Introduction:

The goal of this project is to explore the effects of noise on frequency modulation (FM), phase modulation (PM), and two kinds of amplitude modulation (conventional AM, SSB AM). As people attempt to utilize these modulation schemes to transmit information over a significant distance, we discover the pros and cons of them:

FM:

- Power-efficient (theoretical high SNR) => high quality
- Infinite bandwidth
- Short transmission distance

PM:

- Simple demodulation
- Power-efficient (theoretical high SNR) => high quality
- Infinite bandwidth

AM:

- Efficient transmission distance
- Poor quality
-

SSB:

- Bandwidth-efficient
- Strict hardware requirement for implementation to ensure quality

Table of Contents


```
clc; clear; close all;
```

Setup

```
% initializing the signal m from .wav file and setting sampling
 frequence
% fs.
load handel.mat;
filename='handel.wav';
[y,Fs]=audioread(filename);
samples=[1,length(y)-(5*Fs)];
[m,fs]=audioread(filename,samples);

%sound(m, fs);
% normalizing m.
m = transpose(m(:,1) ./ max(abs(m(:,1))));

N = length(m);
T = N/fs;
fc = 55000;

%upsampling
scale = 3*ceil(fc/fs);
N_up = scale * N;
fs_up = scale * fs;
t = linspace(0,T,N);
t_up = linspace(0,T,N_up);
wd = linspace(-pi, pi, N_up);
f = wd *fs_up / (2 * pi);
fdemod = linspace(-pi, pi, N) * fs / (2 * pi);
fc = 55000;

sSin = sin(2*pi * fc * t_up);
m_P = rms(m).^2;
W_m = obw(m, fs);
```

<== Power of the input audio file

<== Bandwidth of the input audio file

FM Mod & Demod

```
Ac_FM = 1.21;
k_FM = 70000;
```

<== carefully chosen values to make sure the power of the demodulated signal matches the input power

```
                                        Carrier Frequency
                                        ||
m_FM = FM_mod(m, fs, Ac_FM, 55000, k_FM);
m_FMfreq = fftshift(fft(m_FM)/fs_up);

figure('Position', [0 0 1024 1024]);
subplot(2,1,1)
plot(t_up,m_FM);
title("FM Modulated m(t)");xlabel("Time (s)"); ylabel("u(t)");
subplot(2,1,2)
semilogy(f, abs(m_FMfreq));
title("Fourier Transform of FM Modulated m(t)");
xlabel("Frequency (Hz)"); ylabel("Amplitude");
suptitle("FM Modulation");                   Cutoff Frequency
                                             ||
m_FM_demod = FM_demod(m_FM, fs_up, 2000,f, scale);
m_FM_demodfreq = fftshift(fft(m_FM_demod)/fs);

figure('Position', [0 0 1024 1024]);
subplot(2,1,1)
plot(t,m_FM_demod);
title("FM Demodulated m(t)");xlabel("Time (s)"); ylabel("u(t)");
subplot(2,1,2)
semilogy(fdemod, abs(m_FM_demodfreq));
title("Fourier Transform of FM Demodulated m(t)");
xlabel("Frequency (Hz)"); ylabel("Amplitude");
suptitle("FM Demodulation");

%sound(m_FM_demod, fs);

m_FM_P = rms(m_FM_demod).^2;    <== Power of the demodulated signal
W_FM = obw(m_FM_demod, fs);
                                 <== Bandwidth of the demodulated signal

var = [0.01, 0.05, 0.1];
FM_noiseless = rms(m_FM_demod).^2;

for i = 1:3

    v = var(i);
    noise = sqrt(var(i)) * randn(1,length(m_FM));   <== Generating AGWN
    m_FM_noise = m_FM + noise;   <== Adding noise to signal
    m_FM_noisefreq = fftshift(fft(m_FM_noise)/fs_up);
    m_FM_noise_demod = FM_demod(m_FM_noise, fs_up, 2000,f, scale);
    m_FM_noise_demod = lowpass(m_FM_noise_demod, 2000, fs);
    m_FM_noise_demodfreq = fftshift(fft(m_FM_noise_demod)/fs);

    figure('Position', [0 0 1024 1024]);
    subplot(2, 2, 1);
    semilogy(f, abs(m_FMfreq));
    title("  Magnitude of Noise Free Input");
    xlabel("Frequency (Hz)"); ylabel("Amplitude");
    subplot(2, 2, 2);
    semilogy(fdemod, abs(m_FM_demodfreq));
    title("  Magnitude of Modulated Noise Free Input");
    xlabel("Frequency (Hz)"); ylabel("Amplitude");
    subplot(2, 2, 3);
```

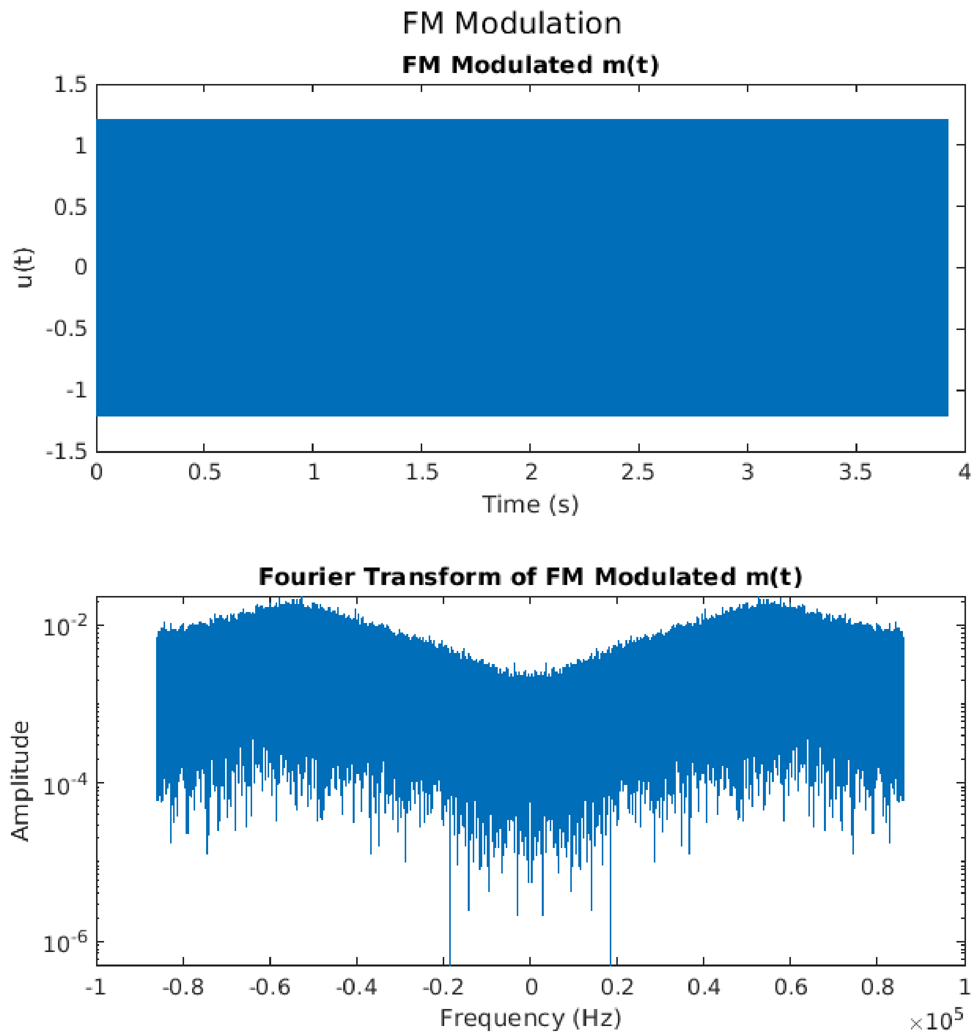
```
        semilogy(f, abs(m_FM_noisefreq));
        title("  Magnitude of Modulated Noisy Signal");
        xlabel("Frequency (Hz)"); ylabel("Amplitude");
        subplot(2, 2, 4);
        semilogy(fdemod, abs(m_FM_noise_demodfreq));
        title("  Magnitude of Demodulated Noisy Signal");
        xlabel("Frequency (Hz)"); ylabel("Amplitude");
        suptitle("FM with Noise \sigma^2 = " + var(i));

        noisy = rms(m_FM_noise_demod - m_FM_demod).^2;
        snr_FM = 10 * log10(FM_noiseless/noisy);
        display("SNR for FM with variance" + var(i) + ": " + snr_FM);

        snr_FM_theo = (Ac_FM^2 * k_FM^2 * m_FM_P) / (4 * var(i) * W_FM^3 /
     fs);
        display("Theoretical SNR for FM with variance " + var(i) + ": " +
     snr_FM_theo);
    end

    figure('Position', [0 0 1024 1024]);
    for j = 1:3
        var = 0.01;
        Ac_FM = 1.21;
        k_FM = 70000 * 2^(j-2);
        m_FM = FM_mod(m, fs, Ac_FM, 55000, k_FM);
        m_FMfreq = fftshift(fft(m_FM)/fs_up);
        m_FM_demod = FM_demod(m_FM, fs_up, 2000,f, scale);
        m_FM_demodfreq = fftshift(fft(m_FM_demod)/fs);
        FM_noiseless = rms(m_FM_demod).^2;
        noise = sqrt(var) * randn(1,length(m_FM));
        m_FM_noise = m_FM + noise;
        m_FM_noisefreq = fftshift(fft(m_FM_noise)/fs);
        m_FM_noise_demod = PM_demod(m_FM_noise, 1, fs_up, 2000, scale,
     sSin);
        m_FM_noise_demod = lowpass(m_FM_noise_demod, 2000, fs);
        m_FM_noise_demodfreq = fftshift(fft(m_FM_noise_demod)/fs);

        noisy = rms(m_FM_noise_demod - m_FM_demod).^2;
        snr_FM = 10 * log10(FM_noiseless/noisy);
        display("SNR for FM with k = " + k_FM + "(variance = "+var+") is:
     " + snr_FM);

        snr_FM_theo = (Ac_FM^2 * k_FM^2 * m_FM_P) / (4 * var * W_FM^3 /
     fs);
        display("Theoretical SNR for FM with k = " + k_FM + "(variance =
     "+var+") is: " + snr_FM_theo);

        subplot(2, 2, 1);
        str = strcat('k = ' , int2str(k_FM));
        semilogy(f, abs(m_FMfreq), 'DisplayName', str);
        title("  Magnitude of Noise Free Input");
        xlabel("Frequency (Hz)"); ylabel("Amplitude");
        legend;
        hold on
```

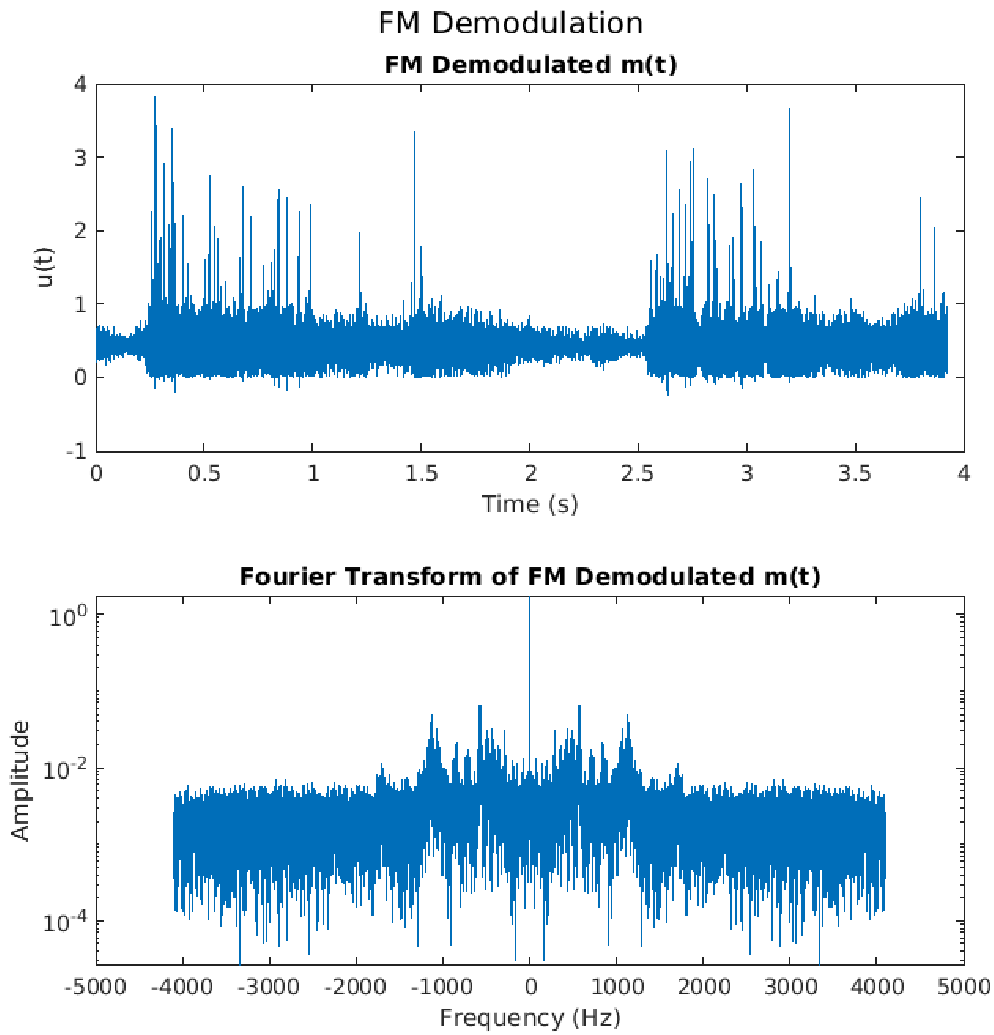
<== Calculating Experimental SNR using the formula given

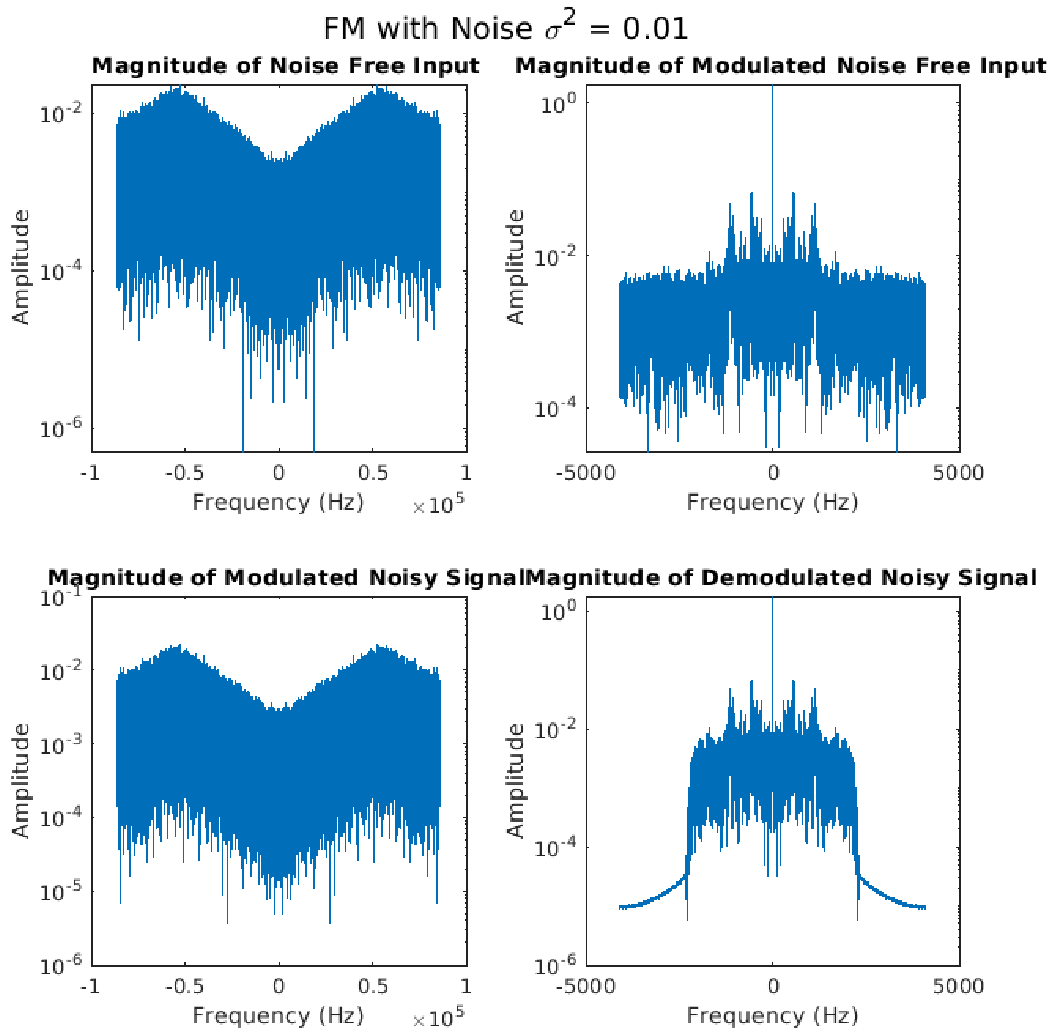
SNR_FM = Ac^2(kf^2)Pm / (2NoW^3)

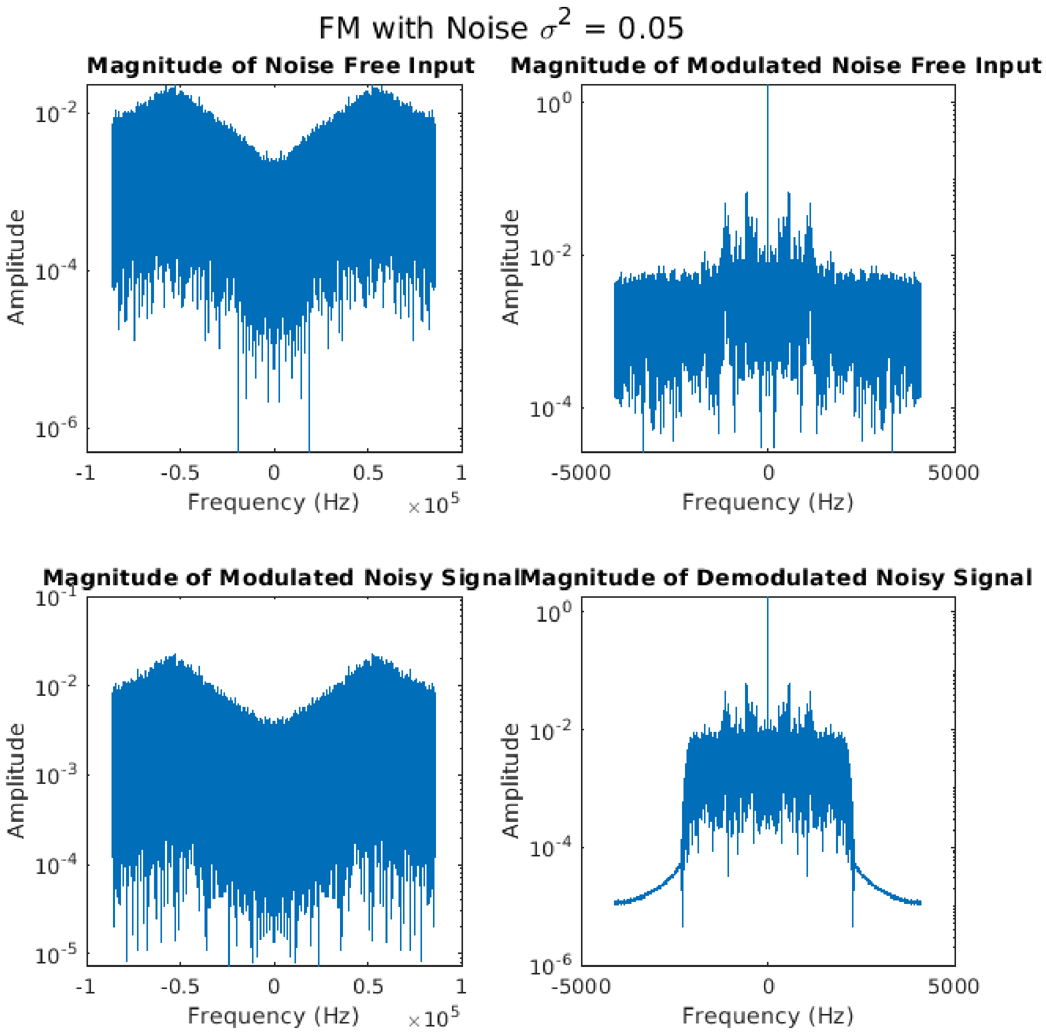
<== Same process to calculate SNR with variance of 0.01 for different k value

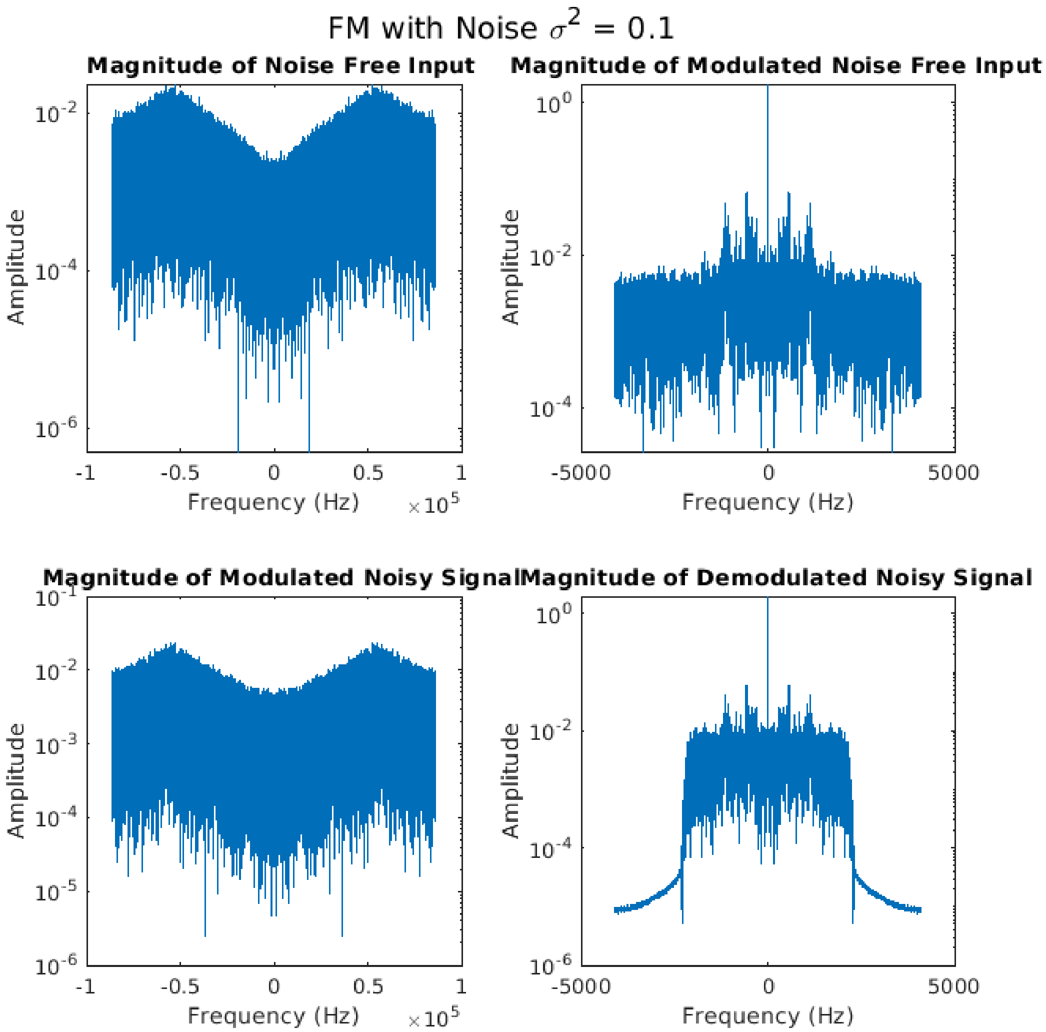
```
    subplot(2, 2, 2);
    semilogy(fdemod, abs(m_FM_demodfreq), 'DisplayName', str);
    title("  Magnitude of Modulated Noise Free Input");
    xlabel("Frequency (Hz)"); ylabel("Amplitude");
    legend;
    hold on

    subplot(2, 2, 3);
    semilogy(f, abs(m_FM_noisefreq), 'DisplayName', str);
    title("  Magnitude of Modulated Noisy Signal");
    xlabel("Frequency (Hz)"); ylabel("Amplitude");
    legend;
    hold on

    subplot(2, 2, 4);
    semilogy(fdemod, abs(m_FM_noise_demodfreq), 'DisplayName', str);
    title("  Magnitude of Demodulated Noisy Signal");
    xlabel("Frequency (Hz)"); ylabel("Amplitude");
    legend;
    hold on;

end
suptitle("FM with Noise \sigma^2 = " + var);
hold off;
```

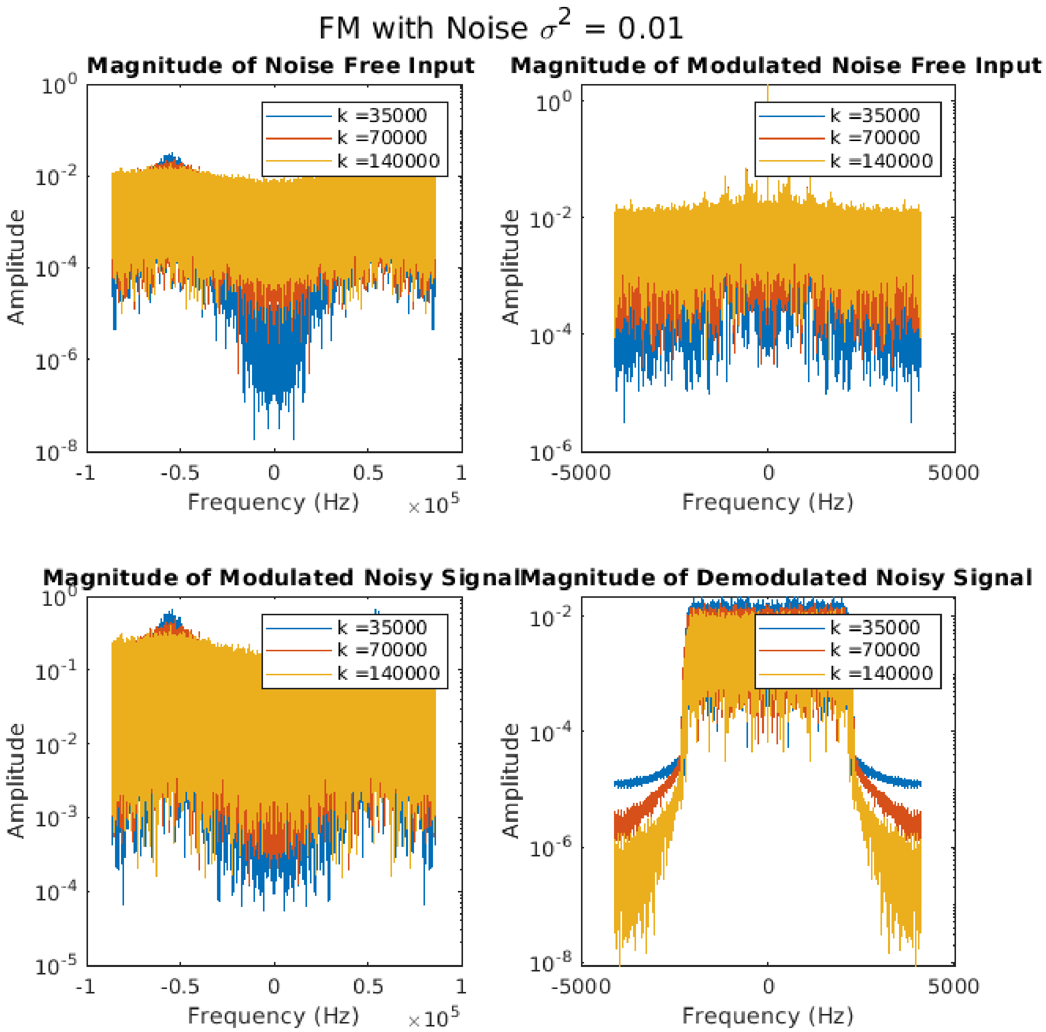

FM Modulation
FM Modulated m(t)
u(t)
Time (s)
Fourier Transform of FM Modulated m(t)
Amplitude
Frequency (Hz)
×10^5


FM Demodulation
FM Demodulated m(t)
u(t)
Time (s)
Fourier Transform of FM Demodulated m(t)
Amplitude
Frequency (Hz)


FM with Noise σ^2 = 0.01
Magnitude of Noise Free Input
Magnitude of Modulated Noise Free Input
Magnitude of Modulated Noisy Signal
Magnitude of Demodulated Noisy Signal
Amplitude
Frequency (Hz)
$\times 10^5$
10^{-2}
10^{-4}
10^{-6}
10^{0}
10^{-1}
10^{-3}
10^{-5}
-1
-0.5
0
0.5
1
-5000
5000


FM with Noise σ^2 = 0.05
Magnitude of Noise Free Input
Magnitude of Modulated Noise Free Input
Magnitude of Modulated Noisy Signal
Magnitude of Demodulated Noisy Signal
Amplitude
Frequency (Hz)
$\times 10^5$


FM with Noise σ^2 = 0.1
Magnitude of Noise Free Input
Magnitude of Modulated Noise Free Input
Magnitude of Modulated Noisy Signal
Magnitude of Demodulated Noisy Signal
Amplitude
Frequency (Hz)
$\times 10^5$



PM Mod & Demod

```
Ac_PM = 0.6019;
k_PM = 2;
m_PM = PM_mod(m, fs, Ac_PM, 55000, k_PM);
m_PMfreq = fftshift(fft(m_PM)/fs_up);

figure('Position', [0 0 1024 1024]);
subplot(2,1,1)
plot(t_up,m_PM);
title("PM Modulated m(t)");xlabel("Time (s)"); ylabel("u(t)");
subplot(2,1,2)
semilogy(f, abs(m_PMfreq));
title("Fourier Transform of PM Modulated m(t)");
xlabel("Frequency (Hz)"); ylabel("Amplitude");
suptitle("PM Modulation");
```

```
m_PM_demod = PM_demod(m_PM, 2, fs_up, 2000, scale, sSin);
m_PM_demodfreq = fftshift(fft(m_PM_demod)/fs);

figure('Position', [0 0 1024 1024]);
subplot(2,1,1)
plot(t,m_PM_demod);
title("PM Demodulated m(t)");xlabel("Time (s)"); ylabel("u(t)");
subplot(2,1,2)
semilogy(fdemod, abs(m_PM_demodfreq));
title("Fourier Transform of PM Demodulated m(t)");
xlabel("Frequency (Hz)"); ylabel("Amplitude");
suptitle("PM Demodulation");

%sound(m_PM_demod, fs);
m_PM_P = rms(m_PM_demod).^2;
W_PM = obw(m_PM_demod, fs);

var = [0.01, 0.05, 0.1];
PM_noiseless = rms(m_PM_demod).^2;

for i = 1:3
```

SNR_PM = Ac^2(kf^2)Pm / (2NoW)

```
    v = var(i);
    noise = sqrt(var(i)) * randn(1,length(m_PM));
    m_PM_noise = m_PM + noise;
    m_PM_noisefreq = fftshift(fft(m_PM_noise)/fs);
    m_PM_noise_demod = PM_demod(m_PM_noise, 1, fs_up, 2000, scale,
 sSin);
    m_PM_noise_demod = lowpass(m_PM_noise_demod, 2000, fs);
    m_PM_noise_demodfreq = fftshift(fft(m_PM_noise_demod)/fs);

    figure('Position', [0 0 1024 1024]);
    subplot(2, 2, 1);
    semilogy(f, abs(m_PMfreq));
    title("  Magnitude of Noise Free Input");
    xlabel("Frequency (Hz)"); ylabel("Amplitude");
    subplot(2, 2, 2);
    semilogy(fdemod, abs(m_PM_demodfreq));
    title("  Magnitude of Modulated Noise Free Input");
    xlabel("Frequency (Hz)"); ylabel("Amplitude");
    subplot(2, 2, 3);
    semilogy(f, abs(m_PM_noisefreq));
    title("  Magnitude of Modulated Noisy Signal");
    xlabel("Frequency (Hz)"); ylabel("Amplitude");
    subplot(2, 2, 4);
    semilogy(fdemod, abs(m_PM_noise_demodfreq));
    title("  Magnitude of Demodulated Noisy Signal");
    xlabel("Frequency (Hz)"); ylabel("Amplitude");
    suptitle("PM with Noise \sigma^2 = " + var(i));

    noisy = rms(m_PM_noise_demod - m_PM_demod).^2;
    snr_PM = 10 * log10(PM_noiseless/noisy);
    display("SNR for PM with variance" + var(i) + ": " + snr_PM);
```

```
    snr_PM_theo = (Ac_PM^2 * k_PM^2 * m_PM_P) / (4 * var(i) * W_PM /
 fs);
    display("Theoretical SNR for PM with variance " + var(i) + ": " +
 snr_PM_theo);
end

figure('Position', [0 0 1024 1024]);
for j = 1:3
    var = 0.01;
    Ac_PM = 10827/10000;
    k_PM = 2^(j-1);
    m_PM = PM_mod(m, fs, Ac_PM, 55000, k_PM);
    m_PMfreq = fftshift(fft(m_PM)/fs_up);
    m_PM_demod = PM_demod(m_PM, 2, fs_up, 2000, scale, sSin);
    m_PM_demodfreq = fftshift(fft(m_PM_demod)/fs);
    PM_noiseless = rms(m_PM_demod).^2;
    noise = sqrt(var) * randn(1,length(m_PM));
    m_PM_noise = m_PM + noise;
    m_PM_noisefreq = fftshift(fft(m_PM_noise)/fs);
    m_PM_noise_demod = PM_demod(m_PM_noise, 1, fs_up, 2000, scale,
 sSin);
    m_PM_noise_demod = lowpass(m_PM_noise_demod, 2000, fs);
    m_PM_noise_demodfreq = fftshift(fft(m_PM_noise_demod)/fs);

    noisy = rms(m_PM_noise_demod - m_PM_demod).^2;
    snr_PM = 10 * log10(PM_noiseless/noisy);
    display("SNR for PM with k = " + k_PM + "(variance = "+var+") is:
 " + snr_PM);

    snr_PM_theo = (Ac_PM^2 * k_PM^2 * m_PM_P) / (4 * var * W_PM / fs);
    display("Theoretical SNR for PM with k = " + k_PM + "(variance =
 "+var+") is: " + snr_PM_theo);

    subplot(2, 2, 1);
    str = strcat('k = ' , int2str(k_PM));
    semilogy(f, abs(m_PMfreq), 'DisplayName', str);
    title("  Magnitude of Noise Free Input");
    xlabel("Frequency (Hz)"); ylabel("Amplitude");
    legend;
    hold on

    subplot(2, 2, 2);
    semilogy(fdemod, abs(m_PM_demodfreq), 'DisplayName', str);
    title("  Magnitude of Modulated Noise Free Input");
    xlabel("Frequency (Hz)"); ylabel("Amplitude");
    legend;
    hold on

    subplot(2, 2, 3);
    semilogy(f, abs(m_PM_noisefreq), 'DisplayName', str);
    title("  Magnitude of Modulated Noisy Signal");
    xlabel("Frequency (Hz)"); ylabel("Amplitude");
    legend;
    hold on
```

```
    subplot(2, 2, 4);
    semilogy(fdemod, abs(m_PM_noise_demodfreq), 'DisplayName', str);
    title("  Magnitude of Demodulated Noisy Signal");
    xlabel("Frequency (Hz)"); ylabel("Amplitude");
    legend;
    hold on;

end
suptitle("PM with Noise \sigma^2 = " + var);
hold off;
```

"SNR for PM with variance0.01: 5.7486"

"Theoretical SNR for PM with variance 0.01: 6.1769"

"SNR for PM with variance0.05: 5.5341"

"Theoretical SNR for PM with variance 0.05: 1.2354"

"SNR for PM with variance0.1: 5.2732"

"Theoretical SNR for PM with variance 0.1: 0.61769"

"SNR for PM with k = 1(variance = 0.01) is: 5.7787"

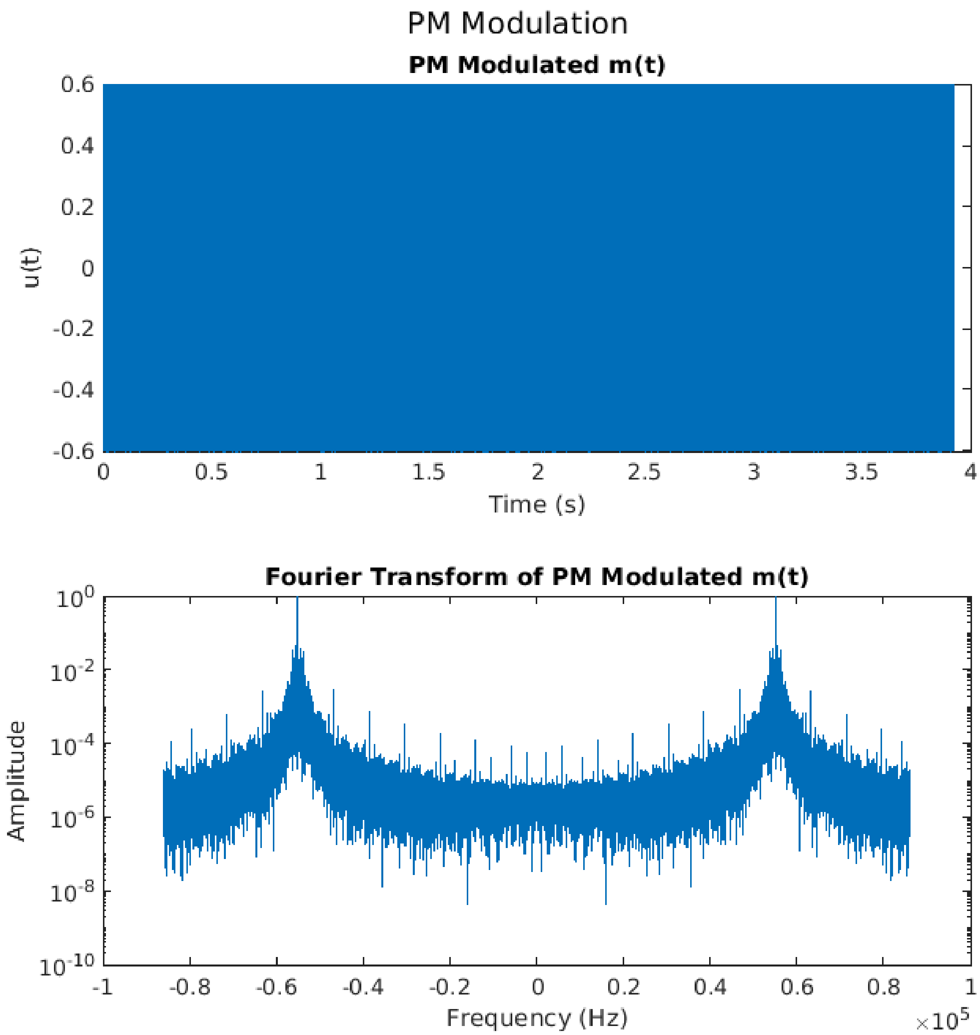
"Theoretical SNR for PM with k = 1(variance = 0.01) is: 4.9966"

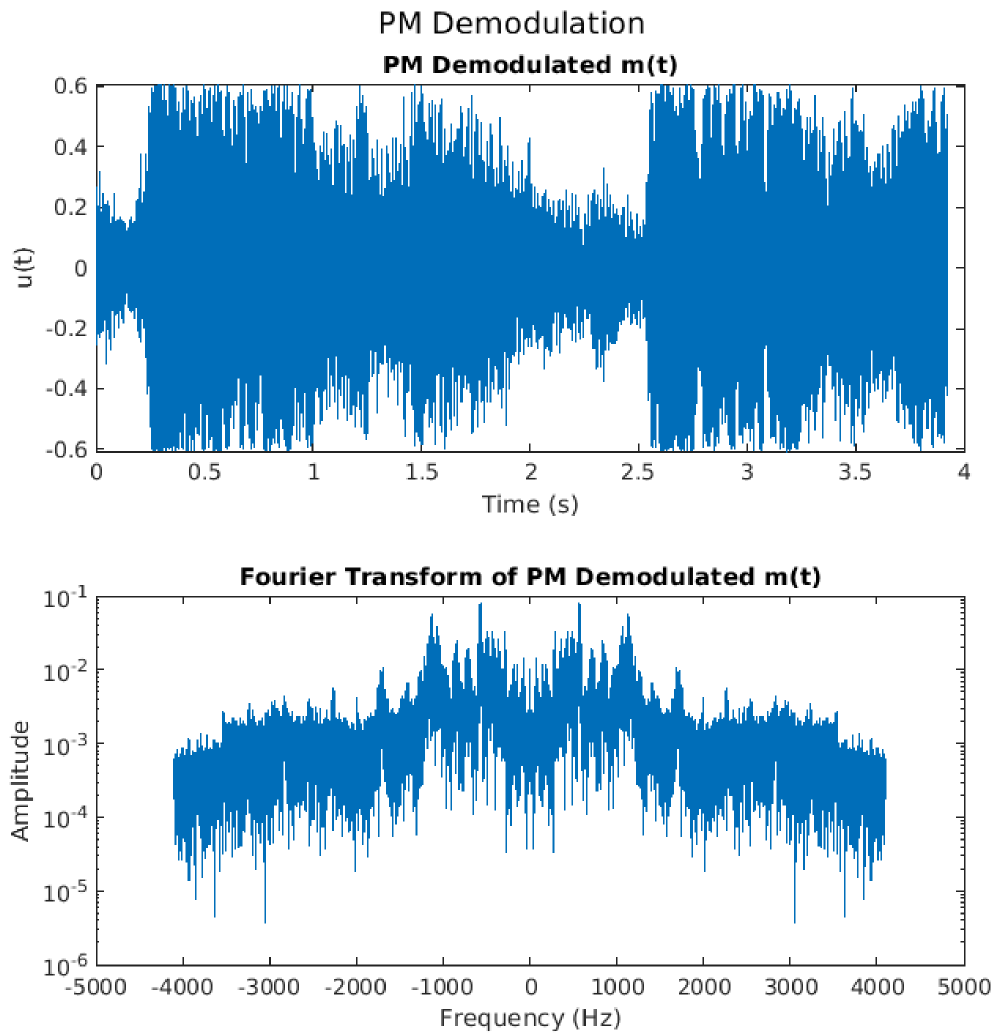
"SNR for PM with k = 2(variance = 0.01) is: 5.7818"

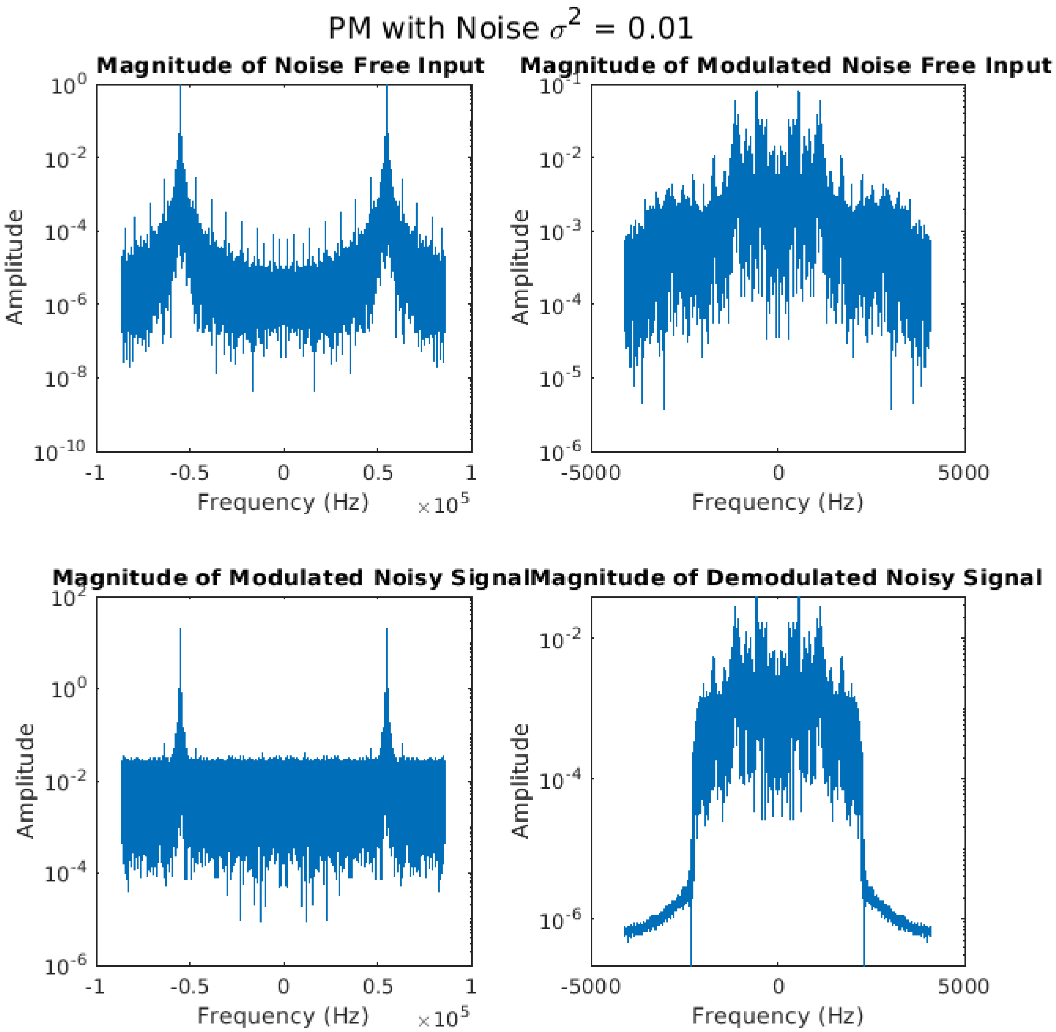
"Theoretical SNR for PM with k = 2(variance = 0.01) is: 19.9864"

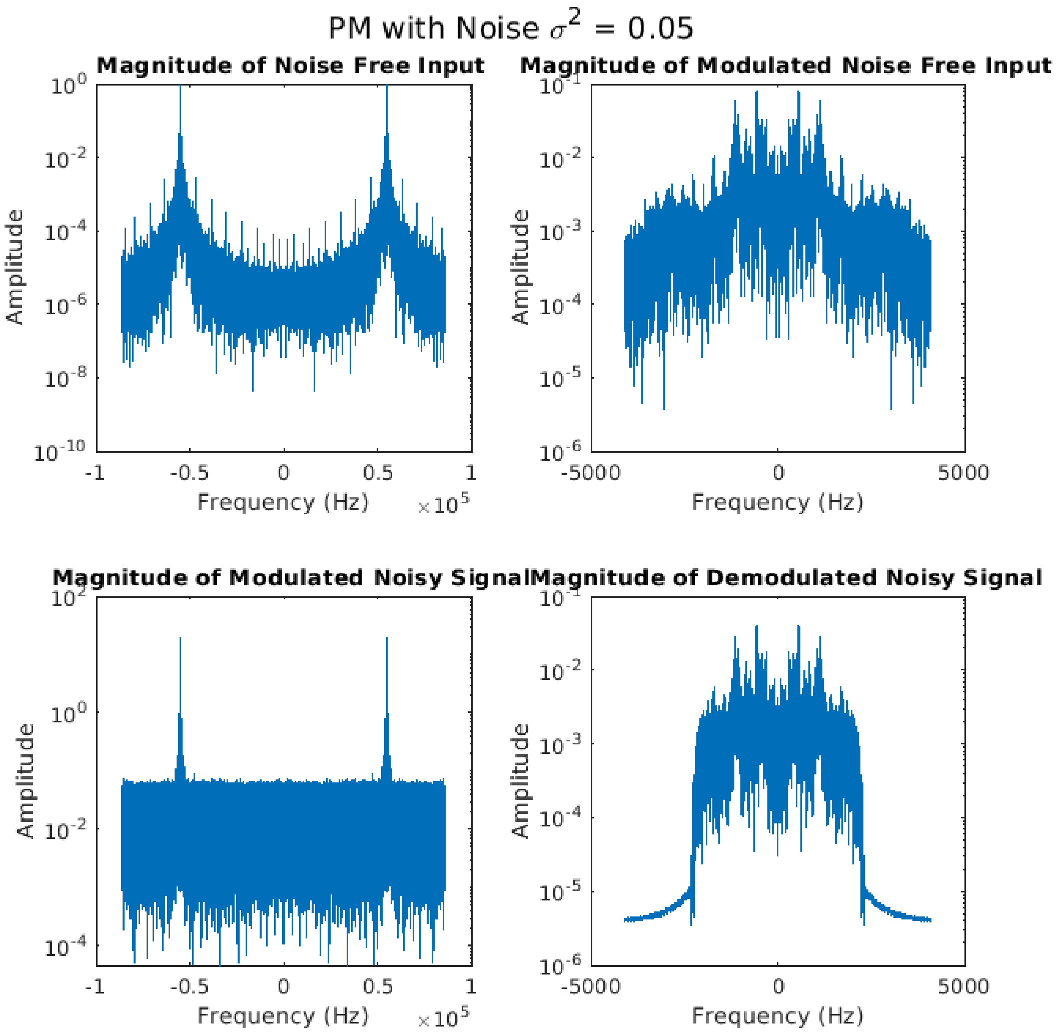
"SNR for PM with k = 4(variance = 0.01) is: 5.3667"

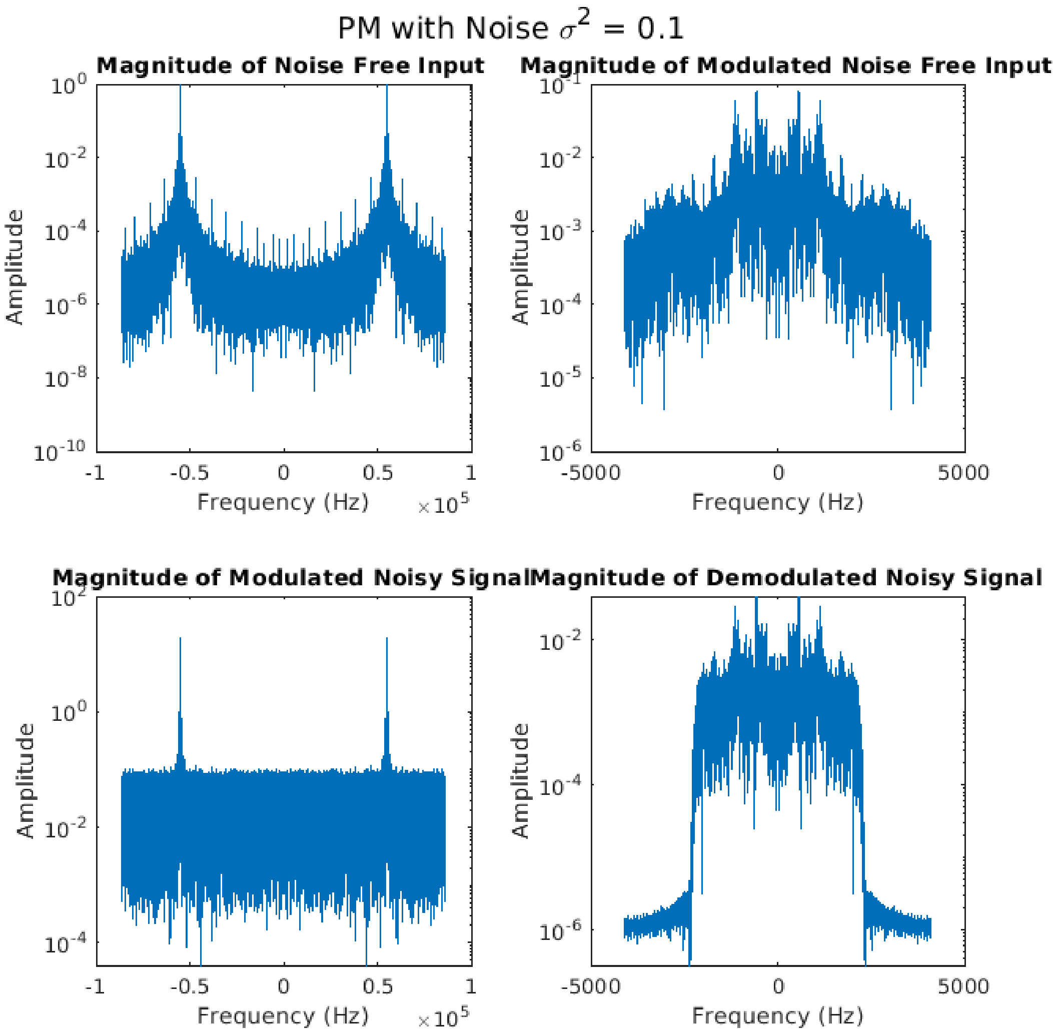
"Theoretical SNR for PM with k = 4(variance = 0.01) is: 79.9458"


PM Modulation
PM Modulated m(t)
u(t)
Time (s)
Fourier Transform of PM Modulated m(t)
Amplitude
Frequency (Hz)

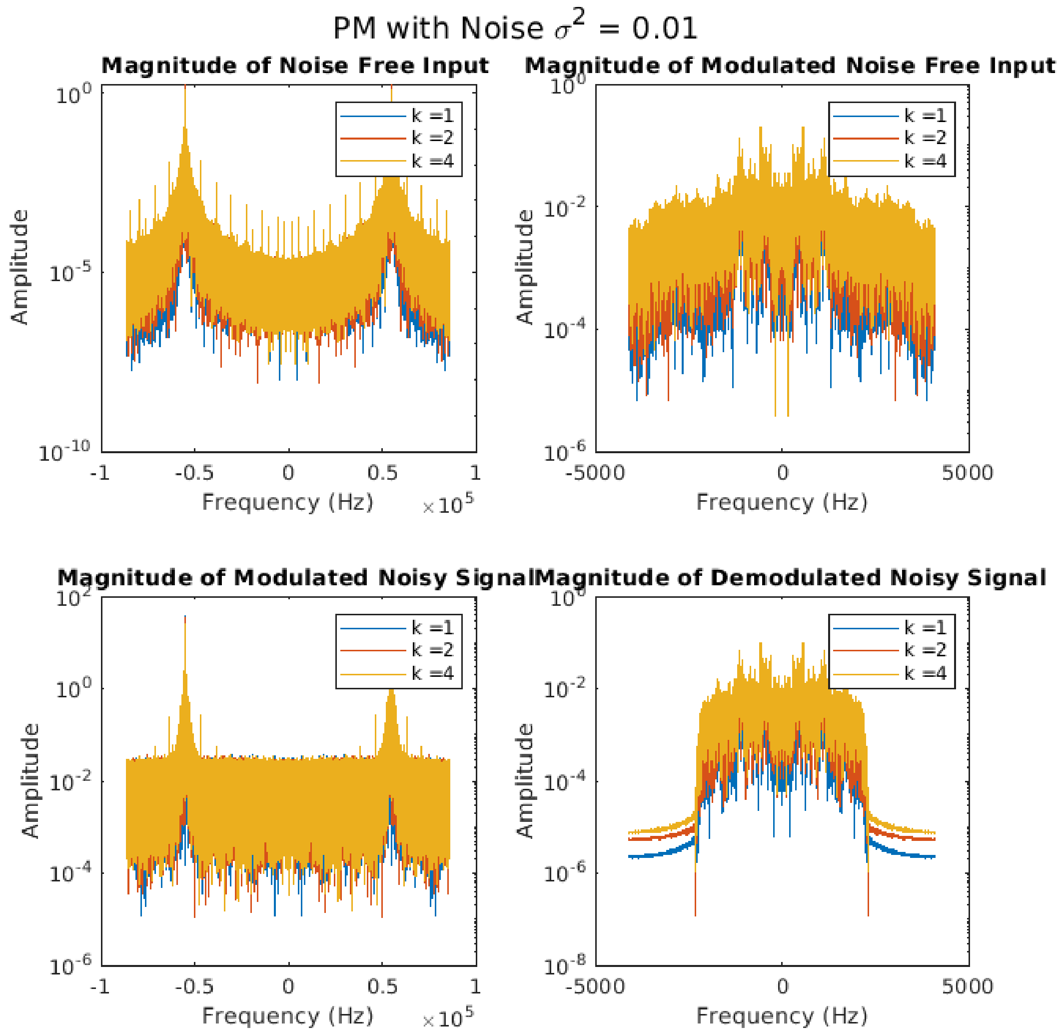

PM Demodulation
PM Demodulated m(t)
u(t)
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
0
0.5
1
1.5
2
2.5
3
3.5
4
Time (s)
Fourier Transform of PM Demodulated m(t)
Amplitude
10^{-1}
10^{-2}
10^{-3}
10^{-4}
10^{-5}
10^{-6}
-5000
-4000
-3000
-2000
-1000
0
1000
2000
3000
4000
5000
Frequency (Hz)


PM with Noise σ2 = 0.01
Magnitude of Noise Free Input
Magnitude of Modulated Noise Free Input
Magnitude of Modulated Noisy Signal
Magnitude of Demodulated Noisy Signal
Amplitude
Frequency (Hz)
×10^5


PM with Noise σ^2 = 0.05
Magnitude of Noise Free Input
Magnitude of Modulated Noise Free Input
Magnitude of Modulated Noisy Signal
Magnitude of Demodulated Noisy Signal
Amplitude
Frequency (Hz)
$\times 10^5$



PM with Noise σ^2 = 0.1
Magnitude of Noise Free Input
Magnitude of Modulated Noise Free Input
Magnitude of Modulated Noisy Signal
Magnitude of Demodulated Noisy Signal
Amplitude
Frequency (Hz)
$\times 10^5$



AM Mod & Demod

```
Ac_AM = 87/250;          <== carefully chosen values to make sure
a = 2;                       the power of the demodulated signal
                             matches the input power
m_AM = AM_mod(m, fs, Ac_AM, 55000,a);
m_AMfreq = fftshift(fft(m_AM)/fs_up);

figure('Position', [0 0 1024 1024]);
subplot(2,1,1)
plot(t_up,m_AM);
title("AM Modulated m(t)");xlabel("Time (s)"); ylabel("u(t)");
subplot(2,1,2)
semilogy(f, abs(m_AMfreq));
title("Fourier Transform of AM Modulated m(t)");
xlabel("Frequency (Hz)"); ylabel("Amplitude");
suptitle("AM Modulation");
```

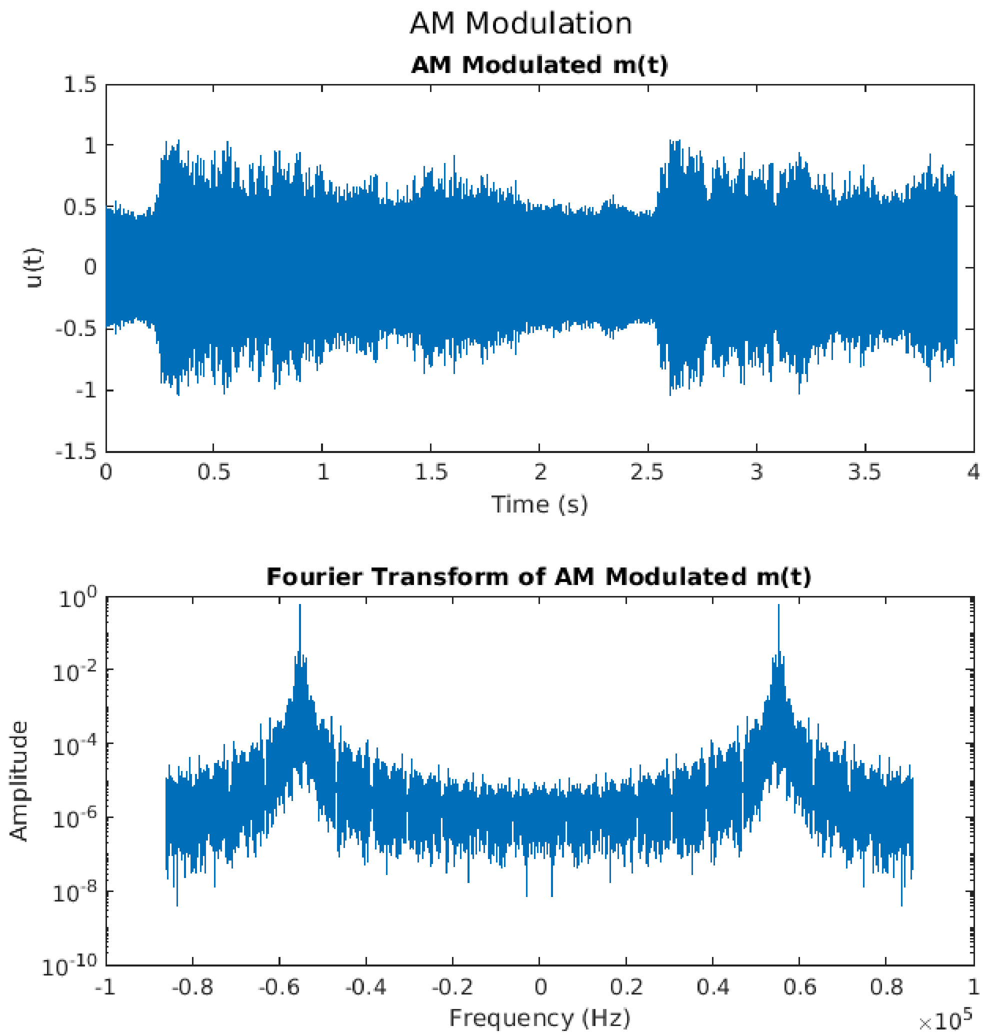
```
m_AM_demod = AM_demod(m_AM, fs_up, 2000, scale);
m_AM_demodfreq = fftshift(fft(m_AM_demod)/fs);

figure('Position', [0 0 1024 1024]);
subplot(2,1,1)
plot(t,m_AM_demod);
title("AM Demodulated m(t)");xlabel("Time (s)"); ylabel("u(t)");
subplot(2,1,2)
semilogy(fdemod, abs(m_AM_demodfreq));
title("Fourier Transform of AM Demodulated m(t)");
xlabel("Frequency (Hz)"); ylabel("Amplitude");
suptitle("AM Demodulation");

%sound(m_AM_demod, fs);
m_AM_P = rms(m_AM_demod).^2;
W_AM = obw(m_AM_demod, fs);

var = [0.01, 0.05, 0.1];
AM_noiseless = rms(m_AM_demod).^2;

for i = 1:3

    v = var(i);
    noise = sqrt(var(i)) * randn(1,length(m_AM));
    m_AM_noise = m_AM + noise;
    m_AM_noisefreq = fftshift(fft(m_AM_noise)/fs);
    m_AM_noise_demod = AM_demod(m_AM_noise, fs_up, 2000, scale);
    m_AM_noise_demod = lowpass(m_AM_noise_demod, 2000, fs);
    m_AM_noise_demodfreq = fftshift(fft(m_AM_noise_demod)/fs);
    figure('Position', [0 0 1024 1024]);
    subplot(2, 2, 1);
    semilogy(f, abs(m_AMfreq));
    title("  Magnitude of Noise Free Input");
    xlabel("Frequency (Hz)"); ylabel("Amplitude");
    subplot(2, 2, 2);
    semilogy(fdemod, abs(m_AM_demodfreq));
    title("  Magnitude of Modulated Noise Free Input");
    xlabel("Frequency (Hz)"); ylabel("Amplitude");
    subplot(2, 2, 3);
    semilogy(f, abs(m_AM_noisefreq));
    title("  Magnitude of Modulated Noisy Signal");
    xlabel("Frequency (Hz)"); ylabel("Amplitude");
    subplot(2, 2, 4);
    semilogy(fdemod, abs(m_AM_noise_demodfreq));
    title("  Magnitude of Demodulated Noisy Signal");
    xlabel("Frequency (Hz)"); ylabel("Amplitude");
    suptitle("AM with Noise \sigma^2 = " + var(i));

    noisy = rms(m_AM_noise_demod - m_AM_demod).^2;
    snr_AM = 10 * log10(AM_noiseless/noisy);
    display("SNR for AM with variance" + var(i) + ": " + snr_AM);

    snr_AM_theo = (Ac_AM^2 * a^2 * m_AM_P) / (4 * var(i) * W_AM / fs);
```

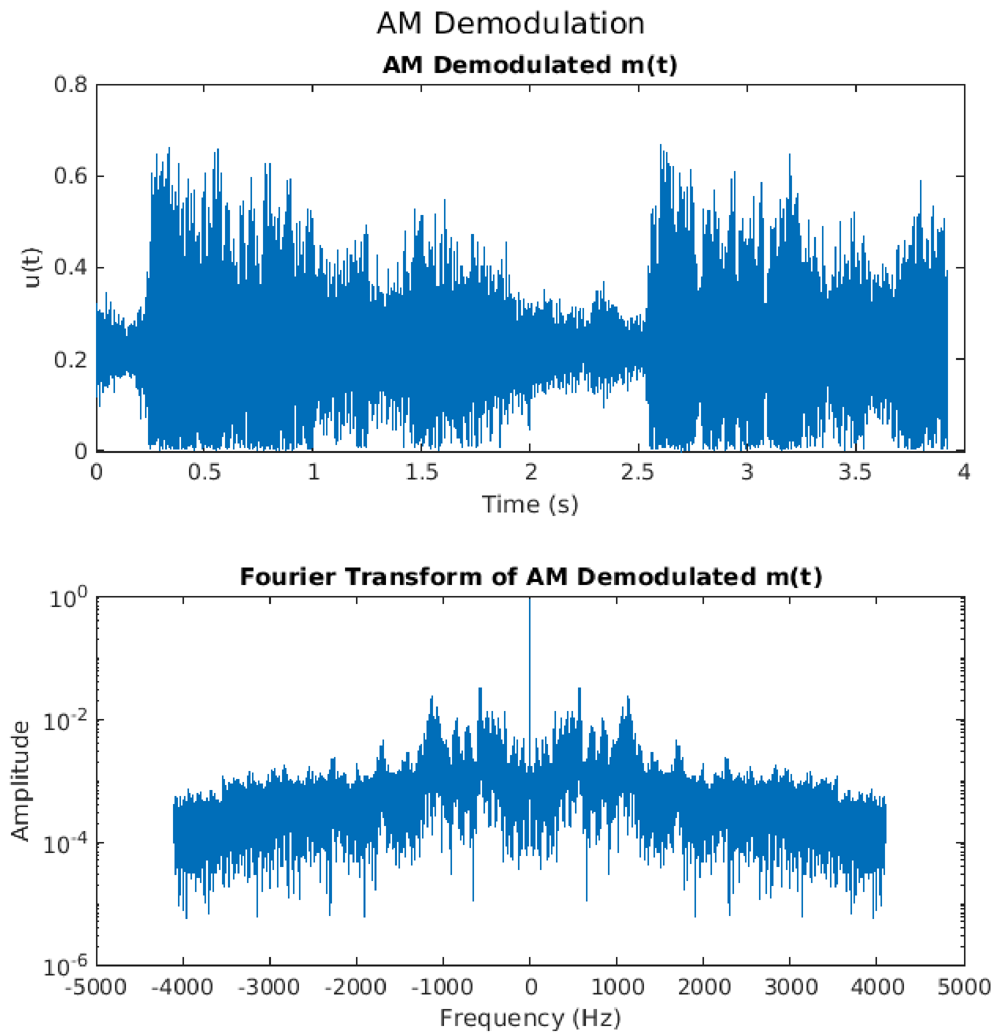
SNR_AM = Ac^2(a^2)Pm / (2NoW)

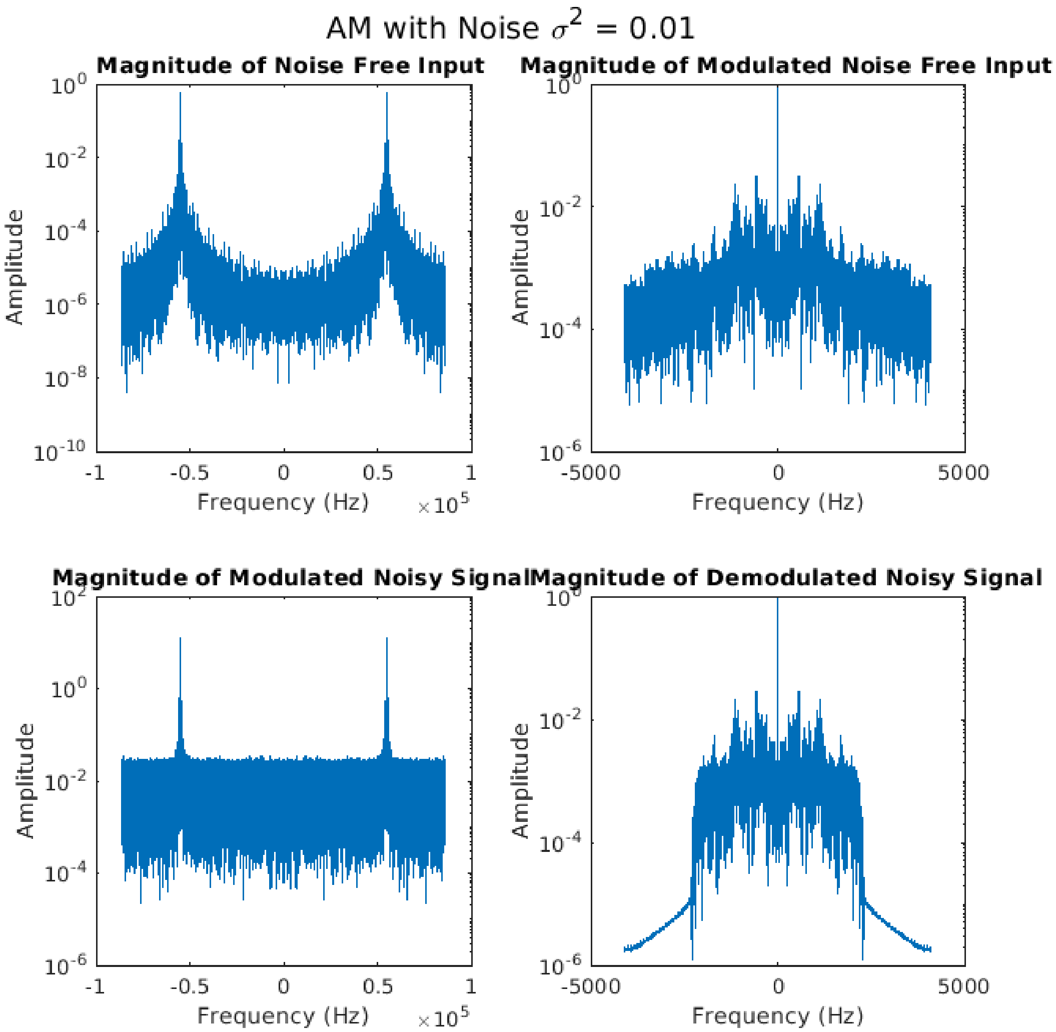
```
    display("Theoretical SNR for AM with variance " + var(i) + ": " +
 snr_AM_theo);
end
%sound(m_AM_noise_demod, fs);
figure('Position', [0 0 1024 1024]);
for j = 1:3
    var = 0.01;
    Ac_AM = 87/250;
    a = 2^(j-1);

    m_AM = AM_mod(m, fs, Ac_AM, 55000,a);
    m_AMfreq = fftshift(fft(m_AM)/fs_up);
    m_AM_demod = AM_demod(m_AM, fs_up, 2000, scale);
    m_AM_demodfreq = fftshift(fft(m_AM_demod)/fs);
    AM_noiseless = rms(m_AM_demod).^2;
    noise = sqrt(var) * randn(1,length(m_AM));
    m_AM_noise = m_AM + noise;
    m_AM_noisefreq = fftshift(fft(m_AM_noise)/fs);
    m_AM_noise_demod = AM_demod(m_AM_noise, fs_up, 2000, scale);
    m_AM_noise_demod = lowpass(m_AM_noise_demod, 2000, fs);
    m_AM_noise_demodfreq = fftshift(fft(m_AM_noise_demod)/fs);

    noisy = rms(m_AM_noise_demod - m_AM_demod).^2;
    snr_AM = 10 * log10(AM_noiseless/noisy);
    display("SNR for AM with a = " + a + "(variance = "+var+") is: " +
 snr_AM);

    snr_AM_theo = (Ac_AM^2 * a^2 * m_AM_P) / (4 * var * W_AM / fs);
    display("Theoretical SNR for AM with a = " + a + "(variance =
 "+var+") is: " + snr_AM_theo);

    subplot(2, 2, 1);
    str = strcat('a = ' , int2str(a));
    semilogy(f, abs(m_AMfreq), 'DisplayName', str);
    title("  Magnitude of Noise Free Input");
    xlabel("Frequency (Hz)"); ylabel("Amplitude");
    legend;
    hold on

    subplot(2, 2, 2);
    semilogy(fdemod, abs(m_AM_demodfreq), 'DisplayName', str);
    title("  Magnitude of Modulated Noise Free Input");
    xlabel("Frequency (Hz)"); ylabel("Amplitude");
    legend;
    hold on

    subplot(2, 2, 3);
    semilogy(f, abs(m_AM_noisefreq), 'DisplayName', str);
    title("  Magnitude of Modulated Noisy Signal");
    xlabel("Frequency (Hz)"); ylabel("Amplitude");
    legend;
    hold on

    subplot(2, 2, 4);
```

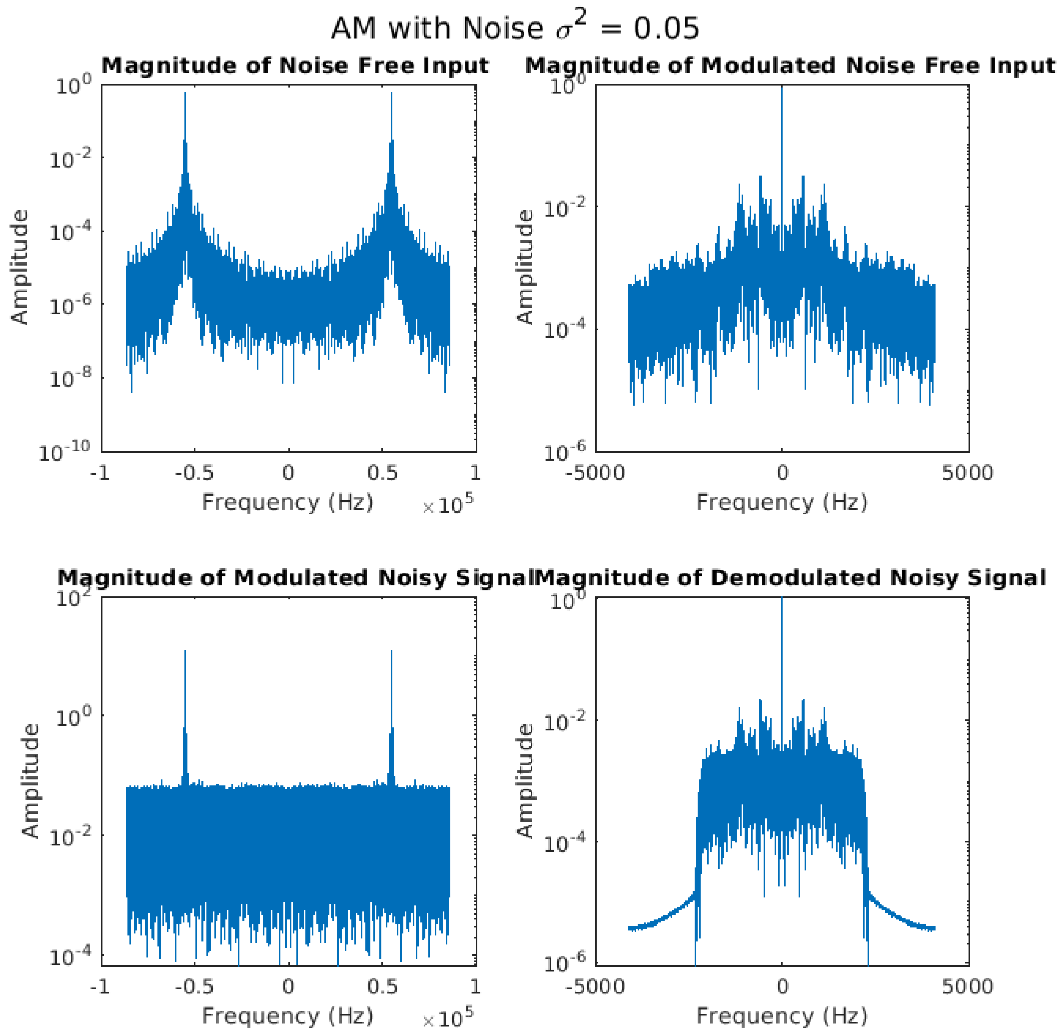
```
    semilogy(fdemod, abs(m_AM_noise_demodfreq), 'DisplayName', str);
    title("  Magnitude of Demodulated Noisy Signal");
    xlabel("Frequency (Hz)"); ylabel("Amplitude");
    legend;
    hold on;

end
suptitle("AM with Noise \sigma^2 = " + var);
hold off;
```

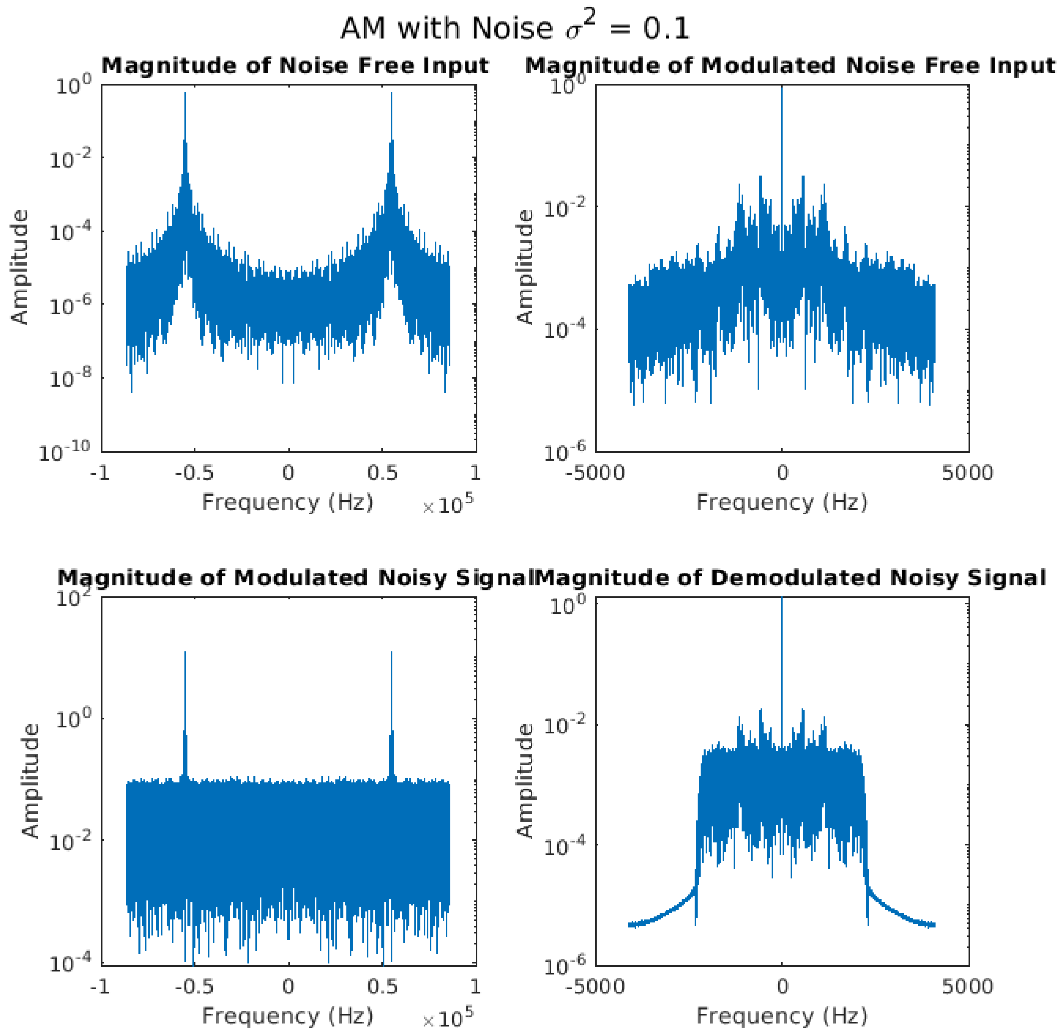
```
"SNR for AM with variance0.01: 18.7191"

"Theoretical SNR for AM with variance 0.01: 3.4931"

"SNR for AM with variance0.05: 10.1124"

"Theoretical SNR for AM with variance 0.05: 0.69863"

"SNR for AM with variance0.1: 5.742"

"Theoretical SNR for AM with variance 0.1: 0.34931"

"SNR for AM with a = 1(variance = 0.01) is: 20.1247"

"Theoretical SNR for AM with a = 1(variance = 0.01) is: 0.87328"

"SNR for AM with a = 2(variance = 0.01) is: 18.7091"

"Theoretical SNR for AM with a = 2(variance = 0.01) is: 3.4931"

"SNR for AM with a = 4(variance = 0.01) is: 16.4272"

"Theoretical SNR for AM with a = 4(variance = 0.01) is: 13.9726"
```

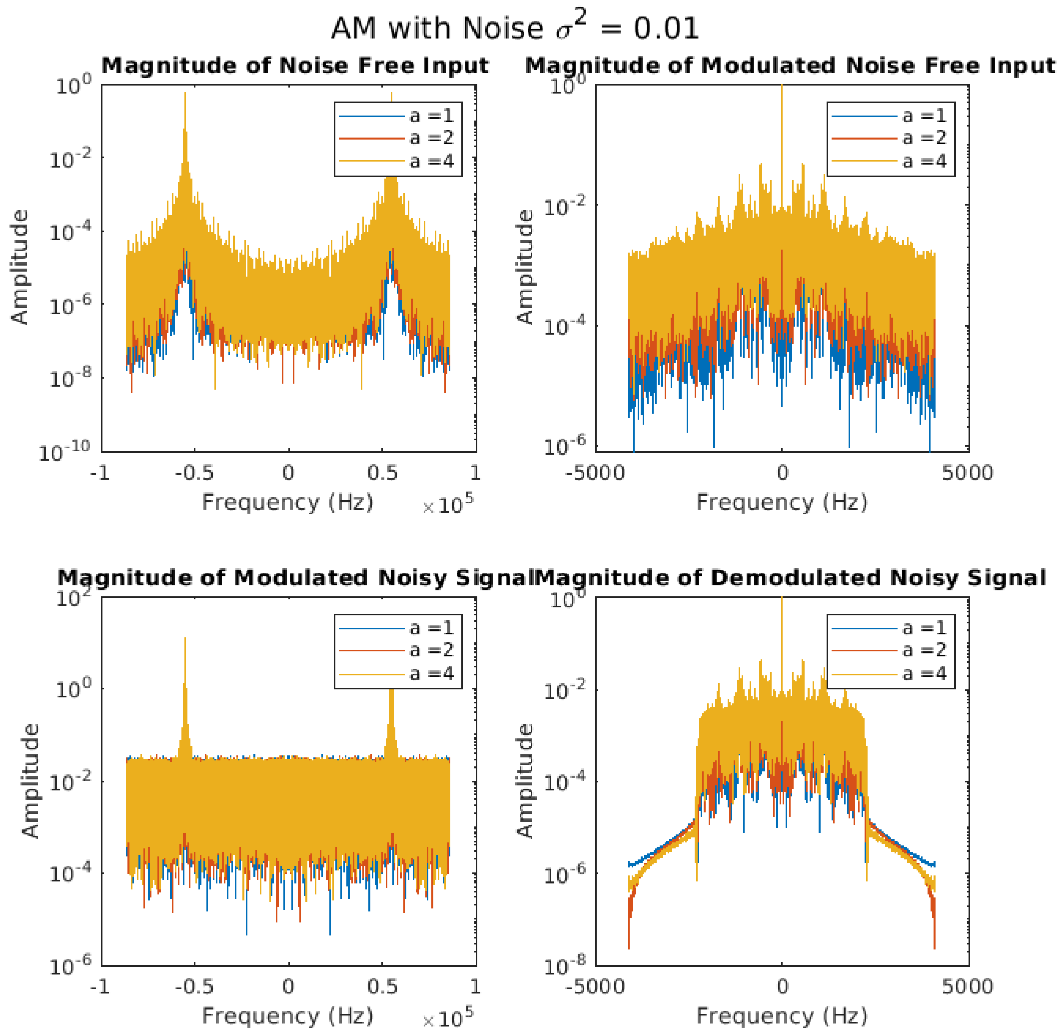

AM Modulation
AM Modulated m(t)
1.5
1
0.5
0
-0.5
-1
-1.5
u(t)
0
0.5
1
1.5
2
2.5
3
3.5
4
Time (s)
Fourier Transform of AM Modulated m(t)
10^{0}
10^{-2}
10^{-4}
10^{-6}
10^{-8}
10^{-10}
Amplitude
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
0.8
1
Frequency (Hz)
$\times 10^{5}$


AM Demodulation
AM Demodulated m(t)
0.8
0.6
0.4
0.2
0
u(t)
0
0.5
1
1.5
2
2.5
3
3.5
4
Time (s)
Fourier Transform of AM Demodulated m(t)
10^0
10^{-2}
10^{-4}
10^{-6}
Amplitude
-5000
-4000
-3000
-2000
-1000
0
1000
2000
3000
4000
5000
Frequency (Hz)


AM with Noise σ^2 = 0.01
Magnitude of Noise Free Input
Magnitude of Modulated Noise Free Input
Magnitude of Modulated Noisy Signal
Magnitude of Demodulated Noisy Signal
Amplitude
Frequency (Hz)
$\times 10^5$


AM with Noise σ^2 = 0.05
Magnitude of Noise Free Input
Magnitude of Modulated Noise Free Input
Magnitude of Modulated Noisy Signal
Magnitude of Demodulated Noisy Signal
Amplitude
Frequency (Hz)
$\times 10^5$


AM with Noise σ^2 = 0.1
Magnitude of Noise Free Input
Magnitude of Modulated Noise Free Input
Magnitude of Modulated Noisy Signal
Magnitude of Demodulated Noisy Signal
Amplitude
Frequency (Hz)
$\times 10^5$



SSB Mod & Demod

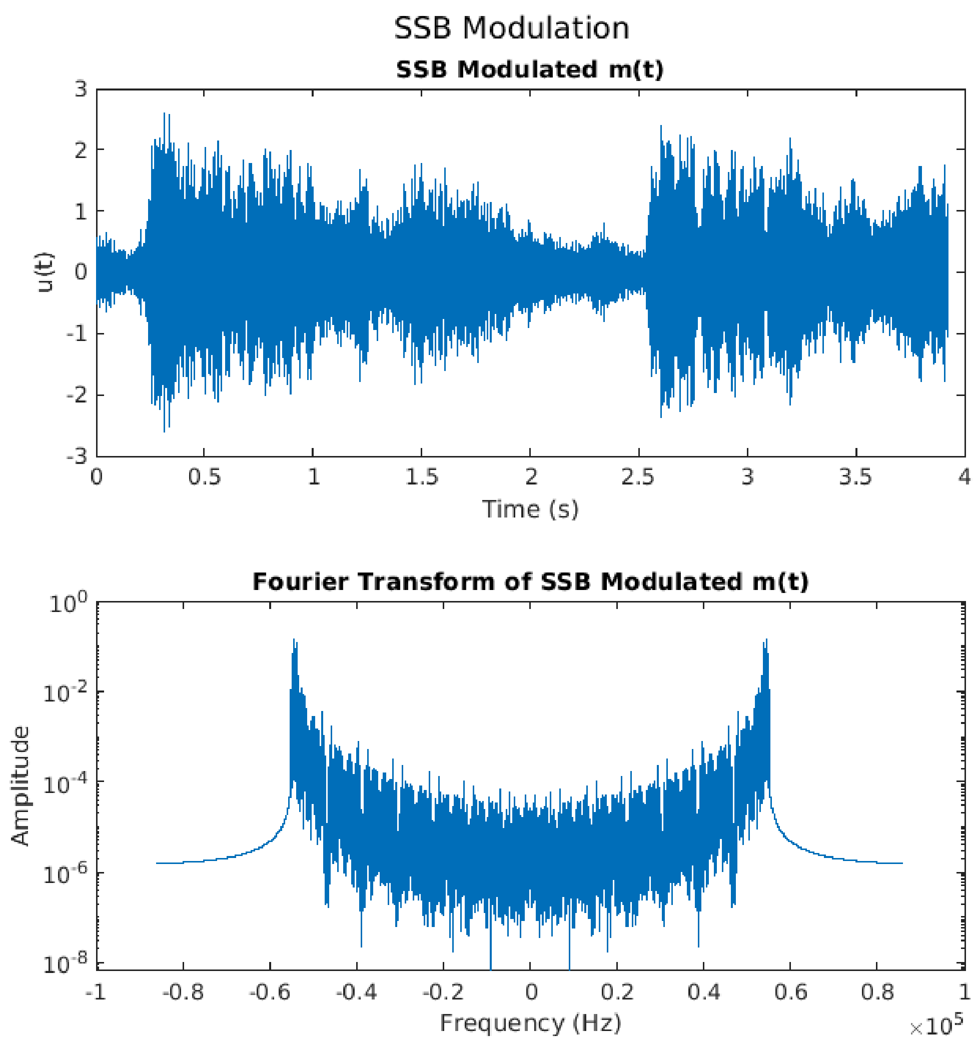
```
Ac_SSB = 262/125;

m_SSB = SSB_mod(m, fs, Ac_SSB, 55000);
m_SSBfreq = fftshift(fft(m_SSB)/fs_up);

figure('Position', [0 0 1024 1024]);
subplot(2,1,1)
plot(t_up,m_SSB);
title("SSB Modulated m(t)");xlabel("Time (s)"); ylabel("u(t)");
subplot(2,1,2)
semilogy(f, abs(m_SSBfreq));
title("Fourier Transform of SSB Modulated m(t)");
xlabel("Frequency (Hz)"); ylabel("Amplitude");
suptitle("SSB Modulation");
```

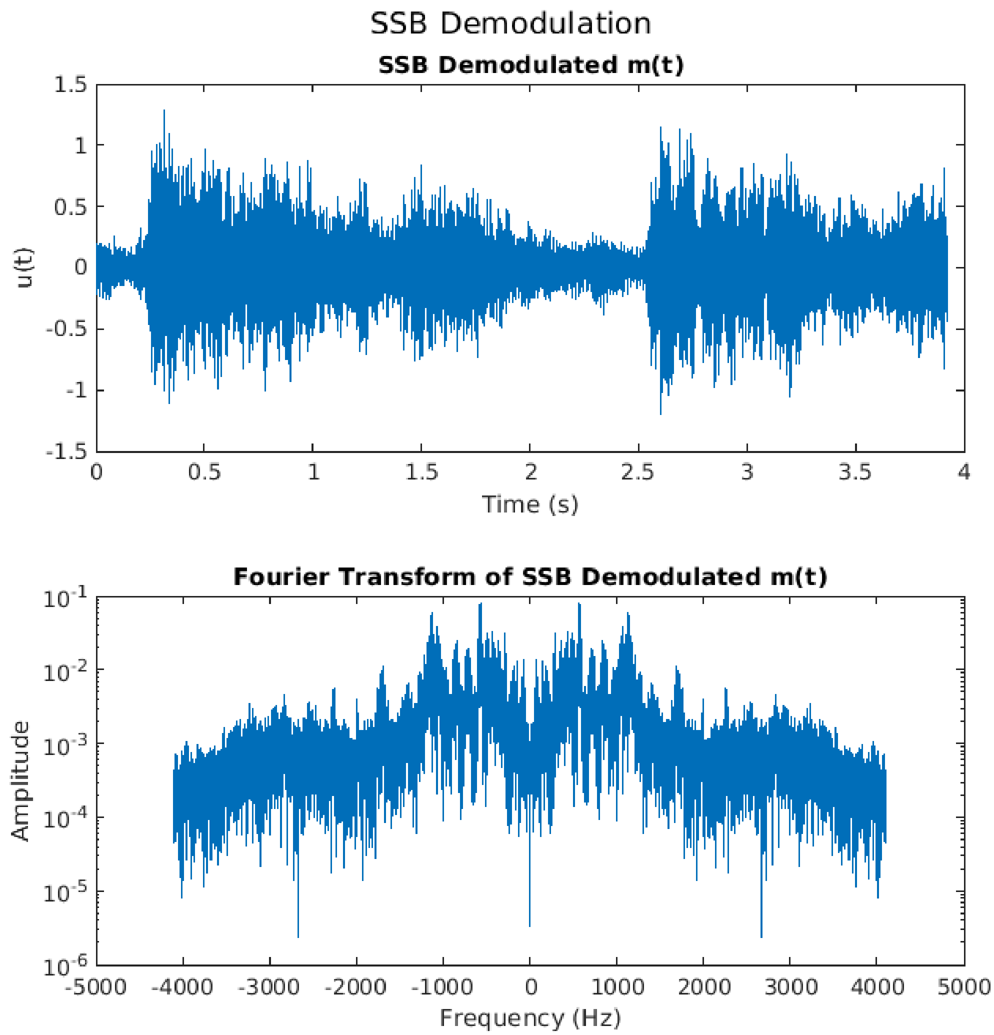
```
m_SSB_demod = SSB_demod(m_SSB, fs_up, 2000, scale, sSin);
m_SSB_demodfreq = fftshift(fft(m_SSB_demod)/fs);

figure('Position', [0 0 1024 1024]);
subplot(2,1,1)
plot(t,m_SSB_demod);
title("SSB Demodulated m(t)");xlabel("Time (s)"); ylabel("u(t)");
subplot(2,1,2)
semilogy(fdemod, abs(m_SSB_demodfreq));
title("Fourier Transform of SSB Demodulated m(t)");
xlabel("Frequency (Hz)"); ylabel("Amplitude");
suptitle("SSB Demodulation");

%sound(m_SSB_demod, fs);

m_SSB_P = rms(m_SSB_demod).^2;
W_SSB = obw(m_SSB_demod, fs);

var = [0.01, 0.05, 0.1];
SSB_noiseless = rms(m_SSB_demod).^2;

for i = 1:3

    v = var(i);
    noise = sqrt(var(i)) * randn(1,length(m_SSB));
    m_SSB_noise = m_SSB + noise;
    m_SSB_noisefreq = fftshift(fft(m_SSB_noise)/fs);
    m_SSB_noise_demod = SSB_demod(m_SSB_noise, fs_up, 2000, scale,
 sSin);
    m_SSB_noise_demod = lowpass(m_SSB_noise_demod, 2000, fs);
    m_SSB_noise_demodfreq = fftshift(fft(m_SSB_noise_demod)/fs);

    figure('Position', [0 0 1024 1024]);
    subplot(2, 2, 1);
    semilogy(f, abs(m_SSBfreq));
    title("  Magnitude of Noise Free Input");
    xlabel("Frequency (Hz)"); ylabel("Amplitude");
    subplot(2, 2, 2);
    semilogy(fdemod, abs(m_SSB_demodfreq));
    title("  Magnitude of Modulated Noise Free Input");
    xlabel("Frequency (Hz)"); ylabel("Amplitude");
    subplot(2, 2, 3);
    semilogy(f, abs(m_SSB_noisefreq));
    title("  Magnitude of Modulated Noisy Signal");
    xlabel("Frequency (Hz)"); ylabel("Amplitude");
    subplot(2, 2, 4);
    semilogy(fdemod, abs(m_SSB_noise_demodfreq));
    title("  Magnitude of Demodulated Noisy Signal");
    xlabel("Frequency (Hz)"); ylabel("Amplitude");
    suptitle("SSB with Noise \sigma^2 = " + var(i));

    noisy = rms(m_SSB_noise_demod - m_SSB_demod).^2;
    snr_SSB = 10 * log10(SSB_noiseless/noisy);
    display("SNR for SSB with variance" + var(i) + ": " + snr_SSB);
```

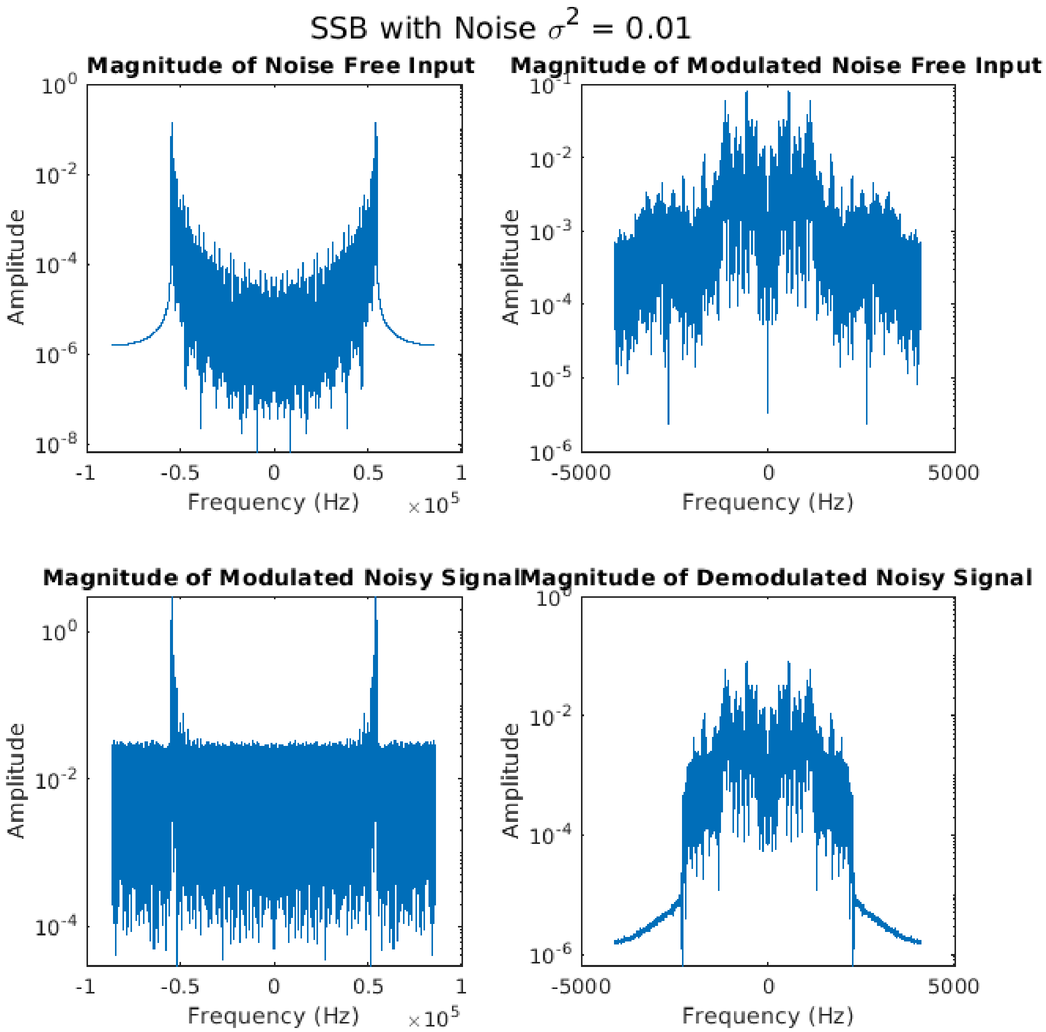
SNR_SSB = Ac^2Pm / (NoW)

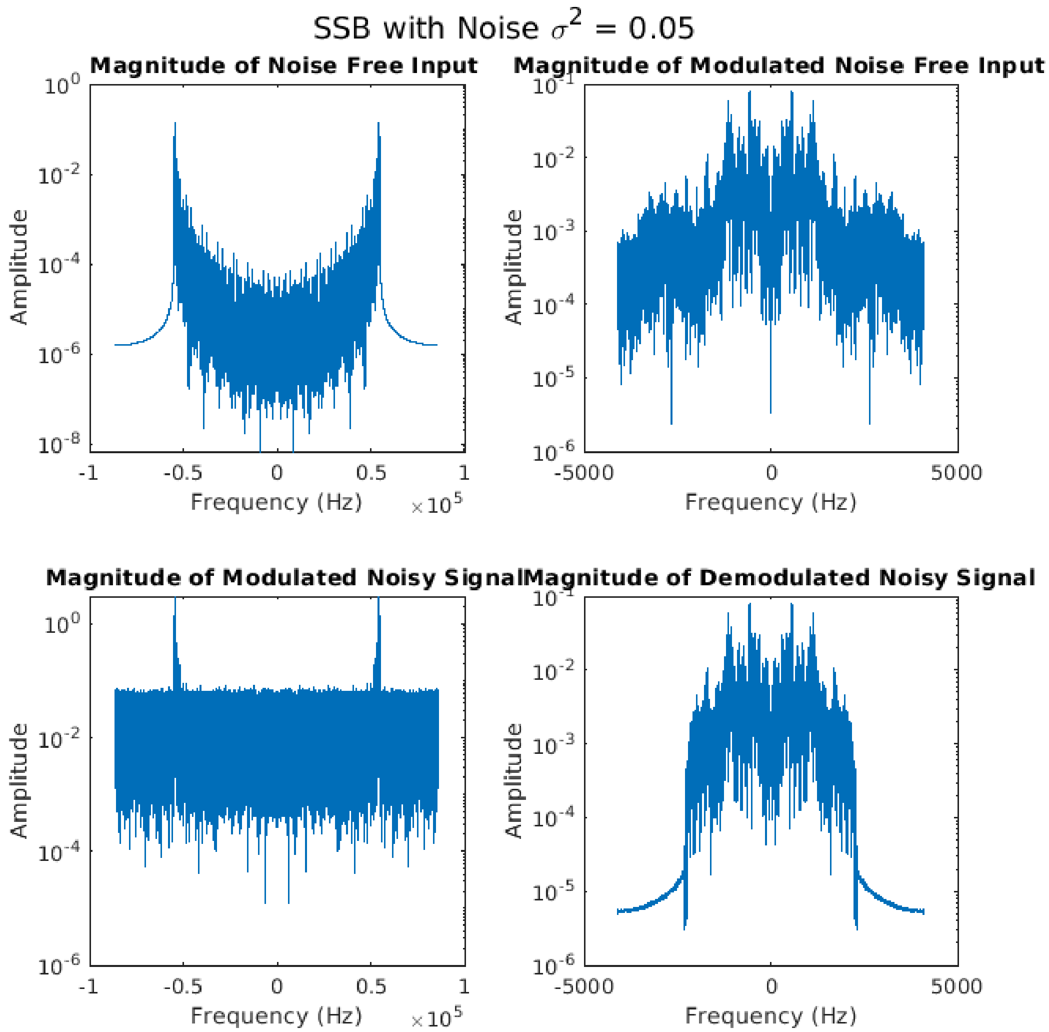
```
    snr_SSB_theo = (Ac_SSB^2 * m_SSB_P) / (2 * var(i) * W_SSB / fs);
    display("Theoretical SNR for SSB with variance " + var(i) + ": " +
 snr_SSB_theo);
end
```

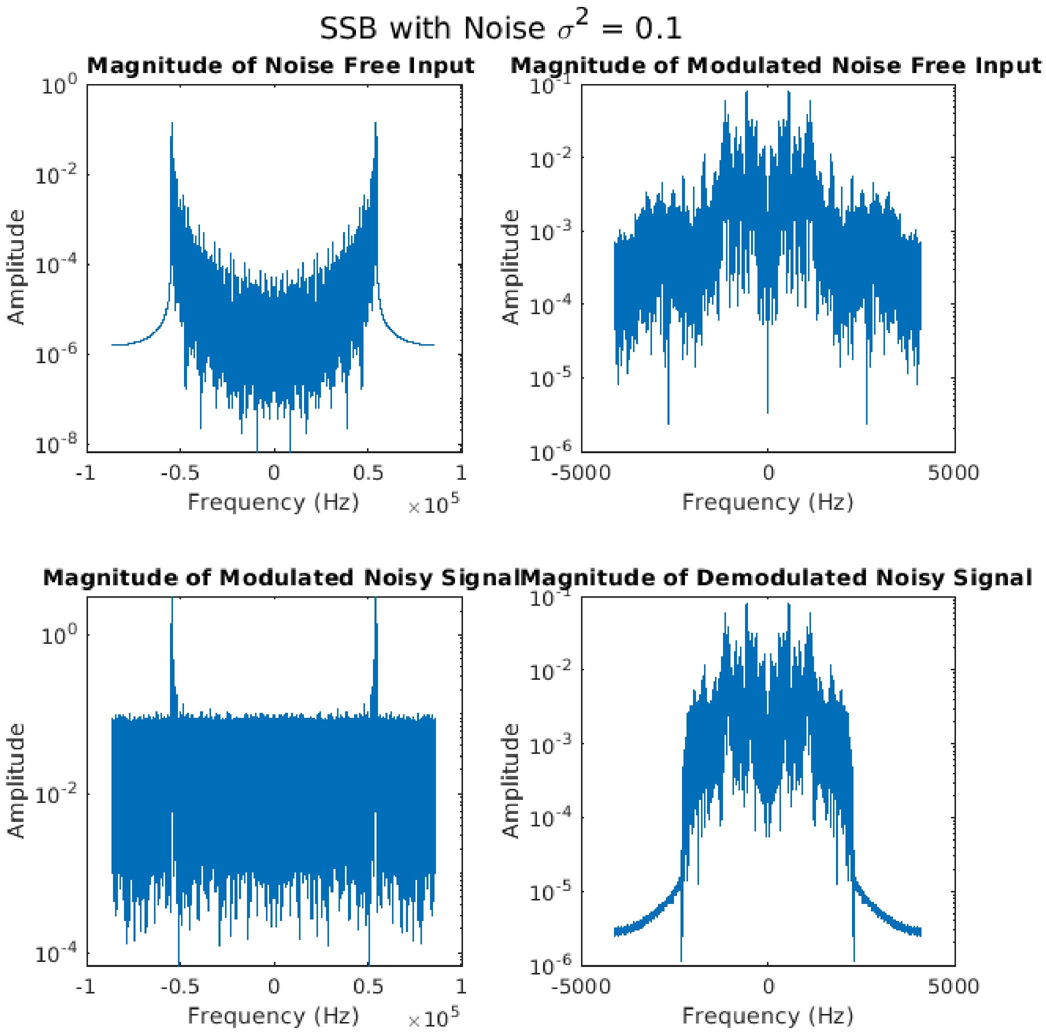
```
    "SNR for SSB with variance0.01: 17.8563"

    "Theoretical SNR for SSB with variance 0.01: 39.2897"

    "SNR for SSB with variance0.05: 15.2197"

    "Theoretical SNR for SSB with variance 0.05: 7.8579"

    "SNR for SSB with variance0.1: 13.2694"

    "Theoretical SNR for SSB with variance 0.1: 3.929"
```




SSB Demodulation
SSB Demodulated m(t)
u(t)
Time (s)
Fourier Transform of SSB Demodulated m(t)
Amplitude
Frequency (Hz)


SSB with Noise σ^2 = 0.01
Magnitude of Noise Free Input
Magnitude of Modulated Noise Free Input
Magnitude of Modulated Noisy Signal
Magnitude of Demodulated Noisy Signal
Amplitude
Frequency (Hz)
$\times 10^5$


SSB with Noise σ^2 = 0.05
Magnitude of Noise Free Input
Magnitude of Modulated Noise Free Input
Magnitude of Modulated Noisy Signal
Magnitude of Demodulated Noisy Signal
Amplitude
Frequency (Hz)
$\times 10^5$



Functions

```
%%%%%%%%%%%%%%%%%%%% FM
function [output] = FM_mod(m, fs, Ac, fc, k)
    N = length(m);
    T = N/fs;
    t = linspace(0,T,N);
    wc = fc * 2 * pi;

    %upsampling
    scale = 3*ceil(fc/fs);
    N_up = scale * N;
    fs_up = scale * fs;
    t_up = linspace(0,T,N_up);
    m_up = interp1(t,m,t_up);
```

<== u(t) = Ac cos(2pi fc t + phi(t))

phi(t) = f m(tau) dtau, with limits -inf (lower) and tau (upper)

```
    theta = cumsum(m_up)/fs_up * 2 * pi * k;
    output = (Ac * cos(wc * t_up + theta));

end

function [output] = FM_demod(m, fs, fcut, f, scale)
    %m = abs(m);
    w = 2 * pi * f;
    output = real(ifft(((fft(m)/fs)) .* (1j * w)));
    output(output<0) = 0;
    output = lowpass(output,fcut,fs);
    output = downsample(output,scale);
end

%%%%%%%%%%%%%%%%%%%% PM
function [output] = PM_mod(m, fs, Ac, fc, k)
```

<== u(t) = Ac cos(2pi fc t + phi(t))

phi(t) = kp * m(t)

```
    N = length(m);
    T = N/fs;
    t = linspace(0,T,N);
    wc = fc * 2 * pi;

    %upsampling
    scale = 3*ceil(fc/fs);
    N_up = scale * N;
    %fs_up = scale * fs;
    t_up = linspace(0,T,N_up);
    m_up = interp1(t,m,t_up);

    theta = m_up * k;
    output = (Ac * cos(wc * t_up + theta));
end

function [output] = PM_demod(m, Ac, fs, fcut, scale, sSin)
    output = Ac .* m .* sSin;
    output = lowpass(output, fcut, fs);
    output = downsample(output, scale);

end

%%%%%%%%%%%%%%%%%%%% AM
function [output] = AM_mod(m, fs, Ac, fc,a)
    N = length(m);
    T = N/fs;
    t = linspace(0,T,N);
    wc = fc * 2 * pi;
```

<==u(t) = Ac(1+a*m(t))cos(2pi fc *t)

```
    %upsampling
    scale = 3*ceil(fc/fs);
    N_up = scale * N;
    %fs_up = scale * fs;
    t_up = linspace(0,T,N_up);
    m_up = interp1(t,m,t_up);

    cSig = cos(wc * t_up);
```

```
    output = Ac * (1+ a*m_up) .* cSig;
    %output = highpass(output, 1000, fs);
end

function [output] = AM_demod(m, fs, fcut, scale)

    %m = m+0.5;
    %m(m<0) = 0;
    m = abs(m);
    %t_up = linspace(0, length(m)/(fs/scale), length(m)*scale);
    %output = cos(2*pi*55000.*t_up).*m;
    output = lowpass(m,fcut,fs);
    %output = highpass(output, 2, fs);
    output = downsample(output, scale);
    %output = lowpass(output, 20000, fs);
end

%%%%%%%%%%%%%%%%%%%% SSB
function [output] = SSB_mod(m, fs, Ac, fc)
    N = length(m);
    T = N/fs;
    t = linspace(0,T,N);                    <==u(t) = m(t)cos(2pi fc *t) - mhat(t)sin(2pi fc *t)
    wc = fc * 2 * pi;

    %upsampling
    scale = 3*ceil(fc/fs);
    N_up = scale * N;
    fs_up = scale * fs;
    t_up = linspace(0,T,N_up);
    m_up = interp1(t,m,t_up);
    wd = linspace(-pi, pi, N_up);
    f = wd *fs_up / (2 * pi);

    cSig = cos(wc * t_up);
    sSig = sin(wc * t_up);
    mH = real(ifft(fft(m_up).*(-1j.*sign(f))));
    output = Ac * (m_up .* cSig - mH .* sSig);
    %output = highpass(output, 5000, fs);
end

function [output] = SSB_demod(m, fs, fcut, scale, sSin)
    m = m .* sSin;
    output = lowpass(m,fcut,fs);
    output = downsample(output, scale);
end

    "SNR for FM with variance0.01: 15.9078"

    "Theoretical SNR for FM with variance 0.01: 8035.1212"

    "SNR for FM with variance0.05: 13.1083"

    "Theoretical SNR for FM with variance 0.05: 1607.0242"
```

```
   "SNR for FM with variance0.1: 10.7714"

   "Theoretical SNR for FM with variance 0.1: 803.5121"

   "SNR for FM with k = 35000(variance = 0.01) is: -1.2257"

   "Theoretical SNR for FM with k = 35000(variance = 0.01) is:
2008.7803"

   "SNR for FM with k = 70000(variance = 0.01) is: -0.61758"

   "Theoretical SNR for FM with k = 70000(variance = 0.01) is:
8035.1212"

   "SNR for FM with k = 140000(variance = 0.01) is: -0.2562"

   "Theoretical SNR for FM with k = 140000(variance = 0.01) is:
32140.4848"
```

Published with MATLAB® R2020b

	$\sigma^2 = 0.01$	$\sigma^2 = 0.05$	$\sigma^2 = 0.1$
FM Experimental	15.9048	13.0716	10.7612
FM Theoretical	8035.1212	1607.0242	803.5121
PM Experimental	5.7486	1.2354	0.61769
PM Theoretical	6.1769	5.2732	5.7787
AM Experimental	18.7191	10.1124	5.742
AM Theoretical	3.4931	0.69863	0.34931
SSB Experimental	17.8541	15.2751	13.3274
SSB Theoretical	39.2897	7.8579	3.929

```
"SNR for SSB with variance0.01: 17.8541"

   "Theoretical SNR for SSB with variance 0.01: 39.2897"

   "SNR for SSB with variance0.05: 15.2751"

   "Theoretical SNR for SSB with variance 0.05: 7.8579"

   "SNR for SSB with variance0.1: 13.3274"

   "Theoretical SNR for SSB with variance 0.1: 3.929"
```

Discussion:

1. Carson's rule

   FM and PM modulations both have infinite bandwidth; however, according to Carson's rule, 98% of an FM/PM modulated signal's power is bounded in bands centered around -fc and fc ($B_c = 2(\beta + 1)W$).

   For FM modulation we performed in this project, (graph on pg. 6), the bandwidth is approximately 160 kHz. $\beta_P = k_f \max(m(t)) / W = 70000(1)/3000 = 23.3$; $W \approx 3$ kHz => $B_c = 2(23.3 + 1)3000 = 146$ kHz which fairly close to our modulated bandwidth of 160 kHz.

   For PM modulation we performed in this project, (graph on pg. 14), the bandwidth is approximately 16 kHz. $\beta_P = k_p \max(m(t)) = 2(1) = 2$; $W \approx 3$ kHz => $B_c = 2(2+1)3000 = 18$ kHz => pretty close to 16kHz.

2. There is no noise, so they ought to sound like what you put in more-or-less exactly, save the case of conventional AM – what is different in the sound at the output of the conventional AM system, qualitatively? Why? Make sure the signal for conventional AM does not sound too bad!

   When transmitting a message using conventional AM, we are encoding the message signal in the envelope of the transmitted signal. As we rectify the signal, we are throwing away the negative quadrant of information, passing through a lowpass filter that filters out the high-frequency components and it will result in an approximation of our message.

3. Isolate a noise process that makes conventional AM sound qualitatively noisy, but audible. Vary the modulation index by a factor of 2 both downward and upward, and observe the results of this noise process on the three new modulated signals (making similar plots to those you made for the previous problem but with three curves plotted on each subplot). What effect do you expect this to have?

As the modulation increase, an increase in SNR value is also expected.

AM case:

"SNR for AM with a = 1(variance = 0.01) is: 20.1247"

"Theoretical SNR for AM with a = 1(variance = 0.01) is: 0.87328"

"SNR for AM with a = 2(variance = 0.01) is: 18.7091"

"Theoretical SNR for AM with a = 2(variance = 0.01) is: 3.4931"

"SNR for AM with a = 4(variance = 0.01) is: 16.4272"

"Theoretical SNR for AM with a = 4(variance = 0.01) is: 13.9726"

	a = 1	a = 2	a = 4
Experimental	20.1247	18.7091	16.4272
Theoretical	0.87328	3.4931	13.9726

As the modulation index increases, the theoretical SNR increases; however, the experimental results do not seem to change much, and even decreased a bit.

FM case:

"SNR for FM with k = 35000(variance = 0.01) is: -1.2248"

"Theoretical SNR for FM with k = 35000(variance = 0.01) is: 2008.7803"

"SNR for FM with k = 70000(variance = 0.01) is: -0.61361"

"Theoretical SNR for FM with k = 70000(variance = 0.01) is: 8035.1212"

"SNR for FM with k = 140000(variance = 0.01) is: -0.25642"

"Theoretical SNR for FM with k = 140000(variance = 0.01) is: 32140.4848"

	k = 35000	a = 70000	a = 140000
Experimental	-1.2248	-0.61361	-0.25642
Theoretical	2008.7803	8035.1212	32140.4848

As the modulation index increases, the theoretical SNR increases as well as the experimental results; however, both of them seem to be very wrong.

PM case:

"SNR for PM with k = 1(variance = 0.01) is: 5.7787"

"Theoretical SNR for PM with k = 1(variance = 0.01) is: 4.9966"

"SNR for PM with k = 2(variance = 0.01) is: 5.7818"

"Theoretical SNR for PM with k = 2(variance = 0.01) is: 19.9864"

"SNR for PM with k = 4(variance = 0.01) is: 5.3667"

"Theoretical SNR for PM with k = 4(variance = 0.01) is: 79.9458"

	k = 35000	a = 70000	a = 140000
Experimental	5.7787	5.7818	5.3667
Theoretical	4.9966	19.9864	79.9458

As the modulation index increases, the theoretical SNR increases; while the experimental results do not vary much.

4. The results of the experiment do not seem accurate. The experimental SNR does not match the theoretical values of SNR despite numerous attempts of adjusting the modulation index and Ac values and modifying the modulation and demodulation functions. The author has run out of time for further adjustment, potentially including modifying the SNR calculation methods. This project demonstrated the difficulties when it comes to modulating and demodulating signals as many problems were encountered.

	$\sigma^2 = 0.01$	$\sigma^2 = 0.05$	$\sigma^2 = 0.1$
FM Experimental	15.9048	13.0716	10.7612
FM Theoretical	8035.1212	1607.0242	803.5121
PM Experimental	5.7486	1.2354	0.61769
PM Theoretical	6.1769	5.2732	5.7787
AM Experimental	18.7191	10.1124	5.742
AM Theoretical	3.4931	0.69863	0.34931
SSB Experimental	17.8541	15.2751	13.3274
SSB Theoretical	39.2897	7.8579	3.929