

# EECS150: Components and Design Techniques for Digital Systems

University of California

Dept. of Electrical Engineering and Computer Sciences

David E Culler & Greg D Gibeling

Fall 2004

## Homework 10: Due Friday 12/3 2:10 pm. Controller Design, Timing & RTL

This homework assignment is a continuation of homework #9. The following steps listed pick up exactly where homework #9 left off around step #3.

### 1. Finish your controller

Now that you have timing diagrams drawn out for interfacing to DRAM, and a basic controller design, you must combine the two. In this step you may find that you need to add counters and states to your controller in order to keep the timing constraints satisfied.

At the end of this step you should have a final controller design in whatever form you wish: block diagram, logic equations, bubble-and-arc and Verilog are all acceptable, though writing Verilog on a computer may help you make changes more easily.

```
module Controller(Clock, Reset,
                 Start, End, Dest,
                 ADDR, RAS_, CAS_, WE_, OE_,
                 SE);

    input          Clock, Reset;

    input          [7:0] Start, End, Dest;
    output         [3:0] ADDR;
    output         RAS_, CAS_, WE_, OE_, SE;

    // Put your code here if you choose to use Verilog

endmodule
```

### 2. Determine the Maximum Clock Frequency

Now that you have a completed controller, you must go back and determine how fast your design can run. As a first step you might wish need to draw out the complete circuit including both the datapath and controller.

Using the timing information in table 1 from homework #9, calculate the maximum clock frequency at which your running sum computer can operate without timing problems.

While calculating timing you will need to take your controller design into account. Remember that a counter is a register and an adder and an FSM is a register and a fair amount of n-Input logic gates.

### **3. Optimize your Controller**

This step is not required, but we highly recommend that you spend at least half an hour trying to find ways to increase not the maximum clock frequency of your design, but its efficiency.

We define efficiency as  $1/(\text{Time to perform one iteration of the RTL loop})$

### **4. What to hand-in**

For this problem we expect you to hand in the following:

- Your controller design in Verilog, block diagram, bubble-and-arc or logic equation form.
- The schematic you drew to compute timing in step 2, with the critical path or paths highlighted.
- The maximum clock frequency of your design, along with the calculation you went through to find it.
- The efficiency of your design as defined in step 3.

Efficiency will not be considered in grading this homework.