University of California
Department of Electrical and Computer Engineering
ECE 156A – FALL 2004

Homework 6

Due: December 9th

This exercise is to practice the synthesis process from RTL code to the gate level simulation. It involves writing and simulating RTL code , synthesizing your code with the Design Compiler, and simulating the output of DC.

Part a) Write a counter that counts from 0 to 10 in RTL (Verilog). Your counter should have an asynchronous reset signal to reset the counter to 0 when reset is 1, an enable signal to allow counting when enable is 1 and to hold the current count when enable is 0, and a clock signal (count increments only during the positive edge of the clock). The only output is the current count.

Write a **complete** test bench that exercises the boundary conditions and normal operation of the counter.

Part b) Synthesize your RTL code with the Design Compiler. Generate a gate-level Verilog version of your counter. (Note you may have to make changes to *part a* to allow your code to be synthesized.)

Part c) Simulate your DC-generated gate-level code using the same test bench in *part a*. You will need to include in your Modelsim project *class.v,* available on the class webpage.

Turn in:

- Verilog RTL code for the counter and test bench
- Waveform of RTL simulation
- DC-generated gate-level Verilog code
- DC-generated output report (area report)
- Waveform of gate-level simulation

Send a copy via e-mail to oguzey@umail.ucsb.edu and turn in a hardcopy during class.

For your enrichment (you may also try these):

- Try various DC optimization techniques
- Create a MITR to simulate your RTL and gate-level code together
- Become aware of the setup and hold time situations, and modify your test bench to account for them