EE434

ASIC and Digital Systems

Final Exam

May 5, 2015. (1pm – 3pm)

Instructor: Dae Hyun Kim (daehyun@eecs.wsu.edu)

Name:

WSU ID:

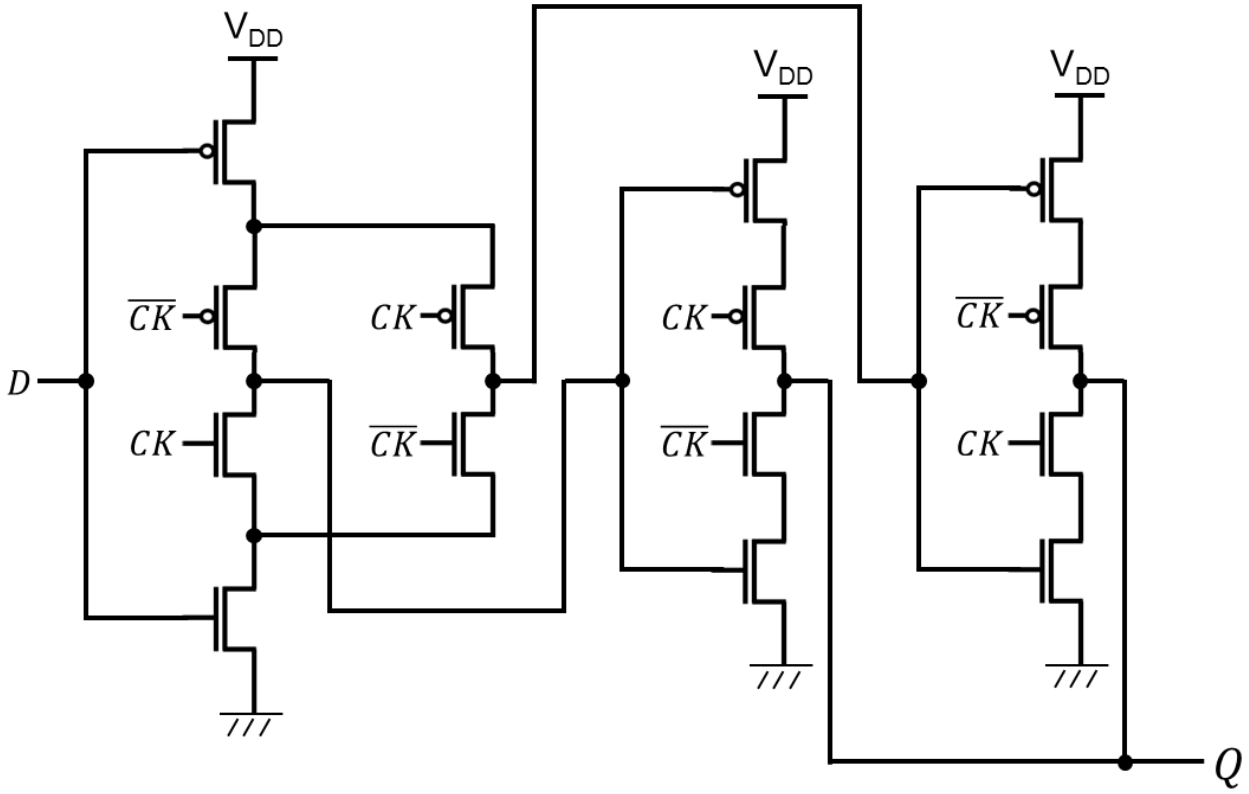
Problem	Points	
1	10	
2	10	
3	10	
4	10	
5	10	
6	10	
7	10	
8	10	
Total	80	

* Allowed: Textbooks, cheat sheets, class notes, notebooks, calculators, watches

* Not allowed: Electronic devices (smart phones, tablet PCs, laptops, etc.) except calculators and watches

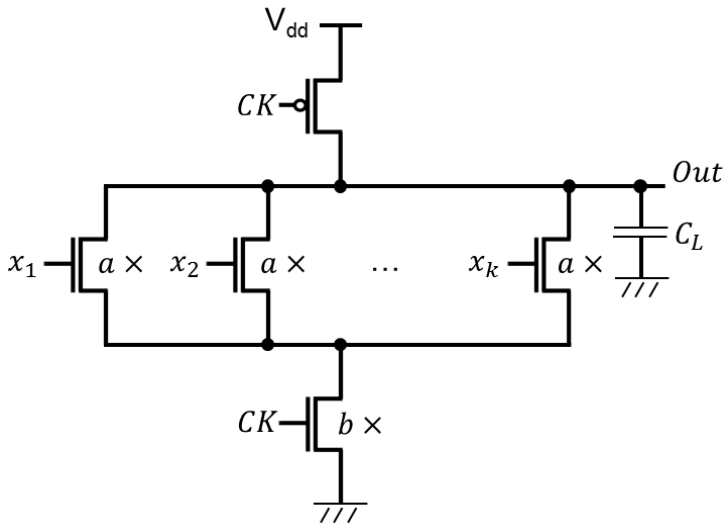
Problem #1 (CMOS gates, 10 points).

What does the following circuit do? Describe the function of the circuit in as much detail as possible (D: input, Q: output, CK: clock).



Problem #2 (Transistor Sizing Under Timing Constraints, 10 points).

Let's design a *k*-input NOR gate using the dynamic CMOS design methodology. The following shows a schematic of the *k*-input NOR gate.



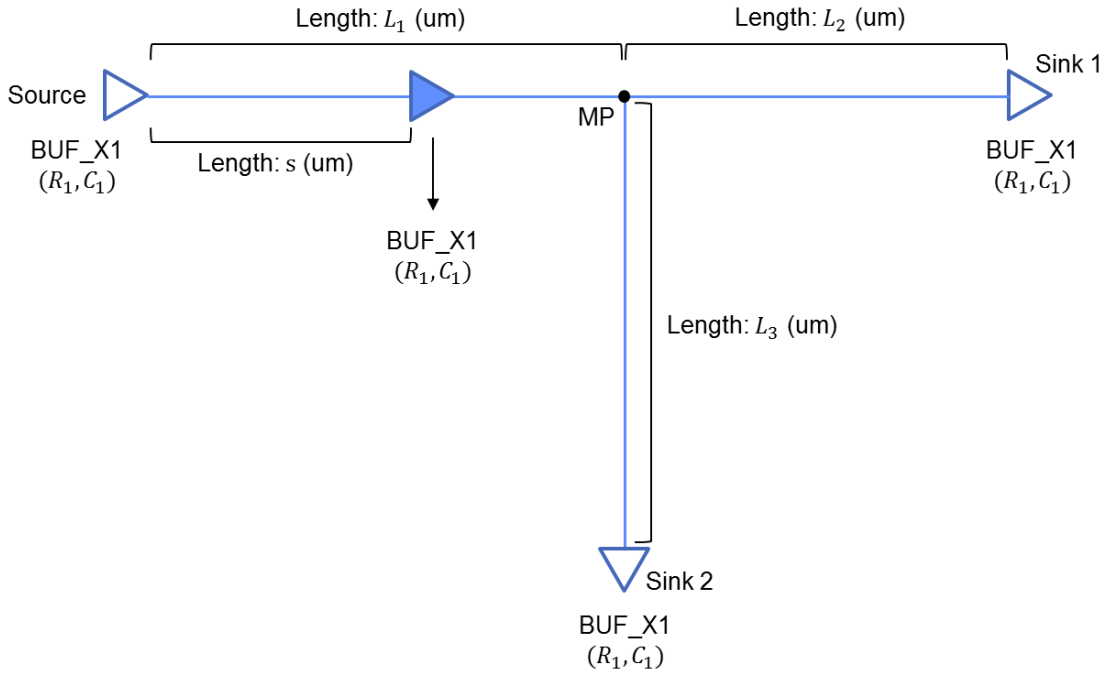
Our objective is to ***minimize*** the total width, $Width = a \cdot k + b$ and satisfy given timing constraints at the same time. Two timing constraints are given to us as follows:

- Setup time: Elmore delay $\leq\ 4 \cdot R_n \cdot C_L$
- Hold time: Elmore delay $\geq \frac{1}{4} \cdot R_n \cdot C_L$

R_n is the resistance of a 1X NMOS transistor. $\mu_n = 2 \cdot \mu_p$. Ignore all the parasitic capacitances. All the transistors for $x_1 \sim x_k$ are upsized to aX and the transistor for CK is upsized to bX (a and b are ***real*** numbers). ***Find*** a and b minimizing the total width and satisfying the timing constraints.

Answer) $a =$ $b =$

Problem #3 (Buffer Insertion, 10 points).



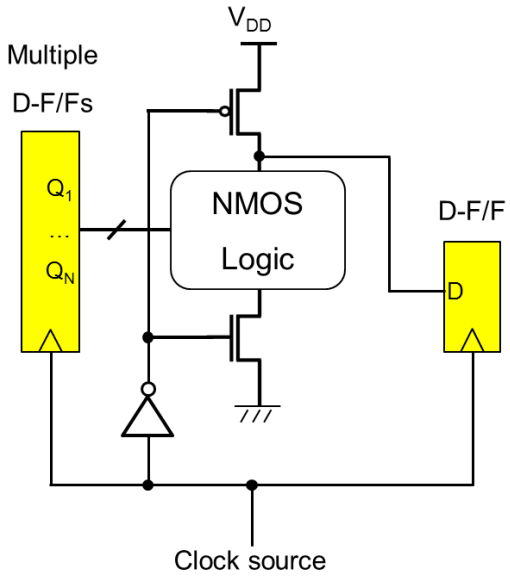
A source drives two sinks through a net and you are supposed to insert a buffer between the source and the branch point (MP) as shown in the above figure. ***Find*** an optimal location of the buffer minimizing the total delay, i.e., ***represent*** "s" as a function of the following parameters:

- Output resistance of BUF_X1: R_1
- Input capacitance of BUF_X1: C_1
- Unit wire resistance: r $(\Omega/\mu m)$
- Unit wire capacitance: c $(F/\mu m)$

Answer) $s =$

Problem #4 (Timing Analysis for Dynamic CMOS Circuits, 10 points).

The following figure shows a dynamic CMOS circuit between two pipeline stages.



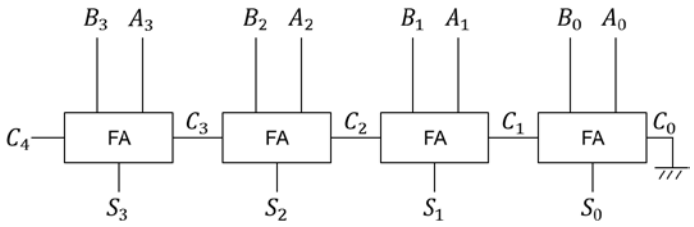
- Setup time: T_s
- Clock period: T_{CLK}
- D-F/F internal delay: T_{CQ}
- Clock skew: 0
- NMOS logic delay: T_{logic}
- Inverter delay: T_v

When the clock goes from low to high, the F/Fs capture their input signals. At the same time, the dynamic CMOS circuit starts pre-charging the output node. When the clock goes from high to low, the dynamic CMOS circuit starts evaluating its inputs. The delay of the dynamic CMOS circuit (T_{logic}) is actually the time spent to discharge the output node. *Derive* a new setup time constraint (inequality) for the dynamic CMOS circuit shown above.

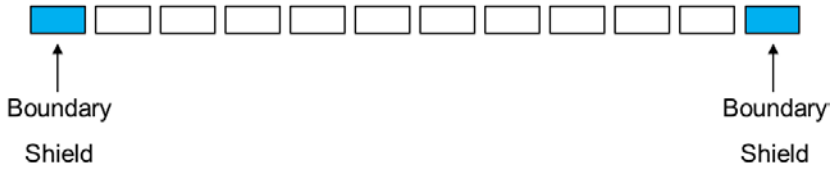
Answer)

Problem #5 (Timing Analysis and Coupling in an RCA, 10 points).

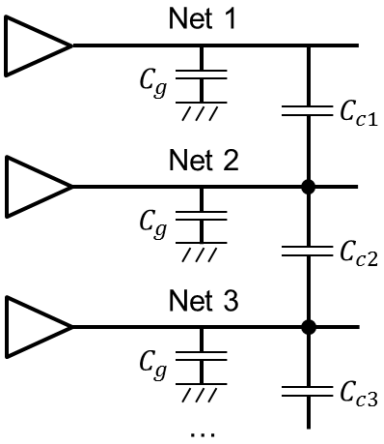
Let's design a four-bit ripple carry adder (RCA) with C_0 tied to ground as shown below.



Due to some physical design constraints, only ten routing tracks are available for the eight primary input signals as follows:



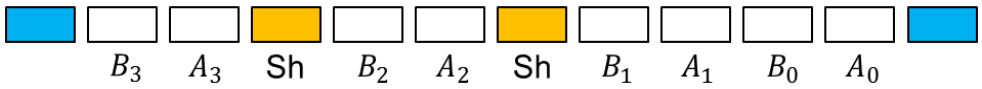
You are supposed to use ***eight*** routing tracks ***for signal*** and the other ***two*** routing tracks ***for shielding*** tied to V_{DD} or V_{SS}. Wire resistance is negligible and each wire has a ground capacitor (C_g) and a coupling capacitor as follows:



- Output resistance of the buffers driving the wires: 500Ω
- C_g: $50fF$
- C_c: $25fF$
- Delay of a full adder (from its inputs to both its Sum and Carry-Out): $40ps$
- Wire delay: $2 \cdot R \cdot C$ (where R is the output resistance of the buffer driving the net and C is the total capacitance of the wire).

Assign the eight primary input signals and the two shielding to the ten routing tracks and *compute* the delay from the primary inputs to S_3 or C_4. You should *minimize* the delay from the primary inputs to S_3 or C_4 when you assign the signals to the routing tracks. (See the next page for an example).

Example) Suppose the following is my assignment result.

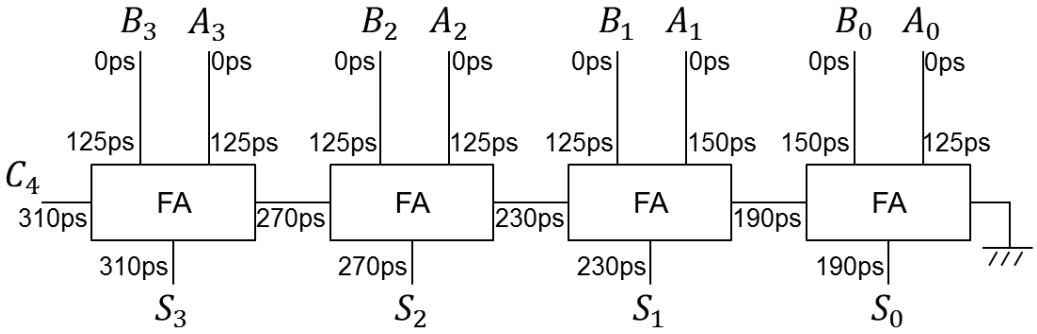


The following shows the total capacitance of each wire:

- B_3: $C_g + 3C_c$ A_3: $C_g + 3C_c$ B_2: $C_g + 3C_c$ A_2: $C_g + 3C_c$
- B_1: $C_g + 3C_c$ A_1: $C_g + 4C_c$ B_0: $C_g + 4C_c$ A_0: $C_g + 3C_c$

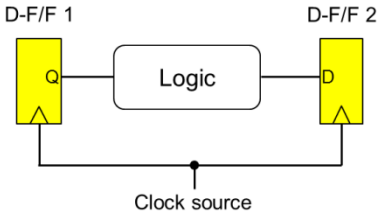
The following shows the arrival time at each node:

- $B_3, A_3, B_2, A_2, B_1, A_0$: $2RC = 2R(C_g + 3C_c) = 2 \cdot (500\Omega) \cdot (50fF + 3 \cdot 25fF) = 125ps$
- A_1, B_0: $2RC = 2R(C_g + 4C_c) = 2 \cdot (500\Omega) \cdot (50fF + 4 \cdot 25fF) = 150ps$



Thus, the delay is 310ps.

Problem #6 (Timing Analysis under PVT Variation, 10 points).



- Delay from the clock source to D-F/F 1 (and D-F/F 2): cd_1 (and cd_2)
- Setup time of the F/Fs: T_{s}
- Hold time of the F/Fs: T_{h}
- D-F/F internal delay: T_{CQ}
- Clock skew: $T_{\mathrm{skew}} = cd_2 - cd_1$
- Logic delay: T_{logic}
- Clock period: T_{CLK}

Ideally, the following inequalities should be satisfied:

1. Setup time: $T_{\mathrm{s}} \leq T_{\mathrm{CLK}} + T_{\mathrm{skew}} - T_{\mathrm{logic}} - T_{\mathrm{CQ}}$
2. Hold time: $T_{\mathrm{h}} \leq T_{\mathrm{CQ}} + T_{\mathrm{logic}} - T_{\mathrm{skew}}$

Process-voltage-temperature (PVT) variation causes serious problems such as delay variation. For example, a transistor can be faster or slower than predicted due to process variation (i.e., μ_p and μ_n change) and wire delay can be increased or decreased depending on the operating temperature. The following shows variations in the timing values due to PVT variation:

- $cd_1 \rightarrow cd_1 \pm \Delta_1$
- $cd_2 \rightarrow cd_2 \pm \Delta_2$
- $T_{\mathrm{CQ}} \rightarrow T_{\mathrm{CQ}} \pm \Delta_3$
- $T_{\mathrm{logic}} \rightarrow T_{\mathrm{logic}} \pm \Delta_4$

Derive a new setup time and a new hold time constraints (inequalities) that should be satisfied under the PVT variations. The new inequalities should consist of the following constants and variables only:

- $T_{\mathrm{s}},\ T_{\mathrm{h}},\ T_{\mathrm{CQ}},\ T_{\mathrm{CLK}},\ T_{\mathrm{skew}},\ T_{\mathrm{logic}},\ \Delta_1,\ \Delta_2,\ \Delta_3,\ \Delta_4$

Answer)

Setup time:

Hold time:

Problem #7 (High-Speed Adder, 10 points).

Compute the sum of A and B and Cin using the conditional sum adder.

- A = 65534 (1111111111111110)
- B = 13421 (0011010001101101)
- Cin = 1

$CI_0 = 1$

	i:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	A_i:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
	B_i:	0	0	1	1	0	1	0	0	0	1	1	0	1	1	0	1
Step 1	S_i^0:																
	CO_i^0:																
	S_i^1:																
	CO_i^1:																
Step 2	S_i^0:																
	CO_i^0:																
	S_i^1:																
	CO_i^1:																
Step 3	S_i^0:																
	CO_i^0:																
	S_i^1:																
	CO_i^1:																
Step 4	S_i^0:																
	CO_i^0:																
	S_i^1:																
	CO_i^1:																
Result																	

Problem #8 (Testing, 10 points).

We want to detect stuck-at-0 and stuck-at-1 faults at all the primary inputs, x_1, x_2, x_3, x_4, and the two internal nodes, a, b. Computation of Z_f to detect a stuck-at-0/1 fault at an internal node can be done by setting the value of the node to constant 0 (for stuck-at-0 faults) or 1 (for stuck-at-1 faults). Find a ***minimal*** set of test vectors that can detect all the s-a-0 and s-a-1 faults at $x_1, x_2, x_3, x_4, a,$ and b.