CS/EE 5830/6830 VLSI Architecture
Possible floating point project

This possible project is to design a floating point unit that mostly follows IEEE standard with some (hopefully) significant simplifying assumptions to allow you to demonstrate a nifty floating point unit without some of the extra complexity that a full IEEE-style floating point unit would have to have. Here are the specs for full credit:

- Exponent width of 8 bits, excess 127 notation as in IEEE.
- Mantissa minimum width of 16 bits to correspond to your multiplier in case you want to use it. If you have an arithmetic unit of a different width that you'd like to recycle from a previous assignment, you can choose a different width (greater than 16). IEEE single precision standard of 24 bits is another good choice. (Remember that the hidden bit counts as a bit. That is, IEEE single precision has 23 encoded bits plus the hidden bit)
- Assume a hidden bit and sign bit as in IEEE.
- Assume that all inputs are normal numbers. That is, no NaN, no denormals, no 0, no infinities. This means no exceptions on the output.
- Your circuit should do either Add, Sub or Mult on floating point numbers.
- Functional simulation at the switch level (i.e. no behavioral code) is sufficient to demonstrate your circuit. Spectre/mixed simulation is not required (but would be great if you want to do it).
- You may use Synopsys Design Compiler to generate pieces of your design. You may **not** use these tools to generate the complete circuit using the DesignWare built-in floating point templates, but you can use the tools to build constituent pieces that you assemble either as schematics or in Verilog.
- You should implement all the IEEE rounding modes since they're not that tricky.
- Report circuit timing as a switch-level timing. This will assume 0.1ns of delay for each transistor so it won't be terribly accurate, but it will be roughly indicative of performance.
- You should try to build a reasonably fast FP unit. That is, after constructing a functional unit from synthesized pieces, you should at least consider the bottlenecks, and see if there are obvious ways to improve the parts that are holding things up.
- You can work individually, or in teams of two (with a team of two, you should do some of the bonus options for full credit)

Opportunities for extra bonus credit include

- Running mixed-mode analog simulations for better timing information
- Implementing both add/sub and mult in your alu
- Computing power for various operations
- Adding denormal numbers and/or NaNs as inputs
- Add exceptions on outputs

- Pipelining the FP unit
- Including division as an operation

Turn in a written report that describes your project along with the actual design directories. You can turn the design directories in with on-line handin (to CAD6), but please turn in a hard copy of the report. The report should describe what your project is, what features you added, what you did to optimize for speed (or power or size), how you tested your unit, and what the test reports showed. You should also report size (number of transistors) and speed for your design. I should be able to tell what you did, why you chose to do things that way, and what your results were by reading the report. That is, don't expect me to dig information out of your design directory. If you think I should see it in order to decide your grade on the project, put it in the report! I'll look at your design directories as part of my evaluation, but I should be able to tell what you did, why, and how it turned out by reading the report.