%, a izlaznom veličinom regulatora zapravo se upravlja referentnom vrijednosti za brzinu vrtnje hidraulične pumpe. Izlaz regulatora ograničen je na ± 30 gdje vrijednost od $Y = 30$ odgovara vrtnji pumpe od 1100 okr./min, dok pri spuštanju izlaz regulatora upravlja otvorenosću ventila za spuštanje vilica za koji vrijednost od $Y = -30$ odgovara otvorenosti od 42%. Na slici 7.19 prikazan je odziv visine vilica za upravljanje u zatvorenoj petlji pomoću PI regulatora, a na slici 7.20 prikazan je odziv izlaza regulatora Y po zadanoj sekventi $0 \text{ mm} \rightarrow 500 \text{ mm} \rightarrow 1000 \text{ mm} \rightarrow 1500 \text{ mm} \rightarrow 750 \text{ mm} \rightarrow 200 \text{ mm} \rightarrow 0 \text{ mm}$.



Slika 7.19: Odziv visine vilica za upravljanje u zatvorenoj petlji



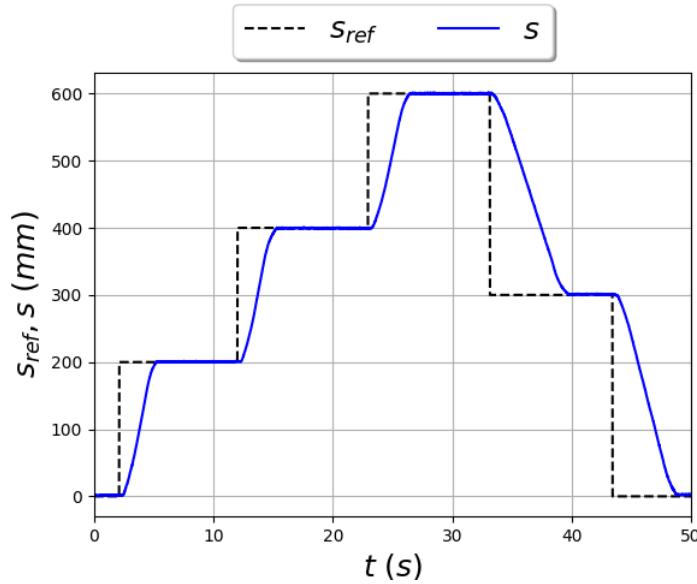
Slika 7.20: Izlaz regulatora za upravljanje visinom vilica u zatvorenoj petlji

Za zatvaranje petlje za bočni pomak vilica postavljeni su parametri regulatora prema tablici 7.6 do kojih se isto došlo eksperimentalnom metodom.

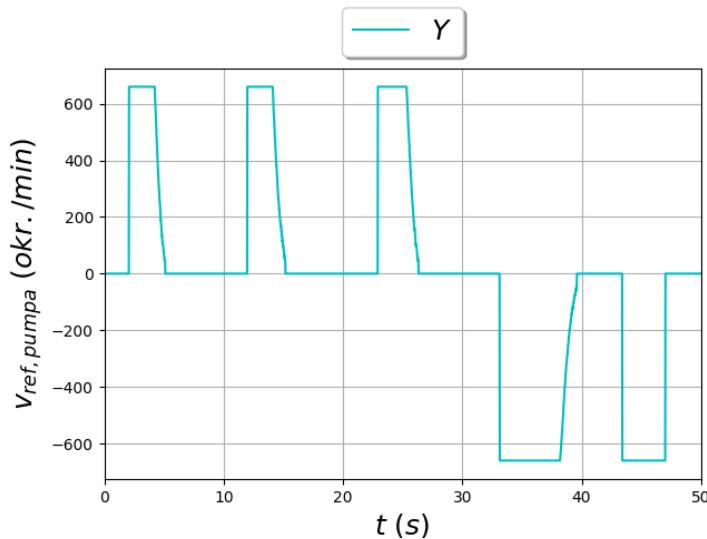
Tablica 7.7: Parametri PID regulatora za bočni pomak vilica

T_s	0.01 s
K_p	0.5
K_i	1e-4
K_d	0
T_f	0
<i>Tolerance</i>	3 mm

Pri bočnom pomaku vilica odgovarajući ventil postavljen je na otvorenost od 50%, a izlaznom veličinom regulatora upravlja se referentnom vrijednosti za brzinu vrtnje hidraulične pumpe. Izlaz regulatora ograničen je na ± 30 gdje ta vrijednost odgovara vrtnji pumpe od 1100 okr./min, a obzirom na predznak izlazne vrijednosti otvara se ventil za bočni pomak vilica ulijevo (pozitivan smjer kretanja vilica) ili bočni pomak vilica udesno (negativan smjer kretanja vilica). Na slici 7.21 prikazan je odziv pozicije bočnog pomaka vilica za upravljanje u zatvorenoj petlji pomoću PI regulatora, dok je na slici 7.22 prikazan odziv izlaza regulatora Y po zadanoj sekvenci $0 \text{ mm} \rightarrow 200 \text{ mm} \rightarrow 400 \text{ mm} \rightarrow 600 \text{ mm} \rightarrow 300 \text{ mm} \rightarrow 0 \text{ mm}$.



Slika 7.21: Odziv pozicije bočnog pomaka vilica za upravljanje u zatvorenoj petlji



Slika 7.22: Izlaz regulatora za upravljanje pozicijom bočnog pomaka vilica u zatvorenoj petlji

Mehanički hod bočnog pomaka vilica iznosi 700 milimetara, a za vrijednost pozicije od 0 mm vilice se nalaze u krajnje desnom položaju, a za vrijednost od 700 mm u krajnje lijevom položaju.

Petlju za nagib vilica se nažalost nije stiglo zatvoriti u vremenskom okviru trajanja zadatka ovog diplomskog rada zbog kašnjenja isporuke komponenata.

8. Zaključak

Kroz ovaj diplomski rad opisan je općeniti model viličara s pet stupnjeva slobode te je objašnjen princip rada električnog i hidrauličnog sustava na vozilu. Napravljen je pregled korištene opreme poput programabilnog logičkog kontrolera, sigurnosnih enkodera, laserskih skenera i ostalih senzora potrebnih za integraciju niskorazinskog upravljačkog sustava na platformu viličara. Opisani su korišteni komunikacijski protokoli kojima se povezuju frekvenički pretvarači na vozilu s upravljačkim sustavom te upravljački sustav sa senzorima. Izvedeni su izrazi potrebni za kvalitativno upravljanje sustavom trakcije poput proračuna raspodjele brzine po pogonskim kotačima te proračun ekvivalentnog kuta virtualnog okretnog kotača. Opisano je postavljanje i konfiguracija uređaja korištenih za upravljanje i sigurnosni nadzor sustava viličara. Definirane su zaštitne zone oko robota i uvjeti pod kojim se aktivira pojedina zona. Opisana je implementacija upravljačkog algoritma na PLC koji je razdijeljen na sigurnosni i standardni PLC projekt. U sigurnosnom PLC projektu implementirana je logika za praćenje linearne brzine i stanja samog robota, logika za dinamičko mijenjanje aktivnih zona na laserskim skenerima te logika za sigurnosno zaustavljanje robota za slučaj u nuždi. U standardnom PLC projektu implementirana je logika za upravljanje sustavom trakcije pomoću napravljenih proračuna te je opisana korištena metoda upravljanja osima za manipulaciju vilicama u otvorenoj i zatvorenoj petlji.

U radu je opisana temeljna upravljačka struktura niskorazinskog upravljačkog sustava koja se može dalnjim razvojnim naporima unaprijediti i time osigurati preciznije i brže odzive aktuatora u osima viličara. Potrebno je zatvoriti petlju za nagib vilica koju se nažalost zbog kašnjenja komponenata nije stiglo zatvoriti. Dodatno, kako bi se postigla veća razina pouzdanosti potrebno je implementirati mehanizme samoprovjere i obrade grešaka na robotu čime bi se upravljački sustav učinio otpornijim na nepoznate scenarije, a i povećala bi se produktivnost jer bi se smanjila potreba za ručnom intervencijom.

Predloženi sustav upravljanja također može poslužiti kao osnova za razvoj naprednijih robotskih sustava koji se mogu nositi sa složenijim zadacima i okruženjima.

LITERATURA

- [1] Felipe Espinosa, Ricardo García, Manuel Mazo, Elena López, i Raúl Mateos. Modeling and simulation of the kinematic and dynamic behavior of a fork-lift-truck. *IFAC Proceedings Volumes*, 1998.
- [2] Jing-Shan Zhao, Zhi-Jing Liu, i Jian Dai. Design of an ackermann type steering mechanism. *Journal of Mechanical Engineering Science*, 2013.
- [3] Šandor Ileš i Jadranko Matuško. Tehnike upravljanja u mehatronici. *Sveučilište u Zagrebu, Fakultet Elektrotehnike i Računarstva, materijali s predavanja*, 2021.

Upravljanje električnim viličarom s pet stupnjeva slobode pomoću programabilnog logičkog kontrolera

Sažetak

U ovom diplomskom radu opisan je općeniti model viličara s pet stupnjeva slobode. Napravljen je detaljan pregled korištene opreme potrebne za implementaciju niskorazinskog upravljačkog sustava. Opisani su korišteni komunikacijski protokoli za povezivanje uređaja. Napravljeni su potrebni proračuni za kvalitativno upravljanje sustavom trakcije. Definirane su zaštitne zone oko viličara te je isprogramirana sigurnosna logika za praćenje stanja vozila i zaustavljanje za slučaj u nuždi. Opisana je implementacija upravljačkog algoritma za manipulaciju osima viličara na programabilni logički kontroler te je ovaj predloženi sustav upravljanja eksperimentalno testiran i validiran.

Ključne riječi: električni viličar, upravljanje, mobilna robotika, programabilni logički kontroler, automatizacija procesa, sigurnosne funkcije

Control of electric forklift with five degrees of freedom using a programmable logic controller

Abstract

In this thesis a general model of a forklift with five degrees of freedom is described. An in-depth examination is made of the essential equipment required to implement a low-level control system. A brief description is made of the communication protocols employed for device connectivity. Necessary calculations are carried out for the effective control of the traction system. Protective zones around the forklift are defined and safety logic is programmed to monitor the condition of the vehicle and stop in case of emergency. The implementation of the control algorithm on a programmable logic controller for the manipulation of the forklift's movements is described, and this proposed control system is experimentally tested and validated.

Keywords: electric forklift, control, mobile robotics, programmable logic controller, process automation, safety functions