

Primena digitalnog filtera za audio signal

Predmet: Multimedijalni signali i sistemi

Predavač: dr Nataša Savić

Asistent: Nikola Milutinović

Primena

Digitalni filteri su osnovni alati digitalne obrade signala (DSP) i koriste se za analizu, obradu i poboljšanje signala u gotovo svim savremenim tehničkim sistemima.

Koriste se za:

1. Uklanjanje šuma i smetnji
2. Izdvajanje korisnog signala
3. Oblikovanje spektra signala
4. Poboljšanje kvaliteta signala
5. Detekciju i analizu signala.

Osnovni tipovi digitalnih filtera (prema frekvencijskom odzivu)

Tipovi digitalnih filtera prema frekvencijskom odzivu

1. Niskopropusni filter (Low-pass) - propušta niske frekvencije, potiskuje visoke
2. Visokopropusni filter (High-pass) - propušta visoke frekvencije, potiskuje niske.
3. Pojasnopropusni filter (Band-pass) - propušta samo određeni frekvencijski opseg.
4. Pojasnozaustavni filter (Band-stop / Notch) - potiskuje uzak ili širok opseg frekvencija.

Tipovi digitalnih filtera prema strukturi

1. IIR filteri (Infinite Impulse Response)

Osobine: beskonačan impulsni odziv, povratna sprega

2. FIR filteri (Finite Impulse Response)

Osobine: konačan impulsni odziv, uvek stabilni, mogu imati linearnu fazu

Tipovi digitalnih filtera prema vremenskim karakteristikama

1. Linearni i nelinearni filteri
2. Vremenski promenljivi i nepromenljivi

%% 1) Osnovni parametri signala

```
fs = 44100;           % Frekvencija odabiranja [Hz]
T = 3;               % Trajanje signala [s]
t = (0:1/fs:T-1/fs).'; % Vremenska osa (kolona)
```

% Osnovne frekvencije koje čine "koristan" audio signal

```
f1 = 220; % Osnovni ton (A3)
f2 = 440; % Drugi harmonik (A4)
f3 = 660; % Treći harmonik
```

% Parametri smetnji i šuma

```
f_interf = 1000; % Uska smetnja (za notch filter)
noiseAmp = 0.08; % Amplituda belog šuma
hfAmp = 0.25; % Amplituda visokofrekventne smetnje
f_hf = 9000; % Visokofrekventna smetnja (za LPF)
```

%% 2) Generisanje sintetičkog audio signala

% Čisti signal (superpozicija sinusnih talasa)

```
x_clean = 0.45*sin(2*pi*f1*t) + ...
          0.30*sin(2*pi*f2*t) + ...
          0.20*sin(2*pi*f3*t);
```

% Dodavanje smetnji i šuma

```
x = x_clean ...
    + 0.20*sin(2*pi*f_interf*t) ... % Uska sinusna smetnja
    + hfAmp*sin(2*pi*f_hf*t) ... % Visokofrekventna smetnja
    + noiseAmp*randn(size(t)); % Beli Gaussov šum
```

% Normalizacija signala radi izbegavanja klipovanja

```
x = x ./ max(abs(x));
```

%% 3) Projektovanje digitalnih filtera

% 3.1 Niskopropusni (Low-pass) IIR filter

% Uklanja visoke frekvencije iznad 3 kHz

```
lpCutoff = 3000; % Granična frekvencija [Hz]
```

```
lpFilt = designfilt('lowpassiir', ...
```

```
    'FilterOrder', 8, ...
```

```
    'HalfPowerFrequency', lpCutoff, ...
```

```
    'SampleRate', fs);
```

% 3.2 Pojasnopropusni (Band-pass) IIR filter

% Propušta opseg 200–1200 Hz

```
bpLo = 200; % Donja granična frekvencija
```

```
bpHi = 1200; % Gornja granična frekvencija
```

```
bpFilt = designfilt('bandpassiir', ...
```

```
    'FilterOrder', 10, ...
```

```
    'HalfPowerFrequency1', bpLo, ...
```

```
    'HalfPowerFrequency2', bpHi, ...
```

```
    'SampleRate', fs);
```

% 3.3 Notch (uski band-stop) filter

% Uklanja smetnju oko 1 kHz

```
w0 = f_interf/(fs/2); % Normalizovana frekvencija (0–1)
```

```
bw = w0/35; % Širina pojasa (podešava se po potrebi)
```

```
[bN, aN] = iirnotch(w0, bw);
```

%% 4) Primena filtera (bez fazne distorzije)

% filtfilt primenjuje filtriranje unapred i unazad (nulta faza)

```
y_lp = filtfilt(lpFilt, x);    % Samo low-pass
y_bp  = filtfilt(bpFilt, x);    % Samo band-pass
y_notch = filtfilt(bN, aN, x);  % Samo notch
y_chain = filtfilt(lpFilt, filtfilt(bN, aN, x)); % Notch + LPF
```

%% 5) Prikaz signala u vremenskom domenu (prvih 40 ms)

```
Nview = round(0.04*fs);
figure('Name','Vremenski domen (prvih 40 ms)');
plot(t(1:Nview)*1000, x(1:Nview), 'LineWidth', 1); hold on;
plot(t(1:Nview)*1000, y_lp(1:Nview), 'LineWidth', 1);
plot(t(1:Nview)*1000, y_notch(1:Nview), 'LineWidth', 1);
plot(t(1:Nview)*1000, y_chain(1:Nview), 'LineWidth', 1);
grid on;
xlabel('Vreme [ms]');
ylabel('Amplituda');
legend('Originalni signal','Low-pass','Notch','Notch + Low-pass');
```

%% 6) Analiza u frekvencijskom domenu (FFT)

```
Nfft = 2^nextpow2(length(x)); % Dužina FFT-a (stepen broja 2)
f = (0:(Nfft/2))*(fs/Nfft);    % Frekvencijska osa
```

% Izračunavanje FFT-a

```
X = fft(x, Nfft);
YLP = fft(y_lp, Nfft);
YN = fft(y_notch, Nfft);
YC = fft(y_chain, Nfft);
```

% Jednostrani amplitudski spektri

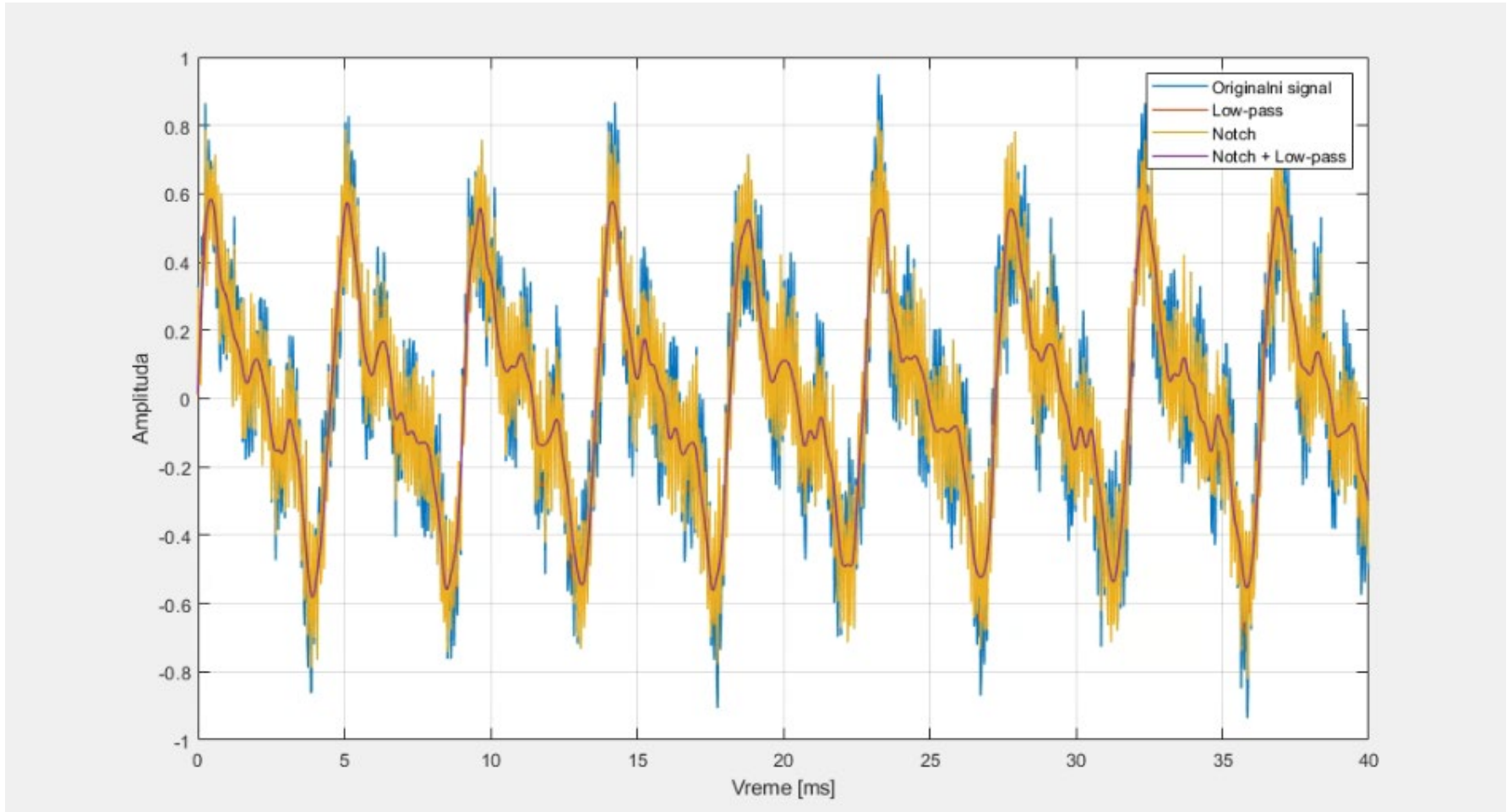
```
magX = abs(X(1:Nfft/2+1));
magLP = abs(YLP(1:Nfft/2+1));
magN = abs(YN(1:Nfft/2+1));
magC = abs(YC(1:Nfft/2+1));
```

```
figure('Name','Frekvencijski domen (FFT)');
plot(f, 20*log10(magX + eps), 'LineWidth', 1); hold on;
plot(f, 20*log10(magLP + eps), 'LineWidth', 1);
plot(f, 20*log10(magN + eps), 'LineWidth', 1);
plot(f, 20*log10(magC + eps), 'LineWidth', 1);
grid on;
xlabel('Frekvencija [Hz]');
ylabel('Amplituda [dB]');
xlim([0 12000]);
legend('Original','Low-pass','Notch','Notch + Low-pass');
```

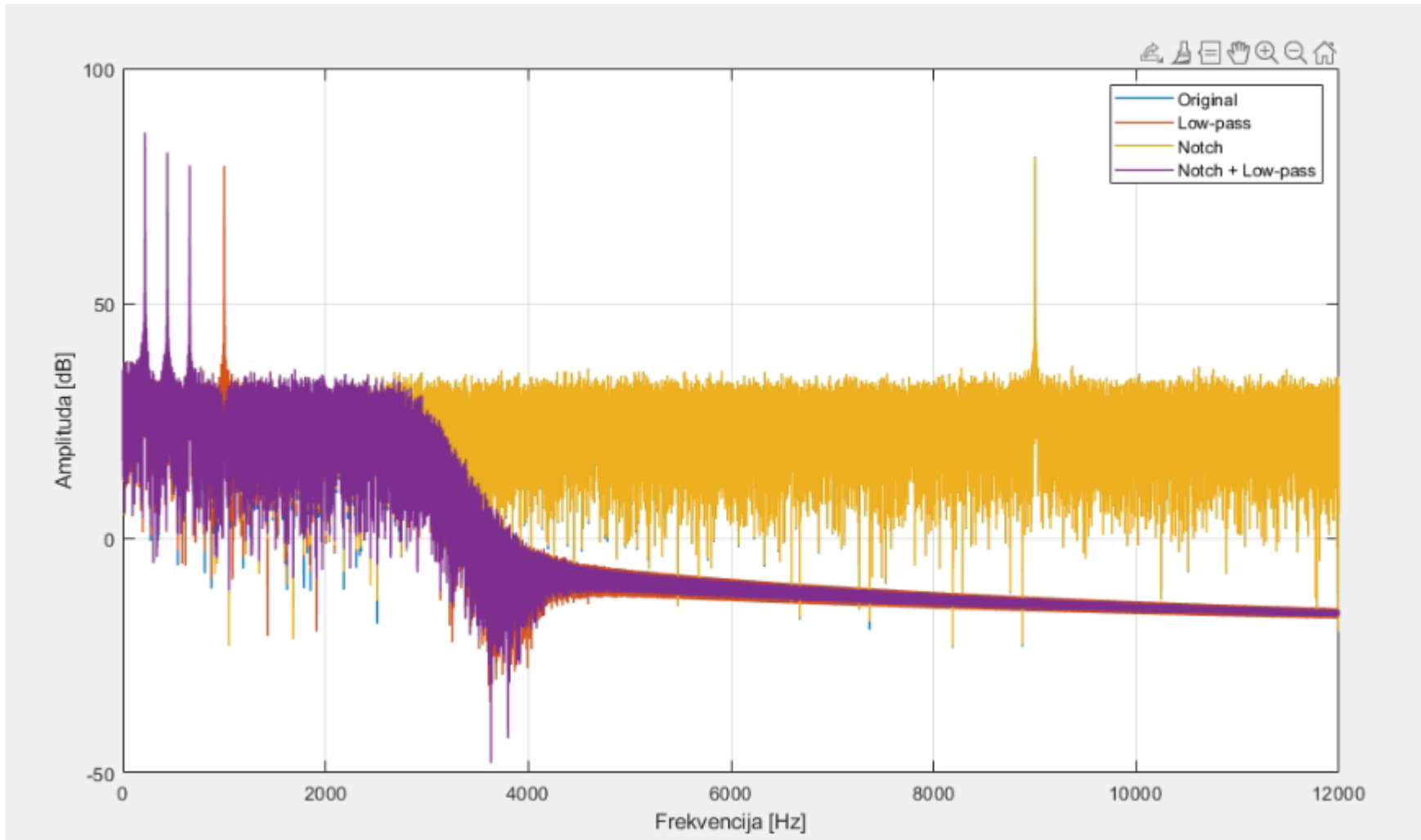
%% 7) Ispis u komandnom prozoru

```
fprintf('Frekvencija odabiranja: %d Hz\n', fs);
fprintf('Trajanje signala: %.2f s\n', T);
fprintf('Low-pass granična frekvencija: %d Hz\n', lpCutoff);
fprintf('Band-pass opseg: %d – %d Hz\n', bpLo, bpHi);
fprintf('Notch filter na %d Hz\n', f_interf);
```

REZULTAT – Vremenski domen



REZULTAT – Frekvencijski domen



%% 1) Osnovni parametri signala

```
fs = 44100;           % Frekvencija odabiranja [Hz]
T = 3;               % Trajanje signala [s]
t = (0:1/fs:T-1/fs).'; % Vremenska osa (kolona)
```

% Osnovne frekvencije "korisnog" audio sadržaja

```
f1 = 220; % A3
f2 = 440; % A4
f3 = 660; % Viši harmonik
```

% Parametri smetnji i šuma

```
f_interf = 1000;      % Uska smetnja (za notch)
noiseAmp = 0.08;     % Amplituda belog šuma
hfAmp = 0.25;        % Amplituda visokofrekventne smetnje
f_hf = 9000;         % Visokofrekventna smetnja (za LPF)
```

%% 2) Generisanje sintetičkog audio signala

```
x_clean = 0.45*sin(2*pi*f1*t) + ...
          0.30*sin(2*pi*f2*t) + ...
          0.20*sin(2*pi*f3*t);
```

% Dodavanje uske smetnje + HF smetnje + belog šuma

```
x = x_clean ...
    + 0.20*sin(2*pi*f_interf*t) ...
    + hfAmp*sin(2*pi*f_hf*t) ...
    + noiseAmp*randn(size(t));
```

% Normalizacija radi izbegavanja klipovanja

```
x = x ./ max(abs(x));
```

%% 3) Projektovanje FIR filtera

% 3.1 FIR niskopropusni filter (low-pass)

```
lpCutoff = 3000;      % Granična frekvencija [Hz]
N_lp = 200;           % Red FIR filtera (povećati za strmiji prelaz)
Wn_lp = lpCutoff/(fs/2); % Normalizovana granična frekvencija (0-1)
b_lp = fir1(N_lp, Wn_lp, 'low'); % Koeficijenti FIR LPF
```

% 3.2 FIR pojasnopropusni filter (band-pass)

```
bpLo = 200;           % Donja granična frekvencija [Hz]
bpHi = 1200;          % Gornja granična frekvencija [Hz]
N_bp = 300;           % Red FIR BPF
Wn_bp = [bpLo bpHi]/(fs/2); % Normalizovane granične frekvencije
b_bp = fir1(N_bp, Wn_bp, 'bandpass');
```

% 3.3 FIR notch filter (uski band-stop oko f_interf)

% Napomena: Notch kao FIR se obično pravi kao band-stop vrlo uskog opsega.

```
notchBW = 40;         % Širina notch opsega [Hz] (npr. +/- 20 Hz)
notchLo = max(1, f_interf - notchBW/2);
notchHi = min(fs/2-1, f_interf + notchBW/2);
N_notch = 400;        % Red FIR notch filtera (veći red -> uži i strmiji notch)
Wn_notch = [notchLo notchHi]/(fs/2);
b_notch = fir1(N_notch, Wn_notch, 'stop'); % Band-stop FIR
```

%% 4) Primena FIR filtera

% Za FIR filtere koristimo filter(b,1,x).

% Pošto FIR uvodi grupno kašnjenje (delay), ovde radimo kompenzaciju kašnjenja

% tako što "odsečemo" početak i dopunimo kraj nulama (opciono).

```
y_lp_raw = filter(b_lp, 1, x);
```

```
y_bp_raw = filter(b_bp, 1, x);
```

```
y_notch_raw = filter(b_notch, 1, x);
```

% Grupno kašnjenje FIR filtera je približno $N/2$ uz linearno-fazne FIR filtere.

```
d_lp = N_lp/2;
```

```
d_bp = N_bp/2;
```

```
d_notch = N_notch/2;
```

% Kompenzacija kašnjenja (poravnanje sa originalom radi poređenja)

```
y_lp = [y_lp_raw(d_lp+1:end); zeros(d_lp,1)];
```

```
y_bp = [y_bp_raw(d_bp+1:end); zeros(d_bp,1)];
```

```
y_notch = [y_notch_raw(d_notch+1:end); zeros(d_notch,1)];
```

% Lančano filtriranje: notch pa low-pass (često praktično)

```
y_chain_raw = filter(b_lp, 1, filter(b_notch, 1, x));
```

```
d_chain = d_notch + d_lp;
```

```
y_chain = [y_chain_raw(d_chain+1:end); zeros(d_chain,1)];
```

%% 5) Prikaz signala u vremenskom domenu (prvih 40 ms)

```
Nview = round(0.04*fs);
```

```
figure('Name','Vremenski domen (prvih 40 ms) - FIR');
```

```
plot(t(1:Nview)*1000, x(1:Nview), 'LineWidth', 1); hold on;
```

```
plot(t(1:Nview)*1000, y_lp(1:Nview), 'LineWidth', 1);
```

```
plot(t(1:Nview)*1000, y_notch(1:Nview), 'LineWidth', 1);
```

```
plot(t(1:Nview)*1000, y_chain(1:Nview), 'LineWidth', 1);
```

```
grid on;
```

```
xlabel('Vreme [ms]');
```

```
ylabel('Amplituda');
```

```
legend('Originalni signal','FIR Low-pass','FIR Notch','FIR Notch + FIR Low-pass');
```

%% 6) Analiza u frekvencijskom domenu (FFT)

```
Nfft = 2^nextpow2(length(x));
```

```
f = (0:(Nfft/2))*(fs/Nfft);
```

```
X = fft(x, Nfft);
```

```
YLP = fft(y_lp, Nfft);
```

```
YBP = fft(y_bp, Nfft);
```

```
YN = fft(y_notch, Nfft);
```

```
YC = fft(y_chain, Nfft);
```

```
magX = abs(X(1:Nfft/2+1));
```

```
magLP = abs(YLP(1:Nfft/2+1));
```

```
magBP = abs(YBP(1:Nfft/2+1));
```

```
magN = abs(YN(1:Nfft/2+1));
```

```
magC = abs(YC(1:Nfft/2+1));
```

```
figure('Name','Frekvencijski domen (FFT) - FIR');
plot(f, 20*log10(magX + eps), 'LineWidth', 1); hold on;
plot(f, 20*log10(magLP + eps), 'LineWidth', 1);
plot(f, 20*log10(magN + eps), 'LineWidth', 1);
plot(f, 20*log10(magC + eps), 'LineWidth', 1);
grid on;
xlabel('Frekvencija [Hz]');
ylabel('Amplituda [dB]');
xlim([0 12000]);
legend('Original','FIR Low-pass','FIR Notch','FIR Notch + FIR Low-pass');
```

%% 7) Opcioni prikaz frekvencijskog odziva filtera

% freqz omogućava proveru karakteristika FIR filtera.

```
figure('Name','Frekvencijski odziv FIR filtera');
freqz(b_lp, 1, 4096, fs); grid on; title('FIR Low-pass');
```

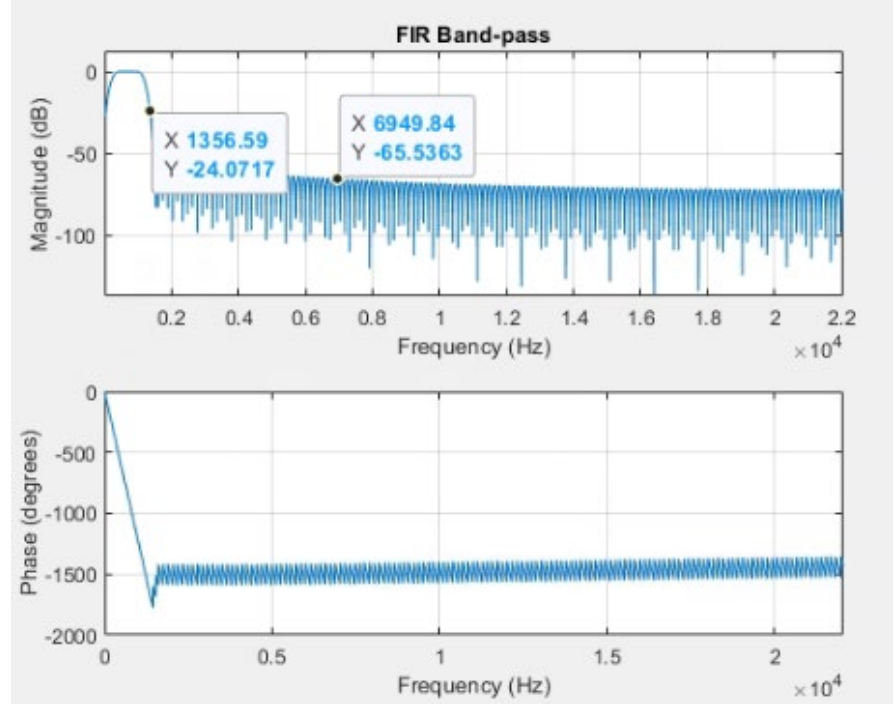
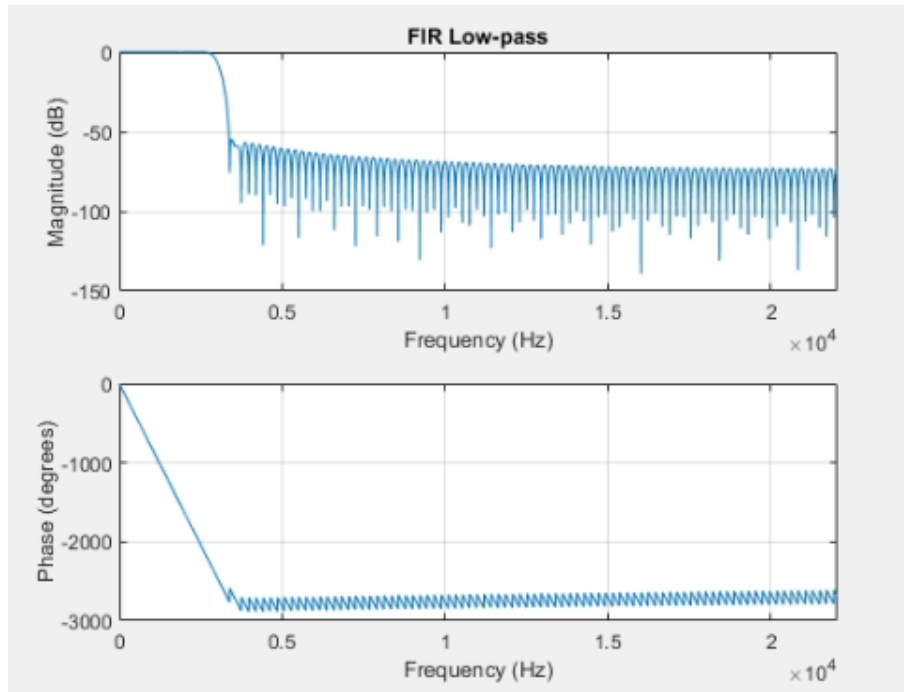
```
figure('Name','Frekvencijski odziv FIR filtera');
freqz(b_bp, 1, 4096, fs); grid on; title('FIR Band-pass');
```

```
figure('Name','Frekvencijski odziv FIR filtera');
freqz(b_notch, 1, 4096, fs); grid on; title('FIR Notch (band-stop)');
```

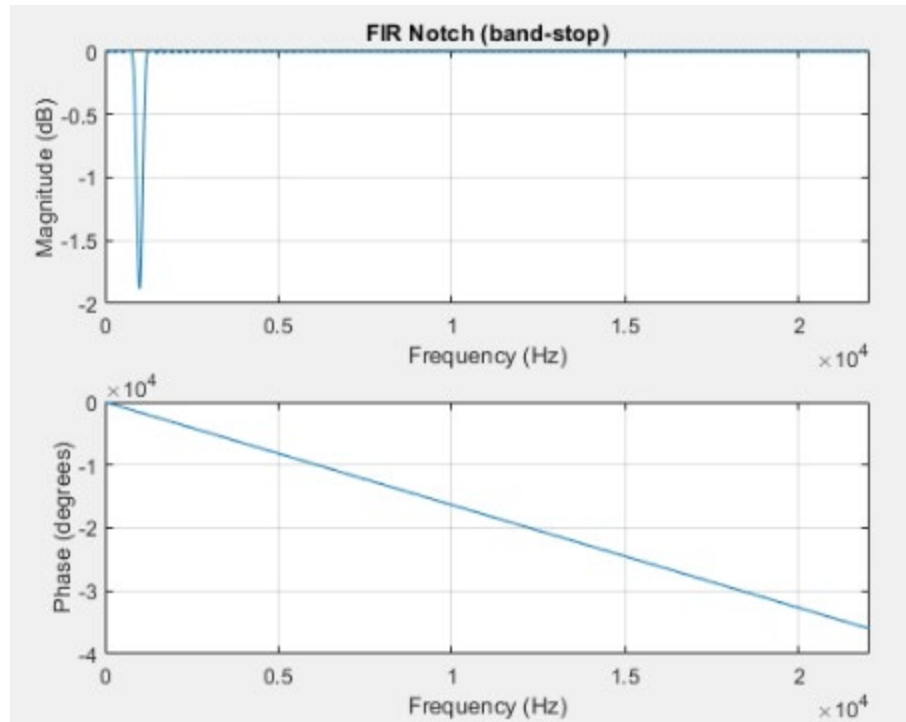
%% 9) Informativni ispis

```
fprintf('Frekvencija odabiranja: %d Hz\n', fs);
fprintf('Trajanje signala: %.2f s\n', T);
fprintf('FIR LPF: red N=%d, fc=%d Hz\n', N_lp, lpCutoff);
fprintf('FIR BPF: red N=%d, opseg=%d-%d Hz\n', N_bp, bpLo, bpHi);
fprintf('FIR Notch: red N=%d, opseg=%d-%d Hz (oko %d Hz)\n', ...
    N_notch, round(notchLo), round(notchHi), f_interf);
```

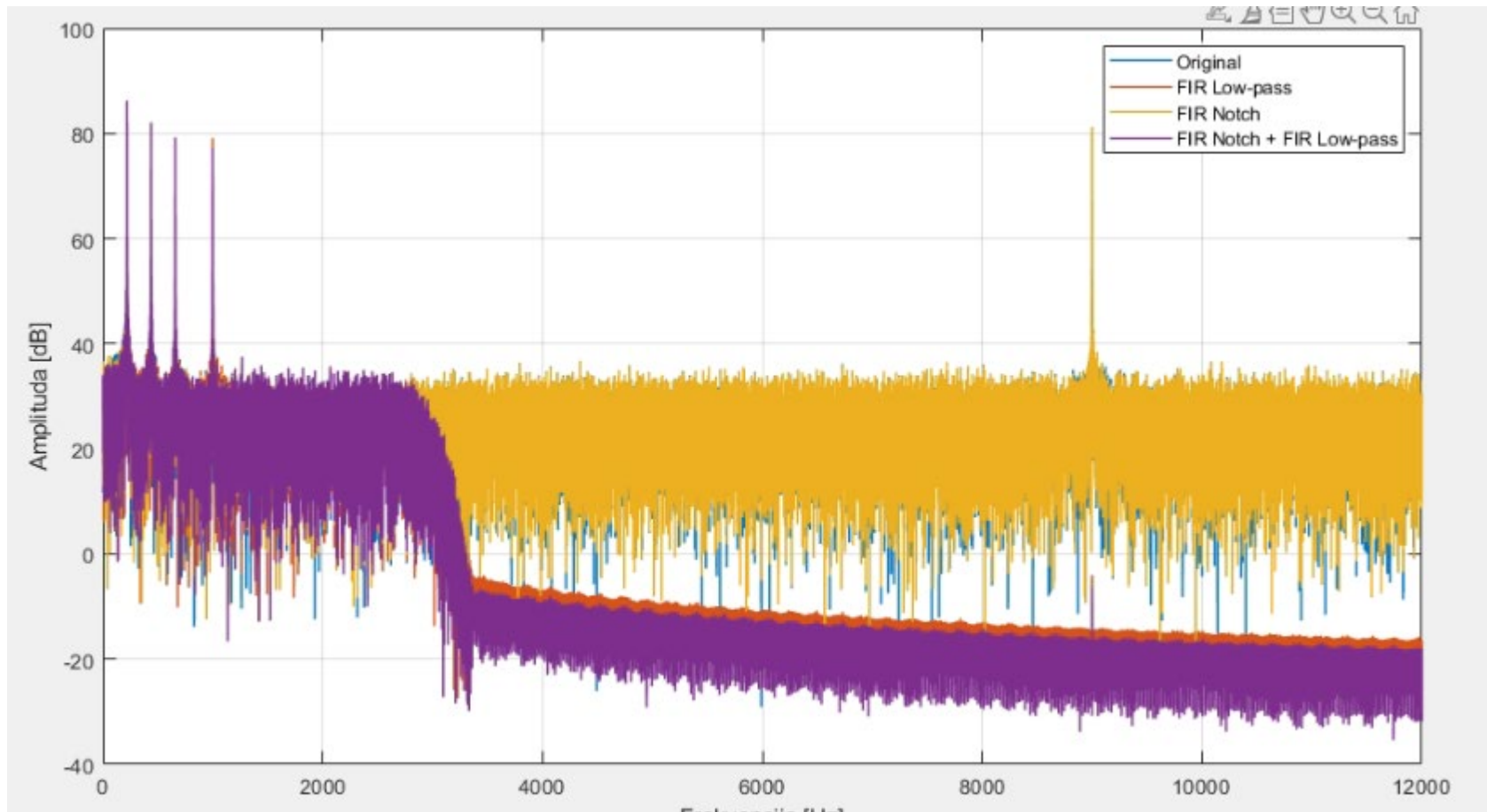
REZULTAT – Frekvencijski odziv



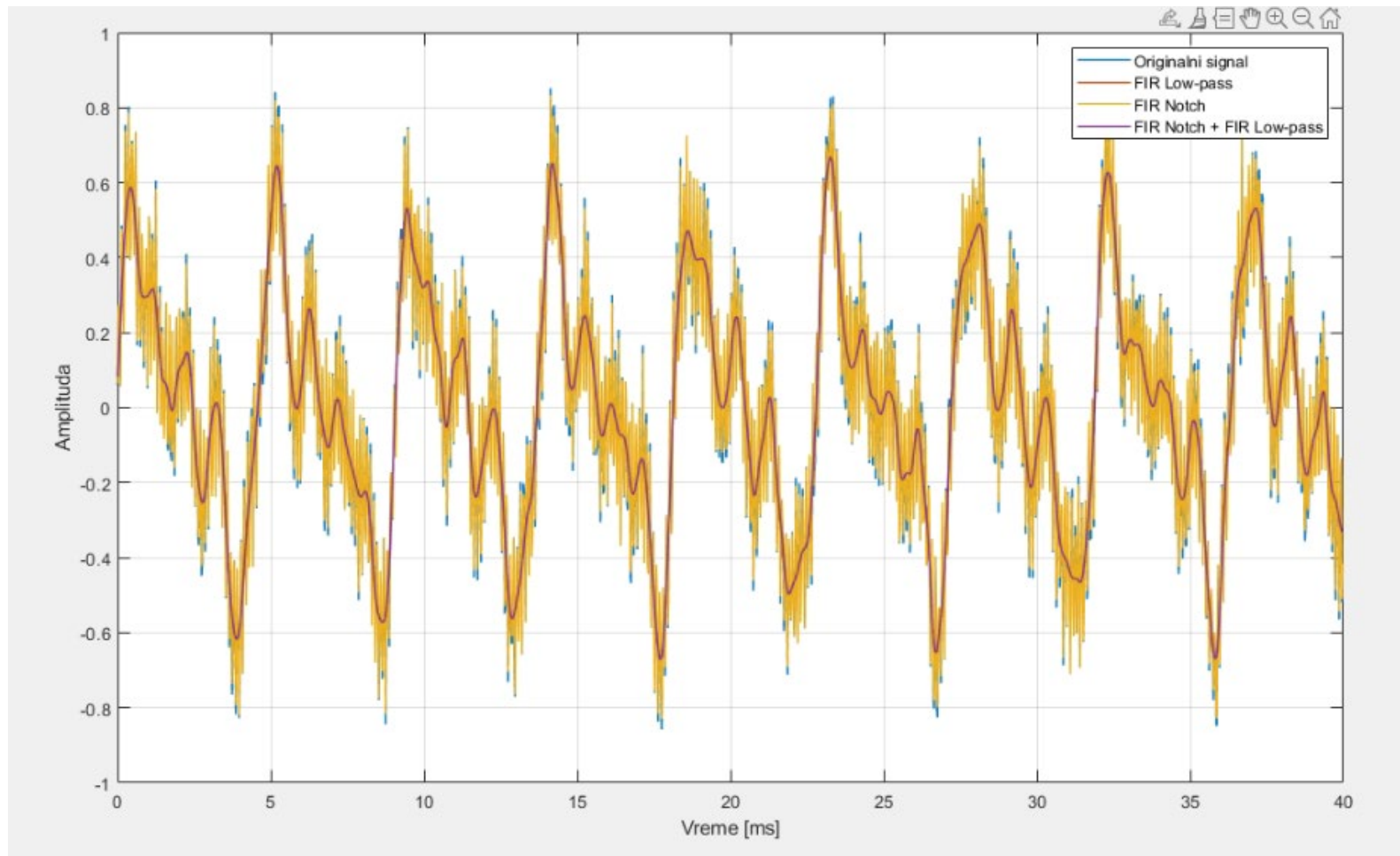
REZULTAT – Frekvencijski odziv



REZULTAT – Frekvencijski domen



REZULTAT – Vremenski domen



Hvala na pažnji!

PITANJA?

