Signals and Systems Laboratory (EC2P002)

EXPERIMENT-6

Shorya Sharma
19EE01017

Aim of the experiment:

- To sample a signal at Nyquist Rate.
- To observe the effects of under sampling (aliasing).
- To perform a discrete Fourier transform (DFT) of a given signal.
- To obtain the DFT of your own voice signal.

Theory:

- **Sampling Theorem:**
  Sampling is a process of converting a signal (for example, a function of continuous-time and/or space) into a sequence of values (a function of discrete-time and/or space). Shannon's version of the theorem states:
  If a function x(t) contains no frequencies higher than B hertz, it is completely determined by giving its ordinates at a series of points spaced (1/2B) seconds apart. A sufficient sample-rate is therefore anything larger than 2B samples per second. Equivalently, for a given sample rate fs, perfect reconstruction is guaranteed possible for a bandwidth B<2fs.

- **Discrete Fourier Transform (DFT)**
  The Discrete Fourier Transform (DFT) is the equivalent of the continuous Fourier Transform for signals known only at instants separated by sample times T (i.e. a finite sequence of data). Consider f[k] be a sequence of N sampled instances, the DFT is defined as:

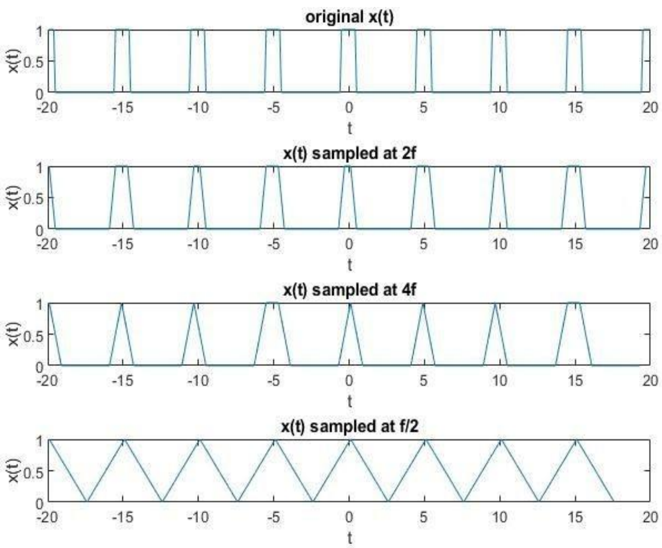
$$F[n] = \sum_{k=0}^{N-1} f[k]\, e^{\frac{2}{N}nk} \quad n=0,\ldots\ldots,N-1$$

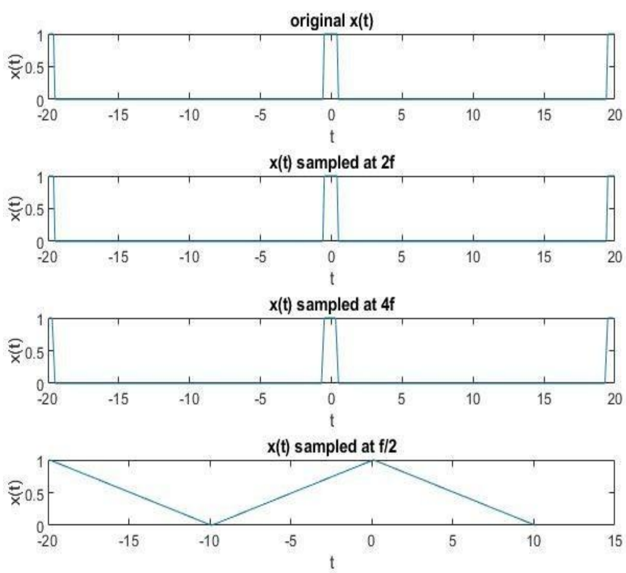
Results

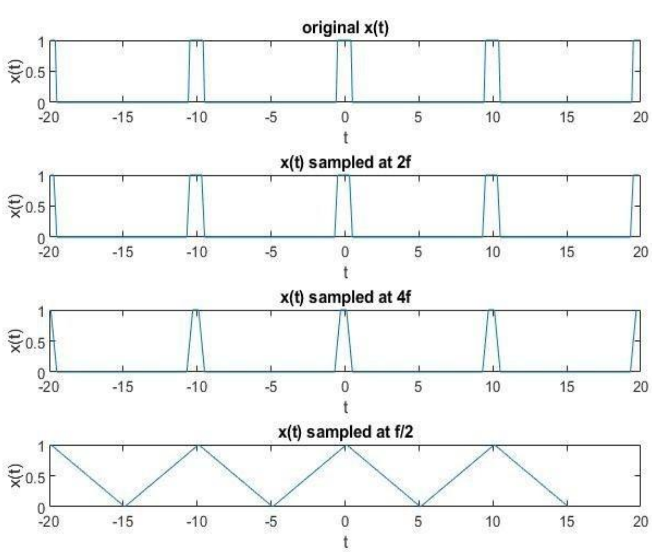
Question 1:

Sample the following signals at 2f, f/2, 4f (where f is the frequency of the signal).
(a)T1=0.5 ms, T=5 ms
(b) T1=0.5 ms, T=10ms
(c) T1=0.5 ms, T=20 ms

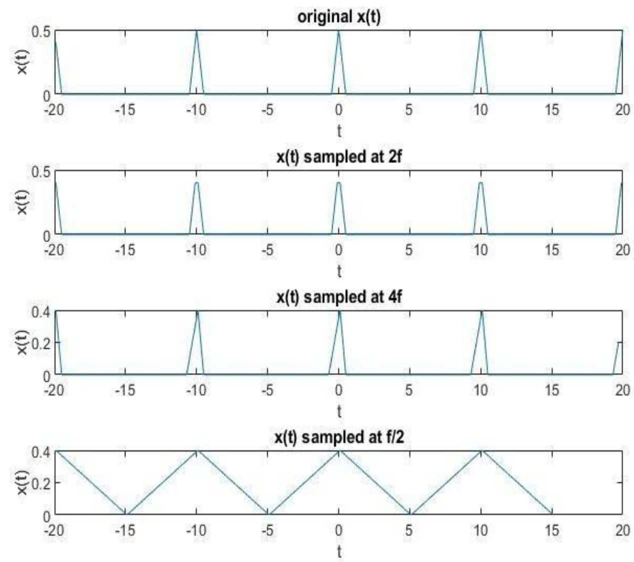
Rectangular Signal T=5,10,20

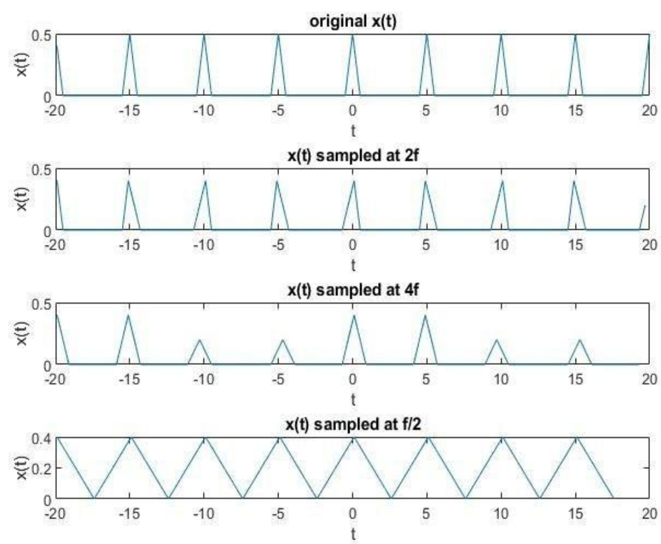


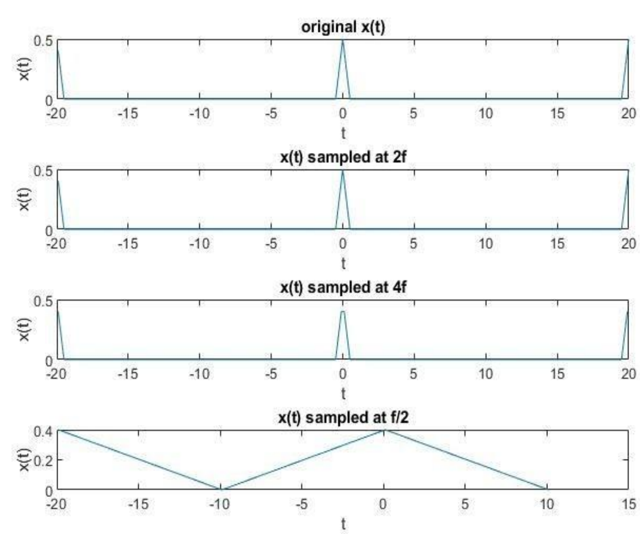




Triangular Signal T=5,10,20

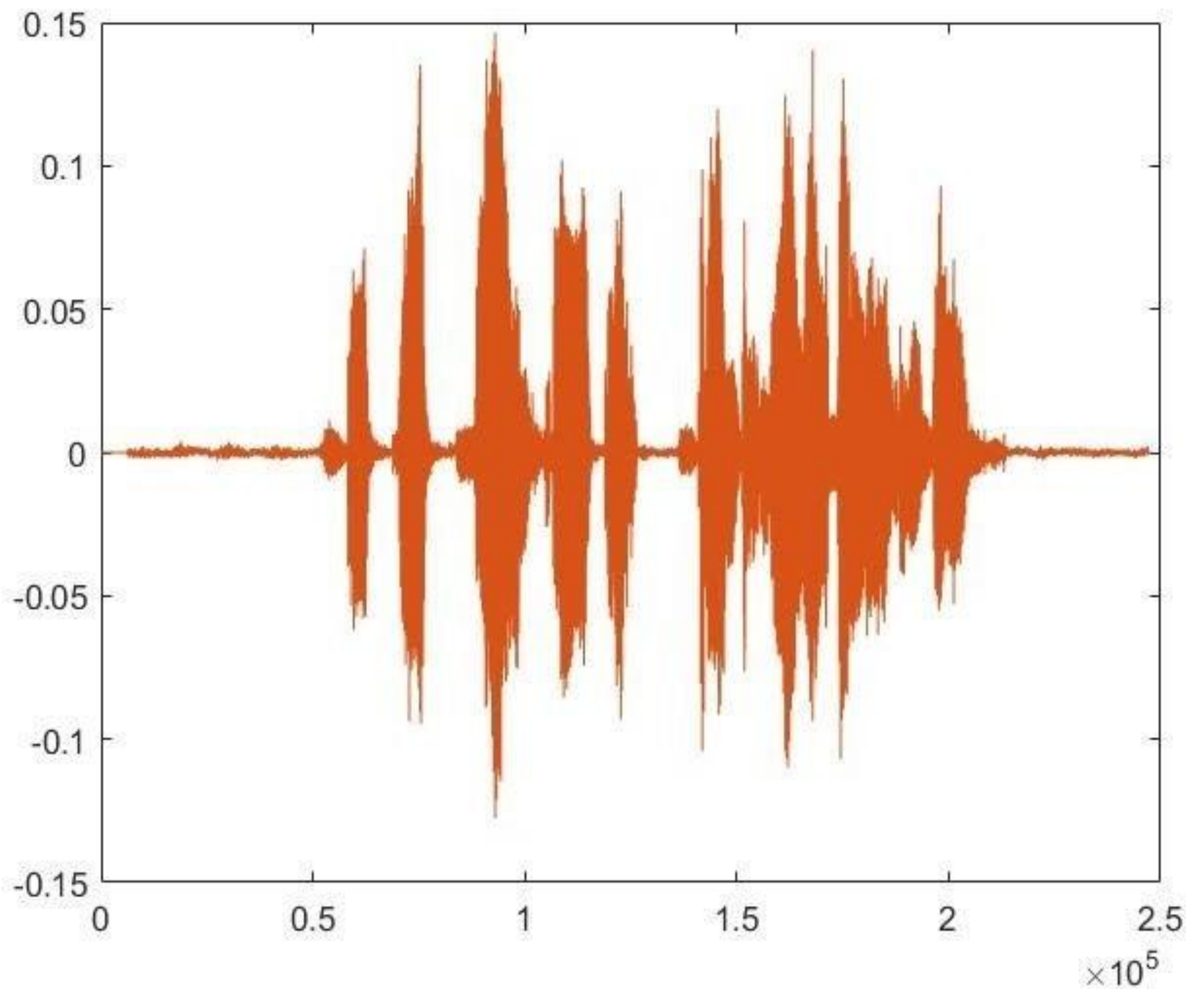






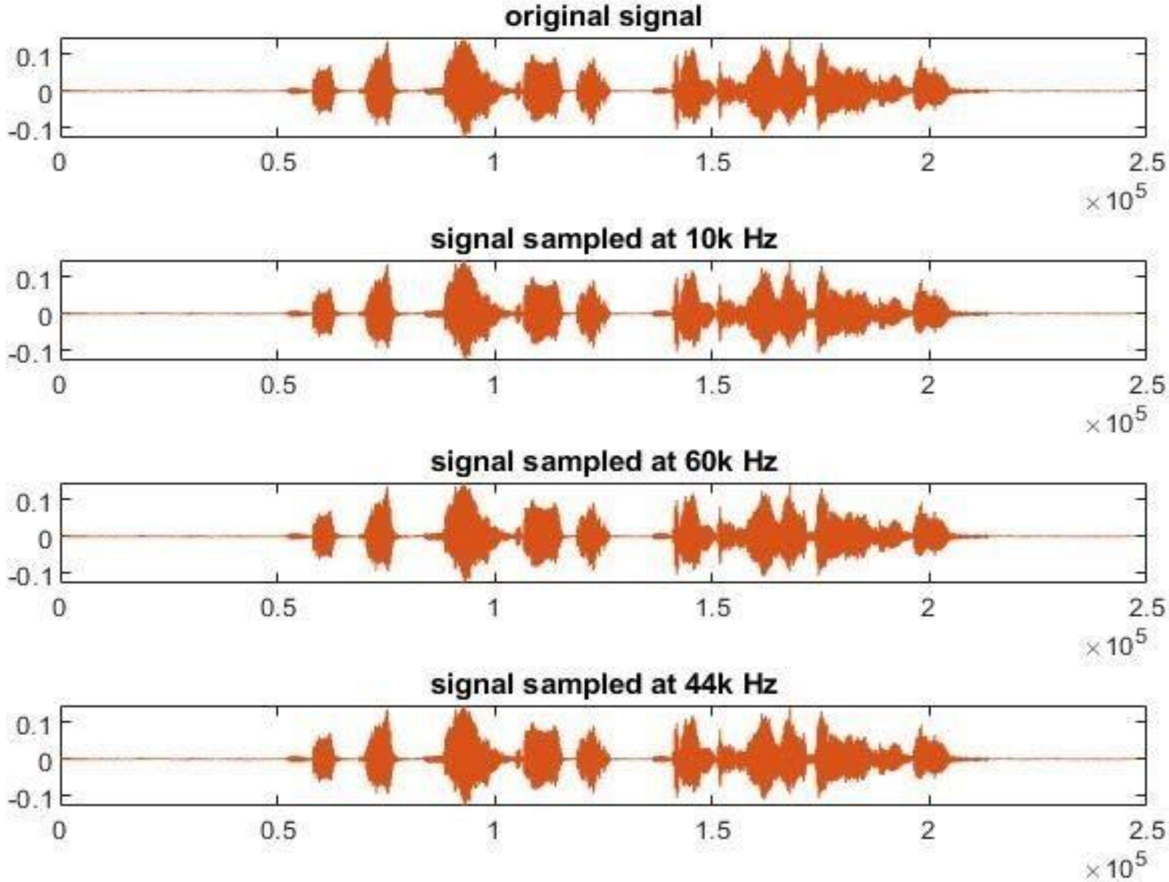
Question 2

Create your own audio file that is at least 4 seconds long and playback the recorded audio. The recorded audio should be stereo. Plot the recorded signal.
Read the following line while recording your audio:
"**THE QUICK BROWN FOX JUMPS OVER THE LAZY DOG"**



Question 3

Sample the recorded signal at 10Hz, 60 kHz, and 44 kHz and playback the sampled signal. Plot and report your observation.

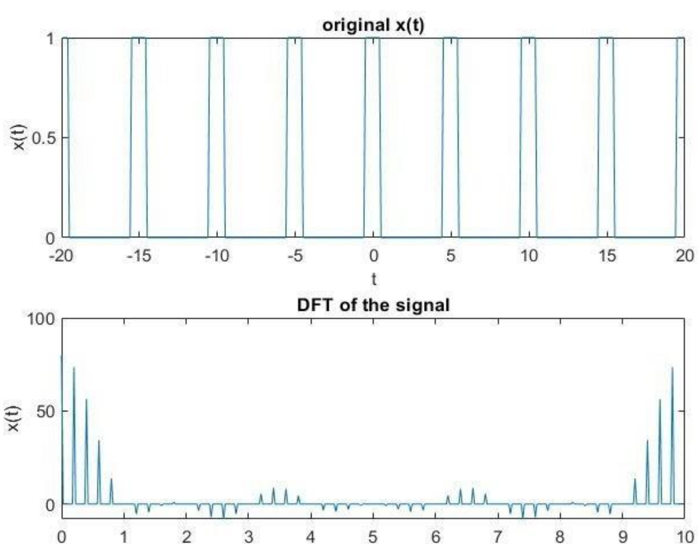


Question 4:

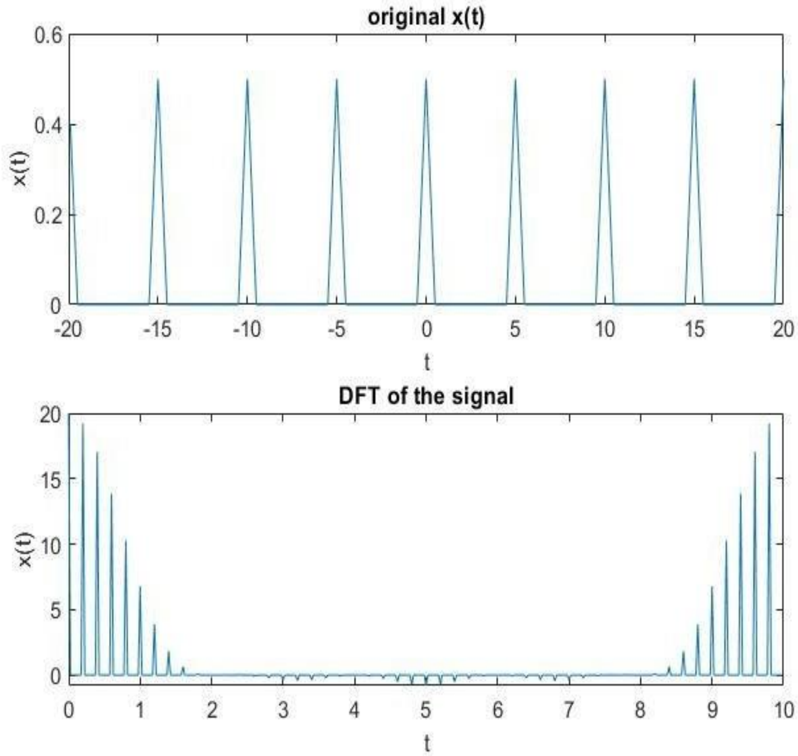
a) Calculate and plot the DFT of the signals in figures 1 and 2.
b) Calculate and plot the DFT of your audio signals.
c) Calculate and plot the DFT of your sampled audio signals.

(A)

Rectangular Wave:

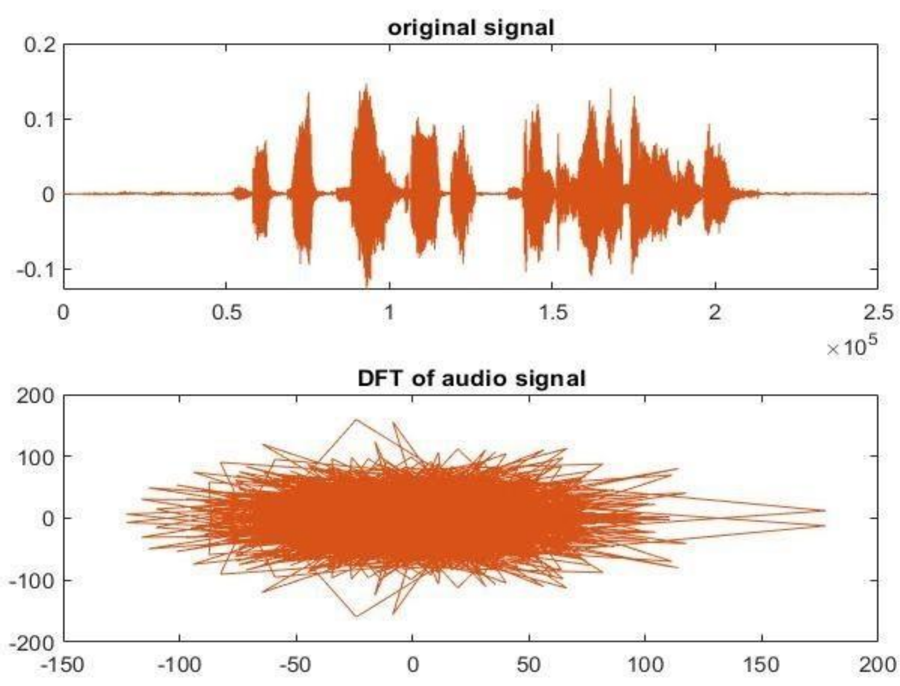


Triangular Wave:

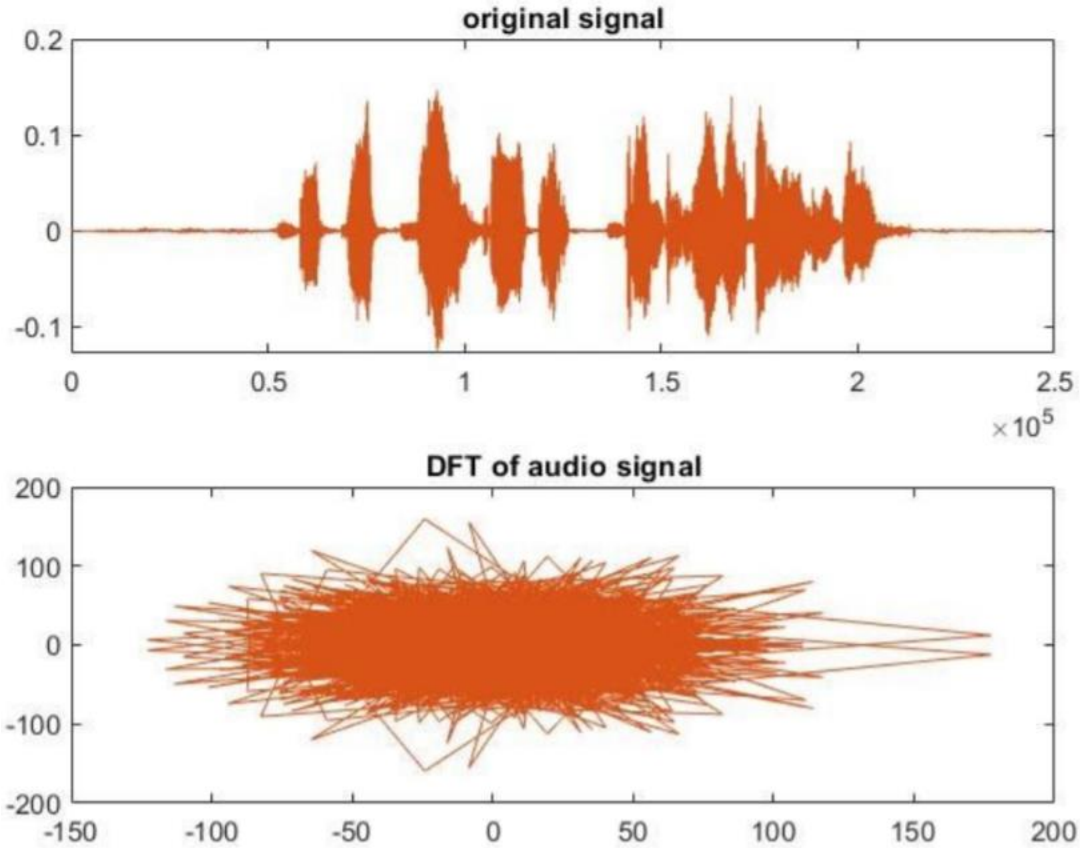


DFT calculated only for T=5 for both rectangular and triangular signals. similarly, the results for T=10,20 can be obtained from the same MATLAB code.

(B) Audio Signal



(C) Sampled Audio Signal



DFT calculated only for the original signal and signal sampled at 2fs. similarly, the results for Different sampling rates can be obtained from the same MATLAB code.

DISCUSSION

It was really interesting to see how MATLAB performs simulations and the graphical data that was obtained was really helpful to understand the concept of Sampling on a signal more often MATLAB provides us with a large number of tools to perform such simulations.

Applications of sampling:

- To maintain sound quality in music recordings.
- Sampling process applicable in the conversion of analog to discrete form.
- Speech recognition systems and pattern recognition systems.
- Modulation and demodulation systems
- In sensor data evaluation systems
- Digital watermarking and biometric identification systems,
- surveillance systems.

CONCLUSION

By this experiment, we have successfully written the algorithm for sampling the given signals at Nyquist rate and also above and below it i.e. over and under sampling. Also, we have performed DFT on the signals by converting them into discrete signals. Above operations are also done on the recorded audio signal.

APPENDIX

Question-1

Part-A: -

```
clc
close all
T1 = 0.5;
%T = 5,10,20
T = 5;
t = -19.9:0.1:20;
x = zeros(1,length(t));
%initial signal
for i=1:length(t)
 n = round(t(i)/T);
 if(t(i)>=(n*T-T1) && t(i)<(n*T+T1))
 x(i)= 1;
 end
end
%signal sampled at 2f
t1 = -19.9:2/T:20;
x1 = zeros(1,length(t1));
for i=1:length(t1)
 n = round(t1(i)/T);
 if(t1(i)>=(n*T-T1) && t1(i)<(n*T+T1))
 x1(i)= 1;
 end
end
%signla sampled at 4f
t2 = -19.9:4/T:20;
x2 = zeros(1,length(t2));
for i=1:length(t2)
 n = round(t2(i)/T);
 if(t2(i)>=(n*T-T1) && t2(i)<(n*T+T1))
 x2(i)= 1;
 end
end

%signal sampled at f/2
t3 = -19.9:1/2*T:20;
x3 = zeros(1,length(t3));
for i=1:length(t3)
 n = round(t3(i)/T);
 if(t3(i)>=(n*T-T1) && t3(i)<(n*T+T1))
 x3(i)= 1;
 end
end
subplot(411)
plot(t, x);
title('original x(t)');
xlabel('t');
ylabel('x(t)');
subplot(412)
plot(t1, x1);
title('x(t) sampled at 2f');
xlabel('t');
ylabel('x(t)');
subplot(413)
plot(t2, x2);
title('x(t) sampled at 4f');
xlabel('t');
ylabel('x(t)');
subplot(414)
plot(t3, x3);
title('x(t) sampled at f/2');
xlabel('t');
ylabel('x(t)');
```

Part-B: -

```
clc
close all
T1 = 0.5;
%T = 5,10,20
T = 5;
t = -19.9:0.1:20;
x = zeros(1,length(t));
%initial signal
n = 0;
for i=1:length(t)
 n = round(t(i)/T);
 if(t(i)>=(n*T-T1) && t(i)<(n*T))
 x(i)= t(i)-(n*T-T1);
 end
 if(t(i)>=(n*T) && t(i)<(n*T+T1))
 x(i)= (n*T+T1)-t(i);
 end
end
%signal sampled at 2f
t1 = -19.9:2/T:20;
x1 = zeros(1,length(t1));
n = 0;
for i=1:length(t1)
 n = round(t1(i)/T);
 if(t1(i)>=(n*T-T1) && t1(i)<(n*T))
 x1(i)= t1(i)-(n*T-T1);
 end
 if(t1(i)>=(n*T) && t1(i)<(n*T+T1))
 x1(i)= (n*T+T1)-t1(i);
 end
end
%signla sampled at 4f
t2 = -19.9:4/T:20;
x2 = zeros(1,length(t2));
n = 0;
for i=1:length(t2)
 n = round(t2(i)/T);

 if(t2(i)>=(n*T-T1) && t2(i)<(n*T))
 x2(i)= t2(i)-(n*T-T1);
 end
 if(t2(i)>=(n*T) && t2(i)<(n*T+T1))
 x2(i)= (n*T+T1)-t2(i);
 end
end
%signal sampled at f/2
t3 = -19.9:1/2*T:20;
x3 = zeros(1,length(t3));
n = 0;
for i=1:length(t3)
 n = round(t3(i)/T);
 if(t3(i)>=(n*T-T1) && t3(i)<(n*T))
 x3(i)= t3(i)-(n*T-T1);
 end
 if(t3(i)>=(n*T) && t3(i)<(n*T+T1))
 x3(i)= (n*T+T1)-t3(i);
 end
end
subplot(411)
plot(t, x);
title('original x(t)');
xlabel('t');
ylabel('x(t)');
subplot(412)
plot(t1, x1);
title('x(t) sampled at 2f');
xlabel('t');
ylabel('x(t)');
subplot(413)
plot(t2, x2);
title('x(t) sampled at 4f');
xlabel('t');
ylabel('x(t)');
subplot(414)

plot(t3, x3);
title('x(t) sampled at f/2');
xlabel('t');
ylabel('x(t)');
```

Question-2

```
clc
clear all
close all
%reading audio file
[y,fs] = audioread('SS lab 6
voice.m4A');
%playing the original audio
sound(y,fs);
plot(y);
%change the sampling rate
fs2= 2*fs;
pause(5);
audiowrite('SSlab6_1.wav',y,
fs2)
%Read the data back into
MATLAB using audioread.
[y,fs2] =
audioread('SSlab6_1.wav');
sound(y,fs2);
```

Question-3

```
clc
clear all
close all
%reading audio file
[x,fs] = audioread('SS lab 6
voice.m4A');
%playing the original audio
sound(x,fs);
subplot(411);
plot(x);
title('original signal');
%change the sampling rate
fs2= 10000;
pause(5);
audiowrite('SSlab6_1.wav',x,
fs2)
%Read the data back into
MATLAB using audioread.
[x,fs2] =
audioread('SSlab6_1.wav');
sound(x,fs2);
subplot(412);
plot(x);
title('signal sampled at 10k
Hz');
fs3= 60000;
pause(20);
audiowrite('SSlab6_1.wav',x,
fs3)
%Read the data back into
MATLAB using audioread.
[x,fs2] =
audioread('SSlab6_1.wav');
sound(x,fs2);
subplot(413);
plot(x);
title('signal sampled at 60k
Hz');
fs2= 44000;
pause(5);
audiowrite('SSlab6_1.wav',x,
fs2)
%Read the data back into
MATLAB using audioread.
[x,fs2] =
audioread('SSlab6_1.wav');
sound(x,fs2);
subplot(414);
plot(x);
title('signal sampled at 44k
Hz');
```

Question-4

```
clc
clear all
close all
T1 = 0.5;
%T = 5,10,20
T = 5;
t = -19.9:0.1:20;
f = (0:length(t)-
1)*10/length(t);
x =
zeros(1,length(t));
%initial signal
for i=1:length(t)
 n = round(t(i)/T);
 if(t(i)>=(n*T-T1) &&
t(i)<(n*T+T1))
 x(i)= 1;
 end
end
figure(1);
subplot(211)
plot(t, x);
title('original
x(t)');
xlabel('t');
ylabel('x(t)');
subplot(212)
plot(f,fft(x));
xlabel('t');
ylabel('x(t)');
title('DFT of the
signal');
%given signal 2
x1 =
zeros(1,length(t));
n = 0;
for i=1:length(t)
 n = round(t(i)/T);
 if(t(i)>=(n*T-T1) &&
t(i)<(n*T))
 x1(i)= t(i)-(n*T-T1);
 end
 if(t(i)>=(n*T) &&
t(i)<(n*T+T1))
 x1(i)= (n*T+T1)-t(i);
 end
end
figure(2);
subplot(211)
plot(t, x1);
title('original
x(t)');
xlabel('t');
ylabel('x(t)');
subplot(212)
plot(f,fft(x1));
xlabel('t');
ylabel('x(t)');
title('DFT of the
signal');
%dft of audio signal
[x2] = audioread('SS
lab 6 voice.m4A');
%playing the original
audio
figure(3);
subplot(211);
plot(x2);
title('original
signal');
subplot(212);
plot(fft(x2));
title('DFT of audio
signal')
%dft of sampled audio
signal
[x3] =
audioread('SSlab6_1.wa
v');
%playing the original
audio
figure(4);
subplot(211);
plot(x3);
title('original
signal');
subplot(212);
plot(fft(x3));
title('DFT of audio
signal')
```