Implementacija modulacije prostornog vektora na CRIO FPGA platformi

Ugljar, Marjan

Master's thesis / Diplomski rad

2018

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **Josip Juraj Strossmayer University of Osijek, Faculty of Electrical Engineering, Computer Science and Information Technology Osijek / Sveučilište Josipa Jurja Strossmayera u Osijeku, Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:200:169966

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-07-27**

Repository / Repozitorij:

Faculty of Electrical Engineering, Computer Science and Information Technology Osijek

SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU

FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I INFORMACIJSKIH TEHNOLOGIJA OSIJEK

Sveučilišni studij

IMPLEMENTACIJA MODULACIJE PROSTORNOG VEKTORA NA CRIO FPGA PLATFORMI

Diplomski rad

Marjan Ugljar

Osijek, 2018.

FERIT
FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I INFORMACIJSKIH TEHNOLOGIJA OSIJEK

Obrazac D1: Obrazac za imenovanje Povjerenstva za obranu diplomskog rada

Osijek, 11.07.2018.

Odboru za završne i diplomske ispite

Imenovanje Povjerenstva za obranu diplomskog rada

Ime i prezime studenta:	Marjan Ugljar
Studij, smjer:	Diplomski sveučilišni studij Elektrotehnika
Mat. br. studenta, godina upisa:	D 1011, 24.09.2017.
OIB studenta:	03195130187
Mentor:	Doc.dr.sc. Marinko Barukčić
Sumentor:	Tin Benšić
Sumentor iz tvrtke:	
Predsjednik Povjerenstva:	Izv.prof.dr.sc. Denis Pelin
Član Povjerenstva:	Tin Benšić
Naslov diplomskog rada:	Implementacija modulacije prostornog vektora na CRIO FPGA platformi
Znanstvena grana rada:	**Elektrostrojarstvo (zn. polje elektrotehnika)**
Zadatak diplomskog rada:	Potrebno je teorijski opisati modulaciju prostornog vektora na trofaznom punoupravljivom izmjenjivaču. Posebno treba prikazati valne oblike trenutnih vrijednosti faznog i linijskog napona te opisati problematiku trećih harmonika kod modulacije prostornog vektora. Praktično implementirati algoritam na CompactRIO razvojni sustav u FPGA modulu koji za zadani referentni prostorni vektor generira upravljačke impulse za ventile izmjenjivača. (Rezervirano: Marijan Ugljar) (Tin Benšić)
Prijedlog ocjene pismenog dijela ispita (diplomskog rada):	Izvrstan (5)
Kratko obrazloženje ocjene prema Kriterijima za ocjenjivanje završnih i diplomskih radova:	Primjena znanja stečenih na fakultetu: 3 bod/boda Postignuti rezultati u odnosu na složenost zadatka: 3 bod/boda Jasnoća pismenog izražavanja: 3 bod/boda Razina samostalnosti: 3 razina
Datum prijedloga ocjene mentora:	11.07.2018.

Potpis mentora za predaju konačne verzije rada u Studentsku službu pri završetku studija:	Potpis:
	Datum:

FERIT
FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I INFORMACIJSKIH TEHNOLOGIJA **OSIJEK**

IZJAVA O ORIGINALNOSTI RADA

Osijek, 16.07.2018.

Ime i prezime studenta:	Marjan Ugljar
Studij:	Diplomski sveučilišni studij Elektrotehnika
Mat. br. studenta, godina upisa:	D 1011, 24.09.2017.
Ephorus podudaranje [%]:	0

Ovom izjavom izjavljujem da je rad pod nazivom**: Implementacija modulacije prostornog vektora na CRIO FPGA platformi**

izrađen pod vodstvom mentora Doc.dr.sc. Marinko Barukčić

i sumentora Tin Benšić

moj vlastiti rad i prema mom najboljem znanju ne sadrži prethodno objavljene ili neobjavljene pisane materijale drugih osoba, osim onih koji su izričito priznati navođenjem literature i drugih izvora informacija.
Izjavljujem da je intelektualni sadržaj navedenog rada proizvod mog vlastitog rada, osim u onom dijelu za koji mi je bila potrebna pomoć mentora, sumentora i drugih osoba, a što je izričito navedeno u radu.

Potpis studenta:

Sadržaj

1 UVOD **1**

2 TEORIJSKI PRIKAZ MODULACIJE PROSTORNOG VEKTORA **3**
2.1 Analiza rada trofaznog izmjenjivača u mosnom spoju 4
2.2 Dvoosna teorija 8
2.2.1 Prostorni vektor 9
2.2.2 Clarkova transformacija 10
2.2.3 Parkova transformacija 12
2.3 Algoritam prostorno-vektorske modulacije 13
2.4 Optimalan obrazac sklapanja ventila 18
2.5 Valni oblici faznog i linijskog napona kod prostorno-vektorske modulacije 20

3 PREGLED RAZLIČITIH MODULACIJSKIH TEHNIKA **22**
3.1 Six-Step od 120° 22
3.2 Pulsno-širinska modulacija 23
3.3 Napredne modulacijske tehnike 24

4 IMPLEMENTACIJA PROSTORNO-VEKTORSKE MODULACIJE **26**
4.1 Simulacija prostorno-vektorske modulacije u Matlab/Simulinku 26
4.2 Rezultati implementacije u Matlab/Simulinku 31

5 EKSPERIMENT I REZULTATI **38**
5.1 LabVIEW 38
5.2 CompactRio RTOS/FPGA platforma 39
5.2.1 Real-Time operacijski sustav 40
5.2.2 FPGA procesor 40
5.2.3 Komunikacija 42
5.3 Implementacija na cRio-9024 platformi 44
5.4 Rezultati implementacije na cRio-9024 platformi 54

6 ZAKLJUČAK **60**

Literatura **62**

Popis slika **66**

Popis tablica **67**

Sažetak **68**

Abstract **68**

Životopis **69**

PRILOG 1 - MATLAB SKRIPTA **70**

PRILOG 2 - RASPORED IZLAZA NI 9401 MODULA **72**

PRILOG 3 - ESTIMIRANE VRIJEDNOSTI UPOTREBE RESURSA KORIŠTENIH FUNKCIJA CRIO-9114 FPGA PROCESORA **73**

PRILOG 4 - NEPODRŽANE FUNKCIJE U SINGLE CYCLE TIMED LOOPU **74**

1. UVOD

Po svojoj pojavi, istosmjerni motori su imali najveću primjenu u industriju unatoč postojanju izmjeničnih motora zbog jednostavnosti upravljanja te unatoč svoj nepouzdanosti zbog trošnih dijelova kao što su četkice. Izmjenični motori su se smatrali motorima konstante brzine vrtnje iako su za njih postojale određene tehnike upravljanja brzinom vrtnje koje su bile kompliciranе i skupe za izvest. Takvo stanje se zadržalo sve do pojave poluvodičkih komponenti kada počinje razvoj modulacijskih tehnika. Pojavom sinusne pulsno-širinske modulacije izmjenični motori polako počinju zamjenjivati istosmjerne motore u industriji [1]. Pojava poluvodičkih komponenti te razvoj na njima baziranih pretvarača je bitno smanjila troškove implementacije i upravljanja izmjeničnim motorima [1]. Na osnovi sinusne pulsno-širinske modulacije su kasnije razvijane modulacijske tehnike koje su dodatno smanjivale gubitke te povećale isplativost primjene izmjeničnih motora [2]. Pulsno-širinska modulacija je bila preferirana modulacijska tehnika sve do razvoja modulacije prostornog vektora gdje su se vremena vođenja poluvodičkih komponenti određivala na osnovu prostornog vektora napona [3, 4]. Razvojem izmjenjivača su napravljene brojne varijacije na modulaciju prostornog vektora odnosno napravljena je njena prilagodba višerazinskim izmjenjivačima koji su danas sve aktualniji i sve češće primjenjivani [5, 6, 7, 8].

Tema ovog rada je razvoj algoritma prostorno-vektorske modulacije te implementacija istog na CompactRio platformu. Kroz rad će biti napravljena analiza prostorno-vektorske modulacije i razvoj algoritma za implementaciju modulacije na ranije spomenutu platformu te će algoritam biti poopćen kako bi se mogao primijeniti na bilo kojem ugradbenom sustavu.

U drugom poglavlju će biti objašnjen pojam prostornog vektora te će biti objašnjena Clarkova i Parkova transformacija kao jedan od koraka u implementaciji algoritma. Biti će prikazana analiza rada trofaznog izmjenjivača u mosnom spoju za kojeg se primarno koristi ova modulacijska tehnika. Nadalje je dana teorijska analiza prostorno-vektorske modulacije odnosno izveden je izraz za određivanje vremena vođenja upravljivih ventila izmjenjivača, predložen je optimalan obrazac sklapanja ventila u svrhu smanjenja gubitaka te su na kraju dani valni oblici linijskog i faznog napona koje generira izmjenjivač uz objašnjenje pojave trećeg harmonika napona.

U trećem poglavlju je obavljen pregled modulacijskih tehnika. Uz Six-Step modulaciju i modulaciju prostornog vektora koje su objašnjenje u drugom poglavlju, dodatno je pojašnjen princip rada jedne varijante Six-Stepa, sinusno pulsno-širinske modulacije te naprednih modulacijskih tehnika u vidu višerazinske modulacije prostornog vektora.

Četvrto poglavlje se bavi implementacijom razvijenog algoritma u Matlab/Simulink programu. Dan je opis podsustava koji su implementirani prema razvijenom algoritmu. Prikazani su generirani upravljački impulsi odnosno obrasci sklapanja te valni oblici linijskog i faznog napona za tri sklopne frekvencije koje generira simulirani trofazni izmjenjivač u mosnom spoju.

Peto poglavlje opisuje korišteni hardver i softver odnosno LabView i cRio-9024 RTOS/FPGA platformu. Dan je kratak pregled programskog paketa LabView te su detaljno objašnjene

mogućnosti Real-Time operacijskog sustava cRio platforme. Nadalje je objašnjen FPGA procesor kao i FPGA tehnologija te način komunikacije između Real-Time i FPGA procesora. Objašnjen je svaki podsustav implementiran prema algoritmu na cRio platformu kao i neke preinake i prilagodbe koje su učinjene naspram implementiranog algoritma u Matlab/Simulinku i koje su specifične za cRio platformu. Na kraju su dani rezultati implementacije u vidu generiranih impulsa na izlaznim modulima cRio sustava te analiza dobivenih rezultata.

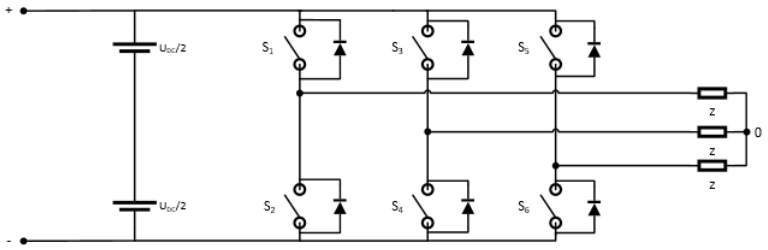
2. TEORIJSKI PRIKAZ MODULACIJE PROSTORNOG VEKTORA

Trofazni naponski izmjenjivač pronalazi mnoge primjene u današnjim tehnologijama npr. u obnovljivim izvorima energije gdje postoji potreba pretvaranje istosmjernog napona i struje u izmjenični [9, 10]. Nalaze još široku primjenu u frekvencijskim pretvaračima, neprekidnim izvorima napajanja, regulaciji djelatne i jalove snage, poboljšanju dinamičkih performansi mreže i električnim automobilima [11, 12]. Za upravljanje izmjenjivačima razvile su se razne modulacijske tehnike od kojih najbolje performanse daju pulsno-širinske modulacijske tehnike kao što su sinusna pulsno-širinska modulacija, sinusna pulsno-širinska modulacija s injektiranjem trećeg harmonika i prostorno-vektorska modulacija [1, 13, 14]. Najbolje karakteristike pokazuje prostorno-vektorska modulacija koja, naspram ostalih modulacijskih tehnika, pruža bolju iskoristivost napona istosmjernog međukruga, ima niži harmonijski spektar zahvaljujući unaprijed poznatom broju sklopnih stanja izmjenjivača iz čega proizlaze niži sklopni gubitci, digitalno je jednostavnija za implementirati naspram ostalih modulacijskih tehnika te funkcionira dobro sa tehnikama vektorskog upravljanja koje se primjenjuju u električnim automobilima [3, 4, 15, 16].

Prije pojave pulsno-širinskih modulacijskih tehnika i poluvodičkih komponenti izmjenični motori su imali fiksnu brzinu vrtnje čija je brzina bila određena frekvencijom napona. Postojale su određene tehnike upravljanja brzinom vrtnje kao što je promjena broja pari polova no bile su komplicirane i skupe tako da su istosmjerni motori prevladavali u industriji [1]. 1964. Schönung i Stemmler među prvima predlažu sinusno pulsno-širinsku modulaciju nakon čega ona postaje preferirana modulacijska tehnika sve do pojave tehnika koje su zasnovane na njoj kao što je ranije spominjana sinusno pulsno-širinska modulacija s injektiranjem trećeg harmonika [1, 13]. 1982. godine Busse i Holtz predlažu modulacijsku tehniku koju su zvali kutno-amplitudna modulacija (*engl. Angle and Amplitude Modulation*) koja se bazirala na prostornom vektoru izvedenom iz trofaznog sinusnog napona [17]. Holtz, Lammert i Lotzkat 1987. nazivaju tu tehniku prostorno-vektorska modulacija te je detaljnije objašnjavaju u [18]. Van der Broeck, Skudelny i Lotzkat 1988. godine daju detaljnu analizu prostorno-vektorske modulacije te je prvi implementiraju i daju usporedbu sa sinusno pulsno-širinskom modulacijom [14]. U zadnjim godinama se učinjeni daljnji pomaci u razvijanju prostorno-vektorske modulacije u smislu da se izrađuju modulacijske tehnike bazirane na njoj za višerazinske izmjenjivače.

2.1. Analiza rada trofaznog izmjenjivača u mosnom spoju

Kao što je ranije rečeno, trofazni izmjenjivač nalazi brojne primjene kao što su obnovljivi izvori energije, električni automobili, frekventni pretvarači itd. Trofazni izmjenjivač u mosnom spoju se može dobiti spajanjem tri jednofazna izmjenjivača u polumosnom spoju. Sastoji se od 6 sklopki (dvije u istoj grani) i svaka sklopka se sastoji od 2 poluvodičke komponente, jedna se može modelirati pomoću upravljivog ventila (MOSFET, IGBT), a druga je neupravljiva (dioda). Nadomjesna shema trofaznog izmjenjivača u mosnom spoju je prikazana na slici 2.1.



Sl. 2.1: *Nadomjesna shema trofaznog izmjenjivača u mosnom spoju*

Trofazni izmjenjivač sa slike ima 8 sklopnih stanja koja su prikazana u tablici 2.1.

Tab. 2.1: *Sklopna stanja*

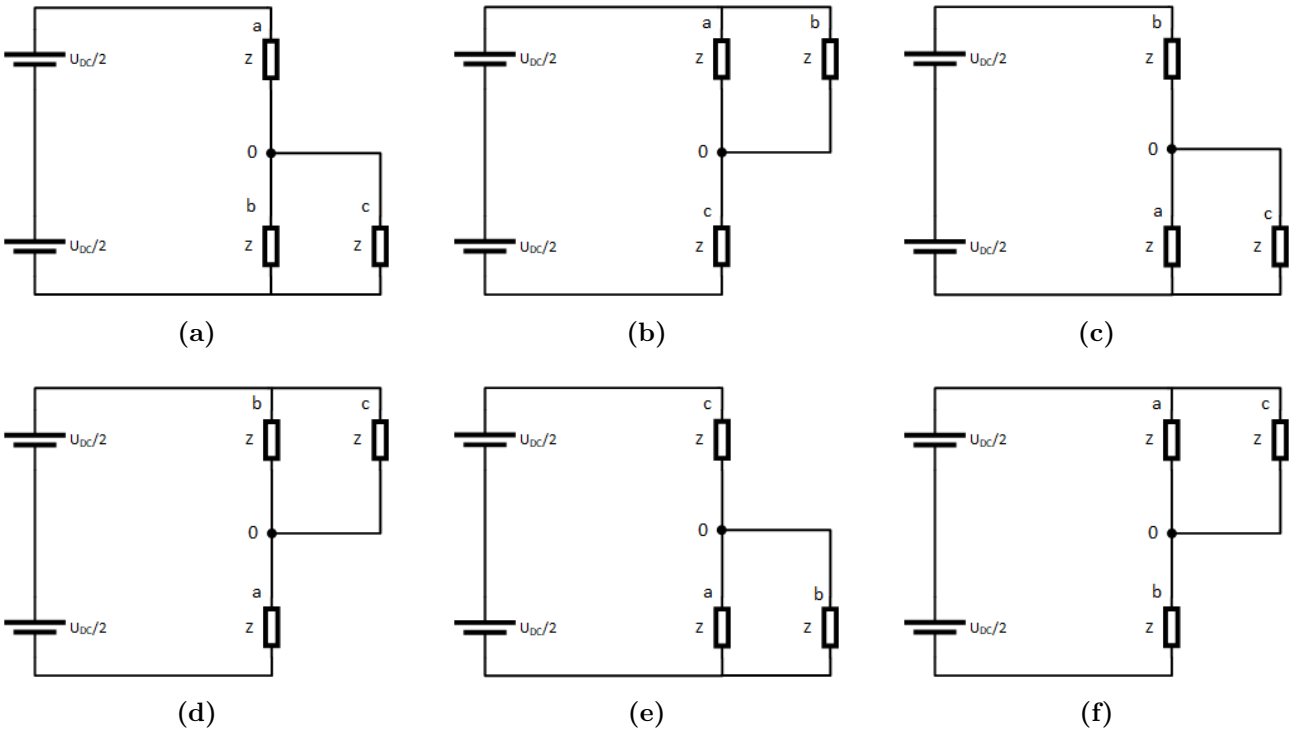
	V_0	V_1	V_2	V_3	V_4	V_5	V_6	V_7
S_1	0	1	1	0	0	0	1	1
S_3	0	0	1	1	1	0	0	1
S_5	0	0	0	0	1	1	1	1

S_1, S_3 i S_5 su tzv. gornje ili pozitivne sklopke izmjenjivača, dok su S_2, S_4 i S_6 donje ili negativne sklopke koje su komplementarne gornjima. V_0 - V_7 predstavljaju svako moguće sklopno stanje izmjenjivača prikazanog na slici 2.1. Svako sklopno stanje V_n odgovara jednom prostornom vektoru napona U_n gdje je $n = 0, 1, 2, ..., 7$.

Analiza rada izmjenjivača će biti urađena za Six-Step ili pravokutnu modulaciju gdje će svaki ventil voditi pola periode te se koriste samo sklopna stanja V_1 - V_6. Za trošilo je pretpostavljen asinkroni motor u zvijezda spoju gdje su otpor i induktivitet namota modelirani pomoću impedancije Z (Sl. 2.1).

Nacrta li se nadomjesna shema za svako sklopno stanje dobije se slika 2.2.

Prema slici 2.2 je moguće izračunati fazne i linijske napone trošila gdje su fazni naponi



Sl. 2.2: *Nadomjesna shema za sklopna stanja: a) U_1, b) U_2 c) U_3 d) U_4 e) U_5 f) U_6*

naponi između nule trošila i pojedine faze, a linijski naponi:

$$\begin{aligned} U_{ab} &= U_a - U_b \\ U_{bc} &= U_b - U_c \\ U_{ca} &= U_c - U_a \end{aligned} \tag{2-1}$$

Tablica 2.2 prikazuje iznose faznih i linijskih napona u slučaju simetričnog trošila. U_a, U_b i U_c su fazni naponi, a U_{ab}, U_{bc} i U_{ca} su linijski naponi.

Tab. 2.2: *Fazni i linijski naponi za pojedino sklopno stanje*

Sklopno stanje	$\boldsymbol{U_a}$	$\boldsymbol{U_b}$	$\boldsymbol{U_c}$	$\boldsymbol{U_{ab}}$	$\boldsymbol{U_{bc}}$	$\boldsymbol{U_{ca}}$
V_1	$\frac{2}{3}U_{DC}$	$-\frac{U_{DC}}{3}$	$-\frac{U_{DC}}{3}$	U_{DC}	0	$-U_{DC}$
V_2	$\frac{U_{DC}}{3}$	$\frac{U_{DC}}{3}$	$\frac{2}{3}U_{DC}$	0	U_{DC}	$-U_{DC}$
V_3	$-\frac{U_{DC}}{3}$	$\frac{2}{3}U_{DC}$	$\frac{U_{DC}}{3}$	$-U_{DC}$	U_{DC}	0
V_4	$-\frac{2}{3}U_{DC}$	$\frac{E}{3}$	$\frac{U_{DC}}{3}$	$-U_{DC}$	0	U_{DC}
V_5	$-\frac{U_{DC}}{3}$	$-\frac{U_{DC}}{3}$	$\frac{2}{3}U_{DC}$	0	$-U_{DC}$	U_{DC}
V_6	$-\frac{U_{DC}}{3}$	$-\frac{2}{3}U_{DC}$	$\frac{U_{DC}}{3}$	U_{DC}	$-U_{DC}$	0

Prema tablici 2.2 je sada moguće nacrtati valne oblike napona trošila. Slika 2.3 prikazuje

slijed upravljačkih signala, te valne oblike linijskog i faznog napona trofaznog izmjenjivača u mosnom spoju [38, 39, 40]. S_1 - S_6 su upravljački impulsi upravljivih ventila izmjenjivača. Nadalje prema faznim naponima na slici 2.3 da se izračunati da je efektivna vrijednost faznog napona [39]:

$$U = \frac{2}{3}U_{DC} \tag{2-2}$$

S obzirom da je asinkroni motor simetrično trošilo odnosno vrijedi jednadžba (2-9) koja također vrijedi i za osnovne harmonike faznih i linijski napona te stoga i oni tvore trofazni simetrični sustav, za efektivnu vrijednost osnovnog harmonika faznog napona će vrijediti [39]:

$$U_i(1) = \frac{U_{ij}(1)}{\sqrt{3}} = \frac{\sqrt{2}}{3}U_{DC}, \quad i = a, b, c; j = a, b, c; i \neq j \tag{2-3}$$

Općenito prema [41], fazni naponi sa slike 2.3 rastavom u Fourierov red se mogu zapisati kao:

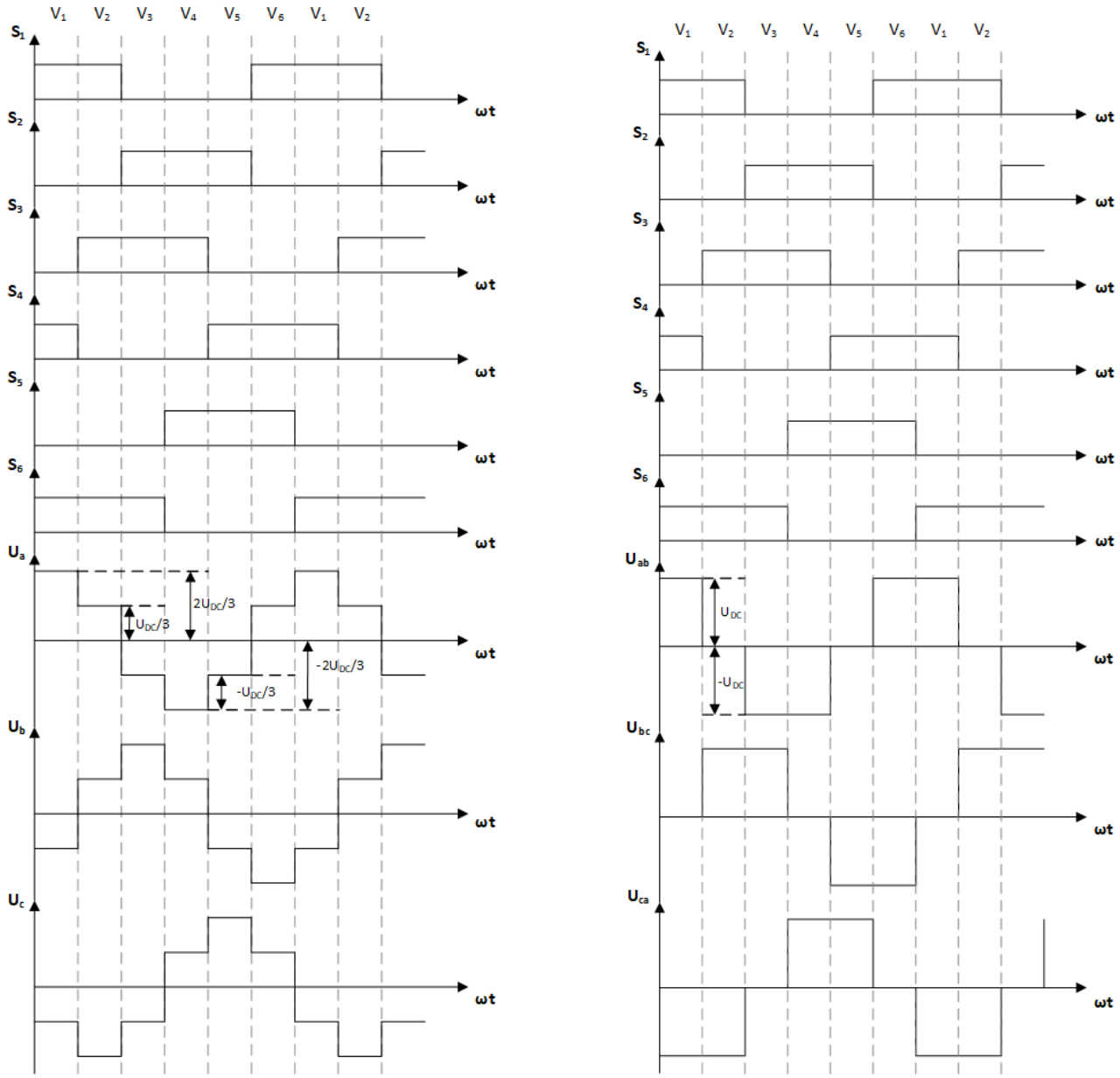
$$\begin{aligned} u_a &= \frac{2U_{DC}}{\pi}[\frac{1}{k}\cos(k\omega t)] \\ u_b &= \frac{2U_{DC}}{\pi}[\frac{1}{k}\cos(k(\omega t - \frac{2\pi}{3}))], \quad k = 6n \pm 1 \\ u_c &= \frac{2U_{DC}}{\pi}[\frac{1}{k}\cos(k(\omega t + \frac{2\pi}{3}))] \end{aligned} \tag{2-4}$$

a linijski naponi kao:

$$\begin{aligned} u_{ab} &= \frac{2U_{DC}}{\sqrt{3}\pi}[\frac{1}{k}\cos(k\omega t)] \\ u_{bc} &= \frac{2U_{DC}}{\sqrt{3}\pi}[\frac{1}{k}\cos(k(\omega t - \frac{2\pi}{3}))], \quad k = 6n \pm 1 \\ u_{ca} &= \frac{2U_{DC}}{\sqrt{3}\pi}[\frac{1}{k}\cos(k(\omega t + \frac{2\pi}{3}))] \end{aligned} \tag{2-5}$$

gdje je n cijeli broj odnosno $n = 1, 2, 3, ...$, a k red harmonika koji se javljaju u izlaznom naponu trofaznog izmjenjivača u mosnom spoju.

S obzirom da se svako sklopno stanje trofaznog izmjenjivača u mosnom spoju može prikazati prostornim vektorom te da je to osnova modulacije prostornog vektora, u nastavku će biti objašnjen pojam prostornog vektora, kao i njegov kompleksni oblik do kojeg se dolazi pomoću Clarkove i Parkove transformacije.



Sl. 2.3: *Slijed upravljačkih signala i: a) Fazni naponi, b) Linijski naponi*

2.2. Dvoosna teorija

Dvoosna teorija je nastala iz potrebe da se uklone svi induktiviteti ovisni o položaju rotora iz jednadžbi koje se koriste za analizu dinamičkih stanja sinkronih i asinkronih strojeva [19]. 1929. R. H. Park [20] je predložio upravo taj pristup analizi električnih strojeva. Predložio je transformaciju statorskih veličina sinkronog stroja u novi dvoosni koordinatni sustav koji je fiksiran za os rotora te prema tome rotira brzinom rotora. Upravo je ta transformacija, danas poznata kao Parkova transformacija, uklonila sve induktivitete ovisne o položaju rotora iz naponskih jednadžbi sinkronog stroja.

H. C. Stanley [21] je 1938. predložio sličnu transformaciju na asinkronom stroju. Pokazao je da promjenjivi međuinduktiviteti u naponskim jednadžbama asinkronog stroja mogu biti eliminirani transformacijom varijabli vezanih za rotorski namot u fiktivne varijable koje bi bile vezane za statorski namot. Drugačije rečeno, rotorski koordinatni sustav je transformiran u koordinantni sustav vezan za stator. Danas je ta transformacija poznata kao Clarkova transformacija.

G. Kron [22] je 1951. predložio transformaciju i statorskih i rotorskih varijabli simetričnog asinkronog stroja u referentni okvir koji rotira sinkronom brzinom.

Brereton, Lewis i Young [23] su 1957. primijenili Parkovu transformaciju na asinkroni stroj odnosno, transformirali su statorske varijable u referentni okvir fiksiran za os rotora koji prema tome rotira brzinom rotora.

Do 1965. su se sve gore navedene transformacije u literaturi tretirale odvojeno i smatrale su se svojstvenim za određene primjene. Tek su Krause i Thomas [24] prepoznali da su sve transformacije koje se koriste u analizi asinkronih strojeva sadržane u jednoj općoj transformaciji koja eliminira sve međuinduktivitete ovisne o položaju rotora. Ta se transformaciji mogla vršiti transformacijom statorskih i rotorskih varijabli u referenti okvir koji može rotirati proizvoljnom brzinom ili mirovati. Sve do tada poznate transformacije su se jednostavno mogle izvesti korištenjem prikladne brzine vrtnje i transformacijom varijabli u tzv. proizvoljni referentni okvir. Ista transformacija se mogla primijeniti i na sinkroni stroj [25]. No uočeno je da se promjenjivi induktiviteti kod sinkronog stroja mogu eliminirati samo primjenom Parkove transformacije odnosno transformacijom statorskih veličina u referentni okvir koji rotira brzinom rotora [19].

2.2.1 Prostorni vektor

Trofazni sustav može se zapisati pomoću jednog prostornog vektora. Koncept prostornog vektora je razvijen iz teorije trofaznih električnih strojeva gdje su svi statorski namoti međusobno razmaknuti za 120° [26, 27].

Prostorni vektor je istodobni prikaz sve tri fazne veličine te je funkcija vremena. Nacrta li se trofazni koordinatni sustav s osima a,b i c te tim osima pripadajućim jediničnim vektorima u_a°, u_b° i u_c°. Vrijedi da je svaki vektor koji se nalazi u definiranom koordinatnom sustavu linearna kombinacija ta tri jedinična vektora. Rezultanti vektor se može pisati kao zbroj njegovih komponenti:

$$\vec{u}_s = \frac{2}{3}(\vec{u}_a + \vec{u}_b + \vec{u}_c) \tag{2-6}$$

Konstanta $\frac{2}{3}$ se uvodi radi očuvanja amplitude vektora. Naime vektorski zbroj $\vec{u}_a, \vec{u}_b$ i $\vec{u}_c$ daje rezultantni vektor čija je amplituda 1.5 veća od stvarnih veličina, primjeni li se to npr. na veličine toka, napona i struje električnog stroja. S obzirom da su komponente rezultantnog vektora međusobno pomaknute za 120° mogu se zapisati u sljedećem obliku:

$$\begin{aligned} \vec{u}_a &= u_a e^{j0} \\ \vec{u}_b &= u_b e^{j\frac{3\pi}{4}} \\ \vec{u}_c &= u_c e^{-j\frac{3\pi}{4}} \end{aligned} \tag{2-7}$$

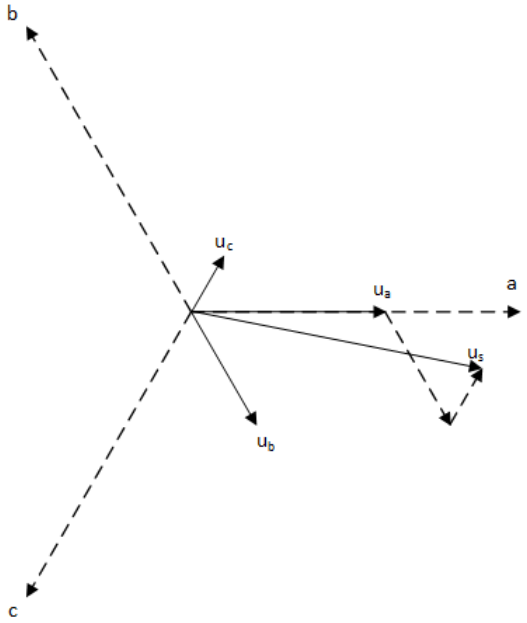
U konačnici se ukupni rezultantni vektor može pisati kao:

$$\vec{u}_s = \frac{2}{3}(u_a e^{j0} + u_b e^{j\frac{3\pi}{4}} + u_c e^{-j\frac{3\pi}{4}}) \tag{2-8}$$

Cijeli postupak je prikazan na slici 2.4.

Gledano s aspekta teorije električnih strojeva, rezultantni vektor odnosno prostorni vektor opisan sa (2-8) može predstavljati trenutne vrijednosti napona, struje ili toka, uz određeni faktor skaliranja o kojemu će više biti riječ kasnije, kao i o kompleksnom obliku prostornog vektora koji će biti objašnjen kroz Parkovu i Clarkovu transformaciju. U objašnjenju za ovo poglavlje je uzeta oznaka "u" jer će se Clarkova transformacija u prostorno-vektorskoj modulaciji, koja je tema ovoga rada, primjenjivati na faznim naponima. Važno je napomenuti da uporaba prostornog vektora u prikazu napona, struja i tokova električnih strojeva ide uz pretpostavku da su iste simetrične odnosno da nema nulte komponente spomenutih veličina tj. da vrijedi [28, 29, 30, 31]:

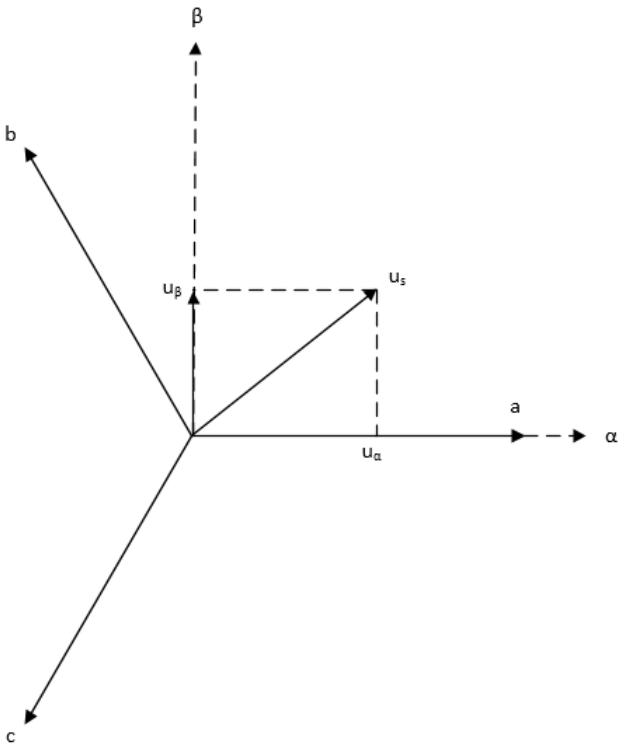
$$u_a + u_b + u_c = 0 \tag{2-9}$$



Sl. 2.4: *Prostorni vektor u trofaznom koordinatnom sustavu*

2.2.2 Clarkova transformacija

Ranije je rečeno da je Clarkova transformacija transformacija rotirajućeg troosnog koordinatnog sustava u mirujući dvoosni sustav. Prvo je, radi lakšeg razumijevanja transformacije, potrebno na slici 2.4 nacrtati dvoosni $\alpha - \beta$ sustav čija je os-α postavljena u os-a a,b,c sustava što je prikazano slikom 2.5.



Sl. 2.5: *Clarkova transformacija*

Raspiše li se jednadžba (2-7) u kompleksnom obliku dobije se:

$$\begin{aligned}\vec{u}_a &= u_a \\ \vec{u}_b &= u_b(-\frac{1}{2} + j\frac{\sqrt{3}}{2}) \\ \vec{u}_c &= u_c(-\frac{1}{2} - j\frac{\sqrt{3}}{2})\end{aligned} \tag{2-10}$$

Nadalje, uvrsti li se jednadžba (2-10) u jednadžbu (2-6) slijedi:

$$\vec{u}_s = \frac{2}{3}(u_a - \frac{1}{2}u_b + j\frac{\sqrt{3}}{2} - \frac{1}{2}u_c - j\frac{\sqrt{3}}{2}) \tag{2-11}$$

Postavi li se α-os kao realna os te β-os kao imaginarna os i rastavi li se jednadžba (2-11) na realni i imaginarni dio, može se pisati u matričnom obliku.

$$\begin{bmatrix} u_\alpha \\ u_\beta \end{bmatrix} = \frac{2}{3}\begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix}\begin{bmatrix} u_a \\ u_b \\ u_c \end{bmatrix} \tag{2-12}$$

Jedndžba (2-12) predstavlja Clarkovu transformaciju rotirajućeg trofaznog sustava u mirujući dvofazni. Jednadžba se množi sa $\frac{2}{3}$ kako bi se dobio realan prikaz vršnih vrijednosti stvarnih veličina [19, 32, 33, 34, 35]. Kada su poznate realna i imaginarna komponenta dvoosnog koordinatnog sustava preko njih se može izraziti prostorni vektor koji glasi:

$$u_s = u_\alpha + ju_\beta \tag{2-13}$$

Za koeficijent skaliranja se ne mora nužno odabrati $\frac{2}{3}$. Prema [29, 36, 37] može se odabrati i 1 ili $\sqrt{(\frac{2}{3})}$. Odabere li se $\frac{2}{3}$ kao faktor skaliranja očuvat će se amplituda odnosno vršna vrijednost transformirane veličine te će novi dvofazni sustav imati jednake impedancije kao trofazni, ali će snaga dvofaznog sustava biti 1.5 puta veća od snage trofaznog sustava. $\sqrt{(\frac{2}{3})}$ se koristi kada se želi očuvati snaga. Naime pretpostavi li se asinkroni motor u stacionarnom stanju, snaga u svakoj fazi se može izračunati kao umnožak efektivnih vrijednosti napona i struje i faktora snage odnosno:

$$P = UI\cos\varphi \tag{2-14}$$

Pretpostavi li se da je $\cos\varphi = 1$ ukupna snaga motora se može dobiti kao suma snaga pojedine faze. Pretpostavi li se asinkroni motor kojemu efektivne vrijednosti napona i struje iznose U_{abc} i I_{abc}, ukupna snaga motora P_{uk} se može izračunati kao:

$$P_{uk} = 3U_{abc}I_{abc} \tag{2-15}$$

Ako se na taj asinkroni motor primjeni Clarkova transformacija s faktorom skaliranja $\sqrt{\frac{2}{3}}$ dobije se da su struje i naponi ekvivalentnog dvofaznog sustava $\sqrt{\frac{3}{2}}$ puta veće nego naponi i

struje trofaznog sustava. Prema tome se snage pojedinih faza dvofaznog $\alpha - \beta$ sustava mogu izračunati kao [36]:

$$\begin{aligned} P_\alpha &= \sqrt{\frac{3}{2}} U_{abc} \sqrt{\frac{3}{2}} I_{abc} = \frac{3}{2} U_{abc} I_{abc} \\ P_\beta &= \sqrt{\frac{3}{2}} U_{abc} \sqrt{\frac{3}{2}} I_{abc} = \frac{3}{2} U_{abc} I_{abc} \end{aligned} \tag{2-16}$$

Ukupna snaga je jednaka zbroju snaga P_α i P_β odnosno:

$$P_{uk} = P_\alpha + P_\beta = \frac{3}{2} U_{abc} I_{abc} + \frac{3}{2} U_{abc} I_{abc} = 3 U_{abc} I_{abc} \tag{2-17}$$

Prema [36] tok, napon i struja se ne moraju skalirati sa jednakim faktorom, ali se u literaturi i primjeni najčešće koristi jednak faktor za sve tri veličine.

2.2.3 Parkova transformacija

Ranije je rečeno da je Parkova transformacija, transformacija statorskih veličina u dvofazni sustav fiksiran za os rotora stroja koji se vrti brzinom rotora. S obzirom da je Park u [20] transformaciju primjenjivao na sinkronom stroju, brzina vrtnje rotora je bila sinkrona brzina ω_s. Rečeno je da su Krause i Thomas prepoznali da se Parkova transformacije može primijeniti za bilo koju proizvoljnu brzinu vrtnje i trofaznog i dvofaznog koordinatnog sustava tako da je opća transformacija u biti Parkova transformacija samo s korištenom proizvoljnom brzinom. Ta transformacija transformira trofazne statorske i rotorske veličine u dvofazni koordinatni sustav koji može rotirati proizvoljnom brzinom ili mirovati. Transformacija trofaznog abc koordinatnog sustava koji rotira brzinom ω_1 u dvofazni dq koordinatni sustav koji rotira brzinom ω_2 je prikazana slikom 2.6. Kut između abc i dq sustava je φ, kut koju d-os zatvara sa zamišljenom referentnom osi je θ_2, a kut koju a-os zatvara sa zamišljenom referentnom osi je θ_1.

Prema slici 2.6 kut φ je:

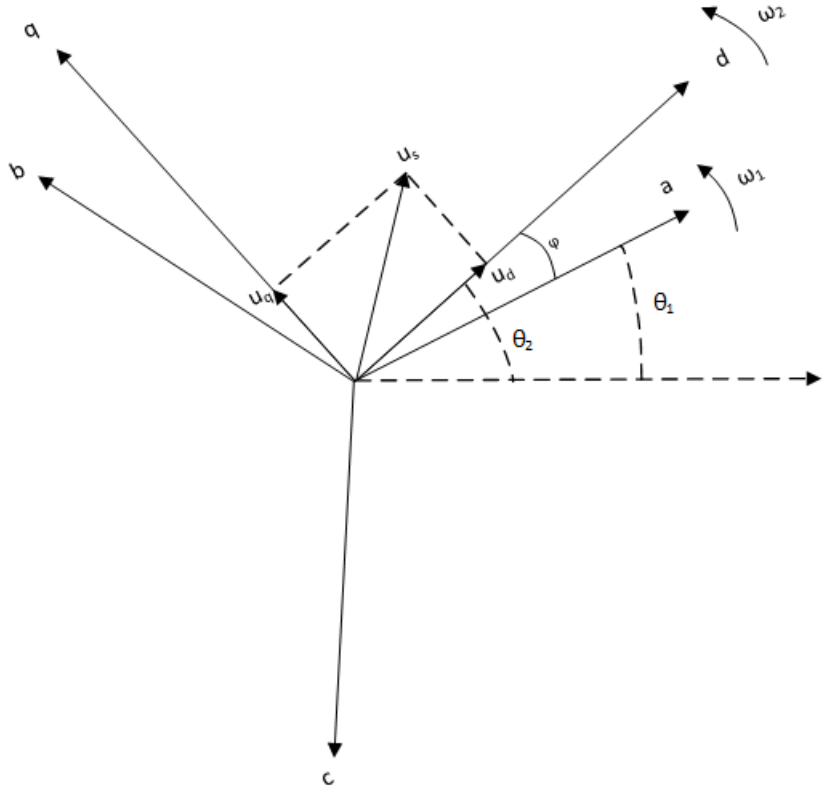
$$\varphi = \theta_1 - \theta_2 \tag{2-18}$$

a brzine vrtnje koordinatnih sustava su:

$$\begin{aligned} \omega_1 &= \frac{d\theta_1}{dt} \\ \omega_2 &= \frac{d\theta_2}{dt} \end{aligned} \tag{2-19}$$

Opet na isti način kao i kod Clarkove transformacije, raspišu li se komponente vektora u_s u abc sustavu u kompleksnom obliku i uvrste u jednadžbu (2.4) dobije se:

$$\vec{u}_s = \frac{2}{3}[u_a(\cos\varphi + j\sin\varphi) + u_b(\cos(\varphi - \frac{2\pi}{3})) + j\sin(\varphi - \frac{2\pi}{3}) + u_c(\cos(\varphi + \frac{2\pi}{3}) + j\sin(\varphi + \frac{2\pi}{3})] \tag{2-20}$$



Sl. 2.6: *Opća transformacija*

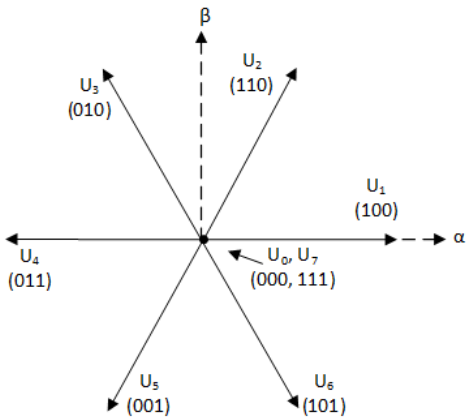
Rastavi li se jednadžba (2-20) na realni i imaginarni dio, gdje je d komponenta vektora realni dio, a q komponenta vektora imaginarni dio, i zapiše li se to u matričnom obliku dobije se izraz za Parkovu transformaciju:

$$\begin{bmatrix} u_d \\ u_q \end{bmatrix} = \frac{2}{3} \begin{bmatrix} \cos\varphi & \cos(\varphi - \frac{2\pi}{3}) & \cos(\varphi + \frac{2\pi}{3}) \\ -\sin\varphi & -\sin(\varphi - \frac{2\pi}{3}) & -\sin(\varphi + \frac{2\pi}{3}) \end{bmatrix} \begin{bmatrix} u_a \\ u_b \\ u_c \end{bmatrix} \tag{2-21}$$

2.3. Algoritam prostorno-vektorske modulacije

Sa sve bržim razvojem mikroprocesora, DSP-a (engl. Digital Signal Processor) i ugradbenih sustava prostorno-vektorska modulacija postaje preferirana modulacijska tehnika za primjenu u brojnim aplikacijama. Njene prednosti nad ostalim modulacijama su ranije rečene te je već objašnjen postupak dobivanja prostornog vektora. Rečeno je da svako sklopno stanje predstavlja jedan položaj prostornog vektora napona koji ima odgovarajući kut i amplitudu. Predstave li se ta sklopna stanja prostornim vektorima, dobivenim Clarkovom transformacijom prema (2-12), u kompleksnoj $\alpha - \beta$ ravnini dobije se slika 2.7.

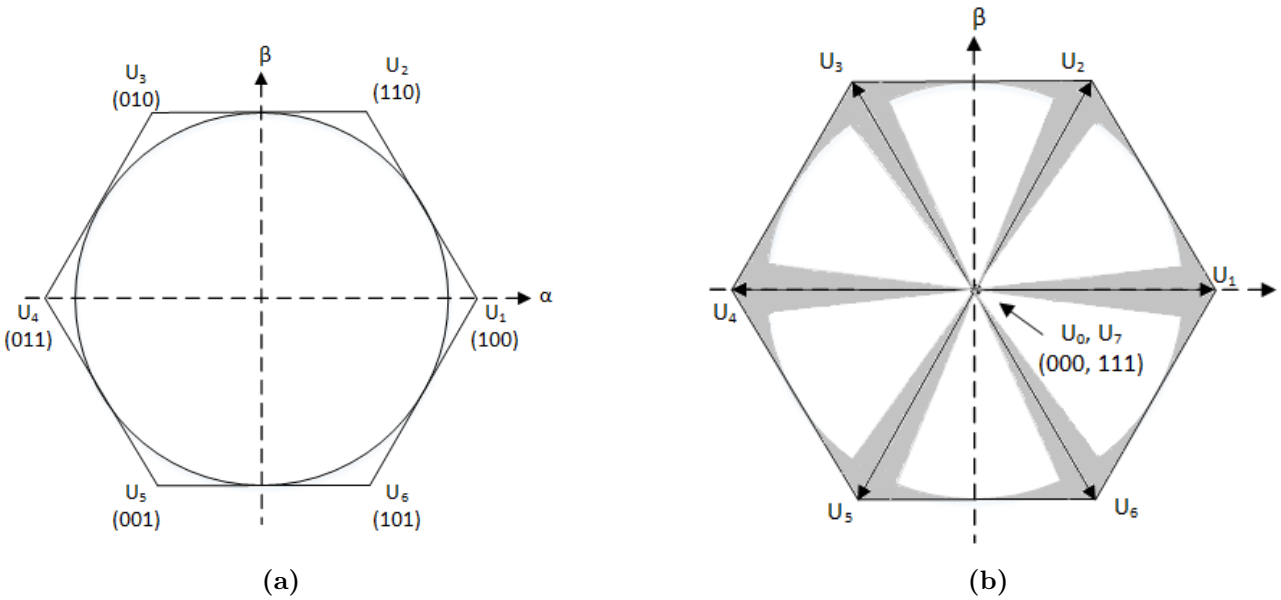
Prostorni vektori dijele $\alpha - \beta$ ravninu na 6 dijelova koji se nazivaju sekstanti ili sektori gdje svaki sektor iznosi 60°. Vektori U_0 i u U_7 su tzv. nul vektori. Binarni brojevi u zagradi označavaju stanje ventila odnosno "1" ako je uklopljen, "0" ako je isklopljen. Prvi broj u



Sl. 2.7: *Položaji prostornih vektora u $\alpha - \beta$ ravnini*

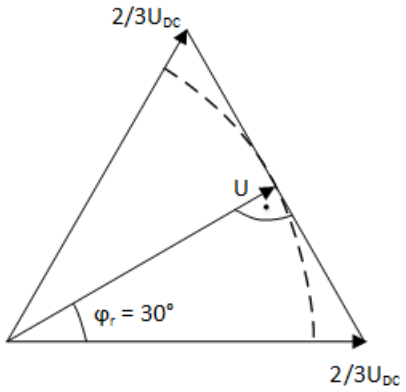
zagradi se odnosi na ventil S_1, drugi na ventil S_3 i treći na ventil S_5 dok su S_2, S_4 i S_6 njima komplementarni. Nacrta li se lokus ili polarni dijagram i upiše li mu se kružnica dobije se maksimalna amplituda vektora napona, koja je važna prilikom implementacije prostorno-vektorske modulacije, što je prikazano slikom 2.8a.

Ograničenje maksimalne amplitude napona nije jedina restrikcija prilikom implementacije prostorno-vektorske modulacije. Postoji i ograničenje s obzirom na vremena vođenja koja mogu biti realizirana. S obzirom da u praksi ne postoje idealne sklopke, odnosno da ne uklapaju i isklapaju trenutno nego im za to treba određeno vrijeme, potrebno je prilikom implementacije i to uzeti u obzir [28]. Obije restrikcije su grafički prikazane na slici 2.8b.



Sl. 2.8: *a) Polarni (Lokus) dijagram, b) Vektori koji se ne mogu realizirati [28]*

Promatra li se samo prvi sektor slike 2.8a može se odrediti maksimalna amplituda vektora napona što je prikazano slikom 2.9.



Sl. 2.9: *Određivanje maksimalne amplitude vektora napona*

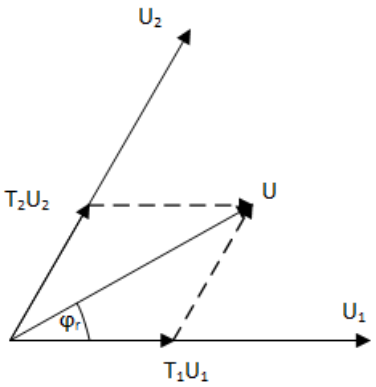
Prema slici 2.9 maksimalna amplituda vektora iznosi [41]:

$$\hat{U} = \frac{2}{3} U_{DC} \cos(\frac{\pi}{6}) = \frac{\sqrt{3}}{3} U_{DC} \tag{2-22}$$

Promatra li se samo prvi sektor na slici 2.7 kojeg omeđuju vektori U_1 i U_2 može se odrediti rezultantni ili referentni vektor napona. Rezultantni vektor, u slučaju prvog sektora, će se dobiti određenim obrascem sklapanja ventila odnosno aktiviranjem vektora U_1, U_2 i U_0 ili U_7. O idealnim obrascima sklapanja ventila će biti rečeno nešto više kasnije. Ako su vektori U_1 i U_2 konstantni te ako je nul vektor iznosa nula, onda se referentni ili rezultantni vektor dobije kao:

$$UT_s = \sum_{i=0}^{2} U_i T_i \tag{2-23}$$

gdje je U rezultantni ili referentni vektor, T_s je sklopni period, U_i aktivni vektor, a T_i vrijeme koje određuje koliko će pojedini vektor biti aktivan. Aktivna vremena pojedinih vektora ili vremena vođenja se određuju prema slici 2.10.



Sl. 2.10: *Određivanje referentnog vektora*

Aktivno vrijeme, prema slici 2.10, svakog vektora se računski može odrediti prema [14]:

$$\begin{aligned} \frac{t_1}{T_s} &= \frac{3}{\sqrt{3}}\frac{|U|}{U_{DC}}\sin(\frac{\pi}{3}-\varphi_r) \\ \frac{t_2}{T_s} &= \frac{3}{\sqrt{3}}\frac{|U|}{U_{DC}}\sin(\varphi_r) \\ T_s &= \sum_{i=0}^{2} t_i \end{aligned} \tag{2-24}$$

gdje je $|U|$ amplituda vektora napona, U_{DC} napon istosmjernog međukruga, a φ_r kut koji referentni vektor zatvara sa realnom osi $\alpha - \beta$ sustava.

Jednadžba (2-24) se nadalje može poopćiti da vrijedi za svaki sektor:

$$\begin{aligned} \frac{t_1}{T_s} &= \frac{3}{\sqrt{3}}\frac{|U|}{U_{DC}}\sin(k\frac{\pi}{3}-\varphi_r) \\ \frac{t_2}{T_s} &= \frac{3}{\sqrt{3}}\frac{|U|}{U_{DC}}\sin(\varphi_r-(k-1)\frac{\pi}{3}), \quad k=1,2,...,6 \\ T_s &= t_0 + t_k + t_{k+1} \end{aligned} \tag{2-25}$$

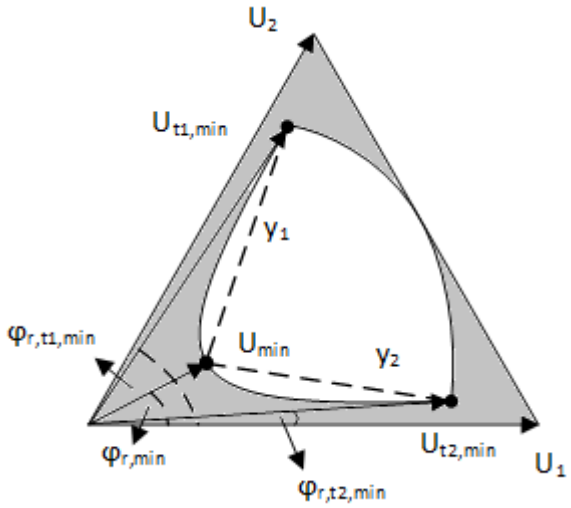
gdje je k broj sektora u kojemu se referentni vektor trenutno nalazi.

Ranije je određena maksimalna amplituda prostornog vektora napona te se sada pomoću jednadžbi (2-24) i (2-25), prema slici 2.8b mogu odrediti vremena vođenja koja se mogu realizirati. Promatra li se samo jedan sektor slike 2.8b odnosno slika 2.11 potrebno je odrediti izraze za minimalna vremena vođenja $t_{1,min}$ i $t_{2,min}$ te minimalnu amplitudu prostornog vektora napona U_{min}. Ako je poznata jednadžba (2-23) te ako se zabranjene zone aproksimiraju pomoću dva pravca y_1 i y_2, minimalna vremena vođenja se mogu izračunati pomoću izraza (2-24) i (2-25) preko kutova vektora čiji je vrh postavljen u karakteristične točke naznačene na slici. Amplituda tih vektora je jednaka maksimalnoj amplitudi $\hat{U}$ izračunatoj u izrazu (2-22). Prema tome vrijedi:

$$\begin{aligned} \varphi_{r,t1,min} &= \frac{\pi}{3} - \arcsin(\frac{\sqrt{3}U_{DC}t_{1,min}}{3\hat{U}T_s}) \\ \varphi_{r,t2,min} &= \arcsin(\frac{\sqrt{3}U_{DC}t_{1,min}}{3\hat{U}T_s}) \end{aligned} \tag{2-26}$$

odnosno u poopćenom obliku za svaki sektor:

$$\begin{aligned} \varphi_{r,t1,min} &= k\frac{\pi}{3} - \arcsin(\frac{\sqrt{3}U_{DC}t_{1,min}}{3\hat{U}T_s}) \\ \varphi_{r,t2,min} &= \arcsin(\frac{\sqrt{3}U_{DC}t_{1,min}}{3\hat{U}T_s} + (k-1)\frac{\pi}{3}), \quad k=1,2,...,6 \end{aligned} \tag{2-27}$$



Sl. 2.11: *Određivanje zabranjenih zona*

Minimalna amplituda vektora napona iznosi:

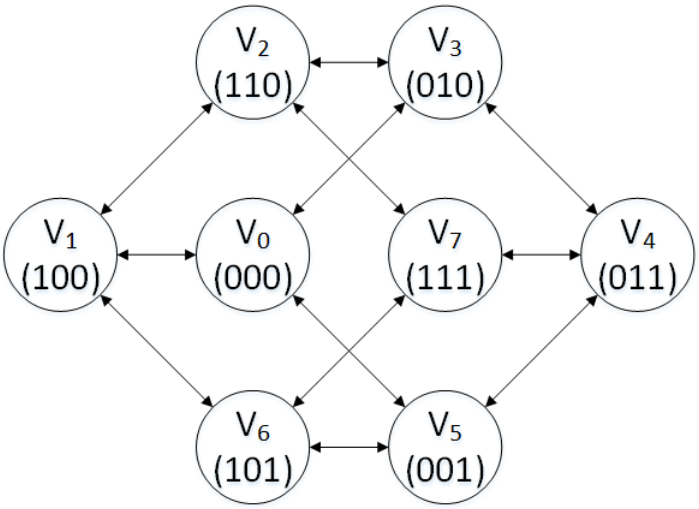
$$|U_{min}| = \frac{\sqrt{3}U_{DC}t_{min}}{3T_s\varphi_{r,min}} \tag{2-28}$$

2.4. Optimalan obrazac sklapanja ventila

Nakon izračunatih vremena vođenja potrebno je odrediti idealan obrazac sklapanja s ciljem minimiziranja sklopnih gubitaka i poboljšanja harmonijskog spektra valnih oblika. Postoji više obrazaca sklapanja za generiranje referentnog vektora U, no svi oni bi trebali zadovoljavati sljedeća dva uvjeta kako bi ranije istaknuti ciljevi bili ispunjeni [34, 42]:

- Tranzicija iz jednog sklopnog stanja u drugo uključuje samo dva ventila u istoj grani izmjenjivača, jedan ventil isklapa dok drugi uklapa.
- Tranzicija referentnog vektora napona U u $\alpha-\beta$ ravnini iz jednog sektora u drugi zahtjeva nimalo ili minimalan broj sklapanja

Prema dva navedena uvjeta i prema [34] moguće je nacrtati dijagram mogućih prijelaza iz jednog sklopnog stanja u drugo koji je prikazan na slici 2.12.



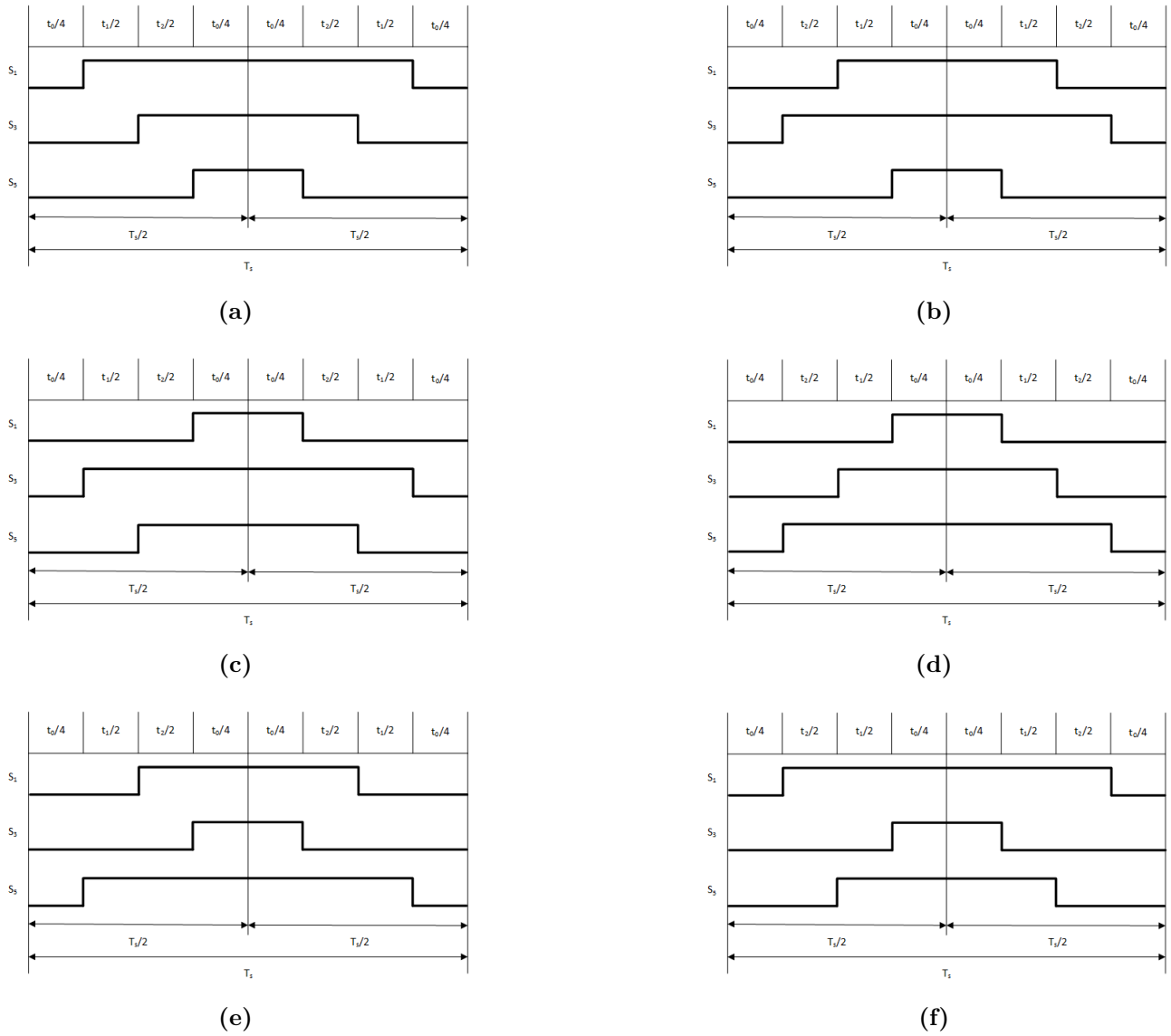
Sl. 2.12: *Mogući prijelazi među sklopnim stanjima [34]*

Jedna od mogućih sekvenci ili obrazaca sklapanja je tzv. simetrični ili centrirani obrazac. Simetrični obrazac sklapanja za sve sektore je prikazan na slici 2.13. Oznake S_1, S_3 i S_5 se odnose na upravljive ventile gornje grane izmjenjivača, dok su S_2, S_4 i S_6 njima komplementarni. Obrazac je podijeljen na 7 dijelova ili segmenata te se još naziva 7-segmentni obrazac sklapanja.

Uzme li se za primjer samo prvi sektor odnosno slika 2.13a te usporedi li se sa uvjetima za minimiziranje sklopnih gubitaka i poboljšanja harmonijskog spektra može se zaključiti da [42]:

- Da je prvi uvjet zadovoljen jer se iz sklopnog stanja (000) u sklopno stanje (100) dolazi tako što ventil S_1 uklopi dok ventil S_2 u isto vrijeme isklopi.
- Ako se sumiraju sva vremena vođenja prikazana na slici 2.13a dobije se sklopni period T_s odnosno vrijedi:

$$T_s = \sum_{i=0}^{2} t_i \tag{2-29}$$



Sl. 2.13: *Simetrični obrazac sklapanja ventila za: a) Sektor 1, b) Sektor 2, c) Sektor 3, d) Sektor 4, e) Sektor 5, f) Sektor 6*

- Broj sklapanja se smanjuje upotrebom nul-vektora tako da svaki sklopni period počinje i završava sa sklopnim stanjem V_0 dok su središnja dva segmenta sekvence sklopno stanje V_7.
- Svaki ventil izmjenjivača u sklopnoj periodi se pali i gasi samo jednom.

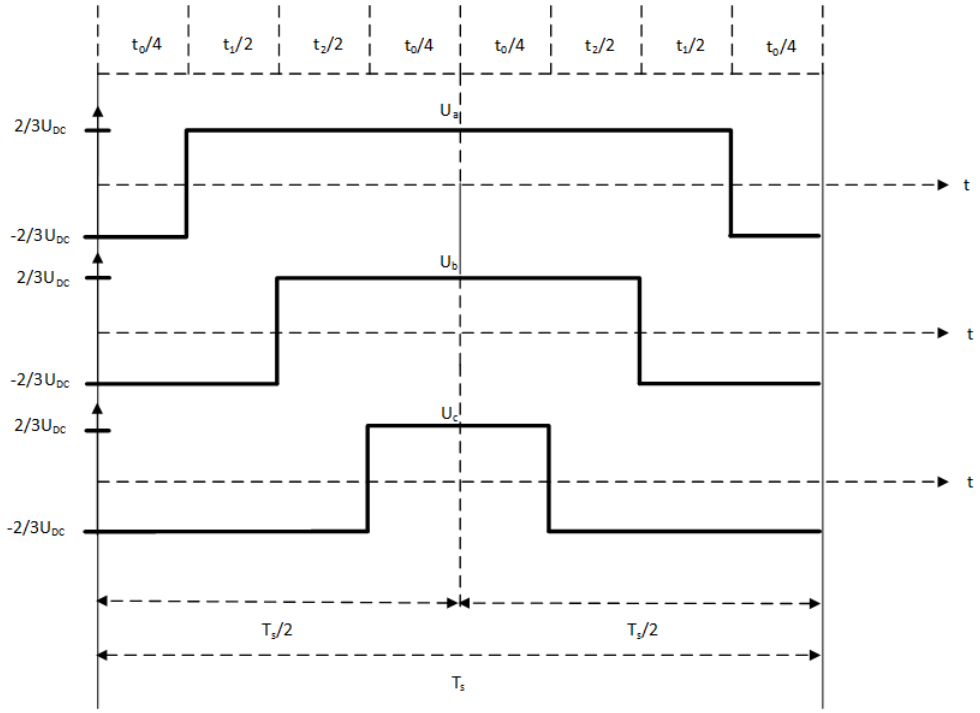
Ako se obrazac sklapanja prikaže tablično, što je učinjeno tablicom 2.3, primjećuje se da je ispunjen i drugi uvjet odnosno da prijelaz referentnog vektora iz jednog sekstanta u drugi zahtjeva nimalo ili minimalan broj sklapanja.

Tab. 2.3: *Obrasci sklapanja po sektorima*

	Segment						
Sektor	**1**	**2**	**3**	**4**	**5**	**6**	**7**
1	V_0 (000)	V_1 (100)	V_2 (110)	V_7 (111)	V_2 (110)	V_1 (100)	V_0 (000)
2	V_0 (000)	V_3 (010)	V_2 (110)	V_7 (111)	V_2 (110)	V_3 (010)	V_0 (000)
3	V_0 (000)	V_3 (010)	V_4 (011)	V_7 (111)	V_4 (011)	V_3 (010)	V_0 (000)
4	V_0 (000)	V_5 (001)	V_4 (011)	V_7 (111)	V_4 (011)	V_5 (001)	V_0 (000)
5	V_0 (000)	V_5 (001)	V_6 (101)	V_7 (111)	V_6 (101)	V_5 (001)	V_0 (000)
6	V_0 (000)	V_1 (100)	V_6 (101)	V_7 (111)	V_6 (101)	V_1 (100)	V_0 (000)

2.5. Valni oblici faznog i linijskog napona kod prostorno-vektorske modulacije

Nacrtaju li se pomoću obrazaca na slici 2.13 fazni naponi na izlazu izmjenjivača za sve tri faze dobije se slika 2.14 [38].



Sl. 2.14: *Fazni naponi za sektor 1*

Analogno slici 2.14 te prema slici 2.13 se mogu nacrtati i slike za ostale sektore. Odredi

li se srednja vrijednost valnog oblika sa slike 2.14 prema izrazu:

$$U_{a,sr} = \frac{1}{T}\int_0^{T_s} U_a(t)dt \tag{2-30}$$

dobije se za srednju vrijednost napona faze a izraženog preko vremena vođenja ventila i napona istosmjernog međukruga:

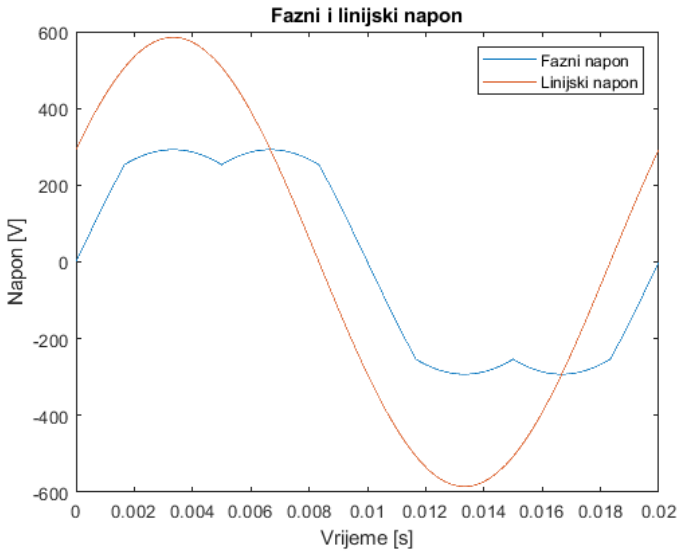
$$U_{a,sr} = \frac{2}{3}U_{DC}\frac{T_s - t_0}{T_s} \tag{2-31}$$

Primjeni li se isti postupak na sve sektore za sva tri fazna napona dobije se tablica 2.4.

Tab. 2.4: *Srednja vrijednost faznih napona za sve sektore*

	Sektor 1	**Sektor 2**	**Sektor 3**	**Sektor 4**	**Sektor 5**	**Sektor 6**
$\|\boldsymbol{U_a(t)}\|$	$\frac{2}{3}U_{DC}\frac{T_s-t_0}{T_s}$	$\frac{2}{3}U_{DC}\frac{t_1-t_2}{T_s}$	$\frac{2}{3}U_{DC}\frac{t_0-T_s}{T_s}$	$\frac{2}{3}U_{DC}\frac{t_0-T_s}{T_s}$	$\frac{2}{3}U_{DC}\frac{t_2-t_1}{T_s}$	$\frac{2}{3}U_{DC}\frac{T_s-t_0}{T_s}$
$\|\boldsymbol{U_b(t)}\|$	$\frac{2}{3}U_{DC}\frac{t_2-t_1}{T_s}$	$\frac{2}{3}U_{DC}\frac{T_s-t_0}{T_s}$	$\frac{2}{3}U_{DC}\frac{T_s-t_0}{T_s}$	$\frac{2}{3}U_{DC}\frac{t_1-t_2}{T_s}$	$\frac{2}{3}U_{DC}\frac{t_0-T_s}{T_s}$	$\frac{2}{3}U_{DC}\frac{t_0-T_s}{T_s}$
$\|\boldsymbol{U_c(t)}\|$	$\frac{2}{3}U_{DC}\frac{t_0-T_s}{T_s}$	$\frac{2}{3}U_{DC}\frac{t_0-T_s}{T_s}$	$\frac{2}{3}U_{DC}\frac{t_2-t_1}{T_s}$	$\frac{2}{3}U_{DC}\frac{T_s-t_0}{T_s}$	$\frac{2}{3}U_{DC}\frac{T_s-t_0}{T_s}$	$\frac{2}{3}U_{DC}\frac{t_1-t_2}{T_s}$

Nadalje se prema tablici 2.4 mogu nacrtati valni oblici faznih i linijskih napona. Slika 2.15 prikazuje valni oblik srednjih vrijednosti faznog napona jedne faze i linijskog napona. Valni oblici su radi jednostavnosti crtani pomoću Matlab/Simulinka gdje je napon istosmjernog međukruga iznosio $U_{DC} = 660V$, a sklopna frekvencija $T_s = 10kHz$.



Sl. 2.15: *Valni oblik faznog i linijskog napona*

Promotri li se fazni napon primjećuje se prisustvo trećeg harmonika, dok je u linijskom naponu on nepostajan odnosno poništava se zbog simetričnosti faznih napona. Treći harmonik se prema [14] javlja jer Clarkova transformacija nije u potpunosti reverzibilna tj. invarijantna je na treći harmonik.

3. PREGLED RAZLIČITIH MODULACIJSKIH TEHNIKA

Kao što je ranije napomenuto, postoje razne modulacijske tehnike od kojih su trenutno najpopularnije tzv. pulsno-širinske modulacije. Već je spominjana Six-Step modulacija pri kojoj svaki ventil vodi pola periode, te je na osnovu nje napravljena analiza rada trofaznog izmjenjivača u mosnom spoju. Također je dan i opis algoritma prostorno-vektorske modulacije. U ovom poglavlju će još biti opisana Six-Step modulacija pri kojoj svaki ventil vodi 120° i sinusna pulsno-širinska modulacija te će se na kraju dati osvrt na napredne modulacijske tehnike.

3.1. Six-Step od 120°

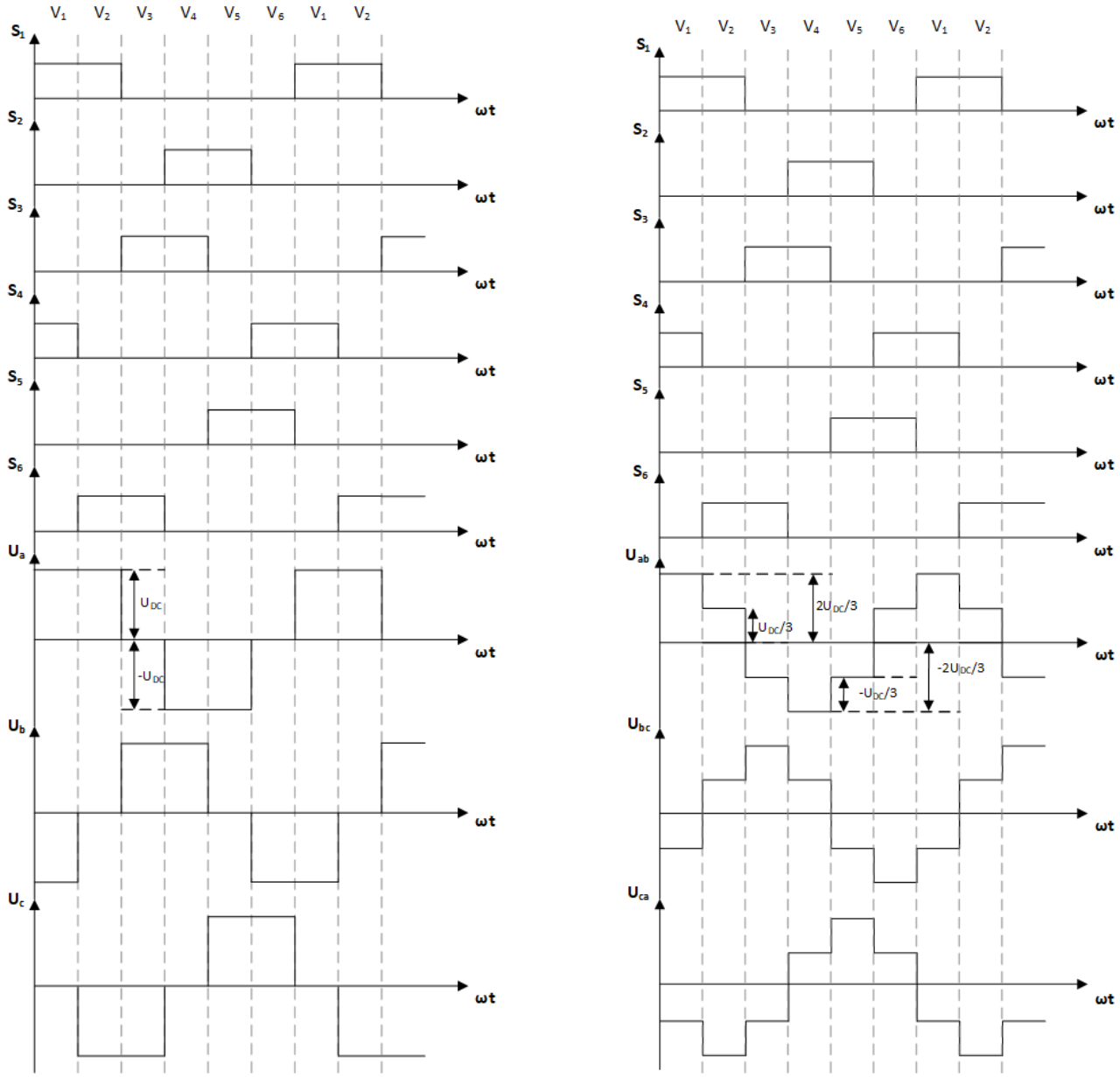
Glavna razlika između Six-Stepa od 180° i Six-Stepa od 120° je u vremenu vođenja ventila. Uzme li se to u obzir može se nacrtati tablica sklopnih stanja 3.1.

Tab. 3.1: *Sklopna stanja za Six-Step od 120°*

	Sklopno stanje					
Sektor	**1**	**2**	**3**	**4**	**5**	**6**
S_1	1	1	0	0	0	0
S_2	0	0	0	1	1	0
S_3	0	0	1	1	0	0
S_4	1	0	0	0	0	1
S_5	0	0	0	0	1	1
S_6	0	1	1	0	0	0

Usporedi li se tablica 3.1 sa tablicom 2.1 primjećuje se kako za razliku od Six-Stepa od 180°, kod Six-Stepa od 120° u svakom trenutku vodi jedan ventil u pozitivnoj i negativnoj strani izmjenjivača, dok kod Six-Stepa od 180° u svakom trenutku vode dva ventila u jednoj grani te jedan ventil u drugoj grani. To je i najveća prednost ovog načina modulacije jer se otklanja mogućnost da dva ventila u istoj grani provedu u istu vrijeme te se tako izbjegava stvaranje kratkog spoja na istosmjernom međukrugu što se kod Six-Stepa od 180° može dogoditi [43]. Naime, kod Six-Stepa od 180° u trenutku isklapanja jednog ventila njemu komplementaran ventil uklapa. S obzirom da je realnim ventilima potrebno određeno vrijeme da uklope i isklope, te da se može dogoditi kašnjenje u generiranju upravljačkih impulsa, može se dogoditi da dva ventila u istoj grani istovremeno budu uklopljena čime se događa kratki spoj istosmjernog međukruga. Kod Six-Stepa od 120° je ta mogućnost otklonjena jer, prema tablici 3.1, nakon što isklopi jedan ventil, postoji razdoblje od 60° kada su oba ventila u istoj grani isklopljena

nakon čega uklapa drugi ventil. Prema tablici se može nacrtati i slijed upravljačkih impulsa kao i fazni i linijski naponi na isti način kao i za Six-Step od 180° što je prikazano slikom 3.1.

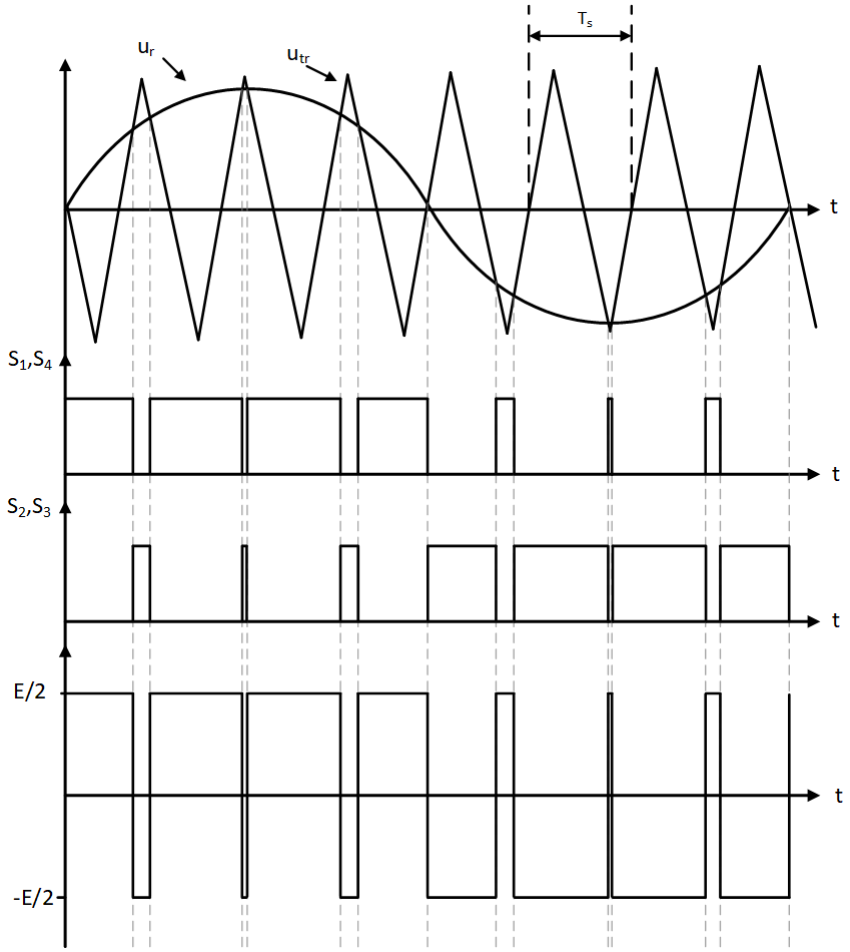


Sl. 3.1: *Six Step od 120° slijed upravljačkih signala i: a) Fazni naponi, b) Linijski naponi*

3.2. Pulsno-širinska modulacija

Jedne od najpopularnijih modulacijskih tehnika danas predstavljaju pulsno-širinske modulacije. Kod pulsno-širinske modulacije se referentni sinusni napon (u_r) uspoređuje s trokutastim valnim oblikom (u_{tr}) visoke frekvencije pri čemu izlazni napon ima frekvenciju sinusnog napona [40, 41]. Princip sinusne pulsno-širinske modulacije je prikazan na slici 3.2 [39].

Na slici 3.2 je prikazana modulacija za jednofazni izmjenjivač u mosnom spoju, no nju je



Sl. 3.2: *Sinusna pulsno-širinska modulacija*

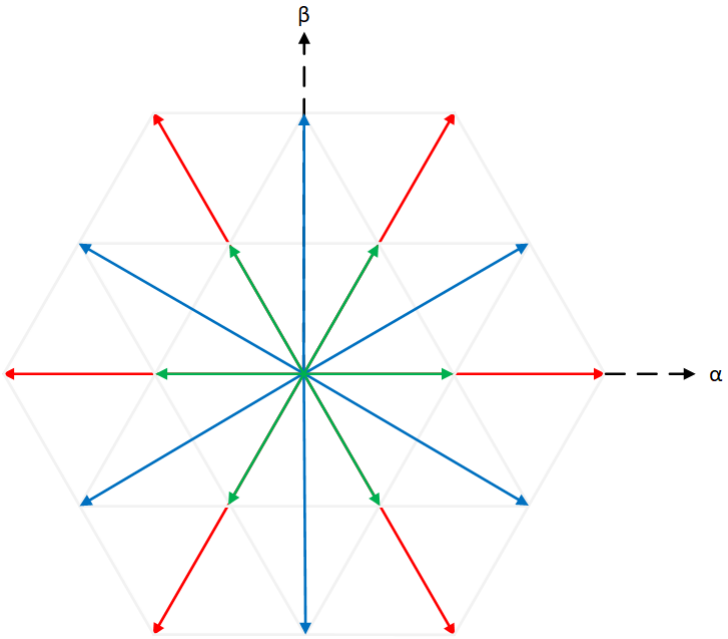
lako proširiti na trofazni izmjenjivač na način da se sa trokutastim valnim obliku uspoređuju tri sinusna napona međusobno fazno pomaknuta za 120°.

3.3. Napredne modulacijske tehnike

Napredne modulacijske tehnike su uglavnom varijacije ili bolje reći nadogradnje na postojeće modulacijske tehnike. Pretežito se koriste kod višerazinskih izmjenjivača kao što su "Neutral Point Clamped Inverter" (NPC), "Capacitor Clamped Inverter", kaskadni višerazinski izmjenjivači itd. [5, 6]. Za upravljanje njihovim ventilima se uglavnom koriste višerazinska sinusna pulsno-širinska modulacija i višerazinska prostorno-vektorska modulacija [5, 7].

Višerazinska prostorno-vektorska modulacija se bazira na sličnom algoritmu kao što je ranije opisan. Razlika je što kod, u ovom slučaju trorazinskog izmjenjivača, svaki sektor koji je prikazan na slici 2.7 još podijeljen na 4 trokuta, što je prikazano slikom 3.3 [40, 44].

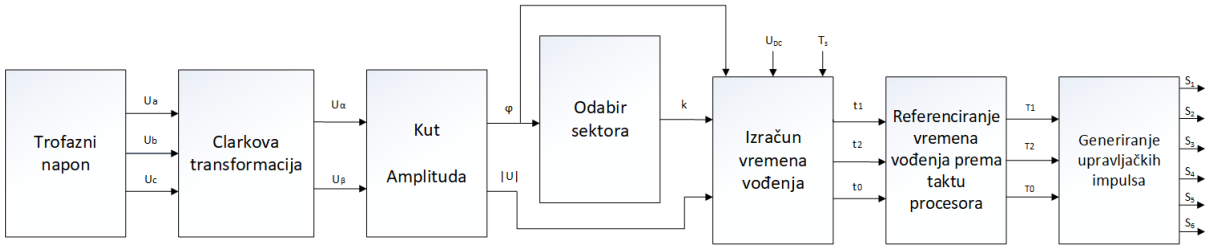
Prema [45] postoje ukupno 33 sklopna stanja kojima odgovara 19 prostornih vektora. Svaka faza kod trorazinskog izmjenjivača može biti spojena na pozitivnu i negativnu stranu istosmjernog međukruga te na nulu prema čemu postoje tri nul-vektora [8]. Vektori se dijele na dugačke (crvena), srednje (plava) i kratke (zelena) te je svih 19 prostornih vektora prikazano slikom 3.3



Sl. 3.3: *Položaji prostornih vektora kod trorazinske prostorno-vektorske modulacije*

4. IMPLEMENTACIJA PROSTORNO-VEKTORSKE MODULACIJE

Algoritam koji će biti implementiran je podijeljen na 7 dijelova koji su prikazani na slici 4.1.

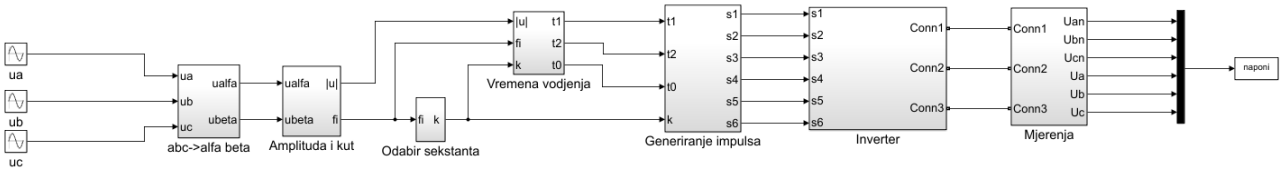


Sl. 4.1: *Blok dijagram algoritma prostorno-vektorske modulacije*

Prvo se generiraju tri sinusna napona međusobno pomaknuta za 120°. Zatim se trofazni sustav koji rotira brzinom $\omega = 2\pi f$ pomoću jednadžbe (2-12) transformira u dvofazni mirujući sustav. Nadalje se određuju kut i amplituda rezultirajućeg vektora. Nakon što su određeni kut i amplituda, određuje se sektor u kojem se prostorni vektor trenutno nalazi. Dalje se prema izrazu (2-25) određuju aktivna vremena pojedinih vektora koji sudjeluju u stvaranju rezultantnog vektora. Sljedeći korak je referenciranje tih aktivnih vremena prema taktu procesora. I u konačnici se generiraju impulsi. Svaki od ovih koraka će biti detaljnije objašnjen prilikom objašnjenja same implementacije.

4.1. Simulacija prostorno-vektorske modulacije u Matlab/Simulinku

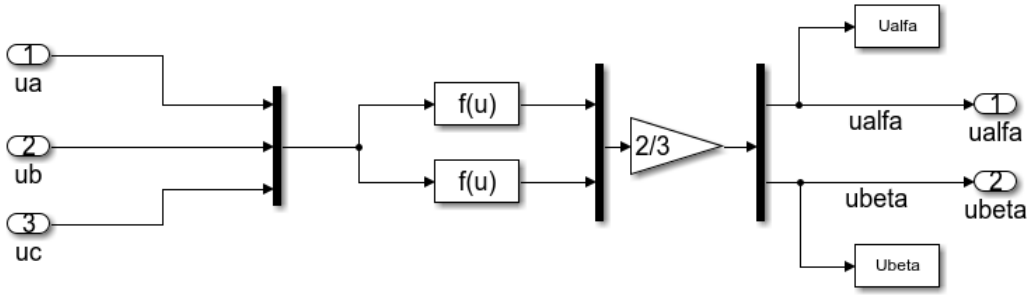
Prije same implementacije na cRio platformu provedeno je testiranje algoritma u Matlab/Simulinku. Prema blok dijagramu na slici 4.1 je izrađena simulacija u Simulinku prikazana na slici 4.2.



Sl. 4.2: *Simulacijski blok dijagram prostorno-vektorske modulacije izrađen u Matlab/Simulinku*

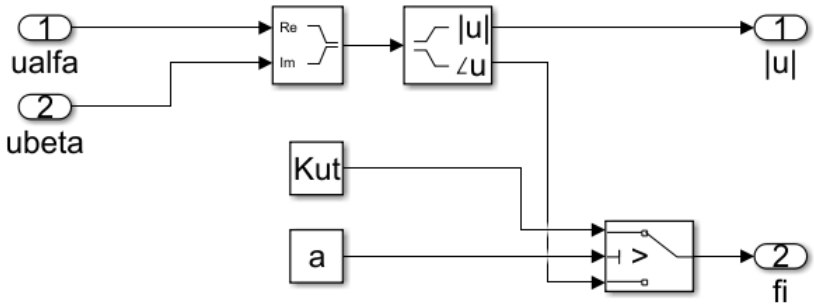
Pomoću "Sine Wave" blokova se generiraju tri sinusna napona međusobno pomaknuta za 120°. Nadalje se, kao što je ranije rečeno, ta tri sinusna napona pomoću Clarkeove transformacije (jednadžba (2-12)) transformiraju iz trofaznog u dvofazni sustav što je prikazano slikom 4.3.

Sada kada su određena α i β komponente napona, moguće je odrediti amplitudu i kut vektora kojeg one prema jednadžbi (2-13) čine. Prema slici 4.4 prvo se pomoću bloka "Real-Imag To Complex" α komponenta napona postavlja kao realna, a β komponenta kao imaginarna. Zatim se pomoću bloka "Complex To Magnitude-Angle" određuju amplituda i kut vektora



Sl. 4.3: *Clarkova transformacija izvedena u simulinku*

napona. U podsustavu "Amplituda i kut" prikazanog slikom 4.4 je omogućeno ručno unošenje proizvoljnog kuta preko bloka "Switch" radi mogućnosti testiranja implementiranog algoritma. Ako se u napisanoj Matlab skripti, koja će biti dana u prilogu, za varijablu "a" unese "1", Simulink će u simulaciji koristiti proizvoljno unesenu vrijednost kuta, dok će za "a=0" sam određivati vrijednost kuta. Vrijednost kuta se unosi u radijanima.

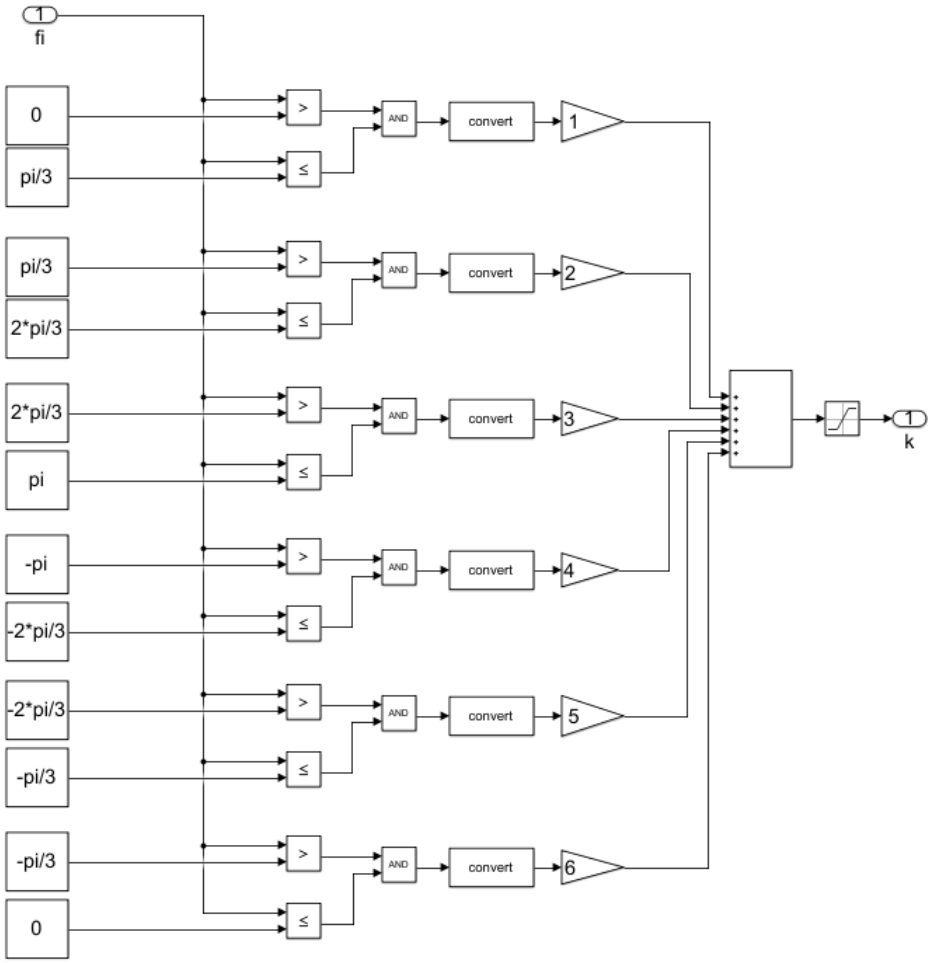


Sl. 4.4: *Određivanje kuta i amplitude*

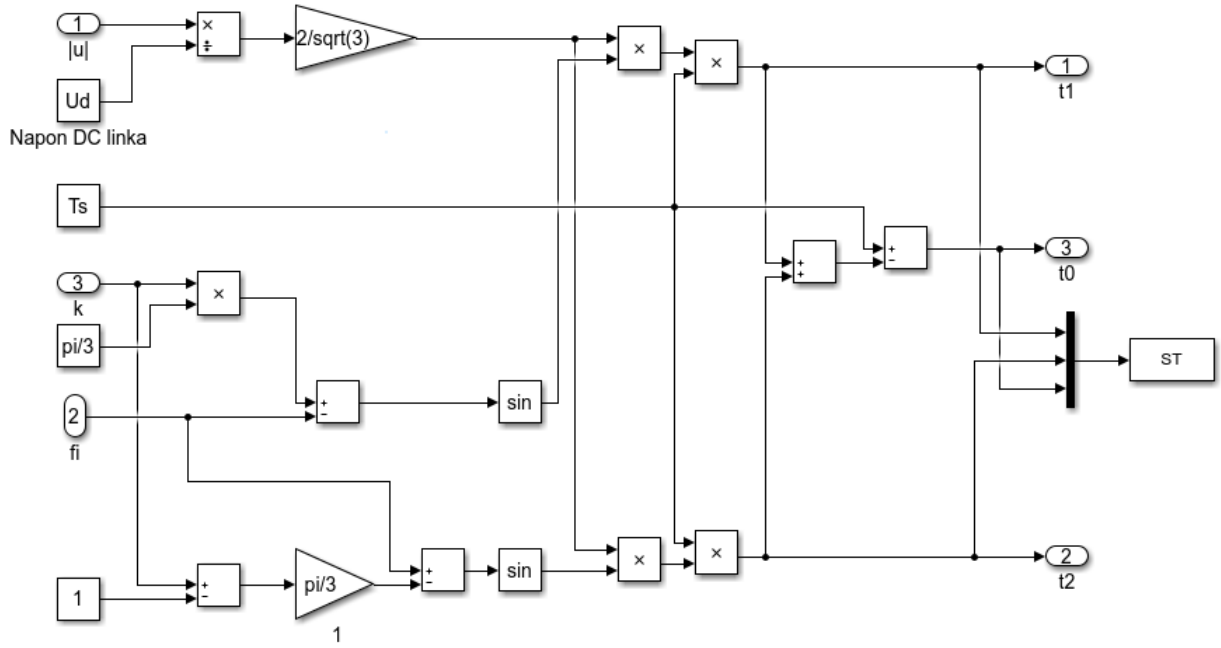
Dalje se određuje sektor u kojem se vektor nalazi što je prikazano na slici 4.5. Podsustav će biti objašnjen na primjeru prvog sektora. Prvo se provjera je li kut veći od 0° i manji ili jednak od 60°. Ako su oba uvjeta zadovoljena "AND" blok na svom izlazu daje logičku jedinicu koja se zatim pretvara u iz Boolean tipa podataka u Integer te se množi s brojem sektora, u ovom slučaju 1. Ista provjera se radi za ostalih 5 sektora samo za njima odgovarajuće kuteve te se množenje radi s odgovarajućim brojem sektora.

Nakon što su određeni sektor, amplituda i kut može se prema jednadžbi (2-25) pristupiti proračunu aktivnih vremena vektora koji čine rezultantni vektor što je prikazano slikom 4.6.

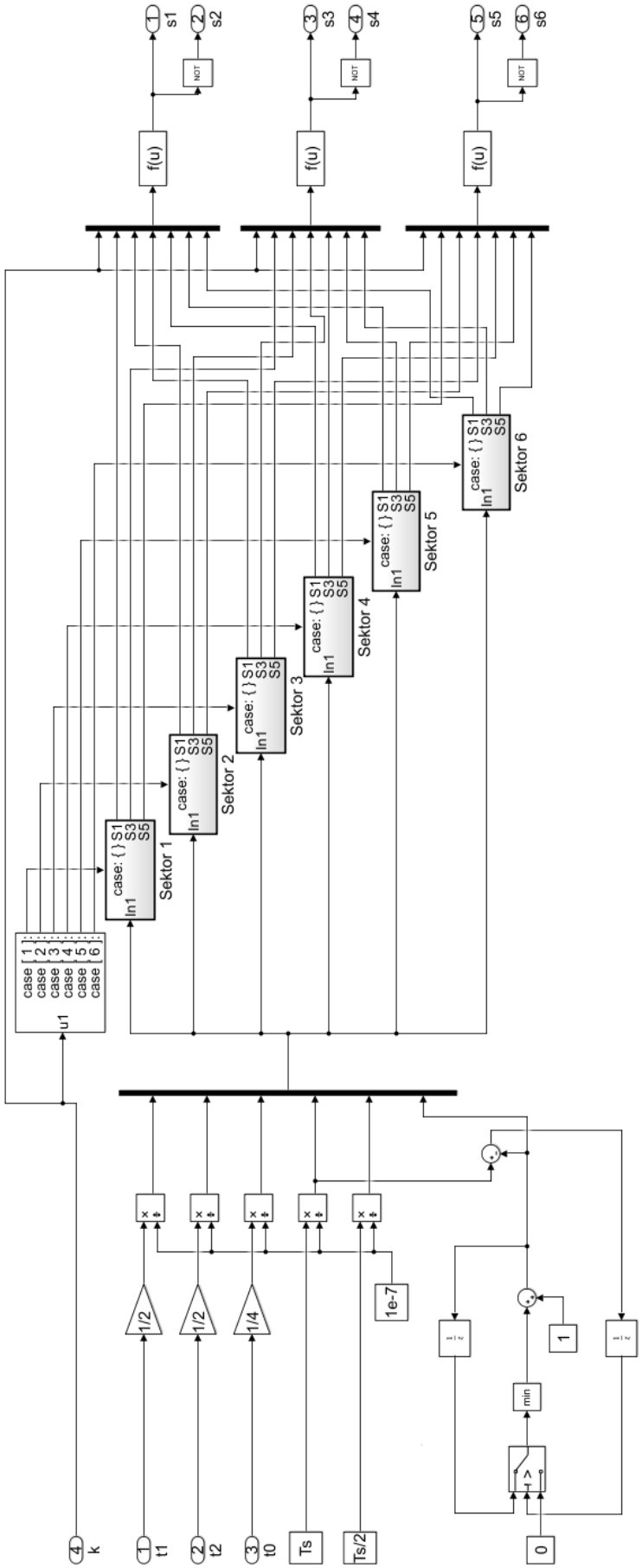
Sljedeći korak je generiranje samih impulsa što je prikazano slikom 4.7. Jedan od načina generiranja upravljačkih impulsa kod prostorno-vektorske modulacije je usporedbom kontrolnih valnih oblika ([27, 33]) ili vremena vođenja odnosno "duty cyclea" ([11, 15, 46]) sa trokutastim signalom kao kod pulsno-širinske modulacije, ili spremanjem već unaprijed proračunatih vremena vođenja ventila u LookUp tablice i prosljeđivanje istih timerima [9, 47]. U ovom slučaju se proračunata vremena množe sa taktom procesora, ili u slučaju Simulinka sa Step Sizeom postavljenim u Solveru, kako bi se dobilo koliko otkucaja procesora ili koraka iznosi određeno vrijeme vođenja.



Sl. 4.5: *Podsustav "Odabir sekstanta"*

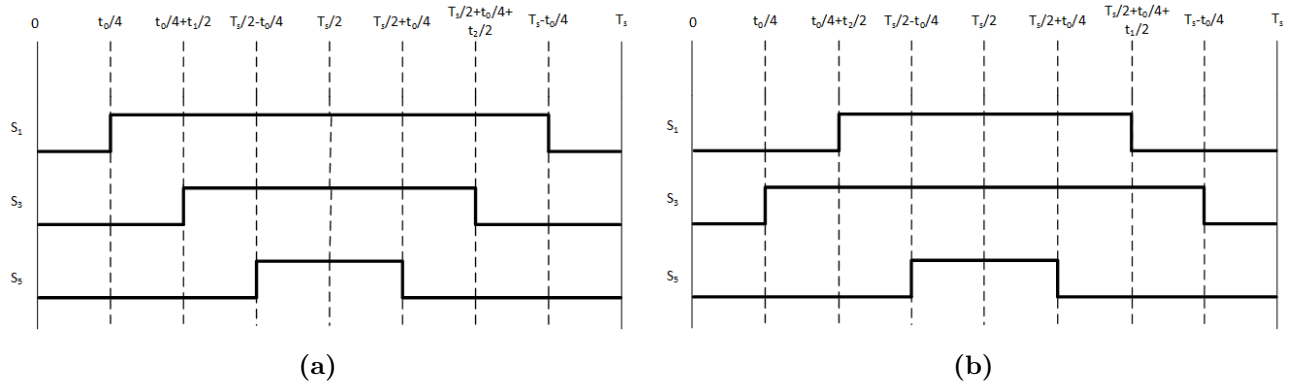


Sl. 4.6: *Proračun aktivnih vremena vektora u Simulinku*



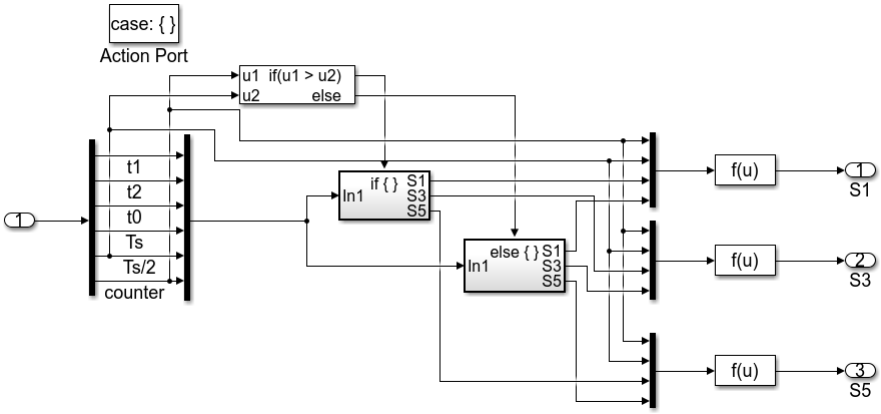
Sl. 4.7: *Podsustav generiranje impulsa*

Proračunat broj taktova se dalje uspoređuje s izrađenim brojačem i istovremeno prosljeđuje Simulinkovoj "Switch Case" naredbi koja prema sektoru u kojem se vektor trenutno nalazi i vremenskim trenutcima određenim prema slici 2.13 generira impulse za simetrični obrazac sklapanja. Vremenski trenutci prema kojima uklapaju i isklapaju ventili su prikazani na slici 4.8.



Sl. 4.8: *Trenutci uklopa i isklopa ventila za: a) Neparne sektore, b) Parne sektore*

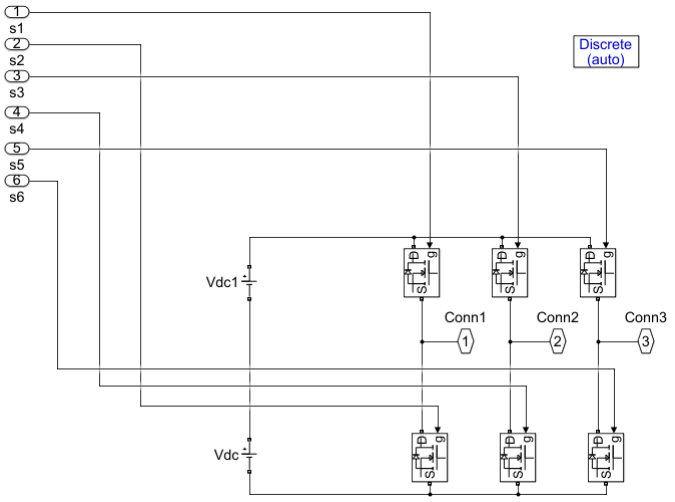
Uzme li se za primjer samo prvi sektor "Switch Case" naredbe, prikazan na slici 4.9, vidi se kako se u njegovom podsustavu nalazi "If Else" blok koji uspoređuje je li brojač veći ili manji od $T_s/2$. Ako je manji, ventili gornje ili pozitivne grane izmjenjivača uklapaju prema vremenskim trenutcima manjim od $T_s/2$ prikazanima na slici 4.8. Ako je veći, ventili gornje grane pretvarača isklapaju prema trenutcima na slici 4.8.



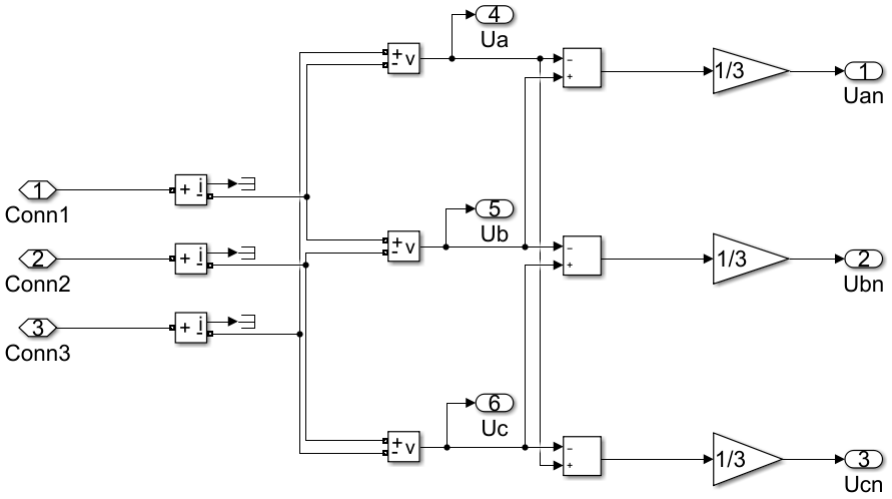
Sl. 4.9: *Switch Case za sektor 1*

Generirani impulsi se dalje proslijeđuju trofaznom izmjenjivaču u mosnom spoju modeliranom u Simulinku gdje se kao upravljivi ventili koriste IGBTi. Izmjenjivč je prikazana slikom 4.10.

Posljedni podsustav korišten u simulaciji je izrađen za mjerenje faznog i linijskog napona te je prikazan na slici 4.11.



Sl. 4.10: *Izmjenjivač modeliran u Simulinku*



Sl. 4.11: *Podsustav mjerenja*

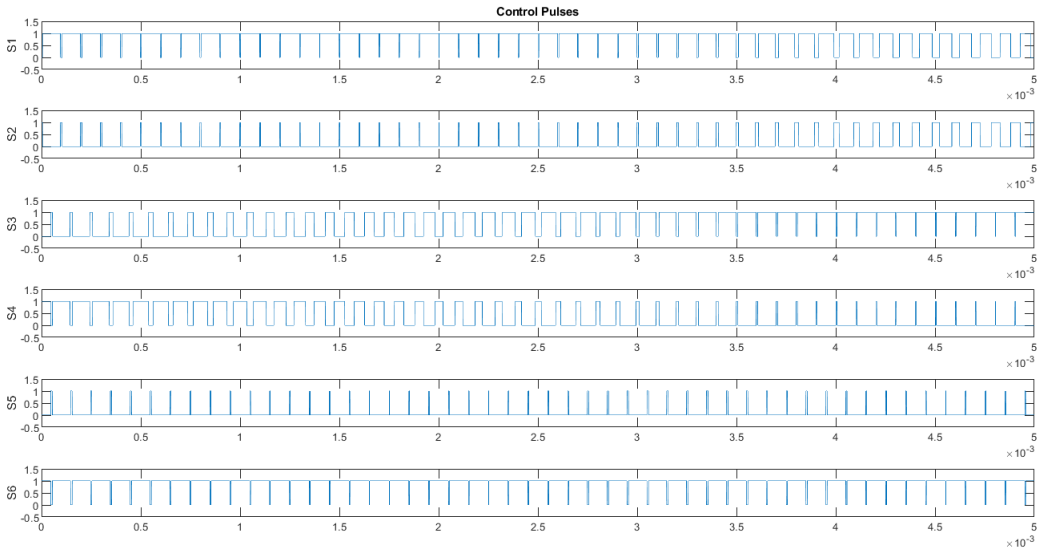
4.2. Rezultati implementacije u Matlab/Simulinku

U nastavku će biti prikazani rezultati simulacije za tri sklopne frekvencije: 1 kHz, 5 kHz i 10 kHz. Napon istosmjernog međukruga je postavljen na 660 V. Matlab skripta korištena pri simulaciji će biti dana u prilogu. Slika 4.12 prikazuje generirane upravljačke impulse pri sklopnoj frekvenciji od 10 kHz.

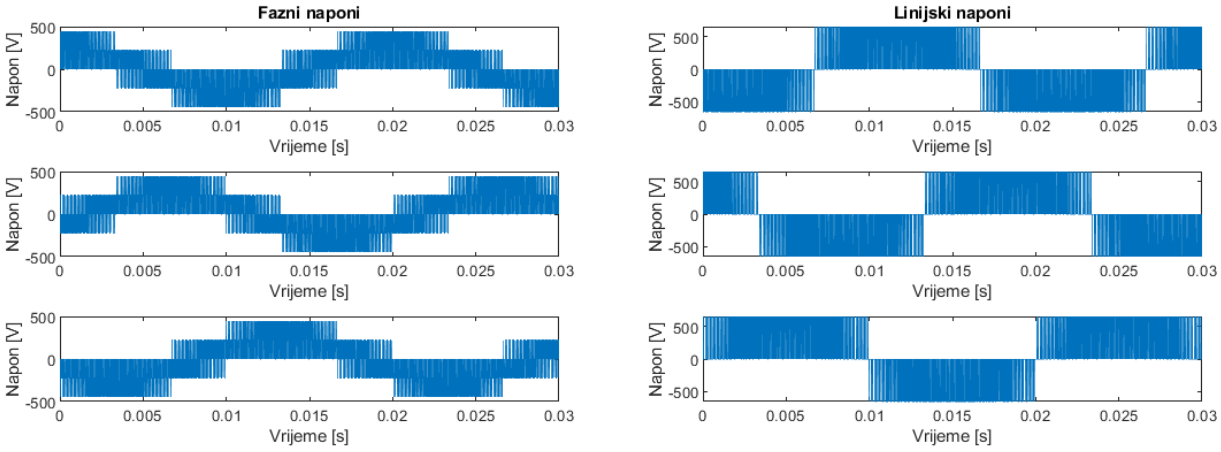
Prema slici 4.12 se vidi da su upravljački signali komplementarni, odnosno da neće doći do kratkog spoja na istosmjernom međukrugu zbog istovremenog uklapanja dva ventila.

Nadalje su na slici 4.13 prikazani valni oblici faznog i linijskog napona na izlazu simuliranog izmjenjivača pri sklopnoj frekvenciji od 10 kHz.

Usporede li se valni oblici faznog i linijskog napona s obavljenom analizom rada odnosno s valnim oblicima na slici 2.3 primjećuje se kako se oni podudaraju odnosno da fazni napon kao i u analizi rada ima 4 naponske razine ($\pm 2U_{DC}/3, \pm U_{DC}/3$), dok linijski napon ima 2 razine ($\pm U_{DC}$).



Sl. 4.12: *Generirani upravljački impulsi*



(a) **(b)**

Sl. 4.13: *a) Fazni naponi, b) Linijski naponi*

Slika 4.14 prikazuje simetrični obrazac sklapanja za svih 6 sektora pri sklopnoj frekvenciji od 10 kHz. Usporedi li se slika 4.14 sa slikom 2.13 vidi se da se slike podudaraju odnosno da je postignut simetričan obrazac sklapanja ventila s obzirom na polovicu sklopne periode $Ts/2$. Prema tome su zadovoljena dva uvjeta dana u poglavlju 2.4 te bi harmonijski sadržaj faznih i linijskih napona trebao biti optimalan. U nastavku su također prikazani fazni i linijski naponi za sklopnu frekvenciju od 5 kHz (slika 4.17) i 1 kHz (slika 4.18), te obrazac skalpanja ventila svih sektora za iste sklopne frekvencije (slike 4.15 i 4.16).

Slike 4.15 i 4.16 se također podudaraju sa slikom 2.13 odnosno implementacijom je postignut simetričan obrazac sklapanja ventila izmjenjivača. Na slikama 4.16b do 4.16f se može primijetiti nagli propad u upravljačkom impulsu točno na pola sklopne periode što je rezultat

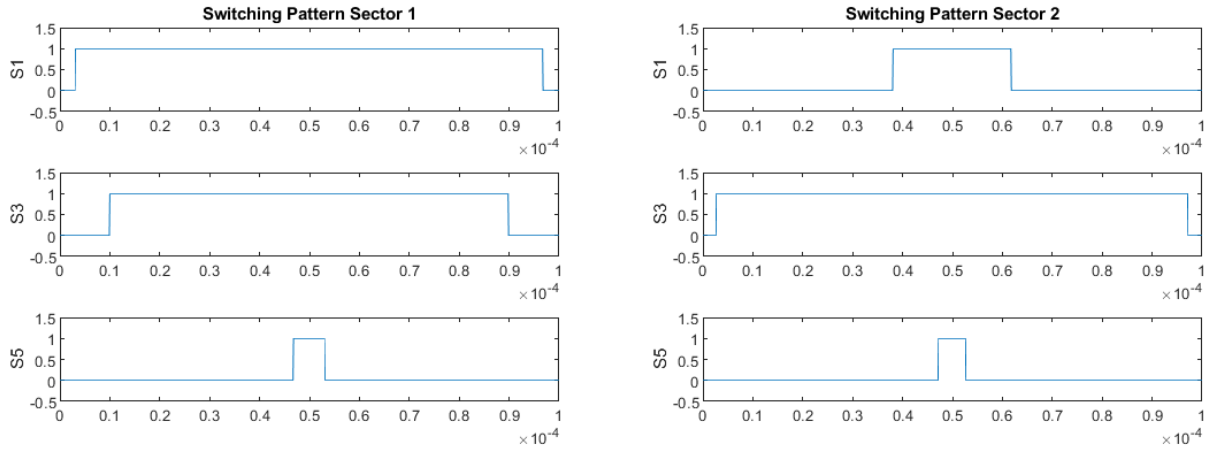
same izvedbe u Simulinku odnosno načinu rada simulacijskog programa. Naime u simulaciji nije točno određeno što činiti točno u trenutku $T_s/2$ tako da "If-else" naredba taj trenutak ne čini ništa odnosno daje 0 na izlazu što dolazi do izražaja tek pri nižim sklopnim frekvencijama kada je korak kojim se vrti simulacija veći. Pri realizaciji samog algoritma na nekom od ugradbenih sustava to može biti problem ovisno o upravljivim ventilima koji se koriste i njihovim vremenskim karakteristikama. Naime, ako upravljivi ventil ima vrijeme uključenja i isključenja koje je manje ili približno jednako vremenu propada signala, ventil će odreagirati i na taj impuls te će doći do narušavanja željenog obrasca sklapanja, te time do povećanja sklopnih gubitaka i pogoršanja harmonijskog sadržaja napona i struja na izlazu iz izmjenjivača.

Usporede li se slike 4.13, 4.17 i 4.18 primjećuje se kako su valni oblici faznih i linijskih napona sve "rjeđi" odnosno da su stanja vođenja i nevođenja ventila sve veća. Prema [48] posljedica toga je lošiji THD valnih oblika napona i struje zbog pogoršanja harmonijskog sadržaja. Također zbog dužeg vremena vođenja ventila može doći do gubitaka uslijed grijanja pretvaračkih komponenti. Prema [49] do povećanja udjela harmonika dolazi već pri sklopnim frekvencijama od 2 kHz dok pri sklopnim frekvencijama nižim od 1 kHz udio harmonika gotovo eksponencijalno raste.

Tablica 4.1 prikazuje proračunata i realizirana vremena vođenja odnosno aktivna vremena, te odnos između aktivnog i neaktivnog stanja ventila. Prikazani rezultati su urađeni za prvi sektor za sve tri sklopne frekvencije. T_{si} je neaktivno stanje ventila kada ventil ne vodi, a T_{sa} je stanje kada ventil vodi.

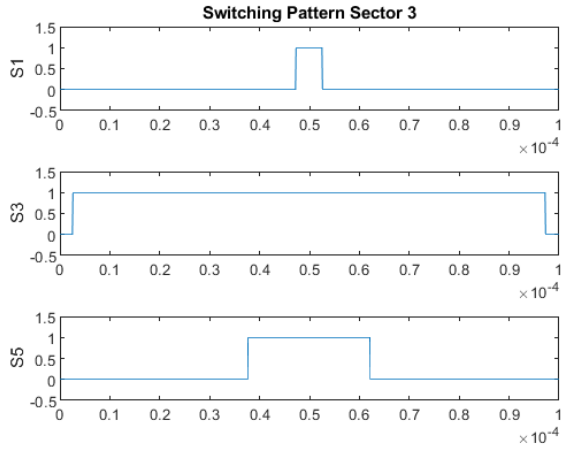
Tab. 4.1: *Izračunata i realizirana vremena vođenja*

$\mathbf{f_s}$ **željeni [kHz]**	1	5	10
$\mathbf{f_s}$ **realizirano [kHz]**	1.07	5	10.04
$\mathbf{t_1}$ **računski** [μs]	137.9	27.579	13.79
$\mathbf{t_1}$ **realizirano** [μs]	140.1	28.4	13.6
$\mathbf{t_2}$ **računski** [μs]	736.49	147.3	73.649
$\mathbf{t_2}$ **realizirano** [μs]	676.6	148.2	73.6
$\mathbf{t_0}$ **računski** [μs]	125.61	25.123	12.561
$\mathbf{t_0}$ **realizirano** [μs]	120.6	23.2	12.4
$\mathbf{T_{si}/T_{sa}}$ **računski**	0.067	0.067	0.067
$\mathbf{T_{si}/T_{sa}}$ **realizirano**	0.0688	0.0616	0.0664

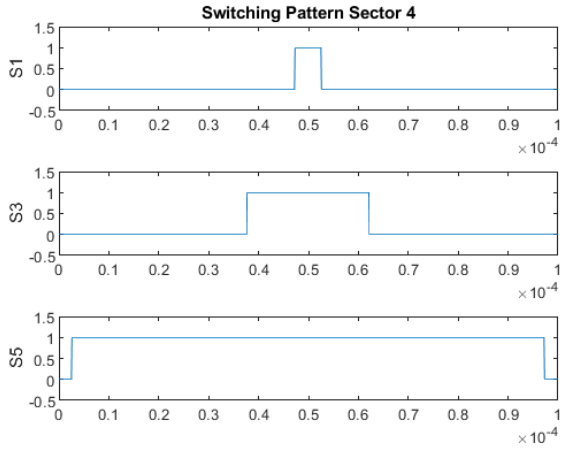


(a)

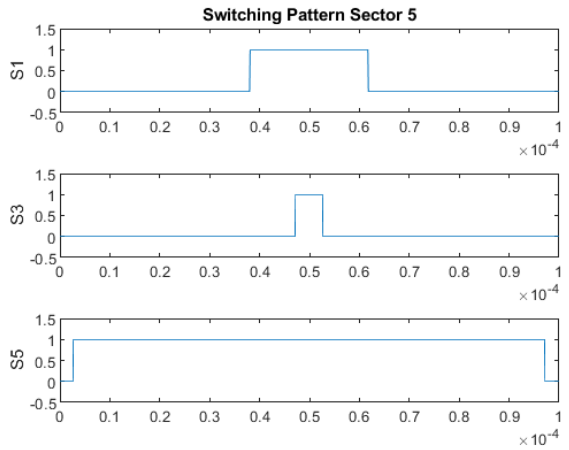
(b)



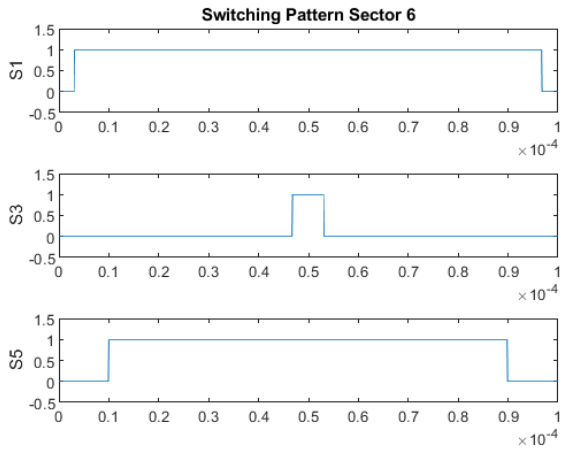
(c)



(d)

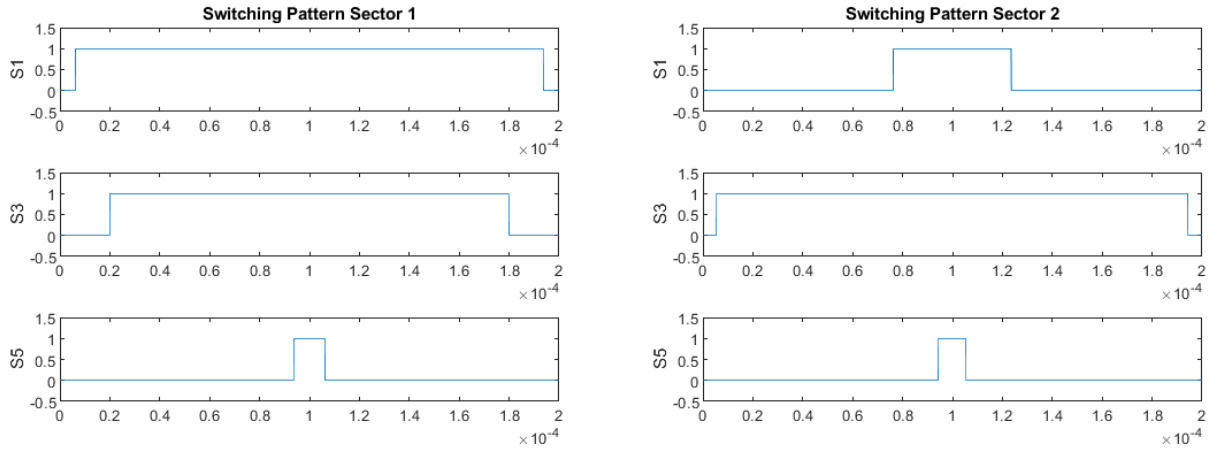


(e)



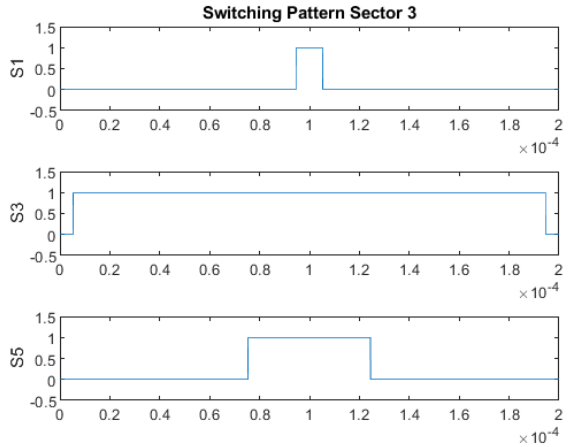
(f)

Sl. 4.14: *Simetrični obrazac sklapanja ventila postignut u Simulinku pri* $f_s = 10kHz$ *za: a) Sektor 1, b) Sektor 2, c) Sektor 3, d) Sektor 4, e) Sektor 5, f) Sektor 6*

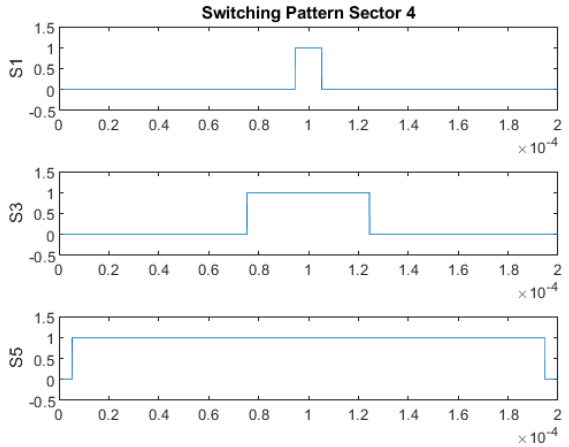


(a)

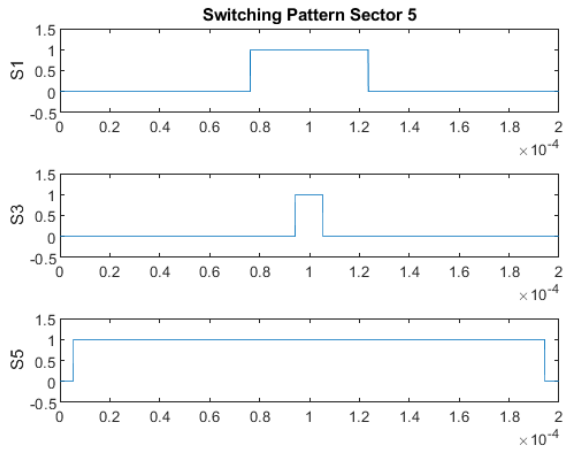
(b)



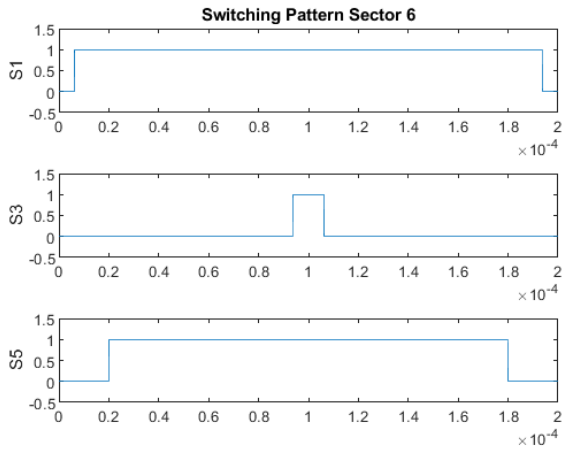
(c)



(d)

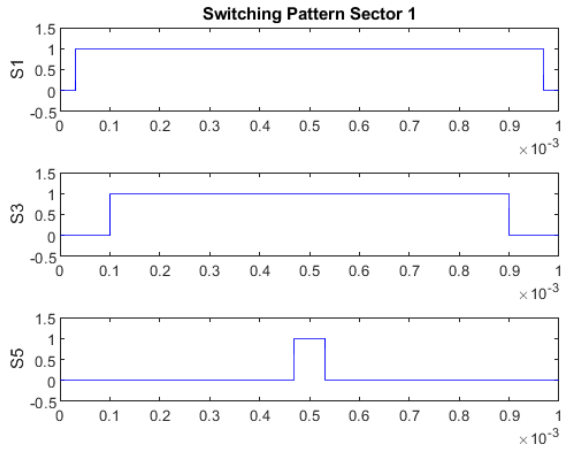


(e)

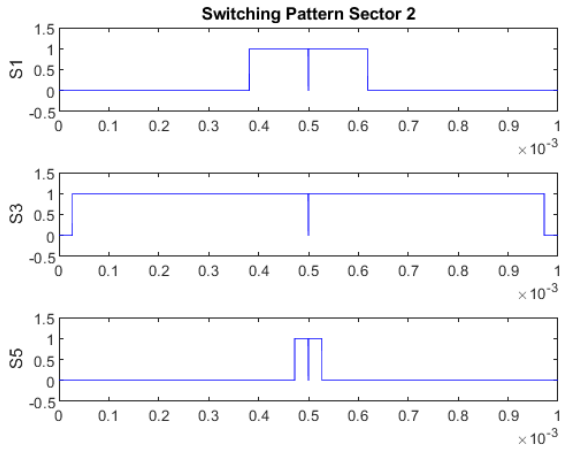


(f)

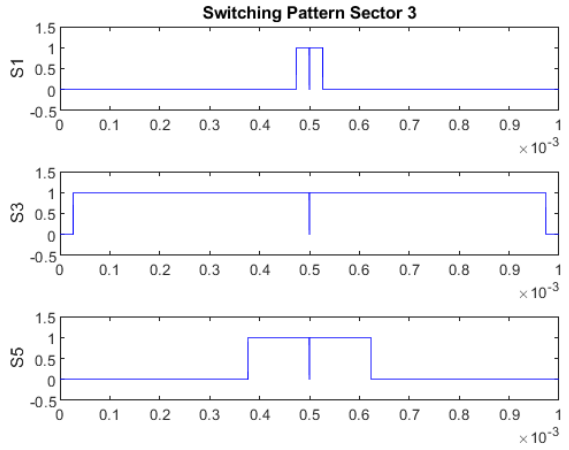
Sl. 4.15: *Simetrični obrazac sklapanja ventila postignut u Simulinku pri* $f_s = 5kHz$ *za: a) Sektor 1, b) Sektor 2, c) Sektor 3, d) Sektor 4, e) Sektor 5, f) Sektor 6*



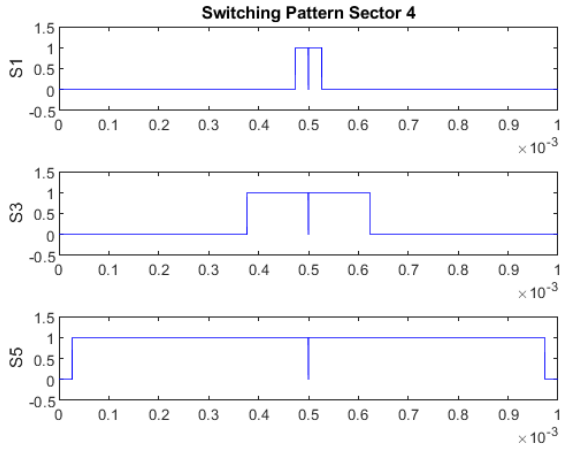
(a)



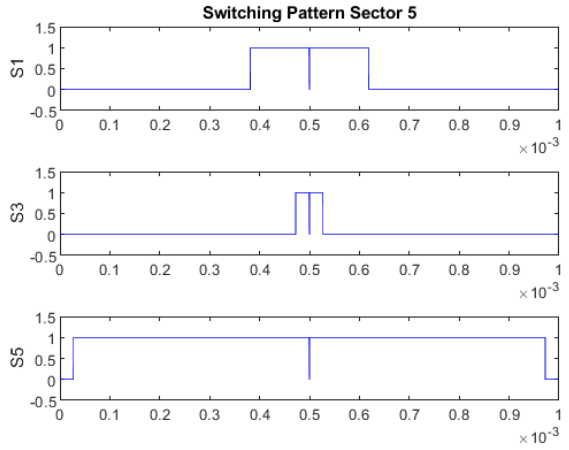
(b)



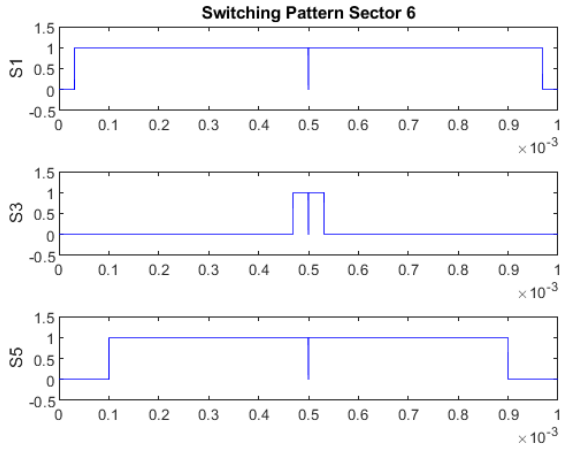
(c)



(d)

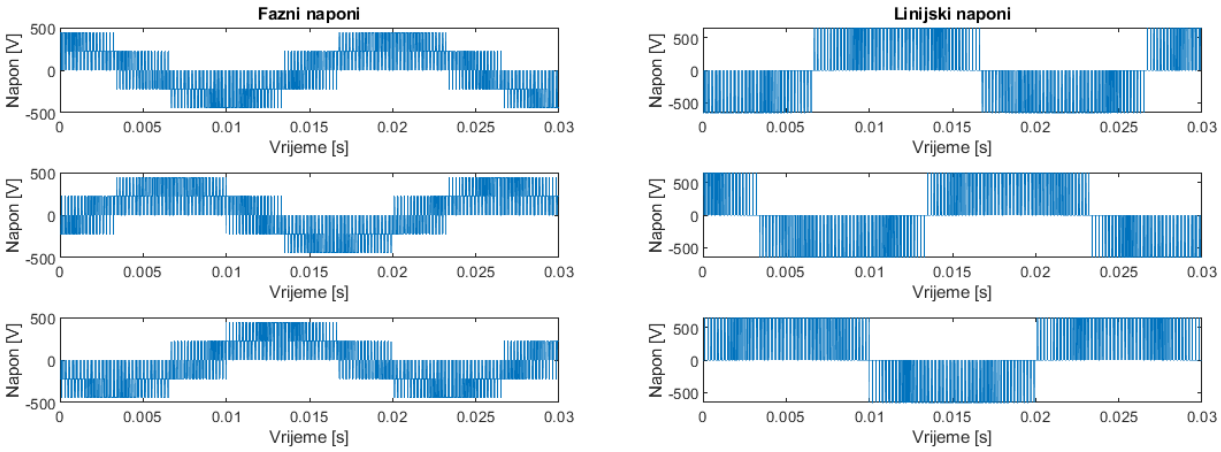


(e)



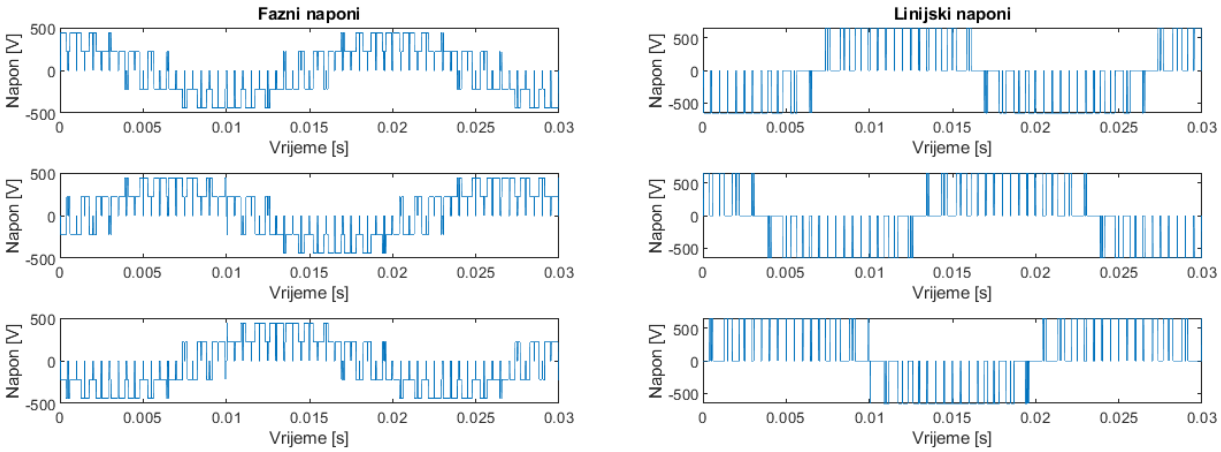
(f)

Sl. 4.16: *Simetrični obrazac sklapanja ventila postignut u Simulinku pr $f_s = 1kHz$ za: a) Sektor 1, b) Sektor 2, c) Sektor 3, d) Sektor 4, e) Sektor 5, f) Sektor 6*



(a) **(b)**

Sl. 4.17: *Sklopna frekvencija 5 kHz: a) Fazni naponi, b) Linijski naponi*



(a) **(b)**

Sl. 4.18: *Sklopna frekvencija 1 kHz: a) Fazni naponi, b) Linijski naponi*

Prema tablici 4.1 se vidi kako je s povećanjem sklopne frekvencije razlika između izračunatih i realiziranih vremena sve manja prema čemu se može zaključiti kako je primijenjena metoda implementacije prostorno-vektorske modulacije odnosno generiranja impulsa prigodnija za veće sklopne frekvencije. Omjer aktivnog i neaktivnog stanja se značajno ne mijenja.

5. EKSPERIMENT I REZULTATI

Implementacija algoritma prostorno-vektorske modulacije će se vršiti na National Instrumentsovoj (NI) CompactRio (cRio) platformi. CRio je RTOS/FPGA (*engl. Real-Time Operating System/Field Programmable Gate Array*) ugradbeni sustav koji se programira pomoću grafičkog programiranja odnosno slaganjem blokova u LabVIEW softveru od istog proizvođača.

5.1. LabVIEW

LabVIEW je skraćeno od "Laboratory Virtual Instrument Engineering Workbench". To je programsko okruženje gdje se programi stvaraju tzv. grafičkim programiranjem naspram konvencionalnih programskih jezika kao što su C, C++, C#, Java i sl. gdje se programira upisujući linije koda. LabVIEW programsko okruženje radi na računalima koja pokreću Windows, Mac OS X, Linux ali se u njemu mogu razvijati programi i za razne ugradbene sustave kao što su FPGA (engl. Field Programmable Gate Array), DSP (engl. Digital Signal Processor) i mikroprocesori. LabVIEW je izvorno izdan za Apple Macintosh 1986. godine te se danas uobičajeno koristi za prikupljanje podataka (engl. Data Acquisition), kreiranje virtualnih instrumenta, GPIB (engl. General Purpose Interface Bus), analizu prikupljenih podataka itd. [50, 51].

Program u LabVIEWu se sastoji od jednog ili više virtualnih instrumenata (VI) koji se nazivaju tako jer izgledom i načinom rada oponašaju stvarne instrumente. U pozadini su VIi zapravo samo funkcije, programi ili podrutine standardnih programskih jezika kao što je npr. C. VI se sastoji od tri glavna dijela: Front Panela, Block Diagrama i ikone [50].

Front Panel je interaktivno korisničko sučelje koje se naziva tako je simulira izgled stvarnih instrumenata. Može sadržavati pokazivače, indikatore, tastere, grafove odnosno mnoge druge ulazne i izlazne funkcije [50].

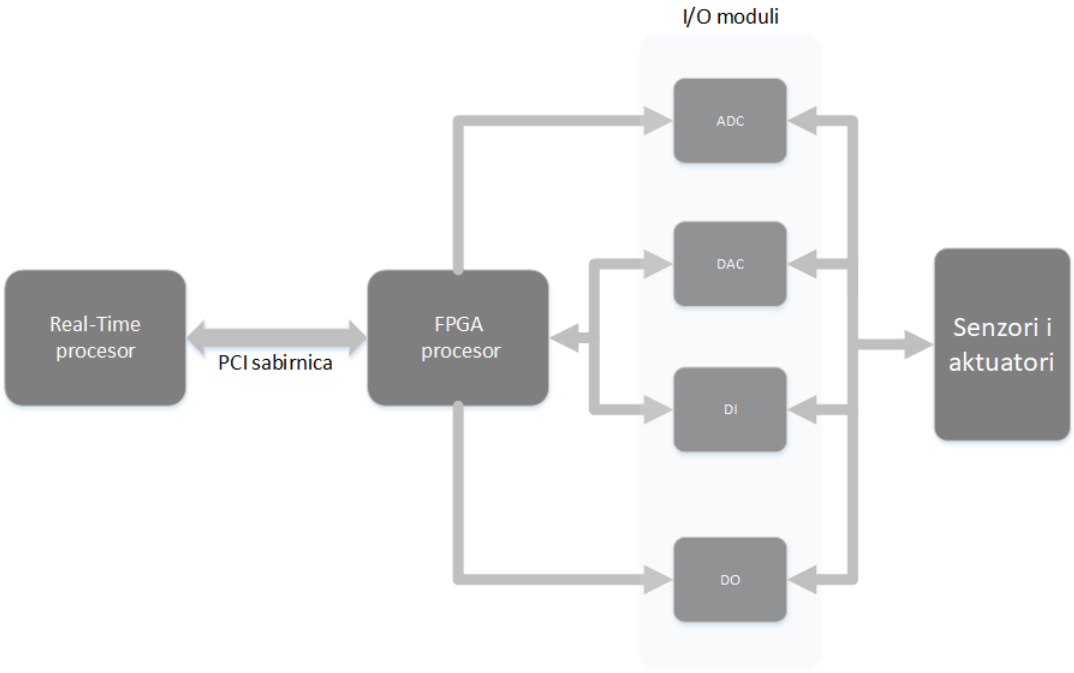
Blok Diagram je "kod" VIa koji se "piše" slaganjem raznih blokova gdje svaki blok predstavlja jednu funkciju. Objekti s Front Panela imaju odgovarajuće terminale u Blok Diagramu kako bi se mogli razmjenjivati podaci između korisnika i programa [50].

Ikone ili SubVIi su podrutine u blok dijagramu odnosno to su VIi unutar drugih VIa. Svaki SubVI ima jedan ili više konektora preko kojih razmjenjuje podatke s njemu nadređenim VIom [50].

Izvršavanje programa u LabVIEWu je određeno strukturom blok dijagrama u kojem su linijama povezane različite funkcije. Te linije služe kao komunikacija između različitih funkcija odnosno one "provode" vrijednosti od funkcije do funkcije te se funkcija izvršava onog trenutku kada dobije vrijednost. S obzirom da prema tome više funkcija istovremeno može dobiti vrijednosti, LabVIEW je sposoban paralelno izvršavati zadatke [51].

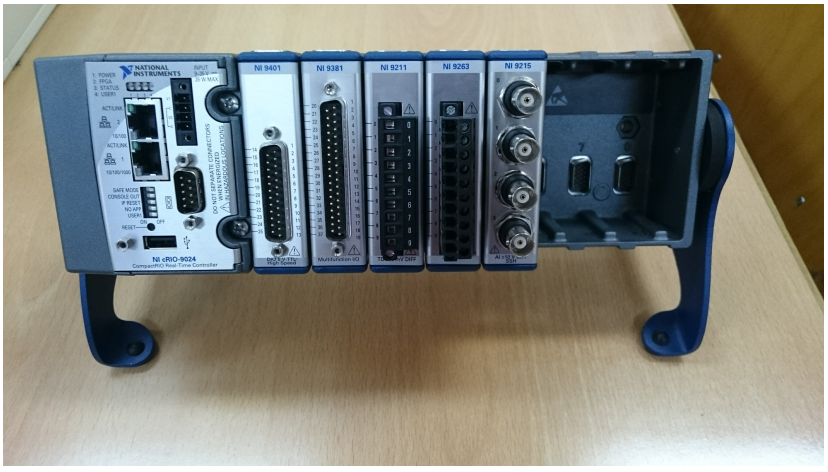
5.2. CompactRio RTOS/FPGA platforma

Za praktičnu implementaciju prostorno-vektorske modulacije potrebna je pouzdana, predvidiva, deterministička i vremenski precizna platforma koja može obavljati zahtjevne matematičke operacije. Takva platforma je National Instrumentsov CompactRio-9024 ugradbeni sustav. CRio se sastoji od tri glavne komponente: procesora koji pokreće Real-Time operacijski sustav (RTOS), FPGA procesora i ulazno-izlaznih modula. Arhitektura cRia je prikazana slici 5.1 [52].



Sl. 5.1: *Arhitektura cRio platforme*

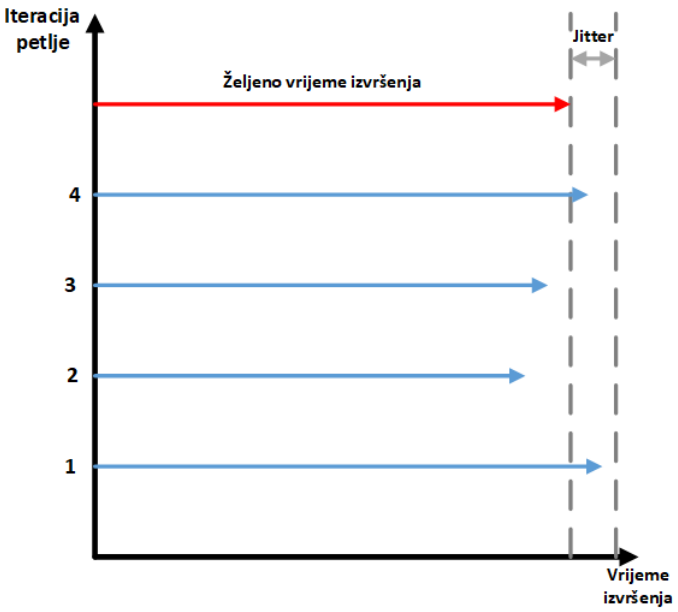
CRio-9024 posjeduje Real-Time procesor takta 800 MHz, 512 MB dinamičke RAM memorije te može spremiti do 4 GB podataka. Prikazan je na slici 5.2.



Sl. 5.2: *cRio-9024 embedded sustav*

5.2.1 Real-Time operacijski sustav

Općenito, operacijski sustav je zadužen za upravljanje hardverskim resursima računala i upravljanje aplikacijama koje se izvršavaju na računalu. RTOS također obavlja te zadatke ali je dizajniran tako da aplikacije izvršava vremenski precizno sa visokim stupnjem pouzdanosti. Moglo bi se reći da je glavni parametar RTOSa vrijeme odnosno vremenska preciznost. Kako bi se operacijski sustav mogao smatrati "Real-Time" mora se poznavati maksimalno vrijeme koje mu je potrebno za izvršavanje određenog zadatka. Prema tome se RTOSi dijele na dva tipa: "Hard Real-Time" i "Soft Real-Time". Hard Real-Time operacijski sustavi mogu apsolutno garantirati maksimalno vrijeme izvršenja određene aktivnosti, dok Soft Real-Time sustavi većinom mogu garantirati to vrijeme odnosno postoji određena greška i odstupanje [53, 54]. Usporede li se sa konvencionalnim operacijskim sustavima kao što je Windows, RTOSi su dizajnirani da izvršavaju kritične aplikacije i zadatke pouzdano i vremenski precizno po prioritetima koje određuje programer, dok su operacijski sustavi opće upotrebe dizajnirani tako da "pravedno" raspoređuju resurse između usluga i programa koji su pokrenuti [53]. Važan pojam koji se upotrebljava pri karakteriziranju RTOSa je "Jitter" odnosno količina pogreške u vremenskoj preciznosti koja se javlja prilikom višestrukog izvođenja programa ili petlje. RTOS su optimizirani kako bi pružili malu grešku ako su ispravno programirani. Svaka iteracija petlje bi trebala imati približno jednaku količinu vremena da se izvrši što je prikazano slikom 5.3.



Sl. 5.3: *Grafičko objašnjenje Jitter-a*

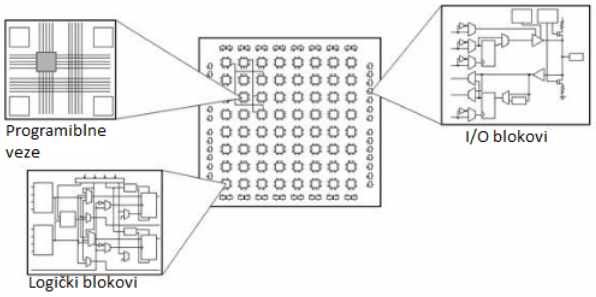
5.2.2 FPGA procesor

FPGA (engl. Field-Programmable Gate Array) su, ukratko rečeno, reprogramabilni silikonski čipovi. Ross Freeman, suosnivač Xilinxa, ih je osmislio 1985. godine [55]. FPGAi se sastoje od logičkih blokova koji se mogu reprogramirati po potrebi za obavljanje različitih zadataka,

naspram npr. integriranim sklopovima koji su napravljeni samo kako bi obavljali određene zadatke. Prednost FPGA je što može izvršavati više zadataka paralelno te što sadrže funkcije koje im omogućavaju korištenje jako visokih taktova. Nedostatak im je mala količina resursa koji se stavljaju programeru na raspolaganje [56].

Jedan od razloga sve veće primjene FPGA procesora leži u činjenica da, iako su po prirodi digitalni sklopovi, mogu se koristiti za obradu analognih signala ako su isti vremenski ovisni. Nadalje mogu obavljati prijenos podataka sa niskom razinom šuma [56].

Svaki FPGA procesor se sastoji od ograničenog broja predefiniranih resursa sa programabilnim vezama kako bi se implementirao reprogramabilni digitalni sklop i I/O blokovi koji omogućavaju sklopu komunikaciju sa senzorima i aktuatorima. Preciznije rečeno, FPGA se sastoji od programabilnih logičkih blokova, funkcijskih logičkih blokova kao što su multiplikatori, i memorijskih blokova kao što je RAM. Dijelovi od kojih se sastoji FPGA su prikazani na slici 5.4 [55].

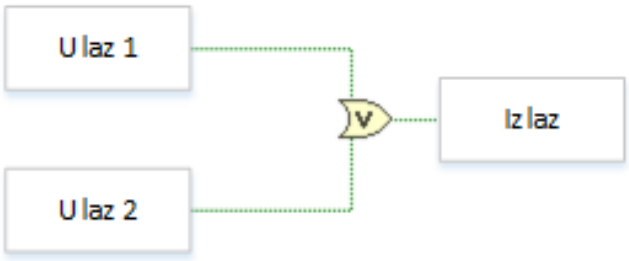


Sl. 5.4: *Dijelovi FPGA procesora [55]*

Programabilni logički blokovi (engl. Configurable Logic Blocks - CLBs) su osnovna logička jedinica svakog FPGA procesora koja se još naziva "Slice-ovi" ili "logičke ćelije". CLBi se sastoje od dvije osnovne komponente: Flip-Flopovi i LookUp Tablice (LUTs). Još dva važna dijela FPGA i njegovih resursa su "DSP Slices" koji imaju istu funkciju kao DSPi i blok memorija tj. RAM. Flip-Flopovi su binarni Shift registri koji se koriste za sinkronizaciju logike i spremanje logičkih stanja između dva takta FPGA procesora. Pri svakom taktu procesora, Flip-Flop prima na svom ulazu logičku 1 ili 0 (True ili False) i sprema vrijednost do idućeg takta [55].

LookUp tablice su CLBi koji su implementirani koristeći male iznose RAMa. Sva kombinatorna logika u FPGA procesorima je implementira u obliku tablica istine u tzv. LUT memoriju koja nije ništa drugo nego RAM memorija namijenjena isključivo za LookUp tablice. Tablica istine je predefiniran popis izlaza za sve moguće ulaze. Uzme li se za primjer booleovu "ili" funkciju (slika 5.5), umjesto da FPGA za svaki novi takt procesora i za svaki novi podatak na ulazu iznova računa izlaz, on se jednostavno pozove na predefiniranu LookUp tablicu i odabere odgovarajući izlaz što štedi FPGA resurse i ubrzava cijeli sustav [55].

S obzirom da matematičke operacije izvedene hardverski zahtijevaju velik broj sklopova, npr. množenje dva 32-bitna broja zahtjeva više od 2000 operacija, koriste se multiplikatori i DSP Sliceovi koji su zapravo predefinirani sklopovi multiplikatora kako bi se uštedjelo na upotrebi LookUp tablica i Flip-Flopova pri obavljanju matematičkih operacija. DSP sliceovi se ukratko



(a)

Ulaz 1	Ulaz 2	Izlaz
0	0	0
0	1	1
1	0	1
1	1	1

(b)

Sl. 5.5: *a) Logička OR operacija u LabVIEWu, b) Simbol i tablica istine za OR funkciju*

definiraju kao računski visoko učinkoviti makroi koji su dostupni u danas najpopularnijim FPGA procesorima [55, 57].

Blokovi RAM memorije su korisni za spremanje većih količina podataka kao što su vektori ili za razmjenu podataka između paralelnih zadataka. Uvijek postoji opcija za implementaciju vektora putem Flip-Flopova, ali oni zauzimaju veliku količinu FPGA resursa. Npr. 32-bitni vektor od 100 elemenata zauzima više od 30% Flip-Flopova dok implementiran u obliku RAMa zauzima samo 1% resursa. Važnost RAMa dolazi do izražaja prilikom spomenutog paralelizma zadataka pogotovo ako se dva zadatka izvode na različitim taktovima procesora. Upotrebom RAMa u tom slučaju neće doći do zagušenja i gubitka podataka [55].

5.2.3 Komunikacija

S obzirom da se cRio sastoji od Real-Time i FPGA procesora, za maksimalno iskorištavanje cijelog sustava potrebno je koristiti prednosti oba sustavu za što je potrebno uspostaviti komunikaciju između njih. CRio izbjegava upotrebu klasične sabirnice za komunikaciju između dva procesora tako što su oba procesora smještena na istu tiskanu pločicu što ubrzava razmjenu podataka između njih. Na cRiu se razlikuju dva tipa komunikacije: interprocesna i interprocesorska. Interprocesna komunikacija se uglavnom odnosi na razmjenu podataka između dvije ili više petlji, dok se interprocesorska odnosi na razmjenu podataka između FPGA i Real-Time procesora ili između računala i cRio procesora [53].

Interprocesna komunikacija se može odvijati putem varijabli, blokova memorije, registara ili FIFOa (engl. First-In First-Out). Od varijabli LabVIEW podržava lokalne i globalne varijable. Upotrebom njih se podaci spremaju u Flip-Flopove FPGA procesora. Varijable spremaju samo zadnji podatak koji je zapisan u njih te su dobar izbor ako nije potrebno koristiti svaki podatak koji se obrađuje [53].

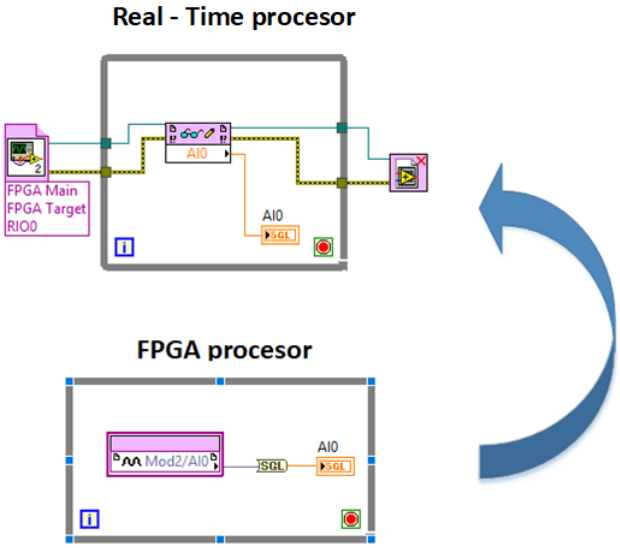
Još jedan od načina na koji se mogu prenositi samo posljednji podaci, ali i veća količina podataka, su blokovi memorije. Upotreba blokova memorije troši nekoliko logičkih blokova ali većim dijelom koristi RAM. FPGA procesor ima dva tipa memorije: onu koja se odnosi na cijeli procesor i onu koja se odnosi samo na VI kojim je definirana memorija. Prvom tipu memorije može pristupiti svaki VI koji se nalazi na FPGA procesoru, dok se VI definirana memorija

uglavnom koristi ako VI na koji se odnosi obrađuje veću količinu podataka. Svakom podatku koji se sprema u memoriju se može definirati adresa te je tako prikladna za spremanje vektora podataka [53].

Registri se također koriste ako je potrebno prenijeti samo posljednju vrijednost iz jedne petlje u drugu no prednost naspram varijabli i blokova memorije im je što dobro funkcioniraju ako se podaci prenose između petlji koje rade na različitim taktovima procesora [58].

Ako se komuniciraju poruke ili ažuriranja, ili se prenosi velika količina podataka između dvije ili više petlji koriste se FIFOi. FIFO je podatkovna struktura koja sprema elemente redoslijedom kojim ih prima. Slično kao kod blokova memorije postoje dvije vrste FIFOa: FIFOi koji se odnose na cijeli procesor i FIFOi koje definira VI te im je funkcija ista kao i kod blokova memorije. Kada se definira FIFO, potrebno je odrediti u kojim FPGA resursima će FIFO spremati podatke. Flip-Flopovi se koriste za spremanje podataka kod prilično malih FIFOa (do 100 bytova) te im je prednost što imaju najveću brzinu. LookUp tablice se koriste kod FIFOa do 300 bytova dok se FIFOi koji spremaju podatke u blokove memorije koriste prilikom prijenosa velike količine podataka te ako se Flip-Flopovi i LookUp tablice čuvaju za druge funkcije [53].

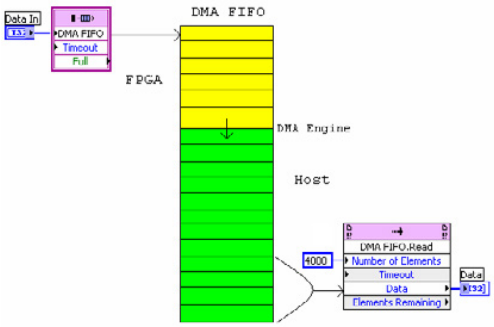
Za razmjenu podataka između procesora se mogu koristiti dva načina komunikacije: kontrole i indikatori Front Panela i DMA (engl. Direct Memory Access) FIFOi. Kontrole i indikatori Front Panela se koriste samo kada je potrebno prenijeti zadnju vrijednost. Vrijednost se tada sprema u kontrolu ili indikator te joj se može pristupiti koristeći "Read/Write Control" funkciju. Slika 5.6 prikazuje primjer takvog načina komunikacije. FPGA procesor, koji komunicira izravno sa ulazno/izlaznim modulima, čita vrijednost na analognom ulazu te ju sprema u indikator, dok RTOS pomoću "Read/Write Control" funkcije čita tu spremljenu vrijednost i prikazuje ju, u ovom slučaju, na indikatoru u svom Front Panelu [53].



Sl. 5.6: *Interprocesorska komunikacija upotrebom kontrola i indikatora Front Panela*

Ako je potrebno prenijeti veliku količinu podataka između FPGA i Real-Time procesora

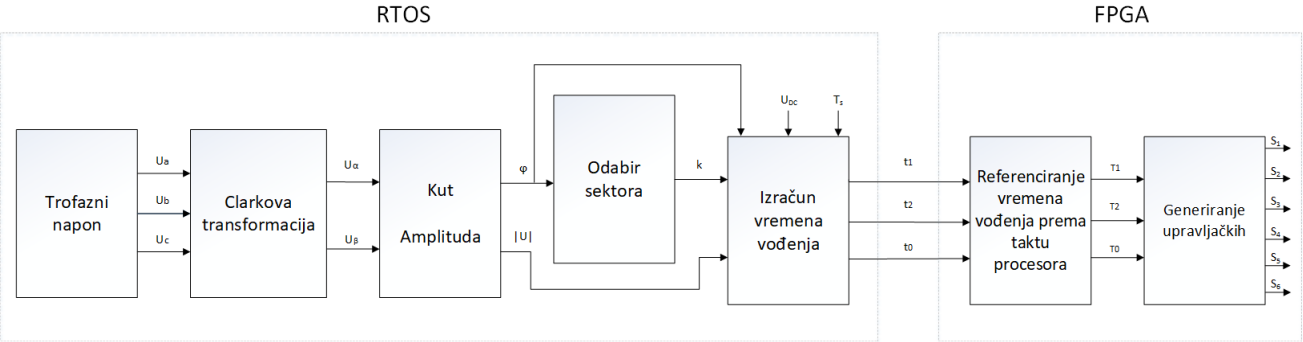
ili je potreban mali buffer za komunikaciju zapovijedi ili poruka koriste se DMA FIFOi. DMA ne koristi Real-Time procesor kada čita podatke s FPGA što ga čini najbržom metodom za razmjenu podataka. Problem s DMA FIFOima može nastati ako se prenose podaci s više kanala ili ako se ne pazi jesu li Real-Time procesor i FPGA sinkronizirani što može dovesti do zagušenja buffera i gubitka podataka. Slika 5.7 prikazuje mehanizam DMA FIFOa [53].



Sl. 5.7: *DMA FIFO način razmjene podataka između procesora [52]*

5.3. Implementacija na cRio-9024 platformi

Implementacija na cRio platformi je, kao i implementacija u Matlab/Simulinku, izvršena prema algoritmu danom na slici 4.1. Algoritam je dodatno podijeljen na dva dijela te je jedan dio implementiran na Real-Time procesoru, a drugi na FPGA procesoru. Dodatna podjela algoritma je prikazana na slici 5.8.



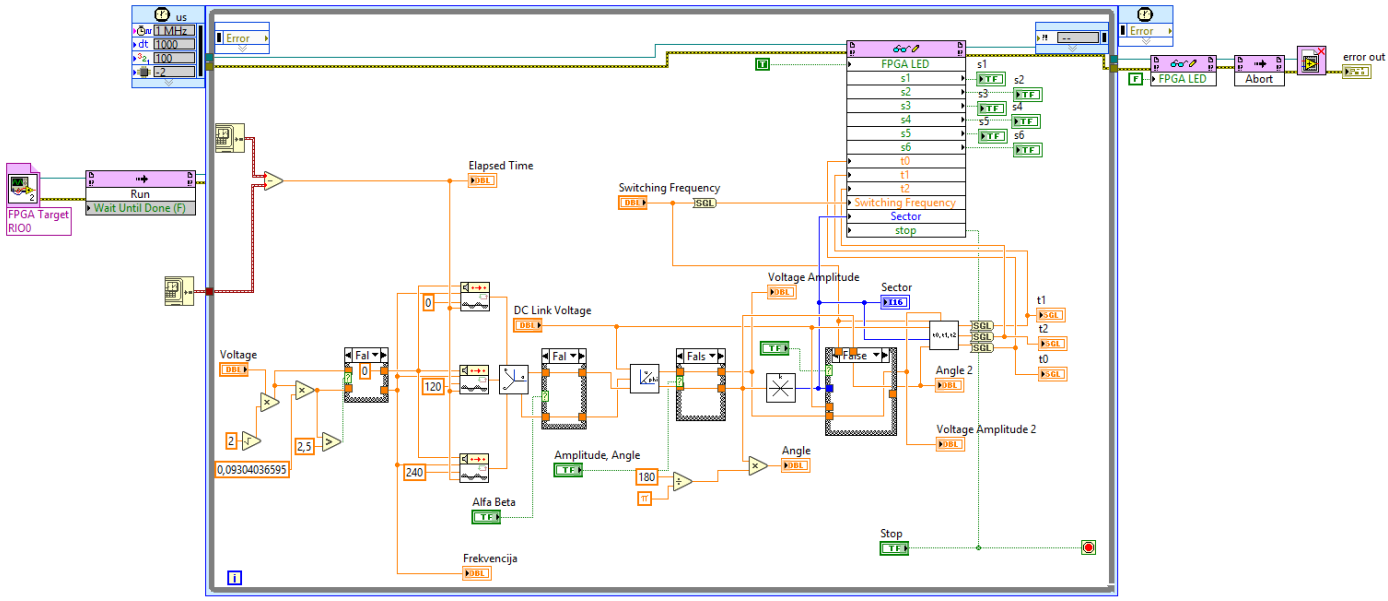
Sl. 5.8: *Blok dijagram algoritma implementiranog na cRio-9024 platformi*

Algoritam je podijeljen tako da RTOS obavlja računski zahtjevne zadatke koje bi FPGA procesoru uzimale puno resursa. Na RTOS je dakle implementirano generiranje trofaznog napona, Clarkova transformacija, proračun kuta i amplitude, odabir sektora, te proračun vremena vođenja. Na FPGA procesoru je implementirano referenciranje vremena vođenja prema taktu procesora te generiranje upravljačkih signala. Podjela je napravljena tako da se uzme u obzir upotreba resursa FPGA procesora, kao i količina podataka koja dva procesora međusobno komuniciraju. Naime implementacija se mogla izvesti tako da se i referenciranje vremena vođenja prema taktu procesora prebaci na RTOS čime bi se dodatno rasteretio FPGA procesor.

No posljedica toga bi bilo dodatno opterećenje Real-Time procesora što bi njegovu vremensku preciznost i determinizam dovelo u pitanje. Također bi se povećala količina podataka koja se razmjenjuje. Naime, ako se blok "Time-Clock Calculation" prema slici 5.8 ostavi na FPGA procesoru komuniciraju se samo tri podatka svaki takt procesora, a to su vremena vođenja t_0, t_1 i t_2. Ako se taj isti blok prebaci na RTOS, te uzevši u obzir sliku 4.8 odnosno da je za generiranje impulsa potrebno odrediti vremenske trenutke označene na toj slici, dolazi se do 5 podataka koje je potrebno komunicirati ($t_0/4$, $t_1/2$, $t_2/2$, $T_s/2$, T_s) koji se računaju u tom bloku, što će se vidjeti kasnije.

Upotreba resursa je jedan od najvećih zahtjeva na koji treba paziti, kako na cRiu, tako i na svakom FPGAu. Poželjno je otprilike znati koliko koja funkcija koristi resursa te pokušati izbjeći ili zaobići upotrebu više zahtjevnih funkcija. Npr. operacija dijeljenja koristi više resursa nego operacija množenja, "Select" funkcija u slučaju bool, signed i unsigned 8 i 16 bitnih integera koristi manje resursa od "Case Structure" itd. Tablica koja pokazuje približnu upotrebu resursa pojedinih funkcija će biti dana u prilogu.

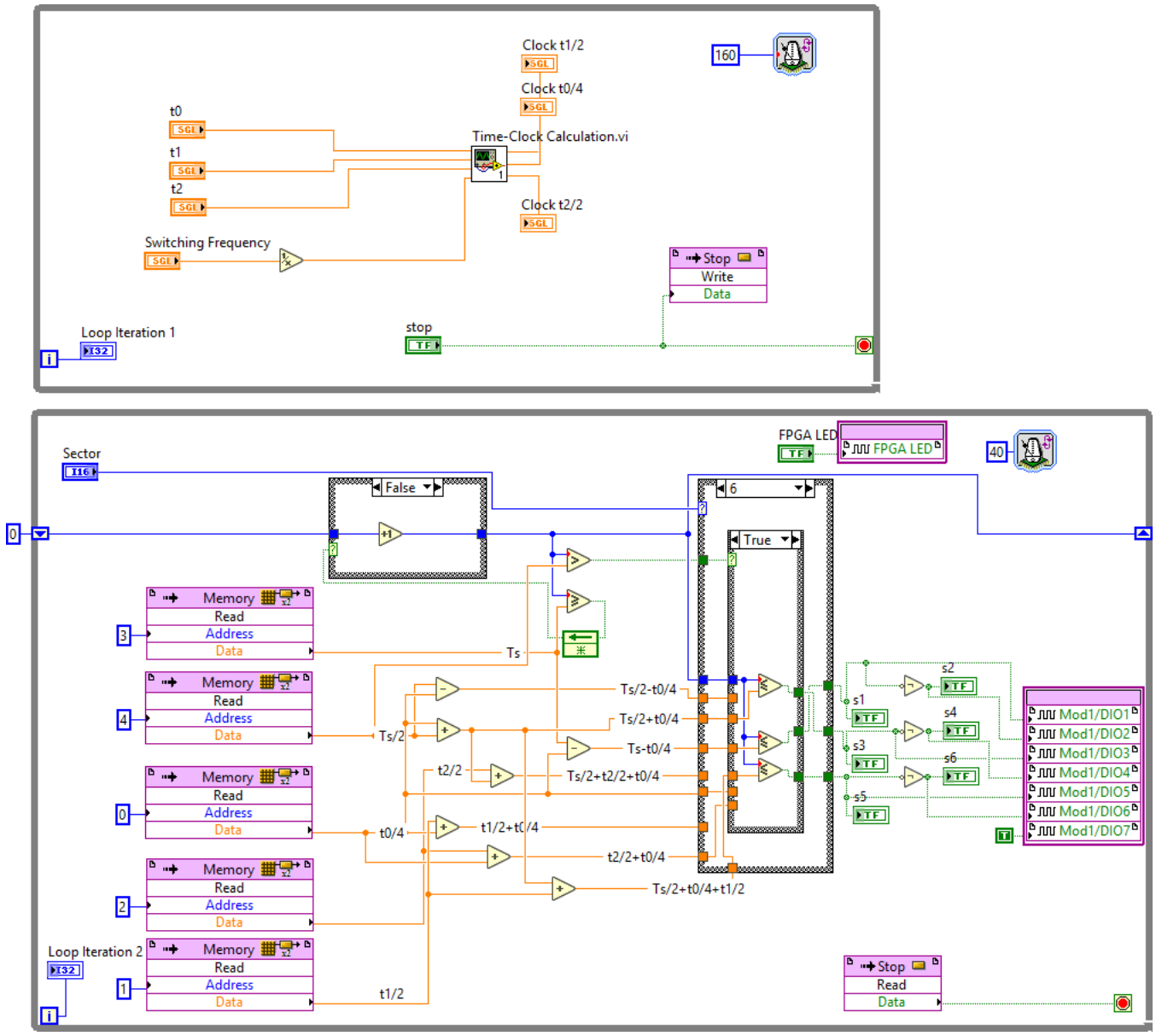
Slike 5.9 i 5.10 prikazuju implementirane algoritme na cRio platformi. Implementacija se izvodila na isti način kao i u Matlab/Simulinku uz određene preinake karakterističnе za cRio platformu.



Sl. 5.9: *Dio algoritma implementiran na Real-Time procesoru*

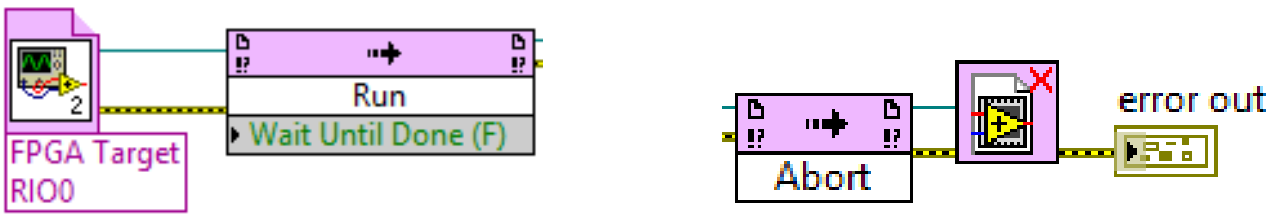
Promotri li se slika 5.9 primjećuje se upotreba petlje, točnije "Timed" petlje koja garantirana da će se algoritam unutar nje izvršiti u specificiranom vremenu što je u ovom slučaju 1 MHz. CRio ugradbeni sustav se može koristi kao HIL (engl. Hardware in Loop) te da bi algoritme kao što je prostorno-vektorska modulacija izvršavao ciklički nužna je upotreba petlji. CRio na izbor ima While, For i Timed petlje. Timed petlja je odabrana jer pruža zahtijevanu razinu vremenske preciznosti koja je nužna za ovu modulaciju.

Blokovi prikazani slikom 5.11 koriste se samo ako se koristi interprocesorska komunikacija. S obzirom da se na FPGA procesor može implementirati više algoritama koji mogu raditi



Sl. 5.10: *Dio algoritma implementiran na FPGA procesoru*

paralelno, ovi blokovi određuju RTOSu sa kojim VIom na FPGA procesoru komunicira te omogućuju komunikaciju.



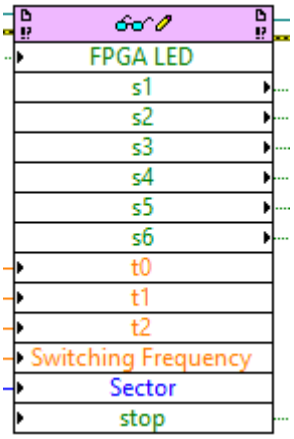
(a) **(b)**

Sl. 5.11: *Blok: a) Open VI Reference, b) Close VI Reference*

"Invoke Method" blok na slici 5.11a je postavljen kao "Run" funkcija što znači da on po-

kreče izvršenje dijela algoritma na FPGA procesoru, dok je isti taj blok na slici 5.11b postavljen kao "Abort" funkcija, odnosno on zaustavlja algoritam na FPGA procesoru. Time je omogućeno upravljanje cijelim algoritmom s Real-Time procesora ili izradom Real-Time aplikacije.

Komunikaciju između dva procesora se odvija preko kontrola i indikatora Front Panela pomoću bloka "Read/Write Control" prikazanog slikom 5.12. Podaci koji se šalju na FPGA su naredba za paljenje LEDa na cRiu, vremena vođenja, sklopna frekvencija, sektor u kojem se vektor trenutno nalazi te naredba "Stop" kako bi se dijelovi algoritma na RTOSu i FPGAu istovremeno zaustavili. RTOS s FPGA procesora prima upravljačke impulse te ih prikazuje u obliku LED lampica na Front Panelu.



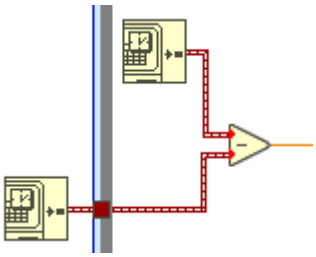
Sl. 5.12: *Blok Read/Write Control*

Na FPGA procesoru se nalaze dvije paralelne petlje, u jednom je "Time-Clock Calculation" SubVI a druga je zadužena za generiranje impulsa. Za razliku od RTOSa, na FPGA procesoru su korištene dvije While petlje. "Single-Cycle Timed Loop" (SCTL) se ne mogu koristiti sa mnogim LabView funkcijama te iz tog razloga nisu ni u ovom slučaju korištene. Za komunikaciju između te dvije petlje se koriste blokovi memorije i registri. Blokovi memorije služe za prijenos vremena vođenja izraženim u otkucajima procesora, dok se registar koristi za prijenos "Stop" naredbe. Također se obije petlje vrte različitim brzina. Gornja petlja na slici 5.10 se vrti sporije od donje što omogućuje dodatno selektiranje broja generiranih vektora po jednoj periodi.

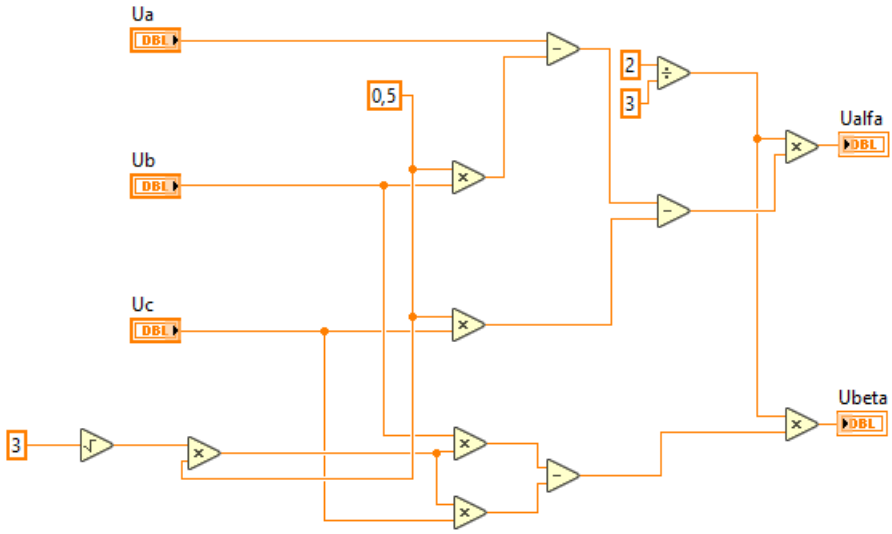
Kao i u Matlab/Simulinku, implementacija na cRiu je također prema blok dijagramu na slici 4.1 podijeljena na 7 dijelova. Njih 6 je implementirano u obliku podsustava odnosno SubVIa osim generiranja impulsa koje radi jednostavnosti nije smješteno u podsustav.

Trofazni napon se generira pomoću "Sine Wave PtByPt VI" bloka kojem se zadaju amplituda napona, frekvencija, fazni pomak i referenca vremena. Fazni pomak između napona je 120°, dok se amplitudom i frekvencijom upravlja preko Front Panela i to U/f regulacijom. Referenca vremena se postiže pomoću bloka "Get Date/Time in Seconds". Jedan blok se postavi izvan petlje, dok se drugi postavi unutar petlje, njihova razlika je proteklo vrijeme od pokretanja izvršetka algoritma. Dobivanje proteklog vremena je prikazano slikom 5.13.

Sljedeći korak je Clarkeova transformacija koja se provodi na isti način kao i u Simulinku te je prikazana slikom 5.14.



Sl. 5.13: *Proračun proteklog vremena od trenutka pokretanje izvršenja algoritma*

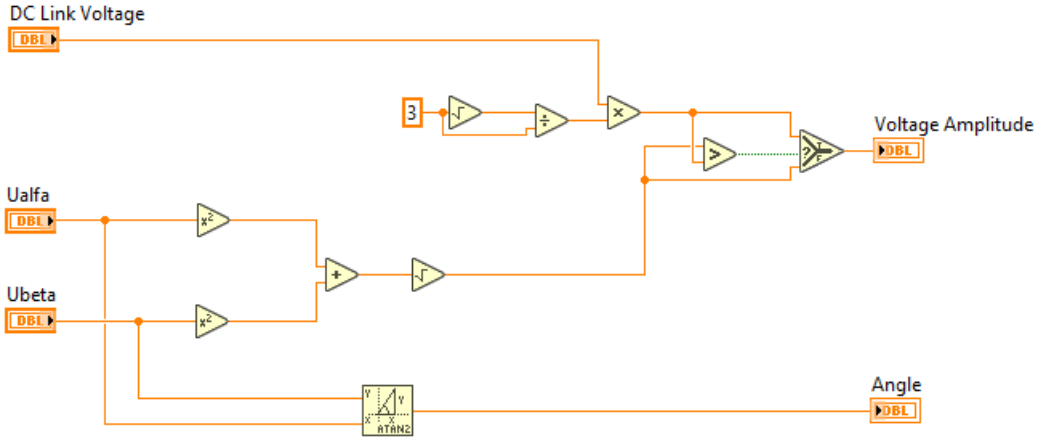


Sl. 5.14: *SubVI Clarkeova transformacija*

Za razliku od Simulinka gdje su se za određivanje kuta i amplitude koristili blokovi "Real-Imag to Complex" i "Complex to Magnitude-Angle" u LabViewu se određuju prema sljedećem izrazu:

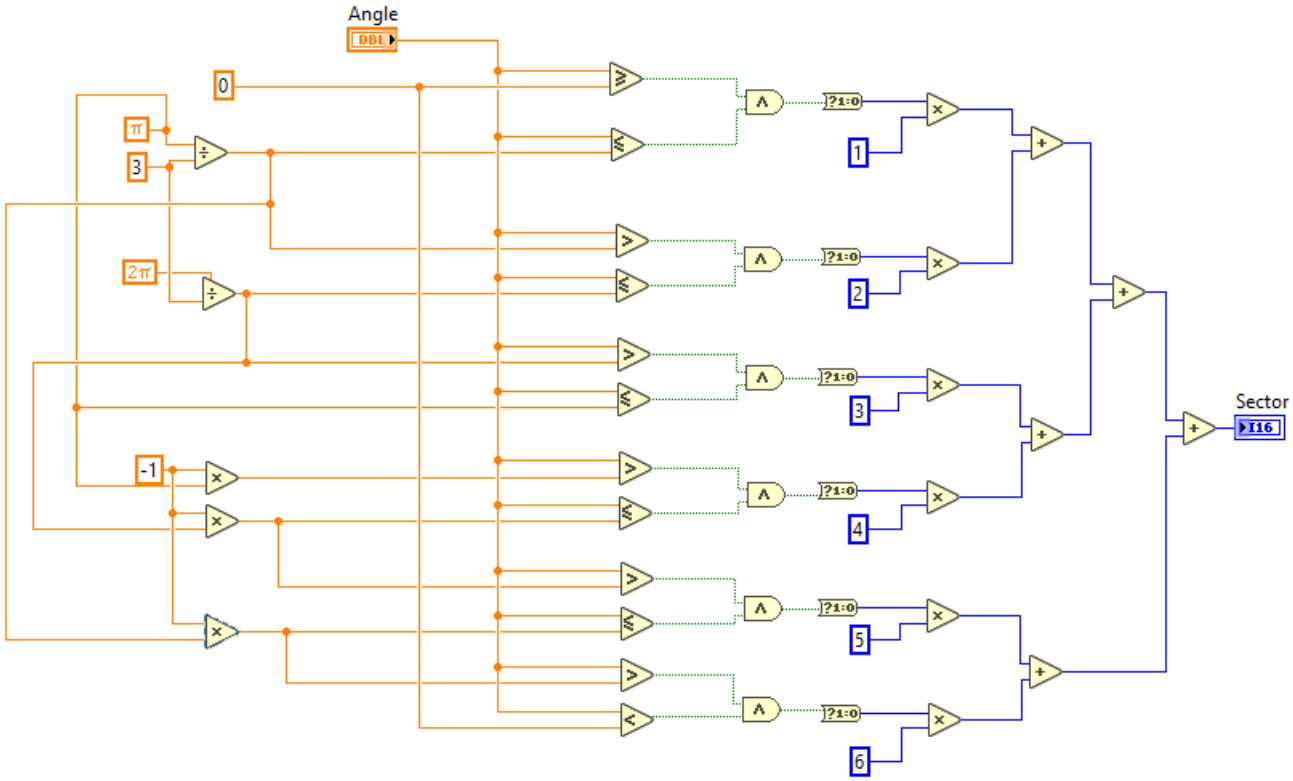
$$\begin{aligned} \varphi &= \arctan(\frac{u_\beta}{u_\alpha}) \\ |U| &= \sqrt{u_\alpha^2 + u_\beta^2} \end{aligned} \tag{5-1}$$

Određivanja kuta i amplitude u LabViewu je prikazano slikom 5.15.



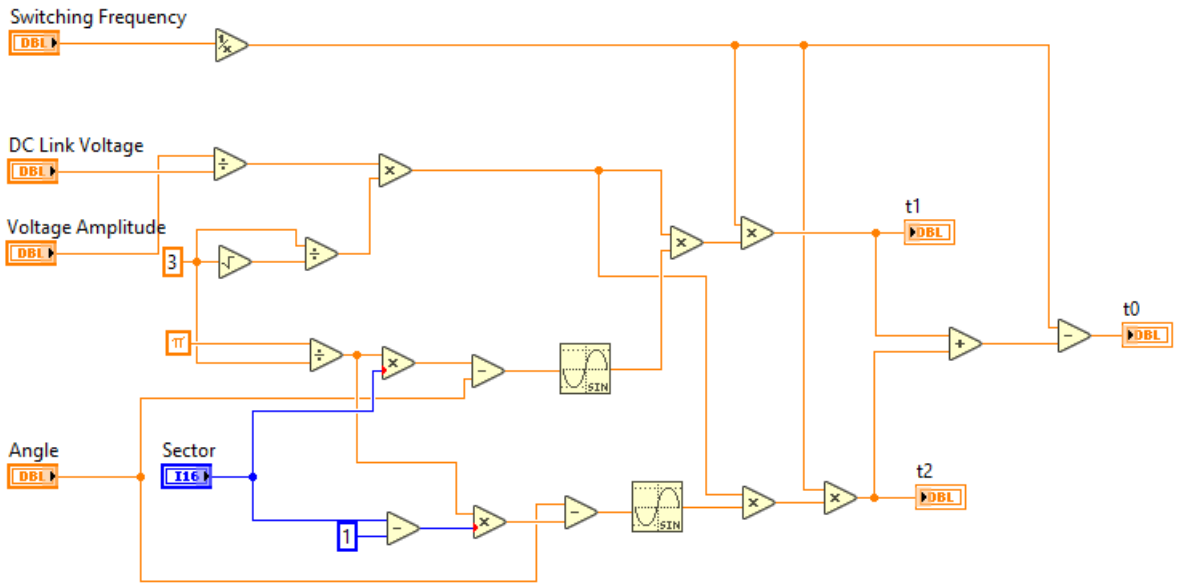
Sl. 5.15: *Određivanje kuta i amplitude*

Određivanje sektora i proračun vremena vođenja se također implementiralo istom logikom kao i u Simulinku te su oba SubVIa prikazana na slikama 5.16 i 5.17.



Sl. 5.16: *SubVI za određivanje sektora*

Sljedeći korak je slanje podataka na FPGA procesor no prije nego se pošalju moraju se pretvoriti u Single-Precision Floating-Point (SGL) tip podataka. CRiov FPGA procesor podržava samo SGL i Fixed-Point (FXP) tipove podataka za prikaz decimalnih brojeva. Razliku između SGL i FXP je najjednostavnije objasniti pomoću slika 5.18a i 5.18b. SGL tip podataka se sastoji od predznaka, mantise koja predstavlja broj u znanstvenom zapisu i eksponenta s kojim se taj broj skalira. FXP tip podataka se sastoji od Integera i frakcije koja slijedi iza decimalnog zareza. Dakle razlika je u načinu kako FPGA procesor sprema i obrađuje varijable.

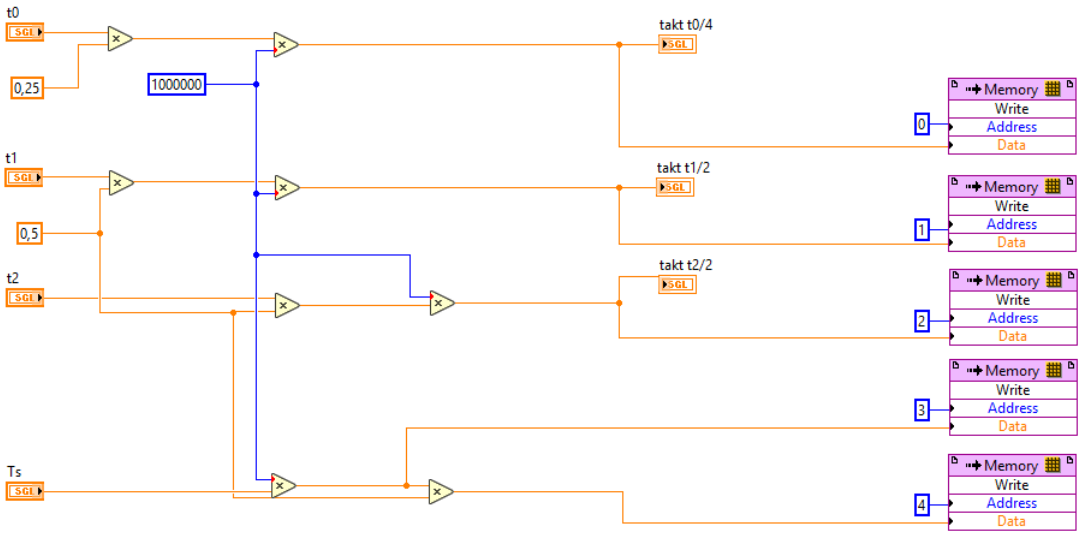


Sl. 5.17: *SubVI za proračun vremena vođenja*

(a) **(b)**

Sl. 5.18: *a) SGL tip podataka, b) FXP tip podataka*

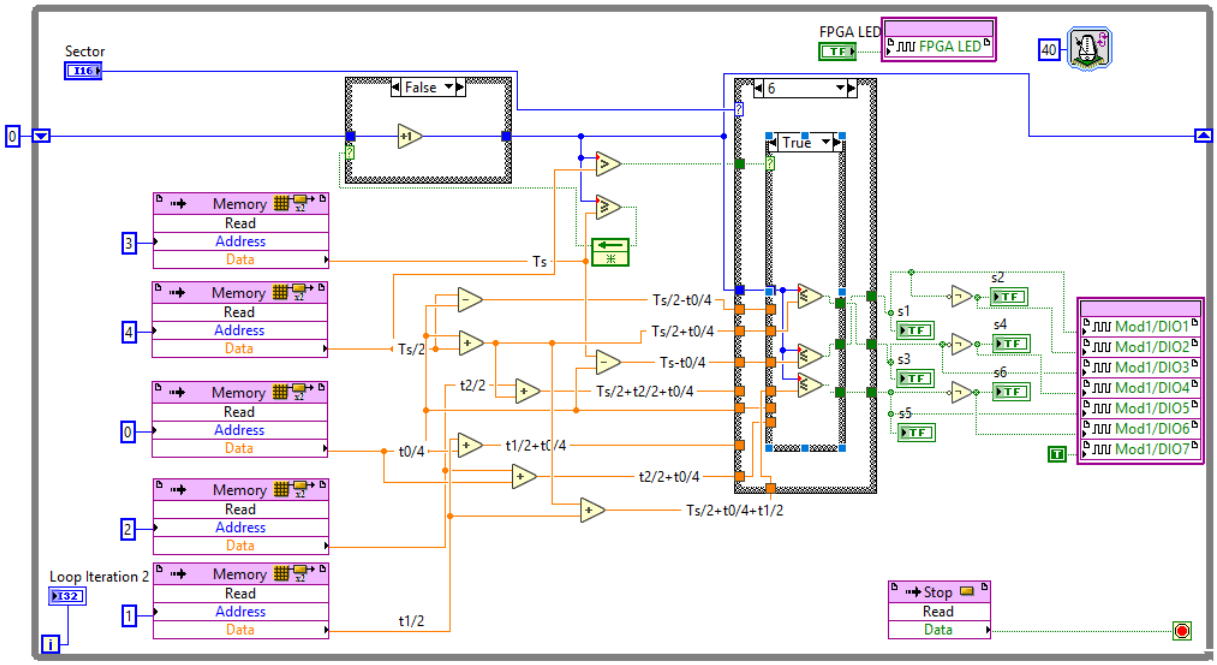
Podaci se dalje šalju "Time-Clock Calculation" SubVIu na FPGA procesoru koji proračunata vremena vođenja referencira prema taktu procesora. SubVI je prikazan slikom 5.19.



Sl. 5.19: *Time-Clock Calculation SubVI*

U konačnici ostaje generiranje impulsa što je prikazano slikom 5.20. Impulsi se generiraju

tako da je izrađen brojač koji broji do iznosa jedne sklopne periode izraženog u otkucajima procesora. Sklopni obrazac je dodatno podijeljen na dva dijela, na vrijednosti manje ili jednake $T_s/2$ i na vrijednosti koje su veće. Ta dva slučaju implementirana su preko "Case" strukture koja ima "True" i "False" slučaj. Kada je brojač veći od $T_s/2$ aktivan je "True" slučaj, kad je manji aktivan je "False" slučaj. Ta "Case" struktura se nalazi unutar druge koja služi za prebacivanje između sektora. Na primjeru sektora 1, kada je brojač manji od $T_s/2$ te u onom trenutku kada postane veći od $t_0/4$ izlaz na cRiu koji generira impulse za ventil S_1 postaje aktivan, a izlaz zadužen za ventil S_2 postaje neaktivan. Tako se redom pale i gase izlazi prema vremenskim trenutcima određenim na slici 4.8. Slučaj kada je brojač manji od $T_s/2$ je zadužen za paljenje ventila pozitivne grane izmjenjivača, dok gasi ventile negativne grane. Kada brojač postane veći od $T_s/2$ "Case" struktura mijenja slučaj u "True" te se opet na isti način gase ventili pozitivne grane izmjenjivača, a pale ventili negativne grane. Analogno vrijedi za ostale sektore. Komunikacija između FPGA procesora i I/O modula se vrši preko "FPGA I/O Node" bloka koji je potrebno prvo konfigurirati da služi kao digitalni ulaz ili izlaz.



Sl. 5.20: *Generiranje impulsa implementirano na FPGA procesoru SubVI*

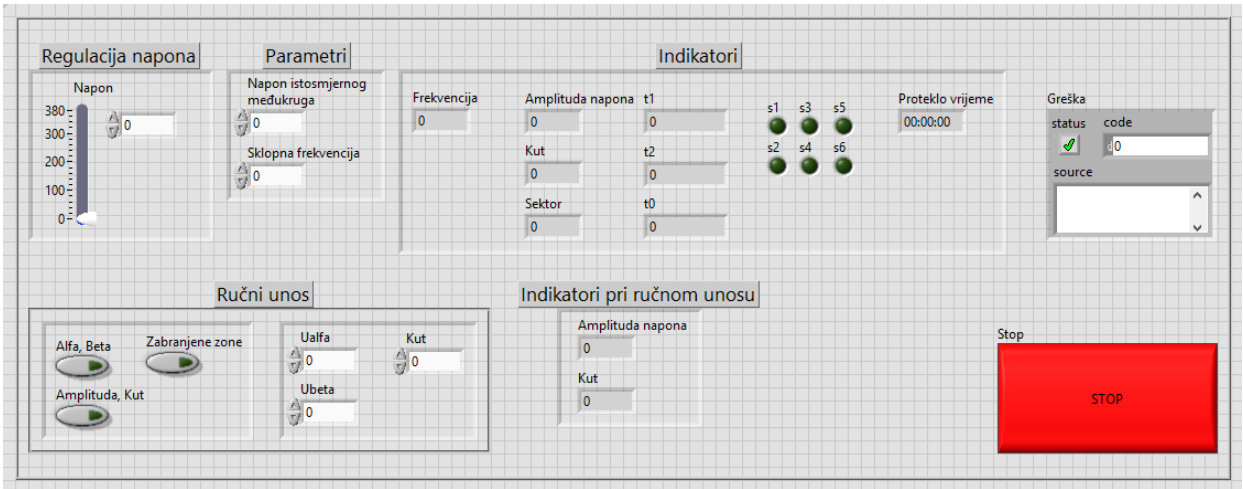
Na kraju, kako bi se algoritam mogao izvoditi na cRio platformi potrebno ga je prevesti. Algoritam se može prevesti na tri načina. Prvi je lokalno preko instaliranih Xilinx Compilation Toolsa što je dugotrajan postupak koji, ovisno o veličini algoritma, može potrajati duže od 24 sata. Drugi je da se na jedno računalo, ako je dostupno, instalira Linux operacijski sustav te se od računala napravi server za prevođenje što ubrzava vrijeme prevođenja za 20% - 50% ovisno o specifikacijama računala. I treći, ujedno i najbrži način, je korištenjem National Instrumentsovog Compile Clouda gdje postupak prevođenja traje uglavnom do sat vremena. Po završetku prevođenja LabView izbaci status prevođenja odnosno izvještaj gdje se mogu pregledati postavke korištene za prevođenje te utilizacija resursa na FPGA procesoru. Upotreba

resursa FPGA prcesora za implementirani algoritam je prikazana tablicom 5.1.

Tab. 5.1: *Upotreba resursa FPGA procesora za implementirani algoritam prostorno-vektorske modulacije*

Resurs	**Iskorišteno**	**Ukupno**	**Iskorištenost u postotcima**
Ukupno Sliceova	4591	7200	63.8 %
Slice registri	8470	28800	29.4 %
Slice LUT	13388	28800	46.5 %
Block RAM	1	48	2.1 %
DSP48	8	48	16.7 &

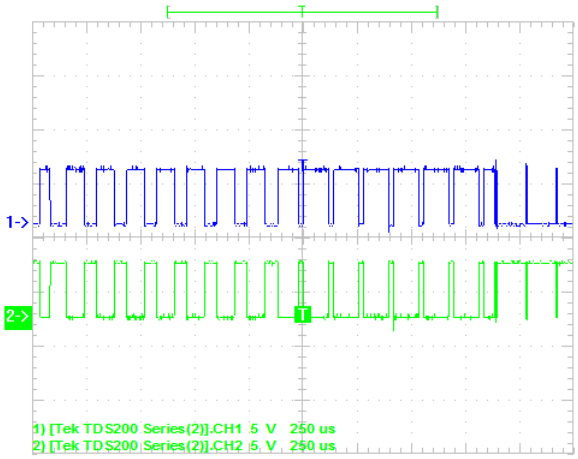
Na kraju je potrebno naglasiti da je cijeli algoritam izveden tako da se njime može upravljati samo sa RTOSa odnosno preko njegovog Front Panela što je prikazano slikom 5.21. Na Front Panelu se nalazi klizač preko kojeg se regulira amplituda napona. S obzirom da se radi o U/f regulaciji prema iznosu napona se automatski skalira vrijednost frekvencije koja se vidi na postavljenom indikatoru za frekvenciju. Nadalje su još postavljeni indikatori za proračunati kut i sektor kao i za vremena vođenja t_0, t_1 i t_2. Također su postavljene kontrole za unošenje napona istosmjernog međukruga i sklopne frekvencije. Također se još nalaze LED lampice koje se pale i gase prema generiranim upravljačkim impulsima. Omogućeno je praćenje proteklog vremena te grešaka koje cRio javlja u slučaju neuspjele komunikacije između dva procesora. S obzirom da je algoritam primarno namijenjen za testiranje mogućnosti cRio platforme te same prostorno-vektorske modulacije dodani su tasteri koji su povezani na "Case" strukture te služe za omogućivanje ručnog unosa α i β komponenti napona, amplitude i kuta u radijanima. Dodan je i taster koji također preko "Case" strukture omogućuje testiranje utjecaja zabranjenih zona prikazanih slikama 2.8b i 2.11. Međutim pri sklopnim frekvencijama korištenima u eksperimentu vektor napona ne ulazi u nijednu od navedenih zabranjenih zona tako da njihov utjecaj nije prikazan ovim radom. U konačnici se tu nalazi "Stop" taster koji zaustavlja algoritam na oba procesora.



Sl. 5.21: *Front Panel RTOSa*

5.4. Rezultati implementacije na cRio-9024 platformi

Upravljački impulsi generirani preko cRia mjereni su Tektronix TDS 224 osciloskopom na cRi-ovom NI 9401 modulu. Osciloskop je bio spojen s računalom RS232 protokolom te se akvizicija upravljačkih impulsa vršila Wavestar softverom. Napon istosmjernog međukruga prema kojemu su se računala vremena vođenja iznosio je 660 V te je snimanje upravljačkih impulsa vršeno za tri frekvencije: 1, 5 i 10 kHz. Slika 5.22 prikazuje upravljačke impulse ventila S_1 i S_2.

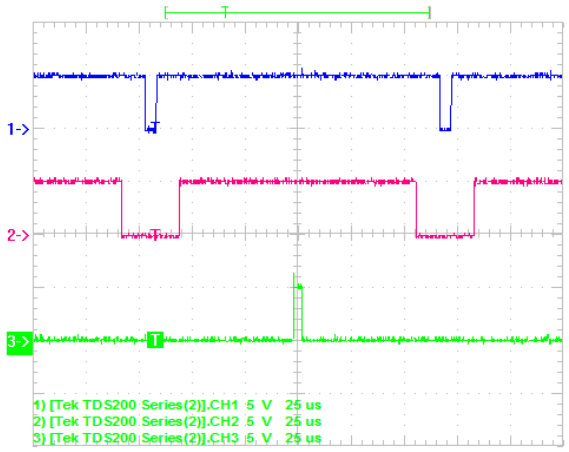


Sl. 5.22: *Upravljački impulsi ventila S_1 i S_2*

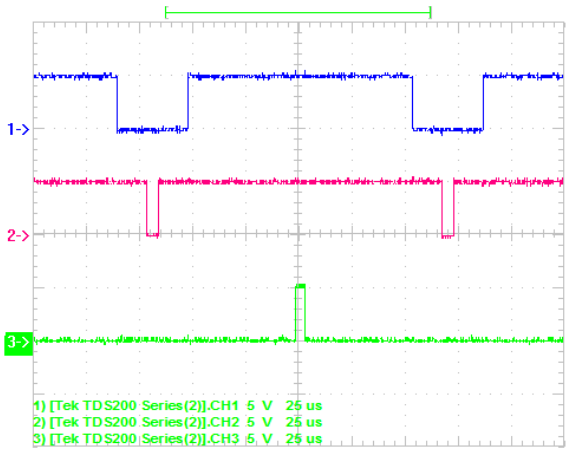
Usporede li se upravljački impulsi na slici 5.22 sa impulsima na slici 4.12 vidi se da se one podudaraju odnosno kao što je bio slučaj kod implementacije u Simulinku, i u ovom slučaju su signali komplementarni što znači da neće doći do kratkog spoja istosmjernog međukruga.

U nastavku će na slikama 5.23, 5.24 i 5.25 biti prikazani obrasci sklapanja za sve sektore i to redom za frekvencije 10, 5 i 1 kHz.

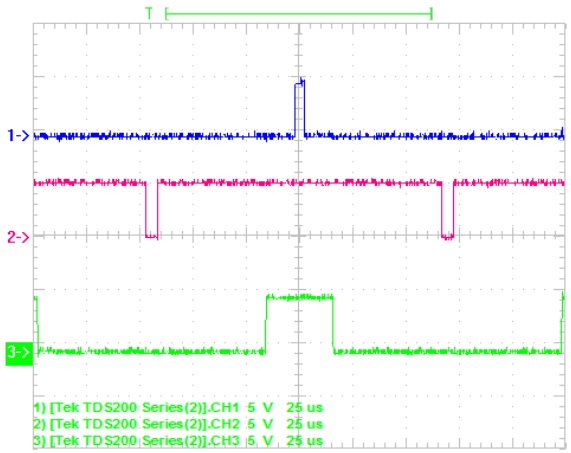
Usporede li se slike 5.23, 5.24 i 5.25 sa slikama 4.14, 4.15 i 4.16 te slikom 2.13 primjećuje se da se podudaraju odnosno da je postignut simetričan obrazac sklapanja ventila. Na slikama kanal 1 (plava boja) označava impulse ventila S_1, kanal 2 (ljubičasta boja) su impulsi ventila S_3 i kanal 3 (zelena boja) su impulsi ventila S_5.



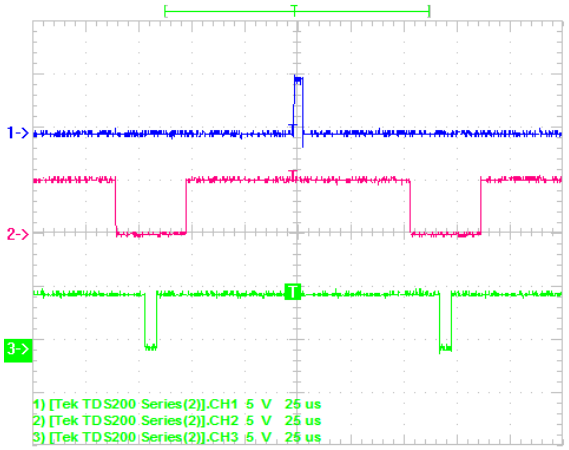
(a)



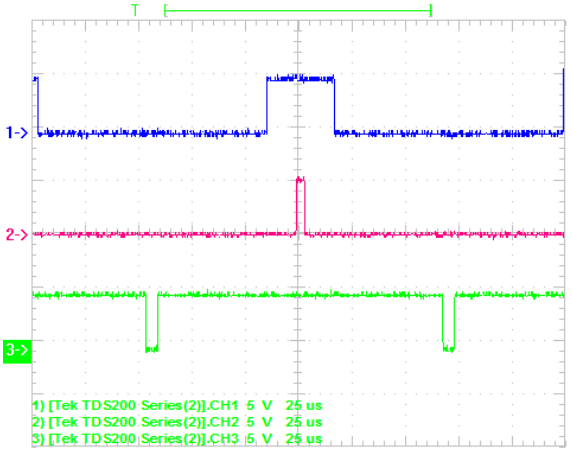
(b)



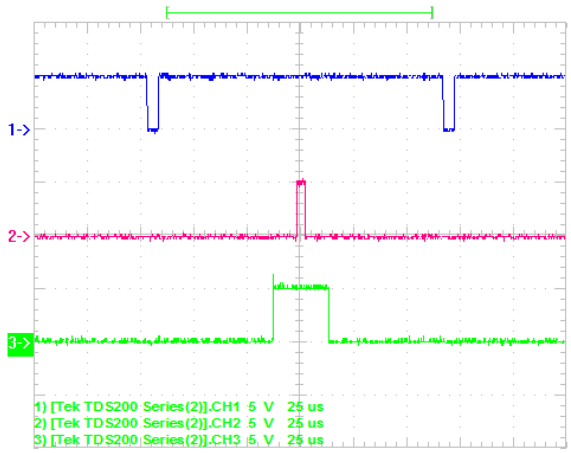
(c)



(d)

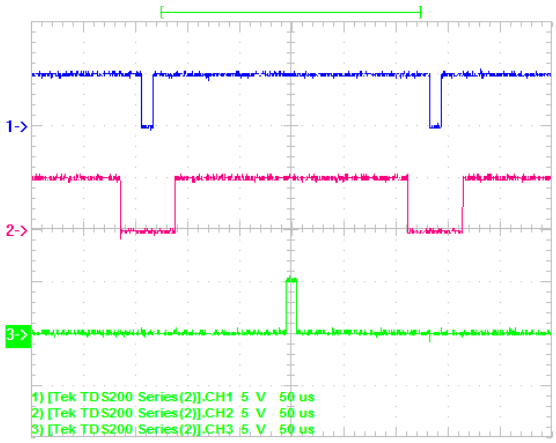


(e)

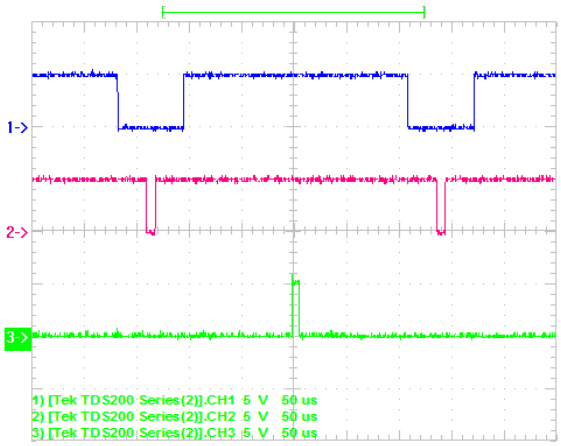


(f)

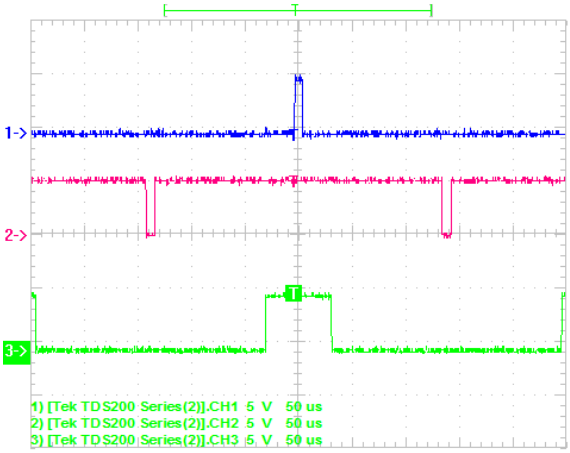
Sl. 5.23: *Simetrični obrazac sklapanja ventila postignut na cRiu pri* $f_s = 10kHz$ *za: a) Sektor 1, b) Sektor 2, c) Sektor 3, d) Sektor 4, e) Sektor 5, f) Sektor 6*



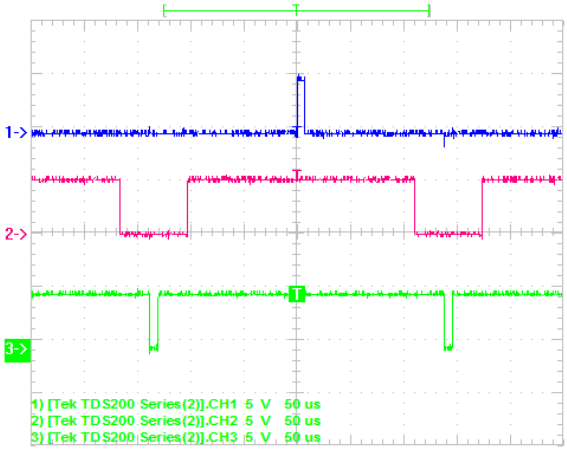
(a)



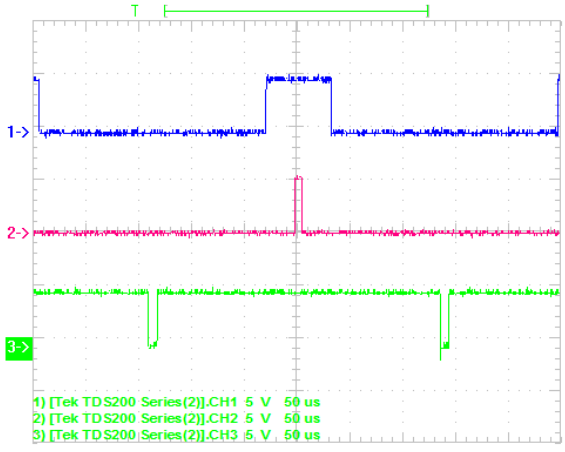
(b)



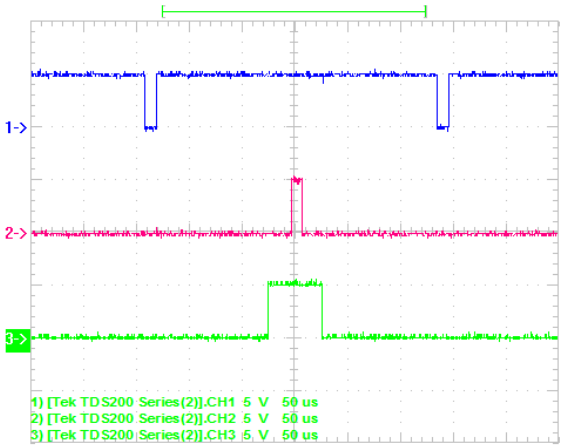
(c)



(d)

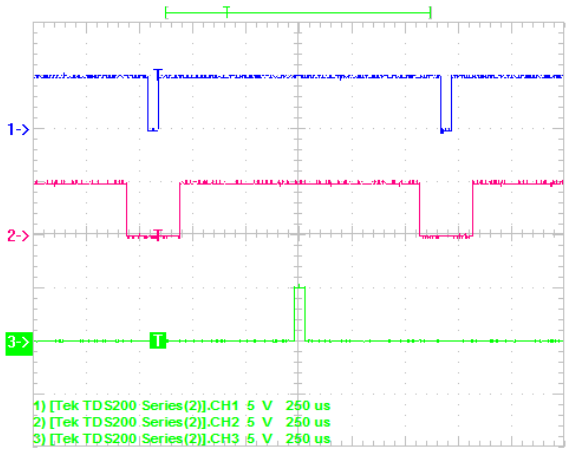


(e)

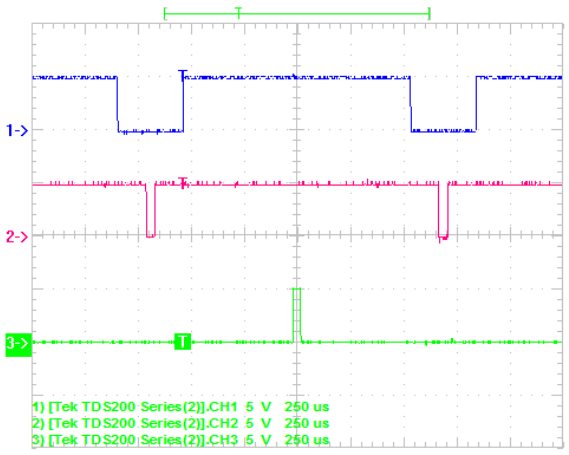


(f)

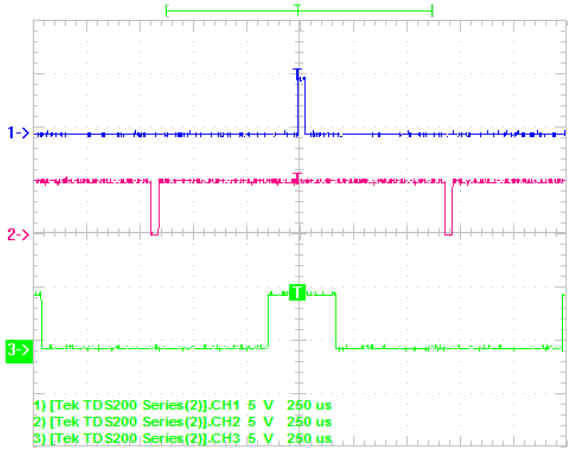
Sl. 5.24: *Simetrični obrazac sklapanja ventila postignut na cRiu pri $f_s = 5kHz$ za: a) Sektor 1, b) Sektor 2, c) Sektor 3, d) Sektor 4, e) Sektor 5, f) Sektor 6*



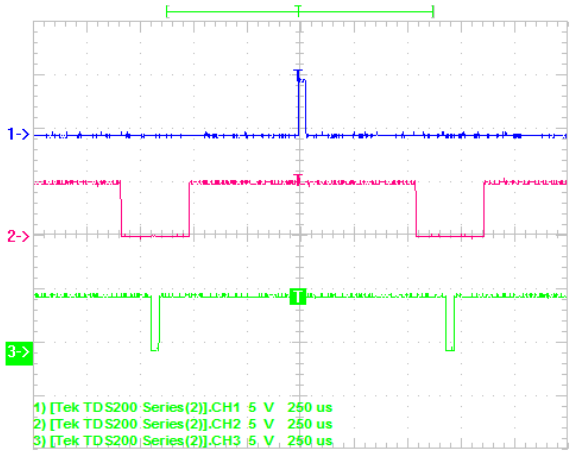
(a)



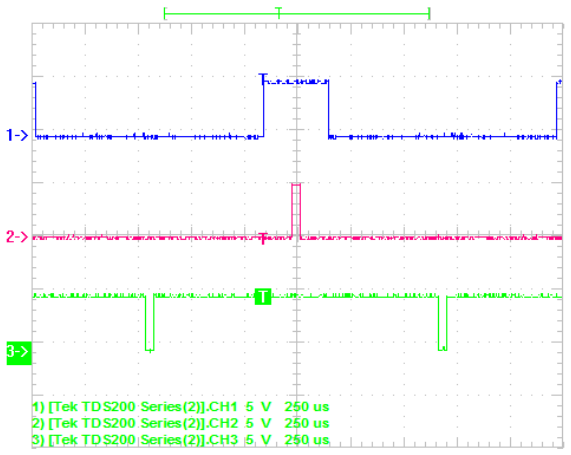
(b)



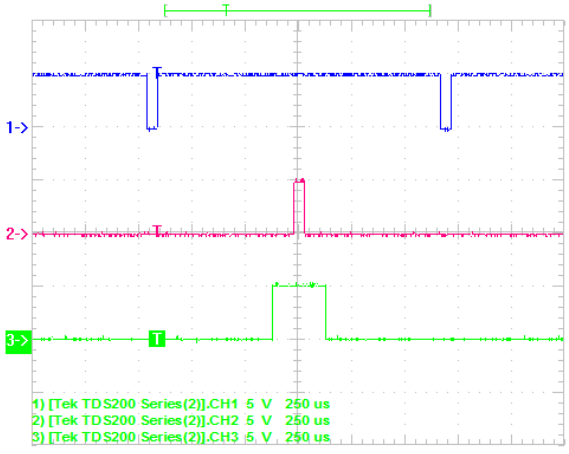
(c)



(d)



(e)



(f)

Sl. 5.25: *Simetrični obrazac sklapanja ventila postignut na cRiu pri $f_s = 1kHz$ za: a) Sektor 1, b) Sektor 2, c) Sektor 3, d) Sektor 4, e) Sektor 5, f) Sektor 6*

Tablica 5.2 prikazuje proračunata i realizirana vremena vođenja odnosno aktivna vremena, te odnos između aktivnog i neaktivnog stanja ventila. Prikazani rezultati su urađeni za prvi sektor za sve tri sklopne frekvencije. T_{si} je neaktivno stanje ventila odnosno stanje kada ventil ne vodi, a T_{sa} je aktivno stanje ventila odnosno stanje kada ventil vodi.

Tab. 5.2: *Izračunata i realizirana vremena vođenja na cRio platformi*

$\mathbf{f_s}$ **željeni [kHz]**	1	5	10
$\mathbf{f_s}$ **realizirano [kHz]**	0.8	4	7.14
$\mathbf{t_1}$ **računski** [μs]	146.667	29.3333	14.6667
$\mathbf{t_1}$ **realizirano** [μs]	150	30	22.5
$\mathbf{t_2}$ **računski** [μs]	783.327	156.665	78.3327
$\mathbf{t_2}$ **realizirano** [μs]	1000	200	107.5
$\mathbf{t_0}$ **računski** [μs]	70.0065	14.0013	7.00065
$\mathbf{t_0}$ **realizirano** [μs]	100	20	10
$\mathbf{T_{si}/T_{sa}}$ **računski**	0.03627	0.03627	0.03627
$\mathbf{T_{si}/T_{sa}}$ **realizirano**	0.04167	0.0.4167	0.03702

Analizira li se tablica 5.2 primjećuju se odstupanja između izračunatih i realiziranih vremena vođenja no detaljnijom analizom se dolazi do zaključka kako su odstupanja između proračunatih i realiziranih vremena vođenja, kada bi se izrazili u postotcima približno ista. Prema tome se može zaključiti kako unatoč odstupanjima cRio daje predvidive rezultate te prema tome ima predvidivo, deterministično ponašanja. Tolika odstupanja poglavito leže u tome što izvedena implementacija nije potpuna FPGA implementacije čime bi odstupanja bila svedena na minimum. S obzirom da je algoritam podijeljen na dva dijela radi optimizacije upotrebe resursa te je stoga potrebna komunikacija između dva procesora, odstupanja se događaju radi vremena koje je potrebno za komunikaciju, te radi upotrebe "While" petlji na FPGA procesoru umjesto SCTLa. SCTL garantira izvođenje algoritma u jednom taktu procesora uz minimalni "jitter" dok "While" petlja ne garantira istu razinu determinizma i vremenske preciznosti. Da cRio ima predvidivo ponašanje se također vidi po omjeru neaktivnog i aktivnog vremena aktivnosti ventila gdje su razlike minimalne. Omjeri između proračunatih i realiziranih vremena vođenja i vremena aktivnosti ventila, kao i omjer između željene i realizirane frekvencije sklapanja izraženi su tablično (5.3) u postotcima kako bi se predvidivost i determinizam cRia lakše uočili i analizirali.

Tab. 5.3: *Omjeri iskazani u postotcima*

$\mathbf{f_s}$ [kHz]	$\frac{\mathbf{f_{s,želj.}}}{\mathbf{f_{s,real.}}}$ [%]	$\frac{\mathbf{t_{1,rač.}}}{\mathbf{t_{1,real.}}}$ [%]	$\frac{\mathbf{t_{2,rač.}}}{\mathbf{t_{2,real.}}}$ [%]	$\frac{\mathbf{t_{0,rač.}}}{\mathbf{t_{0,real.}}}$ [%]	$\frac{\mathbf{T_{si,rač.}/T_{sa,rač.}}}{\mathbf{T_{si,real.}/T_{sa,real.}}}$ [%]
1	125	97.78	78.33	70.01	87.04
5	125	97.78	78.33	70.01	87.04
10	140.06	65.19	72.87	70.01	97.97

Prema tablici 5.3 se vidi kako omjeri proračunatih i realiziranih vremena vođenja uglavnom jednaki. Veliko odstupanje se primjećuje jedino kod vremena vođenja t_1 pri sklopnoj frekvenciji 10 kHz iz kojeg proizlazi i veliko odstupanje u realiziranoj sklopnoj frekvenciji. Naime to odstupanje može biti i rezultate analize snimljenih valnih oblika pri čemu se za određivanje vremena vođenja nisu koristili kursori koje nude osciloskop i Wavestar nego su određivani očitanjima prema podiocima i vremenskoj bazi koja je namještena na osciloskopu.

6. ZAKLJUČAK

Kroz rad je razvijen i implementiran algoritam prostorno-vektorske modulacije. Objašnjen je prikaz trofaznog sinusnog napona preko jednog prostornog vektora. Dan je osvrt na Clarkovu i Parkovu transformaciju te je prema njima izveden izraz za opću transformaciju. Clarkova transformacija se koristi samo prilikom transformacije troosnog koordinatnog sustavu fiksiranog za rotor asinkronog motora u dvoosni koordinatni sustav fiksiran za stator motora. Parkova transformacija transformira mirujući troosni koordinatni sustav fiksiran za stator sinkronog stroja u rotirajući koordinatni sustav fiksiran za rotor stroja. Opća transformacija je razvijena kako bi sadržavala sve do tada poznate transformacije u sebi. Uvrštavanje prikladne brzine vrtnje koordinatnog sustava u opću transformaciju mogu se dobiti sve do tada razvijene transformacije. Kroz analizu rada trofaznog izmjenjivača u mosnom spoju prikazano je 8 mogućih sklopnih stanja izmjenjivača u kojem svako stanje odgovara jednom prostornom vektoru od kojih su dva nul-vektora. Na osnovu sklopnih stanja i nadomjesne sheme izmjenjivača je napravljena analiza rada te su prikazani valni oblici faznog i linijskog napona. Dan je matematički model prostorno-vektorske modulacije odnosno izveden je izraz za proračun vremena vođenja. Izvedena su ograničenja prostorno-vektorske modulacije u smislu maksimalnog napona prostornog vektora i vektora koji su ostvarivi.

Poluvodičke komponente u teorijskoj se analizi modeliraju pomoću idealnih sklopki no u stvarnosti je njima potrebno određeno vrijeme da uklope i isklope. Ako generirani upravljački impuls ima vremensko trajanje manje od zbroja vremena uklopa i isklopa upravljivog ventila, ventil na nju neće reagirati što dovodi do povećanih sklopnih gubitaka. Na osnovu toga je izveden izraz za određivanje neostvarivih vremena vođenja ventila. Na osnovu obrazaca sklapanja i faznih napona izvedeni su valni oblici na izlazu izmjenjivača kod prostorno vektorske modulacije. U faznom naponu je uočljiva prisutnost harmonika trećeg reda koji su rezultat Clarkove transformacije koja nije potpuno reverzibilna. Prisustvo harmonika trećeg reda prostorno-vektorskoj modulaciji omogućuje veću iskorištenost napona istosmjernog međukruga nego kod sinusno pulsno-širinske modulacije. To i veća sloboda u odabiru broja generiranih vektora kao i obrazaca sklapanja joj daje prednost nad ostalim modulacijskim tehnikama.

Kroz rezultate implementacije modulacije u Matlab/Simulinku te kroz analizu rezultata je vidljivo kako primijenjena metoda modulacije ima predvidive rezultate te kako je prikladnija za veće frekvencije sklapanja.

Implementacija modulacije na cRio platformi je pokazala prednosti cRiovog predvidivog i determinističkog ponašanja. Iako su odstupanja postojana, ona se zadržavaju na otprilike istoj razini. Odstupanja se javljaju jer implementacija nije potpuna FPGA implementacija nego se koriste prednosti Real-Time procesora, koji nema ograničenja u vidu dostupnih resursa i broja funkcija. Nadalje Real-Time procesor zbog većih dostupnih resursa može obavljati matematički zahtjevne operacije, kao što je modulacija prostornog vektora, u kratkom vremenu. Upotreba Real-Time procesora za računski zahtjevan dio modulacije zahtjeva komunikaciju između Real-Time i FPGA procesora za koju je potrebno određeno vrijeme da se obavi što unosi odstupanja

između proračunatih i realiziranih vrijednosti vremena vođenja. Dodatna odstupanja unosi nemogućnost upotrebe SCTLa na FPGA procesoru koji garantiraju izvršenje implementiranog dijela algoritma u zadanom vremenu. I na kraju još jedan razlog odstupanja je pogreška pri mjerenju koja dolazi do izražaju pri višim sklopnim frekvencijama gdje su vremena vođenja sve manja te ih je sve teže očitati bez upotrebe preciznih instrumenata. Odstupanja također unosi i izrađeni brojač, koji i ima inkrement od +1 što je Integer tip podataka, dok su vremena vođenja prema kojim se referira SGL tip podataka. Problem je također što, iako računski moćan, Real-Time procesor proračun vremena vođenja izvodi svaki takt procesora što usporava izvršenje algoritma i blago povećava Jitter. Iako je to povećanje malo, kod vremenski preciznih algoritama kao što je modulacija prostornog vektora ipak dolazi do izražaja. Ipak, ovakva implementacija ima prednosti prilikom testiranja i istraživanja gdje se pri promjeni sklopne frekvencije i/ili iznosa napona gotovo trenutno dobije odziv.

Iz svega ovoga se mogu izvući preporuke za budući rad u vidu poboljšanja i optimizacije algoritma. Moguće je prilagoditi algoritam na FPGA procesoru kako bi bila omogućena upotreba SCTLa umjesto While petlje. Također je moguće cijeli algoritam implementirati na FPGA procesor te usporediti njegove rezultate sa implementacijom izvedenom u ovom radu no treba imati u vidu da to onemogućuje upotrebu SCTLa. Moguće je i vremena vođenja proračunati unaprijed te ih spremiti u Look-Up tablicu odakle će FPGA procesor samo po programiranom obrascu povlačiti vremena što bi kao rezultat trebalo imati drastično rasterećenje procesora.

Literatura

[1] B. K. Bose, Adjustable speed AC drives—A technology status review, Proceedings of the IEEE, vol. 70, no. 2, pp. 116-135, Feb. 1982.

[2] J. A. Houldsworth and D. A. Grant, The Use of Harmonic Distortion to Increase the Output Voltage of a Three-Phase PWM Inverter, in IEEE Transactions on Industry Applications, vol. IA-20, no. 5, pp. 1224-1228, Sept. 1984.

[3] A. Mehrizi-Sani, S. Filizadeh, Digital implementation and transient simulation of space-vector modulated converters, 2006 IEEE Power Engineering Society General Meeting, Montreal, Que., 2006, pp. 7 pp.-.

[4] D. W. Feng, B. Wu, S. Wei, D. Xu, Space vector modulation for neutral point clamped multilevel inverter with even order harmonic elimination, Canadian Conference on Electrical and Computer Engineering 2004 (IEEE Cat. No.04CH37513), 2004., pp. 1471-1475 Vol.3.

[5] J. Rodriguez, J. S. Lai, F. Z. Peng, Multilevel inverters: a survey of topologies, controls, and applications, IEEE Transactions on Industrial Electronics, vol. 49, no. 4, pp. 724-738, Aug 2002.

[6] I. Colak, E. Kabalci, R. Bayindir, Review of Multilevel Voltage Source Inverter Topologies and Control Schemes, Energy Conversion and Management, 2011.

[7] S. Vadhiraj, K. N. Swamy, B. P. Divakar, Generic SPWM technique for multilevel inverter, 2013 IEEE PES Asia-Pacific Power and Energy Engineering Conference (APPEEC), Kowloon, 2013, pp. 1-5.

[8] N. Celanovic, D. Boroyevich, A comprehensive study of neutral-point voltage balancing problem in three-level neutral-point-clamped voltage source PWM inverters, in IEEE Transactions on Power Electronics, vol. 15, no. 2, pp. 242-249, Mar 2000.

[9] A. Abdalrahman, A. Zekry, A. Alshazly, Simulation and Implementation of Grid-connected Inverters, International Journal of Computer Applications, vol. 60, December 2012.

[10] B. Tolunay, Space Vector Pulse Width Modulation for Three-Level Converters - a LabVIEW Implementation, Uppsala Universitet, Diplomski rad, 2012.

[11] A. Iqbal, A. Lamine, I. Ashraf, Mohibullah, Matlab/Simulink Model of Space Vector PWM for Three-Phase Voltage Source Inverter, Proceedings of the 41st International Universities Power Engineering Conference, Newcastle-upon-Tyne, 2006, pp. 1096-1100.

[12] A. M. Hava, R. J. Kerkman, T. A. Lipo, Carrier-based PWM-VSI overmodulation strategies: analysis, comparison, and design, IEEE Transactions on Power Electronics, vol. 13, no. 4, pp. 674-689, Jul 1998.

[13] K. Zhou, D. Wang, Relationship between space-vector modulation and three-phase carrier-based PWM: a comprehensive analysis, IEEE Transactions on Industrial Electronics, vol. 49, no. 1, pp. 186-196, Feb 2002.

[14] H. W. van der Broeck, H. C. Skudelny, G. V. Stanke, Analysis and realization of a pulsewidth modulator based on voltage space vectors, IEEE Transactions on Industry Applications, vol. 24, no. 1, pp. 142-150, Jan/Feb 1988.

[15] F. Yusivar, R. J. Sembiring, Implementation of Space Vector Pulse Width Modulation using CompactRIO, 2013 Joint International Conference on Rural Information & Communication Technology and Electric-Vehicle Technology (rICT & ICeV-T), Bandung, 2013, pp. 1-6.

[16] G. K. Nisha, S. Ushakumari, Z. V. Lakaparampil, Harmonic Elimination of Space Vector Modulated Three Phase Inverter, 2012 Proceedings of the International MultiConference of Engineers and Computer Scientists, Hong Kong, 2012., vol. 2

[17] A. Busse, and J. Holtz, Multiloop control of a unity power factor fast switching AC to DC converter, 1982 IEEE Power Electronics Specialists conference, Cambridge, MA, USA, 1982, pp. 171-179.

[18] J. Holtz, P. Lammert, and W. Lotzkat, High-Speed Drive System with Ultrasonic MOSFET PWM Inverter and Single-Chip Microprocessor Control, in IEEE Transactions on Industry Applications, vol. IA-23, no. 6, pp. 1010-1015, Nov. 1987.

[19] P. C. Krause, O. Wasynczuk, S. Sudhoff, S. Pekarek, Analysis of Electric Machinery and Drive Systems, Wiley-IEEE Press, 2013.

[20] R. H. Park, Two-Reaction Theory of Synchronous Machines, Transactions of the American Institute of Electrical Engineers, vol. 48, no. 3, pp. 716-727, July 1929.

[21] H. C. Stanley, An Analysis of the Induction Machines, Electrical Engineering, vol. 57, no. 12, pp. 751-757, Dec. 1938.

[22] G. Kron, Equivalent Circuits of Electric Machinery, John Wiley & Sons, 1951.

[23] D. S. Brereton, D. G. Lewis, C. C. Young, Representation of induction motor loads, Electrical Engineering, vol. 76, no. 5, pp. 393-393, May 1957.

[24] P. C. Krause, C. H. Thomas, Simulation of Symmetrical Induction Machinery, IEEE Transactions on Power Apparatus and Systems, vol. 84, no. 11, pp. 1038-1053, Nov. 1965.

[25] P. C. Krause, F. Nozari, T. L. Skvarenina, D. W. Olive, The Theory of Neglecting Stator Transients, IEEE Transactions on Power Apparatus and Systems, vol. PAS-98, no. 1, pp. 141-148, Jan. 1979.

[26] M. Uzair, Implementation of Space-Vector Modulation in a Three-Phase VSI-Type Grid-Connected Inverter, Tampere University of Technology, 2015.

[27] P. H. Tran, Matlab/Simulink Implementation and Analysis of Three Pulse-Width-Modulation (PWM) Techniques Boise State University, 2012.

[28] N. P. Quang, J. A. Dittrich, Vector Control of Three-Phase AC Machines, Springer-Verlag, Berlin, 2015.

[29] P. Vas, Sensorless Vector and Direct Torque Control, Oxford University Press, 1998.

[30] K. P. Kovacs, Symmetrische Komponenten in Wechselstrommaschinen, Springer Basel AG, 1962.

[31] R. Krishnan, Electric Motor Drives: Analysis, Modeling, and Control, Prentice Hall, 2001.

[32] P. Kundur, Power System Stability and Control, McGraw-Hill, 1994.

[33] N. Mohan, Advanced Electric Drives: Analysis, Control, and Modeling Using MATLAB/Simulink, Wiley, 2014.

[34] G. Erceg, Upravljanje elektromotornim pogonima, Zagreb, 2006.

[35] T. Benšić, Analiza dinamičkih stanja sustava vlastite potrošnje hidroelektrane, Diplomski rad, Osijek.

[36] S. N. Vukosavic, Electrical Machines, Springer, 2013.

[37] W. A. Lewis, A Basic Analysis of Synchronous Machines-Part I, in Transactions of the American Institute of Electrical Engineers. Part III: Power Apparatus and Systems, vol. 77, no. 3, pp. 436-453, April 1958.

[38] D. G. Holmes, T. A. Lipo, Pulse Width Modulation for Power Converters: Principles and Practice, Wiley-IEEE Press, 2003.

[39] I. Flegar, Elektronički energetski pretvarači, KIGEN, Zagreb, 2010.

[40] B. K. Bose, Power Electronics and Motor Drives: Advances and Trends, Elsevier, 2006.

[41] B. K. Bose, Modern Power Electronics and AC Drives, Prentice Hall PTR, 2002.

[42] B. Wu, M. Narimani, High Power Converters and AC Drives, Wiley-IEEE Press, 2017.

[43] M. H. Saied, M. Z. Mostafa, T.M. Abdel-Moneim, H. A. Yousel, On Three-Phase Six-Switches Voltage Source Inverter: a 150° Conduction Mode, 2006 IEEE International Symposium on Industrial Electronics, Montreal, Que., 2006, pp. 1504-1509.

[44] N. Soleimanpour, Advanced Modulation Techniques for Neutral-Point Clamped Three-Level Inverters in Automotive Applications, Diplomski rad, Concordia University, Montreal, Canada, 2014.

[45] Y. Jiao, F. C. Lee, S. Lu, Space Vector Modulation for Three-Level NPC Converter With Neutral Point Voltage Balance and Switching Loss Reduction, IEEE Transactions on Power Electronics, vol. 29, no. 10, pp. 5579-5591, Oct. 2014.

[46] H. Hu, W. Yao, Z. Lu, Design and Implementation of Three-Level Space Vector PWM IP Core for FPGAs, IEEE Transactions on Power Electronics, vol. 22, no. 6, pp. 2234-2244, Nov. 2007.

[47] M. A. A. Badran, A. M. Tahir, W. F. Faris, Digital Implementation of Space Vector Pulse Width Modulation Technique Using 8-bit Microcontroller, World Applied Science Journal 21, pp. 21-28, 2013.

[48] A. K. Rathore, J. Holtz, T. Boller, Synchronous Optimal Pulsewidth Modulation for Low-Switching-Frequency Control of Medium-Voltage Multilevel Inverters, IEEE Transactions on Industrial Electronics, vol. 57, no. 7, pp. 2374-2381, July 2010.

[49] K. I. Al Qaisi, N. Bowring, Investigation of the Effects of Sampling Frequency on the THD of 3-Phase Inverters Using Space Vector Modulation International Journal of Electronics and Communication Engineering, vol. 9, no. 2, pp. 215-219., 2015.

[50] J. Travis, J. Kring, LabVIEW for Everyone: Graphical Programming Made Easy and Fun, Prentice Hall, 2006.

[51] H. P. Halvorsen, Introduction to LabVIEW, University College of Southeast Norway, 2016.

[52] NI LabVIEW for CompactRIO Developer's Guide: Recommended LabVIEW Architectures and Development Practices for Control and Monitoring Applications

[53] National Instruments, "What is a Real-Time Operating System (RTOS)?", 2013., dostupno na: http://www.ni.com/white-paper/3938/en/ [01.06.218.]

[54] A. S. Tanenbaum, Modern Operating Systems, Prentice Hall of India, 2007.

[55] National Instruments, "FPGA Fundamentals", 2012., dostupno na: http://www.ni.com/white-paper/6983/en/. [01.06.2018]

[56] H. F. W. Sadrozinski, J. Wu, Applications of Field-Programmable Gate Arrays in Scientific Research, CRC Press, 2011.

[57] K. S. Unnikrishnan, S. Madhavan, Parallel Computation Using DSP Slices in FPGA

[58] LabVIEW In-Product Help

Popis slika

2.1 Nadomjesna shema trofaznog izmjenjivača u mosnom spoju 4
2.2 Nadomjesna shema za sklopna stanja: a) U_1, b) U_2 c) U_3 d) U_4 e) U_5 f) U_6 . . . 5
2.3 Slijed upravljačkih signala i: a) Fazni naponi, b) Linijski naponi 7
2.4 Prostorni vektor u trofaznom koordinatnom sustavu 10
2.5 Clarkova transformacija 10
2.6 Opća transformacija 13
2.7 Položaji prostornih vektora u $\alpha - \beta$ ravnini 14
2.8 a) Polarni (Lokus) dijagram, b) Vektori koji se ne mogu realizirati [28] 14
2.9 Određivanje maksimalne amplitude vektora napona 15
2.10 Određivanje referentnog vektora 15
2.11 Određivanje zabranjenih zona 16
2.12 Mogući prijelazi među sklopnim stanjima [34] 18
2.13 Simetrični obrazac sklapanja ventila za: a) Sektor 1, b) Sektor 2, c) Sektor 3, d) Sektor 4, e) Sektor 5, f) Sektor 6 19
2.14 Fazni naponi za sektor 1 20
2.15 Valni oblik faznog i linijskog napona 21
3.1 Six Step od 120° slijed upravljačkih signala i: a) Fazni naponi, b) Linijski naponi 23
3.2 Sinusna pulsno-širinska modulacija 24
3.3 Položaji prostornih vektora kod trorazinske prostorno-vektorske modulacije . . . 25
4.1 Blok dijagram algoritma prostorno-vektorske modulacije 26
4.2 Simulacijski blok dijagram prostorno-vektorske modulacije izrađen u Matlab/Simulinku 26
4.3 Clarkova transformacija izvedena u simulinku 27
4.4 Određivanje kuta i amplitude 27
4.5 Podsustav "Odabir sekstanta" 28
4.6 Proračun aktivnih vremena vektora u Simulinku 28
4.7 Podsustav generiranje impulsa 29
4.8 Trenutci uklopa i isklopa ventila za: a) Neparne sektore, b) Parne sektore 30
4.9 Switch Case za sektor 1 30
4.10 Izmjenjivač modeliran u Simulinku 31
4.11 Podsustav mjerenja 31
4.12 Generirani upravljački impulsi 32
4.13 a) Fazni naponi, b) Linijski naponi 32
4.14 Simetrični obrazac sklapanja ventila postignut u Simulinku pri $f_s = 10kHz$ za: a) Sektor 1, b) Sektor 2, c) Sektor 3, d) Sektor 4, e) Sektor 5, f) Sektor 6 34
4.15 Simetrični obrazac sklapanja ventila postignut u Simulinku pri $f_s = 5kHz$ za: a) Sektor 1, b) Sektor 2, c) Sektor 3, d) Sektor 4, e) Sektor 5, f) Sektor 6 35

4.16 Simetrični obrazac sklapanja ventila postignut u Simulinku pr $f_s = 1kHz$ za: a) Sektor 1, b) Sektor 2, c) Sektor 3, d) Sektor 4, e) Sektor 5, f) Sektor 6 36
4.17 Sklopna frekvencija 5 kHz: a) Fazni naponi, b) Linijski naponi 37
4.18 Sklopna frekvencija 1 kHz: a) Fazni naponi, b) Linijski naponi 37
5.1 Arhitektura cRio platforme 39
5.2 cRio-9024 embedded sustav 39
5.3 Grafičko objašnjenje Jitter-a 40
5.4 Dijelovi FPGA procesora [55] 41
5.5 a) Logička OR operacija u LabVIEWu, b) Simbol i tablica istine za OR funkciju 42
5.6 Interprocesorska komunikacija upotrebom kontrola i indikatora Front Panela . . 43
5.7 DMA FIFO način razmjene podataka između procesora [52] 44
5.8 Blok dijagram algoritma implementiranog na cRio-9024 platformi 44
5.9 Dio algoritma implementiran na Real-Time procesoru 45
5.10 Dio algoritma implementiran na FPGA procesoru 46
5.11 Blok: a) Open VI Reference, b) Close VI Reference 46
5.12 Blok Read/Write Control 47
5.13 Proračun proteklog vremena od trenutka pokretanje izvršenja algoritma 48
5.14 SubVI Clarkeova transformacija 48
5.15 Određivanje kuta i amplitude 49
5.16 SubVI za određivanje sektora 49
5.17 SubVI za proračun vremena vođenja 50
5.18 a) SGL tip podataka, b) FXP tip podataka 50
5.19 Time-Clock Calculation SubVI 50
5.20 Generiranje impulsa implementirano na FPGA procesoru SubVI 51
5.21 Front Panel RTOSa 53
5.22 Upravljački impulsi ventila S_1 i S_2 54
5.23 Simetrični obrazac sklapanja ventila postignut na cRiu pri $f_s = 10kHz$ za: a) Sektor 1, b) Sektor 2, c) Sektor 3, d) Sektor 4, e) Sektor 5, f) Sektor 6 55
5.24 Simetrični obrazac sklapanja ventila postignut na cRiu pri $f_s = 5kHz$ za: a) Sektor 1, b) Sektor 2, c) Sektor 3, d) Sektor 4, e) Sektor 5, f) Sektor 6 56
5.25 Simetrični obrazac sklapanja ventila postignut na cRiu pri $f_s = 1kHz$ za: a) Sektor 1, b) Sektor 2, c) Sektor 3, d) Sektor 4, e) Sektor 5, f) Sektor 6 57

Popis tablica

2.1 Sklopna stanja 4
2.2 Fazni i linijski naponi za pojedino sklopno stanje 5
2.3 Obrasci sklapanja po sektorima 20
2.4 Srednja vrijednost faznih napona za sve sektore 21
3.1 Sklopna stanja za Six-Step od 120° 22
4.1 Izračunata i realizirana vremena vođenja 33
5.1 Upotreba resursa FPGA procesora za implementirani algoritam prostorno-vektorske modulacije 52
5.2 Izračunata i realizirana vremena vođenja na cRio platformi 58
5.3 Omjeri iskazani u postotcima 59

Sažetak

Kroz diplomski rad je opisan matematički model modulacije prostornog vektora. Definiran je pojam prostornog vektora te su dani izrazi za proračun vremena vođenja. Predložen je optimalan obrazac sklapanja ventila te je na osnovu obrazaca sklapanja prikazan valni oblik faznog i linijskog napona uz objašnjenje prisustva harmonika trećeg reda. Dan je kratak pregled ostalih modulacijskih tehnika. Provedena je implementacija u Matlab/Simulinku te je obavljena analiza rezultata. Također je obavljena implementacija na cRio RTOS/FPGA platformi te je obavljena analiza dobivenih rezultata.

Ključne riječi: Prostorni vektor, modulacija prostornog vektora, simetrični obrazac sklapanja ventila, Matlab/Simulink, LabView, RTOS, FPGA, CompactRio platforma

Abstract

Implementation of the Space Vector Modulation on CRIO FPGA Platform

Through this thesis the mathematical model of the Space Vector Modulation is presented. The concept of the space vector is described and expressions for switching time calculation are given. An optimal switching pattern is suggested and phase to neutral and phase to phase voltage waveforms are presented with the explanation of triplen harmonics presence. A short overview of modulation techniques is presented. The Space Vector Modulation algorithm is implemented in Matlab/Simulink and the results are studied. The implementation on the cRio RTOS/FPGA platform is also presentedd and the results are studied.

Key Words: Space Vector, Space Vector Modulation, Symmetrical Switching Pattern, Matlab/Simulink, LabView, RTOS, FPGA, CompactRio Platform

Životopis

Marjan Ugljar rođen je 09.06.1994. godine u Vinkovcima. U Vrbanji završava „Osnovnu školu Mare Švel-Gamiršek“, nakon čega upisuje opći smjer Gimnazije Županja. Po završetku srednjoškolskog obrazovanja upisuje preddiplomski studij elektrotehnike na Elektrotehničkom fakultetu Osijek. Na 2. godini studija se opredjeljuje za smjer elektroenergetika. 2016. godine završava preddiplomski studij elektrotehnike te upisuje diplomski studij elektrotehnike na Fakultetu elektrotehnike, računarstva i informacijskih tehnologija Osijek. Opredjeljuje se za smjer elektroenergetika, izborni blok DEC - Industrijska elektroenergetika. Tijekom školovanja sudjelovao je na županijskim natjecanjima iz fizike, biologije i njemačkog jezika te na državnom natjecanju iz njemačkog jezika. Također je sudjelovao na natjecanjima iz nogometa, rukometa i stolnog tenisa.

U Osijeku, srpanj 2018. Marjan Ugljar

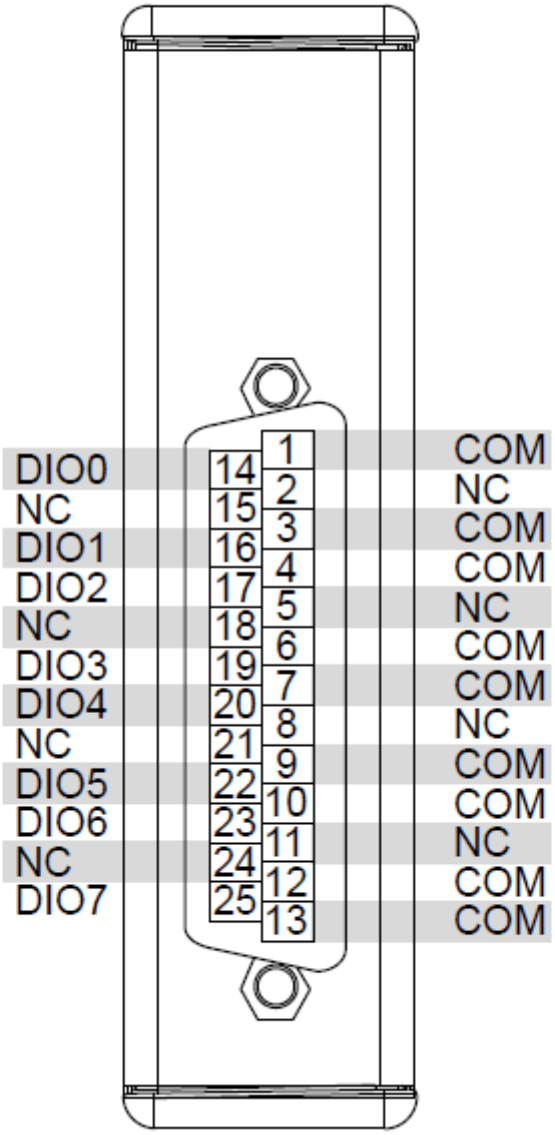
(Vlastoručni potpis)

PRILOG 1 - MATLAB SKRIPTA

```
close all; clear all; clc;

A=sqrt(2)*380;
f=50;
fs=1e3;
Ts=1/fs;
Tc=Ts/2;
Ud=660;
a=0;
Kut=0;

sim('SVMAlfaBeta4');

set(0,'DefaultAxesFontSize', 12);

figure(1);
hold on;
subplot(3,1,1);
plot(t,naponi(:,1));
title('Fazni naponi');
xlabel('Vrijeme [s]');
ylabel('Napon [V]');

subplot(3,1,2);
plot(t,naponi(:,2));
xlabel('Vrijeme [s]');
ylabel('Napon [V]');

subplot(3,1,3);
plot(t,naponi(:,3));
xlabel('Vrijeme [s]');
ylabel('Napon [V]');
hold off;


figure(2);
hold on;
subplot(3,1,1);
plot(t,naponi(:,4));
title('Linijski naponi');
xlabel('Vrijeme [s]');
ylabel('Napon [V]');

subplot(3,1,2);
plot(t,naponi(:,5));
xlabel('Vrijeme [s]');
ylabel('Napon [V]');

subplot(3,1,3);
plot(t,naponi(:,6));
xlabel('Vrijeme [s]');
ylabel('Napon [V]');

figure(3);
hold on;
subplot(6,1,1);
plot(t,S(:,1));
ylabel('S1');
xlim([0,0.005]);
ylim([-0.5,1.5]);
title('Control Pulses');

subplot(6,1,2);
plot(t,S(:,2));
ylabel('S2');
xlim([0,0.005]);
ylim([-0.5,1.5]);

subplot(6,1,3);
plot(t,S(:,3));
ylabel('S3');
xlim([0,0.005]);
ylim([-0.5,1.5]);

subplot(6,1,4);
plot(t,S(:,4));
```

```
77  ylabel('S4');
78  xlim([0,0.005]);
79  ylim([-0.5,1.5]);
80
81  subplot(6,1,5);
82  plot(t,S(:,5));
83  ylabel('S5');
84  xlim([0,0.005]);
85  ylim([-0.5,1.5]);
86
87  subplot(6,1,6);
88  plot(t,S(:,6));
89  ylabel('S6');
90  xlim([0,0.005]);
91  ylim([-0.5,1.5]);
92
93  a=1; %1 za rucno unosenje kuta
94  Kut = 0.9;
95  sim('SVMAlfaBeta4');
96
97  figure(6);
98  hold on;
99  subplot(3,1,1);
100 plot(t,S(:,1),'b');
101 title('Switching Pattern Sector 6');
102 ylabel('S1');
103 xlim([0,10e-4]);
104 ylim([-0.5,1.5]);
105
106 subplot(3,1,2);
107 plot(t,S(:,3),'b');
108 ylabel('S3');
109 xlim([0,10e-4])
110 ylim([-0.5,1.5]);
111
112 subplot(3,1,3);
113 plot(t,S(:,5),'b');
114 ylabel('S5');
115 xlim([0,10e-4])
116 ylim([-0.5,1.5]);
```

PRILOG 2 - RASPOREDIZLAZA NI 9401 MODULA



PRILOG 3 - ESTIMIRANE VRIJEDNOSTI UPOTREBE RESURSA KORIŁTENIH FUNKCIJA CRIO-9114 FPGA PROCESORA

Funkcija	**Tip podataka**	**Izvan SCTLa**			**Unutar SCTLa**		
		Flip Flops	**LUTs**	**Mult.**	**Flip Flops**	**LUTs**	**Mult.**
While Loop, i terminal wired	/	2	6	/	Nije podržano		
Case Structure	/	5	15	/	0	0	/
Feedback Node	Boolean	3	5	/	2	2	/
Add	SGL	Nepoznato					
Subtract	SGL	Nepoznato					
Multiply	SGL	Nepoznato					
Increment	I16/U16	17	18	0	0	16	0
Reciprocal	SGL	Nepoznato					
Numeric Constant	/	0	0	0	0	0	0
Not	Boolean	4	3	/	0	1	0
True Constant	/	0	0	0	0	0	0
False Constant	/	0	0	0	0	0	0
Greater?	I16/U16	2	20	/	0	18	/
Greater or Equal?	I16/U16	2	21	/	0	19	/
Less or Equal?	I16/U16	2	21	/	0	19	/
Loop Timer	/	72	43	/	Nije podržano		
Memory Block	SGL	Nepoznato					
Control	Boolean	2	5	/	Nepoznato		
	I16/U16	27	21	/	Nepoznato		
	SGL	Nepoznato					
Indicator	Boolean	2	5	/	Nepoznato		
	I32/U32	52	39	/	Nepoznato		
	SGL	Nepoznato					

PRILOG 4 - NEPODRŽANE FUNKCIJE U SINGLE CYCLE TIMED LOOPU

- Analog Period Measurement VI
- Butterworth Filter VI
- Discrete Delay VI
- Matematička operacija dijeljenja
- FIFO Clear funkcija
- FPGA I/O Method Node osim sa određenim FPGA
- FPGA I/O Property Node osim sa određenim FPGA
- Interrupt VI
- Look-Up-Table 1D VI sa označenim Interpolate Data
- Loop Timer VI
- Više FPGA I/O čvorova koji koriste isti I/O resurs ako je barem jedan unutar petlje i barem jedan van petlje
- Non-reentrant SubVIi ako ih se koristi više
- Notch Filter VI
- PID (FPGA) VI
- Quotient & Remainder funkcija
- Reciprocal funkcija
- Rotate 1D Array funkcija
- Sine Wave Generator VI
- Operacija sa SGL tipom podataka
- Square Root function
- Timed Loop
- Wait Express VI
- Wait on Occurrence funkcija
- While loop