

Univerzitet u Beogradu – Elektrotehnički fakultet



Katedra za elektroniku

Digitalna obrada signala (19E043DOS)

Drugi domaći zadatak 2025/26.

Cilj drugog domaćeg zadatka je da studenti samostalno probaju osnovne metode projektovanja IIR i FIR filtara i da projektovane filtre iskoriste za filtriranje signala u određenim primerima, kao i za promenu učestanosti odabiranja.

Domaći zadatak se sastoji iz tri dela. Prvi deo domaćeg zadatka se sastoji od projektovanja banke IIR filtara nepropusnika opsega koji služe za poboljšanje kvaliteta pokvarenog zvučnog signala. Cilj ovog dela je i da studenti primene znanja iz prvog domaćeg zadatka kako bi znali sami da procene koje gabarite filtara je potrebno odabrati. Drugi deo domaćeg zadatka je implementacija IIR filtara koji služe za ekvalizaciju zvuka. Treći deo domaćeg zadatka je realizacija funkcija za promenu učestanosti odabiranja pomoću FIR filtara.

Deo 1: Filtriranje zvučnog signala

(9 poena)

Tokom snimanja jedne pesme, jedno dete se igralo gitarom i stalno okidalo jednu žicu. Zbog toga se preko originalne pesme u snimljenom signalu čuje ton A odsviran na gitari. Fajl *allplusA.wav* se sastoji od jedne sekvence pesme „*All the things you are*“ i na nju dodate note A čiji je segment dat u fajlu *noteA.wav*.

1. Učitati signale iz fajla *allplusA.wav* i *noteA.wav* korišćenjem naredbe **wavfile.read**. Prikazati vremenski oblik dela signala. Vremenska osa treba da bude u sekundama.
2. Napisati funkciju **[b, a] = bandstop_filter_Cheb2(fs, fa, fp, Aa, Ap)** kojom se projektuje IIR filtar nepropusnik opsega učestanosti korišćenjem inverznog Čebiševljevog filtra za analogni prototip. Funkcija kao argumente prima učestanost odabiranja f_s ,

granične učestanosti nepropusnog opsega f_a kao vektor dve vrednosti, granične učestanosti propusnog opsega f_p kao vektor dve vrednosti i odgovarajuća slabljenja u nepropusnom (α_a) i propusnom (α_p) opsegu. Kao povratnu vrednost, funkcija vraća koeficijente polinoma u imeniocu i brojiocu funkcije prenosa dobijenog filtra. Obratiti pažnju na način zadavanja parametara kod transformacije učestanosti za Čebiševljev inverzni filter (vidi u materijalima).

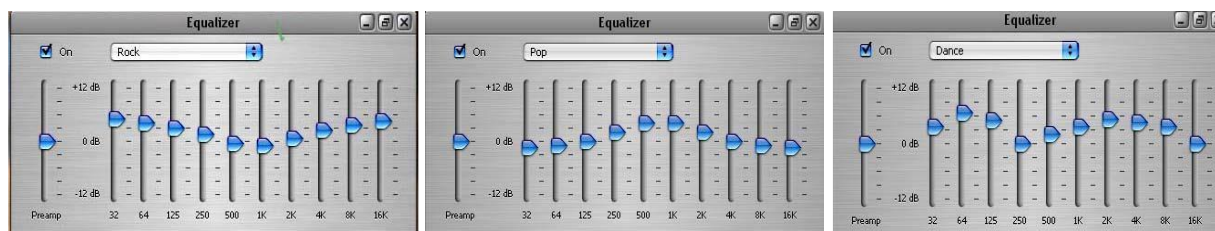
3. Na istoj slici, jedan pored druge, nacrtati spektrograme signala iz fajlova *allplusA.wav* i *noteA.wav*. Spektrograme crtati u smislenom opsegu učestanosti.
4. Na osnovu dobijenih spektrograma korišćenjem funkcije **bandstop_filter_Cheb2** projektovati filtre kojima se filtrira originalni signal tako da on bude očišćen od dodate note A. Nacrtati spektrogram izlaznog signala i filtrirani signal pustiti pomoću IPython plejera.
5. Nacrtati amplitudske karakteristike filtera iz tačke 4 i crtanjem odgovarajućih linija za gabarite pokazati da filteri zadovoljavaju zadate specifikacije. Amplitudske karakteristike crtati tako da se jasno vidi nepropusni opseg, tj. zumirati frekvencijsku osu na smislene učestanosti. Takođe, crtati više karakteristika na jednoj slici, na više podslika korišćenjem **subplot**-a.

U kodu komentarima jasno naznačiti koji deo koda se odnosi na koji deo zadatka. Sve vremenske ose u ovoj tački treba da budu u sekundama. Neophodno je obeležiti sve ose odgovarajućim oznakama/tekstom.

Deo 2: Ekvalizacija zvučnog signala

(9 poena)

Ekvalizacija zvučnog signala predstavlja podešavanje pojačanja različitih frekvencijskih komponenti u spektru sa ciljem postizanja prijatnijeg efekta na slušaoc. Tako, za različite vrste muzike (pop, rok, klasična, dance, ...) treba različito pojačati ili oslabiti određene delove spektra kako bi se postigao željeni kvalitet. Na slici 1 je prikazan profil ekvalizacije za rok, pop i dance muziku iz aplikacije iTunes. Može se primetiti da se različiti frekvencijski opsezi pojačavaju za maksimalno 12 dB ili se oslabljuju za maksimalno 12 dB. Ovo se realizuje bankom filtera propusnika opsega koji se dodatno množe odgovarajućim pojačanjem. Signal se filtrira svakim od filtera iz banke, a rezultati se na kraju sabiraju kako bi se dobio ekvalizovani signal.



Slika 1 – Različiti profili ekvalizacije iz aplikacije iTunes

Idealno, filteri imaju sledeće popusne opsege:

- Filtar 1: 31.25 Hz – 62.5 Hz
- Filtar 2: 62.5 Hz – 125 Hz
- Filtar 3: 125 Hz – 250 Hz
- Filtar 4: 250 Hz – 500 Hz
- Filtar 5: 500 Hz – 1000 Hz
- Filtar 6: 1000 Hz – 2000 Hz
- Filtar 7: 2000 Hz – 4000 Hz
- Filtar 8: 4000 Hz – 8000 Hz
- Filtar 9: 8000 Hz – 16000 Hz
- Filtar 10: > 16000 Hz

Profili pojačanja u dB za pop, rok i dance sa slike 1 su sledeći:

- Rok: 5, 3.75, 3, 1.5, -0.5, -1.5, 1, 2.5, 3.75, 4.5
- Pop: -1.5, -1, 0, 1.5, 4, 4, 2, 0, -1, -1.5
- Dance: 4, 7, 5, 0, 2, 4, 5, 4.5, 3.5, 0

Kako bi se izbeglo projektovanje filtera sa uskim prelaznim zonama, npr, svega 1 Hz pri učestanosti odabiranja od 44.1 kHz, ekvalizator se najvećim delom pravi korišćenjem jednog digitalnog filtera, ali na koji se dovode signali različitih učestanosti odabiranja. Tako će se u ovom zadatku projektovati samo filter propusnik visokih učestanosti (filtar 10) i jedan filter propusnik opsega (filtar 9), a zatim će se na isti filter propusnik opsega dovoditi signali različite učestanosti odabiranja.

Kao što je poznato, digitalni filter propusnik opsega graničnih relativnih učestanosti F_{p1} i F_{p2} projektovan za učestanost odabiranja f_s propušta frekvencijske komponente između $F_{p1} \cdot f_s$ i $F_{p2} \cdot f_s$. Ako bi se isti taj filter koristio u sistemu u kome je učestanost odabiranja $f_s/2$, rezultujući sistem bi propuštao komponente između $F_{p1} \cdot f_s/2$ i $F_{p2} \cdot f_s/2$. To znači da je moguće koristiti isti digitalni filter za filtriranje različitih frekvencijskih komponenti, ako je moguće smanjiti učestanost odabiranja ulaznog signala. Kako

se u našem sistemu izlazi svih filtara sabiraju, potrebno je vratiti učestanost odabiranja svih izlaznih signala na originalnu pre sabiranja. Postupak promene učestanosti odabiranja je opisan u dodatku na kraju teksta ovog domaćeg zadatka i neophodno ga je razumeti posebno zbog implementacija funkcija u trećem delu domaćeg zadatka. Tokom rada na domaćem zadatku, koristiti gotovu Scipy funkciju **resample_poly** koja radi smanjenje ili povećanje učestanosti odabiranja za određeni faktor, što zavisi od parametara. Na primer, **resample_poly(x, up=2, down=1)** će povećati učestanost odabiranja dva puta, dok će npr. **resample_poly(x, up=1, down=8)** smanjiti učestanost odabiranja 8 puta (vidi dokumentaciju za *Scipy*).

1. Napisati funkciju **[b, a] = bandpass_filter(fs, fa, fp, Aa, Ap)** kojom se projektuje IIR filter propusnik opsega učestanosti. Odabrati odgovarajuću funkciju za analogni prototip. Funkcija kao argumente prima učestanost odabiranja f_s , granične učestanosti nepropusnog opsega f_a kao vektor dve vrednosti, granične učestanosti propusnog opsega f_p kao vektor dve vrednosti i odgovarajuća slabljenja u nepropusnom (α_a) i propusnom (α_p) opsegu. Kao povratnu vrednost, funkcija vraća koeficijente polinoma u imeniocu i brojiocu funkcije prenosa dobijenog filtra.
2. Napisati funkciju **[b, a] = highpass_filter(fs, fa, fp, Aa, Ap)** kojom se projektuje IIR filter propusnik visokih učestanosti. Odabrati istu funkciju za analogni prototip kao u prethodnoj tački. Funkcija kao argumente prima učestanost odabiranja f_s , graničnu učestanost nepropusnog opsega f_a , graničnu učestanost propusnog opsega f_p i odgovarajuća slabljenja u nepropusnom (α_a) i propusnom (α_p) opsegu. Kao povratnu vrednost, funkcija vraća koeficijente polinoma u imeniocu i brojiocu funkcije prenosa dobijenog filtra.
3. Projektovati filter 9 i filter 10 korišćenjem funkcija iz prethodnog zadatka. Prelazne zone treba da budu širine manje od 200 Hz, dok slabljenja u nepropusnom i propusnom opsegu treba da budu $\alpha_a = 0,05$ dB i $\alpha_p = 60$ dB. Ove filtre projektovati tako da im je pojačanje u propusnom opsegu jednako 1. Nacrtati njihove amplitudske i fazne karakteristike i crtanjem odgovarajućih linija za gabarite pokazati da filteri zadovoljavaju zadate specifikacije
4. Napisati funkciju **y = IIR_equalizer(x, fs, style)** kojom se radi zvučna ekvalizacija. Funkcija kao argumente prima odbirke ulaznog signala x , učestanost odabiranja f_s i tip ekvalizacije koja se zadaje kao string 'POP', 'ROCK', 'DANCE' ili 'CUSTOM' na osnovu koga se biraju pojačanja u različitim propusnim opsezima. 'CUSTOM' je profil koji treba zadati prema sopstvenom nahođenju. U okviru ove funkcije, korišćenjem funkcije **resample_poly**, kreirati 9 različitih ulaznih signala koji imaju različite učestanosti odabiranja, svaki duplo manju od prethodnog (npr. $x_1, x_2, x_4, x_8, \dots, x_{256}$). Ove signale koristiti kao ulaze u digitalne filtre pomnožene odgovarajućim pojačanjem i korišćenjem naredbe **lfilter**, kreirati 10 različitih izlaza koje treba vratiti na osnovnu učestanost odabiranja korišćenjem funkcije **resample_poly** u režimu povećanja učestanosti odabiranja. Zbir svih signala je ekvalizovani signal.
5. Dovođenjem impulsa za ulaz, nacrtati impulsni odziv ekvalizatora.
6. DFT-om impulsnog odziva nacrtati približnu frekvencijsku karakteristiku ekvalizatora.
7. Učitati signal iz fajla rock.wav i uraditi ekvalizaciju tog signala proizvoljnim stilom ('POP', 'ROCK', 'DANCE' ili 'CUSTOM'), a rezultat pustiti u IPython playeru.

8. U izveštaju prikazati sve amplitudske karakteristike projektovanih filtara i crtanjem odgovarajućih linija za gabarite pokazati da filtri zadovoljavaju zadate specifikacije, kao i sve druge relevantne dijagrame.

U kodu komentarima jasno naznačiti koji deo koda se odnosi na koji deo zadatka. Obeležiti sve ose odgovarajućim oznakama/tekstom.

Deo 3: Promena učestanosti odabiranja

(12 poena)

U drugom delu domaćeg zadatka projektovan je jedan višebrzinski sistem obrade signala gde se jedan filter koristio na različitim učestanostima odabiranja. Ovaj sistem je potrebno proširiti sopstvenim funkcijama za interpolaciju i decimaciju. Pre nastavka obavezno pročitati dodatak u kome je teorijski objašnjena promena učestanosti odabiranja.

1. Napisati funkciju `h = lowpass_filter_kaiser(Wc, Bt, Aa, Ap)` kojom se projektuje FIR filter propusnik niskih učestanosti. Funkcija kao argumente prima idealnu graničnu učestanost Ω_c , širinu prelazne zone B_t i odgovarajuća slabljenja u nepropusnom (α_a) i propusnom (α_p) opsegu. Kao povratnu vrednost, funkcija vraća impulsni odziv FIR filtra projektovanog odsecanjem idealnog impulsnog odziva Kajzerovim prozorom. Ukoliko gabariti digitalnog filtra nisu zadovoljeni, funkcija treba da pooštrava specifikacije kojima se procenjuju parametar β i red filtra, dok oni ne postanu zadovoljeni. Računati frekvencijske karakteristike u dovoljno velikom broju tačaka prilikom provere, npr. više od 10000.
2. Napisati funkciju `y = dos_resample(x, up_down, r)` kojom se menja učestanost odabiranja ulaznog signala x , za celobrojni faktor r . Učestanost odbiranja se povećava, ako je parametar `up_down` na vrednosti `false`, a smanjuje ako je na vrednosti `true`. Filter potreban za decimaciju ili interpolaciju ima idealnu graničnu učestanost $\Omega_c = \pi/r$ i prelaznu zonu širine $B_t = 0,4\pi/r$. Filter se projektuje korišćenjem funkcije iz prethodne tačke. Slabljenje u nepropusnom opsegu treba da bude minimalno 70 dB, a varijacija amplitude u propusnom opsegu ne sme da pređe 0,05 dB. Primetiti da impulsni odzivi decimacionih/interpolacionih filtera za ovaj slučaj imaju vrednost 0 na indeksima koji su umnožak faktora r . Obratiti pažnju da se filtriranjem i umetanjem nula menja snaga signala, pa je izlaz ili koeficijente filtra potrebno skalirati. Kako bi se uradila ispravna ekvalizacija potrebno je iskompenzovati fazno kašnjenje koje unosi filter za decimaciju/interpolaciju tako da izlazni signal bude u fazi sa ulaznim signalom (o ovome nije bilo potrebno voditi računa u drugom delu jer je ugrađena funkcija već radila kompenzaciju grupnog kašnjenja).
3. Modifikovati funkciju `y = IIR_equalizer(x, fs, style)` iz drugog dela domaćeg zadatka tako da koristi funkciju `dos_resample` umesto Scipy ugrađene funkcije.
4. Dovođenjem impulsa za ulaz, nacrtati impulsni odziv ekvalizatora.
5. DFT-om impulsnog odziva nacrtati približnu frekvencijsku karakteristiku ekvalizatora. Frekvencijsku osu crtati u logaritamskoj razmeri.
6. Učitati signal iz fajla rock.wav i uraditi ekvalizaciju tog signala proizvoljnim stilom ('POP', 'ROCK', 'DANCE' ili 'CUSTOM'), a rezultat pustiti u IPython playeru.
7. U izveštaju prikazati reprezentativne amplitudske karakteristike projektovanih filtera za decimaciju i interpolaciju i crtanjem odgovarajućih linija za gabarite pokazati da filteri zadovoljavaju zadate specifikacije, kao i sve druge relevantne dijagrame.

U kodu komentarima jasno naznačiti koji deo koda se odnosi na koji deo zadatka. Obeležiti sve ose odgovarajućim oznakama/tekstom.

8. Napisati funkciju `y = dos_resample_rat(x, p, q)` kojom se menja učestanost odabiranja ulaznog signala x , za racionalni faktor p/q . Proceniti granične učestanosti filtra potrebnog za decimaciju ili interpolaciju. Filtar se projektuje korišćenjem Parks-MekKlelanovog optimizacionog postupka. Slabljenje u nepropusnom opsegu treba da bude minimalno 70 dB, a varijacija amplitude u propusnom opsegu ne sme da pređe 0,05 dB. Potrebno je iskompenzovati fazno kašnjenje koje unosi filter za decimaciju/interpolaciju tako da izlazni signal bude u fazi sa ulaznim signalom.
9. Izgenerisati reprezentativan signal dovoljne dužine od nekoliko sinusoidalnih komponenti i testirati ispravnost funkcije iz prethodne tačke. Testirati funkciju za nekoliko različitih vrednosti uzajamno prostih parametara p i q ($p < q$, $p > q$). Prikazati rezultate na vremenskim dijagramima, tako što će se na istom grafiku prikazati originalni signal x i izlazni signal y , kao na slici 5 u tekstu drugog domaćeg zadatka.

U kodu komentarima jasno naznačiti koji deo koda se odnosi na koji deo zadatka. Obeležiti sve ose odgovarajućim oznakama/tekstom.

Uputstvo za slanje rešenja domaćeg zadatka

Jedini fajl koji potrebno dostaviti je fajl *dos_dz2_godinaupisa_brojindeksa.ipynb*. Pre slanja **obavezno** očistiti sve izlaze ćelija klikom na *Edit* → *Clear All Outputs* i sačuvati takav fajl.

NE slati fajlove koji su dati kao prilog zadatku jer oni samo povećavaju veličinu fajla i postoji mogućnost da će u tom slučaju mejl biti isfiltriran.

Smatrati da su svi ulazni signali na putanji "*dz2_signali*".

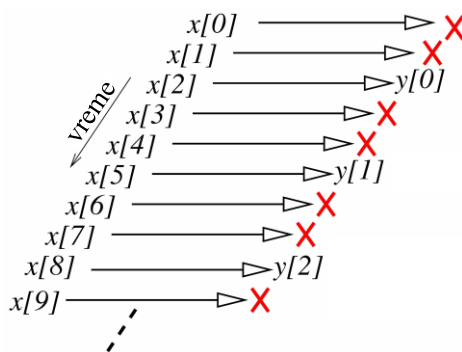
Rešenje zadatka poslati na mejl petrovicv@etf.rs. Naslov mejla treba da bude **19E043DOS – Drugi domaći zadatak GGGG/BBBB**, gde je GGGG godina upisa, a BBBB broj indeksa. Vrlo je važno da mejl bude naslovljen kako je napisano, u suprotnom će biti isfiltriran. Rešenje domaćeg zadatka se mora poslati najmanje 5 dana pre ispita u ispitnom roku u kome se polaže. Posebno, u januarskom ispitnom roku je ovaj rok 21.2.2026. u 23.59, a u februarskom ispitnom roku do dana ispita.

Opšte: Trudite se da napisani kod bude pregledan i detaljno komentarisano. To će vam uštedeti vreme. Nemojte pisati komentare samo da biste zadovoljili zahtev domaćeg zadatka, pokušajte da izvučete prednosti iz preglednog i lepo komentarisano g koda. **Domaći zadatak se radi samostalno. Prepisivanje povlači 0 poena na domaćim zadacima i disciplinsku prijavu.** Domaći zadaci se brane zajedno kada se pošalje drugi domaći zadatak.

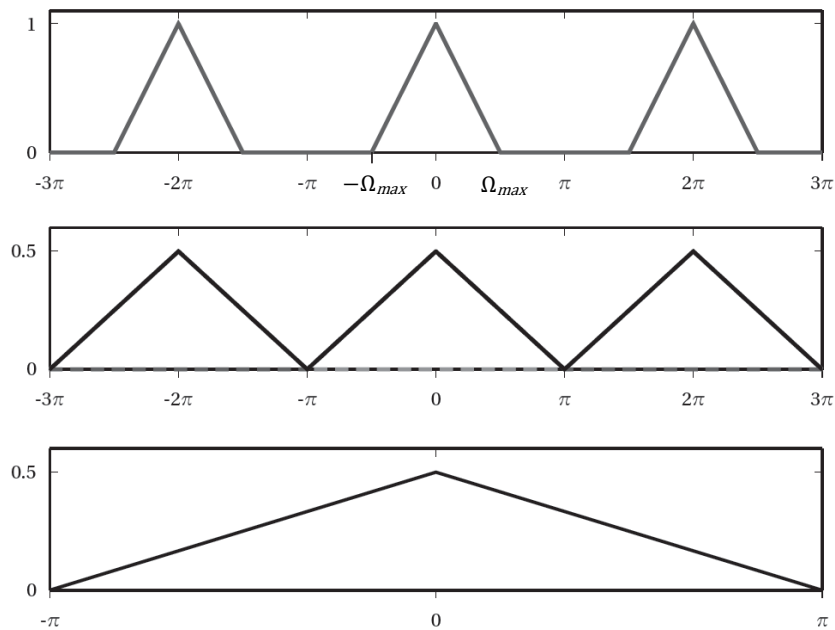
Dodatak: Promena učestanosti odabiranja signala

U sistemima za digitalnu obradu signala često se ukazuje potreba da se promeni učestanost odabiranja. Sistem u kome se koristi više učestanosti odabiranja naziva se višebnzinski (engl. *multirate*) sistem za digitalnu obradu signala. Postupak pri kome se smanjuje učestanost odabiranja signala najčešće se naziva decimacija (engl. *decimation* ili *downsampling*), dok se postupak pri kome se povećava učestanost odabiranja naziva interpolacija (engl. *interpolation* ili *upsampling*).

Decimacija se najlakše radi odbacivanjem odbiraka (slika 2). Na primer, ako se želi smanjiti učestanost odabiranja 2 puta, potrebno je uzeti svaki drugi odbirak iz originalnog signala. S obzirom na to da se menja učestanost odabiranja a originalni analogni signal je isti, spektar decimiranog signala je duplo širi jer sada rellativnoj kružnoj učestanosti π odgovara relativna kružna učestanost $\pi/2$ u originalnom signalu (slika 3).

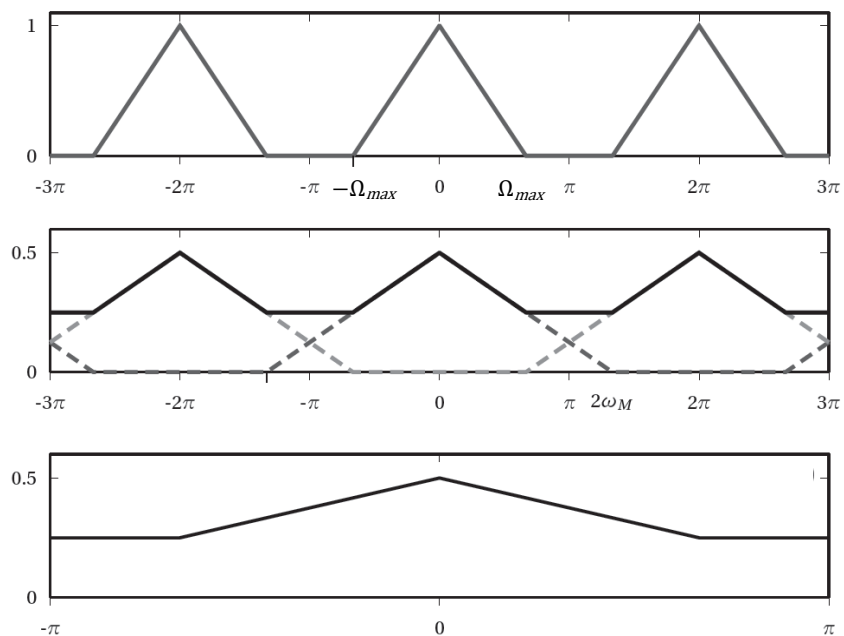


Slika 2 – Ilustracija decimacije



Slika 3 – Spektar originalnog signala koji nema spektralne komponente na učestanostima većim od $\Omega_{max} = \pi/2$ i spektar decimiranog signala kod koga je uzet svaki drugi odbirak iz originalnog signala.

Već naslućujete da se ovde može pojaviti ozbiljan problem. Ako je faktor decimacije veliki, može se pojaviti preklapanje u spektru novoformiranog signala što će izazvati neželjene efekte koji nastaju i prilikom neispravnog odabiranja analognih signala. Primer je na slici 4.



Slika 4 - Spektar originalnog signala koji nema spektralne komponente na učestanostima većim od $\Omega_{max} = 2\pi/3$ i spektar decimiranog signala kod koga je uzet svaki drugi odbirak iz originalnog signala. Primetno je preklapanje u spektru.

Zbog toga je nužno pre odbacivanja odbiraka uraditi niskofrekventno filtriranje (*antialiasing filter*) koje će sprečiti preklapanje u spektru. Jasno je, niskofrekventno filtriranje će uticati na gubitak informacija, ali je svakako mnogo manje štetno od preklapanja u spektru do kog dolazi ako se niskofrekventno filtriranje ne primenjuje.

Na slici 5 je prikazan primer 3 tipa interpolacije. Najjednostavniji način je dodavanjem nula između dva susedna odbiraka. Ovaj način neće dati dobre rezultate, a videćemo i zašto. Bolji rezultati se dobijaju ponavljanjem odbiraka što predstavlja konvoluciju signala kome su dodate nule sa signalom [1 1 1]. Navedeni signal ima dejstvo niskofrekventnog filtra jer je pravouga-oni signal u vremenu *sinc* u frekvencij-skom domenu. Bolji NF filter je linearni interpolator čiji je impulsni odziv signal [0,25 0,5 0,75 1 0,75 0,5 0,25] i rezultat linearne interpolacije je dat na četvrtom grafiku sa slike 4.

Kako bismo detaljnije razumeli postupak interpolacije, posmatrajmo spektar signala kome su dodate nule između svaka dva odbirka, tj. signala

$$y[n] = \begin{cases} x[n/U], & n \bmod U = 0 \\ 0, & n \bmod U \neq 0 \end{cases}$$

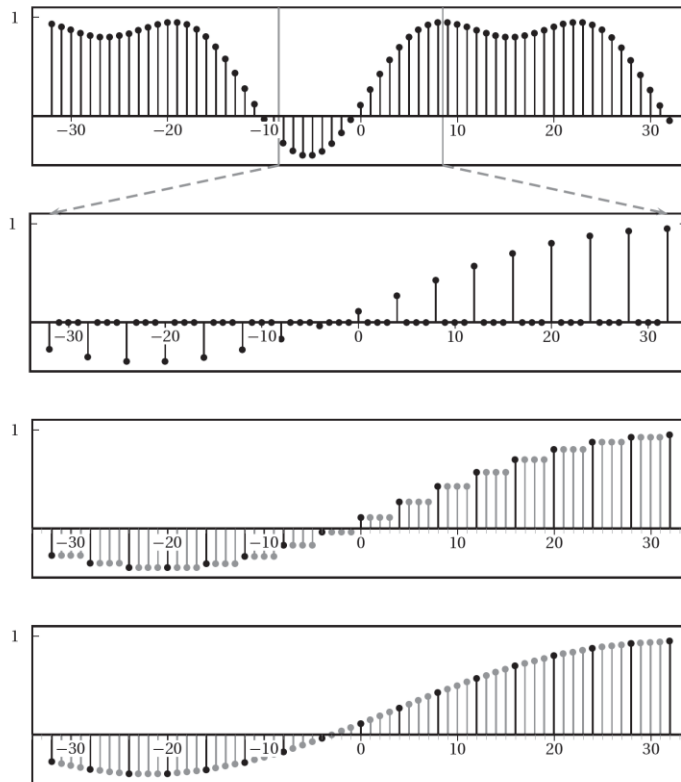
gde je $x[n]$ originalni signal, a $U = 4$ u primeru sa slike 2. Z transformacija signala $y[n]$ je

$$Y(z) = \sum_{n=-\infty}^{\infty} y[n] z^{-n} = \sum_{m=-\infty}^{\infty} x[m] z^{-mU} = X(z^U).$$

Na osnovu veze između Furijeove i Z transformacije gde je $z = e^{j\Omega}$, zaključujemo da je spektar signala $y[n]$ isti spektar kao i spektar signala $x[n]$, samo što mu je frekvencijska osa sabijena U puta.

Na slici 6 je prikazan spektar nekog signala i signala kod koga je $U = 3$. Periodično ponovljeni spektri na visokim učestanostima formiraju nagle promene signala. Interpolacija se dobija poništavanjem ovih periodično ponovljenih delova spektra i zadržavanjem samo osnovnog opsega. Kako bi se ovo uradilo, potrebno je projektovati filter kojim se izdvaja samo osnovni spektar signala.

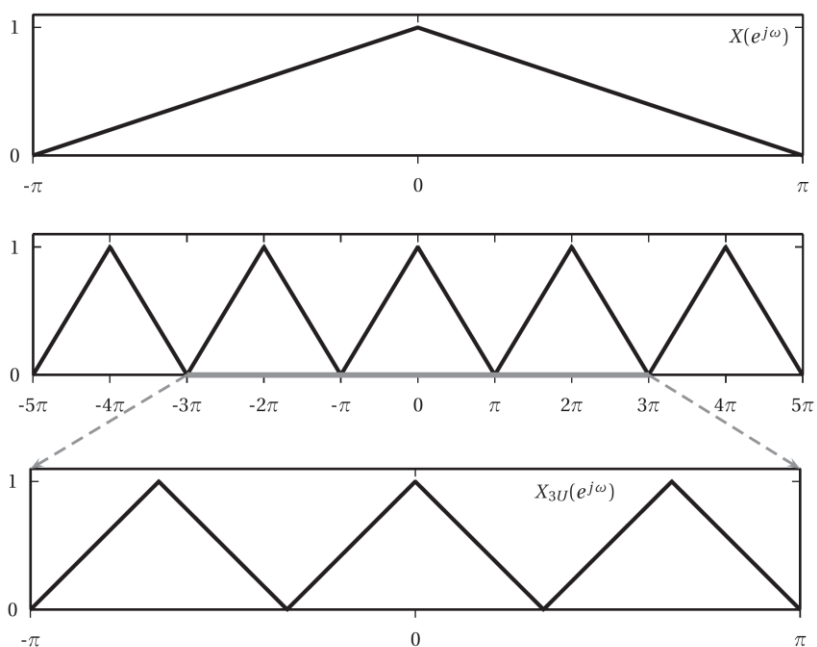
Gorepomenuti načini za decimaciju i interpolaciju mogu da smanje odnosno povećaju učestanost odabiranja signala isključivo celobrojan broj puta. Međutim, u mnogim primenama je potrebno promeniti učestanost odabiranja necelobrojan broj puta $U = P/Q$. U ovakvim situacijama, pogodno je najpre povećati učestanost odabiranja P puta, a zatim interpoliranim signalu smanjiti učestanost odabiranja Q puta. Postupak necelobrojne promene učestanosti odabiranja je prikazan na slici 7. Dakle, najpre se uradi dopunjavanje nulama, zatim niskofrekventno filtriranje koje daje interpolirani signal.



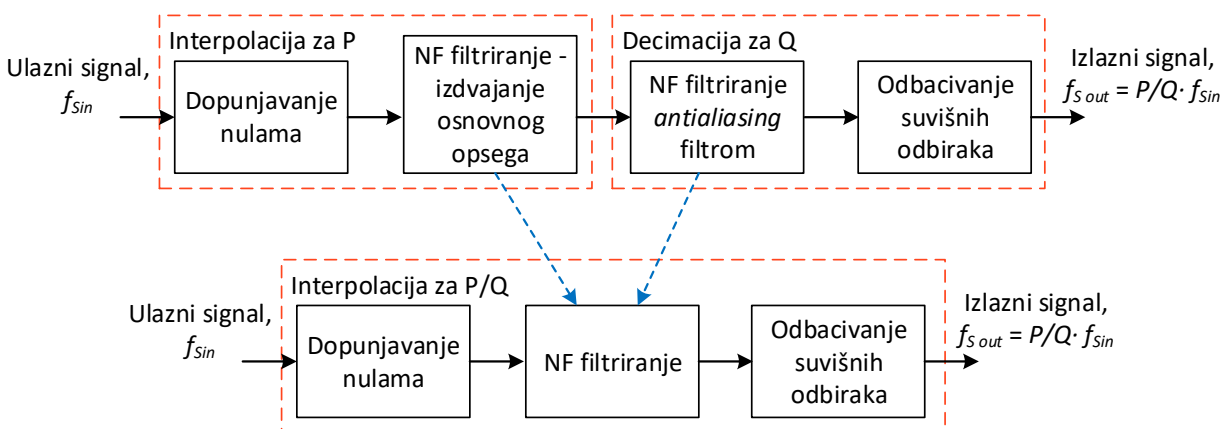
Slika 5 – Primeri interpolacije dodavanjem nula, ponavljanjem odbiraka i linearnom interpolacijom.

Nakon toga, je potrebno filtrirati signal kako ne bi došlo da preklapanja u spektru i na kraju uraditi decimaciju.

Primetićete da u sistemu postoje dva NF filtra jedan iza drugog. Sistem se može pojednostaviti postavljanjem samo jednog NF filtra kod koga je granična učestanost manja od dve granične učestanosti prvobitnih NF filtara. Na taj način je omogućeno da je potrebno projektovati samo jedan NF filter i za necelobrojne promene učestanosti odabiranja.



Slika 6 - Spektar originalnog signala i signala kome su dodate nule tako da je dužina signala 3 puta veća



Slika 7 – Blok šema sistema za necelobrojnu interpolaciju signala

Audio sistemi su vrlo često višebnzinski sistemi. Postoji dosta različitih kvaliteta audio fajlova i oni se razlikuju prema učestanosti odabiranja i prema broju bita za predstavu odbiraka. Tako na primer, signal

govora je dovoljno odabirati sa 8 kHz, dok je za muzički signal neophodno 44,1 kHz. Audio signali na CD-ovima su signali kod kojih je učestanost odabiranja baš 44,1 kHz. Ova učestanost je neophodna jer je opseg zvučnih signala koje čovek može da čuje između 20 Hz i 20 kHz. Međutim, danas se sve češće koriste veće učestanosti odabiranja (88,2 kHz, 96 kHz, 192 kHz...) kako bi se postigao što kvalitetniji snimak.