Synthesis of Sinusoidal Signals using Tuning Forks

Tuning forks are physical systems that generate sinusoidal signals. When a tuning fork is exposed to vibration, it disturbs nearby air molecules, creating regions of higher-than-normal pressure (called compressions) and regions of lower-than-normal pressure (called rarefactions).

Pre-Lab:

Read the lab handout and section 2.7 in the textbook on tuning forks. Consider the following question. This lab uses tuning forks with frequencies ranging from 128 Hz to 4000 Hz. Assuming that the stiffness of the tuning forks is the same, comment on the mass of the tuning forks as the frequency increases. Verify your hypothesis when you get to the lab.

1 Overview

In this lab, you will be using the microphone sensor connected to the CyDAQ and your computer to digitize a sinusoidal signal from different tuning forks. The microphone will convert the sound signal generated by the tuning fork to an electrical signal, which in turn is converted to a sequence of numbers stored in a digital file which can be displayed on computer screen as a sinusoidal curve. Characteristics of the sound wave, such as its period T and frequency can be determined from this curve. Knowing the waves period, its frequency f is easily computed using the formula:

$f = 1 / T$

2 Post-Collection Processing

In this section, you will write an M-File function that takes input data signal, the sampling rate, and related time vector at a specific sampling rate and stores it as an output file in .wav format. The input signal will be the tuning fork signal when you strike it and expose to vibration.

3 Labeling Plots in MATLAB

It is very important to be able to label all plots and graphs turned in with your lab reports. Basic functions for labeling axes and putting titles on your graphs are given in this section.
xlabel, ylabel: these commands put a text label on the x and y axes of a plot. For example, to label the x- axis as time in msec: xlabel('time (msec)'); The characters within the quotes are plotted on the active graph.

title: this command puts a title above the graph. For example, to label a plot 'Time vs. Tuning Fork Response', use: title('Time vs. Tuning Fork Response');
axis: this command sets the beginning and end values for the graph axes. For example, if the time should start at 2 sec and end at 5 sec and the recorded signal should range between 0 and 10 volts, use the command: axis([2, 5, 0 10]); axis([xmin xmax ymin ymax]).

4 Frequency view of signals in Matlab

Signals can also be viewed in terms of frequency as well as time. The Fourier Transform is used to show what frequencies are present in a signal. We will cover exactly how this works in class in the next few weeks. The command below plots the frequency spectrum of a signal where Fs is the sampling frequency:

```
freqz(tuningfork1,[1],Fs);
```

5 Lab Procedure Signal Collection

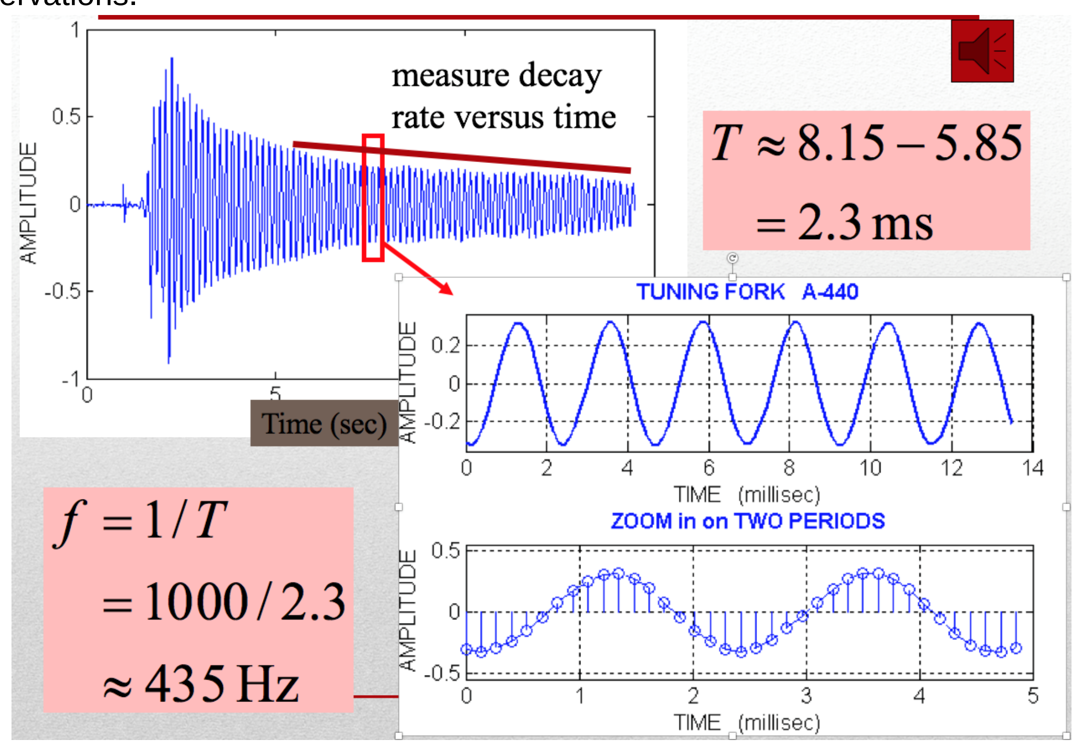
Part 1:

A. Collect audio signals. Using the microphone sensor connected to ADC0 and collect signals for two different tuning forks or generated tones at frequencies at or under 2028 Hz using a sampling frequency of 4056 samples/sec. Do not use any filters. Repeat at 8000 samples/sec sampling frequency. Note: Save the files as .mat and give each signal a different name to avoid overwriting.

   Next, collect data at 8000 samples/sec using the 6th Order Bandpass filter centered close to your tone/tuning fork's frequency.

B. Measure the frequency of these signals using a plot of the signal. Include the plots that you used in your lab report. Be sure to label your graph and all the time units using the methods given in section 3 above. Show how you measured the signal. To have the proper time index on the plot, the sampling interval (the space between samples) used in this signal must be calculated (Ts=1/Fs). The time vector must then be scaled to give the proper time values.

C. Compare the measured frequency with that on the tuning fork. What are possible sources of error? Use the zoom and cursor functions in Matlab figures to improve your observations.



D. Measure how the energy in the signal falls off over time for each of the tuning forks. Provide an annotated figure that shows how you measured the signal. How does the bandpass filtered signal compare to the unfiltered signal, sampled at the same rate?

E. Compute the magnitude spectrum of the signals in step A using the method given in section 4 of this lab. Verify that the peaks occur at the frequency of the tuning fork. Turn in your plots. (Note: If your signal is not close, then your time window for collection is probably too long.) What is the amplitude in decibels of the signal in each case? Compare the tuning fork frequency estimated by the frequency spectrum with the one estimated using the time signal.

F. Comment on the other signals present in the spectrum. Why don't you see a nice clean delta function like in the notes?

G. Search on your browser for Matlab's butter and buttord functions. Could these help? Implement a bandpass butterworth filter with the following parameters:

Let f = your tuning fork/tone frequency in Hz

Passband Frequencies: f - 30 Hz to f + 30 Hz

Stopband Frequencies: f – 60 Hz and f + 60 Hz

Passband Ripple: 3 dB

Stopband Ripple: 35 dB

Apply your filter your filter using Matlab's filter function:

```
filtered_signal = filter(b, a, data);
```

Apply freqz and compare it to the other plots. How does it compare? Better? Worse?

H. Give some reasons for why the CyDAQ has a max sampling rate? What are some system level limitations that prevent this device from being faster? Note, if you used the DAQ in the computer, you would be able to sample around 11,025 samples/sec.

5. Lab Report

A. Answer all questions in section 4 of this lab and include all requested plots and calculations of the frequency and magnitude decay estimation with clearly labeled graphs. Put them in a pdf file and submit electronically using the appropriate lab account. Fill in the table below as part of your report:

Signal Source	Frequency Estimate from Time Plot	Magnitude Decay Rate V/sec	Frequency Estimate from frequency plot
Tuning fork labeled 426.6 at 11,025 samp/sec	430 Hz	.02	428

B. Include all your MATLAB code in an appendix of your report. You do not need to include the tuning fork signals.