CSEE 3827: Fundamentals of Computer Systems

Midterm Exam
March 9, 2011

SOLUTIONS

Read all of the following information before starting the exam:

- Be sure to write your name on each page of the exam.
- Use the exam itself for your solutions (no blue books or spare sheets of paper). You may use the backside of pages if you need more space.
- Show your work in order to earn partial credit.
- You may use your textbook and class notes, but *absolutely no electronic devices (laptops, cell phones, etc.)*
- Good luck!

Problem	**Point Value**	**Points Earned**
1	12	
2	10	
3	8	
4	10	
5	10	
Total	50	

1. **(12 points)** Salt and pepper.

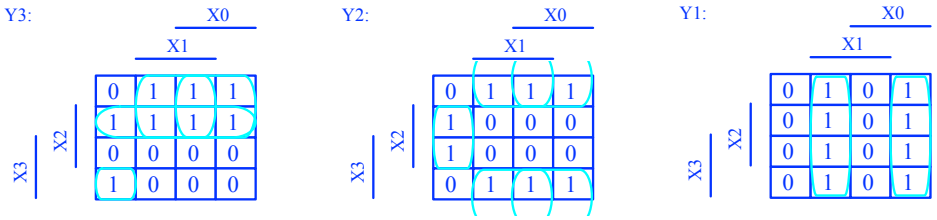
   (a) **(2 points)** Prove algebraically that $BC + AB + \overline{A}B = B$.

   - $BC + AB + \overline{A}B$
   - $B(C + A + \overline{A})$
   - $B(C + 1)$
   - $B(1)$
   - B

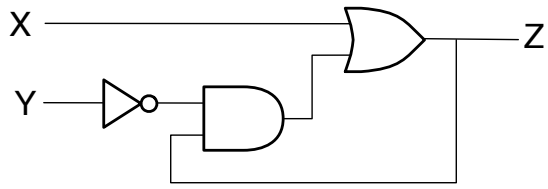
   (b) **(4 points)** Draw a schematic for a *minimal* circuit that uses only *and*, *or*, and *not* gates that performs the two's complement operation on a four bit input value. Let the input be $X_{3:0}$ and the output be $Y_{3:0}$.

X_3	X_2	X_1	X_0	Y_3	Y_2	Y_1	Y_0
0	0	0	0	0	0	0	0
0	0	0	1	1	1	1	1
0	0	1	0	1	1	1	0
0	0	1	1	1	1	0	1
0	1	0	0	1	1	0	0
0	1	0	1	1	0	1	1
0	1	1	0	1	0	1	0
0	1	1	1	1	0	0	1
1	0	0	0	1	0	0	0
1	0	0	1	1	1	1	1
1	0	1	0	0	1	1	0
1	0	1	1	0	1	0	1
1	1	0	0	0	1	0	0
1	1	0	1	0	0	1	1
1	1	1	0	0	0	1	0
1	1	1	1	0	0	0	1

$$Y_3 = \overline{X_3} \cdot X_2 + \overline{X_3} \cdot X_0 + \overline{X_3} \cdot X_1 + X_3 \cdot \overline{X_2} \cdot \overline{X_1} \cdot \overline{X_0}$$
$$Y_2 = X_2 \cdot \overline{X_1} \cdot \overline{X_0} + X_0 \cdot \overline{X_2} + X_1 \cdot \overline{X_2}$$
$$Y_1 = \overline{X_1} \cdot X_0 + X_1 \cdot \overline{X_0}$$
$$Y_0 = X_0$$



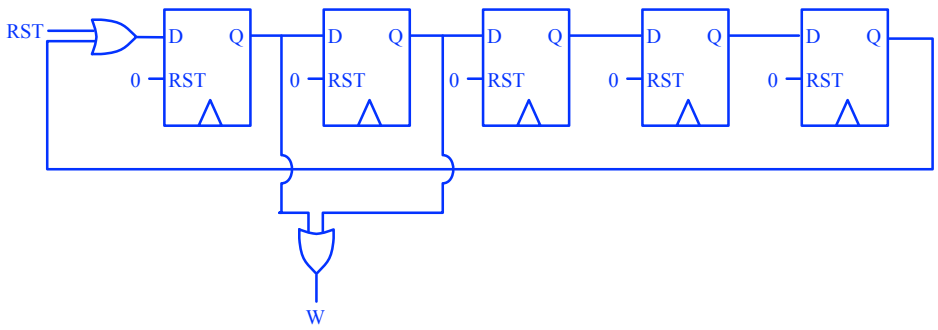
(c) **(2 points)** Examine the sequential circuit below. First, give a complete truth table, then describe conceptually what the circuit does.



X	Y	Z	
0	0	Z_{prev}	(hold)
0	1	0	(reset)
1	0	1	(set)
1	1	1	(set)

This is an RS latch, but instead of instability when both inputs are 1, this latch sets.

(d) **(4 points)** Design a Moore machine robot controller which causes a robot to wave (output $W = 1$) in 40% of the time (I.e,, in 40% of its states, any 40% will do) Implement this controller using D flip-flops.



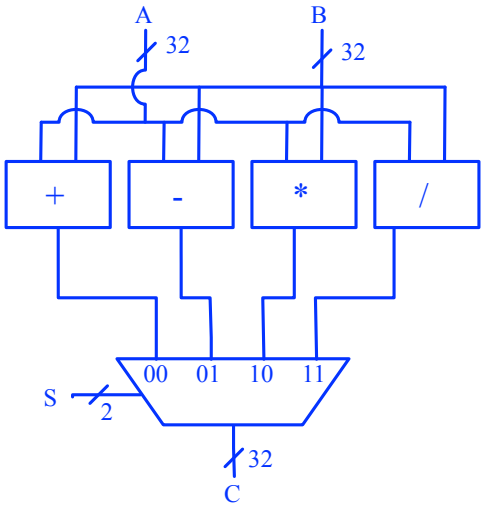
2. **(10 points)** Design a 32-bit ALU (arithmetic logic unit) which computes either $+$, $-$, $\times$, $\div$ of two 32-bit values A and B putting the result on 32-bit output C. You may assume the presence of existing a 32-bit adder, subtractor, multiplier, and divider, and you may incorporate them in your design as black boxes. The particular operation is controlled by an additional two bit input S, where

   NB: In addition to the arithmetic modules, you may incorporate muxes, decoders, and other standard components as you find helpful.

   - if $S = 00$, $C = A + B$
   - if $S = 01$, $C = A - B$
   - if $S = 10$, $C = A \times B$
   - if $S = 11$, $C = A \div B$

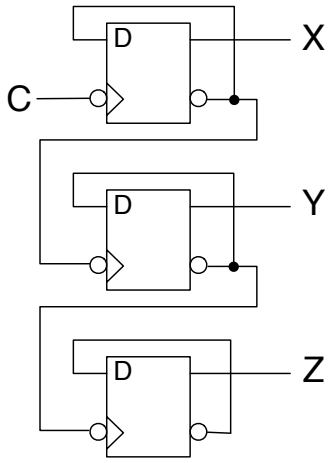
   (a) Draw a schematic for your ALU design.

   (b) Give an expression for the propagation delay (TP_{ALU}) and contamination delay (TC_{ALU}) of your ALU, in terms of the propagation and contamination delays of the sub-modules. For example, TP_+ is the propagation delay of the adder, and TC_{MUX} is the contamination delay of a mux.

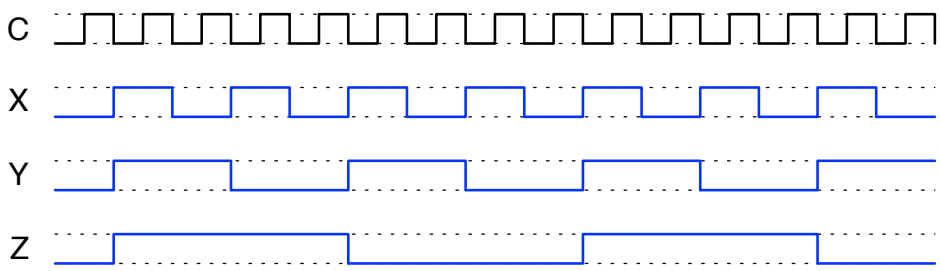


$$TP_{ALU} = max(TP_+, TP_-, TP_\times, TP_\div) + TP_{MUX}$$
$$TC_{ALU} = min(TC_+, TC_-, TC_\times, TC_\div) + TC_{MUX}$$

3. **(8 points)** Examine the circuit below. Note that these D flip-flops are negative, or falling, edge triggered.



(a) Complete the following timing diagram. You may assume wires and gates have zero delay (i.e., all transitions are instantaneous), and that at the start $X = Y = Z = 0$.



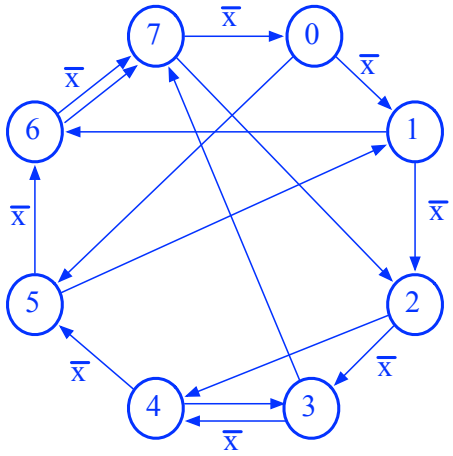
(b) Describe conceptually what this circuit does.

This is a 3-bit up-counter.

4. **(10 points)** Use D flip-flops and basic combinational gates (AND, OR, NOT) to design the following counter/pseudorandom number generator. The circuit has a CLK and RST and one control input x. It provides a three bit output indicating the numeric output $N_{2:0}$. When $x = 0$, the circuit should operate a binary up-counter. Otherwise, it should operate as a pseudorandom number generator according to the following function table.

   NB: While your design must use only the primitive elements listed above, it need not be minimal.

Present State	Next State	
	Binary Up-Counter ($x = 0$)	Pseudorandom No. Gen. ($x = 1$)
0	1	5
1	2	6
2	3	4
3	4	0
4	5	3
5	6	1
6	7	7
7	0	2



(unlabeled edges transition on x)

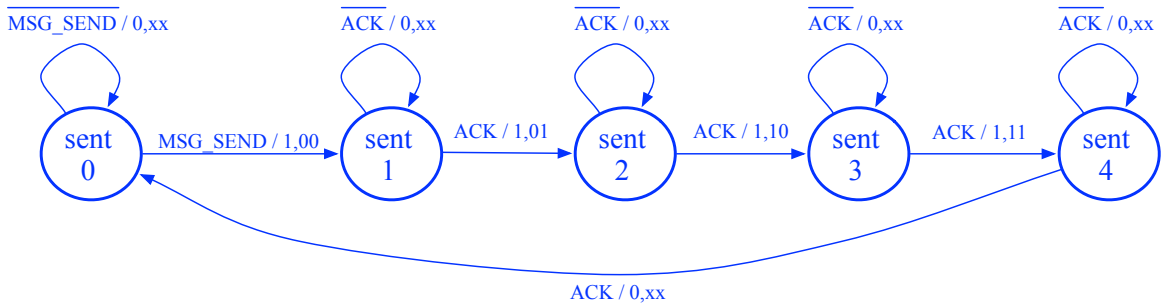
- Use 8 D flipflops, with a one hot state encoding (i.e., state $i \Leftrightarrow Q_i$)
- Next state logic
  - $D_0 = \overline{x}Q_7 + xQ_3 + RST$
  - $D_1 = \overline{x}Q_0 + xQ_5$
  - $D_2 = \overline{x}Q_1 + xQ_7$
  - $D_3 = \overline{x}Q_2 + xQ_4$
  - $D_4 = \overline{x}Q_3 + xQ_2$
  - $D_5 = \overline{x}Q_4 + xQ_0$
  - $D_6 = \overline{x}Q_5 + xQ_1$
  - $D_7 = \overline{x}Q_6 + xQ_6$
- Output logic
  - $N_2 = Q_4 + Q_5 + Q_6 + Q_7$
  - $N_1 = Q_2 + Q_3 + Q_6 + Q_7$
  - $N_0 = Q_1 + Q_3 + Q_5 + Q_7$

5. **(10 points)** Give a state transition diagram for a *Message Transmission Controller* Mealy Machine. Messages are broken into four packets for transmission, so sending a message amounts to sending packet 0, packet 1, packet 2, and packet 3. The sender must receive an acknowlegement of packet i before transmitting packet $i + 1$.

   NB: You need only provide the state transition diagram, no implementation necessary.

   The Message Transmission controller has the following interface:

   - Inputs:
     - **MSG_SEND**: 1 bit signal that initiates the transmission of the four packets that comprise the message
     - **PKT_ACK**: 1 bit signal acknowledging receipt of the previous packet
   - Outputs:
     - **PKT_SEND**: 1 bit signal that initiates the transmission of a packet
     - **PKT_ID**: 2 bits indicating which packet is to be sent



(output code: PKT_SEND, PKT_ID)

(Scratch space.)