

FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA
ZAVOD ZA ELEKTRONIČKE SUSTAVE I OBRADBU INFORMACIJA

PROJEKT IZ KOLEGIJA
NAPREDNE METODE DIGITALNE OBRADBE SIGNALA

FREKVENCIJSKA KARAKTERISTIKA LJUDSKOG UHA

STUDENTI:

Nina Marević
Branimir Dropuljić

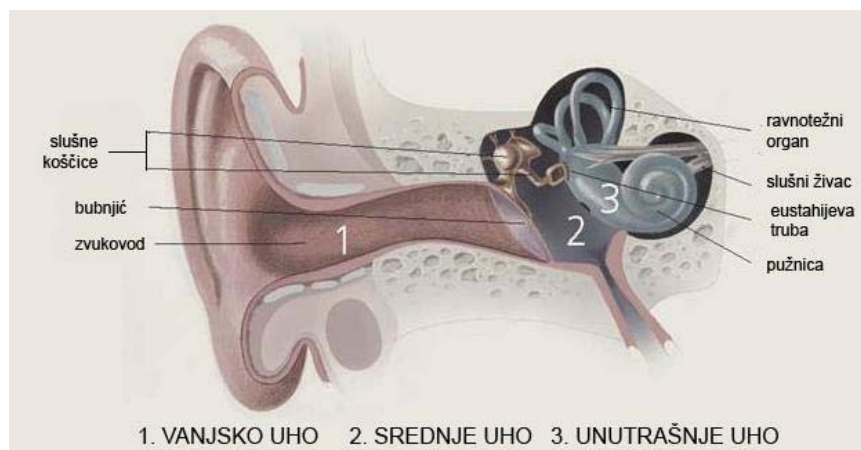
Zagreb, siječanj 2007

- UVOD -

Kako čujemo?

Zvuk iz čovjekove okoline se prenosi kroz vanjsko uho (ušna školjka i zvukovod) do srednjeg uha gdje uzrokuje titranje bubnjića. Titranje se dalje prenosi preko lanca slušnih košćica do unutarnjeg uha (polukružni kanali, pužnica, slušni živac i eustahijeva truba) gdje dolazi do podraživanja osjetnih slušnih stanica. Nakon što su podražene osjetne slušne stanice dolazi do pretvaranja mehaničke energije zvuka u električni impuls koji se provodi putem slušnog živca i slušnog puta do moždane kore gdje nastaje svjesna percepcija zvuka.

Zvuk kojeg ljudsko uho čuje se kreće od 20-tak Hz do 20kHz.



Kako napraviti aproksimaciju ljudskog sluha na računalu?

Sve frekvencije ispod 20 ili iznad 20 000 Hz ne primjećujemo u prirodi i nije ih potrebno propuštati u programskom modelu. Karakteristično je za ljudsko uho da subjektivno ne čuje sve frekvencije jednako.

Ljudsko je uho daleko osjetljivije u nižim i srednjim frekvencijama nego u visokim.

Ova karakteristika ljudskog uha naziva se "**frekventna selektivnost**" i na najnižim je frekvencijama oko 100 Hz, ali na onim višim iznosi i više od 4 kHz.

Zanimljivo je dakle da razliku između 20Hz i 150Hz može osjetiti svatko, a između 15kHz i 17kHz malo tko.

Naime, ljudsko uho prepoznaje pojedine frekvencijske pojaseve kao grupe zvukova, u grubo - niske frekvencije, srednje frekvencije i visoke frekvencije.

Niske frekvencije (bas) se približno nalaze u pojasu od 20Hz do 400Hz, **srednje frekvencije** od 400Hz do 7,000-8,000Hz, a **visoke frekvencije** od 8,000Hz do 20,000Hz.

Ideja je raspon frekvencija koje čovjek čuje pokušati razdijeliti u 30-ak pojaseva preko 30-ak filtara, te promatrati kakav signal izlazi iz filtara, dakle kakve sve frekvencijske komponente signala razaznajemo.

Samo tu frekvencijsku skalu nećemo podijeliti linearno - nego prema mel-skali, jer ona više odgovara prirodnim karakteristikama ljudskog uha.

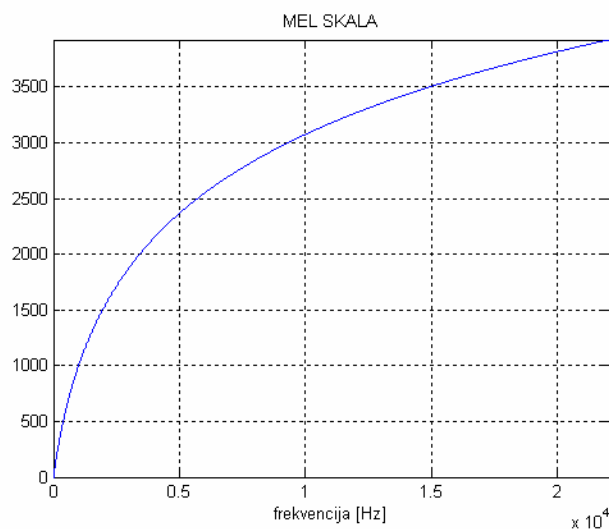
Mi u **mel skali** projektiramo filtarsku banku od 30-ak filtara koji su svi isti i ovisno o koeficijentu preklapanja se preklapaju. Prebačeni u frekvencijsku skalu ti filtri se rasporede baš onako kako nam treba - gušći su na nižim frekvencijama, a rjeđi su na višim frekvencijama.

Veza mel skale s frekvencijskom se definira kao:

$$\text{Mel} = 1127.01048 \log_e(1+f/700)$$

Ova formula pretvara frekvencijsku skalu u mel skalu za svaku frekvenciju kaže koliko je to u melima.

Normirana je tako da joj je 1000 Hz = 1000 mel.



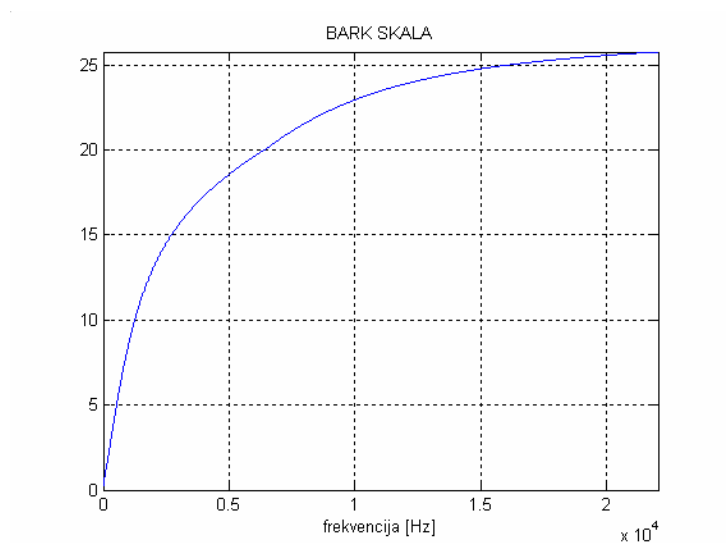
Slično tome, moguće je i projektirati filtre u **bark skali**.

Veza bark skale s frekvencijskom se definira kao:

```
bark = 13arctan(0.00076f)+3.5arctan((f/75000)2)
bark = bark +0.15*(2- bark).*( bark <2)+0.22*( bark -20.1).*
      ( bark >20.1);
```

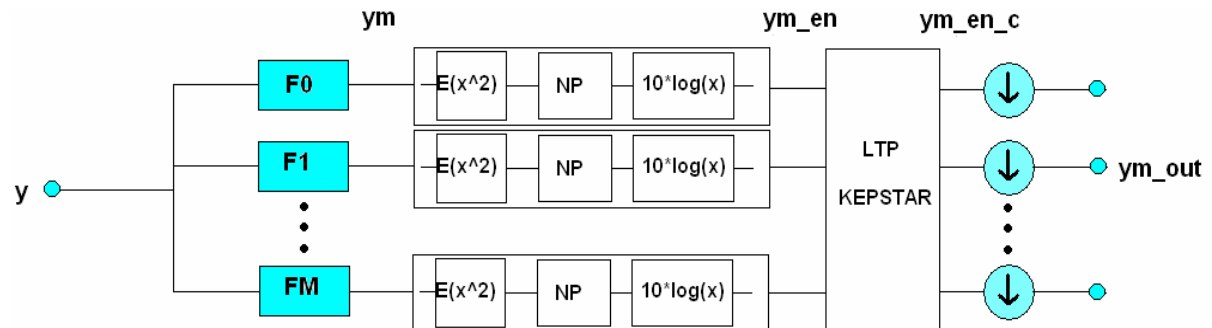
Smisao je isti, al se mel skala češće koristi u praksi.

Zamišljena je tako da se skala rasporedi na rang koji odgovara prvih 24 filtara.



- ŠTO SMO MI NAPRAVILI -

Kako smo osmislili sklop?



Algoritam:

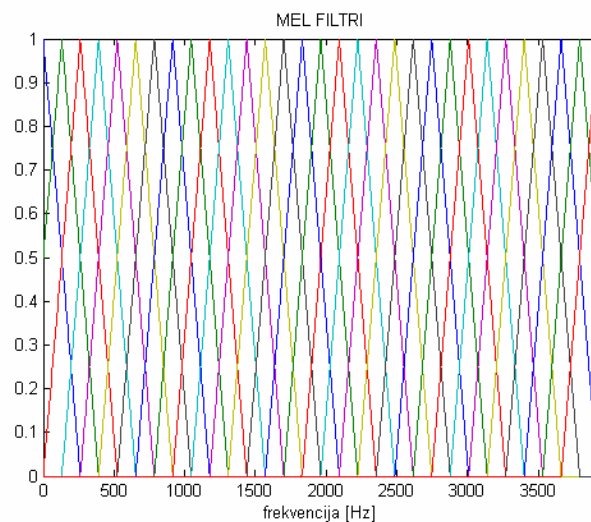
1. projektiranje filtara
2. logatiram anvelope energija svakog kanala
3. primjena brzog kepstra na svaki uzorak zasebno
4. decimiranje signala

Kako smo pisali kod?

Dakle, prvi korak je **projektiranje filtara**.

Mi smo odabrali projektiranje s 31 filtrom, svi filtri su trokutasti, nisu savršeno selektivni nego se preklapaju.

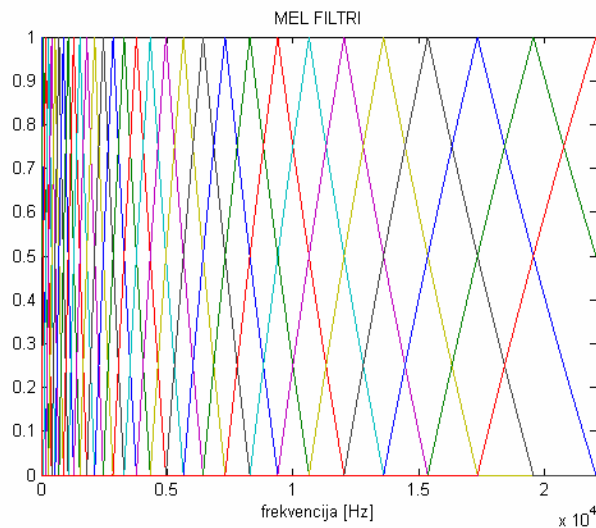
Dakle u mel skali svi su filtri jednaki, kao što možemo primjetiti na slici.



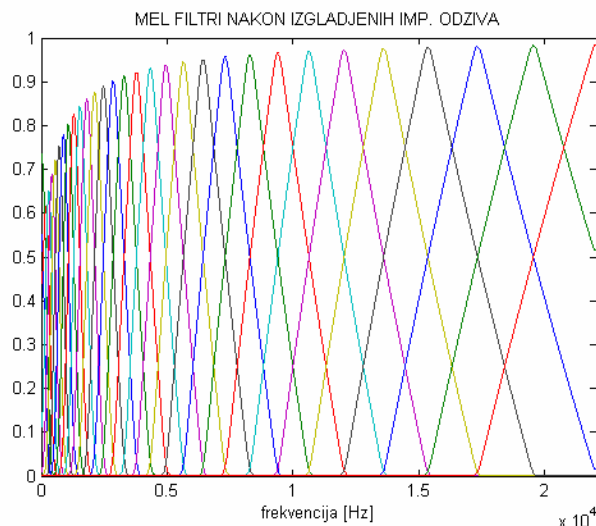
Koliko ih se preklapa ovisi o koeficijentu preklapanja - ako je on 1 to znači da je centar drugog filtra tamo gdje prvi završava. Ako je 2 onda je centar trećeg tamo gdje prvi završava, a centar drugog na pola puta između centra prvog i trećeg. Mi smo odabrali koeficijent preklapanja 2.

Iz slike se vidi da će na svaki uzorak signala u isto vrijeme utjecat 3 filtara.

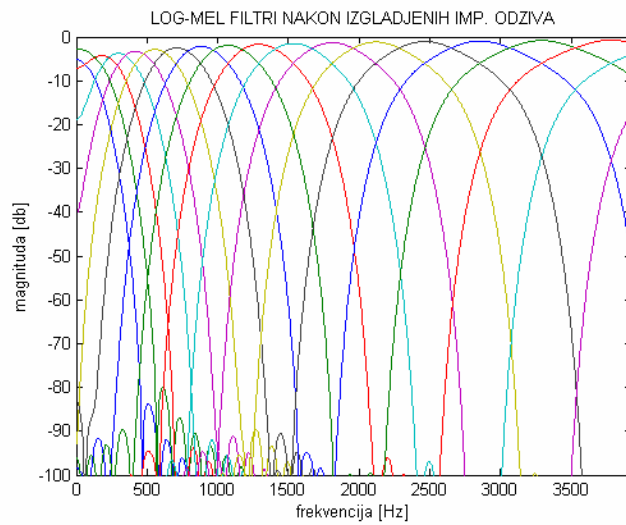
Ako pogledamo ove filtre u frekvencijskoj skali vidimo da su baš takvo kakva je karakteristika ljudskog uha - selektivniji su za niže frekvencije - koje uho dobro razlikuje, a manje selektivni za visoke, koje uho, to jest ljudski sluh ne može razlikovati.



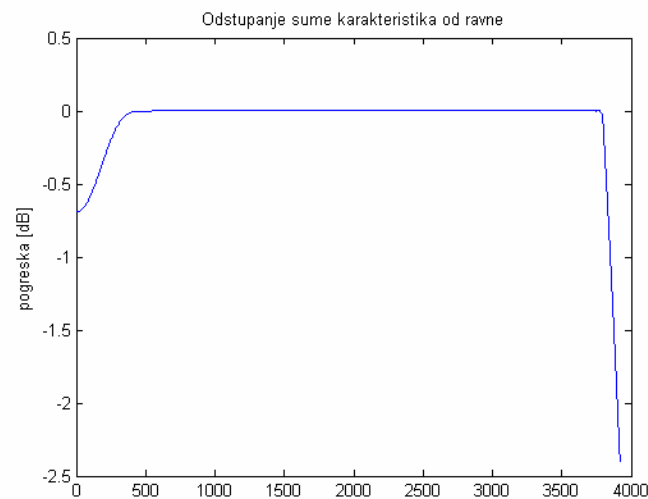
Svaki pojedini filter konvoluiramo s istim blackmannovim otvorom da bi dobili glađi impulsni odziv jednake duljine za svaki filter.



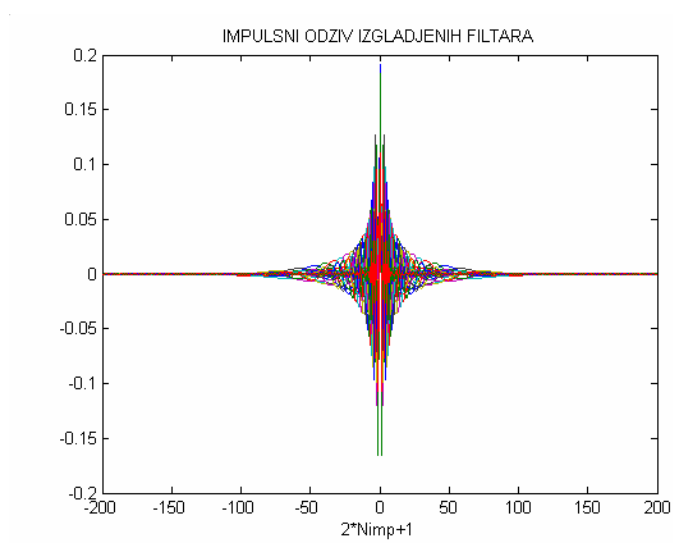
Naposlijetku prikazujemo filtre s y - osi u decibelima:



U idealnom slučaju kad zbrojimo sve filtre po y-osi bi trebali dobiti svepropusni filter iznosa jednakog koeficijentu preklapanja filtara. Pošto smo mi odabrali prvi i zadnji filter da imaju centre na samim rubovima, oni i nekoliko njima najbližih trokutastih filtara s kojima se preklapaju, nemaju potpun oblik trokuta, nego su "prekinuti" - tako da zbroj nemože na tom dijelu biti konstantan. Zato se javljaju propadi na početku i na kraju prikaza odstupanja. Mi smo ciljani zbroj normirali na jedinicu i odstupanje prikazali u logaritamskom mjerilu.



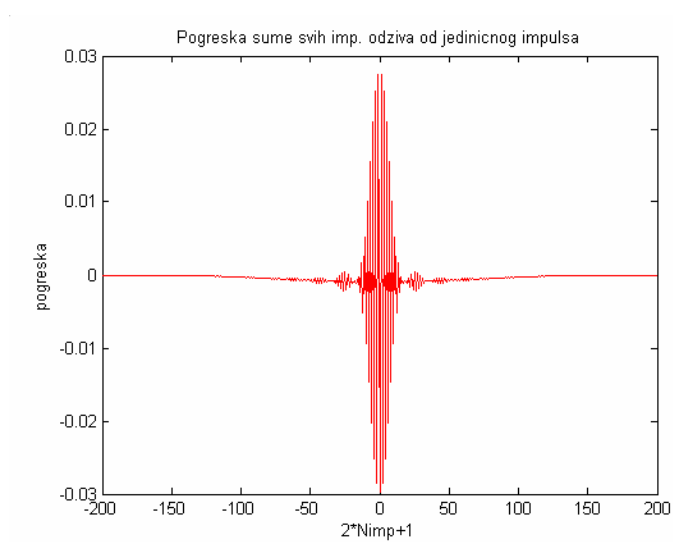
Impulsni odzivi svakog filtra prikazani jedan preko drugog:



Svi impulsni odzivi filtera zbrojeni bi trebali davati jedinični impulsni odziv, odnosno diracha u nuli.

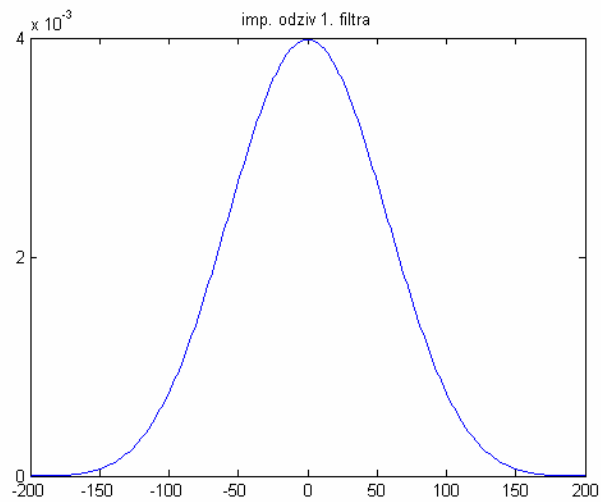
Sljedeća slika prikazuje koliko smo pogriješili, tj. razliku diracha i onoga što smo mi dobili.

Ta razlika je reda 10^{-2} , što je zadovoljavajuće.

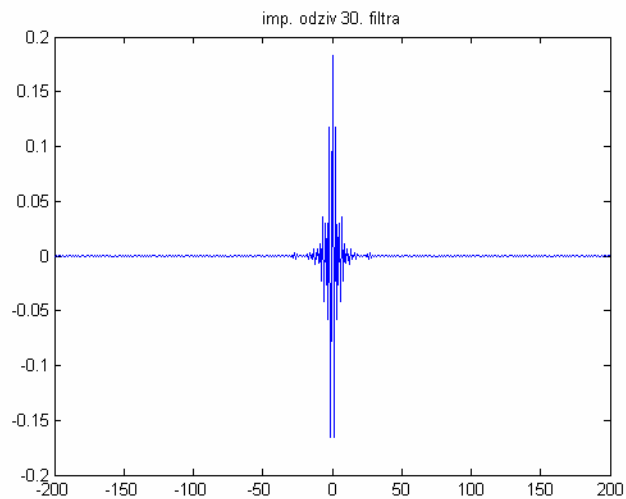


Zbog svoje različitosti u frekvencijskoj skali ne propuštaju svi filtri jednake frekvencije.

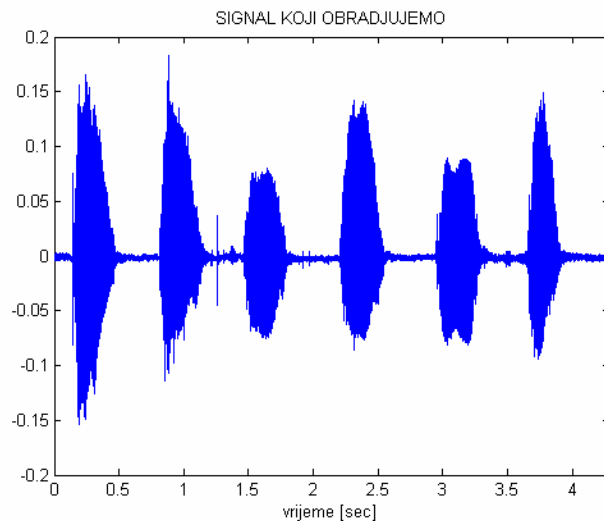
Prvi filter ima najširi i najniži impulsni odziv, te kao takav propušta niže frekvencije.



Zadnji filter ima najuži i najviši impulsni odziv, te kao takav propušta više frekvencije.



Signal koji smo obrađivali je izgovor 6 samoglasnika "a", "e", "i", "o", "u", "z". Frekvencija otipkavanja je 44.1 kHz.



Na izlazu pojedinih filtara imamo signale onakve kakvima ih "čuže" pojedini kanal.

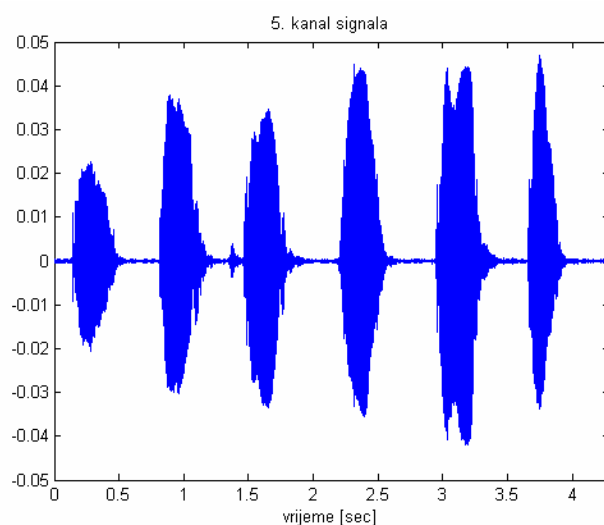
Filtar koji obuhvati jedan od formanta nekog glasa prikaže taj glas s većom energijom od ostalih.

Slušajući izlaze filtra primjećujemo da relativno dobro (glasno) čujemo glas čije formante smo obuhvatili (i to duboki izgovor glasa ako je uhvaćen prvi formant, a piskutavi ako je obuhvaćen treći).

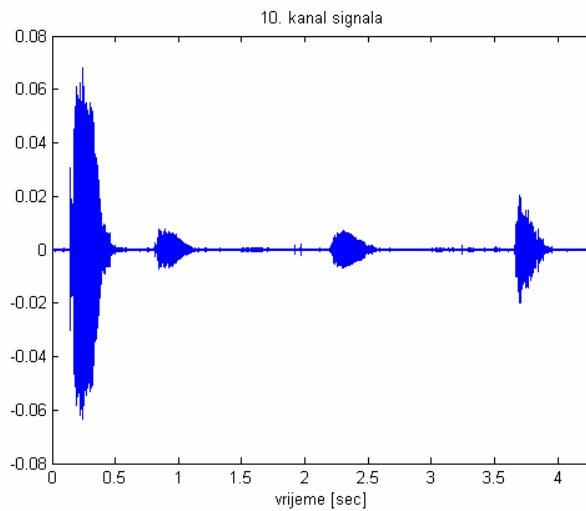
To vrijedi za ovaj konkretni primjer samoglasnika.

Općenito možemo zaključiti da će se na izlazu pojedinog kanala jasno čuti onaj dio zvučnog signala čiji spektar ima nadvišenje u frekvencijskom području koje je obuhvaćeno pojedinim kanalom.

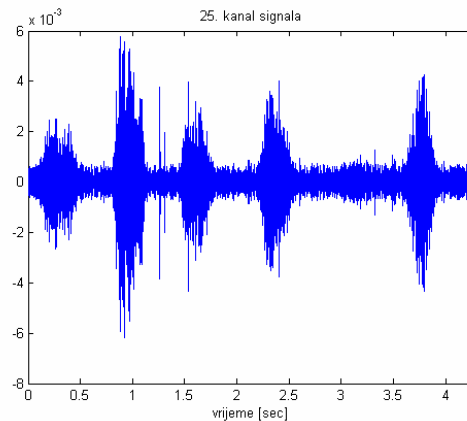
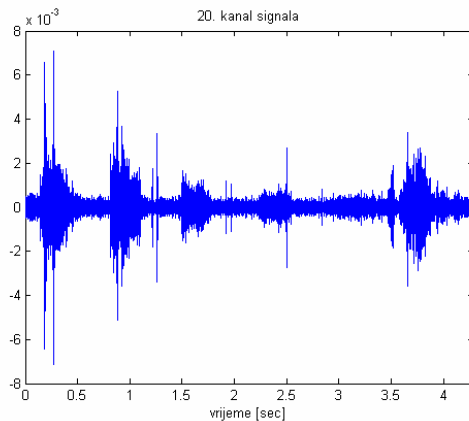
Kod 5. kanala signala vidimo da su gotovo formanti gotovo svih samoglasnika dobro pogođeni, što rezultira relativno rezgovjetnom izgovoru dubokih frekvencija tih samoglasnika.



Kod 10. kanala signala vidimo da ima najveću energiju za glas "a" - tako da pri reprodukciji izlaza iz ovog čujemo dobro glas a na srednjim frekvencijama. Ostali su prigušeni. Mora se napomenuti da su amplitude na izlazu iz pojedinih filtara, čak i kod dobro poklapajućih slučajeva, dosta manje nego što je energija samog glasa, pa je signale potrebno slušati naredbom soundsc - sound scale.

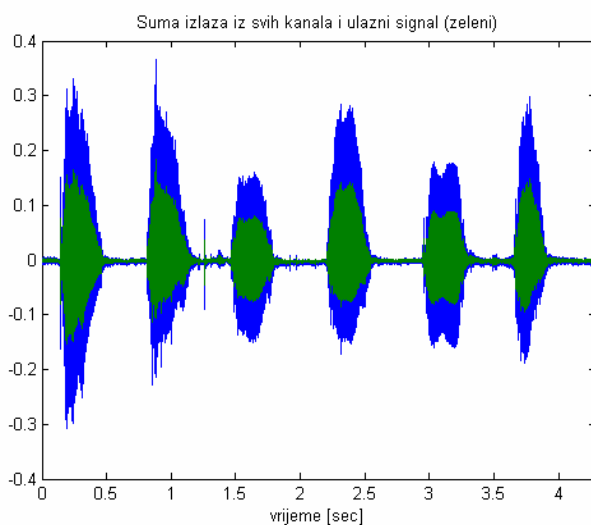


20., te 25. kanal propuštaju samo šum - vidimo da su amplitude signala na izlazu oko $2 \cdot 10^{-3}$



Iz ovoga možemo zaključiti da nam je za neki prosječni ljudski govor (bez ekstremno piskutavih glasova), puno važnije područje nižih frekvencija.

Kada pozbrojimo izlaze svih kanala - dobijemo opet izgovor gotovo identičan ulaznom signalu.



Pojednostavljeno govoreći - možemo ulazni signal zamisliti kao ono što nam neko kaže, a sumu, tj. izlazni signal - kao ono što naše uho čuje.

Na slici primjećujemo da je suma 2 puta veće amplitude - razlog tome je koeficijent preklapanja koji iznosi dva, te za koji suma filtera izgleda kao svepropusni član koji pojačava ulaz dva puta. Da smo uzeli koeficijent preklapanja 1, ulazni signal i suma bi bili gotovo istih amplitude.

Preslušavajući sumu izlaza kanala primjećujemo da je razumljivost zadovoljavajuća.

Dalje smo **uređivali** te iste signale za **ljepši prikaz**, s ciljem da još bolje vizualno predočimo rezultate projekta.

Za prikaz nam je najpogodnija anvelopa od logarima energije.

Na to se još primjene niske komponente kepstra.

Naime, ako pogledamo neki uzorak kako ga vide svi kanali - dobivamo poprilično izlomljenu krivulju.

Izlomljene krivulje se inače glade ograničavanjem spektra - prelomi predstavljaju visoke frekvencijske komponente, a mi ih ograničavanjem spektra uklanjamo.

Pošto naša krivulja predstavlja spektar, ne možemo koristiti baš ovu metodu, već pribjegavamo računanju kepstara.

Ograničavajući se na niske kepstre dobivamo zaglađeni spektar.

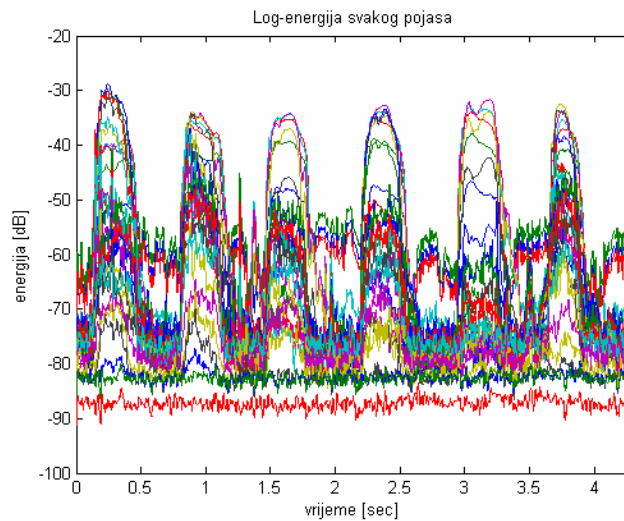
Nakon izlaza iz filtera primjenjuje se metoda za dobivanje anvelope logaritma energije. Signal se prvo kvadrira i propušta kroz jedan filter (uzeli smo FIRGAUSS) koji ima impulsni odziv veličine oko

1000 uzoraka, gladak je, i oblika gaussove krivulje. Treba napomenuti da je glavna latica najšireg impulsnog odziva bila oko 400 uzoraka, znači da ćemo konvolucijom s ovim filtrom dobiti podatke o sporim promjenama signala, tj. anvelopu.

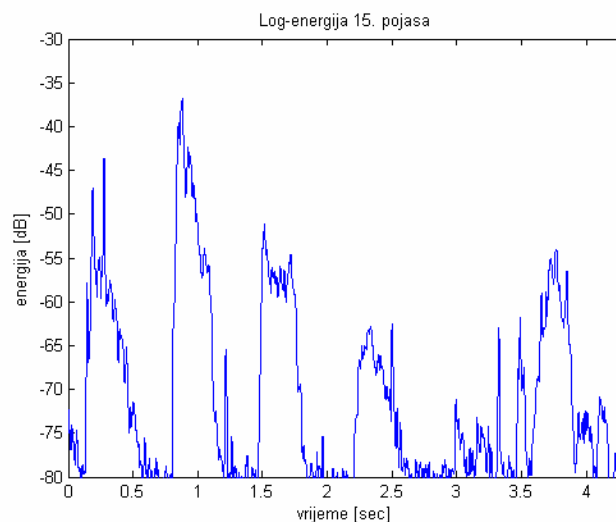
Važan je i glatki izgled gaussove krivulje, koji omogućava još glađu anvelopu. Nakon konvolucije, signal se logaritmirala, i množi s 10 (jer imamo signal na kvadrat), da bi više naglasili djelove nižih iznosa energije, jer se većina promjena energije odvija u tim nižim iznosima.

Ovo se radi za svaki kanal zasebno.

Na izlazu tog sklopa imamo prikaz anvelopa log-energija svakog kanala, koje su prikazane na slici.



Na idućoj slici prikazana je izdvojena energija za 15. pojas. Možemo uočiti da je energija najveća na mjestima gdje se izgovaraju pojedini samoglasnici.



Sljedeća slika prikazuje spektrogram signala.

Spektrogram je prikaz promjene frekvencijskih karakteristika signala u vremenu.

Dakle vidimo izlaze iz svih filtara za svaki pojedini uzorak signala. Visina (boja) je treća dimenzija spektrograma koja prikazuje koliko je jako izražena pojedina frekvencija za pojedini uzorak.

U našem primjeru frekvencija je predstavljena izlazima iz 31-og kanala. Crvena područja su područja najveće energije dok su plavo područja najniže energije.

Kao što smo rekli u primjeru imamo izgovor 6 samoglasnika "a" "e" "i" "o" "u" "э".

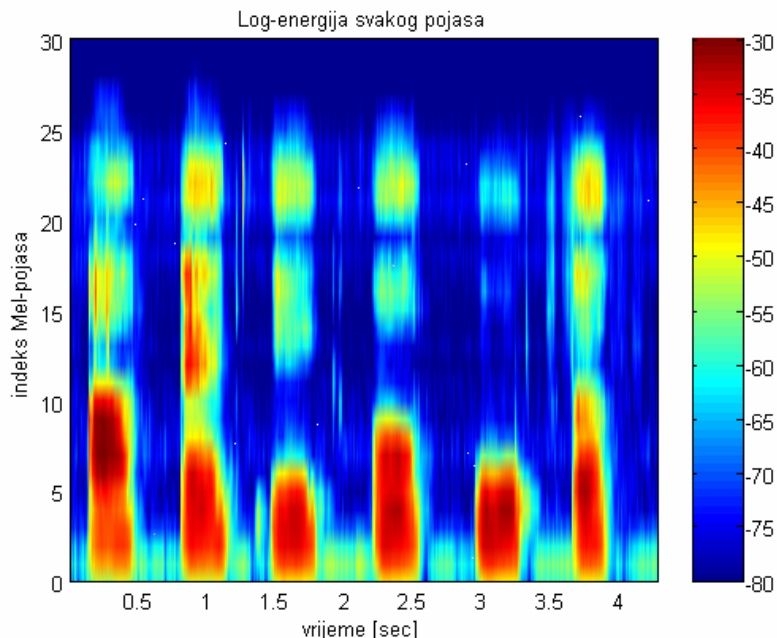
Trajanje izgovora je u "šarenijem" području, gdje je energija veća, dok plava područja između predstavljaju pauzu.

Na slici su jasno vidljiva područja formanata (nadvišenja u spektru, koja su posebno izražena kod samoglasnika).

U tim pauzama se može primjetiti da "niži" filtri propuštaju smetnju gradske mreže.

Pošto su filtri pravljjeni da budu jednaki u mel domeni i automatski da je u frekvencijskoj najviše filtara na najnižim frekvencijama - tako su niže frekvencije bolje pokrivena - što znači da će nam za raspon od 20kHz prvih 10 filtara pokrivati ugrubo rečeno raspon od 3 kHz, idućih 10 raspon od 6kHz, a zadnjih 10 od 11kHz.

To smo i htjeli dobiti jer se takav raspored selektivnosti poklapa s specifičnostima ljudskog sluha.

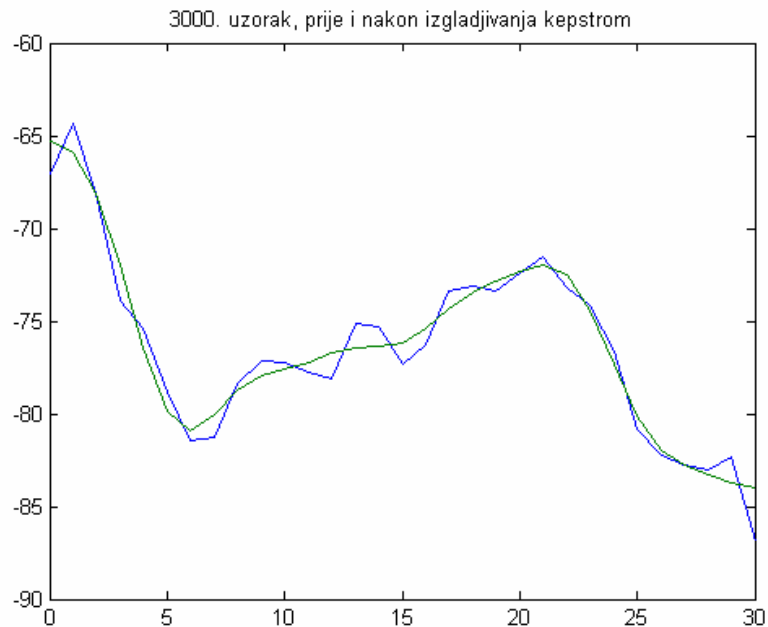


Kao što smo prije spomenuli - promatramo jedan uzorak kakvim ga vide svaki od 31-og kanala. Ta krivulja je izlomljena, što nam ne odgovara - jer uzrokuje grube prelaze u spektrogramu u smjeru y-osi.

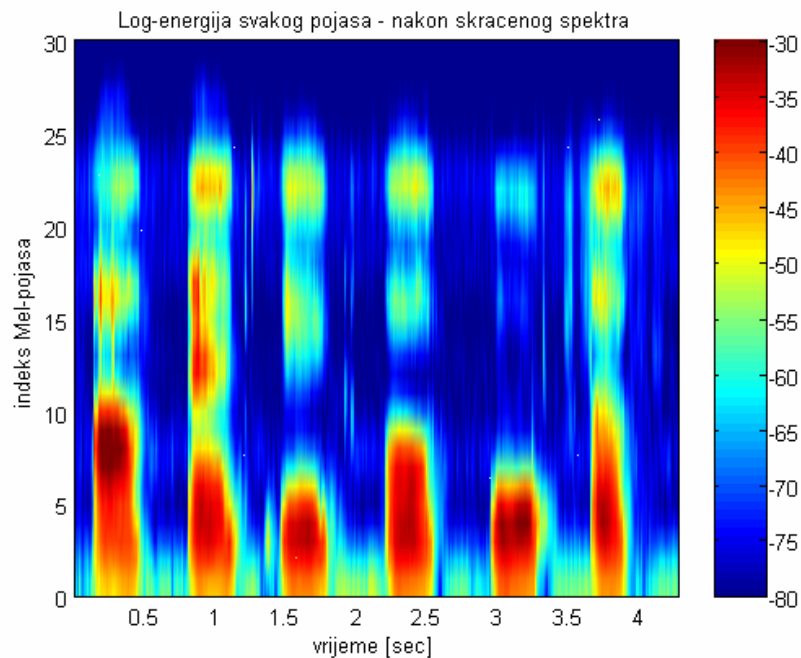
Treći korak je **izgladivanje spektra kratkim kepstrom**.

Pošto nam uzorak preko svih kanala predstavlja spektar, gladenje ćemo postići ograničavajući njegove kepstre na najnižih 12, od prvog do trinaestog.

Ograničavajući se na niske kepstre dobivamo zaglađeni spektar.

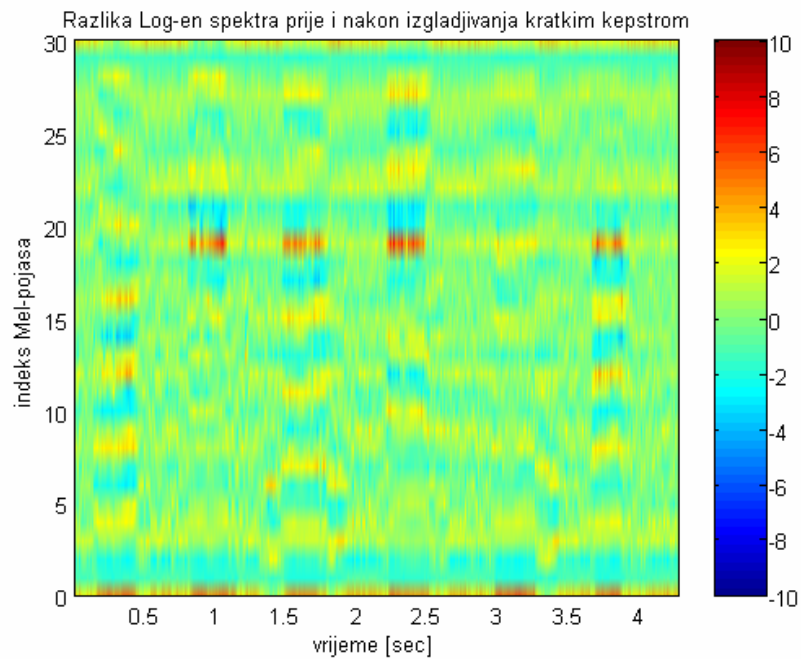


Na slici vidimo da nakon gladenja kepstrom imamo blaže prijelaze po y-osi

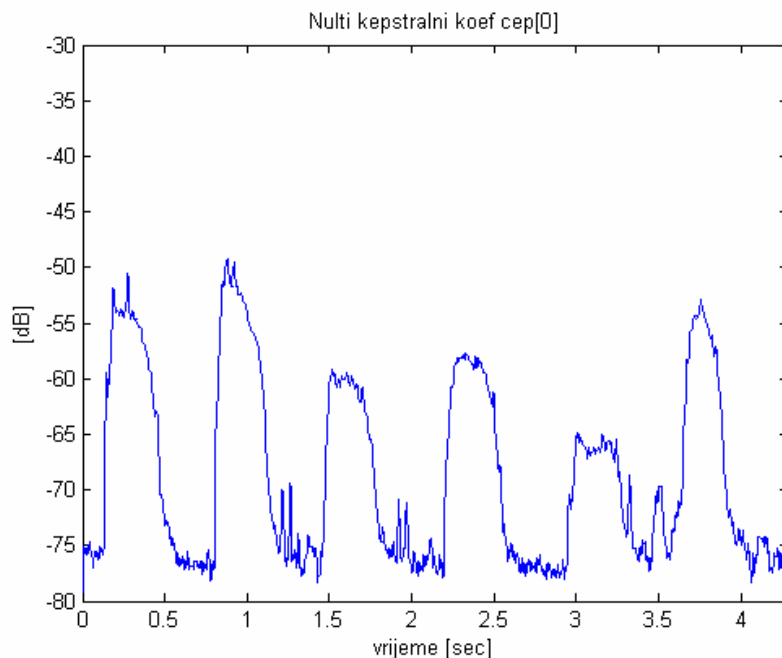


Iduća slika prikazuje razliku log-en-spektara prije i nakon zaglađivanja - možemo uočiti da tamo gdje su bile najveće promjene, skokovi, na razlici slika su maksimumi.

To znači da je zaglađivanje bilo uspješno.



Nulti kepstar predstavlja DC-komponentu spektra, odnosno srednju vrijednost (energiju uzorka signala). Znači prikazujući taj nulti kepstar svakog uzorka možemo lako dobiti podatak o energiji promatranog signala. Iz slike se može vidjeti da su energije u trenutcima izgovora znatno veće od energija u trenutcima tišine, stoga nam nulti kepstar može dobro poslužiti za detekciju govora.



Na kraju se izvrši **decimiranje signala**.

Vrši se decimacija izlaznih signala po vremenu, i to tako da se uzme takav faktor decimacije da na kraju nemamo više od 10000 uzoraka.

- ZAKLJUČAK -

Uspjeli smo prikazati specifičnosti ljudskog uha pomoću programskog koda. U ovom zaključku bismo samo htjeli naglasiti da se u nekoj praktičnoj primjeni ne bi koristili ovdje navedeni koraci projektiranja, jer su oni školski i više namjenjeni razumjevanju funkcioniranja ljudskog sluha, nego što imaju neku komercijalnu iskoristivost.

Ako bi se eventualno radio takav projekt, ulazni signal bi se množio s kratkim vremenskim otvorom dužine oko 25 ms, i od tako ograničenog signala bi se računao DFT u većem broju točaka da se dobije više uzoraka spektra.

Pošto je taj otvor bio simetričan i spektar će biti simetričan. Nakon toga pravi se matrica koja u svakom redu ima jedan od dosad spomenutih filtara, dužine $N/2$ zbog simetrije. Njihova apsolutna vrijednost na kvadrat pomnožena s kvadriranim vektorom koji predstavlja spektar daje jedan vektor energije.

Vremenski otvor se pomiče za 10 ms i ponavlja se računanje, sve dok ne dođemo do kraja signala.

Rezultat dobiven tim postupkom je isti kao rezultat našeg projekta, ali nije razdvojen po kanalima već ima prikazan kompletni raspon frekvencija. Taj postupak se naziva računanje MFCC (Mel Frequency Cepstral Coefficient).

- DODATAK: KOD -

```
%% ----- PROJEKT: FREKVENCIJSKA KARAKTERISTIKA Ljudskog UHA -----%%
%% -- CLANOVI TIMA: BRANIMIR DROPULJIC, NINA MAREVIC ----- %%

%% --- IDEJA: Ovaj program koristi se za realniju interpretaciju zvukova iz
%% okoline. Ljudsko uho ne cuje sve frekvencije jednako. Od raspona cujnih
%% frekvencija 20 - 20000 Hz, nize frekvencije razlikuje puno bolje od
%% visih. Frekvencije iznad 15 kHz gotovo i ne razlikuje, a jedva ih i
%% cuje. Racunalo nemoze interpretirati zvukove na nacin na koji to radi uho,
%% nego sve frekvencije, do fs/2 prikaze u jednakom rasponu. Za digitalnu obradu
%% govora vazno je da se zvuk prikaze na nacin na koji ga mi cujemo, da bi dobili
%% stvarnije i tocnije rezultate. Zato je potrebno zvukove koje promatramo
%% predobraditi tj. prikazati ih onakvima, kakve ih uho stvarno i doživljava.

%% Glavna funkcija koja se koristi u programu, je funkcija 'interpretacija_zvukova'.
%% Ona pokrece podfunkcije, i od ulaznog zvucnog signala radi realni
%% signal. Koristi podfunkciju 'filtri', koja napravi pojasne filtre koji su
%% odredjene sirine i odredjenog preklapanja (nisu idealno selektivni), te
%% su u odredjenoj skali ekvidistantni. Najcesce koristene skale su MEL i
%% BARK, koje postizu nelinearnu frekvencijsku os. Najnize frekvencije
%% imaju najveću razlucivost, a najviše najmanju. Signal koji prolazi kroz
%% taj slog filtara predstavlja podatke o zvuku koji se dalje salje prema
%% mozgu pomocu neurona. Za prikaz signala na kompjuteru koristili smo
%% dvije mogu?nosti, slusanje signala, za koje je dovoljno ostaviti signale
%% takve kakvi jesu, i slikovno prikaz, za koje ih je potrebno obraditi.
%% Obrade se vrse podfunkcijama, 'log_anelopa_en' i 'skraceni_kepstar'. Nakon
%% tog prikaza, signali se decimiraju i predstavljaju izlaze iz sklopa,
%% pogodne za daljnju obradu.

%% Na pocetku ove uvodne skripte - 'projekt' - koja dalje poziva glavnu
%% funkciju 'interpretacija_zvukova', odabire se zvucni signal koji se zeli
%% obraditi, oblik filtara, te skalu. Takodjer se odabire da li se zeli
%% prikazati slike filtara, te prikazati signali (zvukom i slikama) na
%% ulazu i izlazima iz kanala, i da li zelimo pohraniti izlaze iz kanala.
%% Ona nakon, sto pozove funkciju 'interpretacija_zvukova' i napravi sve sto
%% treba, prikazuje slike i zvukove te ih sprema ovisno o odabranom nacinu.

close all;
clear all;

% Odabir zvuka iz foldera 'zvukovi\'. 'zvuk_po_kanalima\' je folder u koji
% ce se kasnije spremati izlazi na kanalima.

pref='nina_sam';
zvukovi='zvukovi\';
zvuk_po_kanalima='zvuk_po_kanalima\';
zvuk=sprintf('%s%s',zvukovi,pref);

[y,FS,NBITS]=wavread(zvuk);
y=y(:);

% Odabir parametara:
% M - broj pojasnih filtara [0:M] (uzima se oko 25 - 30)
% - NAPOMENA!! - mora biti paran!
% BW_f - koeficijent preklapanja filtara
% Nimp - duzina imp odziva
% CL - broj kepstralnih uzoraka kojih uzimamo
% od_skala - odabrana skala

M=30;
BW_f=2;
```

```

Nimp=200;
CL=12;
od_skala=1;      % 1==MEL ,0==BARK
prikaz_slika_filtara=1;      %0==NE, 1==DA
prikaz_signala=1;      %0==NE, 1==DA
zapis_zvuka_po_kanalima=1;  %0==NE, 1==DA

% Poziv glavne funkcije, na ulaz joj dovodimo parametre filtera i ulazni
% signal. Na izlazu daje realizirane signale, spektre i impulsne odzive.

[skala,s_max,f,kor_dec,SP,SP_W,imp,imp_tr,imp_trw,ym,ym_en,ym_en_c,cep,ym_out,raz]=interpretacija_zvukova(y,FS,M,BW_f,Nimp,CL,od_skala);

% skala - odabrana skala
% s_max - njen maksimum
% f - frekvencijska skala
% kor_dec - odabrani korak decimacije
% SP - spektar odabrane skale
% SP_W - spektar odabrane skale nakon izgladjivanja impulsnih odziva
% imp - impulsni odziv
% imp_tr - odrezani impulsni odziv na 2*Nimp+1 uzoraka
% imp_trw - izgladjeni i skraceni impulsni odziv
% ym - matrica, signali na izlazu iz kanala
% ym_en - energija signala, nakon log_anvelopa_en funkcije
% ym_en_c - signali nakon primjene kratkog kepstra
% ym_out - decimirani ym_en_c - signal za dalje koristenje
% raz - razlika ym_en i ym_en_c

```

NAPOMENA: DALJE SLIJEDI DIO KODA KOJI SAMO PLOTA SVE ZANIMLJIVE SIGNALA I SPREMA ZVUKOVE PO KANALIMA PA ĆE BITI STAVLJEN KA KRAJ.

```

%% ----- FUNKCIJA: INTERPRETACIJA_ZVUKOVA ----- %%

function
[skala,s_max,f,kor_dec,SP,SP_W,imp,imp_tr,imp_trw,ym,ym_en,ym_en_c,cep,ym_out,raz]=interpretac
ija_zvukova(y,FS,M,BW_f,Nimp,CL,od_skala);

% GLAVNA FUNKCIJA PROGRAMA
% NACIN RADA:

% Na ulaz se dovodi zvuk kojeg ona prikazuje u odabranoj skali, koja
% aproksimira uho. Dovode se i podatci o filtrima - opisani u skripti
% 'projekt'.

% Za zadani ulazni zvucni signal funkcija radi pojasne selektivne filtre
% koji su ekvidistantni u odabranoj (MEL ili BARK) skali, i ovise o ulaznim
% parametrima, gore navedenim, ali razvuceni u frekvencijskoj skali. Kroz
% te filtre propusta ulazni signal, nakon sto im je skratila impulsne
% odzive sve na jednaku duljinu, i izgadila ih konvolucijom sa
% blackmannovim otvorom. Za izradu filtera koristi podfunkciju 'filtri'.
% Nakon konvolucije dobivamo ulazni signal rastavljen po kanalima koji
% predstavljaju zivcana vlakna koja putuju prema mozgu. Za slikovni prikaz,
% te daljnju obradu signala zanimljivije nam je signale dodatno
% "uljep sati". Stoga se redom primjenjuje estimacija prosjecne log-energije
% svakog kanala. Za to se koristi Firgaussov otvor za
% usrednjavanje zeljene sirine u vremenu. On je duzine oko 1000 uzoraka,
% sto nam je dovoljno da bi dobili najmanju frekvenciju u signalu (anvelopu).
% Logaritmiranje se provodi tek nakon usrednjavanja energije. Sve to radi
% funkcija 'log_anvelopa_en'.
% Na kraju se primjenom funkcije 'skraceni_kepstar' napravi IDFT energije
% svakog uzorka i time se dobiva kepstar. Uzimanjem prvih CL uzoraka kepstra i
% vracanjem DFT-om, dobivamo izgadjenu promjenu energije po kanalima. Takav
% glatki signal, decimiran korakom kor_dec, pogodan nam je za daljnju odradu te
% prikaz. Izracuna se i razlika prije i nakon primjene kepstara.

% Na pocetku, funkcija odabire broj uzoraka za prikaz u ovisnosti o
% frekvenciji otipkavanja.

if FS<=1e4;
    K=2^12;
elseif FS<=3e4;
    K=2^13;
else
    K=2^14;
end;

% Isto tako odabire decimaciju na temelju duzine ulaznog signala, tako da
% nebude na kraju, nakon decimacije, vise od 10000 uzoraka.

if length(y)>=1e7;
    kor_dec=1e4;
elseif length(y)>=1e6;
    kor_dec=1e3;
elseif length(y)>=1e5;
    kor_dec=1e2;
elseif length(y)>=1e4;
    kor_dec=1e1;
end;

% Korak i frekvencijska os.

k=[0:K/2];

```

```

f=k/K*FS;

% U ovisnosti o odabiru racuna se mel ili bark skala. Te maksimalni iznos.
% Bark skala ima tri matematicke formule od kojih je svejedno koja se
% odabire. Takodjer ima i ogranicenja koja su uvrstena u formulu.

if od_skala;
    skala = log(1+f/700)*1127.01048;
    s_max = log(1+(FS/2)/700)*1127.01048;
else
    skala=13*atan(0.00076*f)+3.5*atan((f/7500).^2);
    %skala=26.81*f./(1960+f)-0.53;
    %skala=26.28-52547.6*(1960+f).^(-1);
    skala=skala+0.15*(2-skala).*(skala<2)+0.22*(skala-20.1).*(skala>20.1);
    s_max=skala(1,K/2+1);
end;

% Poziva se podfunkcija 'filtri', koja projektira filtre.

[SP,SP_W,imp,imp_tr,imp_trw]=filtri(M,s_max,BW_f,skala,K,Nimp);

% Konvolucija ulaznog signala sa svim filtrima.

for i=1:M+1,
ym(:,i)=conv(y,imp_trw(:,i));
end;
ym=ym(Nimp+1:length(ym)-Nimp,:);

% Racunanje logaritma anvelope energije po svakom kanalu.

[ym_en]=log_anvelopa_en(M,ym);

% Primjena kratkog kepstra na svaki uzorak.

[cep,ym_en_c]=skraceni_kepstar(CL,M,ym_en);

% Racunanje razlike signala prije i poslije izgadjivanja kepstrima. Ovo se
% radi zbog kasnije usporedbe.

raz=ym_en_c-ym_en;

% Decimacija

ym_out=ym_en_c(1:kor_dec:end,:);

```

```
%% ----- FUNKCIJA: FILTRI ----- %%
```

```
function [SP,SP_W,imp,imp_tr,imp_trw]=filtri(M,s_max,BW_f,skala,K,Nimp);
```

```
% Funkcija koja na temelju odredjenih parametara projektira filtre, i  
% njihove impulsne odzive.  
% Projektirani filtri su ekvidistantni u odabranoj skali i proizvoljnih  
% sirina.
```

```
% Prvo se postave centri filtara u jednolikom rasponu do maksimalne  
% frekvencije skale.
```

```
cen=[0:M]'/M*s_max;
```

```
% BW_nom se namjestja jednak za sve.  
% Ako smo odabrali BW_f=1, ondasu filtri takvi da prvi završava tamo  
% di je iduci u maksimumu. Time se postize da je njihov zbroj jednak jedan.  
% Ako je pak BW_f=2 onda prvi završava di je treci u maksimumu, sada je zbroj jednak  
% dva. Dakle, namjestja se tako da zbroj bude jednak BW_f.
```

```
BW_nom=s_max/M;
```

```
BW=BW_nom*BW_f;
```

```
skala_m=ones(M+1,1)*skala;
```

```
cen_m=cen*ones(1,K/2+1);
```

```
% Projektiranje filtara
```

```
SP=1-abs((skala_m-cen_m)/BW);
```

```
SP=SP.*(SP>0);
```

```
SP=[SP fliplr(SP(:,2:K/2))];
```

```
% Impulsni odziv simetricnog spektra ima fazu nula ali zbog numericke  
% pogreske matlaba rucno to namjestamo.
```

```
imp=real(iff(SP));
```

```
SP=SP(1:K/2+1,:);
```

```
% Impulsni odzivi se sad skracuju svi na duzinu 2*Nimp+1 i izgadjuju  
% blackmannovim filtrom iste duzine. Na kraju imamo impulsni odziv imp_trw  
% i pripadni spektar SP_W.
```

```
imp=[imp(K/2+1:K,:); imp(1:K/2,:)];
```

```
imp_tr=imp(K/2+1+[-Nimp:Nimp],:);
```

```
wi=blackman(2*Nimp+1);
```

```
wi=wi(:)*ones(1,M+1);
```

```
imp_trw=imp_tr.*wi;
```

```
SP_W=fft(imp_trw,K);
```

```
SP_W=SP_W(1:K/2+1,:);
```

```

%% ----- FUNKCIJA: LOG_ANVELOPA_EN ----- %%

function [ym_en]=log_anvelopa_en(M,ym);

% Funkcija koja služi za ljepsi prikaz izlaznih signala.
% Radi se o estimaciji energije ( $E(x^2)$ ) svakog kanala, i konvolucija niskopropusnim
% filtrom (firgauss), koji ima impulsni odziv dužine nekoliko puta veće
% nego što je najšira latica impulsnih odziva ulaznih filtara. Time
% dobivamo anvelopu signala. Na kraju se još signal logaritmirala.

b=firgauss(4,256); %K=4,N=256, length(b)=K*(N-1)+1
b=b/sum(b);

for i=1:M+1,
    ym_en(:,i)=conv(b,ym(:,i).^2);
end;

% Nakon konvolucije signal koji dobijemo je dužine ulaznog signala + reda
% firgauss filtra, pa ga smanjimo na početku i na kraju za pola reda filtra.

ym_en=ym_en((length(b)-1)/2+1:length(ym_en)-(length(b)-1)/2,:);

% Logaritmiranje i množenje s 10 (jer je  $x^2$ ).

ym_en=10*log10(ym_en);

```

```
%% ----- FUNKCIJA: SKRACENI KEPSTAR ----- %%
```

```
function [cep,ym_en_c]=skraceni_kepstar(CL,M,ym_en);
```

```
% Log-spektar u odabranoj skali se primjenom IDFT-a  
% vraca u vremensku domenu cime se dobiva kepstar. Ova operacija  
% se provodi matricnim mnozenjem i to tako da se odmah racunaju  
% samo oni kepstralni uzorci koji se ne odbacuju. Time se  
% dobiva kepstralna matrica, koja se zatim primjenom DFT matrice  
% nazad vraca u Mel (Bark) log-spektralnu domenu. Pri tome, ova  
% kombinacija IDFT-a i DFT-a zapravo simulira efekt odsjecanja  
% kepstralne sekvence na CL uzoraka (npr. CL=12).
```

```
% rade se dvije matrice: idft_mtx i dft_mtx.
```

```
% matrica idft_mtx je dimenzija (CL+1)x(2*M) i sadrzi clan  
%  $\exp(j*2*\pi/(2*M)*m*n)$ . idft_mtx od 2 do M postaje zbroj njih samih i  
% idft_mtx-a na indeksima od 2*M do M+2. uzimamo samo realni dio (cos) i  
% dijelimo ga s 2*M, te uzimamo idft_mtx na indeksima od 1 do M+1.
```

```
n=[0:CL];  
m=[0:2*M-1]';  
idft_mtx=exp(sqrt(-1)*m*n*2*pi/(2*M));  
idft_mtx(2:M,:)=idft_mtx(2:M,:)+idft_mtx(2*M:-1:M+2,:);  
idft_mtx=real(idft_mtx(1:M+1,:))/(2*M);
```

```
% dft_mtx je matrica sa  $\exp(-j*2*\pi/(2*M)*m*n)$ , i dimenzija je (2*M)x(M+1).  
% Nakon slicnog postupka kao gore, dobivamo matricu (CL+1)x(M+1).
```

```
n=[0:2*M-1]';  
m=[0:M];  
dft_mtx=exp(-sqrt(-1)*n*m*2*pi/(2*M));  
dft_mtx(2:M,:)=dft_mtx(2:M,:)+dft_mtx(2*M:-1:M+2,:);  
dft_mtx=real(dft_mtx(1:CL+1,:));
```

```
% Te dvije matrice nam sluze za racunanje kratkog (brzog) kepstra, koji je  
% vrlo korisan za operacije na dugim signalima s vise kanala. On operira  
% samo s prvih CL komponenti kepstra, a ne svih 2*M.
```

```
% Kao provjeru tocnosti mozemo pomnoziti te matrice na sljedeci nacin:  
% disp(max(max(abs(dft_mtx*idft_mtx-eye(CL+1))))))
```

```
% Te dvije matrice moraju biti pseudo-inverz jedna drugoj.
```

```
% Racunanje skracenog kepstra matricnim mnozenjem, te izgledjenog spektra  
% pomocu kepstra.
```

```
cep=ym_en*idft_mtx;
```

```
ym_en_c=cep*dft_mtx;
```



```

% Dalje slijede prikazi svih signala, imp odziva i filtara, u ovisnosti da
% li to zelimo. Takodjer naredbom soundsc slusanje izlaza iz pojedinih
% kanala, i snimanje '.wav' fileova u gore odabrani folder.

```

```

if prikaz_slika_filtara;

    figure(1)
    plot(f,skala);
    if od_skala;
        title('MEL SKALA');
    else
        title('BARK SKALA');
    end;
    xlabel('frekvencija [Hz]');
    axis([0 FS/2 0 s_max])
    grid;
    pause;

    figure(2)
    plot(skala,SP);
    if od_skala;
        title('MEL FILTRI');
    else
        title('BARK FILTRI');
    end;
    xlabel('frekvencija [Hz]');
    axis([skala(1,1) s_max 0 1])
    pause;

    plot(f,SP);
    if od_skala;
        title('MEL FILTRI');
    else
        title('BARK FILTRI');
    end;
    xlabel('frekvencija [Hz]');
    axis([0 FS/2 0 1])
    pause;

    figure(3)
    plot(f,abs(SP_W));
    if od_skala;
        title('MEL FILTRI NAKON IZGLADJENIH IMP. ODZIVA');
    else
        title('BARK FILTRI NAKON IZGLADJENIH IMP. ODZIVA');
    end;
    xlabel('frekvencija [Hz]');
    axis([0 FS/2 0 1])
    pause;

    figure(4)
    plot(skala,abs(SP_W));
    if od_skala;
        title('MEL FILTRI NAKON IZGLADJENIH IMP. ODZIVA');
        xlabel('frekvencija [Mel]');
    else
        title('BARK FILTRI NAKON IZGLADJENIH IMP. ODZIVA');
        xlabel('frekvencija [Bark]');
    end;
    axis([skala(1,1) s_max 0 1])
    pause;

```

```

figure(5)
plot(f,20*log10(abs(SP_W)));
if od_skala;
    title('LOG-MEL FILTRI NAKON IZGLADJENIH IMP. ODZIVA');
else
    title('LOG-BARK FILTRI NAKON IZGLADJENIH IMP. ODZIVA');
end;
xlabel('frekvencija [Hz]');
ylabel('magnituda [db]')
axis([skala(1,1) s_max -100 0])
pause;

```

```

plot(skala,20*log10(abs(SP_W)));
if od_skala;
    title('LOG-MEL FILTRI NAKON IZGLADJENIH IMP. ODZIVA');
    xlabel('frekvencija [Mel]');
else
    title('LOG-BARK FILTRI NAKON IZGLADJENIH IMP. ODZIVA');
    xlabel('frekvencija [Bark]');
end;
ylabel('magnituda [db]')
axis([skala(1,1) s_max -100 0])
pause;

```

```

figure(6)
plot(skala,20*log10(sum(abs(SP_W)')/BW_f))
ax=axis;
axis([skala(1,1) s_max ax(3) ax(4)])
title('Odstupanje sume karakteristika od ravne');
ylabel('pogreska [dB]');
pause;

```

```

figure(7)
plot([-Nimp:Nimp],imp_trw);
title('IMPULSNI ODZIV IZGLADJENIH FILTARA')
xlabel('2*Nimp+1')
pause;

```

```

figure(8)
plot([-Nimp:Nimp],sum(imp_trw')-[zeros(1,Nimp) BW_f zeros(1,Nimp)],'r')
title('Pogreska sume svih imp. odziva od jedinичnog impulsa');
xlabel('2*Nimp+1')
ylabel('pogreska');
pause;

```

```

figure(9)
plot([-Nimp:Nimp],imp_trw(:,1));
title('imp. odziv prvog filtra');
pause;

```

```

figure(10)
plot([-Nimp:Nimp],imp_trw(:,M+1));
title('imp. odziv zadnjeg filtra');
pause;

```

end;

if prikaz_signala;

```

figure(1+10*prikaz_slika_filtara)
plot([1:length(y)]/FS,y);

```

```

ax=axis;
axis([ax(1) length(y)/FS ax(3) ax(4)]);
title('SIGNAL KOJI OBRADJUJEMO')
xlabel('vrijeme [sec]')
soundsc(y,FS);
pause;

plot([1:length(y)]/FS,ym(:,5));
ax=axis;
axis([ax(1) length(y)/FS ax(3) ax(4)]);
title('5. kanal signala')
xlabel('vrijeme [sec]')
soundsc(ym(:,5),FS);
pause;

plot([1:length(y)]/FS,ym(:,10));
ax=axis;
axis([ax(1) length(y)/FS ax(3) ax(4)]);
title('10. kanal signala')
xlabel('vrijeme [sec]')
soundsc(ym(:,10),FS);
pause;

plot([1:length(y)]/FS,ym(:,15));
ax=axis;
axis([ax(1) length(y)/FS ax(3) ax(4)]);
title('15. kanal signala')
xlabel('vrijeme [sec]')
soundsc(ym(:,15),FS);
pause;

plot([1:length(y)]/FS,ym(:,20));
ax=axis;
axis([ax(1) length(y)/FS ax(3) ax(4)]);
title('20. kanal signala')
xlabel('vrijeme [sec]')
soundsc(ym(:,20),FS);
pause;

plot([1:length(y)]/FS,ym(:,25));
ax=axis;
axis([ax(1) length(y)/FS ax(3) ax(4)]);
title('25. kanal signala')
xlabel('vrijeme [sec]')
soundsc(ym(:,25),FS);
pause;

plot([1:length(y)]/FS,sum(ym'))
ax=axis;
axis([ax(1) length(y)/FS ax(3) ax(4)]);
title('Suma izlaza iz svih kanala')
xlabel('vrijeme [sec]')
soundsc(sum(ym'),'FS');
pause;

plot([1:length(y)]/FS,[sum(ym')' y])
ax=axis;
axis([ax(1) length(y)/FS ax(3) ax(4)]);
title('Suma izlaza iz svih kanala i ulazni signal (zeleni)')
xlabel('vrijeme [sec]')
pause;

```

```

figure(2+10*prikaz_slika_filtara)
plot([1:kor_dec:length(y)]/FS,ym_en(1:kor_dec:end,:));
axis([0 length(y)/FS -100 -20]);
title('Log-energija svakog pojasa');
xlabel('vrijeme [sec]');
ylabel('energija [dB]');
pause;

plot([1:kor_dec:length(y)]/FS,ym_en(1:kor_dec:end,15));
axis([0 length(y)/FS -80 -30]);
title('Log-energija 15. pojasa');
xlabel('vrijeme [sec]');
ylabel('energija [dB]');
pause;

pcolor([1:kor_dec:length(y)]/FS,[0:M],ym_en(1:kor_dec:end,:));
shading interp;
caxis([-80 -30]);
colorbar;
title('Log-energija svakog pojasa');
xlabel('vrijeme [sec]');
if od_skala;
    ylabel('indeks Mel-pojasa');
else
    ylabel('indeks Bark-pojasa');
end;
pause;

figure(3+10*prikaz_slika_filtara)
plot([0:M],[ym_en(3000,:) ym_en_c(3000,:)])
ax=axis;
axis([0 M ax(3) ax(4)]);
title('3000. uzorak, prije i nakon izgladjivanja kepstrom')
pause;

plot([1:length(y)]/FS,cep(:,1));
axis([0 length(y)/FS -80 -30]);
title('Multi kepstralni koef cep[0]');
xlabel('vrijeme [sec]');
ylabel('[dB]');
pause;

pcolor([1:kor_dec:length(y)]/FS,[0:M],ym_en_c(1:kor_dec:end,:));
shading interp;
caxis([-80 -30]);
colorbar;
title('Log-energija svakog pojasa - nakon skracenog spektra');
xlabel('vrijeme [sec]');
if od_skala;
    ylabel('indeks Mel-pojasa');
else
    ylabel('indeks Bark-pojasa');
end;
pause;

figure(4+10*prikaz_slika_filtara)
pcolor([1:kor_dec:length(y)]/FS,[0:M],raz(1:kor_dec:end,:));
shading interp;
caxis([-10 10]);
colorbar;
title('Razlika Log-en spektra prije i nakon izgladjivanja kratkim kepstrom');
xlabel('vrijeme [sec]');

```

```
if od_skala;
    ylabel('indeks Mel-pojasa');
else
    ylabel('indeks Bark-pojasa');
end;
pause;

end;

if zapis_zvuka_po_kanalima;

    for i=1:M+1,
        zvuk_kanal=sprintf('%s%s_%d',zvuk_po_kanalima,pref,i);
        wavwrite(ym(:,i)/max(abs(ym(:,i))),FS,NBITS,zvuk_kanal);
    end;
    zvuk_kanal=sprintf('%s%s_%s',zvuk_po_kanalima,pref,'sum');
    wavwrite(sum(ym'),' ',FS,NBITS,zvuk_kanal);

end;
```

- LITERATURA -

1. SLAJDOVI SA PREDAVANJA IZ KOLEGIJA NAPREDNE METODE DIGITALNE OBRADNE SIGNALA
2. PRIPREMA ZA LABORATORIJSKE VJEŽBE IZ KOLEGIJA NAPREDNE METODE DIGITALNE OBRADNE SIGNALA
3. SKRIPTA IZ KOLEGIJA DIGITALNA OBRADA GOVORA
4. PRIPREME ZA LABORATORIJSKE VJEŽBE IZ KOLEGIJA DIGITALNA OBRADA GOVORA
5. INTERNET