Wawrzynek & Weaver
Sp 2018

EECS 151/251A
Digital Design

Final Exam

PRINT your name: ____________________________, ____________________________
(last) (first)

I am aware of the Berkeley Campus Code of Student Conduct and acknowledge that any academic misconduct on this exam will be reported to the Center for Student Conduct and may lead to a "F"-grade for the course.

SIGN your name: ____

You may consult four sheets of notes (each double-sided). You may not consult other notes, textbooks, etc. Calculators, computers, and other electronic devices are not permitted. Please write your answers in the spaces provided in the test.

You have 170 minutes. There are 11 questions, of varying credit (108 points total). The questions are of varying difficulty, so avoid spending too long on any one question.

Do not turn this page until your instructor tells you to do so.

Question:	1	2	3	4	5	6	7	8	9	10	11	Total
Points:	21	8	8	8	6	8	5	12	12	12	8	108
Score:												

Problem 1 ***Keep Your Answers Short And Tweet*** **(21 points)**

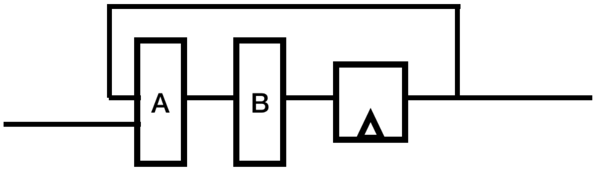
Every answer in this section should be less than 120 characters.

(a) (3 points) You want to optimize a design for low NRE cost. Should you use an FPGA or ASIC and why?

(b) (3 points) You want to optimize a design for low total cost and are producing millions of devices. Sholud you use an FPGA or ASIC and why?

(c) (3 points) If you lower the clock rate but don't change the voltage, how does this affect the energy per computation?

(d) (3 points) Consider the following simple circuit. A units take 1ns, B units take 2ns, the Flip flops have a 1ns setup time, 2.5ns hold time, and a 1ns clk to q time. What is the maximum clock frequency in MHz?



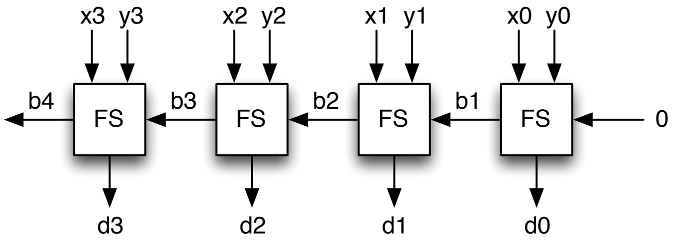
(e) (3 points) Ben Bitdiddle wants to retime that previous circuit 2-slow by putting an additional flip-flop between A and B. Explain to Ben why that would not improve the clock rate for this particular design.

(f) (3 points) You need an external part that is optimized for latency of random access. Do you want to use a DRAM or SRAM and why?

(g) (3 points) How many transistors are needed for each SRAM cell for a true dual-ported SRAM?

Problem 2 ***Combinational Subtractor Circuit*** **(8 points)**

Derive the minimized logic equations for a full-subtractor (FS) cell that computes $x - y$. (A full-subtractor cell is a one-bit wide subtractor used in wider substractors such as the borrow-ripple subtractor shown below). Hint: Derive the truth table first.



Problem 3 ***Combinational Comparator Circuit*** **(8 points)**

Consider a combinational logic circuit for bitwise equal-comparison of two 4-bit numbers. The circuit takes A (a3, a2, a1, a0) and B (b3, b2, b1, b0) and generates a one-bit output x, which is equal to one iff A=B. Implement this circuit using only 2-input NAND gates. Draw your result.

Problem 4 *Moore and Mealy Finite State Machines* (8 points)

Consider the design of a finite state machine that takes as input a stream of bits and generates a one at its output when it first sees a one at its input and then again when it sees a third one at its input. After that it starts again. For example:

```
input  0010001010001...
output 0010000010001...
```

The timing of the output relative to the input will depend on the choice of Moore versus Mealy machine. Show the state-transision diagrams for both a Moore and a Mealy implementation.

Problem 5 ***Flip-flop Circuit*** **(6 points)**

A standard positive edge-triggered CMOS circuit uses back to back transparent latches. Imagine you are using an implementation technology *without* transmission gates. However, you have inverters, nand-gates, nor-gates, and inverting tri-state buffers. Draw the circuit diagrams that show how to implement a latch using just the above components, and how you would use two of those latches to create a flip-flop.

Problem 6 *Multiplexor Circuits* (8 points)

Draw two different CMOS transistor-level circuits for implementing a 2-input multiplexor.

Problem 7 ***Transistor Sizing*** **(5 points)**

Consider the design of a pad-driver circuit. The specification says that the external capacitive load will be 10nF (including the pad itself). Assume the input capacitance of a unit-size inverter is 1pF (and that it has balanced pull-up and pull-down strength). We decide to use 4 stages of inverters in our driver circuit, with the first stage being a unit-size inverter. What should be the sizes of the other inverters in the driver to minimize delay?

Problem 8 ***Carry-Select Adders*** **(12 points)**

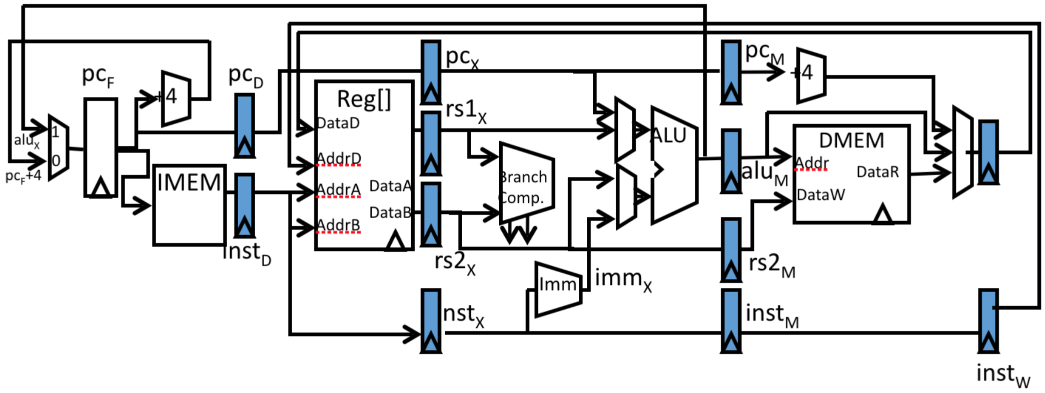
In this problem you are asked to calculate worst case delay through various carry-select adders. Assume that for a full-adder cell (FA) the delay from any input to any output is 1, and similarly for a multiplexor (MUX), the delay from any input to output is 1.

For each of the following, calculate the worst-case delay:

1. 4-bit ripple adder
2. 4-bit carry-select adder
3. 16-bit carry-select adder as presented in class (4 4-bit ripple stages)
4. 16-bit hierarchical carry-select adder. This adder exploits a hierarchical structure by using carry-select for sub-adders instead of ripple adders. The longest ripple chain in this adder would be two stages.

Problem 9 *Working On The Pipeline* (12 points)

Consider the following diagram of a 5-stage, 32b microprocessor pipeline with no forwarding. We want to add forwarding logic to minimize the number of stall cycles and we only want to add forwarding logic after the ID/EX pipeline registers.



(a) (4 points) Highlight on the above diagram where you need to insert two muxes for this forwarding.

(b) (2 points) How many 32b data inputs do these muxes need? How many control inputs (assuming a compact encoding)?

(c) (2 points) If we instead implement these muxes as tri-state buffers, how many tri-state buffers will we need?

(d) (4 points) Provide a sequence of two instructions that will require at least one pipeline stall to process even with your forwarding logic.

Problem 10 *FIFO* (12 points)

A common primitive in digital design is the FIFO: A first-in, first-out buffer. If possible it is preferable to implement a FIFO using a dual-ported memory, but sometimes dual-ported memories are simply unavailable.

An alternate approach can be to use a single-ported memory and logic that is clocked 2x faster than the rest of the logic. You will implement the controller for such a FIFO.

It has four inputs: `clk` which is the 2x clock internal to the logic, `rst` which should reset the FIFO so that it is empty (reset is a synchronous reset), `read` which indicates that the consumer has read the data and the FIFO should move onto the next address for reading, `write` which indicates there is data to write.

It also has four outputs: `addr` which is the address transmitted to the single ported SRAM controlled by the FIFO controller, `we` which is the write-enable for the SRAM, `drdy` which is asserted when the read address is not equal to the write address (indicating that there is more data available), and `wrdy` which is asserted as long as the read address does not equal the write address + 1 (indicating that more data can be written).

You also have three internal registers. `toggle`, which should alternate every clock cycle, `read_addr` which is the address that the SRAM should read at and `write_addr` which is the address the SRAM should be written at. These should be incremented to specify the current address being read or written.

Both read and write are clocked at 1/2 the frequency of clk, so you should only act on read or write every other clock cycle, reading on cycles where toggle is 0 and writing on cycles when toggle is 1.

```
module fifo(clk, rst, read, write, addr, we, drdy, wrdy) begin
   input clk, rst, read, write;
   output [7:0] addr;
   output we, drdy, wrdy;

   reg we, drdy, wrdy, toggle;
   reg [7:0] addr;
   reg [7:0] read_addr;
   reg [7:0] write_addr;

   // Combinational logic here
   always @* begin











   end
   // Sequential logic here
   always @(posedge clk) begin











   end
end
```

Problem 11 ***Constant Multiplication*** **(8 points)**

Neatly sketch the circuit diagram for a multiplier that takes a 4-bit unsigned number X (x3,x2,x1,x0) and outputs $11_{10}\times$ X. Your circuit can only use full-adder cells (FA). Minimize the total cost in number of FA cells.