

ZADACI — SINTEZA LVN DISKRETNIH SISTEMA

*Rešeni ispitni zadaci — FIR filtri, realizacije, SIMULINK i kvantizacija
(MATLAB)*

Sadržaj

ISPIT 1 — FIR prozorska metoda (NF filter, Hanov prozor)

a) projektovanje · b) direktna i transponovana + SIMULINK · c) kvantizacija 8 bita

ISPIT 2 — Parks-McClellan (VF filter)

a) projektovanje · b) kaskada (transp. sekcije) + SIMULINK · c) kvantizacija [4 2]

Dodatak — slike struktura i SIMULINK kontrolna lista

ISPIT 1 — FIR prozorska metoda

Zadatak je pretkucan sa poslate slike. Delovi koji se na slici ne vide jasno dopunjeni su standardnom formulacijom.

TEKST ZADATKA

1. a) (5b) Korišćenjem metode sinteze digitalnih FIR filtara zasnovane na Furijeovoj transformaciji i prozorskim funkcijama projektovati sledeći digitalni filter: **NF (niskopropusni) filter 7. reda** sa graničnom učestanošću od **1800 Hz** korišćenjem **Hanovog prozora**. Nacrtati amplitudsku i faznu karakteristiku, raspored polova i nula u Z-ravni i odrediti i nacrtati odziv na ulazni signal:

$$x(n) = \sin(2\pi \cdot 900/fs \cdot n) + \sin(2\pi \cdot 2800/fs \cdot n) + \sin(2\pi \cdot 4800/fs \cdot n)$$

Učestanost odabiranja $f_s = 11200$ Hz.

b) (5b) Projektovani filter realizovati korišćenjem **direktne** i **transponovane direktne** realizacije. Formirati odgovarajuće SIMULINK modele i simulacijom odrediti odzive na $x(n)$ za svaku strukturu. Uporediti rezultate iz MATLAB-a i SIMULINK-a.

c) (5b) Za svaki model izvršiti kvantizaciju koeficijenata prenosne funkcije korišćenjem **8 bita**. Za svaku realizaciju nacrtati amplitudsku i faznu karakteristiku i raspored polova i nula u Z-ravni. Detaljno komentarisati rezultate.

a) Projektovanje filtra

Prozorska metoda množi idealni (beskonačni) impulsni odziv niskopropusnog filtra Hanovim prozorom konačne dužine. U MATLAB-u to radi funkcija `firl`:

Ispit1_a.m (1/2)

```
%% ===== ISPIT 1: NF FIR filter 7. reda, Hanov prozor =====
clear; close all; clc;

%% a) PROJEKTOVANJE
N = 7; % red filtra
M = N + 1; % broj koeficijenata = 8
fs = 11200; % ucestanost odabiranja [Hz]
fc = 1800; % granicna ucestanost [Hz]
Wn = fc/(fs/2); % normalizovana granica (Nyquist = fs/2)

% prozorska metoda: idealni NF odziv * Hanov prozor duzine M
b = firl(N, Wn, 'low', hann(M).'); % vektor koeficijenata (h(n))
a = 1; % FIR -> imenilac = 1

% (alternativno 'rucno', da se vidi princip:)
% wc = 2*pi*fc/fs; m = (0:N) - N/2;
% hd = sin(wc*m)./(pi*m); % idealni NF impulsni odziv (sinc)
% hd(m==0) = wc/pi; % granicna vrednost u m=0
% b = hd .* hann(M).'; % * Hanov prozor
```

Crtanje karakteristika, Z-ravni i odziva na zadati ulaz:

Ispit1_a.m (2/2)

```
% Amplitudska i fazna karakteristika (preko FFT)
Nfft = 1024;
H = fft(b, Nfft);
Ha = abs(H(1:Nfft/2));
Hf = angle(H(1:Nfft/2));
k = 0:Nfft/2-1;
w = pi*k/(Nfft/2-1);          % normalizovana ucestanost [0, pi]

figure('Name','ISPIT1 a) karakteristike');
subplot(2,1,1), plot(w, 20*log10(Ha)), grid on
xlabel('\omega [rad/od]'), ylabel('|H| [dB]')
title('Amplitudska karakteristika')
subplot(2,1,2), plot(w, Hf), grid on
xlabel('\omega [rad/od]'), ylabel('faza [rad]')
title('Fazna karakteristika')

figure('Name','ISPIT1 a) Z-ravan'); zplane(b, a)
title('Raspored polova i nula')

%% Odziv na ulazni signal
n = 0:99;
x = sin(2*pi*900*n/fs) + sin(2*pi*2800*n/fs) + sin(2*pi*4800*n/fs);
y = filter(b, a, x);

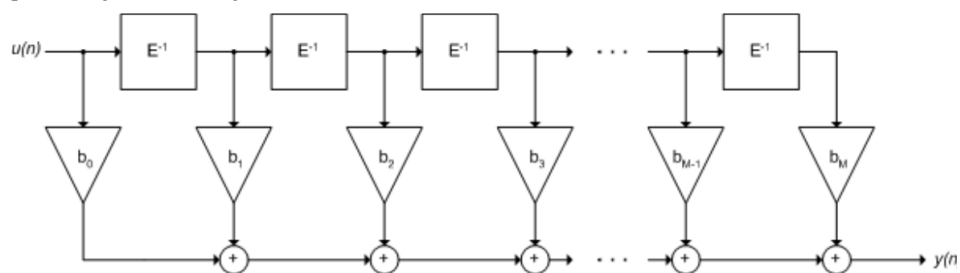
figure('Name','ISPIT1 a) odziv');
subplot(3,1,1), stem(0:M-1, b), grid on, title('Impulsni odziv h(n)=b')
subplot(3,1,2), stem(n, x), grid on, title('Ulazni signal x(n)')
subplot(3,1,3), stem(n, y), grid on, title('Izlazni signal y(n)')
```

✓ Šta očekuješ na grafiku (a)

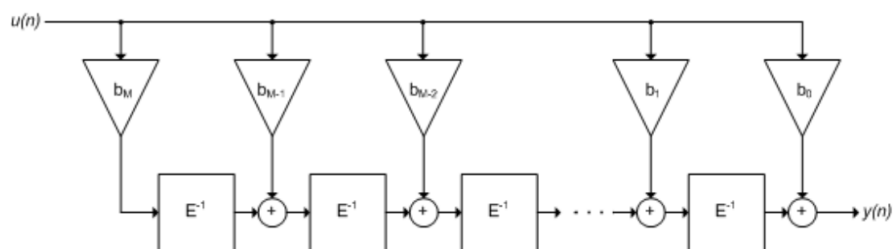
Granična učestanost 1800 Hz je na $\omega = 2\pi \cdot 1800 / 11200 \approx 1.01$ rad. Komponenta na **900 Hz prolazi** (ispod granice), a **2800 Hz i 4800 Hz se slabe** (iznad granice). Zato izlaz $y(n)$ u ustaljenom stanju $\approx \sin(2\pi \cdot 900 / fs \cdot n)$ — ostane samo najniža komponenta. Faza je linearna (FIR sa simetričnim h).

b) Direktna i transponovana direktna realizacija + SIMULINK

FIR direktna forma sledi direktno iz RUI: lanac kašnjenja z^{-1} , sa svakog odvoda množenje koeficijentom b_k i sabiranje. Transponovana forma koristi **iste koeficijente** b — razlikuje se samo raspored elemenata (kraći najduži propagacioni put).



Direktna realizacija FIR sistema (opšti oblik).



Transponovana direktna forma FIR sistema (isti koeficijenti b).

Ispit1_b.m

```
%% b) PRIPREMA ZA SIMULINK
t = n/fs; % VREMENSKA OSA U SEKUNDAMA (ključno!)
x_sim = [t; x].'; % matrica [vreme, vrednost] za From Workspace
% Stop time simulacije = t(end) = 99/fs
% Koeficijenti b su ISTI za obe forme - razlikuje se samo struktura bloka.
```

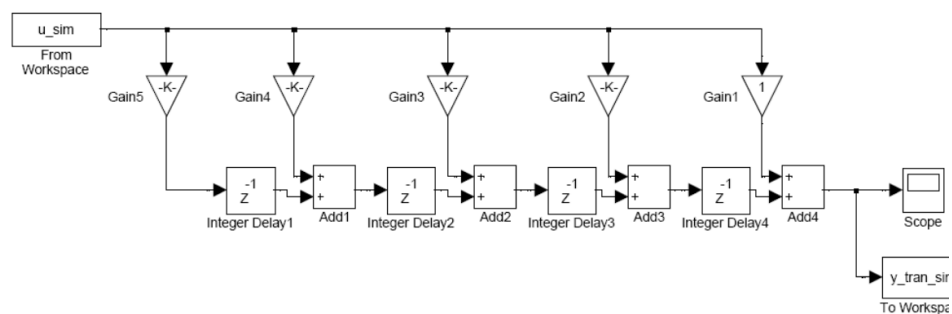
⚠ Priprema ulaza — najčešća greška!

Blok From Workspace traži matricu **[vreme, vrednost]** gde je **vreme u sekundama**. Ako se kao vreme stavi indeks n (0,1,2,...), SIMULINK misli da signal traje n sekundi i pročita samo prvi odbirak → izlaz ispadne **sve nule**. Zato $t = n/fs$, ne $[n; x]$.

SIMULINK model (dva modela — direktni i transponovani):

1. **From Workspace**: Data = x_sim , Sample time = $1/fs$, Output after final value = **Holding final value**.
2. **Discrete FIR Filter**: Coefficients = b ; Filter structure = Direct form za prvi, Transposed direct form za drugi model.
3. **To Workspace**: Variable name = y_dir_sim / y_tran_sim , **Save format = Array**.
4. **Scope** za proveru. Stop time = $99/fs$; Solver (Ctrl+E) = Fixed-step, discrete, korak $1/fs$.

Ovako izgleda SIMULINK model transponovane direktne forme (sastavljen od osnovnih Gain / Add / z^{-1} blokova):



SIMULINK model transponovane direktne FIR realizacije.

i Ako Discrete FIR Filter traži DSP System Toolbox

Opcija **Transposed direct form** u tom bloku zahteva DSP System Toolbox. Ako ga nemaš: transponovanu formu sastavi ručno od osnovnih blokova (Gain $b_0 \dots b_M$, Add, z^{-1}) kao na slici iznad, ili koristi blok Discrete Filter sa b i $a=1$. Izlaz $y(n)$ je u svakom slučaju identičan — strukture su matematički ekvivalentne.

Ispit1_b.m (poređenje)

```
%% b) POREĐENJE MATLAB &lt;-&gt; SIMULINK (posle simulacija)
y_dir = out.y_dir_sim; % ili y_dir_sim (starije verzije)
y_tran = out.y_tran_sim;

delta_dir = y - y_dir.';
delta_tran = y - y_tran.';

figure('Name','ISPIT1 b) poredjenje');
subplot(3,1,1), stem(n, y), title('y (filter)')
subplot(3,1,2), stem(n, y_dir), title('y direktna (SIMULINK)')
subplot(3,1,3), stem(n, y_tran), title('y transponovana (SIMULINK)')

disp(['max greska direktna: ' num2str(max(abs(delta_dir)))])
disp(['max greska transponov.: ' num2str(max(abs(delta_tran)))])
% obe greske ~1e-15 &gt; sve tri metode daju isti rezultat
```

c) Kvantizacija koeficijenata na 8 bita

FIR koeficijenti iz `fir1` su po modulu manji od 1, pa je prirodno uzeti **1 bit za znak + 7 razlomljenih bita** (format [8 7], opseg $[-1, 1]$):

Ispit1_c.m

```
%% c) KVANTIZACIJA KOEFICIJENATA NA 8 BITA
q = quantizer('fixed','round','saturate',[8 7]); % [ukupno=8, razlom.=7]
b_q = quantize(q, b); % kvantizovani b

Hq = fft(b_q, Nfft);
Haq = abs(Hq(1:Nfft/2));
Hfq = angle(Hq(1:Nfft/2));

figure('Name','ISPIT1 c) kvantizovano');
subplot(3,1,1), stem(0:M-1, b_q), grid on, title('Kvantizovani b_q')
subplot(3,1,2), plot(w,20*log10(Ha), w,20*log10(Haq)), grid on
    legend('original','kvantizovano'), title('Amplitudska karakteristika')
subplot(3,1,3), plot(w, Hf, w, Hfq), grid on
    legend('original','kvantizovano'), title('Fazna karakteristika')

figure('Name','ISPIT1 c) Z-ravan kvant. '); zplane(b_q, 1)
    title('Polovi i nule posle kvantizacije')
```

✓ Komentar rezultata (c)

Sa 8 bita (7 razlomljenih) zaokruživanje je relativno fino, pa se karakteristika samo blago promeni — nule se malo pomere, slabljenje u nepropusnom opsegu se malo pogorša. **Direktna i transponovana forma imaju iste koeficijente b**, pa je njihova kvantizacija **identična** → iste karakteristike i isti raspored nula. FIR ostaje **stabilan** (nema polova van koordinatnog početka), bez obzira na broj bita.

ISPIT 2 — Parks-McClellan (VF filter)

TEKST ZADATKA

Zadatak 1 (16. 5. 2024.)

a) (4b) Korišćenjem **Parks-McClellan** algoritma projektovati FIR filter: **VF (visokopropusni) filter reda 6**, $f_p = 6500$ Hz, $f_s = 6000$ Hz, $a_p = 1$ dB, $a_s = 40$ dB. Nacrtati amplitudsku i faznu karakteristiku, raspored polova i nula u Z-ravni i odrediti i nacrtati odziv na ulazni signal:

$$u(n) = 5 \cdot \cos(2\pi \cdot 2000 / f_{sr} \cdot n) + \cos(2\pi \cdot 5389 / f_{sr} \cdot n) + 4 \cdot \cos(2\pi \cdot 8500 / f_{sr} \cdot n)$$

$f_{sr} = 24050$ Hz.

b) (6b) Projektovani filter realizovati **kaskadnom realizacijom sa sekcijama drugog reda u transponovanoj direktnoj formi**. Formirati SIMULINK model i simulacijom odrediti odziv na $u(n)$. Uporediti MATLAB i SIMULINK.

c) (5b) Definirati sistem q za predstavljanje brojeva: omogućava pozitivne i negativne brojeve; **2 bita za celobrojni deo, 2 bita za razlomljeni deo**; zaokruživanje **ka $+\infty$** ; vrednosti van opsega svode se na najbližu (najveću ili najmanju) dozvoljenu — zasićenje. Kvantizovati koeficijente; nacrtati amplitudsku i faznu karakteristiku i raspored polova i nula. Komentarisati uticaj kvantizacije i da li je narušena stabilnost.

a) Projektovanje (Parks-McClellan)

Parks-McClellan (`firpmord` + `firpm`) daje optimalni FIR sa ravnomernim talasanjem. Za **visokopropusni** filter važi $f_s < f_p$ (nepropusni opseg ispod propusnog):

Ispit2_a.m (1/2)

```
%% ===== ISPIT 2: VF FIR filter reda 6, Parks-McClellan =====
clear; close all; clc;

%% a) PROJEKTOVANJE
N = 6; % red filtra
M = N + 1; % broj koeficijenata = 7
fsr = 24050; % ucestanost odabiranja [Hz]
fp = 6500; % granica propusnog opsega [Hz]
fs = 6000; % granica nepropusnog opsega [Hz]
ap = 1; % talasanje u propusnom opsegu [dB]
as = 40; % slabljenje u nepropusnom opsegu [dB]

f = [fs fp]; % VF: fs < fp ; ivice prelaznog opsega
a = [0 1]; % zeljeno pojacanje: 0 (stop), 1 (pass)
dev = [10^(-as/20) (10^(ap/20)-1)/(10^(ap/20)+1)]; % odstupanja

[~, fo, ao, wt] = firpmord(f, a, dev, fsr); % red ostaje fiksiran = 6
b = firpm(N, fo, ao, wt); % Parks-McClellan koef.
a_den = 1;
```

Ispit2_a.m (2/2)

```
% Karakteristike, Z-ravan i odziv
Nfft = 1024;
H = fft(b, Nfft);
Ha = abs(H(1:Nfft/2)); Hf = angle(H(1:Nfft/2));
k = 0:Nfft/2-1; w = pi*k/(Nfft/2-1);

figure('Name','ISPIT2 a) karakteristike');
subplot(2,1,1), plot(w,20*log10(Ha)), grid on, title('Amplitudska')
xlabel('\omega [rad/od]'), ylabel('|H| [dB]')
subplot(2,1,2), plot(w,Hf), grid on, title('Fazna')
xlabel('\omega [rad/od]'), ylabel('faza [rad]')

figure('Name','ISPIT2 a) Z-ravan'); zplane(b,a_den), title('Polovi i nule')

n = 0:49;
u = 5*cos(2*pi*2000*n/fsr) + cos(2*pi*5389*n/fsr) + 4*cos(2*pi*8500*n/fsr);
y = filter(b, a_den, u);

figure('Name','ISPIT2 a) odziv');
subplot(3,1,1), stem(0:M-1,b), grid on, title('h(n)=b')
subplot(3,1,2), stem(n,u), grid on, title('Ulaz u(n)')
subplot(3,1,3), stem(n,y), grid on, title('Izlaz y(n)')
```

✓ Šta očekuješ na grafiku (a)

VF filter propušta visoke učestanosti. Komponente na **2000 Hz i 5389 Hz** su ispod $f_s = 6000 \text{ Hz} \rightarrow$ **slabe se**; komponenta na **8500 Hz** je iznad $f_p = 6500 \text{ Hz} \rightarrow$ **prolazi**. Zato izlaz u ustaljenom stanju $\approx 4 \cdot \cos(2\pi \cdot 8500 / f_{sr} \cdot n)$.

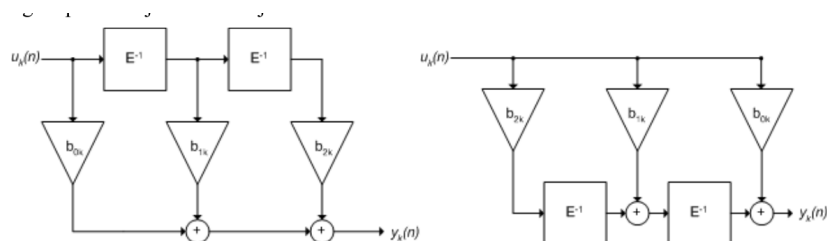
b) Kaskada (transponovane sekcije 2. reda) + SIMULINK

Kaskadna realizacija rastavlja $H(z)$ u proizvod sekcija 2. reda. Svaka sekcija je ovde u **transponovanoj direktnoj formi**. Niz sekcija:



Slika 5.21. Kaskadna realizacija FIR sistema
Kaskadna realizacija FIR sistema (niz sekcija 2. reda).

Gradivna ćelija 2. reda — levo direktna, **desno transponovana** (ovu koristimo za zadatak b):



Slika 5.22. Struktura ćelija drugog reda u slučaju korišćenja direktne forme (levo) i
Ćelija 2. reda: direktna forma (levo) i transponovana direktna forma (desno).

Faktorizacija $H(z)$ u sekcije — nule se uparuju u konjugovane parove funkcijom `cplxpair` da bi koeficijenti sekcija bili realni:

Ispit2_b.m

```
% b) KASKADNA REALIZACIJA (3 sekcije 2. reda, transponovana forma)
% H(z) = b(1) * proizvod sekcija 2. reda
zo = cplxpair(roots(b)); % 6 nula, slozenih u konjugovane parove
g0 = b(1) * poly(zo(1:2)); % prva sekcija (sa pojačanjem b(1))
g1 = poly(zo(3:4)); % druga sekcija
g2 = poly(zo(5:6)); % treća sekcija
g0 = real(g0); g1 = real(g1); g2 = real(g2); % uklanjanje ~0 imag. deo

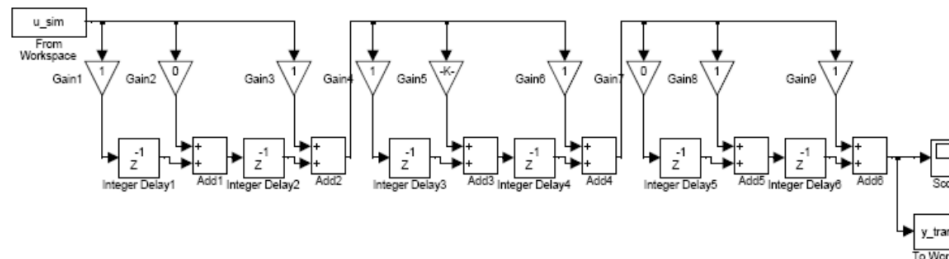
% proverava faktORIZACIJE (treba ~1e-15):
disp(max(abs( conv(conv(g0,g1),g2) - b )))

% priprema ulaza za SIMULINK
t = n/fsr; % VREME U SEKUNDAMA (ključno!)
u_sim = [t; u].'; % [vreme, vrednost]; Stop time = t(end)=49/fsr
```

i Ispravka u odnosu na poslati kod (K2.txt)

U poslatom kodu linija $g0=b(1)*conv([1 \ -zo(2), \ p[1 \ -zo(3)]]);$ ima grešku u kucanju (višak p, fali zatvorena zagrada) i ručno biranje nula $zo(2), zo(3)$ ne garantuje konjugovani par (pa koeficijenti sekcija ne bi bili realni). Verzija sa `cplxpair + poly` iznad je ispravna i uvek daje realne sekcije.

SIMULINK model (kaskada od 3 transponovane sekcije): From Workspace (u_sim) → tri **Discrete FIR Filter** bloka u nizu, Filter structure = Transposed direct form, sa koeficijentima $g0, g1, g2$ → To Workspace (y_sim , Array). Primer izgleda modela:



SIMULINK model kaskadne FIR realizacije (sekcije 2. reda u nizu).

Ako nemaš DSP toolbox, svaku sekciju sastavi ručno prema desnoj ćeliji sa slike iznad (tri Gain-a $b0/b1/b2$, dva sabirača, dva z^{-1}).

Ispit2_b.m (poređenje)

```
% b) POREĐENJE
y_sim = out.y_sim; % iz To Workspace
delta = y - y_sim.';

figure('Name','ISPIT2 b) poredjenje');
subplot(3,1,1), stem(n,y), title('y (filter)')
subplot(3,1,2), stem(n,y_sim), title('y_{sim} (SIMULINK kaskada)')
subplot(3,1,3), stem(n,delta), title('greska y - y_{sim}')
disp(['max greska: ' num2str(max(abs(delta)))]) % ~1e-15
```

c) Kvantizacija sistemom q

Tražene osobine sistema q : sa znakom (pozitivni i negativni brojevi) → 'fixed'; zaokruživanje ka $+\infty$ → 'ceil'; zasićenje → 'saturate'; 2 bita celobrojni + 2 bita razlomljeni → format [4 2] (ukupno 4 bita; razlomljenih 2; celobrojni deo zajedno sa znakom = 2).

Ispit2_c.m

```
% c) KVANTIZACIJA: sistem q (2 cela + 2 razlom. bita, ka +inf, zasicenje)
q = quantizer('mode','fixed', 'roundmode','ceil', ...
    'overflowmode','saturate', 'format',[4 2]);
b_q = quantize(q, b);

Hq = fft(b_q, Nfft);
Haq = abs(Hq(1:Nfft/2)); Hfq = angle(Hq(1:Nfft/2));

figure('Name','ISPIT2 c) kvantizovano');
subplot(4,1,1), stem(0:M-1,b_q), grid on, title('Kvantizovani b_q')
subplot(4,1,2), plot(w,20*log10(Ha), w,20*log10(Haq)), grid on
    legend('original','kvant.'), title('Amplitudska')
subplot(4,1,3), plot(w,Hf, w,Hfq), grid on
    legend('original','kvant.'), title('Fazna')
subplot(4,1,4), zplane(b_q,1), title('Polovi i nule (kvant.)')
```

⚠ Komentar rezultata i odgovor o stabilnosti

Sa samo **2 razlomljena bita** korak kvantizacije je $2^{-2} = 0.25$ — vrlo grubo. Koeficijenti se jako zaokruže, nule se osetno pomere, pa se **slabljenje u nepropusnom opsegu drastično pogorša** (više nije -40 dB) i propusni opseg se izobličići. Karakteristika je znatno lošija nego sa 8 bita.

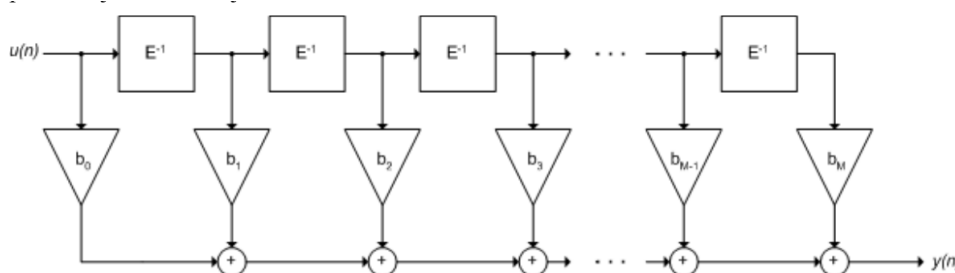
Da li je narušena stabilnost? NE. Ovo je **FIR** filter — ima samo nule, nema polova. Kvantizacija pomera nule, ali polova (osim u koordinatnom početku) nema, pa sistem **ostaje stabilan** bez obzira na grubu kvantizaciju. Kvantizacija bi mogla ugroziti stabilnost samo kod **IIR** sistema (kad bi pol izašao iz jediničnog kruga).

Pregled realizacija — blok dijagrami

Za podsećanje, pregled svih osnovnih struktura za realizaciju. Sve forme datog tipa daju **matematički isti** izlaz — razlikuju se po broju elemenata, dužini najdužeg propagacionog puta i osetljivosti na kvantizaciju.

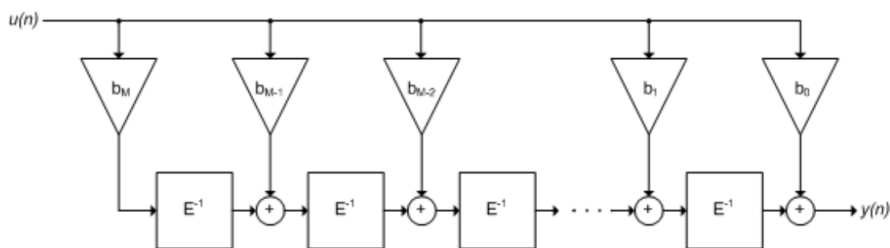
FIR realizacije

1) Direktna forma — lanac kašnjenja, sa svakog odvoda množenje koeficijentom b_k i sabiranje. $M+1$ množača, M sabirača, M pomerača.



FIR — direktna realizacija.

3) Transponovana direktna forma — transpozicija grafa (svim granama se promeni smer, čvorovi $\leftrightarrow$ sabirači, ulaz $\leftrightarrow$ izlaz). Isti koeficijenti b , kraći najduži propagacioni put.

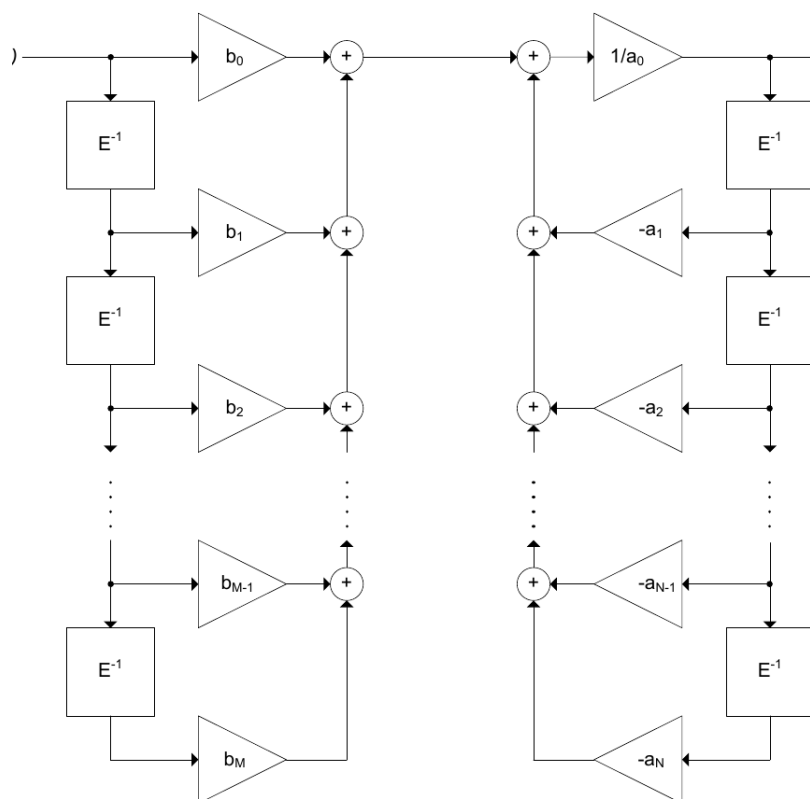


FIR — transponovana direktna forma.

IIR realizacije

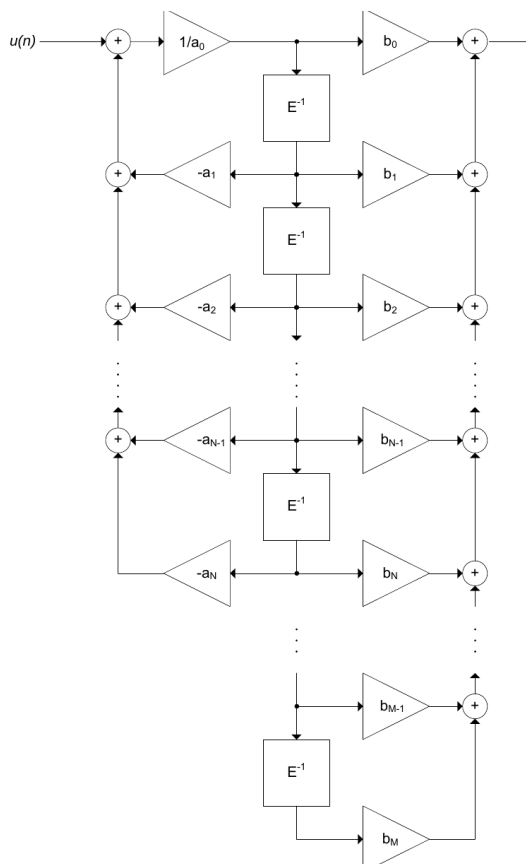
Kod IIR-a postoje i nule i polovi, pa strukture imaju povratne veze (koeficijenti $-a_k$). Sve četiri direktne forme su ekvivalentne; transponovana direktna II ima najkraći kritični put.

Direktna forma I — prvo deo po ulazu (brojilac), zatim deo po izlazu (imenilac). $M+N+1$ množača, $M+N$ sabirača, $M+N$ pomerača (najviše memorije).



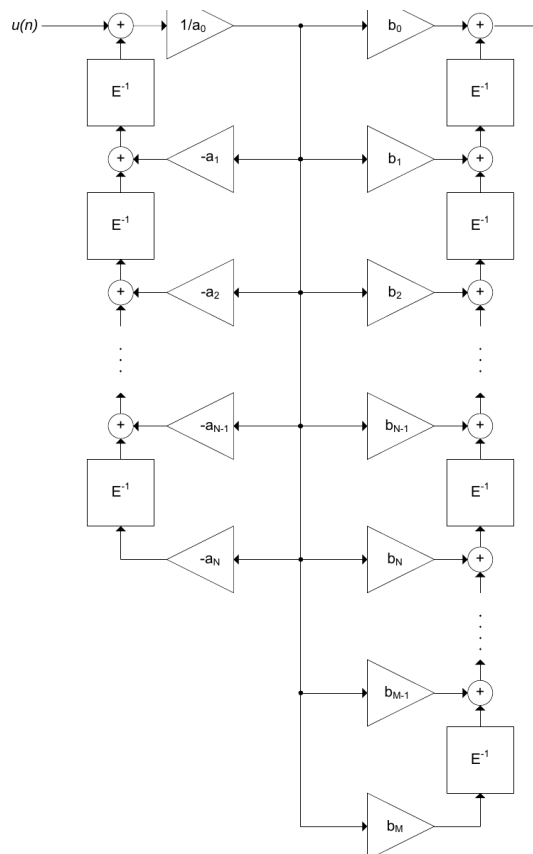
IIR — direktna realizacija I.

Direktna forma II — zamenom redosleda podsistema dele se isti pomerači → minimalno $\max(M, N)$ pomerača (kanonska po memoriji).



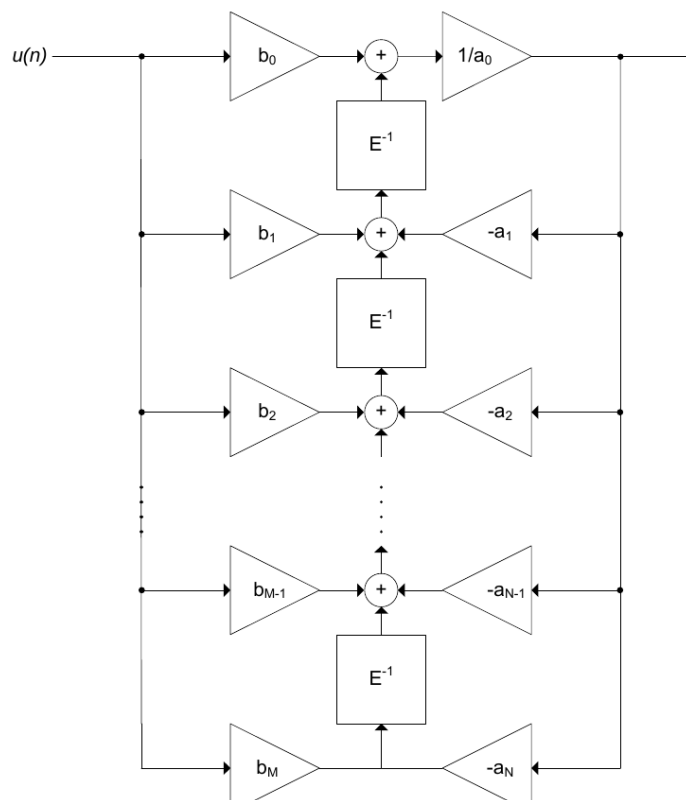
IIR — direktna realizacija II (slučaj $M > N$).

Transponovana direktna forma I — transpozicija direktne I.



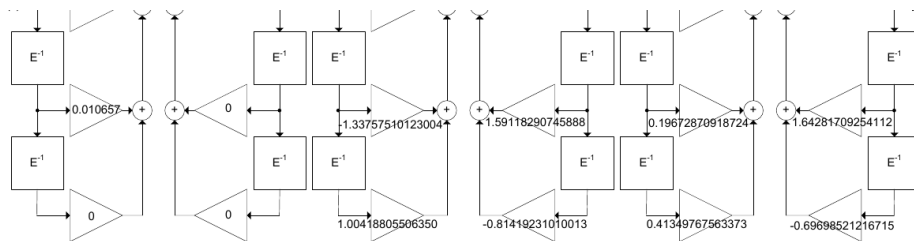
IIR — transponovana direktna realizacija I.

Transponovana direktna forma II — najkraći najduži propagacioni put (na kritičnom putu samo jedan množak i jedan sabirač, uz normalizaciju $a_0 = 1$).



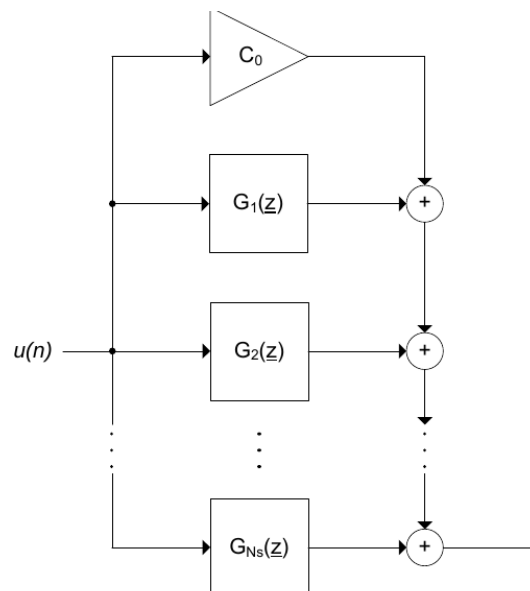
IIR — transponovana direktna realizacija II ($M = N$).

Kaskadna realizacija — $H(z)$ se rastavlja u proizvod sekcija 2. reda vezanih u niz; konjugovani polovi/nule se grupišu u istu sekciju (realni koeficijenti). Smanjuje osetljivost na kvantizaciju.



IIR — kaskadna realizacija (konkretan primer 4. reda, sekcije u direktnoj formi I).

Paralelna realizacija — $H(z)$ se razvija u parcijalne razlomke: konstanta C_0 (i eventualni FIR deo) plus sekcije 2. reda vezane **paralelno**, koje se sabiraju na izlazu.



IIR — paralelna realizacija (opšti oblik: $C_0 +$ paralelne sekcije).

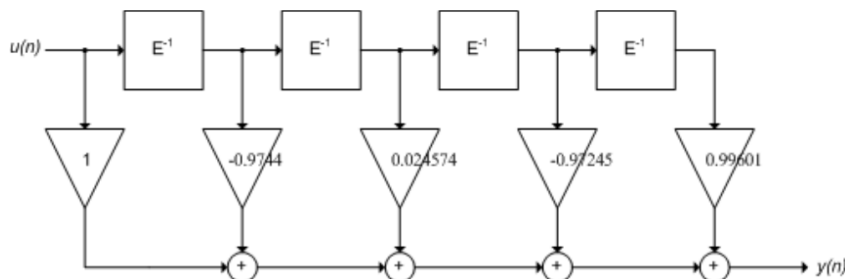
i Čelije 2. reda (za kaskadu i paralelu)

Same sekcije 2. reda koje se ulančavaju mogu biti u 4 varijante: direktna I, direktna II, transponovana direktna I i transponovana direktna II — kao kod FIR ćelije prikazane uz Ispit 2.

Dodatak — pomoćne slike i kontrolna lista

Numerički primer direktne FIR realizacije 4. reda (za RUI $y(n)=u(n)-0.9744u(n-1)+0.024574u(n-2)-0.97245u(n-3)+0.99601u(n-4)$) — pokazuje kako se konkretni koeficijenti upisuju u množače:

forme. Sledeća slika prikazuje blok dijagram realizovanog sistema.



Slika 5.4. Direktna realizacija FIR sistema četvrtog reda

Direktna realizacija FIR sistema 4. reda sa upisanim koeficijentima.

SIMULINK — kontrolna lista (da izlaz ne ispadne sve nule)

1. Ulaz: $x_sim = [n/fs; x].'$ — **vreme u sekundama**, ne indeks n .
2. From Workspace: Sample time $1/fs$, Output after final value = **Holding final value**.
3. To Workspace: **Save format = Array**.
4. Stop time = $(length(n)-1)/fs$.
5. Solver (Ctrl+E): Fixed-step, discrete (no continuous states), korak $1/fs$.
6. Provera tačnosti: $\max(abs(y - y_sim.')) \approx 10^{-15}$.

Korisne MATLAB komande

- `fir1(N,Wn,'low'/'high',hann(N+1))` — prozorska metoda ($Wn = fc/(fs/2)$).
- `firpmord + firpm` — Parks-McClellan.
- `filter(b,a,u)` — odziv (referentni rezultat).
- `freqz(b,a)` — karakteristike; `zplane(b,a)` — polovi i nule.
- `roots(b)`, `cplxpair`, `poly` — faktorizacija u sekcije 2. reda.
- `quantizer('fixed',roundmode,overflowmode,[w f]) + quantize(q,b)` — kvantizacija.

Srećno na ispitu! Strukture su matematički ekvivalentne — ako SIMULINK izlaz ne prati funkciju filter, greška je skoro uvek u pripremi ulaza (vremenska osa) ili u podešavanju Solver-a.