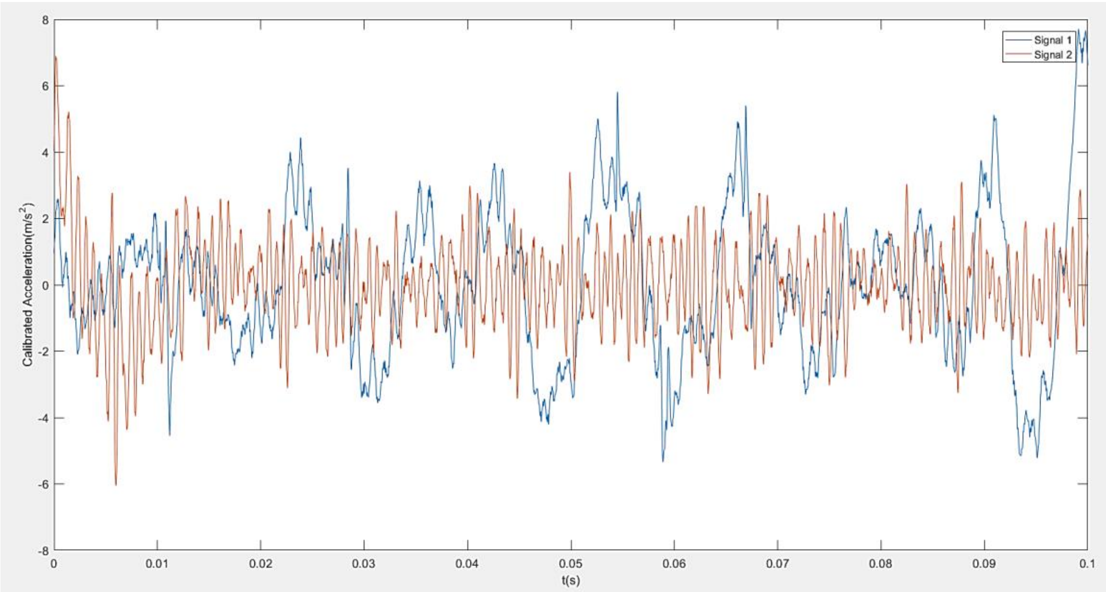
Problem Set #4

The MatLab codes are shown after each question.

Question 1

Task 1

We know that the sensitivities of accelerometer 1 and 2 are 60.2 and 59.9 mV/g. After applying it into the raw data, plot the first 0.1 s calibrated acceleration signals.



Task 2

After calculated by MatLab, the value of mean, peak, peak-to-peak and RMS of the calibrated acceleration signals are shown in the following table.

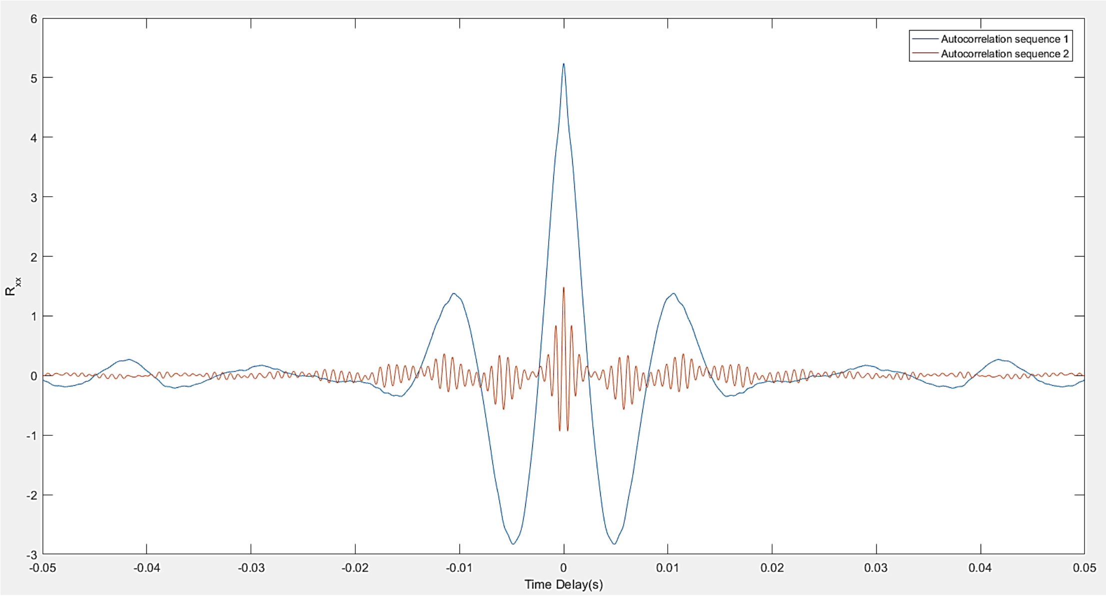
	Mean	Peak	Peak-to-Peak	RMS
Signal 1	0.037924	7.708293	13.04967068	2.288988
Signal 2	0.003764	6.894621	12.94291498	1.219409

Task 3

Calculate and plot the autocorrelation sequences of the calibrated acceleration signals. The figure is shown in the next page with the x-axis limits to a time lag range of -0.05 to 0.05 s.

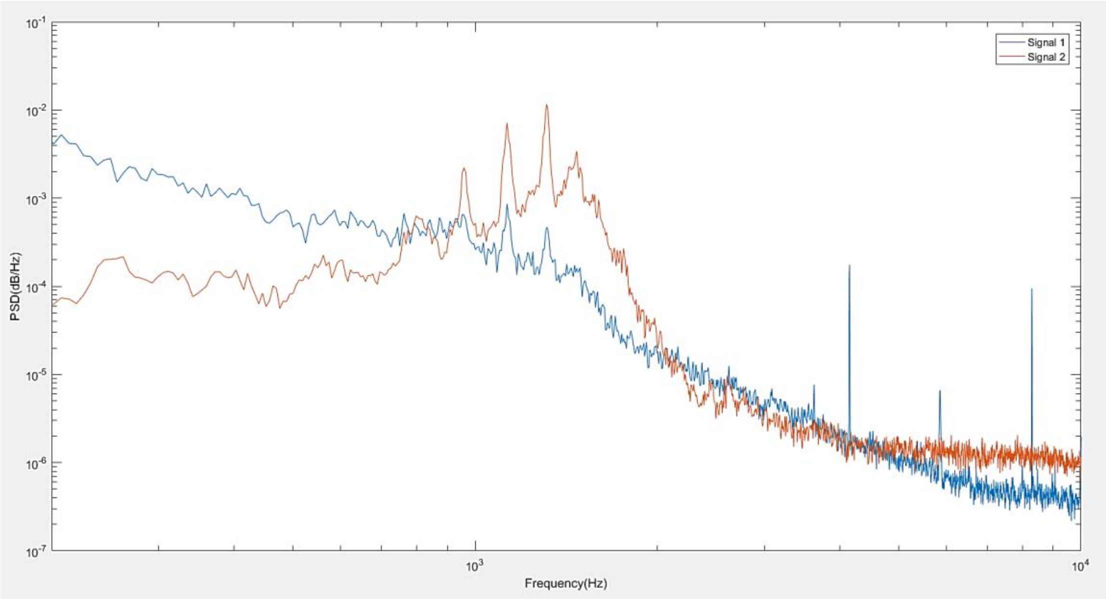
Task 4

From the figure, we can find that signal 1 contains a time repeating component. Because around t=0 and t=±0.01, it has a broad peak in the autocorrelation figure, which indicates that it correlated with some time, and has time repetitive component. On another hand, in autocorrelation figure of signal 2, each peak is very sharp, which indicates that it is not correlated with itself, and has no time repetitive component.



Task 5

Calculate and plot (on the same figure) the power spectral density (PSD) of the calibrated acceleration signals by MatLab. The figure is shown below with a loglog scale and set the x-axis limits to a frequency range of 200 Hz to 10 kHz.



Task 6

With the increase in frequency, the overall trend of power spectral density of the two signals is declining. However, there are 3 to 4 peaks from 800 Hz to 1470 Hz.

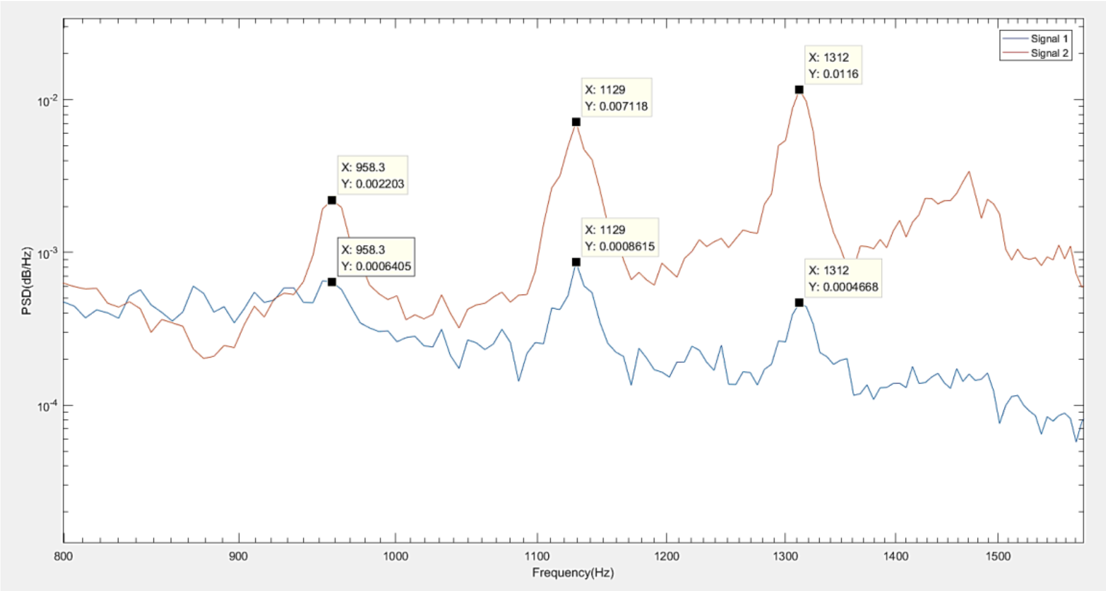
The frequency resolution of the spectra is calculated by the formula:

$$f_{res} = \frac{Fs}{window} = \frac{50000}{2^{13}} = \ 6.1035$$

The zoom in figure of power spectral density (PSD) with the x-axis limits to a frequency range of 800 Hz to 1470 Hz. The blade pass frequency of the fan is calculate by the formula:

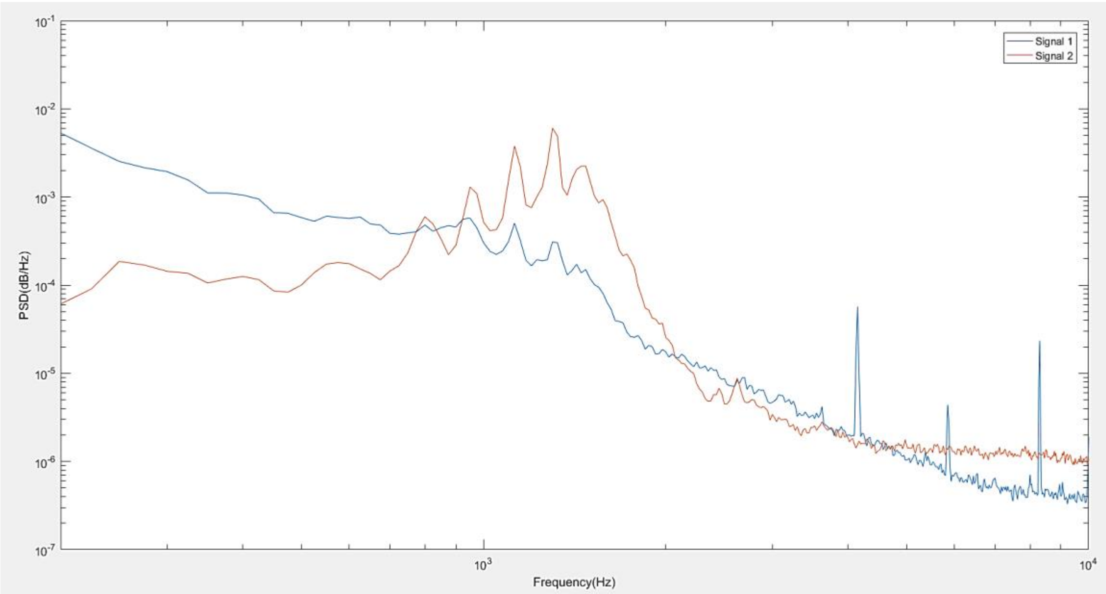
$$\mathrm{BPF} = \frac{nt}{60}$$

Where BPF = Blade Pass Frequency (Hz), n = rotation velocity (rpm), t = number of blades
The machine contains a small 7 bladed fan that operates at between 1450 and 1600 rpm, and the motor of the machine operates at 2975 rpm. Thus, after converted to Hz, the value of the fan is range from 169Hz to 187Hz, and the motor is 50 Hz. The difference between each peak in the figure is 171 Hz and 183 Hz. So we can conclude that the peak components visible at frequencies between 800 and 1470 Hz is the response of the fan.



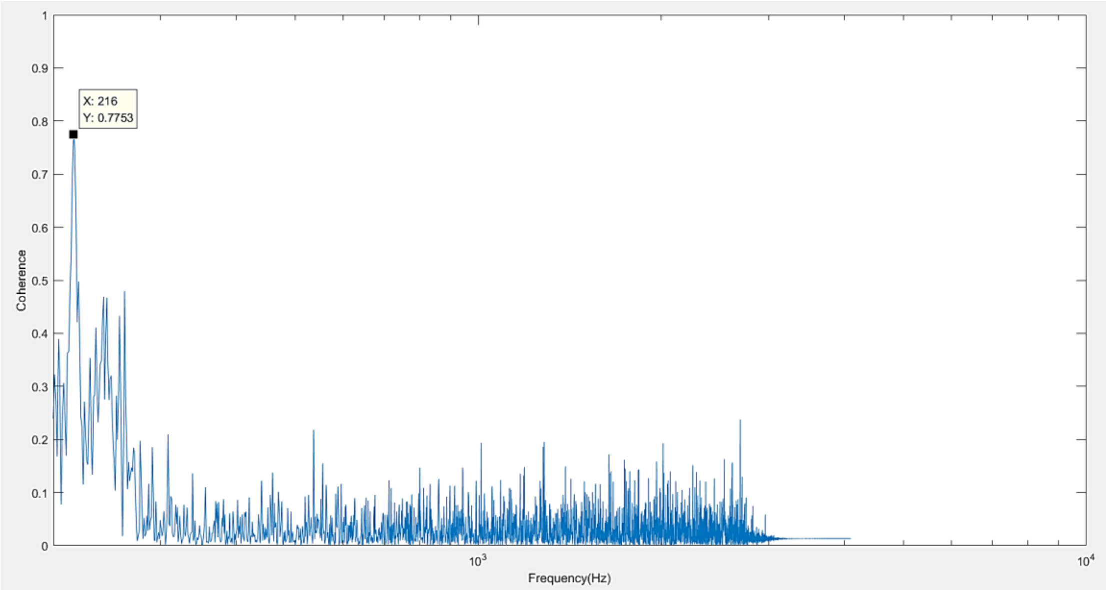
Task 7

Plot the PSD of the calibrated acceleration signals with a frequency resolution of 25 Hz and using 25% overlap. The frequency resolution refers to the ability to discriminate the spectral features. As the frequency resolution increased, the figure looks smooth.



Task 8

Calculate and plot the magnitude-squared coherence of the calibrated acceleration signals with a log scale on the x-axis only and set the x-axis limits to a frequency range of 200 Hz to 10 kHz. The figure is shown in the next page.



Task 9

The accelerometer signals are most coherent at the frequency more than 0.5, thus, at 216 Hz shown in the figure above.

MatLab Code:

```
clc;
clear;
%Task 1
load('VelocityData.mat')
x1 = Vdata_1(1:5000)/0.0602*9.8;
x2 = Vdata_2(1:5000)/0.0599*9.8;
t = linspace(0,0.1,5000);
figure(1);
plot(t,x1);
hold on;
plot(t,x2);
hold off
legend('Signal 1','Signal 2');
xlabel('t(s)');
ylabel('Calibrated Acceleration(m/s^2)');

%Task 2
Mean1 = mean(x1);%mean of signal 1
Mean2 = mean(x2);%mean of signal 2
Max1 = max(x1);
Min1 = min(x1);
Max2 = max(x2);
Min2 = min(x2);
```

```
PTP1 = Max1-Min1;%peak to peak of signal 1
PTP2 = Max2-Min2;%peak to peak of signal 2
RMS1 = rms(Vdata_1/0.0602*9.8);%RMS of signal 1
RMS2 = rms(Vdata_2/0.0599*9.8);%RMS of signal 1

%Task 3
as1 = xcorr(Vdata_1/0.0602*9.8,2500)/200000;
as2 = xcorr(Vdata_2/0.0599*9.8,2500)/200000;
t2 = linspace(-0.05,0.05,5001);
figure(2);
plot(t2,as1);
hold on
plot(t2,as2);
hold off
legend('Autocorrelation sequence 1','Autocorrelation
sequence 2');
xlabel('Time Delay(s)');
ylabel('R_x_x');

%Task 5
[Gxx1,~] =
pwelch(Vdata_1/0.0602*9.8,2^13,2^11,2^13,50000);
[Gxx2,f] =
pwelch(Vdata_2/0.0599*9.8,2^13,2^11,2^13,50000);
figure(3);
loglog(f,Gxx1);
hold on
loglog(f,Gxx2);
axis([200 10^4 10^(-7) 10^(-1)]);
legend('Signal 1','Signal 2');
xlabel('Frequency(Hz)');
ylabel('PSD(dB/Hz)');

%Task 7
[Gxx3,~] =
pwelch(Vdata_1/0.0602*9.8,2000,500,2000,50000);
[Gxx4,f2] =
pwelch(Vdata_2/0.0599*9.8,2000,500,2000,50000);
figure(4);
loglog(f2,Gxx3);
hold on
loglog(f2,Gxx4);
axis([200 10^4 10^(-7) 10^(-1)]);
legend('Signal 1','Signal 2');
```

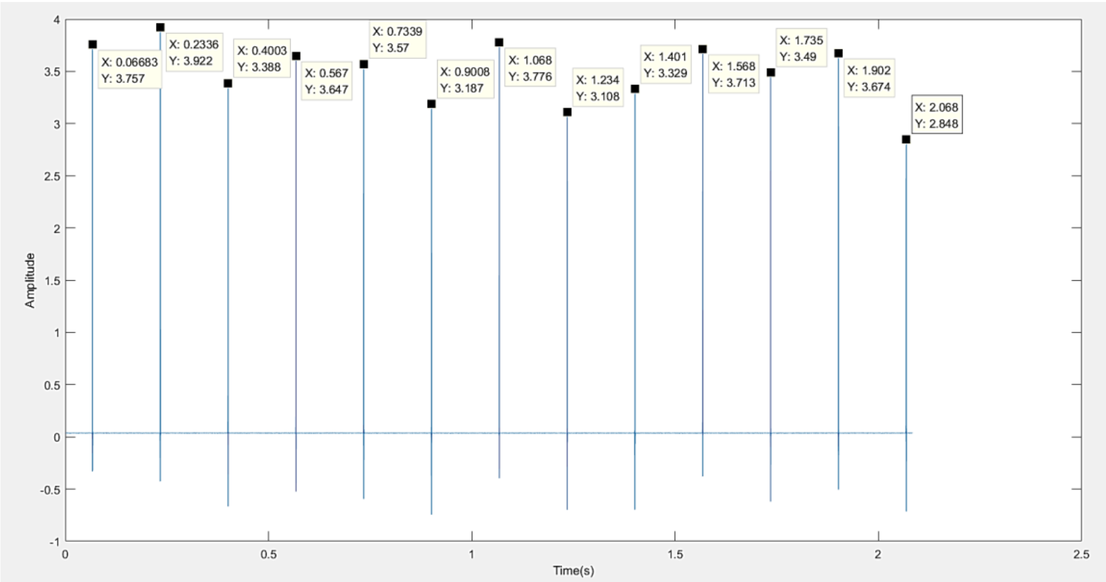
```
xlabel('Frequency(Hz)');
ylabel('PSD(dB/Hz)');

%Task 8
[cxy, f3] =
mscohere(Vdata_1/0.0602*9.8,Vdata_2/0.0599*9.8,2^13,2
^11,2^13,50000);
figure(5);
semilogx(cxy);
set(gca, 'XScale', 'log');
axis([200 10^4 0 1]);
xlabel('Frequency(Hz)');
ylabel('Coherence');
```

Question 2

Task 1

Preliminary calculations:

The speed of the shaft in the test rig should be identified by analyzing the tacho signal by using the mean time between pulses. The following figure shows when each pulse happens.



The mean time between pulses is $\frac{2.068-0.06683}{12} = 0.1668$s. Thus, the shaft rotational speed is

$f_r = \frac{1}{0.1668} = 5.9955$. The relevant potential bearing fault frequencies should be calculated by the formulas:

$$BPFO = \frac{n f_r}{2}\left\{1-\frac{d}{D}\cos\phi\right\} \quad \textbf{Outer race}$$

$$BPFI = \frac{n f_r}{2}\left\{1+\frac{d}{D}\cos\phi\right\} \quad \textbf{Inner race}$$

$$FTF = \frac{f_r}{2}\left\{1-\frac{d}{D}\cos\phi\right\} \quad \textbf{Cage}$$

$$BSF = \frac{f_r D}{2d}\left\{1-\left(\frac{d}{D}\cos\phi\right)^2\right\} \quad \textbf{Rolling element}$$

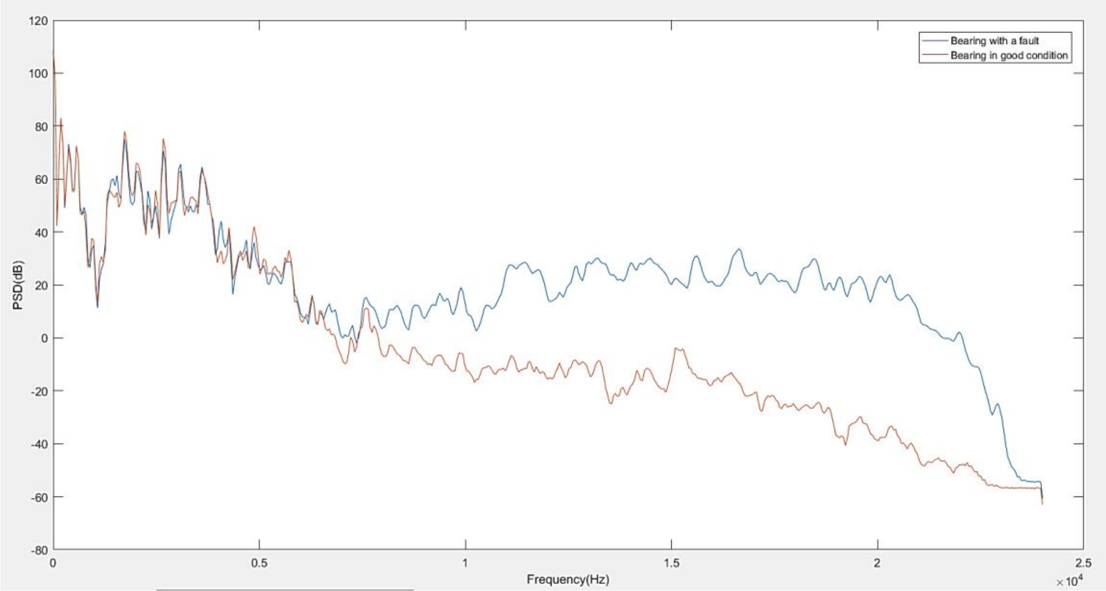
The values are shown in the following table.

BPFO	BPFI	FTF	BSF
29.32036	42.62571	2.443363	15.6553698

Task 2

Compare PSDs and select demodulation band:

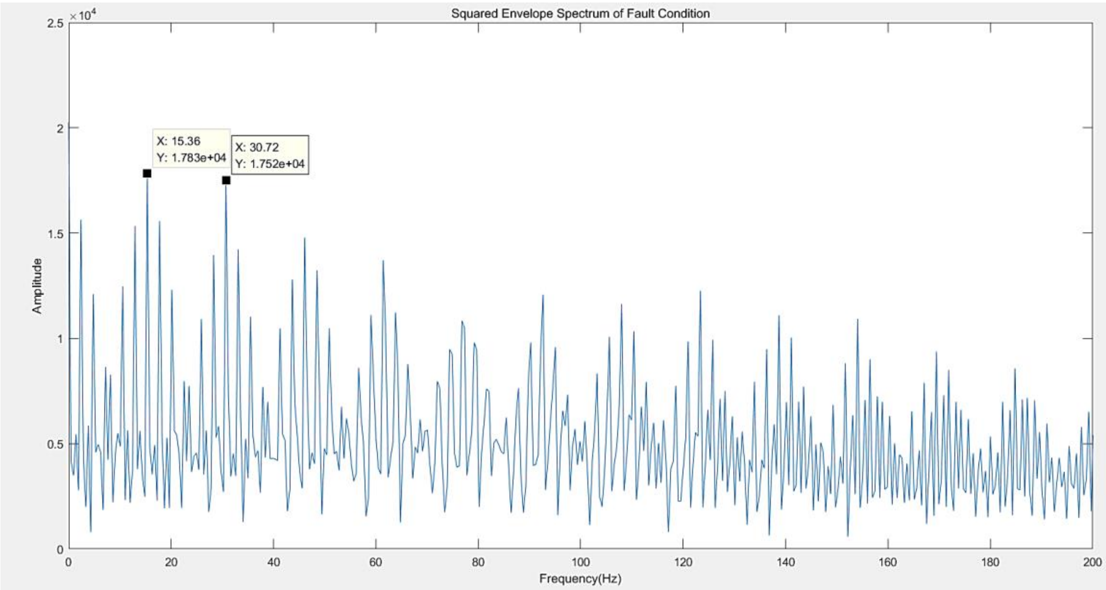
Compare the PSD of the faulty bearing signal with that of the reference (fault-free) condition. The settings are 1024 point transform and Hamming window length, and 50% overlap. The figure is shown in the next page according to the question requirements. The figure indicates that the band should be chosen from 10 kHz to 20 kHz. We choose the number of samples as 2000. And the width of the band is 48000/100000*2000=960 Hz.

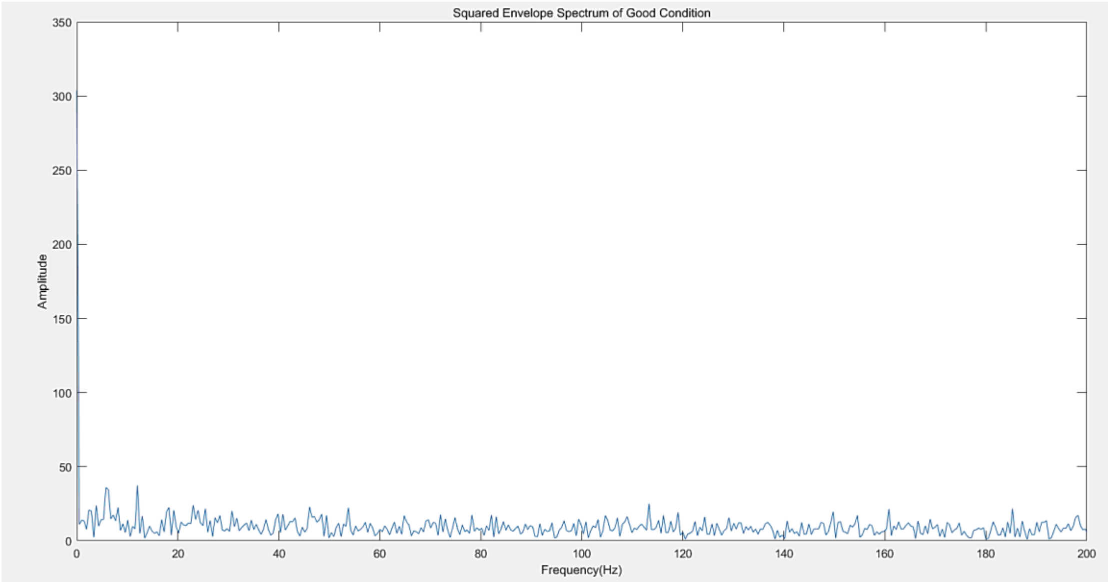


Task 3

Conduct amplitude demodulation and form the squared envelope spectrum:

Following the steps introduced in the question, the squared envelope spectrums of fault and good condition are plotted in the figure below.





Task 4:

Analyse envelope spectrum and make diagnosis:

Envelope spectrum will have harmonics of BSF, with sidebands spaced at cage speed. The even harmonics of BSF are often dominant (sometimes the component at the fundamental frequency (1 x BSF) is quite small). As the figure shown above, the fault condition indicates that at 15.36 Hz, the energy reaches a peak, and at 30.72 Hz, the component dominates. The good condition indicates that the energy keeps few.

Task 5:

According to the result in task 1, the value of BSF is 15.6553698. Thus, the fault appears at rolling element.

MatLab Code:

Task 1&2:

```
clc
clear
load('bf.mat');
load('bg.mat');

%Task 1
%system parameters
d = 7.12*10^(-3);
D = 38.5*10^(-3);
n = 12;
phi = 0;
tf = 100000/48000;
t = linspace(0,tf,100000);

figure(1);
plot(t,bg(:,2));
xlabel('Time(s)');
ylabel('Amplitude');
[pk,tpk] =
findpeaks(bg(:,2),t,'MinPeakProminence',2);
fr = 1/((tpk(length(tpk))-tpk(1))/(length(tpk)-1));

BPFO = n*fr/2*(1-d/D*cos(phi));%Outer race
BPFI = n*fr/2*(1+d/D*cos(phi));%Inner race
FTF = fr/2*(1-d/D*cos(phi));%Cage
BSF = fr*D/2/d*(1-(d/D*cos(phi))^2);%Rolling element

%Task 2
[Gxxf,f] = pwelch(bf(:,1),1024,512,1024,48000);
[Gxxg,f] = pwelch(bg(:,1),1024,512,1024,48000);
Gxxflog = 20*log10(Gxxf/(10^(-6)));
Gxxglog = 20*log10(Gxxg/(10^(-6)));
figure(2);
plot(f,Gxxflog)
hold on
plot(f,Gxxglog);
legend('Bearing with a fault','Bearing in good
condition');
xlabel('Frequency(Hz)');
ylabel('PSD(dB)');
```

Task 3:

```
clc
clear
load('bf.mat');
load('bg.mat');
fs=48000;
N=100000;
dfs=fs/N;

bF=bf(:,1);
bF1=fft(bF);
bF2=zeros(2000,1);
for ii=1:1000
   bF2(ii,1)=bF1((75000+ii),1);
end
bF3=ifft(bF2);
bF4=(abs(bF3)).^2;
bF5=fft(bF4);
f=(0:length(bF5)-1)*dfs;
figure(1)
plot(f,abs(bF5));
xlim([0 200]);
xlabel('Frequency(Hz)');
ylabel('Amplitude');
title('Squared Envelope Spectrum of Fault Condition')

bG=bg(:,1);
bG1=fft(bG);
bG2=zeros(2000,1);
for ii=1:1000
   bG2(ii,1)=bG1((75000+ii),1);
end
bG3=ifft(bG2);
bG4=(abs(bG3)).^2;
bG5=fft(bG4);
f=(0:length(bG5)-1)*dfs;
figure(2)
plot(f,abs(bG5));
xlim([0 200]);
xlabel('Frequency(Hz)');
ylabel('Amplitude');
title('Squared Envelope Spectrum of Good Condition')
```