Budapesti Műszaki és Gazdaságtudományi Egyetem
Méréstechnika és Információs Rendszerek Tanszék

Digitális szűrők - (BMEVIMIM278)
FIR-szűrő tervezése ablakozással
Házi Feladat

Név: Szőke Kálmán Benjamin
Neptun: SLZ0UE
Dátum: 2014. december 17.

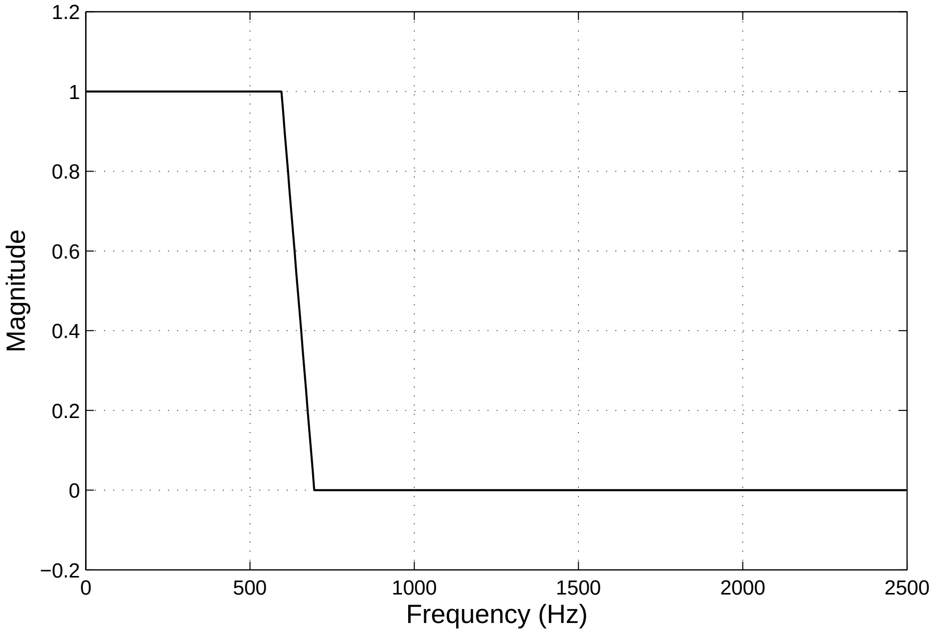
Tartalomjegyzék

Ábrák jegyzéke

1. Elméleti összefoglaló

1.1. Tervezés Fourier sorfejtéssel

Legyen az amplitúdó követelményt leíró függvény $H_d(\omega)$. Ez egy általunk (a tolerancia séma közepében felvett) valós, tipikusan szakaszonként konstans értékű, a mintavételi frekvencia szerint periodikus függvény. Ez a függvény előállítható Fourier-sorával:



1. ábra. $H_d(\omega)$ (terv)

$$H_d(\omega) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} h_d(n)\, e^{-j\omega n} dn$$

A Fourier-sor együtthatói:

$$h_d(n) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} H_d(\omega)\, e^{-j\omega n} d\omega$$

Mivel $H_d(\omega)$ valós, páros függvény, ezért a $h_d(n)$ sorozat is valós, páros sorozat, vagyis $h_d(-n) = h_d(n)$. Első lépésben az átviteli függvényt a $H_d(\omega)$ véges Fourier-sorával fogjuk közelíteni. Az így kapott véges, nem-kauzális sorozatot ezután az időtartományban eltoljuk annyival, hogy a sorozat kauzális (realizálható) legyen.

$H_d(\omega)$-át valós függvénynek vettük fel, annak fázisa azonosan zérus volt. Az időtartományban eltolt sorozathoz tartozó átviteli függvény fáziskarakterisztikája az eltolási tételnek megfelelően, egzaktul lineáris lesz. A fáziskarakterisztika meredeksége (a csoport futási idő) az eltolás nagyságával egyezik meg. A szűrő által okozott jel késleltetés

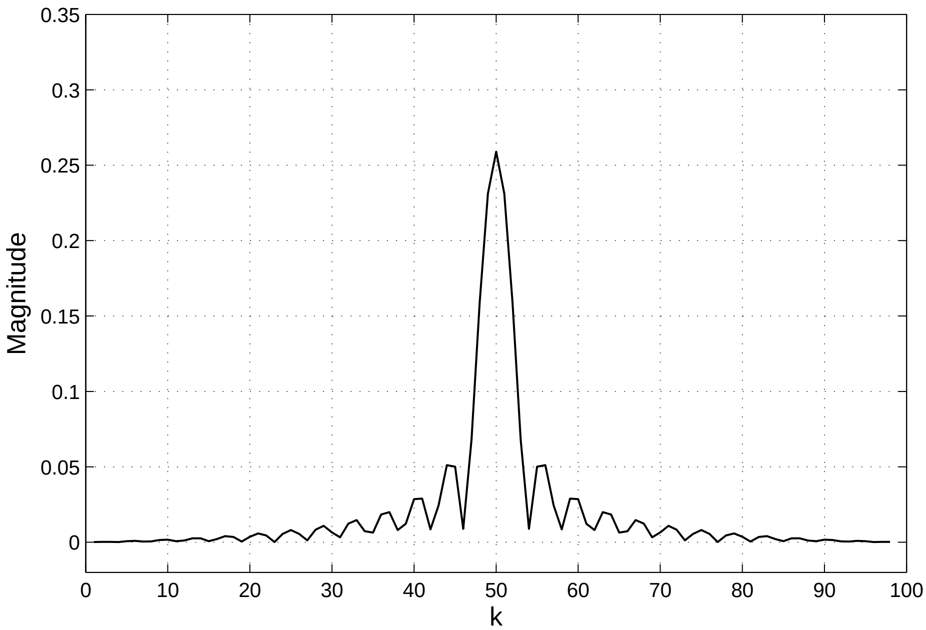
jelentős lesz, melynek nagysága csak a tervezés során alakul ki (ugyanis a választott fokszámtól függ). A fenti gondolatmenetet követve válasszuk a szűrő fokszámát páratlanra ($N-1=2R$)!
A megengedett kauzális $H(z)$ átviteli függvény így a következő:

$$H(z) = z^{-R} \sum_{n=0}^{N-1} h(n)\, z^{-(n-R)} = z^{-R} G(z)$$

$$G(z) = \sum_{k=-R}^{R} h(k+R)\, z^{-k} = \sum_{k=-R}^{R} g(k)\, z^{-k}$$

$$g(k) = h(k+R)\,, \quad k = -R, \ldots + R$$

A $g(k)$ sorozatot az eltolási tétel segítségével állítottuk elő, $h(n)$ eltolásával. A frekvencia



2. ábra. $g(k)$

tartományba áttérve $H(\omega)$ a következő ($k=n-R$):

$$H(\omega) = e^{-j\omega R} G(\omega)$$

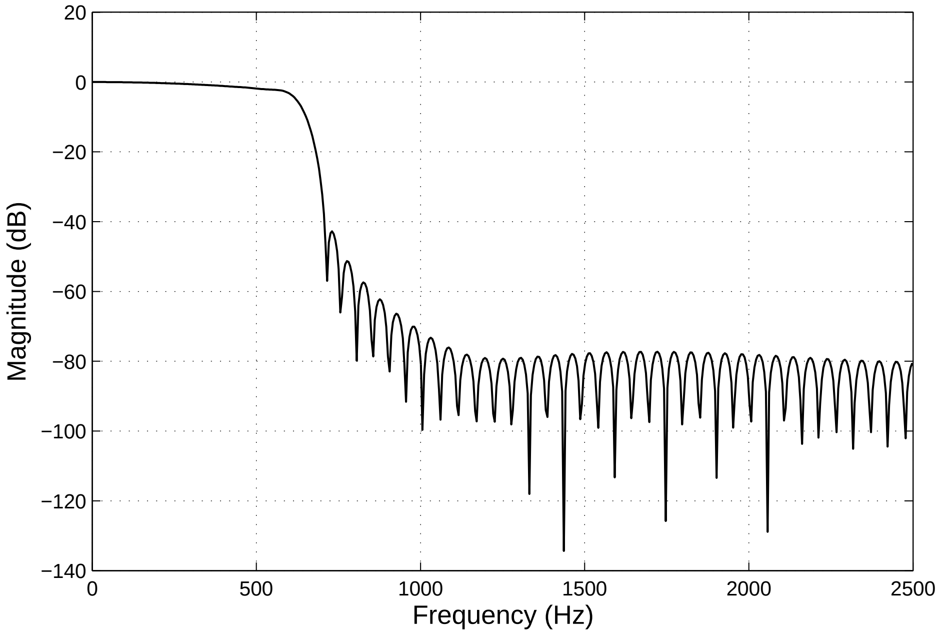
$$G(\omega) = \sum_{k=-R}^{R} g(k)\, e^{-jk\omega}$$

A szűrő $\varphi(\omega)$ fáziskarakterisztikája lineáris:

$$\varphi(\omega) = -\omega R$$

A sorfejtést megfelelően nagy indexhatárig (R) elvégezve:

$$G(\omega) \approx H_d(\omega)$$



3. ábra. $G(\omega)$ (csonkolás utáni eredmény)

```
a = [ 1 1 0 0 ];
Fs = 5000;
f = [ 0 600 700 Fs/2 ];
L = 1000;
N = 99;
```

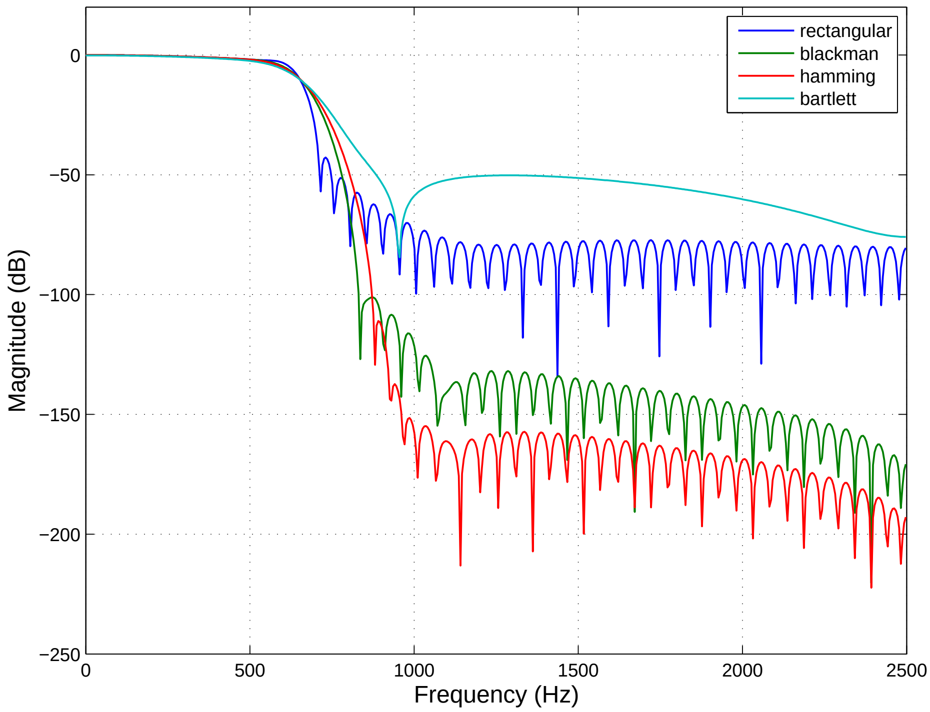
1.2. Tervezés ablakozással

A FIR szűrők ablakozással történő tervezését úgy fogjuk végezni, hogy a kívánt $H_d(\omega)$ karakterisztikát Fourier-sorba fejtjük, majd a $h_d(k)$ együtthatókat egy megfelelően választott $w(k)$ ablakfüggvénnyel súlyozzuk.

$$h_d(n) = \frac{1}{\sqrt{2\pi}} \int_{-\pi}^{\pi} H_d(\omega)\, e^{j\omega n} d\omega$$

$$g(k) = w(k) \cdot h_d(k)$$

Az ablakozással a szűrő erősítési szintje megváltozik. Megfelelő ablakfüggvény választása mellett így a Gibbs oszcilláció csökkenthető a záró tartományban.



4. ábra. Különböző ablakfüggvények eredménye

```
a = [ 1 1 0 0 ];
Fs = 5000;
f = [ 0 600 700 Fs/2 ];
L = 1000;
N = 99;
```

2. Megvalósítás

2.1. Matlab függvény használata

Az általam készített `myfir()` Matlab FIR szűrő tervező függvényhez 6 paramétert kell a tervezés elején rögzítenünk. Ezek közül az első a kívánt amplitúdó karakterisztika (`a`) 0 és 1 numerikus értékekkel jellemezve és egy vektorba tárolva, második az ezekhez a tartományokhoz tartozó frekvenciák (`f`) szintén vektorba tárolva, továbbá a mintavételi frekvencia (`Fs`), az amplitúdó követelményt leíró $H_d(\omega)$ terv függvény diszkrét nagysága (`L`), az ablakfüggvény diszkrét nagysága (`N`), és hatodikként megadhatjuk az alkalmazandó ablakfüggvény típusát is (`'rectangular', 'blackman', 'hamming', 'bartlett'`).

Amennyiben az utolsó paramétert nem adjuk meg, úgy automatikus a négyszögletes (`'rectangular'`) ablakfüggvénnyel történik a tervezés. A függvény eredményként a kívánt (`N`) fokszámú szűrő együtthatóit adja vissza.

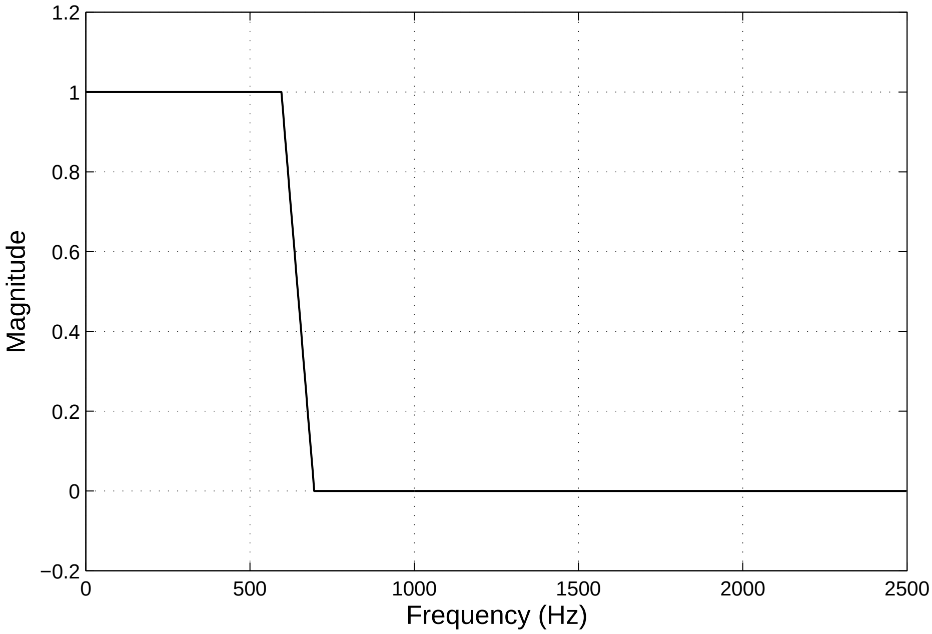
```
a = [ 1 1 0 0 ];
Fs = 5000;
f = [ 0 600 700 Fs/2 ];
L = 1000;
N = 99;
h = myfir( a, f, Fs, L, N, 'hamming' );
```

2.2. Működés releváns példák segítségével

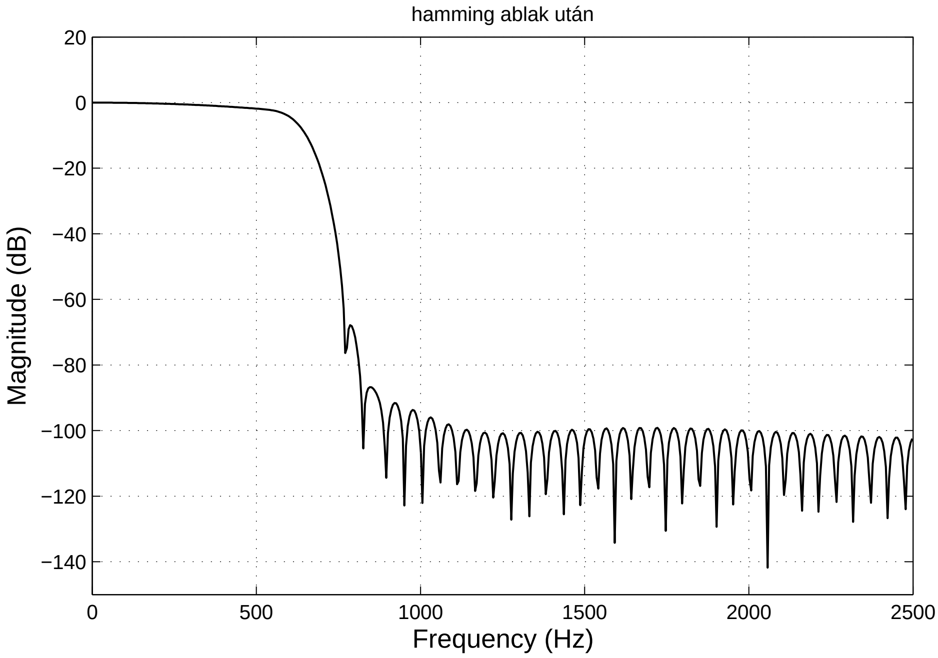
A működős teszteléséhez generáltam egy $f_1 = 100\,Hz$-es és egy $f_2 = 900\,Hz$-es szinuszjelet. Ezek összeadása után egy aluláteresztő szűrővel a 900 Hz-es szinuszt leszűrtem hamming ablakot használva. Végeredményben sikeresen visszakaptam 100 Hz-es frekvenciájú komponenst, a tervezési eljárás során várható késleltetéssel együtt.

```
close all
clear all

a = [ 1 1 0 0 ];
Fs = 5000;
f = [ 0 600 700 Fs/2 ];
L = 1000;
N = 99;
h = myfir( a, f, Fs, L, N, 'hamming' );

Ts=1/Fs;
f1=100;
f2=900;
t = 0:Ts:10*(1/f1);
x100=2*sin(2*pi*f1.*t);
x900=0.5*sin(2*pi*f2.*t);
x=x100+x900;
y = filter(h,1,x);

figure;
```

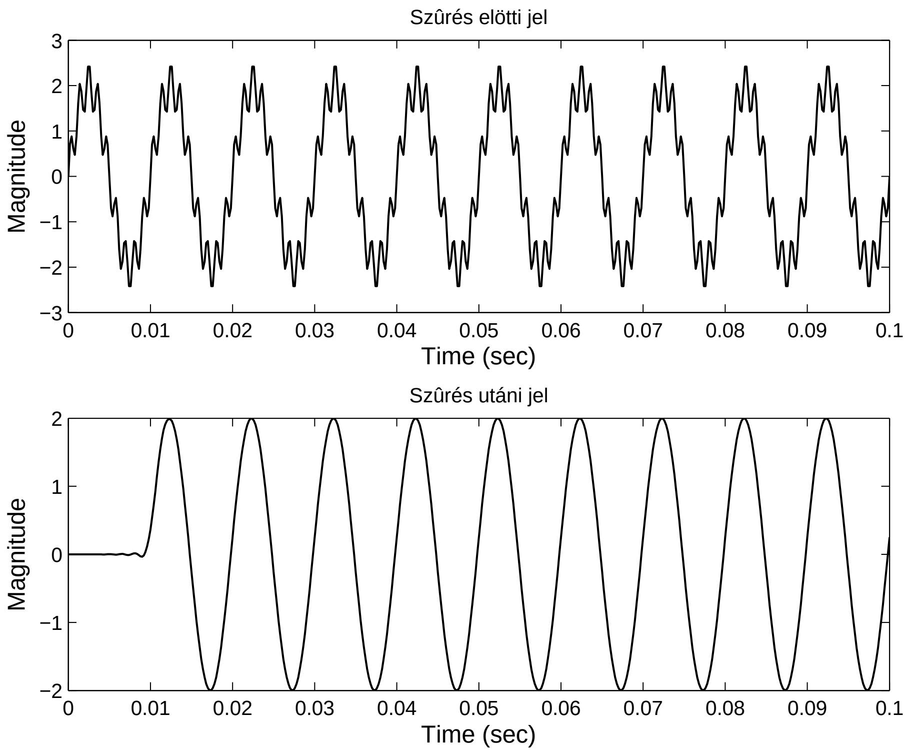
```
subplot(2,1,1);
plot(t,x,'k','LineWidth',1)
title('Szűrés elötti jel');
ylabel('Magnitude','FontSize',12)
xlabel('Time (sec)','FontSize',12)

subplot(2,1,2);
plot(t,y,'k','LineWidth',1)
title('Szűrés utáni jel');
ylabel('Magnitude','FontSize',12)
xlabel('Time (sec)','FontSize',12)
```



5. ábra. Tervezett aluláteresztő szűrő amplitúdó menete



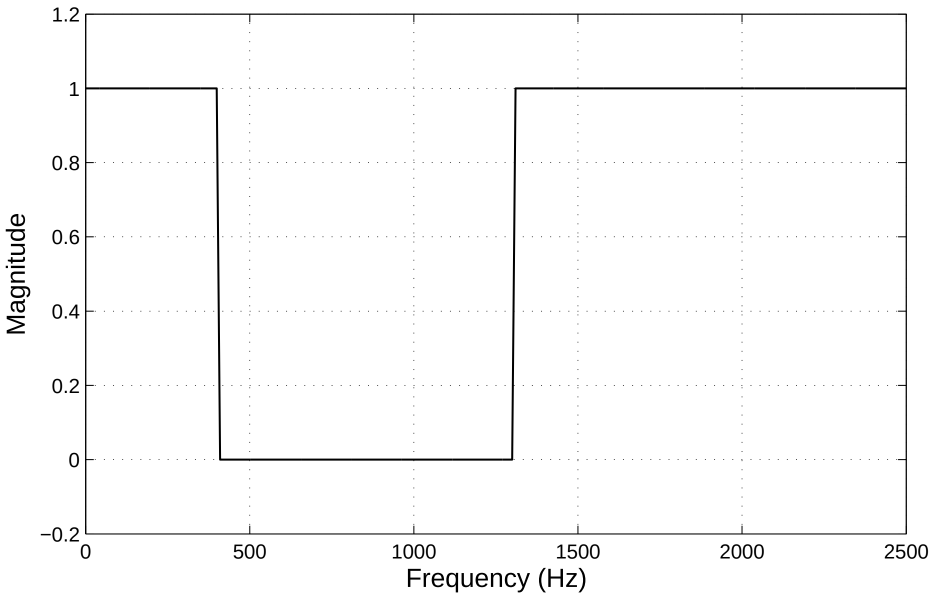
6. ábra. Tervezés eredménye hamming ablakfüggvényt használva



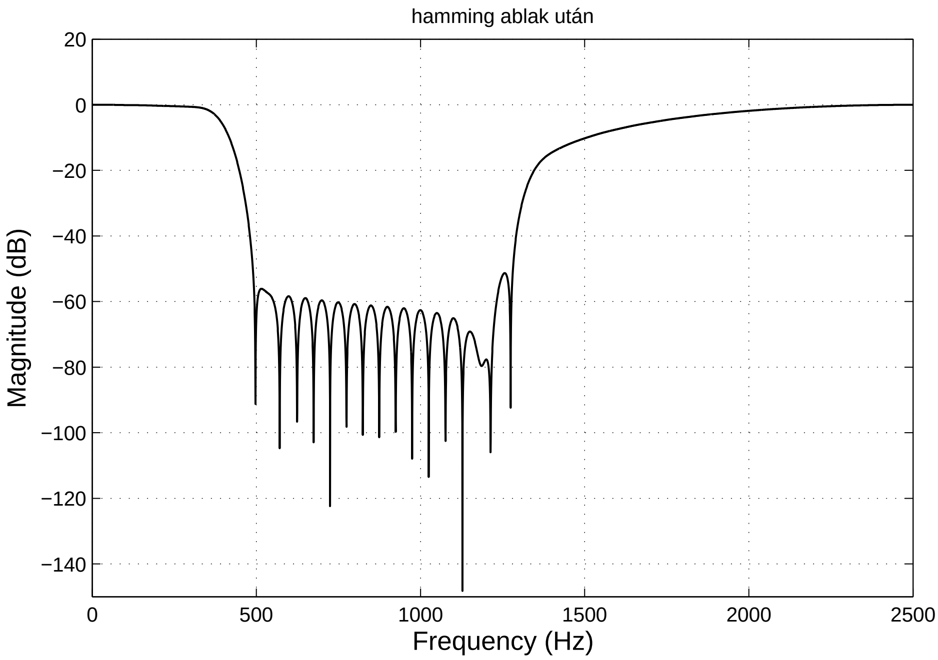
7. ábra. Szűrés eredménye

Az általam elkészített Matlab FIR szűrő tervezővel tetszőleges amplitúdó menet tervezhető.

```
a = [ 1 1 0 0 1 1 ];
Fs = 5000;
f = [ 0 400 410 1300 1310 Fs/2 ];
L = 10000;
N = 99;
h = myfir( a, f, Fs, L, N, 'hamming' );
```



8. ábra. Sávzáró szűrő terve



9. ábra. Sávzáró szűrő eredménye hamming ablakfüggvényt használva

3. Melléklet

Matlab kód

```
function [ g_win ] = myfir( a, fd, Fs, L, N, win )

if nargin < 6
    win = 'rectangular';
end

close all
delta_f = Fs/(L-1);
frek = 0:delta_f:Fs;
leng = length(frek);
halfleng = leng/2;

if( leng < N )
    disp('N túl nagy');
    return;
end

fd_size = length(fd);
for i=1:fd_size
 if( Fs/2 < fd(i) )
    disp('Fs kicsi');
    disp(Fs);
    return;
 end
end



d = a(1) * ones(1,L);
k = length(a);
for i=2:k

    if a(i-1)~=a(i)
        x0=round(fd(i-1)/delta_f);
        x1=round(fd(i)/delta_f);

        p = polyfit([x0 x1],[a(i-1) a(i)],1);

        x=x0:1:x1;
        d( x0 : x1) = polyval(p,x);
        d( x1 :  end) = a(i);

    else
        x=round(fd(i)/delta_f);
        d( x :  end) = a(i);
    end
end
d((L/2)+1:end)=fliplr( d(1:(L/2)) );
```

```
figure;
plot(frek(1:halfleng),d(1:halfleng),'k','LineWidth',1);
title('H_d - terv')
ylabel('Magnitude','FontSize',13)
xlabel('Frequency (Hz)','FontSize',13)
grid on

hd = real(ifft(d));
leng_hd = length(hd);


figure;
plot(frek,abs(hd),'k','LineWidth',1);
title('h_d')
ylabel('Magnitude','FontSize',13)
xlabel('Frequency (Hz)','FontSize',13)
grid on

h = zeros(1,L);
g = [ hd(round(leng_hd-N/2+1):end), hd(round(1:N/2)) ];
g_size = length(g);
h((L/2)-(g_size/2):(L/2)+(g_size/2)-1) = g;

figure;
plot(abs(g),'k','LineWidth',1);
title('g - csonkolás')
ylabel('Magnitude','FontSize',13)
xlabel('N','FontSize',13)
grid on

H0 = real(fft(h));

figure;
plot(frek(1:halfleng),20*log10(abs(H0(1:halfleng))),'k','LineWidth',1);
title('H - csonkolás után')
ylabel('Magnitude (dB)','FontSize',13)
xlabel('Frequency (Hz)','FontSize',13)
grid on

switch win
    case 'blackman'
        w = blackman(g_size)';
        g_win = h((L/2)-(g_size/2):(L/2)+(g_size/2)-1).*w;
        h((L/2)-(g_size/2):(L/2)+(g_size/2)-1) = g_win;
    case 'hamming'
        w = hamming(g_size)';
        g_win = h((L/2)-(g_size/2):(L/2)+(g_size/2)-1).*w;
        h((L/2)-(g_size/2):(L/2)+(g_size/2)-1) = g_win;
    case 'bartlett'
        w = bartlett(g_size)';
        a_win = h((L/2)-(a_size/2):(L/2)+(a_size/2)-1).*w;
        h((L/2)-(a_size/2):(L/2)+(a_size/2)-1) = a_win;
    case 'kaiser'
        w = kaiser(g_size)';
```

```
        g_win = h((L/2)-(g_size/2):(L/2)+(g_size/2)-1).*w;
        h((L/2)-(g_size/2):(L/2)+(g_size/2)-1) = g_win;
    otherwise
        g_win = g;
end
H = real(fft(h));

f=frek(1:halfleng);
figure;
plot(frek(1:halfleng),20*log10(abs(H(1:halfleng))),'k','LineWidth',1);
title(['H - csonkolás és ',win,' ablak után'])
ylabel('Magnitude (dB)','FontSize',13)
xlabel('Frequency (Hz)','FontSize',13)
grid on

end
```