

# Univerzitet u Beogradu – Elektrotehnički fakultet



## Katedra za elektroniku

### Digitalna obrada signala (19E043DOS)

#### Domaći zadatak 2025/26.

Cilj ovog domaćeg zadatka je da studenti razumeju diskretnu Furijeovu transformaciju, samostalno probaju osnovne metode frekvencijske analize signala korišćenjem diskretne Furijeove transformacije, razumeju efekte preklapanja u spektru, implementiraju algoritam za računanje konvolucije korišćenjem diskretne Furijeove transformacije, kao i da ovladaju elementarnim korišćenjem programskog jezika Pajton u digitalnoj obradi signala.

Domaći zadatak se sastoji iz četiri dela. Prvi deo domaćeg zadatka daje uvid u osnovne probleme koji mogu nastati usled lošeg odabiranja, uvodi diskretnu Furijeovu transformaciju, kao i primene prozorskih funkcija. Drugi deo domaćeg zadatka zahteva implementaciju funkcije za izračunavanje spektrograma i njenu primenu u frekvencijskoj analizi signala. Treći deo domaćeg zadatka predstavlja implementaciju jednog algoritma za brzo izračunavanje konvolucije korišćenjem brze diskretne Furijeove transformacije. Četvrti deo zadatka je simulacija elementarnog digitalnog telekomunikacionog sistema.

Ceo domaći zadatak treba da bude u fajlu *dos\_dz1\_GGGG\_BBBB.ipynb*, pri čemu treba zameniti *GGGG* i *BBBB* odgovarajućim brojevima koji odgovaraju broju indeksa. Sva tekstualna objašnjenja otkucati u tekstualnim ćelijama. Na ovaj način treba integrisati izveštaj i sve kodove u jedan fajl. Potrebno je modifikovati već postojeći fajl i samo njega poslati kao rešenje. Ovo je jedini fajl koji se šalje kao rešenje zadatka! Ne treba slati fajlove sa sajta predmeta. Pročitati uputstvo za slanje rešenja na kraju ovog dokumenta.

## Deo 1: Odabiranje, frekvencijska analiza signala i preklapanje u spektru (10 poena)

### 1.1

Dat je analogni signal  $x(t) = \cos(2\pi f_1 t) + 0,5 \cos(2\pi f_2 t) + 3 \cos(2\pi f_3 t)$ ,  $f_1 = 1$  Hz,  $f_2 = 3$  Hz,  $f_3 = 7$  Hz. Signal se diskretizuje učestanošću  $f_s = 100$  Hz.

1. U Pajtonu proceniti **Furijeovu transformaciju** u dovoljnom broju tačaka i nacrtati amplitudsku karakteristiku digitalnih signala  $x_1[n]$ ,  $x_2[n]$  i  $x_3[n]$  koji su dobijeni odabiranjem signala  $x(t)$  u  $N_1 = 32$ ,  $N_2 = 128$  i  $N_3 = 1024$  tačaka respektivno.
2. Pronaći tri najveće frekvencijske komponente svakog od signala  $x_1[n]$ ,  $x_2[n]$  i  $x_3[n]$  i odrediti da li se iz spektara mogu tačno proceniti učestanosti  $f_1$ ,  $f_2$  i  $f_3$  i sa kojom greškom. Program treba da izračuna tri dominantne učestanosti u spektru sva tri signala, a u *markdown* ćeliji prokomentarisati dobijene rezultate i greške. Primer traženja pozicija maksimalnih vrednosti je dat u skripti *usefulMethods.ipynb* koja prati ovaj tekst.
3. Na signale iz tačke 1 primeniti neku prozorsku funkciju čiji odbirci opadaju do nule ka krajevima. Trajanje te prozorske funkcije je jednako trajanju signala  $N_1 = 32$ ,  $N_2 = 128$  ili  $N_3 = 1024$ . Nacrtati Furijeovu transformaciju modifikovanih signala u istom broju tačaka kao i u tački 1.
4. Kao i u tački 2 pronaći tri najveće frekvencijske komponente svakog od signala  $x_1[n]$ ,  $x_2[n]$  i  $x_3[n]$  i odrediti da li se iz spektara mogu tačno proceniti učestanosti  $f_1$ ,  $f_2$  i  $f_3$  i sa kojom greškom. Prokomentarisati dobijene rezultate i greške.
5. Da li se korišćenjem neke druge prozorske funkcije mogu popraviti rezultati i zašto?

U kodu komentarima jasno naznačiti koji deo koda se odnosi na koji deo zadatka.

Sve vremenske ose u ovom delu treba da budu u sekundama. Frekvencijske karakteristike crtati tako da je frekvencijska osa u hercima od 0 do  $f_s/5$ . Neophodno je obeležiti sve ose odgovarajućim oznakama/tekstom.

### 1.2

Dat je analogni signal  $x(t) = 5 \cos(2\pi f_1 t) + 1000 \cos(2\pi f_2 t) + 10 \cos(2\pi f_3 t)$ ,  $f_1 = 1$  kHz,  $f_2 = 1,3$  kHz,  $f_3 = 1,32$  kHz. Signal se diskretizuje učestanošću  $f_s = 12,5$  kHz.

6. Odrediti minimalan broj odbiraka  $N$  digitalnog signala  $x[n]$  za koju prilikom izračunavanja diskretne Furijeove transformacije ne dolazi do curenja spektra. Obrazložiti u izveštaju. Nacrtati ovaj signal u  $N$  tačaka, kao i njegovu diskretnu Furijeovu transformaciju (realni i imaginarni deo na istoj slici, jedan grafik ispod drugog).

### 1.3

Naredni deo zadatka se odnosi na probleme preklapanja u spektru:

7. U prilogu ovog fajla u direktorijumu *dz1\_signali* nalazi se fajl *singing.wav* u kome je snimljen audio signal. Učitati ovaj signal, nacrtati njegov vremenski oblik i njegov spektar, a zatim pustiti ga na zvučnicima korišćenjem IPython audio plejera.
8. Napisati funkciju koja računa frekvencijski opseg koji zauzima navedeni signal **obw = occupiedBandwidth(x, energyPct, fs)** koja određuje učestanost do koje signal ima **energyPct** procenata ukupne energije signala.

U telekomunikacijama, Occupied Bandwidth (OBW), ili na srpskom zauzeta širina opsega, se najčešće definiše kao opseg učestanosti koji sadrži 99% (ili 99.99%) ukupne energije/snage signala. Kako za

diskretne signale konačnog trajanja važi  $E = \sum_{n=0}^{N-1} |x[n]|^2$  (energija) i  $P = \frac{1}{N} \sum_{n=0}^{N-1} |x[n]|^2$  (srednja snaga), to je ekvivalentno definisati zauzeti širinu ospega preko energije ili preko srednje snage. Da bismo mogli da izračunamo zauzeti opseg potrebno je definisati energiju pomoću koeficijenata diskretne Furijeove transformacije. Za te potrebe možemo iskoristiti Parsevalovu teoremu, koja garantuje da su srednje snage jednake u vremenskom i u frekvencijskom domenu, a za DFT važi:

$$\sum_{n=0}^{N-1} |x[n]|^2 = \sum_{k=0}^{N-1} |X[k]|^2$$

Iz gorenavedenog je lako moguće pronaći koeficijent  $k_{obw}$  za koje je srednja snaga izračunata do tog koeficijenta jednaka **energyPct** procenata ukupne srednje snage.

Ukratko, funkcija treba da izračuna FFT signala, izračuna spektralnu gustinu snage kvadriranjem amplitidske karakteristike FFT-a, izračuna ukupnu energiju, a zatim iterativno odredi za koje  $k_{obw}$  je energija od  $k = 0$  do  $k = k_{obw}$  barem **energyPct** ukupne snage. Iz indeksa  $k_{obw}$  i učestanosti odabiranja  $f_s$ , lako se određuje učestanost koja definiše zauzeti opseg.

9. Korišćenjem navedene funkcije odrediti zauzeti opseg signala iz fajla *singing.wav*. Na osnovu ove informacije zaključiti da li se učestanost odabiranja signala može smanjiti i za koliko najviše. Komentarisati u tekstualnoj ćeliji.
10. Uzimanjem svakog  $m$ -tog odbirka ( $m$  određeno na osnovu rezultata iz tačke 9.) iz polaznog signala  $x[n]$  izgenerisati signal  $x_{dm}[n]$ , Korišćenjem *IPython* biblioteke pustiti novodobijeni signal na zvučnicima. Ne zaboraviti da je sada učestanost odabiranja promenjena.
11. Smanjiti učestanost odabiranja  $l > m$  puta i ponoviti tačku 10. Efekte koji se dobijaju u ovoj i prethodnoj tački objasniti u tekstualnim ćelijama.

U kodu komentarima jasno naznačiti koji deo koda se odnosi na koji deo zadatka.

Sve vremenske ose u ovom delu treba da budu u sekundama. Frekvencijske karakteristike crtati tako da je frekvencijska osa u hercima od 0 do  $f_s/2$ . Neophodno je obeležiti sve ose odgovarajućim oznakama/tekstom.

## Deo 2: Spektrogram

(5 poena)

1. Napisati funkciju `S, f, t = dosSpectrogram(x, fs, window, noverlap, nfft, fMaxShow)` koja kao ulazne argumente prima signal `x` čiji je spektrogram potrebno pronaći, učestanost odabiranja `fs`, vektor odbiraka prozorske funkcije kojom se odsecaju pojedinačni delovi signala `window`, broj tačaka preklapanja susednih prozora `nooverlap`, broj tačaka izračunavanja diskretne Furijeove transformacije (uz dopunjavanje nulama) `nfft`, i maksimalnu učestanost koju treba prikazati `fMaxShow`. Kao izlazne argumente funkcija vraća matricu `S` u kojoj susedne kolone predstavljaju vektore DFT-a pojedinačnih prozora, niz učestanosti `f` kojima odgovaraju koeficijenti DFT-a iz matrice `S` i niz vremenskih trenutaka `t` u kojima počinje svaki od prozora. Ukoliko je ulazni signal realan, funkcija vraća jednostrani spektar gde su učestanosti raspoređene od 0 do `fMaxShow`. Ukoliko je ulazni signal kompleksan, funkcija vraća dvostrani spektar gde su učestanosti u opsegu od `-fMaxShow` do `fMaxShow`. Dozvoljeno je koristiti ugrađenu funkciju za brzu Furijeovu transformaciju, ali nije dozvoljeno koristiti ugrađene funkcije za izračunavanje spektrograma.
2. Kao test primer za napisanu funkciju potrebno je iskoristiti chirp signal. Izgenerisati sinusoidalni signal  $x_{chirp}[n]$  trajanja 5 s kome se relativna učestanost menja linearno sa vremenom od 0 do  $1/2$ , pri čemu je  $f_s = 8$  kHz (*chirp*<sup>1</sup> signal). Korišćenjem *IPython* biblioteke pustiti ovaj signal na zvučnicima.
3. Prikazati vremenski oblik signala i spektrogram chirp signala korišćenjem funkcije napisane u tački 2.1. Nacrtati spektrogram i korišćenjem ugrađene funkcije i uporediti rezultate. Vremenska osa treba da bude u sekundama.
4. Nacrtati spektrogram za minimalno tri različite dužine prozora `window`. Komentarisati u Markdown celiji vremensku i frekvencijsku rezoluciju za svaku dužinu prozora.

Sve vremenske ose u ovoj tački treba da budu u sekundama. Neophodno je obeležiti sve ose odgovarajućim oznakama/tekstom.

---

<sup>1</sup> Linearno frekvencijski modulirani kontinualni *chirp* signal ima sledeći oblik:  $x(t) = \sin\left(\theta_0 + 2\pi\left(f_0 t + \frac{\beta}{2} t^2\right)\right)$ , gde je  $\theta_0$  početna faza,  $f_0$  učestanost u početnom trenutku, a  $\beta$  nagib lineane funkcije po kojoj učestanost raste.

### Deo 3: Implementacija konvolucije dugačkog signala sa kratkim impulsnim odzivom (8 poena)

U prilogu ovog fajla u direktorijumu *dz1\_signali* nalazi se fajl *birds\_airplane.wav* u kome je snimljen cvrkut ptica. Međutim, ptice koje su snimane se nalaze u blizini aerodroma, pa je pored cvrkuta snimljen i zvuk aviona koji poleće sa piste. Za uklanjanje zvuka aviona koristi se sistem čiji je impulsni odziv dat u fajlu *impulse\_response\_birds.pkl* (primer učitavanja je dat u fajlu *usefulMethods.ipynb*).

Potrebno je uraditi konvoluciju signala iz fajla *birds\_airplane.wav* sa impulsnim odzivom korišćenjem metoda "selektuj i sačuvaj" (*Select/Overlap and Save*). Pojedinačne konvolucije se računaju primenom diskretne Furijeove transformacije.

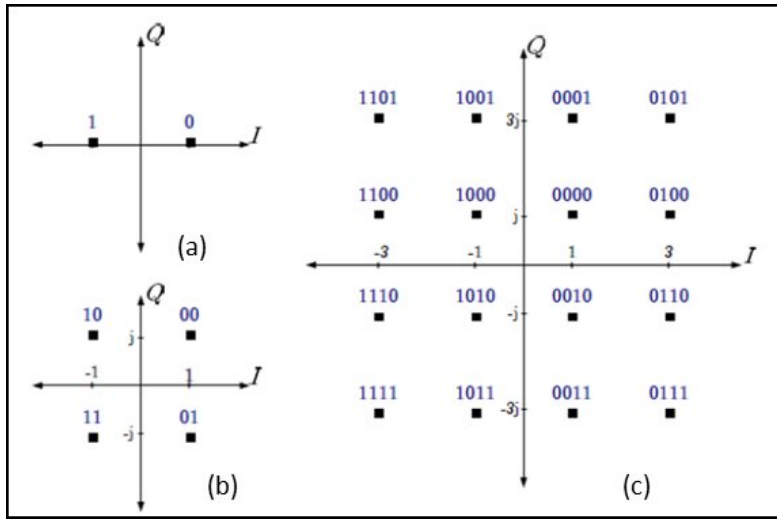
1. Napisati funkciju **blockConvolution(x, h, blockLength)** koja kao ulazne argumente ima vektor odbiraka signala koji se obrađuje, vektor odbiraka impulsnog odziva i dužinu blokova na koje se deli ulazni signal. Kao izlazni argument funkcija vraća vektor obrađenog signala.
2. Korišćenjem funkcije **blockConvolution** izračunati konvoluciju signala iz fajla *birds\_airplane.wav* i pomenutog impulsnog odziva.
3. Korišćenjem funkcije **signal.fftconvolve**, izračunati konvoluciju signala iz prethodne tačke. Nacrtati obe konvolucije i nacrtati signal razlike.
4. Na istoj slici jedan pored druge, nacrtati spektrograme ulaznog i očišćenog signala. Na osnovu spektrograma, zaključiti kakav je filter sistem čiji je impulsni odziv učitao iz fajla *impulse\_response\_birds.pkl*.
5. Generisati proizvoljne signale **x** i **h**, trajanja  $N_x = 100.000$  i  $N_h = 128$  odbiraka nad kojima je potrebno izvršiti blok konvoluciju. Nacrtati grafik zavisnosti vremena izvršavanja od dužine bloka **blockLength** koja treba da uzima vrednosti iz skupa  $blockLength \in \{2^8, 2^9, 2^{10}, 2^{11}, 2^{12}, 2^{13}, 2^{14}, 2^{15}, 2^{16}\}$ . Radi manje zavisnosti od ostalih procesa, prilikom merenja vremena, izvršiti konvoluciju za jednu konfiguraciju barem 100 puta.

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. Frekvencijske karakteristike crtati tako da je frekvencijska osa u hercima od 0 do  $f_s/2$ . Neophodno je obeležiti sve ose odgovarajućim oznakama/tekstom.

## Deo 4: Realizacija elementarnog sistema za prenos digitalnog signala (7 poena)

U digitalnim telekomunikacijama, prenos informacija počinje mapiranjem poruke predstavljene nizom bita  $b[n]$  u niz simbola  $s[n]$ . Umesto direktnog slanja bita, oni se mapiraju u simbole iz tzv. konstelacionog dijagrama. Mapiranje podrazumeva korak u kome se niz digitalnih bita 0 i 1 mapira u niz kompleksnih simbola koji reprezentuju grupe bita. Najjednostavnije mapiranje je BPSK (Binary Phase Shift Keying) kod koga se logička nula mapira u simbol „+1“, a logička jedinica u simbol „-1“ (ili neke druge dve međusobno udaljene kompleksne vrednosti). Viši modulacioni nivoi koriste uređene parove (trojke, četvorke itd.) bita. Nekoliko primera je dato na slici 4.1.



Slika 4.1 – Ilustracija BPSK, QPSK i 16QAM mapiranja

Tako se u opštem obliku mapiranje grupe od  $M$  bita u kompleksni simbol može zapisati kao funkcija čiji su argumenti biti  $b[M \cdot i], b[M \cdot i + 1], \dots, b[(M + 1) \cdot i - 1]$ :

$$s[i] = f(b[M \cdot i], b[M \cdot i + 1], \dots, b[(M + 1) \cdot i - 1]) = A[i]e^{j\theta[i]}.$$

Kao što je rečeno, i za BPSK modulaciju se mogu koristiti drugi simboli iz kompleksne ravni, na primer,  $i$ -ti simbol u nizu se može dobiti na sledeći način:

$$s[i] = \begin{cases} \sqrt{2}(1 + j), & \text{za } b[i] = 0 \\ \sqrt{2}(-1 - j), & \text{za } b[i] = 1 \end{cases}.$$

Niz simbola predstavlja signal koji se može prebaciti u analogni domen DA konverzijom, čime se dobija signal:

$$s(t) = h_{DAC}(t) * \sum_{n=0}^{N-1} s[n]\delta(t - nT_s) = \sum_{n=0}^{N-1} s[n]h_{DAC}(t - nT_s)$$

Ranije smo videli da DA konvertor ima impulsni odziv oblika jediničnog impulsa trajanja  $T_s$ :

$$h_{DAC}(t) = h(t + T_s/2) - h(t - T_s/2) = \begin{cases} 1, & -T_s/2 < t < T_s/2, \\ 0, & \text{inače} \end{cases}$$

gde je sa  $h(t)$  označena Hevisajdova funkcija.

Navedeni signal se može modulisati množenjem sa nosiocem. Uobičajeno je da je nosilac (engl. *carrier*) kompleksna sinusoida:

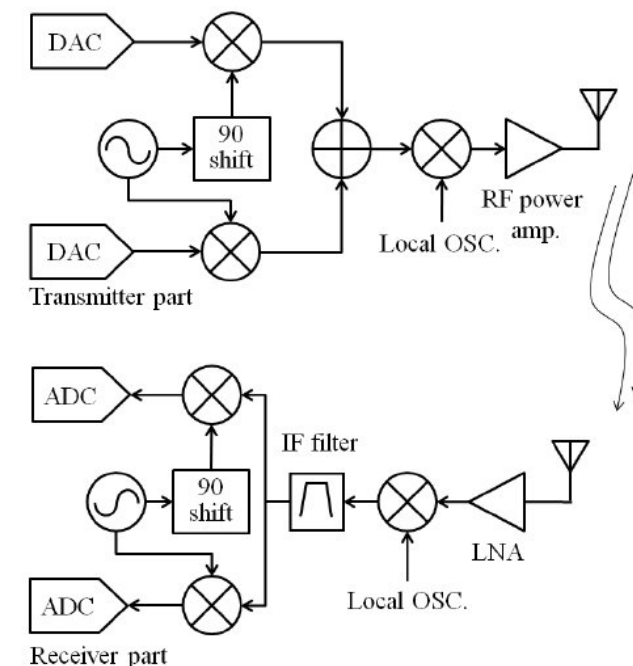
$$c(t) = e^{2\pi f_c t}.$$

Modulisani signal je tada:

$$\begin{aligned} m(t) &= s(t)c(t) = s(t)e^{2\pi f_c t} \\ &= e^{2\pi f_c t} \sum_{n=0}^{N-1} s[n]h_{DAC}(t - nT_s) = e^{2\pi f_c t} \sum_{n=0}^{N-1} A[n]e^{j\theta[n]}h_{DAC}(t - nT_s) \\ &= \sum_{n=0}^{N-1} A[n]e^{2\pi f_c t + j\theta[n]}h_{DAC}(t - nT_s). \end{aligned}$$

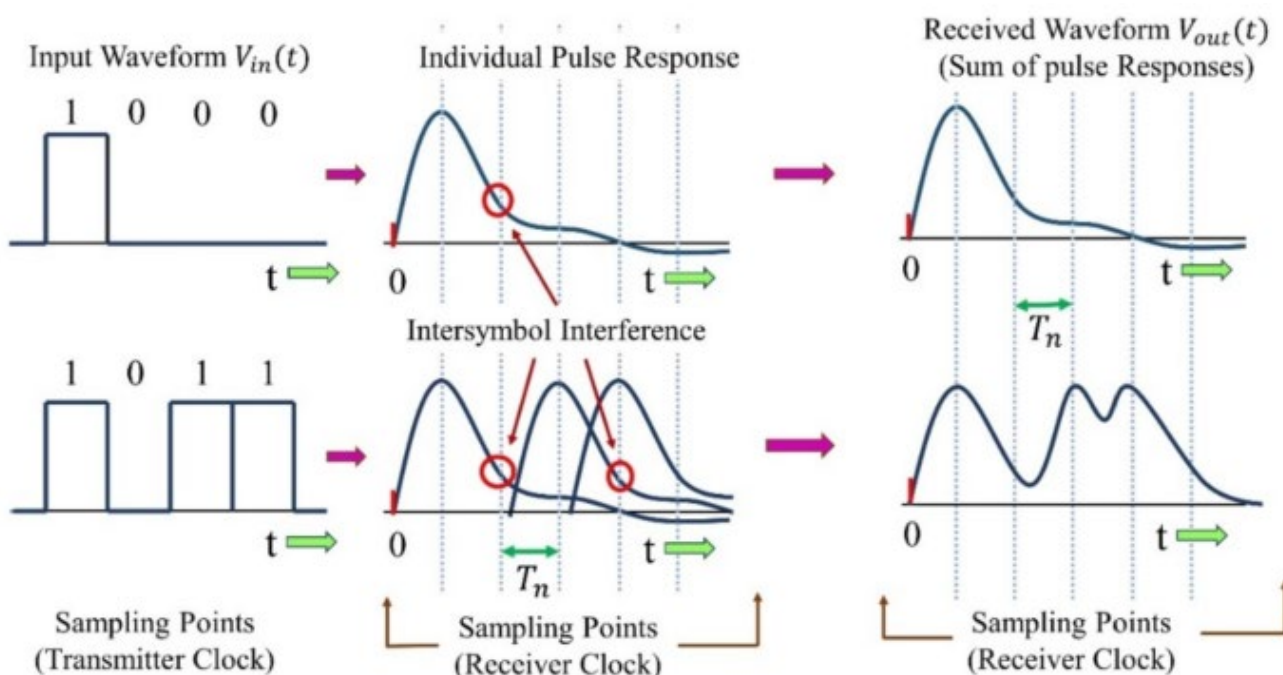
Iz prethodnog izraza se vidi da se modulacijom zapravo menja amplituda i faza nosioca na svakih  $T_s$ , i da je u njima sadržana informacija o poruci.

Matematički je sve čisto, ali postavlja se pitanje kako fizički preneti kompleksan signal. Kako je  $c(t) = e^{2\pi f_c t} = \cos 2\pi f_c t + j\sin 2\pi f_c t$  to je moguće modulisati realni deo signala simbola (deo u fazi, engl. *in-phase*, I) množenjem kosinusnom funkcijom, a imaginarni deo signala simbola (deo u kvadraturi, engl. *quadrature*, Q) množenjem sinusnom funkcijom. Zbir rezultujućih signala se dalje može poslati na pojačavač snage a zatim i antenu direktno ili nakon još jednog množenja radi dodatnog pomeraja signala u spektru. Sinus i kosinus su ortogonalne funkcije pa je na ovaj način omogućeno da se u jednom realnom signalu pošalju informacije iz kompleksnog signala. Inverzan postupak podrazumeva pojačanje niskošumnim pojačavačem, opciono množenje radi pomeraja u spektru unazad, filtriranje opsega od značaja, a zatim ponovo množenje sinusoidalnom i kosinusnom funkcijom. Kako su sinus i kosinus međusobno ortogonalne funkcije, u rezultatima množenja će biti izdvojeni modulisani signali simbola iz kojih se mogu odabiranjem izdvojiti realni i imaginarni deo simbola (kratko objašnjenje: [https://en.wikipedia.org/wiki/Quadrature\\_amplitude\\_modulation#Demodulation](https://en.wikipedia.org/wiki/Quadrature_amplitude_modulation#Demodulation)).



Slika 4.2 – Prikaz kvadraturnog predajnika i prijemnika

Ranije smo videli da signal nakon DA konverzije ima praktično beskonačan spektar, što u telekomunikacijama nije dozvoljeno jer je potrebno da više korisnika dele dostupan elektromagnetski spektar. Nakon DA konverzije se svakako postavlja filter propusnik niskih učestanosti kojim se potiskuju kopije spektra. Ipak, u telekomunikacijama je bitan još jedan faktor, a to je intersimbolska interferencija (engl. *inter-symbol interference, ISI*). ISI nastaje kada se na prijemu usled nesavršenosti analognih filtera u trenutku odabiranja, na mestu  $i$ -tog simbola pojavi linearna kombinacija više poslatih simbola. Idealno bi bilo da je primljeni simbol samo verna predstava isključivo jednog poslatog simbola kako bi greške u prenosu bile minimalne. Ilustracija nastanka ISI je prikazana na slici 4.3. Na primeru je pojednostavljena modulacija gde se šalju samo nule ili jedinice, ali efekti su identični. Ako je odziv jednog impulsa takav da on nije nula u trenucima odabiranja drugih simbola na prijemu, jasno je da će se njegov doprinos pojaviti i u drugim simbolima. Dakle, on interferira sa drugim simbolima.



Slika 4.3 – Prikaz nastanka intersimbolske interferencije

Zbog svega ovoga je najčešće brzina komuniciranja manja od same učestanosti odabiranja, kako bi se svi prelazni procesi koji nastaju usled impulsnog odziva analognih filtera završili pre trenutka evaluacije narednog simbola. Dakle, jedan simbol u analognom domenu će biti predstavljen pomoću nekoliko odbiraka signala  $s(t)$ .

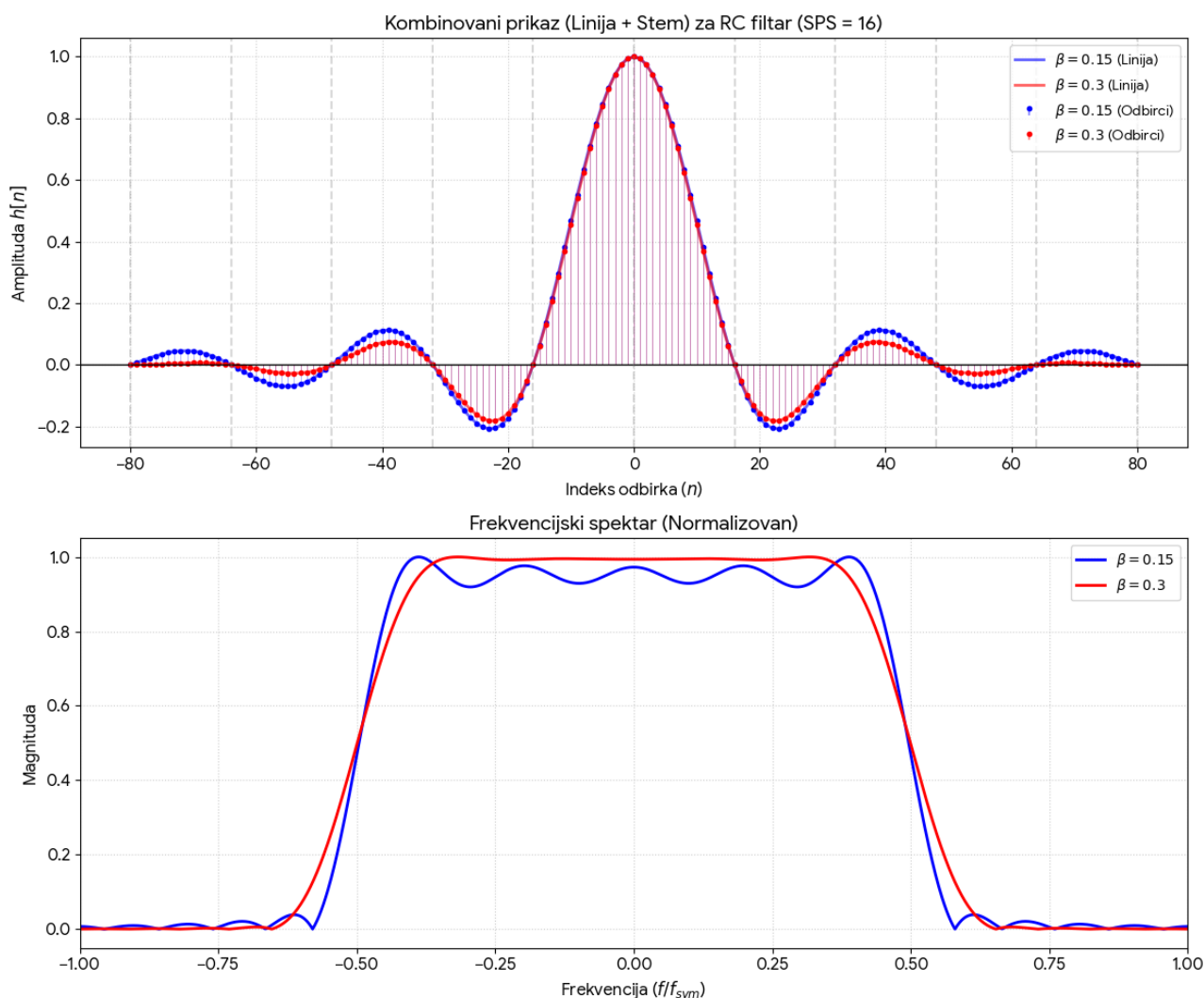
Kako bi signal  $s(t)$  bio ograničen u spektru potrebno je svaki simbol predstaviti impulsom čija je frekvencijska karakteristika uska. Da bi se izbegla ISI u digitalnom domenu, ekvivalentni impulsni odziv od predaje do prijema mora biti takav da ima jedinicu na mestu evaluacije simbola od interesa, a nule na mestima evaluacije susednih simbola. Takav filter je tzv. raised cosine (RC) filter čiji impulsni odziv ima oblik:

$$h_{RC}[n] = \text{sinc}\left(\frac{n}{N_{SPS}}\right) \frac{\cos\left(\frac{\pi\beta n}{N_{SPS}}\right)}{1 - \left(\frac{2\beta n}{N_{SPS}}\right)^2},$$

gde je  $N_{SPS}$  broj odbiraka po simbolu, a  $\beta$  je parameter kojim se podešava selektivnost filtera, tj. širina prelazne zone. Na slici 4.4 je prikazan primer impulsnog odziva RC filtera ako je  $N_{SPS} = 16$ . Primetiti da je na



indeksima  $kN_{SPS}$  impulsni odziv jednak nuli. Osa za frekvencijsku karakteristiku je predstavljena relativno u odnosu na frekvenciju simbola  $f_{sym}$ .



Slika 4.4 – Impulsni odziv RC filtra i odgovarajuće frekvencijske karakteristike

Kako bi sistem imao ekvivalentan impulsni odziv koji odgovara RC filtru, i predajnik i prijemnik filtriraju simbole filtrom čija konvolucija sa samim sobom daje RC filtar. Takav filtar se zove root-raised cosine (RRC) filtar i njegov impulsni odziv je:

$$h_{RRC}[n] = \frac{4\beta}{\pi\sqrt{N_{SPS}}} \frac{\cos\left(\frac{(1+\beta)\pi n}{N_{SPS}}\right) + \frac{N_{SPS}}{4\beta n} \sin\left(\frac{(1-\beta)\pi n}{N_{SPS}}\right)}{1 - \left(\frac{4\beta n}{N_{SPS}}\right)^2}$$

Modulaciju je moguće raditi i u digitalnom domenu, množenjem kompleksnom sinusoidom. Ako je učestanost odabiranja DA konvertora dovoljno velika, moguće je direktno iz digitalnog domena generisati signal za predaju. U ovom domaćem zadatku, primer je pojednostavljen i modulacija će biti rađena samo u digitalnom domenu.

1. U fajlu `usefulMethods.ipynb` dostupan je kod za konverziju teksta u niz bita koji predstavljaju informaciju koju je potrebno preneti. Iskoristiti taj kod za generisanje binarne poruke proizvoljnog teksta dužine oko 100 karaktera.

2. U fajlu `usefulMethods.ipynb` dostupan je kod za funkciju `rrcFilter(beta, Nsps, filter_len_symbols)` koja vraća koeficijente impulsnog odziva RRC filtra korišćenjem gorenavedene formule. Parametar `filter_len_symbols` predstavlja dužinu impulsnog odziva u simbolima.
3. Za potrebe domaćeg zadatka, koristićemo  $N_{\text{sps}} = 16$  i  $\beta = 0.3$ . Generisati impulsni odziv RRC filtra trajanja 8 simbola ( $8 \cdot 16 + 1 = 129$  odbiraka). Nacrtati vremenski oblik i Furijeovu transformaciju generisanog signala. Zatim, nacrtati vremenski oblik i Furijeovu transformaciju konvolucije dva RRC filtra.

#### **Predajnik**

4. Signal poruke mapirati BPSK modulacijom i to tako da je

$$s[i] = \begin{cases} \sqrt{2}(1 + j), & \text{za } b[i] = 0 \\ \sqrt{2}(-1 - j), & \text{za } b[i] = 1 \end{cases}$$

5. Zatim proširiti BPSK signal umetanjem 15 nula između svaka dva simbola.
6. Izračunati konvoluciju dobijene proširene poruke (odvojeno njenog realnog i imaginarnog dela) i impulsnog odziva RRC filtra. Nacrtati prvih 128 odbiraka realnog i imaginarnog dela dobijene konvolucije. Primetiti da signal kasni 64 odbirka u odnosu na ulaz. Takođe, primetiti da simboli sada liče na impulsni odziv RRC filtra. Zbog toga se i RRC filter naziva *pulse-shaping* filter. Nacrtati Furijeovu transformaciju dobijenog signala. Primetiti da je spektar ograničen prema spektru RRC filtra.
7. Sada je moguće uraditi modulaciju. Ako je učestanost odabiranja DA konvertora 1.6 GHz, jasno je da je brzina komuniciranja 100 MSps (megasimbola u sekundi) odnosno 100 MHz. Ako je učestanost nosioca 400 MHz, modulirati signal iz prethodne tačke množenjem kompleksnom sinusoidom nosioca. Nacrtati spektar dobijenog modulisanog signala. Frekvencijska osa u ovoj tački treba da bude u hercima. Primetiti da je spektar ograničen prema spektru RRC filtra i pomeren oko učestanosti nosioca.

#### **Kanal**

8. Kanal sa aditivnim belim Gausovim šumom (engl. *additive white Gaussian noise, AWGN*) je jedan od najjednostavnijih kanala u telekomunikacijama. Na poslati signal ovaj kanal samo dodaje beli Gausov šum, tj. signal koji predstavlja Gausov slučajni proces (svaki odbirak predstavlja kompleksan broj gde su i realni i imaginarni deo slučajne promenljive generisane iz Gausove raspodele). Za početak, ovaj korak možemo preskočiti. Šum ćemo dodati tek na kraju, kada budemo uspešno rekonstruisali signal u sistemu bez šuma.

#### **Prijemnik**

9. Na strani prijemnika, potrebno je modulirani signal vratiti u osnovni opseg množenjem odbircima kompleksne sinusoide

$$c_r(t) = e^{-2\pi f_c t}$$

10. Dobijeni signal filtrirati istim RRC filtrom. Ovaj filter se naziva prilagođeni filter (engl. *matched filter*). Jedan od najvažnijih rezultata u teoriji signala je Teorema o prilagođenom filteru. Ona kaže da je za detekciju signala poznatog oblika u prisustvu belog šuma, optimalni linearni filter onaj čiji je impulsni odziv "ogledalo" samog signala.
11. Rekonstruisane BPSK simbole je moguće dobiti uzimanjem svakog 16. odbiraka iz rezultujućeg signala. Ipak, pošto svaki od dva filtra unosi kašnjenje od po 64 odbirka, potrebno je odbaciti prvih 128 odbiraka pre evaluacije BPSK simbola.
12. Dobijene BPSK simbole  $\hat{s}[n]$  prebaciti u niz bita  $\hat{b}[n]$  korišćenjem tzv. tvrdog odlučivanja na osnovu euklidskog rastojanja:

$$\hat{b}[i] = \begin{cases} 0, & \text{ako je } \sqrt{(\hat{s}_R[n] - \sqrt{2})^2 + (\hat{s}_I[n] - \sqrt{2})^2} < \sqrt{(\hat{s}_R[n] + \sqrt{2})^2 + (\hat{s}_I[n] + \sqrt{2})^2} \\ 1, & \text{ako je } \sqrt{(\hat{s}_R[n] - \sqrt{2})^2 + (\hat{s}_I[n] - \sqrt{2})^2} \geq \sqrt{(\hat{s}_R[n] + \sqrt{2})^2 + (\hat{s}_I[n] + \sqrt{2})^2} \end{cases}$$

13. Korišćenjem postojećeg koda iz `usefulMethods.ipynb` konvertovati niz bita u tekstualnu poruku.  
Ukoliko je tekst identičan, najveći deo zadatka je uspešno urađen.
14. Vratiti se u tačku 8 i na signal dodati šum. Ponoviti izvršavanje koda dok se ne uoče greške u rekonstruisanom tekstu.

## Uputstvo za slanje rešenja domaćeg zadatka

Jedini fajl koji potrebno dostaviti je fajl `dos_dz1_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 `"dz1_signali"`, baš kao u primerima koji prate ovaj tekst.

Rešenje zadatka poslati asistentu na mejl [petrovicv@etf.rs](mailto:petrovicv@etf.rs) najkasnije **do nedelje 25.1. u 23:59**. Naslov mejla treba da bude **19E043DOS – Prvi domaci 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. Obaveštenja o odbrani domaćeg zadatka biće naknadno postavljena na sajtu predmeta.

**Opšte:** Trudite se da napisani kod bude pregledan i detaljno komentaran. To će vam uštedeti vreme. Nemojte pisati komentare samo da biste zadovoljili zahtev domaćeg zadatka, pokušajte da izvučete prednosti iz preglednog i lepo komentaranog koda. **Domaći zadaci se rade samostalno. Prepisivanje povlači 0 poena na domaćem zadatku i disciplinsku prijavu.**