

```

% DSP LAB 5
% design diff types of filters used in loudspeaker equalizer
% define parameters

fs = 44100;
f0 = 5000;
q = 2;
gain_db = 6;

[x, fs_file] = audioread('/MATLAB Drive/6_11.wav');
if fs_file ~= fs
    x = resample(x, fs, fs_file);
end

sound(x, fs);
subplot(2, 2, 1);
plot(x);
title('Original Sound');

subplot(2, 2, 2);
z = abs(fft(x));
plot(z);
title('Frequency Spectrum of Original Sound');

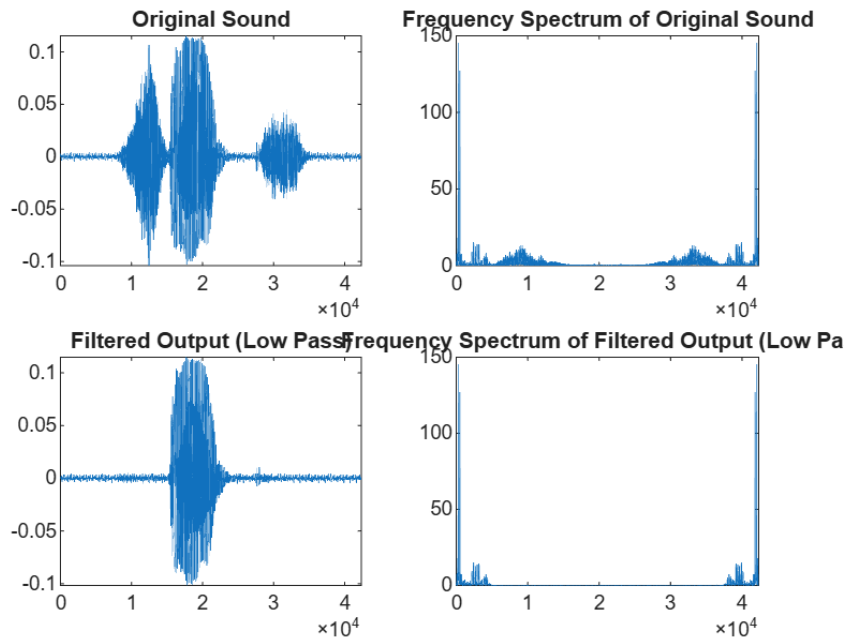
% Design a low pass filter
lp_filter = designfilt('lowpassfir', ...
    'FilterOrder', 64, ...
    'CutoffFrequency', f0, ...
    'SampleRate', fs);

y_filtered = filter(lp_filter, x);
c = 10 * y_filtered;

pause(1);
sound(c, fs);
subplot(2, 2, 3);
plot(y_filtered);
title('Filtered Output (Low Pass)');

z1 = abs(fft(y_filtered));
subplot(2, 2, 4);
plot(z1);
title('Frequency Spectrum of Filtered Output (Low Pass)');

```



```
% Design a high pass filter
hp_filter = designfilt('highpassfir', ...
    'FilterOrder', 64, ...
    'CutoffFrequency', f0, ...
    'SampleRate', fs);

y_filtered = filter(hp_filter, x);
c = 10 * y_filtered;

pause(1);
sound(c, fs);
figure;
subplot(2, 2, 1);
plot(y_filtered);
title('Filtered Output (High Pass)');

z1 = abs(fft(y_filtered));
subplot(2, 2, 2);
plot(z1);
title('Frequency Spectrum of Filtered Output (High Pass)');

% Design a band pass filter
bandpass_filter = designfilt('bandpassfir', ...
    'FilterOrder', 64, ...
    'CutoffFrequency1', 1000, ...
    'CutoffFrequency2', 8000, ...
    'SampleRate', fs);

y_filtered = filter(bandpass_filter, x);
c = 10 * y_filtered;

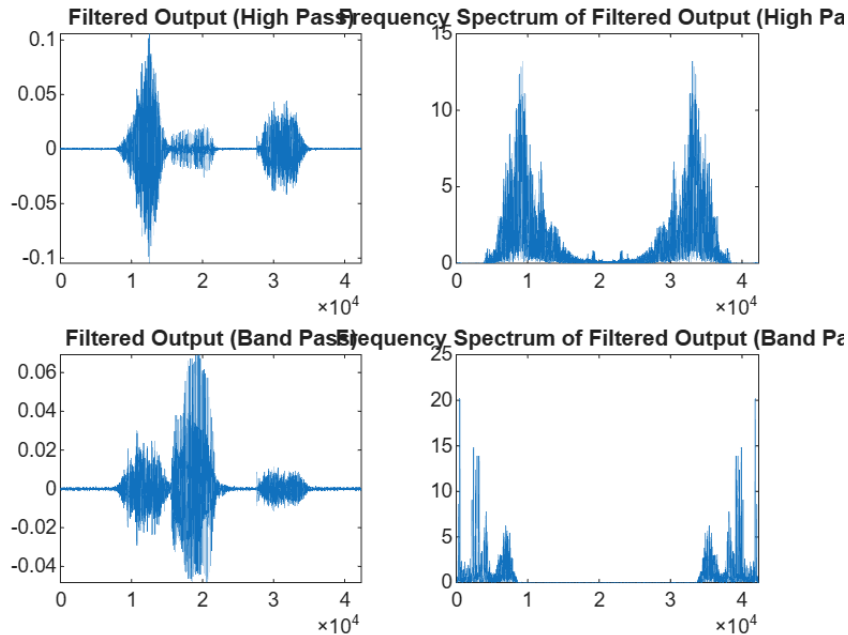
pause(1);
```

```

sound(c, fs);
subplot(2, 2, 3);
plot(y_filtered);
title('Filtered Output (Band Pass)');

z1 = abs(fft(y_filtered));
subplot(2, 2, 4);
plot(z1);
title('Frequency Spectrum of Filtered Output (Band Pass)');

```



```

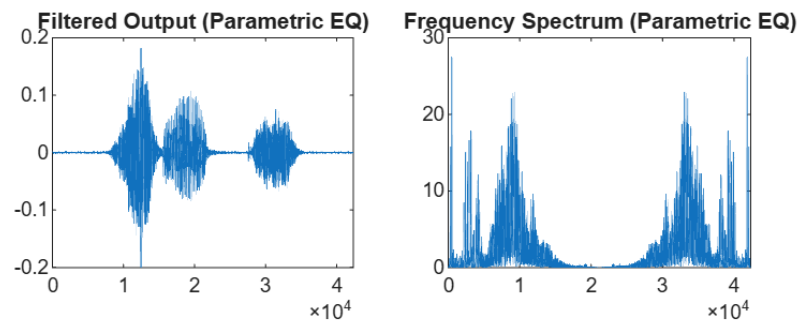
% Parametric equalizer filter
gain = 10^(gain_db/20);
b_gain = [1 0 -1];
a_gain = [1 (1/(gain*q)) (1/(gain*q))];
eq_fil = dfilt.df2(conv(b_gain, a_gain));

y_filtered = filter(eq_fil, x);
c = 10 * y_filtered;

pause(1);
sound(c, fs);
figure;
subplot(2, 2, 1);
plot(y_filtered);
title('Filtered Output (Parametric EQ)');

z1 = abs(fft(y_filtered));
subplot(2, 2, 2);
plot(z1);
title('Frequency Spectrum (Parametric EQ)');

```

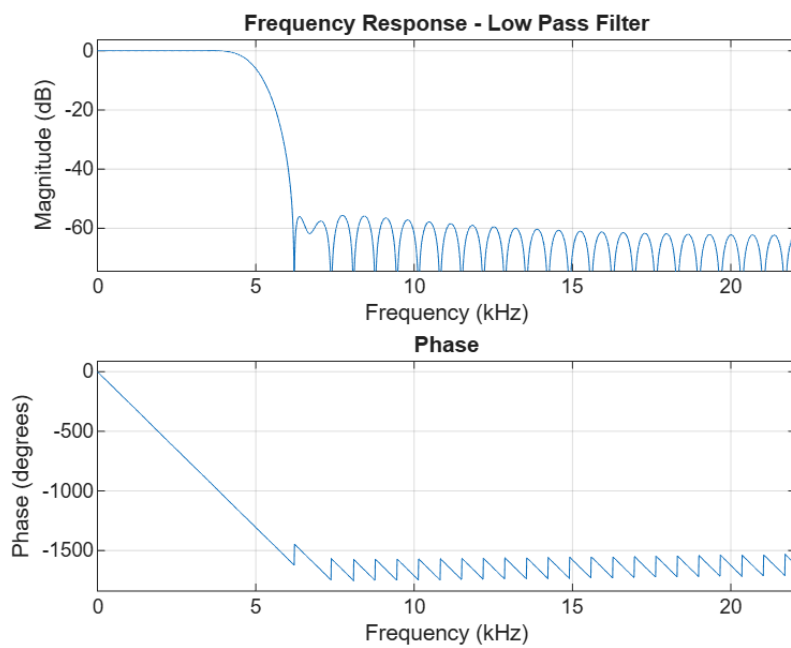


% Frequency responses

figure;

freqz(lp_filter, 2048, fs);

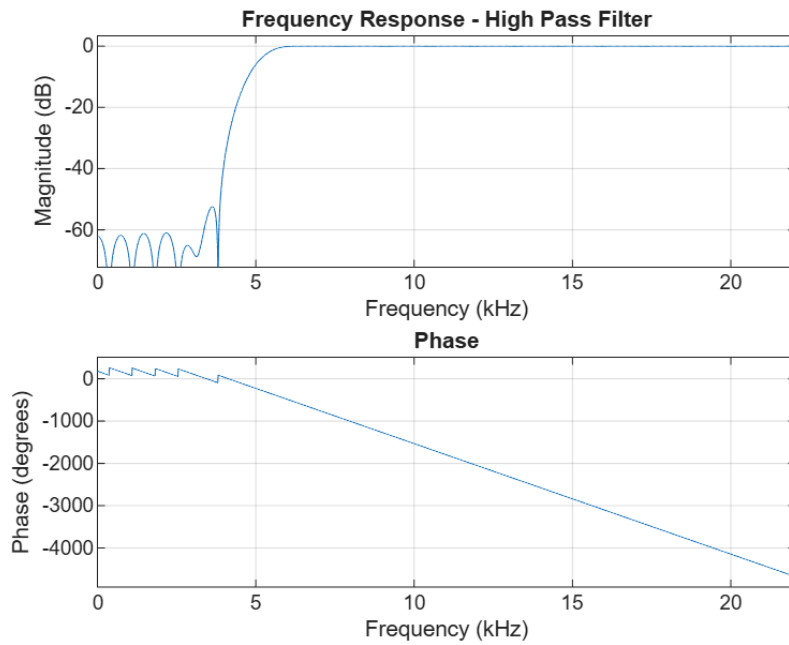
title('Frequency Response - Low Pass Filter');



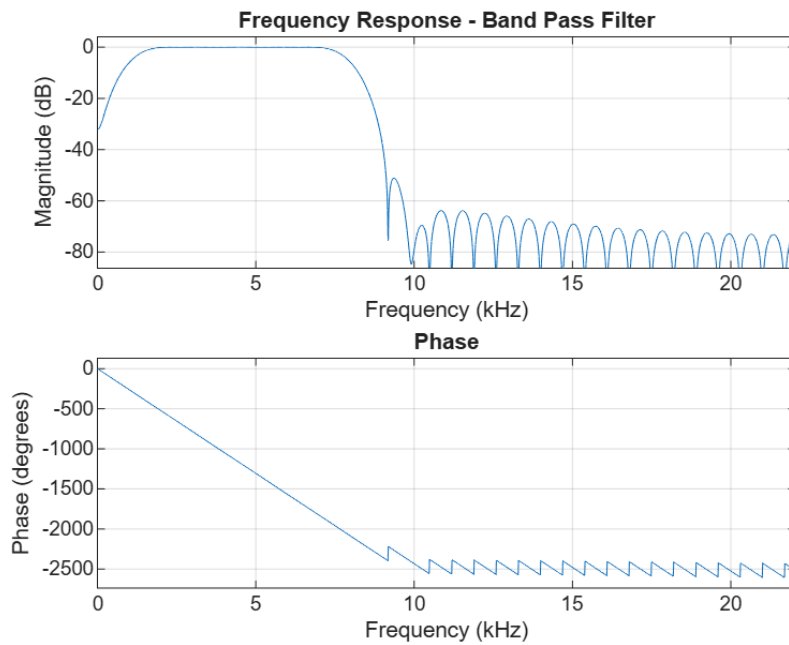
figure;

freqz(hp_filter, 2048, fs);

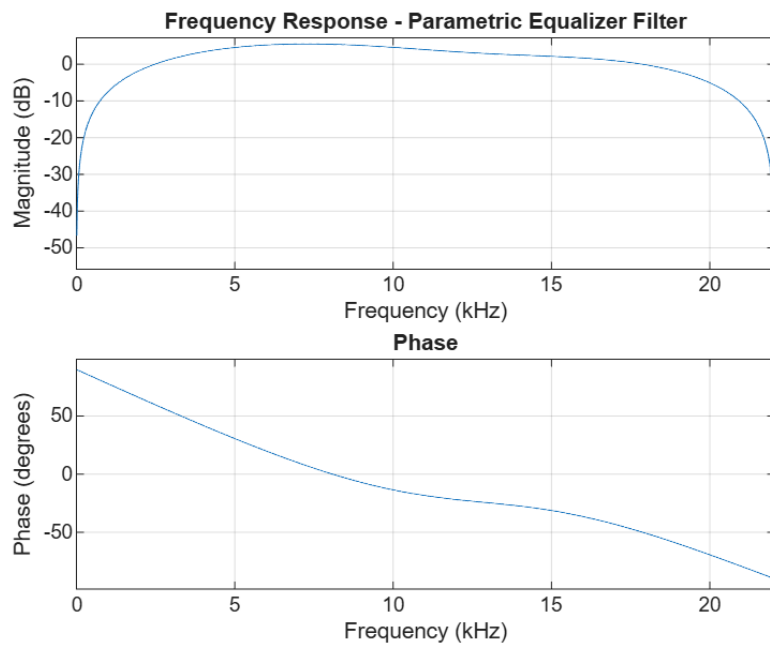
title('Frequency Response - High Pass Filter');



```
figure;
freqz(bandpass_filter, 2048, fs);
title('Frequency Response - Band Pass Filter');
```



```
figure;
freqz(eq_fil, 2048, fs);
title('Frequency Response - Parametric Equalizer Filter');
```



```
disp('All filters executed successfully.');
```

All filters executed successfully.