ECE423
Final Project Ideas
Spring 2014

Schedule

Here is the schedule for the Final Project. I have updated the course schedule webpage with this information. Note that we will continue to meet on Tuesdays until the end of the semester.

Tuesday, 4/22/2014	No lecture – discuss possible final project ideas; teams begin making decision on project.
Thursday, 4/24/2014	E-mail me with Final Project choice and begin work on project.
Tuesday, 4/29/2014	Teams present project plan.
Tuesday, 5/6/2014	Interim progress report – informal.
Tuesday, 5/13/2014	Oral presentation of key results. Need not be a polished slide show.
Friday, 5/16/2014, 3PM	Final Project Report due.

Project ideas

You may have noticed suggestions for possible final projects in the End Notes sections of the Labs. There are also a number of suggestions in Chapter 10 of the textbook, along with some starting code. The textbook ideas are good as starting points, but I don't want you to simply download files from the included CD-ROM, compile and run them. You need to extend the projects by new analysis, application, or features.

Here are some additional ideas for projects. They tend to involve solutions in the area of high-speed communication. Note that some projects may require two DSKs; this can be accommodated.

1. Adaptive filter.

    a. A straightforward example of an adaptive filter is a notch filter which can adjust to and reject a single tone within a specific frequency band, say 500 Hz to 12KHz. The idea is that the filter is able, through an adaption algorithm, to adjust the filter coefficients such that any tone in the range will be rejected, and the filter has some ability to "track" and null out the tone as it's frequency changes over time.

    b. A more sophisticated and useful adaptive filter would compensate for the distortion introduced by a communication channel. Some background: digital signals are commonly transmitted as a serial bit stream over a cable, board trace or fiber using a simple encoding scheme ("NRZ") in which a 1 is represented by a positive pulse and a 0 by a negative pulse. At the receiving end, a simple comparator (slicer) recovers the digital stream. The pulses may be launched into the channel as ideal rectangular pulses but bandwidth limitations of the channel result in rounded and in some cases overlapping pulses at the receiving end; this distortion causes slicer errors. The compensating filter partially undoes the channel distortion, separating and squaring up the received pulses, thus reducing the bit error rate. This project involves modeling the channel distortion with a low-pass filter and implementing an adaptive compensating filter based on the Least Mean-Square algorithm.

2. High performance FFT: the FFT algorithms we will study in the lab are a straightforward implementation which can be optimized considerably through pipelining, assembly coding, and use of special routines in the supplied DSK library. This project would involved evaluating these techniques and using them to perform fast convolution on filters which are too complex to implement via the convolution sum.

3. Software PLL: the textbook has a good starting point for this project. The project would involve comparing two types of phase detectors (linear and bang-bang) and two types of loop filters (first order and second order). Measures of performance such as lock range, phase error, and maximum phase/frequency slewing rate could be compared for the different implementations.

4. Advanced communication techniques. This project is an extension of Lab 7 into other techniques, such as FM modulation/demodulation or digital communication using techniques such as phase-shift keying and PAM. Your text has some starting points.

5. Digital PID controller. Both the controller and “plant” can be modeled in the DSK.

6. Voice procession (detection, scrambling). The text has starting point code which could be extended to be able to detect the voice of different individuals or similar extensions.