Class works:

Task 1 : Design of IIR filters

Code:

```
clc;
clear all;
close all;
FN=1000/2;
fc=300; %cut off freq
N=5; %filter order
[z,p,k]=buttap(N); %creating analog filter
w=linspace(0,FN/fc,1000);
h=freqs(k*poly(z),poly(p),w);
f=fc*w;
plot(f,20*log10(abs(h))),grid
ylabel('Magnitude(dB)')
xlabel('frequency(Hz)')
```

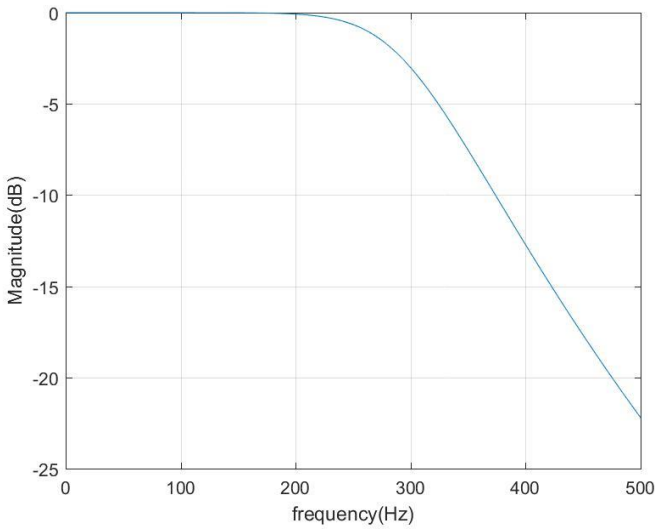

Figure 6.1 : IIR filter design

Task 2: Design of IIR filters by impulse invariant method.

Code:

```
Fs=1000; %sampling freq
fc=300; %cutoff freq
WC=2*pi*fc; %cutoff freq in radian
N=5;
[b,a]=butter(N,WC,'s'); %creating analog filter
[z,p,k]=butter(N,WC,'s');
[bz,az]=impinvar(b,a,Fs); %determine coeffs of IIR filter
[h,f]=freqz(bz,az,512,Fs); %512 points are taken,can be changed
plot(f,20*log10(abs(h))),grid
ylabel('Magnitude(dB)')
xlabel('frequency(Hz)')
```

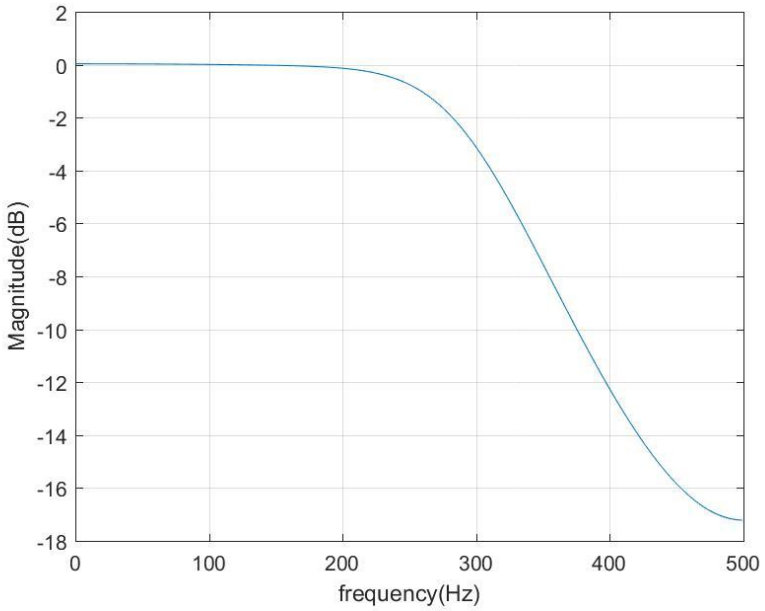

Figure 6.2 : Design of IIR filters by impulse invariant method

Task 3: Design of IIR filters by impulse invariant method.

Code:

```
N=2; %filter order
Fs=1280; %sampling freq
fc=150; %cutoff freq
WC=2*pi*fc; %cutoff freq in radian
 [b,a]=butter(N,WC,'s'); %creating analog filter
[z,p,k]=butter(N,WC,'s');
[bz,az]=impinvar(b,a,Fs); %determine coeffs of IIR filter
subplot(2,1,1) %plot magnitude freq. response
[H,f]=freqz(bz,az,512,Fs); %512 points are taken,can be changed
plot(f,20*log10(abs(H)))
ylabel('Magnitude Response(dB)')
xlabel('frequency(Hz)')
subplot(2,1,2) %plot pole zero diagram
zplane(bz,az)
zz=roots(bz) %poles and zeros
pz=roots(az) %poles in z plane
```

Output:

```
zz = 0

pz =
  0.5154 + 0.2955i
  0.5154 - 0.2955i
```

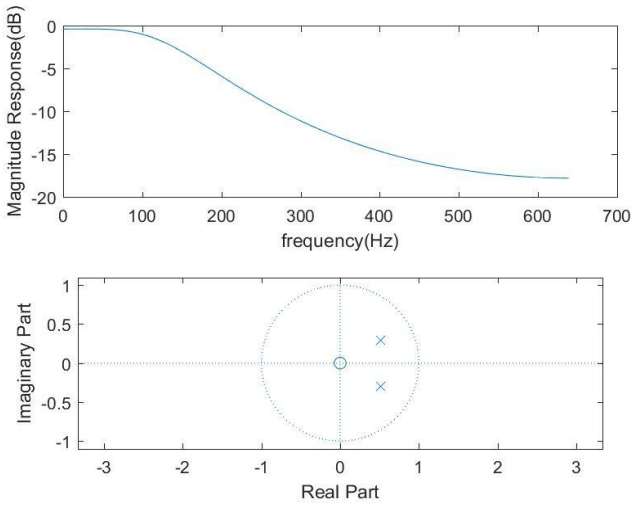

Figure 6.3 : Design of IIR filters by impulse invariant method

Task 4:Design of IIR filters by filter,overlap,impulse invariant, bilinear method

```
clc;
clear all;
close all;
fs = 1e4;
t = 0:1/fs:5;
sw = sin(2*pi*262.62*t);
n = 0.1 * randn(size(sw));
swn = sw + n;
%butterworth filter of order 2 with a cutoff at 400H
N = 2;
FN = fs/2;
fc = 150;
FC = fc/FN;
[b, a] = butter(2, 400/(fs/2));
figure(1);
w = linspace(0, FN/fc, 400);
[h, f] = freqz(b, a);
plot(f, 20*log10(abs(h)));
grid;
ylabel('Magnitude (dB)');
xlabel('Frequency (Hz)');
%filter with filter
y1 = filter(b,a,swn);
figure(2);
subplot(311);
plot(t,sw), axis([0 0.04 -1.1 1.1]), title('Original Signal');
soundsc(sw,1e4)
subplot(312);
plot(t,swn), axis([0 0.04 -1.1 1.1]), title('Noisy Signal');
soundsc(swn,1e4)
subplot(313);
plot(t,y1), axis([0 0.04 -1.1 1.1]), title('Using Filter');
soundsc(y1,1e4)
```

```
%zero-phase filter
y2 = filtfilt(b,a,swn);
figure(3);
subplot(311);
plot(t,sw), axis([0 0.04 -1.1 1.1]), title('Original Signal');
soundsc(sw,1e4)
subplot(312);
plot(t,swn), axis([0 0.04 -1.1 1.1]), title('Noisy Signal');
soundsc(swn,1e4)
subplot(313);
plot(t,y2), axis([0 0.04 -1.1 1.1]), title('Using Filtfilt(zero-phase
filter)');
soundsc(y2,1e4)
%impulse invariant method
[bz1, az1] = impinvar(b, a, fs);
y3 = filter(bz1,az1,swn);
figure(4);
subplot(311);
plot(t,sw), axis([0 0.04 -1.1 1.1]), title('Original Signal');
soundsc(sw,1e4)
subplot(312);
plot(t,swn), axis([0 0.04 -1.1 1.1]), title('Noisy Signal');
soundsc(swn,1e4)
subplot(313);
plot(t,y3), axis([0 0.04 -1.1 1.1]), title('Using Filter(Impulse
Invariant)');
soundsc(y3,1e4)
%bilinear transfer method
[bz2, az2] = bilinear(b, a, fs);
y4 = filter(bz2,az2,swn);
figure(5);
subplot(311);
plot(t,sw), axis([0 0.04 -1.1 1.1]), title('Original Signal');
soundsc(sw,1e4)
subplot(312);
plot(t,swn), axis([0 0.04 -1.1 1.1]), title('Noisy Signal');
soundsc(swn,1e4)
subplot(313);
plot(t,y4), axis([0 0.04 -1.1 1.1]), title('Using Filter(Bilinear)');
soundsc(y4,1e4)
%% overlap add method method
y5 = fftfilt(b,swn);
figure(6);
subplot(311);
plot(t,sw), axis([0 0.04 -1.1 1.1]), title('Original Signal');
%%
%%
soundsc(sw,1e4)
subplot(312);
plot(t,swn), axis([0 0.04 -1.1 1.1]), title('Noisy Signal');
 soundsc(swn,1e4)
subplot(313);
plot(t,y5), axis([0 0.04 -1.1 1.1]), title('Using Filter(Overlap Add)');
soundsc(y5,1e4)
```

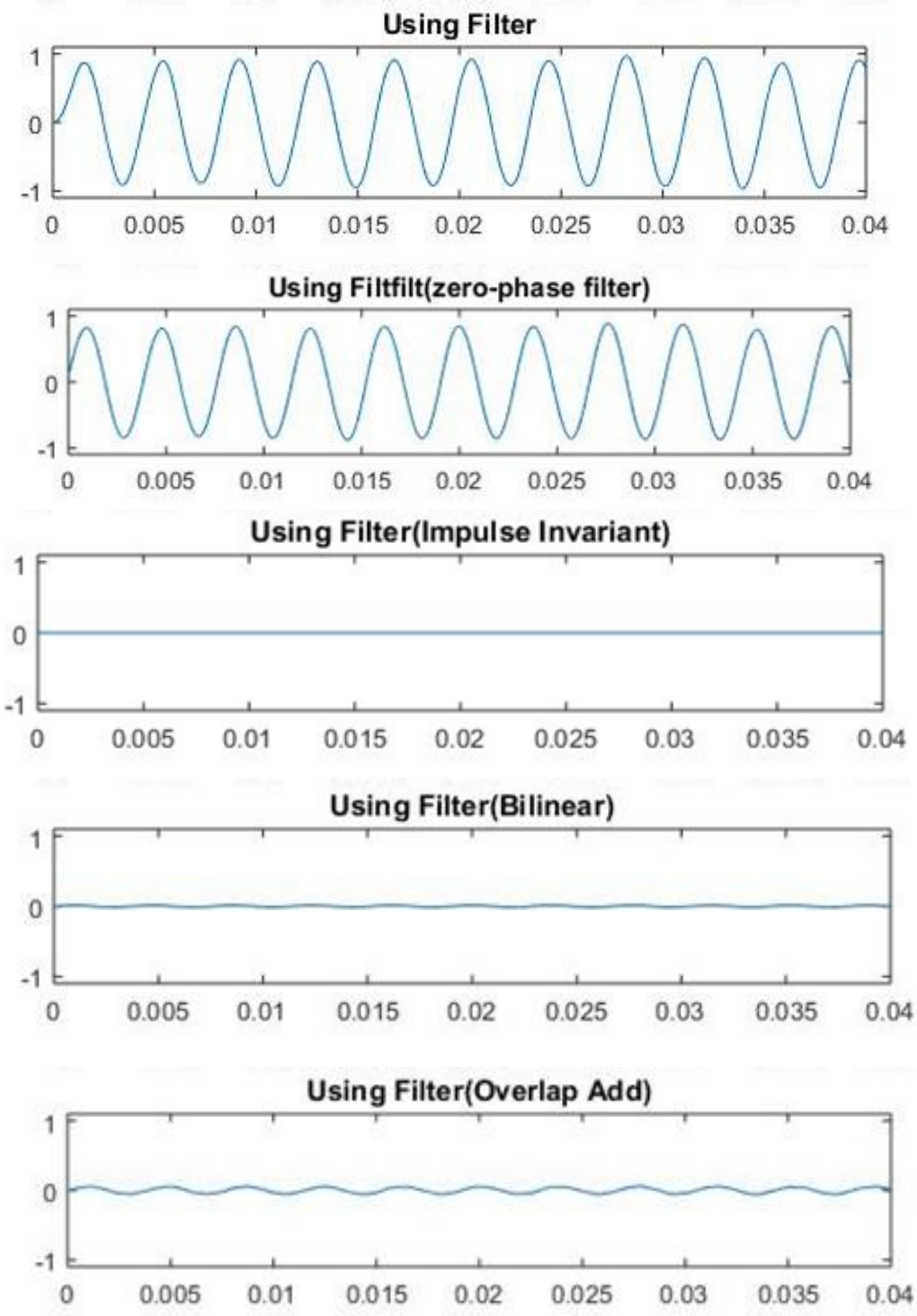

Figure 6.10 : Output comparison for different filters

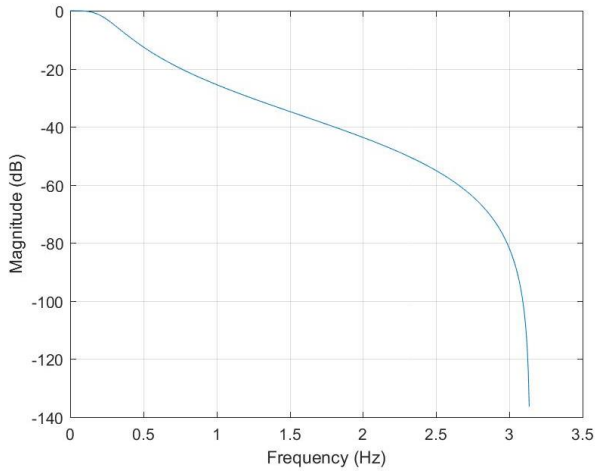

Figure 6.4: butterworth filter of order 2 with a cutoff at 400H

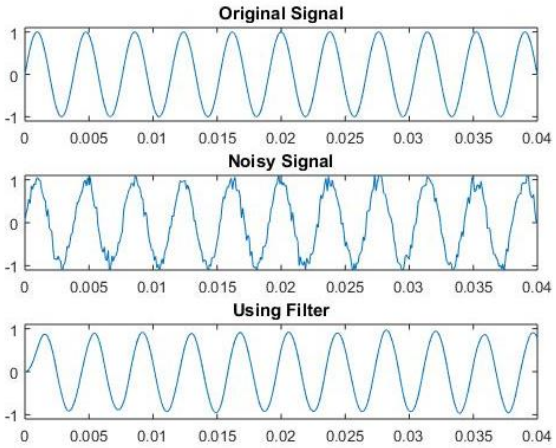

Figure 6.5: filter with filter

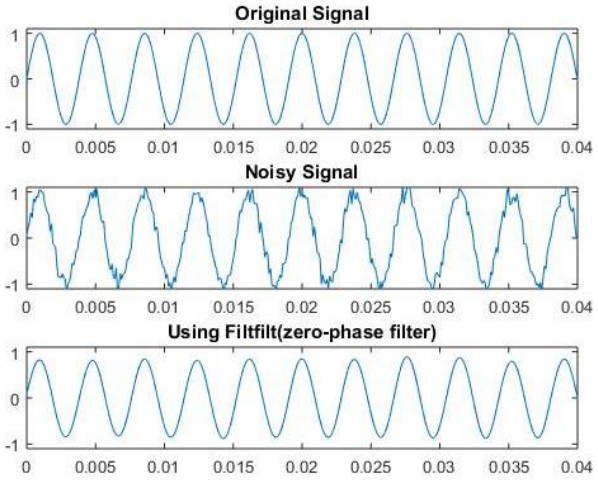

Figure 6.6: zero-phase filter

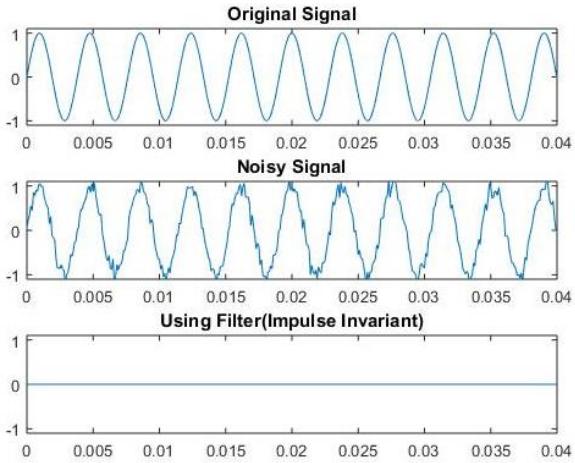

Figure 6.7:filter with impulse invariant method

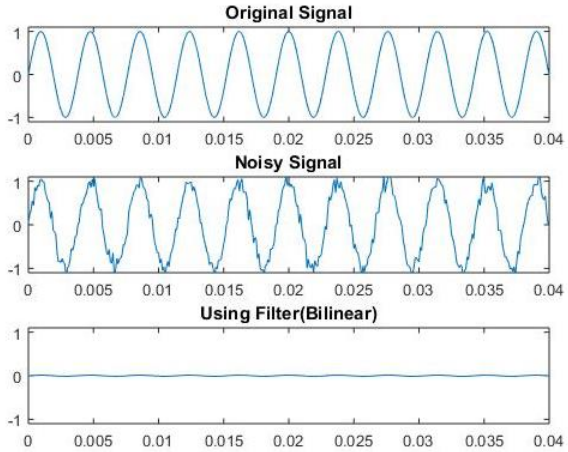

Figure 6.8:filter with bilinear transfer method

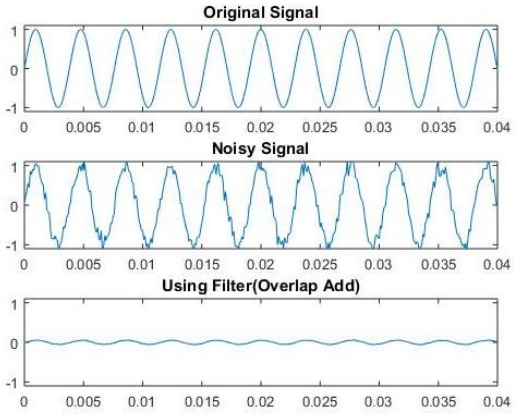

Figure 6.9: overlap add method method