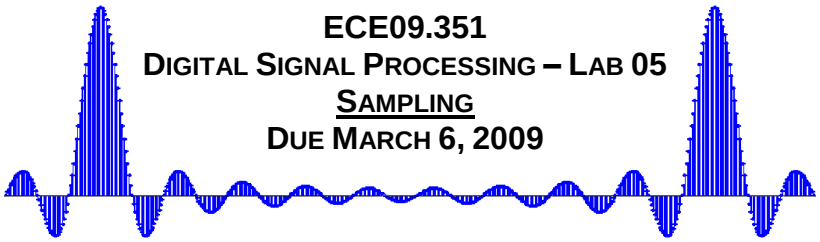

ECE09.351
DIGITAL SIGNAL PROCESSING – LAB 05
SAMPLING
DUE MARCH 6, 2009

Consider the sampling example given in class. In this lab, you are asked to write your own sampling and reconstruction function. However, this function is to be more general than the one in class. Specifically:

1. The analog signal to be sampled will be a user input. It can be of any length. Since an analog signal will be simulated using a discrete time signal in Matlab, it is assumed that the analog signal is originally sampled at a very high rate to simulate the analog signal. You should know - ahead of time - the highest frequency in the signal, so that you can see the effect of the sampling. You may use canonical signals (sine, cosine, etc. at known frequencies as your signal).
2. The sampling rate will also be a user input. Using this sampling rate, you will sample the (pseudo)analog signal.
3. Plot the spectra of both the original and sampled signals, as done in class example.
4. Reconstruct the signal using the **sinc()** interpolation. You may NOT use Matlab's builtin **sinc()** function. You are asked to write your own **sinc()** function, based on the **sin()** function. Use the equations provided in the class as well as Matlab's help files.
5. Plot the reconstructed signal, as well as its spectrum.
6. Use sampling rates below, at and above the Nyquist rate to demonstrate the sampling theorem. Make sure that you clearly show the effect of aliasing in case of an undersampling.
7. Demonstrate your code on an actual audio signal. Create or download a wave file (keep it in 30 – 60 second interval) and read it into Matlab using the **wavread()** command. Run it through your code and write the sampled reconstructed signal to an other wavfile using the **wavwrite()** command. You will be asked to demonstrate your code live to me.
8. Design a simulink model of the system that takes an audio input as in Q7 above, and plays the reconstructed signal as an audio file.

You may use the example code given in class as a reference, but you may not copy it. Your code should be ***sufficiently different*** than the example code given in class, as well as other class members' codes. You are highly recommended, however that you understand how the example code is structured, because it uses very efficient Matlab programming, particularly for reconstructing the signal.

The Simulink model is worth 25 points of the total lab grade. You may start with the sampling1.mdl model provided to you (see class webpage). You may also get ideas from the sampling2.mdl model, however since that model mostly answers this question, the parameters of the model is not available to you. You are highly encouraged to personalize that model. Creative designs and/or models that add meaningful bells & whistles may be eligible for extra credit.