Midterm Project

Quantization Noise

Name: Jinhan Zhang

Class: ECE 251A Digital Signal Processing I

Date: 02/14/2017

Quantization Noise

• Objective

Use Matlab to analyze the properties of noise introduced by uniform quantization, and verify the correctness of the model we established for uniform quantization errors.

• Background

The A/D convertor is widely used in real-world systems to convert a voltage or current amplitude at its input into a binary code which is quantized closest to the amplitude of the input. This process introduces noise as our system noise and certainly we want to analyze that noise and make sure that it won't be too high. So in this project, we'd like to analyze the characteristics of the quantization noise with different values of fractional bits and compare them with our theoretical values calculated by our model.

• Approach

Firstly, we use a single cycle of a 256-point long sinusoid as original signal, and investigate quantization in case of 1, 3 and 7 fractional bits. Secondly, we investigate the |FFT|(dB) of both the original and quantized signals, then subtract original signal from the quantized signal to get the error signal for all 3 cases respectively. Thirdly, we investigate the histogram, autocorrelation and power spectrum estimation of the error signal e(n) to see its properties, and also the cross-correlation between error signal and original signal. Finally, we look at the case of "dithering" input for b=1 by adding an uncorrelated noise sequence to the original signal.

• Results

Fig. 1 shows the 256-point long single-cycle sinusoid as our original signal and Fig. 2 shows its continuous form. Fig. 3 is the |FFT|(dB) of the original signal $\mathrm{x(n)}$. We can see 2 impulses at ± 0.003906 cycle/sample, which corresponds to the frequency $f_x = \frac{1}{N}$, where N=256.

Fig. 4-6 show the quantized signal $\mathrm{Q[x(n)]}$ for fractional bits $\mathrm{b} = 1, 3, 7$ respectively. We can see that as the increase of the number of fractional bits, the quantized signal tends to be smoother. Fig. 7-9 show the |FFT|(dB) of the quantized signal $\mathrm{Q[x(n)]}$. Due to that we have cut off large negative values, we can only see 1 impulse in the plot. And also we can see that the noise will have a lower dB magnitude with a higher number of fractional bits.

Fig. 10-12 show the error signal $\mathrm{e(n) = Q[x(n)] - x(n)}$ for fractional bits $\mathrm{b} = 1, 3, 7$ respectively. We can see from the figures that when b=1, 3 the error signals are more ordered, while for b=7 the error signal is more like a uniformly distributed noise. Fig 13-15 are the histograms for different b values, we can see that they can be approximately assumed as uniform distribution, which matches our quantization error model as uniformly distributed noise.

Fig. 16-18 show autocorrelation sequence estimation for $\mathrm{b} = 1{,}3{,}7$ respectively when $\mathrm{m} = \{0,1,\dots,63\}$. We can see from the plot that in the b=1 case, $\mathrm{e}(\mathrm{n})$ is much more correlated than other 2 cases. Then I will calculate theoretical expected value for $\emptyset_{ee}(0)$.

$$\mathrm{E}[\emptyset_{ee}(0)] = \frac{1}{N}\sum_{n=0}^{N-1} E[e^2(n)]$$

Because $E[X^2]$=$(E[X])^2 + Var[X]$, so we have

$$\mathrm{E}[\emptyset_{ee}(0)] = \frac{1}{N}\sum_{n=0}^{N-1} \{(E[e])^2 + Var[e]\}$$

As we assume that in our model $\mathrm{e}(\mathrm{n})$ is a random variable uniformly distributed from $-\Delta/2$ to $\Delta/2$, where Δ is the quantization level length. For random variable uniformly distributed in $[-\Delta/2,\ \Delta/2]$, we have $E[e] = 0$ and $Var[e] = \frac{\Delta^2}{12}$, so

$$\mathrm{E}[\emptyset_{ee}(0)] = \frac{\Delta^2}{12}$$

For $\mathrm{b} = 1, 3$ and 7, Δ= $\frac{1}{2}$, $\frac{1}{8}$ and $\frac{1}{128}$ respectively. Thus we can draw a table to compare the theoretical values computed above with empirical result in Fig. 16-18 as Table 1.

Table 1. theoretical expected values of $\emptyset_{ee}(0)$ vs. empirical values

b \ source	Theoretical	Empirical
1	2.083×10^{-2}	1.736×10^{-2}
3	1.302×10^{-3}	1.186×10^{-3}
7	5.086×10^{-6}	4.590×10^{-6}

We can see from Table 1 that the empirical values will have smaller relative errors with theoretical values with the increase of b. This is mainly because that our model will fit the actual cases better and better with fractional bits growing, thus we are expected to get a better estimation with higher b values.

Fig. 19-21 show the power spectral density estimation for $\mathrm{b} = 1,3,7$ respectively. I use single periodogram to do power spectral estimation and normalize them by $(f_s MU)^{-1}$, where $f_s = 1\mathrm{Hz}$, M=256 and U is sum of the square of window function which is 256-point Hamming here. We can see from the figures that with a smaller b value the power spectrum will have a relatively bumpier trend while larger b value gives us a flatter power spectrum. Then I will calculate the theoretical power spectrum using the quantization error model.

We have

$$\sigma_e^2 = \frac{\Delta^2}{12}$$

where $\Delta = 2^{-b}X_m$ and where $X_m = 1$ here.

So the autocorrelation function would be

$$\emptyset_{ee}[m] = \sigma_e^2\delta[m]$$

Since e(n) is a sample sequence of a stationary random process, we can apply Wiener–Khinchin theorem to get

$$P_{ee}(e^{jw}) = \sigma_e^2 = \frac{2^{-2b}}{12} \qquad |\omega| \leq \pi.$$

Thus the power spectrum estimation in dB would be

$$10log_{10}\left(P_{ee}(e^{jw})\right) = 10log_{10}\left(\frac{1}{12}\right) - 2b \times 10log_{10}(2)$$

So we have

$$10log_{10}\left(P_{ee}(e^{jw})\right) = -6.02b - 10.79$$

The theoretical power spectrum level is obviously constant when b is fixed , so the power spectrum theoretically is flat. The theoretical power spectrum levels for different b are shown as Table 2.

Table 2. Theoretical power spectrum levels for different b

b	Theoretical power spectrum level
1	-16.81 dB
3	-28.85 dB
7	-52.93 dB

Fig. 22 shows the power spectrum(dB) for all 3 cases in one plot. We can see that larger number of fractional bits will result in lower power spectrum level. And the empirical power spectrum estimations are all close to the theoretical levels but the empirical ones are slightly lower. This is mainly due to that we are using $2^{b+1}+1$ quantization levels instead of 2^{b+1}. One more level will result in lower variance, so empirical $P_{ee}(e^{jw})$ would be smaller than theoretical value.

Fig. 23-25 show the cross-correlation for $\mathrm{m}=\{-63,\ldots,0,\ldots,63\}$ between the error signal and the original signal. We can see that when b is small the error signal and the original signal are highly correlated, while they tend to be uncorrelated with the increase of b. So when we have small number of fractional bits, the assumption "The error sequence is uncorrelated with the sequence x[n]." doesn't hold true, thus our model won't fit the empirical results well when b=1.

Fig. 26 shows the error signal for the case we dither the original signal by adding an uncorrelated zero-mean noise uniformly distributed between $\pm 2^{-b}/2$ when b=1. Fig. 27 is its corresponding autocorrelation estimation. We can see that compared with Fig. 16 $\emptyset_{ee}(0)$ is roundly doubled while $\emptyset_{ee}(m)(m \neq 0)$ becomes much smaller. Thus its power spectrum estimation would be flatter as Fig. 28 shown compared with Fig. 19. Fig. 29-31 shows the similar case but the additional noise distribution range changes into $\pm 2^{-b}/20$. We can see from the plots that they are very similar to Fig. 10, Fig. 16 and Fig. 19 respectively. This is because the range of the noise is relatively small, so they won't make as much sense as above.

- **Summary**

In this project, we have analyzed the properties of quantization noise. We see that when b is small the quantization error time series e(n) would be highly correlated and also the power spectrum estimation would not be much like what we get using our theoretical model. And we have also tried to dither the original signal with uniformly distributed noise of different ranges. For the b=3,7 cases the model calculation fits the empirical result much better and the error sequences are much less correlated.

- **Plots**

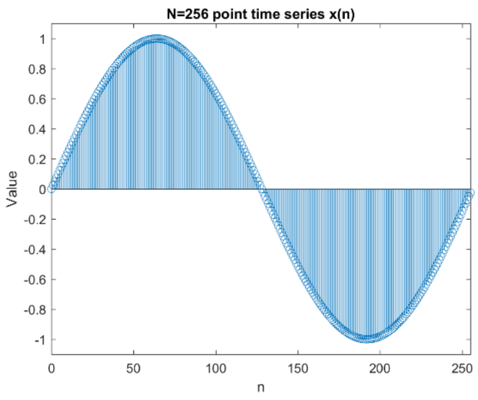

Fig. 1

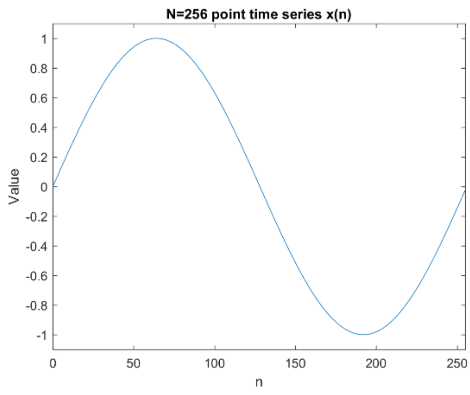

Fig. 2

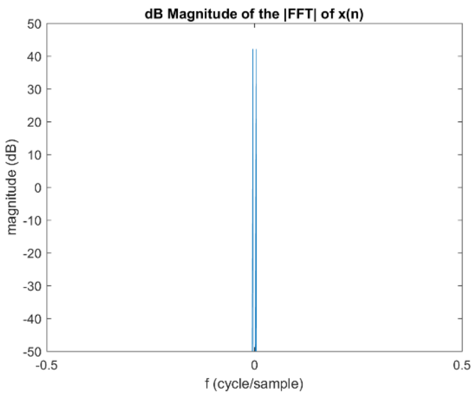

Fig. 3

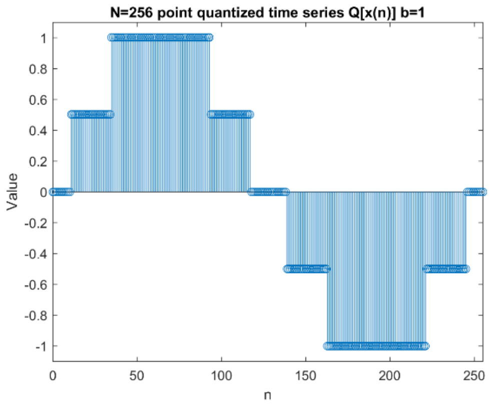

Fig. 4

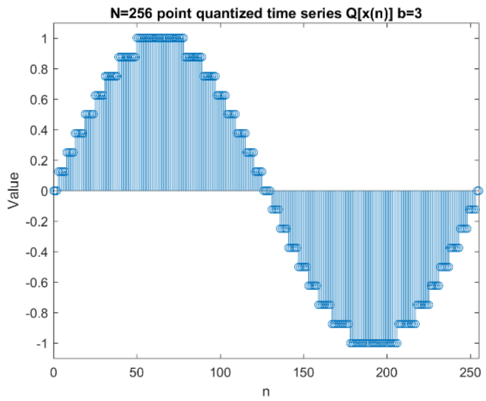

Fig. 5

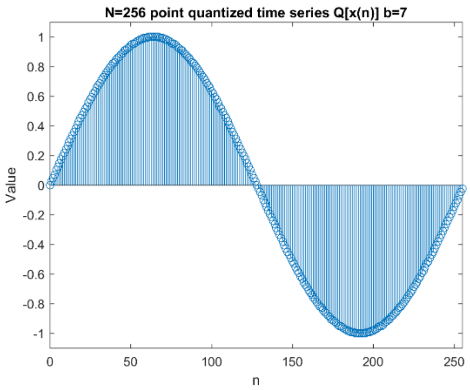

Fig. 6

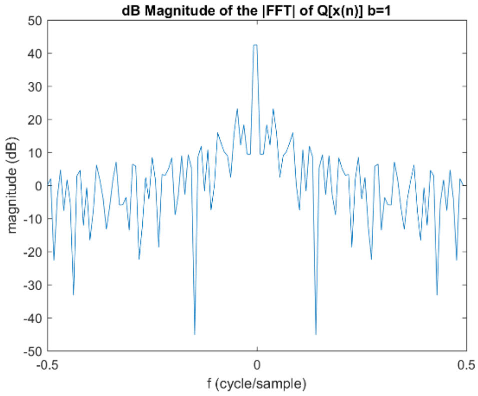

Fig. 7

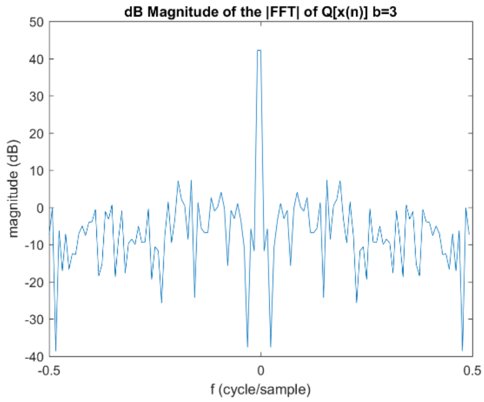

Fig. 8

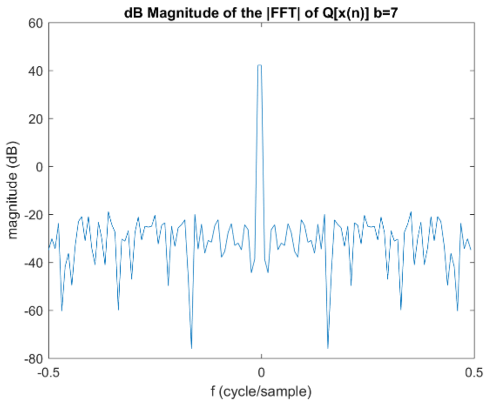

Fig. 9

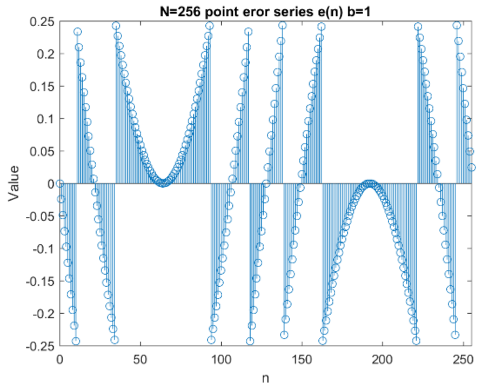

Fig. 10

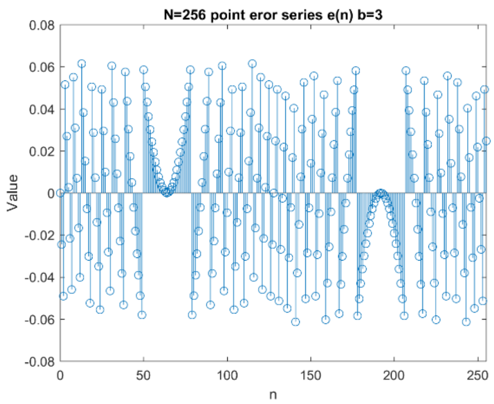

Fig. 11

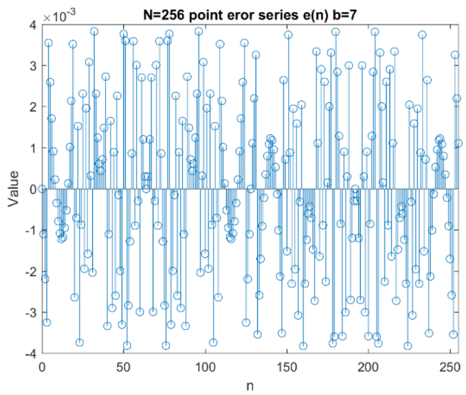

Fig. 12

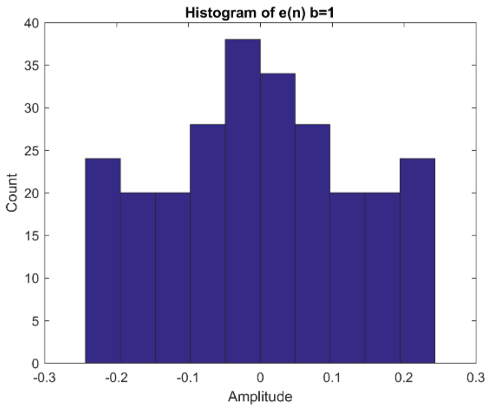

Fig. 13

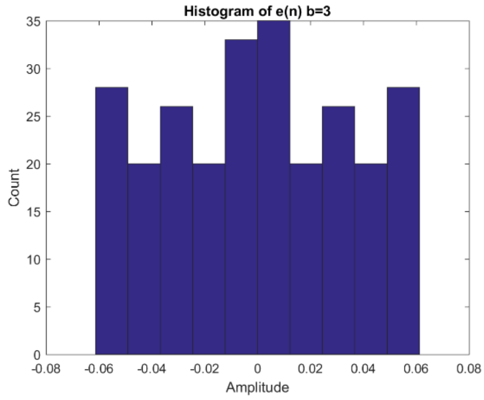

Fig. 14

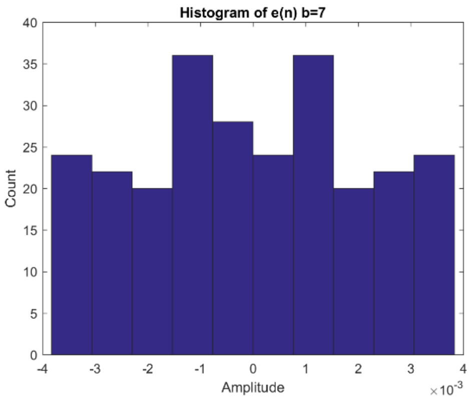

Fig. 15

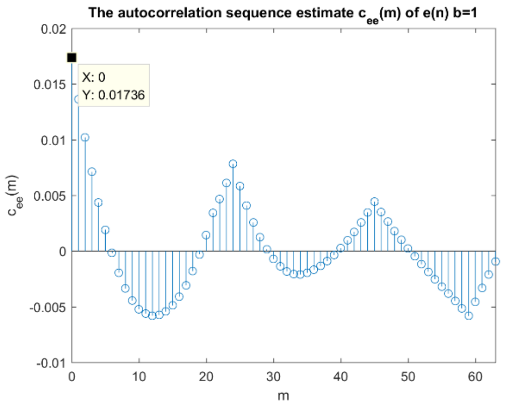

Fig. 16

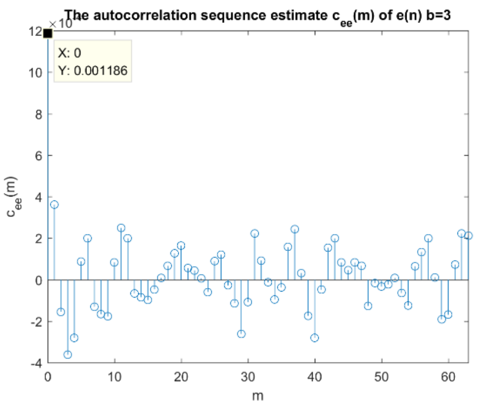

Fig. 17

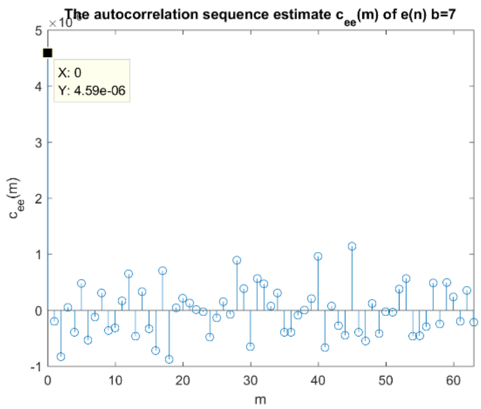

Fig. 18

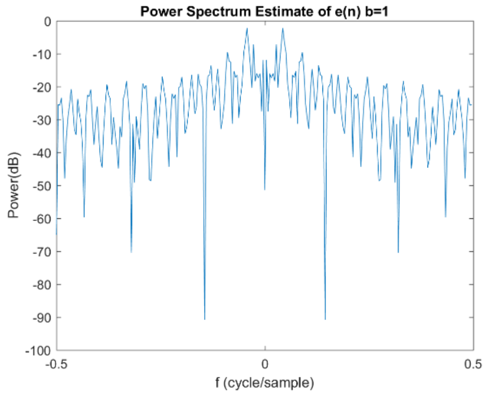

Fig. 19

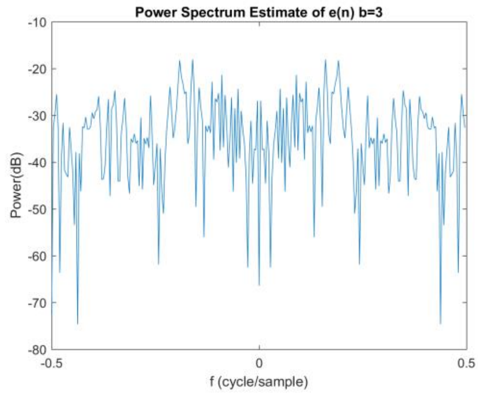

Fig. 20

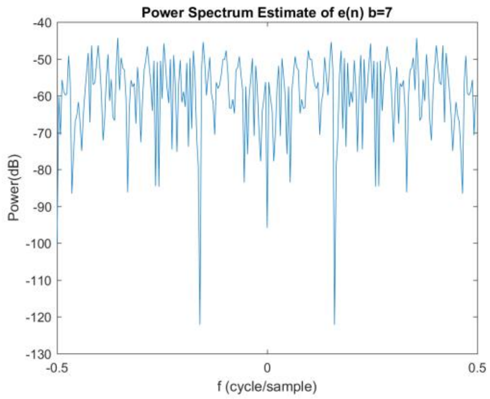

Fig. 21

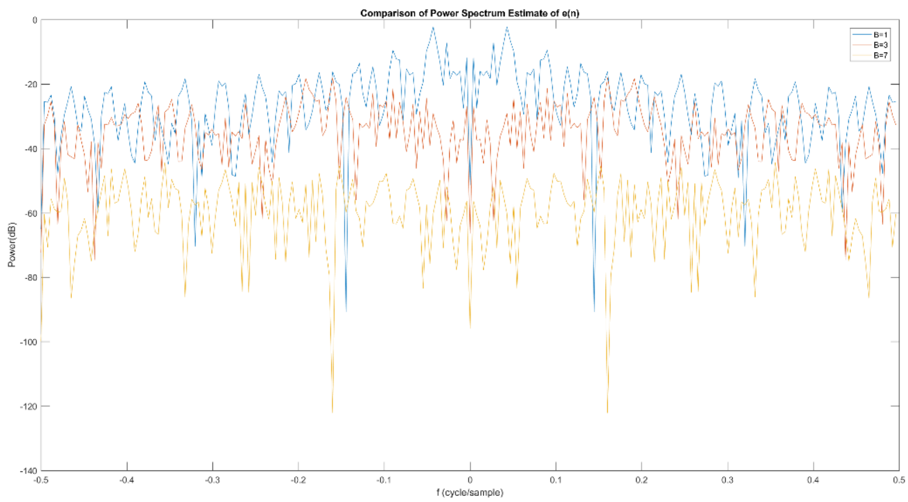

Fig. 22

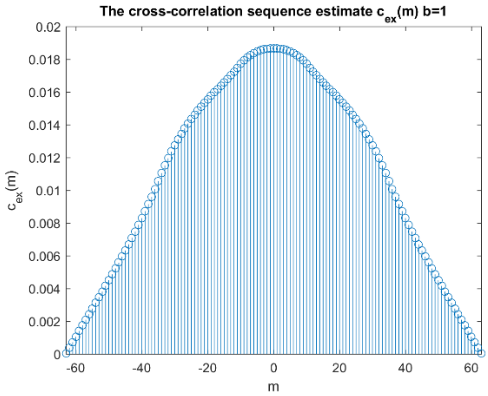

Fig. 23

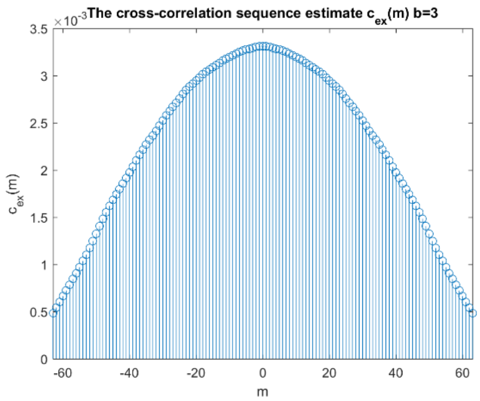

Fig. 24

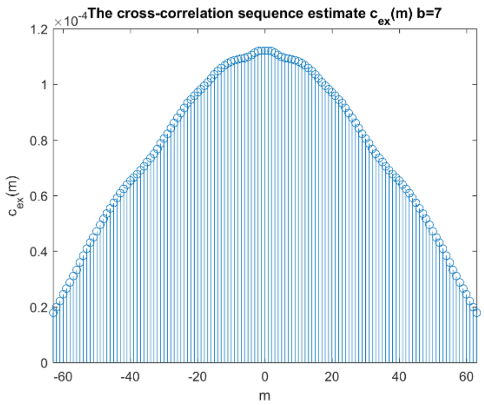

Fig. 25

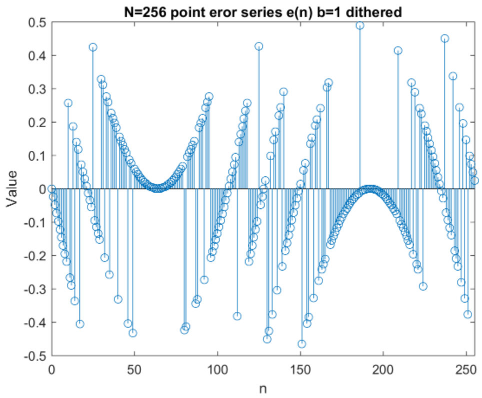

Fig. 26

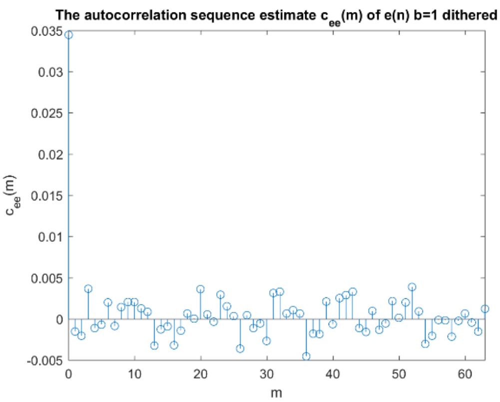

Fig. 27

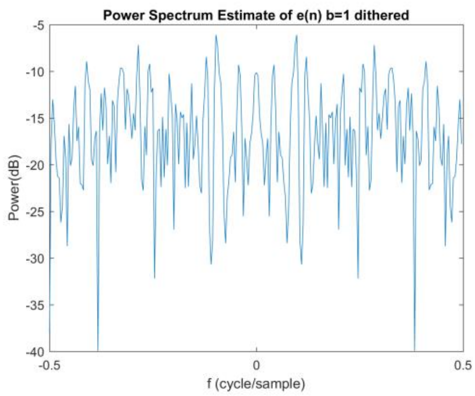

Fig. 28

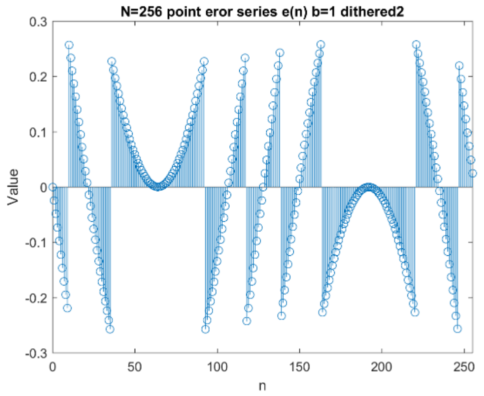

Fig. 29

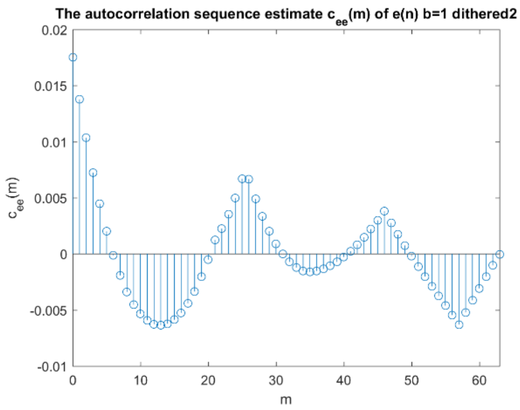

Fig. 30

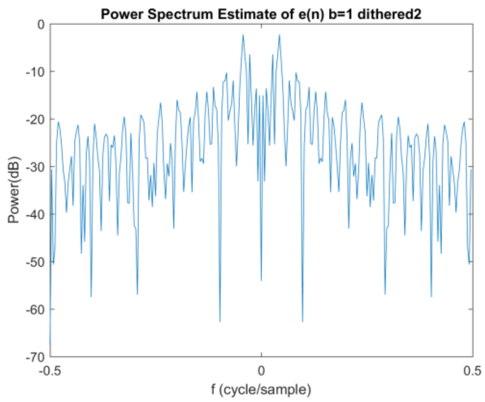

Fig. 31

- # Appendix

Script:

```
clear;clc;close all;
%% 1
N=256;
n=0:N-1;
xn=sin(2*pi/N*n);
% xx=xn;
figure(1);
stem(n,xn);
axis([0 length(xn)-1 -1.1 1.1]);
xlabel('n');ylabel('Value');title('N=256 point time
series x(n)');

b=1;            %%1,3,7
% noise=-2^(-b) + (2^(-b)+2^(-b)).*rand(1,N);
% xn=xn+noise;
```

```
xn_q=xn.*(2^b);
xn_q=round(xn_q);
xn_q=xn_q.*(2^(-b));
figure(2);
stem(n,xn_q);
axis([0 length(xn)-1 -1.1 1.1]);
xlabel('n');ylabel('Value');title(['N=256 point
quantized time series Q[x(n)] b=' num2str(b)]);

en=xn_q-xn;   %%change for dither
figure(3);
stem(n,en);
xlim([0 length(xn)-1]);
xlabel('n');ylabel('Value');title(['N=256 point eror
series e(n) b=' num2str(b)]);
%% 2
XN=fftshift(fft(xn,N));
figure(4);
plot([-0.5:1/N:0.5-1/N],20*log10(abs(XN)));
title('dB Magnitude of the |FFT| of x(n)');
xlabel('f (cycle/sample)');ylabel('magnitude (dB)');
axis([-0.5 0.5 -50 50]);

figure(5);
XN_Q=abs(fftshift(fft(xn_q,N)));
XN_Q=XN_Q(2:2:end);                         %%cut-off large
negative
plot([-0.5:2/N:0.5-2/N],20*log10(abs(XN_Q)));
title(['dB Magnitude of the |FFT| of Q[x(n)] b='
num2str(b)]);
xlabel('f (cycle/sample)');ylabel('magnitude (dB)');

%% 3
figure(6);
hist(en);
title(['Histogram of e(n) b=' num2str(b)]);
xlabel('Amplitude');ylabel('Count');

m=63;
figure(7);
e_auto=xcorr(en,m,'biased');
e_auto=e_auto(m+1:2*m+1);
stem(0:m,e_auto);
```

```
title(['The autocorrelation sequence estimate c_{ee}(m)
of e(n) b=' num2str(b)]);
xlabel('m');ylabel('c_{ee}(m)');
xlim([0 m]);

window=hamming(N)';
wn=en.*window;
U=sum(window.^2);
Pee=10*log10(abs(fftshift(fft(wn))).^2/U);
figure(8);
f=[-0.5:1/N:0.5-1/N];
plot(f,Pee);
xlabel('f (cycle/sample)');ylabel('Power(dB)');
title(['Power Spectrum Estimate of e(n) b='
num2str(b)]);

figure(9);
bex=xcorr(xn,en,m,'biased'); stem(-m:m,bex);
title(['The cross-correlation sequence estimate
c_{ex}(m) b=' num2str(b)]);
xlabel('m');ylabel('c_{ex}(m)');
xlim([-m m]);
```