



UNIVERZITET U NIŠU
ELEKTRONSKI FAKULTET



Dejan (Dragan) Mirković

**PROJEKTOVANJE SELEKTIVNIH IIR
DIGITALNIH FILTERA SA LINEARNOM
FAZOM UPOTREBOM ANALOGNIH
PROTOTIPOVA**

DOKTORSKA DISERTACIJA

Niš, 2017.



UNIVERSITY OF NIŠ
FACULTY OF ELECTRONIC ENGINEERING



Dejan (Dragan) Mirkovic

**DESIGN OF SELECTIVE IIR DIGITAL
FILTERS WITH LINEAR PHASE
UTILIZING ANALOG PROTOTYPES**

DOCTORAL DISSERTATION

Niš, 2017.

Podaci o doktorskoj disertaciji

Mentor:	Dr Miona Andrejević Stošović, docent
Naslov:	Projektovanje selektivnih IIR digitalnih filtara sa linearnom fazom upotrebom analognih prototipova
Rezime:	<p>Nepostojanje univerzalnog načina prevođenja analognog prototipa filtra u digitalni domen otvara mogućnost za dodatna poboljšanja postojećih i izvođenje novih metoda. U disertaciji će biti predložen i obrađen nov metod u vidu transformacije za preslikavanje kompleksne frekvencije s u z-domen. Digitalna IIR reprezentacija analognih filtara baziranih na specijalnoj klasi selektivnih filtara sa kritično monotonom amplitudskom karakteristikom će biti prezentovana po prvi put. Kompletan postupak projektovanja IIR filtara na bazi analognih prototipova iz ove klase će biti implementiran u vidu softverskog rešenja. Takođe, biće ispitan metod sa invarijantnom faznom karakteristikom u cilju projektovanja IIR filtara sa linearnom fazom. U tezi će biti razmatrana hardverska realizacija filtara. Praktični deo disertacije se odnosi na projektovanje IIR filtara u vidu integrisanog kola veoma visokog stepena integracije metodom standardnih ćelija. Reprezentativni primeri IIR filtara su realizovani do nivoa maksimi za fabrikaciju (lejaut). Na osnovu parazitnih elemenata kola ekstrahovanih iz lejauta date su procene potrošnje projektovanih kola.</p>
Naučna oblast:	Projektovanje integrisanih kola
Naučna disciplina:	Projektovanje digitalnih filtara sa beskonačnim impulsnim odzivom
Ključne reči:	digitalni filtri, integrisana kola
UDK:	(621.372.54:004.382)+621.3.049.77
CERIF klasifikacija:	T170 Elektronika
Tip licence Kreativne zajednice:	CC BY-NC-ND

Data on Doctoral Dissertation

Doctoral Supervisor:	Miona Andrejevic Stosovic
Title:	Design of selective IIR digital filters with linear phase utilizing analog prototypes
Abstract:	<p>Lack of universal approach for transforming analog filter prototypes into digital domain opens a possibility for further research in the field thus enabling both, improving existing and deriving novel methods. In the dissertation, a novel method in a form of transformation function for mapping complex angular frequency s into z domain will be covered. Digital counterparts of analog filters based on a special class of filters exhibiting critical monotonic pass-band amplitude characteristic will be presented for the first time. Complete procedure for designing IIR filters based on this class of filters will be implemented as a software solution. Phase-Invariance Method will be exploited for designing linear phase IIR filters. Hardware realization of the filters will be considered. The practical part of the thesis is devoted to IIR filter design in a form of (very large-scale integration) integrated circuit exploiting standard cells. Representative examples of IIR filter designs are realized up to the fabrication masks level (layout). Power consumption estimations based on layout parasitic elements are given as well.</p>
Scientific Field:	Integrated circuits design
Scientific Discipline:	Design of the infinite impulse response digital filters
Key Words:	digital filters, integrated circuits
UDC:	(621.372.54:004.382)+621.3.049.77
CERIF Classification:	T170 Electronics
Creative Commons License Type:	CC BY-NC-ND

Mojoj porodici, Urošu, Đorđu i Mileni

Teza je realizovana u okviru projekta „TR32004 Napredne tehnologije elektronskog merenja, upravljanja i komunikacije na električnoj distributivnoj mreži“ čiju realizaciju finansira Ministarstvo prosvete, nauke i tehnološkog razvoja Republike Srbije.

Zahvaljujem se svim kolegama iz LEDA laboratorije za projektovanje elektronskih kola na podršci i dugogodišnjoj saradnji.

Zahvaljujem se dr Srđanu Milenkoviću iz firme Lime Micorsystems na savetima i podršci vezanoj za praktični deo ove disertacije.

Posebno se zahvaljujem prof. dr Vanču Litovskom na svesrdnoj pomoći i brojnim savetima prilikom izrade, pisanja i realizacije ove disertacije.

Konačno, zahvaljujem se mentoru, doc. dr Mioni Andrejević Stošović, na saradnji i savetima prilikom realizacije i pisanja disertacije.

Sadržaj

1	Uvod	9
2	Projektovanje digitalnih FIR i IIR filtara	13
2.1	Izbor fizičke realizacije (tipa aproksimacione funkcije i arhitekture sistema)	15
2.2	Ispunjenje projektantskih zahteva u frekvencijskom i vremenskom domenu	19
2.3	Složenost rešenja odnosno cena proizvodnje	26
2.4	Primenljivosti u integrisanim kolima vrlo male potrošnje	26
2.5	Maksimalna brzina rada FIR i IIR filtara	27
3	Transformacije iz s u z domen za projektovanje IIR filtara na osnovu zahteva iz frekvencijskog domena	33
3.1	Upotreba bilinearne transformacije za sintezu IIR filtara	33
3.2	Nova transformacija drugog reda	36
3.3	CMAC filtri u z domenu	40
4	Uticaj konačne preciznosti predstavljanja koeficijenata filtra	47
4.1	Kvantovanje koeficijenata	51
5	VLSI implementacija digitalnih filtara	61
6	Primena transformacija sa specijalnom namenom	74
6.1	Metod invarijantne amplitudske karakteristike (MIM)	74
6.2	Metod invarijantne fazne karakteristike (PIM)	80
7	Projektovanje selektivnih IIR digitalnih filtara sa linearnom fazom upotrebom analognih prototipova	83
7.1	Sinteza korektora grupnog kašnjenja filtra sa selektivnom amplitudskom karakteristikom	83
7.2	Sinteza korektora amplitudske karakteristike polinomskih filtra sa linearnom faznom karakteristikom	87
7.3	Simultana sinteza amplitudske karakteristike i karakteristike konstantnog grupnog kašnjenja	94
8	Zaključak	97
9	Dodatak	100
9.1	MATLAB funkcije	100
9.2	Podrška projektovanju polinomskih CMAC filtara u z-domenu	107

9.3 VHDL kodovi	110
Lista pojmov	122
Literatura	124

1 Uvod

Obrada signala predstavlja jednu od najvažnijih operacija u prenosu podataka putem bilo kog elektronskog medijuma gde se kao ključni koncept nameće filtriranje. Pojam filtriranja se odnosi na frekvencijski domen, tj. filter treba da obezbedi oblikovanje frekvencijskog spektra ulaznog signala shodno datoj primeni. U idealnom slučaju filter treba simultano da potpuno potisne, sa beskonačnim slabljenjem van propusnog opsega, signale neželjenih frekvencija i očuva, sa nultim slabljenjem unutar propusnog opsega, signale poželjnih frekvencija. Pri tome poželjno je da propusni i nepropusni opseg budu jedan do drugog bez tzv. prelazne oblasti. Nažalost, ovakvu filterarsku funkciju nije moguće praktično realizovati. Prema tome, izučavanje elektronskih filtera je uvek bila i ostaće veoma atraktivna oblast istraživanja što proizilazi iz potrebe za novim konceptima i metodama u cilju približavanja idealnoj filterarskoj funkciji. Razvoj ove oblasti je u tesnoj vezi sa razvojem elektronike i telekomunikacija koje su doživele ekspanziju šezdesetih godina dvadesetog veka. Rezultati publikovani u radovima poput [OrTe68, SaU158] potvrđuju činjenicu da je projektovanje filtera od samog početka bilo u žiži interesovanja naučne i inženjerske zajednice. Razvojem računara koncept filtriranja je dobio širi smisao čime je omogućena diseminacija ovog koncepta u razne tradicionalno netehničke oblasti. Praktično, gde god se javlja potreba za obradom neke vrste podataka koji više ne moraju obavezno biti interpretirani u domenu elektrotehnike, koncept filtriranja se pojavljuje kao nezaobilazni alat. Dakle, pojam signal i podatak postaju sinonimi u savremenom konceptu filtriranja. Takođe, bitno je naglasiti da će u nastavku teksta termini kontinualni i diskretni često biti zamenjeni terminima analogni i digitalni kako bi se bliže pojasnio domen kojim se disertacija bavi.

Generalno, proces projektovanja filtera se može posmatrati kao optimizacioni problem gde se svaka od ciljanih osobina može okarakterisati funkcijom greške koju treba minimizovati. Dakle, podrazumevaju se znanja iz oblasti kompleksne analize, linearne algebre, numeričke analize, optimizacije i aproksimacije. Naravno, ovo predstavlja samo jedan skup problema koji se obično navodi kao problem sinteze filterarskih funkcija. Dakle, u ovoj ravni projektovanje filtera se posmatra kao čisto matematički problem koji podrazumeva sintezu racionalne, polinomske, funkcije koja aproksimira idealnu filterarsku funkciju. Tradicionalno, ovaj problem se razmatra u frekvencijskom tj. s domenu gde je s kompleksna kružna frekvencija. Ovom domenu odgovara vremenski domen gde se podrazumevaju signali kontinualni u vremenu. U proteklim decenijama ovde su postignuti značajni rezultati koji su

praktično utemeljili oblast izučavanja problema sinteze filtarskih funkcija. Za ovaj rad od ključnog značaja su [AnZL14, Rako70, RaLi73, RaDj72, RaLa72, RaPo80, RaPD77, ToLA15b]. Drugi deo problema se odnosi na fizičku realizaciju filtara ili jednostavno sintezu (implementaciju) filtara. Često se problem sinteze filtarske funkcije posmatra integralno sa implementacijom filtra. U teoriji kola dobro je poznato da ista prenosna funkcija filtra može imati različite realizacije. Prvo se može razmatrati izbor tehnologije za realizaciju. Na raspolaganju su mnoge tehnologije za implementaciju filtarske funkcije poput pasivnih/aktivnih elektronskih, elektro-mehaničkih, piezoelektričnih (kristalnih), optičkih kao i drugih. Sama implementacija filtra prvenstveno zavisi od prirode signala koji se obrađuju. Tako npr. ukoliko se radi o kontinualnim signalima najčešće se u praksi koriste pasivne (LC [LiMi83]) i/ili aktivne (RC, gmC [AID110, ToLA15b]) mreže.

Razvoj CMOS tehnologije i integrisanih kola veoma visokog stepana integracije (*Very Large Scale Integration* – VLSI) omogućio je efikasnu implementaciju koncepta filtriranja u diskretnom domenu. Praktično, došlo se do novog koncepta tzv. digitalne obrade signala (*Digital Signal Processing* – DSP). Ova nova paradigma je iznedrila dva tipa digitalnih filtara: filtri sa beskonačnim impulsnim odzivom (*Infinite Impulse Response* – IIR) i filtri sa konačnim impulsnim odzivom (*Finite Impulse Response* – FIR). U ovoj tezi fokus će biti na projektovanju IIR filtara i glavni doprinosi se odnose na ovaj tip filtara. Naravno, FIR filtriranje je veoma važan i neosporno dominantan koncept u savremenim DSP sistemima pa će shodno tome biti diskutovan na raznim nivoima kroz disertaciju.

Tradicionalno, projektovanje IIR filtra podrazumeva sledeće korake:

1. Sintezu normalizovane, racionalne, filtarske funkcije propusnika niskih frekvencija (NF, *Low Pass* – LP) tzv. NF prototip. Kao NF prototip ovde će biti razmatrane filtarske funkcije iz klase funkcija sa kritično/granično monotonom amplitudskom karakteristikom (*Critical Monotonic Amplitude Characteristic* – CMAC). Četiri osnovne klase su: Least Square Monotonic (LSM) [RaLi73], *Optimal-L* (na bazi Ležandrovih (*Legendre*) polinoma prvog reda) (L) [Fuka59], *Halpern* (H) [Halp69], Batervort (*Butterworth*, B) [Butt30] i detaljno su obrađene u [ToLA15b].
2. Transliranje prototipa po frekvencijskoj osi na željeni frekvencijski opseg sa realnim graničnim frekvencijama. Ovde se podrazumevaju četiri osnovna tipa: propusnik niskih frekvencija (NF/LP), propusnik visokih frekvencija (VF, *High Pass* – HP), propusnik opsega frekvencija (PO, *Band Pass* – BP) i nepropusnik (prigušnik) opsega frekvencija (NPO, *Band Stop* – BS).

3. Preslikavanja iz s u z domen.

4. Realizacija (implementacija) filtra (hardver/softver).

Dok se korak 4 može smatrati univerzalnim, koraci 1-3 nisu jedini način projektovanja IIR filtara. Naime, sinteza filtarskih funkcija se može sprovesti direktno u z domenu [RaKD91, TCHR01]. Naravno, ovakav način obično ne rezultuje rešenjem u zatvorenom obliku i podrazumeva upotrebu izuzetno kompleksnih, optimizacionih, algoritama. Ova činjenica je jedan od glavnih motiva za reviziju standardnog načina projektovanja IIR filtara polazeći od dobro definisanih prototipskih funkcija u analognom domenu. Rezultati vezani za korake 1 i 2 su detaljno prezentovani u navedenoj literaturi i biće tretirani kao osnova za građenje ove doktorske disertacije. Dakle, u disertaciji će glavni fokus biti na koracima 3 i 4.

Motiv za istraživanje 3. koraka se može objasniti na sledeći način. Do sada su razvijene brojne metode koje se baziraju na transformaciji (preslikavanju) filtarskih funkcija iz kontinualnog s , u diskretni z domen. U jednu grupu spadaju metode ili tehnike koje imaju za cilj očuvanje neke od karakteristika filtra prilikom prelaska u diskretni domen. Prema tome, zavisno od primene filtra može se izabrati neka od tehnika koja će očuvati bitnu karakteristiku za datu primenu. U ovoj grupi najzastupljenije su tehnike poput: invarijantni impulsni odziv (*Impulse Response Invariant*) prvi put uvedena od strane Kajzera (*Kaiser*) [Kais66] i detaljnije diskutovana u [HoLi12], modifikovani invarijantni impulsni odziv (*Modified Impulse Response Invariant*) [Nela07], invarijantni jedinični odziv (*Step Invariant*) [NéKo00], invarijantna amplitudska karakteristika (*Magnitude Invariance*) [Paar98] uvedene od strane Parmana (*Paarmman*) i invarijantna fazna karakteristika (*Phase Invariance*) [PaAt06] uvedene od strane Parmana i Artisa (*Artis*).

Druge metode se baziraju na smeni kompleksne frekvencije s adekvatnim izrazom po kompleksnoj promenljivoj, z . Na ovaj način se obezbeđuje direktno prevođenje racionalne filtarske funkcije iz s u z domen putem jednostavne smene promenljivih. Matematički, ovaj proces se može posmatrati kao vrsta kompornog preslikavanja. U ovu grupu spada upareni- z (*Matched- z*) metod [ShWZ12], tu su još i metode koje se baziraju na aproksimaciji analognog integratora digitalnim. Praktično, postupak integraljenja (ili diferenciranja) se diskretizuje u vremenu i aproksimira odgovarajućom numeričkom formulom. U ove metode spadaju Prednji/Zadnji Ojler (*Forward/Backward Euler*) i Trapezno pravilo tzv. bilinearna transformacija (*Bilinear/Trapezoidal/ Tustin*) [PaPK08].

Nijedna od navedenih metoda se ne može smatrati univerzalnom što ostavlja prostor za dalja istraživanja. U ovoj disertaciji dat je konkretan doprinos u smislu izvođenja nove transformacije koja se bazira na aproksimaciji analognog integratora. Takođe, biće ispitana i

bilinearna transformacija kao najpopularnija metoda iz ove grupe. Praktično, bilinearna transformacija će predstavljati referentnu tačku u odnosu na koju će rezultati biti vrednovani.

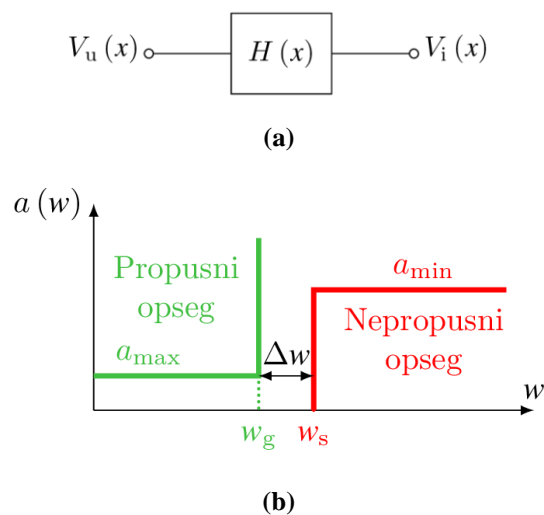
Istraživanje 4. koraka treba da da rešenja kako na arhitekturnom, tako i na tehnološkom nivou. U ovom delu fokus će biti na hardverskoj realizaciji, konkretno u integrisanim kolima po narudžbini (ASIC - *Application Specific Integrated Circuit*) na bazi standardnih ćelija. Razmatraće se aritmetika realnih brojeva u nepokretnom zarezu (*fixed-point*) kao generalno prihvaćen koncept u elektronskim DSP kolima. Kada se razmatra arhitekturni nivo obično se govori o formama za realizaciju IIR filtara. Generalno, forme se mogu podeliti na kanoničke (*canonical*), gde broj elemenata za kašnjenje odgovara redu filtra, i nekanoničke (*non-canonical*) gde je broj elemenata za kašnjenje veći od reda filtra. Jasno je da za hardversku realizaciju dolaze u obzir samo kanoničke forme. Neke od najzastupljenijih formi su: direktna (*Direct*), direktna transponovana I (*Transposed Direct - I*), direktna transponovana II (*Transposed Direct - II*) i *State Space*. Kada je u pitanju veći red filtra direktna implementacija filtarske funkcije je neefikasna pogotovu kada se radi o hardverskoj realizaciji filtra. Naime, opšte je poznato, a kasnije će biti i pokazano, da je direktna implementacija izuzetno osetljiva na konačnu preciznost predstavljanja koeficijenata. Prema tome pri hardverskoj realizaciji filtara višeg reda pribegava se serijskoj (kaskadna, *Cascade*, *Bi-quad coupled*) ili paralelnoj (kaskodna, *Cascade/Parallel*) implementaciji filtarske funkcije. Generalno, direktna implementacija je uglavnom pogodna za softversku realizaciju, (aritmetika pokretnog zareza, *floating-point*) dok se za hardversku realizaciju najčešće koristi serijska ili paralelna.

Posebna pažnja će biti posvećena uticaju konačne dužine digitalne reči na karakteristike filtra. Kod IIR filtara ovo je od izuzetnog značaja s obzirom na postojanje rekurzije izlaza. Konačna dužina reči je osnovni uzrok šuma usled zaokruživanja (*round-off noise*) i zasićenja kako izlaza tako i vrednosti unutrašnjih stanja (*overflow/overwrap/overload*). Drugi aspekt realizacije koji treba razmotriti jeste način implementacije osnovnih gradivnih blokova tj. množača, sabirača i linija za kašnjenje. Izbor arhitekture ovih gradivnih blokova u mnogome određuje maksimalnu brzinu rada i potrošnju kola. Konačno, za hardversku realizaciju filtara biće korišćen savremeni, nanometarski, CMOS tehnološki proces.

2 Projektovanje digitalnih FIR i IIR filtara

Istraživanje, koje će biti sprovedeno u disertaciji, ima za cilj da ukaže na važnost i perspektivu IIR pristupa obradi signala, pogotovu u VLSI sistemima sa niskom potrošnjom (LPVLSI – *Low Power Very Large Scale Integration*). Motivi za istraživanje IIR filtara se mogu jasnije identifikovati poređenjem FIR i IIR realizacije filtara. Poređenje će biti izloženo na osnovu nekoliko ključnih kriterijuma i to:

- Izbor fizičke realizacije (tipa aproksimacione funkcije i arhitekture sistema).
- Ispunjenje projektantskih zahteva u frekvencijskom i vremenskom domenu.
- Složenost rešenja odnosno cena proizvodnje.
- Primenljivosti u integrisanim kolima vrlo male potrošnje.
- Maksimalna brzina rada.



Slika 2.1 a) Filtar kao sistem i b) definicije veličina na karakteristici slabljenja

Pre nego pristupimo daljim izlaganjima, jasnoće radi, biće uvedene neke osnovne definicije koje se odnose na problematiku koja se razmatra.

Za sistem sa Sl. 2.1a definišemo prenosnu funkciju kao

$$H(x) = \frac{V_i(x)}{V_u(x)}, \quad (2.1a)$$

gde je x kompleksna analogna (s) ili digitalna (z) frekvencija. Za sistem sa Sl. 2.1a pretpostavlja se da je linearan i kauzalan što znači da će njegova prenosna funkcija da bude racionalna (količnik polinom po x) i da red polinoma brojioca nije veći od reda polinoma imenioca. Koeficijenti polinoma predstavljaju koeficijente filtra.

Amplitudsku karakteristiku definišemo kao

$$|H(\omega)| = \sqrt{[H(x) \cdot H(-x)]_{x=j\omega}}, \quad (2.1b)$$

gde je w analogna (ω) ili digitalna (θ) kružna frekvencija.

Fazna karakteristika definiše se kao

$$\phi(x) = \arg\{H(x)\}_{x=j\omega} = \frac{1}{2j} \ln \left[\left(\frac{H(x)}{H(-x)} \right)_{x=j\omega} \right]. \quad (2.1c)$$

Kao izvedene veličine obično se definišu slabljenje filtra:

$$a(\omega) = 20 \cdot \log(1/|H(\omega)|), \quad (2.1d)$$

i grupno kašnjenje filtra

$$t_d = -\frac{d\phi(x)}{dx}. \quad (2.1e)$$

Opseg na frekvencijskoj osi u kome filter ne unosi slabljenje naziva se propusni opseg, a van njega razlikujemo dva područja: nepropusni opseg i prelaznu oblast.

Kada se ima u vidu oblik amplitudske karakteristike definiše se više tipova. Naime, ako filter propušta opseg niskih frekvencija, a prigušuje ostatak frekvencijskog spektra signala, onda se naziva niskopropusni ili low-pass filter. U suprotnom, kada propušta samo visoke frekvencije naziva se propusnikom visokih frekvencija ili high-pass filter. Ako filter propušta jedan deo frekvencijskog spektra oko neke konačne frekvencije različite od nule, nazivamo ga propusnikom opsega frekvencija ili band-pass filter, a u suprotnom slučaju, ako prigušuje jedan opseg frekvencija oko neke konačne frekvencije različite od nule kažemo da se radi o prigušniku opsega ili band-reject (*notch*) filter. Najzad, ako filter propušta sve frekvencije nazivamo ga propusnikom svih frekvencija ili all-pass filter.

Na Sl. 2.1b, za slučaj niskopropusnog filtra, prikazani su zahtevi koji se obično postavljaju u pogledu amplitudske karakteristike. Ovde se jasno prepoznaju tri područja na karakteristici slabljenja: propusni opseg (gde slabljenje treba da bude što je moguće manje – u idealnom slučaju jednako nuli); nepropusni opseg (gde slabljenje treba da bude što je moguće veće – u idealnom slučaju beskonačno); i prelazna oblast (koja treba da bude što je moguće uža – u idealnom slučaju njena širina treba da bude jednaka nuli).

Na osnovu ovog dijagrama definišu se tri značajne veličine koje se koriste za karakterizaciju filtera. Najpre, definiše se granična frekvencija (w_g) kao vrednost promenljive

w za koju se na kraju (ili krajevima) propusnog opsega postiže zadato maksimalno slabljenje (na Sl. 2.1b obeleženo sa a_{\max}). Obično se uzima da je granična frekvencija ona na kojoj slabljenje dostiže 3 dB što bi značilo da se potrošaču koji opterećuje filter predaje polovina snage generatora koji pobuđuje isti.

Zatim se definiše selektivnost kao recipročna vrednost količnika širine prelazne oblasti ($\Delta\omega$) i širine propusnog opsega. Za filter propusnik niskih frekvencija sa Sl. 2.1b ova vrednost bi bila definisana kao

$$\frac{1}{S} = \frac{\Delta\omega}{\omega_g} = \frac{\omega_s - \omega_g}{\omega_g}. \quad (2.2)$$

Selektivnost govori o tome koliko blizu može da se smesti susedni kanal u sistemima sa frekvencijskim multipleksom i zato je od izuzetnog značaja. Što je uža prelazna oblast, to je veća selektivnost.

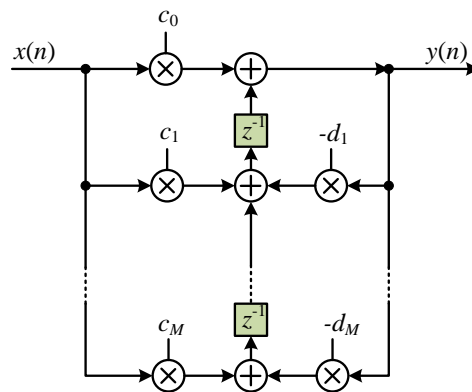
Najzad, definiše se slabljenje u nepropusnom opsegu kao minimalno dozvoljena vrednost slabljenja na koju se potiskuje spektar signala kako se ne bi pojavljivao u susednim kanalima, odnosno kako ne bi došlo do preslušavanja (aliasing). Na Sl. 2.1b ova vrednost obeležena je sa a_{\min} . Ukoliko je a_{\min} veće susedni kanali će biti više zaštićeni od preslušavanja. Frekvencija na kojoj karakteristika slabljenja po prvi put (za niskopropusni filter) dostiže a_{\min} , obeležena je sa w_s [StMa03].

Od izuzetne je važnosti ne mešati termine selektivnost i slabljenje u nepropusnom opsegu. Naime, to su dva posebna zahteva koji govore o različitim osobinama karakteristike slabljenja. Jedan filter može biti veoma selektivan, a da ispoljava malo slabljenje u nepropusnom opsegu i obrnuto. Filter koji ispoljava veoma veliko slabljenje u nepropusnom opsegu, a pri tome ima široku prelaznu oblast, nije selektivan. Treba imati na umu da neke klase filtera, tzv. polinomski niskopropusni filteri, na primer, imaju (baš svi) veoma veliko slabljenje na veoma visokim frekvencijama ali nisu svi selektivni.

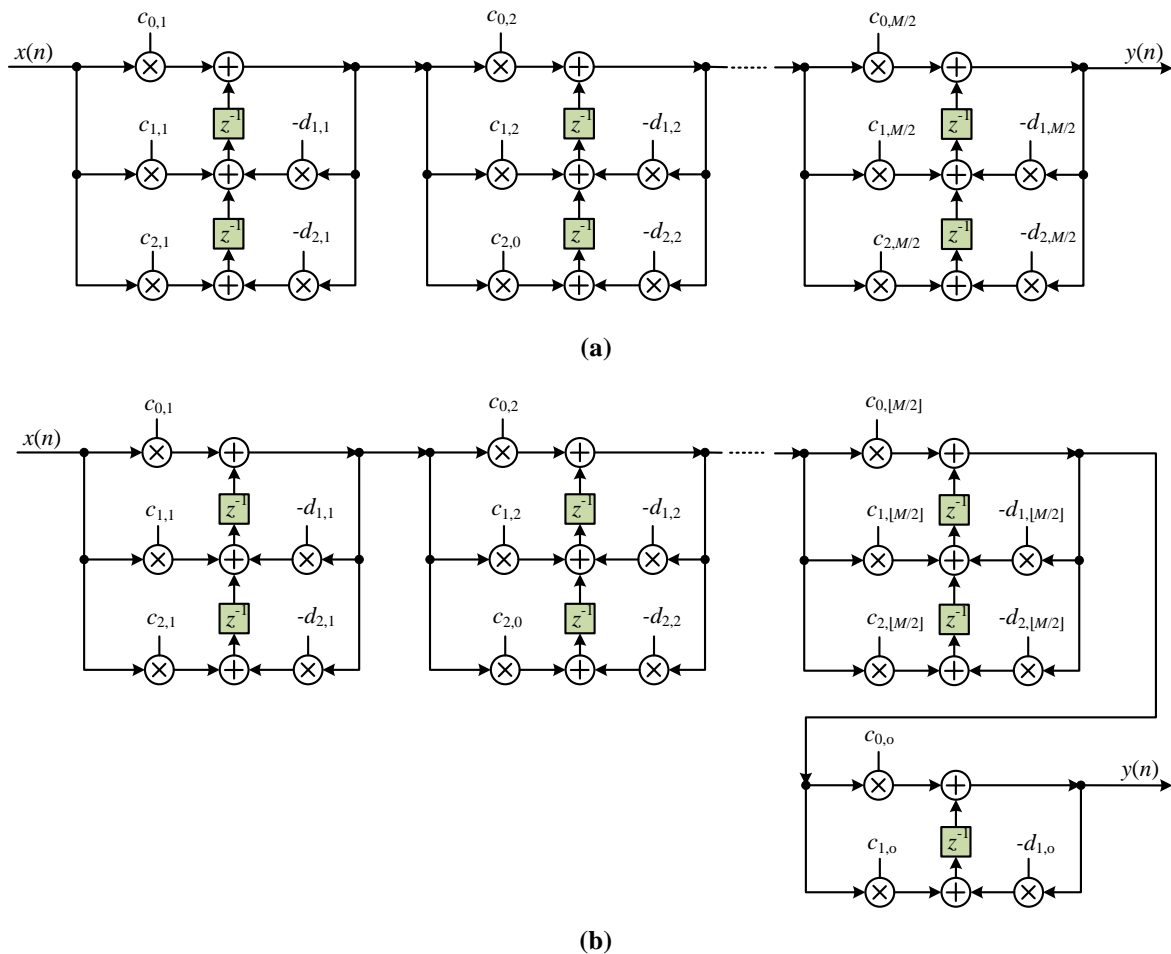
2.1 Izbor fizičke realizacije (tipa aproksimacione funkcije i arhitekture sistema)

Kada se govori o postupku projektovanja digitalnih filtera, projektant prvenstveno treba da izabere IIR ili FIR koncept. Nakon izbora sledi sinteza filterske funkcije (problem aproksimacije), a zatim i izbor arhitekture/topologije. Činjenica je da se ista prenosna funkcija može realizovati različitim arhitekturnim rešenjima kao što je ilustrovano na Sl. 2.2, 2.3 i 2.4. Sve tri prikazane forme su kanoničke pri čemu se podrazumeva da je frekvencija uzorkovanja

suviše visoka pa se ne mogu koristiti arhitekture poput MAC (*Multiplay and Accumulate*) [Mari08]. Na Sl. 2.2 je prikazana direktna forma filtra. Ova forma direktno implementira (2.1a) za $x = z$. Prednost ovakve realizacije jeste to što ne zahteva nikakvu dalju manipulaciju na arhitekturnom nivou nakon određivanja koeficijenata filtra. Poznato je da je direktna realizacija filtra izuzetno osetljiva na konačnu preciznost predstavljanja koeficijenata, pa se obično pribegava kaskadnoj realizaciji [Litw00].



Slika 2.2 Direktna realizacija IIR filtra reda M .



Slika 2.3 Serijska (kaskadna) realizacija IIR filtra reda M , (a) M parno i (b) M neparno.

Kaskadna realizacija (Sl. 2.3) se postiže faktorisanjem imenioca i brojioca prenosne funkcije i kombinovanjem konjugovano kompleksnih parova nula/polova što rezultuje proizvodom prenosnih funkcija drugog (bikvadratnih sekcija) i/ili prvog reda.

Ovakav pristup, često korišćen u praksi, gotovo uvek prati *overflow* efekat čak i kada je red filtra nizak. Tako npr. ako izlaz (ili interni čvor) neke od prethodnih sekcija u kaskadnoj vezi ima tendenciju da osciluje ili je u zasićenju, verovatno je da će se taj efekat propagirati u narednim sekcijama, pa i do samog izlaza filtra. Ovo može ugroziti stabilnost pa je potrebno dodatno skalirati koeficijente i/ili ulaz. Neke od metoda su date u [Parh07a, PrSa89]. Proračun faktora za skaliranje podrazumeva:

1. Dodatnu analizu arhitekture filtra u cilju identifikacije čvorova u kojima potencijalno može doći do *overflow*, *limit cycle* i sličnih neželjenih efekata. Za veći red filtra analiza filtra je nepraktična tako da se obično pribegava računarskoj simulaciji u cilju određivanja mesta gde se neželjeni efekti javljaju.
2. Proračun impulsnog odziva do kritičnih čvorova.
3. Izbor načina proračuna faktora za skaliranje po pojedinim sekcijama. Ovaj faktor se obično određuje normiranjem impulsnog odziva u kritičnim tačkama filtra. U ovu svrhu se najčešće koristi Čebiševljeva (*Chebyshev*) norma ili neke od tzv. *l* (Lebeg, *Lebesgue*) normi poput l_1 i l_2 [Rudi64]. Definicije ovih normi za diskretni sistem čiji je impulsni odziv, $h(n)$, a frekvencijski $H(\theta)$ su date u Tabeli 2.1.
4. Izbor nekog od kriterijuma za skaliranje koeficijenata sekcija (*Fixed Scaling*, *Split Fixed Scaling*, *Adaptive Scaling* [Aboa99]).

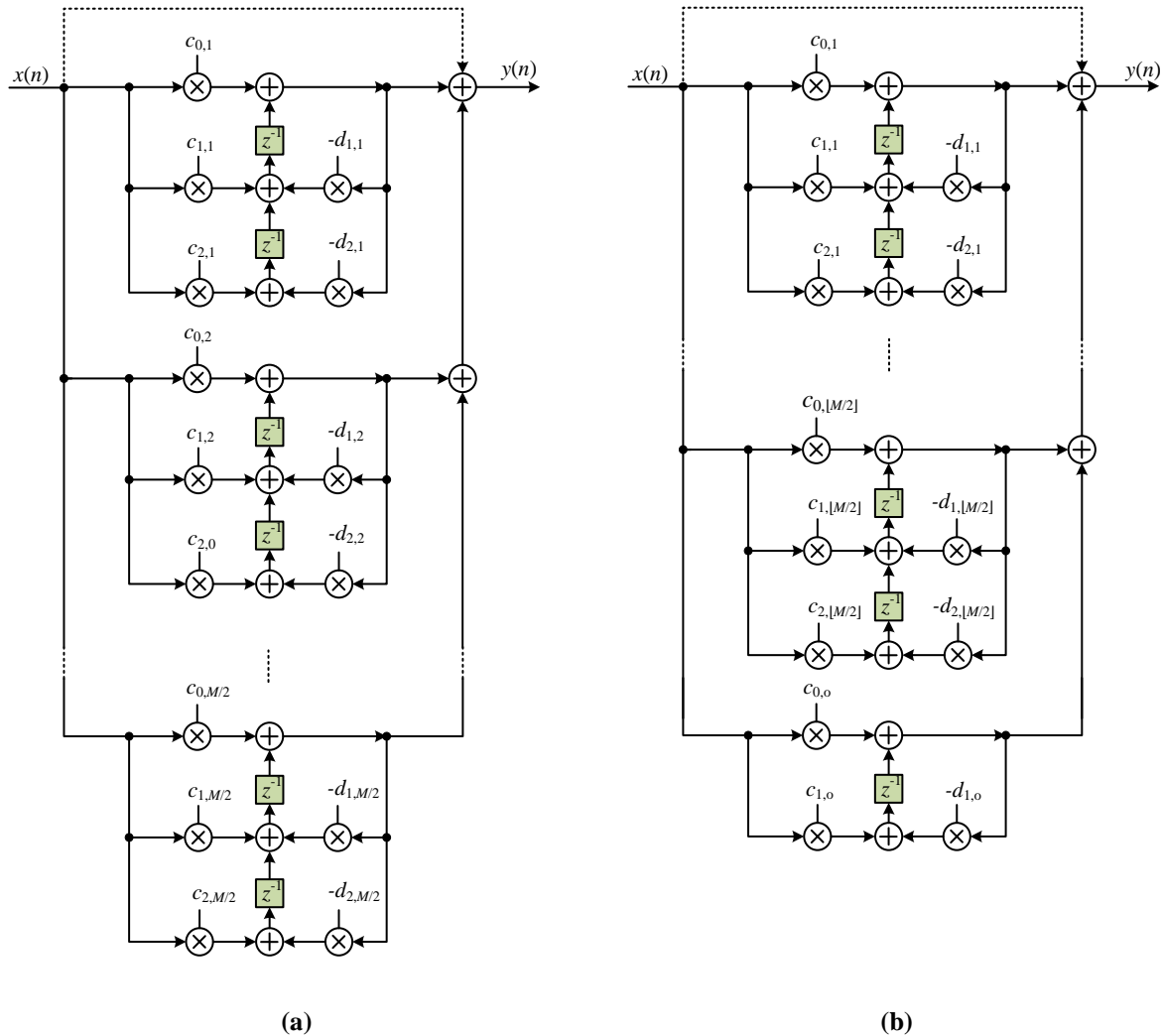
U [Aboa99] je demonstrirana primena ove procedure na eliptičkom filtru četvrtog reda realizovanog pomoću dve sekcije drugog reda. Treba napomenuti da je od značaja i sam redosled sekcija. Tako, na primer, često se bira da sekcije čiji su polovi bliži jediničnom krugu (one sa najvećim Q-faktorom) budu bliže izlazu filtra [Vaid93].

Tabela 2.1 Definicije najčešće korišćenih normi

Tip norme	Definicija
<i>Chebyshev</i>	$\ H\ _C = \max_{\theta} H(\theta) $
l_1	$\ h\ _1 = \sum_n h(n) $
l_2	$\ h\ _2 = \sqrt{\sum_n h(n) ^2}$

Evidentno je dakle da kaskadna realizacija zahteva dodatne korake u postupku projektovanja IIR filtra. Međutim, ako se upotrebi paralelna realizacija (Sl. 2.4), potreba za

skaliranjem koeficijenata se značajno redukuje. U ovakvoj realizaciji obično je dovoljno skalirati ulaz (koji je ujedno ulaz za sve sekcije filtra) fiksnim koeficijentom.



Slika 2.4 Paralelna (kaskodna) realizacija IIR filtera reda M , (a) M parno i (b) M neparno.

Da bi se došlo da paralelne realizacije potrebno je funkciju razviti na parcijalne sabirke odnosno odrediti ostatke u polovima prenosne funkcije. Dakle, svaki par konjugovano kompleksnih polova rezultuje odgovarajućim parom konjugovano kompleksnih ostataka. Praktično, dobijaju se imaginarne, racionalne, funkcije prvog reda gde se u imeniocu nalazi polinom prvog reda po s čiji je slobodni član pol, a u imeniocu koeficijent koji odgovara ostatku u polu imenioca. Da bi se dobile sekcija drugog reda sa realnim koeficijentima treba udružiti (sabirati) parcijalne imaginarne, racionalne, funkcije koje odgovaraju paru konjugovano kompleksnih polova/ostataka. Problem nalaženja ostataka u polovima se smatra “loše definisanim“ pogotovu kada se radi o prenosnoj funkciji sa višestrukim polovima (koje

su izuzetno retke) pa se uglavnom izbegava [OpSc75]. Pristup koji je izabran u ovom radu jeste da se ostaci u polovima pronađu u s -domenu gde je problem bolje definisan (gotovo nikada se ne javljaju višestruki polovi), a zatim odgovarajućom transformacijom, dobijene sekcije preslikaju u z -domen. Za realizaciju sekcija drugog reda upotrebljena je transponovana direktna forma II (TDFII) [Anto79]. Ukoliko prenosna funkcija filtra ima isti red polinoma u imeniocu i brojiocu (HP i BS filter) pre traženja ostataka u polovima potrebno je obezbediti da polinom u imeniocu bude višeg reda od polinoma brojioca. To se postiže deljenjem polinoma pri čemu nastaje konstantni, slobodni član, nezavisan od frekvencije, koji daje direktnu vezu između ulaza i izlaza što je ilustrovano isprekidanom linijom na Sl. 2.4 .

2.2 Ispunjenje projektantskih zahteva u frekvencijskom i vremenskom domenu

Filtarska funkcija iznad svega treba da ispuni jedan osnovni cilj. Da filtrira ulazni signal. To znači, treba da se “prijateljski” odnosi prema jednom delu spektra ulaznog signala koji na frekvencijskoj osi definiše propusni opseg filtra i da se istovremeno “neprijateljski” odnosi prema ostatku spektra ulaznog signala za koji se pretpostavlja da predstavlja šum koji će ometati rad susednih kanala i zato treba biti eliminisan. Ograničavanje propusnog opsega sistema (lanca obrade signala) samo po sebi minimizuje šum s obzirom na beli odnosno termički šum koji dolazi sa ulaza i koji se dodatno generiše unutar sistema. Prema tome, nesumnjivo je da je amplitudska karakteristika od primarnog značaja za kvalifikaciju sistema. Sistem sa neravnomernim ili velikim slabljenjem u propusnom opsegu i sa malim slabljenjem van njega, nije dobar filter. Tome treba dodati potrebu za oštrim razgraničenjem propusnog i nepropusnog opsega što se postiže sužavanjem intervala između njih odnosno sužavanjem prelazne oblasti. Filter sa uskom prelaznom oblašću je selektivan filter, a filter koji nije selektivan pre će biti namenjen nekoj drugoj funkciji (linija za kašnjenje, na primer), a ne filtriranju.

Kada se prenose impulsi, dakle kada se problem posmatra u vremenskom domenu, važna je verna reprodukcija istih. Da bi se to postiglo potrebno je da sistem koji prenosi impulse ima linearnu faznu karakteristiku ili što je ekvivalentno, konstantno grupno kašnjenje.

Projektovanje filtera koji ispunjavaju istovremeno zahteve za velikom selektivnošću i linearnom fazom je zadatak značajne težine. Naime, kod minimalno-faznih kola (ona koja u s -domenu imaju nule prenosne funkcije u levoj poluravni ili na imaginarnoj osi), amplitudska i fazna karakteristika su međusobno povezane i ispunjavanjem zahteva po jednom kriterijumu

(amplitude ili faze) dovodi do narušavanja zahteva po drugom. Zato kada se sintetizuju prenosne funkcije filtara sa linearnom fazom možemo da govorimo o trima situacijama:

1. Sinteza prenosne funkcije na osnovu zahteva za linearnom fazom,
2. Sinteza prenosnih funkcija na osnovu zahteva za velikom selektivnošću i
3. Sinteza prenosnih funkcija koje simultano zadovoljavaju oba zahteva.

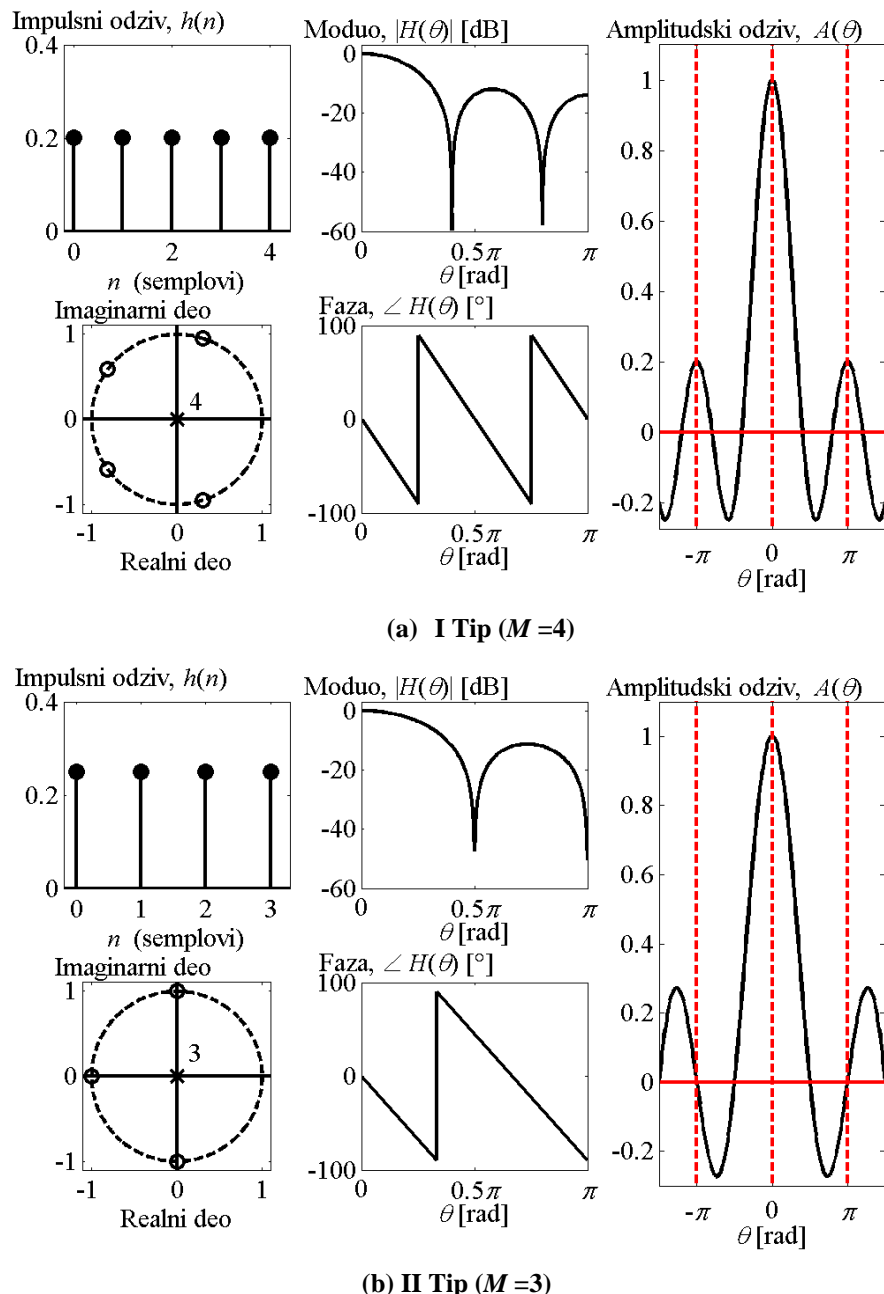
Ukoliko se obezbedi simetrija koeficijenata u prenosnoj funkciji FIR filtara nastaje sistem sa linearnom fazom (posebno bitno kod npr. obrade slike). To se može bolje videti u Tabeli 2.2 gde je dat amplitudski, $A(\theta)$, i fazni, $\phi(\theta)$, odziv za četiri osnovna tipa FIR filtara sa linearnom fazom [WTAFO6]. Frekvencijski odziv je definisan kao $H(\theta)=A(\theta)e^{j\phi(\theta)}$. Bitno je napomenuti da u opštem slučaju $A(\theta)$ nije uvek isto što i $|H(\theta)|$ kao što ni $\phi(\theta)$ nije uvek isto što i $\arg\{H(\theta)\}$. Kasnije će biti pokazano da se analizom $A(\theta)$ mogu otkriti neke bitne osobine FIR filtra sa linearnom fazom. Tip filtra je dat u prvoj koloni Tabele 2.2. Druga kolona prikazuje uslov vezan za red filtra, M . Za FIR filter obično se definiše i dužina filtra koja odgovara broju koeficijenata filtra, c_k tj. $M + 1$. I i III tip FIR filtra podrazumeva paran red (neparnu dužinu), a II i IV neparan red (parnu dužinu). Uslovi vezani za simetriju koeficijenata filtra su dati u trećoj koloni. Dakle, prva dva reda tabele prikazuju karakteristike FIR filtara sa linearnom fazom za koje važi simetrija koeficijenata (I i II tip). Slično, kod III i IV tipa filtra koeficijenati su asimetrični (poslednja dva reda tabele). Bitno je napomenuti da kod FIR filtara koeficijenti ujedno predstavljaju i impulsni odziv filtra, $h(n)$.

Tabela 2.2 Karakteristike četiri tipa FIR filtara sa linearnom fazom

Tip	M	c_k	$A(\theta)$	$\phi(\theta)$
I	parno	$c_k = c_{M-k}$,	$c_{M/2} + \sum_{k=0}^{M/2-1} 2c_k \cos\left(\left(\frac{M}{2}-1+k\right)\theta\right)$	$-\theta\frac{M}{2}$
II	neparno		$\sum_{k=0}^{(M-1)/2} 2c_k \cos\left(\left(\frac{M}{2}-k\right)\theta\right)$	$-\theta\frac{M}{2}$
III	parno	$c_k = -c_{M-k}$,	$c_{M/2} + j \sum_{k=0}^{M/2-1} 2c_k \sin\left(\left(\frac{M}{2}-1+k\right)\theta\right)$	$-\theta\frac{M}{2} + \frac{\pi}{2}$ za $c_{M/2} = 0$
IV	neparano		$\sum_{k=0}^{(M-1)/2} 2c_k \sin\left(\left(\frac{M}{2}-k\right)\theta\right)$	$-\theta\frac{M}{2} + \frac{\pi}{2}$

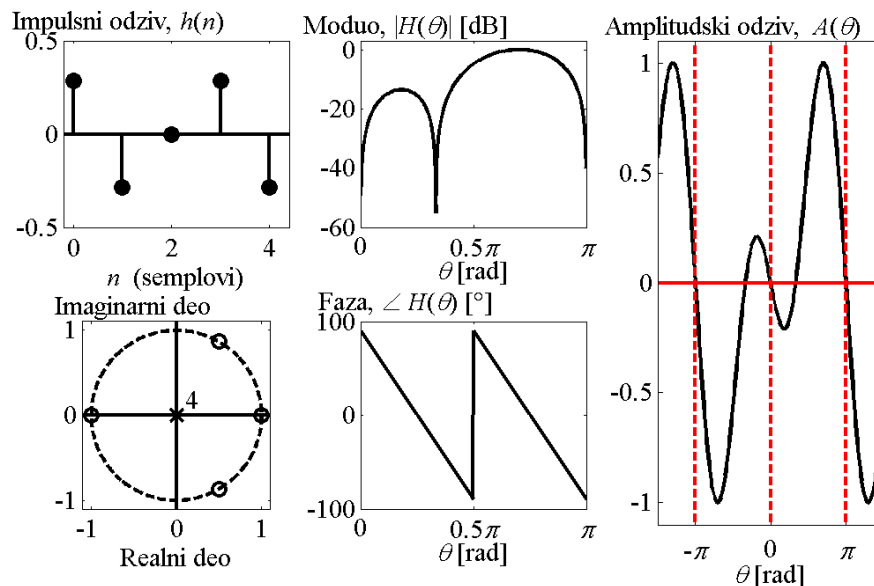
Amplitudski i fazni odziv filtara izvedeni pod navedenim uslovima su dati u poslednje dve kolone, respektivno. Na osnovu faznog odziva jasno se vidi da je faza linearna funkcija digitalne kružne frekvencije, θ . Treba uočiti da kod FIR filtra III tipa centralni koeficijent,

$c_{M/2}$, treba da bude jednak nuli kako bi se ostvarila linearna faza. Na osnovu (2.1e) i poslednje kolone očigledno je da je grupno kašnjenje konstantno ($\tau_d = M/2$) i da zavisi samo od reda filtra. Dakle, što veće M veća i latencija filtra (vremenski interval od zadavanja vrednosti ulaza do pojave prve validne vrednosti izlaza). Osobine četiri osnovna tipa FIR filtara sa linearnom fazom su ilustrovane na Sl. 2.5 i 2.6. Na osnovu ovih slika mogu se izvući veoma bitni zaključci vezani za primenu FIR filtra sa linearnom fazom.

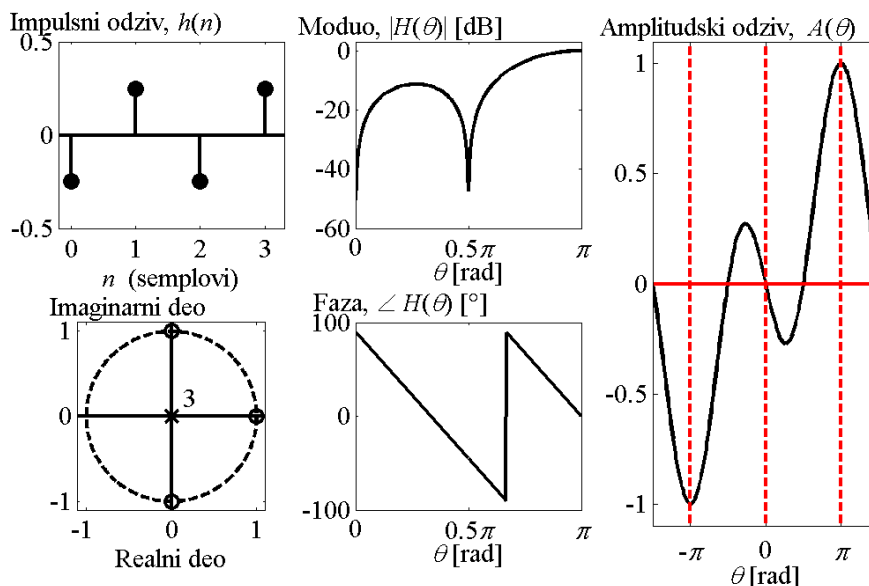


Slika 2.5 Ilustrativni prikaz osobina FIR filtara sa linearnom fazom, I (a) i II (b) tipa. Dat je primer klasičnog Sinc (*Moving Average*) filtra čiji su koeficijetni skalirani sa $1/(M+1)$ kako bi se ostvarilo jedinično pojačanje na $\theta=0$ rad. Nule su označene sa \circ , a polovi sa \times .

Za svaki tip filtra dati su: impulsni odziv, položaj nula/polova u kompleksnoj z ravni, moduo i faza prenosne karakteristike i amplitudski odziv. Posmatranjem Sl. 2.5a (I Tip) može se uočiti da amplitudski odziv ima nenultu vrednost za $\theta = \{0, \pm\pi\}$. Ove kritične tačke odgovaraju realnim frekvencijama $f = \{0, \pm f_N\}$, gde je sa $f_N = f_s/2$ označena Nikvistova (Nyquist) frekvencija, a sa $f_s = 1/T$ frekvencija uzorkovanja gde je T perioda uzorkovanja.



(a) III Tip ($M=4, c_{M/2}=0$)



(b) IV Tip ($M=3$)

Slika 2.6 Ilustrativni prikaz osobina FIR filtera sa linearnom fazom, III (a) i IV (b) tipa. Dati su arbitrarni primeri. Kod III tipa koeficijenti su skalirani tako da se ostvari jedinično pojačanje na $\theta=3\pi/2$. Za IV tip koeficijenti su skalirani sa $\sum |c_k|$ pa je jedinično pojačanje postignuto na $\theta=\pi$ rad. Nule su označene sa \circ , a polovi sa \times .

Pošto, amplitudski odziv ima vrednost različitu od nule u ovim graničnim tačkama FIR filter I tipa se može upotrebiti za realizaciju sve četiri varijante filtra tj. LP, HP, BP i BS. To

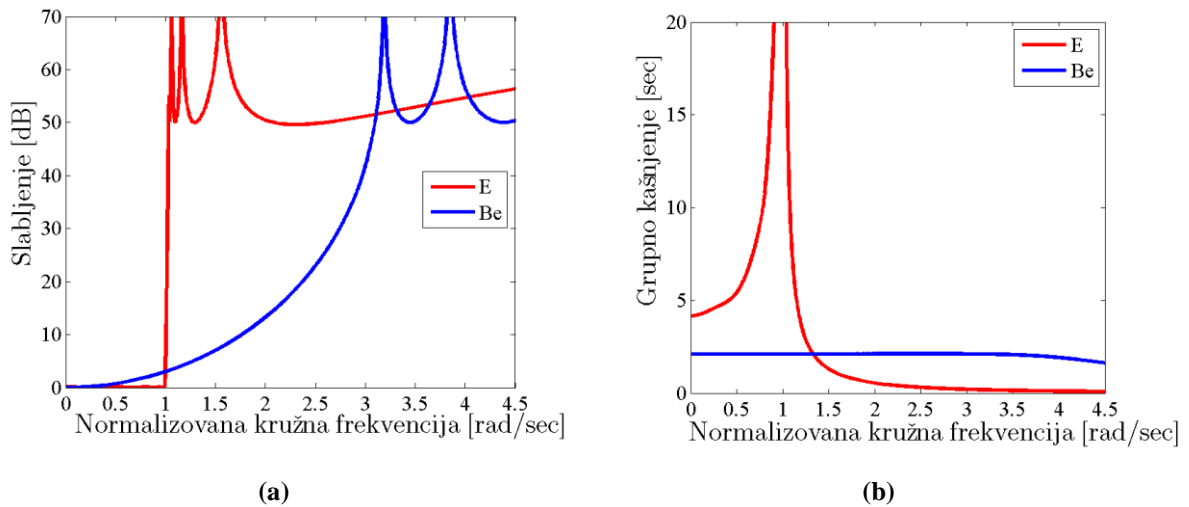
je zapravo i razlog zašto se baš ovaj tip filtra najčešće eksploatiše. Ovo nije slučaj kod preostala tri tipa. Naime, II tip (Sl. 2.5 b) bi bio više pogodan za realizaciju LP i BS, a ne za realizaciju HP i BP filtra. Za HP filter sa graničnom frekvencijom u blizini f_N ovaj tip filtra bi bio pratklično neupotrebljiv pošto amplitudska karakteristika konvergira ka nuli za $\theta \rightarrow \pm\pi$ (amplitudski odziv neparna funkcija u okolini $\theta = \pm\pi$, Sl. 2.5b).

Kod III tipa (Sl. 2.6a) amplitudski odziv ima nultu vrednost za $\theta = \{0, \pm\pi\}$ tako da je pogodan samo za BP ili BS filtre. Slično važi i za IV tip, gde se nulta vrednost amplitudske karakteristike postiže za $\theta \rightarrow 0$ pa je ovim tipom moguće realizovati HP ili BP filter. Međutim, iako ograničeni, III i IV tip se uspešno koriste za realizaciju kola diferencijatora i/ili Hilbertovog (*Hilbert*) transformatora pošto za $\theta = \{0, \pm\pi\}$ obezbeđuju fazni pomak od tačno $\pi/2$ (90°) između ulaza i izlaza (fazne karakteristike za $\theta=0$ rad na Sl. 2.6) [The05]. Na osnovu ove analize jasno je da linearnost faze uzrokuje određena ograničenja sa stanovišta ispunjenja projektantskih zahteva u frekvencijskom i vremenskom domenu.

Izrazi za $A(\theta)$ (četvrta kolona Tabele 2.2) govore o mogućoj selektivnosti FIR filtera. Naime, ako se digitalna frekvencija θ zameni sa vremenom t , prikazani izrazi za amplitudsku karakteristiku mogu da budu posmatrani kao delovi Furijeovog (*Fourier*) reda po sinusima ili po kosinusima. Ako je funkcija koja se aproksimira četvrtasti impuls (što odgovara idealnoj amplitudskoj karakteristici) Furijeov red zahteva beskonačno mnogo sabiraka čak i kada se dozvoli da suma (odziv) menja znak što kod aproksimacije idealne amplitudske karakteristike nikako ne može biti slučaj [Tols97]. Brojevni primer koji ilustruje ovu tvrdnju biće dat kasnije u ovom tekstu. Visoki red FIR filtera koji bi zadovoljavao zahteve u pogledu velike selektivnosti postaje prepreka njegovoj primeni u hardverskom obliku odnosno kao podsistema u telekomunikacionom integrisanom kola, što će takođe biti ilustrovano kasnije.

Imajući u vidu da se IIR filterarske funkcije mogu dobiti pogodnim preslikavanjem iz analognih prototipova i imajući u vidu da se u s -domenu mogu sintetizovati ekstremno selektivne amplitudske karakteristike sa relativno niskim redovima filtera [Zver05] (Primer jedne takve funkcije dat je na Sl. 2.7a, obeležen sa **E**. Širina prelazne oblasti ovde je 10% od širine propusnog opsega tj. $S=10$) može se zaključiti da su IIR filteri očigledno rešenje u slučajevima kada se traži velika selektivnost uz jednostavnu šemu kao i u slučajevima kada linearnost faze nije kritični zahtev. Pri ovome treba imati posebno u vidu primene u telekomunikacijskim sistemima sa frekvencijskim multipleksom gde se na frekvencijskoj osi kanali smeštaju jedan do drugog i gde je od izuzetnog značaja da budu što bliže jedan drugom, a istovremeno sprečiti preslušavanje između kanala. Ne treba zaboraviti da je

frekvencijski spektar prirodni resurs i da se povećanjem selektivnosti (pakovanjem više kanala u isti opseg) postiže “ekološki” efekat odnosno štednja prirodnog resursa.



Slika 2.7 (a) Amplitudske karakteristike filtera 10-tog reda sa 4 nule prenosa: (a) Eliptičnog (E) sa slabljenjem u propusnom opsegu od 0.1 dB i Besselovog (Be) filtra sa korigovanom amplitudskom karakteristikom radi povećanja selektivnosti. Oba filtra su renormalizovani tako da na graničnoj frekvenciji ispoljavaju slabljenje od 3 dB; (b) Karakteristike grupnog kašnjenja filtera.

Ako se, međutim, postavlja zahtev da se sintetizuje IIR filtar koji pored selektivnosti poseduje i linearnu fazu, na raspolaganju stoji nekoliko rešenja:

- Direktna simultana aproksimacija i amplitudske i karakteristike grupnog kašnjenja (neminimalno-faznom funkcijom) u z -domenu [QuPP04].
- Direktna simultana aproksimacija i amplitudske i karakteristike grupnog kašnjenja (neminimalno-faznom funkcijom) u s -domenu [Lito79] sa potonjim preslikavanjem u z -domen.
- Sinteza selektivnog filtra i odgovarajućeg korektora grupnog kašnjenja u s -domenu sa potonjim preslikavanjem u z -domen [ToLA15a]. U literaturi je opisan primer simulatne sinteze amplitudske i karakteristike grupnog kašnjenja (neminimalno-faznom funkcijom) u s -domenu pri čemu je još simultano sintetizovan i korektor.
- Sinteza filtra sa linearnom fazom i korigovanom amplitudskom karakteristikom u s -domenu sa potonjim preslikavanjem u z -domen [MiLL15].

Ovo poslednje po svojoj prirodi najviše podseća na sintezu FIR filtera sa linearnom fazom pa će ovde biti razmotreno sa nešto više detalja.

Najpre se izabere prenosna funkcija sa svim nulama u beskonačnosti (polinomski filtar ili all-pole funkcija) u s -domenu koja ispoljava veoma dobru linearnost faze npr. upotrebom Beselovih (*Bessel*) polinoma [Thom49] oblika:

$$H(s) = a_0 / P_n(s) \quad (2.3)$$

gde je $P_n(s)$ Beselov polinom n -tog reda i $a_0 = P_n(0)$, a s normalizovana kompleksna kružna frekvencija. Ista fazna karakterisitka sa poboljšanom (korigovanom) selektivnošću se može ostvariti prenosnom funkcijom,

$$H_s(s) = \frac{a_0 \cdot \prod_{i=1}^{\lfloor n/2 \rfloor} (1 + s^2 / \omega_i^2)}{P_n(s)} \quad (2.4)$$

u kojoj su $\omega_i, i=1,2,\dots,\lfloor n/2 \rfloor$ nule prenosa (polovi slabljenja) čija se lokacija može dobiti jednostavnom optimizacionom procedurom pod ograničenjem koje za cilj ima ostvarivanje Čebiševljeve aproksimacije amplitudske karakteristike van propusnog opsega [Lito76, MiLL15].

Kao što će kasnije biti pokazano, sinteza filtara sa korigovanom amplitudskom karakteristikom koja je iskazana sa (2.4) dovodi do rešenja koje ima konstantno grupno kašnjenje u celom propusnom opsegu odnosno u celom opsegu prenosa gde se nalazi korisni deo spektar ulaznog signala. Primer amplitudske karakteristike jednog ovakvog filtra dat je na Sl. 2.7a, obeležen sa **Be**. Širina prelazne oblasti ovde je 2.1 puta (210%) veća od širine propusnog opsega. Nesumnjivo, selektivnost ovog filtra ne može se porediti sa selektivnošću eliptičnog i obrnuto, linearnost faze eliptičnog filtra ne može se porediti sa linearnošću faze Beselovog filtra što se lako vidi poređenjem karakteristika grupnog kašnjenja prikazanih na Sl. 2.7b.

Na osnovu ovih razmatranja, međutim, zaključujemo da postoji osnov za uverenje da je moguće ostvariti bolje performanse u smislu selektivnosti amplitudske karakteristike pod ograničenjem linearne faze IIR realizacijom nego što je to slučaj kod FIR filtra. Naravno, potrebno je pronaći i odgovarajuću transformacionu funkciju koja će očuvati kvalitet karakteristika i u z domenu.

Kod FIR filtara izlazna vrednost zavisi samo od linearne kombinacije trenutne i prethodnih vrednosti ulaza tako da su ovi filtri uvek stabilni (svi polovi prenosne funkcije sistema uvek u koordinatom početku u z -ravni). Naravno, ovo važi ukoliko su vrednosti ulaznih odmeraka ograničene što znači da su FIR filtri BIBO stabilni (*Bounded Input Bounded Output*). Nasuprot tome, IIR filtri mogu biti nestabilni zbog prisustva povratne sprege, tj. rekurzije izlaza. Dakle, IIR filtri mogu ispoljiti oscilatorno ponašanje (*limit cycles*). Ovaj problem se može značajno umanjiti korišćenjem paralelne realizacije filtra (Sl. 2.4).

Iz prethodnog razmatranja direktno proizilazi činjenica da su IIR filtri više osjetljivi na efekat konačne dužine digitalne reči. Ovaj problem se minimizuje skaliranjem ulaznog signala i korišćenjem isključivo binarne aritmetike u drugom komplementu kao što je ranije pomenuto.

Takođe, jedna od neospornih činjenica jeste da je sinteza FIR filtarske funkcije jednostavnija (pogotovu kod prozorskih FIR) nego u IIR slučaju. Međutim, ova pogodnost se obično plaća znatno većim redom filtra koji je potreban da bi se ispunili zahtevi. Ovo dalje uslovljava značajne optimizacije po pitanju potrošnje i površine silicijuma u kasnijim fazama projektovanja [Raba09]. Po ovom kriterijumu IIR filtri nesumnjivo imaju prednost. Naime, ista selektivnost se može postići sa značajno manjim redom IIR filtra nego što je to slučaj sa FIR. Samim tim latencija, tj. vreme obrade koje je potrebno da se dobije konačna vrednost izlazne reči i količina hardvera za obradu signala se drastično redukuju. To dalje omogućava da se i potrošnja kola značajno umani. Ovo je posebno važno kada je hardverska realizacija u pitanju. Generalno, kada se traži velika selektivnost i niska potrošnja, a pri tome nisu postavljeni ekstremno strogi zahtevi za linearnost faze, treba izabrati IIR koncept obrade signala. Ovakav koncept je izuzetno primenljiv u senzorskoj i mernoj/telekomunikacionoj instrumentaciji/opremi.

Komparativna analiza FIR i IIR filtara je sumirana u Tabeli 2.3 [Litw00, MSPL15]. Stavke koje bitno utiču na hardversku realizaciju su podvučene.

2.3 Složenost rešenja odnosno cena proizvodnje

Mada će o ovom aspektu biti još govora, ovde ćemo podsetiti da je složenost fizičke (pa i softverske) realizacije određena redom filtra. Kao što se vidi iz Tabele 2.3 IIR filtri ostvaruju postavljene zahteve sa nižim redovima što upućuje na zaključak da će i njihova realizacija biti jeftinija. Tome treba dodati i činjenicu da će složenije rešenje zahtevati duže vreme realizacije jednog ciklusa obrade. O ovome će takođe biti reči kasnije mada je jasno, naročito kada se radi o softverskoj realizaciji bez paralelnog procesiranja, da će vreme obrade (latencija) biti veće kod filtara višeg reda odnosno kod FIR filtara.

2.4 Primenljivosti u integrisanim kolima vrlo male potrošnje

Kada se radi o ovom aspektu poređenja, kao što će kasnije biti demonstrirano na jednom primeru, viši red filtra nedvosmisleno dovodi do veće površine na čipu. Dva puta veći broj množača će zahtevati dva puta veću površinu! Posebno je važno, međutim, da će zahtevati i dva puta veću snagu za svoj rad. Drugim rečima, red aproksimacije definiše potrošnju rešenja

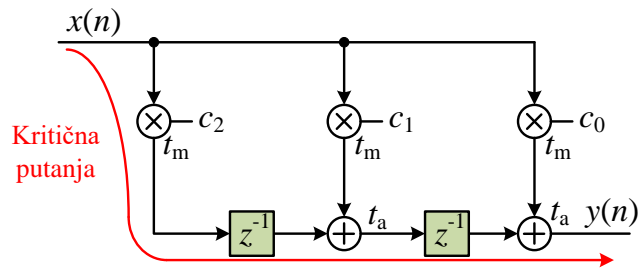
Tabela 2.3 Komparativna analiza IIR i FIR realizacije filtara

Osobina	IIR		FIR	
Selektivnost	Velika i lako se postiže	✓	Mala i zahteva ekstremno visoke redove filtara	✗
Faza	Uvek je moguće ostvariti linearnu fazu u celom propusnom opsegu. U slučajevima vrlo velike selektivnosti kolo za korekciju faze postaje složeno	✓✗	Uvek moguće ostvariti linearnu fazu u i van propusnog opsega po cenu složenosti sistema odnosno redukcije selektivnosti	✓✗
Stabilnost	Mogu biti nestabilni (<i>limit Cycles</i>)	✗	Uvek stabilni (svi polovi u 0 u z domenu tj. u $-\infty$ u s domenu)	✓
Osetljivost na konačnu preciznost	Veća	✗	Manja	✓
<u>Red</u>	Manji	✓	Veći (Zahteva više memorije odnosno površinu na čipu integrisanog kola.)	✗
Kontrola slabljenja u propusnom/nepropusnom opsegu	Kontinualna	✓	Diskretna	✗
<u>Potrošnja</u>	Manja	✓	Veća	✗
<u>Latencija</u>	Manja	✓	Veća	✗
Ostalo	Redom filtra se može kontrolisati maksimalno slabljenje u propusnom opsegu, minimalno slabljenje u nepropusnom opsegu i selektivnost.	✓	Red filtra ne utiče na minimalno slabljenje u nepropusnom opsegu	✗
	U cilju kontrole slabljenja u nepropusnom opsegu moguće simultano podešavati red filtra i broj i položaj nula prenosa.	✓	Polifazna implementacija moguća (<i>multirate signal processing</i>)	✓

u integrisanom obliku što može da bude odlučujući faktor pri donošenju konačne odluke o primenljivosti.

2.5 Maksimalna brzina rada FIR i IIR filtara

U hardverskoj realizaciji filtra brzina rada kola, tj. minimalna perioda T_{\min} uzorkovanja, je uglavnom ograničena propagacionim kašnjenjem aritmetičnikh blokova (množači, sabirači). Na Sl. 2.8 prikazan je FIR filtar drugog reda (dužine 3).



Slika 2.8 FIR filter drugog reda (tri tupa) bez primene pipeline tehnike

Filter se može opisati u vremenskom domenu diferencnom jednačinom

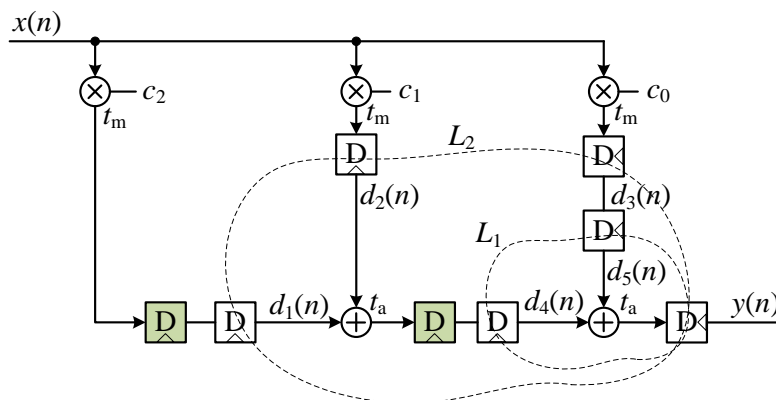
$$y(n) = c_0x(n) + c_1x(n-1) + c_2x(n-2) \quad (2.5)$$

Primenom Z transformacije na (2.5) dobija se

$$Y = (c_0 + c_1z^{-1} + c_2z^{-2}) \cdot X \quad (2.6)$$

U ovom slučaju (bez primene protočnih (pipeline) algoritama) minimalna perioda uzorkovanja je ograničena sa $T_{\min} \geq t_m + 2t_a$, gde su t_m i t_a propagaciona kašnjenja kola množača i sabirača, respektivno. Jednostavnosti radi, ali bez umanjavanja opštosti, usvojeno je da svi množači imaju isto propagaciono kašnjenje, t_m , dok svi sabirači imaju, t_a .

Kod FIR filtera primena protočnih algoritama se jednostavno izvodi uvođenjem dodatnih elemenata za kašnjenje (registara). Ovo se ostvaruje pažljivom analizom kritične putanje (putanja sa maksimalnim kašnjenjem).



Slika 2.9 Pipelined FIR filter drugog reda

Primena protočnih algoritama se može izvesti na više načina. Jedan od načina se može opisati sledećom procedurom:

1. Identifikovati kritičnu putanju (putanja sa maksimalnim kašnjenjem).
2. Počevši od izlaza u šemi izabrane arhitekture ucrtati koncentrične konture koje presecaju glavni izlaz i izlaze/ulaze svih aritmetičkih blokova u kritičnoj putanji.
3. Na preseku svake od krivih sa ulazima/izlazima aritmetičkih blokova postaviti registar.

Primer filtra sa Sl. 2.8 sa ubačenim *pipeline* registrima je prikazan na Sl. 2.9. Registri za ostvarivanje kašnjenja (z^{-1}) su zasenčeni, dok su *pipeline* registri prozračni.

Treba primetiti da u svakoj grani od ulaza do izlaza ima isti broj *pipeline* registara. Generisanje signala na izlazima registra pojedinih *pipeline* stepena (L_1 i L_2) za šest taktних intervala dato je u Tabeli 2.4. Analizom izlaznog signala, y , na kraju šestog taktnog intervala ($n = 5$) dolazi se do diferencne jednačine filtra:

Tabela 2.4 Generacije signala po pojedinim *pipeline* stepenima

Takt, n	x	d_1	d_2	d_3	d_4	d_5	y
0	x_0	0	0	0	0	0	0
1	x_1	0	c_1x_0	c_0x_0	0	0	0
2	x_2	c_2x_0	c_1x_1 ,	c_0x_1	0	c_0x_0	0
3	x_3	c_2x_1	c_1x_2	c_0x_2	c_1x_0	c_0x_1	c_0x_0
4	x_4	c_2x_2	c_1x_3	c_0x_3	$c_1x_1 + c_2x_0$	c_0x_2	$c_0x_1 + c_1x_0$
5	x_5	c_2x_3	c_1x_4	c_0x_4	$c_1x_2 + c_2x_1$	c_0x_3	$c_0x_2 + c_1x_1 + c_2x_0$
...

$$y(n) = c_0x(n-3) + c_1x(n-4) + c_2x(n-5) \quad (2.7)$$

Primenom Z transformacije dolazi se do oblika

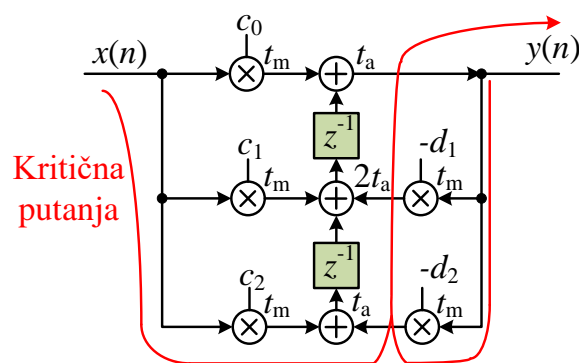
$$Y = z^{-3}(c_0 + c_1z^{-1} + c_2z^{-2}) \cdot X \quad (2.8)$$

Na osnovu ovoga zaključujemo da se primenom protočnih algoritama T_{min} redukuje na $T_{min} \geq \max\{t_m, t_a\}$, međutim latencija se povećala tri puta. Bitno je naglasiti da *pipeline* tehnika podrazumeva konstantan protok podataka. Zapravo, *pipeline* tehnikom se povećava protok (*throughput*). Dakle, postoji jasan kompromis (*trade-off*) između maksimalne brzine rada i latencije.

Izvođenje protočnih algoritama kod IIR filtara, nažalost, nije uvek jednostavan zadatak. Zbog postojanja rekurzije izlaza jednostavno uvođenje *pipeline* registara nije moguće.

Ukoliko bi se, na primer, uveli dodatni registri u povratnu petlju, prenosna funkcija IIR filtra bi bila izmenjena. Kod IIR filtra se primenjuje drugačiji pristup. Naime, koristi se tzv. *Scattered Look-Ahead* (SLA) tehnika uvedena u [PaMe89]. Ideja potiče od klasične *Look-Ahead* tehnike koja se često koristi kod povećavanja brzine rada *ripple-carry* brojača gde se praktično uvode dodatna kola koja unapred proračunavaju bitove prenosa potpunih sabirača [Parh07b]. Autori [PaMe89] su došli do zaključka da se sličan koncept može primeniti i na IIR filter.

Na Sl. 2.10 je prikazana jedna realizacija IIR filtra drugog reda koji se može opisati diferencnom jednčinom:



Slika 2.10 IIR filter drugog reda (Direktna Transponovana Forma II).

$$y(n) = c_0x(n) + c_1x(n-1) + c_2x(n-2) - d_1y(n-1) - d_2y(n-2). \quad (2.9)$$

Prenosna funkcija je data sa

$$H(z) = \frac{Y}{X} = \frac{c_0 + c_1z^{-1} + c_2z^{-2}}{1 + d_1z^{-1} + d_2z^{-2}} \quad (2.10)$$

Za proračun validne vrednosti izlaza potrebno je da, pored trenutne i zakašnjenih vrednosti ulaza, budu dostupne i prethodne vrednosti izlaza. Dakle, minimalna perioda uzorkovanja u ovom slučaju bi bila ograničena sa $T_{\min} \geq t_m + t_m + 4t_a$, gde prvo t_m predstavlja propagaciono kašnjenje operacije množenja sa koeficijentom c_2 , a drugo sa $-d_2$ (Sl. 2.10).

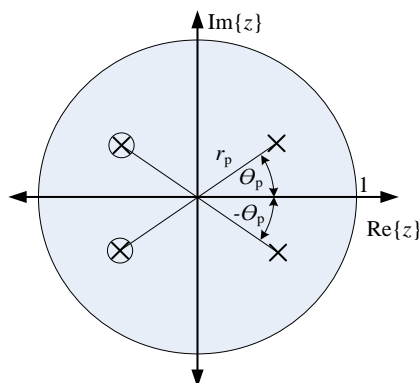
Neka su dalje, polovi (2.10) dati sa $z_{p,1/2} = \rho_p e^{\pm j\theta_p}$, $\rho_p < 1$, $\theta_p < \pi/2$. Ukoliko se u prenosnu funkciju IIR filtra uvedu dodatni polovi i nule za kompenzaciju istih na $z_{sla,1/2} = \rho_p e^{\pm j(\theta_p + \pi)}$, prenosna funkcija filtra dobija oblik

$$\begin{aligned}
H(z) &= \frac{c_0 + c_1 z^{-1} + c_2 z^{-2}}{1 + d_1 z^{-1} + d_2 z^{-2}} \cdot \frac{1 - d_1 z^{-1} + d_2 z^{-2}}{1 - d_1 z^{-1} + d_2 z^{-2}} \\
&= \frac{\hat{c}_0 + \hat{c}_1 z^{-1} + \hat{c}_2 z^{-2} + \hat{c}_3 z^{-3} + \hat{c}_4 z^{-4}}{1 + \hat{d}_2 z^{-2} + \hat{d}_4 z^{-4}}, \tag{2.11}
\end{aligned}$$

gde su

$$\begin{aligned}
\hat{c}_0 &= c_0 & \hat{d}_2 &= 2d_2 - d_1^2 \\
\hat{c}_1 &= c_1 - c_0 d_1 & \hat{d}_4 &= d_2^2 \\
\hat{c}_2 &= c_2 + c_0 d_2 - c_1 d_1 \\
\hat{c}_3 &= c_1 d_2 - c_2 d_1 \\
\hat{c}_4 &= c_2 d_2
\end{aligned}$$

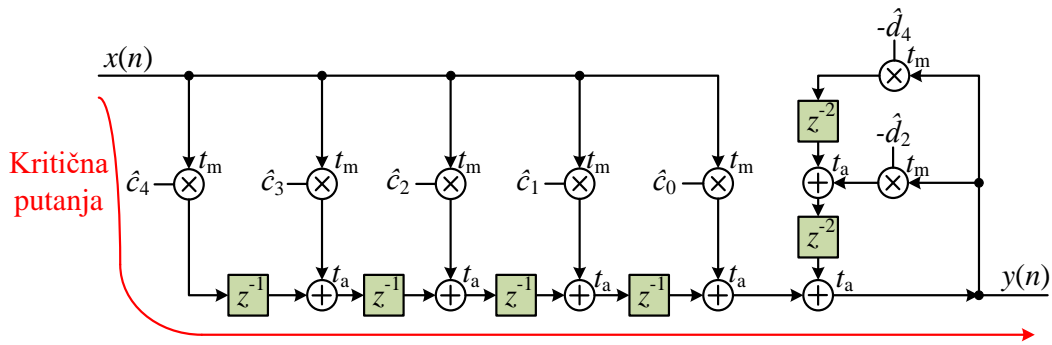
Položaj originalnih polova i novih nula i polova u (2.11) je ilustrovan na Sl. 2.11.



Slika 2.11 Položaj nula (o) i polova (x) *Scattered Look-Ahead* IIR filtra drugog reda

Ovim je obezbeđeno da izlaz ne zavisi od sukcesivnih, zakašnjjenih, vrednosti izlaza. U ovom slučaju zakašnjjene vrednosti razmaknute su za dva taktna intervala ili, što je ekvivalentno, dva *pipeline* nivoa. Da bi se ostvarilo L *pipeline* nivoa, treba uvesti dodatne polove/nule $z_{sla,k} = r_p e^{\pm j((2\pi/L)k + \theta_p)}$, $k = 1, 2, \dots, L-1$ u originalnu prenosnu funkciju IIR filtra.

SLA se može posmatrati i na sledeći način. Generalno, prenosna funkcija IIR filtra se može predstaviti i kao proizvod dve, nezavisne, prenosne funkcije, od kojih je jedna FIR (*feed-forward*, polinom u brojiocu (2.10)), a druga IIR (*feed-back all-pole*, razlomljeni deo tj. imenilac (2.10)) tipa. Praktično, uvođenjem SLA tehnike kritični deo prenosne funkcije (IIR deo) se usporava, na račun produženja FIR dela. Pošto je obezbeđeno da IIR deo filtra radi sporije zahtev za propagaciona kašnjenja aritmetičkih blokova u tom delu filtra se redukuje. Realizacija filtra sa prenosnom funkcijom (2.11) je data na Sl. 2.12.



Slika 2.12 Scattered Look-Ahead IIR filtar drugor reda

Sada je minimalna perioda uzorkovanja ograničena sa $T_{\min} \geq t_m + 5t_a$. Kako je operacija množenja obično značajno „skuplja“ po pitanju kašnjenja od operacije sabiranja, tj. $t_m \gg t_a$, T_{\min} je uglavnom ograničeno vremenom realizacije jedne operacije množenja, tj. t_m .

Kada se ovaj rezultat uporedi sa klasičnom *pipeline* tehnikom kod FIR filtra gde je za minimalnu periodu dobijeno $T_{\min} \geq \max\{t_m, t_a\} \approx t_m$, dolazi se do zaključka da se IIR i FIR filtar može realizovati sa približno istim performansama po pitanju brzine rada. Naravno, za manji red FIR filtri su u prednosti po ovom kriterijumu. Međutim, kada se uzme u obzir da je red IIR filtra značajno manji od FIR filtra za iste gabarite, ova prednost se retko praktično ispoljava. Treba istaći da je glavni nedostatak SLA tehnike to što zahteva veći broj bitova za predstavljanje koeficijenata. Razlog tome je što uvedeni parovi polova i nula moraju da budu dobro upareni (sa dovoljnom preciznošću). U suprotnom može doći do značajnog izobličenja prenosne funkcije filtra usled sistematske greške. Pošto uvedeni polovi zadržavaju poluprečnik (moduo) polova originalne prenosne funkcije, SLA tehnika garantuje stabilnost ukoliko je i polazni filtar stabilan.

3 Transformacije iz s u z domen za projektovanje IIR filtara na osnovu zahteva iz frekvencijskog domena

U uvodnom izlaganju pobrojane su neke od najpopularnijih metoda za preslikavanje filtarske funkcije iz s u z domen. Ovde će biti detaljnije obrađene transformacije koje se baziraju na aproksimaciji operacije diferenciranja. Prvo će biti obrađena bilinearna transformacija. Zatim će biti prezentovana nova transformacija drugog reda. Korišćenjem ovih transformacija izveden je postupak za projektovanje CMAC IIR filtara prilagođen paralelnoj realizaciji. Takođe će biti ispitan i uticaj predstavljanja koeficijenata filtra konačnim brojem cifara (aritmetika fiksnog zareza). Rezultati VLSI realizacije u 65nm CMOS procesu će dati uvid u performanse i potrošnju kola selektivnih IIR i FIR filtara.

3.1 Upotreba bilinearne transformacije za sintezu IIR filtara

Neka je prenosna funkcija analognog filtra data sa (3.1).

$$H_A(s) = \frac{\sum_{k=0}^m b_k s^k}{\sum_{k=0}^n a_k s^k} \quad (3.1)$$

Ovde su koeficijenti polinomske funkcije u brojiocu i imeniocu dati kao vektori \mathbf{a} i \mathbf{b} veličine $n+1$, i $m+1$, respektivno. Pretpostavlja se $a_n=1$. Predstava data u (3.1) potiče od diferencijalne jednačine (3.2).

$$\sum_{k=0}^n a_k \frac{d^k y(t)}{dt^k} = \sum_{k=0}^m b_k \frac{d^k x(t)}{dt^k} \quad (3.2)$$

Primenom Laplasove (*Laplace*) transformacije na (3.2) dobija se oblik dat sa (3.1). Na osnovu linearnosti Laplasove i Z transformacije jasno je da se pri prelasku u z domen analogni sabirači jednostavno zamenjuju digitalnim. Ostaje samo da se nađe preslikavanje kontinualne funkcije diferenciranja u diskretni domen

$$\dot{g}(t) = f(t) \quad (3.3)$$

U (3.3) $g(t)$ se može posmatrati kao izlaz, a $f(t)$ kao ulaz analognog integratora. Primenom Laplasove transformacije na (3.3) dobija se

$$s \cdot G(s) = F(s) \quad (3.4)$$

Dakle, da bi se prenosna funkcija data sa (3.1) preslikala u digitalni domen treba naći adekvatnu predstavu (3.4) u z domenu. Ovaj zadatak se može ostvariti pomoću sledećeg uopštenog postupka:

1. Aproksimirati izvod funkcije, tj. diskretizovati vreme i transformisati diferencijalnu jednačinu u diferencnu,
2. Primeniti Z transformaciju na dobijenu diferencnu jednačinu i
3. Korišćenjem analogije sa (3.4) naći smenu promenljive s aproksimativnim izrazom dobijenim u koraku 2.

Izvod funkcije se može numerički interpretirati pomoću linearne višekoračne formule (*linear multistep formulae*) [SaSh68] na sledeći način:

$$\dot{y}_r = -\frac{\alpha_0}{\beta_0 T} y_r - \frac{1}{\beta_0 T} \cdot \sum_{j=1}^l (\alpha_j y_{r-j} + T \cdot \beta_j \dot{y}_{r-j}), \quad (3.5)$$

ovde je r brojač vremenskih koraka gde se procena izvoda obavlja u svakom rT trenutku, T je perioda uzorkovanja (korak diskretizacije vremenske ose), l je red formule. Takođe se koriste i dva vektora α i β veličine $l+1$ koji definišu koeficijente formule. Izborom reda formule i vrednosti koeficijenta mogu se dobiti različite formule za aproksimaciju izvoda u diskretnom domenu.

Tako npr. izborom $l=1$, $\alpha_0=-1$, $\alpha_1=1$, $\beta_0=1$, i $\beta_1=0$ dobija se:

$$\dot{y}_r = \frac{1}{T} (y_r - y_{r-1}), \quad (3.6)$$

odnosno primenom Z transformacije,

$$\dot{Y} = \frac{1}{T} (1 - z^{-1}) Y, \quad (3.7)$$

što odgovara *Backward Euler* aproksimaciji. Na osnovu analogije sa s domenom tj.,

$$\dot{Y} = s Y \quad (3.8)$$

dolazi se do aproksimativnog izraza za kompleksnu kružnu frekvenciju s u z domenu,

$$s = \frac{1}{T} (1 - z^{-1}) = \frac{z-1}{Tz}. \quad (3.9)$$

Slično, izborom koeficijenata $l=1$, $\alpha_0=-2$, $\alpha_1=2$, $\beta_0=1$, $\beta_1=1$ dobija se:

$$\dot{y}_r = -\dot{y}_{r-1} + \frac{2}{T}(y_r - y_{r-1}) \quad (3.10)$$

$$(1 + z^{-1})\dot{Y} = \frac{2}{T}(1 - z^{-1})Y \quad (3.11)$$

trapezno pravilo (bilinearna transformacija),

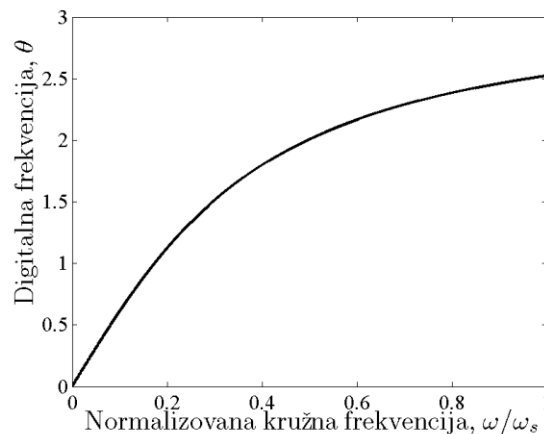
$$s = \frac{2}{T} \frac{(1 - z^{-1})}{(1 + z^{-1})} = \frac{2}{T} \frac{(z - 1)}{(z + 1)}. \quad (3.12)$$

Aproksimacija (3.12) se smatra najpopularnijom pošto čuva stabilnost i unosi izuzetno malu distorziju u amplitudsku karakteristiku što je najčešći zahtev prilikom projektovanja filtera. Imaginarna osa, $j\omega$, iz s domena se slika na jedinični krug u z domenu tako da sve nule na realnim frekvencijama uvek ostaju na jediničnom krugu i obezbeđuju beskonačno slabljenje kao i u originalnoj prenosnoj funkciji u s domenu. Ovo je veoma važno kod selektivnih filtera.

Pošto je (3.12) racionalna funkcija po z jasno je da preslikavanje realnih frekvencija nije linearno. Do formule za preslikavanje (*wrapping*) realnih, analognih, frekvencija (ω) u digitalne (θ) se dolazi smenom $z = e^{j\theta}$ i $s = j\omega$ u (3.12):

$$\theta = 2 \arctg\left(\frac{\omega T}{2}\right). \quad (3.13)$$

Pošto u digitalnom domenu ima smisla posmatrati samo spektar signala do frekvencije odmeravanja (ili do polovine frekvencije odmeravanja), dovoljno je preslikati samo $[0:\omega_s]$, gde je $\omega_s = 2\pi/T$. Na Sl. 3.1 je ilustrovana relacija (3.13).



Slika 3.1 Preslikavanje realnih, analognih frekvencija u digitalne korišćenjem bilinearne transformacije

3.2 Nova transformacija drugog reda

Karakteristično za sve do sada razmatrane metode iz ove klase jeste da red digitalnog filtra odgovara redu analognog, tj. za red formule je uvek birano $l=1$. U cilju izvođenja nove aproksimacione formule u [MiPL14] izabran je drugačiji pristup. Naime usvojeno je da formula bude drugog reda. Motiv za ovakav izbor leži u činjenice da formula drugog reda ispoljava lokalnu grešku zaokruživanja (*Local Truncation Error* – LTE) proporcionalnu trećem izvodu vremenskog odziva kao što je pokazano u [LiZw96]. Treći izvod teži izuzetno maloj vrednosti ukoliko se radi o aperiodičnim funkcijama pogotovu kada se traži monoton odziv.

Kako bi se izabrali odgovarajući koeficijenti za (3.5) iskorišćene su tzv. Girove (*Gear's*) formule [Gear71, SaSh68]. Ovo su najčešće korišćene formule za aproksimaciju izvoda u programima za simulaciju elektronskih kola, SPICE [Nage75]. Izbor koeficijenata $l=2$, $\alpha_0=1$, $\alpha_1=4/3$, $\alpha_2=-1/3$, $\beta_0=2/3$, $\beta_1=\beta_2=0$ u (3.1) daje

$$\dot{y}_r = \frac{3}{2T} y_r - \frac{2}{T} \cdot y_{r-1} + \frac{1}{2T} y_{r-2}, \quad (3.14)$$

odnosno,

$$s = \frac{1}{2T} (3 - 4z^{-1} + z^{-2}) = \frac{1}{2T} \cdot \frac{3z^2 - 4z + 1}{z^2}. \quad (3.15)$$

Ovo je formula drugog reda, dakle digitalni filter će biti dvostruko većeg reda od originalnog analognog. Drugim rečima moguće je dobiti samo digitalni filter parnog reda.

Razvijanjem (3.15) dobija se kvadratna jednačina,

$$z^{-2} - 4z^{-1} + (3 - 2Ts) = 0, \quad (3.16)$$

čija rešenja daju zavisnost kompleksne promenljive z od kompleksne kružne frekvencije s ,

$$\begin{aligned} (z^{-1})_{1/2} &= \frac{4 \pm \sqrt{16 - 4(3 - 2Ts)}}{2} = 2 \pm \sqrt{1 + 2Ts} \Rightarrow \\ (z)_{1/2} &= \frac{1}{2 \pm z_s}, \end{aligned} \quad (3.17)$$

gde je

$$z_s = \sqrt{1 + 2Ts}. \quad (3.18)$$

U cilju ispitivanja karakteristika nove transformacije potrebno je doći do veze između modula i argumenta kompleksne promenljive z i realnog i imaginarnog dela kompleksne

kružne frekvencije s . Do ove veze se dolazi daljom analizom (3.18) pri čemu se uvodi smena $s = \sigma + j\omega$.

$$\begin{aligned}
 z_s &= \sqrt{1 + 2T\sigma + j2T\omega} \\
 &= \sqrt{\sqrt{(1 + 2T\sigma)^2 + (2T\omega)^2}} e^{j \arctg \frac{2T\omega}{1 + 2T\sigma}} \\
 &= \sqrt{\sqrt{(1 + 2T\sigma)^2 + (2T\omega)^2}} z_e,
 \end{aligned} \tag{3.19}$$

gde je $z_e = \sqrt{e^{j \arctg \frac{2T\omega}{1 + 2T\sigma}}} = e^{j \frac{\alpha}{2}}$, $\alpha = \arctg \frac{2T\omega}{1 + 2T\sigma}$.

Primenom trigonometrijskih identiteta,

$$\begin{aligned}
 \cos^2 \frac{\alpha}{2} &= \frac{1 + \cos \alpha}{2}, \quad \sin^2 \frac{\alpha}{2} = \frac{1 - \cos \alpha}{2} \\
 \cos \frac{\alpha}{2} &= \sqrt{\frac{1 + \cos \alpha}{2}}, \quad \sin \frac{\alpha}{2} = \sqrt{\frac{1 - \cos \alpha}{2}}, \\
 \cos \alpha &= \frac{1}{\sqrt{1 + \operatorname{tg}^2 \alpha}},
 \end{aligned} \tag{3.20}$$

za z_e se dobija,

$$\begin{aligned}
 z_e &= \cos \frac{\alpha}{2} + j \sin \frac{\alpha}{2} \\
 &= \sqrt{\frac{1 + \cos \alpha}{2}} + j \sqrt{\frac{1 - \cos \alpha}{2}} \\
 &= \sqrt{\frac{1 + \frac{1}{\sqrt{1 + \operatorname{tg}^2 \alpha}}}{2}} + j \sqrt{\frac{1 - \frac{1}{\sqrt{1 + \operatorname{tg}^2 \alpha}}}{2}} \\
 &= \sqrt{\frac{\sqrt{1 + \operatorname{tg}^2 \alpha} + 1}{2\sqrt{1 + \operatorname{tg}^2 \alpha}}} + j \sqrt{\frac{\sqrt{1 + \operatorname{tg}^2 \alpha} - 1}{2\sqrt{1 + \operatorname{tg}^2 \alpha}}} = \\
 &= \sqrt{\frac{\sqrt{1 + \left(\frac{2T\omega}{1 + 2T\sigma}\right)^2} + 1}{2\sqrt{1 + \left(\frac{2T\omega}{1 + 2T\sigma}\right)^2}}} + j \sqrt{\frac{\sqrt{1 + \left(\frac{2T\omega}{1 + 2T\sigma}\right)^2} - 1}{2\sqrt{1 + \left(\frac{2T\omega}{1 + 2T\sigma}\right)^2}}} \\
 &= \sqrt{\frac{\sqrt{(1 + 2T\sigma)^2 + (2T\omega)^2} + (1 + 2T\sigma)}{2\sqrt{(1 + 2T\sigma)^2 + (2T\omega)^2}}} + j \sqrt{\frac{\sqrt{(1 + 2T\sigma)^2 + (2T\omega)^2} - (1 + 2T\sigma)}{2\sqrt{(1 + 2T\sigma)^2 + (2T\omega)^2}}}.
 \end{aligned} \tag{3.21}$$

Smenom (3.21) u (3.19) z_s postaje,

$$z_s = x_s + jy_s, \quad (3.22)$$

pri čemu su x_s i y_s definisani sa,

$$\begin{aligned} x_s &= \sqrt{\frac{\sqrt{(1+2T\sigma)^2 + (2T\omega)^2} + (1+2T\sigma)}{2}} \\ y_s &= \sqrt{\frac{\sqrt{(1+2T\sigma)^2 + (2T\omega)^2} - (1+2T\sigma)}{2}}. \end{aligned} \quad (3.23)$$

Smenom (3.22) u (3.17) i predstavljanjem kompleksne promenljive z kao $(z)_{1/2} = \rho e^{j\theta}$ gde je $\rho = |z|_{1/2}$ moduo, a $\theta = \arg\{z_{1/2}\}$ argument dobija se

$$\begin{aligned} \rho &= \frac{1}{|2 \pm (x_s + jy_s)|} \\ &= \frac{1}{\sqrt{(2 \pm x_s)^2 + y_s^2}} = \frac{1}{\sqrt{4 \pm 4x_s + x_s^2 + y_s^2}} \\ &= \frac{1}{\sqrt{4 \pm 2\sqrt{2} \sqrt{\sqrt{(1+2T\sigma)^2 + (2T\omega)^2} + (1+2T\sigma)} + \sqrt{(1+2T\sigma)^2 + (2T\omega)^2}}} \\ &= \frac{1}{\sqrt{4 \pm \sqrt{8 \left(\sqrt{(1+2T\sigma)^2 + (2T\omega)^2} + (1+2T\sigma) \right) + \sqrt{(1+2T\sigma)^2 + (2T\omega)^2}}}}, \end{aligned} \quad (3.24)$$

za moduo i

$$\begin{aligned} \theta &= -\operatorname{arctg} \frac{\pm y_s}{2 \pm x_s} \\ &= \operatorname{arctg} \frac{\mp \sqrt{\sqrt{(1+2T\sigma)^2 + (2T\omega)^2} - (1+2T\sigma)}}{2\sqrt{2} \pm \sqrt{\sqrt{(1+2T\sigma)^2 + (2T\omega)^2} + (1+2T\sigma)}} \\ &= \operatorname{arctg} \frac{\mp \sqrt{\sqrt{(1+2T\sigma)^2 + (2T\omega)^2} - (1+2T\sigma)}}{2\sqrt{2} \pm \sqrt{\sqrt{(1+2T\sigma)^2 + (2T\omega)^2} + (1+2T\sigma)}}, \end{aligned} \quad (3.25)$$

za argument.

Na ovaj način kompletna s ravan je preslikana u z ravan. Od posebnog značaja je preslikavanje granice između leve i desne s poluravnini pošto je stabilnost osigurana samo ukoliko se polovi prenosne funkcije nalaze isključivo u levoj poluravnini ($\sigma < 0$). Da bi se

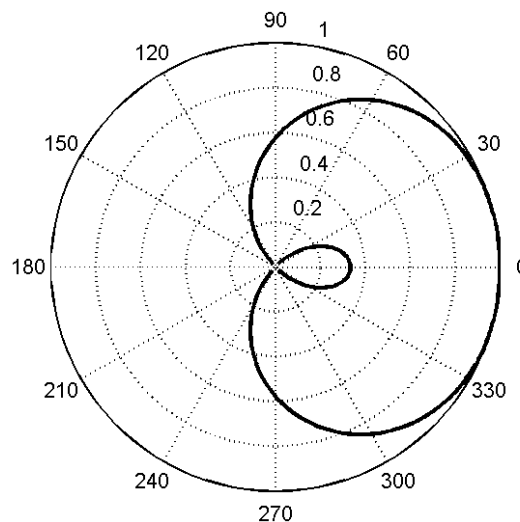
ispitala stabilnost aproksimacione funkcije drugog reda potrebno je preslikati granicu leve i desne poluravnine tj. imaginarnu, $j\omega$, osu u z ravan. Za $s=j\omega$ ($\sigma = 0$) dobijaju se sledeće formule za preslikavanje u z domen

$$\rho \Big|_{\sigma=0} = \frac{1}{\sqrt{4 \pm \sqrt{8(\sqrt{1+(2T\omega)^2} + 1) + \sqrt{1+(2T\omega)^2}}} \quad (3.26)$$

i

$$\theta \Big|_{\sigma=0} = \arctg \frac{\mp \sqrt{\sqrt{1+(2T\omega)^2} - 1}}{2\sqrt{2} \pm \sqrt{\sqrt{1+(2T\omega)^2} + 1}}. \quad (3.27)$$

Na Sl. 3.2 je prikazan rezultat preslikavanja (3.26) i (3.27) u polarnom koordinatnom sistemu u z ravni. Može se videti da se kompletna leva poluravan iz s domena preslikava u površinu ograničenu krivom kardiodnog oblika unutar jediničnog kruga u z ravni. Dobro je poznato da se sistem može smatrati stabilnim u z domenu ukoliko se svi polovi nalaze unutar jediničnog kruga. To praktično znači da, ukoliko je sistem stabilan u s domenu (svi polovi u levoj poluravnini), preslikavanje novom formulom drugog reda obezbeđuje stabilnost sistema i u z domenu. Dakle, po pitanju stabilnosti nova transformacija je uporediva sa bilinearnom.



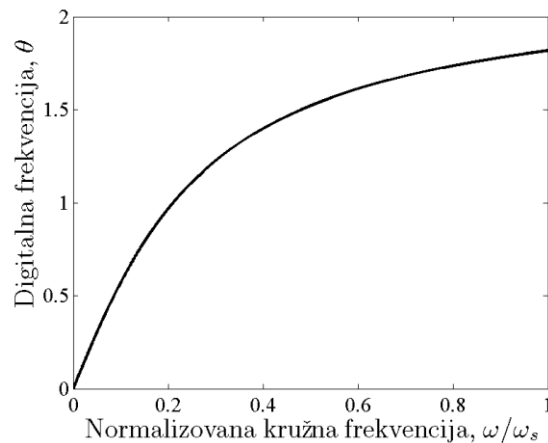
Slika 3.2 Imaginarna osa $j\omega$ preslikana iz s u z ravan upotrebom nove transformacije (3.15).

Preslikavanje analognih u digitalne frekvencije je dato sa (3.27). Međutim, relacija (3.27) nudi dva rešenja. Pošto prvo rešenje generiše negativne vrednosti za θ u intervalu od interesa ($[0:\omega_s]$), treba izabrati drugo rešenje. Pošto se radi o arkus tangens funkciji potrebno je još odrediti za koju vrednost analogne frekvencije imenilac menja znak i korigovati rezultat na

odgovarajući način. Formula za preslikavanje analognih frekvencija u digitalne primenom transformacije drugog reda je,

$$\theta = k\pi + \arctg \frac{\sqrt{\sqrt{1+(2T\omega)^2} - 1}}{2\sqrt{2} - \sqrt{\sqrt{1+(2T\omega)^2} + 1}}, k = \begin{cases} 0, & \omega T \leq \sqrt{12} \\ 1, & \omega T > \sqrt{12} \end{cases}. \quad (3.28)$$

Kriva na Sl. 3.3 prikazuje relaciju (3.28).



Slika 3.3 Preslikavanje realnih, analognih frekvencija u digitalne korišćenjem transformacije drugog reda

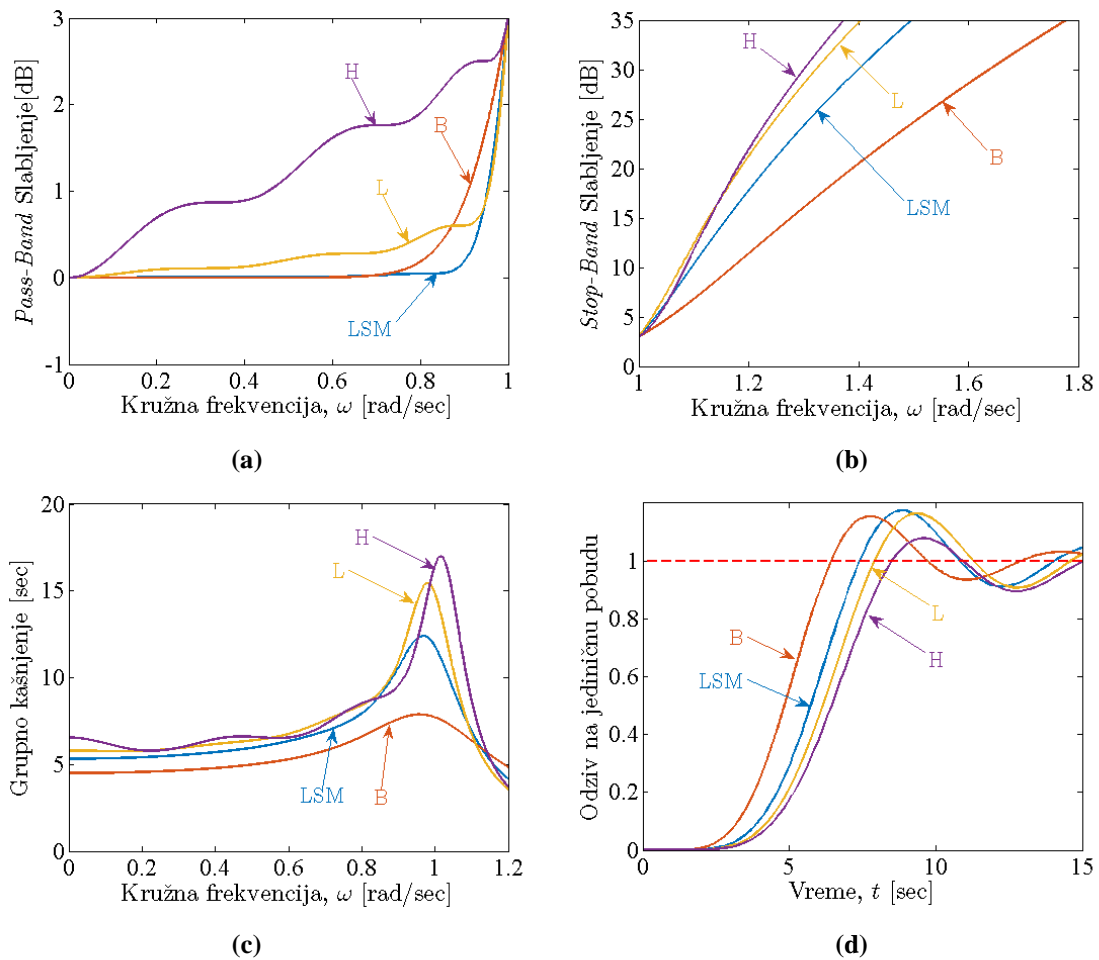
Nakon primene bilo koje od aproksimacija i smene promenljivih u (3.1) dobija se reprezentacija analognog filtra u digitalnom domenu,

$$H_D(s) = \frac{\sum_{k=0}^m \hat{b}_k z^{-k}}{\sum_{k=0}^n \hat{a}_k z^{-k}}, \quad (3.29)$$

gde se podrazumeva $\hat{a}_0 = 1$. Jednostavnosti radi, na dalje će koeficijenti \hat{a}_k i \hat{b}_k biti predstavljeni sa c_k i d_k , respektivno.

3.3 CMAC filtri u z domenu

Da bi se stekao intuitivni osećaj za osobine četiri klase CMAC filtera, karakteristike za *all-pole* filterarske funkcije sedmog reda ($n=7$) u s -domenu date su na Sl. 3.4. Generalno, može se videti da LSM filtri nude najbolju aproksimaciju u propusnom opsegu, dok istovremeno ispoljavaju dobru selektivnost u nepropusnom opsegu. Kao što je i očekivano, H filtri imaju najveće slabljenje u nepropusnom opsegu.



Slika 3.4 Karakteristike četiri tipa, B, L, H i LSM, CMAC filtarskih funkcija sedmog reda: slabljenje unutar (*Pass-Band*) (a) i van (*Stop-Band*) (b) propusnog opsega, grupno kašnjenje (c) i odziv na jediničnu pobudu (*Step response*) (d).

L filtri se mogu smatrati kompromisom između H i LSM filtara. Osim maksimalno ravne karakteristike, druge prednosti B filtara u odnosu na ostala tri CMAC tipa nisu primetne.

Kada se posmatra grupno kašnjenje (Sl. 3.4c) i jedinični odziv (Sl. 3.4d) može se odmah primetiti da je izobličenje grupnog kašnjenja direktno povezana sa selektivnošću filtra. Veća selektivnost uzrokuje veći ekstremum grupnog kašnjenja. Sa druge strane, svi filtri ispoljavaju premašenje (*overshoot*) i podbačaj (*undershoot*) u jediničnom odzivu. Sada filtar sa najvećom selektivnošću u nepropusnom opsegu, H filtar, ima najmanje premašenje ali i najveći podbačaj. Na Sl. 3.4d treba takođe primetiti da B filtri imaju najmanje kašnjenje. Nažalost, ova prednost se gubi usled povećanja reda filtra potrebnog da bi se kompenzovao manjak selektivnosti.

Da bi se došlo do konačne, paralelne, realizacije u z -domenu potrebno je prvo originalnu funkciju u s -domenu razviti na parcijalne sabirke. Treba krenuti od faktorisanog oblika prenosne funkcije filtra u s -domenu

$$H_a(s) = \frac{\sum_{k=0}^m b_k s^k}{\sum_{i=0}^n a_i s^i} = \frac{b_m \prod_{k=0}^{m-1} (s - z_k)}{a_n \prod_{i=0}^{n-1} (s - p_i)} = \frac{b_m}{a_n} H(s), \quad (3.30)$$

gde je $n > m$, b_m i a_n koeficijenti uz najviši stepen polinoma u brojiocu i imeniocu, z_k i p_i nule i polovi prenosne funkcije, respektivno. Ukoliko je $m = n$ pre faktorizacije potrebno je najpre deljenjem polinoma brojioca polinomom imenioca doći do forme (3.31).

$$H_a(s) = \frac{b_n}{a_n} \left(1 + \frac{\prod_{k=0}^{n-2} (s - q_k)}{\prod_{i=0}^{n-1} (s - p_i)} \right) = \frac{b_n}{a_n} (1 + H(s)), \quad (3.31)$$

gde su q_k koreni polinoma,

$$P(s) = \sum_{k=0}^{n-1} \left(\frac{b_k}{b_n} - \frac{a_k}{a_n} \right) s^k. \quad (3.32)$$

U (3.31), član koji ne zavisi od frekvencije predstavlja direktni član (isprekidana linija na Sl. 2.4). Sledeći korak je proračun ostataka u polovima. Ukoliko se radi o jednostrukim polovima može se koristiti

$$H(s) = \sum_{i=0}^{n-1} \frac{r_i}{(s - p_i)}, \quad (3.33)$$

gde je r_i ostatak u polu p_i .

Pažljivim kombinovanjem konjugovano-kompleksnih parova polova sa odgovarajućim ostacima dolazi se do sume parcijalnih sabiraka koja omogućava paralelnu realizaciju filtra:

$$H(s) = \begin{cases} \sum_{i=0}^{n/2} H_e(s), & N - \text{parno} \\ H_o(s) + \sum_{i=1}^{\lfloor n/2 \rfloor} H_e(s), & N - \text{neparno.} \end{cases} \quad (3.34)$$

U (3.34) indeks e označava da se radi o sekcijama drugog reda dobijenim kombinovanjem konjugovano-kompleksnih parova polova i ostataka, a indeks o označava sekciju prvog reda dobijenu od ostatka u jednostavnom, realnom, polu.

Prenosna funkcija drugog reda je oblika

$$H_e(s) = G_i \frac{s + b_{0,i}}{s^2 + a_{1,i}s + a_{0,i}}, \quad (3.35)$$

čiji je pojačanje $G_i = 2\text{re}\{r_i\}$, a koeficijenti $b_{0,i} = -\left(\text{re}\{p_i\} + \frac{\text{im}\{r_i\}\text{im}\{p_i\}}{\text{re}\{r_i\}}\right)$, $a_{1,i} = -2\text{re}\{p_i\}$,

$a_{0,i} = |p_i|^2$. Sa „im{ }“ i „re{ }“ su označeni imaginarni i realni deo i -tog pola/ostatka.

Oblik prenosne funkcije prvog reda je

$$H_o(s) = G_o \frac{1}{s + a_o}, \quad (3.36)$$

sa pojačanjem $G_o = r_o$, i koeficijentom $a_o = -p_o$.

Sada se primenom bilinearne transformacije (3.12) sekcije mogu preslikati u z -domen.

Sekcija drugog reda je

$$H_e(z) = \frac{c_{0,i} + c_{1,i}z^{-1} + c_{2,i}z^{-2}}{1 + d_{1,i}z^{-2} + d_{2,i}z^{-2}}, \quad (3.37)$$

čiji su koeficijenti definisani sa

$$\begin{aligned} c_{0,i} &= G_i T \frac{b_{0,i}T + 2}{a_{0,i}T^2 + 2a_{1,i}T + 4} & d_{1,i} &= \frac{2a_{0,i}T^2 - 8}{a_{0,i}T^2 + 2a_{1,i}T + 4} \\ c_{1,i} &= G_i T \frac{2Tb_{0,i}}{a_{0,i}T^2 + 2a_{1,i}T + 4} & d_{2,i} &= \frac{a_{0,i}T^2 - 2a_{1,i}T + 4}{a_{0,i}T^2 + 2a_{1,i}T + 4} \\ c_{2,i} &= G_i T \frac{b_{0,i}T - 2}{a_{0,i}T^2 + 2a_{1,i}T + 4} \end{aligned}$$

Za sekciju prvog reda dobija se,

$$H_o(z) = \frac{c_{0,o} + c_{1,o}z^{-1}}{1 + d_{1,o}z^{-1}}, \quad (3.38)$$

gde je $c_{0,o} = c_{1,o} = \frac{G_o T}{a_o T + 2}$, $d_{1,o} = \frac{a_o T - 2}{a_o T + 2}$.

Ukoliko, se koristi nova transformacija drugog reda (kvadratna) (3.15), za sekciju drugog reda se dobija

$$H_e(z) = \frac{c_{0,i} + c_{1,i}z^{-1} + c_{2,i}z^{-2}}{1 + d_{1,i}z^{-1} + d_{2,i}z^{-2} + d_{3,i}z^{-3} + d_{4,i}z^{-4}}, \quad (3.39)$$

gde su koeficijenti definisani sa

$$\begin{aligned} c_{0,i} &= 2G_i T \frac{2b_{0,i}T + 3}{4a_{0,i}T^2 + 6a_{1,i}T + 9} & d_{1,i} &= -\frac{8Ta_{1,i} + 24}{4a_{0,i}T^2 + 6a_{1,i}T + 9} \\ c_{1,i} &= -8G_i T \frac{1}{4a_{0,i}T^2 + 6a_{1,i}T + 9} & d_{2,i} &= \frac{2a_{1,i}T + 22}{4a_{0,i}T^2 + 6a_{1,i}T + 9} \\ c_{2,i} &= 2G_i T \frac{1}{4a_{0,i}T^2 + 6a_{1,i}T + 9} & d_{3,i} &= -\frac{8}{4a_{0,i}T^2 + 6a_{1,i}T + 9} \\ & & d_{4,i} &= \frac{1}{4a_{0,i}T^2 + 6a_{1,i}T + 9}. \end{aligned}$$

Slično, za sekciju prvog reda dobija se

$$H_o(z) = \frac{c_{0,o}}{1 + d_{1,o}z^{-1} + d_{2,o}z^{-2}}, \quad (3.40)$$

gde je $c_{0,o} = 2TG_o \frac{1}{3 + 2Ta_o}$, $d_{1,o} = -\frac{4}{3 + 2Ta_o}$, $d_{2,o} = \frac{1}{3 + 2Ta_o}$.

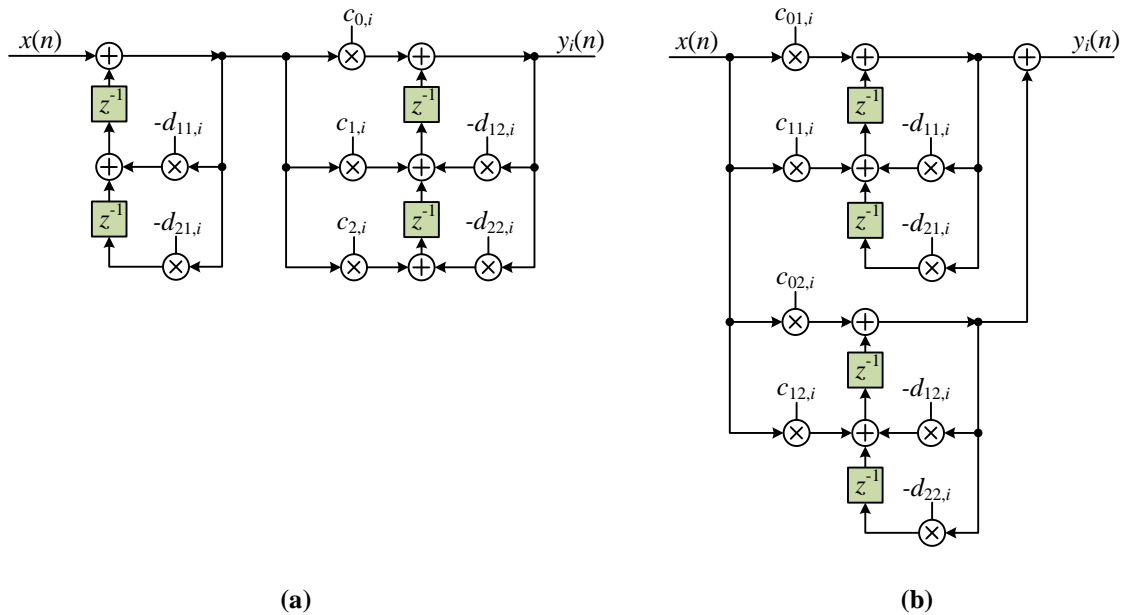
Kao što je i očekivano, rezultujuće sekcije u z -domenu imaju dvostruko veći red od polaznih sekcija u s -domenu kada se za preslikavanje koristi transformacija drugog reda. Takođe treba primetiti da polazne sekcije drugog reda, nakon preslikavanja, postaju četvrtog reda. Dakle, postoje tri mogućnosti za realizaciju.

Prva je da se (3.39) implementira direktno (Sl. 2.2). Kao što je ranije diskutovano ova opcija je najgori izbor.

Druga varijanta je da se koristi kaskadna realizacija. Pošto (3.39) ima dvostruku nulu u nuli dovoljno je faktorisati samo polinom imenioca, a zatim kombinacijom odgovarajućih konjugovano kompleksnih parova polova formirati sekcija drugog reda. Kaskadna realizacija (3.39) bi bila,

$$H_e(z) = \frac{1}{1 + d_{11,i}z^{-1} + d_{21,i}z^{-2}} \cdot \frac{c_{0,i} + c_{1,i}z^{-1} + c_{2,i}z^{-2}}{1 + d_{12,i}z^{-1} + d_{22,i}z^{-2}}, \quad (3.41)$$

gde su koeficijenti imenioca sekcija drugog reda, $d_{1k,i} = -2 \operatorname{re}\{w_{k,i}\}$, $d_{2k,i} = |w_{k,i}|^2$, $k = 1, 2$, a $\{w_{k,i}, w_{k,i}^*\}$ polovi prenosne funkcije. Realizacija (3.41) je data na Sl. 3.5a.



Slika 3.5 Realizacija sekcija četvrtog reda: (a) serijska (kaskadna) i (b) paralelna (kaskodna)

Treća varijanta je da se koristi paralelna realizacija. Za to je potrebno naći ostatake u polovima, ali ovoga puta u z -domenu. Pošto je polinom imenioca po z^{-1} većeg reda od polinoma brojioca nema potrebe za deljenjem polinoma. Drugim rečima direktni član je jednak nuli, pa za (3.39) može da se piše

$$H_e(z) = \frac{c_{01,i} + c_{11,i}z^{-1}}{1 + d_{11,i}z^{-1} + d_{21,i}z^{-2}} + \frac{c_{02,i} + c_{12,i}z^{-1}}{1 + d_{12,i}z^{-1} + d_{22,i}z^{-2}}, \quad (3.42)$$

pri čemu su koeficijenti definisani kao

$$\begin{aligned} c_{0k,i} &= 2 \operatorname{re}\{u_{k,i}\} & d_{1k,i} &= -2 \operatorname{re}\{w_{k,i}\} \\ c_{1k,i} &= -\left(\operatorname{re}\{w_{k,i}\} + \frac{\operatorname{im}\{u_{k,i}\} \operatorname{im}\{w_{k,i}\}}{\operatorname{re}\{u_{k,i}\}} \right) & d_{2k,i} &= |w_{k,i}|^2 \end{aligned} \quad k = 1, 2.$$

Ostaci u polovima $\{w_{k,i}, w_{k,i}^*\}$ su označeni sa $\{u_{k,i}^*, u_{k,i}\}$. Realizacija (3.42) je data na Sl. 3.5b.

Očigledno je da ova procedura zahteva dosta proračuna. Da bi se olakšao posao projektovanja IIR filtera na osnovu CMAC prototipa razvijena je jedna softverska rutina u vidu MATLAB skripta. Pored projektovanja, rutina nudi mogućnost kreiranja VHDL (*Very high speed integrated circuits Hardware Description Language*) kôda, odgovarajućeg testbenča, komandnog fajla logičkog simulatora (.do fajl) i simulaciju pobuđivanjem pre-

instaliranog simulatora u paketskom (*batch*) ili grafičkom (*gui*) režimu rada. VHDL opis praktično predstavlja RTL model filtra koji se može koristiti za funkcionalnu verifikaciju, a u kasnijoj fazi, i za logičku sintezu filtra korišćenjem, ASIC ili FPGA, RTL kompajlera. Više detalja o ovom softverskom rešenju je dato u prilogu.

Na ovaj način omogućeno je projektovanje digitalnih IIR filtara upotrebom četiri klase CMAC *all-pole* filtarskih funkcija za red prototipa, $n=[3:10]$.

4 Uticaj konačne preciznosti predstavljanja koeficijenata filtra

U digitalnom domenu moguće je predstaviti samo konačan broj vrednosti. Dakle, u praktičnoj realizaciji moguće je predstaviti vrednosti koeficijenata samo sa ograničenim brojem značajnih cifara. Broj značajnih cifara je određen dužinom, tj. brojem bitova N , izabrane digitalne reči za dati sistem. Nemogućnost predstavljanja koeficijenata filtra u punoj preciznosti uzrokuje distorziju u karakteristikama filtra. Pored degradacije fazne i karakteristike slabljenja, može biti ugrožena i stabilnost usled migracije vrednosti polova.

Za procenu promene položaja polova usled procesa kvantizacije koeficijenata može se upotrebiti Rušeova (*Rouché*) teorema [Géhe71, Kran99]. Neka je polinom imenioca prenosne funkcije filtra definisan kao

$$D_n(z) = \sum_i^n d_i z^i, \quad \text{sa } d_n = 1. \quad (4.1)$$

Na osnovu Rušeove teoreme u [Géhe71] je pokazano da za gornju granicu vrednosti korena polinoma D_n važi

$$\bar{R} = \max(1, \alpha), \quad (4.2)$$

gde je \bar{R} radijus kruga koji obuhvata sve korene polinoma D_n u kompleksnoj ravni, a α

$$\alpha = \frac{1}{|d_n|} \sum_{i=0}^{n-1} |d_i| = \sum_{i=0}^{n-1} |d_i|. \quad (4.3)$$

Ukoliko se pretpostavi da se svaki i -ti koeficijent (osim d_n koji je fiksiran) uveća za neko Δd_i onda α postaje

$$\alpha(\mathbf{d} + \Delta \mathbf{d}) = \sum_{i=0}^{n-1} |d_i + \Delta d_i|, \quad (4.4)$$

gde su \mathbf{d} i $\Delta \mathbf{d}$ vektori koeficijenata i priraštaja, respektivno. Generalno, Δd_i može biti pozitivno ili negativno. Negativni priraštaj u kombinaciji sa pozitivnim koeficijentom predstavlja odsecanje (*truncation*), a negativni ili pozitivni priraštaj u kombinaciji sa negativnim koeficijentom predstavlja zaokruživanje (*rounding*). Pod pretpostavkom da su svi koeficijenti različiti od nule, konačna razlika $\Delta \alpha$ se može izraziti kao,

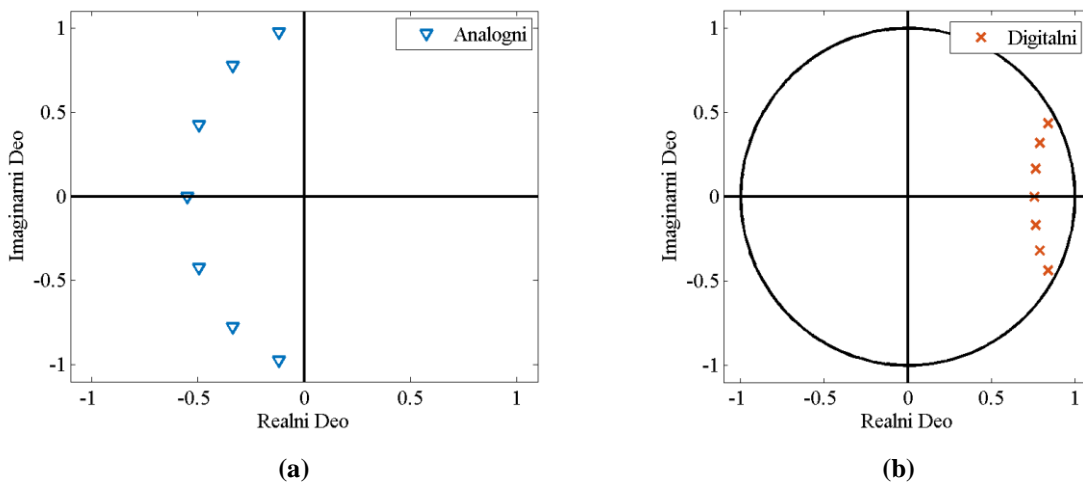
$$\Delta\alpha = \alpha(\mathbf{d} + \Delta\mathbf{d}) - \alpha(\mathbf{d}) = \sum_{i=0}^{n-1} |d_i| \left(\left| 1 + \frac{\Delta d_i}{d_i} \right| - 1 \right). \quad (4.5)$$

U najgorem slučaju ($\Delta d_i/d_i > 0 \forall i=0,1,\dots,n-1$), pa je

$$\Delta\alpha_{\max} = \sum_{i=0}^{n-1} |\Delta d_i|. \quad (4.6)$$

Može se zaključiti da se \bar{R} uvećava dvostruko. Prvenstveno zbog postojanja samih priraštaja Δd_i , a zatim i zbog njihovog broja. Drugim rečima, što je veći red filtra veći je i radijus \bar{R} .

Uticao predstavljanja koeficijenata filtra sa redukovanim brojem značajnih cifara ispitan je na primeru LSM, *all-poles*, analognog prototipa sedmog reda ($n=7$). Položaj polova analognog prototipa i odgovarajuće digitalne realizacije dobijene uz pomoć bilinearne transformacije gde je za periodu uzorkovanja izabrano $T=0.5$ prikazan je na Sl. 4.1.



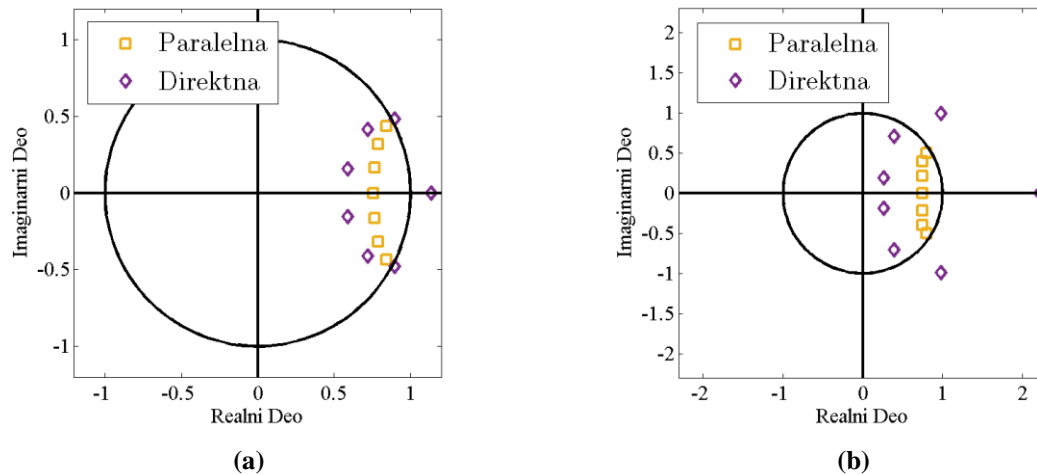
Slika 4.1 Položaj polova LSM filtra, reda $n=7$, (a) analogni filter i (b) digitalni filter gde su koeficijenti dati u punoj preciznosti.

U Tabeli 4.1 su date vrednosti polova sa Sl. 4.1. Položaj polova digitalnog filtra kada su koeficijenti dati u punoj preciznosti nadalje će biti tretiran kao tačan položaj polova.

Tabela 4.1 Tačan položaj polova LSM filtra, reda $n=7$ u s i z domenu za $T=0.5$.

s-domain		z-domain	
Re	Im	Re	Im
-0.5510897460	0.0	0.7578207520	0.0
-0.1179475625	± 0.9751626241	0.8395567140	± 0.4356216113
-0.3342221750	± 0.7735798237	0.7887919250	± 0.3192668225
-0.4935853895	± 0.4252967357	0.7645095574	± 0.1670025358

Ukoliko se koeficijenti filtra predstave sa četiri značajne cifre direktna realizacija (Sl. 2.2) proizvodi sistem koji je nestabilan, dok je kod paralelne (Sl. 2.4) stabilnost i dalje očuvana. Ovaj slučaj je prikazan na Sl. 4.2a.



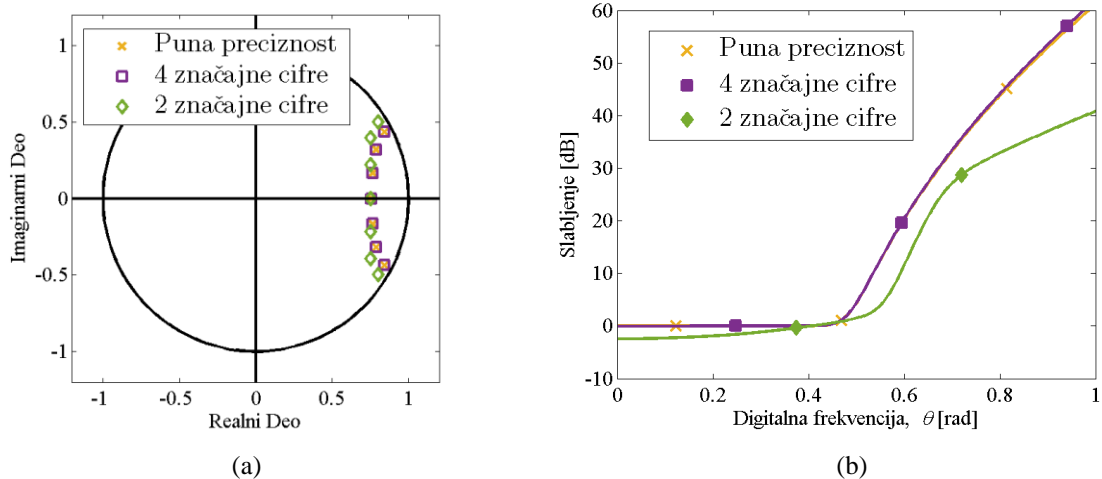
Slika 4.2 Položaj polova digitalnog LSM filtra, reda $n=7$ za dve različite realizacije gde su koeficijenti filtra predstavljeni sa četiri (a) i dve (b) značajne cifre.

Ovaj trend je još više izražen u slučaju kada se broj značajnih cifara za prezentaciju koeficijenata smanji na dva (Sl. 4.2b). Dakle, jasno je da paralelna realizacija daje robustniji sistem. U tabeli 4.2 su date vrednosti polova sa Sl. 4.2.

Tabela 4.2 Položaj polova LSM filtra, reda $n=7$, sa redukovanom preciznošću koeficijenata

Prezentacija koeficijenata filtra	Direktna realizacija (Sl. 4.2a)		Paralelna realizacija (Sl. 4.2b)	
	Re	Im	Re	Im
četiri značajne cifre	1.1360207906	0	0.7578000000	0
	0.8939351215	± 0.4816186107	0.8395000000	± 0.4357060362
	0.7208726759	± 0.4140006562	0.7885000000	± 0.3199496054
	0.5886818073	± 0.1575365844	0.7645000000	± 0.1668524798
dve značajne cifre	2.2161297265	0	0.7500000000	0
	0.9837501045	± 0.9948786645	0.8000000000	± 0.5000000000
	0.3926899052	± 0.7054768233	0.7500000000	± 0.3968626967
	0.2654951271	± 0.1886575446	0.7500000000	± 0.2179449472

Prethodna tri slučaja prezentacije koeficijenata za paralelnu realizaciju su sumirana na Sl. 4.3 gde je dat položaj polova i karakteristika slabljenja.



Slika 4.3 Položaj polova (a) i slabljenje (b) za LSM filter, reda $n=7$, za tri varijante prezentacije koeficijenata: puna preciznost, četiri i dve značajne cifre.

Može se uočiti da je u oba slučaja prezentovanja koeficijenata sa redukovanim brojem značajnih cifara stabilnost očuvana (Sl. 4.3a). Isto se može zaključiti poređenjem zadnje dve kolone tabela 4.1 i 4.2. Tek kod prezentacije koeficijenata sa dve značajne cifre dolazi do odstupanja u karakteristici slabljenja (Sl. 4.3b).

Validnost procene date sa (4.6) se može potvrditi ukoliko se npr. posmatra migracija minimalnog radijusa diska u kompleksnoj ravni koji obuhvata sve polove za slučaj kada su koeficijenti prezentovani sa dve značajne cifre u odnosu na slučaj kada su koeficijenti dati u punoj preciznosti. U slučaju prezentacije koeficijenata sa dve značajne cifre zapravo minimalni radijus diska odgovara vrednosti realnog pola koja iznosi $\bar{R}_2=2.2161297265$. Kod pune preciznosti (tačan položaj polova) ovaj radijus odgovara modulu pola čiji je moduo maksimalan. U ovom slučaju to je par polova sa maksimalnim realnim delom tj. vrednost radijusa je $\bar{R}_0=0.9458444186$. Dakle, promena radijusa iznosi $\Delta\bar{R}=|\bar{R}_0 - \bar{R}_2| \approx 1.2703$. Na osnovu (4.6) dobija se $\Delta\alpha_{\max} \approx 1.9563$. Ako se uzme u obzir da je (4.6) izvedena pod uslovom najgoreg slučaja, može se zaključiti da je predikcija poprilično dobra. U Tabeli 4.3 su dati koeficijenti imenioca prenosne funkcije digitalnog, LSM filtra, reda $n=7$, u punoj preciznosti i prezentaciji sa dve značajne cifre. U zadnjoj koloni dati su priraštaji polova na osnovu kojih se došlo do procene za $\Delta\alpha_{\max}$.

Tabela 4.3 Koeficijenti imenioca digitalnog, LSM filtra, reda $n=7$

	Puna preciznost	Dve značajne cifre	$ \Delta d_i $
d_7	1	1.0	0
d_6	-5.5435371447	-5.5	0.0435371447
d_5	13.4863201902	13	0.4863201903
d_4	-18.6187031540	-18	0.6187031541
d_3	15.7256233042	15	0.7256233043
d_2	-8.1149764659	-8.1	0.0149764659
d_1	2.3665384182	2.3	0.0665384182
d_0	-0.3006277069	0.3	0.0006277070

4.1 Kvantovanje koeficijenata

Uticaj kvantovanja koeficijenta na karakteristike filtra se najbolje može demonstrirati uz pomoć primera. Za primer je izabran filter sa sledećim karakteristikama:

- Aproksimacija amplitudske karakteristike u propusnom opsegu: LSM
- Tip filtra: BP
- Red LP prototipa: 7
- Centralna frekvencija: 3kHz
- Propusni opseg: 900Hz
- Frekvencija odmeravanja: 50kHz

Za LP \rightarrow BP transliranje upotrebljena je poznata relacija definisana u [OrTe68],

$$\omega \leftarrow \frac{1}{BW_r} \cdot \frac{\Omega_0^2 - \Omega^2}{\Omega \cdot \Omega_0}, \quad (4.7)$$

gde je ω kružna frekvencija LP prototipa, Ω kružna frekvencija BP filtra, Ω_0 centralna kružna frekvencija BP filtra, BW_r relativni propusni opseg definisan kao $BW_r = BW/\Omega_0$ i BW propusni opseg izražen u rad/sec. Nakon uvođenja kompleksnih promenljivih, $s_{LP} = j\omega$ i $s_{BP} = j\Omega$, (4.7) postaje algebarska jednačina drugog reda sa kompleksnim koeficijentima koja se obično rešava primenom Gefeovog (*Geffe*) algoritma [Geff74]. Nova prenosna funkcija ima četrnaest polova dobijenih rešavanjem (4.7). Vrednosti polova LP prototipa i rezultujućeg BP filtra su date u Tabeli 4.4.

Naravno, vrednosti BP polova treba denormalizovati množenjem sa $2\pi f_0$. Granice propusnog opsega, Ω_L i Ω_H , se mogu odrediti rešavanjem sistema jednačina,

$$\begin{aligned} f_0^2 &= f_H f_L \\ f_{BW} &= f_H - f_L, \end{aligned} \quad (4.8)$$

gde je $f_{BW} = BW/2\pi$.

Tabela 4.4 Vrednosti normalizovanih polova BP i LP LSM filtra u s-domenu

Br.	BP		LP	
	Realni deo	Imaginarni deo	Realni deo	Imaginarni deo
1/2	-0.08266346190	±0.99657751935	-0.1179475625	±0.9751626241
3/4	-0.02025317565	±1.15676424786	-0.3342221750	±0.7735798237
5/6	-0.01513109310	±0.86421546064	-0.4935853895	±0.4252967357
7/8	-0.05591895577	±1.12151432405	-0.5510897460	0.0
9/10	-0.04434769671	±0.88944037693		
11/12	-0.07876429908	±1.06309950994		
13/14	-0.06931131778	±0.93551048922		

Na osnovu (4.8) za konkretni primer BP filtra dobija se $\{f_L, f_H\} = \{2.5836\text{kHz}, 3.4836\text{kHz}\}$. Nakon proračuna pojedinih sekcija pomoću (3.35) i (3.36), do digitalnog IIR filtra se dolazi primenom (3.37) i (3.38) (bilinearna transformacija).

Sledeći korak podrazumeva kvantovanje koeficijenata tj. njihovo predstavljanje u brojnomo sistemu sa fiksnim zarezom. Usvojena je sledeća notacija, $Q[N F]$, gde N predstavlja ukupnu dužinu reči u bitovima, a F broj bitova za predstavljanje brojeva iza decimalne tačke. Dakle, $Q[N F]$ format pokriva opseg,

$$RNG = \left[-\frac{2^{N-1}}{2^F}, \frac{2^{N-1}-1}{2^F} \right], \quad (4.9)$$

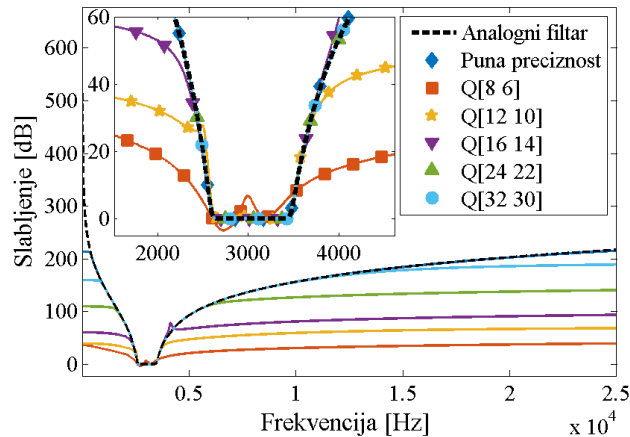
u koracima $INC = 2^{-F}$, gde je F definisano kao, $F = N - I$. Vrednost parametra I se određuje na osnovu koeficijenta sekcije filtra koji ima maksimalnu, apsolutnu, vrednost

$$I = \left\lceil \log_2 \left(\left[\max \left\{ |c_{k,i}|, |d_{l,i}| \right\} \right] \right) \right\rceil + 1, \quad i = 1, \dots, N_S, \quad k = 0, 1, 2, \quad l = 1, 2, \quad (4.10)$$

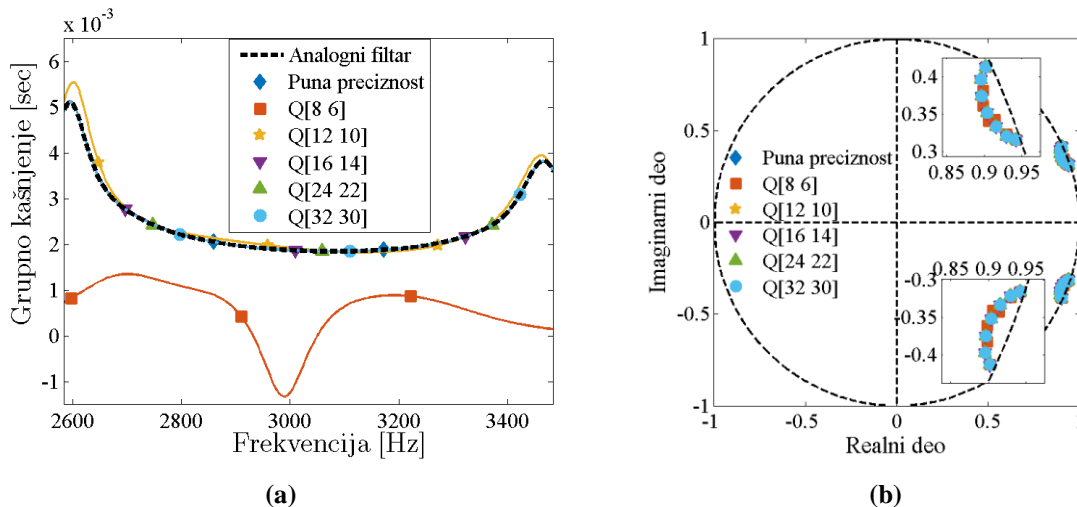
gde je N_S broj sekcija. Jasno je da I određuje i broj bitova za reprezentaciju celobrojnog dela koeficijenta. Dakle, za pravilno kvantovanje koeficijenata potrebne su dve operacije. Prvo treba odrediti parametar I korišćenjem (4.10) koji određuje broj bitova za celobrojni (a ujedno i decimalni) deo koeficijenta, a zatim zaokružiti vrednost koeficijenta na najbližu iz opsega RNG . Da bi se smanjio efekat zasićenja, za prezentaciju koeficijanata koristi se isključivo drugi komplement.

Na Sl. 4.4 prikazan je uticaj zaokruživanja koeficijenata na karakteristiku slabljenja u propusnom i nepropusnom opsegu. Puna preciznost odgovara tridesetdvo-bitnoj prezentaciji u formatu pokretnog zareza (*floating-point*). Može se uočiti da smanjenje broja bitova unosi značajno izobličenje u karakteristiku slabljenja. Tako na primer, dužina reči od osam bitova je neprihvatljiva po pitanju slabljenja kako u propusnom tako i nepropusnom opsegu. Reprezentacija sa dvanaest bitova solidno aproksimira slabljenje u propusnom opsegu, međutim unosi značajno izobličenje u nepropusni opseg. Šesnaestobitna reprezentacija može

biti prihvatljiva ukoliko se ne zahteva slabljenje veće od 60dB van propusnog opsega. Konačno, reprezentacije sa dvadeset četiri i trideset i dva bita omogućuju slabljenje van propusnog opsega veće od 100dB. Dakle, povećavanjem broja bitova za reprezentaciju koeficijenata uticaj kvantovanja koeficijenata se redukuje. Ovaj uticaj je značajno manji na grupno kašnjenje što se može videti na Sl. 4.5a.



Slika 4.4 Slabljenje BP filtra van (donji deo slike) i unutar (gornji deo slike) propusnog opsega za $N=\{8, 12, 16, 24, 32\}$ dobijeno bilinearnom transformacijom.



Slika 4.5 Grupno kašnjenje u propusnom opsegu (a) i položaj polova (b) BP filtra za $N=\{8, 12, 16, 24, 32\}$ dobijeno bilinearnom transformacijom.

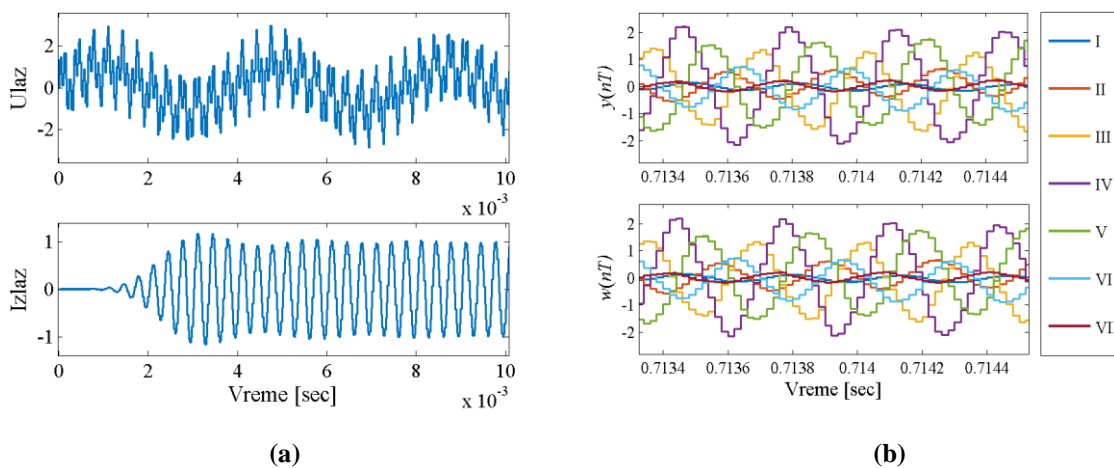
Može se uočiti da je grupno kašnjenje u okolini centralne frekvencije (3kHz) približno 2ms. Izuzetak je naravno kvantovanje sa malim brojem bitova ($N=8$). Na Sl. 4.5b je dat prikaz položaja polova u z -ravni za razne formate kvantovanja. Može se videti da je stabilnost očuvana za sve izabrane vrednosti N .

Na Sl. 4.6a je dat vremenski odziv filtra za $Q[16 14]$ na ulazni, testni, signal oblika,

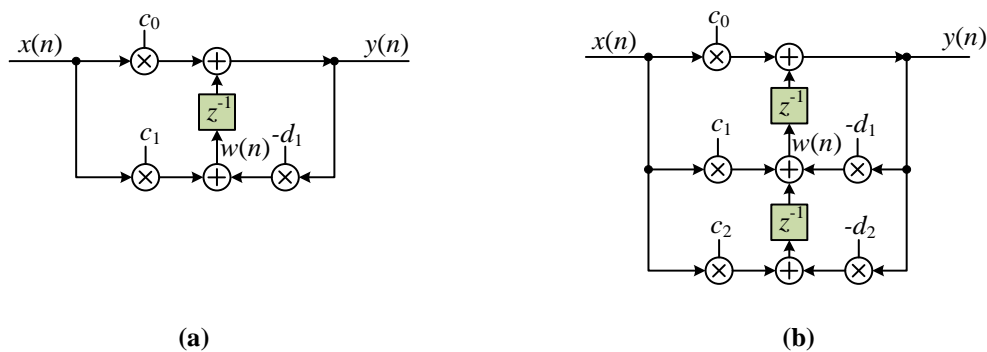
$$a(t) = \sin(2\pi \cdot f_0 t) + \sin(2\pi \cdot f_1 t) + \sin(2\pi \cdot f_2 t), \quad (4.11)$$

gde su $f_1=257.873\text{Hz}$, $f_2= 9.3315\text{kHz}$. Posmatranjem Sl. 4.6a može se uočiti da izlaz filtera ispoljava blago oscilatorno ponašanje. Na Sl. 4.6b je dat odziv u kritičnim tačkama pojedinih sekcija filtera. Ove tačke su označene na Sl. 4.7 za sekciju prvog i drugog reda. Jedna od tačaka je izlaz, y , dok je druga izlaz dvostrukog sabirača, w .

Za izabrani format, Q[16 14], moguće je predstaviti vrednosti iz opsega $RNG=[-2.0:1.99993896484375]$. Očigledno je da vrednosti ulaza i kritičnih tačaka (y i w) IV sekcije izlaze iz ovog opsega. To ukazuje da će u hardverskoj realizaciji biti potrebno skaliranje. U ovom slučaju, gde je ulaz unapred poznat, moguće je jednostavno odrediti faktor skaliranja, koji može biti određen kao $\max\{|y_i(n)|, |w_i(n)|, |x(n)|\}$, $i=1, \dots, N_s$. Radi lakše hardverske realizacije, faktor skaliranja se obično bira da bude stepen broja dva. Na ovaj način skaliranje se svodi na jednostavnu operaciju pomeranja (*shift*). U ovom slučaju dovoljno je podeliti ulaz fiksnim faktorom 2.



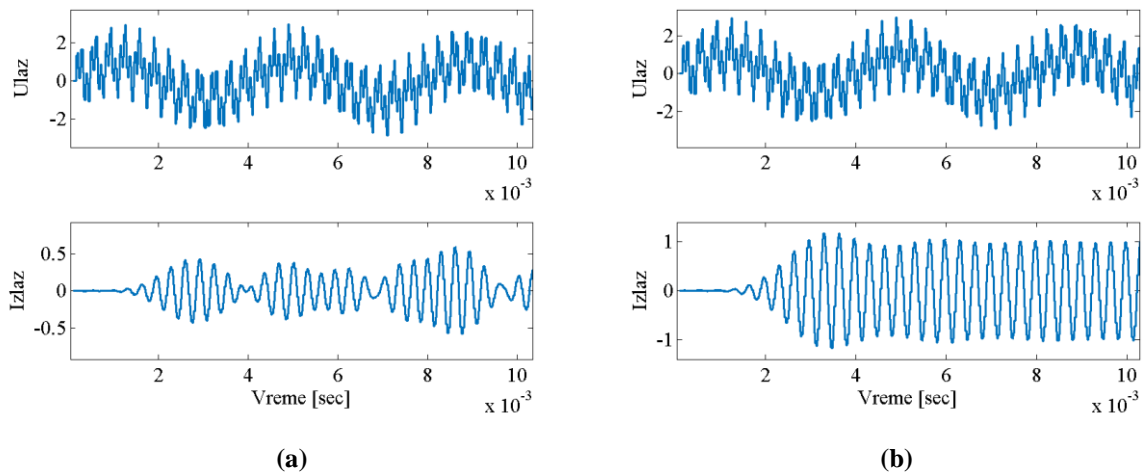
Slika 4.6 Vremenski odziv LSM BP filtera dobijenog bilinearnom transformacijom: (a) ulaz i izlaz, (b) ukupni izlaz ($y(nT)$) i kritični čvorovi ($w(nT)$) pojedinih sekcija). Koefficienti su predstavljeni u formatu Q[16 14].



Slika 4.7 Kritične tačke po pitanju potencijalnog prekoračenja vrednosti u sekciji prvog (a) i drugog (b) reda.

Naravno, nakon filtriranja izlaz treba ponovo skalirati inverznim faktorom tj. pomnožiti sa 2.

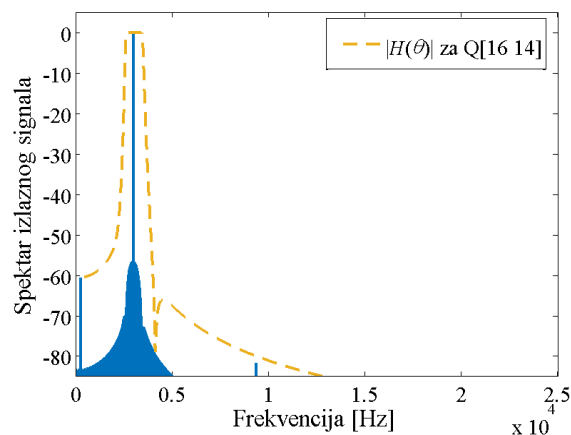
Odziv RTL modela filtra sa i bez skaliranja je dat na Sl. 4.8. Sa Sl. 4.8a se može viditi da filter ne daje validan izlaz ukoliko ulaz nije skaliran. Pošto je korišćen drugi komplement, zasićenje izlaza je izbegnuto, međutim oscilatorno ponašanje je evidentno (*limit-cycle*). Ukoliko se ulaz skalira, RTL model filtra daje isti odziv kao i matematički model (Sl. 4.6a). Ovaj slučaj je prikazan na Sl. 4.8b.



Slika 4.8 Vremenski odziv RTL modela LSM BP filtra dobijenog bilinearnom transformacijom, bez skaliranja (a) i sa skaliranjem (b). Koeficijenti filtra su predstavljeni u formatu Q[16 14].

Najzad, na Sl. 4.9 je dat spektar izlaznog signala dobijen simulacijom RTL modela. U spektar izlaznog signala je ucrtana i amplitudska karakteristika filtra čiji su koeficijenti predstavljeni formatom Q[16 14]. Zanimljivo je primetiti da se u prenosnoj funkciji javlja nula u okolini 4kHz.

Generalno, kvantovanje koeficijenata može izazvati različite efekte na karakteristike filtra koje treba pažljivo analizirati pre konačne realizacije.



Slika 4.9 Spektar izlaznog (filtriranog) signala RTL model LSM BP filtra dobijenog bilinearnom transformacijom.

Date su tri reprezentacije: puna preciznost, decimalna i heksadecimalna po formatu Q[16 14]. Koeficijenti brojioca i imenioca projektovanog BP filtra korišćenjem bilinearne transformacije su dati u Tabeli 4.5 i 4.6, respektivno.

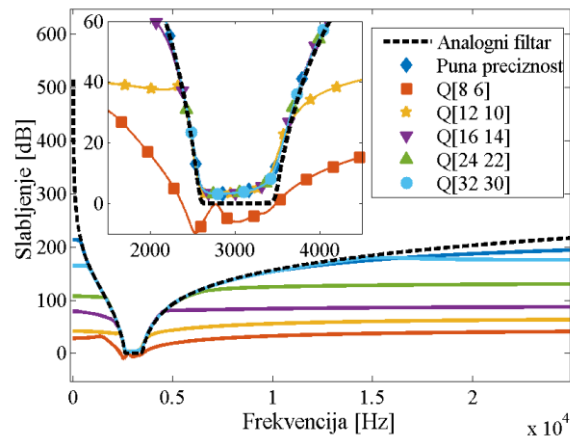
Tabela 4.5 Koeficijenti brojioca sekcija LSM BP filtra dobijenog bilinearnom transformacijom

		Brojilac		
		c_0	c_1	c_2
Puna preciznost	sekcija I	+0.0033086494281706663	-0.0018617614854014842	-0.0051704109135721505
	sekcija II	+0.0067062285319410561	+0.0085906290469084413	+0.0018844005149673854
	sekcija III	-0.044039453119340377	-0.0120564809558326	+0.031982972163507775
	sekcija IV	+0.06277875243767074	0	-0.062778752453249653
	sekcija V	-0.029334088252293972	+0.015384970101239978	+0.044719058353533951
	sekcija VI	-0.006864819539353288	-0.013389525398100878	-0.00652470585874759
	sekcija VII	+0.0074447307119548771	+0.0032630379244648162	-0.0041816927874900617
Q[16 14] decimalna	I	+0.00329589843750	-0.00189208984375	-0.00518798828125
	II	+0.00671386718750	+0.00860595703125	+0.00189208984375
	III	-0.04406738281250	-0.01208496093750	+0.03198242187500
	IV	+0.06280517578125	+0.00000000000000	-0.06280517578125
	V	-0.02935791015625	+0.01538085937500	+0.04473876953125
	VI	-0.00683593750000	-0.01336669921875	-0.00653076171875
	VII	+0.00744628906250	+0.00323486328125	-0.00421142578125
Q[16 14] heksadecimalna	I	0036	FFE1	FFAB
	II	006E	008D	001F
	III	FD2E	FF3A	020C
	IV	0405	0000	FBFB
	V	FE1F	00FC	02DD
	VI	FF90	FF25	FF95
	VII	007A	0035	FFBB

Tabela 4.6 Koeficijenti imenioca sekcija LSM BP filtra dobijenog bilinearnom transformacijom

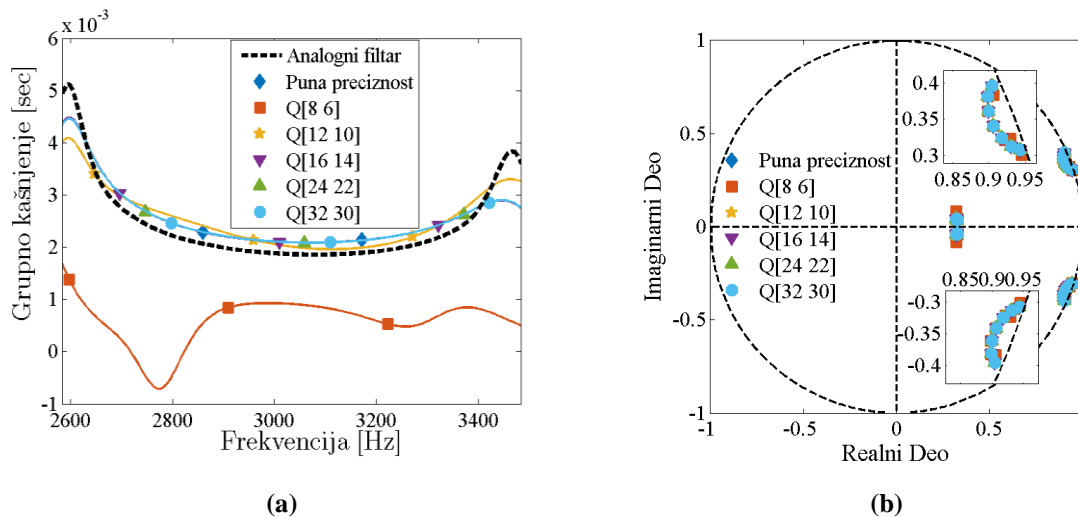
		Imenilac	
		d_1	d_2
Puna preciznost	sekcija I	-1.886085860514888	+0.98894784642499967
	sekcija II	-1.8601293281281488	+0.96799935587139696
	sekcija III	-1.8323004517946606	+0.95057717438073264
	sekcija IV	-1.8083338880121447	+0.94157013740084117
	sekcija V	-1.7935719318070145	+0.94450186279953363
	sekcija VI	-1.7923157567661698	+0.96044412883386077
	sekcija VII	-1.8052459566148433	+0.98552821289667847
Q[16 14] decimalna	I	-1.88610839843750	+0.98895263671875
	II	-1.86010742187500	+0.96801757812500
	III	-1.83227539062500	+0.95056152343750
	IV	-1.80834960937500	+0.94158935546875
	V	-1.79357910156250	+0.94451904296875
	VI	-1.79229736328125	+0.96044921875000
	VII	-1.80523681640625	+0.98553466796875
Q[16 14] heksadecimalna	I	874A	3F4B
	II	88F4	3DF4
	III	8ABC	3CD6
	IV	8C44	3C43
	V	8D36	3C73
	VI	8D4B	3D78
	VII	8C77	3F13

Ukoliko se za mapiranje u z -domen koristi transformacija drugog reda dobija se karakteristika slabljenja prikazana na Sl. 4.10.



Slika 4.10 Slabljenje LSM BP filtra van (donji deo slike) i unutar (gornji deo slike) propusnog opsega za $N=\{8, 12, 16, 24, 32\}$ dobijeno transformacijom drugog reda

Može se uočiti da transformacija drugog reda nudi bolju selektivnost i slabljenje u nepropusnom opsegu za istu dužinu reči, $N=16$, nego bilinearna transformacija. Sa druge strane aproksimacija slabljenja u propusnom opsegu je lošija. Na centralnoj frekvenciji, f_0 , prisutno je dodatno slabljenje od $|\Delta a| \approx 3.4\text{dB}$. Slično važi i za grupno kašnjenje u propusnom opsegu koje je dato na Sl. 4.11a.



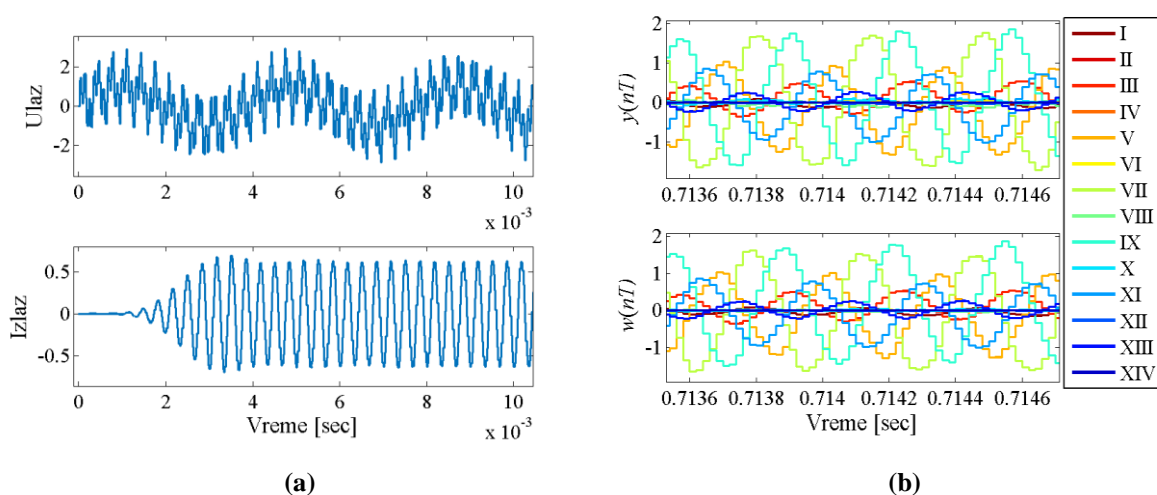
Slika 4.11 Grupno kašnjenje u propusnom opsegu (a) i položaj polova (b) LSM BP filtra za $N=\{8, 12, 16, 24, 32\}$ dobijeno transformacijom drugog reda.

Apsolutna razlika grupnog kašnjenja u odnosu na analogni prototip je $|\Delta \tau_g| = 223\mu\text{s}$. Aproksimacija slabljenja i grupnog kašnjenja u propusnom opsegu može se poboljšati izborom veće frekvencije odmeravanja. Tako na primer udvostručavanjem frekvencije odmeravanja, $f_s=100\text{kHz}$, dodatno slabljenje pada na $|\Delta a| \approx 0.58\text{dB}$, a razlika u grupnom

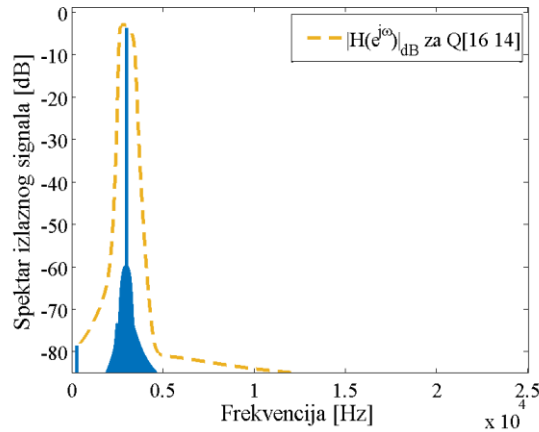
kašnjenju na $|\Delta\tau_g|=82\mu\text{S}$. Nažalost ovo će takođe za najmanje dvostruko uvećati i dinamičku potrošnju kola. Može se zaključiti da je transformacija drugog reda osjetljivija na kvantovanje koeficijenata i da zahteva veću frekvenciju odmeravanja u odnosu na bilinearnu transformaciju. Razlog tome leži u činjenici da transformacija drugog reda proizvodi digitalni filter dvostruko većeg reda, a samim tim i skoro dvostruko većim hardverom nego što je to slučaj kada se koristi bilinearna transformacija. Međutim, transformacija drugog reda daje bolje slabljenje van propusnog opsega što ukazuje da se potencijalno može izabrati niži red analognog prototipa i time redukuje količina hardvera (potrošnja, cena). Ovome treba dodati i činjenicu da primer koji posmatramo predstavlja širokopojasni filter ($BW/f_0=0.3$) što znači da se izobličenja delimično unose već pri transliranju u analognom domenu.

Položaj polova je dat na Sl. 4.11b. Na osnovu položaja polova može se zaključiti da je stabilnost očuvana. Pošto se radi o transformaciji drugog reda dobijena su dva skupa od po četrnaest polova. Prvi skup se nalazi na istom mestu kao i u slučaju kada se koristi bilinearna transformacija, dok je drugi skup grupisan unutar jediničnog kruga.

Na Sl. 4.12 prikazan je vremenski odziv filtra za koeficijente predstavljene u formatu Q[16 14]. Pošto filter unosi slabljenje, interne tačke pojedinih sekcija ostaju u opsegu definisanom formatom Q[16 14]. Međutim, vrednosti ulaza izlaze iz opsega pa je potrebno koristiti faktor skaliranja 2. Vremenski odziv RTL modela je dat na Sl. 4.12a za testni signal definisan u (4.11). Spektar izlaznog signala RTL modela filtra dobijenog transformacijom drugog reda prikazan je na Sl. 4.13. Koeficijenti brojioca i imenioca pojedinih sekcija su dati u Tabeli 4.7.

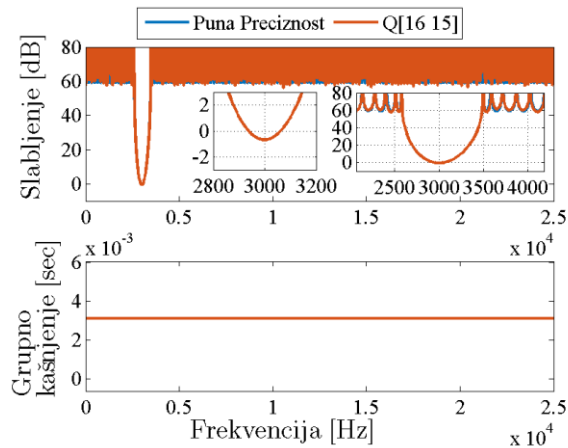


Slika 4.12 Vremenski odziv BP filtra dobijenog transformacijom drugog reda: (a) ulaz i izlaz, (b) izlaz, $y(nT)$, i kritični čvor, $w(nT)$ pojedinih sekcija. Koeficijenti su predstavljene u formatu Q[16 14].



Slika 4.13 Spektar izlaznog (filtriranog) signala RTL model BP filtra dobijenog transformacijom drugog reda

Poređenja radi, kada bi se za date zahteve koristio FIR filter dobijen Parks-Mecklilan (Parks-McClellan) optimizacionim algoritmom, [PaMc72], bio bi potreban filter reda $n=311!$ Na Sl. 4.14 je prikazano slabljenje i grupno kašnjenje FIR filtra 311. reda, II tipa, koji obezbeđuje minimalno slabljenje u nepropusnom opsegu od 60dB i maksimalnu talasnost u propusnom opsegu od 1dB. Sa slike se može uočiti da je FIR filter manje osetljiv na kvantizaciju koeficijenta. Takođe, pošto je obezbeđena simetrija koeficijenta dobijeno je konstantno grupno kašnjenje tj. linearnost faze je zagarantovana. Međutim, red filtra je veoma veliki što bi u hardverskoj realizaciji rezultovalo izuzetno visokom latencijom, potrošnjom i površinom. Ovde zapravo leži glavni motiv za ispitivanje alternativnih IIR realizacija.



Slika 4.14 Slabljenje i grupno kašnjenje BP FIR filtra, II tipa, dobijenog Parks-McClellan optimizacionim algoritmom sa maksimalnim slabljenjem u nepropusnom opsegu od 60dB i maksimalnom talasnošću u propusnom opsegu od 1dB.

Tabela 4.7 Koeficijenti sekcija LSM BP filtra dobijenog transformacijom drugog reda

	sekcija	Brojilac		Imenilac	
		c_0	c_1	d_1	d_2
Puna preciznost	I	0.0077190204842510664	-0.010699383358084754	-1.8872099552974775	0.98524078698326789
	II	-0.0038066700309112018	0.0011610817721585168	-0.65053803937547972	0.10691692041195887
	III	0.005558210634258768	0.0098851417306181476	-1.8628530588983545	0.96533891403832006
	IV	0.0050143123707962652	-0.0010990805328273177	-0.65152896204814248	0.1073313096476797
	V	-0.073171429576353489	0.048384553665106182	-1.8367344417783302	0.94851468803294747
	VI	0.013632657480313483	-0.0054800005416238921	-0.65139760336851948	0.10742810691477651
	VII	0.11816733781085864	-0.1114290613895637	-1.8142108960637189	0.93925782280139647
	VII	-0.037343335148773693	0.012703851810712761	-0.64983111464296373	0.10708330523583864
	IX	-0.066881686022103634	0.088659303947732596	-1.8000383337033519	0.94063182865849049
	X	0.032514911062720218	-0.010018050339672365	-0.64689771846417909	0.1062866116809822
	XI	-0.0026190920229319367	-0.019857185660973015	-1.7979248307670899	0.95354731406821069
	XII	-0.0090509997138257282	0.0021905194995660692	-0.64317888241102261	0.10518932646787348
	XII	0.011227664372711008	-0.0050563626951203771	-1.808569828995628	0.97488687393793283
	XIV	-0.00096090049635530798	0.00053998408089546055	-0.63974608211543948	0.10411108228221963
Q[16 14]	I	0.00769042968750	-0.01068115234375	-1.88720703125000	0.98522949218750
	II	-0.00378417968750	0.00115966796875	-0.65051269531250	0.10693359375000
	III	0.00555419921875	0.00988769531250	-1.86285400390625	0.96533203125000
	IV	0.00500488281250	-0.00109863281250	-0.65155029296875	0.10736083984375
	V	-0.07318115234375	0.04840087890625	-1.83673095703125	0.94848632812500
	VI	0.01361083984375	-0.00549316406250	-0.65136718750000	0.10742187500000
	VII	0.11816406250000	-0.11145019531250	-1.81420898437500	0.93927001953125
	VII	-0.03735351562500	0.01269531250000	-0.64984130859375	0.10705566406250
	IX	-0.06689453125000	0.08868408203125	-1.80004882812500	0.94061279296875
	X	0.03253173828125	-0.01000976562500	-0.64691162109375	0.10626220703125
	XI	-0.00262451171875	-0.01983642578125	-1.79791259765625	0.95355224609375
	XII	-0.00903320312500	0.00219726562500	-0.64318847656250	0.10516357421875
	XII	0.01123046875000	-0.00506591796875	-1.80859375000000	0.97491455078125
	XIV	-0.00097656250000	0.00054931640625	-0.63977050781250	0.10412597656250
heksadecimalna	I	007E	FF51	8738	3F0E
	II	FFC2	0013	D65E	06D8
	III	005B	00A2	88C7	3DC8
	IV	0052	FFEE	D64D	06DF
	V	FB51	0319	8A73	3CB4
	VI	00DF	FFA6	D650	06E0
	VII	0790	F8DE	8BE4	3C1D
	VII	FD9C	00D0	D669	06DA
	IX	FBB8	05AD	8CCC	3C33
	X	0215	FF5C	D699	06CD
	XI	FFD5	FE8B	8CEF	3D07
	XII	FF6C	0024	D6D6	06BB
	XII	00B8	FFAD	8C40	3E65
	XIV	FFF0	0009	D70E	06AA

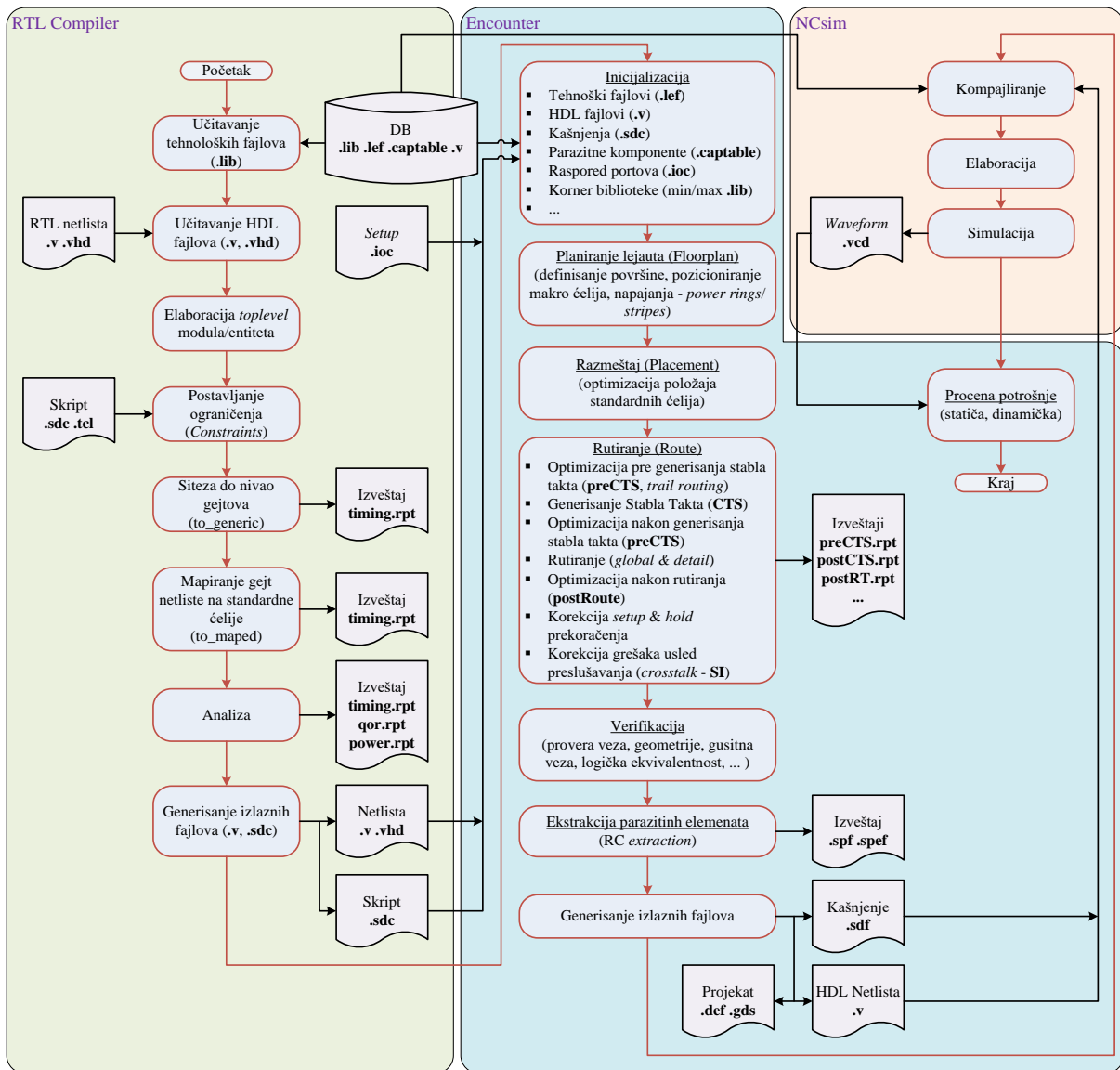
5 VLSI implementacija digitalnih filtara

Konačni cilj projektovanja jeste fizička realizacija filtra. Za fizičku implementaciju filtra izabran je 65nm CMOS proces proizvođača TSMC (*Taiwan Semiconductor Manufacturing Company*). Postupak implementacije savremenih, digitalnih, VLSI kola na bazi standardnih ćelija podrazumeva upotrebu EDA (*Electronic Design Automation*) i CAD (*Computer Aided Design*) softvera. U ovim softverskim alatima implementirani su napredni optimizacioni algoritmi uz pomoć kojih je moguće manipulirati velikim brojem logičkih gejtova i veza između njih. Za fizičko projektovanje filtara upotrebljeni su alati kompanije *Cadence Design Systems, Inc* (CDS) za digitalni *frontend* i *backend*. U ovom kontekstu *frontend* podrazumeva sintezu kola do nivoa gejtova i preslikavanje istih na biblioteku standardnih ćelija. *Backend* obuhvata sve akcije potrebne da se od sintetizovane netliste dođe do fizičke predstave kola, tj. lejauta. Tok fizičkog projektovanja (*flow*) je ilustrovan na Sl. 5.1. Tok se bazira na upotrebi dva osnovna CDS alata: *RTL Compiler*[®] za logičku sintezu (*frontend*) i *Encounter*[®] (*backend*) za implementaciju [Syst09a, Syst09b, Syst09c, Syst08c]. Automatizacija je obezbeđena kroz set Tcl (*Tool Command Language*) skriptova [Oust89].

Logička sinteza podrazumeva niz koraka. Najpre se učitavaju podaci standardnih ćelija vezani za potrošnju, kašnjenja, uslove rada itd.. Informacije o ćelijama su smeštene u odgovarajuće bibliotečke fajlove (.lib) koje obezbeđuje proizvođač. Osnovni preduslov za uspešnu sintezu jeste ispravno opisati funkcionalnost digitalnog kola u nekom od HDL jezika (VHDL, Verilog). HDL jezici pružaju veliki broj jezičkih konstrukcija ali je samo ograničeni skup sintetizabilan.

Dakle, HDL kôd mora biti prilagođen sintezi [Zwol04]. Nakon uspešnog kompajliranja odgovarajućih HDL fajlova i elaboracije modula/entiteta koji je najviši u hijerarhiji (*top level*) sledi zadavanje ograničenja pod kojima će biti izveden postupak sinteze do nivoa osnovnih kombinacionih i sekvencijalnih kola, tj. gejtova (*gate level*). Ograničenja (*Constraints*) se obično odnose na vremenski domen. Tako npr. mogu se zadati maksimalna vremena uspostavljanja (*setup*) i držanja (*hold*) flip-flopova (registara) u kolu. Ograničenja se zadaju u SDC (*Synopsys Delay Constraint*) formatu ili direktno komandama *RTL Compiler*[®]. Pod zadatim ograničenjima postupkom optimizacije se dolazi do predstave kola na nivou gejtova (još uvek nezavisne od tehnologije). Nakon toga sledi postupak preslikavanja na konkretne instance iz biblioteke standardnih ćelija koja je dostupna u formi baze podataka (DB). Postupak sinteze se završava štampanjem odgovarajuće Verilog (.v) netliste i novih

ograničenja (.sdc) koji predstavljaju ulazne parametar za postupak implementacije. Ova nova ograničenja sadrže detaljniji opis modela kašnjenja gejtova od onog zadatog od strane projektanta na početku procesa sinteze.



Slika 5.1 Tok fizičkog projektovanja digitalnih kola u CMOS procesu na bazi standardnih ćelija u Cadence Design Systemes, Inc okruženju.

Implementacija predstavlja niz optimizacionih postupaka na osnovu zadatih ograničenja po raznim kriterijumima (kašnjenje, potrošnja, površina itd.) koji imaju za cilj kreiranje optimalne fizičke reprezentacije kola. Postupak započinje inicijalizacijom tj. učitavanjem fajlova vezanih za raspored portova (.ioc), vremenska ograničenja (.lib), parazitne komponente (.captive) standardnih ćelija, itd.. Tri osnovna koraka su: Planiranje lejauta (floor plan), Razmeštaj (placement) i Povezivanje (routing). Nakon povezivanja sledi niz provera poput verifikacije geometrije, logičke ekvivalencije sa sintetizovanim kolom, potencijalne parazitne antene itd. Na ovom nivou moguće je dobiti realniji model kašnjenja

kola. Da bi se to ostvarilo, potrebno je ekstrahovati parazitne komponente (paraziti) iz geometrije leajauta. Paraziti se beleže u odgovarajućem SDF (*Standard Parasitic Format*) formatu koji se kasnije može koristiti za verifikaciju ispravnosti rada kola simulacijom. Ceo proces je praćen štampanjem različitih izveštaja (.rpt) pomoću kojih inženjer može pratiti da li je odgovarajući korak uspešno realizovan. Nakon uspešne implementacije projekt se eksportuje u standardnim formatima, poput GDS (*Graphics Database System*) ili DEF (*Design Exchange Format*) koji obezbeđuju portabilnost projekta kako između pojedinih alata tako i prema proizvođaču.

Dinamička potrošnja kola procenjena je na osnovu aktivnosti (*switching activity*) internih čvorova gejtova. Aktivnost jednog čvora predstavlja statistički parametar i definiše se kao broj promena logičkog stanja čvora po jednom ciklusu obrade normalizovan brojem ciklusa, pri čemu je broj ciklusa velika vrednost [Kaes08]. Na osnovu ekstrahovanih parazita proračunata su kašnjenja i kreirani odgovarajući SDF (*Standard Delay Format*) i Verilog fajlovi. Ovi fajlovi se dalje mogu koristiti za simulaciju na osnovu koje se dobija aktivnost internih čvorova (*switching activity*). U tu svrhu je korišćen NCsim simulator [Syst08a, Syst08b, Syst09d].

U cilju ispitivanja potencijalnih ušteda za analizu je izabran primer selektivnog BS (*notch*) filtra koji treba da ispuni sledeće zahteve: centralna frekvencija, $f_0 = 5\text{MHz}$ i nepropusni opseg 2MHz. Za frekvenciju odmeravanja izabrana je vrednost od $f_s = 61.44\text{MHz}$ koja se često koristi u MSR (*Multi-Standard Radio*) radio stanicama.

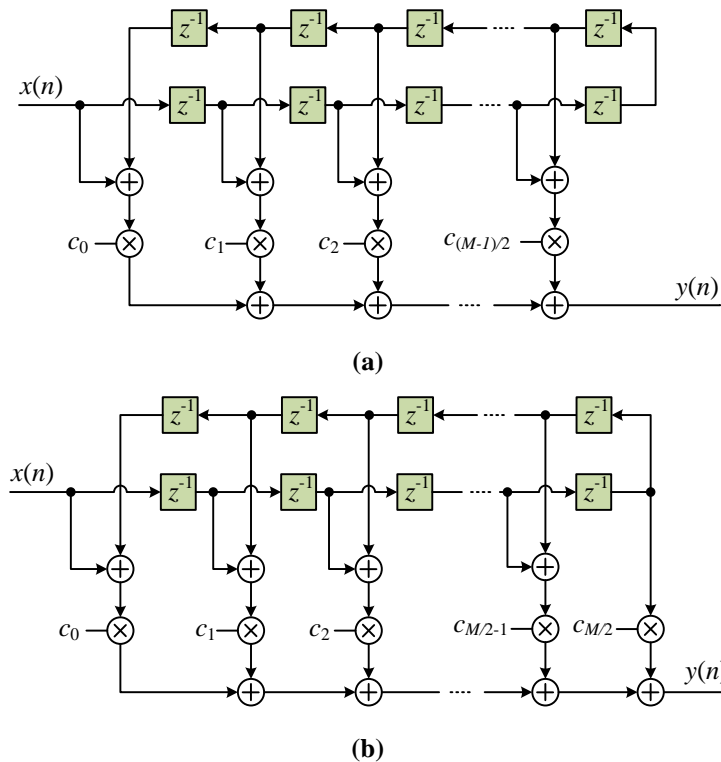
Ispitane su tri realizacije filtra: FIR, IIR dobijen bilinearnom transformacijom i IIR dobijen transformacijom drugog reda. Kada se koristi FIR realizacija filtra potreban je red 170 (dužine 171). Za sintezu filterske funkcije upotrebljen je ranije pomenuti Parks-McClellan optimizacioni algoritam. Da bi se obezbedila linearnost faze obezbeđena je simetrija koeficijenata tako da je dobijen filter I tipa. Kada su koeficijenti filtra simetrični moguće je minimizovati hardver upotrebom tzv. FDL (*Folded Delay Line*) arhitekture. FDL arhitektura za FIR filtre sa linearnom fazom je prikazana na Sl. 5.2.

Da bi se realizovale IIR varijante filtera potrebno je translirati LP prototip. Za $LP \rightarrow BS$ transliranje koristi se relacija,

$$\omega \leftarrow BW_r \frac{\Omega_0 \Omega}{\Omega_0^2 - \Omega^2}. \quad (5.1)$$

Dakle, BS filter je inverzni BP filter [OrTe68].

Za realizaciju IIR filtera korišćena je paralelna realizacija (Sl. 2.4b). Za IIR filter dobijen bilinearnom transformacijom korišćen je LSM prototip četvrtog reda. Vrednosti polova prototipa i normalizovane vrednosti polova dobijenog BS filtera su date u tabeli 5.1. Vrednosti BS filtera su normalizovane sa centralnom kružnom frekvencijom, $\Omega_0=2\pi f_0$. Naravno, pored polova, prenosna funkcija BS filtera ima i četiri para konjugovano-kompleksnih nula $\pm j\Omega_0$.



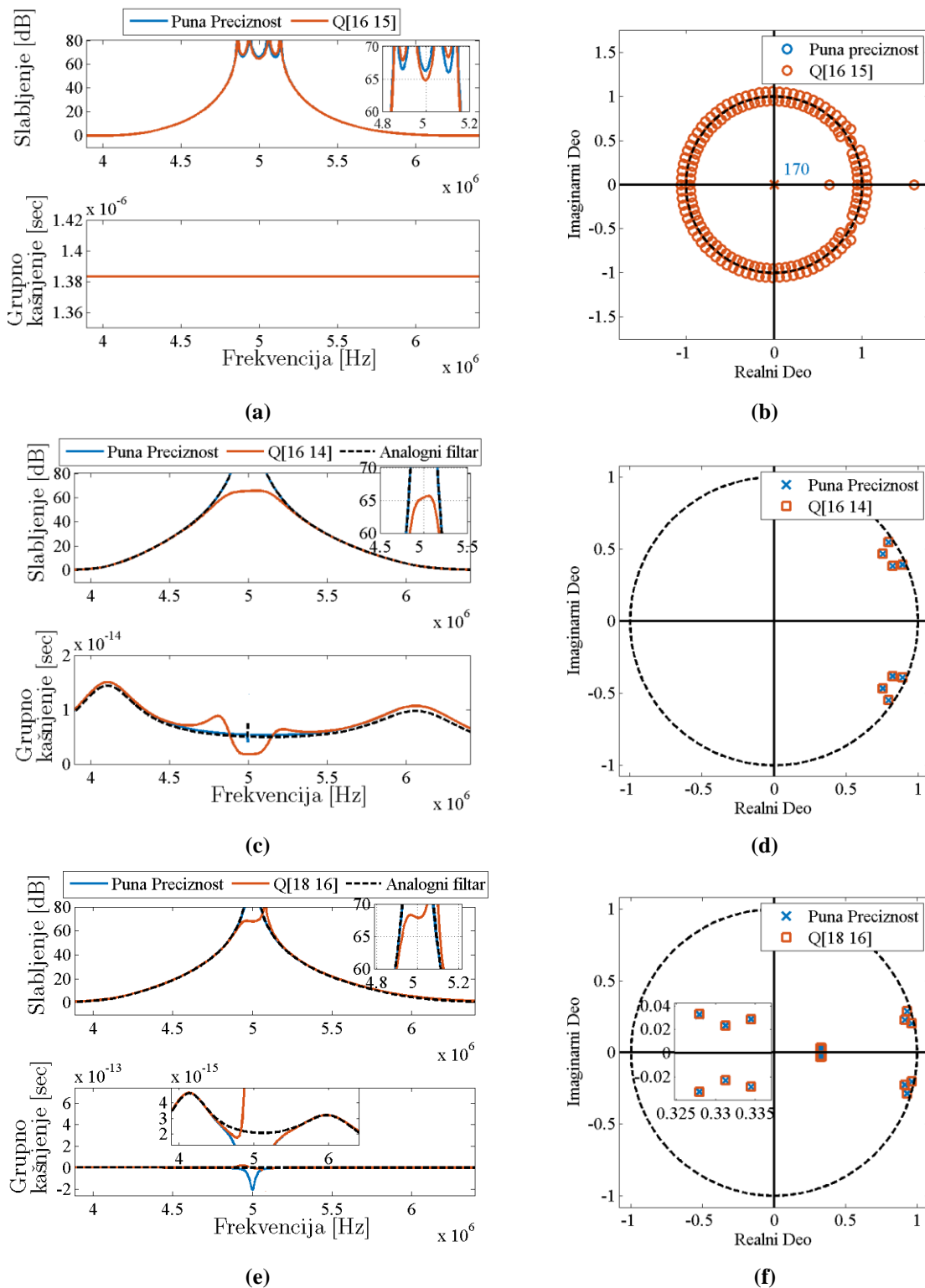
Slika 5.2 *Folded Delay Line* arhitektura FIR filtera sa linearnom fazom za neparni (a) i parni (b) red filtera M .

Tabela 5.1 Vrednosti polova LP prototipa i normalizovane vrednosti LSM BS filtera za bilinearnu transformaciju

Br.	BS		LP	
	Realni deo	Imaginarni deo	Realni deo	Imaginarni deo
1/2	-0.0487295044	± 0.8202217798	-0.2838434341	± 0.9265437853
3/4	-0.1961900822	± 0.8605934781	-0.6886065659	± 0.3750262747
5/6	-0.2518125277	± 1.1045829462		
7/8	-0.0721770666	± 1.2148944003		

Transformacija drugog reda (za razliku od bilinearne) rezultuje filtrom duplo većeg reda u odnosu na analogni prototip. U cilju minimizacije harvera upotrebljen je LSM prototip trećeg reda. Međutim, da bi se ispunili zahtevi usvojena je duplo veća frekvencija uzorkovanja tj. $f_s=122.88$ MHz. Polovi izabranog LSM LP prototipa i normalizovane vrednosti BS filtera za transformaciju drugog reda date su u Tabeli 5.2. U ovom slučaju BS prenosna funkcija ima tri para konjugovano-kompleksnih nula $\pm j\Omega_0$.

Na Sl. 5.3 je dato slabljenje, grupno kašnjenje i položaj nula/polova za FIR i dve varijante IIR filtara.



Slika 5.3 Slabljenje i grupno kašnjenje BS filtra: (a) FIR, $n=170$, (c) IIR, $n=8$, dobijen bilinearnom transformacijom i (e) IIR, $n=12$ dobijen transformacijom drugog reda. Položaj nula i polova FIR filtra (b), položaj polova IIR filtra dobijenog bilinearnom (d) i transformacijom drugog reda (f).

Tabela 5.2 Vrednosti polova LP prototipa i normalizovane vrednosti LSM BS filtra za transformaciju drugog reda

Br.	BS		LP	
	Realni deo	Imaginarni deo	Realni deo	Imaginarni deo
1/2	-0.0715481706	± 0.8257686640	-0.4076505823	± 0.8728824408
3/4	-0.2512882179	± 0.9679123057	-0.7958988355	0.0
5/6	-0.1041438339	± 1.2019694406		

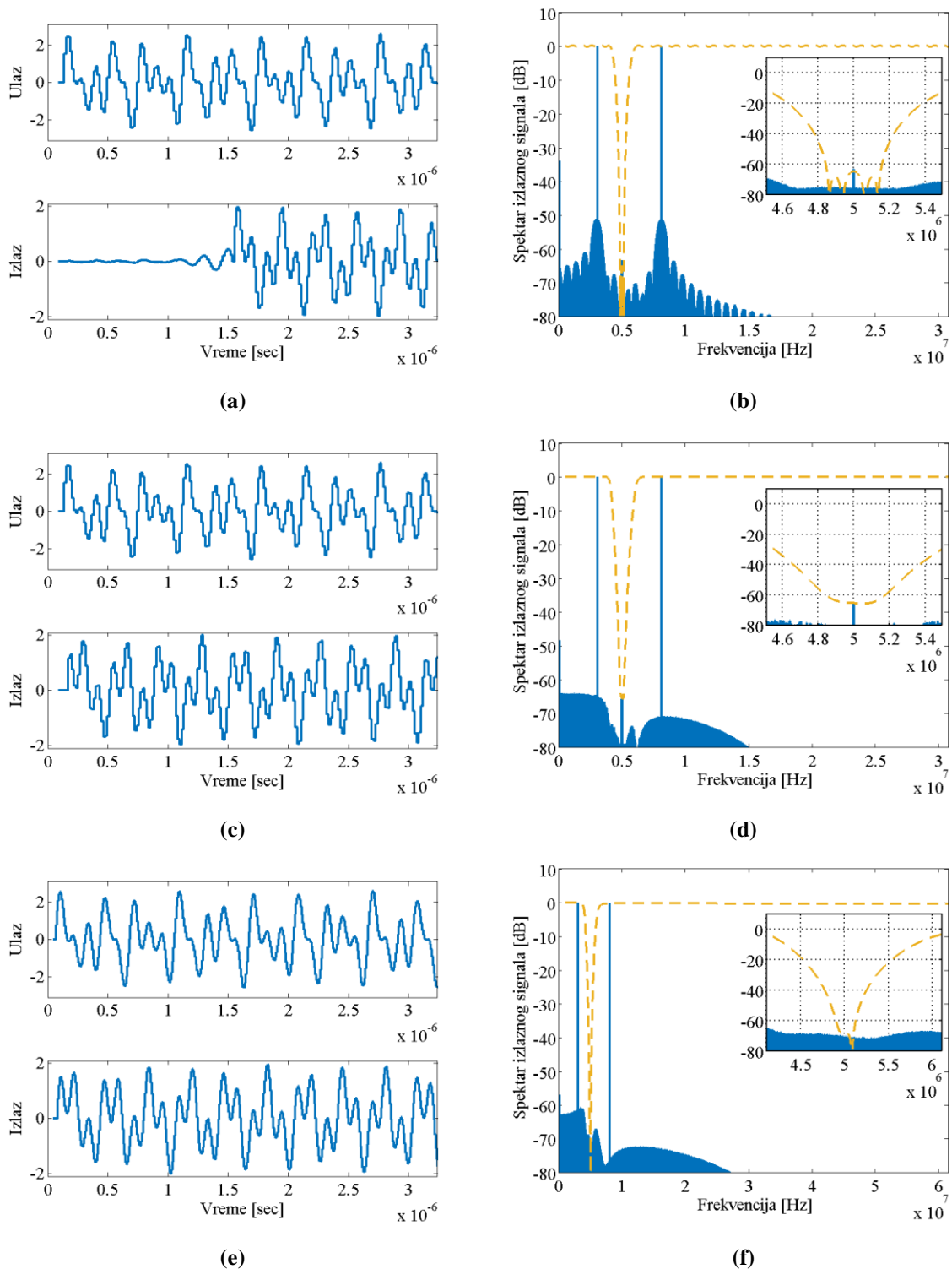
Koeficijenti FIR, IIR dobijenog bilinearnom i transformacijom drugog reda su kvantovani po formatu Q[16 15], Q[16 14] i Q[18 16], respektivno. Sa Sl. 5.3a, c i e se može videti da je kod FIR filtra linearnost faze obezbeđena dok kod IIR filtara to nije slučaj. Sa druge strane, vrednost grupnog kašnjenja kod IIR realizacije je osam do devet redova veličine manja nego kod FIR realizacije. Odstupanje grupnog kašnjenja u propusnom opsegu u odnosu na analogni prototip je nešto veće kod IIR filtra dobijenog na osnovu bilinearne transformacije nego kod IIR filtra na bazi transformacije drugog reda. Takođe se može uočiti da je postignuta dobra aproksimacija slabljenja kod svih realizacija filtara. Međutim, IIR realizacija na bazi transformacije drugog reda daje najveće slabljenje u nepropusnom opsegu. Položaj polova i nula FIR filtra je prikazan na Sl. 5.3b, a položaj polova za dve varijante IIR filtara na Sl. 5.3d i f. Kod obe varijante IIR filtra stabilnost je očuvana.

Vremenski odziv filtra i spektar izlaznog signal RTL modela su dati na Sl. 5.4. Testni signal je oblika (4.11) gde su izabrane frekvencije $f_1=3.09\text{MHz}$ i $f_2=8.09\text{MHz}$. Evidentno je da FIR realizacija rezultuje znatno većom latencijom u odnosu na IIR varijante (slike 5.4a, c, e). Na osnovu spektra izlaznog signala može se videti prisustvo jednosmerne (DC) komponente. Kod FIR filtra je ona najveća -33dB , kod IIR filtra dobijenog bilinearnom transformacijom -48dB , a kod IIR filtra dobijenog transformacijom drugog reda -56dB . Takođe se može uočiti da je selektivnost FIR realizacije i dalje manja u odnosu IIR realizaciju dobijenu transformacijom drugog reda čak iako je upotrebljen izuzetno veliki red FIR filtra.

Rezultati implementacije filtara u 65nm CMOS TSMC procesu su sumirani u Tabeli 5.3. Prve četiri vrste Tabele 5.3 sadrže podatke vezane za geometriju kola. Peta vrsta prikazuje procenu cene na osnovu površine. Procena apsolutnih vrednosti cena je izvedena na osnovu *low cost EURO PRACTICE* ponude za akademiju [Euro00]. U [Euro00] se može videti da je cena za ciljni CMOS process $4055\text{€}/\text{mm}^2$. Sva relativna odstupanja data u Tabeli 5.3 su dobijena po formuli

$$\delta_X = \frac{|X_{\text{FIR}} - X_{\text{IIR}}|}{X_{\text{FIR}}}, \quad (5.2)$$

gde je $X = \{\epsilon, P_{uk}, E_{uk}\}$, a indeksi FIR i IIR označavaju realizaciju filtra. Ušteda u ceni za IIR realizaciju na bazi bilinearne transformacije je oko 72%, a na bazi transformacije drugog reda oko 54%.



Slika 5.4 Vremenski odziv i spektar izlaznog signala RTL modela BS filtra: (a, b) FIR, $n=170$, (c, d) IIR, $n=8$, dobijenog bilinearom transformacijom i (e, f) IIR, $n=12$, dobijenog transformacijom drugog reda.

Tabela 5.3 Rezultati implementacije u 65nm CMOS tehnološkom procesu

Realizacija	Jedinica	FIR Parks- McClellan ($f_s=61.44\text{MHz}$)	IIR bilinearna transformacija ($f_s=61.44\text{MHz}$)	IIR transformacija drugog reda ($f_s=122.88\text{MHz}$)	
Broj standardnih ćelija	-	29979	6991	15499	
Broj metala	-	6	5	6	
Dužina veza	[μm]	323167.645	51574.795	99738.690	
Ukupna površina	[μm^2]	118336	32786	53436	
Cena	€	479	133	217	
	$\delta\epsilon$ [%]	-	72.23	54.70	
Potrošnja	WC $V_{DD}=1.08\text{V}$ $T=125^\circ\text{C}$	P_{int} [mW]	5.066	1.24	3.458
		P_{eks} [mW]	2.042	0.5202	2.004
	P_{cur} [mW]	0.01077	0.002383	0.004416	
	P_{uk} [mW]	7.1190	1.763	5.467	
	δP [%]	-	75.24	23.21	
	E_{uk} [$\mu\text{W}/\text{MHz}$]	115.869	28.694	44.491	
	δE [%]	-	75.23	61.60	
	BC $V_{DD}=1.32\text{V}$ $T=0^\circ\text{C}$	P_{int} [mW]	8.024	1.969	5.456
		P_{eks} [mW]	3.206	0.8213	3.155
		P_{cur} [mW]	0.02288	0.004692	0.008119
P_{uk} [mW]		11.250	2.795	8.619	
δP [%]		-	75.16	23.40	
E_{uk} [$\mu\text{W}/\text{MHz}$]		183.105	45.491	70.141	
	δE [%]	-	75.15	61.70	

Dakle, ukoliko se koristi IIR realizacija ostvaruju se značajne uštede sa ekonomskog aspekta. U ostalim vrstama je data procena potrošnje kola za dva kornera: najgori slučaj (WC – *worst case*), najbolji slučaj (BC – *best case*).

U terminologiji VLSI projektovanja najbolji/najgori slučaj se odnose na kašnjenje kola. Po pitanju potrošnje situacija je obrnuta, tj. BC je zapravo najgori slučaj. Više detalja vezanih za potrošnju i minimizaciju iste u savremenim VLSI kolima se može naći u [Jova16]. Ovde će biti date samo neke osnovne definicije u cilju boljeg razumevanja ostvarenih rezultata. *Encounter*[®] proračunava četiri komponente snage disipacije: P_{int} – interna snaga, P_{eks} – eksterna snaga, P_{cur} – snaga curenja i P_{uk} – ukupna snaga.

Interna i eksterna snaga disipacije predstavljaju dinamičku komponentu snage. U *Encounter*[®] okruženju eksterna snaga se označava kao *switching power*. Interna snaga disipacije potiče od struje prelaznog režima kojom se pune i prazne interne parazitne kapacitivnosti unutar samih standardnih ćelija. Eksterna snaga disipacije obuhvata sve parazitne elemente (veza, ulaza narednih standardnih ćelija, ...) van standardne ćelije. Generalno, dinamička komponenta snage u čvoru k može se izraziti kao

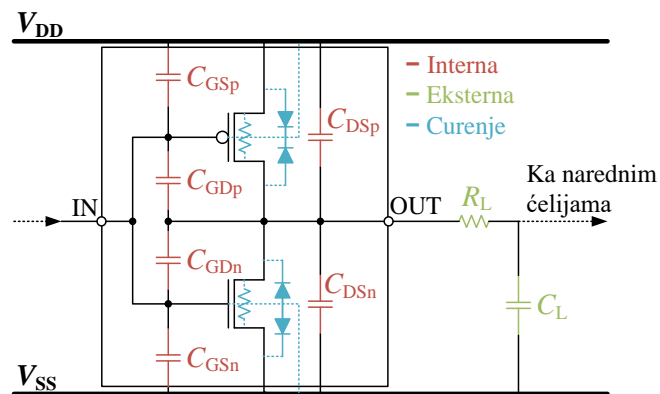
$$P_{d,k} = f_S \frac{\alpha_k}{2} C_k V_{DD}^2, \quad (5.3)$$

gde α_k aktivnost čvorova (*switching activity*), C_k parazitna ili kapacitivnost opterećenja čvora i V_{DD} napon napajanja [Jova13].

Snaga curenja spada u statičku komponentu snage. Prema [Kaes08] četiri ključna fenomena koji uzrokuju ovu disipaciju su:

- struja curenja kanala u podpragovskom režimu rada tj. kada je tranzistor zakočen,
- struja inverzno polarisanog spoja drejn-balk i sors-balk,
- struje inverzno polarisanih susednih *well*-ova i između *well*-a i supstrata i
- tunelovanje elektrona kroz oksid gejta (*gate leakage*).

P_{uk} je ukupna snaga i predstavlja zbir interne, eksterne i snage disipacije usled struja curenja. Na Sl. 5.5 je dat ilustrativni prikaz parazitnih komponenti koje uzrokuju snage definisane u *Encounter*[®] okruženju.



Slika 5.5 Ilustrativni prikaz definicije snaga disipacije u *Encounter*[®] okruženju.

Ako se pogledaju podaci u Tabeli 5.3 vezani za potrošnju vidi se da 65nm CMOS proces ima izuzetno nisku komponentu snage disipacije koja potiče od struja curenja. Ova komponenta je za red veličine manja kod IIR nego kod FIR realizacije što je i očekivano obzirom na značajno manju količinu hardvera. Glavne komponente potrošnje su interna i eksterna. U ovom primeru učešće interne snage disipacije u P_{uk} je oko 70%, a eksterne oko 30%.

Kada se pogleda relativno odstupanje δ_p može se videti da se korišćenjem IIR realizacije na bazi bilinearne transformacije ostvaruje ušteda od oko 75% u oba kornera. Za slučaj IIR realizacije na bazi nove transformacije drugog reda δ_p iznosi oko 23% za oba kornera.

Takođe je data i procena utrošene energije u jednom ciklusu procesiranja $1/f_s$, E_{uk} , tzv. efikasnost [Kaes08]. Ovaj parametar objedinjuje performanse po pitanju potrošnje i brzine rada pa se često daje kao mera za ocenu digitalnog kola. IIR realizacija na bazi bilinearne transformacije daje uštedu utrošene energije od oko 75%. Ovaj procenat je nešto niži za IIR

realizaciju na bazi transformacije drugog reda (oko 61%). Ovde treba naglasiti da je odnos $\delta_E > \delta_P$ kod IIR realizacije drugog reda posledica izbora duplo veće frekvencije uzorkovanja. Naime, za jedan ciklus obrade kod IIR realizacije na bazi bilinearne idu dva ciklusa obrade kod IIR realizacije na bazi transformacije drugog reda.

Generalno, manja efikasnost i veća potrošnja IIR realizacije na bazi transformacije drugog reda u odnosu na realizaciju na bazi bilinearne transformacije je rezultat upotrebe duplo veće frekvencije uzorkovanja i veće količine hardvera. Međutim, na račun manje efikasnosti ostvareno je bolje slabljenje u propusnom opsegu i veća selektivnost.

Tabela 5.4 Koeficijenti LSM BS FIR filtra reda 170 predstavljeni u punoj preciznosti

Puna preciznost			
<i>i</i>	<i>c_i</i>	<i>i</i>	<i>c_i</i>
0	0.0069704500658114149	43	0.0035739211819640431
1	-0.0064003425092590749	44	0.0025454274594954602
2	-0.0040826305518089842	45	0.00024682429655550035
3	-0.0027224004247593746	46	-0.0029702971746923832
4	-0.0020554426132332192	47	-0.0063007531924037996
5	-0.0018120078277654153	48	-0.008695231320297276
6	-0.0017002667695778229	49	-0.0091143582067691568
7	-0.0014556626287832693	50	-0.0069290004611307745
8	-0.00090363090514677296	51	-0.0022194753298267599
9	-0.000025293241124169994	52	0.0041028397903020188
10	0.0010260492304403122	53	0.010442279548603467
11	0.0019722402994001454	54	0.014910736819917612
12	0.0024985778302017213	55	0.015874627248864917
13	0.0023703682828878831	56	0.012489527790789053
14	0.0015293921089423537	57	0.0050909715231313686
15	0.00013112917735095633	58	-0.0047533214703346502
16	-0.0014717262520926807	59	-0.014536139260721417
17	-0.002833375924923962	60	-0.021459275160081056
18	-0.0035379101752092343	61	-0.023214706936356286
19	-0.0033416944135561842	62	-0.018742040384091749
20	-0.0022459070663904406	63	-0.0086533820272711968
21	-0.00052040860882907893	64	0.004764513283588203
22	0.0013802349997124918	65	0.01807172570651951
23	0.0029390113307212608	66	0.027554325412565674
24	0.0037381048624691086	67	0.030278156735077177
25	0.0035757228961850879	68	0.024978182972397438
26	0.0025300162572275222	69	0.012550293969825139
27	0.00092256496680556764	70	-0.0040610863456312773
28	-0.00077908622721326439	71	-0.020557023222046857
29	-0.0021192550072012665	72	-0.032422239683640448
30	-0.002786092587363986	73	-0.036175371845017984
31	-0.0027085013391362788	74	-0.030412108382370313
32	-0.0020425322923614615	75	-0.016277139965055344
33	-0.00111613229655598	76	0.0027293639862347382
34	-0.00026855968689061289	77	0.02165297296160407
35	0.00025376217686769929	78	0.035380952700364968
36	0.00042219669005289829	79	0.040085175054072941
37	0.00038484332411974976	80	0.034277761793793764
38	0.00043549024323252935	81	0.019313910194333064
39	0.00080752530297031032	82	-0.00096018255794495582
40	0.0015948754103021932	83	-0.021208134344506781
41	0.0026273866573892374	84	-0.036025281118388354
42	0.0034448294781808907	85	0.95855235463113142

Koeficijenti LSM BS FIR filtra reda 170 u punoj preciznosti su dati u Tabeli 5.4, a kvantovane vrednosti koeficijenata u Tabeli 5.5. Za IIR realizaciju na bazi bilinearne transformacije koeficijenti brojioca i imenioca po pojedinim sekcijama su dati u Tabeli 5.6 i 5.7, respektivno. Najzad, za IIR relizaciju na bazi transformacije drugog reda koeficijenti su dati u Tabeli 5.8.

Tabela 5.5 Kvantovani koeficijenti BS FIR filtra reda 170

Q[16 15]							
decimalna				heksadecimalna			
<i>i</i>	<i>c_i</i>	<i>i</i>	<i>c_i</i>	<i>i</i>	<i>c_i</i>	<i>i</i>	<i>c_i</i>
0	0.006958007812500	43	0.003570556640625	0	00E4	43	0075
1	-0.006408691406250	44	0.002532958984375	1	FF2E	44	0053
2	-0.004089355468750	45	0.000244140625000	2	FF7A	45	0008
3	-0.002716064453125	46	-0.002960205078125	3	FFA7	46	FF9F
4	-0.002044677734375	47	-0.006286621093750	4	FFBD	47	FF32
5	-0.001800537109375	48	-0.008697509765625	5	FFC5	48	FEE3
6	-0.001708984375000	49	-0.009124755859375	6	FFC8	49	FED5
7	-0.001464843750000	50	-0.006927490234375	7	FFD0	50	FF1D
8	-0.000915527343750	51	-0.002227783203125	8	FFE2	51	FFB7
9	-0.000030517578125	52	0.004089355468750	9	FFFF	52	0086
10	0.001037597656250	53	0.010437011718750	10	0022	53	0156
11	0.001983642578125	54	0.014923095703125	11	0041	54	01E9
12	0.002502441406250	55	0.015869140625000	12	0052	55	0208
13	0.002380371093750	56	0.012481689453125	13	004E	56	0199
14	0.001525878906250	57	0.005096435546875	14	0032	57	00A7
15	0.000122070312500	58	-0.004760742187500	15	0004	58	FF64
16	-0.001464843750000	59	-0.014526367187500	16	FFD0	59	FE24
17	-0.002838134765625	60	-0.021453857421875	17	FFA3	60	FD41
18	-0.003540039062500	61	-0.023223876953125	18	FF8C	61	FD07
19	-0.003356933593750	62	-0.018737792968750	19	FF92	62	FD9A
20	-0.002258300781250	63	-0.008666992187500	20	FFB6	63	FEE4
21	-0.000518798828125	64	0.004760742187500	21	FFEF	64	009C
22	0.001373291015625	65	0.018066406250000	22	002D	65	0250
23	0.002929687500000	66	0.027557373046875	23	0060	66	0387
24	0.003723144531250	67	0.030273437500000	24	007A	67	03E0
25	0.003570556640625	68	0.024963378906250	25	0075	68	0332
26	0.002532958984375	69	0.012542724609375	26	0053	69	019B
27	0.000915527343750	70	-0.004058837890625	27	001E	70	FF7B
28	-0.000793457031250	71	-0.020568847656250	28	FFE6	71	FD5E
29	-0.002105712890625	72	-0.032409667968750	29	FFBB	72	FBDA
30	-0.002777099609375	73	-0.036163330078125	30	FFA5	73	FB5F
31	-0.002716064453125	74	-0.030426025390625	31	FFA7	74	FC1B
32	-0.002044677734375	75	-0.016265869140625	32	FFBD	75	FDEB
33	-0.001129150390625	76	0.002716064453125	33	FFDB	76	0059
34	-0.000274658203125	77	0.021667480468750	34	FFF7	77	02C6
35	0.000244140625000	78	0.035369873046875	35	0008	78	0487
36	0.000427246093750	79	0.040100097656250	36	000E	79	0522
37	0.000396728515625	80	0.034271240234375	37	000D	80	0463
38	0.000427246093750	81	0.019317626953125	38	000E	81	0279
39	0.000793457031250	82	-0.000946044921875	39	001A	82	FFE1
40	0.001586914062500	83	-0.021209716796875	40	0034	83	FD49
41	0.002624511718750	84	-0.036010742187500	41	0056	84	FB64
42	0.003448486328125	85	0.958557128906250	42	0071	85	7AB2

Tabela 5.6 Koeficijenti brojioca LSM BS filtra 8. reda dobijenog bilinearnom transformacijom

		Brojilac			
		sekcija	c_0	c_1	c_2
Puna preciznost	I		-0.016398696220208495	-0.010840132971092944	0.0055585632491155484
	II		-0.11221057157164607	-0.010240482888061134	0.10197008868358495
	III		-0.11019287746605651	0.016015892907438919	0.12620877037349543
	IV		0.00015330163696891563	0.022425349138667242	0.022272047501698325
Q[16 14] decimnalna	I		-0.01641845703125	-0.01086425781250	0.00555419921875
	II		-0.11218261718750	-0.01025390625000	0.10198974609375
	III		-0.11016845703125	0.01599121093750	0.12622070312500
	IV		0.00018310546875	0.02239990234375	0.02227783203125
Q[16 14] heksadeci- malna	I		FEF3	FF4E	005B
	II		F8D2	FF58	0687
	III		F8F3	0106	0814
	IV		0003	016F	016D

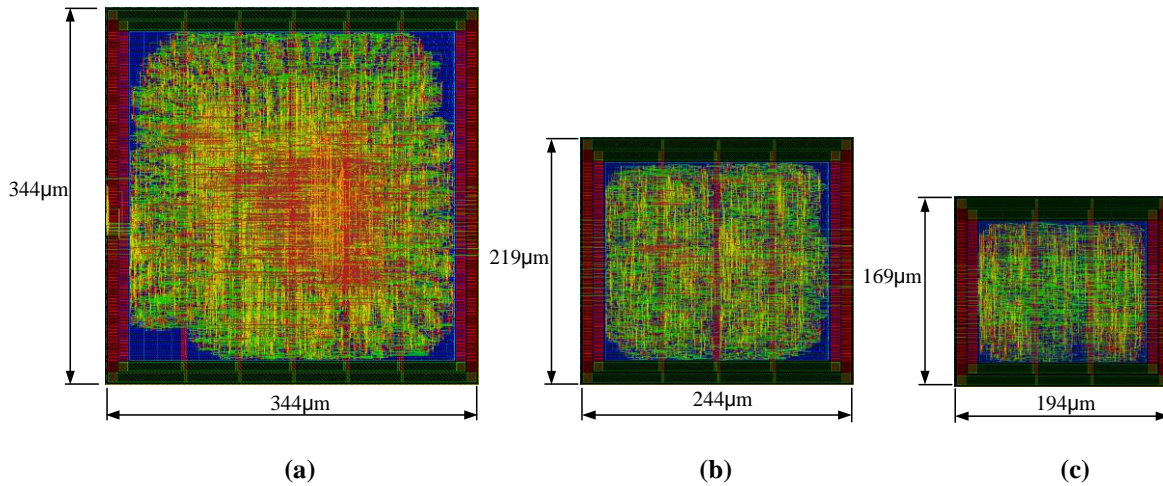
Tabela 5.7 Koeficijenti imenioca LSM BS filtra 8. reda dobijenog bilinearnom transformacijom

		Imenilac		
		sekcija	d_1	d_2
Puna preciznost	I		-1.7882674878828715	0.95338516696825548
	II		-1.6487819031997475	0.8257235454351094
	III		-1.5109088557315891	0.78764174263605213
	IV		-1.5933094079882946	0.93489392658498971
Q[16 14] decimnalna	I		-1.788269042968750	0.953369140625000
	II		-1.648803710937500	0.825744628906250
	III		-1.510925292968750	0.787658691406250
	IV		-1.593322753906250	0.934875488281250
Q[16 14] heksadeci- malna	I		8D8D	3D04
	II		967A	34D9
	III		9F4D	3269
	IV		9A07	3BD5

Tabela 5.8 Koeficijenti brojioca i imenioca LSM BS filtra 12. reda dobijenog transformacijom drugog reda

		Brojilac		Imenilac		
		sekcija	c_0	c_1	d_1	d_2
Puna preciznost	I		-0.035870204443545377	0.031804094524434411	-1.9219949647375596	0.96467956601583105
	II		0.010443270724096333	-0.003636044685470793	-0.66257143338345592	0.1102882525550067
	III		-0.12322841884997347	0.11927346079526567	-1.8258013037047089	0.88452641498185802
	IV		0.045082892366942649	-0.015195586303550938	-0.66890872451824401	0.11268975836710904
	V		-0.045770967751013844	0.050397960886568741	-1.8621390511284264	0.94881087918976847
	VI		0.017843955505611669	-0.0057713638627927537	-0.65598647162481827	0.10865385671267927
Q[18 16] decimnalna	I		-0.0358734130859375	0.0317993164062500	-1.9219970703125000	0.9646759033203125
	II		0.0104370117187500	-0.0036315917968750	-0.6625671386718750	0.1102905273437500
	III		-0.1232299804687500	0.1192779541015625	-1.8258056640625000	0.8845214843750000
	IV		0.0450897216796875	-0.0151977539062500	-0.6689147949218750	0.1126861572265625
	V		-0.0457763671875000	0.0503997802734375	-1.8621368408203125	0.9488067626953125
	VI		0.0178375244140625	-0.0057678222656250	-0.6559906005859375	0.1086578369140625
Q[18 16] heksadeci- malna	I		3F6D1	00824	213F8	0F6F5
	II		002AC	3FF12	35662	01C3C
	III		3E074	01E89	22C98	0E270
	IV		00B8B	3FC1C	354C2	01CD9
	V		3F448	00CE7	2234B	0F2E5
	VI		00491	3FE86	35811	01BD1

Na Sl. 5.6 je dat prikaz leajuta tri realizacije LSM BS filtra u 65nm CMOS TSMC procesu.



Slika 5.6 Lejaut tri realizacije BS filtra: (a) FIR ($118336 \mu\text{m}^2 \approx 0.118\text{mm}^2$), (b) IIR dobijen kvadratnom ($52216 \mu\text{m}^2 \approx 0.052\text{mm}^2$) i (c) IIR dobijen bilinearnom transformacijom ($32786 \mu\text{m}^2 \approx 0.033\text{mm}^2$)

Ostvarene uštede IIR u odnosu na FIR realizaciju su dobijene na osnovu značajno manje količine hardvera potrebne da se implementira zadata funkcija, naravno ako se izuzme linearnost faze. Na osnovu rezultata u Tabeli 5.3 i Sl. 5.3 i 5.4 se jasno može videti kompromis između površine (cene), potrošnje i performansi kola. Na račun ušteta u potrošnji/površini/ceni može se čak poboljšati funkcija kola usvajanjem veće frekvencije uzorkovanja i/ili većeg broja bitova za kvantizaciju koeficijenata kao što je pokazano na primeru IIR realizacije na bazi transformacije drugog reda.

6 Primena transformacija sa specijalnom namenom

Pored do sada razmatranih transformacija, koje se baziraju na aproksimaciji analognog integratora, postoje transformacije koje se mogu klasifikovati kao transformacije sa posebnom, tj. specijalnom namenom. Tako se npr. u cilju očuvanja fazne karakteristike analognog prototipa može koristiti ranije pomenuti *Phase Invariance Method* (PIM) metod. PIM metod je uveden od strane Parmana i Artisa 2006. i proistekao je iz *Magnitude Invariance Method* (MIM) metoda razvijenog od strane Parmana 1998. MATLAB implementacija MIM i PIM metoda je data u [Vadh07] i u ovom radu je iskorišćena u svrhe poređenja sa bilinearnom i novom transformacijom drugog reda. Kompletnosti radi u nastavku će biti rekapitulirane neke bitne činjenice i zaključci u vezi MIM, a kasnije i PIM metoda prezentovani u [Paar98, PaAt06]. Naravno, za ovu disertaciju od posebnog interesa je PIM metod. Međutim, obzirom da se većina matematičkog aparata PIM metoda oslanja na MIM metod, prvo će biti razmatran MIM metod.

6.1 Metod invarijantne amplitudske karakteristike (MIM)

Kod MIM metoda se polazi od odgovarajućeg kontinualnog sistema, tj. analognog prototipa. Principijelno, cilj je dobiti diskretni sistem sa gotovo identičnom amplitudskom karakteristikom kao kod polaznog analognog prototipa. MIM transformacija podrazumeva kauzalni, diskretni sistem sa minimalnom fazom za koji važi:

$$\left|H(e^{j\theta})\right| = \left|H_c(j\omega/T)\right|, \quad |\omega| \leq \pi, \quad (6.1)$$

gde je $H(e^{j\theta})$ frekvencijski odziv diskretnog, a $H_c(j\omega/T)$ kontinualnog sistema i T perioda uzorkovanja. U [PaAt06] je pokazano da se autokorelaciona funkcija diskretnog sistema može dobiti konvolucijom skalirane autokorelacione funkcije originalnog, analognog, sistema i *sinc* funkcije. Prema [PaAt06] za autokorelacionu funkciju diskretnog sistema definisanog sa (6.1) važi

$$r(n) = h(n) * \bar{h}(n). \quad (6.2)$$

U (6.2) $h(n)$ predstavlja impulsni odziv diskretnog sistema sa minimalnom, a $\bar{h}(n)=h(-n)$ sa maksimalnom fazom. Do vrednosti impulsnog odziva, $h(n)$, diskretnog sistema se dolazi dekonvolucijom dobijene autokorelacione funkcije. To se može postići uz pomoć homomorfno filtriranja. Ovaj koncept je razvijen od strane autora [OpSS68] i podrazumeva

nelinearno mapiranje signala u drugi domen, primenu linearnog filtriranja, a zatim vraćanje u polazni domen putem inverznog nelinearnog mapiranja. Jedno nelinearno mapiranje na bazi primene DTFT (*Discrete Time Fourier Transform*) predstavlja *cepstrum*¹. Pojam *cepstrum* je prvi put uveden od strane grupe autora [BoHT63]. Generalizovanu definiciju za kompleksni *cepstrum* dali su Openhajm (*Oppenheim*) i Šafer (*Schafer*) gde je obrađen kompletan matematički aparat za homomorfno procesiranje [OpSc10]. Detaljna analiza teorije homomornog procesiranja prevazilazi okvire ove disertacije tako da će nadalje biti interpretirane ključne definicije i pojmovi u cilju pojašnjenja MIM metoda.

U [OpSc10] kompleksni *Cepstrum* se definiše kao diskretna sekvenca $\hat{x}(n)$ za čiju z -transformaciju važi:

$$\hat{X}(z) = \log(X(z)), \quad (6.3)$$

gde je $X(z)$ z -transformacija stabilne diskretne sekvence $x(n)$, a baza logaritma prirodna². $\hat{x}(n)$ se može sintetisati primenom IDFT (*Inverse Discrete Time Fourier Transform*),

$$\begin{aligned} \hat{x}(n) &= \frac{1}{2\pi} \int_{-\pi}^{\pi} \log(X(e^{j\theta})) e^{jn\theta} d\theta \\ &= \frac{1}{2\pi} \int_{-\pi}^{\pi} \left(\log|X(e^{j\theta})| + j \arg\{X(e^{j\theta})\} \right) e^{jn\theta} d\theta, \end{aligned} \quad (6.4)$$

gde je $X(e^{j\theta})$ DTFT sekvence $x(n)$. Pored kompleksnog, definiše se još i realni *cepstrum* sekvence $x(n)$ kao,

$$c(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} \log(|X(e^{j\theta})|) e^{jn\theta} d\theta, \quad (6.5)$$

Realni *cepstrum* praktično predstavlja realni deo kompleksnog *cepstrum*-a $\hat{x}(n)$ i može se takođe dobiti iz,

$$c(n) = \frac{\hat{x}(n) + \hat{x}^*(-n)}{2}, \quad (6.6)$$

gde je $\hat{x}^*(n)$ konjugovano-kompleksni, vremenski invertovani *cepstrum*. Do relacije (6.6) se dolazi jednostavno kombinovanjem (6.4) i (6.5).

¹ *Cepstrum* je kovanica nastala invertovanjem redosleda prva četiri slova pojma *spectrum*. To je učinjeno kako bi se naznačila razlika u odnosu na klasično poimanje "spektra" (*spectrum*). Tako su autori [BoHT63] razvili čitav vokabular pojmova poput *quefreny* i *liftering* kao pandan klasičnim pojmovima *frequency* i *filtering*.

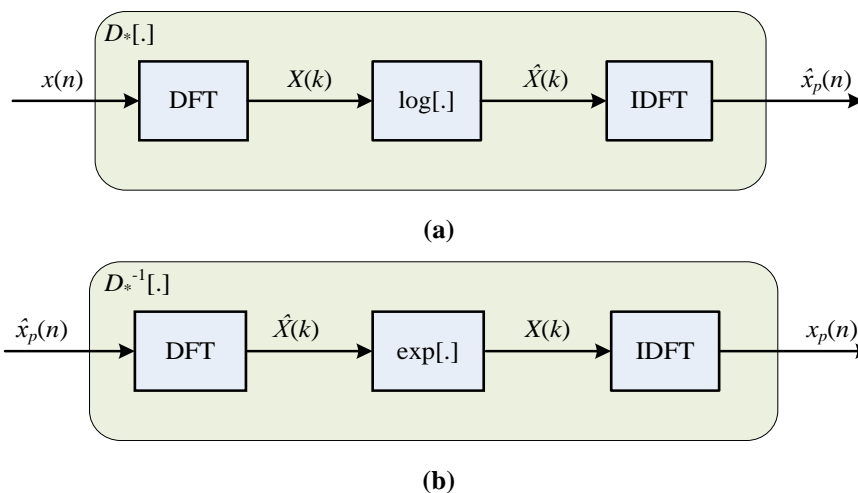
² Prema [OpSc10] u opštem slučaju \log može biti logaritam bilo koje osnove, ali se najčešće uzima prirodna osnova.

Neke od osnovnih osobina kompleksnog cepstrum-a, $x(n)$, su:

1. Sekvenca $x(n)$ će imati beskonačno trajanje čak i kada sekvenca $x(n)$ ima konačno trajanje.
2. Ukoliko je $\log\left(|X(e^{j\theta})|\right)$ parna, a $\arg\{X(e^{j\theta})\}$ neparna funkcija po θ onda je $x(n)$ realna sekvenca.
3. Ako je $x(n)$ realna sekvenca onda je i $x(n)$ realna sekvenca.
4. Za $x(n)$ važi $\hat{x}(n) = 0$ za $n < 0$ (kauzalni sistem) samo i samo ako je $x(n)$ sistem sa minimalnom fazom, tj. $X(z)$ ima sve nule i polove unutar jediničnog kruga.
5. Za $x(n)$ važi $\hat{x}(n) = 0$ za $n > 0$ (nekauzalni sistem) samo i samo ako je $x(n)$ sistem sa maksimalnom fazom tj. $X(z)$ ima sve nule i polove van jediničnog kruga.

Generalno, traženje cepstrum-a sekvence $x(n)$ se može opisati izrazima datim u (6.7). Praktično, definisane relacije se procenjuju numeričkim putem. Dakle, umesto DTFT/IDTFT treba koristiti DFT/IDFT (*Discrete Fourier Transform/Inverse Discrete Fourier Transform*), a frekvencijsku osu, θ , diskretizovati ($\theta_k=2\pi k/N$). Sistem za proračun cepstrum-a sekvence $x(n)$ je blokovski prikazan na Sl. 6.1a, a odgovarajući sistem jednačina je (6.8). Ceo proces se označava operatorom D_* . Inverzan proces se označava sa D_*^{-1} (Sl. 6.1b).

$$\begin{aligned}
 X(e^{j\theta}) &\stackrel{DTFT}{=} \sum_{n=-\infty}^{\infty} x(n)e^{-jn\theta} \\
 \hat{X}(e^{j\theta}) &= \log\left(X(e^{j\theta})\right) \\
 \hat{x}(n) &\stackrel{IDTFT}{=} \frac{1}{2\pi} \int_{-\pi}^{\pi} \hat{X}(e^{j\theta})e^{jn\theta} d\theta.
 \end{aligned} \tag{6.7}$$



Slika 6.1 Blokovski prikaz numeričke aproksimacije procesa: (a) za proračun cepstrum-a $\hat{x}(n)$ na osnovu sekvence $x(n)$ i (b) proračun sekvence $x(n)$ na osnovu cepstrum-a $\hat{x}(n)$.

Bitno je naglasiti da se u ovom slučaju dobija numerička aproksimacija *cepstrum*-a $\hat{x}(n)$ koja je označena sa $\hat{x}_p(n)$. Pošto se za proračun $\hat{x}_p(n)$ koriste odmerci $\hat{X}(e^{j\theta})$ moguća je pojava aliasinga (*aliasing*) u vremenskom domenu. Tu se kao primarni problem implementacije (6.8) nameće kontinualnost faze $X(k)$ (*phase unwrapping*). Za ovaj problem ne postoji univerzalno rešenje. Neki od najpopularnijih algoritama za *phase unwrapping* su predloženi u [Itoh82, StDi82, Trib77].

$$\begin{aligned}
 X \left(e^{jn \left(\frac{2\pi}{N} \right) k} \right) &= X(k) \stackrel{DFT}{=} \sum_{n=-\infty}^{\infty} x(n) e^{-jn \left(\frac{2\pi}{N} \right) k} \\
 \hat{X}(k) &= \log(X(k)) \\
 \hat{x}_p(n) &\stackrel{IDTFT}{=} \frac{1}{N} \sum_{k=0}^{N-1} \hat{X}(k) e^{jn \left(\frac{2\pi}{N} \right) k} \\
 \hat{x}_p(n) &= D_*[x(n)].
 \end{aligned} \tag{6.8}$$

Takođe, treba uočiti da se numerička procena poboljšava (tj. greška se smanjuje) ukoliko se uzme veći broj odmeraka (veće N). Međutim, to direktno utiče na efikasnost algoritma, povećavajući značajno vreme proračunavanja.

D_* operator ima ključnu ulogu u teoriji homomorfni sistema, tj. pomoću njega se definiše generalizacija koncepta superpozicije. Kod homomorfno filtriranja konvolucije signal operator D_* se naziva „karakteristični sistem za konvoluciju“ (*characteristic system for convolution*) pošto omogućava konverziju operacije konvolucije u operaciju sabiranja. To se može pokazati na sledeći način. Neka je sekvence $x(n)$ definisana sa

$$x(n) = x_1(n) * x_2(n), \tag{6.9}$$

gde je sa $*$ označena operacija konvolucije. Odgovarajuća z -transformacija sekvence $x(n)$ je,

$$X(z) = X_1(z) \cdot X_2(z). \tag{6.10}$$

Na osnovu definicije (6.3) važi,

$$\begin{aligned}
 \hat{X}(z) &= \log(X(z)) \\
 &= \log(X_1(z)) + \log(X_2(z)) \\
 &= \hat{X}_1(z) + \hat{X}_2(z),
 \end{aligned} \tag{6.11}$$

odakle sledi,

$$\hat{x}(n) = D_*[x_1(n) * x_2(n)] = \hat{x}_1(n) + \hat{x}_2(n). \quad (6.12)$$

Slično važi i u suprotnom smeru. Npr. neka je $\hat{y}(n)$,

$$\hat{y}(n) = \hat{y}_1(n) + \hat{y}_2(n), \quad (6.13)$$

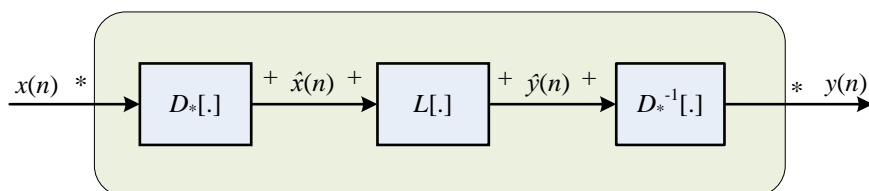
odakle sledi,

$$\begin{aligned} Y(z) &= e^{\hat{Y}(z)} \\ &= e^{\{\log(Y_1(z)) + \log(Y_2(z))\}} \\ &= Y_1(z) \cdot Y_2(z), \end{aligned} \quad (6.14)$$

pa je,

$$y(n) = D_*^{-1}[\hat{y}_1(n) + \hat{y}_2(n)] = y_1(n) * y_2(n). \quad (6.15)$$

Generalizacija koncepta superpozicije kod homomornog filtriranja konvolucije signala je ilustrovana na Sl. 6.2. $L[.]$ predstavlja linearni operator, tj. sistem (npr. filter) koji ne mora obavezno biti i vremenski invarijantan pošto obrađuje *cepstrum* sekvence, a ne sekvencu direktno. Simboli „+“ i „*“ označavaju operator koji definiše superpoziciju ulaznih/izlaznih signala pojedinih blokova. Praktično, svi homomorfni sistemi se uglavnom razlikuju po operatoru $L[.]$.



Slika 6.2 Generalizacija koncepta superpozicije kod homomornog filtriranja konvolucije signala

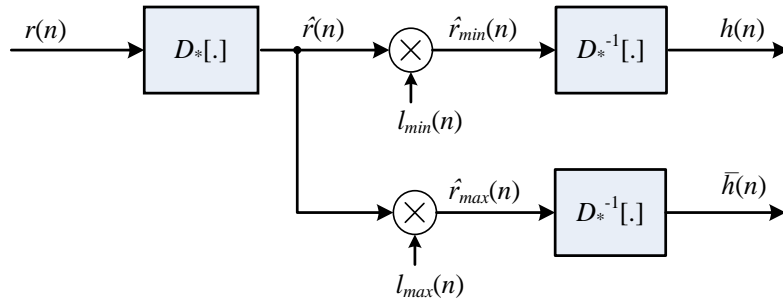
Pošto autokorelaciona funkcija $r(n)$ definisana u (6.2) predstavlja konvoluciju impulsnih odziva sistema sa minimalnom, $h(n)$, i maksimalnom, $\bar{h}(n)$, fazom potrebno je primeniti homomorfno filtriranje konvolucije signala pomoću koga se izdvajaju sistemi sa minimalnom i maksimalnom fazom. Ako se $r(n)$ definiše kao,

$$r(n) = r_{\min}(n) * r_{\max}(n), \quad (6.16)$$

onda je *cepstrum* od $r(n)$,

$$\hat{r}(n) = \hat{r}_{\min}(n) + \hat{r}_{\max}(n). \quad (6.17)$$

Homomorfni sistema za dekonvoluciju (6.16) je prikazanog na Sl. 6.3.



Slika 6.3 Homomorfno filtriranje autokorelacione funkcije

Pošto je $\hat{r}_{\min}(n)$ *cepstrum* sistema sa minimalnom fazom onda je $\hat{r}_{\min}(n) = 0$ za $n < 0$ (4. osobina, str. 76). Dakle, $\hat{r}_{\min}(n)$ se može jednostavno izdvojiti uzimanjem vrednosti $\hat{r}(n)$ za $n > 0$. Slično, za $\hat{r}_{\max}(n)$ važi da je $\hat{r}_{\max}(n) = 0$ za $n > 0$ (5. osobina, str. 76) pošto predstavlja *cepstrum* sistema sa maksimalnom fazom. Sada treba izdvojiti vrednosti $\hat{r}(n)$ za $n < 0$. Praktično, l_{\min} i l_{\max} su jednostavno prozorske funkcije koje izdavaju odgovarajući deo *cepstrum*-a. Vrednosti $\hat{r}_{\max}(n)$ i $\hat{r}_{\min}(n)$ za $n=0$ se preklapaju tako da ostaju kao slobodni parametar pod ograničenjem,

$$\hat{r}(0) = \hat{r}_{\min}(0) + \hat{r}_{\max}(0). \quad (6.18)$$

Kako MIM podrazumeva kauzalni diskretan sistem sa minimalnom fazom (stabilan sistem) dovoljno je procesirati samo $\hat{r}_{\min}(n)$. U [PaAt06] je za l_{\min} izabrana prozorska funkcija,

$$l_{\min} = \begin{cases} 0 & \text{za } n < 0 \\ 1/2 & \text{za } n = 0 \\ 1 & \text{za } n > 0. \end{cases} \quad (6.19)$$

Kada se dobije impulsni odziv kauzalnog diskretnog sistema sa minimalnom fazom, $h(n)$, ostaje da se proračunaju koeficijenti filtra. Pošto se radi o sistemu sa minimalnom fazom svi polovi će biti unutar jediničnog kruga, a nule unutar ili na (granično minimalnofazni sistem) jediničnom krugu. Ukoliko bi se primenila Z transformacija direktno na $h(n)$ dobila bi se FIR realizacija filtra čiji su koeficijenti vrednosti $h(n)$. Međutim, u [PaAt06] je postavljena hipoteza koja se može interpretirati na sledeći način. Pošto je $h(n)$ dobijeno na osnovu analognog prototipa, mora postojati adekvatna IIR realizacija sistema istog reda kao i polazni analogni prototip, sa istim brojem polova i nula, koja zadovoljava kriterijume za minimalnost faze i čiji je odziv identičan $h(n)$. Dakle, pretpostavka je da postoje realni koeficijenti c_i i d_i IIR filtera za čiju, racionalnu, prenosnu funkciju važi,

$$\sum_{n=0}^{\infty} h(n)z^{-n} = \frac{\sum_{m=0}^M c_m z^{-m}}{1 + \sum_{m=1}^M d_m z^{-m}}. \quad (6.20)$$

Grupisanjem članova uz odgovarajuće stepene z^{-1} dolazi se do sistema linearnih jednačina čija su rešenja koficijenti filtra [GrKM75].

6.2 Metod invarijantne fazne karakteristike (PIM)

Iz teorije sistema je poznato da postoji analitička veza između imaginarnog i realnog dela DTFT/DFT stabilne, realne i kauzalne, sekvence $x(n)$. Ovakva veza je poznata pod nazivom Hilbertova (*Hilbert*) transformacija. U [OpSc10] je data relacija (6.21) koja daje vezu između realnog, $X_R(k)$, i imaginarnog, $X_I(k)$, dela DFT sekvence $x(n)$. Kada se (6.21) primeni na DFT *cepstrum*-a sekvence $x(n)$ dobija se veza između $\hat{X}_R(k)$ i $\hat{X}_I(k)$, gde je $\hat{X}_R(k) = \log(|X(k)|)$ i $\hat{X}_I(k) = \arg\{X(k)\}$. Pošto DFT predstavlja uzorkovanu verziju DTFT onda su $|X(k)|$ i $\arg\{X(k)\}$ ništa drugo do odmerci modula i argumenta DTFT (koja je za diskretne signale kontinualna funkcija), respektivno.

$$\begin{aligned} X_R(k) &= \begin{cases} \frac{1}{N} \sum_{m=0}^{N-1} X_I(m) V_N(k-m) + x(0) + (-1)^k x(N/2), & 0 \leq k \leq N-1 \\ 0 & 0 > k > N-1 \end{cases} \\ jX_I(k) &= \begin{cases} \frac{1}{N} \sum_{m=0}^{N-1} X_R(m) V_N(k-m) & 0 \leq k \leq N-1 \\ 0 & 0 > k > N-1 \end{cases} \\ V_N(k) &= \begin{cases} -j \cot(\pi k / N) & k - \text{neparno} \\ 0 & k - \text{parno}. \end{cases} \end{aligned} \quad (6.21)$$

Dakle, ako se zadaju odmerci fazne karakteristike sistema sa minimalnom fazom, $\Phi(k)$, može se doći do logaritma modula amplitudske karakteristike $|H(k)|$.

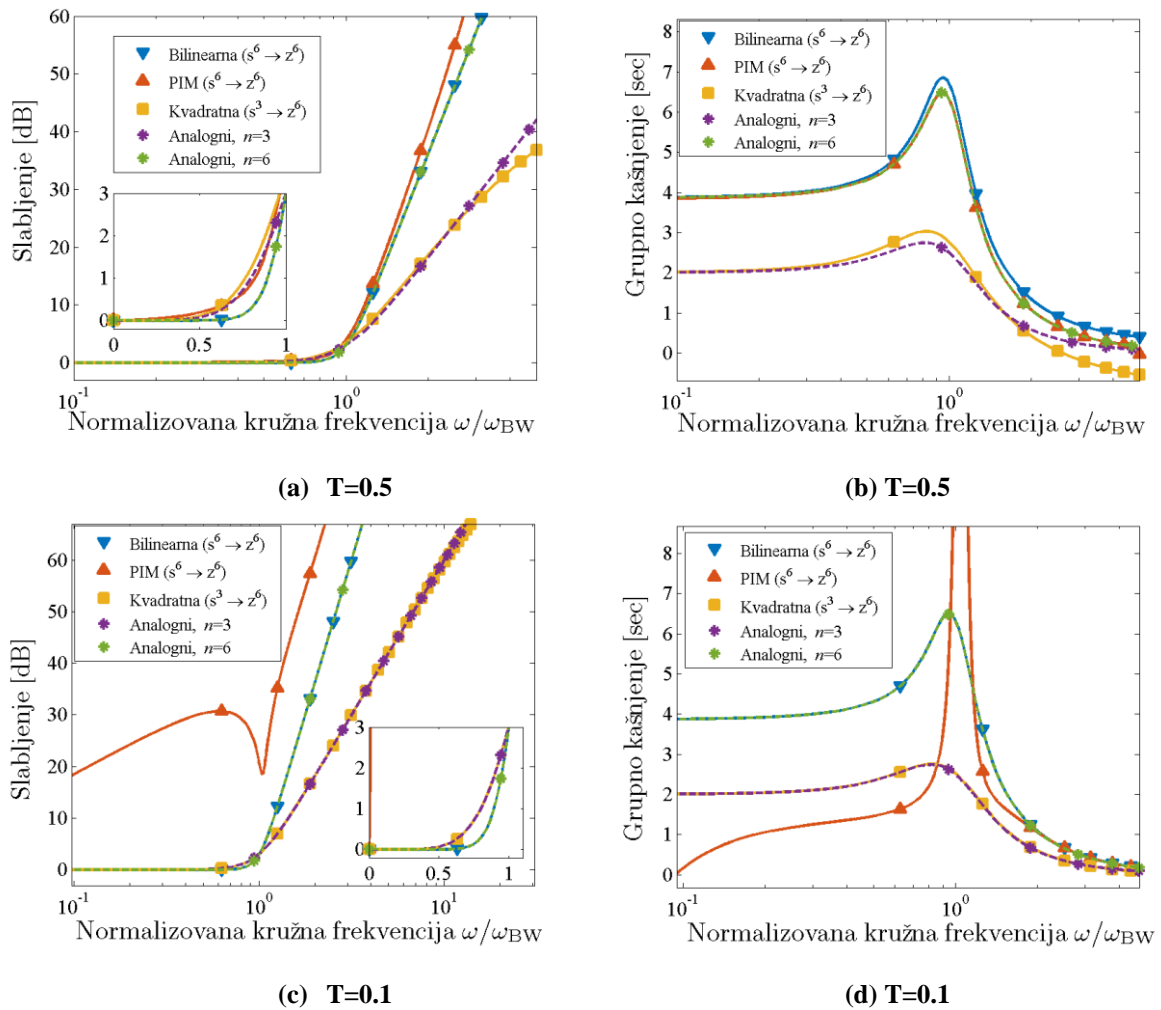
Prema Viner-Hinčinovoj (*Wiener-Khinchin*) teoremi za slučaj diskretne sekvence, spektralna gustina snage odgovara DFT autokorelacione funkcije [Cohe98]. Takođe je poznato da spektralna gustina snage realnih sekvenci odgovara kvadratu modula DFT. Dakle, važe sledeće relacije,

$$\begin{aligned}
 & \xrightarrow{DFT} \\
 r(n) & \rightarrow R(k) \\
 R(k) & = |H(k)|^2.
 \end{aligned}
 \tag{6.22}$$

To praktično znači da se do impulsnog odziva sistema sa minimalnom fazom, $h(n)$, može doći identičnim homomorfim procesiranjem primenjenim kod MIM metoda.

MIM i PIM metode koje su prikazane u ovom poglavlju biće ilustrovane sa dva primera koji koriste PIM metode. Razlog tome je činjenica da je u duhu ove disertacije da demonstrira svojstva IIR filtara dobijenih transformacijom kada se postavljaju zahtevi u pogledu fazne karakteristike gde FIR filtri po svojoj prirodi imaju prednost. Primer koji sledi se odnosi na prototip koji ima selektivnu karakteristiku slabljenja.

Na Sl. 6.4 su date uporedne karakteristike slabljenja i grupnog kašnjenja za tri realizacije IIR filtra na bazi Batervortovog prototipa.



Slika 6.4 Uporedne karakteristike slabljenja i grupnog kašnjenja za tri realizacije IIR filtra dobijenog na osnovu Batervortovog prototipa 6. (Bilinearna i PIM) i 3. (Kvadratna transformacije). Perioda uzorkovanja za (a) i (b) je $T=0.5$ a za (c) i (d) $T=0.1$

Za IIR realizacije na bazi bilinearne transformacije i PIM metoda upotrebljen je analogni prototip šestog reda. U ovom slučaju rezultujući digitalni filter je istog reda kao i polazni analogni prototip tj. šestog reda.

U slučaju realizacije na bazi nove transformacije drugog reda (kvadratna) upotrebljen je prototip trećeg reda. Dobijeni digitalni filter je sada trećeg reda (duplo veći red u odnosu na polazni analogni prototip). Dakle, u digitalnom domenu sve realizacije su istog, šestog, reda pri čemu se transformacijom drugog reda dobija skoro duplo manje grupno kašnjenje u odnosu na realizacije PIM metodom i bilinearnom transformacijom. Naravno, ovaj rezultat je dobijen po cenu lošije karakteristike slabljenja.

Ispitana su dva slučaja za dve različite periode uzorkovanja. U prvom slučaju je izabrana relativno velika perioda uzorkovanja, $T=0.5$ (mala frekvencija uzorkovanja, Sl. 6.4a i b). Posmatranjem slike 6.4a može se uočiti da PIM metod lošije aproksimira karakteristiku slabljenja u odnosu na bilinearnu i kvadratnu transformaciju (što je i očekivano s obzirom na namenu PIM metoda). Suprotno važi za karakteristku grupnog kašnjenja (Sl. 6.4b). Takođe se može uočiti da bilinearna transformacija daje bolju aproksimaciju karakteristike slabljenja u odnosu na kvadratnu transformaciju (u odeljku 4.1 je pokazano da transformacija drugog reda zahteva veću frekvenciju odmeravanja).

Kada se za periodu uzorkovanja uzme relativno mala vrednost (oversemplovanje), $T=0.1$ situacija je značajno drugačija. Na osnovu Sl. 6.4c i d može se videti da bilinearna i kvadratna transformacija verno preslikavaju karakteristike analognog prototipa u digitalni domen. Međutim, sada realizacija na bazi PIM metoda neuspešno aproksimira i karakteristiku slabljenja i karakteristiku grupnog kašnjenja. Na osnovu [Vadh07] može se zaključiti da jedan od razloga loših performansi PIM metoda pri izboru relativno visoke frekvencije uzorkovanja leži u numeričkoj nestabilnosti implementacije PIM metoda (što potencijalno otvara mogućnost za poboljšanja iste). Ovo se posebno odnosi na deo gde se određuje faza na osnovu frekvencijskog odziva. Sa druge strane metod se zasniva na odmeravanju u frekvencijskom domenu i proračunu IDFT, tako da može doći do aliasinga u vremenskom domenu. Generalno, može se zaključiti da PIM metod ispoljava loše karakteristike kada se radi o značajno većim frekvencijama odmeravanja od Nikvistove (što je čest slučaj kod telekomunikacionih interfejsa), dok je transformacija drugog reda uporediva sa bilinearnom.

7 Projektovanje selektivnih IIR digitalnih filtara sa linearnom fazom upotrebom analognih prototipova

U odeljku 2.2 ustanovljeno je da postoje tri načina da se sintetizuje filtarska funkcija IIR filtara (na osnovu analognih filtara uz upotrebu transformacije iz s - u z -domen) koja istovremeno zadovoljava zahteve i u pogledu amplitudske i u pogledu fazne karakteristike:

1. Sinteza prenosne funkcije na osnovu zahteva za linearnom fazom.
2. Sinteza prenosne funkcije na osnovu zahteva za velikom selektivnošću i
3. Sinteza prenosne funkcije koja simultano zadovoljava oba zahteva.

U ovom poglavlju najpre ćemo opisati jedan postupak za sintezu selektivnih filtara koji koristi selektivnu filtarsku funkciju koja se proširuje faznim korektorom (ili, što je isto, korektorom grupnog kašnjenja) kako bi se dobilo konstantno grupno kašnjenje u propusnom opsegu. Koristićemo naziv: "izvorna funkcija"+korektor. Za ovaj postupak razvijen je novi računarski program i njegovim korišćenjem biće generisan primer primene.

Zatim će biti opisan postupak sinteze koji polazi od polinomske prenosne funkcija koja aproksimira konstantno grupno kašnjenje i dodaje joj korektorsku funkciju kojom se obezbeđuje poboljšanje selektivnosti (2. način sinteze IIR funkcije). U ovom slučaju se praktično radi o korekciji amplitudske karakteristike u nepopusnom opsegu sa ciljem da se poveća selektivnost. I za ovaj postupak je razvijen novi program koji će biti upotrebljen za generisanje primera.

Za ilustraciju trećeg metoda biće korišćen gotov primer iz literature koji se odnosi na s -domen.

Rezultati dobijeni pomoću ova tri postupka biće upoređeni i u s - i u z - domenu s tim što će biti korišćena bilinearna transformacija.

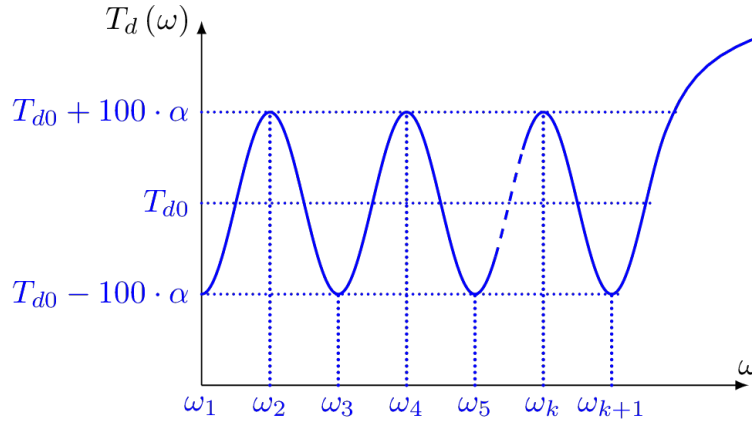
7.1 Sinteza korektora grupnog kašnjenja filtra sa selektivnom amplitudskom karakteristikom

Problem korekcije grupnog kašnjenja je razmatran u literaturi relativno dugo [ŽiLa06]. Bio je razmatran kao problem nelinearne optimizacije [GrTe78] koji će ovde biti rešavan varijantom Remezovog (*Remez*) algoritma [Reme34].

Ako je data prenosna funkcija filtra čiju ćemo zavisnost grupnog kašnjenja od frekvencije da obeležimo sa $T_{df}(\omega)$, treba naći prenosnu funkciju all-pass filtra čije se grupno kašnjenje (ovde obeleženo sa $T_{dc}(\omega)$) sabira sa $T_{df}(\omega)$ tako da ukupno grupno kašnjenje:

$$T_d(\omega) = T_{df}(\omega) + T_{dc}(\omega), \quad (7.1)$$

aproksimira konstantu u Čebiševljevom smislu (*equi-ripple*) sa propisanom greškom od α (%). U stvari, saglasno Sl. 7.1 (za korektor parnog reda), treba iterativno rešiti sledeći sistem nelinearnih jednačina



Slika 7.1 Definicija problema aproksimacije korektora grupnog kašnjenja (slučaj k parno)

$$T_d(\mathbf{p}, \omega_i) = T_{d0} + 100 \cdot \alpha \cdot (-1)^{r+i}, \quad i = 1, 2, \dots, k+1, \quad (7.2)$$

gde je k red korektora, \mathbf{p} je vektor nepoznatih koordinata nula (polova) all-pass korektora (dva broja po kompleksnoj nuli (polu) i jedna dodatna ako je k neparno), T_{d0} je nepoznata (željena) vrednost konstantnog grupnog kašnjenja u propusnom opsegu, a $r=1$ za k neparno i $r=0$ za k -parno.

Treba imati na umu da je, pošto su frekvencije ω_i , $i=1, 2, \dots, k+1$ nepoznate, u svakoj iteraciji korekcije \mathbf{p} , potrebno da se odrede njihove vrednosti. Za realizaciju ovog postupka potrebno je početno rešenje \mathbf{p}^0 što, u opštem slučaju, može predstavljati prepreku za njegovu primenu. Ako se, međutim, primeni tzv. prigušenje iteracija [LiZw96], odnosno ako se na početku iterativnog postupka prethodnoj vrednosti \mathbf{p}^m dodaje samo deo korekcije $\Delta \mathbf{p}^m$, kao početno rešenje za kompleksne nule (polove) može da se koristi sledeći izraz

$$p_l = \sigma_l + j\omega_l = -\frac{1}{k+n} \pm j \frac{2 \cdot l - 1}{k+1}, \quad l = 1, 2, \dots, \lfloor k/2 \rfloor, \quad (7.3)$$

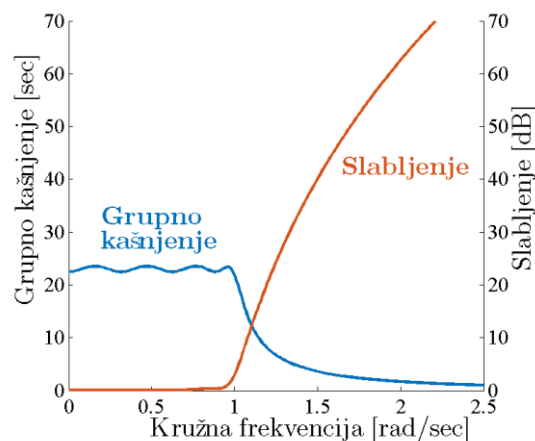
a za realni pol (ako je k neparno) može se uzeti $p_k = -1/(k+n)$, gde je n , kao i ranije, red izvornog filtra. Pored toga, kao početna vrednost, može se koristiti $(T_d)^0 = (n+2 \cdot k) \cdot \pi/2$. U

gornjim izrazima [] je upotrebljeno da se iskaže funkcija “najmanje celo od” (zaokruživanje ka nuli).

Primena ovog algoritma biće ilustrovana na sintezi all-pass korekora 6. reda sa greškom grupnog kašnjenja $\alpha=2.5\%$ kojim će biti korigovana karakteristika grupnog kašnjenja LSM filtra 8. reda.

Nule i polovi novodobijene prenosne funkcije dati su u prvoj koloni Tabele 7.1. Sl. 7.2 ilustruje karakteristiku grupnog kašnjenja i karakteristiku slabljenja dobijenog rešenja u s -domenu. Za $a_{\max}=50\text{dB}$ izračunata je selektivnost $S=1.43$ što znači da je širina prelazne oblasti toliko puta uža od širine propusnog opsega.

Ovako dobijena prenosna funkcija LSM+korektor transformisana je u z -domen pri čemu je uzeto da je granična frekvencija filtra $f_g=20\text{kHz}$, a da je frekvencija uzorkovanja $f_s=800\text{kHz}$. Rezultati transformacije u z -domen dati su u prvoj koloni Tabele 7.2. Grafički prikaz preslikanih rezultata ostavljen je za kasnije kako bi se postigla veća kompaktnost teksta.



Slika 7.2 Karakteristike slabljenja i grupnog kašnjenja LSM filtra 8. reda sa korektorom 6. reda. Greška grupnog kašnjenja je 2.5%

Tabela 7.1 Nule i polovi u s-domenu selektivnih filtara sa linearnom fazom (nule i polovi korektora grupnog kašnjenja su dati podebljanim (*bold*) kurzivom)

	LSM filtar sa korektorom grupnog kašnjenja ($n=6, m=14$)	Equi-ripple grupno kašnjenje sa amplitudskim korektorom ($n=12, m=14$)	Simultana aproksimacija amplitude i grupnog kašnjenja ($n=11, m=13$)
Nule	0.1992951 ± j0.7877623	± j3.1335590	0.1588429 ± j0.8789036
	0.2148503 ± j0.4730444	± j3.7630714	0.1657072 ± j0.5893035
	0.2174384 ± j0.1580652	± j4.7859023	0.1674894 ± j0.2950981
		± j6.0011218	0.0000000 ± j1.2699075
		± j7.3087231	0.0000000 ± j1.6118200
		± j8.6935655	0.1682225
Polovi	-0.1992951 ± j0.7877623	-0.9342617 ± j0.6731098	-0.1588429 ± j0.8789036
	-0.2148503 ± j0.4730444	-0.9320231 ± j0.2017198	-0.1657072 ± j0.5893035
	-0.2174384 ± j0.1580652	-0.9267569 ± j0.3354389	-0.0842760 ± j1.0947876
	-0.5065306 ± j0.4900545	-0.9165233 ± j0.4677895	-0.3512084 ± j1.0332394
	-0.6322950 ± j1.6699013	-0.8960402 ± j0.5975705	-0.7817460 ± j0.5441479
	-0.0947391 ± j0.9814274	-0.8473631 ± j0.7221984	-0.1670604 ± j0.2943957
	-0.2938454 ± j0.7747231	-0.6810420 ± j0.8359018	-0.1664136

Tabela 7.2 Nule i polovi u z-domenu selektivnih filtara sa linearnom fazom (Granična frekvencija 20 kHz, frekvencija uzorkovanja 800 kHz)

	LSM filtar sa korektorom grupnog kašnjenja ($n=14, m=14$)	Equi-ripple grupno kašnjenje sa amplitudskim korektorom ($n=14, m=14$)	Simultana aproksimacija amplitude i grupnog kašnjenja ($n=13, m=13$)
Nule	1.0238076 ± j0.1272055	-0.3640670 ± j0.9313727	0.9684544 ± j0.2491907
	1.0314267 ± j0.0767687	-0.5043186 ± j0.8635177	0.9803005 ± j0.1975118
	1.0344240 ± j0.0256950	-0.6364615 ± j0.7713085	1.0154186 ± j0.1408797
	-1	-0.7524050 ± j0.6587008	1.0219261 ± j0.0948164
	-1	-0.8393336 ± j0.5436168	1.0255426 ± j0.0475717
	-1	-0.8857789 ± j0.4641074	1.0267781
	-1	-1	-1
	-1	-1	-1
	-1	-1	-1
	-1	-1	-1
Polovi	0.9050808 ± j0.0238037	0.3674110 ± j0.8521465	0.8812569 ± j0.0757489
	0.9208466 ± j0.0711024	0.4617671 ± j0.7773966	0.9342501 ± j0.1527518
	0.9479937 ± j0.1158551	0.5671968 ± j0.6871740	0.9662169 ± j0.1340535
	0.9618968 ± j0.1195133	0.6695843 ± j0.5722168	0.9701924 ± j0.0900164
	0.9641894 ± j0.0717642	0.7582639 ± j0.4317912	0.9724571 ± j0.1684854
	0.9661254 ± j0.0239985	0.8238372 ± j0.2692425	0.9730704 ± j0.0450301
	0.9736743 ± j0.1510094	0.8587697 ± j0.0915481	0.9741971

7.2 Sinteza korektora amplitudske karakteristike polinomskih filtra sa linearnom faznom karakteristikom

U ovom odeljku biće opisan algoritam i rezultati njegove primene za sintezu amplitudskog korektora polinomskih filtarskih funkcija sa linearnom faznom karakteristikom.

Prenosna funkcija data sa (2.4) biće ovde prepisana sa modifikacijom koja na eksplicitan način prikazuje amplitudsku korekciju

$$H_s(s) = \frac{a_0 \cdot \prod_{i=1}^{\lfloor n/2 \rfloor} (1 + s^2 / \omega_i^2)}{P_n(s)} = F(s) \cdot C(s) \quad (7.4)$$

gde je

$$F(s) = \frac{a_0}{P_n(s)} \quad (7.4a)$$

izvorna prenosna funkcija polinomskog tipa, $P_n(s)$ je polinom dobijen nekim postupkom aproksimacije konstantnog grupnog kašnjenja, a

$$C(s) = \prod_{i=1}^{\lfloor n/2 \rfloor} (1 + s^2 / \omega_i^2) \quad (7.4b)$$

je korektorska funkcija.

Slabljenje ranije definisano za opšti slučaj sa (2.1d), za s -domen se definiše kao

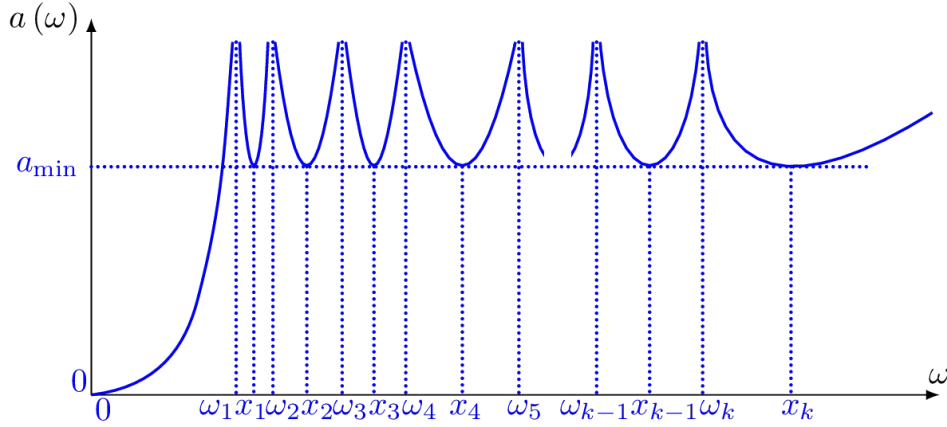
$$a(\omega) = -10 \cdot \log \left\{ H_s(s) \cdot H_s(-s) \Big|_{s=j\omega} \right\} = -10 \cdot \log \left\{ \left[|H_s(s)|_{s=j\omega} \right]^2 \right\}. \quad (7.5)$$

U slučaju korigovane amplitudske karakteristike (smenom (7.4)) slabljenje je dato sa

$$a(\omega) = F(\omega^2) - \sum_{i=1}^{\lfloor n/2 \rfloor} 10 \cdot \log \left(1 - \omega^2 / \omega_i^2 \right)^2 \quad (7.6)$$

gde je

$$F(\omega^2) = -10 \cdot \log \left\{ \left[F(s) \cdot F(-s) \Big|_{s=j\omega} \right] \right\}. \quad (7.7)$$



Slika 7.3 Ilustracija postupka sinteze amplitudskog korektora

Imajući u vidu Sl. 7.3 nova filtarska funkcija treba da zadovolji sledeći sistem nelinearnih jednačina

$$a(x_k) = a_{\min} \quad k = 1, 2, \dots, \lfloor n/2 \rfloor, \quad (7.8)$$

Odnosno

$$g_k(\boldsymbol{\omega}, x_k) = F(x_k^2) - \sum_{i=1}^{\lfloor n/2 \rfloor} 10 \cdot \log \left(1 - x_k^2 / \omega_i^2 \right)^2 - a_{\min} = 0 \quad k = 1, 2, \dots, \lfloor n/2 \rfloor, \quad (7.9)$$

gde je $\boldsymbol{\omega} = [\omega_1 \ \omega_2 \ \dots \ \omega_{\lfloor n/2 \rfloor}]$ vektor nepoznatih koordinata nula prenosa, a $\mathbf{x} = [x_1 \ x_2 \ \dots \ x_{\lfloor n/2 \rfloor}]$ vektor izračunatih koordinata (na ω osi) minimuma slabljenja van propusnog opsega.

Za dato početno rešenje $\boldsymbol{\omega}^0$, postupak (varijanta Remezovog algoritma) određivanja $\boldsymbol{\omega}$ je:

1. Odrede se koordinate minimum slabljenja, \mathbf{x} .
2. Reši se sistem jednačina

$$\sum_{r=1}^{\lfloor n/2 \rfloor} \frac{\partial g_k(\boldsymbol{\omega}, x_k)}{\partial \omega_r} \cdot \Delta \omega_r = g_k(\boldsymbol{\omega}, x_k) \quad k = 1, 2, \dots, \lfloor n/2 \rfloor$$

po priraštajima koordinata nula prenosa i

3. koriguju se koordinate nula prenosa kao

$$\boldsymbol{\omega}^{m+1} = \boldsymbol{\omega}^m + \Delta \boldsymbol{\omega}_r$$

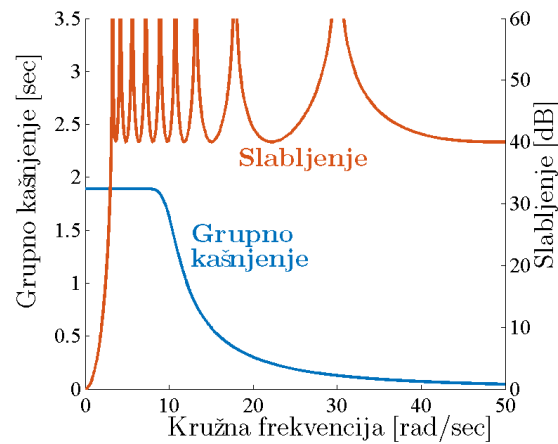
gde je m brojač iteracija. Po potrebi primenjuje se prigušenje iteracija [LiZw96] kao

$$\boldsymbol{\omega}^{m+1} = \boldsymbol{\omega}^m + \alpha \cdot \Delta \boldsymbol{\omega}_r$$

gde je $\alpha < 1$.

Primena ovog postupka biće ilustrovana rezultatima korekcije amplitudskih karakteristika dvaju filtara.

Najpre, sintetizovan je polinomski filter koji aproksimira konstantno grupno kašnjenje u maksimalno ravnom smislu odnosno Beselov (*Bessel*) filter [Thom49]. Izabran je red filtra $n=20$.



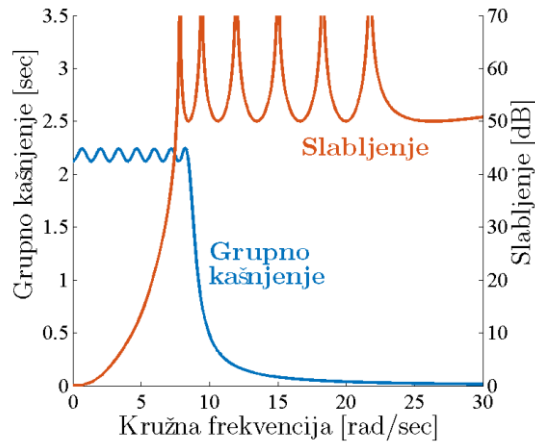
Slika 7.4 Karakteristika slabljenja i grupnog kašnjenja Besselovog filtra 20. reda sa korigovanom amplitudskom karakteristikom ($k=9$)

Njegova amplitudska karakteristika je korigovana proširenjem od $k=9$ nula prenosa. Novodobijena prenosna funkcija je renormalizovana tako da slabljenje na graničnoj frekvenciji iznosi 3 dB. Nule i polovi ove funkcije dati su u Tabeli 7.3. Slika 7.4 sadrži karakteristiku grupnog kašnjenja i karakteristiku slabljenja novodobijenog filtra u s -domenu.

Tabela 7.3 Nule i polovi u s -domenu Besselovog filtra 20. reda sa korigovanom amplitudskom karakteristikom ($k=9$)

Nule	Polovi
0.000000 ± j2.936770	-7.088627 ± j1.376966
0.000000 ± j3.768226	-7.170455 ± j0.458605
0.000000 ± j5.034722	-6.668299 ± j3.227307
0.000000 ± j6.486311	-6.317271 ± j4.165709
0.000000 ± j8.023782	-6.922797 ± j2.298927
0.000000 ± j9.654264	-5.268014 ± j6.095311
0.000000 ± j11.844577	-5.857201 ± j5.119151
0.000000 ± j16.038776	-3.533336 ± j8.181951
0.000000 ± j27.032032	-2.151555 ± j9.392526
	-4.515493 ± j7.107362

Drugi primer će poslužiti za poređenje ovog postupka sa postupkom opisanim u prethodnom odeljku. Najpre je algoritmom sličnim onim koji je opisan u prethodnom odeljku sintetizovan polinomski filter 14. reda koji aproksimira konstantno grupno kašnjenje sa greškom od 2.5%. Zatim je njegova amplitudska karakteristika korigovana tako da minimalno slabljenje u nepropusnom opsegu iznosi 50 dB pri čemu je upotrebljeno $k=6$. Nule i polovi novodobijene filtarske funkcije dati su u Tabeli 7.1 (druga kolona). Slika 7.5 predstavlja frekvencijski odziv u s -domenu ovog rešenja.

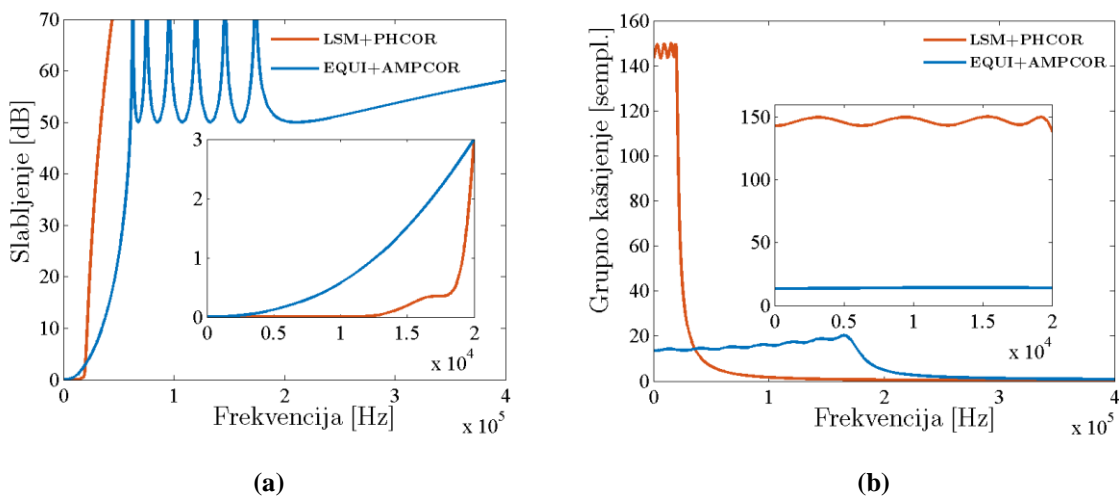


Slika 7.5 Karakteristika slabljenja i grupnog kašnjenja filtra sa equi-ripple aproksimacijom grupnog kašnjenja i korigovanom amplitudskom karakteristikom ($k=6$)

Selektivnost ovog rešenja je $S=0.489$ (za minimalno slabljenje od $a_{\min}=50\text{dB}$) što znači da je širina prelazne zone više nego dva puta veća od širine propusnog opsega.

Ovako dobijena prenosna funkcija, sa equi-ripple aproksimacijom konstantnog grupnog kašnjenja, čija je amplitudska karakteristika korigovana, transformisana je u z -domen pri čemu je, kao i ranije, uzeto da je granična frekvencija filtra $f_g=20\text{kHz}$, a frekvencija uzorkovanja $f_s=800\text{kHz}$. Rezultati transformacije dati su u drugoj koloni Tabele 7.2.

Karakteristike filtera koji aproksimiraju konstantno grupno kašnjenje u Čebiševljevom smislu i čije su nule i polovi dati u prve dve kolone Tabele 7.1, nakon transformacije u z -domen, prikazane su na Sl. 7.6. Radi lakšeg poređenja na Sl. 7.6a su date karakteristike slabljenja rešenja LSM+korektor (odjeljak 7.1) i aktuelnog rešenja (Sl. 7.5).



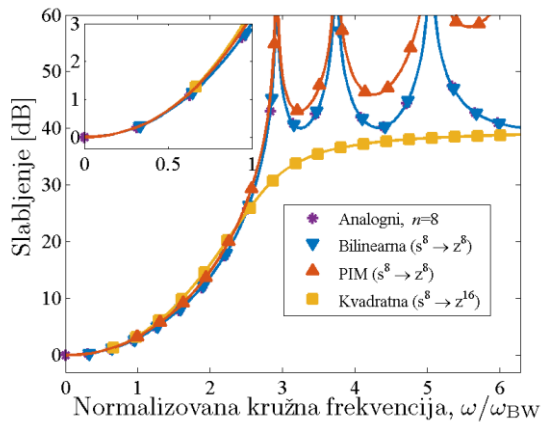
Slika 7.6 Karakteristike slabljenja (a) i karakteristike grupnog kašnjenja (b) digitalnih filtera koji aproksimiraju konstantno grupno kašnjenje u Čebiševljevom smislu: LSM+korektor (LSM+PHCOR) i Equi-ripple grupno kašnjenje sa korigovanom amplitudskom karakteristikom (EQUI+AMPCOR).

Na Sl. 7.6b su date uporedne karakteristike grupnog kašnjenja za oba rešenja u z -domenu. Ponovo je uzeto da je granična frekvencija filtra $f_g=20\text{kHz}$, a frekvencija uzorkovanja $f_s=800\text{kHz}$.

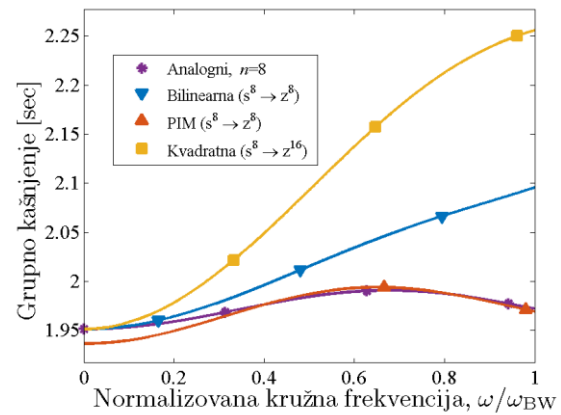
U cilju poređenja bilinearne transformacije, transformacije drugog reda (kvadratna) i PIM metoda, za slučaj filtarskih prototipova koji aproksimiraju fazu, realizovana su tri IIR filtra na bazi analognog prototipa čije su nule i polovi dati u Tabeli 7.4.

Tabela 7.4 Nule i polovi u s -domenu analognog prototipa 8. reda sa nulama prenosa koji aproksimira karakteristiku grupnog kašnjenja u Čebiševljevom smislu sa equ-ripple greškom od $\pm 1\%$.

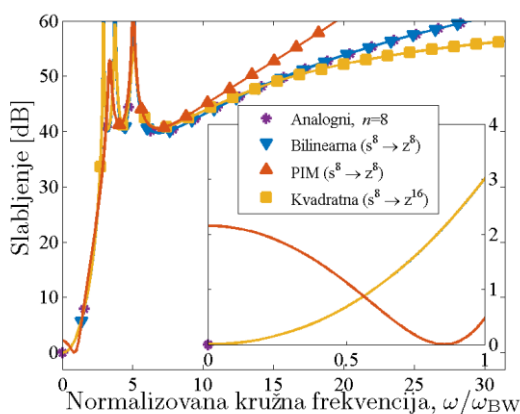
Nule	Polovi
$0.000000 \pm j2.917597$	$-1.177311 \pm j0.681299$
$0.000000 \pm j3.738559$	$-1.152160 \pm j2.029670$
$0.000000 \pm j5.046967$	$-1.077362 \pm j3.327213$
	$-0.837877 \pm j4.516940$



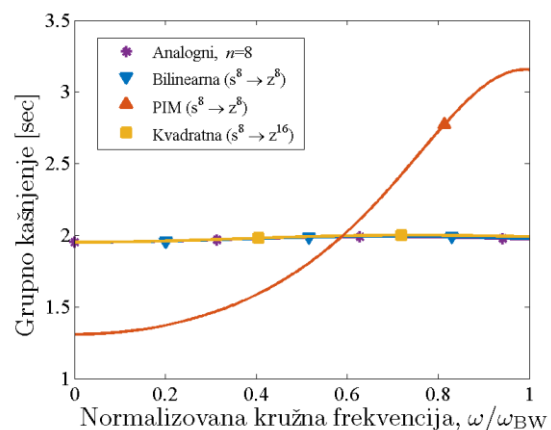
(a) $T=0.5$



(b) $T=0.5$



(c) $T=0.1$



(d) $T=0.1$

Slika 7.7 Uporedne karakteristike slabljenja i grupnog kašnjenja za tri realizacije IIR filtra dobijenog na osnovu prototipa 8. reda sa nulama prenosa koji aproksimira karakteristiku grupnog kašnjenja u Čebiševljevom smislu sa greškom od $\pm 1\%$. Perioda odmeravanja za slike (a) i (b) je $T=0.5$, a za (c) i (d) $T=0.1$

Prototip je 8. reda ($n=8$) sa 6 nula prenosa (transmission zeros, $m=6$). Na Sl. 7.7 su date uporedne karakteristike slabljenja i grupnog kašnjenja za tri realizacije IIR filtera. Pošto se sada radi o granično minimalno faznom sistemu PIM metod poprilično dobro aproksimira karakteristiku slabljenja u propusnom opsegu za $T=0.5$ (Sl. 7.7a), dok van propusnog opsega unosi dodatno slabljenje na frekvencijama različitim od frekvencija nula prenosa. Posmatranjem Sl. 7.7b može se uočiti da PIM metod veoma dobro aproksimira karakteristiku grupnog kašnjenja pri relativno niskim frekvencijama uzorkovanja što nije slučaj kod bilinearne i kvadratne transformacije. Kao što je ranije pokazano (odjeljak 4) kvadratna transformacija ispoljava loše performanse pri niskim frekvencijama uzorkovanja što se može uočiti i u ovom primeru.

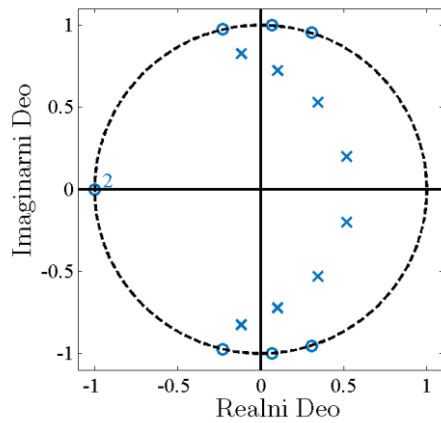
Kada se izabere frekvencija uzorkovanja značajno veća od Nikvistove, Sl. 7.7c i d, bilinearne i kvadratna transformacija dobro aproksimiraju i karakteristiku slabljenja i karakteristiku grupnog kašnjenja. U ovom slučaju PIM metod donekle uspešno aproksimira karakteristiku slabljenja van propusnog opsega, dok je aproksimacija u propusnom opsegu neprihvatljiva. Na osnovu Sl. 7.7d jasno je da PIM metod neuspešno aproksimira karakteristiku grupnog kašnjenja pri većim frekvencijama uzorkovanja.

Na Sl. 7.8 je prikazan položaj nula i polova za tri razmatrane realizacije IIR filtra. Sva tri načina preslikavanja rezultuju stabilnim sistemima u digitalnom domenu. Na osnovu Sl. 7.8a i b može se zaključiti da bilinearne transformacije pravilno preslikava nule prenosa u z -domen nezavisno od vrednosti periode uzorkovanja. Nule prenosa se nalaze tačno na jediničnom krugu u z -ravni (na koji se inače i preslikava imaginarna osa iz s -ravni). Međutim, kod kvadratne transformacije i PIM metoda to nije uvek slučaj.

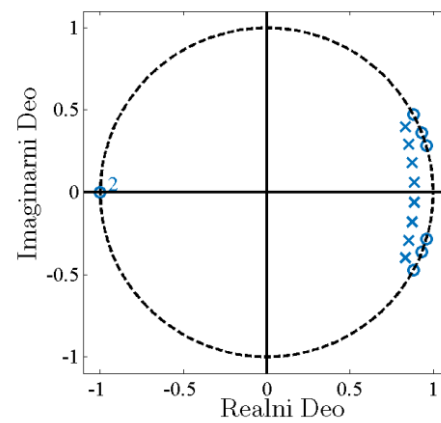
Za $T=0.5$ kvadratna transformacija preslikava sve nule prenosa unutar jediničnog kruga što rezultuje konačnim slabljenjem van propusnog opsega (Sl. 7.8c i Sl. 7.7a). Kod PIM metoda nule prenosa se slikaju na jedinični krug (kao i u slučaju bilinearne) što objašnjava relativno dobru aproksimaciju karakteristike slabljenja (Sl. 7.8e i Sl. 7.7a).

Situacija je značajno drugačija pri većoj frekvenciji uzorkovanja, tj. za $T=0.1$. Sada kvadratna transformacija daje isti rezultat kao i bilinearne. Na Sl. 7.8d se može videti da kvadratna transformacija proizvodi dve konstelacije nula i polova. Jedna je identična onoj koja se dobija bilinearnom transformacijom, a druga predstavlja recipročnu sliku unutar jediničnog kruga (uvećani detalj na Sl. 7.8d) čime je zagarantovana stabilnost. Pri ovakvim

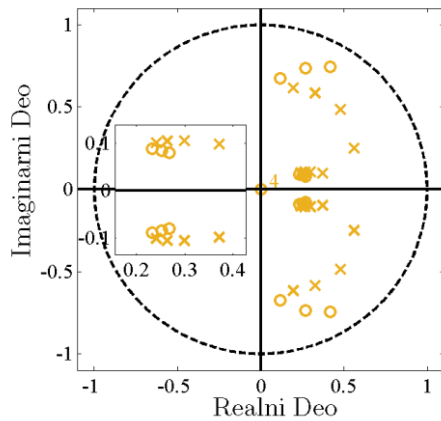
uslovima uzorkovanja, PIM metod dve od šest nula prenosa preslikava na negativni deo realne ose u z -ravni (Sl. 7.8f).



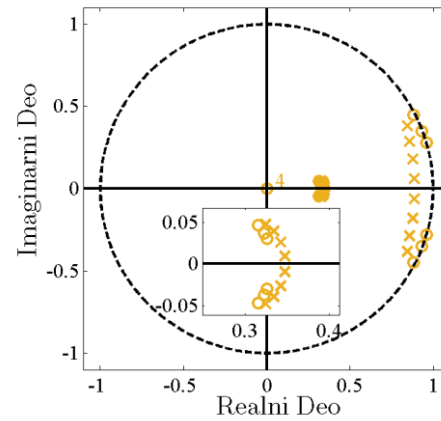
(a) Bilinearna, $T=0.5$



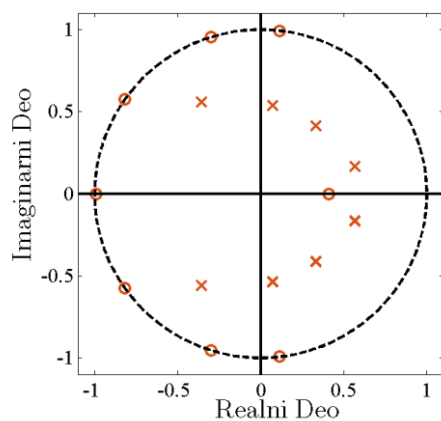
(b) Bilinearna, $T=0.1$



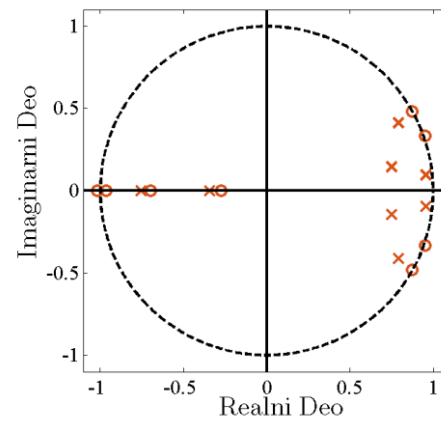
(c) Kvadratna, $T=0.5$



(d) Kvadratna, $T=0.1$



(e) PIM, $T=0.5$



(f) PIM, $T=0.1$

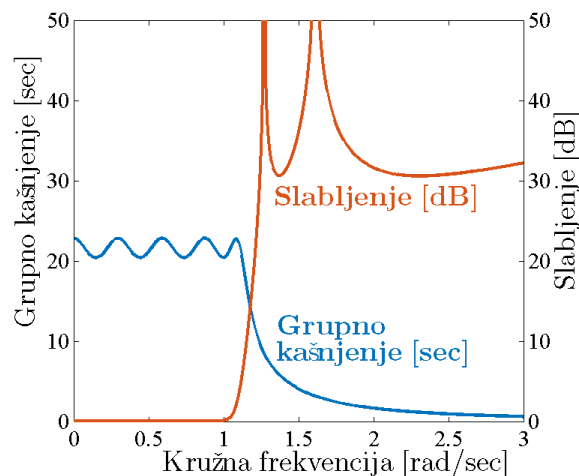
Slika 7.8 Položaj nula i polova u z -domenu za tri realizacije IIR filtra dobijenog na osnovu prototipa 8. reda sa nulama prenosa koji aproksimira karakteristiku grupnog kašnjenja u Čebiševljevom smislu sa greškom od $\pm 1\%$. Perioda uzorkovanja za slike (a), (c) i (e) je $T=0.5$, a za (b), (d) i (f) $T=0.1$

Slično važi i za dva pola koji gotovo kompenzuju nule na negativno delu realne ose u z -ravni čineći da slabljenje bude konačno u okolini ovih nula (Sl. 7.7c). Posledica ovakvog preslikavanja je distorzija kako karakteristike slabljenja tako i karakteristike grupnog kašnjenja.

7.3 Simultana sinteza amplitudske karakteristike i karakteristike konstantnog grupnog kašnjenja

U [LiMi83] je opisan metod koji omogućava da se sintetizuje prenosna funkcija koja se sastoji iz dva dela. Prvi deo je funkcija niskopropusnog tipa koja istovremeno dozvoljava nule prenosa u beskonačnosti, na osi stvarnih frekvencija kao i u desnoj poluravni, ravni kompleksne frekvencije s . Drugi deo je *all-pass* tipa. Suština opisanog postupka je u tome da se nule i polovi obeju funkcija smatraju jedinstvenim vektorom nepoznatih i istovremeno određuju iz niza uslova: oblika karakteristike slabljenja u propusnom opsegu, oblika karakteristike slabljenja u nepropusnom opsegu i konstantnog grupnog kašnjenja u propusnom opsegu određenog iz uslova Čebiševljeve aproksimacije.

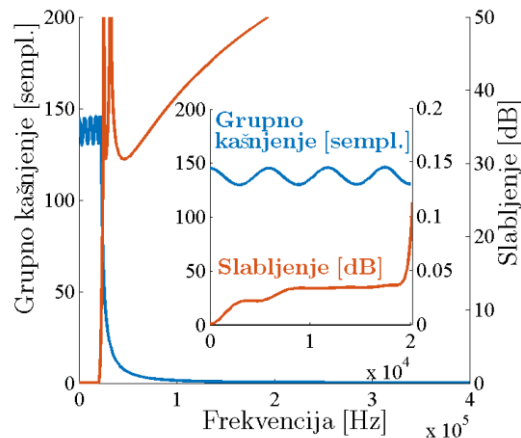
Nule i polovi jednog rešenja gde se zahtevala monotona karakteristika slabljenja u propusnom opsegu, minimalno slabljenje od 30dB u nepropusnom opsegu i konstantno grupno kašnjenje sa greškom od 5% dati su u Tabeli 7.1 (Treća kolona). Odgovarajuće karakteristike slabljenja i grupnog kašnjenja date su na Sl. 7.9.



Slika 7.9 Karakteristike slabljenja i grupnog kašnjenja (s -domen) filtra 13. reda koji simultano aproksimira amplitudsku i karakteristiku grupnog kašnjenja.

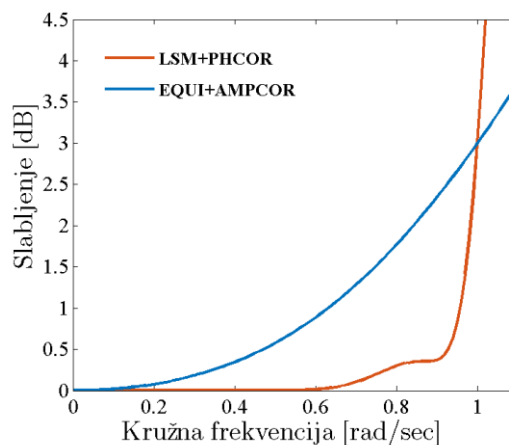
I ova prenosna funkcija je preslikana u z -domen sa $f_g=20\text{kHz}$ i $f_s=800\text{kHz}$. Rezultujuće karakteristika slabljenja i fazna karakteristika prikazane su na Sl. 7.10.

Razmatrajući rezultate prikazane u ovom odeljku a koji se odnose na sintezu prenosnih funkcija selektivnih IIR filtara sa linearnom fazom, možemo da izvedemo sledeće zaključke.



Slika 7.10 Karakteristika slabljenja (z-domen) filtra 13. reda koji je dobijen preslikavanjem odgovarajućeg prototipa iz s-domena.

Najpre, upotreba filtarskih funkcija koje polaze od polinomskih prototipova sa linearnom faznom karakteristikom, bilo da se radi o maksimalno ravnoj ili Čebiševljevoj aproksimaciji grupnog kašnjenja, dovodi do rešenja sa izuzetno lošom amplitudskom karakteristikom u propusnom opsegu. To se može lako zaključiti sa Sl. 7.11 na kojoj su date karakteristike slabljenja u propusnom opsegu za filtre čije su nule i polovi dati u prvoj i drugoj koloni Tabele 7.1.



Slika 7.11 Karakteristike slabljenja u propusnom opsegu (s-domen) filtarskih funkcija prikazanih na Sl. 7.6. (LSM+PHCOR) LSM+korektor i (EQUI+AMPCOR) Equi-ripple grupno kašnjenje sa korigovanom amplitudskom karakteristikom.

Drugo, upotreba filtarskih funkcija koje polaze od polinomskih prototipova sa linearnom faznom karakteristikom (Besel) zahteva izuzetno visoke redove filtara kako bi se postigla selektivnost mada se vrednost minimalnog slabljenja u nepropusnom opsegu može slobodno kontrolisati bez ograničenja. Primer filtra 20. reda koji je dat na Sl. 7.4 odlično ilustruje ovu tvrdnju. Njegova selektivnost je $S=0.55$ (za minimalno slabljenje od $a_{\min}=40\text{dB}$) što znači

da i pored izuzetno visokog reda filtra širina prelazne oblasti je još uvek skoro dva puta veća od širine propusnog opsega.

Fazna karakteristika filtarskih funkcija koje polaze od polinomskih prototipova sa linearnom faznom karakteristikom je linearna u opsegu koji je daleko širi od propusnog opsega što govori o redundantnosti po ovom aspektu aproksimacije. Pošto je opseg aproksimacije linearne faze proporcionalan redu filtra, on se ne može suziti, a da se pritom selektivnost još više ne pogorša.

S druge strane, upotrebom selektivnih filtarskih funkcija (polinomskih ali i onih sa bilo kakvim nulama prenosa) može se postići i velika selektivnost, i željeno potiskivanje susednih kanala (slabljenje u nepropusnom opsegu) ali i linearna faza u celom propusnom opsegu uz razumnu složenost celokupnog rešenja.

8 Zaključak

Sa većom primenom portabilnih elektronskih aparata uključujući mobilnu telefoniju, personalne računare i tablete, kao i dislocirane telekomunikacione uređaje i satelitsku elektroniku i njene telekomunikacione podsisteme, problem potrošnje integrisanih kola postaje sve značajniji.

Ranije je sopstvena potrošnja odnosno grejanje komponente bila razmatrana samo u kontekstu stepena integracije s obzirom da veća gustina pakovanja otežava odvođenje toplote. Danas, međutim, potrošnja određuje i vreme upotrebe bez ponovnog punjenja baterija uređaja u koji je komponenta ugrađena.

Sve to govori da je primena rešenja koja dovode do manje potrošnje od vitalnog značaja za dalji razvoj integrisane elektronike.

Ova disertacija ima za cilj da upotpuni istraživanja koja se odnose na nalaženje takvih rešenja.

Naime, predlaže se upotreba IIR digitalnih filtara koji se dobijaju transformacijama iz analognih prototipova, a omogućavaju zadovoljavanje svih funkcionalnih zahteva uz

1. minimizaciju potrošnje i
2. minimizaciju površine čipa odnosno količine silicijuma odnosno cene proizvodnje.

Najpre su razmatrani osnovni pojmovi koji karakterišu jednu filtarsku funkciju i uveden je kvantitativni pokazatelj selektivnosti filtra. Naime, naglašava se da je selektivnost kritični parameter filtra jer filter koji nije selektivan može da ima primene isključivo u nekim veoma specifičnim situacijama.

Izboru tipa implementacije digitalnog filtra, odnosno izboru između FIR i IIR digitalnih filtara je posvećena posebna pažnja. Pokazano je da se, pri jednakoj složenosti, primenom IIR filtara mogu jednako zadovoljiti zahtevi u pogledu fazne karakteristike u propusnom opsegu filtra, a da se zahtevi u pogledu amplitudske ispunjavaju neuporedivo bolje nego kod FIR filtara, što naročito važi za selektivnost. Analiza je sprovedena i sa stanovišta maksimalne brzine rada pri čemu je pokazano da je latencija FIR filtara neuporedivo veća.

Smatrajući problem stabilnosti kritičnim za primenu IIR filtara, posebna pažnja je posvećena studiji ovog problema i pokazano je da se upotrebom pogodne hardverske implementacije ovaj problem može u potpunosti prevazići. Posebna pažnja posvećena je analizi uticaja konačne dužine digitalne reči na stabilnost IIR filtara kao i na konačna svojstva rešenja sa stanovišta ispunjavanja projektantskih zahteva u pogledu amplitudske i fazne

karakteristike. Rezultati ovakvih analiza prate praktično sva rešenja koja se nude u okviru teze.

Polazeći od ovih zaključaka pristupilo se istraživanju transformacije koja će na najpogodniji način da preslika originalnu (prototipsku) funkciju iz kontinualnog (analognog) u diskretni (digitalni) domen. Pored postojećih kao što su bilinearna, MIM i PIM transformacija, predložena je i jedna nova koja kao osnovu koristi integraciono pravilo drugog reda tzv. Gir-Šičmanovu (*Gier-Schichman*) formulu. Tokom celog rada nova filtarska rešenja koja se uvode su sistematski korišćena za istraživanje svojstava ovih transformacija i ustanovljavanju prednosti i nedostataka pojedinačne primene svake od njih.

Teza nudi nekoliko novih rešenja filtarskih funkcija.

Najpre, po prvi put je ponuđena digitalna verzija tzv. CMAC filtara. Radi se o selektivnim polinomskim filtrima koji ispoljavaju monotonu amplitudsku karakteristiku u propusnom opsegu. Posle primene preslikavanja u digitalni domen data je analiza svojstava ove klase filtara u frekvencijskom i vremenskom domenu u obliku komparativne studije njenih podklasa. Da bi studija ove klase filtara bila kompletna, sintetizovan je filter propusnik opsega i analizirana su njegova svojstva sa više stanovišta uključujući frekvenciju uzorkovanja, tip transformacije u diskretni domen i dužinu digitalne reči. Analiza je sprovedena i u frekvencijskom i u vremenskom domenu. Ovaj filter je upoređen sa FIR filtrom koji bi zadovoljavao slične zahteve kako bi se pokazala dramatična razlika u složenosti u korist IIR CMAC filtra.

Poređenje filtarskih funkcija filtara prigušnika opsega omogućilo je sagledavanje svojstava rešenja ne samo u domenu karakteristika filtarskih funkcija već i u domenu fizičke realizacije u obliku CMOS integrisanog kola. Pri tome su razvijena (isprojektovana integrisana kola) tri rešenja: jedno FIR tipa, drugo IIR CMAC tipa sa upotrebom formule drugog reda za preslikavanje u diskretni domen i treće IIR tipa sa upotrebom bilinearne transformacije (trapeznog pravila) za preslikavanje u diskretni domen. Pokazano je da treće rešenje nedvosmisleno vodi ka najmanjoj površini i potrošnji čipa. U stvari, prednosti su više nego značajne.

Poslednja tema kojom se teza bavi je sinteza selektivnih filtara sa linearnom fazom. Ovde su ponuđena tri rešenja IIR filtarskih funkcija i to:

1. Funkcije koje aproksimiraju linearnu fazu kao originalni polinomski filteri, a kojima je prenosna funkcija proširena tako da se poveća selektivnost a sačuva linearnost faze. Kaže se da je korišćena amplitudska korekcija. Za generisanje originalnih polinomskih filtara sa

linearnom fazom razvijeni su programi koji fazu aproksimiraju u maksimalno ravnom i Čebiševljevom smislu.

2. Funkcije koje ispoljavaju selektivnu amplitudsku karakteristiku čija se prenosna funkcija proširuje faznim korektorom tako da se amplitudska karakteritika ne remeti, a koriguje se fazna karakteristika u propusnom opsegu do željene greške u Čebiševljevom smislu. Za ovu svrhu razvijen je program koji određuje položaj nula i polova faznog korektora.

3. Funkcije koje simultano aproksimiraju amplitudsku i faznu karakteristiku. U ovom slučaju korišćeni su rezultati iz literature.

Za prvi slučaj je pokazano da se prilikom aproksimacije fazne karakteristike dobija aproksimacioni opseg koji je nepotrebno znatno širi od propusnog opsega filtra što ima za posledicu ograničenu selektivnost. Zaključeno je da ovaj tip aproksimacije, mada bolji, najviše podseća na FIR aproksimaciju sa linearnom fazom.

U ostala dva slučaja dobijaju se veoma selektivna rešenja koja aproksimiraju linearnu fazu u celom propusnom opsegu sa željenom greškom. Analiziran je uticaj tipa transformacije u diskretni domen uključujući i frekvenciju uzorkovanja što je dovelo do preporuka o primeni u konkretnim situacijama. Ovi se tipovi preporučuju kao konačna rešenja.

Kao opšti zaključak istraživanja koja su sprovedena u okviru Laboratorije za projektovanje Elektronskih kola Elektronskog Fakulteta u Nišu (LEDA) može se tvrditi da se primenom IIR filtarskih funkcija mogu zadovoljiti strogi zahtevi u pogledu aproksimacije amplitudske i fazne karakteristike, a da se u isto vreme ostvare značajne uštede u latenciji rada kola, površini integrisanog kola i potrošnji integrisanog kola koje realizuje filtarsku funkciju.

9 Dodatak

U cilju kreiranja rezultata prezentovanih u ovoj disertaciji razvijene su brojne softverske rutine. Deo rutina vezan za poglavlja 2 i 3 je pisan u MATLAB jeziku, dok je za implementaciju sofisticiranijih algoritama opisanih u 7. poglavlju korišćen C jezik [KeRi06].

9.1 MATLAB funkcije

S obzirom na tematiku kojom se teza bavi MATLAB (*MATrix LABoratory*) je bio logičan izbor. Ovaj programski jezik je razvijen od strane američkog matematičara Kliva Molera (*Cleve Moler*) sedamdesetih godina dvadesetog veka. MATLAB je inicijalno pisan u FORTRAN jeziku uz primenu LINPACK i EINSACK, FORTRAN biblioteka u kojima su implementirane rutine za matrični račun (linearna algebra) [DBMS79, HuNu58, SBGI13]. U početku je MATLAB bio zamišljen tako da obezbedi funkcionalost tada popularnog FORTRAN jezika koristeći jednostavniju sintaksu. Razvojem personalnih računara osamdesetih godina prošlog veka pojavila se i prva komercijalna verzija MATLAB interpretera kompanje MathWorks® implementirana u C jeziku [Math16]. Ovo je omogućilo da se MATLAB obogati dodatnim setom jezičkih konstrukcija dostupnih u C jeziku što je MATLAB učinilo prenosivim i proširljivim. Generalno, MATLAB je danas zreo jezik (*full-featured*) koji podržava moderne koncepte poput polimorfnih funkcija, objektno orijentisanog i funkcionalnog programiranja.

Ubrzo nakon pojave prve stabilne verzije interpretera, MATLAB je prihvaćen od strane velikog broja članova naučne i inženjerske zajednice koji su koristili MATLAB da rešavaju probleme iz svojih disciplina i oblasti. Skup rutina koje rešavaju probleme iz neke oblasti se u MATLAB terminologiji naziva tulboks (*toolbox*). U protekle tri decenije razvijen je veliki broj tulboksova koji pokrivaju širok dijapazon disciplina, tako da se korišćenjem odgovarajućeg tulboksa mogu efikasno i brzo rešiti problemi iz zadate oblasti. Rutine koje su prezentovane u ovom radu se baziraju na upotrebi dva najpopularnija i najstarija tulboksa: *signal processing* (rutine vezane za oblast obrade signala) i *control* (rutine vezane za oblast automatskog upravljanja).

Gotovo paralelno sa razvojem komercijalne verzije MATLAB interpretera aktivirano je nekoliko javnih web projekata u cilju razvijanja nekomercijalne, open source, varijante MATLAB interpretera. Tako je nastao jedan od najpopularnijih MATLAB interpretera GNU OCTAVE, pisan u C++, koji je široko prihvaćen u akademskoj zajednici [Hans11]. Tu su još i Python biblioteke NUMPY, SCIPY, PYPLOT, MATPLOTLIB (obično se portuju kao

objedinjena PYLAB biblioteka) koje su takođe u javnom domenu [HuOt07, JOPO01]. Neke od ovih biblioteka čak imaju opcije koje nisu dostupne u komercijalnoj verziji MATLAB interpretera. OCTAVE varijanta MATLAB interpretera ima veoma visoku kompatibilnost (preko 90%) sa komercijalnom verzijom pa se često nalazi kao zamena za MathWorks® MATLAB. Takođe, podržava gotovo sve standardne tulboksove koji su korišćeni za rutine prezentovane u ovom radu³. Dakle, sve MATLAB funkcije prezentovane u ovom radu se mogu interpretirati OCTAVE interpreterom bez obaveze licenciranja. To praktično znači da prilikom razvijanja rutina nisu korišćeni sofisticirani MathWorks® MATLAB tulboksovi poput paketa za simboličku analizu, itd.. Čak su i neke od funkcija koje su dostupne u MATLAB tulboksovima od interesa posebno pisane u cilju portabilnosti i bolje kontrole implementiranih algoritama.

Bitno je napomenuti da sve varijante MATLAB interpretera (bilo komercijalne ili nekomercijalne) nude C/C++ API (Application Programming Interface) tako da je moguće koristiti kôd pisan u C jeziku unutar MATLAB programa i obrnuto.

Progresivnim radom na tezi nastao je velik broj MATLAB funkcija/skriptova koji pored CAD podrške projektovanju CMAC IIR filtera imaju i dekorativni karakter (priprema grafika, štampanje rezultata, itd.). Ovde će biti prezentovane samo suštinski bitne funkcije. Treba reći još da su sve MATLAB funkcije polimorfne, tj. argumenti i vrednosti koje vraćaju mogu uzimati različitu formu. Tako npr. argument može biti skalar ili vektor. U nastavku je dat opis razvijenih MATLAB funkcija.

amp_resp_typ1 Proračunava amplitudski odziv FIR filtra sa linearnom fazom I tipa (prvi red Tabela 2.2).

Upotreba

`a = amp_resp_typ1(c, th)`

Argumenti

`c` Vektor koeficijenta filtra. Broj koeficijenata mora da bude neparan broj simetričan oko centralnog elementa npr. [1 2 3 2 1].
`th` Digitalna frekvencija.

Izlaz

`a` Amplitudski odziv.

amp_resp_typ2 Proračunava amplitudski odziv FIR filtra sa linearnom fazom II tipa (drugi red Tabela 2.2).

Upotreba

`a = amp_resp_typ2(c, th)`

Argumenti

`c` Vektor koeficijenta filtra. Broj koeficijenata mora da bude paran broj simetričan oko sredine sekvence npr. [1 2 3 3 2 1].
`th` Digitalna frekvencija.

³ U GNU OCTAVE *signal processing* tulboksa se zove *signal*.

Izlaz

a Amplitudski odziv.

amp_resp_typ3 Proračunava amplitudski odziv FIR filtra sa linearnom fazom III tipa (treći red Tabela 2.2).

Upotreba

a = amp_resp_typ3(c, th)

Argumenti

c Vektor koeficijenta filtra. Broj koeficijenata mora da bude neparan broj asimetričan oko centralnog elementa pri čemu je centralni element jednak nuli
npr. [1 -2 0 2 -1].

th Digitalna frekvencija.

Izlaz

a Amplitudski odziv.

amp_resp_typ4 Proračunava amplitudski odziv FIR filtra sa linearnom fazom IV tipa (četvrti red Tabela 2.2).

Upotreba

a = amp_resp_typ3(c, th)

Argumenti

c Vektor koeficijenta filtra. Broj koeficijenata mora da bude paran broj asimetričan oko sredine sekvence npr. [1 -2 2 -1].

th Digitalna frekvencija.

Izlaz

a Amplitudski odziv.

filter_tdf_ii Model IIR filtra drugog reda (Sl. 4.7).

Upotreba

[y, w] = filter_tdf_ii(c, d, x)

Argumenti

c Koeficijenti brojioca

d Koeficijenti imenioca.

x Ulazni signal.

Izlaz

y Izlazni signal.

w Kritična tačka na izlazu internog sabirača.

frmpars Paralelna realizacija filtra u s-domenu (3.34)

Upotreba

[k, secs, rr, pp] = frmparsz(ns, ds, Ts, tranType, breakVer)

Argumenti

ns Koeficijenti brojioca prenosne funkcije u s-domenu

ds Koeficijenti imenioca prenosne funkcije u s-domenu

Izlaz

k Slobodni član u (3.31)

secs Matrica veličine $n \times 6$ čiji redovi sadrže koeficijente imenioca (kolone [1:3]) i brojioca (kolone [4:6]) pojedinih sekcija u paralelenoj realizaciji (3.35)-(3.36).

rr Ostatci u polovima

pp Polovi prenosne funkcije

frmparsz Paralelna realizacija IIR filtra (Sl. 2.4) na bazi bilinearne i transformacije drugog reda.

Upotreba

[k, secs, ss] = frmparsz(ns, ds, Ts, tranType, breakVer)

Argumenti

ns	Koeficijenti brojioca prenosne funkcije u s-domenu
ds	Koeficijenti imenioca prenosne funkcije u s-domenu
Ts	Perioda odmeravanja
tranType	Tip transformacije, 'bilinear'- (3.12), 'quadratic'-(3.15)
breakVer	Načini realizacije sekcija četvrtog reda kada se koristi transformacija drugog reda ('quadratic'), 1-(Sl. 3.5b), 2-(Sl. 3.5a)

Izlaz

k	Slobodni član u (3.31)
secs	Matrica veličine $n \times 6$ čiji redovi sadrže koeficijente imenioca (kolone [1:3]) i brojioca (kolone [4:6]) pojedinih sekcija u paralelenog realizaciji (3.37)-(3.42).
ss	Sekcije paralelene realizacije filtra u s-domenu dobijene <i>frmpars</i> funkcijom

mapOmg Preslikava kružnu frekvenciju ω u digitalnu frekvenciju θ i obrnuto za bilinearnu transformaciju (3.13)

Upotreba

$[wy] = \text{mapOmg}(wx, Ts, str)$

Argumenti

wx	Originalna frekvencija
Ts	Perioda odmeravanja
str	's'- $wx=\omega$ $wy=\theta$, 's'- $wx=\theta$ $wy=\omega$

Izlaz

wy	Preslikana frekvencija
----	------------------------

mapOmgQuad Preslikava kružnu frekvenciju ω u digitalnu frekvenciju θ za transformaciju drugog reda (3.28).

Upotreba

$wd = \text{mapOmgQuad}(wa, Ts)$

Argumenti

wa	Kružna frekvencija [rad/sec]
Ts	Perioda odmeravanja

Izlaz

wd	Digitalna frekvencija [rad]
----	-----------------------------

grpdelaywa Proračunava grupno kašnjenje analognog filtra

Upotreba

$gda = \text{grpdelaywa}(na, da, wa)$

Argumenti

nd	Koeficijenti brojioca prenosne funkcije u s-domenu
da	Koeficijenti imenioca prenosne funkcije u s-domenu
wa	Kružna frekvencija [rad/sec]

Izlaz

gda	Grupno kašnjenje analognog filtra [sec]
-----	---

grpdelaywd Proračunava grupno kašnjenje digitalnog filtra

Upotreba

$gdz = \text{grpdelaywd}(nz, dz, wd)$

Argumenti

nd	Koeficijenti brojioca prenosne funkcije u s-domenu
da	Koeficijenti imenioca prenosne funkcije u s-domenu
wd	Kružna frekvencija [rad/sec]

Izlaz

gdz	Grupno kašnjenje izrađeno u broju odmeraka (semplova) [smp]. Da bi se dobilo grupno kašnjenje u [sec] <i>gdz</i> treba pomnožiti sa T_s .
-----	---

paarmann Preslikavanje iz s- u z-domen korišćenjem PIM (MIM) metoda. Preuzeto iz [Vadh07]

Upotreba

```
[nz, dz] = paarmann(ns, ds, Ts, method, M, N)
```

Argumenti

nd	Koeficijenti brojioca prenosne funkcije u s-domenu.
da	Koeficijenti imenioca prenosne funkcije u s-domenu.
Ts	Perioda smplovanja.
method	'mim' (Poglavlje 6.1), 'pim' (Poglavlje 6.2).
M	Red rezultujućeg filtra u z-domenu
N	Broj odmeraka amplitudkse/fazne karakteristike u s-domenu.

Izlaz

gdz	Grupno kašnjenje izrađeno u broju odmeraka (smplova) [smp]. Da bi se dobilo grupno kašnjenje u [sec] <i>gdz</i> treba pomnožiti sa T_s .
-----	--

s2z_numeric Transformiše nule i polove prenosne funkcije u s-domenu u z-domne korišćenjme PIM (MIM) metoda, bilinearne ili transformacije drugog reda. Funkcija takođe formira kaskadnu realizaciju IIR filtra (Sl. 2.3) i proračunava frekvencijski odziv i grupno kašnjenje u zadatim frekvencijama.

Upotreba

```
[zz, zp, kz, ndz, hd, gd, wd] = ...  
s2z_numeric(sz, sp, Ts, wa, tranType, methType)
```

Argumenti

sz	Nule prenosne funkcije u s-domenu
sp	Polovi prenosne funkcije u s-domenu
Ts	Perioda odmeravanja
wa	Kružna frekvencija
tranType	Tip transformacije, 'bilinear'-(3.12), 'quadratic'-(3.15) ili 'paarmann'-(Poglavlje 6).
methType	Izbor metoda za 'paarmman' transformaciju, 'mim' (Odeljak 6.1), 'pim' (Odeljak 6.2).

Izlaz

zz	Nule prenosne funkcije u z-domenu
zp	Polovi prenosne funkcije u z-domenu
kz	Pojačanje nastalo kao posledica transforacije nula/polova
ndz	Matrica veličine $n \times 6$ čiji redovi sadrže koeficijente imenioca (kolone [1:3]) i brojioca (kolone [4:6]) pojedinih sekcija u kasadnoj realizaciji.
hd	Frekvencijski odziv
gd	Grupno kašnjenje
wd	Digitalna frekvencija za 'bilinear'-(3.13), 'quadratic'-(3.28), 'paarmann'- $\omega_a T_s$

sigfigar Reprezentuje realne vrednosti sa zadatim brojem značajnih cifara

Upotreba

```
arsf = sigfigar(ar, sfig)
```

Argumenti

ar	Originalna vrednost
sfig	Broj značajnih cifara

Izlaz

arsf	Originalna vrednost prezentovana sa <i>sfig</i> brojem značajnih cifara
------	---

quantar Kvantuje vrednosti po Q[*prec, frac*] formatu (Odeljak 4.1)

Upotreba

```
[arc, arb, arh] = quantar(ar, prec, frac)
```

Argumenti

ar	Originalna vrednost.
prec	Dužina digitalne reči u bitovima.

frac	Broj bitova iza decimalnog zaraza.
Izlaz	
arc	Kvantovana vrednost u decimalnom obliku.
arb	Kvantovana vrednost u binarnom obliku.
arh	Kvantovana vrednost u heksadecimalnom obliku.

genFirTypeI_FdlRtlVhdl Štampa RTL model FIR filtra sa linearnom fazom I tipa FDL arhitekture (Sl. 5.2) u formi VHDL koda.

Upotreba

```
hdl_file = ...
    genFirTypeI_FdlRtlVhdl(cfCell, qfrm, ord, pth, entName )
```

Argumenti

cfCell	Koeficijenti filtra u binarnom obliku
qfrm	Vektor veličine 1×2 čije su vrednosti Q[qfrm(1) qfrm(2)].
ord	Red filtra.
pth	Putanja do direktorijuma gde će biti smešten VHDL fajl
entName	Ime VHDL entiteta.

Izlaz

hdl_file Ime kreiranog VHDL fajla (*entName.vhd*).

genFirTypeI_RtlVhdlPkg Štampa VHDL *package* fajl za RTL model FIR filtra sa linearnom fazom I tipa. Ovaj fajl sadrži deklaraciju komponenti korišćenih u testbenču i definicije novih tipova.

Upotreba

```
pkg_file = genFirTypeI_RtlVhdlPkg( qfrm, ord, pth, entName )
```

Argumenti

qfrm	Vektor veličine 1×2 čije su vrednosti Q[qfrm(1) qfrm(2)].
ord	Red filtra.
pth	Putanja do direktorijuma gde će biti smešten VHDL fajl
entName	Ime VHDL entiteta.

Izlaz

pkg_file Ime kreiranog VHDL *package* fajla (*entName_pkg.vhd*).

genFirTypeI_RtlVhdlTb Štampa VHDL testbenč fajl za RTL model FIR filtra sa linearnom fazom I tipa. Ovaj fajl uključuje *package* fajl kreiran *genFirTypeI_RtlVhdlPkg* funkcijom.

Upotreba

```
tb_file = ...
    genFirTypeI_RtlVhdlTb(freq, qfrm, NFFT, hdlFFT, ...
        scalePower, ord, infnm, outfnm, pth, entName )
```

Argumenti

freq	Vektor frekvencija oblika [f_{bw} f_d f_g f_s $f_{in,0}$ $f_{in,1}$... $f_{in,n}$], gde je f_{bw} - propusni pseg, [f_d , f_g] - donja i gornja granična frekvencija, f_s - frekvencija odmreavanja i [$f_{in,0}$ $f_{in,1}$... $f_{in,n}$] - testne frekvenicije. Sve vrednosti su izražene u [Hz].
qfrm	Vektor veličine 1×2 čije su vrednosti Q[qfrm(1) qfrm(2)].
NFFT	Broj tačaka za FFT (<i>Fast Fourier Transform</i>) analizu. Treba da bude stepen broja dva.
hdlFFT	Određuje da li će rezultati simulacije biti podvrgnuti FFT analizi ('true') ili ne ('false').
scalePower	Stepn broja dva koji određuje faktor skaliranja ulaznog signala.
ord	Red filtra.
infnm	Ime tekstualnog fajla za smeštanje odmeraka ulaznog signala
outfnm	Ime tekstualnog fajla za smeštanje odmeraka izlaznog signala
pth	Putanja do direktorijuma gde će biti smešten VHDL fajl
entName	Ime VHDL entiteta.

Izlaz

tb file Ime kreiranog VHDL *package* fajla (*entName_tb.vhd*).

genFirTypeI_FdlRtlDoTcl Štampa Tcl *do* fajl za RTL model FIR filtra sa linearnom fazom I tipa. Ovaj fajl sadrži komande logičkog simulatora.

Upotreba

```
do_file = ...
  genFirTypeI_FdlRtlDoTcl( waveValRng, simStr, modStr,...
    runStr, pth, entName )
```

Argumenti

waveValRng Opseg iscrtavanja vremenskih dijagrama [*-waveValRng:waveValRng*]. Za ovu vrednost se obično uzima nešto veća vrednost od očekivane, maksimalne, vrednosti signala.

simStr Izbor logičkog simulatora: 'avhdl' - Active-HDL, 'vsim' - ModelSim.

modStr Izbor režima rada logičkog simulatora: 'batch' - paketni, 'gui' - grafički.

runStr Vreme trajanja simulacije npr. '1ns', '10ns' ili '-all' (vreme trajanja simulacije određeno testbenčom).

pth Putanja do direktorijuma gde će biti smešten Tcl fajl

entName Ime VHDL entiteta.

Izlaz

pkg_file Ime kreiranog Tcl *do* fajla (*entName_simStr_modStr.do*).

genRtlVhdlCode Štampa za RTL model IIR filtra. Moguće je implementirati paralelnu (Sl. 2.4) i serijsku (Sl. 2.3) arhitekturu pri čemu su ćelije prvog i drugog reda u TDF II formi (Sl. 4.7). Takođe se može odabrati dužina reči unutar ćelija i tip aritmetičkog bloka za sabiranje (oduzmač, sabirač).

Upotreba

```
hdl_file = ...
  genRtlVhdlCode( sbin, qfrm, ksec, substr, parser, ...
    num1st, num2nd, breakVer, tranType, ...
    interPrec, pth, entName )
```

Argumenti

sbin Koeficijenti filtra u binarnom obliku

qfrm Vektor veličine 1×2 čiji su vrednosti $Q[qfrm(1) \ qfrm(2)]$.

substr Izbor aritmetičkog bloka za sabiranje u TDF II formi (4.7): 1- oduzimač, 0- sabirač

parser Izbor arhitekture: 1- paralelna, 2- serijska

num1st Broj sekcija prvog reda u (3.31)

num2nd Broj sekcija drugog reda u (3.31)

breakVer Načini realizacije sekcija četvrtog reda kada se koristi transformacija drugog reda ('quadratic'), 1-(Sl. 3.5b), 2-(Sl. 3.5a)

tranType Tip transformacije, 'bilinear'- (3.12), 'quadratic'-(3.15)

interPrec Dužina reči unutar TDF II ćelije (preciznost interne aritmetike): 'double'- $2 \times qfrm(1)$, 'single'- $qfrm(1)$

pth Putanja do direktorijuma gde će biti smešten VHDL fajl

entName Ime VHDL entiteta.

Izlaz

hdl_file Ime kreiranog Tcl *do* fajla (*entName.vhd*).

genRtlVhdlPkg Štampa VHDL *package* fajl za RTL model IIR filtra. Ovaj fajl sadrži deklaraciju komponenti korišćenih u testbenču i definicije novih tipova.

Upotreba

```
pkg_file = genRtlVhdlPkg( qfrm, pth, entName )
```

Argumenti

qfrm Vektor veličine 1×2 čiji su vrednosti $Q[qfrm(1) \ qfrm(2)]$.

pth Putanja do direktorijuma gde će biti smešten VHDL fajl

entName Ime VHDL entiteta.

Izlaz

pkg_file Ime kreiranog VHDL *package* fajla (*entName_pkg.vhd*).

genRtlVhdlTb Štampa VHDL testbenč fajl za RTL model IIR filtra. Ovaj fajl uključuje *package* fajl kreiran *genRtlVhdlPkg* funkcijom.

Upotreba

```
tb_file = ...
genRtlVhdlTb( freq, qsec, NFFT, hdlFFT, scalePower, ...
             infnm, outfnm, pth, entName )
```

Argumenti

freq Vektor frekvencija oblika $[f_{bw} f_d f_g f_s f_{in,0} f_{in,1} \dots f_{in,n}]$, gde je f_{bw} - propusni pseg, $[f_d, f_g]$ - donja i gornja granična frekvencija, f_s - frekvencija odmreavanja i $[f_{in,0} f_{in,1} \dots f_{in,n}]$ - testne frekvenicije. Sve vrednosti su izražene u [Hz].

qfrm Vektor veličine 1×2 čiji su vrednosti $Q[qfrm(1) qfrm(2)]$.

NFFT Broj tačaka za FFT (*Fast Fourier Transform*) analizu. Treba da bude stepen broja dva.

hdlFFT Određuje da li će rezultati simulacije biti podvrgnuti FFT analizi ('true') ili ne ('false').

scalePower Stepn broja dva koji određuje faktor skaliranja ulaznog signala.

infnm Ime tekstualnog fajla za smeštanje odmeraka ulaznog signala

outfnm Ime tekstualnog fajla za smeštanje odmeraka izlaznog signala

pth Putanja do direktorijuma gde će biti smešten VHDL fajl

entName Ime VHDL entiteta.

Izlaz

tb_file Ime kreiranog VHDL *package* fajla (*entName_tb.vhd*).

genRtlDoTcl Štampa Tcl *do* fajl za RTL model FIR filtra sa linearnom fazom I tipa. Ovaj fajl sadrži komande logičkog simulatora.

Upotreba

```
do_file = ...
genRtlDoTcl( waveValRng, simStr, modStr, ...
            runStr, pth, entName )
```

Argumenti

waveValRng Opseg iscertavanja vremenskih dijagrama $[-waveValRng:waveValRng]$. Za ovu vrednost se obično uzima nešto veća vrednost od očekivane, maksimalne, vrednosti signala.

simStr Izbor logičkog simulatora: 'avhdl' - Active-HDL, 'vsim' - ModelSim.

modStr Izbor režima rada logičkog simulatora: 'batch' - paketni, 'gui' - grafički.

runStr Vreme trajanja simulacije npr. '1ns', '10ns' ili '-all' (vreme trajanja simulacije određeno testbenčom).

pth Putanja do direktorijuma gde će biti smešten Tcl fajl

entName Ime VHDL entiteta.

Izlaz

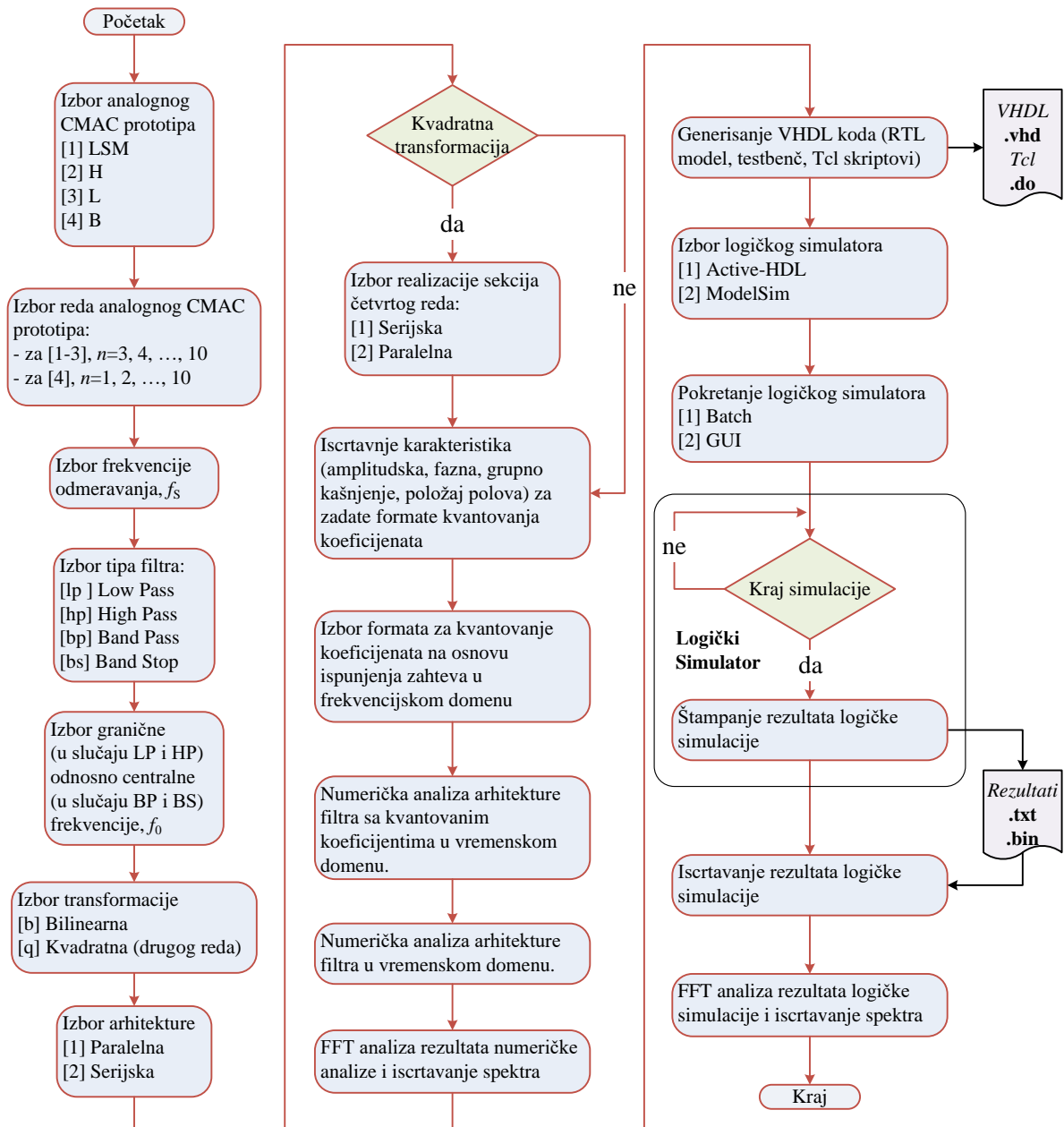
pkg_file Ime kreiranog Tcl *do* fajla (*entName_simStr_modStr.do*).

9.2 Podrška projektovanju polinomskih CMAC filtara u z-domenu

Za projektovanje polinomskih CMAC filtra u z-domenu razvijen je MATLAB skript kojim se implementira tok projektovanja prikazan na Sl. 9.1. Korišćenjem ovog toka su kreirani rezultati prezentovani u Odeljcima 4.1 i 5. Koeficijenti analognih prototipa su preuzeti iz [ToLA15b]. Za transliranje analognog prototipa korišćene su funkcije *signal-processing*

tulboksa (lp2lp, lp2hp, lp2bp i lp2bs). Za grafičku prezentaciju rezultata korišćene su standardne MATLAB funkcije za crtanje (plot, bar, ...). Slično važi i za obradu rezultata simulacije, poput FFT analize.

Štampanje VHDL i do fajlova je ostvareno putem funkcija prezentovanim u prethodnom odeljku (funkcije čije ime počinje sa gen). Pored RTL modela filtra (koji se kasnije koristi za fizičku realizaciju) štampaju se testbenč i package fajlovi. Radi bolje preglednosti primeri ovih fajlova su dati u narednom odeljku.



Slika 9.1 Algoritamski prikaz toka projektovanja IIR filtera.

Pokretanje logičkog simulatora je ostvareno putem sistemskog poziva korišćenjem MATLAB funkcije system. Inače ova funkcija je dostupna u svim programskim jezicima.

Oba simulatora (Active-HDL, ModelSim) su dostupna bez obaveze licenciranja [Alde16, Grap16]. Naravno, implementirani tok funkcioniraju i sa komercijalnim verzijama ovih simulatora. Primer sistemskih poziva izabranih simulatora je dat u Tabeli 9.1.

Tabela 9.1 Pokretanje logičkih simulatora putem sistemskog poziva (argumenti `system` funkcije)

Režim rada	ModelSim	Active-HDL
Pakteski (<i>Bathc</i>)	<code>vsim -c -do design.do</code>	<code>vsimsa -do design.do</code>
Grafički (<i>GUI</i>)	<code>vsim -do design.do</code>	<code>avhdl -do design.do</code>

Ovi simulatori su najzastupljeniji u i najrasprostranjeniji pa su iz tog razloga i izabrani. Interfejs prema njima je ostvaren preko *do* fajlova. Opšte prihvaćeni jezik za pisanje *do* skriptova je Tcl. Generalno, svi moderni simulatori se portuju sa nekom varijantom Tcl interpretera i sa API ka simulatoru. Primer *do* fajla dobijen funkcijom `genRtlDoTcl` za pokretanje ModelSim simulatora je dat u tabeli 9.2.

Tabela 9.2 Primer *do* fajla za simulaciju LSM LP IIR filtra trećeg reda

ModelSim *do* fajl

```
vlib work
vmap work
vcom ../src/tdf_ii_lp_lsm_ord_3_pkg.vhd
vcom ../src/tdf_ii_lp_lsm_ord_3.vhd
vcom ../src/adc.vhd
vcom ../src/dac.vhd
vcom ../src/wtf.vhd
vcom ../src/tdf_ii_lp_lsm_ord_3_tb.vhd
vsim -L work tdf_ii_lp_lsm_ord_3_tb
view structure
view signals
view wave
add wave -label CLK clk
add wave -label RST rst
add wave -label X -decimal x
add wave -label Y -decimal y
add wave -label X_SCALED -decimal x_scaled
add wave -label Y_SCALED -decimal y_scaled
add wave -label AIN -analog-step -min -1 -max 1 -height 100 ain
add wave -label AOUT -analog-step -min -1 -max 1 -height 100 aout
add wave -label AIN_SCALED -analog-step -min -1 -max 1 -height 100 ain_scaled
add wave -label AOUT_SCALED -analog-step -min -1 -max 1 -height 100 aout_scaled
add wave -label simulation_end_flag ENDSIM
run -all
quit -f
```

Nakon logičke simulacije rezultati se smeštaju u odgovarajuće tekstualne ili binarne fajlove. Ovi fajlovi se zatim ponovo učitavaju u MATLAB okruženje gde se dalje sprovodi obrada i grafička reprezentacija dobijenih rezultata. Na ovaj način je ostvaren jednostavan i nezavisan interfejs između MATLAB okruženja i logičkog simulatora.

Konačno, poređenjem rezultata logičke i numeričke analize može se ustanoviti da li filter ispunjava projektantske zahteve. Ukoliko to nije slučaj ceo tok se može ponoviti sa izborom drugog formata kvantovanja koeficijenata (većeg broja bitova za dužinu digitalne reči).

9.3 VHDL kodovi

U ovom odeljku su dati primeri VHDL kodova dobijeni upotrebom funkcija `genRtlVhdlCode`, `genRtlVhdlPkg` i `genRtlVhdlTb` za paralelnu realizaciju IIR filtra. Pošto su VHDL kodovi filtera razmatranih u tezi (odeljci 4.1 i 5) veoma obimni⁴, ovde je prezentovan VHDL kôd za primer LSM LP filter trećeg reda dobijen korišćenjem bilinearne transformacije sa dvostrukom preciznošću unutar pojedinih ćelija pri čemu su koeficijenti kvantovani formatom Q[16 14]. Za granice propusnog opsega usvojeno je, $f_g=1\text{kHz}$, a za frekvenciju uzorkovanja, $f_s=50\text{kHz}$. Koeficijenti filtra koji množe izlazni signal, y , su invertovani tako da su korišćeni samo sabirači. Dati su još i opisi idealnog Analogno-Digitalnog/Digitalno-Analognog (AD/DA) konvertora i komponente koje služe za upisivanje rezultata u tekstualne/binarne fajlove. Ove komponente služe samo u svrhe simulacije, tj. nisu sintetizibilne.

VHDL kôd RTL modela filtra:

```
library IEEE;
use IEEE.STD_LOGIC_1164.all;
use IEEE.NUMERIC_STD.all;

entity tdf_ii_lp_lsm_ord_3 is
  generic(
    N : natural:= 16;
    F : natural:= 14
  );
  port(
    clk: in std_logic;
    rst : in std_logic;
    x: in std_logic_vector( N - 1 downto 0 );
    y: out std_logic_vector( N - 1 downto 0 )
  );
end entity tdf_ii_lp_lsm_ord_3;

architecture tdf_ii_lp_lsm_ord_3_rtl of tdf_ii_lp_lsm_ord_3 is

  -- Filter Coefficients Signals --

  signal c0_1 : std_logic_vector( N - 1 downto 0 );
  signal c1_1 : std_logic_vector( N - 1 downto 0 );
  signal d1_1 : std_logic_vector( N - 1 downto 0 );
  signal c0_2 : std_logic_vector( N - 1 downto 0 );
  signal c1_2 : std_logic_vector( N - 1 downto 0 );
  signal c2_2 : std_logic_vector( N - 1 downto 0 );
  signal d1_2 : std_logic_vector( N - 1 downto 0 );
```

⁴ VHDL kôdovi se mogu dobiti od autora.

```

signal d2_2 : std_logic_vector( N - 1 downto 0 );

-- Signals for connecting section's outputs --

signal y1 : std_logic_vector( N - 1 downto 0 );
signal y2 : std_logic_vector( N - 1 downto 0 );

-- Internal signals for the sections --

signal xc0_1 : std_logic_vector( 2*N - 1 downto 0 );
signal xc1_1 : std_logic_vector( 2*N - 1 downto 0 );
signal yd1_1 : std_logic_vector( 2*N - 1 downto 0 );
signal w0_1 : std_logic_vector( 2*N - 1 downto 0 );
signal w0r_1 : std_logic_vector( 2*N - 1 downto 0 );
signal yy_1 : std_logic_vector( 2*N - 1 downto 0 );
signal xc0_2 : std_logic_vector( 2*N - 1 downto 0 );
signal xc1_2 : std_logic_vector( 2*N - 1 downto 0 );
signal xc2_2 : std_logic_vector( 2*N - 1 downto 0 );
signal yd1_2 : std_logic_vector( 2*N - 1 downto 0 );
signal yd2_2 : std_logic_vector( 2*N - 1 downto 0 );
signal w0_2 : std_logic_vector( 2*N - 1 downto 0 );
signal w0r_2 : std_logic_vector( 2*N - 1 downto 0 );
signal w1_2 : std_logic_vector( 2*N - 1 downto 0 );
signal w1r_2 : std_logic_vector( 2*N - 1 downto 0 );
signal yy_2 : std_logic_vector( 2*N - 1 downto 0 );

begin

-- Assigning Coefficients Values --

c0_1 <= "0000001100011010";
c1_1 <= "0000001100011010";
d1_1 <= "0011100111100111";
c0_2 <= "1111110011101001";
c1_2 <= "1111111111111110";
c2_2 <= "0000001100010101";
d1_2 <= "0111100011100101";
d2_2 <= "1100011000110111";

-- Instantiation of sections --

-----
--
-- SECTION: 1
-- TDFII, 1st order, double precision inside
--
-----

-- Delay --
D_REG_1 : process( clk, rst ) is
begin
    if (rst = '0') then
        w0r_1 <= ( others => '0' );
    elsif ( rising_edge( clk ) ) then
        w0r_1 <= w0_1;
    end if;
end process D_REG_1;

-- Direct path --
xc0_1 <= std_logic_vector(
        signed( x )

```

```

        *      signed( c0_1 ) );
xc1_1 <= std_logic_vector(
        signed( x )
        *      signed( c1_1 ) );

-- Feedback path --
yd1_1 <= std_logic_vector(
        signed( yy_1( N + F - 1 downto F ) )
        *      signed( d1_1 ) );

-- Delay (shared) path --
w0_1 <= std_logic_vector(
        signed( xc1_1 )
        +      signed( yd1_1 ) );

-- Output double precision --
yy_1 <= std_logic_vector(
        signed( xc0_1 )
        +      signed( w0r_1 ) );

-- Output single precision--
y1 <= yy_1( N + F - 1 downto F );

-----
--
-- SECTION: 2
-- TDFII, 2nd order, double precision inside
--
-----

-- Delay --
D_REG_2 : process( clk, rst ) is
begin
    if (rst = '0') then
        w0r_2 <= ( others => '0' );
        w1r_2 <= ( others => '0' );
    elsif ( rising_edge( clk ) ) then
        w0r_2 <= w0_2;
        w1r_2 <= w1_2;
    end if;
end process D_REG_2;

-- Direct path --
xc0_2 <= std_logic_vector(
        signed( x )
        *      signed( c0_2 ) );
xc1_2 <= std_logic_vector(
        signed( x )
        *      signed( c1_2 ) );
xc2_2 <= std_logic_vector(
        signed( x )
        *      signed( c2_2 ) );

-- Feedback path --
yd1_2 <= std_logic_vector(
        signed( yy_2( N + F - 1 downto F ) )
        *      signed( d1_2 ) );
yd2_2 <= std_logic_vector(
        signed( yy_2( N + F - 1 downto F ) )
        *      signed( d2_2 ) );

```



```

-- Delay (shared) path --
w0_2 <= std_logic_vector(
    signed( xc2_2 )
    + signed( yd2_2 ) );
w1_2 <= std_logic_vector(
    signed( xc1_2 )
    + signed( yd1_2 )
    + signed( w0r_2 ) );

-- Output double precision --
yy_2 <= std_logic_vector(
    signed( xc0_2 )
    + signed( wlr_2 ) );

-- Output single precision --
y2 <= yy_2( N + F - 1 downto F );

-----
--
-- Adders
--
-----

y <= std_logic_vector(
    signed( y1 )
    + signed( y2 ) );

end architecture tdf_ii_lp_lsm_ord_3_rtl;

```

VHDL kôd idealnog AD kovertora:

```

library IEEE;
use IEEE.STD_LOGIC_1164.all;
use IEEE.MATH_REAL.all; -- round
use IEEE.STD_LOGIC_ARITH.all; -- conv...

entity adc is
    generic ( N : natural := 12; -- word length
             F : natural := 10); -- fraction length
    port (
        clk: in std_logic;
        rst: in std_logic;
        ain : in real;
        dout: out std_logic_vector(N-1 downto 0)
    );
end adc;

architecture adc of adc is
    -- analog 2 digital --
    procedure a2d (
        signal a: in real; -- analog input
        constant prec: in integer; -- precision (whole word length)
        constant frac: in integer; -- fraction
        signal d: out std_logic_vector(N-1 downto 0) ) is

        variable aa : real:=0.0;

    begin

        if ( a < 0.0 ) then

```

```

        aa := 2.0**(prec) - round(abs(a)*(2.0**frac));
    elsif a > 0.0 then
        aa := round(a*(2.0**frac));
    end if;

    d <= conv_std_logic_vector(integer(aa), prec);

end procedure a2d;

begin

p_adc: process (rst, clk) is
begin
    if (rst = '0') then
        dout <= (others => '0');
    elsif( rising_edge( clk) ) then
        a2d(ain, N, F, dout);
    end if;
end process p_adc;

end adc;

```

VHDL kôd idealnog DA kovertora:

```

library IEEE;
use IEEE.STD_LOGIC_1164.all;
use IEEE.STD_LOGIC_ARITH.all;
use IEEE.MATH_REAL.all;
use IEEE.NUMERIC_STD.all;

entity dac is
    generic ( N : natural := 12;
              F : natural := 10);
    port (
        clk: in std_logic;
        rst: in std_logic;
        din: in std_logic_vector(N-1 downto 0);
        aout: out real
    );
end dac;

architecture dac of dac is

    function b2r ( -- binary to real
        b : in std_logic_vector(N-1 downto 0)
    ) return real is

        variable m : real:= 0.0;
        variable r : real:= 0.0;
        variable bb : std_logic_vector(N-1 downto 0);

    begin
        bb := b;
        lfor: for k in 0 to N-1 loop
            if bb(k) = '1' then
                m := 1.0;
            elsif b(k) = '0' then
                m := 0.0;
            end if;
            r := r + (2.0**k)*m;
        end loop lfor;
        return r;
    end b2r;

end architecture dac;

```

```

end function b2r;

-- digital 2 analog --
procedure d2a(
  signal d : in std_logic_vector(N-1 downto 0); -- digital input
  constant prec : in integer; -- precision (word length)
  constant frac : in integer; -- fraction part
  signal a : out real ) is -- analog output
  variable dd : real := 0.0;
  variable dd_buff : real := 0.0;
  constant rmin : real := 0.0;
  constant rmax : real := 2.0**(prec) - 1.0;
begin
  dd_buff := b2r(d);
  if ( dd_buff < rmax ) then
    if (dd_buff > rmin ) then
      dd := dd_buff;
    else
      dd := rmin;
    end if;
  else
    dd := rmax;
  end if;

  if d(N-1) = '1' then
    a <= (dd - 2.0**prec)/( 2.0**(frac) );
  elsif d(N-1) = '0' then
    a <= dd/(2.0**frac);
  end if;

end procedure d2a;

begin

p_adc: process (rst, clk) is
begin
  if (rst = '0') then
    aout <= 0.0;
  elsif( rising_edge(clk) ) then
    d2a(din, N, F, aout);
  end if;
end process p_adc;

end dac;

```

VHDL kôd komponente za upis rezultat u tekstualni i/ili binarni fajl (*wtf*):

```

library IEEE;
use IEEE.STD_LOGIC_1164.all;
use IEEE.STD_LOGIC_TEXTIO.all;
use STD.TEXTIO.all;
use IEEE.MATH_REAL.all; -- round
use IEEE.STD_LOGIC_ARITH.all; -- conv_
use IEEE.NUMERIC_STD.all;

entity wtf is
  generic ( N : natural := 8;
           file_name : string;
           txt_nbin: boolean:= true
  );
  port (

```

```

    signal clk : in std_logic;
    signal rst : in std_logic;
    signal ES: in boolean;
    signal a : in real;
    signal d : in std_logic_vector(N-1 downto 0)
);
end wtf;

architecture wtf of wtf is

    constant file_name_b : string:= file_name & ".bin";
    constant file_name_t : string:= file_name & ".txt";

    -- converts time to real --
    function time2real( t : time ) return real is
        constant simres : time := 1ps;
        constant resreal : real := 1.0e-12;
    begin
        return real(t / simres) * resreal;
    end function time2real;

    type REAL_FILE is file of real;

    file fnmr : REAL_FILE;
    file fnmt : text;

begin

    -- Opne files --
    file_open(fnmt, file_name_t, write_mode);
    file_open(fnmr, file_name_b, write_mode);

    p_wtf: process(clk, rst, ES) is
        variable ln : line;
    begin
        if ES = false then
            if ( rising_edge(clk) and rst = '1' ) then
                if txt_nbin = true then
                    -- Write to txt file
                    write(ln, time2real(now) );
                    write(ln, string'(" "));
                    write(ln, a);
                    write(ln, string'(" "));
                    write(ln, d);
                    writeline( fnmt, ln );
                else
                    -- Write to binary file
                    write(fnmr, a);
                end if;
            end if;
        else
            file_close(fnmt);
            file_close(fnmr);
        end if;
    end process p_wtf;

end architecture wtf;

```

VHDL kôd paketa (*package*):

```
-- Library declaration --
```

```

library IEEE;
use IEEE.STD_LOGIC_1164.all;
use IEEE.MATH_REAL.all;
use IEEE.NUMERIC_STD.all;

-- Package Declaration --

package tdf_ii_lp_lsm_ord_3_pkg is

  -- Component Declaration --

  component tdf_ii_lp_lsm_ord_3 is
    generic(
      N : natural := 16;
      F : natural := 14
    );
    port(
      clk: in std_logic;
      rst : in std_logic;
      x: in std_logic_vector( N - 1 downto 0 );
      y: out std_logic_vector( N - 1 downto 0 )
    );
  end component tdf_ii_lp_lsm_ord_3;

  component adc is
    generic(
      N : natural := 16;
      F : natural := 14
    );
    port(
      clk: in std_logic;
      rst : in std_logic;
      ain: in real;
      dout: out std_logic_vector( N - 1 downto 0 )
    );
  end component adc;

  component dac is
    generic(
      N : natural := 16;
      F : natural := 14
    );
    port(
      clk: in std_logic;
      rst : in std_logic;
      din: in std_logic_vector( N - 1 downto 0 );
      aout: out real
    );
  end component dac;

  component wtf is
    generic(
      N : natural := 16;
      file_name : string;
      txt_nbin : boolean := true
    );
    port(
      clk: in std_logic;
      rst : in std_logic;
      ES : in boolean;

```

```

        a: in real;
        d: in std_logic_vector( N - 1 downto 0 )
    );
end component wtf;

-- Type definitions --

type real_vector is array (natural range<>) of real;

end package tdf_ii_lp_lsm_ord_3_pkg;

```

VHDL kôd testbenča:

```

-- Library declaration --

library IEEE;
use IEEE.STD_LOGIC_1164.all;
use IEEE.MATH_REAL.all;
use IEEE.NUMERIC_STD.all;
use WORK.tdf_ii_lp_lsm_ord_3_pkg.all;

-- Test-Bench Entity Declaration --

entity tdf_ii_lp_lsm_ord_3_tb is
    generic(
        N : natural := 16;
        F : natural := 14
    );
end entity tdf_ii_lp_lsm_ord_3_tb;

-- Test-Bench Architecture Declaration --

architecture tdf_ii_lp_lsm_ord_3_test of tdf_ii_lp_lsm_ord_3_tb is

    -- Input Signals (Stimulated) --

    signal rst : std_logic;
    signal clk : std_logic;
    signal x : std_logic_vector( N - 1 downto 0 );

    -- Output Signals (Observed) --

    signal y : std_logic_vector( N - 1 downto 0 );

    -- Additional Test-Bench signals --

    signal aout : real;
    signal aout_scaled : real;
    signal ain : real;
    signal ain_scaled : real;
    signal x_scaled : std_logic_vector( N - 1 downto 0 );
    signal y_scaled : std_logic_vector( N - 1 downto 0 );
    signal ENDSIM : boolean := false;

    -- Test-Bench Constants --

    constant FBW: real := 1.0000e+003;
    constant FD: real := 0.0000e+000;
    constant FG: real := 1.0000e+003;
    constant FS: real := 5.0000e+004;

```

```

constant FIN: real_vector( 0 to 3 ) :=
(
    4.997253e+002,
    1.999664e+003,
    3.999329e+003,
    5.999756e+003
);
constant NFFT: real := 65536.000000;
constant FFT : boolean := false; -- Set FFT to "true" for longer sim.
constant TS : time := ( 1.0 / FS ) * 1.0sec;
constant M : integer := integer( round( FS / FIN(1) ) );
constant TSIM : time := 10.0*( 1.0 / FIN(1) ) * 1.0sec;
constant TFFT : time := NFFT * TS;
constant POW : integer := 1024; -- Power of scaling factor
constant SCALE_FACTOR : real := 2.0 ** POW; -- scaling factor
constant IR: boolean := false; -- Impulse Response

begin

-- Unit Under Test --

UUT : tdf_ii_lp_lsm_ord_3
generic map(
    N => N,
    F => F
)
port map(
    rst => rst,
    clk => clk,
    x => x_scaled,
    y => y_scaled
);

-- Ideal ADC for input signal --

A2D : adc
generic map(
    N => N,
    F => F
)
port map(
    rst => rst,
    clk => clk,
    ain => ain_scaled,
    dout => x_scaled
);

-- Ideal DAC for dumping output signal --

D2A : dac
generic map(
    N => N,
    F => F
)
port map(
    rst => rst,
    clk => clk,
    din => y_scaled,
    aout => aout_scaled
);

```

```

-- Write output to file --

WTF0: wtf
  generic map(
    N => N,
    file_name => "out_tdf_ii_lp_lsm_ord_3",
    txt_nbin => true
  )
  port map(
    clk => clk,
    rst => rst,
    ES => ENDSIM,
    a => aout,
    d => y
  );

-- Write input to file --

WTF1: wtf
  generic map(
    N => N,
    file_name => "in_tdf_ii_lp_lsm_ord_3",
    txt_nbin => true
  )
  port map(
    clk => clk,
    rst => rst,
    ES => ENDSIM,
    a => ain,
    d => x
  );

-- Generate clock signal for processing --

CLK_GEN: process is
begin
  if ( ENDSIM = false ) then
    clk <= '0';
    wait for TS/2.0;
    clk <= '1';
    wait for TS/2.0;
  else
    wait;
  end if;
end process CLK_GEN;

-- Scale input/output --

SCALE: process (rst, clk) is
begin
  if ( rst = '0' ) then
    aout <= 0.0;
    ain <= 0.0;
  elsif ( rising_edge ( clk ) ) then
    aout <= SCALE_FACTOR * aout_scaled; -- scale up output
    ain <= SCALE_FACTOR * ain_scaled; -- scale up input
    -- scale up output
    y <= y_scaled( N-1-POW downto 0 ) & ( POW-1 downto 0 => '0');
    -- scale up input
    x <= x_scaled( N-1-POW downto 0 ) & ( POW-1 downto 0 => '0');
  end if;

```



```

end process SCALE;

-- Generate input signal --

p_sig : process( clk, rst ) is
  variable k : real := 0.0;
  variable a : real := 0.0;
begin
  if (rst = '0') then
    a := 0.0;
  elsif ( rising_edge( clk ) ) then
    if IR = false then
      a := 0.0;
      for_loop: for j in FIN'range loop
        a := sin( MATH_2_PI * FIN( j ) * k / FS ) + a;
      end loop for_loop;
    else
      if k = 0.0 then
        a := 1.0;
      else
        a := -1.0;
      end if;
    end if;
    k := k + 1.0;
  end if;
  ain_scaled <= a / ( 2.0 ** POW );
end process p_sig;

-- Master simulation process --

p_sim: process is
begin
  rst <= '0';
  wait for 5.0 * TS;
  rst <= '1';
  if FFT = false then
    wait for TSIM;
  else
    wait for TFFT;
  end if;
  ENDSIM <= true;
  wait;
end process p_sim;

end architecture tdf_ii_lp_lsm_ord_3_test;

```

Lista pojmova

LC	Električno kolo koje sadrži induktivnosti i kapacitivnosti.
RC	Električno kolo koje sadrži otpornosti i kapacitivnosti.
gmC	Električno kolo koje aproksimira funkciju kalema u modernim VLSI kolima korišćenjem operacionog transkonduktansnog pojačavča (gm stepen) i kondenzatora.
VLSI	Integrirana kola veoma visokog stepena integracije (<i>Very Large Scale Iteration</i>)
DSP	Digitalno obrada signala (<i>Digital Signal Processing</i>)
IIR	Diskretni sistem sa beskonačnim impulsnim odzivom (<i>Infinite Impulse Response</i>)
FIR	Diskretni sistem sa konačnim impulsnim odzivom (<i>Finite Impulse Response</i>)
LP	Propusnik niskih frekvencija (<i>Low Pass</i>)
HP	Propusnik visokih frekvencija (<i>High Pass</i>)
BP	Propusnik opsega frekvencija (<i>Band Pass</i>)
BS	Prigušnik opsega frekvencija (<i>Band Stop</i>)
CMAC	Granično monotona amplitudska karakteristika (<i>Critical Monotonic Amplitude Characteristic</i>)
LSM	Tip filtarske funkcije (<i>Least Squares Monotonic</i>)
IC	Integrirana kola (<i>Integrated Circuits</i>)
ASIC	Integrirana kola pravljena specifično za određenu primenu (<i>Application Specific Integrated Circuits</i>)
TDFII	Direktna transponovana forma II (<i>Transpose Discrete Form II</i>)
MIM	Metod preslikavanja iz s u z domen na bazi analognog prototipa gde je ključni kriterijum očuvanje amplitudske karakteristike analognog prototipa (<i>Magnitude Invariance Method</i>)
PIM	Metod preslikavanja iz s u z domen na bazi analognog prototipa gde je ključni kriterijum očuvanje fazne karakteristike analognog prototipa (<i>Phase Invariance Method</i>)
LPVLSI	VLSI kola sa niskom potrošnjom (<i>Low Power</i>)
HDL	Jezik za opis hardvera (<i>Hardware Description Language</i>)
RTL	Model digitalnog kola na nivou registara (<i>Register Transfer Level</i>)

EDA	Automatizacija procesa projektovanja elektronskih kola (<i>Electronic Design Automation</i>)
CAD	Projektovanje uz pomoć računara (<i>Computer Aided Design</i>)
CDS	Proizvođač CAD/EDA softvera (<i>Cadence Design Systems, Inc</i>)
LIB	Format za opis vremenskih karakteristika biblioteke standardnih IC ćelija (<i>Liberty</i>)
LEF	Format za opis fizičkih karakteristika biblioteke standardnih IC ćelija (<i>Library Exchange Format</i>)
SDC	Format za opis ograničenja po pitanju kašnjenja u digitalnim IC kolima (<i>Synopsys Delay Constraint</i>)
SPF	Format za opis parazitnih komponenti u digitalnim IC kolima (<i>Standard Parasitic Format</i>)
SDF	Format za opis kašnjenja uzrokovanih parazitnim komponentama u digitalnim IC kolima (<i>Standard Delay Format</i>)
GDS	Format za opis leajauta u CAD alatima (<i>Graphics Database System</i>)
DEF	Format za opis projekta u <i>Cadence Design Systems, Inc</i> CAD alatima (<i>Design Exchange Format</i>)
MSR	Radio uređaj koji podržava veliki broj standarda za prenos radio signala (<i>Multi-Standard Radio</i>)
DTFT	Furijeova transformacija u diskretnom vremenu (<i>Discrete Time Fourier Transform</i>)
IDTFT	Inverzna Furijeova transformacija u diskretnom vremenu (<i>Inverse Discrete Time Fourier Transform</i>)
DFT	Diskretna Furijeova transformacija (<i>Discrete Fourier Transform</i>)
IDFT	Inverzna Diskretna Furijeova transformacija (<i>Inverse Discrete Fourier Transform</i>)
SLA	Tip protočnog (<i>pipeline</i>) algoritma (<i>Scattered Look Ahead</i>)
FDL	Često korišćena direktna forma FIR filtra sa linearnom fazom (<i>Folded Delay Line</i>)
API	Skup softverskih rutina, protokola i alata za građenje softverskih aplikacija (<i>Application Programming Interface</i>)

Literatura

- [Aboa99] Aboagye, A. K. (1999). Overflow avoidance techniques in cascaded IIR filter implementations on the TMS320 DSPs. *Texas Instruments Application Report*.
- [Alde16] Aldec®. (2016). Active-HDL Student Edition.
- [AID110] Almazan, S. P. R., and d. Leon, M. T. G. (2010). A 3rd Order Butterworth Gm-C Filter for WiMAX Receivers in a 90nm CMOS Process. In *Computer Modelling and Simulation (UKSim), 2010 12th International Conference on* (pp. 625–630). <https://doi.org/10.1109/UKSIM.2010.119>
- [AnZL14] Andrejević Stošović, M. V., Zivanić, J. M., and Litovski, V. B. (2014). Maximally Flat Filter Functions With the Maximum Number of Transmission Zeros Having Maximal Multiplicity. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 61(10), 778–782. <https://doi.org/10.1109/TCSII.2014.2345300>
- [Anto79] Antoniou, A. (1979). *Digital Filters Analysis and Design*, McGrawHill. Inc., New York.
- [BoHT63] Bogert, B. P., Healy, M. J. R., and Tukey, J. W. (1963). The quefrency alanalysis of time series for echoes: Cepstrum, pseudo-autocovariance, cross-cepstrum and saphe cracking. In *Proceedings of the symposium on time series analysis* (Vol. 15, pp. 209–243).
- [Butt30] Butterworth, S. (1930). On the theory of filter amplifiers. *Wireless Engineer*, 7(6), 536–541.
- [Cohe98] Cohen, L. (1998). Generalization of the Wiener-Khinchin theorem. *IEEE Signal Processing Letters*, 5(11), 292–294.
- [DBMS79] Dongarra, J. J., Bunch, J. R., Moler, C. B., and Stewart, G. W. (1979). *LINPACK users' guide*. Siam.
- [Euro00] EUROPRACTICE. (n.d.). 2016 mini@sic Europractice MPW runs Schedule and Prices. Retrieved from http://www.europractice-ic.com/general_runschedule_minisic.php
- [Fuka59] Fukada, M. (1959). Optimum Filters of Even Orders with Monotonic Response. *IRE Transactions on Circuit Theory*, 6(3), 277–281. <https://doi.org/10.1109/TCT.1959.1086558>
- [Gear71] Gear, C. W. (1971). *Numerical initial value problems in ordinary differential equations*. Prentice Hall PTR.

- [Geff74] Geffe, P. R. (1974). Designers guide to active bandpass filters. *Part III*, *EDN*, 19(7).
- [Géhe71] Géher, K. (1971). *Theory of network tolerances*. Akadémiai Kiadó.
- [Grap16] Graphics®, M. (2016). ModelSim PE Student Edition.
- [GrKM75] Graupe, D., Krause, D. J., and Moore, J. (1975). Identification of autoregressive moving-average parameters of time series. *IEEE Transactions on Automatic Control*, 20(1), 104–107.
- [GrTe78] Gregorian, R., and Temes, G. (1978). Design techniques for digital and analog all-pass circuits. *IEEE Transactions on Circuits and Systems*, 25(12), 981–988.
- [Halp69] Halpern, P. (1969). Optimum Monotonic Low-Pass Filters. *IEEE Transactions on Circuit Theory*, 16(2), 240–242. <https://doi.org/10.1109/TCT.1969.1082945>
- [Hans11] Hansen, J. S. (2011). *GNU Octave: Beginner's Guide: Become a Proficient Octave User by Learning this High-level Scientific Numerical Tool from the Ground Up*. Packt Publishing Ltd.
- [HoLi12] Hongyan, C., and Lisheng, W. (2012). Software and hardware implementation of IIR based on Matlab&AccelDSP. In *Proc. of the 2nd Int. Conf. on Computer Application and System Modeling*, Atlantis Press, Paris (pp. 1411–1415).
- [HuNu58] Hughes, R. A., and Nutt, R. (1958). *FORTRAN Automatic Coding System for the IBM 704 Data Processing System*. Public domain.
- [HuOt07] Hunter, J. D., and others. (2007). Matplotlib: A 2D graphics environment. *Computing in Science and Engineering*, 9(3), 90–95.
- [Itoh82] Itoh, K. (1982). Analysis of the phase unwrapping algorithm. *Appl. Opt.*, 21(14), 2470.
- [JOPO01] Jones, E., Oliphant, T., Peterson, P., and others. (2001). Open source scientific tools for Python. Scipy.
- [Jova13] Jovanović, B. (2013). *ANALITIČKI MODEL ZA PROCENU DINAMIČKE POTROŠNJE ARITMETIČKIH KOLA IMPLEMENTIRANIH NA FPGA*. Univerzitet u Nišu, Elektronski Fakultet.
- [Jova16] Jovanović, B. (2016). *Napredni Metodi Projektovanja Digitalnih Integrisanih Kola u Nanometarskim Tehnologijama sa Posebnim Naglaskom na Brzinu, Statičku i Dinamičku Potrošnju*. Doktorska disertacija, Univerzitet u Nišu, Elektronski Fakultet.
- [Kaes08] Kaeslin, H. (2008). *Digital integrated circuit design: from VLSI architectures to CMOS fabrication*. Cambridge University Press.
- [Kais66] Kaiser, J. F. (1966). Digital Filters. Chapter 7 in *System Analysis by Digital*

- Computers, published by Kuo and Kaiser. John Wiley & Sons, New York.
- [KeRi06] Kernighan, B. W., and Ritchie, D. M. (2006). *The C programming language*. Prentice Hall.
- [Kran99] Krantz, S. G. (1999). The Rouché's Theorem, § 8.2 in *Handbook of Complex Variables*. Boston, MA: Birkhäuser.
- [Lito76] Litovski, V. (1976). Gaussian Filter with a Modified Characteristic Function. In *Proc. of the XX Konf. ETAN-a, Opatija*.
- [Lito79] Litovski, V. (1979). Synthesis of monotonic passband sharp cutoff filters with constant group delay response. *IEEE Transactions on Circuits and Systems*, 26(8), 597–602.
- [LiMi83] Litovski, V. B., and Milovanovich, D. P. (1983). New solution for the ideal filter approximation problem. *IEE Proceedings G - Electronic Circuits and Systems*, 130(4), 161–163. <https://doi.org/10.1049/ip-g-1:19830029>
- [LiZw96] Litovski, V., and Zwolinski, M. (1996). *VLSI circuit simulation and optimization*. Springer Science & Business Media.
- [Litw00] Litwin, L. (2000). FIR and IIR digital filters. *IEEE Potentials*, 19(4), 28–31.
- [Mari08] Marinković, M. (2008). *Decimacioni filtri u trofaznom integrisanom meraču potrosnje električne energije*. Magistarska teza, Univerzitet u Nisu, Elektronski Fakultet.
- [Math16] MathWorks. (2016). MATLAB: The Language of Technical Computing.
- [MiLL15] Mirković, D., Litovski, I., and Litovski, V. (2015). On the Synthesis and Realization of Selective Linear Phase IIR Filters. In *2nd International Conference on Electrical, Electronic and Computing Engineering IcETRAN 2015, Silver Lake, Serbia* (p. ELI1.3.1-6). Etran Society.
- [MiPL14] Mirković, D., Petković, P., and Litovski, V. (2014). A second order s-to-z transform and its implementation to IIR filter design. *COMPEL: The International Journal for Computation and Mathematics in Electrical and Electronic Engineering*, 33(5), 1831–1843.
- [MSPL15] Mirković, D., Stošović, M. A., Petković, P., and Litovski, V. (2015). IIR digital filters with critical monotonic pass-band amplitude characteristic. *AEU-International Journal of Electronics and Communications*, 69(10), 1495–1505.
- [Nage75] Nagel, L. W. (1975). SPICE2: A computer program to simulate semiconductor circuits. *ERL Memo ERL-M520*.
- [Nela07] Nelatury, S. R. (2007). Additional correction to the impulse invariance method for the design of IIR digital filters. *Digital Signal Processing*, 17(2), 530–540.

- [NéKo00] Németh, J. G., and Kollar, I. (2000). Step-invariant transform from z-to s-domain: a general framework. In *Instrumentation and Measurement Technology Conference, 2000. IMTC 2000. Proceedings of the 17th IEEE* (Vol. 2, pp. 902–907).
- [OpSc75] Oppenheim, A. V, and Schaffer, R. W. (1975). Digital signal processing. *Englewood Cliffs, New York*, 166–170.
- [OpSc10] Oppenheim, A. V, and Schaffer, R. W. (2010). *Discrete-time signal processing*. Pearson Higher Education.
- [OpSS68] Oppenheim, A. van, Schaffer, R., and Stockham, T. (1968). Nonlinear filtering of multiplied and convolved signals. *IEEE Transactions on Audio and Electroacoustics*, 16(3), 437–466.
- [OrTe68] Orchard, H., and Temes, G. (1968). Filter Design Using Transformed Variables. *IEEE Transactions on Circuit Theory*, 15(4), 385–408. <https://doi.org/10.1109/TCT.1968.1082870>
- [Oust89] Ousterhout, J. K. (1989). *Tcl: An embeddable command language*. Citeseer.
- [Paar98] Paarmann, L. D. (1998). Mapping from the s-domain to the z-domain via the magnitude-invariance method. *Signal Processing*, 69(3), 219–228.
- [PaAt06] Paarmann, L. D., and Atris, Y. H. (2006). Mapping from the s-domain to the z-domain via the phase-invariance method. *Signal Processing*, 86(2), 223–229.
- [Parh07a] Parhi, K. K. (2007a). VLSI digital signal processing systems: design and implementation. John Wiley & Sons.
- [Parh07b] Parhi, K. K. (2007b). *VLSI digital signal processing systems: design and implementation*. John Wiley & Sons.
- [PaMe89] Parhi, K. K., and Messerschmitt, D. G. (1989). Pipeline interleaving and parallelism in recursive digital filters. I. Pipelining using scattered look-ahead and decomposition. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 37(7), 1099–1117. <https://doi.org/10.1109/29.32286>
- [PaPK08] Park, W., Park, K.-S., and Koh, H.-M. (2008). Active control of large structures using a bilinear pole-shifting transform with H_∞ control method. *Engineering Structures*, 30(11), 3336–3344.
- [PaMc72] Parks, T., and McClellan, J. (1972). Chebyshev Approximation for Nonrecursive Digital Filters with Linear Phase. *IEEE Transactions on Circuit Theory*, 19(2), 189–194. <https://doi.org/10.1109/TCT.1972.1083419>
- [PrSa89] Prasad, K. P., and Sathyanarayana, P. (1989). Signal scaling in cascade digital filters. *Circuits, Systems and Signal Processing*, 8(4), 421–426.

- [QuPP04] Quélhas, M. F., Petraglia, A., and Petraglia, M. R. (2004). Efficient group delay equalization of discrete-time IIR filters. In *Signal Processing Conference, 2004 12th European* (pp. 125–128).
- [Raba09] Rabaey, J. (2009). *Low Power Design Essentials* (1st ed.). Springer US.
- [RaKD91] Radecki, J., Konrad, J., and Dubois, E. (1991). Design of finite wordlength IIR filters with prescribed magnitude, group delay and stability properties using simulated annealing. In *Acoustics, Speech, and Signal Processing, 1991. ICASSP-91., 1991 International Conference on* (pp. 1637–1640).
- [Rako70] Rakovich, B. (1970). Linear-phase transmission-line filters with increased selectivity. *IEEE Transactions on Circuit Theory*, 17(1), 41–45. <https://doi.org/10.1109/TCT.1970.1083051>
- [RaLi73] Rakovich, B. D., and Litovski, V. B. (1973). Least-squares monotonic lowpass filters with sharp cutoff. *Electronics Letters*, 9(4), 75–76. <https://doi.org/10.1049/el:19730056>
- [RaDj72] Rakovich, B., and Djurich, B. (1972). Chebyshev approximation of a constant group delay with constraints at the origin. *IEEE Transactions on Circuit Theory*, 19(5), 466–475. <https://doi.org/10.1109/TCT.1972.1083519>
- [RaLa72] Rakovich, B., and Lazovich, S. (1972). Monotonic low-pass filters with improved stopband performance. *IEEE Transactions on Circuit Theory*, 19(2), 218–221. <https://doi.org/10.1109/TCT.1972.1083420>
- [RaPo80] Rakovich, B., and Popovich, M. (1980). Characteristic function of least-mean-square passband filters with finite attenuation poles. *IEEE Transactions on Circuits and Systems*, 27(12), 1225–1233. <https://doi.org/10.1109/TCS.1980.1084771>
- [RaPD77] Rakovich, B., Popovich, M., and Drakulich, B. (1977). Minimum phase transfer functions providing a compromise between phase and amplitude approximation. *IEEE Transactions on Circuits and Systems*, 24(12), 718–724. <https://doi.org/10.1109/TCS.1977.1084295>
- [Reme34] Remez, E. Y. (1934). Sur la détermination des polynômes d'approximation de degré donné. *Comm. Soc. Math. Kharkov*, 10, 41–63.
- [Rudi64] Rudin, W. (1964). *Principles of mathematical analysis* (Vol. 3). McGraw-Hill New York.
- [SaUl58] Saal, R., and Ulbrich, E. (1958). On the Design of Filters by Synthesis. *IRE Transactions on Circuit Theory*, 5(4), 284–327. <https://doi.org/10.1109/TCT.1958.1086481>

- [SaSh68] Sandberg, I. W., and Shichman, H. (1968). Numerical integration of systems of stiff nonlinear differential equations. *Bell System Technical Journal*, 47(4), 511–527.
- [ShWZ12] Shi, R., Wang, S., and Zhao, J. (2012). An unsplit complex-frequency-shifted PML based on matched Z-transform for FDTD modelling of seismic wave equations. *Journal of Geophysics and Engineering*, 9(2), 218.
- [SBGI13] Smith, B. T., Boyle, J. M., Garbow, B. S., Ikebe, Y., Klema, V. C., and Moler, C. B. (2013). *Matrix eigensystem routines-EISPACK guide* (Vol. 6). Springer.
- [StDi82] Steiglitz, K., and Dickinson, B. (1982). Phase unwrapping by factorization. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 30(6), 984–991.
- [StMa03] Stern, H. P. E., and Mahmoud, S. (2003). *Communication systems: analysis and design*. Prentice-Hall, Inc.
- [Syst08a] Systems, C. D. (2008a). Cadence NC-Verilog Simulator Help. San Jose, California, United States.
- [Syst09a] Systems, C. D. (2009a). Command Reference for Encounter RTL Compiler. San Jose, California, United States.
- [Syst09b] Systems, C. D. (2009b). Encounter Text Command Reference. San Jose, California, United States.
- [Syst09c] Systems, C. D. (2009c). Encounter User Guide. San Jose, California, United States.
- [Syst08b] Systemts, C. D. (2008b). Cadence NC-VHDL Simulator Help. San Jose, California, United States.
- [Syst08c] Systemts, C. D. (2008c). Encounter RTL Compiler Synthesis Flows. San Jose, California, United States.
- [Syst09d] Systemts, C. D. (2009d). SimVision Command Language Reference. San Jose, California, United States.
- [TCHR01] Tarczynski, A., Cain, G. D., Hermanowicz, E., and Rojewski, M. (2001). A WISE method for designing IIR filters. *IEEE Transactions on Signal Processing*, 49(7), 1421–1432.
- [Thed05] Thede, L. D. (2005). Practical analog and digital filter design (pp. 165–169). Artech House New Jersey.
- [Thom49] Thomson, W. E. (1949). Delay networks having maximally flat frequency characteristics. *Proceedings of the IEE-Part III: Radio and Communication Engineering*, 96(44), 487–490.
- [Tols97] Tolstov, G. P. (1997). *Fourier Series (Dover Books on Mathematics)*. (Richard A. Silverman, Trans.) (1st ed.). New York, USA: Devor Publications. Retrieved from

<http://store.doverpublications.com/0486633179.html>

- [ToLA15a] Topisirovic, D., Litovski, V., and Andrejevic Stosovic, M. (2015a). CMAC and Chebyshev Filters in Frequency Domain. In *Small Systems Simulation Symposium, Niš, Serbia* (pp. 269–283).
- [ToLA15b] Topisirović, D., Litovski, V., and Andrejević Stošović, M. (2015b). Unified theory and state-variable implementation of critical-monotonic all-pole filters. *International Journal of Circuit Theory and Applications*, 43(4), 502–515. <https://doi.org/10.1002/cta.1956>
- [Trib77] Tribolet, J. (1977). A new phase unwrapping algorithm. *IEEE Transactions on Acoustics Speech and Signal Processing*, 25(2), 170–177.
- [Vadh07] Vadhavkar, P. R. (2007). *Mapping controllers from the s-domain to the z-domain using magnitude invariance and phase invariance methods*. Wichita State University.
- [Vaid93] Vaidyanathan, P. P. (1993). Robust digital filter structures. *Handbook for Digital Signal Processing*, 419–490.
- [WTAF06] Williams, A. B., Taylor, F. J., Arthur, B. W., and Fred, J. T. (2006). *Electronic filter design handbook*.
- [ŽiLa06] ŽIŠKA, P., and LAIPERT, M. (2006). Analog group delay equalizers design based on evolutionary algorithm. *Radioengineering*, 15(1), 1–5.
- [Zver05] Zverev, A. I. (2005). *Handbook of filter synthesis*. Wiley-Blackwell.
- [Zwol04] Zwolinski, M. (2004). *Digital system design with VHDL*. Pearson Education.

Biografija autora

Autor je rođen 12. maja 1982. godine u Užicu, Republika Srbija. Diplomirao je na Elektronskom fakultetu, Univerziteta u Nišu 2007. godine. U periodu od 2007.-2008. volontira u projektnom odeljenju za razvoj softvera firme Atlas Electronics d.o.o., Surdulica. Od 2009.-2010. radi kao saradnik u nastavi za užu naučnu oblast Elektronika na Elektronskom fakultetu, a od 2011. je u zvanju asistent za užu naučnu oblast Elektronika. Od 2008. godine je član Laboratorije za automatizaciju projektovanja elektronskih kola (LEDA) Elektronskog fakulteta u Nišu. Kao član laboratorije učestvovao je na dva nacionalna projekta finansirana od strane Ministarstva prosvete, nauke i tehnološkog razvoja Republike Srbije. Do sada je objavio 45 publikacija, od toga 12 tehničkih rešenja (M85), 13 radova na skupovima nacionalnog značaja (M63), 11 radova na skupovima međunarodnog značaja (M33), 2 rada u međunarodnim časopisima (M23) i učestvovao je u izradi 3 publikacije vezane za edukaciju.

IZJAVA O AUTORSTVU

Izjavljujem da je doktorska disertacija, pod naslovom

**Projektovanje selektivnih IIR digitalnih filtara sa
linearnom fazom upotrebom analognih prototipova**

koja je odbranjena na Elektronskom fakultetu Univerziteta u Nišu:

- rezultat sopstvenog istraživačkog rada;
- da ovu disertaciju, ni u celini, niti u delovima, nisam prijavljivao na drugim fakultetima, niti univerzitetima;
- da nisam povredio autorska prava, niti zloupotrebio/la intelektualnu svojinu drugih lica.

Dozvoljavam da se objave moji lični podaci, koji su u vezi sa autorstvom i dobijanjem akademskog zvanja doktora nauka, kao što su ime i prezime, godina i mesto rođenja i datum odbrane rada, i to u katalogu Biblioteke, Digitalnom repozitorijumu Univerziteta u Nišu, kao i u publikacijama Univerziteta u Nišu.

U Nišu, _____

Potpis autora disertacije:

(Ime, srednje slovo i prezime)

**IZJAVA O ISTOVETNOSTI ELEKTRONSKOG I ŠTAMPANOG OBLIKA
DOKTORSKE DISERTACIJE**

Naslov disertacije:

**Projektovanje selektivnih IIR digitalnih filtara sa
linearnom fazom upotrebom analognih prototipova**

Izjavljujem da je elektronski oblik moje doktorske disertacije, koju sam predao za unošenje u **Digitalni repozitorijum Univerziteta u Nišu**, istovetan štampanom obliku.

U Nišu, _____

Potpis autora disertacije:

(Ime, srednje slovo i prezime)

IZJAVA O KORIŠĆENJU

Ovlašćujem Univerzitetsku biblioteku „Nikola Tesla“ da u Digitalni repozitorijum Univerziteta u Nišu unese moju doktorsku disertaciju, pod naslovom:

**Projektovanje selektivnih IIR digitalnih filtara sa
linearnom fazom upotrebom analognih prototipova**

Disertaciju sa svim priložima predao sam u elektronskom obliku, pogodnom za trajno arhiviranje.

Moju doktorsku disertaciju, unetu u Digitalni repozitorijum Univerziteta u Nišu, mogu koristiti svi koji poštuju odredbe sadržane u odabranom tipu licence Kreativne zajednice (Creative Commons), za koju sam se odlučio.

1. Autorstvo (**CC BY**)
2. Autorstvo – nekomercijalno (**CC BY-NC**)
3. Autorstvo – nekomercijalno – bez prerade (**CC BY-NC-ND**)
4. Autorstvo – nekomercijalno – deliti pod istim uslovima (**CC BY-NC-SA**)
5. Autorstvo – bez prerade (**CC BY-ND**)
6. Autorstvo – deliti pod istim uslovima (**CC BY-SA**)

U Nišu, _____

Potpis autora disertacije:

(Ime, srednje slovo i prezime)