CS150 — Final Exam Review

Kris Pister

Albert Magyar, Ian Juch, Vincent Lee

Digital logic

Questions (a-c) consider the logic function $F = \Sigma m(3, 4, 5, 6, 8, 9, 10, 15)$ on the variables A,B,C,D (with D as the LSB in the numerical encoding).

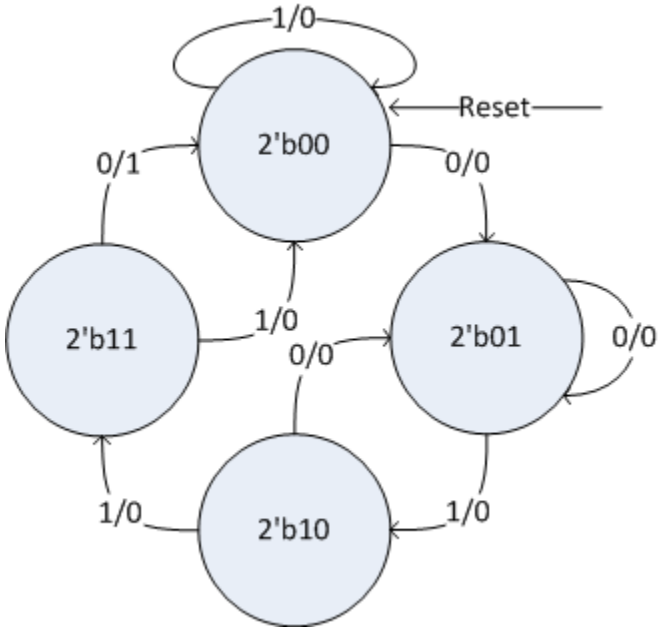
a.) Use a Karnaugh map to reduce the function to minimal sum-of-products form.

b.) Draw a minimal (or relatively close) implementation using only binary or ternary NAND gates.

c.) Draw a minimal (or relatively close) implementation using CMOS logic (assume you have the complement of each signal).

FSM design

Questions (a-d) consider the Mealy machine implementing the state transition diagram below.



a.) What is the purpose of this FSM?

b.) Fill in the following template for a Verilog implementation.

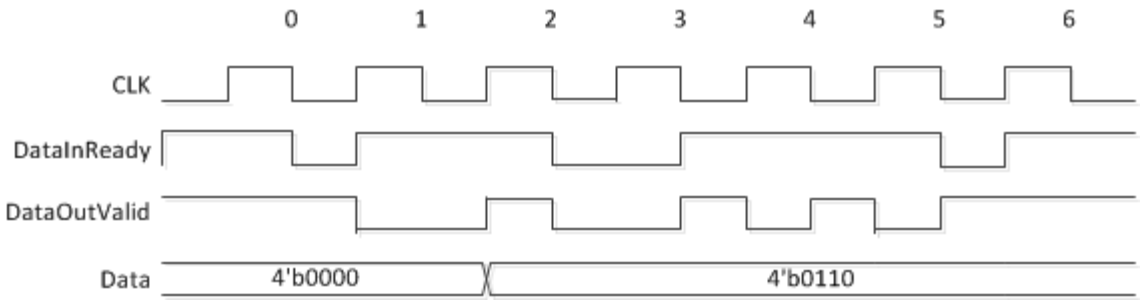
```
module TestFSM (input  clk, rst, in,
                output out);

   // State encoding
   localparam STATE_0 = 2b00, STATE_1 = 2b01, STATE_2 = 2b10, STATE_3 = 2b11;

   // State variables
   reg [1:0] current_state, next_state;

   // Output
   assign out = ____________________________________________;

   // Update state
   always @(posedge clk) begin
      if (rst) _________________________;
      else     _________________________;
   end

   // Choose next_state
   always @(*) begin // FILL IN THIS ENTIRE BLOCK









   end
endmodule
```

c.) Draw a logic-level implementation using only 1-bit D flip-flops, binary AND gates, binary OR gates, and unary NOT gates.

d.) Draw the state transition diagram for a Moore machine with the same functionality (with the necessarily different timing with respect to the input to reflect the difference between Moore and Mealy machines).

Ready/valid interfaces

Questions (a-b) consider the timing diagram below.



a.) What data gets transferred from the source to the sink?

b.) The clock edges are numbered according to the number of the clock period that they *begin*. Given this enumeration, on which positive clockedges does data transfer occur?

c.) A downsampler reduces the bandwidth of a data stream, often by averaging multiple inputs. Implement a 2x downsampler that takes two successive 4-bit inputs and averages them (with minimal loss of precision) and outputs this averaged signal at half the input rate. The module has the following port list:

```
inputs:  Clock, Reset, DataIn[3:0], DataInValid, DataOutReady
outputs: DataOut[4:0], DataInReady, DataOutValid
```

i. Draw a state transition diagram for a Moore machine with the desired functionality.

ii. What happens if there is an odd number of inputs in a given stream? How could this be addressed by changing the format of the Valid signal?

MIPS150

What are the write masks applied to the data cache, instruction cache, and ISR memory for the memory access in the last instruction of each of the following operations?

a.)
```
lui   $t0, 0x3000
addiu $t0, $t0, 0x20
lw    $t0, 40($t0)
```

b.)
```
lui   $t0, 0x1000
addiu $t0, $t0, 0x20
sb    $t0, 43($t0)
```

c.)
```
lui   $t0, 0x3000
sh    $t0, 42($t0)
```

- All questions at the end of the Checkpoint4 document
- Timing and power

1. Given an N stage ring oscillator burning power P from supply V_{DD}, calculate
    a. the delay per stage (assume symmetric rise/fall times)
    b. the input capacitance per stage (assuming now wiring capacitance)
    c. the output resistance
2. Given gate1 with output pull-up resistance R_p and output pull-down resistance R_n, and gate2 with input capacitance C, with gate1 driving gate2
    a. calculate the rise and fall times, t_{plh}, t_{phl}
    b. calculate the energy consumed per charge/discharge cycle from voltage V_{DD}
    c. calculate the power consumed with an average charge/discharge frequency $f_{0\text{-}1}$
3. Given a wiring capacitance of 200fF/mm, how far apart can the gates in the previous problem be placed before the power consumption doubles?

- SDRAM

1. Given the 5-3-3-8 200MHz SDRAM as implemented in your project, and assuming what is the fastest clock speed that your MIPS processor could run and never stall waiting for a fetch from SDRAM? Assume that you can completely redesign the control logic optimally.
2. Given a 5-5-5-12 SDRAM, how fast would its clock need to run to match the read cycle time of the SDRAM in problem 1?
3. How long does it take to refresh a single row in the SDRAM in problem 1? If the four banks can refresh in parallel, how long does it take to refresh the entire chip? If the entire chip must be refreshed every 64ms, what is the minimum fraction of time that must be devoted to refresh?
4. What is the purpose of the burst register?
5. Design the burst register down to transistors, from D/Q pins to the interface to the row register. Use hierarchy. (n.b. for sure there will be at least one “design X down to transistors” question on the final)

- ALU/adders

1. Design a 32 bit ALU down to transistors, with the following functionality:
    a. Two bit control input, selecting one of: ADD, SUB, AND, NOT B
    b. ripple carry adder

    Assuming that all of your basic gates have the same delay, estimate the worst-case delay when performing AND, and when performing ADD.

2. If you change your adder to be a carry bypass adder with 8 bit bypass blocks, what is the worst-case delay for ADD?
3. If you change your adder to be a carry-select adder with 8 bit ripple carry as a building block, what is the worst case delay? What is the increase in area over the ripple carry adder?