**VLSI Design for Test/Power**

**Proj 3: Dual-Level Power Estimation and Gated Clock Low-power Design Optimization**

**Due at 11:59pm, April 25 (Friday), 2021**

The purpose of this project to use Synopsys Power Compiler to estimate the power consumption of a design and further use logic-level to reduce the power consumption of the design.

**Phase 1: Power Estimation by RT-level and Gate-level switching activities 50%**

As we did in Project 2, you are given a GCD machine designed by VHDL using register-transfer (RT)-level representation. The purpose of the first phase is to help you fully understand how power estimation can be done by Power Compiler. Switching activities can be estimated using RT-level simulation (less accurate but faster) and gate-level simulation (more accurate but slower), respectively. Power estimated using different switching activity simulation levels are then compared for RT and gate level circuits. Details can be found in Power Compiler User Manual.

1. Write a testbench to simulate the RTL design to obtain different switching activies. In your testbench, apply at least 8 test patterns to simulate different circuit behavior, four test patterns where GCD can be found, e.g., GCD (12, 8), GCD (14, 7); and other four test pattern where GCD cannot be found, e.g., GCD (15, 13), GCD (11, 3).
2. Use ModelSim to obtain the RTL switching information in VCD (value change dump) format by simulating the testbench developed in step 1 on the GCD calculator. Translate the VCD file to a SAIF (switching activity information format) file understood by Synopsys using the Synopsys-provided vcd2saif utility.
3. Use the SAIF to find the power consumption of the RT-level design.
4. Use Design Compiler to synthesize your RTL GCD calculator into gate-level using a 90-nm cell library called saed90nm\_typ.db. Use the *RT-level switching activities* (SAIF file) obtained in Step 2 to estimate the power consumption of the gate level synthesized.
5. Repeat Step 2 using the gate-level GCD calculator obtained in Step 4, the testbench developed in Step 1, and the cell library to get the *gate-level switching activities* represented by SAIF.
6. Analyze the power consumption using the gate-level GCD calculator and the gate-level SAIF file.
7. Compare the power estimated by Steps 3 (RT SAIF on RT circuit), 4 (RT SAIF on gate-level circuit) and 5 (gate-level SAIF on gate-level circuit).

**Phase 2: Dual-level Gated Clock Design (50%)**

The purpose of this phase is to use Synopsys Power Compiler to analyze the power consumption of the GCD Calculator and explore the clock gating capabilities of Power Compiler.

1. Write a simple 4-bit load-enable register in VHDL or Verilog and compile it with Design Compiler. Observe the schematic. Re-analyze the RTL design and compile it again, this time using **–gate\_clock**. Compare the generated schematic with the previous schematic.
2. Reuse the testbench for the GCD Calculator and use ModelSim to obtain the RTL switching information in VCD format. Translate the VCD file to a SAIF file understood by Synopsys using the Synopsys-provided vcd2saif utility. Compile the GCD Calculator using Design Compiler and the saed90nm\_typ technology library, and read in the SAIF file. Generate a power report to obtain the unoptimized total power consumed by the design.
3. Set some constraints, create the clock, and add some clock gates to the RTL **(i.e., the input is the RT-level design)** design using the –gate\_clock option of the compile\_ultra command. How many clock gates were inserted? How many FFs are gated? Compare the power report of this new design to the report from the last step.
4. Repeat step 2, this time inserting the clock gates into the gate level design **(i.e., the input is gate-level design)**. Compare the power and clock gating reports to the previous two steps.
5. Repeat step 3, this time setting the power\_driven\_clock\_gating variable to true. Compare the power and clock gating reports to the previous steps.