

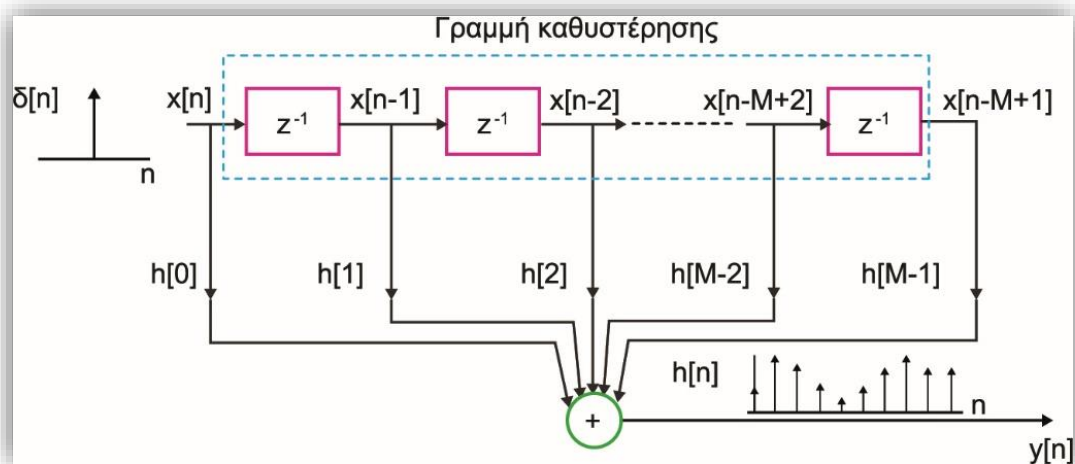


Πανεπιστήμιο Πελοποννήσου
Τμήμα Ηλεκτρολόγων Μηχανικών
και Μηχανικών Υπολογιστών

ΨΗΦΙΑΚΗ ΕΠΕΞΕΡΓΑΣΙΑ ΣΗΜΑΤΩΝ

ΕΡΓΑΣΤΗΡΙΑΚΟΣ ΟΔΗΓΟΣ

Εργαστηριακή Άσκηση 6 «Σχεδίαση Ψηφιακών Φίλτρων»



Παρασκευάς Μιχάλης
Καθηγητής

Φεβρουάριος 2024

Έκδοση: 5.0

Πίνακας περιεχομένων

ΜΕΡΟΣ Α' – Σχεδίαση ψηφιακών φίλτρων FIR	4
1. Στόχος της Εργαστηριακής Άσκησης	4
2. Προαπαιτούμενα	4
3. Μέθοδοι Σχεδίασης Φίλτρων FIR στο Matlab	4
Μέθοδος Παραθύρων	4
Μέθοδος Βέλτιστης Σχεδίασης (Ισοκυματική Μέθοδος)	6
4. Δομές υλοποίησης φίλτρων FIR	10
5. Άλυτες Ασκήσεις	11
ΜΕΡΟΣ Β' – Σχεδίαση ψηφιακών φίλτρων IIR	13
1. Σκοπός της Άσκησης	13
2. Προαπαιτούμενα	13
3. Ψηφιακά φίλτρα άπειρης κρουστικής απόκρισης (IIR)	13
Βαθυπερατό αναλογικό φίλτρο Butterworth	13
Βαθυπερατό αναλογικό φίλτρο Chebyshev τύπου I και II	13
Βαθυπερατό αναλογικό ελλειπτικό φίλτρο	14
Μέθοδος αμετάβλητης κρουστικής απόκρισης	14
Μέθοδος διγραμμικού μετασχηματισμού	15
4. Άλυτες ασκήσεις	18

ΜΕΡΟΣ Α' – Σχεδίαση ψηφιακών φίλτρων FIR

1. Στόχος της Εργαστηριακής Άσκησης

Στόχος της εργαστηριακής άσκησης είναι η εξοικείωση με το σχεδιασμό Φίλτρων FIR διακριτού χρόνου (Ψηφιακών Φίλτρων) στο Matlab.

2. Προαπαιτούμενα

Για την επιτυχή υλοποίηση της άσκησης είναι απαραίτητη η μελέτη της ενότητας 9 της θεωρίας.

3. Μέθοδοι Σχεδίασης Φίλτρων FIR στο Matlab

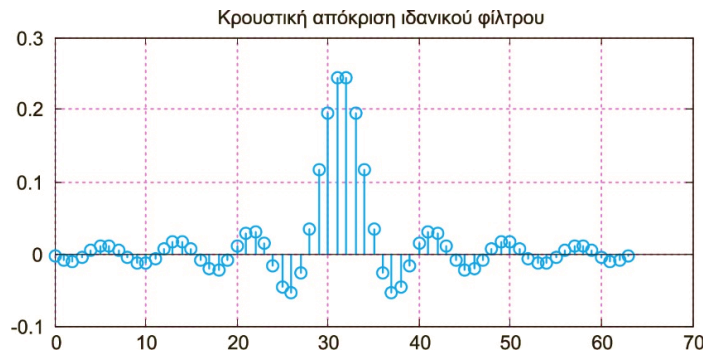
Μέθοδος Παραθύρων

Παράδειγμα 1 – Σχεδιασμός φίλτρου FIR με τη μέθοδο παραθύρων

Να σχεδιαστεί στο Matlab ένα βαθυπερατό φίλτρο με συχνότητα αποκοπής $\omega_c = 0.25\pi$ και μήκος $N = 64$ συντελεστές, για όλα τα είδη των παραπάνω παραθύρων.

Απάντηση: Αρχικά θα υπολογίσουμε την κρουστική απόκριση του ιδανικού βαθυπερατού φίλτρου (LPF) με αντίστροφο DTFT στην απόκριση συχνότητας $H_d(e^{j\omega})$, η οποία για την περίπτωση του ιδανικού βαθυπερατού φίλτρου δίνεται από τη σχέση 3. Λαμβάνουμε:

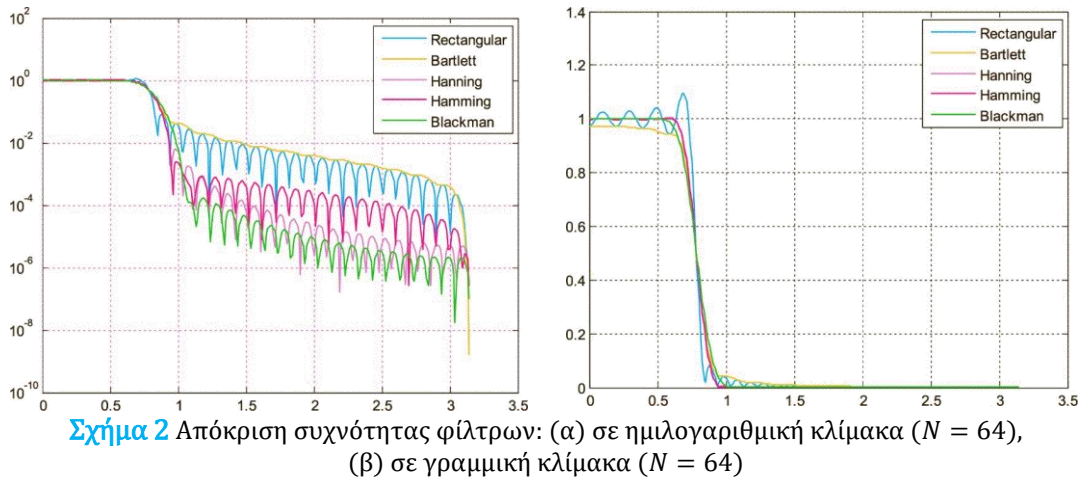
$$h_d[n] = \frac{1}{2\pi} \int_{-\pi}^{\pi} H_d(e^{j\omega}) e^{j\omega n} d\omega = \frac{1}{2\pi} \int_{-\omega_c}^{\omega_c} e^{j\omega n} d\omega = \frac{\sin(\omega_c n)}{\pi n}$$



Σχήμα 1 Κρουστική απόκριση ιδανικού βαθυπερατού φίλτρου (N=64)

Η συνάρτηση `imp_resp()` [Ingle, 2012] επιστρέφει την κρουστική απόκριση (σχήμα 3) του ιδανικού βαθυπερατού φίλτρου με βάση την παραπάνω σχέση.

```
function [hd, n] = imp_resp(wc,N) % wc συχνότητα αποκοπής
a = (N-1)/2; m = [0:1:(N-1)]; n = m - a + eps;
hd = sin(wc*n)./(pi*n); end
```



Ακολουθεί ο κώδικας Matlab που υπολογίζει τα παράθυρα Rectangular, Bartlett, Hanning, Hamming, Blackman χρησιμοποιώντας τις κατάλληλες συναρτήσεις, κατόπιν υπολογίζει τη κρουστική απόκριση του πραγματικού φίλτρου και στη συνέχεια την απόκριση συχνότητας του πραγματικού φίλτρου. Τέλος, σχεδιάζει τις αποκρίσεις συχνότητας των φίλτρων σε ημιλογαριθμική (σχ. 4α) και γραμμική κλίμακα (σχ. 4β).

```

N = 64;
wc = 0.25*pi           % Τάξη φίλτρου και συχνότητα αποκοπής
n = 0:1:N-1;          % Κλίμακα χρόνου
w = 0:0.01:pi;        % Κλίμακα συχνότητας
% Δημιουργία κρουστικής απόκρισης ιδανικού φίλτρου
hd = imp_resp(wc,N);
% Δημιουργία παραθύρων, κρουστικών αποκρίσεων και αποκρίσεων συχνότητας
w1 = boxcar(N)'; h1 = hd.*w1; H1 = freqz(h1,1,w); % Ορθογώνιο παράθυρο
w2 = bartlett(N)'; h2 = hd.*w2; H2 = freqz(h2,1,w); % Παράθυρο Bartlett
w3 = hanning(N)'; h3 = hd.*w3; H3 = freqz(h3,1,w); % Παράθυρο Hanning
w4 = hamming(N)'; h4 = hd.*w4; H4 = freqz(h4,1,w); % Παράθυρο Hamming
w5 = blackman(N)'; h5 = hd.*w5; H5 = freqz(h5,1,w); % Παράθυρο Blackman
% Σχεδιασμός αποκρίσεων συχνότητας σε ημιλογαριθμική κλίμακα
figure(1); semilogy(w,abs(H1),'c'); hold on; grid on;
semilogy(w,abs(H2),'b'); semilogy(w,abs(H3),'m');
semilogy(w,abs(H4),'r'); semilogy(w,abs(H5),'g');
legend('Rectangular','Bartlett','Hanning','Hamming','Blackman');
hold off
% Σχεδιασμός αποκρίσεων συχνότητας σε γραμμική κλίμακα
figure(2); plot(w,abs(H1),'c'); hold on; grid on; plot(w,abs(H2),'b');
plot(w,abs(H3),'m'); plot(w,abs(H4),'r'); plot(w,abs(H5),'g')
legend('Rectangular','Bartlett','Hanning','Hamming','Blackman');
hold off
    
```

Για τον προσδιορισμό της τάξης ενός FIR φίλτρου που σχεδιάζεται με τη μέθοδο παραθύρων με χρήση του παράθυρου Kaiser, το Matlab προσφέρει τη συνάρτηση `kaiserord()`.

Άσκηση: Επαναλάβετε την εκτέλεση του παραπάνω κώδικα αυξάνοντας την τάξη του φίλτρου. Τι παρατηρείτε;

Μέθοδος Βέλτιστης Σχεδίασης (Ισοκυματική Μέθοδος)

Η απαίτηση για σταθερή και ελεγχόμενη κυμάτωση στις ζώνες διέλευσης και αποκοπής οδηγεί στον ορισμό του **ισοκυματικού φίλτρου** (equiripple filter), το οποίο θεωρείται ως το βέλτιστο FIR φίλτρο, με την έννοια ότι επιτυγχάνει τις καλύτερες δυνατές προδιαγραφές στην απόκριση συχνότητας με το μικρότερο πλήθος συντελεστών.

Για την εύρεση της βέλτιστης λύσης εφαρμόζεται επαναληπτική διαδικασία, γνωστή ως **αλγόριθμος Parks – McClellan** (αλγόριθμος εναλλαγής Remez), που στο Matlab υλοποιείται με τη συνάρτηση `firpm()`. Για τον προσδιορισμό της τάξης ενός FIR φίλτρου που σχεδιάζεται με τη μέθοδο Parks – McClellan, το Matlab προσφέρει τη συνάρτηση `firpmord()`.

Παράδειγμα 2 – Σχεδιασμός φίλτρου FIR με τη μέθοδο βέλτιστης σχεδίασης

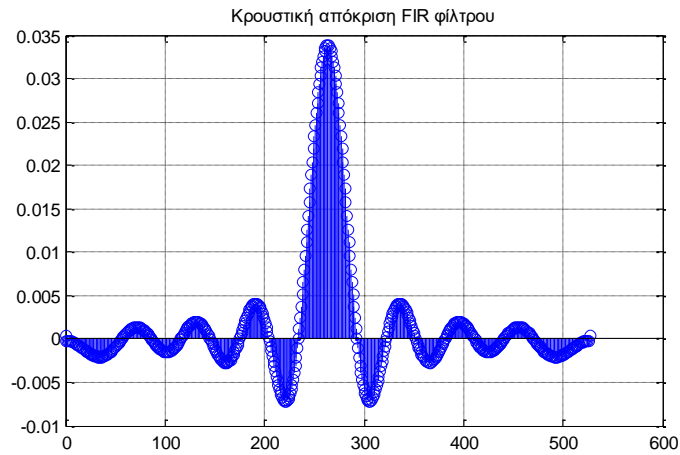
Να σχεδιαστεί στο Matlab με χρήση του αλγόριθμου Parks-McClellan ένα βαθυπερατό FIR φίλτρο με προδιαγραφές: $f_p = 700 \text{ Hz}$, $d_p = 0.1$, $f_s = 850 \text{ Hz}$, $d_s = 0.01$. Να σχεδιαστεί η κρουστική απόκριση, η απόκριση συχνότητας και το διάγραμμα πόλων – μηδενικών του φίλτρου. Θεωρήστε συχνότητα δειγματοληψίας $F_s = 44100 \text{ Hz}$.

Απάντηση:

```

Fs = 44100; % Συχνότητα δειγματοληψίας
fp = 700; % Συχνότητα αποκοπής ζώνης διέλευσης
fs = 850; % Συχνότητα αποκοπής ζώνης αποκοπής
dp = 0.1; % Απόκλιση στη ζώνη διέλευσης
ds = 0.001; % Απόκλιση στη ζώνη αποκοπής
% Προδιαγραφές φίλτρου
F = [fp fs]; A = [1 0]; Deviation = [dp ds];
% Υπολογισμός βέλτιστης τάξης
[N, Fo, Ao, Weights] = firpmord(F, A, Deviation, Fs)
% Υπολογισμός h[n] με Parks-McClellan
h = firpm(N, Fo, Ao, Weights);
% Σχεδιασμός κρουστικής απόκρισης FIR φίλτρου
n = 0:1:N; figure(1); stem(n, h); grid on;
title('Κρουστική απόκριση FIR φίλτρου')
% Σχεδιασμός απόκρισης συχνότητας FIR φίλτρου
NFFT = 2^11; figure(2);
freqz(h, 1, NFFT, Fs); axis([0, Fs/2 -80 10])
title('Απόκριση συχνότητας FIR φίλτρου')

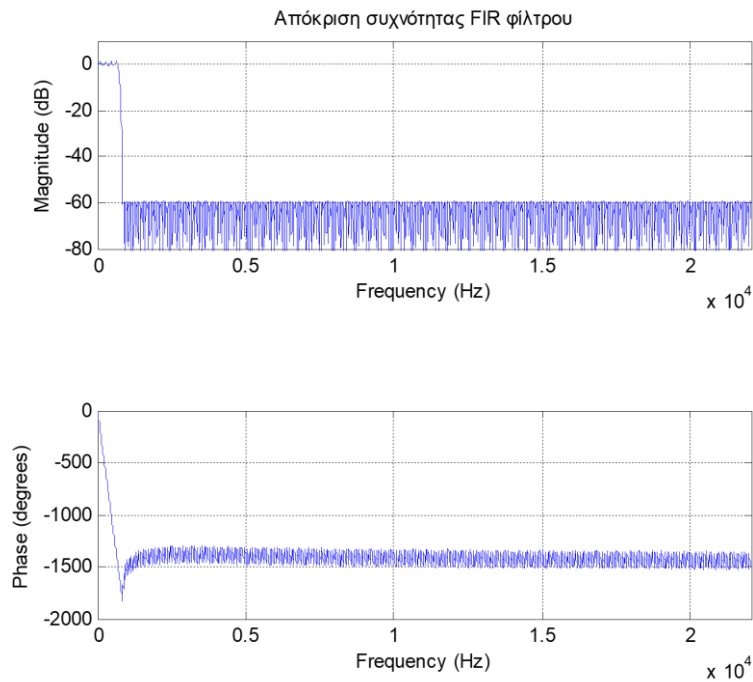
```



Σχήμα 3 Κρουστική απόκριση FIR φίλτρου

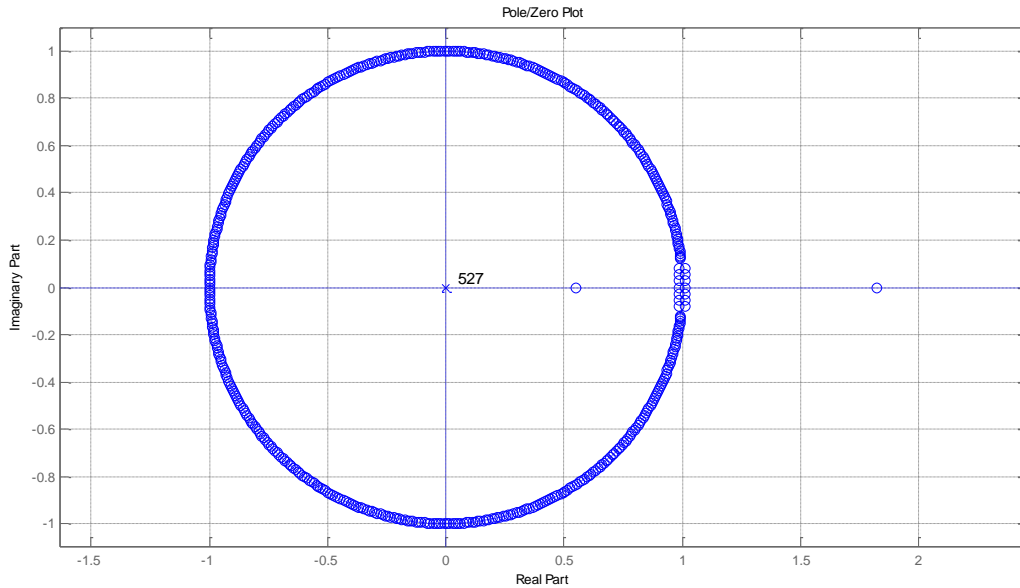
Αποτελέσματα:

N =	Fo =	Ao =	Weights =
527	0	1	1
	0.0317	1	100
	0.0385	0	
	1.0000	0	



Σχήμα 4 Απόκριση συχνότητας FIR φίλτρου

Με την εντολή `fvtool(h,1)` ενεργοποιούμε το Filter Visualization Tool του Matlab, ένα γραφικό interface που επιτρέπει την ανάλυση ψηφιακών φίλτρων. Από την επιλογή Analysis - Pole/Zero plot λαμβάνουμε το ακόλουθο διάγραμμα.



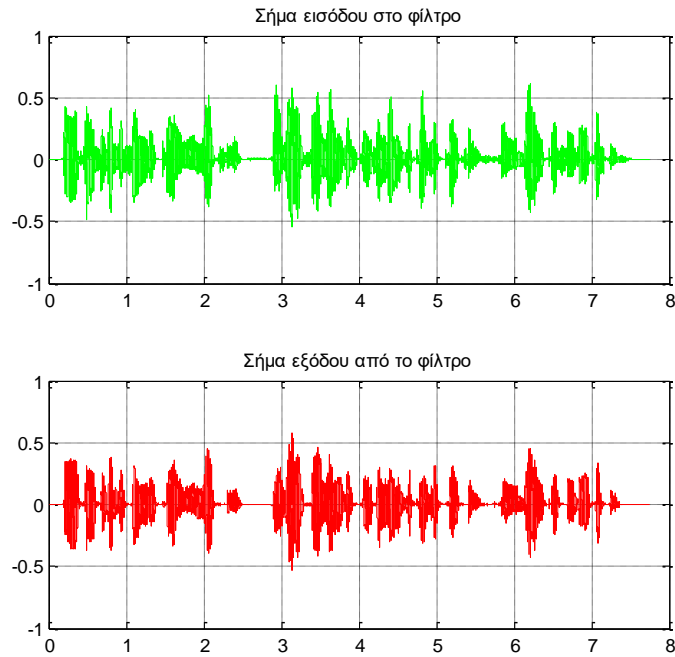
Σχήμα 5 Διάγραμμα πόλων – μηδενικών FIR φίλτρου

Άσκηση: Επαναλάβετε την εκτέλεση του παραπάνω κώδικα αλλάζοντας την τιμή της απόκλισης πλάτους στη ζώνη αποκοπής σε $d_s = 0.1$. Τι παρατηρείτε ως προς την τάξη, την κρουστική απόκριση και την απόκριση συχνότητας του φίλτρου;

Παράδειγμα 3

Στον κώδικα του προηγούμενου παραδείγματος προσθέστε τις ακόλουθες εντολές:

```
% Ανάγνωση αρχείου ήχου και αποθήκευση ενός καναλιού
[signal, FS, Nbits] = wavread('speech.WAV');
x = signal(:,1);
% Υπολογισμός εξόδου FIR φίλτρου μέσω συνέλιξης
y = conv(h, x);
% Σχεδιασμός κυματομορφών εισόδου και εξόδου
L = length(x); n = [0:L-1]/FS;
figure(3); subplot(211); plot(n, x(1:L), 'g'); grid on;
title('Σήμα εισόδου στο φίλτρο')
subplot(212); plot(n, y(1:L), 'r'); grid on;
title('Σήμα εξόδου από το φίλτρο')
% Υπολογισμός φασμάτων πλάτους εισόδου και εξόδου
M = 2^15;
X = 20*log10(abs(fft(x(1:M), M)));
Y = 20*log10(abs(fft(y(1:M), M)));
% Κανονικοποιημένη κλίμακα συχνοτήτων
f = [0:(M/2-1)]*(FS/2)/(M/2);
% Σχεδίαση φασμάτων πλάτους εισόδου και εξόδου
figure(4); subplot(211); plot(f, X(1:M/2), 'g' ); axis([0, FS/2 -60 60])
grid on; title('Φάσμα πλάτους σήματος εισόδου στο φίλτρο')
subplot(212); plot(f, Y(1:M/2), 'r' ); axis([0, FS/2 -60 60])
grid on; title('Φάσμα πλάτους σήματος εξόδου από το φίλτρο')
```

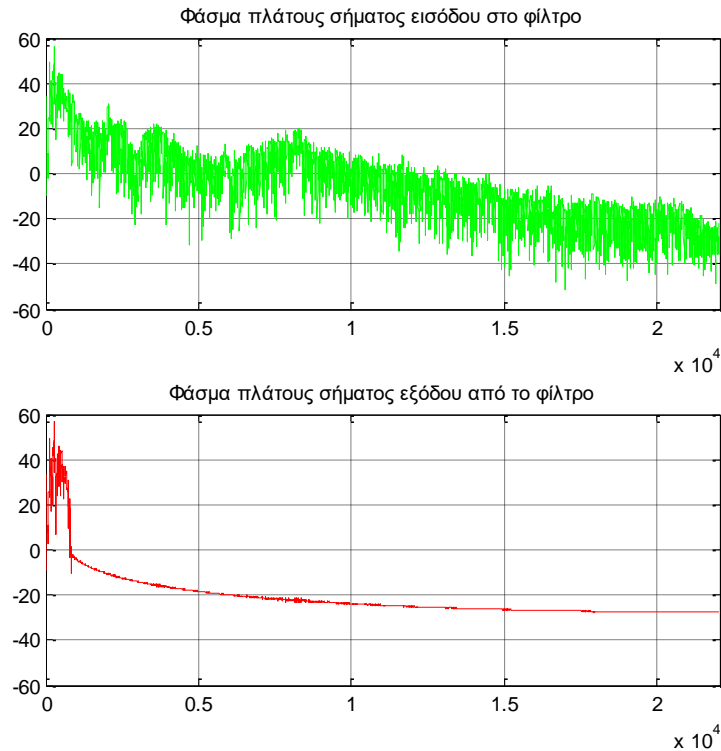
Σχήμα 6 Κυματομορφές: (α) Αρχικού σήματος, (β) Φιλτραρισμένου σήματος

Σχολιασμός:

- Υπάρχουν σημαντικές διαφορές μεταξύ των σημάτων του σχήματος 6;
- Σχολιάστε σε συνδυασμό τα σχήματα 6 και 7.

Προκειμένου να αναπαράγετε από την κάρτα ήχου του υπολογιστή σας το αρχικό καθώς και το φιλτραρισμένο σήμα, προσθέστε στο προηγούμενο πρόγραμμα τις ακόλουθες εντολές:

```
% Ακρόαση σήματος εισόδου
sound(x, FS)
pause % Προσωρινή παύση
% Ακρόαση σήματος εξόδου (φιλτραρισμένο)
sound(y, FS)
```



Σχήμα 7 Φάσματα πλάτους: (α) Αρχικού σήματος, (β) Φιλτραρισμένου σήματος

Παρατήρηση: Για τον ταχύτερο υπολογισμό της συνέλιξης μπορείτε να χρησιμοποιήσετε τη συνάρτηση `FFT_conv(x, h)`. Η συνάρτηση αυτή υπολογίζει τη συνέλιξη των σημάτων A και B μέσω FFT. Συγκεκριμένα, αντικαταστήστε την εντολή `y = conv(x, h)`; με την εντολή `y = FFT_conv(x, h)`;

Σχολιάστε το αποτέλεσμα σχετικά με την ταχύτητα εκτέλεσης.

% Υλοποίηση συνέλιξης μεταξύ των σημάτων x1 και x2 με τη χρήση FFT

```
function y = FFT_conv(x1, x2)
M = length(x1) + length(x2) - 1;
N = round((log10(M)/log10(2.0))+2.2204460492503131e-016));
if 2.0^N < M, N = N + 1; end;
NFFT = 2.0^N;
X1 = fft(x1, NFFT); clear x1;
X2 = fft(x2, NFFT); clear x2;
Y = X1.*X2; clear X1; clear X2;
y = real(ifft(Y)); clear Y;
y = y(1:M);
```

4. Δομές υλοποίησης φίλτρων FIR

Υπάρχουν τέσσερις διαφορετικές δομές με τις οποίες μπορεί να υλοποιηθεί ένα FIR σύστημα:

- **Ευθεία Μορφή:** Η μορφή αυτή προκύπτει από την άμεση εφαρμογή της ΓΕΔΣΣ σε

διάγραμμα βαθμίδων.

- **Μορφή Καταρράκτη:** Στη μορφή αυτή η συνάρτηση μεταφοράς παραγοντοποιείται σε μικρότερα τμήματα δεύτερης τάξης. Όπως και στα IIR, κάθε τμήμα υλοποιείται σε ευθεία μορφή και το συνολικό σύστημα προκύπτει από τη συνένωση των επιμέρους τμημάτων.
- **Γραμμικής φάσης:** Όταν η κρουστική απόκριση ικανοποιεί κάποιες ιδιότητες συμμετρίας τότε η φάση του συστήματος είναι γραμμική. Λόγω της συμμετρίας μειώνεται το πλήθος των απαιτούμενων υπολογισμών.
- **Δειγματοληψίας συχνότητας:** Βασίζεται στον υπολογισμό DFT $H[k]$ της κρουστικής απόκρισης $h[n]$.

Παράδειγμα 4 – Υπολογισμός συντελεστών μορφής καταρράκτη

Να υπολογιστούν στο Matlab οι συντελεστές της μορφής καταρράκτη του FIR φίλτρου με ΓΕΔΣΣ: $y[n] = x[n] + 2x[n - 2] - 0.5x[n - 4]$.

Απάντηση:

```
% FIR: y[n] = x[n] + 2x[n-2] - 0.5x[n-4]
a = 1; b = [1, 0, 2, 0, -0.5]
[sos, g] = tf2sos(b,a);
display(g); display(sos);
```

Αποτελέσματα:

```
g =
    1

sos =
    1.0000    0.0000   -0.2247    1.0000         0         0
    1.0000         0    2.2247    1.0000         0         0
```

5. Άλυτες Ασκήσεις

1. Να σχεδιαστεί στο Matlab με χρήση του αλγόριθμου Parks-McClellan ένα ζωνοπερατό FIR φίλτρο με προδιαγραφές:

$$\begin{aligned}\omega_{1p} &= 0.2\pi, & \omega_{1s} &= 0.25\pi \\ \omega_{2p} &= 0.7\pi, & \omega_{2s} &= 0.75\pi \\ R_p &= 1 \text{ dB}, & A_s &= 50 \text{ dB}\end{aligned}$$

Να σχεδιαστεί η κρουστική απόκριση, η απόκριση συχνότητας και το διάγραμμα πόλων – μηδενικών του φίλτρου. Η συχνότητα δειγματοληψίας είναι $F_s = 10000 \text{ Hz}$.

2. Το σήμα $x(t) = \cos(2000\pi t) + \cos(6000\pi t)$ δειγματοληπτείται με συχνότητα $F_s = 10000 \text{ Hz}$ και διέρχεται από ένα βαθυπερατό φίλτρο με προδιαγραφές:

$$f_p = 1500 \text{ Hz}, f_s = 1700 \text{ Hz}, d_p = 0.001, \quad d_s = 0.1$$

Να υπολογιστεί η κρουστική απόκριση και η τάξη του ισοκυματικού φίλτρου και να σχεδιαστεί η απόκριση συχνότητας του φίλτρου καθώς και τα φάσματα των σημάτων στην είσοδο και στην έξοδο του φίλτρου.

3. Να βρεθούν στο Matlab οι συντελεστές ανάκλασης του FIR φίλτρου πλέγματος τέταρτης τάξης, το οποίο σε ευθεία μορφή περιγράφεται από τη ΓΕΔΣΣ:

$$y[n] = 2x[n] + \frac{11}{5}x[n-1] - \frac{2}{3}x[n-3] + \frac{1}{5}x[n-4]$$

4. Να σχεδιαστεί στο Matlab με χρήση του αλγόριθμου Parks-McClellan ένα υψιπερατό FIR φίλτρο με προδιαγραφές: $f_p = 2200 \text{ Hz}$, $d_p = 0.01$, $f_s = 2000 \text{ Hz}$, $d_s = 0.001$. Να σχεδιαστεί η κρουστική απόκριση, η απόκριση συχνότητας και το διάγραμμα πόλων - μηδενικών του φίλτρου. Θεωρήστε συχνότητα δειγματοληψίας $F_s = 44100 \text{ Hz}$.

Χρησιμοποιήστε σαν σήμα εισόδου στο φίλτρο το αρχείο "music.WAV". Σχολιάστε τα αποτελέσματα.

Αν ο υπολογισμός της συνέλιξης με την ενσωματωμένη συνάρτηση `conv()` καθυστερεί, δοκιμάστε να υπολογίσετε τη συνέλιξη μέσω του FFT με τη συνάρτηση `FFT_conv()`.

ΜΕΡΟΣ Β' – Σχεδίαση ψηφιακών φίλτρων IIR

1. Σκοπός της Άσκησης

Σκοπός της άσκησης είναι η εξοικείωση των φοιτητών με τον σχεδιασμό IIR Ψηφιακών Φίλτρων.

2. Προαπαιτούμενα

Για την επιτυχή υλοποίηση της άσκησης είναι απαραίτητη η μελέτη της ενότητας 10 της θεωρίας.

3. Ψηφιακά φίλτρα άπειρης κρουστικής απόκρισης (IIR)

Βαθυπερατό αναλογικό φίλτρο Butterworth

Για τη σχεδίαση βαθυπερατού αναλογικού φίλτρου Butterworth το Matlab προσφέρει τις συναρτήσεις:

- **[N, Wn] = buttord(Wp, Ws, Rp, As):** Επιστρέφει τη μικρότερη τάξη N ενός αναλογικού φίλτρου Butterworth που ικανοποιεί τις προδιαγραφές. Wn είναι η συχνότητα αποκοπής Ω_c σε κανονικοποιημένη κλίμακα, δηλαδή $0 < W_n < 1$, όπου το 1 αντιστοιχεί στο ήμισυ της συχνότητας δειγματοληψίας.
- **[b, a] = butter(N, Wn):** Επιστρέφει τους συντελεστές $a[n], b[m]$ της συνάρτησης μεταφοράς, τάξης N και συχνότητας αποκοπής Wn. Εναλλακτικά συντάσσεται **[z,p,k] = butter(N, Wn)** οπότε επιστρέφει τους πόλους, τα μηδενικά και τον συντελεστή κέρδους της συνάρτησης μεταφοράς.

Για τη σχεδίαση πρότυπου βαθυπερατού αναλογικού φίλτρου Butterworth το Matlab προσφέρει τη συνάρτηση:

- **[z,p,k] = buttap(N):** Επιστρέφει τους πόλους, τα μηδενικά και το κέρδος ενός κανονικοποιημένου πρότυπου αναλογικού φίλτρου Butterworth τάξης N. Το προκύπτον φίλτρο έχει N πόλους περί του μοναδιαίου κύκλου στο αριστερό μιγαδικό ημιεπίπεδο και δεν έχει μηδενικά.

Βαθυπερατό αναλογικό φίλτρο Chebyshev τύπου I και II

Για τη σχεδίαση βαθυπερατού αναλογικού φίλτρου Chebyshev τύπου I το Matlab προσφέρει τις συναρτήσεις:

- **[N, Wn] = cheb1ord(Wp, Ws, Rp, Rs):** Επιστρέφει τη μικρότερη τάξη N ενός φίλτρου Chebyshev τύπου I που ικανοποιεί τις προδιαγραφές του φίλτρου. Wn είναι η συχνότητα αποκοπής Ω_c σε κανονικοποιημένη κλίμακα, δηλαδή $0 < W_n < 1$, όπου το 1 αντιστοιχεί σε π rad/sample, δηλαδή στο ήμισυ της συχνότητας δειγματοληψίας.
- **[b, a] = cheby1(N, R, Wn):** Επιστρέφει τους συντελεστές $a[n], b[m]$ της συνάρτησης μεταφοράς τάξης N για συχνότητα αποκοπής Wn και κυμάτωση R στη ζώνη αποκοπής. Εναλλακτικά, συντάσσεται **[z,p,k] = cheby1(N, R, Wn)** οπότε επιστρέφει τους πόλους, τα μηδενικά και τον συντελεστή κέρδους της συνάρτησης μεταφοράς.

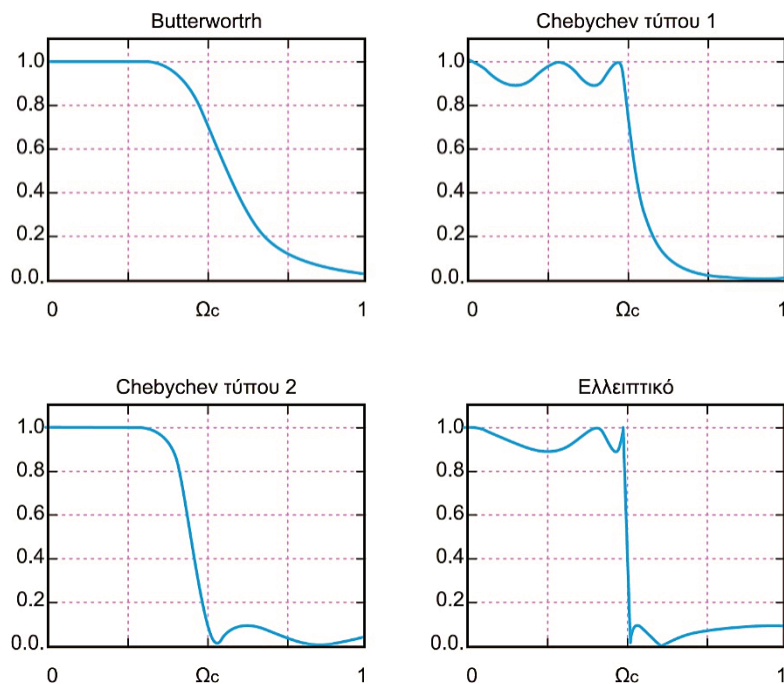
Για τη σχεδίαση πρότυπου βαθυπερατού αναλογικού φίλτρου Chebyshev τύπου I το Matlab προσφέρει τη συνάρτηση:

- $[z,p,k] = \text{cheb1ap}(N, R_p)$: Επιστρέφει τους πόλους, τα μηδενικά και το κέρδος ενός κανονικοποιημένου πρότυπου αναλογικού φίλτρου Chebyshev τύπου I με τάξη N και κυμάτωση R_p στη ζώνη διέλευσης.

Αντίστοιχη σύνταξη και χρήση έχουν οι συναρτήσεις `cheb2ord()`, `cheby2()` και `cheb2ap()` για το φίλτρο Chebyshev τύπου II.

Βαθυπερατό αναλογικό ελλειπτικό φίλτρο

Για τη σχεδίαση βαθυπερατού αναλογικού ελλειπτικού φίλτρου το Matlab προσφέρει τη συνάρτηση `ellip()` και τη συνάρτηση `ellipord()`, για τον υπολογισμό της βέλτιστης τάξης, ενώ για τη σχεδίαση πρότυπου βαθυπερατού αναλογικού ελλειπτικού φίλτρου προσφέρει τη συνάρτηση `ellipap()`.



Σχήμα 1 Αποκρίσεις συχνότητας πρότυπων αναλογικών βαθυπερατών φίλτρων Butterworth, Chebyshev I, Chebyshev II και Ελλειπτικό

Μέθοδος αμετάβλητης κρουστικής απόκρισης

Η μετατροπή του πρότυπου βαθυπερατού αναλογικού φίλτρου σε ψηφιακό στο Matlab με τη μέθοδο της αμετάβλητης κρουστικής απόκρισης γίνεται με τη συνάρτηση $[b_z, a_z] = \text{impinvar}(b, a, F_d)$, όπου b, a είναι οι συντελεστές του αναλογικού φίλτρου, F_d είναι η συχνότητας δειγματοληψίας της κρουστικής απόκρισης και $[b_z, a_z]$ είναι οι συντελεστές του ψηφιακού φίλτρου.

Σε αρκετές περιπτώσεις, η μέθοδος της αμετάβλητης κρουστικής αποτυγχάνει να υλοποιήσει με ακρίβεια το ψηφιακό φίλτρο επειδή η απεικόνιση συχνότητας που υλοποιεί, αντιστοιχίζει πολλά σημεία της αναλογικής συχνότητας σε ένα σημείο της ψηφιακής συχνότητας. Αυτό οφείλεται στην ανεπαρκή συχνότητα δειγματοληψίας. Για τον ίδιο λόγο η μέθοδος αυτή δεν μπορεί να σχεδιάσει υψιπερατά φίλτρα.

Μέθοδος διγραμμικού μετασχηματισμού

Η μετατροπή του πρότυπου βαθυπερατού αναλογικού φίλτρου σε ψηφιακό στο Matlab με τη μέθοδο του διγραμμικού μετασχηματισμού γίνεται με τη συνάρτηση `[bz, az] = bilinear(b, a, Fs)`, όπου **b**, **a** είναι οι συντελεστές του αναλογικού φίλτρου, **Fs** είναι η συχνότητα δειγματοληψίας και `[bz, az]` είναι οι συντελεστές του ψηφιακού φίλτρου.

Παράδειγμα 1 – Σχεδιασμός IIR φίλτρου με τη μέθοδο αμετάβλητης κρουστικής

Να σχεδιαστεί ένα βαθυπερατό ψηφιακό φίλτρο με τις ακόλουθες προδιαγραφές, χρησιμοποιώντας το πρότυπο αναλογικό φίλτρο Butterworth και τη μέθοδο της αμετάβλητης κρουστικής:

$$\omega_p = 0.3\pi, \quad R_p = 0.5 \text{ dB}$$

$$\omega_s = 0.45\pi, \quad A_s = 15 \text{ dB}$$

Απάντηση: Μετατρέπουμε τις προδιαγραφές του ψηφιακού φίλτρου στις αντίστοιχες προδιαγραφές του αναλογικού φίλτρου, χρησιμοποιώντας τη σχέση μετατροπής $\Omega = \omega/T_d$.

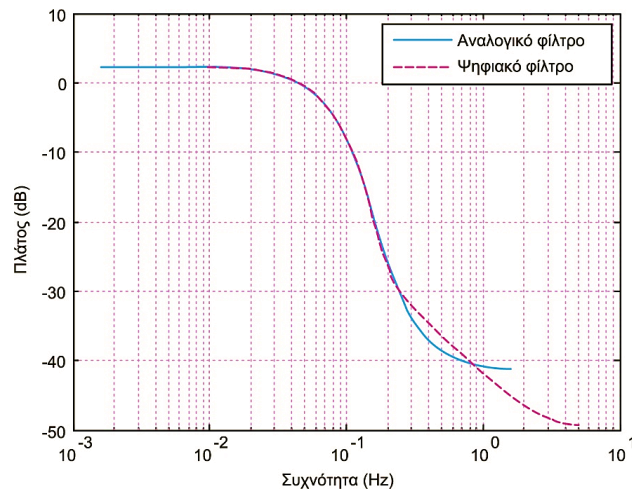
Βήμα 1: Θέτουμε τιμή $T_d = 3 \text{ sec}$, οπότε λαμβάνουμε:

$$\Omega_p = \frac{\omega_p}{3} = 0.1\pi \quad \text{και} \quad \Omega_s = \frac{\omega_s}{3} = 0.15\pi$$

Βήμα 2: Από τις προδιαγραφές Ω_p , Ω_s , R_p και A_s και από τη σχέση 10.β υπολογίζουμε την ελάχιστη τιμή της τάξης που ικανοποιεί τις προδιαγραφές. Κατόπιν υπολογίζουμε τη συνάρτηση μεταφοράς $H(s)$ του πρότυπου βαθυπερατού φίλτρου Butterworth.

```
% Προδιαγραφές ψηφιακού φίλτρου
wp = 0.3*pi; Rp = 0.5; ws = 0.45*pi; As = 15;
% Προδιαγραφές πρότυπου αναλογικού φίλτρου
Td = 3; Wp = wp/Td; Ws = ws/Td;
% Υπολογισμός πρότυπου αναλογικού φίλτρου Butterworth
[N, Wn] = buttord(Wp, Ws, Rp, As)
[b, a] = butter(N, Wn);
% Συνάρτηση μεταφοράς αναλογικού φίλτρου
[Hz, Wz] = freqs(b, a, 512);
% Περίοδος και συχνότητα δειγματοληψίας
Ts = 0.1; Fs = 1/Td;
% Μέθοδος αμετάβλητης κρουστικής απόκρισης
[bz, az] =impinvar(b, a, Fs);
% Συνάρτηση μεταφοράς ψηφιακού φίλτρου
[Hs, Ws] = freqz(bz, az, 512, Fs);
% Σχεδιασμός φασμάτων αναλογικού και ψηφιακού φίλτρου
figure(1); semilogx(Ws/(2*pi), 20*log10(abs(Hs)), 'LineWidth', 2);
hold on; semilogx(Wz, 20*log10(abs(Hz)), 'r--'); grid on;
xlabel('Συχνότητα (Hz)'), ylabel('Πλάτος (dB)');
legend('Αναλογικό φίλτρο', 'Ψηφιακό φίλτρο');
```

Η τάξη του φίλτρου υπολογίζεται σε $N=7$. Η απόκριση συχνότητας των φίλτρων δίνεται στο επόμενο σχήμα.



Σχήμα 2 Απόκριση πλάτους αναλογικού και ψηφιακού φίλτρου με βάση το πρότυπο αναλογικό φίλτρο Butterworth

Παράδειγμα 2 – Σχεδιασμός IIR φίλτρου με τη μέθοδο διγραμμικού μετασχηματισμού

Να σχεδιαστεί ένα βαθυπερατό ψηφιακό φίλτρο με τον διγραμμικό μετασχηματισμό και τα πρότυπα αναλογικά φίλτρα: (α) Butterworth, (β) Chebyshev I, (γ) Chebyshev II και (δ) Ελλειπτικό με τις ακόλουθες προδιαγραφές:

$$\begin{aligned} \omega_p &= 0.2\pi, & R_p &= 0.5 \text{ dB} \\ \omega_s &= 0.3\pi, & A_s &= 20 \text{ dB} \end{aligned}$$

Απάντηση: Γράφουμε το ακόλουθο πρόγραμμα Matlab:

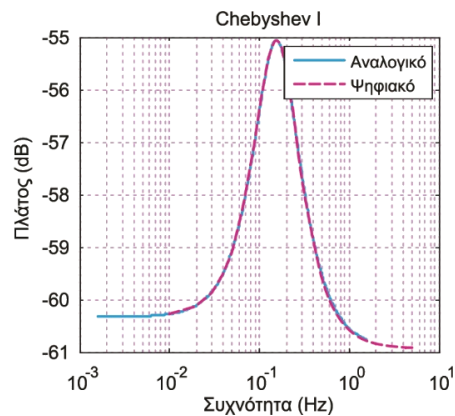
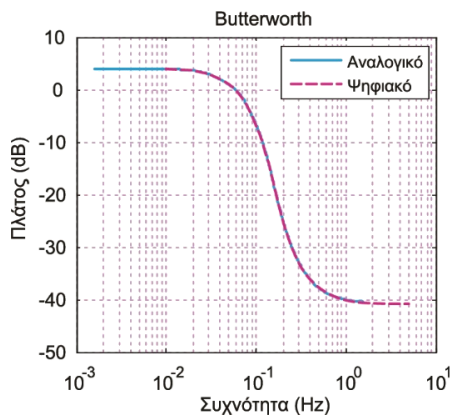
```
% Προδιαγραφές ψηφιακού φίλτρου
wp = 0.2*pi; Rp = 0.5; ws = 0.3*pi; As = 20; T = 2;
% Προδιαγραφές πρότυπου αναλογικού φίλτρου Butterworth
Wp = (2/T)*tan(wp/2); % Αναλογική συχνότητα διέλευσης
Ws = (2/T)*tan(ws/2); % Αναλογική συχνότητα αποκοπής
% Υπολογισμός πρότυπου αναλογικού φίλτρου Butterworth
[N1, Wn1] = buttord(Wp, Ws, Rp, As);
[b1, a1] = butter(N1, Wn1);
% Υπολογισμός πρότυπου αναλογικού φίλτρου Chebyshev I
[N2, Wn2] = cheb1ord(Wp, Ws, Rp, As);
[b2, a2] = cheby1(N2, As, Wn2);
% Υπολογισμός πρότυπου αναλογικού φίλτρου Chebyshev II
[N3, Wn3] = cheb2ord(Wp, Ws, Rp, As);
[b3, a3] = cheby2(N3, As, Wn3);
% Υπολογισμός πρότυπου ελλειπτικού αναλογικού φίλτρου
[N4, Wn4] = ellipord(Wp, Ws, Rp, As);
[b4, a4] = ellip(N4, Rp, As, Wn4);
% Συνάρτηση μεταφοράς αναλογικού φίλτρου
[Hs1, Ws1] = freqs(b1, a1, 512);
[Hs2, Ws2] = freqs(b2, a2, 512);
[Hs3, Ws3] = freqs(b3, a3, 512);
[Hs4, Ws4] = freqs(b4, a4, 512);
% Περίοδος και συχνότητα δειγματοληψίας
```

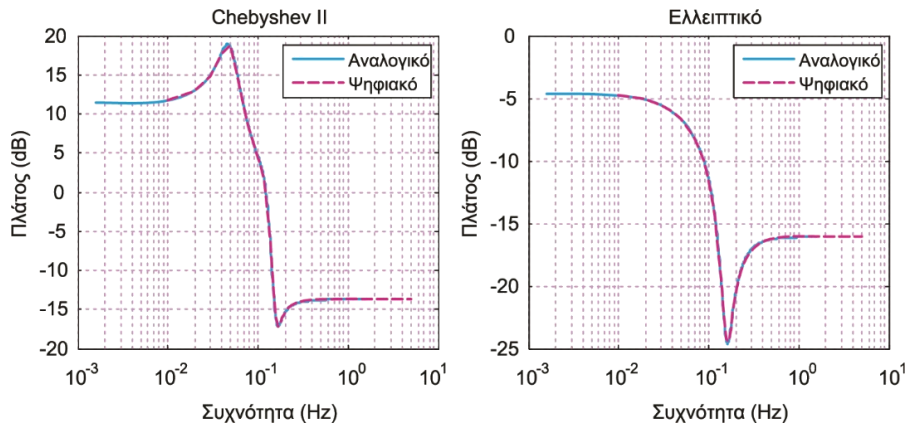


```

Ts = 0.1; Fs = 1/Ts;
% Εφαρμογή διγγραμμικού μετασχηματισμού
[bz1, az1] = bilinear(b1, a1, Fs)
[bz2, az2] = bilinear(b2, a2, Fs)
[bz3, az3] = bilinear(b3, a3, Fs)
[bz4, az4] = bilinear(b4, a4, Fs)
% Συνάρτηση μεταφοράς ψηφιακού φίλτρου
[Hz1, Wz1] = freqz(bz1, az1, 512, Fs);
[Hz2, Wz2] = freqz(bz2, az2, 512, Fs);
[Hz3, Wz3] = freqz(bz3, az3, 512, Fs);
[Hz4, Wz4] = freqz(bz4, az4, 512, Fs);
% Σχεδιασμός φασμάτων αναλογικών και ψηφιακών φίλτρων
figure(1); subplot(121); semilogx(Ws1/(2*pi), 20*log10(abs(Hs1)), 'Linewidth',2);
hold on; semilogx(Wz1, 20*log10(abs(Hz1)), 'r--'); grid on;
title('Butterworth'); xlabel('Συχνότητα (Hz)'), ylabel('Πλάτος (dB)');
legend('Αναλογικό','Ψηφιακό'); hold off;
subplot(122); semilogx(Ws2/(2*pi), 20*log10(abs(Hs2)), 'Linewidth',2);
hold on; semilogx(Wz2, 20*log10(abs(Hz2)), 'r--'); grid on;
title('Chebyshev I'); xlabel('Συχνότητα (Hz)'), ylabel('Πλάτος (dB)');
legend('Αναλογικό','Ψηφιακό'); hold off;
figure(2); subplot(121); semilogx(Ws3/(2*pi), 20*log10(abs(Hs3)), 'Linewidth',2);
hold on; semilogx(Wz3, 20*log10(abs(Hz3)), 'r--'); grid on;
title('Chebyshev II'); xlabel('Συχνότητα (Hz)'), ylabel('Πλάτος (dB)');
legend('Αναλογικό','Ψηφιακό'); hold off;
subplot(122); semilogx(Ws4/(2*pi), 20*log10(abs(Hs4)), 'Linewidth',2);
hold on; semilogx(Wz4, 20*log10(abs(Hz4)), 'r--'); grid on;
title('Ελλειπτικό'); xlabel('Συχνότητα (Hz)'), ylabel('Πλάτος (dB)');
legend('Αναλογικό','Ψηφιακό'); hold off;

```





Σχήμα 3 Απόκριση πλάτους πρότυπων αναλογικών φίλτρων και των αντίστοιχων ψηφιακών με βάση τον διγραμμικό μετασχηματισμό

Η απόκριση συχνότητας των πρότυπων αναλογικών φίλτρων και των αντίστοιχων ψηφιακών φίλτρων που παράγονται με τον διγραμμικό μετασχηματισμό δίνονται στο παραπάνω σχήμα.

4. Άλυτες ασκήσεις

1. Να σχεδιαστεί με τη μέθοδο του διγραμμικού μετασχηματισμού ένα βαθυπερατό ψηφιακό φίλτρο με τις παρακάτω προδιαγραφές:

$$|H(e^{j\omega})|^2 \geq -1 \text{ dB} \quad \text{για} \quad |\omega| \leq 0.4\pi$$

$$|H(e^{j\omega})|^2 \leq -15 \text{ dB} \quad \text{για} \quad 0.5\pi \leq |\omega| \leq \pi$$

για $T = 1 \text{ sec}$, $R_p = 1$, $A_s = 15$

2. Να σχεδιαστεί με χρήση του διγραμμικού μετασχηματισμού στο Matlab ένα βαθυπερατό ψηφιακό IIR φίλτρο με προδιαγραφές: $f_p = 700 \text{ Hz}$, $d_p = 0.1$, $f_s = 850 \text{ Hz}$, $d_s = 0.01$ και χρήση πρότυπου ελλειπτικού αναλογικού φίλτρου. Να σχεδιαστεί η απόκριση συχνότητας του φίλτρου. Θεωρήστε συχνότητα δειγματοληψίας $F_s = 44100 \text{ Hz}$.