University of California at Berkeley
College of Engineering
Department of Electrical Engineering and Computer Science

EECS150, Spring 2013

Homework Assignment 7: Ethernet and CMOS
Due March 19^{th}, 2pm

1. A system is designed to take in video input using ethernet(802.3). The video has a resolution of 1920x1080 and each pixel takes 4 bytes. Assuming the refresh rate is 75MHz, how much bandwidth would the video transmission consume? (You need to take into consideration the overhead in ethernet frames)

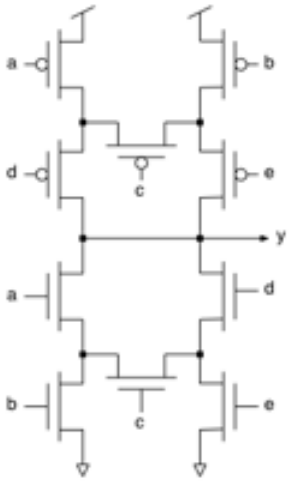
2. Create a circuit to extract the payload from an ethernet packet. You will be accessing the received data through a FIFO. The FIFO has a ready valid interface, and an 8-bit wide data bus. You don't have to worry about the integrity of the received packet. You circuit should write the payload to a block RAM, with the starting address = 0x000. The block RAM has the following inputs: wirteEnable, 11 bit address and 8 bit data. Your design should include a datapath and an FSM which determines when to get data out of the FIFO and write the data into the BRAM. Also after processing one packet, it should be ready to take in the next packet.

3. You have seen how a latch can be constructed using inverters and MOSFET, and how a D flip-flop can be constructed using two latches. Show modifications to the circuit to add the following. Try to minimize the total number of transistors:

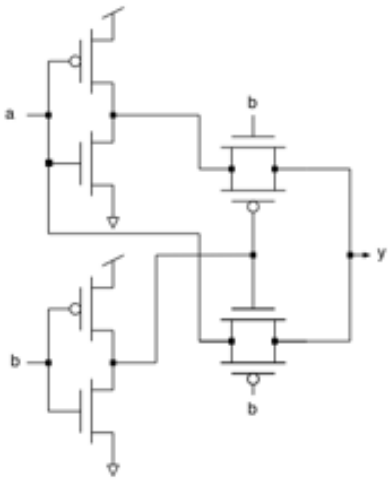
   (a) Synchronous reset
   (b) Synchronous set
   (c) Clock enable
   (d) Asynchronous reset

4. For each of the CMOS circuits shown below, write a Boolean expression for the corresponding function?

(a)



(b)



5. Implement the following function using CMOS transistors, minimize the number of transistor you use. $f = \overline{(a+b)(c+d)}$

6. Assume a product development based on an FPGA has a NRE cost of \$1M and a per unit FPGA cost of \$100. An ASIC version of the product has an NRE cost \$10M and a per unit cost of \$10. If your company is shipping 10,000 units, would you choose FPGA or ASIC? Why?

7. In this problem we will look at how a piece of the programmable interconnect portion of an FPGA might be implemented. Programming the connections generally happens at places where 4 signals/wires (north, south, east, west) meet. Using tri-state buffers, inverters, and/or transmission gates, show how you would implement a circuit that allows any one of the signals to drive any one or more of the other signals. Every signal should be buffered both on its way and on its way out - i.e., you can't just short the wires together, instead each signal must go through at least one tri-state or inverter. Try to minimize the number of transistors required for your implementation.