Electrical Engineering 3TP4
Lab 3
Aliasing in Signal Sampling

Introduction: In this laboratory we investigate the issue of aliasing when analog signals are sampled. We consider the effects of aliasing that would occur if proper filtering is not performed prior to signal sampling in the digital telephone network. In the first section of the lab we illustrate the effects of aliasing on simple sinusoids. Then, we illustrate the aliasing of a frequency chirp signal. In both sets of experiments the signals are sampled using different sampling frequencies. The reconstructed output signals are viewed graphically and listened to using PC sound hardware.

Aliasing in the Telephone System: During a telephone conversation, the digital telephone network transports streams of sampled voice signals between the phone exchanges to which the two connected handsets are attached. Voice signal transmission between the phone exchange and telephone handsets is normally done in analog form. At the phone exchanges, the required A/D and D/A conversions are performed. The design is based on the fact that a large fraction of human voice energy is typically below about 3.5 KHz. Assuming 4 KHz just to be safe, then the Nyquist sampling rate is 8 KHz (i.e., 2×4 KHz), which is the standard sampling rate used in the telephone system. At each sampling instant an 8-bit sample is taken, which results in a total bit rate of 64 Kbps (i.e., $8 \text{ bits} \times 8$ KHz) for each one-way voice stream.

In each telephone exchange, the incoming analog voice signal is first low-pass filtered to reduce the frequency components above 3.5 KHz, so that significant aliasing does not occur. Following this, the signal is sampled as discussed above and transported to its destination. In this part of the lab we will consider what would happen if this pre-filtering operation were *not* performed and therefore aliasing may occur.

1. First we will consider the effects of aliasing on a sinusoid that is sampled and transmitted through the phone system (i.e., Assume that the person on one end of the phone conversation hums various sinusoids into the handset!). We are interested in what emerges at the other telephone.

   Start by writing a Matlab M-file program that takes the analog sinusoid (voice) input, i.e.,
   $$x(t) = \sin(2\pi f t + \phi),$$
   and samples it at periodic intervals of the sampling period, $T_s = 1/f_s$, where f_s is the sampling rate and ϕ can be any phase. The sampled signal is given by

   $$x(nT_s) = \sin(2\pi n f/f_s + \phi).$$

   We will assume that a very high degree of accuracy is used when the sampling is done. Program this function into a Matlab M-file keeping f, f_s and T_s as variables. Set $f_s = 8000$ Hz as discussed above.

Over the interval from 0 to 10 ms, plot $x(nT_s)$ using the Matlab "stem" command when $f = 100$ Hz. This can be easily done in Matlab using the following M-file.

```
%
% Do a plot of a sampled sinusoid with frequency f = 100 Hz
%
f = 100
%
% Sampling frequency and interval
fs = 8000;
Ts = 1/fs;
%
% Set time duration of plot, i.e., 10 msec.
tfinalplot = 10e-3;
%
% Make the time vector for the plot
nplot=0:Ts:tfinalplot;
%
% Sample the sinusoid.
xnT = sin(2*pi*f*nplot);
%
% Make the plot
stem(nplot, xnT);
%
% Uncomment/edit this next line to save the graph.
% print -djpg graph1.jpg
%
```

Describe what you see. Save an electronic copy of the graph so that you can print it later and include it with your writeup.

When this sampled signal is transported to its destination, it is converted back into analog form then transmitted out to the destination telephone. For simplicity we will assume that this digital to analog conversion is done by connecting the voice sample values with straight lines. This can be seen in Matlab simply by using the "plot" command rather than "stem". In practice a more smooth reconstruction can be done by performing analog filtering at the output.

2. Over the interval from 0 to 10 ms, plot $x(nT_s)$ using the "plot" command when $f = 100, 200, 400, 800$ Hz. An example of the code for doing this for a single sinusoid is given as follows. You should start by getting this M-file running.

```
%
% Use sinusoid frequency f = 200 Hz
f = 200
%
```

```
% Sampling frequency and interval
fs = 8000;
Ts = 1/fs;
%
% Set time duration of plot, i.e., 10 msec.
tfinalplot = 10e-3;
%
% Make the time vector for the plot
nplot=0:Ts:tfinalplot;
%
% Make the time vector for replayed sound spurt
% Play the spurt for 2 seconds
tfinal = 2;
nsound=0:Ts:tfinal;
%
% Sample the sinusoid.
xnT = sin(2*pi*f*nsound);
%
% Make the plot
plot(nplot, xnT(1:length(nplot)));
%
% Save xnT as a wav sound file, soundfile.wav.
% wavwrite(xnT', fs, 8, 'soundfile.wav');
%
% Uncomment/edit this next line to save the graph.
% print -djpg graph2.jpg
%
```

Now make the changes to the M-file so that you can plot all four output frequency plots on a single page using the "subplot" command. Label each plot with its value of frequency, f. Save this graph so you can print it later for use in your writeup.

Now concatenate four 2-second tone segments at each of the above 4 frequencies into a single vector. Then store the sound in a wav file using Matlab code similar to that shown commented out above. Once you have created the sound file, play it out through the PC/laptop audio hardware. Describe what you hear.

3. Now repeat this last section, this time using the input frequencies 7200, 7600, 7800, 7900 Hz. Describe what is happening now! What happens to the output frequency as you increase the frequency of the input sinusoid? Explain in detail why this is happening. What do you hear this time?

4. Based on what you have observed discuss why the performance of the telephone system would degrade significantly if anti-aliasing pre-filtering were not used. Explain how this filtering prevents these negative effects. What would happen in the above experiments when this filtering is in place?

Aliasing of a Frequency Chirp Signal: In this section we will illustrate the aliasing of a frequency chirp signal. A frequency chirp is a signal who's frequency is a linear function of time. We will use the following chirp signal,

$$c(t) = \cos(\pi \mu t^2 + 2\pi f_1 t + \phi).$$

By taking the derivative of this signal's phase, it can be seen that its instantaneous frequency is given by

$$f(t) = \mu t + f_1.$$

i.e., The frequency is increasing linearly with time, starting at f_1 Hz.

1. Write a Matlab M-file that samples the above signal. Use $f_1 = 100$ Hz and set $\mu = 2000$. Start by using a sampling frequency of $f_s = 32$ KHz. Sample $c(t)$ for an 8 second period of time. Plot the first 2000 samples to see what the sampled signal looks like. Then using Matlab, store the sound as a wav file and play it out through the PC/laptop audio hardware. Describe and explain what you see and hear in both cases.

2. Repeat the above experiment but this time use a sampling frequency of 16 KHz. Explain what you hear now! Using the theory that you know, explain in detail what you have heard. Try the same thing for 8 KHz, which indicates what would happen if this signal were sent through the digital telephone network without anti-aliasing pre-filtering. What would you hear over a telephone connection that includes the anti-aliasing filtering? Experiment with other f_1, f_s and μ values. In all cases explain what you hear using the theory you know about sampling.

Writeup: Submit a writeup for the lab. Each group (2 maximum) is responsible for their own experiments and writeup. Include in your writeup a description of everything that you did including all data, Matlab programs and graphs. Relate the experimental results to theory in as much detail as possible.