Homework #5

Problem 1 [20 pts]

The purpose of this problem is to familiarize you with the synthesis and place and route process and to give you a rough feeling for the size of a few simplecircuits. Turn in a detailed report with the results for each part.

- For all designs register the inputs and outputs, and add timing constraint for the clock (e.g 400 MHz=2.5ns). Check the Slack, if it couldn't make the timing constraint, slow down the clock until it meets the timing. Report the final clock period and frequency that the design can results to near zero slack (<0.15 ns).
- Synthesize and place and route the following blocks and report their total slice count, and BRAM, registers and other resource counts. Make sure you include registers (flip-flops) at the input and output and **constrain the timing path** as explained in the class. No need to write testbench and simulate, only turn in the source verilog, but your verilog **must compile correctly.**
- After Implementation report the power dissipation of each block for your clock specification that the design can operate. **Note that for power number, your design clock must be set to the number that you found in previous step, and Implementation step must be done again.**
- **Report your results in a table**. Report the numbers in a single table so it can be used as a note sheet in the future.
- Include verilog files for each circuit in the report.
- Include the generated schemetic under Implementation in your report.

Blocks

a) [10 pts] two input 10-bit adder (11-bit output). Use "+" in verilog.
b) [10 pts] 20-bit by 20-bit unsigned multiplier (40-bit output). Use "*" in Verilog.

Problem 2 [40 pts]

Design a circuit that computes the product of 128 element-vectors, a and b; that is a vector p such that *pi=ai*bi (i*= 0 to 127). The elements of a and b are stored in separate SSRAMS and the result *p* is to be written into a third SSRAM. Assume that computation is started by a control signal, *go* when goes high for one cycle. Then when the computation of all 128 elements is complete, the *done* signal is to be set to one. Assume *a* and *b* are 16-bit values.

Design

i. [5 pt] Draw the detailed block diagram with naming the signals

ii. [10 pt] Design a control state machine in verilog for controlling the signals and report these:
    1. Control Sequence/steps (page 7 of State Machine slides)
    2. Table for control signals (Page 8)
    3. State diagram or Transition Function (Page 11 and 16)

iii. [15 pt] Write verilog for the SSRAM memories, datapath and statemachine to have a complete system.

Behavioral Simulations

iv. [5 pt] Write a testbench and simulate the design

Implementation Results [5 pt]

v. Perform Synthesis and Place and Route (Implementation) steps and
vi. Report maximum clk that the design can achieve by adding timing constraints and minimum slack timing.
vii. Report total slice count, and other FPGA resource counts
viii. Report the power consumption at the frequency that it can work

Problem 3 [40 pts]

Design a FIFO to store up to 256 data items of 16-bits each, using 256x 16-bit dual-port SSRAM for the data storage. Assume the FIFO will not be read when it is empty, not to be written when it is full, and that the write and read ports share a common clock.

Design

i. [5 pt] Draw the detailed block diagram with naming the signals

ii. [10 pt] Design a control state machine in verilog for controlling the signalsand report these:
    1. Control Sequence/steps (page 7 of State Machine slides)
    2. Table for control signals (Page 8)
    3. State diagram or Transition Function (Page 11 and 16)

iii. [15 pt] Write verilog for the SSRAM memories, datapath and statemachine tohave a complete system.

Behavioral Simulations

iv. [5 pt] Write a testbench and simulate the design

Implementation Results [5pts]

v. Perform Synthesis and Place and Route (Implementation) steps and
vi. Report maximum clk that the design can achieve by adding timing constraints and minimum slack timing.
vii. Report total slice count, and other FPGA resource counts
viii. Report the power consumption at the frequency that it can work