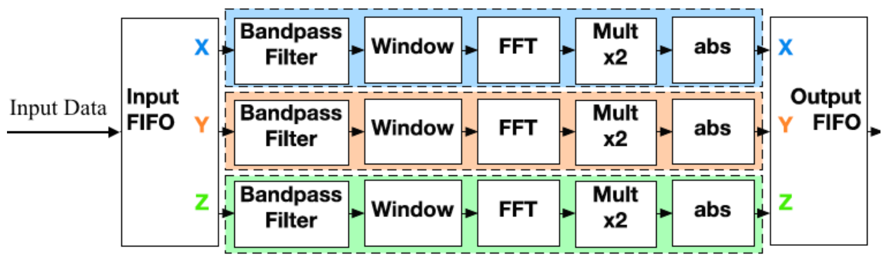
Final Project

ECE 111, Fall 2016

The objective of the this project is to deliver a complete system that aims at performing vibration analysis of a target moving object in a 3D system. The following figure illustrates the high level block diagram of the desired system.



Phase 1: Simulated Data and Spectral Analysis in MATLAB

You should first perform the entire spectral analysis in MATLAB using simulated data. The simulated data will mimic the vibration seen by a target such as a sine wave at a dominant frequency with added noise. A MATLAB script will be needed to filter, window, perform FFT, and finally extract the spectral energy to be displayed graphically along with the filtered time-series.

1. Data will be generated at a sampling frequency of 500 Hz for three streams to represent an accelerometer values. You just need to implement one channel say x.
2. The raw data will be band-pass filtered to pass the frequencies from 10 to 200 Hz with minimal band-pass attenuation. It is necessary to filter the data to eliminate the noise and DC terms of the input. You may use the variables that we already set in the attached MATLAB file for you FIR filter. You need to obtain the "coefficient.coe" for your filter design to load into your FPGA implementation.
3. The filtered data will then be windowed using a rectangular window. The size of the window will depend on the length of the FFT.
4. The windowed time-series data will then be converted to the frequency domain using FFT. The length of the FFT must be set such that a frequency resolution of at least 16 Hz is achieved (e.g., $64 \leq N$).

Note: The time-series data will be generated continuously (one sample at a time) whereas the spectral analysis will require a window of data. For the graphical representation it is advised to display several seconds of the time-series but you could perform spectral analysis only on a smaller fraction of the most recent data. In addition, the windows should overlap between 50-80%, i.e., the window shall consist of 50-80% of old data with 20-50% new data.

Phase 2: Simulated Data and Spectral Analysis in FPGA

In this phase, you should load the simulated data generated by MATLAB into the FPGA. You can do it in a streaming setting using the testbench. The spectral analysis will now be performed on the FPGA. You should save the result to visualize the final results using MATLAB.

Metrics:

- Area overhead: For area metric, you could use Number of Slice LUTs in Xilinx ISE Design Summary. Your target board is Spartan3E.
- Design time overhead: This metric can be calculated by multiplying the clock duration by the number of clock cycles that your design takes to do the operations. Report the number of clock cycles from your ISIM simulations.
- Accuracy: The outputs of your hardware implementation and simulation results should be compared against each other after each stage (filtering, windowing, and FFT). You could use the sum of squared error as your accuracy metric: $(SSE = \Sigma(output_i calculations_i)^2)$
- Frequency resolution

Note: All implementations should be done in floating point format. To compute the absolute value of a complex FFT result you may consider the following approximation to avoid square root computation: $|X| = max(|a|, |b|) + \frac{min(|a|,|b|)}{4}$

Logistics

Group policy:
You can work in groups. Although, you could discuss with other groups regarding the project, no electronic file exchange is allowed.

Submission:
1- Zip your design files and upload it on TED.
2- Write a report (3 pages maximum with font 11) about the project, the steps, challenges and the design trade-offs. Your report should includes your simulation and FPGA implementation's output figures.