

EE434

ASIC and Digital Systems

Midterm Exam 1

Feb. 22, 2017. (4:10pm – 5pm)

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

Name:

WSU ID:

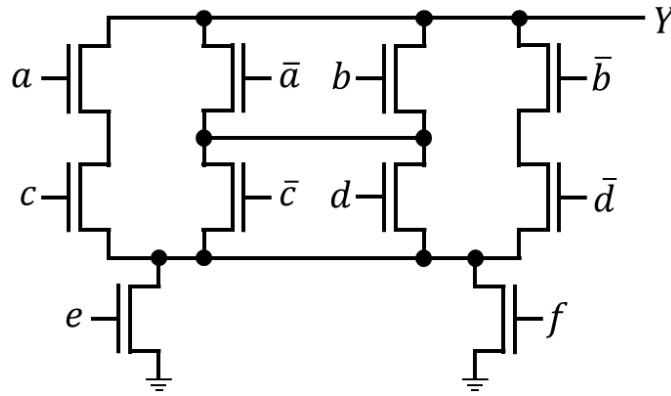
Problem	Points	
1	10	
2	10	
3	10	
4	20	
5	10	
Total	60	

* 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 (Static CMOS gates, 10 points).

The pFET network of the following nFET network is designed by the dual network of the nFET network. Represent Y as a Boolean function of a, b, c, d, e and f . (Try to simplify the Boolean function).



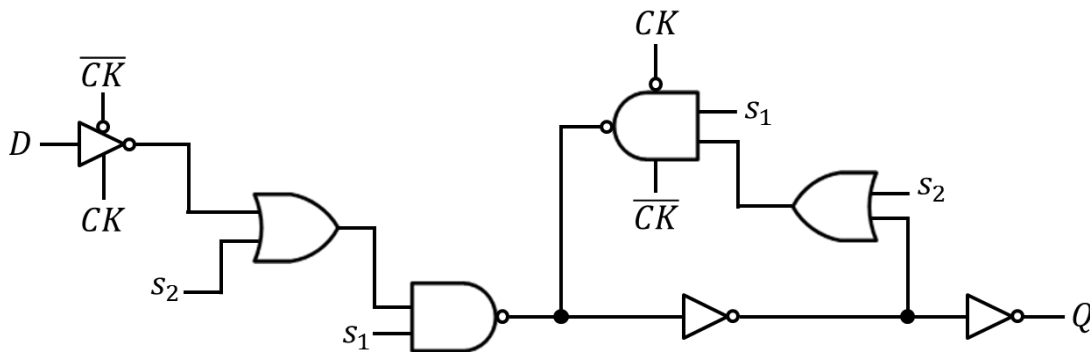
$$\begin{aligned}
 Y &= \overline{ace + acf + \bar{a}\bar{c}e + \bar{a}\bar{c}f + \bar{a}de + \bar{a}df + b\bar{c}e + b\bar{c}f + bde + bdf + \bar{b}\bar{d}e + \bar{b}\bar{d}f} \\
 &= \overline{ac(e + f) + \bar{a}\bar{c}(e + f) + \bar{a}d(e + f) + b\bar{c}(e + f) + bd(e + f) + \bar{b}\bar{d}(e + f)} \\
 &= (e + f) \cdot \overline{\{ac + \bar{a}\bar{c} + \bar{a}d + b\bar{c} + bd + \bar{b}\bar{d}\}} \\
 &= (e + f) \cdot \overline{\{a \oplus c + \bar{a}d + b\bar{c} + b \oplus d\}}
 \end{aligned}$$

Problem #2 (Analysis of CMOS gates, 10 points).

The following circuit is a sequential logic. Describe the function of the circuit in as much detail as possible (D : data input, CK : clock, s_1, s_2 : control inputs).

Note:

- Logic
 - F/F vs. Latch
 - Positive-edge triggered vs. Negative-edge triggered (for F/F)
 - Active-high vs. Active-low (for Latch)
- Control inputs
 - Asynchronous vs. Synchronous
 - Active-high vs. Active-low
 - Set, Reset, Enable, Select, Additional input
 - Dominance (e.g., Set dominates Reset, Enable dominates Set, etc.)

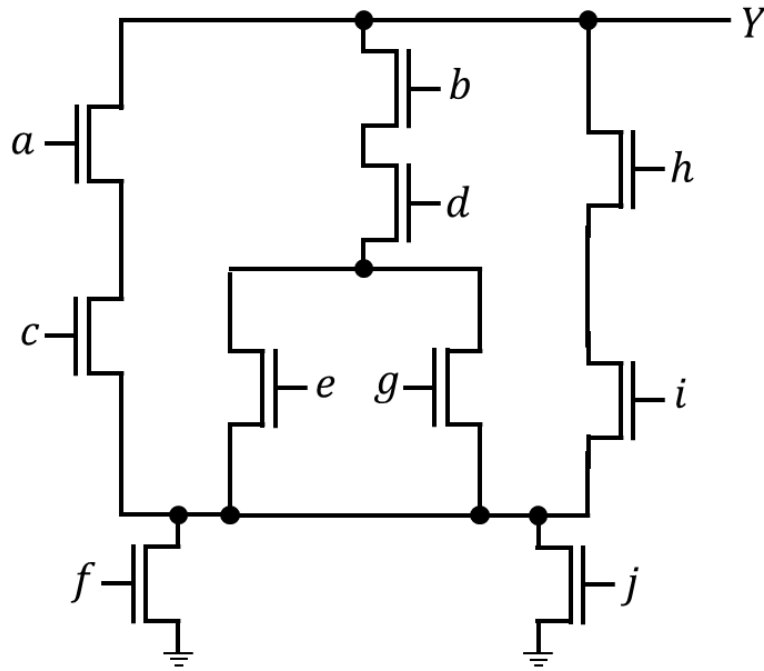


- If $s_1 = 0, Q = 1 \rightarrow s_1$ is an asynchronous active-low set.
- If $s_1 = 1$ and $s_2 = 1, Q = 0 \rightarrow s_2$ is an asynchronous active-high reset.
 - Set dominates Reset.
- When $s_1 = 1$ and $s_2 = 0$
 - If $CK = 0, Q = \text{hold}$.
 - If $CK = 1, Q = D$.

Thus, this is an active-high Latch with an asynchronous active-low set and an asynchronous active-high reset (Set dominates Reset).

Problem #3 (Transistor Sizing, 10 points).

Size the transistors in the following pull-down network. R_n is the resistance of a 1X NMOS transistor. Ignore parasitic capacitances. Target time constant: $\tau_{target} = R_n \cdot C_L$. Try to minimize the total area.



The longest path is b-d-e-f. Each of them is upsized to 4X. Then, we get a=c=h=i=8/3.

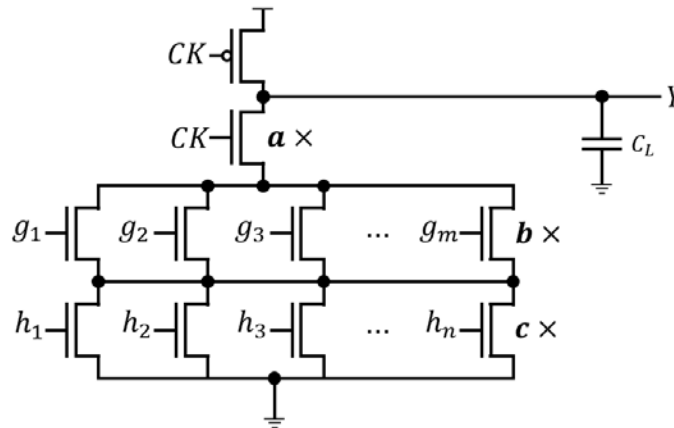
- a: 8/3X
- b: 4X
- c: 8/3X
- d: 4X
- e: 4X
- f: 4X
- g: 4X
- h: 8/3X
- i: 8/3X
- j: 4X

Total width: $(34+2/3)X$

(If we upsize a-c-f first, a=c=h=i=f=j=3X. Then, b=d=e=g=4.5X. Total width = 36X, which is worse than the above one.)

Problem #4 (Transistor Sizing, 20 points).

We want to design the following logic: $Y = \overline{(g_1 + g_2 + \dots + g_m) \cdot (h_1 + h_2 + \dots + h_n)}$ where $g_1 \sim g_m$ and $h_1 \sim h_n$ are the inputs of the gate. However, the static CMOS gate design methodology is not suitable for the design of the gate. Thus, we are going to design it using the dynamic CMOS design methodology. The following shows a schematic of the gate.



R_n is the resistance of a 1X nFET. Target time constant: $\tau_{target} = R_n \cdot C_L$. The nFET connected to CK is upsized to $a \times$, all the nFETs connected to $g_1 \sim g_m$ are upsized to $b \times$, and all the nFETs connected to $h_1 \sim h_n$ are upsized to $c \times$ to satisfy the timing constraint (a, b, c are real numbers). Ignore all the parasitic capacitance. We minimize the total width

$$Width = a + m \cdot b + n \cdot c.$$

Problem 4-1 (8 points): Find a, b , and c minimizing the total width (i.e., represent each of a, b , and c as a function of n and m).

Timing constraint: $\frac{R_n}{a} + \frac{R_n}{b} + \frac{R_n}{c} = R_n \rightarrow \frac{1}{a} + \frac{1}{b} + \frac{1}{c} = 1$. Set $a = \frac{1}{x}, b = \frac{1}{y}, c = \frac{1}{z}$. Then we minimize $F = \frac{1}{x} + \frac{m}{y} + \frac{n}{z}$ with constraint $x + y + z = 1$.

$$\frac{\partial F}{\partial x} = -\frac{1}{x^2} + \frac{n}{z^2} = 0 \rightarrow x = \frac{z}{\sqrt{n}}$$

$$\frac{\partial F}{\partial y} = -\frac{m}{y^2} + \frac{n}{z^2} = 0 \rightarrow y = \frac{\sqrt{m}z}{\sqrt{n}}$$

$$z \left(\frac{1}{\sqrt{n}} + \frac{\sqrt{m}}{\sqrt{n}} + 1 \right) = 1 \rightarrow z = \frac{\sqrt{n}}{1 + \sqrt{m} + \sqrt{n}}$$

$$x = \frac{1}{1 + \sqrt{m} + \sqrt{n}}$$

$$y = \frac{\sqrt{m}}{1 + \sqrt{m} + \sqrt{n}}$$

Thus, we obtain the following:

$$a = (1 + \sqrt{m} + \sqrt{n})X$$

$$b = \frac{1 + \sqrt{m} + \sqrt{n}}{\sqrt{m}} = \left(1 + \frac{1 + \sqrt{n}}{\sqrt{m}}\right)X$$

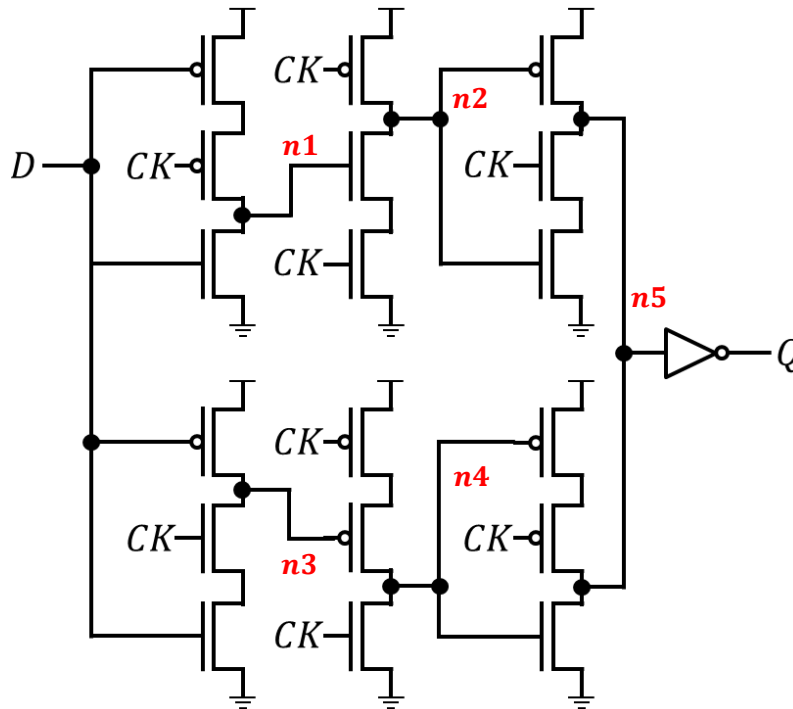
$$c = \frac{1 + \sqrt{m} + \sqrt{n}}{\sqrt{n}} = \left(1 + \frac{1 + \sqrt{m}}{\sqrt{n}}\right)X$$

Problem 4-2 (12 points, True/False. Use your intuition or the formulas you derived in the above problem).)

- If we increase n , a increases. True / False
 - If n goes up, increasing a can significantly reduce the size overhead.
- If we increase m , a increases. True / False
 - n and m are interchangeable, so if m goes up, increasing a can significantly reduce the size overhead.
- If we increase n , b increases. True / False
 - In this case (n increases), a and b can be treated the same way. Thus, b should go up to minimize the total area.
- If we increase m , b increases. (True / False)
 - Increasing m increases a and n or decrease b .
- If we increase n , c increases. (True / False)
- If we increase m , c increases. True / False)

Problem #5 (Logic Analysis, 10 points).

What does the following circuit do? Describe the function of the circuit in as much detail as possible (CK is a clock signal, D is an input, Q is an output). Note: if a node is floating, it holds its previous value.



(M. Afghahi, JSSC'91)

- 1) $CK=0, D=0$: $n3=1$, so $n4$ is floating. If $n4$ was 0, $Q=0$ (hold). If $n4$ was 1, $Q=1$ (hold). In this case, $n2=1$, but $CK=0$, so $n5$ is driven by $n4$. Q =hold.
- 2) $CK=0, D=0 \rightarrow 1$: $n3$ is floating. If $n3$ was 0, $n4$ was 1, so $Q=1$ (hold). If $n3$ was 1, $n4$ was floating, so Q is hold. $n1$ is 0, but $n2$ is 1 because $CK=0$, so $n5$ is driven by $n4$. Q =hold.
- 3) $CK=0, D=1$ or $D=1 \rightarrow 0$: Q =hold.
- 4) $CK=0 \rightarrow 1$: $n4$ is 0, so $n5$ is not driven by $n4$ anymore. If D was 1 right before the clock rising edge, $n1=0$. $n2$ =floating (but it was 1 because CK was 0). Thus, $n5$ is 0, so $Q=1$. If D was 0 right before the clock rising edge, $n1$ was 1. After the rising edge, $n2 = 0$. Thus, $n5=1$, so $Q=0$. In other words, it captures the D value at the clock rising edge (positive-edge D-FF).
- 5) $CK=1 \rightarrow 0$: $n2=1$, so $n5$ is not driven by the upper logic. $n4$ was 0 right before the clock falling edge. If D was 0 right before the clock falling edge, $n3=1$, so $n4$ is floating

(n_4 holds the previous value 0), so $n_5=1$, so $Q=0$. If D was 1 right before the clock falling edge, n_3 was 0. After the clock falling edge, n_3 is floating (holds the previous value 0). Now, $CK=0$, so $n_4=1$, so $n_5=0$, so $Q=1$. In other words, it captures the D value at the clock falling edge (negative-edge D-FF).

This is a dual-edge D-FF.