



Sveučilište u Zagrebu  
Fakultet elektrotehnike i računarstva  
Zavod za elektroničke sustave i obradu informacija

# Primjene DFT i DCT transformacija, projektiranje filtara i filtarski slogovi

## Obrada informacija (FER-2) - Laboratorijska vježba 2.

### 1. Uvod

Laboratorijske vježbe iz Obrade informacija se izvode na računalu, a koristi se programski sustav MATLAB. Osim što posjeduje mogućnost izvođenja raznih jednostavnih i izuzetno složenih matematičkih operacija, MATLAB ima i modul Simulink koji je zamišljen kao alat za brzo i jednostavno simuliranje raznih sustava. S MATLAB-om i Simulink-om ste se upoznali na [LiV-u MATLAB](#).

### 2. Priprema

Prisjetite se kako se pišu MATLAB m-funkcije te kako se koristi Simulink (vježbe [2.](#) i [4.](#) sa [LiV-a MATLAB](#)). Kao podsjetnik vam osim materijala korištenih na [LiV-u MATLAB](#) može poslužiti i priručnik [Kratke upute za korištenje MATLAB-a](#) koji je dostupan na [stranicama predmeta](#).

Prije dolaska na laboratorijske vježbe korisno je pročitati sve zadatke koje ćemo raditi. Primijetite da uz neke od zadatke piše (PRIPREMA). To su zadaci koje morate riješiti prije dolaska na vježbu. Rješenja uredno napišite rukom na papiru. U zaglavlje svakog papira s pripremom napišete vaše ime i prezime te JMBAG.

Pripremni zadaci su: 3.1-1a, 3.1-1b, 3.1-2a, 3.2-1a, 3.2-2a, 3.3-1a, 3.3-2a

### 3. Rad u laboratoriju

Prije početka rada uključite dnevnik (naredba diary). Bez obzira na dnevnik također vam preporučamo da rješenje svakog zadatka spremite kao MATLAB m-skriptu ili Simulink model čije ime je redni broj zadatka.

Bodove iz laboratorija stječete tijekom vježbi. Druga laboratorijska vježba nosi tri boda. Točno napisana priprema vam donosi jedan bod. U zadnjem satu laboratorijskih vježbi (ili prije ako ste prije gotovi s vježbom) dežurni asistent će obići studente te svakog od vas zamoliti da demonstrira rješenja zadanih zadataka. Ako uspješno demonstrirate zadatke<sup>1</sup> (jedan bod) i ako točno odgovorite na postavljeno pitanje vezano uz vježbe (još jedan bod) dobivate preostala dva boda.

---

<sup>1</sup> Demonstracija će vam biti jednostavnija ako ste napisali m-skriptu - jednostavno pozovete tu skriptu. Ako niste pisali m-skripte iskoristite dnevnik. Dnevnik otvorite u ugrađenom editoru i odaberete dio koji odgovara rješenju zadatka kojeg tražimo. Pritiskom na desnu tipku miša

Svaki student može samo jednom demonstrirati rješenje zadatka i odgovarati. Dakle ako krivo odgovorite ili ako je demonstrirano rješenje zadatka krivo ne dobivate bodove. NEMA popravnih zadataka!

Dio zadataka označen je kao (PRIPREMA), dok je dio označen kao (ZA ONE KOJI ŽELE ZNATI VIŠE). Zadatke za pripremu ste riješili prije vježbi, dok zadatke za one koji žele znati više ne trebate rješavati. Predlažemo da ih preskočite te se vratite na njih ako obavezne zadatke završite prije predviđenog vremena.

## 3.1. Spektralna analiza

Prvi dio prve laboratorijske vježbe se bavi diskretnom Fourierovom transformacijom i spektralnom analizom signala.

### 30 minuta **Zadatak 3.1-1 Detekcija DTMF signala korištenjem DFT<sub>205</sub> transformacije**

U telefonskim sustavima se za signalizaciju koristi linearna kombinacija dva tona od kojih svaki može poprimiti neku od četiri predefinirane frekvencije što ukupno omogućuje prenošenje 16 različitih simbola. Takva signalizacija se naziva DTMF što je kratica od *dual-tone multi-frequency*. Odabrane četiri frekvencije za oba tona te značenje svake od mogućih kombinacija prikazano je u tablici 1. Primijetimo da se sve frekvencije nalaze unutar frekvencijskog opsega od 0 do 4 kHz koji je predviđen za telefonski prijenos govornog signala.

Tablica 1. Dozvoljene frekvencije i značenje kombinacije

	1209 Hz	1336 Hz	1477 Hz	1633 Hz
697 Hz	1	2	3	A
770 Hz	4	5	6	B
852 Hz	7	8	9	C
941 Hz	*	0	#	D

U ovom zadatku razmotriti ćemo kako korištenjem diskretne Fourierove transformacije možemo dekodirati poslani DTMF signal. Kako se DTMF signal sastoji od čistih harmonijskih komponenti za dekodiranje signala je potrebno odrediti spektar signala te ispitati koje dvije čiste harmonijske komponente signala se javljaju. Kako ćemo za dekodiranje koristiti računalo i DFT transformaciju u  $N$  točaka postavlja se pitanje koji  $N$  odabrati tako da dobijemo pouzdanu detekciju signala. Pri tome primijetite da smanjivanjem vrijednosti  $N$  smanjujemo računalnu složenost no istodobno kvarimo frekvencijsku razlučivost. Želimo dakle odabrati najmanji  $N$  koji nam daje zadovoljavajuću frekvencijsku razlučivost.

Ako znamo da je telefonski signal otipkan frekvencijom <sup>2</sup> 8 kHz diskretna Fourierova transformacija u  $N$  točaka nam daje spektralne komponente za frekvencije koje su višekratnici od  $8000/N$ . Za DTMF signalizaciju dobar izbor broja točaka  $N$  koji

---

dobivate izbornik iz kojega je potrebno odabrati Evaluate Selection. MATLAB tada izvršava označeni dio te crta sve slike i prikazuje sve rezultate.

<sup>2</sup> Digitalni telefonski signal je kvantiziran s 8 bita i otipkan frekvencijom od 8 kHz što daje ukupnu brzinu prijenosa podataka 64 kb/s.

daje mala odstupanja između DTMF frekvencija i cjelobrojnih višekratnika od  $8000/N$  je  $N = 205$ . Odstupanja za odabrani  $N = 205$  su navedena u tablici 2.

Tablica 2. Odstupanja za DTMF frekvencije i DFT u 205 točaka

Osnovni ton [Hz]	Točna vrijednost k	Najbliži cjelobrojni k	Apsolutna pogreška za k
697	17,861	18	0,139
770	19,731	20	0,269
852	21,833	22	0,167
941	24,133	24	0,113
1209	30,381	31	0,019
1336	34,235	34	0,235
1477	37,848	38	0,152
1633	41,846	42	0,154

- (PRIPREMA) Napišite MATLAB m-skriptu ili funkciju koja za zadani ulazni znak generira DTMF signal trajanja 80 ms otipkan frekvencijom 8000 Hz. Kao predložak možete koristiti funkciju `dtmfcode` dostupnu na stranicama predmeta.
- (PRIPREMA) Napišite MATLAB m-skriptu ili funkciju koja temeljem 205 uzoraka signala otipkanog frekvencijom 8000 Hz dekodira DTMF signal. Kao predložak možete koristiti funkciju `dtmfdecode` dostupnu na stranicama predmeta.
- Ako znate da je minimalno trajanje DTMF signala 50 ms te da između dva različita signala mora biti 45 ms tišine kreirajte i poslušajte signal koji je potrebno poslati telefonskoj centrali ako bi željeli birati broj koji je jednak vašem matičnom broju.

Za potrebe vježbi je napisana jednostavna funkcija `dtmfcode` koju možete preuzeti sa stranica predmeta. Korištenjem te funkcije DTMF signal generiramo i slušamo na sljedeći način:

```
» help dtmfcode
```

```
DTMFCODE - Generiranje DTMF signala za zadani znak.
y = DTMFCODE(x) generira DTMF signal duljine 640 uzoraka otipkan
frekvencijom 8000 Hz za ulazni simbol x.
```

```
y = DTMFCODE(x, fs, N) generira DTMF signal temeljem znakovnog
niza x otipkan frekvencijom fs gdje je svaki znak iz x
zamijenjen DTMF tonom trajanja N uzoraka.
```

```
» broj=dtmfcode('0 0 3 6 3 6 1 8 4 9'); % kreiramo DTMF signal
» soundsc(broj, 8000) % slušamo ga
```

- Korištenjem funkcije za detekciju DTMF signala iz pripreme i korištenjem već pripremljene funkcije `dtmfdecode` dekodirajte DTMF signal iz zadatka b).

```
» help dtmfdecode
```

```
DTMFDECODE - Dekodira DTMF signal otipkan frekvencijom 8000Hz.
y = DTMFDECODE(x) vraća znakovni niz koji sadrži dekodirane
znakove za ulazni zvučni signal x za svaki nepreklapajući blok
duljine 205 uzoraka. Signal x mora biti otipkan frekvencijom
otipkavanja od 8000Hz.
```

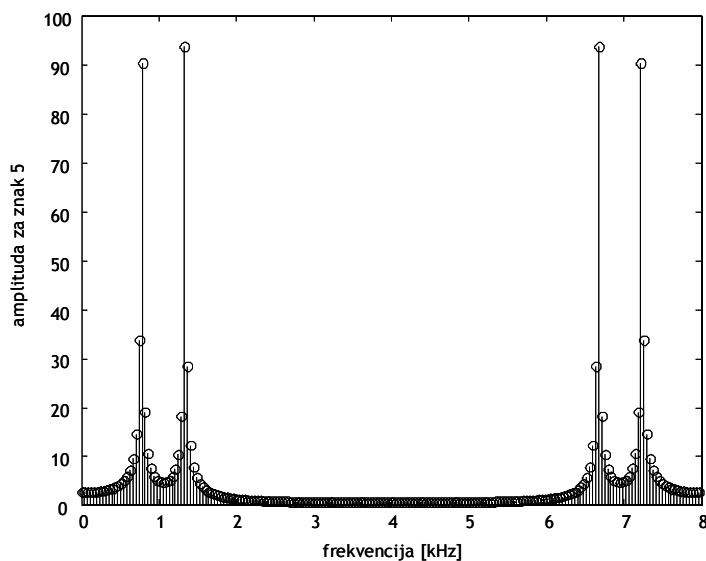
```
» dtmfdecode(broj) % dekodiramo broj
```

ans =

% kako koristimo najjednostavniji dekodier  
% primijetite da je potrebna dodatna obrada

000      00      33      66      333      66      11      88      444      99

- e) Odaberite dva DTMF simbola. Za svaki od odabranih simbola generirajte DTMF signal trajanja 205 uzoraka te nacrtajte pripadni amplitudni spektar  $DFT_{205}$  transformacije. Sastoji li se spektar samo od dvije čiste harmonijske komponente? Možete li objasniti dobiveno razmazivanje spektra (eng. *spectral leakage*)? Što se događa s razmazivanjem ako odaberemo li manji broj točaka, a što ako odaberemo veći?



Slika 1. Amplitudni spektar  $DFT_{205}$  transformacije za DTMF simbol 5

Tablica 3. Odstupanja drugih harmonika DTMF frekvencija za  $DFT_{201}$

Drugi harmonik [Hz]	Točna vrijednost k	Najbliži cjelobrojni k	Apsolutna pogreška za k
1394	35,024	35	0,024
1540	38,692	39	0,308
1704	42,813	43	0,187
1882	47,285	47	0,285
2418	60,752	61	0,248
2672	67,134	67	0,134
2954	74,219	74	0,219
3266	82,058	82	0,058

- f) (ZA ONE KOJI ŽELE ZNATI VIŠE) Dodatni zahtjevi na DTMF dekodier su: 1) samo dvije frekvencije smiju biti prisutne u signalu dok ostalih 6 mora biti slabije od -55 dBm; 2) trajanje DTMF signala mora biti dulje od 40 ms (ANSI T1.401-1988 zahtijeva 50 ms); 3) pauza između dva simbola mora biti dulja od 40 ms (ANSI T1.401-1988 zahtijeva 45 ms); 4) jačina DTMF signala mora biti između -25 dBm i 0 dBm; 5) dozvoljeno odstupanje amplitude između niže i više frekvencije smije biti između -8 dB i +4 dB;

6) eventualni signali na telefonskoj liniji koji nisu DTMF signali moraju biti ispod -6 dB u odnosu na frekvencijski nižu komponentu DTMF signala. Modificirajte vaš DTMF dekodirer tako da zadovoljava navedene zahtjeve.

Za svojstvo 6) se obično uzimaju u obzir frekvencijske komponente navedene u tablici 3.

- g) (ZA ONE KOJI ŽELE ZNATI JOŠ VIŠE) Opisani DTMF dekodireri temelje se na  $DFT_{205}$  i  $DFT_{201}$  transformaciji, no koriste samo 8 uzoraka spektra svake. U slučaju kada nam nisu potrebne vrijednosti spektra za sve frekvencije već samo za manji broj njih koristi se Goertzelov algoritam. Zamijenite  $DFT_{205}$  i  $DFT_{201}$  transformacije u vašem detektoru Goertzelovim algoritmom. Opis Goertzelovog algoritma možete naći u [1], poglavlje 8.

### 15 minuta Zadatak 3.1-2 Spektrogram i DFT filtarski slog

U prošlom zadatku smo se bavili dekodiranjem DTMF signala. Dekodiranje se temeljilo na spektralnoj analizi signala, odnosno na računanju DFT transformacije i analizi odabranih komponenti. Računamo li DFT transformaciju signala blok-po-blok i prikazemo li kvadrat amplitude dobivamo procjenu spektra snage za svaki blok, spektrogram.

Prilikom računanja spektrograma moramo odabrati veličinu bloka (koja određuje DFT) te koliko će biti preklapanje između blokova. Odaberemo li kratki blok računamo DFT u malom broju točaka te dobivamo lošu frekvencijsku rezoluciju. Pripadni spektrogram se naziva širokopojasni spektrogram (eng. *wide-band spectrogram*). Odaberemo li pak dugi blok dobivamo dobru frekvencijsku rezoluciju, no kvarimo vremensku lokalizaciju pojedine frekvencijske komponente. Takav spektrogram se naziva uskopojasni spektrogram (eng. *narrow-band spectrogram*).

- a) (PRIPREMA) Ponovite i navedite svojstva vremenskih otvora (širina glavne latice, gušenje prve bočne latice, širina prijelaznog područja) i to za pravokutni, Bartlettov, Hannov, Hammingov i Blackmanov vremenski otvor.
- b) Korištenjem naredbe `dtmfcode` generirajte signal koji sadrži sve simbole dostupne na običnom telefonskom aparatu. Korištenjem naredbe `specgram` nacrtajte spektrogram DTMF signala uz korištenje pravokutnog vremenskog otvora u 205 točaka. Možete li iz dobivenog spektrograma zaključiti je li moguće razlikovati poslane simbole?
- ```
» broj=dtmfcode('1 2 3 4 5 6 7 8 9 0 * #'); % kreiramo DTMF signal
» N = 205; % blok je duljine 205 uzoraka
» fs = 8000;
» specgram(broj,N,fs,rectwin(N)) % crtamo spektrogram s
                                % duljinom bloka 205 gdje smo
                                % na svaki blok primijenili
                                % pravokutni vremenski otvor
```
- c) Za signal i prethodnog zadatka nacrtajte spektrograme za duljine bloka od 128, 256, 512 i 1024 uzoraka uz korištenje pravokutnog vremenskog otvora. Što se događa s vremenskom, a što s frekvencijskom razlučivošću? Kada je razlikovanje simbola moguće, a kada nije?
- d) Ponovite prethodni zadatak uz korištenje Blackmanovog vremenskog otvora. Kako otvor utječe na frekvencijsku razlučivost? Povežite utjecaj otvora sa svojstvima otvora!
- e) (ZA ONE KOJI ŽELE ZNATI JOŠ VIŠE) Snimite izgovor samoglasnika A, E, I, O i U te nacrtajte pripadni spektrogram. Možete li iz spektrograma razlikovati kada je izgovoren koji samoglasnik? Koristite li uskopojasni ili širokopojasni spektrogram? Zamolite kolegu da vas da snimi izgovor samoglasnika proizvoljnim redom. Prije slušanja snimke samo iz spektrograma pokušajte odrediti koji je redoslijed samoglasnika u snimci.

## 3.2. Projektiranje FIR filtara

Drugi dio druge laboratorijske vježbe bavi se projektiranjem FIR filtara. Iako postoji veliki broj metoda za projektiranje digitalnih FIR i IIR filtara u ovom dijelu vježbe baviti ćemo se isključivo projektiranjem FIR filtara koji imaju *linearnu fazu*, odnosno *konstantno grupno vrijeme kašnjenja*.

FIR filtri su karakterizirani konačnim impulsnim odzivom, odnosno konačnim brojem koeficijenata koje moramo odabrati tako da se za njihovo projektiranje uglavnom koriste optimizacijske metode. No želimo li projektirati FIR filter s *linearnom fazom* ne možemo slobodno birati sve uzorke impulsnog odziva. Linearnost faze naime podrazumijeva bilo simetriju bilo antisimetriju impulsnog odziva što znači da možemo slobodno birati samo polovicu koeficijenata. U MATLAB-u postoje gotove funkcije za projektiranje optimalnih FIR filtara s *linearnom fazom* kao što su `fir1s` (najmanja kvadratna pogreška) i `remez` ili `firpm` (najmanja minimax pogreška).

### 30 minuta **Zadatak 3.2-1 Projektiranje FIR filtara projekcijskom metodom**

Označimo sa  $A_d(\omega)$  željenu amplitudnu karakteristiku. Želimo li naći FIR filter tipa I (paran red filtra  $N$ , neparan broj uzoraka impulsnog odziva, simetričan odziv) koji najbolje aproksimira željenu karakteristiku u smislu srednje kvadratne pogreške možemo koristiti teorem o projekciji. Amplitudna karakteristika FIR filter tipa I je dana izrazom

$$A(\omega) = \sum_{m=0}^{N/2} a[m] \cos(\omega m),$$

pri čemu je veza koeficijenata  $a[m]$  i uzoraka impulsnog odziva  $h[n]$  dana izrazom

$$h[n] = \begin{cases} \frac{1}{2} a[\frac{N}{2} - n], & 0 \leq n < \frac{N}{2} \\ a[0], & n = \frac{N}{2} \\ \frac{1}{2} a[n - \frac{N}{2}], & \frac{N}{2} < n \leq N \end{cases}.$$

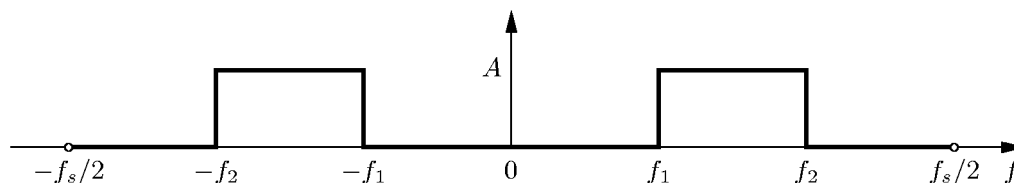
Koeficijenti  $a[m]$  su prema teoremu o projekciji određeni kao

$$a[0] = \frac{1}{2\pi} \int_{-\pi}^{\pi} A_d(\omega) d\omega$$

i

$$a[m] = \frac{1}{\pi} \int_{-\pi}^{\pi} A_d(\omega) \cos(\omega m) d\omega.$$

Korištenjem navedenih izraza možemo projektirati FIR filter tipa I koji je optimalan u smislu srednjeg kvadratnog odstupanja.



Slika 2. Željena karakteristika idealnog pojasno-propusnog filtra

- a) (PRIPREMA) Razmotrite željenu amplitudnu karakteristiku FIR filtra tipa I prikazanu slikom 2. Radi se o idealnom pojasno-propusnom filteru koji propušta frekvencije od  $f_1$  do  $f_2$ . Odredite opće izraze za koeficijente  $a[m]$  i uzorke impulsnog odziva  $h[n]$ .

- b) Korištenjem MATLAB-a provjerite vaš rezultat iz pripreme.

```
» syms f1 f2 w m
» int(cos(w*m),w,-f2,-f1)+int(cos(w*m),w,f1,f2) % rezultat je još
ans = % potrebno skalirati
% i iz njega odrediti
2*(-sin(f1*m)+sin(f2*m))/m % uzorke impulsnog odziva
```

- c) Prema dobivenim izrazima iz pripreme napišite MATLAB funkciju koja za zadane frekvencije  $f_1$  i  $f_2$  te za zadani red filtra  $N$  (koji mora biti paran) vraća vrijednosti impulsnog odziva.

- d) Naredba `freqz` za zadani impulsni odziv crta amplitudnu i faznu frekvencijsku karakteristiku. Provjerite ispravnost napisane funkcije iz c) zadatka za proizvoljno odabrane  $f_1, f_2$  i  $N$ . Što možete reći o aproksimaciji kada se red filtra  $N$  povećava?

- e) (ZA ONE KOJI ŽELE ZNATI VIŠE) Naredba `fir1s` projektira filter koji je optimalan u smislu kvadratnog odstupanja. Naredbi se uz red filtra mora zadati specifikacija amplitudno-frekvencijske karakteristike. Specifikacija se sastoji od dva vektora s parnim brojem uzoraka s time da se u prvom navodi frekvencija, a u drugom željena amplituda na toj frekvenciji. Filter s karakteristikom prikazanom na slici 2. bi projektirali na slijedeći način:

```
» N = 5; % red filtra
» A = 1; % amplituda
» f1 = 0.25; % normirana donja granična frekvencija
» f2 = 0.75; % normirana gornja granična frekvencija
» fir1s(N, [0 f1 f1 f2 f2 1], [0 0 A A 0 0])
ans =
-0.3183 0.0000 0.5000 0.0000 -0.3183
» freqz(ans) % crtamo amplitudnu i faznu karakteristiku
```

Usporedite rezultate vaše funkcije s rezultatima funkcije `fir1s`.

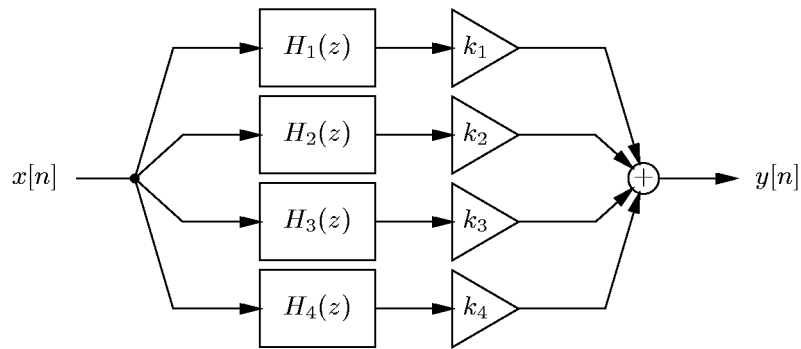
### 30 minuta **Zadatak 3.2-2 Filtarski slog i ekvalizator**

Propuštamo li isti signal paralelno kroz različite filtre govorimo o filtarskom slogu. Jedan mogući filtarski slog prikazan je na slici 3. Odaberemo li prijenosne funkcije filtera u slogu tako da vrijedi

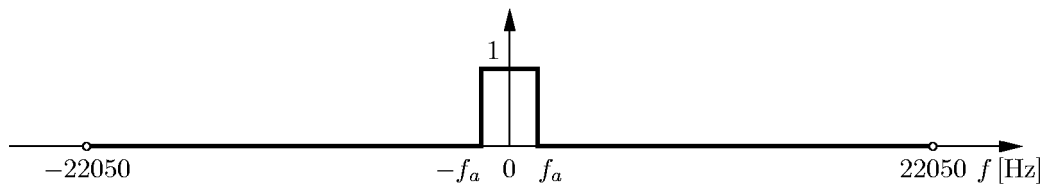
$$H_1(z) + H_2(z) + H_3(z) + H_4(z) = 1$$

uz odabir jediničnih pojačanja ( $k_1 = 1$ ,  $k_2 = 1$ ,  $k_3 = 1$  i  $k_4 = 1$ ) filtarski slog ne mijenja signal. Mijenjanjem pojačanja  $k_i$  možemo utjecati samo na dio signala odabran filtrom  $H_i(z)$ . U ovom zadatku ćemo korištenjem projekcijske metode projektirati četiri FIR filtra koja će nam poslužiti za analizu snimljenog audio signala.

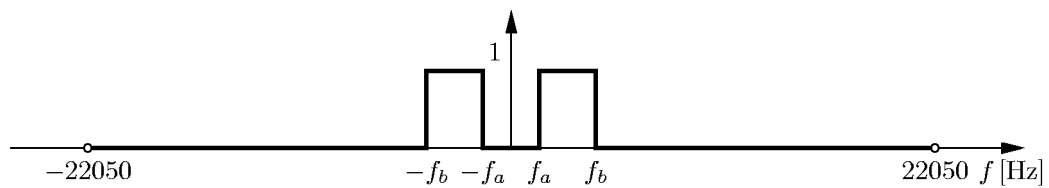
- a) (PRIPREMA) Razmotrite četiri željene prijenosne karakteristike filtra prikazane na slikama 4., 5., 6. i 7. Pokažite da za FIR filtre tipa I projektirane projekcijskom metodom iz zadanih željenih karakteristika bez obzira na odabrani red filtra  $N$  uvijek vrijedi  $H_1(z) + H_2(z) + H_3(z) + H_4(z) = 1$ .



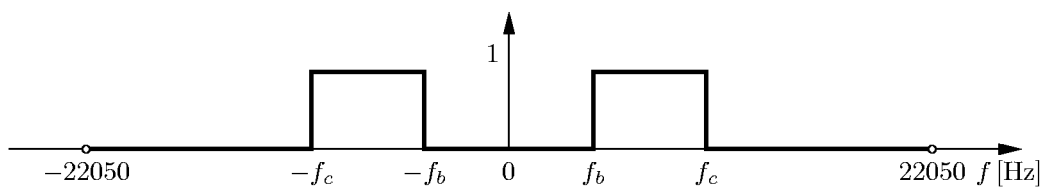
Slika 3. Filtarski slog koji se sastoji od četiri filtra



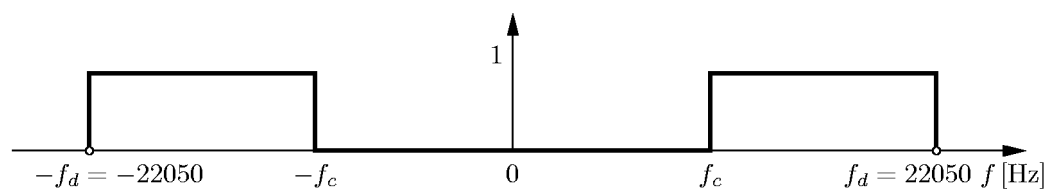
Slika 4. Željena karakteristika prvog filtra u slogu



Slika 5. Željena karakteristika drugog filtra u slogu



Slika 6. Željena karakteristika trećeg filtra u slogu



Slika 7. Željena karakteristika četvrtog filtra u slogu

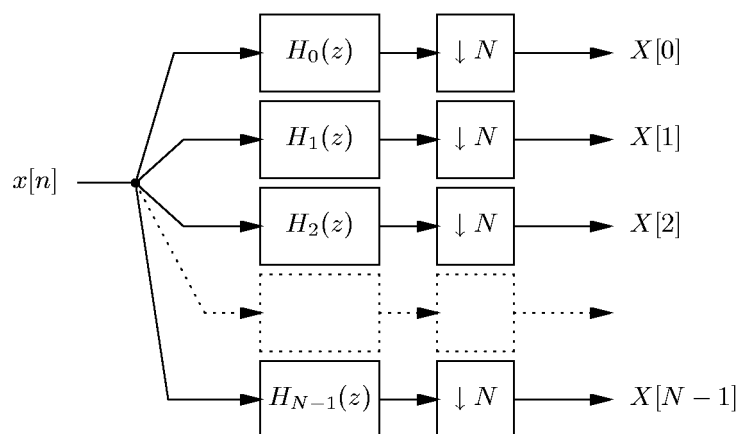


- b) Korištenjem napisanih funkcija iz zadatka 3.2-1c odredite impulsne odzive četiri FIR filtra tipa I koji aproksimiraju željene prijenosne karakteristike prikazane na slikama 4., 5., 6. i 7. Neka je  $N = 32$  i neka je  $f_a = 1470$  Hz,  $f_b = 4410$  Hz,  $f_c = 10290$  Hz i  $f_d = 22050$  Hz.
- c) Korištenjem naredbe `freqz` nacrtajte amplitudne i fazne frekvencijske karakteristike dobivenih filtara te amplitudnu i faznu frekvencijsku karakteristiku njihovog paralelnog spoja.
- d) U Simulinku sastavite model prema slici 3. Kao blok za FIR filter koristite `Overlap-Add FFT Filter` ili `Overlap-Save FFT Filter` (za broj točaka FFT-a odaberite 128; koeficijenti FIR filtra su izračunate vrijednosti impulsnog odziva). Ulaz u filterski slog neka bude izlaz iz `From Audio Device` bloka. Izlaz spojite na `To Audio Device` blok za reprodukciju signala. Svi navedeni blokovi se nalaze u skupini `Signal Processing Blockset`. U `Simulation` izborniku odaberite `Accelerator` način simulacije<sup>3</sup>. Isprobajte razne vrijednosti pojačanja  $k_1$ ,  $k_2$ ,  $k_3$  i  $k_4$ . Koje vrijednosti pojačanja bi odabrali za isticanje niskih frekvencija, a koje za isticanje visokih frekvencija?

Ako vam je simulacija prespora uz `Accelerator` način simulacije ili ako nemate na raspolaganju ubrzanje simulacije blok za snimanje zamijenite s blokom za učitavanje zvučnog signala i blok za reprodukciju zvučnog signala zamijenite s blokom za snimanje signala u datoteku. Nakon završetka obrade snimljeni signal možete poslušati korištenjem bilo kojeg programa za reprodukciju zvuka, a možete koristiti i naredbe `wavread` i `soundsc` unutar MATLAB-a.

### 3.3. Kompresija signala, DCT i MDCT filterski slogovi

Treći dio druge laboratorijske vježbe bavi se DCT i MDCT filterskim slogovima. Kako ćemo razmatrati slogove za veće vrijednosti duljine bloka  $N$  Simulink nije pogodan alat jer je crtanje i povezivanje svih blokova vremenski zahtjevno. Razmotrimo stoga najprije kako bi u MATLAB-u realizirali slog prikazan slikom 8.



Slika 8. Analizirajući filterski slog s decimacijom

<sup>3</sup> `Accelerator` način simulacije temeljem nacrtanog modela generira C kod koji se zatim prevodi. To dovodi do značajnog ubrzanja simulacije, no prilikom svake izmjene modela potrebno je pričekati na prevođenje.

Neka je ulazni signal  $x[n]$  spremljen u varijabli  $x$  i neka su koeficijenti brojnika i nazivnika prijenosne funkcije  $H_0(z)$  spremljeni u vektorima  $A0$  i  $B0$ . Naredbe koje su nam potrebne su `filter` i `downsample`. Naredba `filter` računa odziv LTI sustava na pobudu, dok naredba `downsample` vrši podtipkvanje (ili poduzorkovanje) signala.

```
» x = [1:8] .^ 2; % definiramo ulazni signal
» A0 = [1]; % nazivnik prijenosne funkcije
» B0 = [1 0 -1]; % brojnik prijenosne funkcije
» filter(B0, A0, x) % računamo izlaz filtra

ans =

     1     4     8    12    16    20    24    28

» downsample(ans, 4) % ostavljamo svaki 4 uzorak izlaza

ans =

     1    16
```

Za rekonstrukcijski slog umjesto naredbe `downsample` koristiti ćemo naredbu `upsample` koja vrši pretipkavanje (ili naduzorkovanje) signala.

```
» upsample([1 16], 4) % iza svakog uzorka dodajemo 3 nule

ans =

     1     0     0     0    16     0     0     0
```

Pri tome je važno razlikovati naredbe `upsample` i `downsample` od naredbi `interp` i `decimate`. Naime, pretipkavanje i podtipkavanje s faktorom  $L$  su jednostavne operacije koje ili ostavljaju svaki  $L$ -ti uzorak (podtipkavanje), što možemo opisati izrazom

$$y[n] = x[nL],$$

ili dodaju iza svakog uzorka još  $L - 1$  nula (pretipkavanje), što možemo opisati izrazom

$$y[n] = \begin{cases} x[n/L], & n/L \in \mathbb{Z} \\ 0, & n/L \notin \mathbb{Z} \end{cases}.$$

Interpolacija i decimacija osim te osnovne operacije uvijek pretpostavljaju postojanje filtra koji za je slučaj decimacije konstruiran tako da Nyquistov uvjet bude zadovoljen, dok je za slučaj interpolacije konstruiran tako da računa vrijednosti signala u novo-dodanim uzorcima koje jednostavnim pretipkavanjem postavljamo na nulu<sup>4</sup>.

#### 45 minuta **Zadatak 3.3-1 DCT filtarski slog**

Prvi filtarski slog s decimacijom kojeg ćemo realizirati će biti DCT filtarski slog prikazan na slici 9. koji se temelji na DCT-II transformaciji. DCT-II transformacija je opisana izrazom

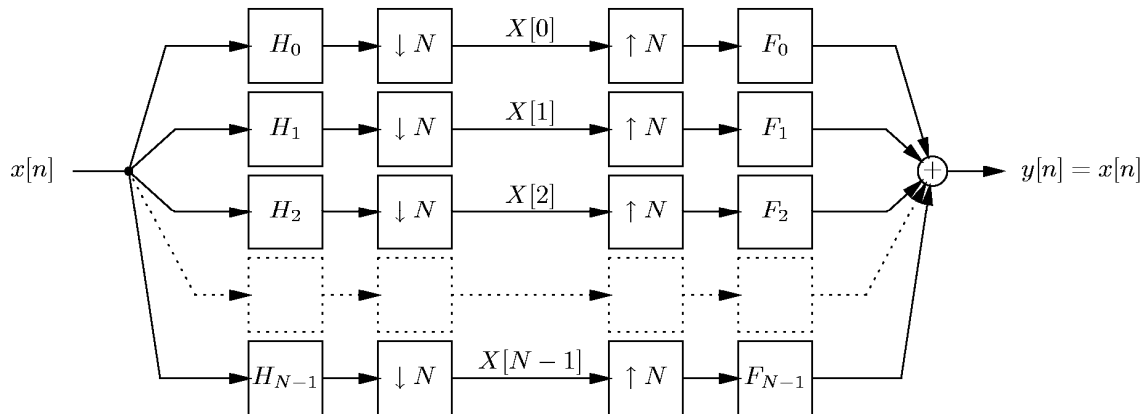
$$X[k] = \sqrt{\frac{2 - \delta[k]}{N}} \sum_{n=0}^{N-1} x[n] \cos \frac{(2n+1)k\pi}{2N},$$

<sup>4</sup> U literaturi se ponekad izrazi miješaju tako da čitatelj uvijek mora biti oprezan te iz konteksta zaključiti radi li se samo o ubacivanju ili izbacivanju uzoraka ili se pak vrši i dodatna filtracija.

dok je inverzna transformacija (koja odgovara DCT-III transformaciji) opisana izrazom

$$x[n] = \sqrt{\frac{2}{N}} \sum_{k=0}^{N-1} X[k] \frac{1}{\sqrt{1+\delta[k]}} \cos \frac{(2n+1)k\pi}{2N},$$

U MATLAB-u naredbe `dct` i `idct` realiziraju navedene transformacije.



Slika 9. DCT filterarski slog s decimacijom

Za realizaciju sloga koristiti ćemo naredbe `filter`, `downsample` i `upsample`. No potrebno je još izračunati impulsne odzive svakog filtra u slogu. Prisjetimo se da koeficijenti FIR filtara u analizirajućem dijelu sloga odgovaraju redcima matrice diskretne kosinusne transformacije, dok koeficijenti rekonstrukcijskih FIR filtara odgovaraju stupcima matrice inverzne diskretne kosinusne transformacije. Pokažimo kako bi u MATLAB-u odredili te koeficijente za  $N = 4$ .

```

» N = 4;
» C = dct(eye(N))
C =
    0.5000    0.5000    0.5000    0.5000
    0.6533    0.2706   -0.2706   -0.6533
    0.5000   -0.5000   -0.5000    0.5000
    0.2706   -0.6533    0.6533   -0.2706

% transformacija je u 4 točke
% matricu transformacije možemo dobiti ako
% za argumet dct funkcije postavimo
% jediničnu matricu

» H1 = C(2, 4:-1:1)
H1 =
   -0.6533   -0.2706    0.2706    0.6533

% impulsni odziv drugog analizirajućeg filtra
% odgovara ZRCALJENOM drugom retku

» CT = idct(eye(4))
CT =
    0.5000    0.6533    0.5000    0.2706
    0.5000    0.2706   -0.5000   -0.6533
    0.5000   -0.2706   -0.5000    0.6533
    0.5000   -0.6533    0.5000   -0.2706

% matricu inverzne transformacije možemo
% dobiti na jednak način
% CT mora biti transponirana matrica C

» F1 = CT(:, 2)
% drugi rekonstrukcijski filter odgovara

```

```

F1 = % drugom stupcu (nema zrcaljenja)
    0.6533
    0.2706
   -0.2706
   -0.6533

```

Pogledajmo sada kako bi izgledao programski odsječak koji računa koeficijente svih analizirajućih i rekonstrukcijskih filtara za proizvoljni  $N$ :

```

1.  N = 32;
2.
3.  % Impulsni odzivi analizirajućih filtara su spremljeni u H{i}
4.  C = dct(eye(N));
5.  for i = 1 : N
6.      H{i} = C(i, N:-1:1);
7.  end
8.
9.  % Impulsni odzivi rekonstrukcijskih filtara su spremljeni u F{i}
10. CT = idct(eye(N));
11. for i = 1 : N
12.     F{i} = CT(:, i);
13. end

```

Ako se ulazni signal nalazi u varijabli  $x$  i ako izlazni signal želimo spremiti u varijablu  $y$  mogući programski odsječak za realizaciju DCT sloga sa savršenom rekonstrukcijom za proizvoljni  $N$  bi bio:

```

1.  % Analizirajući dio DCT sloga.
2.  for i = 1 : N
3.      X{i} = downsample(filter(H{i}, 1, x), N, N-1);
4.  end
5.
6.  % Rekonstrukcijski dio DCT sloga.
7.  y = zeros(numel(X{1})*N, 1);
8.  for i = 1 : N
9.      y = y + filter(F{i}, 1, upsample(X{i}, N));
10. end

```

Ukoliko želimo izvršiti neku manipulaciju s dobivenim koeficijentima rastava  $X\{i\}$  odgovarajući kod bi stavili između linija 4. i 6. Pri razmatranju primjene na kompresiju signala tipično želimo ukloniti one koeficijentima rastava  $X\{i\}$  koji ne nose informaciju.

- a) (PRIPREMA) Prema navedenim odsječcima napišite na računalu<sup>5</sup> m-skriptu (ili funkciju) koja za zadani  $N$  i signal  $x$  računa koeficijente rastava  $X\{i\}$  i rekonstrukciju  $y$ . Funkciju napišite tako da na jednostavan način iz samo nekih komponenti  $X\{i\}$  rastava bude moguće izračunati rekonstrukciju  $y$ .
- b) Korištenjem naredbe `wavrecord` snimite kratki govorni signal. Uz odabir  $N=32$  prvo pokažite da računanjem izlaza  $y$  temeljem svih elemenata rastava  $X\{i\}$  dobivamo savršenu rekonstrukciju signala (do na numeričku pogrešku). Zatim odredite rekonstrukciju  $y$  samo iz prvih 4, 8 i 16 uzoraka. Korištenjem naredbe `soundsc` preslušajte sve dobivene rekonstrukcije. Kako kvaliteta zvuka ovisi o odabiru elemenata rastava  $X\{i\}$  temeljem kojih računamo rekonstrukciju?
- c) (ZA ONE KOJI ŽELE ZNATI VIŠE) Odnos signal-šum definiramo kao omjer snage signala i snage šuma,

<sup>5</sup> Na vježbu nije potrebno nositi ispisanu skriptu ili funkciju - dovoljna je elektronička verzija!

$$SNR = \frac{P_{\text{signal}}}{P_{\text{sum}}} = \frac{\sum x^2[n]}{\sum (y[n] - x[n])^2}.$$

Ako pri računanju rekonstrukcije  $y$  uzimamo samo prvih  $k < N$  komponenti rastava  $X\{i\}$  nacrtajte kako SNR ovisi o  $k$ . Možete li povezati dobivenu krivulju s kvalitetom zvuka?

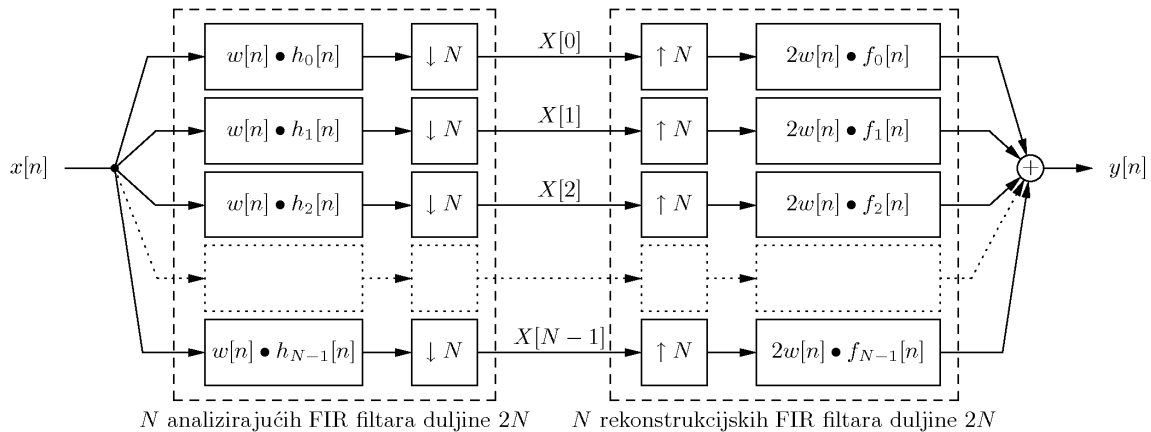
#### 45 minuta **Zadatak 3.3-2 MDCT filtarski slog**

Nešto kompliciraniji od DCT filtarskog sloga jest MDCT filtarski slog koji je prikazan na slici 10. MDCT filtarski slog temelji se na MDCT transformaciji koja je dana izrazom

$$X[k] = \sum_{n=0}^{2N-1} x[n] \cos \frac{(2n+1+N)(2k+1)\pi}{4N},$$

dok je inverzna transformacija dana izrazom

$$y[n] = \frac{1}{N} \sum_{k=0}^{N-1} X[k] \cos \frac{(2n+1+N)(2k+1)\pi}{4N}.$$



Slika 10. MDCT filtarski slog s decimacijom

U MATLAB-u nažalost ne postoje funkcije za implementaciju MDCT i IMDCT transformacije, no zato su za potrebe vježbi napisane funkcije `mdct` i `imdct` koje se ponašaju isto kao i `dct` i `idct` funkcije (možete ih preuzeti sa stranica predmeta). Pogledajmo kako ih iskoristiti za određivanje impulsnih odziva analizirajućih i rekonstrukcijskih FIR filtara:

```

» N = 2;                                % transformacija je u 2 točke
» C = mdct(eye(2*N))                    % duljina bloka je 2N, a matrica
   % transformacije je N sa 2N
C =
    0.3827   -0.3827   -0.9239   -0.9239
   -0.9239    0.9239   -0.3827   -0.3827

» F0 = C(1, end:-1:1)                   % impulсни odzivi analizirajućih filtara su
   % opet jednaki ZRCALJENIM retcima matrice
   % transformacije C
F0 =
   -0.9239   -0.9239   -0.3827    0.3827

```

```

» IC = imdct(eye(N))
IC =
    0.1913    -0.4619
   -0.1913     0.4619
   -0.4619   -0.1913
   -0.4619   -0.1913

» C0 = IC(:,1)
C0 =
    0.1913
   -0.1913
   -0.4619
   -0.4619

```

% matrica inverzne transformacije je  
% dimenzija 2N sa N  
% znate li koliko je C\*IC, a koliko je IC\*C?

% impulsni odzivi rekonstrukcijskih filtara  
% su jednaki stupcima matrice IC

Još jedna značajna razlika u odnosu na DCT filterski slog s decimacijom iz zadatka 3.3-1 jest u primjeni vremenskog otvora  $w[n]$ . Za savršenu rekonstrukciju vremenski otvor  $w[n]$  mora zadovoljiti uvjet

$$w^2[n] + w^2[N+n] = 1.$$

Kako se otvor primjenjuje? Primijetite da ga prema slici 10. primjenjujemo i na filtre analizirajućeg i na filtre rekonstrukcijskog dijela sloga. Pri tome je za analizirajuće filtre potrebno računati  $w[n] \bullet h_i[n]$ , a za rekonstrukcijske filtre  $2w[n] \bullet f_i[n]$ . Oznaka  $\bullet$  označava Schurov (ili Hadamardov) umnožak matrica - radi se o množenju elemenata matrice po članovima (operacija  $.*$  u MATLAB-u). Sada možemo pogledati i kako bi izgledao programski odsječak koji računa koeficijente svih analizirajućih i rekonstrukcijskih filtara za proizvoljni  $N$ :

```

1.  N = 32;
2.
3.  % Prvo računamo vremenski otvor w koji se koristi za Vorbis.
4.  n = 0 : 2*N - 1;
5.  w = sin(pi/2*sin(pi/2/N*(n + 1/2))).^2);
6.
7.  % Impulsni odzivi analizirajućih filtara su spremljeni u H{i}
8.  C = mdct(eye(2*N));
9.  for i = 1 : N
10.     H{i} = w .* C(i,2*N:-1:1);
11. end
12.
13. % Impulsni odzivi rekonstrukcijskih filtara su spremljeni u F{i}
14. IC = imdct(eye(N));
15. for i = 1 : N
16.     F{i} = 2 * w.' .* IC(:,i);
17. end

```

- a) (PRIPREMA) Prema navedenim programskim odsječcima napišite na računalu<sup>6</sup> m-skriptu (ili funkciju) koja za zadani  $N$  i signal  $x$  računa koeficijente rastava  $X\{i\}$  i rekonstrukciju  $y$ . Funkciju napišite tako da na jednostavan način iz samo nekih komponenti  $X\{i\}$  rastava bude moguće izračunati rekonstrukciju  $y$ .
- b) Korištenjem naredbe `wavrecord` snimite kratki govorni signal. Uz odabir  $N=32$  prvo pokažite da računanjem izlaza  $y$  temeljem svih elemenata rastava  $X\{i\}$  dobivamo savršenu rekonstrukciju signala (do na numeričku pogrešku). Zatim odredite

<sup>6</sup> Na vježbu nije potrebno nositi ispisanu skriptu ili funkciju - dovoljna je elektronička verzija!

rekonstrukciju  $y$  samo iz prvih 4, 8 i 16 uzoraka. Korištenjem naredbe `soundsc` preslušajte sve dobivene rekonstrukcije. Kako kvaliteta zvuka ovisi o odabiru elemenata rastava  $X\{i\}$  temeljem kojih računamo rekonstrukciju?

- c) (ZA ONE KOJI ŽELE ZNATI VIŠE) U MDCT filtarskom slogu moguće je koristiti razne vremenske otvore. Ponovite prethodni zadatak za slijedeće vremenske otvore:

1.  $w_1[n] = \frac{1}{\sqrt{2}}$
2.  $w_2[n] = \sin \frac{(2n+1)\pi}{4N}$
3.  $w_3[n] = \sin \left( \frac{\pi}{2} \sin^2 \frac{(2n+1)\pi}{4N} \right)$

- c) (ZA ONE KOJI ŽELE ZNATI VIŠE) Odnos signal-šum definiramo kao omjer snage signala i snage šuma,

$$SNR = \frac{P_{\text{signal}}}{P_{\text{sum}}} = \frac{\sum x^2[n]}{\sum (y[n] - x[n])^2}.$$

Ako pri računanju rekonstrukcije  $y$  uzimamo samo prvih  $k < N$  komponenti rastava  $X\{i\}$  nacrtajte kako SNR ovisi o  $k$ . Možete li povezati dobivenu krivulju s kvalitetom zvuka?

## 3.4. Uvod u digitalnu obradu slike

Zadnji dio ove laboratorijske vježbe bavi se uvodom u korištenje MATLAB-a za digitalnu obradu slike. Teme koje će biti obrađene su kako učitati, prikazati te pohraniti sliku te kako izvršiti najjednostavnije manipulacije sa slikom. Svrha ovog djela vježbe jest priprema za treći ciklus laboratorijskih vježbi.

### 10 minuta **Zadatak 3.4-1 Učitavanje, prikaz i spremanje slike**

Naredba `imread` učitava razne formate slika. Bitno je za naglasiti da se u MATLAB-u svaka slika reprezentira pomoću dvije matrice, matrice vrijednosti  $X$  i palete boja `map`. Pri tome se paleta boja u određenim slučajevima može izostaviti.

```
» [X, map] = imread('cameraman.tif');    % učitavamo sliku
» whos                                    % primijetite da nema palete boja
Name      Size      Bytes    Class
X         256x256    65536    uint8 array
map       0x0        0        double array

Grand total is 65536 elements using 65536 bytes

» image(X)                                % prikazujemo sliku
» colormap(map)                            % vidite li sliku?
» colormap(gray(256))                     % siva paleta boja je dobar izbor
```

- a) Učitajte i prikažite nekoliko slika koje se isporučuju s MATLAB-om: `cameraman.tif`, `canoe.tif`, `football.jpg` i `flowers.tif`. Koje vrijednosti poprima  $X$  (tip i dimenzije matrice), a koje `map` za svaku sliku. Kakvu paletu je potrebno odabrati za slike za koje je `map` prazna matrica? Na temelju čega to možete zaključiti?
- b) Prikažite svaku učitane sliku korištenjem naredbi `image`, `imshow` i `imagesc`. U čemu se naredbe razlikuju?
- c) Korištenjem naredbe `rand` generirajte slučajnu matricu dimenzija 90 sa 90 točaka i snimite je korištenjem naredbe `imwrite`. Učitajte tako snimljenu sliku u novu

varijablu te korištenjem naredbe `imagesc` istodobno prikažite obje slike. U čemu je razlika?

```
» X = random('norm',1,9,90,90); % kreiramo sliku
» imwrite(X, 'proba.tif'); % snimamo je
» Y = imread('proba.tif'); % učitavamo sliku
» figure, imagesc(X)
» figure, imagesc(Y) % jesu li slike iste?
```

#### 10 minuta **Zadatak 3.4-2 Operacije na točki**

Najjednostavnije operacije na slici su bezmemorijske operacije koje primjenjujemo na svaku točku. Kako je u MATLAB-u slika reprezentirana kao matrica operacije nad točkama se jednostavno provode.

- a) U zadatku 3.4-1c ste primijetili da slika nakon snimanja i učitavanja više nije jednaka polaznoj slici. Da bi lakše uočili razlike možemo oduzeti slike i prikazati rezultat. Oduzmite slike i prikažite rezultat! Možete li iz rezultata oduzimanja zaključiti kako se slike razlikuju?

```
» razlika = X - Y; % računamo razliku
??? Error using ==> -
Function '-' is not defined for values of class 'uint8'.

» whos
Name      Size      Bytes  Class
X         90x90      64800  double array
Y         90x90      8100   uint8 array

Grand total is 16200 elements using 72900 bytes

» razlika = X - double(Y); % ovisno o verziji MATLAB-a ponekad je
» imagesc(razlika) % potrebno promijeniti tip podatka
```

- b) Oduzimanje slika se uobičajeno koristi kako bi se istaknule razlike. Učitajte slike `angio0.tif` i `angio1.tif` (dostupne na stranicama predmeta). Na tim slikama je snimka glave prije i nakon ubrizgavanja kontrastnog sredstva. Oduzmite te dvije slike te prikažite rezultat. Što ste dobili?

## 4. Literatura

1. Sanjit K. Mitra, *Digital Signal Processing - A Computer Based Approach*, McGraw-Hill, 1998., <http://www.mhhe.com/engcs/electrical/mitra/>
2. ITU-T Recommendation Q.23 (11/88): *Technical features of push-button telephone sets*, <http://www.itu.int/rec/T-REC-Q.23/en>
3. ITU-T Recommendation Q.24 (11/88): *Multifrequency push-button signal reception*, <http://www.itu.int/rec/T-REC-Q.24/en>
4. G. Goertzel, *An Algorithm for the Evaluation of Finite Trigonometric Series*, The American Mathematical Monthly, sv. 65., br. 1., str. 34-35, siječanj 1958.
5. Stephen A. Martucci, *Symmetric Convolution and the Discrete Sine and Cosine Transforms*, IEEE Transactions on Signal Processing, sv. 42., br. 5., str. 1038-1051, svibanj 1994. <http://dx.doi.org/10.1109/78.295213>



6. Princen, J. i Bradley, A, *Analysis/Synthesis filter bank design based on time domain aliasing cancellation*, IEEE Transactions on Acoustics, Speech and Signal Processing, sv. 34., br. 5., str. 1153-1161, listopad 1986.
7. H. Babić, *Signali i sustavi (zavodska skripta)*, FER, Zagreb 1996.,  
[http://sis.zesoi.fer.hr/predavanja/pdf/sis\\_2001\\_skripta.pdf](http://sis.zesoi.fer.hr/predavanja/pdf/sis_2001_skripta.pdf)
8. Ž. Ban, *Osnove MATLAB-a*, FER, Zagreb, 2006.  
[http://www.fer.hr/\\_download/repository/Osnove\\_matlaba.pdf](http://www.fer.hr/_download/repository/Osnove_matlaba.pdf)
9. *Laboratorij i vještine - MATLAB: Uvod u Simulink*, FER, Zagreb, 2006.  
[http://www.fer.hr/\\_download/repository/simulinkskripta.pdf](http://www.fer.hr/_download/repository/simulinkskripta.pdf)
10. T. Petković, *Kratke upute za korištenje MATLAB-a*, FER, Zagreb, travanj 2005.  
[http://www.fer.hr/\\_download/repository/matlab\\_upute.pdf](http://www.fer.hr/_download/repository/matlab_upute.pdf)