**컴퓨터 통신 팀 프로젝트 보고서**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 과제명 | 팀원 1인의 voice를 녹음해 프로그램을 이용하여 voice 차이를 비교 | | | |
| 팀번호 | 7조 | | | |
| 지도교수 | 이종명 교수님 | | | |
| 작성자 | 이채민 | | | |
| 팀원 | 이 름 | 학 번 | 전화번호 | E-mail |
| 이채민 | 60175076 | 010-4907-3539 | hongdoo0215@naver.com |
| 이준성 | 60171745 | 010-3946-7307 | leejunsung0910@gmail.com |
| 이창민 | 60142013 | 010-2603-6035 | dia\_changmin@naver.com |
| 이진우 | 60142969 | 010-9813-1617 | anoxia\_@naver.com |
| 이승기 | 60151938 | 010-9359-6450 | tmdrlwkd519@naver.com |

**과제 최종보고서**

**1. 과제 개요**

팀원 1인의 voice를 녹음하여 sampling rate를 8,000, 4,000, 2,000, 1,000 samples/sec 로 프로그램 수행하여 voice 차이를 비교하고 quantization levels를 256, 128, 64, 32, 16으로 프로그램 수행하여 voice차이를 비교한다.

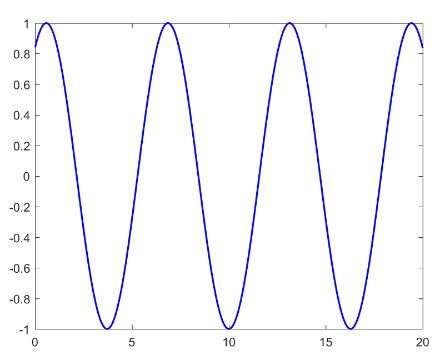
**2. 내용**

**2.1**

MATLAB을 이용하여 sampling rate와 quantization levels 를 다르게 설정하여 2초간의 음성파일의 voice 차이를 비교할 것이다.

**2.2 코드를 설정하기 위한 기초**

MATLAB을 사용하여 샘플링과 양자화를 하기 위해 코드를 작성한다. MATLAB에서의 주석은 % 또는 %{ … %} 기호를 사용하여 추가한다.

그래프에 필요한 x축과 y축의 범위 설정 및 로그 축을 설정한다. 예로 y = sin(x+1)이라고하고 0~ 20사이에 숫자 2000 개를 만든다고 가정하면 x=linspace(0, 20, 2000); \n y=sin(x+1); 를 사용한다. 각 축의 범위설정은 최소, 최대값을 벡터로 넣어주어야 한다. 따라서 x와 y축의 범위 설정할 때 사용해야할 함수는 xlim([0 20]), ylim([-1 1]) 이다.   
 위의 설명으로 이러한 그래프가 나왔다.

오디오 파일 읽기에 필요한 코드 [y, Fs] = audioread(filename) 는 filename이라는 파일에서 데이터를 읽고 샘플링된 데이터 y와 읽어드린 데이터의 샘플 레이트 Fs를 반환합니다.

오디오 파일 쓰기에 필요한 코드 Audiowrite(filename, y, Fs) 는 샘플 레이트가 Fs이고 오디오 데이터로 구성된 y를 filename 파일에 사용, filename의 입력값에 따라 출력 파일 형식이 지정된다. 출력 데이터형은 오디오 데이터 y의 출력 파일 형식과 데이터형에 따라 달라진다.

**2.3 샘플링 코드**

fs=8000;

t=(1:length(sample8000))/fs;

subplot(4,1,1)

stairs(t,sample8000),grid on;

legend('8000-Sample')

xlim([0.001 0.005])

ylim([-0.15 0.15])

xlabel('Time');

ylabel('amplitude');

for j = 1:length(sample8000)/2

sample4000(j)=sample8000(2\*j-1)

end

fs=4000;

t=(1:length(sample4000))/fs;

subplot(4,1,2)

stairs(t,sample4000),grid on;

legend('4000-Sample')

xlim([0.001 0.005])

ylim([-0.15 0.15])

xlabel('Time');

ylabel('amplitude');

for j = 1:length(sample8000)/4

sample2000(j)=sample8000(4\*j-3)

end

fs=2000;

t=(1:length(sample2000))/fs;

subplot(4,1,3)

stairs(t,sample2000),grid on;

legend('2000-Sample')

xlim([0.001 0.005])

ylim([-0.15 0.15])

xlabel('Time');

ylabel('amplitude');

for j = 1:length(sample8000)/8

sample1000(j)=sample8000(8\*j-7);

end

fs=1000;

t=(1:length(sample1000))/fs;

subplot(4,1,4)

stairs(t,sample1000),grid on;

legend('1000-Sample')

xlim([0.001 0.005])

ylim([-0.15 0.15])

xlabel('Time');

ylabel('amplitude');

audiowrite('sampling\_8000.wav',sample8000,8000)

audiowrite('sampling\_4000.wav',sample4000,4000)

audiowrite('sampling\_2000.wav',sample2000,2000)

audiowrite('sampling\_1000.wav',sample1000,1000)

**2.4 양자화코드 (주석은 %로 표시)**

[sample8000,fs] = audioread('SR\_8000.wav');

**% load한 SR\_8000.wav를 audioread를 이용하여 MATLAB으로 읽어 들였다.**

t=(1:length(sample8000))/fs;

**% 8비트 양자화**

N=8; **% N 비트 양자화**

M=2^N; **% N비트 = M = 2^N승, N=4, M=16**

Xmax = max(sample8000) **%신호의 최대치 ex)10**

stepsize = (2\*Xmax)/M; **% 1단계당 크기 ex)20/16 = 1.25**

for i = 1:M **% i는 1부터 M까지**

quantlevel(i) = -Xmax+stepsize\*(i-0.5); **%quantlevel(1)= -10+ 1.25\*0.5 = -9.375**

end

for j = 1:length(sample8000)

for i = 1:M

if(sample8000(j) >= quantlevel(i)-stepsize/2)&&(sample8000(j) <= quantlevel(i) + stepsize/2)

quant8X(j) = quantlevel(i); **%샘플의 데이터가 해당 레벨의 범위에 포함되는 경우 그 레벨로 양자화한다.**

end

end

end

**.**

**.**

**. %(7~5비트 까지의 양자화는 위와 동일하므로 생략)**

**.**

**.**

**% 4비트 양자화**

N=4;

M=2^N;

Xmax = max(sample8000)

stepsize = (2\*Xmax)/M;

for i = 1:M

quantlevel(i) = -Xmax+stepsize\*(i-0.5);

end

for j = 1:length(sample8000)

for i = 1:M

if(sample8000(j) >= quantlevel(i)-st

epsize/2)&&(sample8000(j) <= quantlevel(i) + stepsize/2)

quant1X(j) = quantlevel(i); **%샘플의 데이터가 해당 레벨의 범위에 포함되는경우 그레벨로 양자화함**

end

end

end

subplot(5,1,1)

stairs(t,quant8X),grid on;

legend('8-bit')

xlim([0.001 0.005])

ylim([-0.15 0.15])

xlabel('Time');

ylabel('amplitude');

subplot(5,1,2)

stairs(t,quant7X),grid on;

legend('7-bit')

xlim([0.001 0.005])

ylim([-0.15 0.15])

xlabel('Time');

ylabel('amplitude');

subplot(5,1,3)

stairs(t,quant6X),grid on;

legend('6-bit')

xlim([0.001 0.005])

ylim([-0.15 0.15])

xlabel('Time');

ylabel('amplitude');

subplot(5,1,4)

stairs(t,quant5X),grid on;

legend('5-bit')

xlim([0.001 0.005])

ylim([-0.15 0.15])

xlabel('Time');

ylabel('amplitude');

subplot(5,1,5)

stairs(t,quant4X),grid on;

legend('4-bit')

xlim([0.001 0.005])

ylim([-0.15 0.15])

xlabel('Time');

ylabel('amplitude');

audiowrite('8bits\_quantization.wav',quant8X,8000)

audiowrite('7bits\_quantization.wav',quant7X,8000)

audiowrite('6bits\_quantization.wav',quant6X,8000)

audiowrite('5bits\_quantization.wav',quant5X,8000)

audiowrite('4bits\_quantization.wav',quant4X,8000)

**2.4 sampling 과 Quantization**

Sampling(채집)이란 1초당 추출되는 샘플 개수이며 아날로그 신호로 표본을 채집하여 그 결과에 근거해 펄스를 제작한다. 과제에서의 Sampling rate는 이산적인 신호를 만들기 위해 연속적 신호에서 얻어진 단위시간당 샘플링 횟수를 정의한 것이다. 단위는 Hz로 나타낸다. 일정 간격마다 신호의 진폭을 측정하며 디지털로 변환하기 위해서 펄스 코드 변조를 사용한다. Nyquist 이론으로 채집율은 신호에 포함된 최대 주파수의 최소한 두 배가 되어야 한다. Voice는 1초에 8000개의 샘플을 취하며 최대주파수의 2배의 sample이면 원음 복구가 가능하다. y축이 좁을수록 에러가 줄어든다.

Sampling 한 아날로그 형태로 되어있는 신호나 정보를 디지털화 하는 작업으로 셀 수 없는 아날로그 정보를 셀 수 있을 만큼의 간격으로 만들어 유의미한 정보만 사용하는 것이다. Quantization(양자화)는 채집된 값에 특정 범위에 속하는 정수 값을 할당하는 방법이다. 양자화 시 원래의 아날로그 신호의 진폭값과 근사값 사이에서 오차가 발생하고 복호화 시에 잡음이나 왜곡과 같은 효과를 주어서 양자잡음이라 하는데 이 양자잡음이 품질 저하의 요인이 된다. 양자화 스텝의 사이즈가 작을수록 잡음을 줄어든다. 하지만 양자화 하는데 필요한 스텝 수가 증가하여 부호화를 위한 비트가 증가하게 되어 대역폭의 증가를 불러오게 된다.

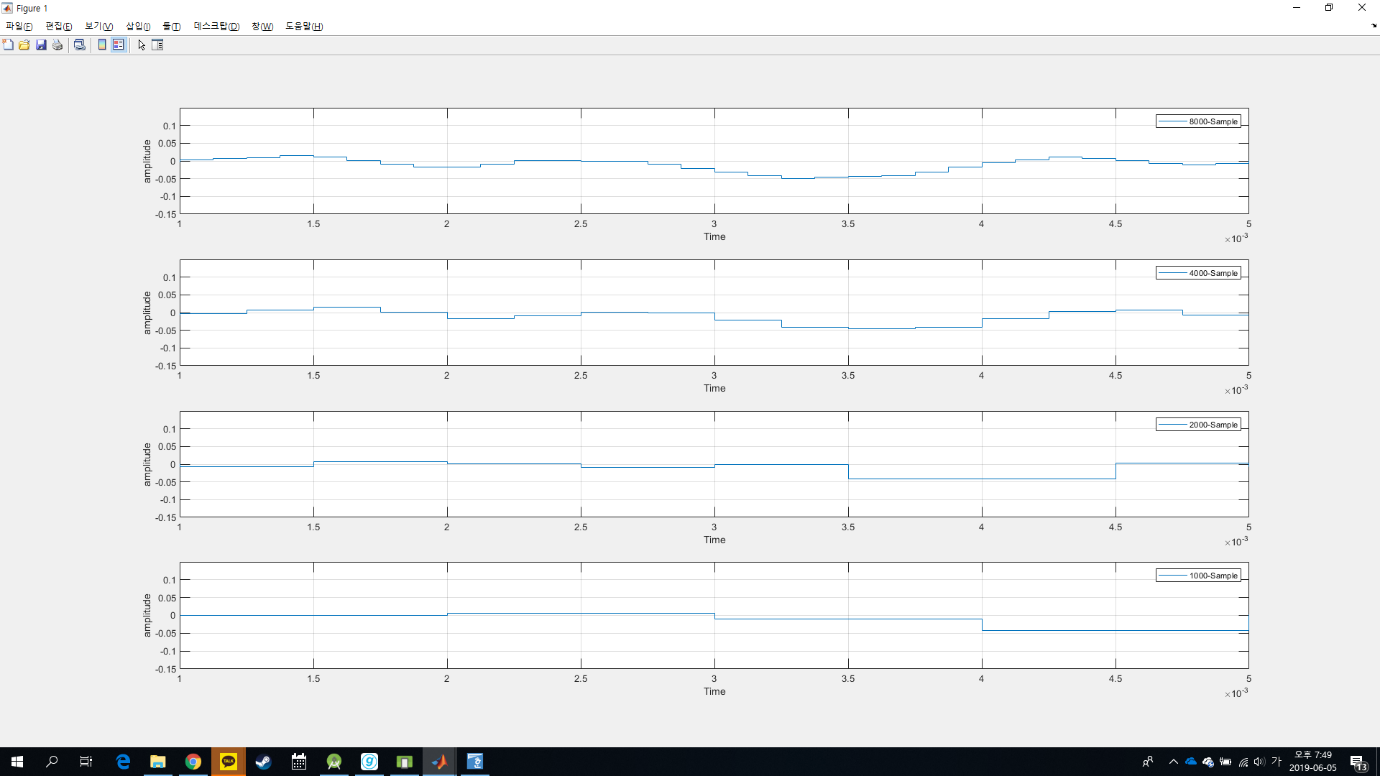
양자화 잡음의 개선방안은 특정 입력 신호 레벨에서 스텝 간격을 조밀하게 하여 양자화잡음을 줄이는 비선형 양자화를 활용하거나 양자화 스텝 수를 증가시켜 잡음을 최소화한다. 밑의 Quantization 결과물을 보면 4bit는 8bit에 비해 잡음이 크게 들리는 것을 확인할 수 있다. 따라서 비트 수를 증가시키면 잡음 개선을 확인할 수 있다. 또한 압신기(압축기와 신장기로 구성됨, 작은 진폭은 크게 증폭시키거나 큰 진폭은 적게 증폭시켜 두 진폭의 차를 줄임)를 활용하면 비선형 양자화기만큼 잡음을 줄일 수 있는데 양자화 계단을 줄이지 않고도 SNR을 향상시킬 수 있다.

.

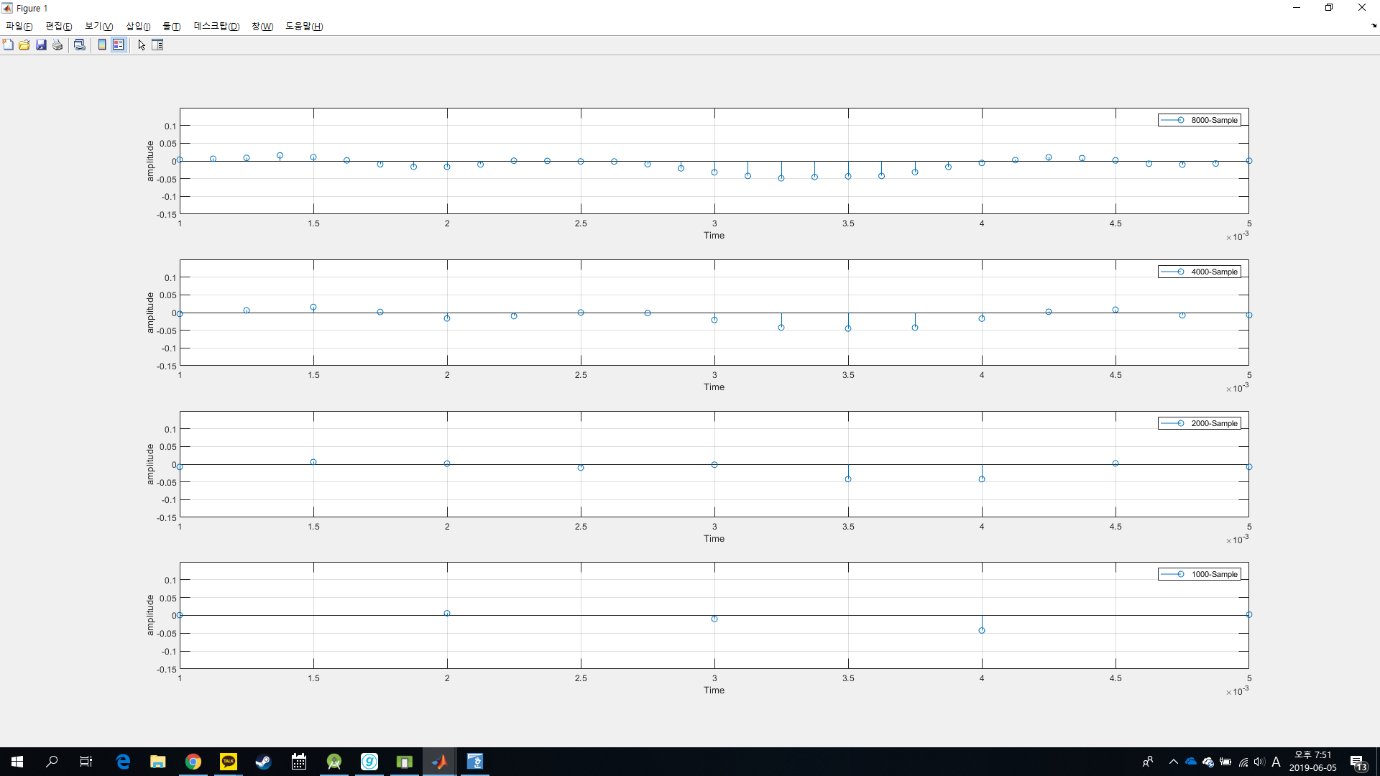
**2.5 결과물**

sampling rate 과 quantization levels 의 파형이다.

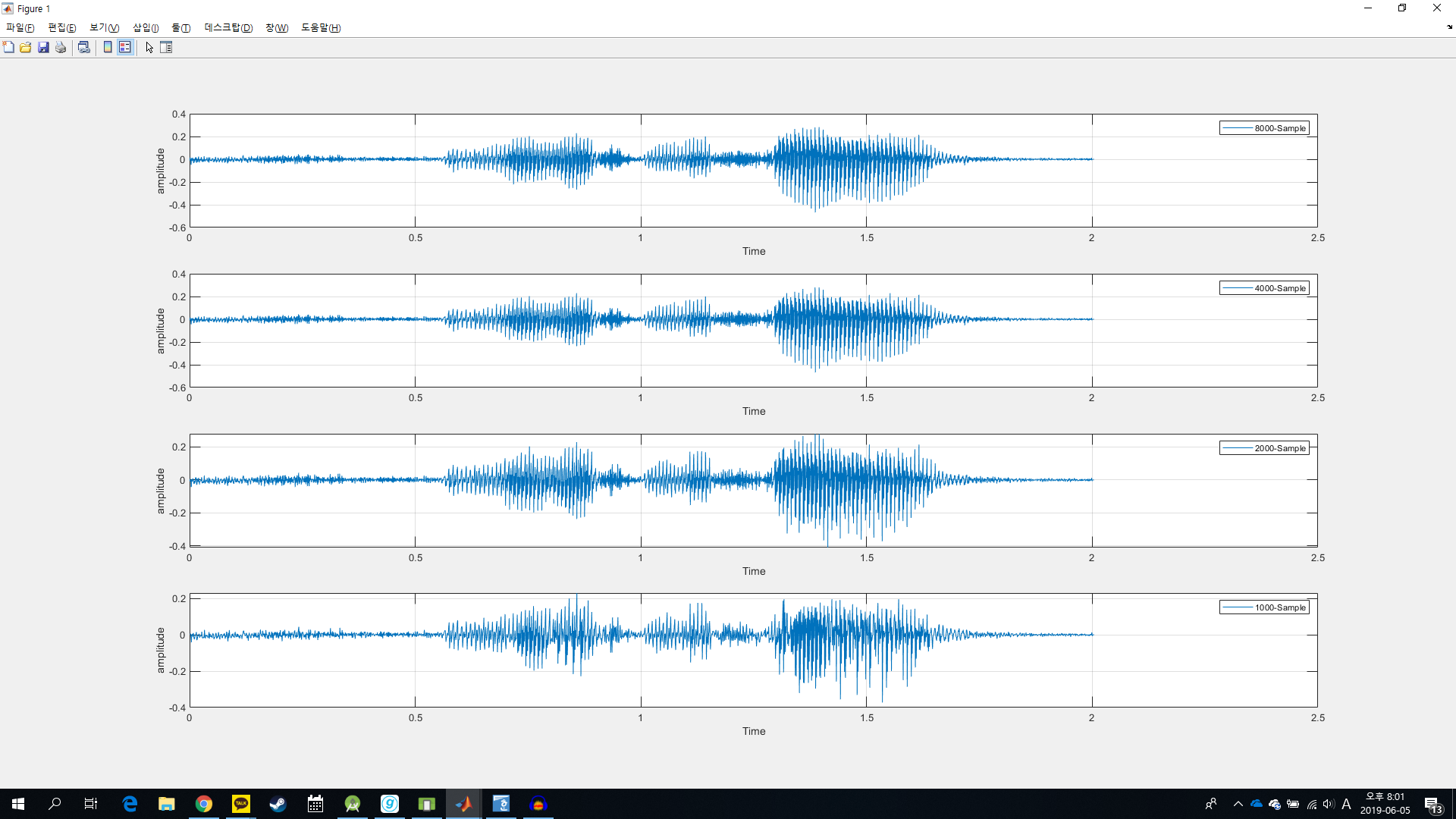
전체 파형으로는 sampling이 잘 안보여서 1ms~5ms까지만 나타내었다.



Sampling (계단식)



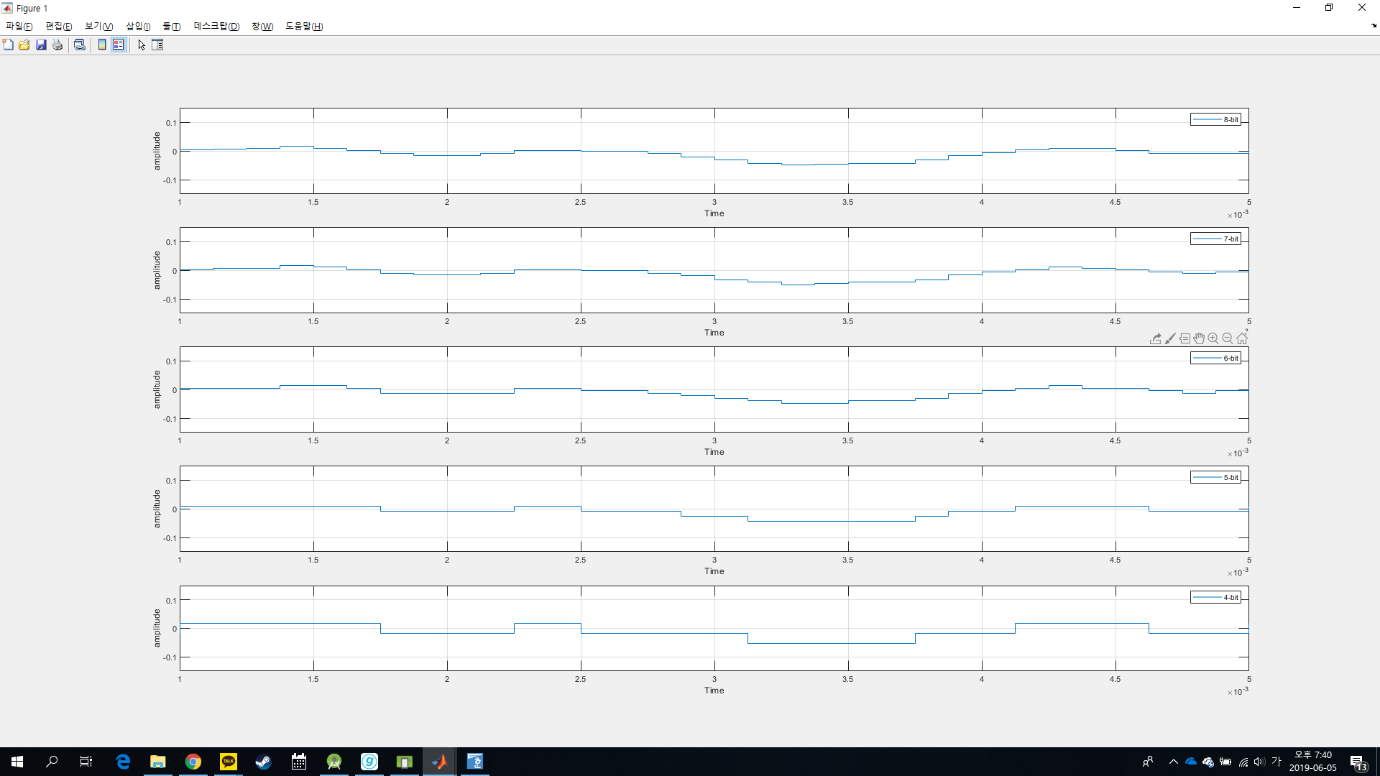
sampling (막대식)



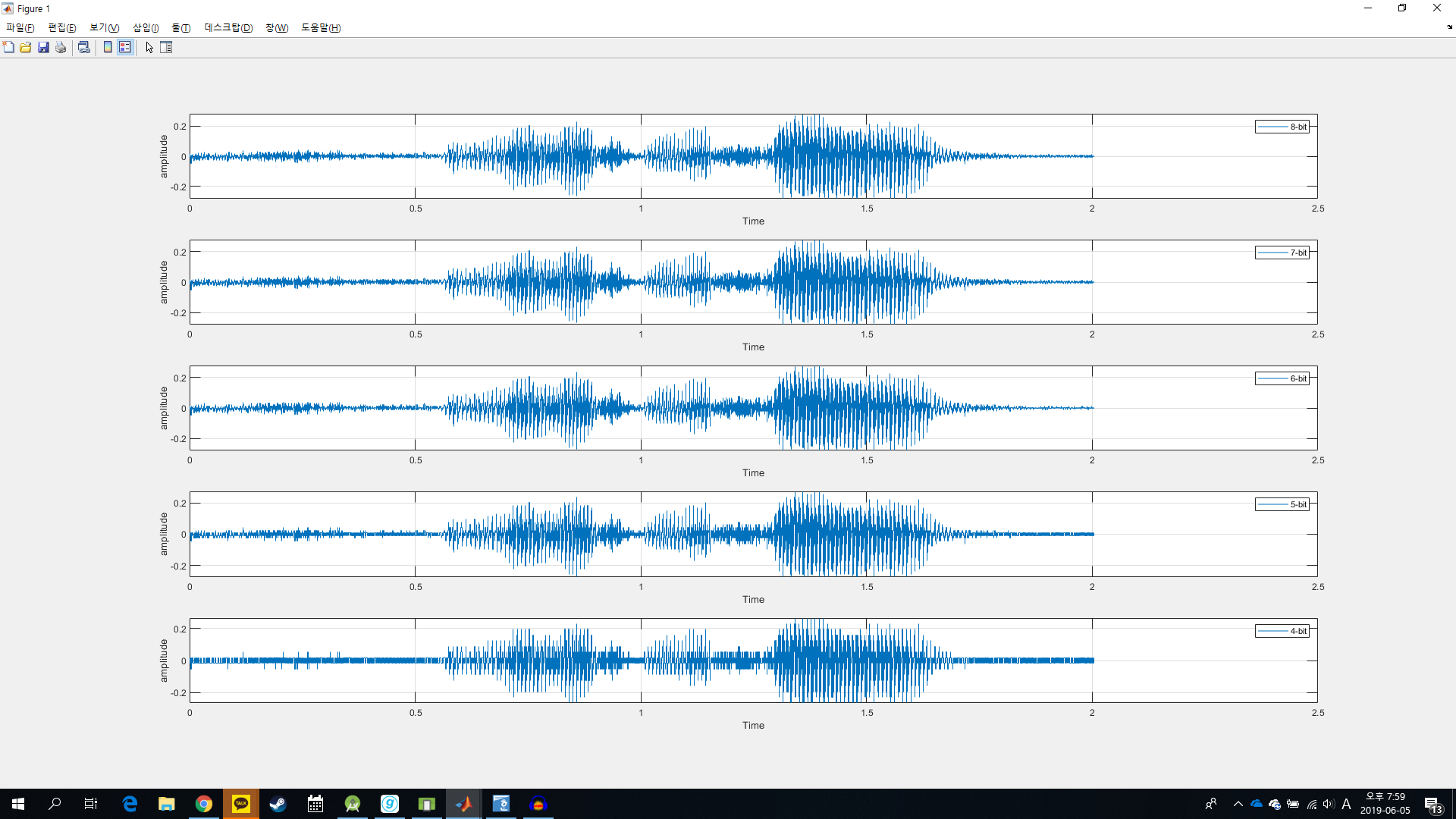
sampling (전체파형)

sampling rate를 8,000 samples/sec 로 하였을 때하고 1,000 samples/sec하였을 때의 음성을 들어보면 음질의 선명도에 변화가 크다. 1,000 samples/sec의 파형을 보면 y축이 넓어진 것을 볼 수 있다. 숫자가 낮아질수록 음질의 선명도는 급격히 떨어지며 소리가 잘 들리지 않는다.

--------------------------------------------------------------------------------------------------------------



quantization levels (계단식)



quantization levels (전체파형)

quantization은 4bit는 8bit에 비해 주변 잡음이 심하게 들린다. 계단식을 보면 4bit의 간격이 8bit에 비해 크다는 것을 알 수 있다.

**2.6 참고문헌**

그래프, 축 범위설정 - https://1312ryu.tistory.com/32

매트랩 audioread - https://kr.mathworks.com/help/matlab/ref/audioread.html

매트랩 audiowrite - https://kr.mathworks.com/help/matlab/ref/audiowrite.html

매트랩 주석 - <https://kr.mathworks.com/help/matlab/matlab_prog/comments.html>

샘플링과 양자화 개념 - <https://twlab.tistory.com/19>

구글링

컴퓨터통신 강의 자료

수업시간 때의 내용

**2.7 소감문**