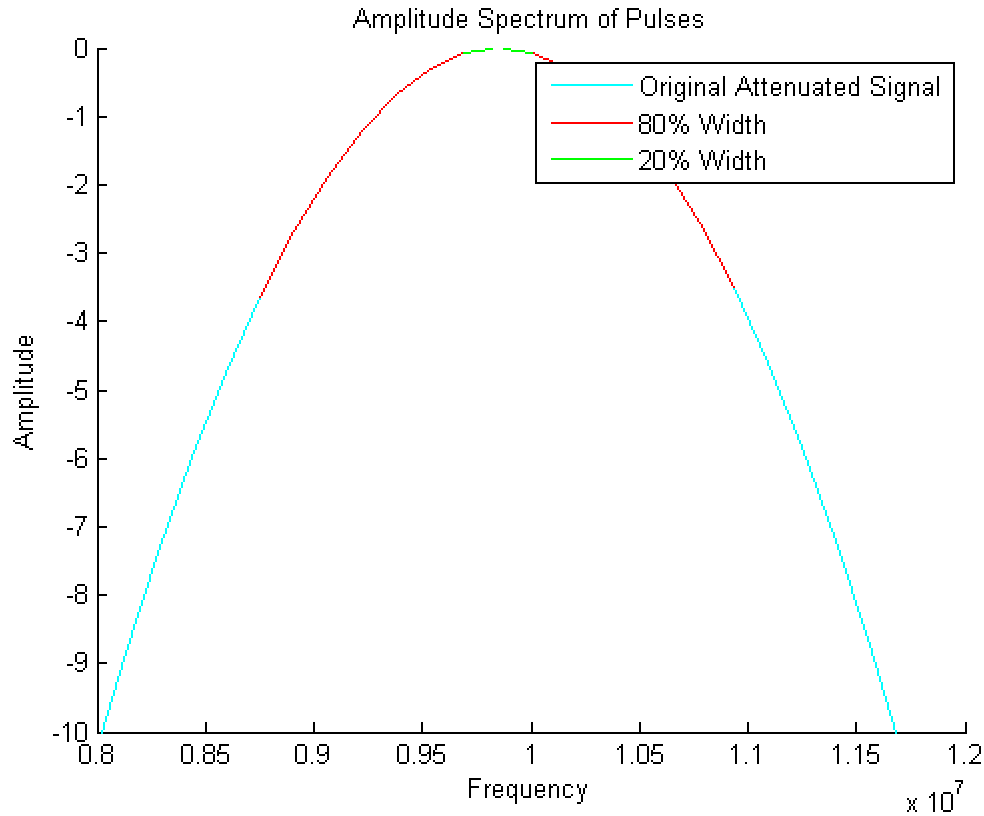
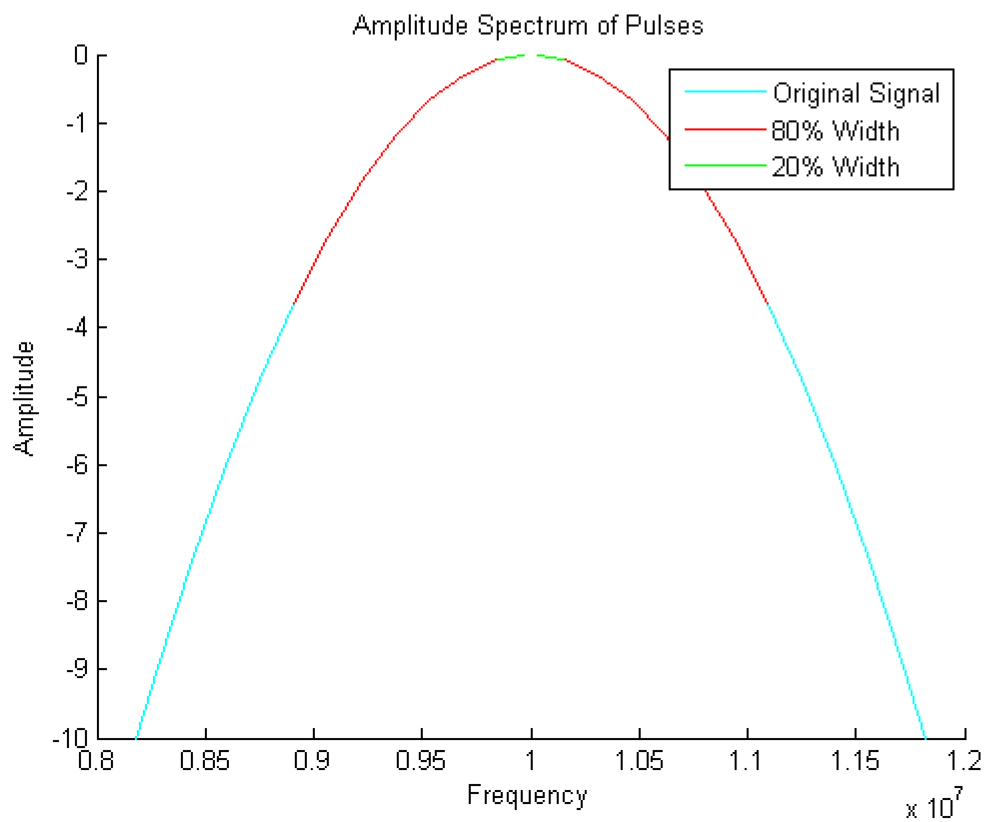
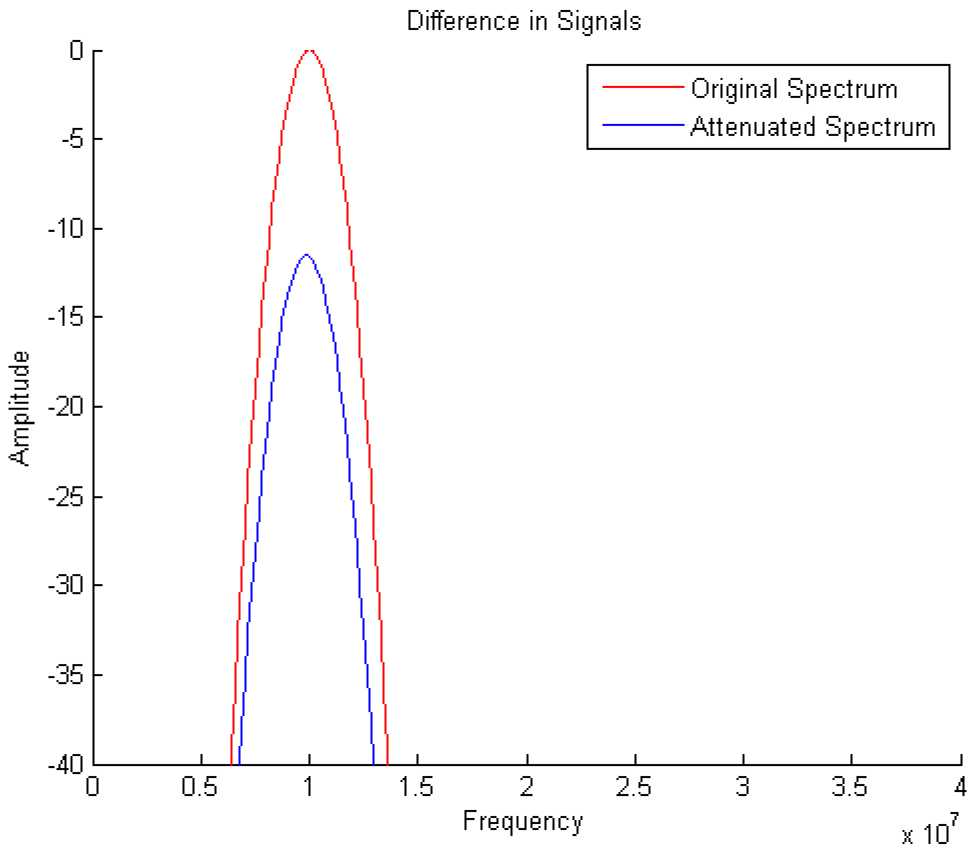
```
% Program to create array beam plot

clear all
clf
j=sqrt(-1);

vel=1540;                                    % Speed of sound - all units MKS
num_elems=64;                                % Number of elements
pitch=0.30e-3;                               % Array element pitch
fc=10e6;                                      % Center frequency
z_foc=50e-3;                                 % Range direction focal distance
theta_steer=45*pi/180;                       % Steer angle


fs=fc/64;                                    % Define a sampling frequency (not cri
f=[fs:fs:8*fc];                              % Define an adequate frequency range
w=2*pi*f;                                    % Angular frequency radians
ns=length(f);                                % Number of samples

tdel=1.0e-6;                                 % Use a fixed time offset so that base

bw=30;                                       % Fractional bandwidth as percent
sig=bw*fc/100;                               % Width of Gausian
gauss_pulse=exp(-pi*((f-fc)/sig).^2);        % Generate Gaussian pulse (frequency d
gauss_pulse=gauss_pulse.*exp(-j*w*tdel);     % Apply timed delay so 0 for t<0
gauss_pulse_db = (20*log10(abs(gauss_pulse)./max((gauss_pulse))));

% Attenuate the signal
k_gauss_pulse = (gauss_pulse) ./ (.38/10^6.*(f));
k_gauss_pulse_db = (20*log10(abs(k_gauss_pulse)./max((k_gauss_pulse))));
k_gauss_pulse_db_normalizedOrig = (20*log10(abs(k_gauss_pulse)./max((gauss_pulse))

% Find the -6 dB range of the attenuated signal.
mask = k_gauss_pulse_db > (-6);
mask = mask*1;
k_gauss_pulse_6db_slice = k_gauss_pulse_db .*mask;

p80_width = 0.8*sum(mask);
[maxVal k_arrayIndex] = max(k_gauss_pulse);
newIndex80 = (k_arrayIndex - floor(0.5*p80_width)):1:(k_arrayIndex + floor(0.5*p80

p20_width = 0.2*sum(mask);
[maxVal k_arrayIndex] = max(k_gauss_pulse);
newIndex20 = (k_arrayIndex - floor(0.5*p20_width)):1:(k_arrayIndex + floor(0.5*p20

% Plot the spectrums of the attenuated signal
figure(1)
hold on
plot(f, k_gauss_pulse_db, 'c')
plot(f(newIndex80), k_gauss_pulse_db(newIndex80) , 'r')
plot(f(newIndex20), k_gauss_pulse_db(newIndex20) , 'g')
axis([8e6 12e6 -10 0])
title('Amplitude Spectrum of Pulses')
xlabel('Frequency')
ylabel('Amplitude')
legend('Original Attenuated Signal', '80% Width', '20% Width');

% Find the -6 dB range of the original signal.
mask = gauss_pulse_db > (-6);
mask = mask*1;
gauss_pulse_6db_slice = gauss_pulse_db .*mask;

p80_width = 0.8*sum(mask);
```

```
[maxVal arrayIndex] = max(gauss_pulse);
newIndex80 = (arrayIndex - floor(0.5*p80_width)):1:(arrayIndex + floor(0.5*p80_wid

p20_width = 0.2*sum(mask);
[maxVal arrayIndex] = max(gauss_pulse);
newIndex20 = (arrayIndex - floor(0.5*p20_width)):1:(arrayIndex + floor(0.5*p20_wid

% Plot the spectrums of the un-attenuated signal
figure(2)
hold on
plot(f, gauss_pulse_db, 'c')
plot(f(newIndex80), gauss_pulse_db(newIndex80) , 'r')
plot(f(newIndex20), gauss_pulse_db(newIndex20) , 'g')
axis([8e6 12e6 -10 0])
title('Amplitude Spectrum of Pulses')
xlabel('Frequency')
ylabel('Amplitude')
legend('Original Signal', '80% Width', '20% Width');

figure(3)
hold on
plot(f, gauss_pulse_db, 'r')
plot(f, k_gauss_pulse_db_normalizedOrig, 'b')
axis([0e6 40e6 -40 0])
title('Amplitude Spectrum of Pulses')
xlabel('Frequency')
ylabel('Amplitude')
legend('Original Spectrum','Attenuated Spectrum')
title('Difference in Signals');
peakFreqDifference = f(arrayIndex) - f(k_arrayIndex)


Warning: Imaginary parts of complex X and/or Y arguments ignored
Warning: Imaginary parts of complex X and/or Y arguments ignored
Warning: Imaginary parts of complex X and/or Y arguments ignored
Warning: Imaginary parts of complex X and/or Y arguments ignored
Warning: Imaginary parts of complex X and/or Y arguments ignored
Warning: Imaginary parts of complex X and/or Y arguments ignored
Warning: Imaginary parts of complex X and/or Y arguments ignored
Warning: Imaginary parts of complex X and/or Y arguments ignored

peakFreqDifference =

      156250
```


Amplitude Spectrum of Pulses
Original Attenuated Signal
80% Width
20% Width
Amplitude
Frequency
x 10^7
0
-1
-2
-3
-4
-5
-6
-7
-8
-9
-10
0.8
0.85
0.9
0.95
1
1.05
1.1
1.15
1.2


Amplitude Spectrum of Pulses
Original Signal
80% Width
20% Width
Amplitude
Frequency
x 10^7
0
-1
-2
-3
-4
-5
-6
-7
-8
-9
-10
0.8
0.85
0.9
0.95
1
1.05
1.1
1.15
1.2



Published with MATLAB® 7.11