

# DIGITAL SIGNAL PROCESSING

## FINAL PROJECT: AUDIO EQUALIZER

Under the supervision of:

Dr. Ahmed Eltrass

Eng. Hossam Hassan

Eng. Esraa Ghazy

Eng. Mohamed Essam

**Names:**

Saad El Dine Ahmed

Morougue Ghazal

**IDs**

7370

7524

**Group:**

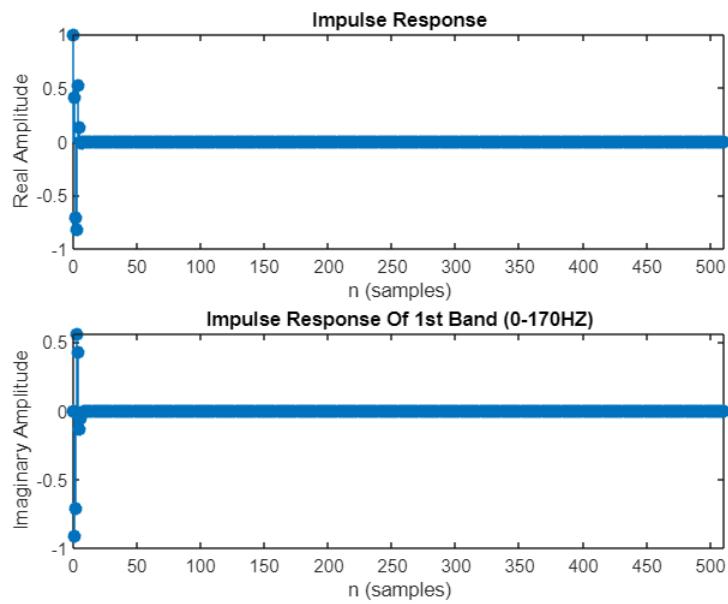
2

**Lab:**

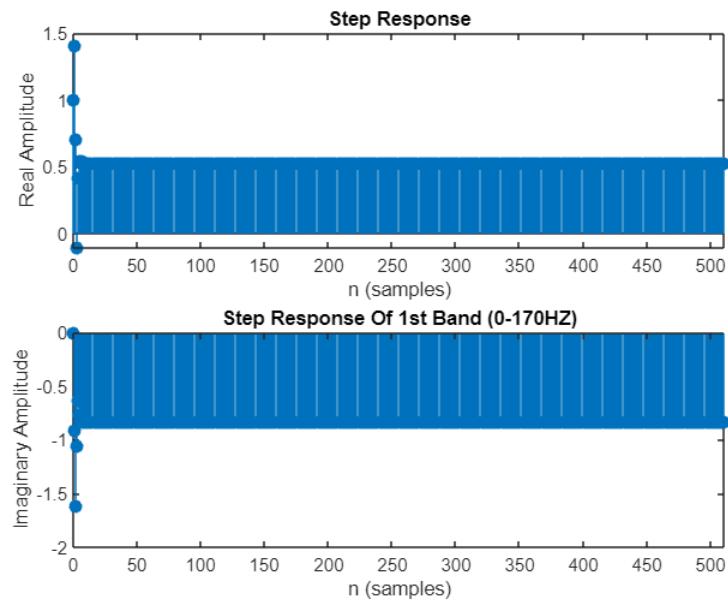
2

# ANALYSIS

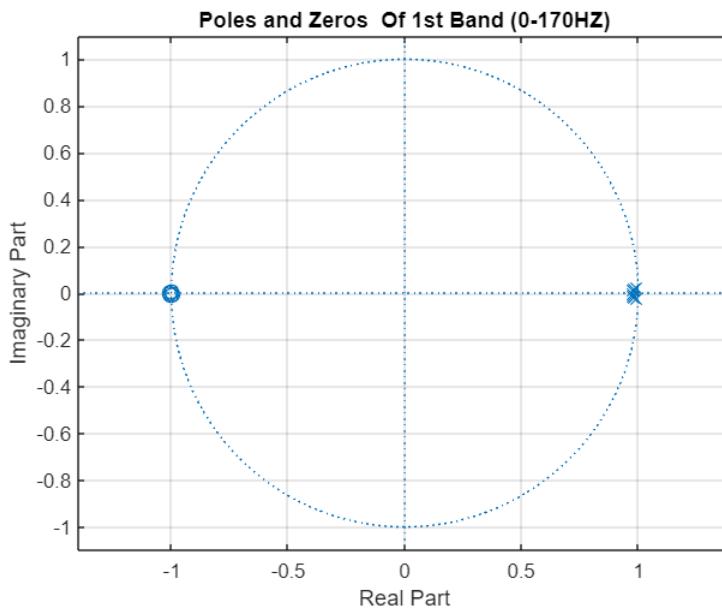
```
Fs = 44100;  
%%%%%%% Calculate Fs Normalized %%%%%%  
F_normalized = Fs / 2;  
%%%%%%% Calculate Frequencies Normalized %%%%%%  
f1 = 170 / F_normalized;  
f2 = [170 / F_normalized, 310 / F_normalized];  
f3 = [310 / F_normalized, 600 / F_normalized];  
f4 = [600 / F_normalized, 1000 / F_normalized];  
f5 = [1000 / F_normalized, 3000 / F_normalized];  
f6 = [3000 / F_normalized, 6000 / F_normalized];  
f7 = [6000 / F_normalized, 12000 / F_normalized];  
f8 = [12000 / F_normalized, 14000 / F_normalized];  
f9 = [14000 / F_normalized, 16000 / F_normalized];  
%  
  
%%%%%%% Define IIR Butter Worth Filters %%%%%%  
[b1, a1] = butter( 7, f1, 'low');  
[b2, a2] = butter( 3, f2, 'bandpass');  
[b3, a3] = butter( 4, f3, 'bandpass');  
[b4, a4] = butter( 5, f4, 'bandpass');  
[b5, a5] = butter( 7, f5, 'bandpass');  
[b6, a6] = butter( 7, f6, 'bandpass');  
[b7, a7] = butter(14, f7, 'bandpass');  
[b8, a8] = butter(14, f8, 'bandpass');  
[b9, a9] = butter(14, f9, 'bandpass');  
  
%%%%%%% Get Transfer Function Of Each IIR Filter %%%%%%  
[HD1, wcd1] = freqz(b1,a1);  
[HD2, wcd2] = freqz(b2,a2);  
[HD3, wcd3] = freqz(b3,a3);  
[HD4, wcd4] = freqz(b4,a4);  
[HD5, wcd5] = freqz(b5,a5);  
[HD6, wcd6] = freqz(b6,a6);  
[HD7, wcd7] = freqz(b7,a7);  
[HD8, wcd8] = freqz(b8,a8);  
[HD9, wcd9] = freqz(b9,a9);  
  
%%%%%%% Analysis Of 1st IIR Filter %%%%%%  
figure;  
impz(HD1);  
title('Impulse Response Of 1st Band (0-170HZ)');
```



```
figure;
stepz(HD1);
title('Step Response Of 1st Band (0-170HZ)');
```



```
figure;
zplane(b1, a1);
grid;
title('Poles and Zeros Of 1st Band (0-170HZ)');
```



```

figure;
subplot(2, 1, 1);
magnitude = 20*log10(abs(HD1));
gain = max(magnitude);
fprintf('IIR Filter 1 Gain: %.2f dB\n', gain);

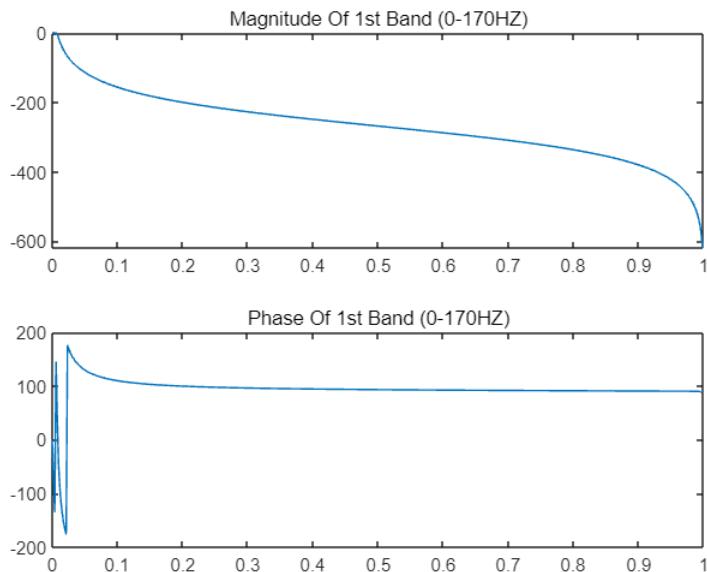
```

IIR Filter 1 Gain: 0.00 dB

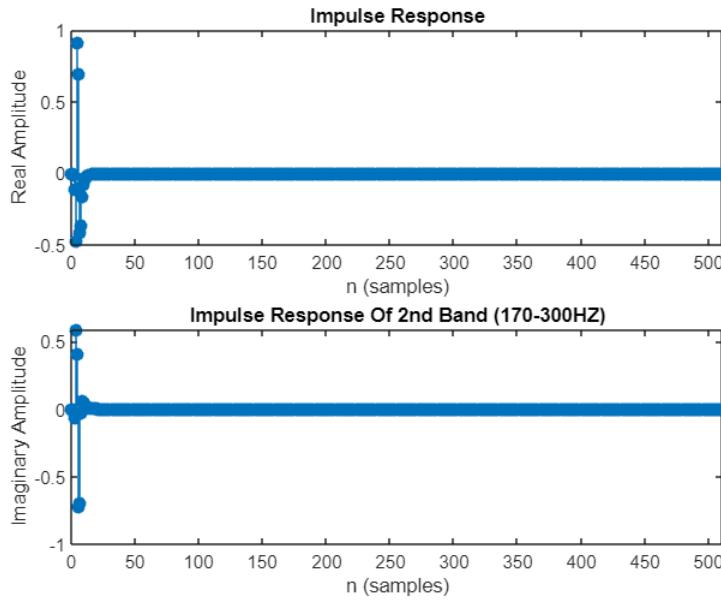
```

plot(wcd1/pi, magnitude);
subtitle('Magnitude Of 1st Band (0-170HZ)');
subplot(2, 1, 2);
plot(wcd1/pi, rad2deg(angle(HD1)));
subtitle('Phase Of 1st Band (0-170HZ)');

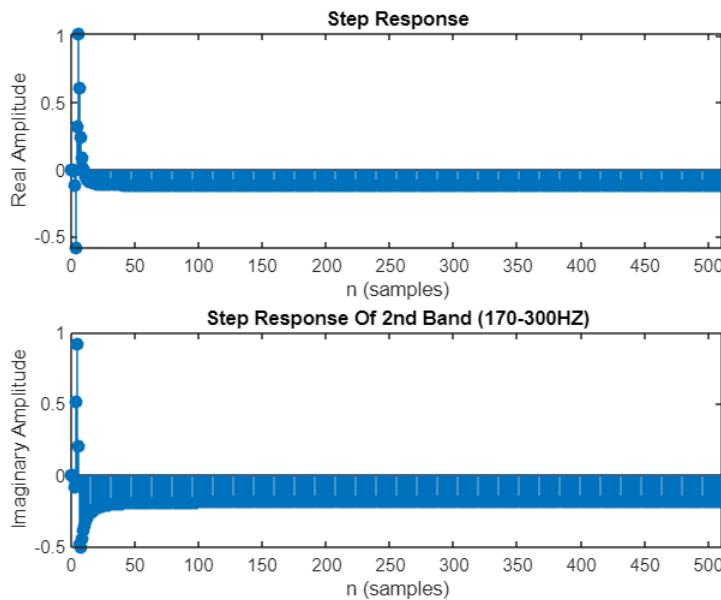
```



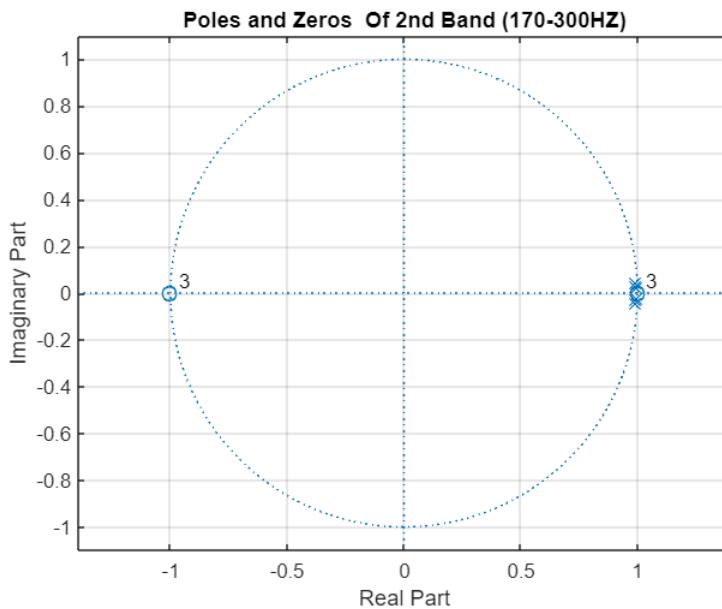
```
%%%%%%%%%%%%% Analysis Of 2nd IIR Filter %%%%%%
figure;
impz(HD2);
title('Impulse Response Of 2nd Band (170-300HZ)');
```



```
figure;
stepz(HD2);
title('Step Response Of 2nd Band (170-300HZ)');
```



```
figure;
zplane(b2, a2);
grid;
title('Poles and Zeros Of 2nd Band (170-300HZ)');
```



```

figure;
subplot(2, 1, 1);
magnitude = 20*log10(abs(HD2));
gain = max(magnitude);
fprintf('IIR Filter 2 Gain: %.2f dB\n', abs(gain));

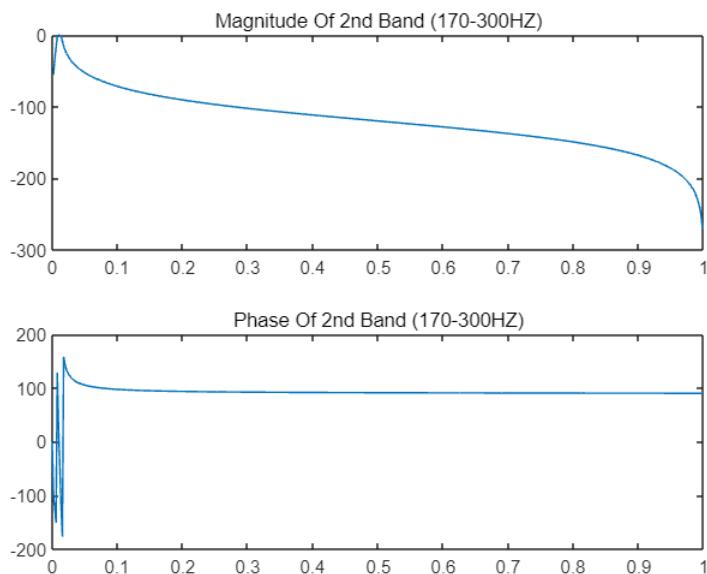
```

IIR Filter 2 Gain: 0.00 dB

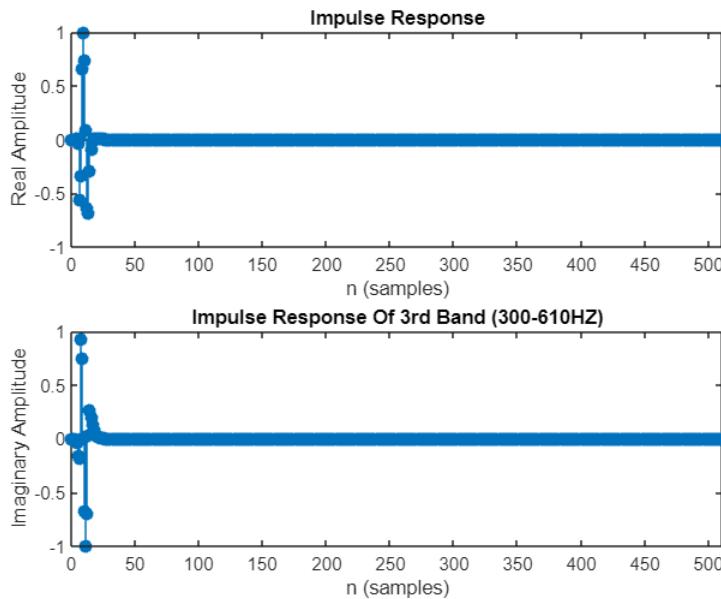
```

plot(wcd2/pi, magnitude);
subtitle('Magnitude Of 2nd Band (170-300HZ)');
subplot(2, 1, 2);
plot(wcd2/pi, rad2deg(angle(HD2)));
subtitle('Phase Of 2nd Band (170-300HZ)');

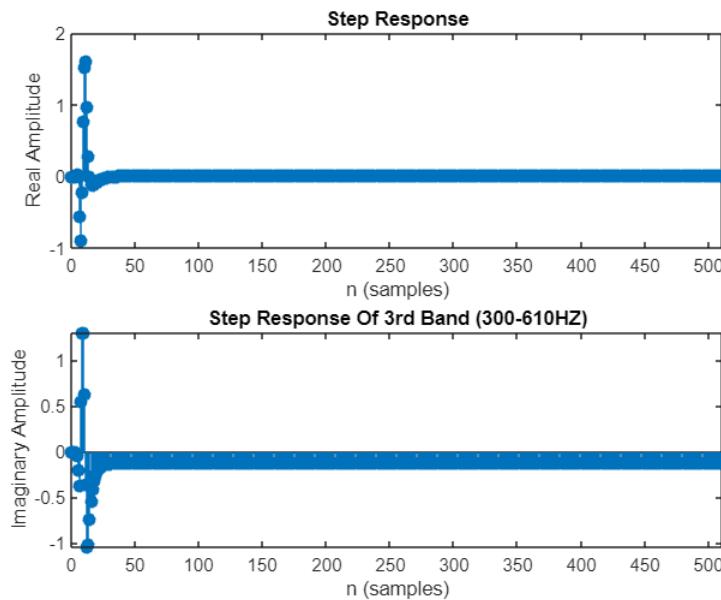
```



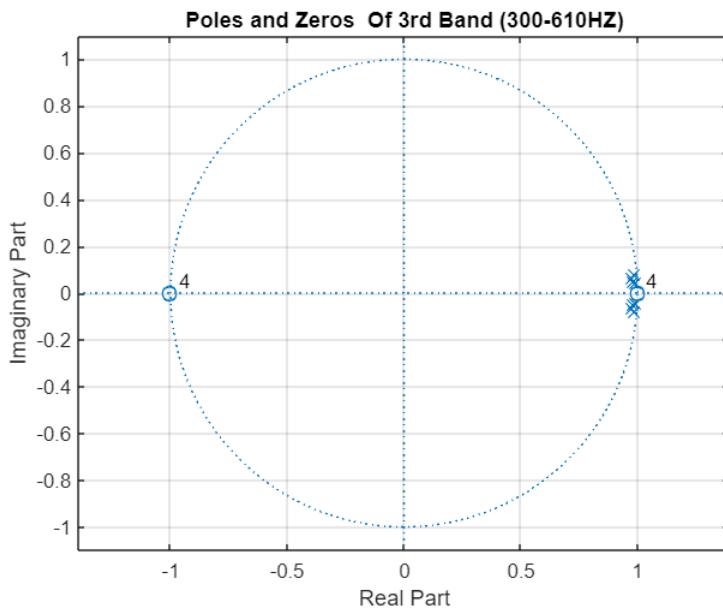
```
%%%%%%%%%%%%% Analysis Of 3rd IIR Filter %%%%%%
figure;
impz(HD3);
title('Impulse Response Of 3rd Band (300-610HZ)');
```



```
figure;
stepz(HD3);
title('Step Response Of 3rd Band (300-610HZ)');
```



```
figure;
zplane(b3, a3);
grid;
title('Poles and Zeros Of 3rd Band (300-610HZ)');
```



```

figure;
subplot(2, 1, 1);
magnitude = 20*log10(abs(HD3));
gain = max(magnitude);
fprintf('IIR Filter 3 Gain: %.2f dB\n', gain);

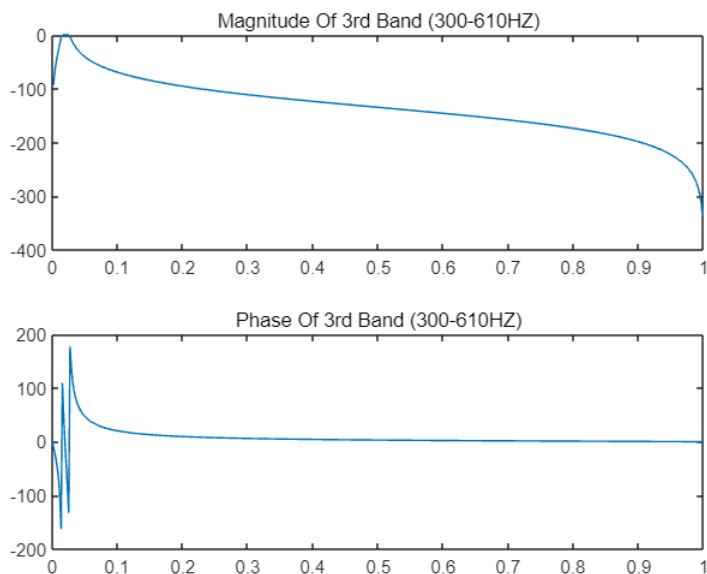
```

IIR Filter 3 Gain: 0.00 dB

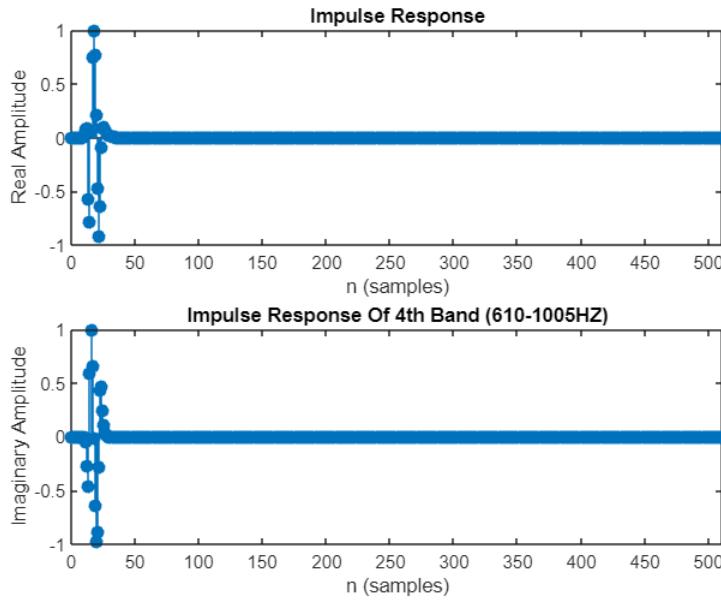
```

plot(wcd3/pi, magnitude);
subtitle('Magnitude Of 3rd Band (300-610HZ)');
subplot(2, 1, 2);
plot(wcd3/pi, rad2deg(angle(HD3)));
subtitle('Phase Of 3rd Band (300-610HZ)');

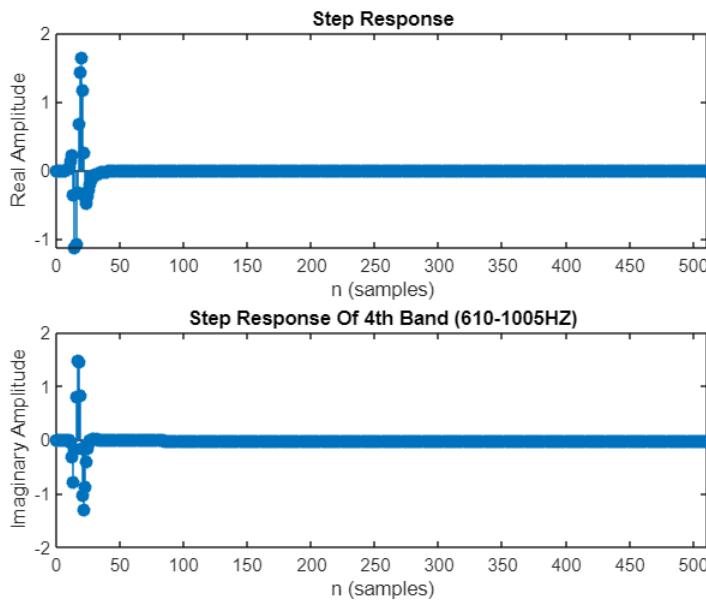
```



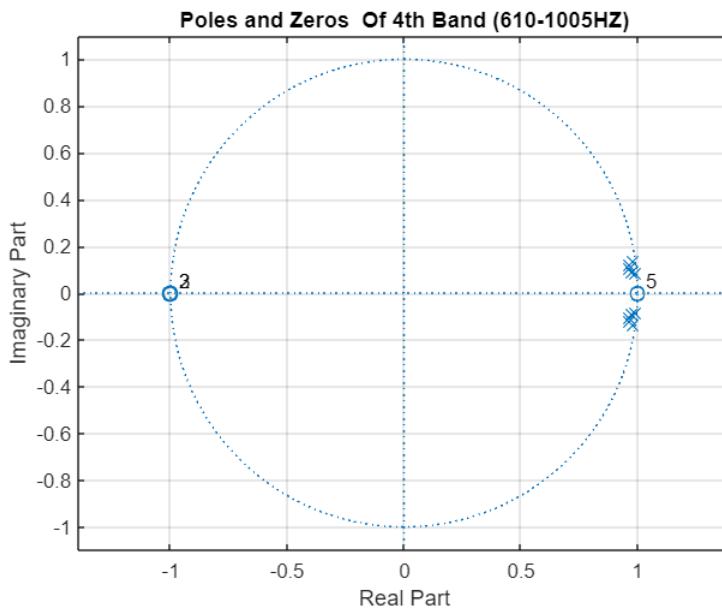
```
%%%%%%%%%%%%% Analysis Of 4th IIR Filter %%%%%%
figure;
impz(HD4);
title('Impulse Response Of 4th Band (610-1005HZ)');
```



```
figure;
stepz(HD4);
title('Step Response Of 4th Band (610-1005HZ)');
```



```
figure;
zplane(b4, a4);
grid;
title('Poles and Zeros Of 4th Band (610-1005HZ)');
```



```

figure;
subplot(2, 1, 1);
magnitude = 20*log10(abs(HD4));
gain = max(magnitude);
fprintf('IIR Filter 4 Gain: %.2f dB\n', gain);

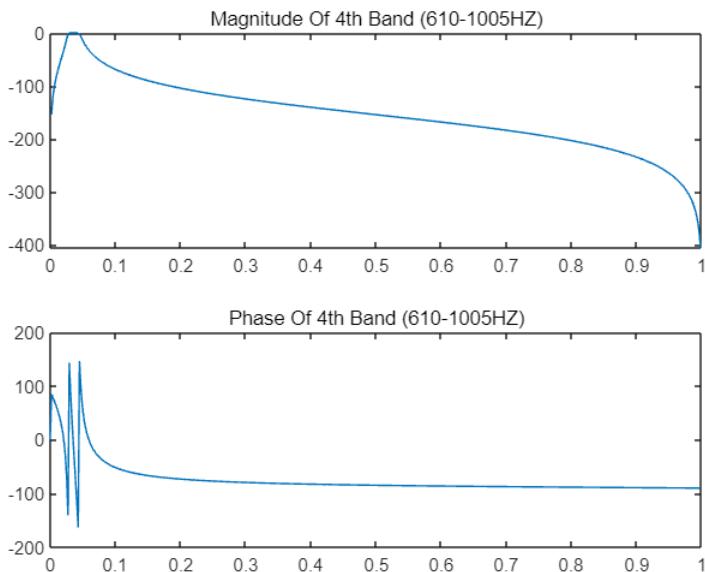
```

IIR Filter 4 Gain: 0.03 dB

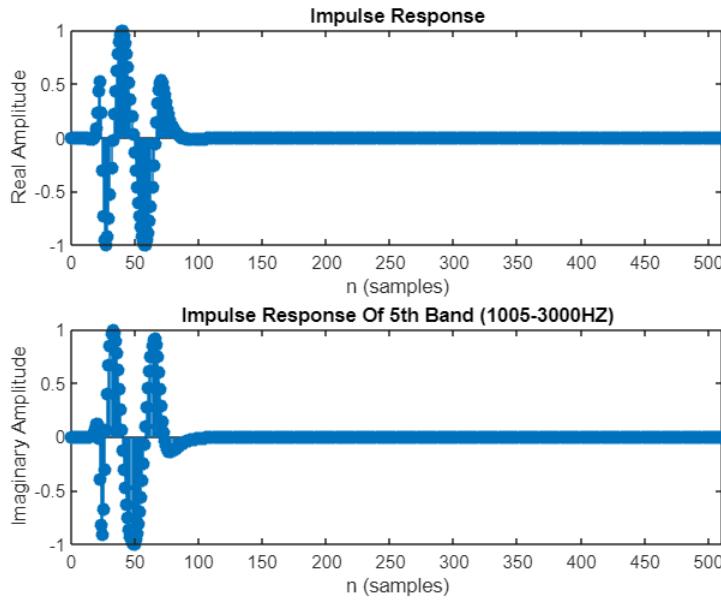
```

plot(wcd4/pi, magnitude);
subtitle('Magnitude Of 4th Band (610-1005HZ)');
subplot(2, 1, 2);
plot(wcd4/pi, rad2deg(angle(HD4)));
subtitle('Phase Of 4th Band (610-1005HZ)');

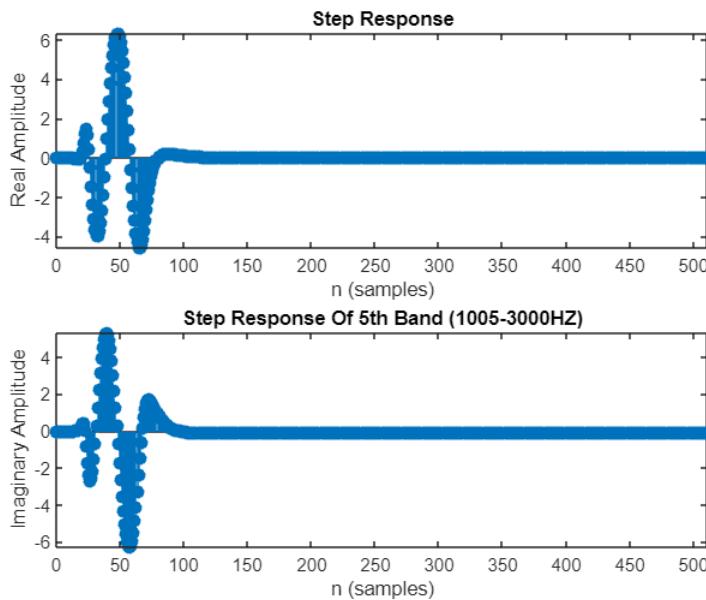
```



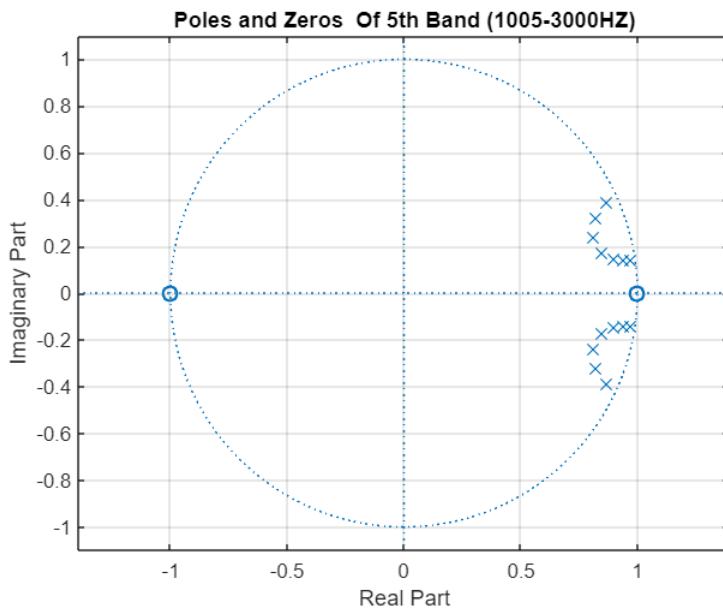
```
%%%%%%%%%%%%% Analysis Of 5th IIR Filter %%%%%%
figure;
impz(HD5);
title('Impulse Response Of 5th Band (1005-3000HZ)');
```



```
figure;
stepz(HD5);
title('Step Response Of 5th Band (1005-3000HZ)');
```



```
figure;
zplane(b5, a5);
grid;
title('Poles and Zeros Of 5th Band (1005-3000HZ)');
```



```

figure;
subplot(2, 1, 1);
magnitude = 20*log10(abs(HD5));
gain = max(magnitude);
fprintf('IIR Filter 5 Gain: %.2f dB\n', gain);

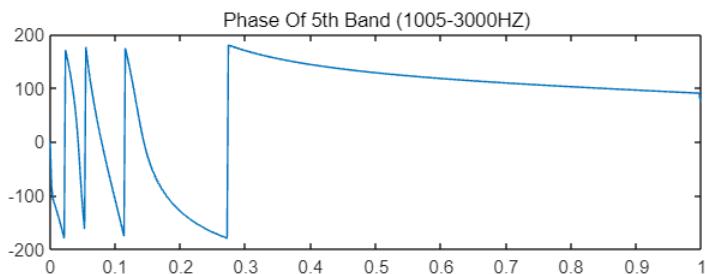
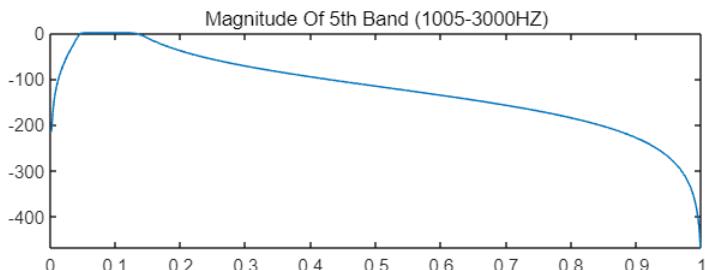
```

IIR Filter 5 Gain: 0.00 dB

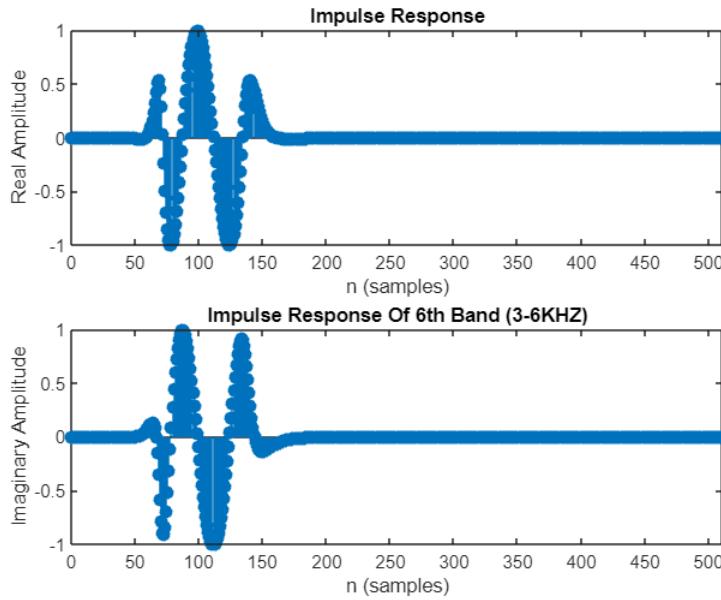
```

plot(wcd5/pi, magnitude);
subtitle('Magnitude Of 5th Band (1005-3000HZ)');
subplot(2, 1, 2);
plot(wcd5/pi, rad2deg(angle(HD5)));
subtitle('Phase Of 5th Band (1005-3000HZ)');

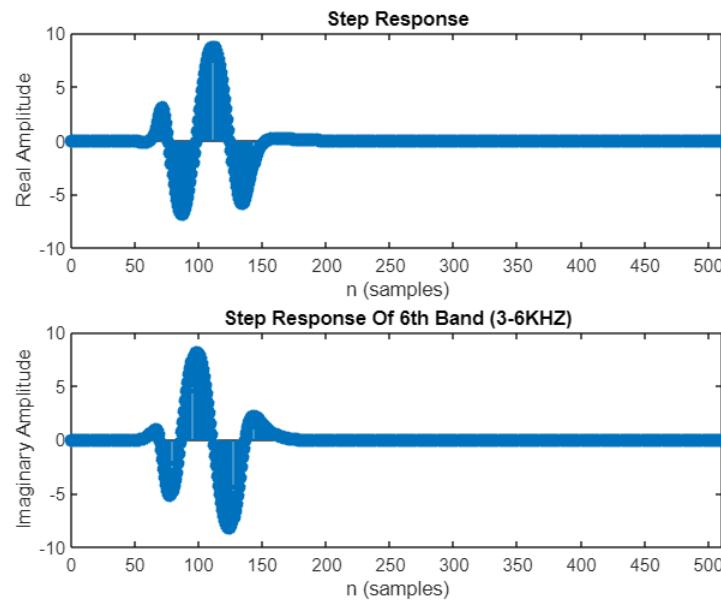
```



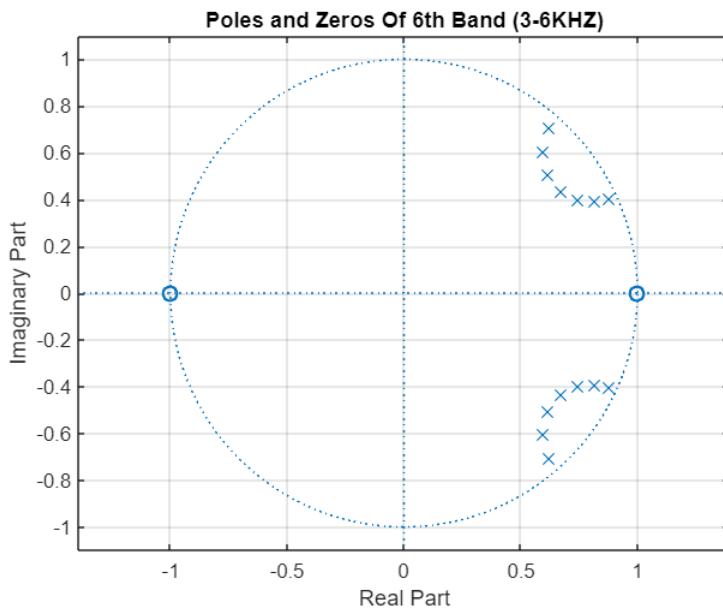
```
%%%%%%%%%%%%% Analysis Of 6th IIR Filter %%%%%%
figure;
impz(HD6);
title('Impulse Response Of 6th Band (3-6KHZ)');
```



```
figure;
stepz(HD6);
title('Step Response Of 6th Band (3-6KHZ)');
```



```
figure;
zplane(b6, a6);
grid;
title('Poles and Zeros Of 6th Band (3-6KHZ)');
```



```

figure;
subplot(2, 1, 1);
magnitude = 20*log10(abs(HD6));
gain = max(magnitude);
fprintf('IIR Filter 6 Gain: %.2f dB\n', gain);

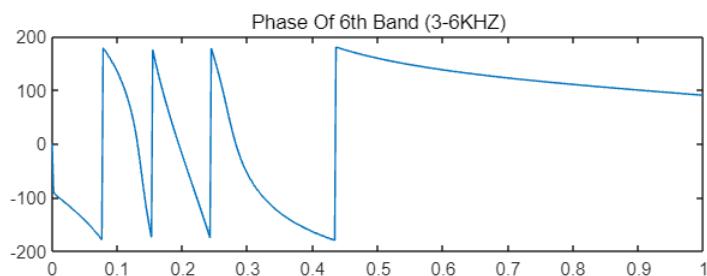
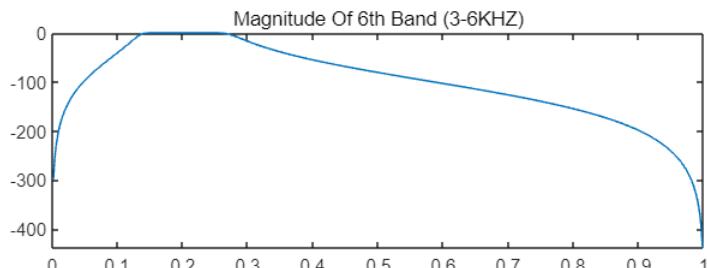
```

IIR Filter 6 Gain: 0.00 dB

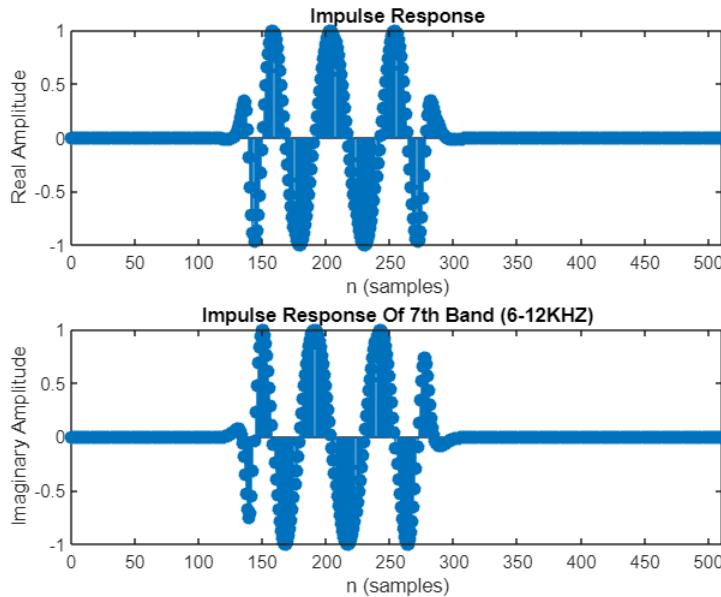
```

plot(wcd6/pi, magnitude);
subtitle('Magnitude Of 6th Band (3-6KHZ)');
subplot(2, 1, 2);
plot(wcd6/pi, rad2deg(angle(HD6)));
subtitle('Phase Of 6th Band (3-6KHZ)');

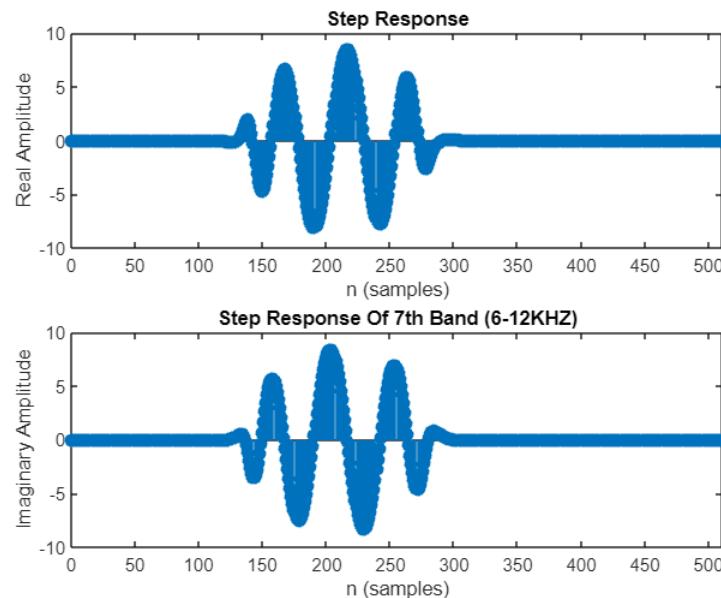
```



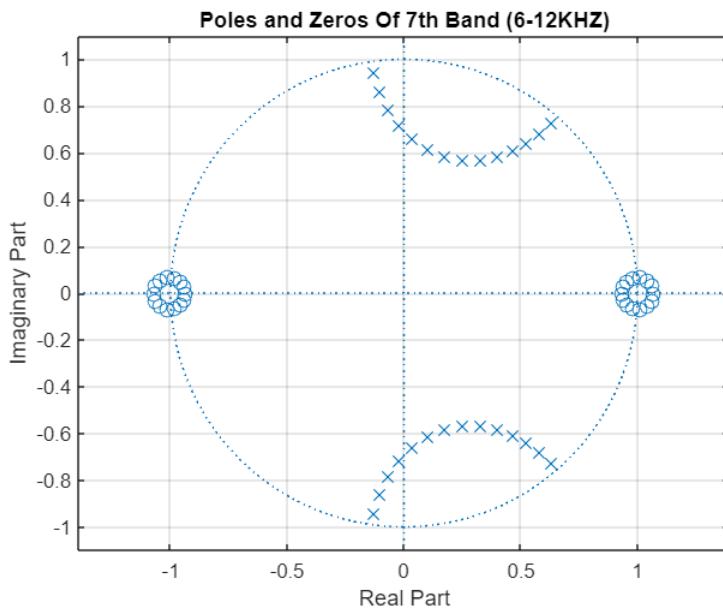
```
%%%%%%%%%%%%% Analysis Of 7th IIR Filter %%%%%%
figure;
impz(HD7);
title('Impulse Response Of 7th Band (6-12KHZ)');
```



```
figure;
stepz(HD7);
title('Step Response Of 7th Band (6-12KHZ)');
```



```
figure;
zplane(b7, a7);
grid;
title('Poles and Zeros Of 7th Band (6-12KHZ)'');
```



```

figure;
subplot(2, 1, 1);
magnitude = 20*log10(abs(HD7));
gain = max(magnitude);
fprintf('IIR Filter 7 Gain: %.2f dB\n', gain);

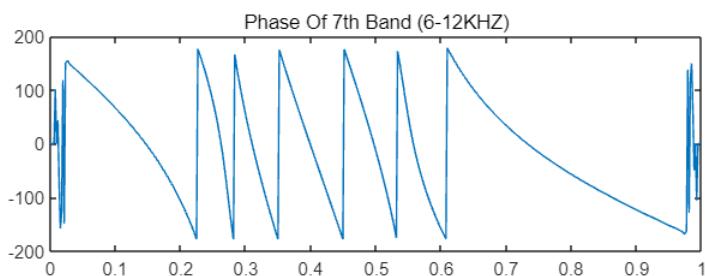
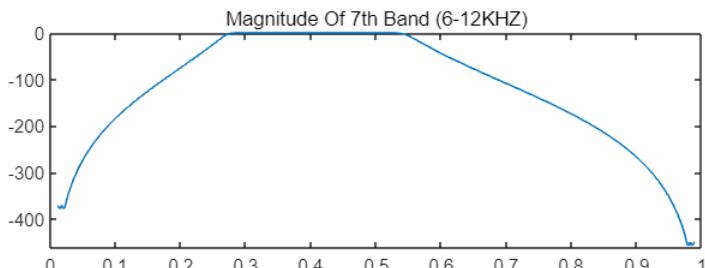
```

IIR Filter 7 Gain: 0.00 dB

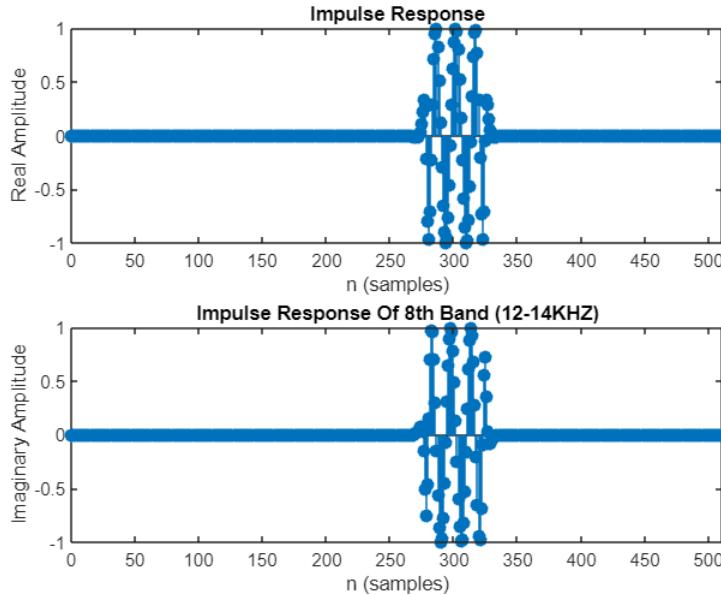
```

plot(wcd7/pi, magnitude);
subtitle('Magnitude Of 7th Band (6-12KHZ)');
subplot(2, 1, 2);
plot(wcd7/pi, rad2deg(angle(HD7)));
subtitle('Phase Of 7th Band (6-12KHZ)');

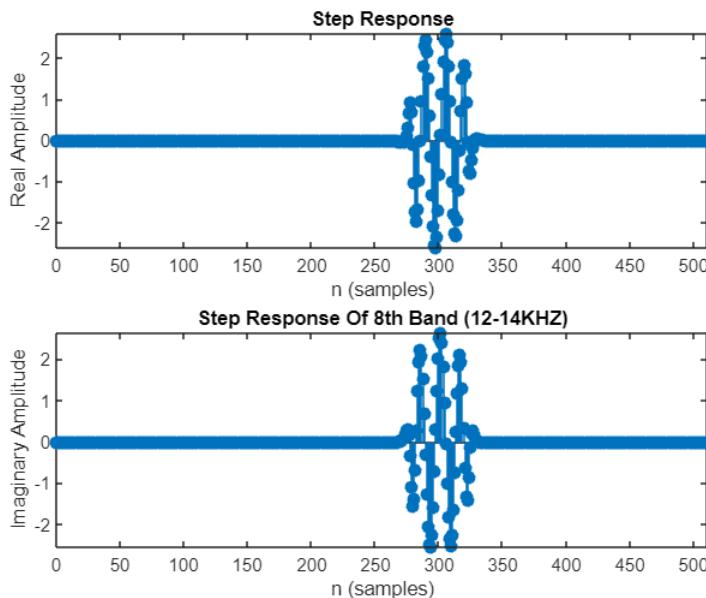
```



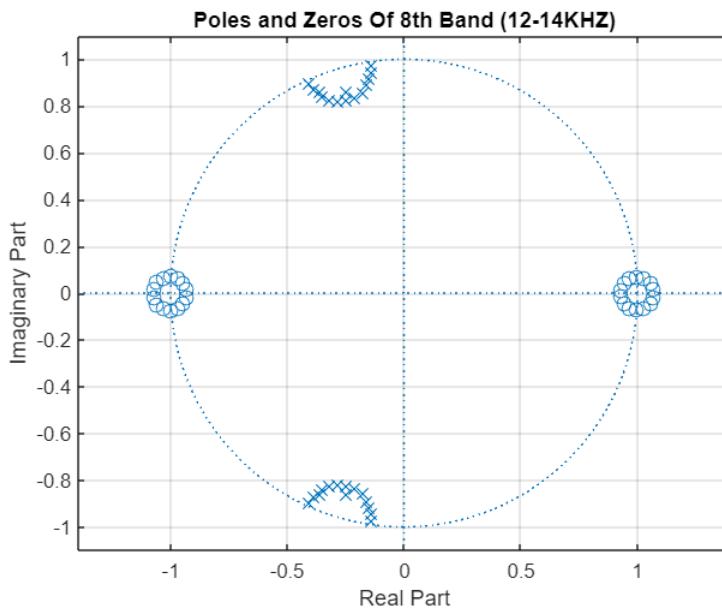
```
%%%%%%%%%%%%% Analysis Of 8th IIR Filter %%%%%%
figure;
impz(HD8);
title('Impulse Response Of 8th Band (12-14KHZ)');
```



```
figure;
stepz(HD8);
title('Step Response Of 8th Band (12-14KHZ)');
```



```
figure;
zplane(b8, a8);
grid;
title('Poles and Zeros Of 8th Band (12-14KHZ)');
```



```

figure;
subplot(2, 1, 1);
magnitude = 20*log10(abs(HD8));
gain = max(magnitude);
fprintf('IIR Filter 8 Gain: %.2f dB\n', gain);

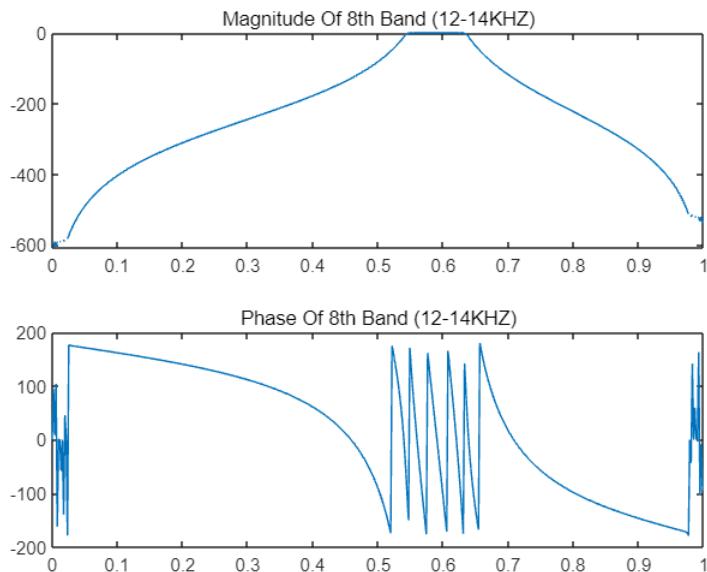
```

IIR Filter 8 Gain: 0.02 dB

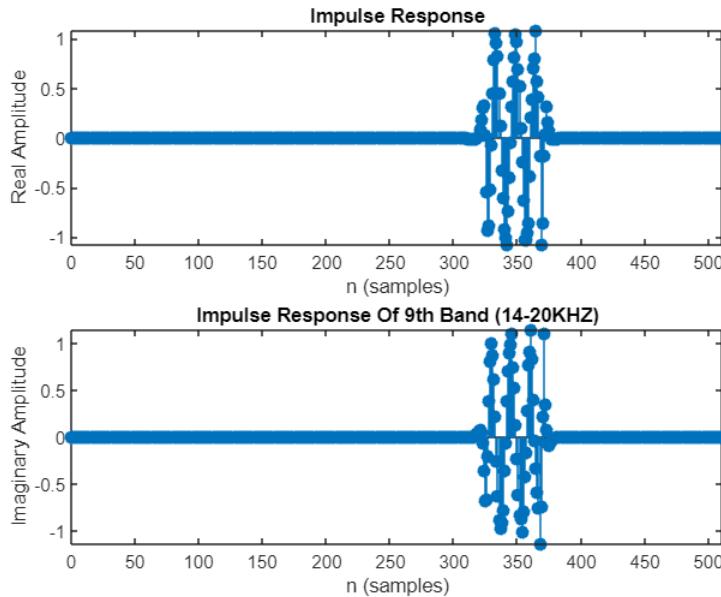
```

plot(wcd8/pi, magnitude);
subtitle('Magnitude Of 8th Band (12-14KHZ)');
subplot(2, 1, 2);
plot(wcd8/pi, rad2deg(angle(HD8)));
subtitle('Phase Of 8th Band (12-14KHZ)');

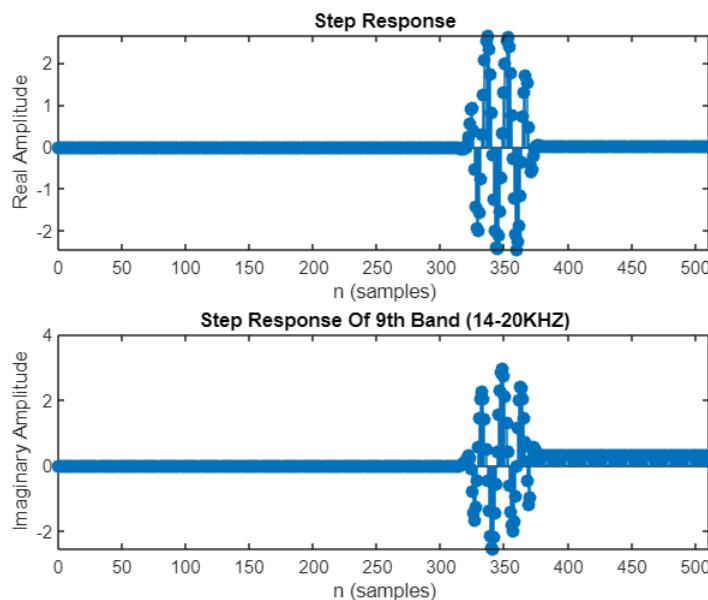
```



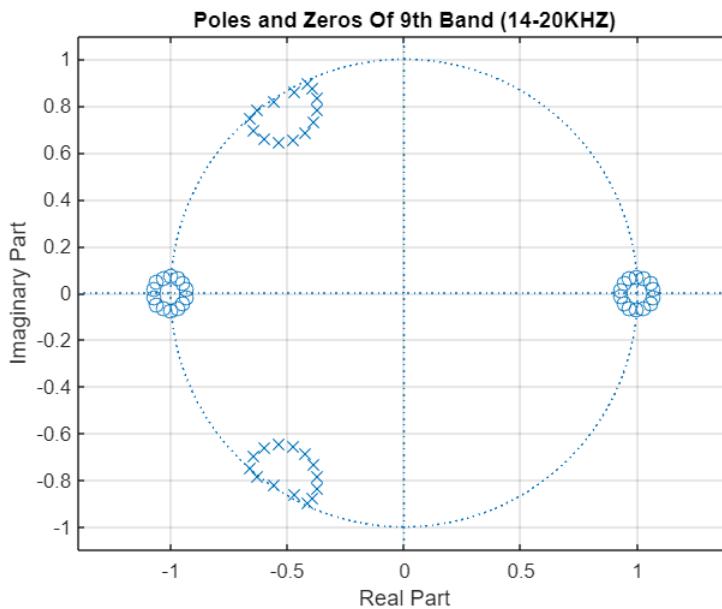
```
%%%%%%%%%%%%% Analysis Of 9th IIR Filter %%%%%%
figure;
impz(HD9);
title('Impulse Response Of 9th Band (14-20KHZ)');
```



```
figure;
stepz(HD9);
title('Step Response Of 9th Band (14-20KHZ)');
```



```
figure;
zplane(b9, a9);
grid;
title('Poles and Zeros Of 9th Band (14-20KHZ)'');
```



```

figure;
subplot(2, 1, 1);
magnitude = 20*log10(abs(HD8));
gain = max(magnitude);
fprintf('IIR Filter 9 Gain: %.2f dB\n', gain);

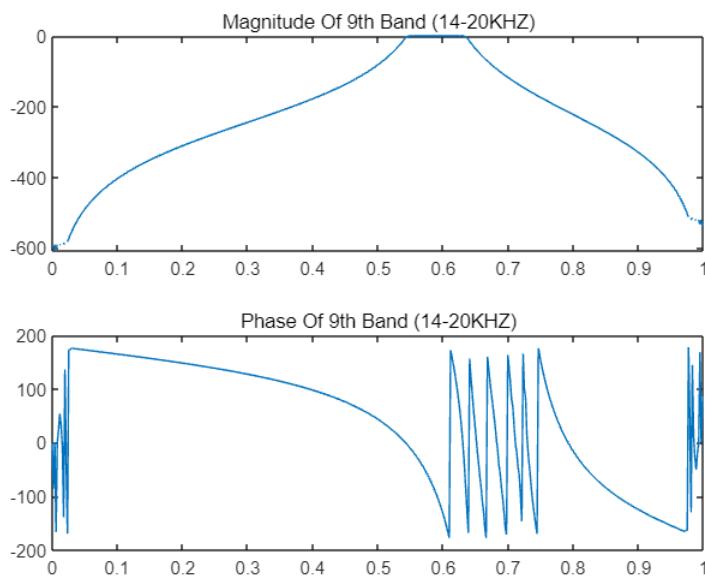
```

IIR Filter 9 Gain: 0.02 dB

```

plot(wcd9/pi, magnitude);
subtitle('Magnitude Of 9th Band (14-20KHZ)');
subplot(2, 1, 2);
plot(wcd9/pi, rad2deg(angle(HD9)));
subtitle('Phase Of 9th Band (14-20KHZ)');

```



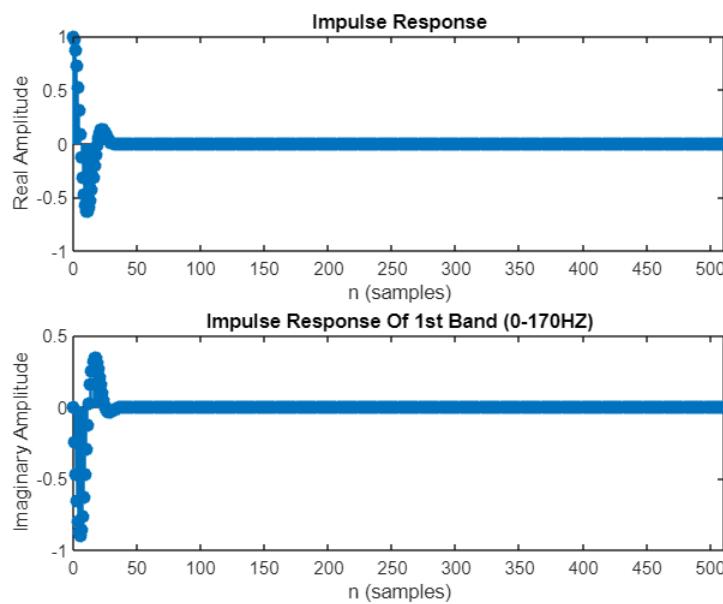
```

%%%%%%%%%%%%% Define FIR Blackman Filters %%%%%%
Order = 80;
TF1 = fir1(Order, f1, blackman(Order + 1));
TF2 = fir1(Order, f2, blackman(Order + 1));
TF3 = fir1(Order, f3, blackman(Order + 1));
TF4 = fir1(Order, f4, blackman(Order + 1));
TF5 = fir1(Order, f5, blackman(Order + 1));
TF6 = fir1(Order, f6, blackman(Order + 1));
TF7 = fir1(Order, f7, blackman(Order + 1));
TF8 = fir1(Order, f8, blackman(Order + 1));
TF9 = fir1(Order, f9, blackman(Order + 1));

%%%%%%%%%%%%% Get Transfer Function Of Each IIR Filter %%%%%%
[TF1, wcd1] = freqz(TF1, 1);
[TF2, wcd2] = freqz(TF2, 1);
[TF3, wcd3] = freqz(TF3, 1);
[TF4, wcd4] = freqz(TF4, 1);
[TF5, wcd5] = freqz(TF5, 1);
[TF6, wcd6] = freqz(TF6, 1);
[TF7, wcd7] = freqz(TF7, 1);
[TF8, wcd8] = freqz(TF8, 1);
[TF9, wcd9] = freqz(TF9, 1);

%%%%%%%%%%%%% Analysis Of 1st FIR Filter %%%%%%
figure;
impz(TF1);
title('Impulse Response Of 1st Band (0-170HZ)');

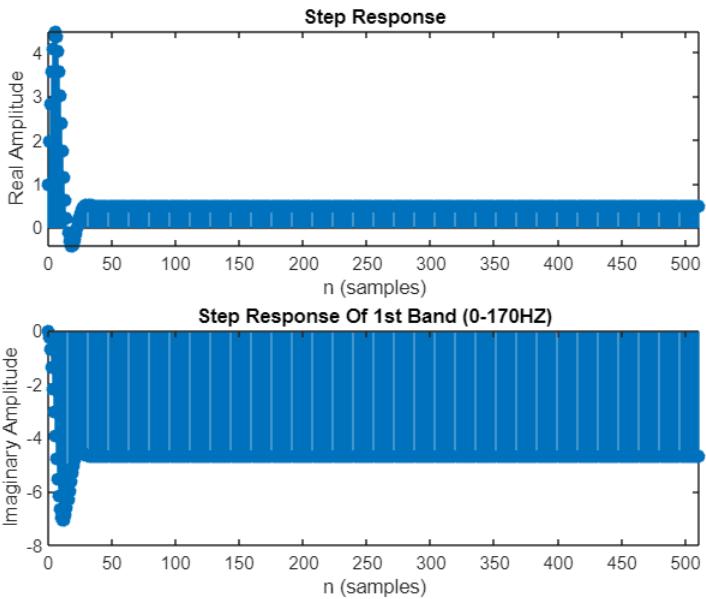
```



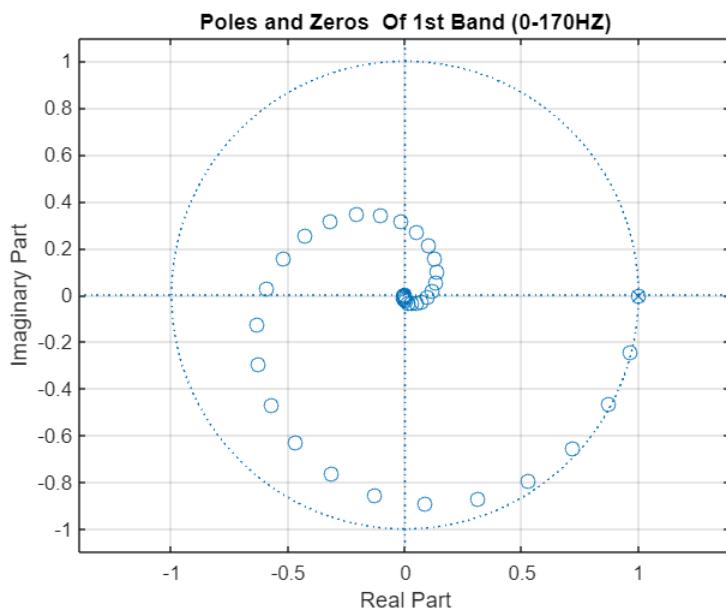
```

figure;
stepz(TF1);
title('Step Response Of 1st Band (0-170HZ)');

```



```
figure;
zplane(TF1, 1);
grid;
title('Poles and Zeros Of 1st Band (0-170HZ)');
```



```
figure;
subplot(2, 1, 1);
magnitude = 20*log10(abs(TF1));
gain = max(magnitude);
fprintf('FIR Filter 1 Gain: %.2f dB\n', gain);
```

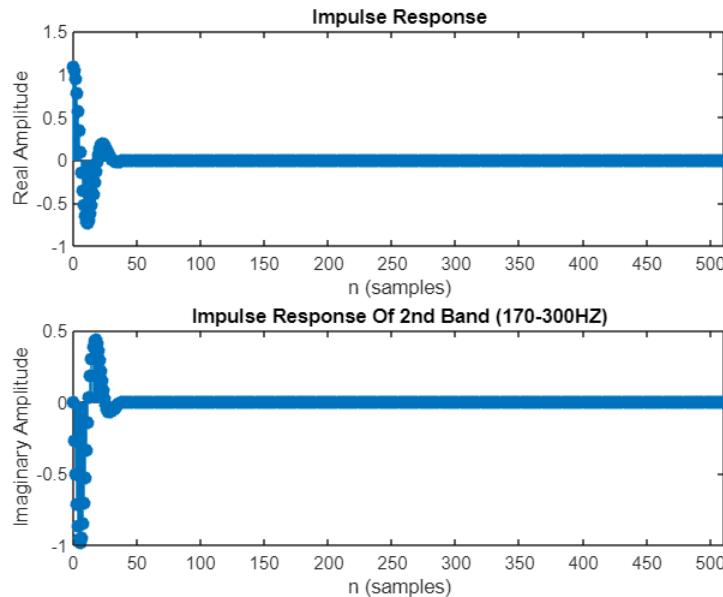
FIR Filter 1 Gain: 0.00 dB

```

plot(wcd1/pi, magnitude);
subplot(2, 1, 1);
title('Magnitude Of 1st Band (0-170HZ)');
subplot(2, 1, 2);
plot(wcd1/pi, rad2deg(angle(TF1)));
title('Phase Of 1st Band (0-170HZ)');

%%%%%%%%%%%%_ Analysis Of 2nd FIR Filter _%%%%%%%%%%%%%
impz(TF2);
title('Impulse Response Of 2nd Band (170-300HZ)');

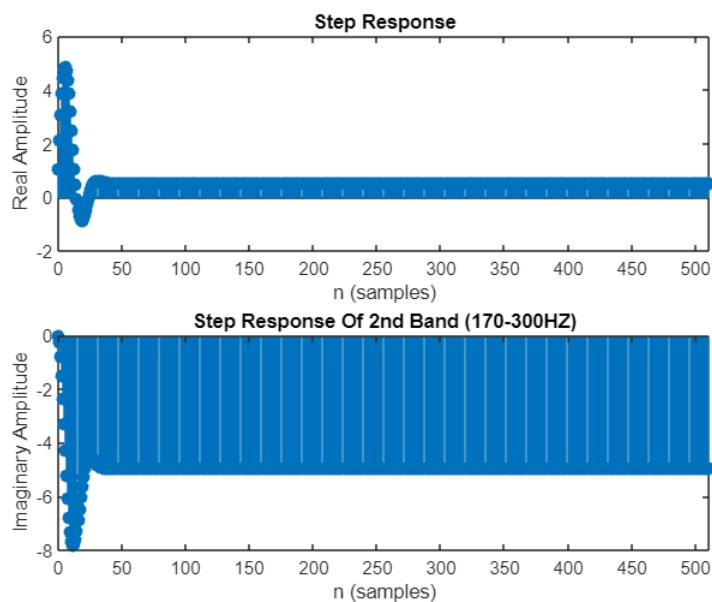
```



```

figure;
stepz(TF2);
title('Step Response Of 2nd Band (170-300HZ)');

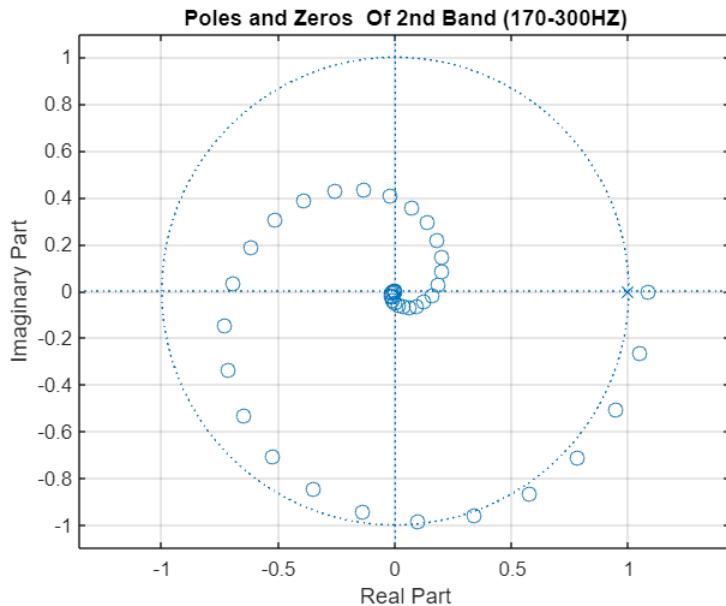
```



```

figure;
zplane(TF2, 1);
grid;
title('Poles and Zeros Of 2nd Band (170-300HZ)');

```



```

figure;
subplot(2, 1, 1);
magnitude = 20*log10(abs(TF2));
gain = max(magnitude);
fprintf('FIR Filter 2 Gain: %.2f dB\n', gain);

```

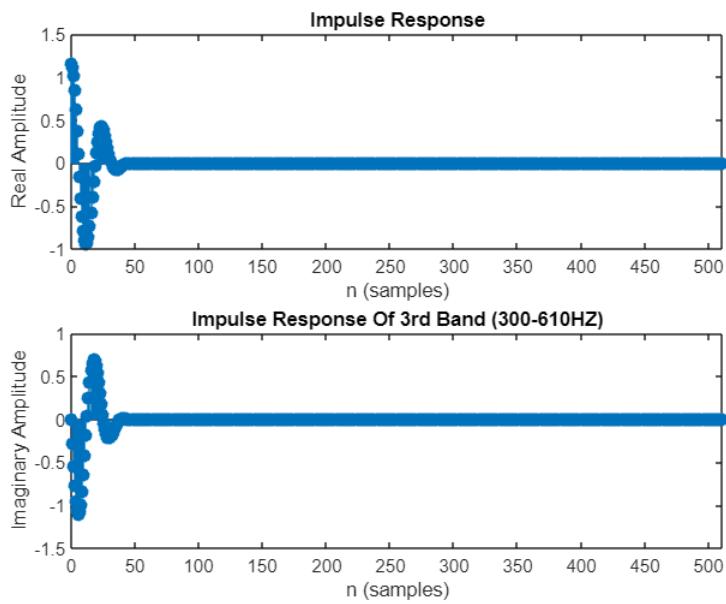
FIR Filter 2 Gain: 0.70 dB

```

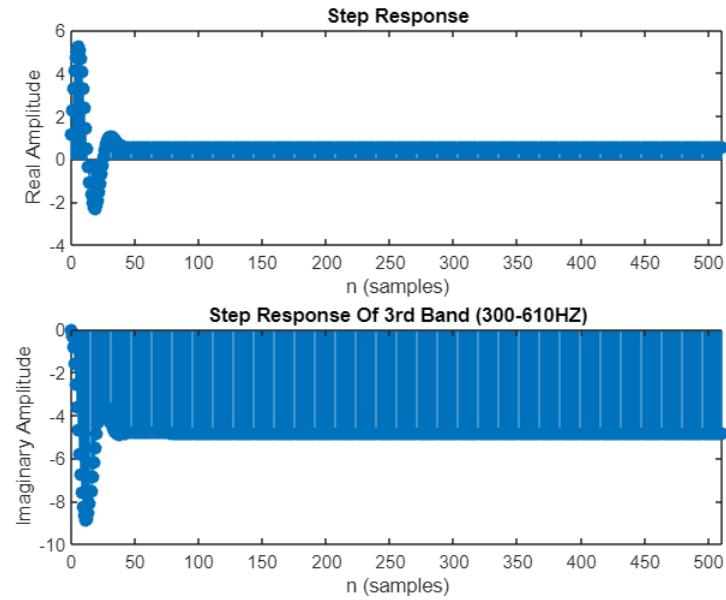
plot(wcd2/pi, magnitude);
subtitle('Magnitude Of 2nd Band (170-300HZ)');
subplot(2, 1, 2);
plot(wcd2/pi, rad2deg(angle(TF2)));
subtitle('Phase Of 2nd Band (170-300HZ)');

%%%%%% Analysis Of 3rd FIR Filter %%%%%%
impz(TF3);
title('Impulse Response Of 3rd Band (300-610HZ)');

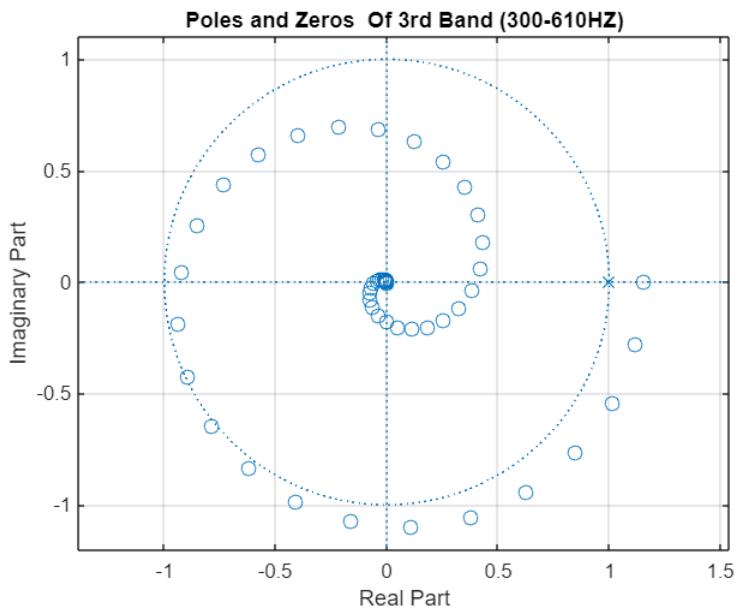
```



```
figure;
stepz(TF3);
title('Step Response Of 3rd Band (300-610HZ)');
```



```
figure;
zplane(TF3, 1);
grid;
title('Poles and Zeros Of 3rd Band (300-610HZ)');
```



```

figure;
subplot(2, 1, 1);
magnitude = 20*log10(abs(TF3));
gain = max(magnitude);
fprintf('FIR Filter 3 Gain: %.2f dB\n', gain);

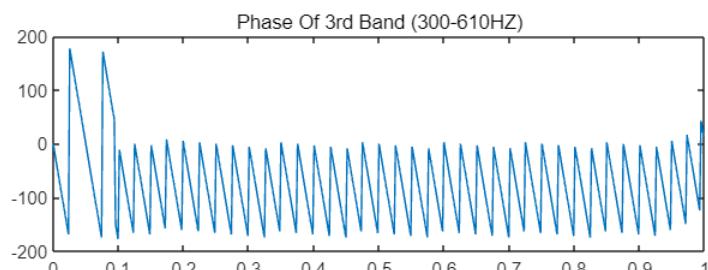
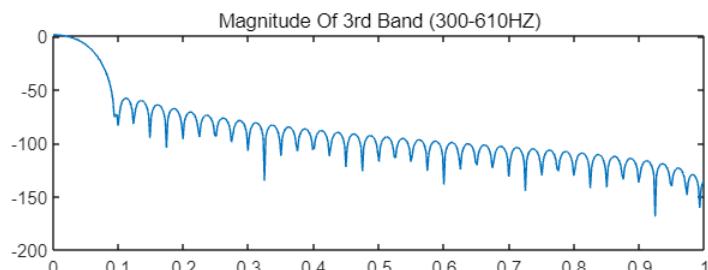
```

FIR Filter 3 Gain: 1.25 dB

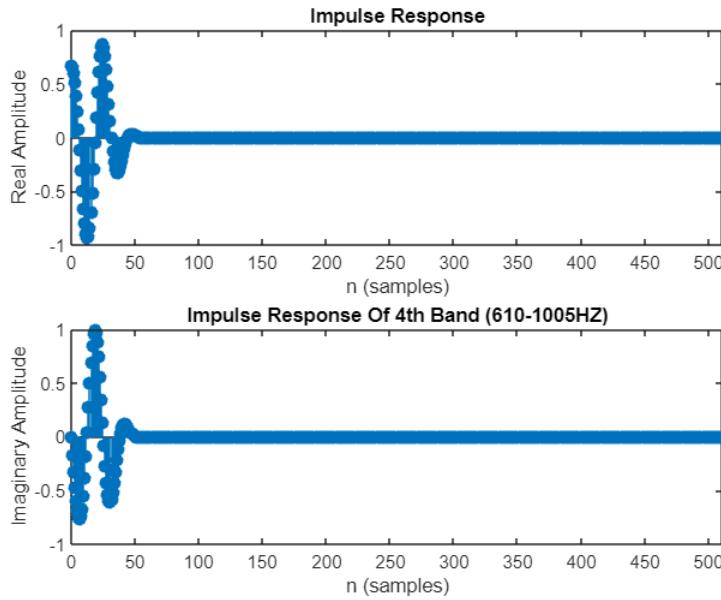
```

plot(wcd3/pi, magnitude);
subtitle('Magnitude Of 3rd Band (300-610HZ)');
subplot(2, 1, 2);
plot(wcd3/pi, rad2deg(angle(TF3)));
subtitle('Phase Of 3rd Band (300-610HZ)');

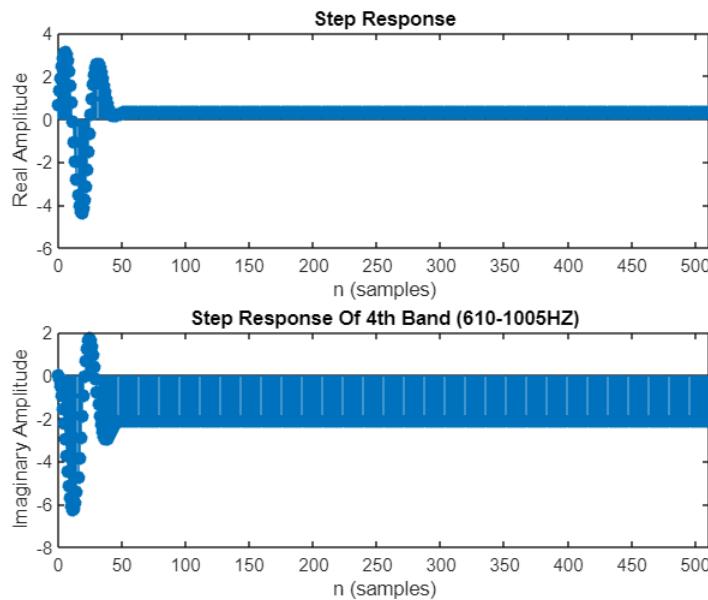
```



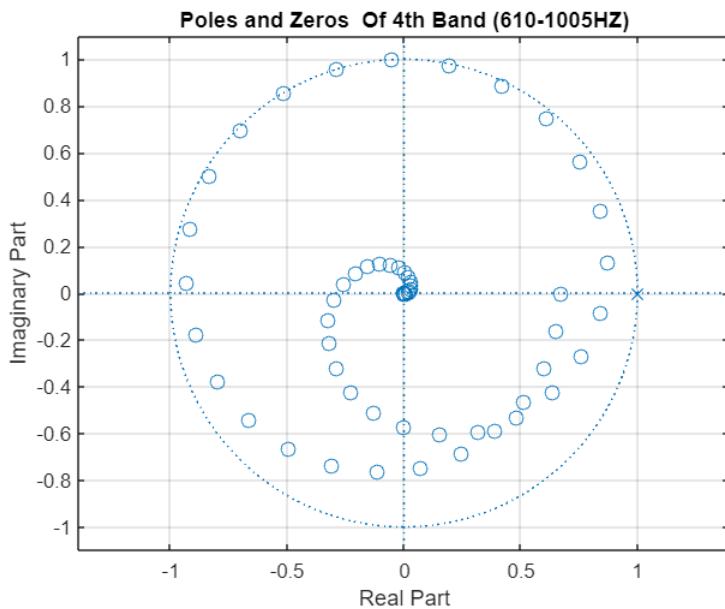
```
%%%%%%%%%%%%_ Analysis Of 4th FIR Filter _%%%%%%%%%%%%%
figure;
impz(TF4);
title('Impulse Response Of 4th Band (610-1005HZ)');
```



```
figure;
stepz(TF4);
title('Step Response Of 4th Band (610-1005HZ)');
```



```
figure;
zplane(TF4, 1);
grid;
title('Poles and Zeros Of 4th Band (610-1005HZ)');
```



```

figure;
subplot(2, 1, 1);
magnitude = 20*log10(abs(TF4));
gain = max(magnitude);
fprintf('FIR Filter 4 Gain: %.2f dB\n', gain);

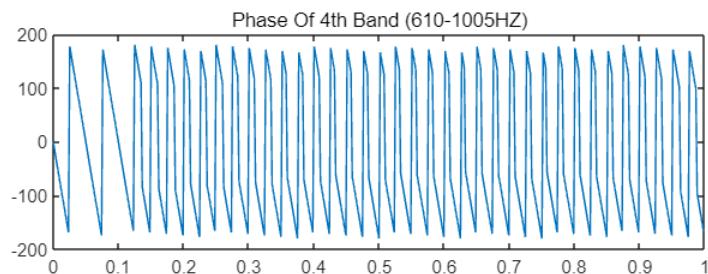
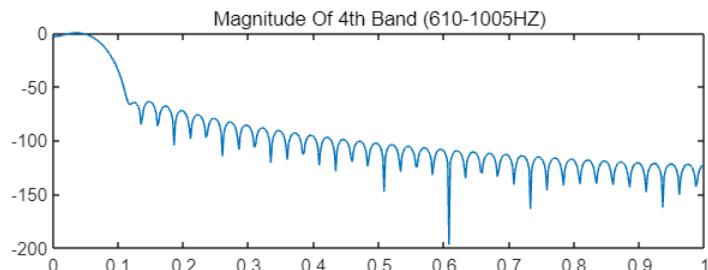
```

FIR Filter 4 Gain: -0.00 dB

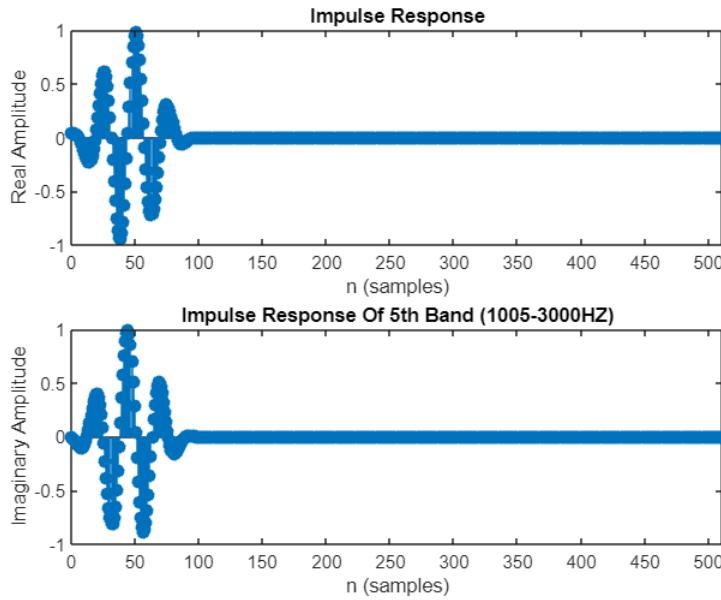
```

plot(wcd4/pi, magnitude);
subtitle('Magnitude Of 4th Band (610-1005HZ)');
subplot(2, 1, 2);
plot(wcd4/pi, rad2deg(angle(TF4)));
subtitle('Phase Of 4th Band (610-1005HZ)');

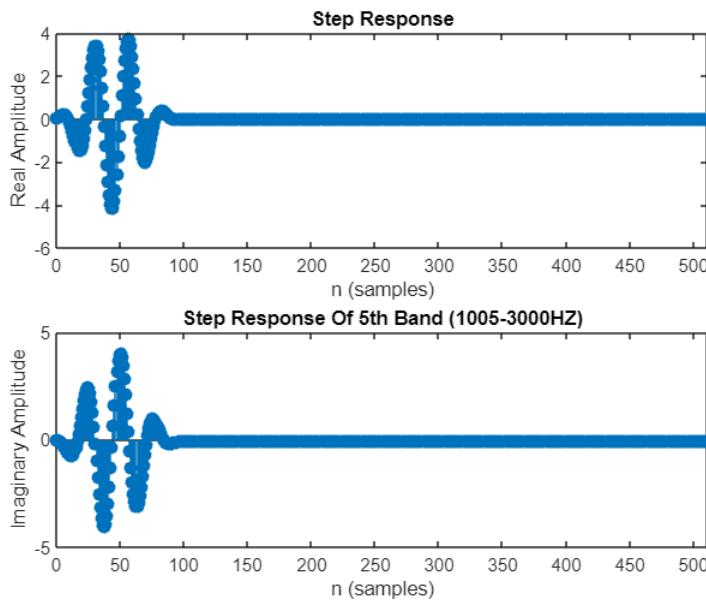
```



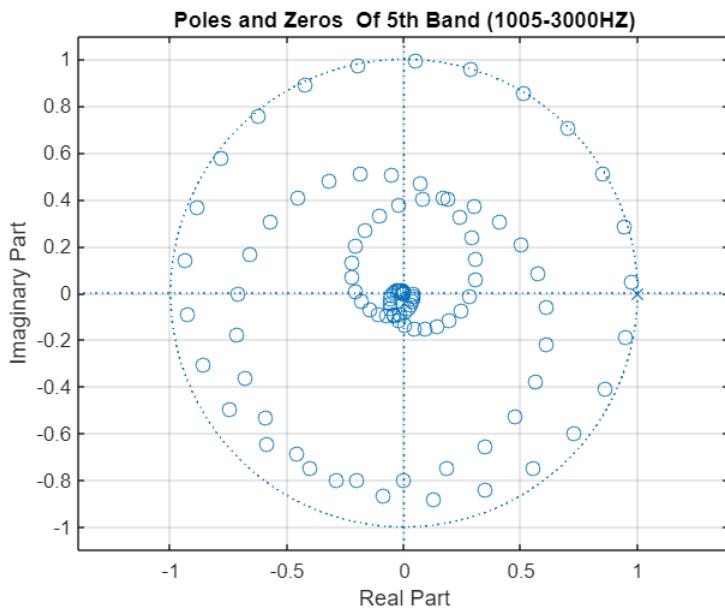
```
%%%%%%%%%%%%_ Analysis Of 5th FIR Filter _%%%%%%%%%%%%%
figure;
impz(TF5);
title('Impulse Response Of 5th Band (1005-3000HZ)');
```



```
figure;
stepz(TF5);
title('Step Response Of 5th Band (1005-3000HZ)');
```



```
figure;
zplane(TF5, 1);
grid;
title('Poles and Zeros Of 5th Band (1005-3000HZ)');
```



```

figure;
subplot(2, 1, 1);
magnitude = 20*log10(abs(TF5));
gain = max(magnitude);
fprintf('FIR Filter 5 Gain: %.2f dB\n', gain);

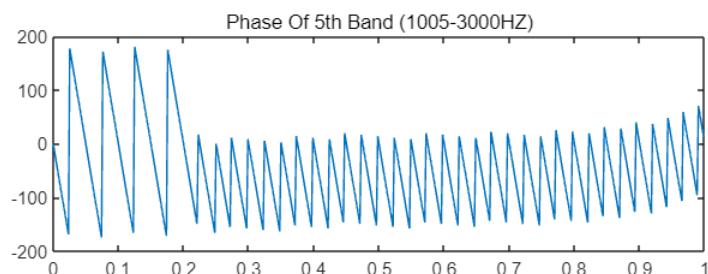
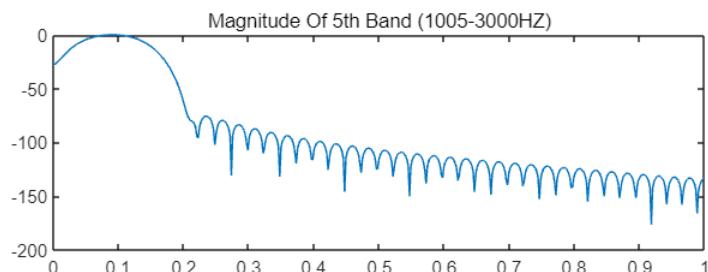
```

FIR Filter 5 Gain: -0.00 dB

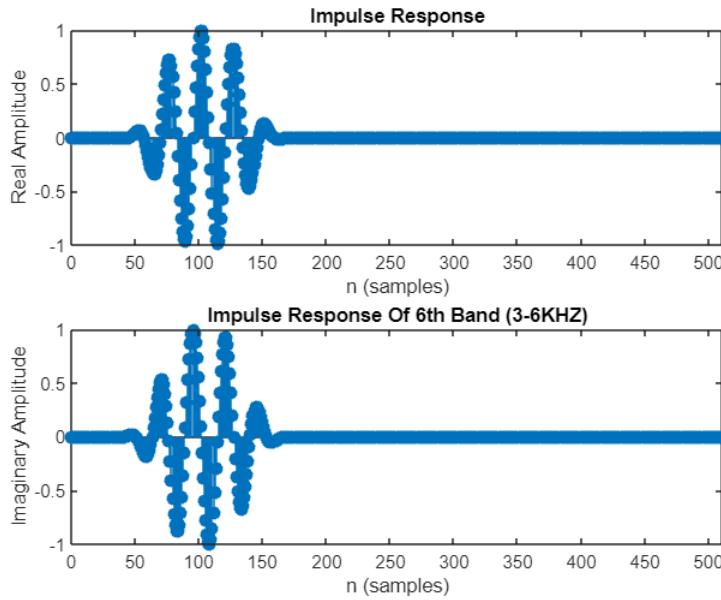
```

plot(wcd5/pi, magnitude);
subtitle('Magnitude Of 5th Band (1005-3000HZ)');
subplot(2, 1, 2);
plot(wcd5/pi, rad2deg(angle(TF5)));
subtitle('Phase Of 5th Band (1005-3000HZ)');

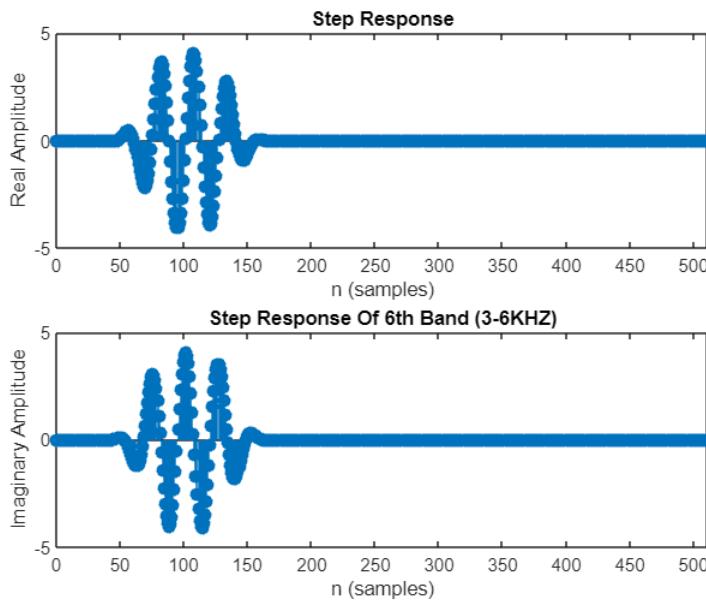
```



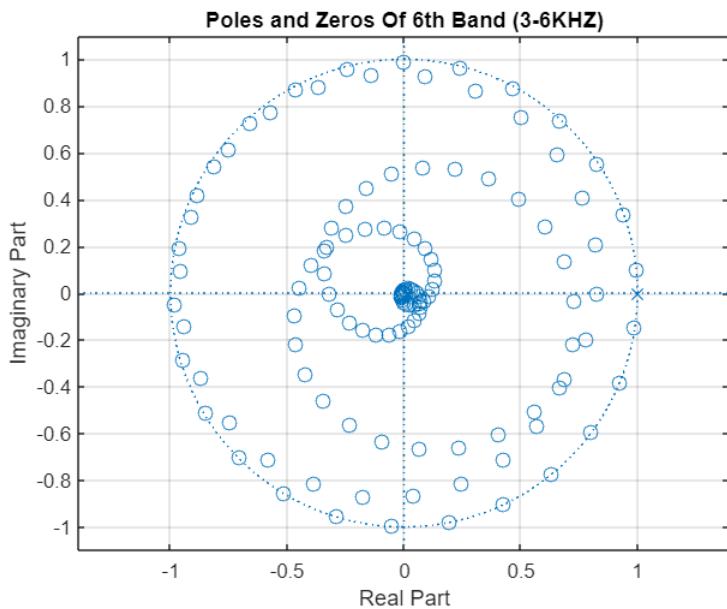
```
%%%%%%%%%%%%_ Analysis Of 6th FIR Filter _%%%%%%%%%%%%%
figure;
impz(TF6);
title('Impulse Response Of 6th Band (3-6KHZ)');
```



```
figure;
stepz(TF6);
title('Step Response Of 6th Band (3-6KHZ)');
```



```
figure;
zplane(TF6, 1);
grid;
title('Poles and Zeros Of 6th Band (3-6KHZ)');
```



```

figure;
subplot(2, 1, 1);
magnitude = 20*log10(abs(TF6));
gain = max(magnitude);
fprintf('FIR Filter 6 Gain: %.2f dB\n', gain);

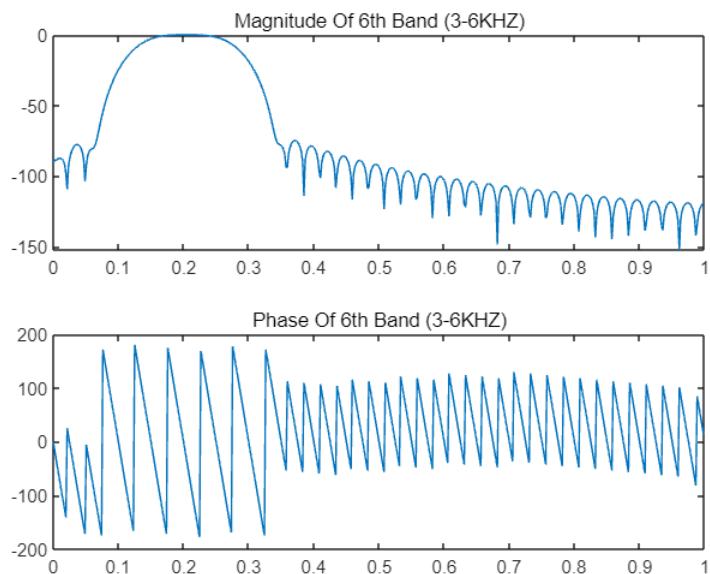
```

FIR Filter 6 Gain: -0.00 dB

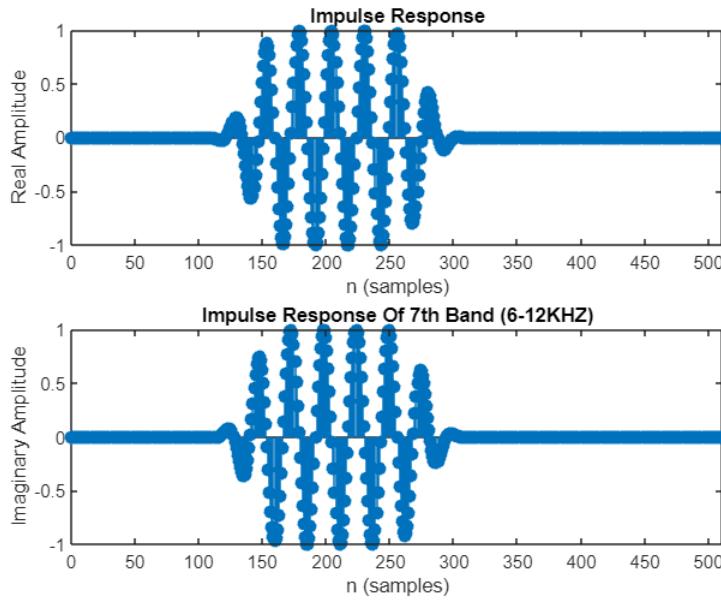
```

plot(wcd6/pi, magnitude);
subtitle('Magnitude Of 6th Band (3-6KHZ)');
subplot(2, 1, 2);
plot(wcd6/pi, rad2deg(angle(TF6)));
subtitle('Phase Of 6th Band (3-6KHZ)');

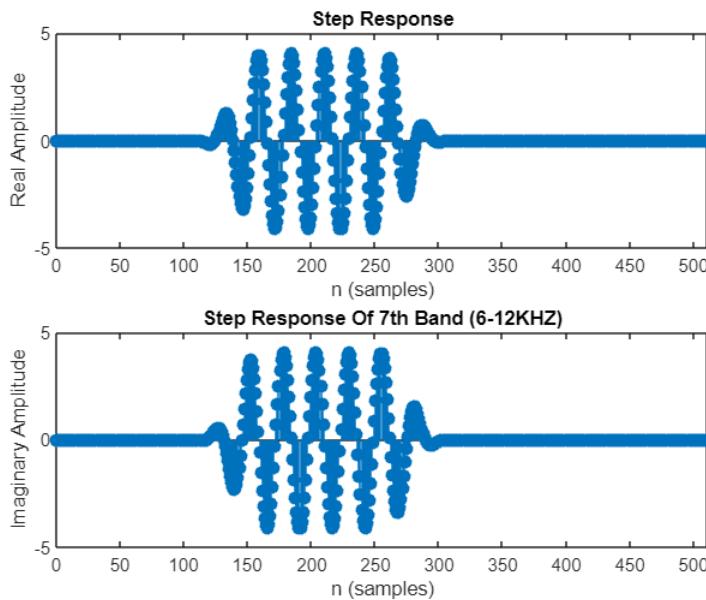
```



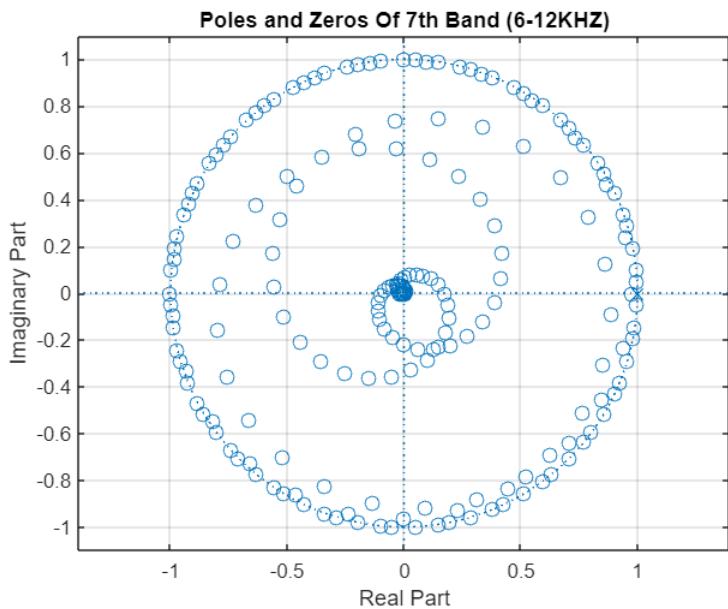
```
%%%%%%%%%%%%_ Analysis Of 7th FIR Filter _%%%%%%%%%%%%%
figure;
impz(TF7);
title('Impulse Response Of 7th Band (6-12KHZ)');
```



```
figure;
stepz(TF7);
title('Step Response Of 7th Band (6-12KHZ)');
```



```
figure;
zplane(TF7, 1);
grid;
title('Poles and Zeros Of 7th Band (6-12KHZ)');
```



```

figure;
subplot(2, 1, 1);
magnitude = 20*log10(abs(TF7));
gain = max(magnitude);
fprintf('FIR Filter 7 Gain: %.2f dB\n', gain);

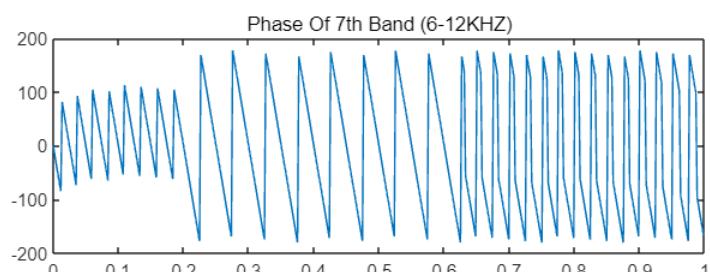
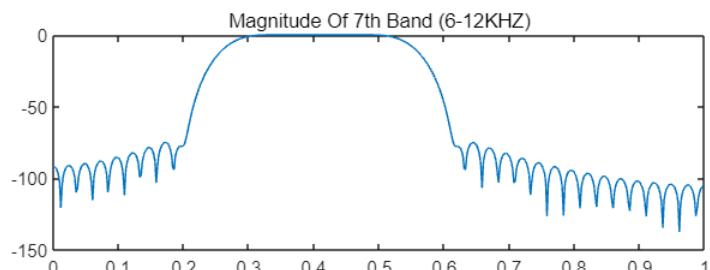
```

FIR Filter 7 Gain: 0.00 dB

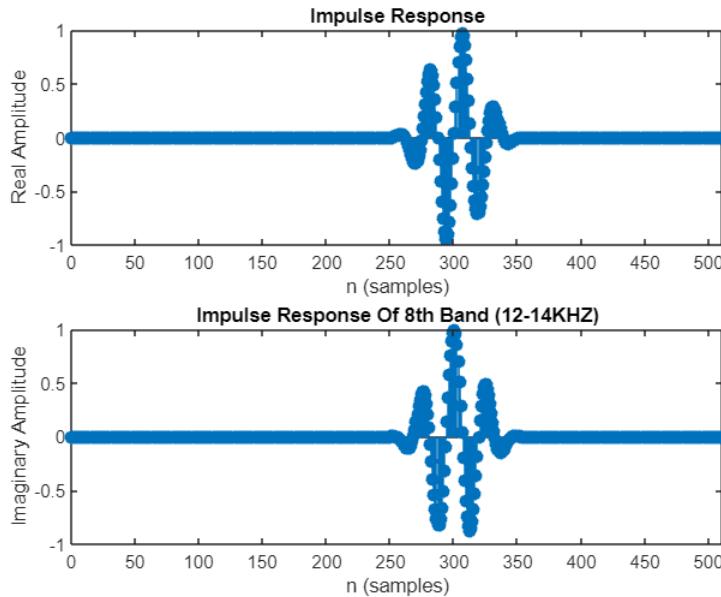
```

plot(wcd7/pi, magnitude);
subtitle('Magnitude Of 7th Band (6-12KHZ)');
subplot(2, 1, 2);
plot(wcd7/pi, rad2deg(angle(TF7)));
subtitle('Phase Of 7th Band (6-12KHZ)');

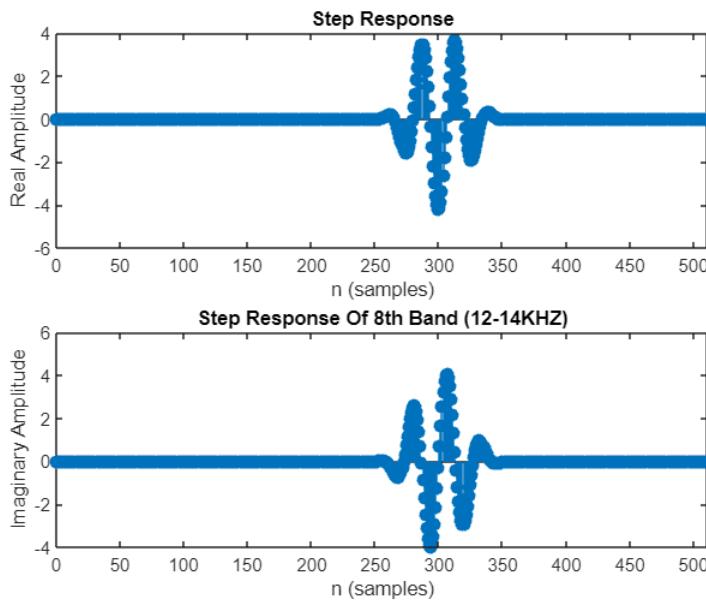
```



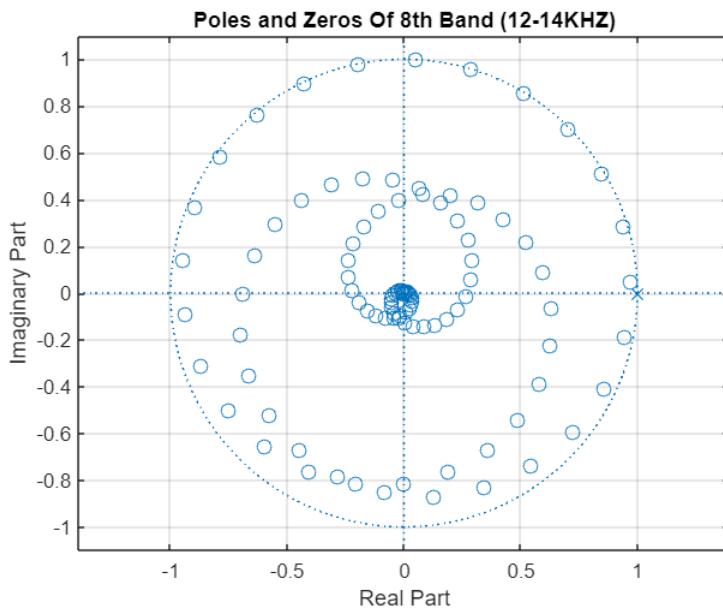
```
%%%%%%%%%%%%_ Analysis Of 8th FIR Filter _%%%%%%%%%%%%%
figure;
impz(TF8);
title('Impulse Response Of 8th Band (12-14KHZ)');
```



```
figure;
stepz(TF8);
title('Step Response Of 8th Band (12-14KHZ)');
```



```
figure;
zplane(TF8, 1);
grid;
title('Poles and Zeros Of 8th Band (12-14KHZ)');
```



```

figure;
subplot(2, 1, 1);
magnitude = 20*log10(abs(TF8));
gain = max(magnitude);
fprintf('FIR Filter 8 Gain: %.2f dB\n', gain);

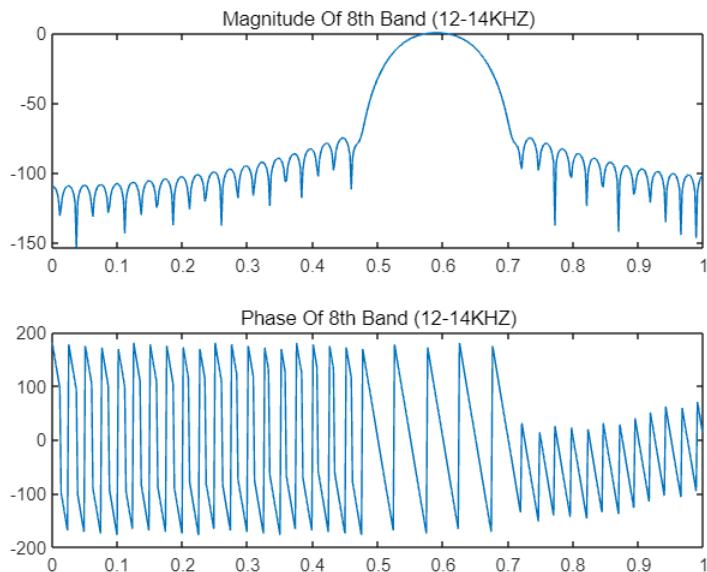
```

FIR Filter 8 Gain: -0.00 dB

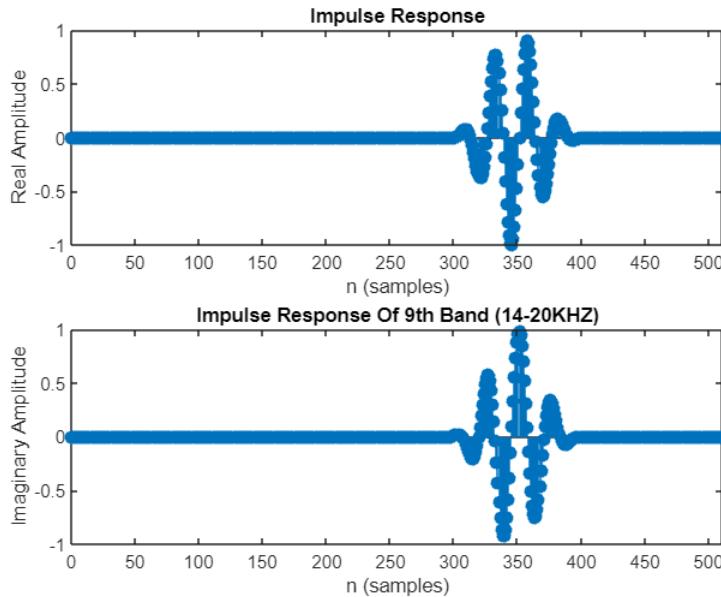
```

plot(wcd8/pi, magnitude);
subtitle('Magnitude Of 8th Band (12-14KHZ)');
subplot(2, 1, 2);
plot(wcd8/pi, rad2deg(angle(TF8)));
subtitle('Phase Of 8th Band (12-14KHZ)');

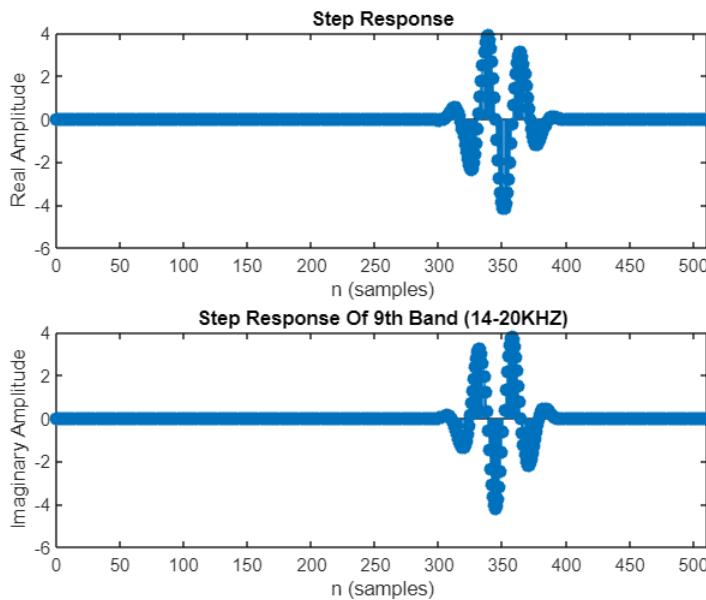
```



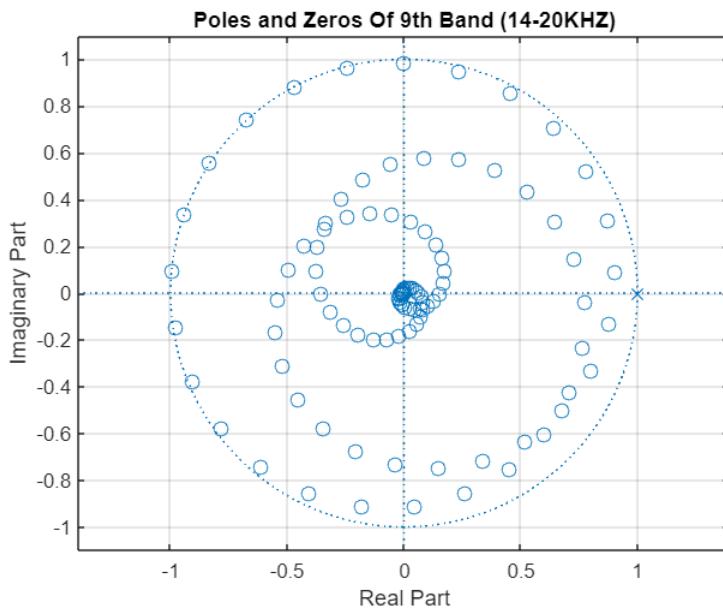
```
%%%%%%%%%%%%_ Analysis Of 9th FIR Filter _%%%%%%%%%%%%%
figure;
impz(TF9);
title('Impulse Response Of 9th Band (14-20KHZ)');
```



```
figure;
stepz(TF9);
title('Step Response Of 9th Band (14-20KHZ)');
```



```
figure;
zplane(TF9, 1);
grid;
title('Poles and Zeros Of 9th Band (14-20KHZ)');
```



```

figure;
subplot(2, 1, 1);
magnitude = 20*log10(abs(TF8));
gain = max(magnitude);
fprintf('FIR Filter 9 Gain: %.2f dB\n', gain);

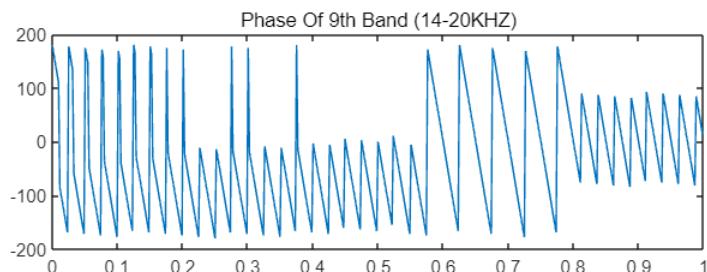
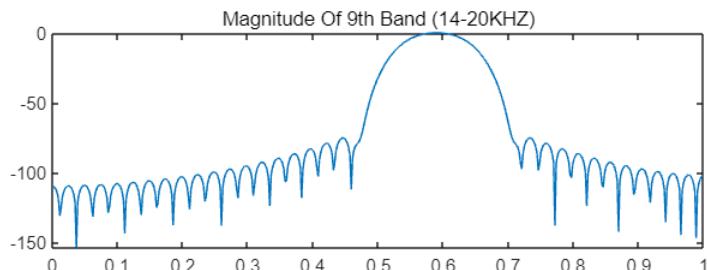
```

FIR Filter 9 Gain: -0.00 dB

```

plot(wcd9/pi, magnitude);
subtitle('Magnitude Of 9th Band (14-20KHZ)');
subplot(2, 1, 2);
plot(wcd9/pi, rad2deg(angle(TF9)));
subtitle('Phase Of 9th Band (14-20KHZ)');

```



# APP DESIGNER CODE

```
1 classdef app1 < matlab.apps.AppBase
2
3     % Properties that correspond to app components
4     properties (Access = public) ...
5
6
7         % Properties section
8     properties (Access = private)
9         Ip % Audio data
10        Fs % Sampling frequency
11        path %Audio Path
12        OpSound %Output Sound
13        CheckConditions = 0; %check wether the user enters all req or not
14        CheckAudio = 0;
15    end
16
17
18    % Callbacks that handle component events
19    methods (Access = private)
20
21        % Button pushed function: BrowseButton
22        function BrowseButtonPushed(app, event)
23            [fileName, app.path] = uigetfile('.wav','Select an Audio file');
24            % Check if the user clicked Cancel
25            if isequal(fileName, 0)
26                return; % Exit the function
27            end
28            app.CheckAudio = 1;
29            [app.Ip, app.Fs] = audioread(fileName);
30            app.SoundEditField.Value = app.path;
31            plot(app.UIAxes,app.Ip, 'Color', [0 0 0]);
32            plot(app.UIAxes_2,abs(fftshift(fft(app.Ip)) / app.Fs), 'Color', [0 0 0]);
33        end
34
35        % Callback function
36        function AudioInfoValueChanging(app, event)
37
38        end
39
40        % Callback function
41        function PanelSizeChanged(app, event)
42            position = app.Panel.Position;
43
44        end
45
46        % Value changing function: HZSlider
47        function HZSliderValueChanging(app, event)
48            changingValue = event.Value;
49
50        end
51
52        % Value changing function: SoundEditField
53        function SoundEditFieldValueChanging(app, event)
54            changingValue = event.Value;
55
56        end
```

```

97
98 % Value changed function: HZSlider_9
99 function HZSlider_9ValueChanging(app, event)
100     changingValue = event.Value;
101 end
102
103 % Callback function
104 function ListBoxClicked(app, event)
105     item = event.InteractionInformation.Item;
106 end
107
108 % Callback function: ApplyButton, HZSlider
109 function ApplyFilterButtonPushed(app, event)
110     if app.CheckAudio == 0
111         msgbox('Please Choose An Audio And Try Again!', 'Error', 'error');
112         return;
113     end
114     app.CheckConditions = 1;
115     %%%%%%%% Calculate Fs Normalized %%%%%%%
116     F_normalized = app.Fs / 2;
117     %%%%%%%% Calculate Frequencies Normalized %%%%%%
118     f1 = 170 / F_normalized;
119     f2 = [170 / F_normalized, 310 / F_normalized];
120     f3 = [310 / F_normalized, 600 / F_normalized];
121     f4 = [600 / F_normalized, 1000 / F_normalized];
122     f5 = [1000 / F_normalized, 3000 / F_normalized];
123     f6 = [3000 / F_normalized, 6000 / F_normalized];
124     f7 = [6000 / F_normalized, 12000 / F_normalized];
125     f8 = [12000 / F_normalized, 14000 / F_normalized];
126     f9 = [14000 / F_normalized, 16000 / F_normalized];
127     %%%%%%%% Convert DB to WT %%%%%%
128     gain1 = 10^(app.HZSlider.Value/10);
129     gain2 = 10^(app.HZSlider_2.Value/10);
130     gain3 = 10^(app.HZSlider_3.Value/10);
131     gain4 = 10^(app.HZSlider_4.Value/10);
132     gain5 = 10^(app.HZSlider_5.Value/10);
133     gain6 = 10^(app.HZSlider_6.Value/10);
134     gain7 = 10^(app.HZSlider_7.Value/10);
135     gain8 = 10^(app.HZSlider_8.Value/10);
136     gain9 = 10^(app.HZSlider_9.Value/10);
137     %%%%%%%% Switch Case %%%%%%
138     switch app.ListBox.Value
139         case 'FIR'
140
141             Order = 80;
142             TF1 = fir1(Order, f1, blackman(Order + 1));
143             Op1 = filter(TF1, 1, app.Ip);
144             Op1 = Op1 * gain1;
145
146             TF2 = fir1(Order, f2, blackman(Order + 1));
147             Op2 = filter(TF2, 1, app.Ip);
148             Op2 = Op2 * gain2;
149
150             TF3=fir1(Order,f3,blackman(Order+1));
151             Op3=filter(TF3,1,app.Ip);
152             Op3=Op3*gain3;
153
154             TF4=fir1(Order,f4,blackman(Order+1));
155             Op4=filter(TF4,1,app.Ip);
156             Op4=Op4*gain4;
157
158             TF5=fir1(Order,f5,blackman(Order+1));
159             Op5=filter(TF5,1,app.Ip);
160             Op5=Op5*gain5;

```

```

161
162         TF6=fir1(Order,f6,blackman(Order+1));
163         Op6=filter(TF6,1,app.Ip);
164         Op6=Op6*gain6;
165
166         TF7=fir1(Order,f7,blackman(Order+1));
167         Op7=filter(TF7,1,app.Ip);
168         Op7=Op7*gain7;
169
170         TF8=fir1(Order,f8,blackman(Order+1));
171         Op8=filter(TF8,1,app.Ip);
172         Op8=Op8*gain8;
173
174         TF9=fir1(Order,f9,blackman(Order+1));
175         Op9=filter(TF9,1,app.Ip);
176         Op9=Op9*gain9;
177
178         app.OpSound = [Op1 + Op2 + Op3 + Op4 + Op5 + Op6 + Op7 + Op8 + Op9];
179
180     case 'IIR'
181         %%%%%% Define IIR Butter Worth Filters %%%%%%
182         [b1, a1] = butter( 7, f1, 'low');
183         [b2, a2] = butter( 3, f2, 'bandpass');
184         [b3, a3] = butter( 5, f3, 'bandpass');
185         [b4, a4] = butter( 5, f4, 'bandpass');
186         [b5, a5] = butter( 7, f5, 'bandpass');
187         [b6, a6] = butter( 7, f6, 'bandpass');
188         [b7, a7] = butter(14, f7, 'bandpass');
189         [b8, a8] = butter(14, f8, 'bandpass');
190         [b9, a9] = butter(14, f9, 'bandpass');
191
192         Op1 = filter(b1,a1,app.Ip);
193         Op1 = Op1 * gain1;
194
195         Op2 = filter(b2,a2,app.Ip);
196         Op2 = Op2 * gain2;
197
198         Op3 = filter(b3,a3,app.Ip);
199         Op3 = Op3 * gain3;
200
201         Op4 = filter(b4,a4,app.Ip);
202         Op4 = Op4 * gain4;
203
204         Op5 = filter(b5,a5,app.Ip);
205         Op5 = Op5 * gain5;
206
207         Op6 = filter(b6,a6,app.Ip);
208         Op6 = Op6 * gain6;
209
210         Op7 = filter(b7,a7,app.Ip);
211         Op7 = Op7 * gain7;
212
213         Op8 = filter(b8,a8,app.Ip);
214         Op8 = Op8 * gain8;
215
216         Op9 = filter(b9,a9,app.Ip);
217         Op9 = Op9 * gain9;
218
219         app.OpSound = [Op1 + Op2 + Op3 + Op4 + Op5 + Op6 + Op7 + Op8 + Op9];
220
221     end
222     NewFs = app.Slider.Value * app.Fs;
223     plot(app.UIAxes,app.OpSound, 'Color', [0 0 0]);
224     plot(app.UIAxes_2,abs(fftshift(fft(app.OpSound)) / NewFs), 'Color', [0 0 0]);
225
226 end

```

```

227         % Button pushed function: PlayOriginalAudioButton
228     function PlayOriginalAudioButtonPushed(app, event)
229         if app.CheckAudio == 1
230             plot(app.UIAxes,app.Ip, 'Color', [0 0 0]);
231             NewFs = app.Slider.Value * app.Fs;
232             plot(app.UIAxes_2,abs(fftshift(fft(app.Ip)) / app.Fs), 'Color', [0 0 0]);
233             sound(app.Ip, NewFs);
234         else
235             msgbox('Please Choose An Audio And Try Again!', 'Error', 'error');
236         end
237     end
238
239         % Button pushed function: PlayFilteredAudioButton
240     function PlayFilteredAudioButtonPushed(app, event)
241         if app.CheckConditions == 1
242             plot(app.UIAxes,app.OpSound, 'Color', [0 0 0]);
243             NewFs = app.Slider.Value * app.Fs;
244             plot(app.UIAxes_2,abs(fftshift(fft(app.OpSound)) / NewFs), 'Color', [0 0 0]);
245             sound(app.OpSound, NewFs);
246         else
247             msgbox('Please Choose Your Requirements, Then Click Apply Filter First!', 'Error', 'error');
248         end
249     end
250
251         % Callback function
252     function HZSlider_9ValueChanged(app, event)
253         value = app.HZSlider_9.Value;
254     end
255
256         % Button pushed function: PlotOriginalButton
257     function PlotOriginalButtonPushed(app, event)
258         if app.CheckAudio == 1
259             plot(app.UIAxes,app.Ip, 'Color', [0 0 0]);
260             plot(app.UIAxes_2,abs(fftshift(fft(app.Ip)) / app.Fs), 'Color', [0 0 0]);
261         else
262             msgbox('Please Choose An Audio And Try Again!', 'Error', 'error');
263         end
264     end
265
266         % Button pushed function: PlotFilteredButton
267     function PlotFilteredButtonPushed(app, event)
268         if app.CheckConditions == 1
269             NewFs = app.Slider.Value * app.Fs;
270             plot(app.UIAxes,app.OpSound, 'Color', [0 0 0]);
271             plot(app.UIAxes_2,abs(fftshift(fft(app.OpSound)) / NewFs), 'Color', [0 0 0]);
272         else
273             msgbox('Please Choose Your Requirements, Then Click Apply Filter First!', 'Error', 'error');
274         end
275     end
276
277         % Button pushed function: StopAudioButton
278     function StopAudioButtonPushed(app, event)
279         clear sound;
280     end
281 end
282
283         % Component initialization
284     methods (Access = private)
285
286         % Create UIFigure and components
287     function createComponents(app)
288
289         % Get the file path for locating images
290         pathToMLAPP = fileparts(mfilename('fullpath'));  

291         function app = app1
292
293             % Create UIFigure and components
294             createComponents(app) ...
295
296             % Delete UIFigure when app is deleted
297             delete(app.UIFigure)
298         end
299     end
300 end

```

# SAMPLE RUN

