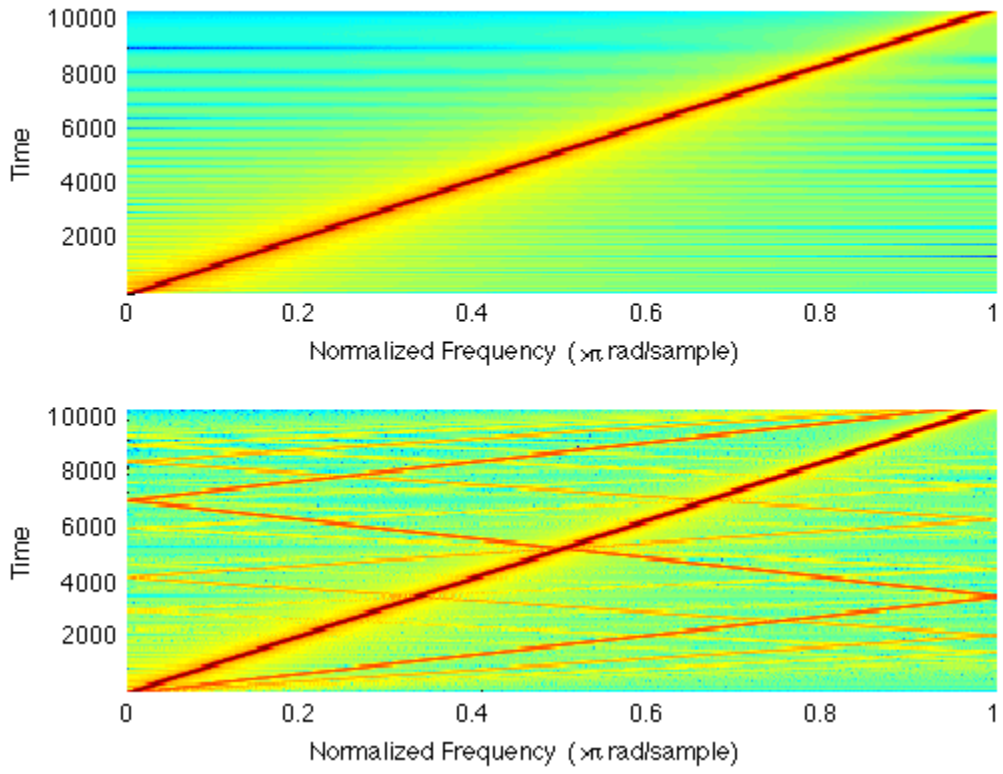
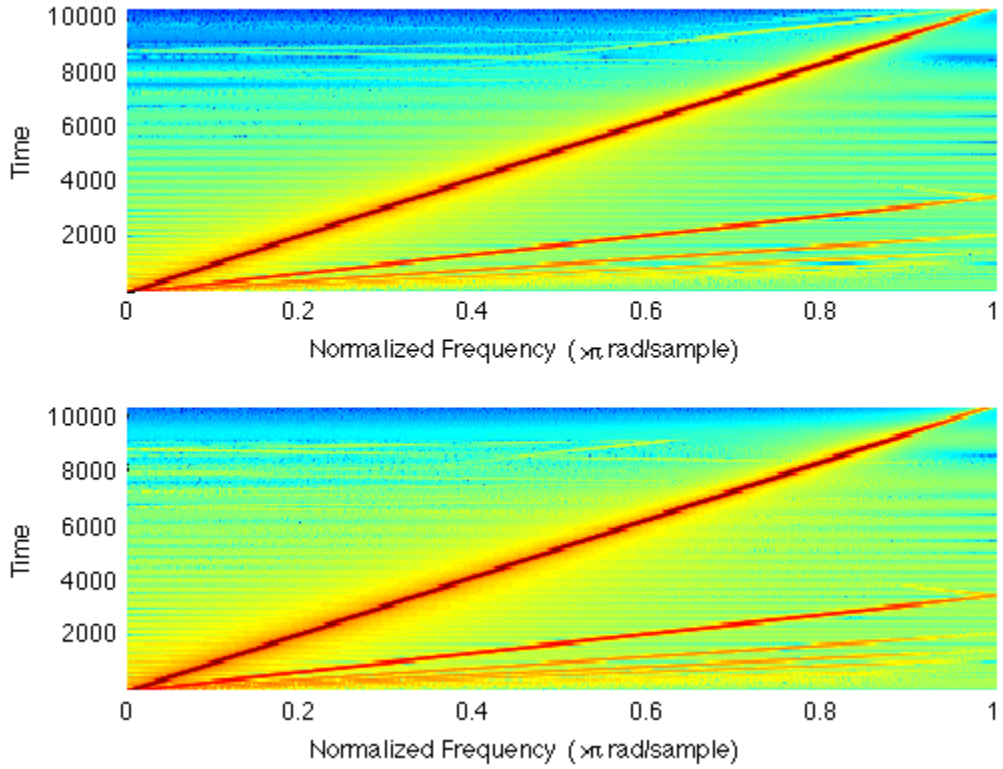
```
% question 1a

[x,fs] = wavread('linearSweep.wav');
[y,fs] = wavread('sweep1a.wav');

subplot(2,1,1), spectrogram(x,1024)
subplot(2,1,2), spectrogram(y(:,1),1024)

% When the frequecy sweeps above the nyquist, we can hear the
% frequencies decreasing, which never actually happens and
% therefore would not be audible were it not for aliasing.
```



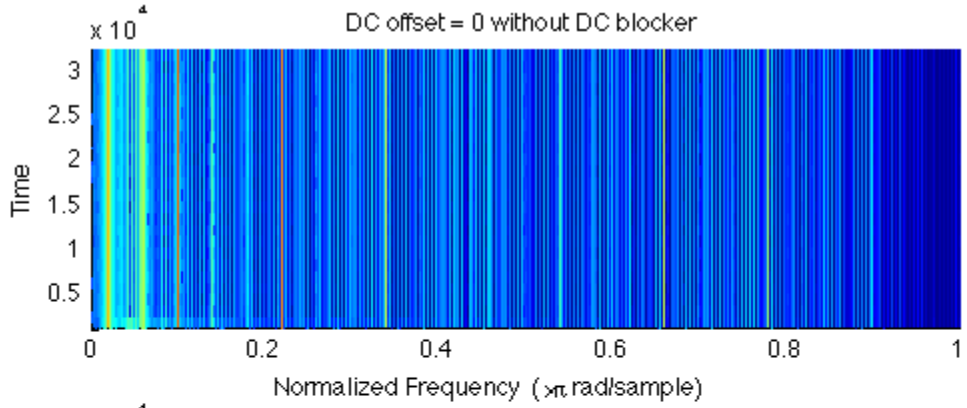
Published with MATLAB® 7.10

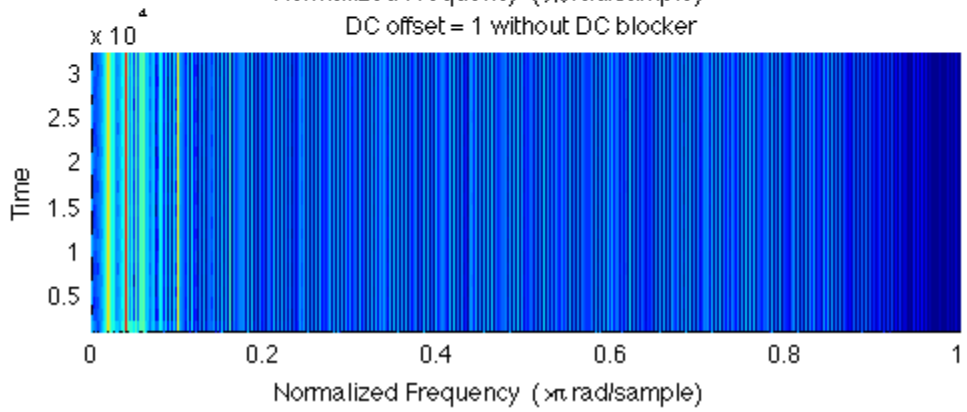
```
% question 1b

% try out the 3 kinds of filters

% [B_butter, A_butter] = butter(12,0.12);
[B_cheby, A_cheby] = cheby2(12,96,0.16);
% [B_ellip, A_ellip] = ellip(12,1/8);


% obtain SOS
% [co,g] = tf2sos(B_butter, A_butter);
[co,g] = tf2sos(B_cheby, A_cheby);

% better display function to copy paste directly to code
matrify(co)
disp(g)

% plot output
[x,fs] = wavread('sweep1bsoft.wav');
[y,fs] = wavread('sweep1bhard.wav');

subplot(2,1,1), spectrogram(x(:,1),1024);
subplot(2,1,2), spectrogram(y(:,1),1024);

%the aliasing is a lot more when a hard clip is used as compared to a soft
%clip because the presence of a hard corner adds infinite bandwidth to the
%signal and hence a lot more of the frequencies above nyquist gain
%amplitude and flip around to alias

% aside from the aliasing the hard clip produces a 'brighter' tone as the
% higher frequencies have more energy than the comparable soft clipped
% signal


{1.0000000000000000,1.1785487532710102,0.9999999999999919,-1.2757184904151924,0.40
{1.0000000000000000,-0.7583238192403949,1.0000000000000502,-1.3457584572238965,0.4
{1.0000000000000000,-1.3959068220445356,0.9999999999984834,-1.4588611015844655,0.5
{1.0000000000000000,-1.6207641789064129,1.0000000000105453,-1.5852516436785695,0.6
{1.0000000000000000,-1.7132116937049666,0.9999999999764242,-1.7090403183235447,0.8
{1.0000000000000000,-1.7485950542998536,1.0000000000145233,-1.8285375590278772,0.9
     4.095208812873113e-05
```

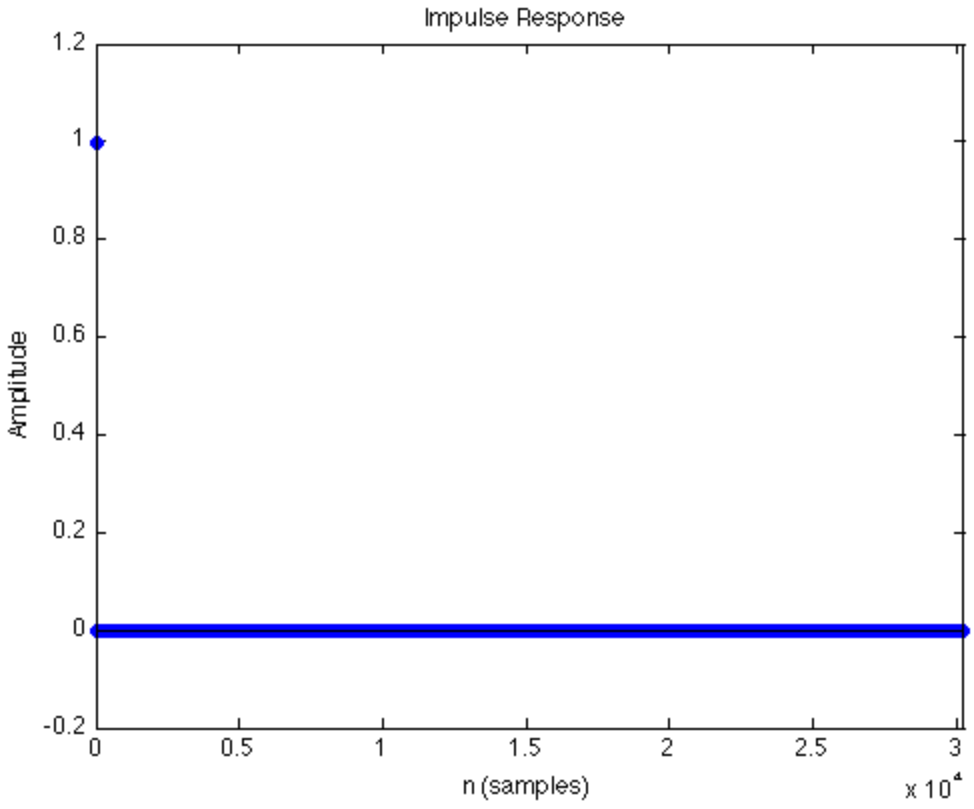


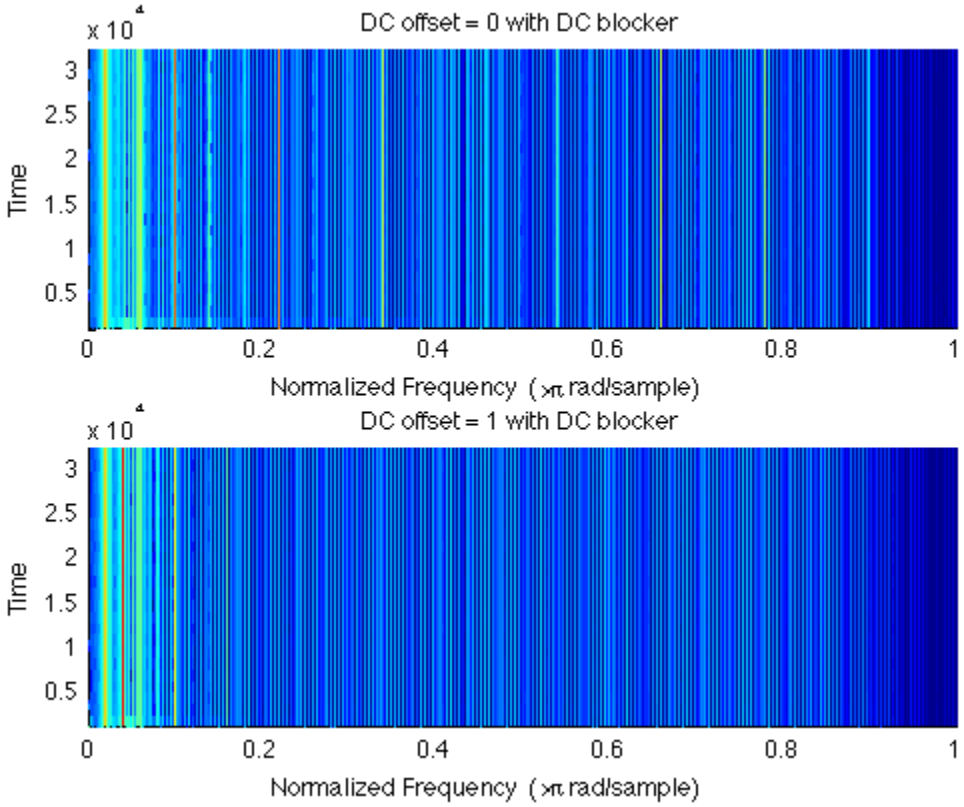
Published with MATLAB® 7.10

```
% question 2a

% if an offset is added to the output of the non-linearity, theoretically
% the sound remains the same since the vibrations created would be the same
%, just about a different 'mean'.

% question 2b


[y,fs] = wavread('sineDist00.wav');

[z,fs] = wavread('sineDist10.wav');

figure()
subplot(2,1,1), spectrogram(y(:,1),blackman(16384));
title('DC offset = 0 without DC blocker')
subplot(2,1,2), spectrogram(z(:,1),blackman(16384));
title('DC offset = 1 without DC blocker')

% the 2 plots are for DC Offset of 0.5 and 0.9
% DC Offset 0.5 has more energy and is spread out near the fundamental and
% its harmonics. For DC Offset 0.9, the energy gets narrowly concentrated
% near the fundamental and their harmonics. Both of them have even an odd
% harmonics

%  question 2c
%
b = [1 0 0];
a = [1 2*2*pi*5 2*pi*5*2*pi*5];


[bz,az] = bilinear(b,a,48000);
figure()
impz(bz,az);

% the impulse response settles in 1 sample according to the above
% calculations.

[y,fs] = wavread('sineDist00dc.wav');
[z,fs] = wavread('sineDist10dc.wav');

figure()
subplot(2,1,1), spectrogram(y(:,1),blackman(16384));
title('DC offset = 0 with DC blocker')
subplot(2,1,2), spectrogram(z(:,1),blackman(16384));
title('DC offset = 1 with DC blocker')
```

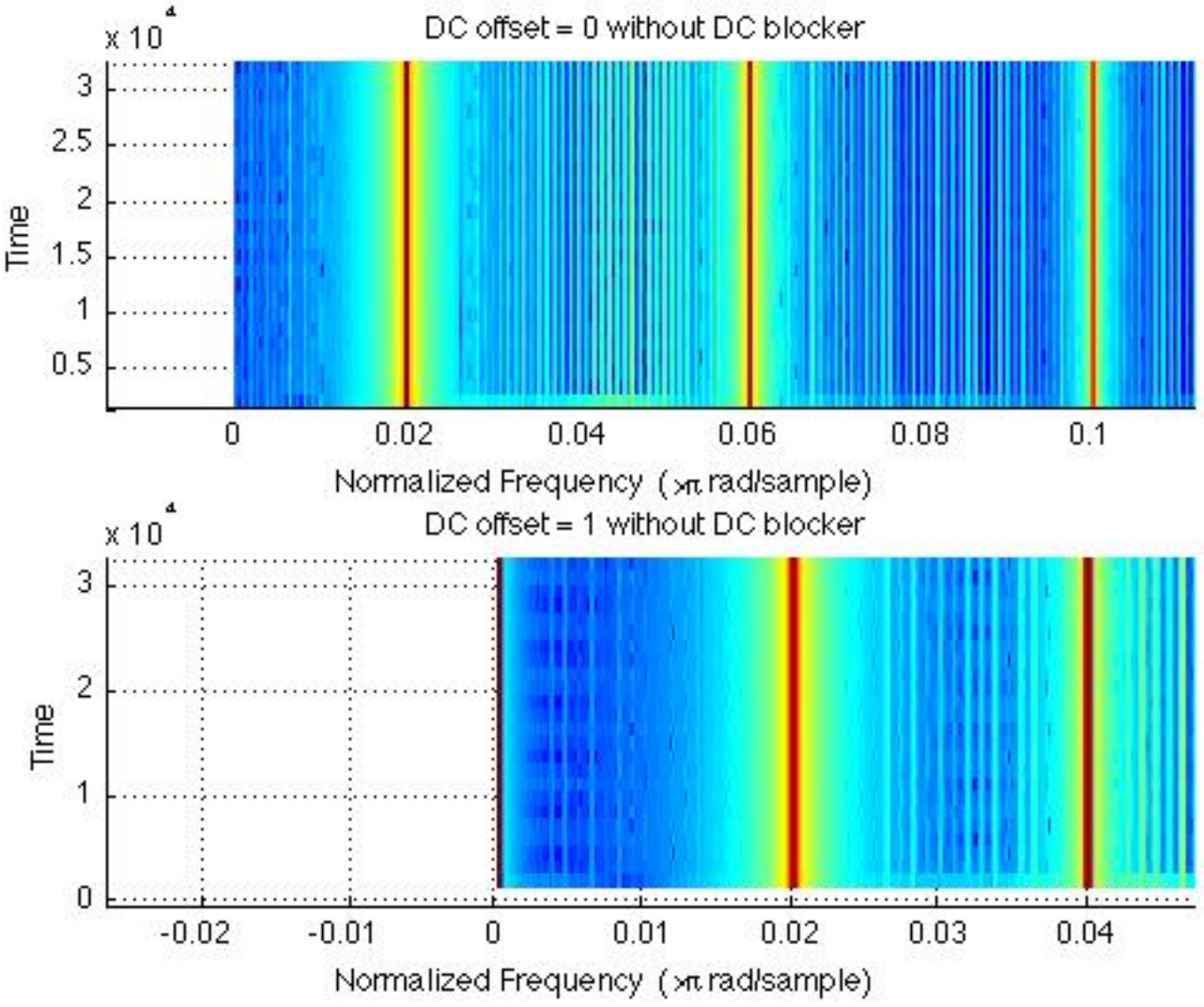

DC offset = 0 without DC blocker
x 10^4
Time
3
2.5
2
1.5
1
0.5
0
0.2
0.4
0.6
0.8
1
Normalized Frequency (×π rad/sample)

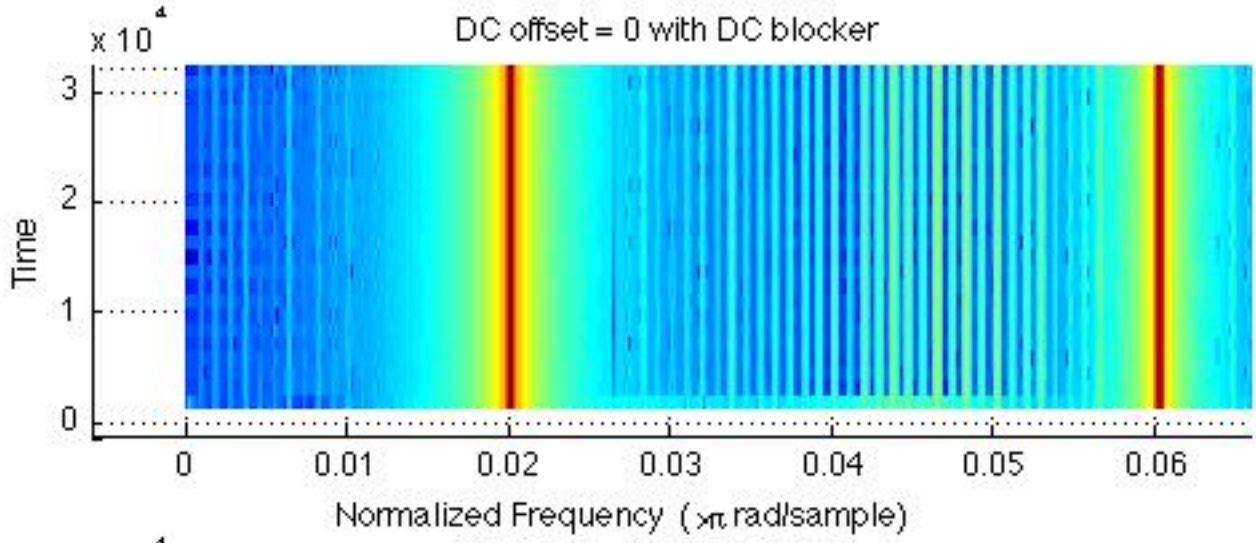

DC offset = 1 without DC blocker
x 10^4
Time
3
2.5
2
1.5
1
0.5
0
0.2
0.4
0.6
0.8
1
Normalized Frequency (×π rad/sample)

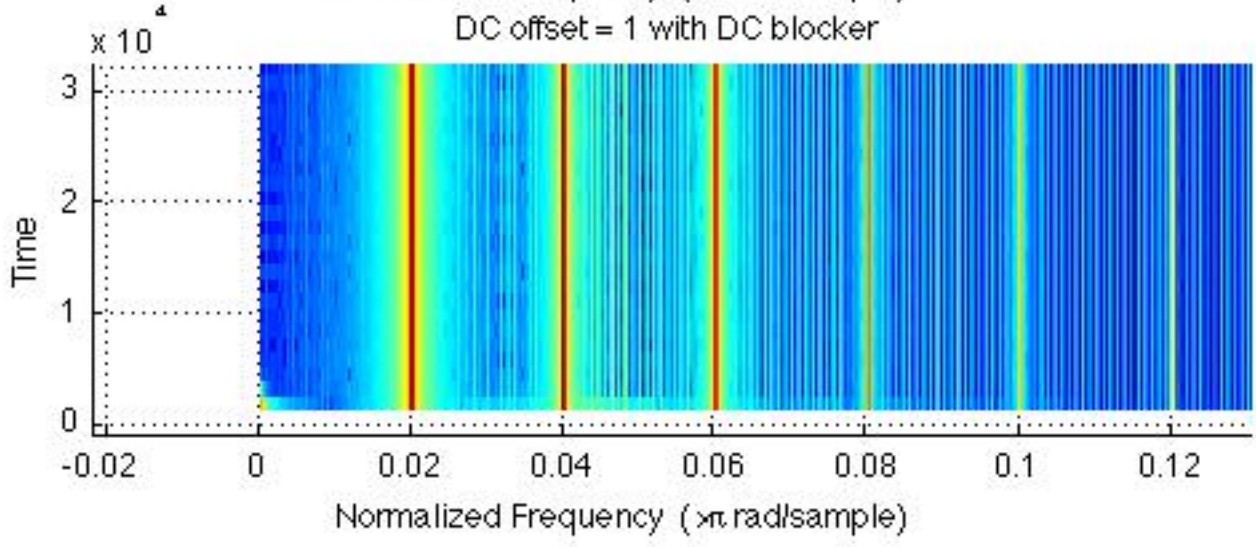

Impulse Response
Amplitude
1.2
1
0.8
0.6
0.4
0.2
0
-0.2
0
0.5
1
1.5
2
2.5
3
n (samples)
x 10^4



Published with MATLAB® 7.10


DC offset = 0 without DC blocker
x 10^4
Time
3
2.5
2
1.5
1
0.5
0
0.02
0.04
0.06
0.08
0.1
Normalized Frequency (×π rad/sample)
DC offset = 1 without DC blocker
x 10^4
Time
3
2
1
0
-0.02
-0.01
0
0.01
0.02
0.03
0.04
Normalized Frequency (×π rad/sample)


DC offset = 0 with DC blocker
x 10^4
Time
3
2
1
0
0
0.01
0.02
0.03
0.04
0.05
0.06
Normalized Frequency (×π rad/sample)


DC offset = 1 with DC blocker
x 10^4
Time
3
2
1
0
-0.02
0
0.02
0.04
0.06
0.08
0.1
0.12
Normalized Frequency (×π rad/sample)