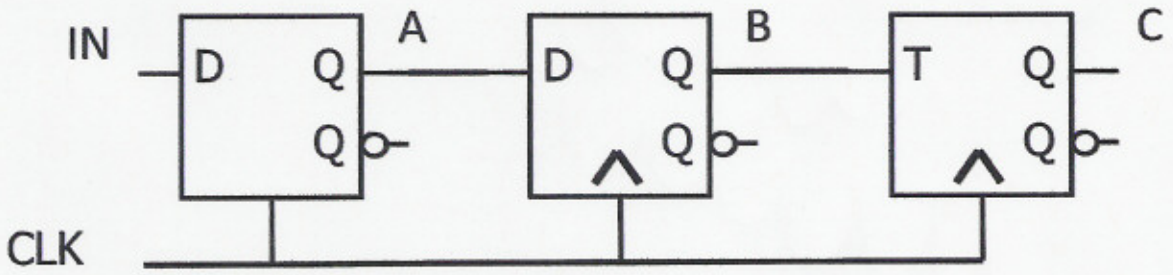
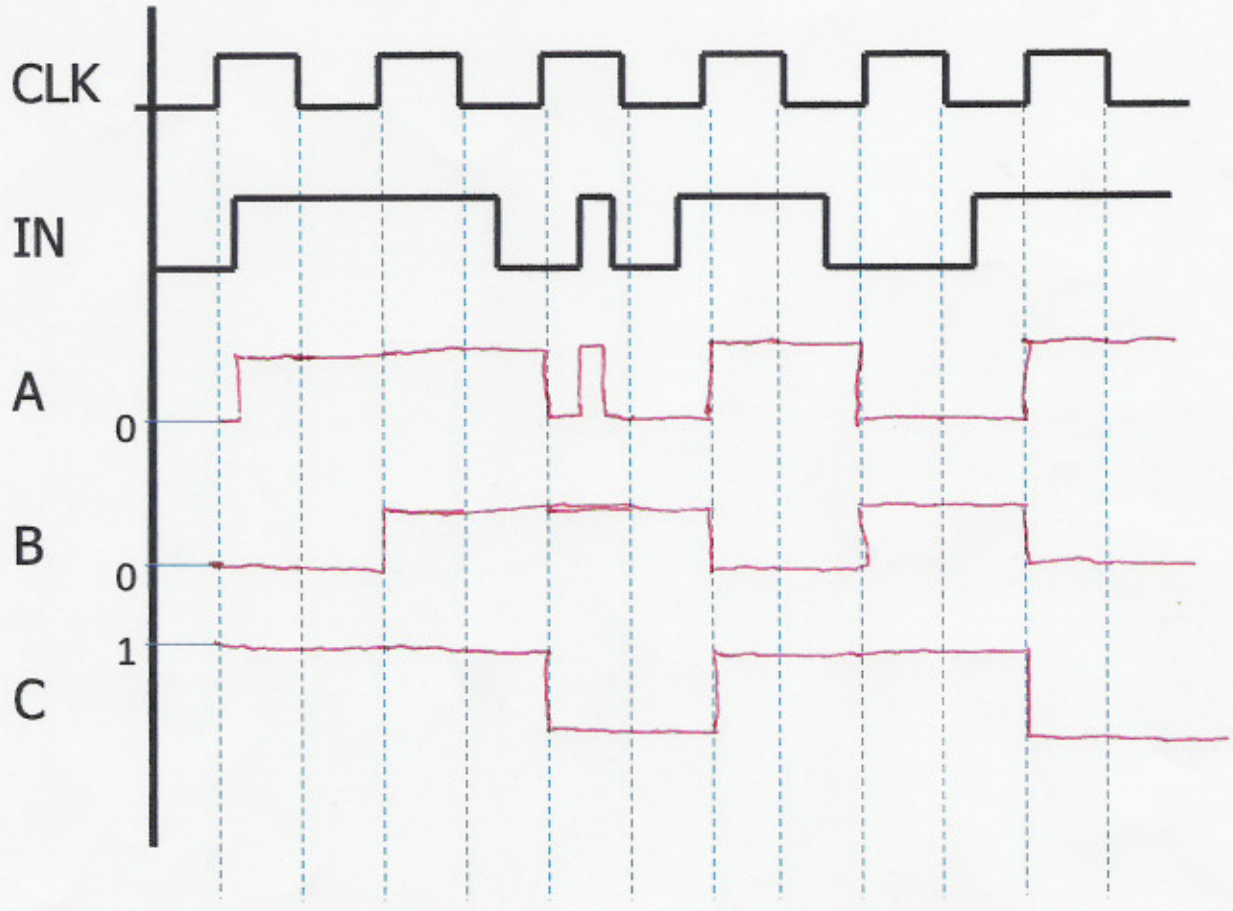
(20pts) Problem 1: Fill in the three empty lines of the timing diagram below.

A starts with 0, B starts with 0, C starts with 1. Assume each setup time, hold time, etc. is respected (so none of the inputs is changing too close to the clock edge).





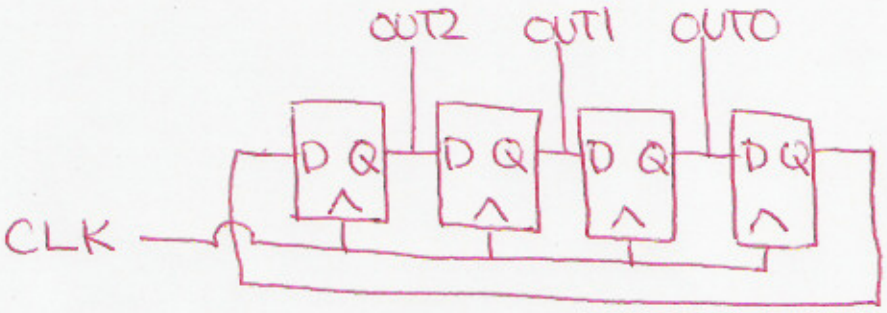
-4 Didn't perserve propagation time

didn't treat latch, flipflop, toggle correctly. -4 each

(40pts) Problem 2:

We have the following 4-state counter. 000 -> 100 -> 010 -> 001 -> 000 -> ...

(a) (8 pts) Build this counter as a ring counter (like shift registers) with D flip flops. (hint: no need to take the four steps in this part of the question.) Label output bits as Out2, Out1, and Out0.



8

(b) (32pts) Now, you want to build this counter using T flip flops. Don't worry about self-starting issues. Take the following four steps: (1) draw a state diagram, (2) fill in the state transition table (hint: don't forget to also fill in the inputs to T flip flops), (3) minimize the logic using k-maps, and (4) implement the design into a circuit.

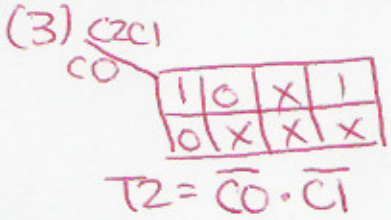
(1)

4

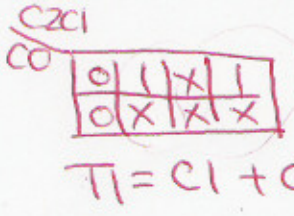
(2)

C2 C1 C0	N2 N1 N0	T2 T1 T0
0 0 0	1 0 0	1 0 0
0 0 1	0 0 0	0 0 1
0 1 0	0 0 1	0 1 1
0 1 1	X X X	X X X
1 0 0	0 1 0	1 1 0
1 0 1	X X X	X X X
1 1 0	X X X	X X X
1 1 1	X X X	X X X

(3)

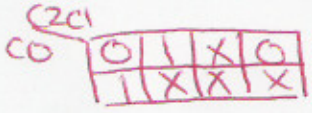


$T2 = \bar{C0} \cdot \bar{C1}$



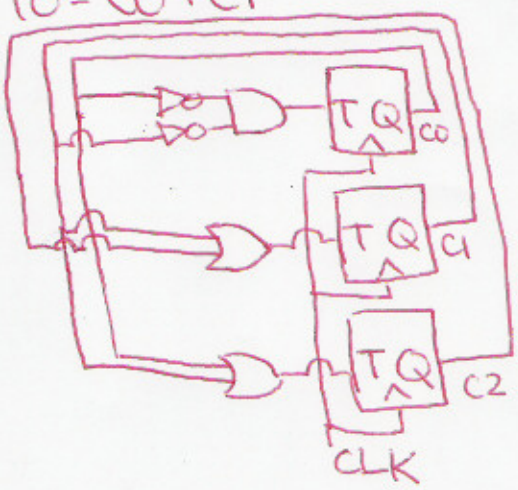
$T1 = C1 + C2$

8



$T0 = C0 + C1$

(4)



18

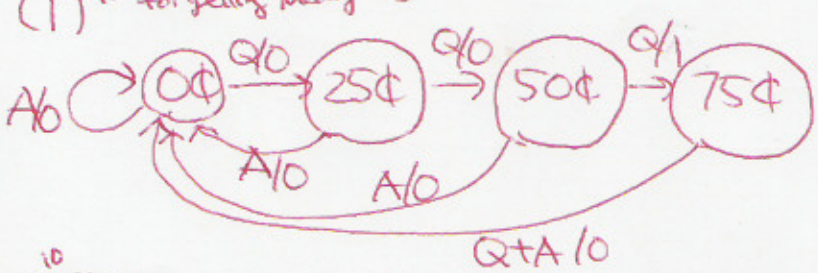
10 + 2 = 12

(40pts) Problem 3:

You are building a "MEALY" vending machine that sells flip-flops (the shoes!) using flip-flops. They cost 75 cents. The machine only takes quarters (Q). You can put in other coins (A), but they will all result in resetting the machine to 0 cents. The machine only has a single slot (can't put in more than one coin at a time), and after you get your flip-flops, putting in any coins (including quarters) results in resetting the machine to 0 cents.

Take the following 4 steps (steps 1 and 2 combine a couple of steps so you are basically going through all 6 steps). (1) Draw a state diagram (minimize), (2) fill in a state transition table (encode the states), (3) minimize the logic using k-maps, and (4) implement the design into a circuit. AGAIN, DON'T FORGET TO BUILD A MEALY MACHINE!

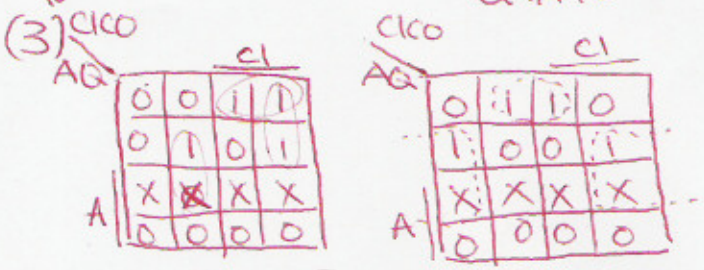
(1) 10 forgetting slash -1 forgetting Mealy -3



(2) 10 combining is not correct.

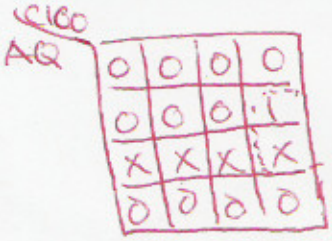
C1 C0	A Q	N1 N0	OUT
0¢	0 0	0 0	0
	0 1	0 1	0
	1 0	0 0	0
	1 1	X X	X
25¢	0 0	0 1	0
	0 1	1 0	0
	1 0	0 0	0
	1 1	X X	X
50¢	0 0	1 0	0
	0 1	0 0	0
	1 0	1 0	0
	1 1	X X	X
75¢	0 0	0 0	0
	0 1	0 0	0
	1 0	0 0	0
	1 1	X X	X

(3) 10



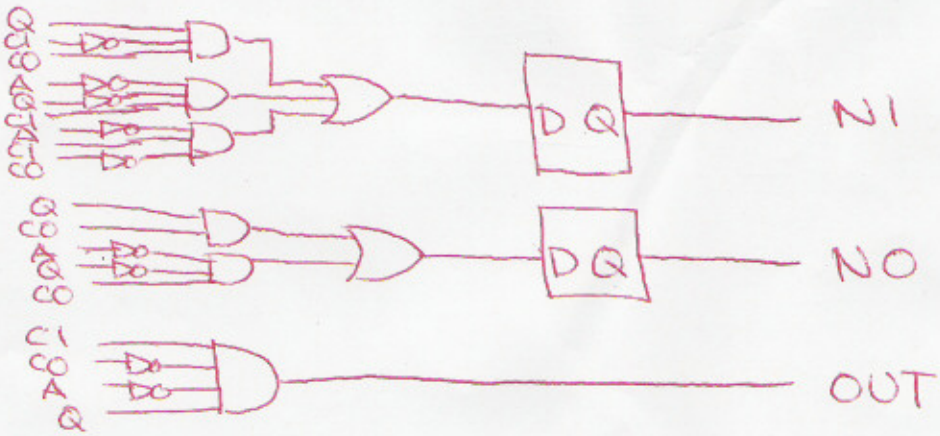
$N1 = Q\cdot\bar{C1}\cdot C0 + \bar{A}\cdot\bar{Q}\cdot C1 + \bar{A}C1\bar{C0}$

$NO = QC0 + AQC0$



$OUT = C1\cdot\bar{C0}\cdot Q$

(4) 10



Problem - 2 grading

Part (a)

(i) 4 FF Version : Full credit
3 FF + NOR : Full credit

- 1 for unlabled diagrams (FFs)
- 2 for unlabled outputs
- 5 for non-functional designs

Part (b)

1. 4 points (-1 partial credit)
2. 12 points (-10 if no T-FF)
3. 8 points
4. 8 points

-8 if D-FF