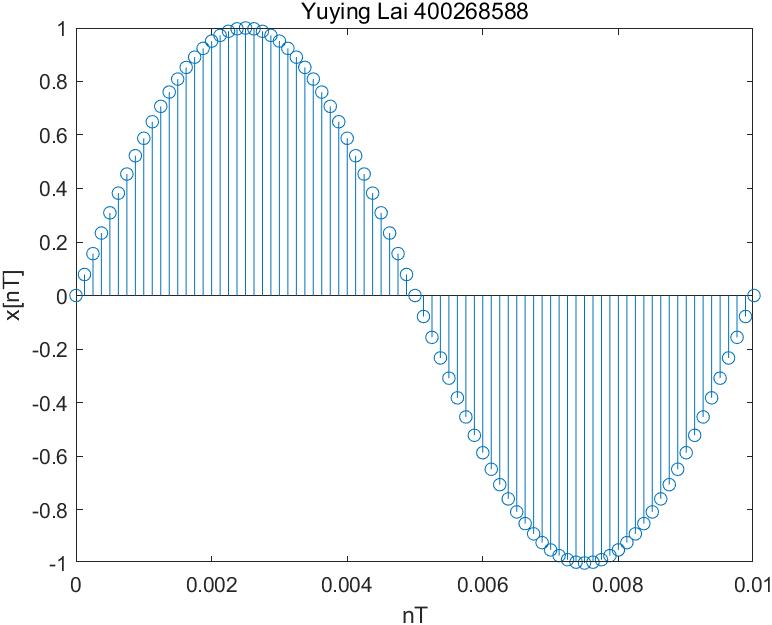
ElecEng 3TP3 Lab3

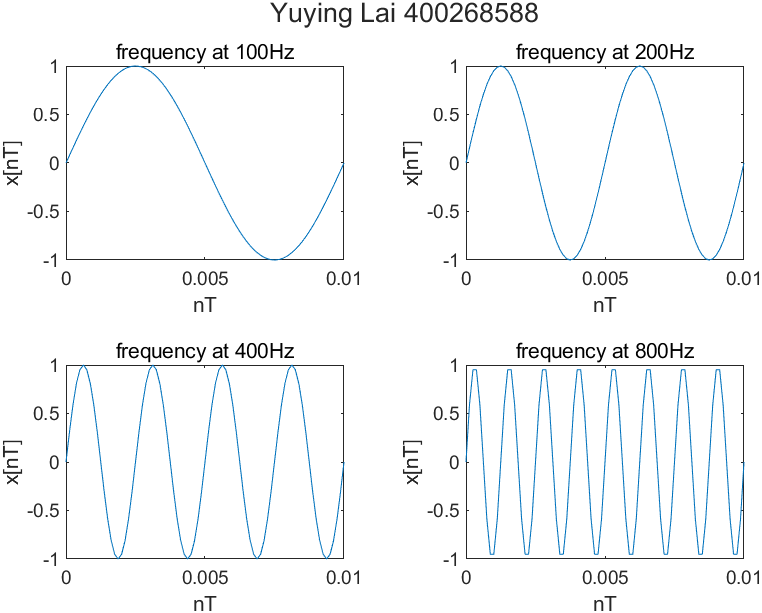
Yuying Lai 400268588 laiy24

Aliasing in the telephone system:

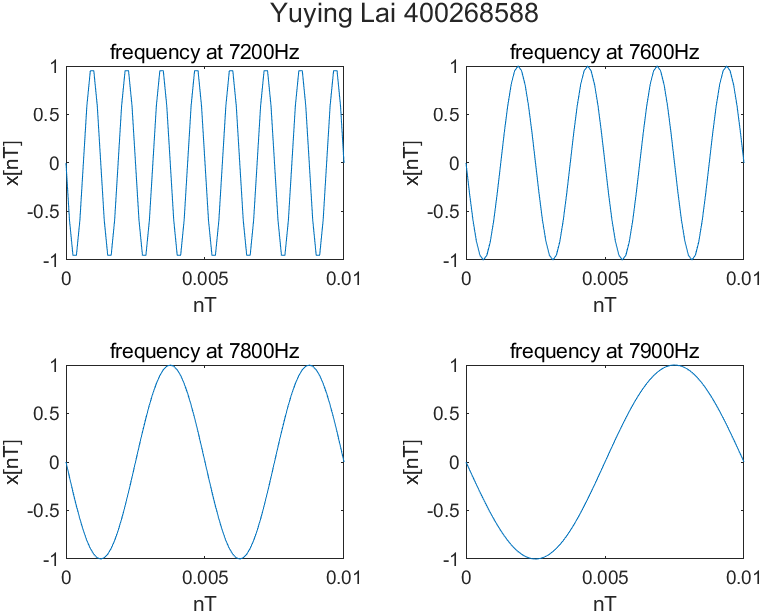
1. Plot of frequency 100Hz, sampling frequency 8000Hz. We can see that the stem plot is a sinusoidal function.



1. Plots of different frequency with sampling frequency of 8000Hz. With higher the frequency, the period is shorter. It means that we see more sin wave in the same amount of time with higher frequency. In the wav file, it produces higher pitch with higher frequency.



1. Plots of different frequency with sampling of frequency 8kHz. We can see that the plot does not have the expected frequency as the input. 7200Hz produces 800Hz frequency plot. 7600Hz produces 400Hz, 7800Hz produces 200Hz and 7900Hz produce 100Hz. The plot is also a negative sin wave. The reason of this problem is that the sampling frequency is too close to the input frequency. Sampling are not fast enough to receive enough info from one period, and the output of one period is actually from many periods of input. This effect is called aliasing. In the wav file, the pitch is lower with higher input frequency.

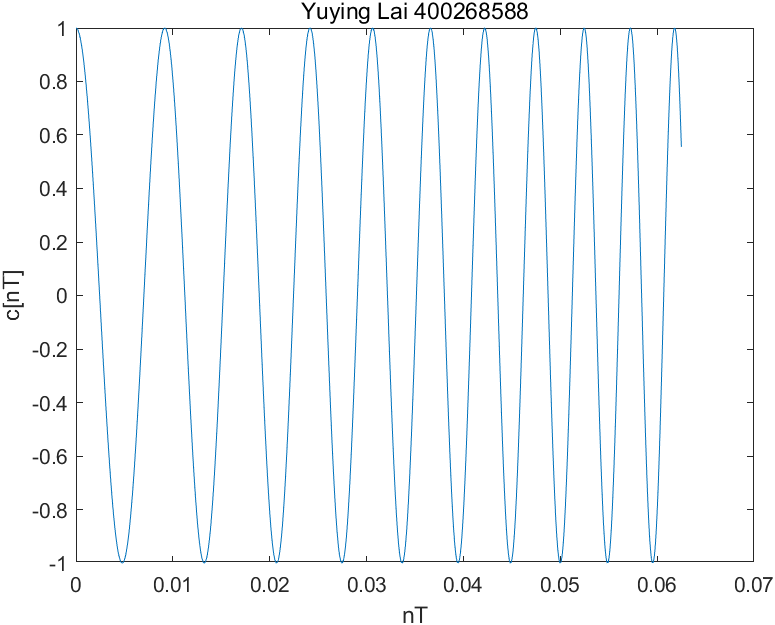


1. The performance degrades if the signal is above half the sampling frequency(Nyquist Frequency). In this lab, signal larger than 4kHz will not give a proper performance with 8kHz sampling frequency. For frequency higher than 4kHz, the output frequency will get smaller with higher input frequency, which produces an unwanted result. In this case, the aliasing frequency is (Fs-Finput);

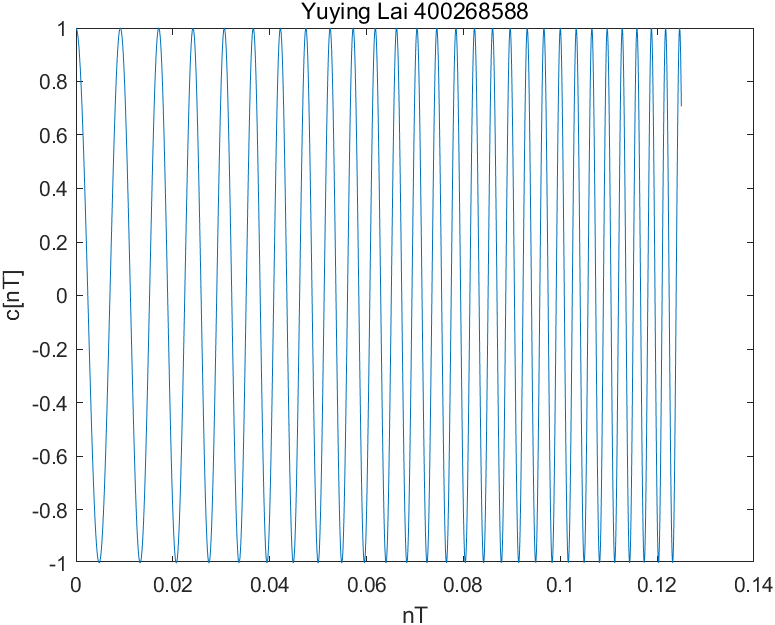
Anti-aliasing pre-filtering prevents this type of negative effect. Theoretically, the filter will eliminate all frequency higher than Nyquist frequency and only remain the low frequency. So that the high frequency will not generate unwanted low frequency due to aliasing. In this lab, all frequency larger than 4kHz will not be sampled or having very small amplitude.

Aliasing of a frequency chirp signal

1. The frequency is increasing over time. The pitch in wav file is also increasing over time.

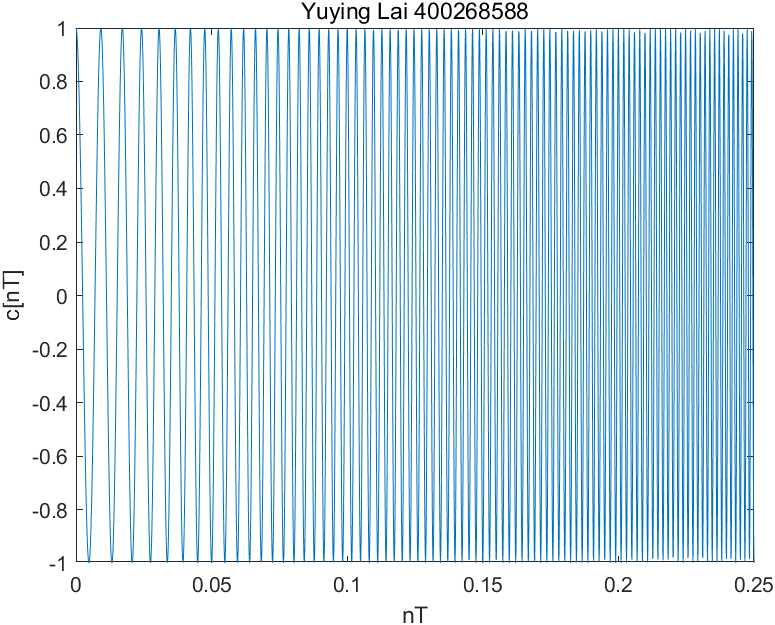


1. Plot with 16kHz sampling frequency. The pitch is getting higher in the first half of the wav file and getting lower in the second half. The same behaviors appear at frequency.



Plot with 8kHz sampling frequency. In the wav file, the pitch is getting higher in the first quarter, getting lower in the second quarter, getting higher again in the third quarter, and getting lower again in the last quarter. The same behaviors appear at frequency.

We know that the frequency should always increasing like that in question 1 in part 2. However, both 16kHz and 8kHz sampling frequency experienced decreasing frequency. This is because the sampling frequency is not large enough to capture all information to present one actual period, which is the same aliasing problem appears in part 1. The Nyquist frequency for 16kHz sampling frequency is 8Khz, and it is 4kHz for 8kHz sampling frequency. Therefore, we will expected the aliasing frequency be Fa = |2m\*Fn - Finput|, where Fn is Nyquist frequency. The number m here will ensure that Fa will always be smaller than Fn.



With higher f1, the beginning pitch is higher, which means the frequency appear at time = 0 will be higher. With higher mu, the increasing speed of the input frequency is faster, which means the output frequency will be increasing or decreasing with a larger value.

Code:

Part 1:

Q1:

% frequency in Hz

f = 100;

fs = 8000; % sampling f

Ts = 1/fs; % interval

% 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);

stem(nplot, xnT);

title('Yuying Lai 400268588');

xlabel("nT"); ylabel("x[nT]");

exportgraphics(gcf,'teleq1.png');

Q2:

clc; clear; close all;

% frequency in Hz

f = [100, 200, 400, 800];

fs = 8000; % sampling f

Ts = 1/fs; % interval

% Set time duration of plot, i.e., 10 msec.

tfinalplot = 10e-3;

% Make the time vector for the plot

nplot=0:Ts:tfinalplot;

% replaed

tfinal = 2;

nsound = 0:Ts:tfinal;

xnT = zeros(size(nsound, 2), 4); % inital

% sample and plot

fig = figure('Name', 'Question 2');

t = tiledlayout(2,2);

title(t, 'Yuying Lai 400268588');

for i = 1:length(f)

xnT\_part = sin(2\*pi\*f(i)\*nsound);

xnT(:, i) = xnT\_part;

nexttile;

plot(nplot, xnT\_part(1:length(nplot)));

titlename = "frequency at " +f(i)+"Hz";

title(titlename);

xlabel("nT"); ylabel("x[nT]");

end

% combine 4 sound

s = xnT(:);

audiowrite('soundfile.wav', s, fs);

exportgraphics(fig,'teleq2.png');

Q3:

clc; clear; close all;

% frequency in Hz

f = [7200, 7600, 7800, 7900];

fs = 8000; % sampling f

Ts = 1/fs; % interval

% Set time duration of plot, i.e., 10 msec.

tfinalplot = 10e-3;

% Make the time vector for the plot

nplot=0:Ts:tfinalplot;

% replaed

tfinal = 2;

nsound = 0:Ts:tfinal;

xnT = zeros(size(nsound, 2), 4); % inital

% sample and plot

fig = figure('Name', 'Question 3');

t = tiledlayout(2,2);

title(t, 'Yuying Lai 400268588');

for i = 1:length(f)

xnT\_part = sin(2\*pi\*f(i)\*nsound);

xnT(:, i) = xnT\_part;

nexttile;

plot(nplot, xnT\_part(1:length(nplot)));

titlename = "frequency at " +f(i)+"Hz";

title(titlename);

xlabel("nT"); ylabel("x[nT]");

end

% combine 4 sound

s = xnT(:);

audiowrite('soundfileq3.wav', s, fs);

exportgraphics(fig,'teleq3.png');

Part 2:

Q1:

% frequency in Hz

f = 100;

fs = 32000; % sampling f

Ts = 1/fs; % interval

u = 2000

% Set time duration of plot, i.e., 10 msec.

tfinalplot = 8;

% Make the time vector for the plot

nsound=0:Ts:tfinalplot;

nplot = 0:Ts: Ts\*2000;

% Sample the sinusoid.

cnT = cos(pi\*u\*nsound.^2 + 2\*pi\*f\*nsound);

plot(nplot, cnT(1:length(nplot)));

title('Yuying Lai 400268588');

xlabel("nT"); ylabel("c[nT]");

audiowrite('chripq1.wav', cnT, fs);

exportgraphics(gcf,'chripq1.png');

16kHz:

% frequency in Hz

f = 100;

fs = 16000; % sampling f

Ts = 1/fs; % interval

u = 2000

% Set time duration of plot, i.e., 10 msec.

tfinalplot = 8;

% Make the time vector for the plot

nsound=0:Ts:tfinalplot;

nplot = 0:Ts: Ts\*2000;

% Sample the sinusoid.

cnT = cos(pi\*u\*nsound.^2 + 2\*pi\*f\*nsound);

plot(nplot, cnT(1:length(nplot)));

title('Yuying Lai 400268588');

xlabel("nT"); ylabel("c[nT]");

audiowrite('chripq116k.wav', cnT, fs);

exportgraphics(gcf,'chripq116k.png');

8kHz:

% frequency in Hz

f = 100;

fs = 8000; % sampling f

Ts = 1/fs; % interval

u = 2000

% Set time duration of plot, i.e., 10 msec.

tfinalplot = 8;

% Make the time vector for the plot

nsound=0:Ts:tfinalplot;

nplot = 0:Ts: Ts\*2000;

% Sample the sinusoid.

cnT = cos(pi\*u\*nsound.^2 + 2\*pi\*f\*nsound);

plot(nplot, cnT(1:length(nplot)));

title('Yuying Lai 400268588');

xlabel("nT"); ylabel("c[nT]");

audiowrite('chripq18k.wav', cnT, fs);

exportgraphics(gcf,'chripq18k.png');