Marcus Domingo
CSC 205
HW4lA

CSC205 section 1 Spring 2015
Homework 4

How to Submit:
Please submit your solutions (parts A and B separately) through Blackboard.
Remember to put your name and homework number on all the documents that you
submit as attachments.

Total possible points in this homework: 4 for Part B
(You receive 1 bonus point towards Part A if you get all Part A questions correct.)

Part A

1. Answer the following questions regarding sequential logic:
   Complete the following table describing the output Q of the JK flip-flop, at time $t+1$,
   with respect to its inputs, J and K, at time t.

Time t	J(t)	K(t)	Q(t)	Reason:
0	0	0	0	*Initial value of Q*
1	1	1	O	No Change
2	0	1	1	Toggle
3	0	0	O	Reset
4	1	1	O	No Change
5	0	0	1	Toggle
6	1	0	1	No change
7	0	0	1	Set

2. You are to design a mini board with the capacity of 32-byte byte-addressable
   memory. You are given as many of the 8-byte byte-addressable memory chips as you
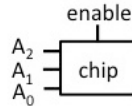   need.
   a. How many 8byte chips are needed? 4 chips
   b. How many address lines go into the board? 5 address lines
   c. Of the address lines that go into the board, how many go to each chip? 3 address lines
   d. How many lines must be decoded for the chip-select inputs? 2
   e. Specify the size of the decoder needed on the board. 2-to-4

Part B
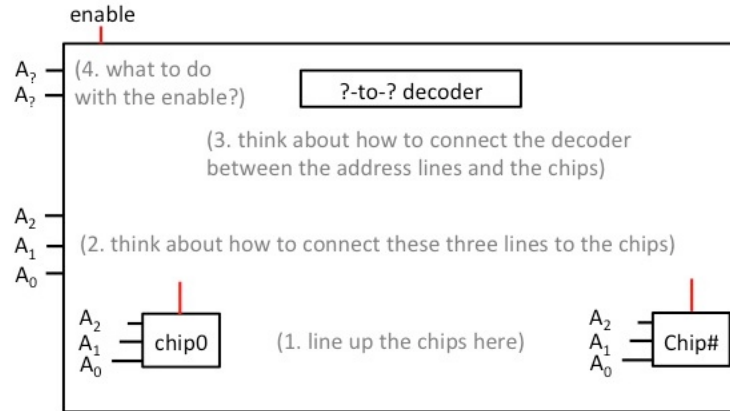
1. [1 pt] This question extends from the previous question. Suppose each 8-byte
   memory chip has three address pins (A2, A1, A0) and has one enable pin. Show the
   circuitry of your 32-byte memory board. You may want to include an enable pin for

your board. (Normally, a chip has a read enable and a write enable, but we simplify them to just one, and ignore any clock pulse here.)

enable

A_2
A_1 chip
A_0

Start by drawing a board like this. And follow the steps below.

enable

$A_?$ — (4. what to do
$A_?$ — with the enable?)   ?-to-? decoder

(3. think about how to connect the decoder
between the address lines and the chips)

A_2 —
A_1 — (2. think about how to connect these three lines to the chips)
A_0 —

A_2
A_1 chip0   (1. line up the chips here) A_2
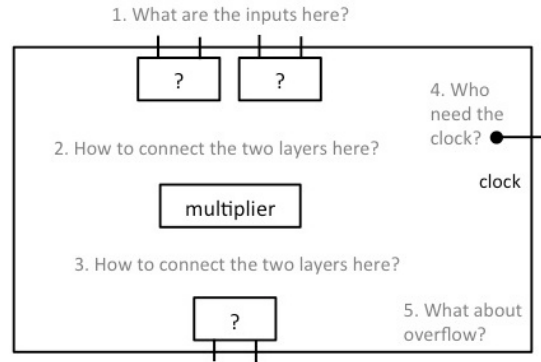A_0 A_1 Chip#
                                                 A_0

2. [Total 3 pts] This question investigates the basic circuitry in and around a *multiplier*, a unit that does multiplication. You are given the following:
    i.   a bunch of logic gates, flip-flops, and simple circuits covered in the class
    ii.   a clock;
    iii. two 2-bit registers, RegA and RegB, storing inputs A=(A1,A0) and B=(B1,B0), which represent two unsigned integers;
    iv.   one 2-bit register, RegD.

a. [0.5 pt] For this part, you may consider the multiplier as a black box. Show how to implement a simple system with two registers, RegA and RegB, and a *multiplier* that multiplies A and B to give the 2-bit result D to be stored in another register, RegD. An overflow line indicates whether an overflow has occurred.

    Hint: Your circuit might look something like this:

1. What are the inputs here?

?   ?

2. How to connect the two layers here?

4. Who need the clock? ● clock

multiplier

3. How to connect the two layers here?

?

5. What about overflow?

b. [0.5 pt] Would you use level-triggered or edge-triggered registers for this set up? Explain.

Hint: Recall that a flip-flop is like a camera. With an edge-triggered flip-flop, the shutter opens and shuts in a very short time (when the clock moves from low to high, for instance). With a level-triggered flip-flop, the shutter is left open as long as the clock is steadily high.

c. [1 pt] Show the circuitry inside the multiplier.

Hints: Express the multiplication of A by B in terms of intermediates (M0, M1, M2 and M3). Express M0 to M1 as Boolean expressions in terms of inputs A and B. Then express the outputs D0 to D3 in terms of the intermediates. Finally, build your circuit based on the Boolean expressions of D0 to D3.

$$
\begin{array}{rrrrr}
 & & & A1 & A0 \\
* & & & B1 & B0 \\
\hline
 & & & M1 & M0 \\
+ & & M3 & M2 & 0 \\
\hline
 & D3 & D2 & D1 & D0 \\
\end{array}
$$

d. [0.5 pt] Show what happens to the value of the register RegD when the following inputs are supplied to the system at different times.

Time	RegA		RegB		RegD	
t	A1	A0	B1	B0	D1	D0
0	0	0	0	0	-	-
1	0	1	0	1		
2	1	1	0	1		
3	1	0	0	1		
4	1	0	1	1		
5	1	0	0	0		

e. [0.5 pt] We have mentioned in class that the delay in a simple combinational circuit is negligible. With large circuits, such as a multiplier that processes two 32-bit integers, the delay is no longer negligible. What constraints does the circuit impose on the CPU's clock for the instruction cycle?