Project 2: Design an 8-bit fast adder

CSc 137, Fall 2016, Faroughi

Due in class on Oct. 19

Design

1. Structural design: Design a 2-bit CLA adder with 2-bit PG unit (PGU), 2-bit CGU, and 2-bit Sum unit (SU) as follows:
    a. Design a structural model using NAND and NOT gates for an XOR gate

    b. Design a structural model (called pgu_su) for the PGU and SU using your model for XOR and AND gates. **Note**, the CLA modules may be packaged in different ways.

    c. Design a structural model using NOR and NOT gates for the CGU. **Hint**: You would need to determine the max-terms and you may need to use Espresso minimization software (see to the class website)

    d. Combine your models for PGU-SU and CGU to create a structural model for the 2-bit CLA.

    ***Note**: Do not use individual names for signals, instead use vectors (e.g., "wire [7:0] p;" to name 8 signals p[7] to p[0] for p7 to p0, respectively.*

2. Structural model: Connect four 2-bit CLA adder modules to create an 8-bit structural adder module. The 8-bit adder, which would be a hybrid CPA and CLA adder, inputs 8-bit *A*, 8-bit *B*, and an initial carry-in *cin* and outputs 8-bit *S* and the last two carry outs as c_7 and c_6. These carry bits may be used later to generate the overflow signal. **Note**: better (easier to read) if you use all lower case signal names.

3. Structural and behavioral models: Design a 8-bit 2's complement adder/subtractor as follows:
    a. Create a behavior model for an 8-bit Invertor using "if-else" statement. **Note**: an "if-else" statement can only appear inside an "always" block. See Chapter 2 for examples.

    b. Combine your Invertor module and the 8-bit adder module in part 2 to create a structural model for an bit adder/subtractor. This module shall input 8-bit A, 8-bit B, mode *m* and output 8-bit sum S and an overflow bit *ovf*. Also, use your XOR module in part 1.

4. Create a test-bench to test your adder/subtractor using the following test vectors in order:

    A = 8'hFF; B = 8'h01; m = 1'b0;

    A = 8'h7F; B = 8'h01; m = 1'b0;

    A = 8'h01; B = 8'hFF; m = 1'b0;

    A = 8'h55; B = 8'hAA; m = 1'b0;

    A = 8'h80; B = 8'h01; m = 1'b1;

A = 8’h6C; B = 8’hCA; m = 1’b1;

Add additional optional test cases here

Report

Merge design files and the simulation output (Screen print) into a single file separated by dash lines for clarity so the single combined file requires fewer pages to print (save trees). Submit your report containing also the following information in order:

a. Cover page/section with

   Name (legible)
   Section number (Sec 1: 3:00-4:15 or Sec 2: 5:30-6:45)
   Project number and title
   Semester (Fall 2016)

b. Block diagram for the 2-bit CLA, the 8-bit adder, and the 8-bit adder/subtractor. Show signal names clearly.
c. Your HDL designs and the simulation outputs.
d. Analysis of the results. Show that the test cases done by hand (also converted to Decimal values) match those generated by simulating your Verilog code. Also, show why or why not the arithmetic operation results in an overflow (if any).

Follow similar instructions as above also on future project reports.