CS150 Spring 2009
Midterm Review: Worksheet 1

Topics:

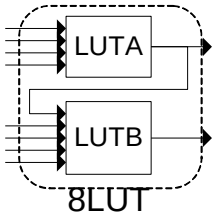
- Platforms for digital design
- FPGA and ASIC platforms
- FPGA Architecture
- Architecture of the Xilinx Virtex® 5 family
- Routing, synthesis
- Processor performance
- Video
- Line Drawing
- Memory-mapped IO
- Memories in hardware
- Memories on the FPGA

*Note: this is **not** a complete list of topics relevant to the midterm.*

Problems

1)
   a. When in the product cycle is the idea of using FPGAs rather than custom silicon (ASIC) cost-effective? You can often buy consumer products with FPGAs inside.
   b. Are FPGAs cheaper or more expensive than ASICs in small quantity?
   c. FPGA manufacturers often provide a low-cost way of mass-producing non-reconfigurable silicon implementing an FPGA design. Can you deduce why? Hint: what happens to a design if it is loaded into a broken FPGA?

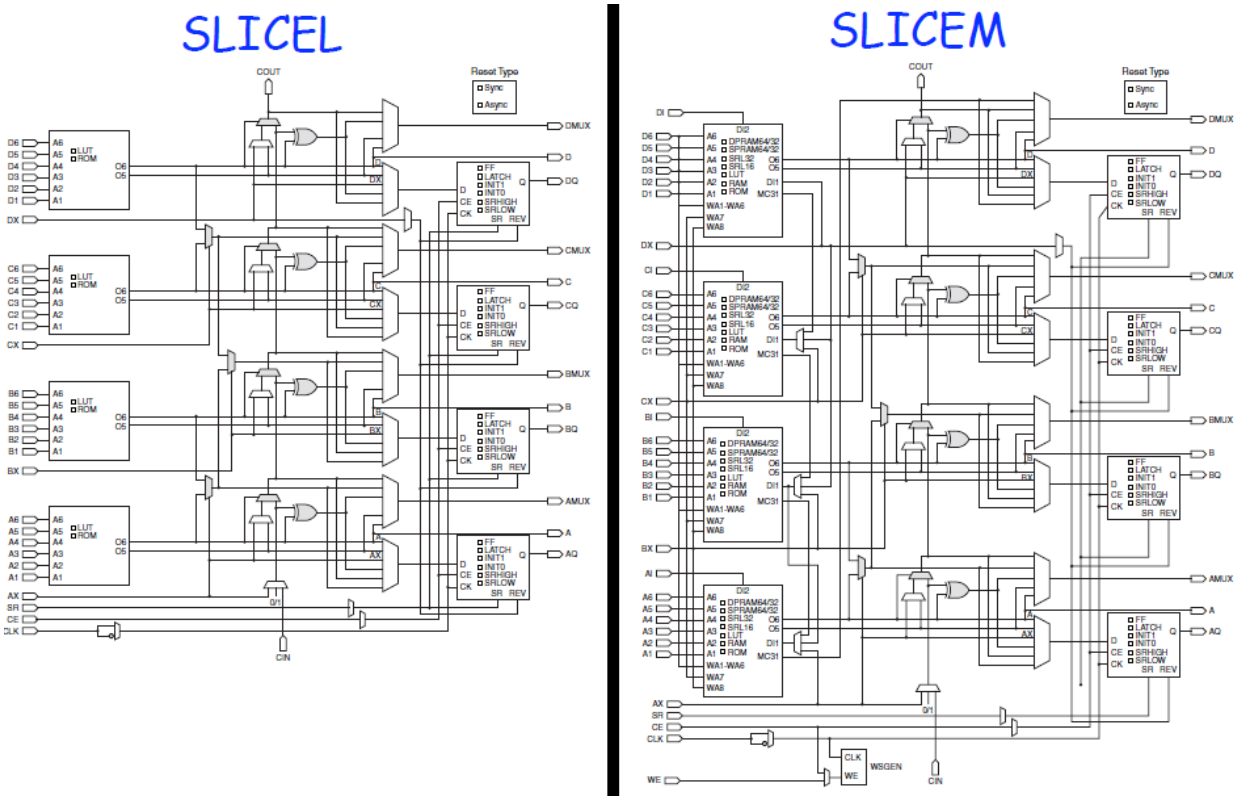
2)
   a. The Virtex E family made use of 4LUTs (4 input bits, 1 output bit). What is the most efficient MUX configuration for this family?
   b. Another FPGA uses "fracturable" 8LUTs (see diagram).



What is the most efficient MUX configuration for area? For performance? Justify your answer.

   c. How many configuration bits are needed to completely specify the function of the fracturable LUT?
   d. The Virtex5 architecture does not provide tristate buffers as FPGA primitives in the FPGA fabric. These primitives are, however, available on each IO pin. How can a tristate buffer be emulated by other FPGA resources (Hint: what are tristate buffers commonly used for?)
   e. Why are clocks usually routed via special wires in an FPGA? What could go disastrously wrong if clocks were routed just like any other wire?

3) The following questions make use of the Virtex 5 SliceL and SliceM diagrams (below).



a. How many configuration bits does a SLICEL appear to have? (*Hint: a mux without a selector input is controlled via configuration bits*). If you **really** want to, answer the same question for a SLICEM.
b. How many 5-input functions (with different inputs) can be implemented using a SLICEL. What about a SLICEM? Does this utilize the slice resources well?
c. How many LUTs are needed to implement a 8-bit AND gate within the Virtex5 family (efficiently in terms of timing)? What about a 9-bit AND gate (Hint: efficiency is a moot point with a gate this large)?
d. Using the SLICEL diagram above, implement two edge detectors (`A&~A'`, where `A'` is the value of `A` last cycle). Assume that any output can be routed back to any input.
e. Using the SLICEM diagram above, implement a run-time reconfigurable 6LUT. Hint: a LUT is a lot like a ROM, while a run-time reconfigurable LUT is a lot like a … (If you do not know how to do this problem, re-read the section of FPGA memories in the Virtex 5 user guide).

4)

a. Why are FPGA wires slow?
b. What is special about clock wires in terms of timing? How is geometry used to achieve this property?
c. What happens to a module with no outputs during synthesis? Why?
d. The input to a Synthesis tool is a textual description of a problem (Verilog). What is the output?

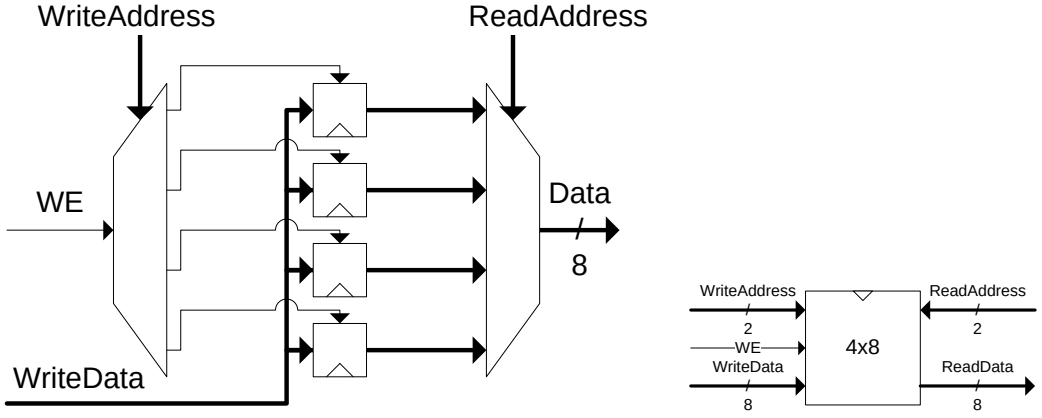
e. What does the MAP procedure do to a synthesized design? What about translation? PAR?
f. You are trying to build a frame buffer using block RAM inferred using Verilog, but PAR is taking 8945 hours to complete. What is likely the problem?

5)
a. How many cycles does a single-cycle CPU take to execute an instruction (ideally)
b. How many cycles does a 5-stage pipelined CPU take to execute an instruction (ideally)? Which offers greater performance? Why?
c. Are branch delay slots helpful in implementing single cycle or pipelined CPUs?
d. You are asked to design a very cheap ultra low-power supercomputer with astoundingly high performance characteristics. What do you say?
e. Why are supercomputers usually very power-hungry and expensive?
f. How many consonants are in "Lebedev"? This is very important.

6)
a. How many bytes are needed to store a 1024x768 image of 24b color?
b. How many bytes are needed if a 4-entry color map was used?
c. Why is a very large memory problematic in terms of timing?
    i. Why is this particularly bad when using BlockRAM on the FPGA?
d. How many pixels are written when drawing the following line using our line algorithm: {(0,0),(5,10)}?

7)
a. What is an alternative to using memory-mapped IO? (Did it say *good* alternative?)
b. What benefit in terms of performance is gained by using the upper half of the address space for IO?

8) Use the diagram below to answer some questions



a. What kind of memory is this? Be thorough.
b. Modify the memory to be simple dual port.
c. Add a read port to this memory.
d. How many 4x8 memories are needed to implement a dual-write, triple-read 8x32 memory.