Tyler Castelli

ECE 3512: Signals – Continuous and Discrete

Department of Electrical and Computer Engineering, Temple University, Philadelphia, PA 1912

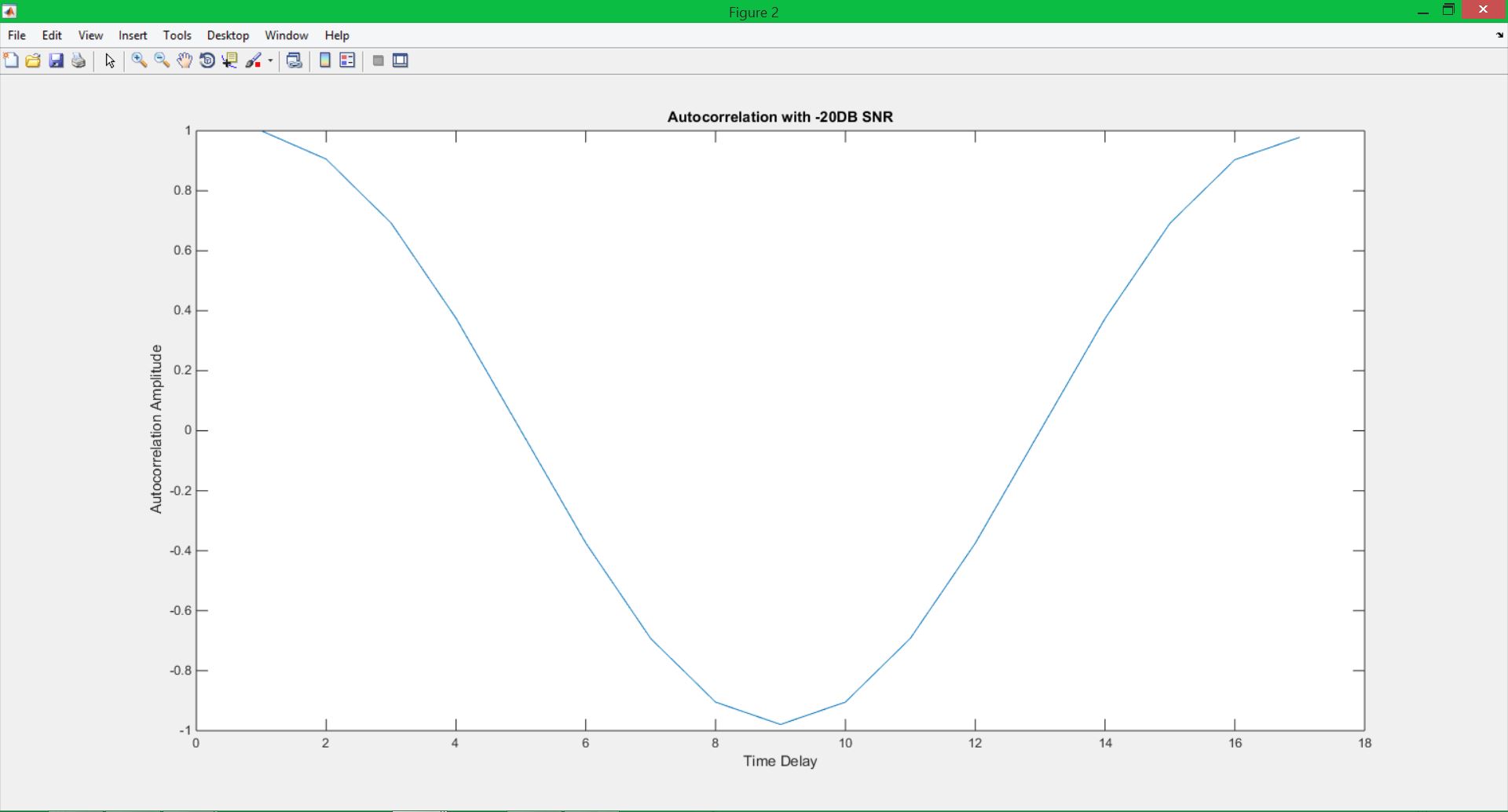
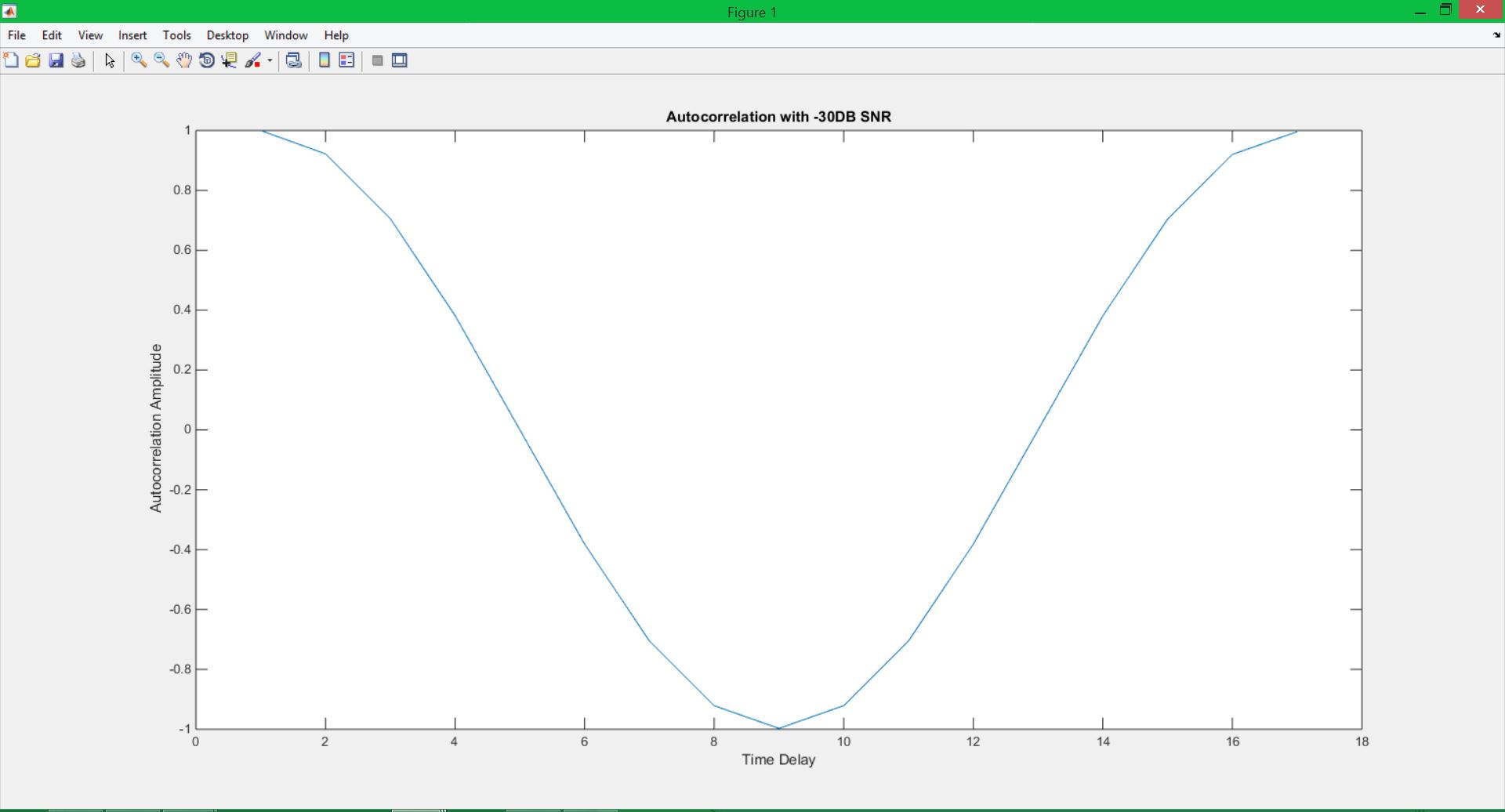
# Problem Statement

The objective of this assignment was to create a sinusoid that also contained Gaussian white noise, and find some characteristics of the signal. These included finding the autocorrelation function at different signal to noise ratios, the magnitude spectrum, and the output of the signal after filtering. Each task was done using a signal of 500 Hz, 1s period, and the sampling frequency and signal t noise ration altered depending on the task.

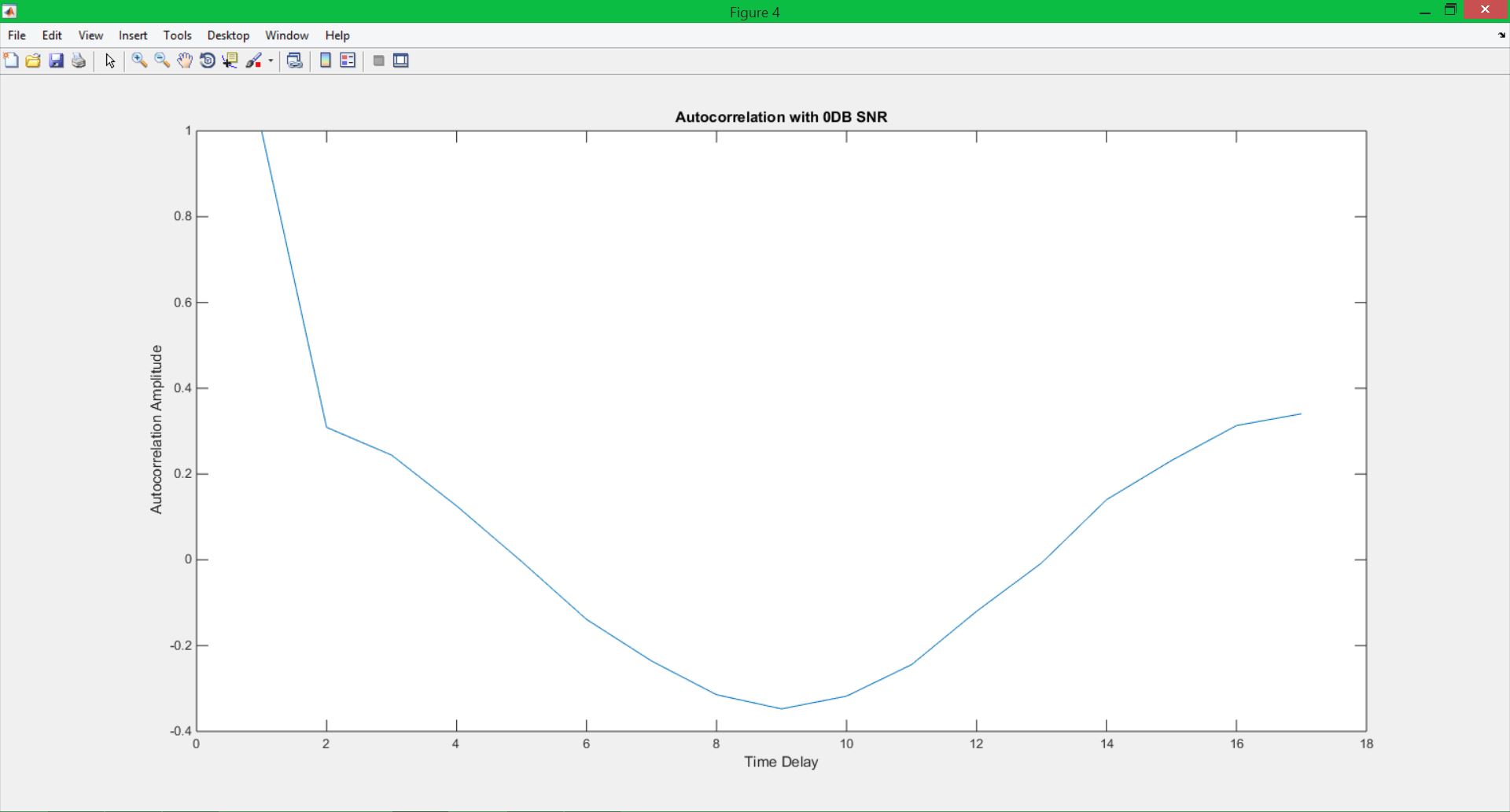
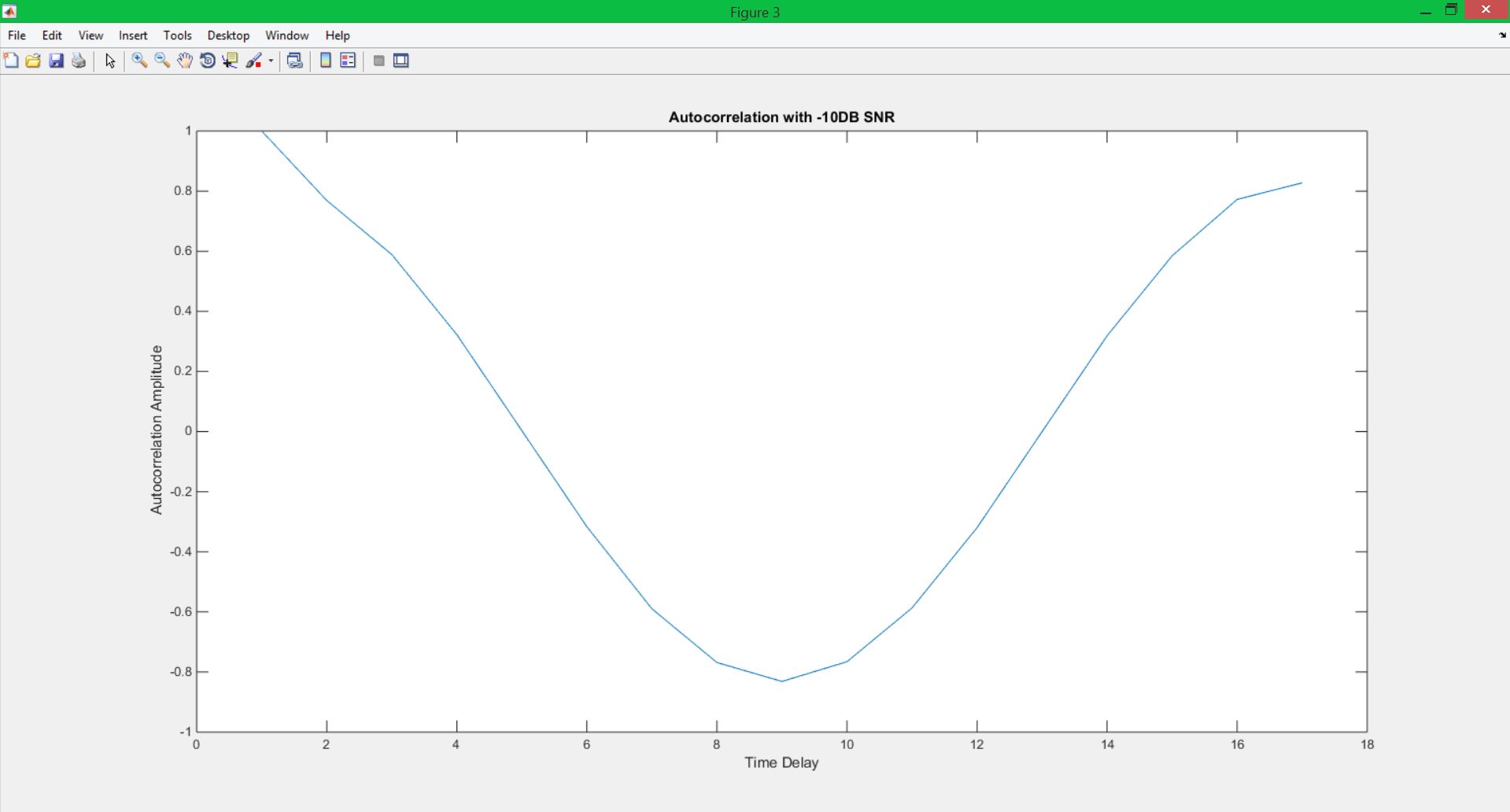
# Approach and Results

Task 1 of this assignment was to simply write the signal generating function. The code for this will be explained in the next section. The next task was to find the autocorrelation function for the first 1 lags of a 500 Hz generated signal sampled at 8 kHz, and repeat this for a range of signal of noise ratios from -30 dB to 30dB. The plots of the autocorrelation functions are shown below in the first 7 figures.

Figures 1 and 2: Autocorrelation at -30 dB and -20 dB respectively



Figures 3 and 4: Autocorrelation at -10 dB and 0 dB respectively



Figures 5 and 6: Autocorrelation at 10 dB and 20 dB respectively

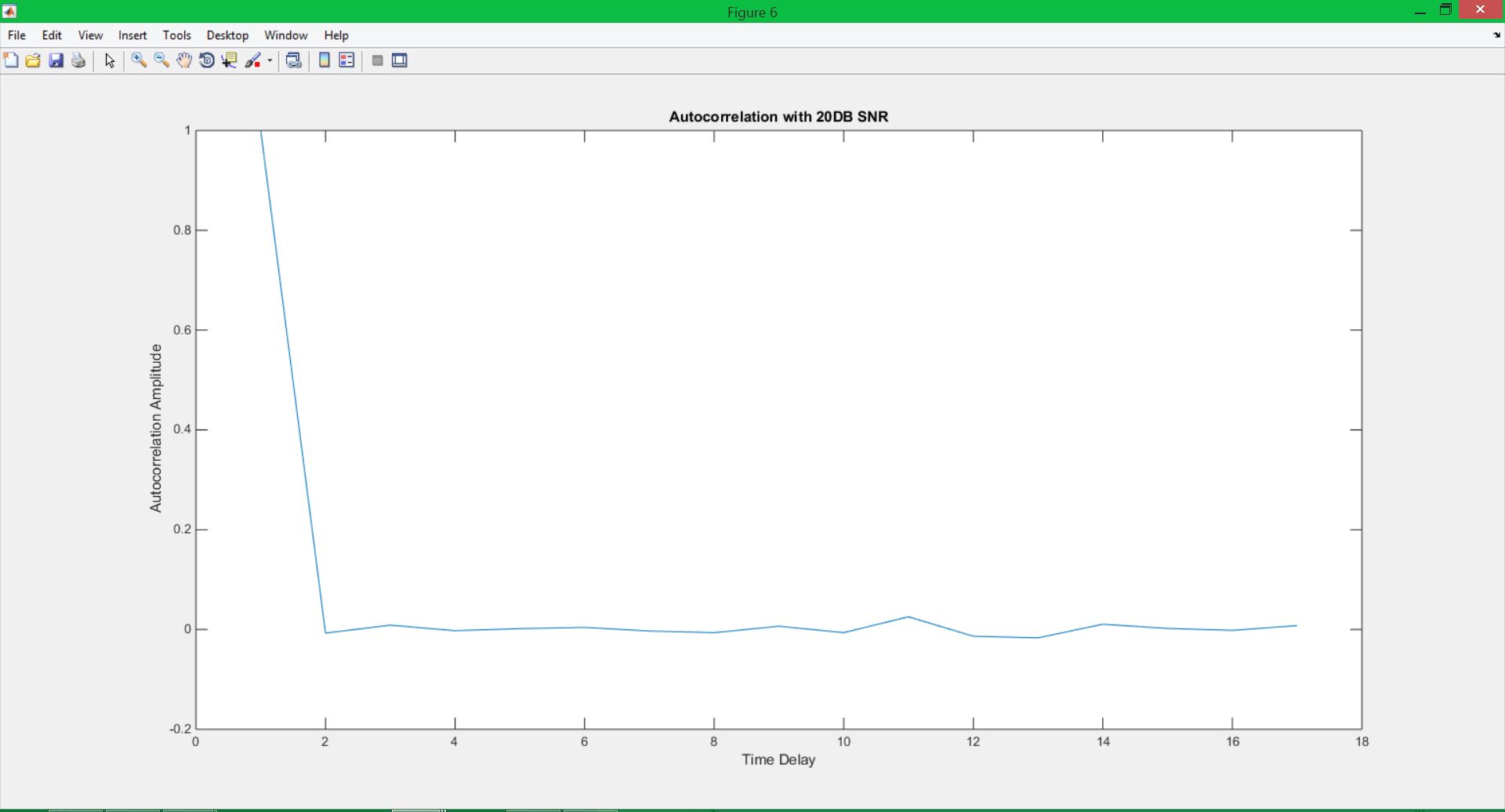
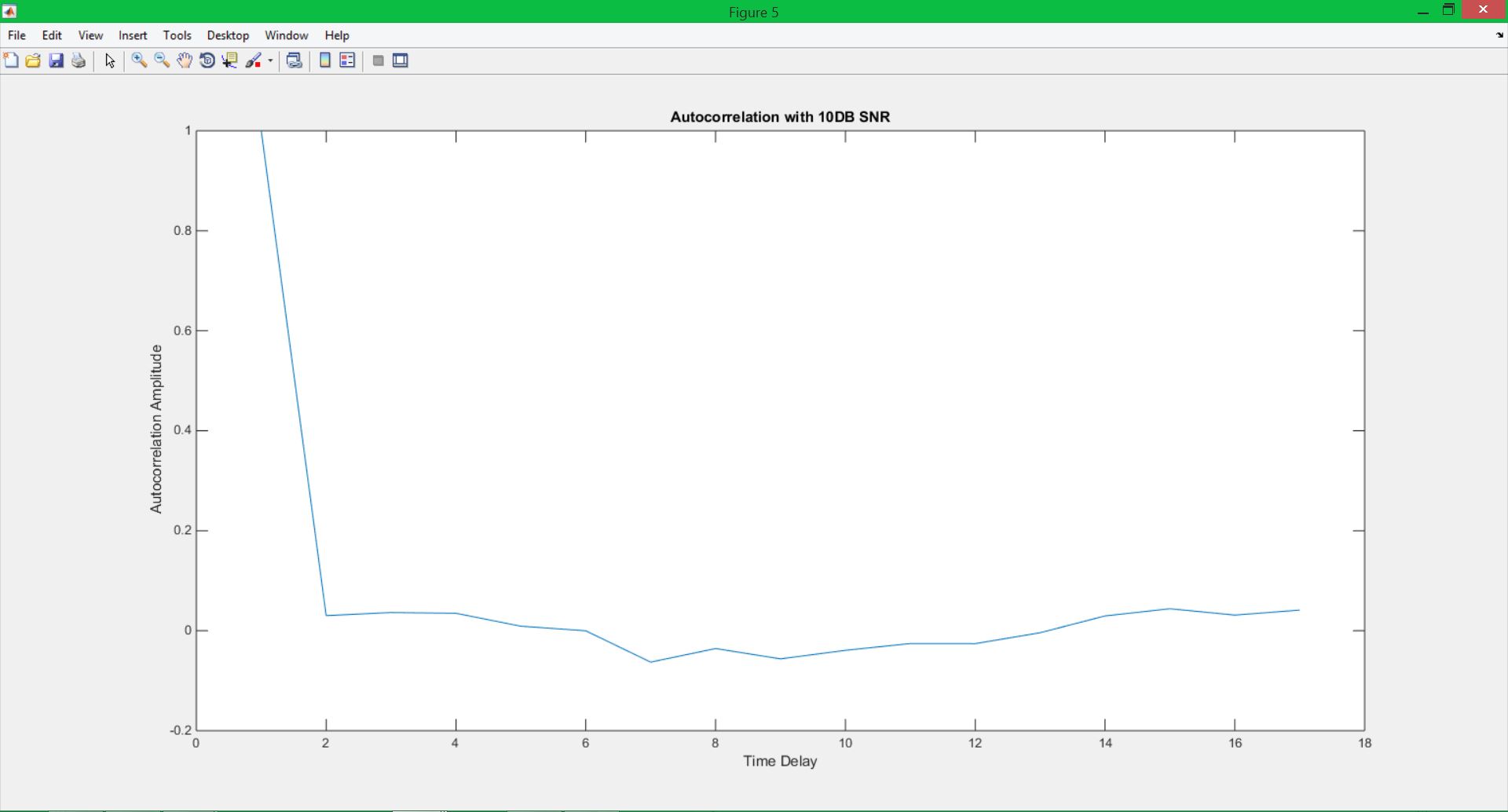
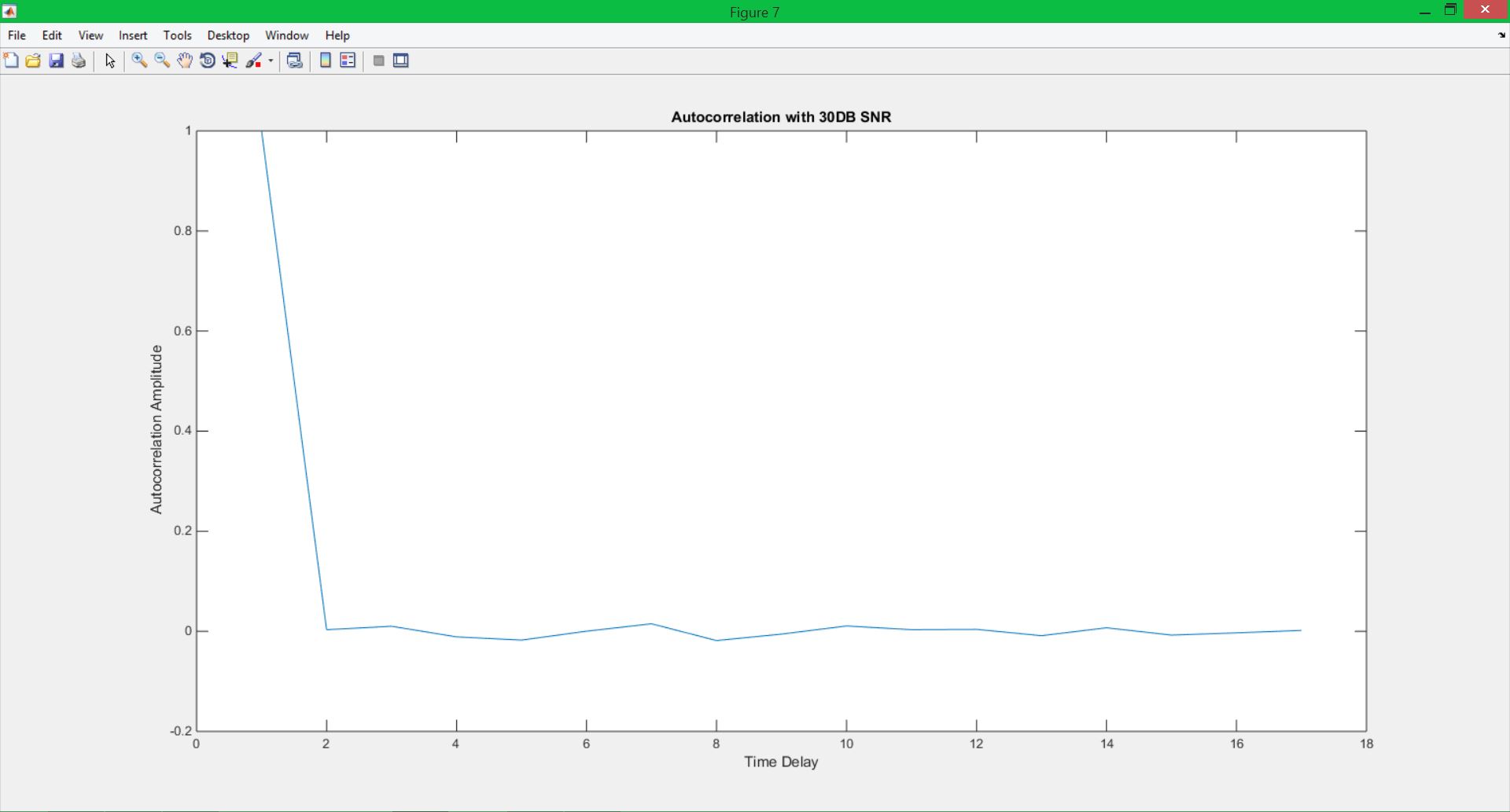


Figure 7: Autocorrelation at 30 dB



We can see that the autocorrelation begins to be significantly effected at 0 dB. This result will be further explained in the conclusion section. Next, the Fourier transform and the magnitude spectrum of the signal are found using a -30 dB SNR and 30 dB SNR. These plots are shown in figures 8 and 9.

Figure 8: Magnitude plot of the Signal at -30 dB SNR

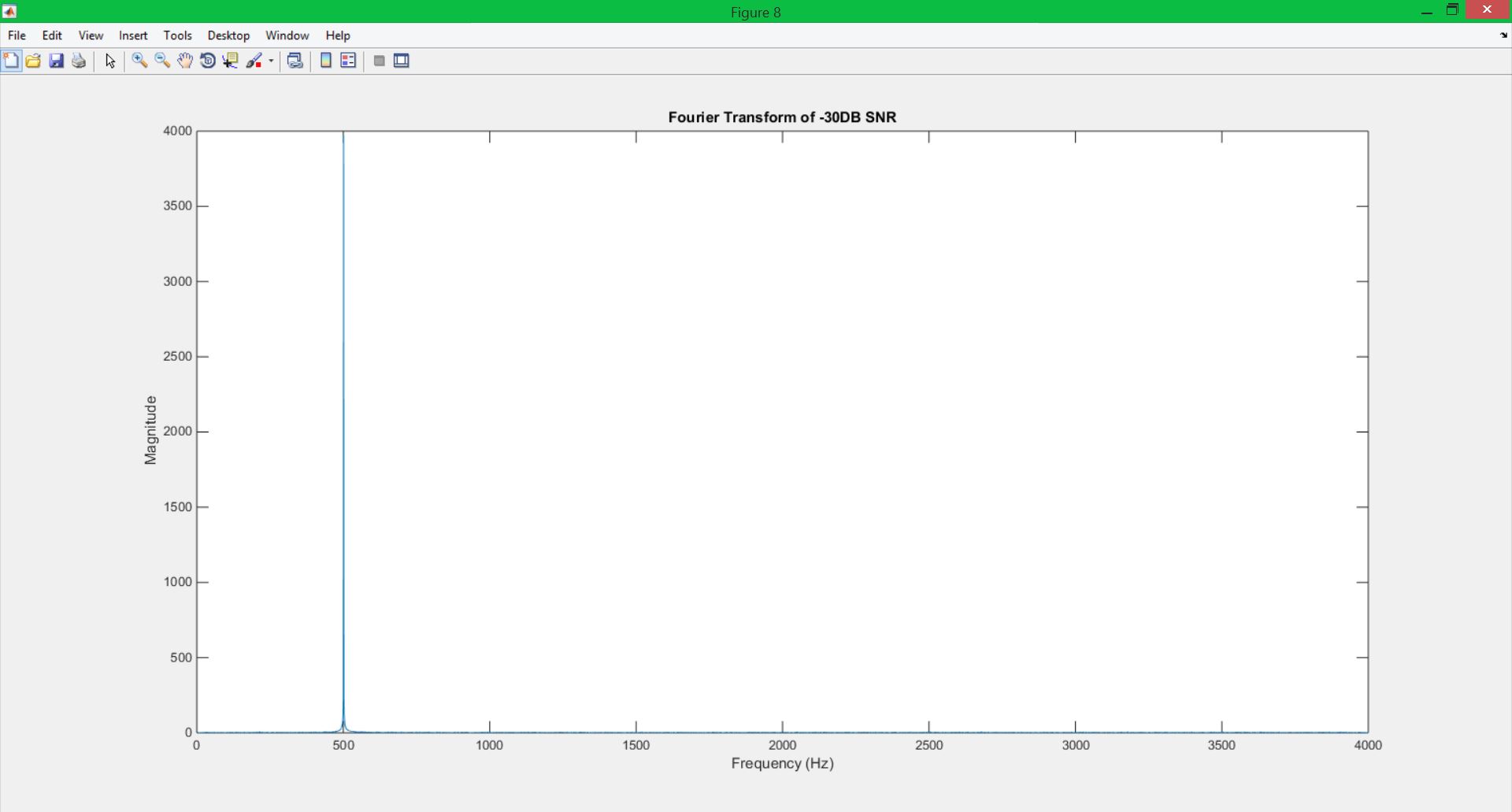
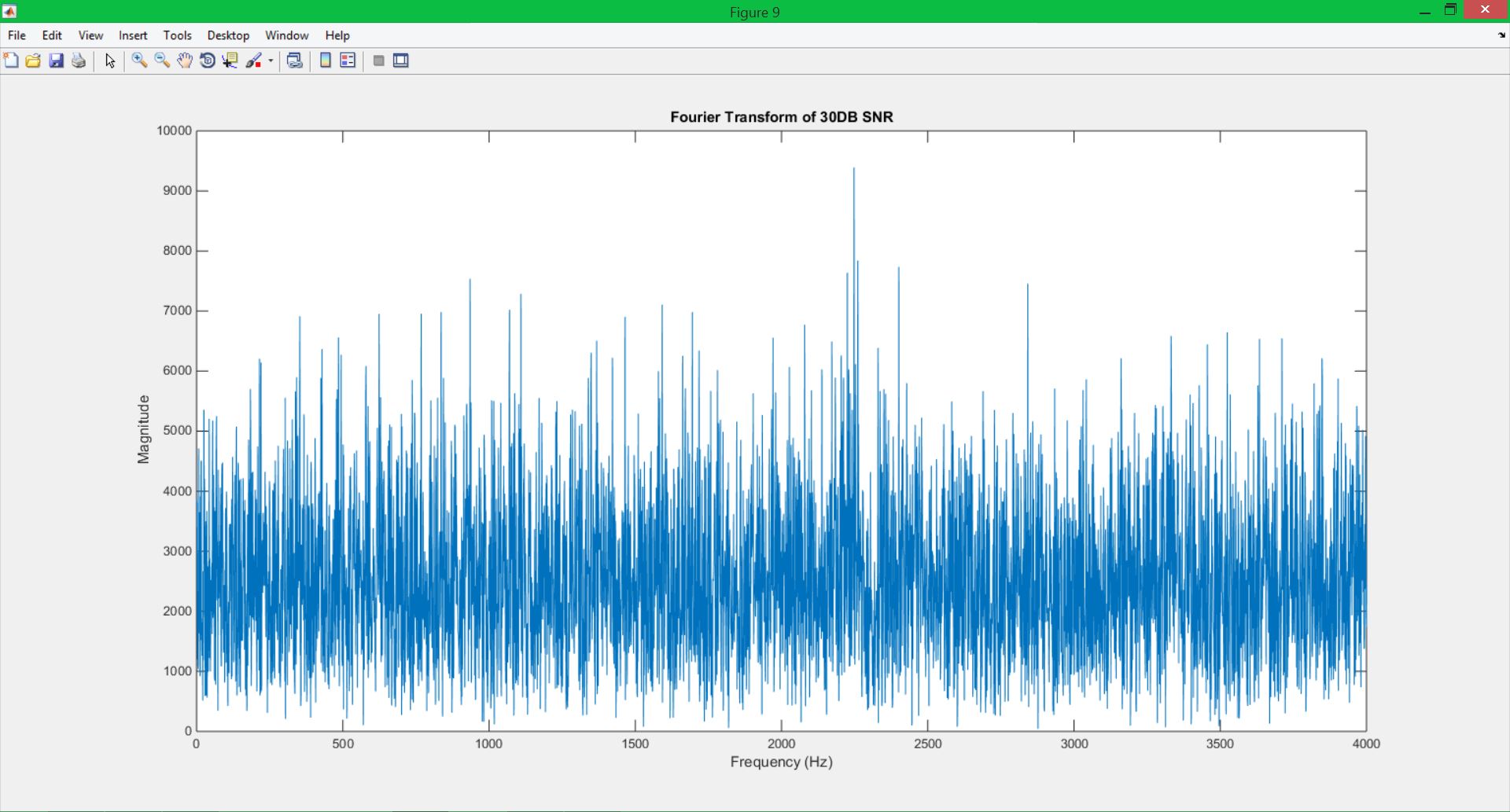


Figure 9: Magnitude plot of Signal at 30 dB SNR



Here we see that at a low signal to noise ration the magnitude spectrum is unaffected. We get the magnitude spectrum of the sine wave itself. However, at a high signal to noise ratio, the magnitude spectrum becomes distorted.

Finally, I filtered the signal through a digital filter of the form:



From this new filtered signal, the autocorrelation and magnitude spectrum were plotted. These are shown in figures 10 and 11.

Figure 10: Autocorrelation of Filtered Signal

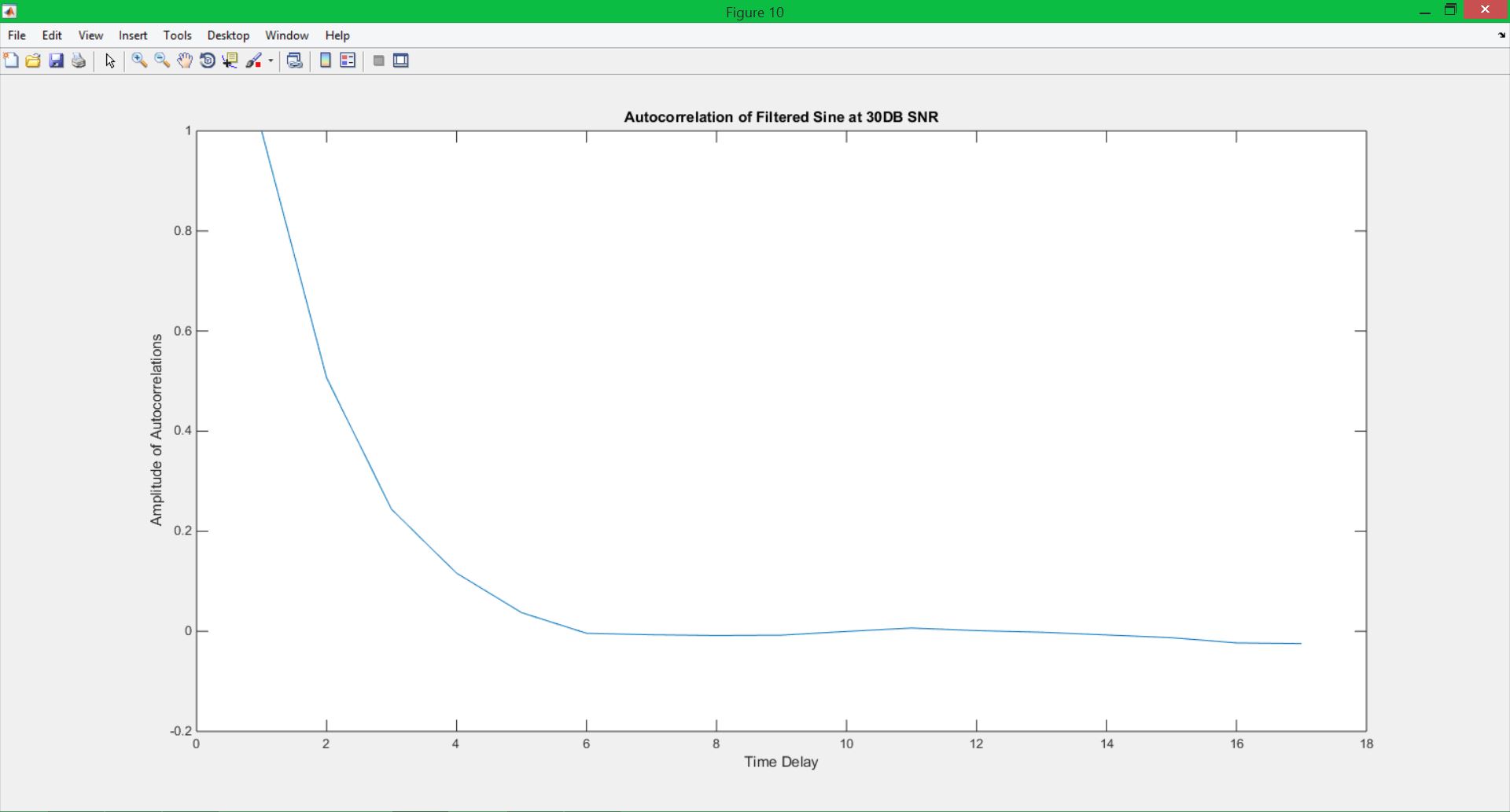
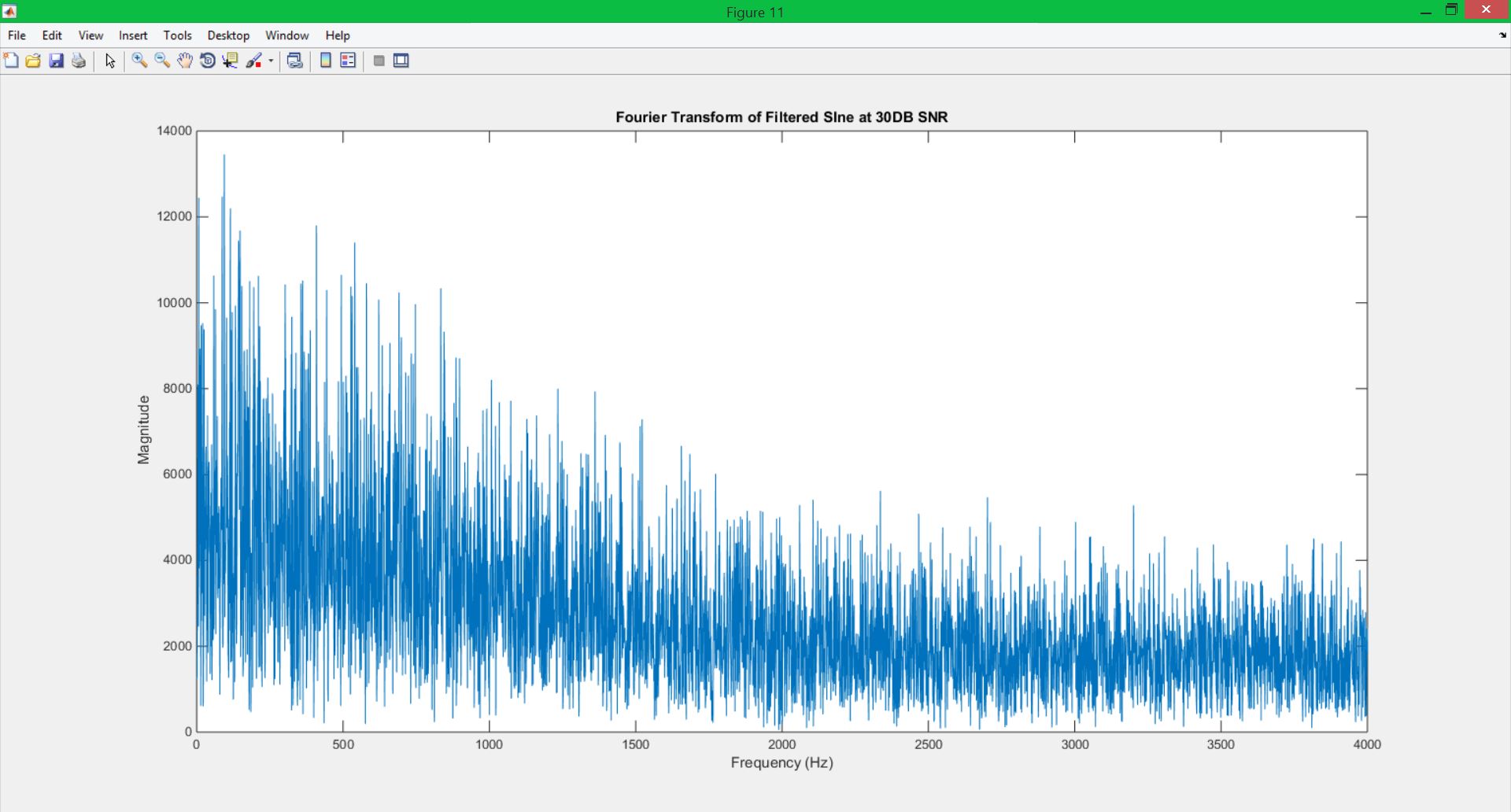


Figure 11: Magnitude spectrum of the filtered signal



These plots will be further explained in the conclusion section.

# MATLAB Code

%question 1

function sineWave = noisySine(f, T, fs, SNR)

t = 0:(1/fs):T;

w = 2\*pi\*f;

sine = sin(w.\*t);

GWN = wgn(1,length(t),SNR);

sineWave = sine+GWN;

end

**This section is the function that generates the sine wave. It simply creates a sinewave based on the given parameters and adds noise to it based on the SNR.**

%%

%question 2

clear all; close all; clc;

SNR = -30:10:30;

f = 500;

fs = 8000;

T = 1;

lags = 16;

autoC = zeros(lags+1,length(SNR));

for i = 1:length(SNR)

sig = noisySine(f,T,fs,SNR(i));

autoC(:,i) = autocorr(sig,lags);

end

for k = 1:7

figure(k);

plot(autoC(:,k));

str = sprintf('Autocorrelation with %dDB SNR',SNR(k));

title(str);

ylabel('Autocorrelation Amplitude');

xlabel('Time Delay');

end

**This section of code is for question 2. The signal is generated using different SNRs and the autocorrelation of each signal is calculated and plotted.**

%%

%question 4

points = 8192;

FFTneg30 = fft(noisySine(f,T,fs,SNR(1)));

FFT30 = fft(noisySine(f,T,fs,30));

mag1 = abs(FFTneg30);

mag2 = abs(FFT30);

k = k+1;

figure(k);

plot(mag1);

str = sprintf('Fourier Transform of %dDB SNR',-30);

title(str);

ylabel('Magnitude');

xlabel('Frequency (Hz)');

xlim([0, 4000]);

figure(k+1);

plot(mag2);

str = sprintf('Fourier Transform of %dDB SNR',30);

title(str);

ylabel('Magnitude');

xlabel('Frequency (Hz)');

xlim([0, 4000]);

**This section takes the signal and finds the magnitude spectrum. The Fast Fourier transforms of the signals with -30 dB and 30 dB SNR are calculated, and the magnitude of each FFT is taken and plotted to generate the Magnitude spectrum.**

%%

%question 5

sig = noisySine(f,T,fs,SNR(7));

y = zeros(1,length(sig));

for i = 1:length(sig)

if ((i-1) <= 0)

y(i) = sig(i);

end

if ((i-1) > 0)

y(i) = 0.5\*y(i-1) + sig(i);

end

end

AC = autocorr(y,lags);

FFTy = abs(fft(y,points));

k = k+2;

figure(k);

plot(AC);

str = sprintf('Autocorrelation of Filtered Sine at %dDB SNR',30);

title(str);

ylabel('Amplitude of Autocorrelations');

xlabel('Time Delay');

figure(k+1);

plot(FFTy);

str = sprintf('Fourier Transform of Filtered SIne at %dDB SNR',30);

title(str);

ylabel('Magnitude');

xlabel('Frequency (Hz)');

xlim([0, 4000]);

**In the last section the signal is filtered by putting it though the equation that defines the filter. The autocorrelation function and the absolute value of the FFT were then taken and plotted.**

# Conclusions

In the first 7 figures we can see that the autocorrelation function begins to be effected significantly by the signal to noise ratio at 0dB. This follows intuition, as that is when there is even parts noise and signal. When the ratio is negative, there is more signal, and when it is positive, there is more noise. This is why the autocorrelation follows that of a sinewave in the lower SNRs, and becomes flat lined and distorted in the higher SNRs.

Next we see the magnitude spectrum at different signal to noise ratios. Again, the result follows intuition, as when there is a low signal to noise ratio, the magnitude spectrum is that of a pure sinewave, an impulse at the frequency of the signal. When there is more noise added to the signal, the magnitude becomes distorted because there is extra disturbance, added to the signal.

Finally, the signal is filtered and the magnitude spectrum and autocorrelation functions are plotted again and compared to those of the previous task. These plots both use a SNR of 30 dB. The filtering smooths the autocorrelation out, but the overall shape still resembles that of the unfiltered autocorrelation at the same SNR. The magnitude plot, while still distorted from the noise, takes on the shape of the autocorrelation. This shows how filtering can help to remove noise form a signal.