ΨΗΦΙΑΚΕΣ ΕΠΙΚΟΙΝΩΝΙΕΣ Ι Ακ. Έτος 2022-23

Κουράκου Σοφία 03120869

Εργαστηριακή Άσκηση 2
Σχεδιασμός/υλοποίηση ψηφιακών φίλτρων FIR με το MATLAB®

Μέρος 1: Εισαγωγή

```
>> X=[-2:2]
>> fftshift(X)
>> ifftshift(X)
>> Y = fftshift(fftshift(X));
>> Z = ifftshift(fftshift(X));
>> isequal(X,Y)
>> isequal(X,Z)
```

Ερώτηση 1: Ποιο εκ των διανυσμάτων Y και Z ισούται με το X;

Απάντηση 1:Ισχύει οτι η σχέση X=ifftshift(fftshift(X)) επιστρέφει το ίδιο διάνυσμα X. Επομένως, ισχύει isequal(X,Y)=0 και isequal(X,Z)=1 (όπως προέκυψε και από την εκτέλεση του προγράμματος). Άρα το διάνυσμα Z ισούται με το X. Προφανώς, αφού το διάνυσμα X έχει περιττό μήκος N=5, ισχύει ότι fftshift και ifftshift δεν δίνουν το ίδιο αποτέλεσμα

Ερώτηση 2: Επαναλάβατε με X=[-1:2]. Τι παρατηρείτε;

Απάντηση 2: Σε αυτή την περίπτωση το μήκος του X είναι άρτιο N=4, επομένως, fftshift και ifftshift δίνουν το ίδιο αποτέλεσμα

```
>> close all; clear all;clc;
>> Xb=[0 0 1 1 1 1 1 0 0] % φάσμα βαθυπερατού σήματος με άρτια συμμετρία
>> figure; subplot (2,1,1); plot([-4:4],Xb); ylabel('Xb');
>> X=ifftshift(Xb) % το φάσμα με τις αρνητικές συνιστώσες στο άνω μέρος
>> x=ifft(X) % IFFT
>> xb=fftshift(x) % πραγματικό σήμα με άρτια συμμετρία όπως αναμένεται
>> subplot (2,1,2); plot([-4:4],xb); ylabel('xb');
```

Ερώτηση 3: Τροποποιείστε το προηγούμενο παράδειγμα ώστε να ξεκινήσετε απευθείας με τον ορισμό του φάσματος του βαθυπερατού σήματος X όπως το αναμένει η ifft.

Απάντηση 3: Για να ξεκινήσουμε απευθείας με τον ορισμό του φάσματος του βαθυπερατού σήματος X, θα γράψουμε το X για να μην χρειαστεί η εντολή X=ifftshift(Xb). Το Xb έχει περιττό αριθμό μήκους N=9 άρα θα βάλουμε τα 5 τελευταία ψηφία του τέλους στην αρχή με αποτέλεσμα X= X=[1 1 1 0 0 0 0 1 1]. Το script θα είναι :

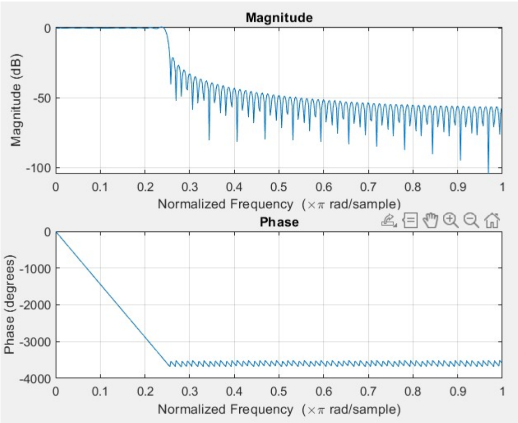
```
X= [1 1 1 0 0 0 0 1 1];  % το σήμα με αρνητικές συνιστώσες στο άνω μέρος
x=ifft(X);               % IFFT
xb=fftshift(x);          % πραγματικό σήμα άρτια συμμετρία όπως αναμένεται
subplot (2,1,2); plot(-4:4,xb); ylabel('xb');
```

Μέρος 2: Σχεδιασμός φίλτρων

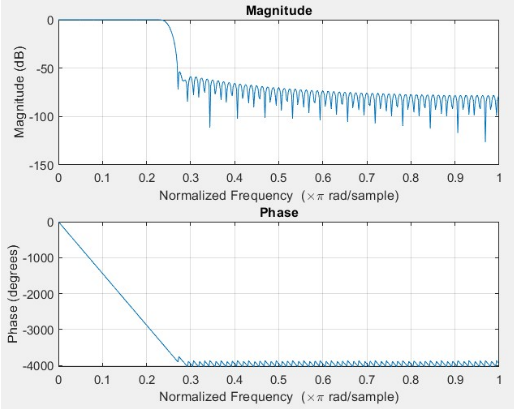
1. Αντί της εντολής h=[h(middle+1:end) h(1:middle)]; Μπορούμε να χρησιμοποιήσουμε την εντολή h= ifftshift(h);
2. Αντί του 64+1 φίλτρου θα χρησιμοποιήσουμε φίλτρο μήκους 160+1 τροποποιώντας κατάλληλα τον κώδικα :

```
middle=length(h)/2;
h=[h(middle+1:end) h(1:middle)];
h32=h(middle+1-16:middle+17);
h128=h(middle+1-64:middle+65);
h160=h(middle+1-80:middle+81);
figure; stem(0:length(h160)-1,h160); grid;
figure; freqz(h160,1);
wh=hamming(length(h160));
wk=kaiser(length(h160),5);
figure; plot(0:160,wk,'r',0:160,wh,'b'); grid;
h_hamming=h160.*wh';
figure; stem(0:length(h160)-1,h_hamming); grid;
figure; freqz(h_hamming,1);
```

Με σχεδιασμό ορθογωνικού παραθύρου προκύπτει :

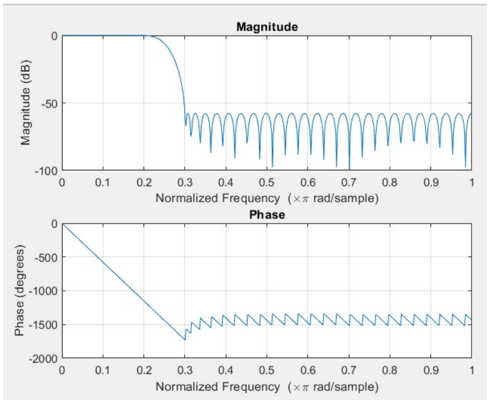


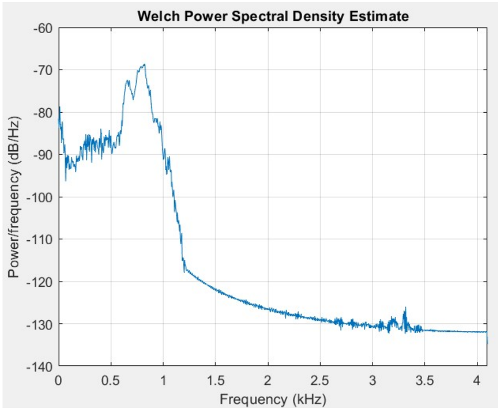
Με σχεδιασμό Hamming παραθύρου προκύπτει :



3. Φίλτρο Parks-McClellan μήκους **64+1** :

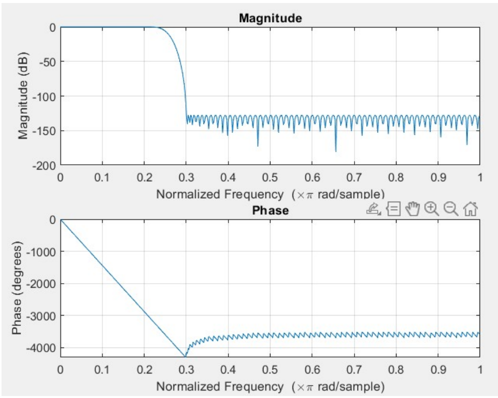
```
% Βαθυπερατό Parks-MacClellan 64+1
hpm=firpm(64, [0 0.10 0.15 0.5]*2, [1 1 0 0]);
figure; freqz(hpm,1);
s_pm=conv(s,hpm);
figure; pwelch(s_pm,[],[],[],Fs);
```

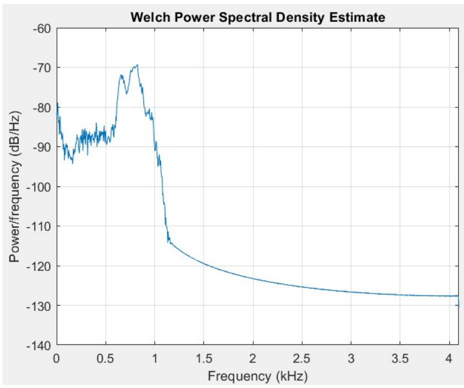




Φίλτρο Parks-McClellan μήκους **160+1** :

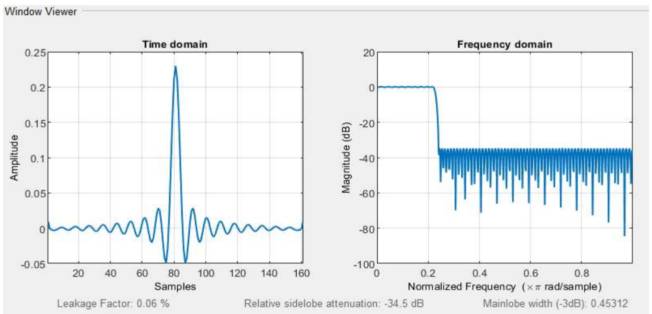
```
% Βαθυπερατό Parks-MacClellan
hpm=firpm(160, [0 0.10 0.15 0.5]*2, [1 1 0 0]);
figure; freqz(hpm,1); %
s_pm=conv(s,hpm);
figure; pwelch(s_pm,[],[],[],Fs);
```

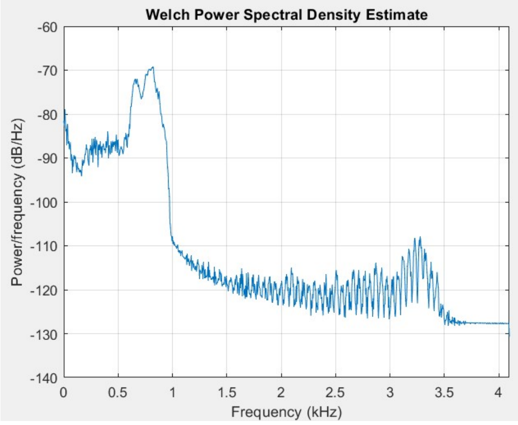


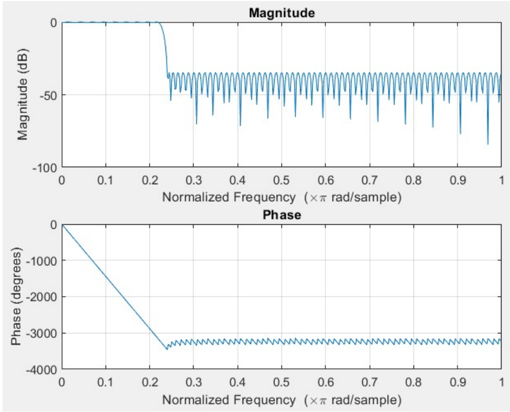


- Η διαφορά στην απόκριση του φίλτρου Parks-McClellan για 64+1 και 160+1 μήκος είναι ότι το 160+1 έχει χαμηλότερους ισοϋψείς λοβούς και έτσι κόβει καλύτερα τις ανεπιθύμητες συχνότητες στο σήμα.

4. Βάζοντας οριακές συχνότητες (0.11, 0.12) προκύπτει :



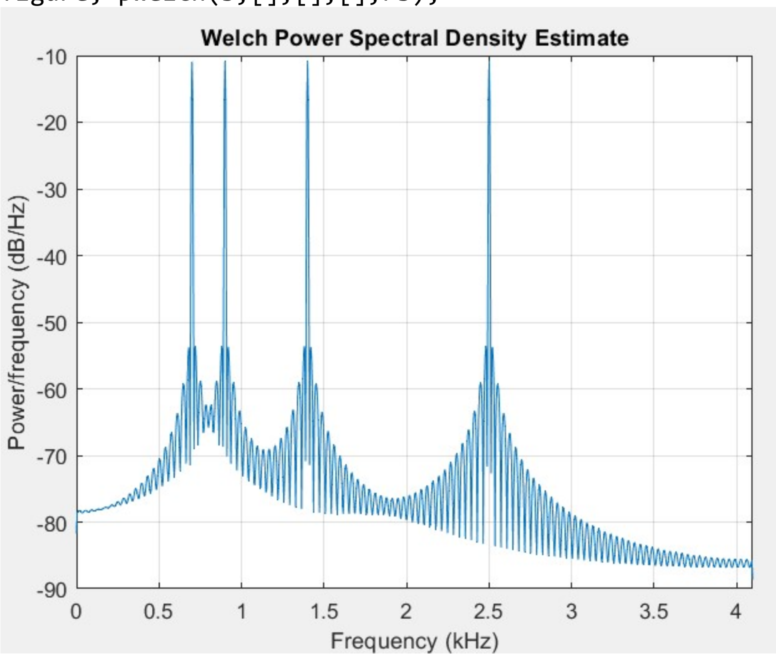




- Το "στένεμα" της κεντρικής συχνότητας καθιστά το φίλτρο χειρότερο από το αρχικό, καθώς οι πλευρικοί λοβοί γίνονται μεγαλύτεροι και η απόκριση συχνότητας δείχνει ότι το φάσμα περιέχει θόρυβο σε συχνότητες πάνω από τη συχνότητα αποκοπής, η απόσβεση στις θορυβικές συχνότητες γίνεται μικρότερη.

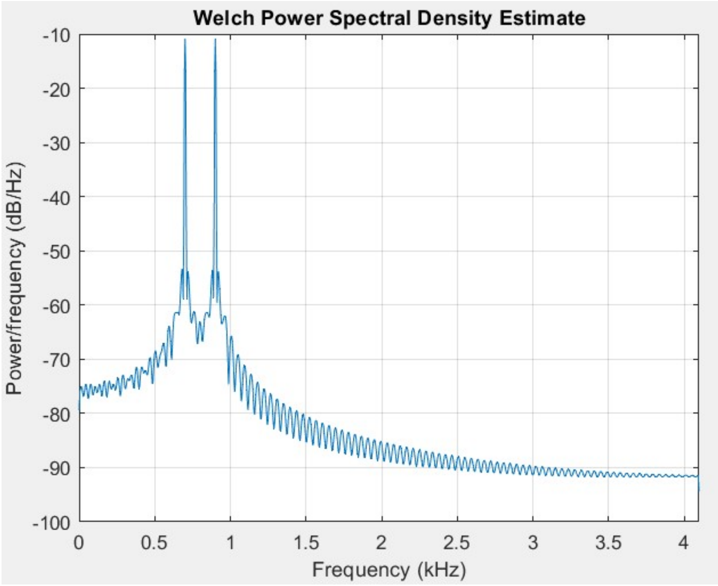
5. Αντικατάσταση του σήματος s με άθροισμα τεσσάρων ημιτονικών συναρτήσεων

```
Fs= 8192 ;
Ts=1/Fs;
t=0:Ts:1;
s=sin(2*pi*700*t)+ sin(2*pi*900*t)+ sin(2*pi*1400*t)+ sin(2*pi*2500*t);
figure; pwelch(s,[],[],[],Fs);
```



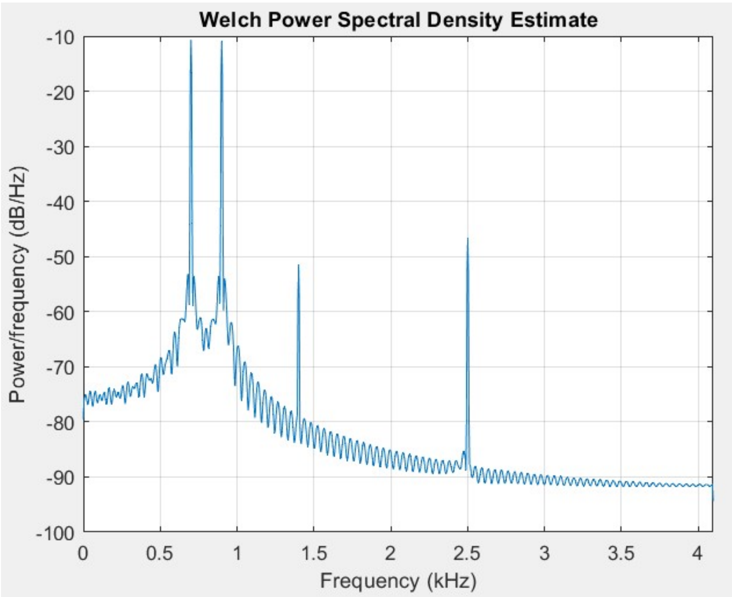
Αυτό σημαίνει ότι ,υπάρχουν τέσσερις συχνότητες Dirac στο ημιτονοειδές κύμα. Αναμένουμε ότι μετά το φιλτράρισμα θα παραμείνουν μόνο οι δύο πρώτες από αριστερά, καθώς βρίσκονται κάτω των 1 kHz .

Φίλτρο Parks-McClellan μήκους 160+1 από ερώτημα 3 με οριακές συχνότητες **(0.1, 0.15)** :



Όπως αναμέναμε, μετά το φιλτράρισμα έμειναν οι 2 πρώτες Dirac και οι υπόλοιπές αποκόπηκαν.

Φίλτρο Parks-McClellan μήκους 160+1 από ερώτημα 4 με οριακές συχνότητες **(0.11, 0.12)** :



Αντιθέτως με αυτό που αναμέναμε , με το φίλτρο με οριακές συχνότητες (0.11, 0.12) είναι ορατές και άλλες δύο συχνότητες Dirac πάνω της 1kHz. Δηλαδή λόγω του μικρού εύρους συχνοτήτων δεν γίνεται επαρκής απόσβεση στις ανεπιθύμητες συχνότητες.

- Στο συγκεκριμένο παράδειγμα, η μέθοδος του ορθογώνιου παραθύρου έχει υψηλούς πλευρικούς λοβούς, οι οποίοι εισάγουν ανεπιθύμητο θόρυβο στο φάσμα του φιλτραρισμένου σήματος, ενώ η μέθοδος Parks-McClellan έχει ισοϋψείς πλευρικούς λοβούς και αποκόπτει καλύτερα το σήμα. Συγκριτικά Hamming και Parks-McClellan, το δεύτερο είναι καλύτερο, λόγω του λιγότερου θορύβου που παρατηρούμε στο φάσμα του φιλτραρισμένου σήματος.

Μέρος 3: Εφαρμογή Α

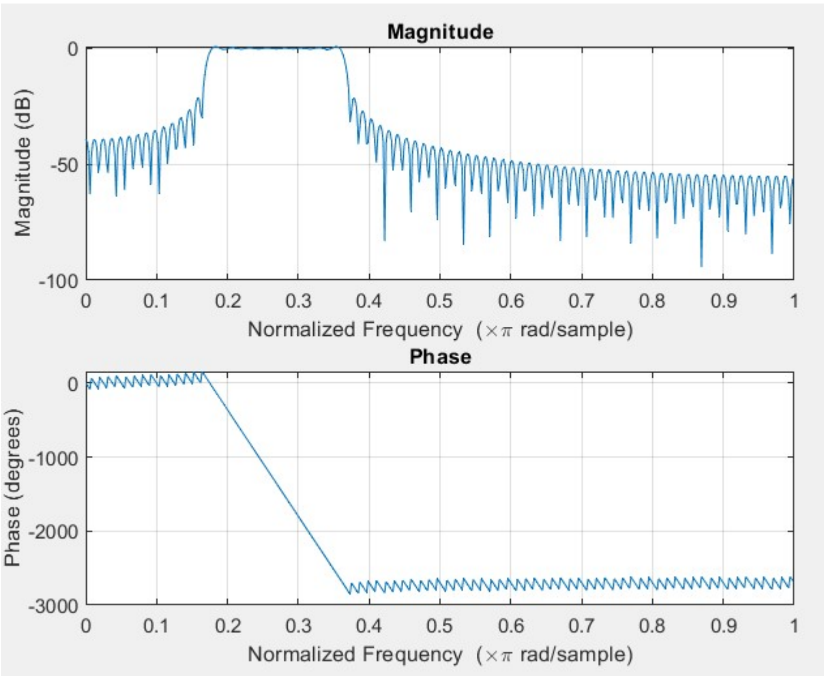
Δημιουργία ζωνοπερατού φίλτρου ζώνης διέλευσης 0.7 – 1.5 kHz => 700-1500 Hz

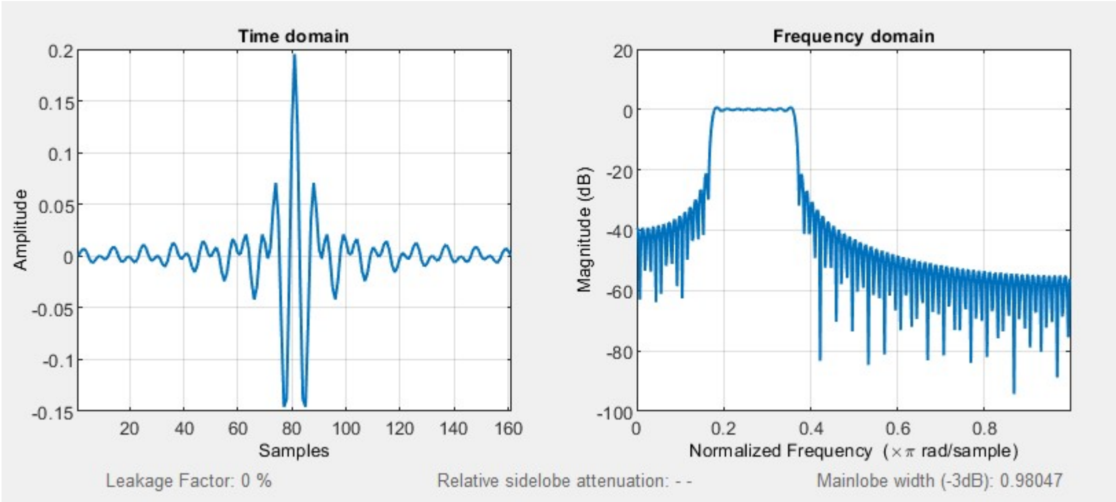
Από το load sima ; έχουμε την Fs = 8192 Hz

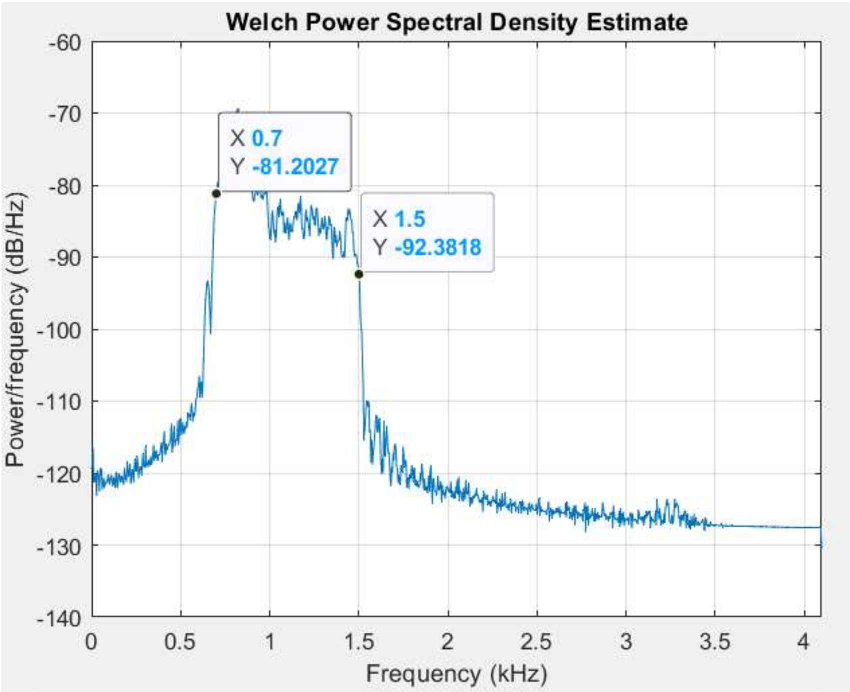
Ορίζεται η ιδανική ζωνοπερατή συνάρτηση H, με συχνότητες αποκοπής f1=700Hz , f2=1500Hz

```
f1=700;  f2=1500;
H=[zeros(1,f1) ones(1,(f2-f1)) zeros(1,(f1+Fs/2-f2)) ones(1,(f2-f1))...
       zeros(1,Fs/2-f2)];
figure(2); freqz(H,1);
% Υπολογίζεται η κρουστική απόκριση με αντίστροφο μετασχ. Fourier
% Εναλλακτικά, μπορεί να χρησιμοποιηθεί η αναλυτική σχέση Sa(x)
h=ifft(H,'symmetric');
middle=length(h)/2;
h=ifftshift(h);
h160=h(middle+1-80:middle+81);
figure(3); stem(0:length(h160)-1,h160); grid;
figure(4); freqz(h160,1);   % σχεδιάζουμε την απόκριση συχνότητας της h160
wvtool(h160);
```

Απόκριση φίλτρου - απόκριση συχνότητας της h160 **-ορθογωνικό**



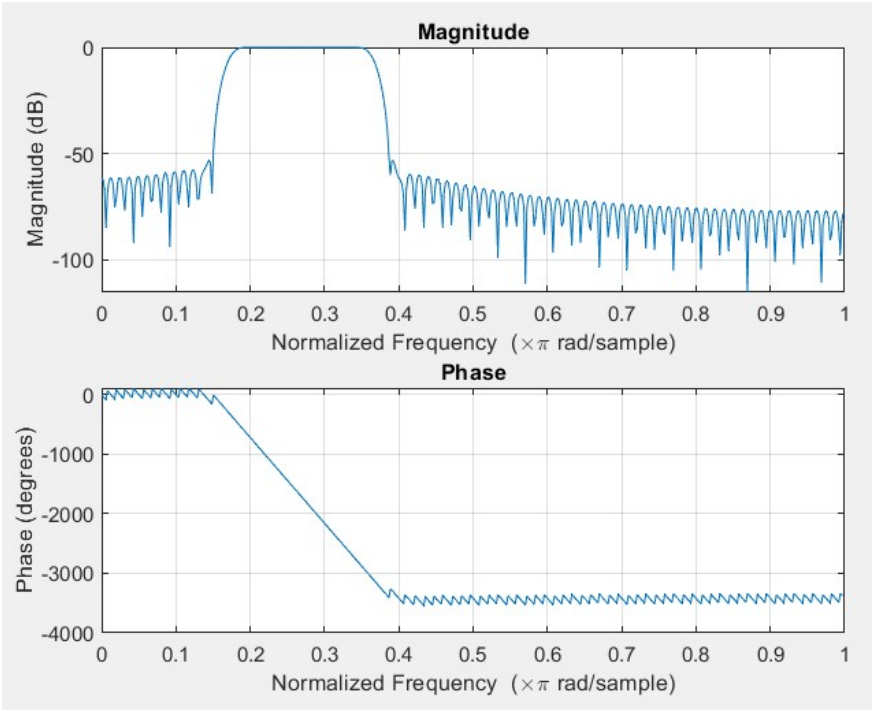


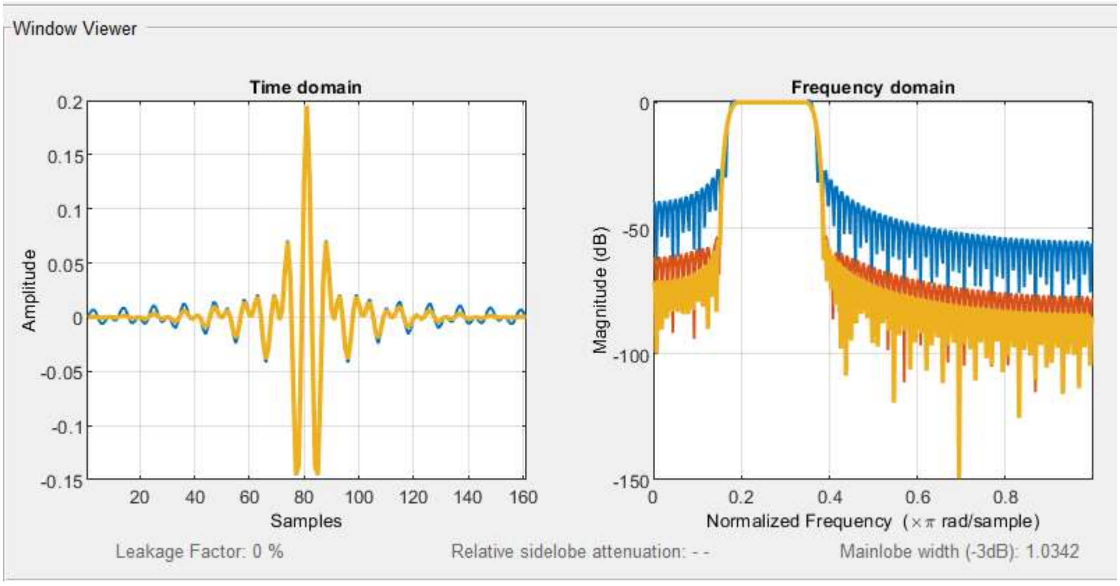


Η επίδραση πάνω στο σήμα με ορθογωνικό, δεν είναι ικανοποιητική ως προς την αποκοπή και τις συχνότητες που μας ενδιαφέρουν.

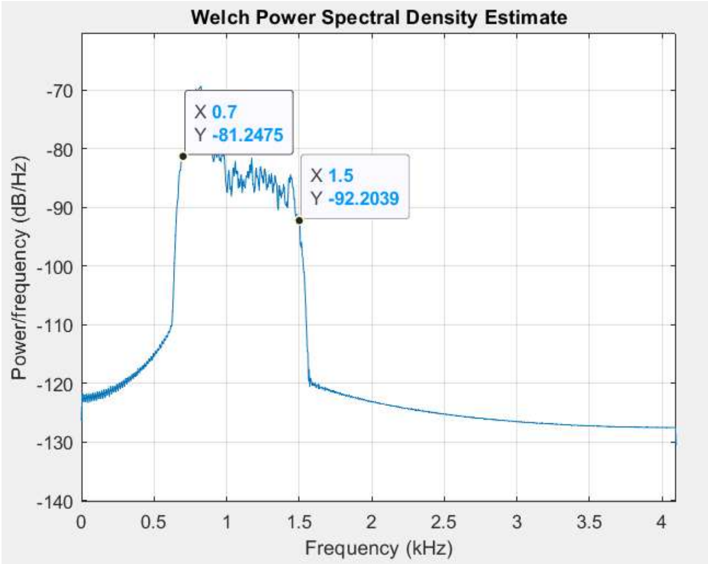
a) Ερώτημα από την άσκηση του βιβλίου – εφαρμογή παραθύρου (π.χ. **Kaiser**)

```
% Χρησιμοποιούμε την h160 και παράθυρα hamming και kaiser
wh=hamming(length(h160));
wk=kaiser(length(h160),5);
figure(5); plot(0:160,wk,'r',0:160,wh,'b'); grid;
h_hamming=h160.*wh';
figure(6); stem(0:length(h160)-1,h_hamming); grid;
figure(7); freqz(h_hamming,1);
h_kaiser=h160.*wk';
wvtool(h160,h_hamming,h_kaiser);
```





- Είναι καλύτερα τα φίλτρα μεγάλου μήκους, όμως τα φίλτρα μικρού μήκους είναι εξίσου αναγκαία προκειμένου να μην υπερφορτώνεται το δίκτυο.



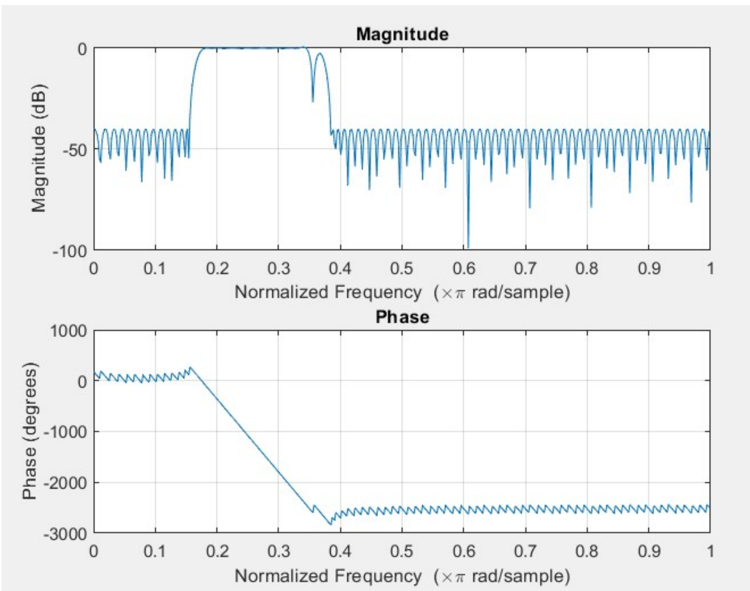
Η επίδραση στο σήμα, με παράθυρο Kaiser είναι αρκετά ικανοποιητική για τις συχνότητες και για της αποκοπή που ενδιαφέρουν.

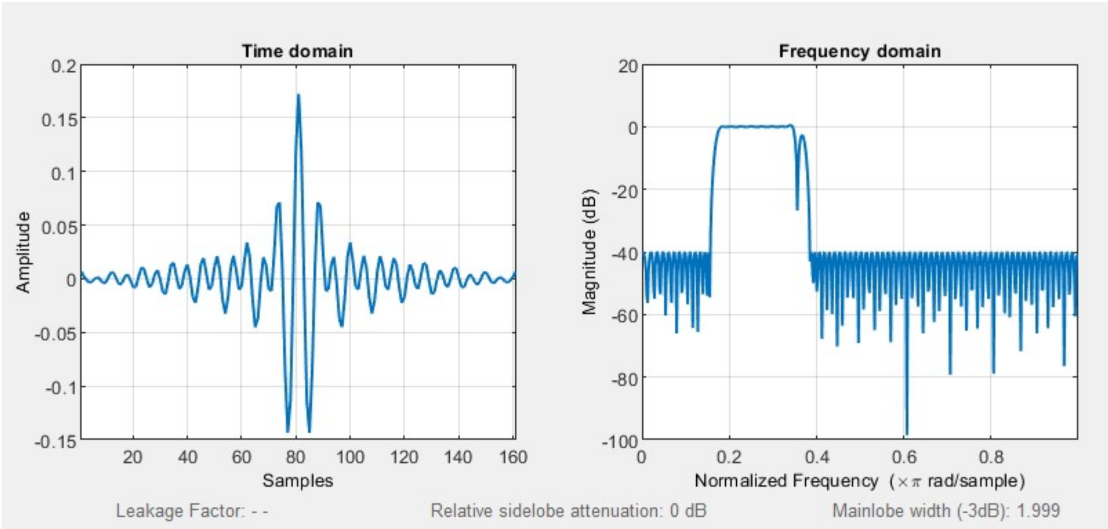
b) ερώτημα από την άσκηση του βιβλίου- **Parks-McClellan**

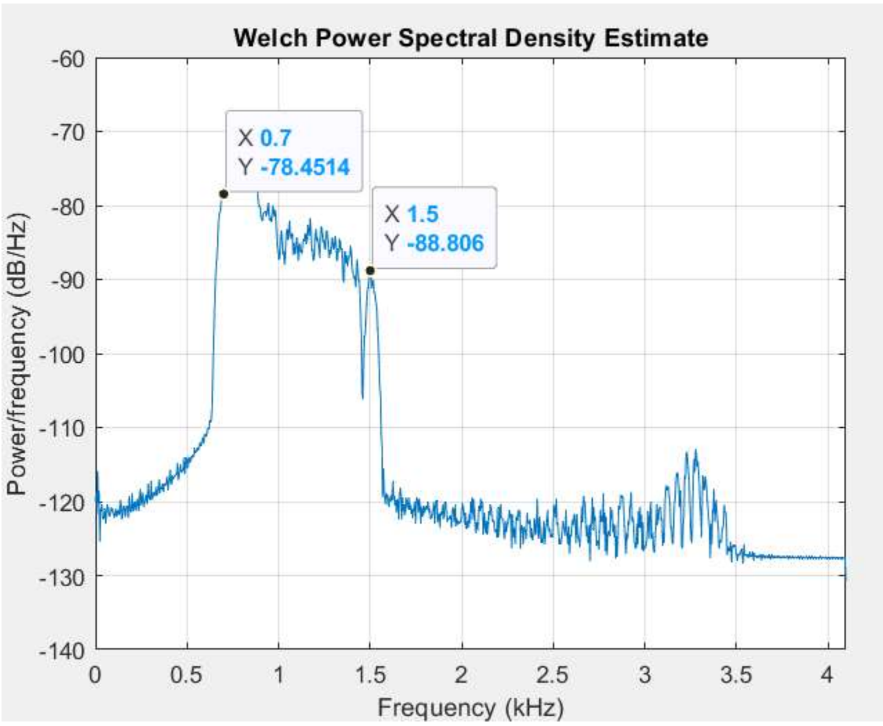
```
% Ζωνοπερατό Parks-MacClellan

f=2*[0 f1*0.91 f1*1.05 f2*0.91 f2*1.05 Fs/2]/Fs;
hpm=firpm(160,f , [0 0 1 1 0 0]);
figure(13); freqz(hpm,1);
s_pm=conv(s,hpm);
figure(14); pwelch(s_pm,[],[],[],Fs);
sound(20*s); % ακούμε το αρχικό σήμα, s
sound(20*s_pm); % ακούμε το φιλτραρισμένο σήμα, s_pm
```

Συγκριτικά με τον κώδικα του μέρους 2- βαθυπερατό φίλτρο, στο ζωνοπερατό έγιναν αλλαγές στην firpm ώστε να βγει μια καλή απεικόνιση, καθώς για μικρό εύρος συχνοτήτων δεν θα είχαμε επαρκής απόσβεση στις ανεπιθύμητες συχνότητες. Χρειαζόμαστε 6 σημεία για 3 σημεία μετάβασης `f=2*[0 f1*0.91 f1*1.05 f2*0.91 f2*1.05 Fs/2]/Fs;`
Είναι κανονικοποιημένα ως προς Fs/2 , το 2* προέκυψε από την ανάλυση των κλασμάτων
Στην `hpm=firpm(160,f , [0 0 1 1 0 0]);` έχουμε ότι 160 είναι το μήκος, f τα κανονικοποιημένα σημεία, και το τελευταίο στοιχείο ορίζει τους μηδενισμούς και τους άσσους.







Η επίδραση στο σήμα, με Parks-McClellan δεν είναι ικανοποιητική στο συγκεκριμένο σήμα.

- Σε αυτό το παράδειγμα παρατηρούμε πως ο σχεδιασμός με Kaiser είναι η καλύτερη επιλογή, δεδομένου των χαμηλότερων πλευρικών λοβών και της μη ύπαρξης πολύ θορύβου στα ενδιαφερόμενα διαστήματα που ορίζουν οι συχνότητες αποκοπής στο φάσμα του φιλτραρισμένου σήματος.