EE466

VLSI Design

Final Exam

Dec. 13, 2019. (3:10pm – 5:10pm)

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

Name:

WSU ID:

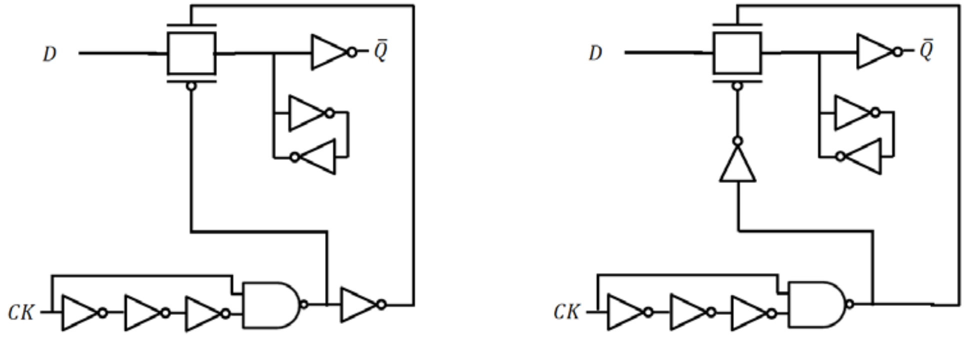
Problem	Points	
1	10	
2	10	
3	10	
4	10	
5	10	
6	20	
7	10	
8	10	
9	40	
Total	130	

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

* Not allowed: Chat apps.

Problem #1 (Sequential Logic, 10 points)

The left one is a positive-edge-triggered explicit-pulsed D flip-flop (epDFF). The right one is a negative-edge-triggered epDFF.

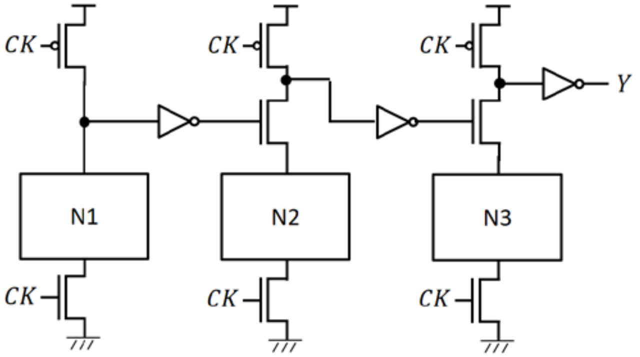


Do the two FFs have the same hold-time constraint for D=0?

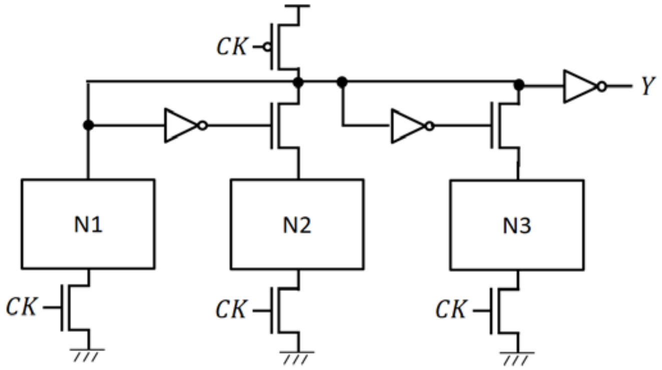
Do the two FFs have the same hold-time constraint for D=1?

Problem #2 (Domino Logic, 10 points)

The following shows a three-stage domino logic for $Y = N1 \cdot N2 \cdot N3$ (the inputs to the NFET networks are not shown). The sizes of all the PFETs and the inverters are fixed (constants). All the given timing constraints are also fixed.

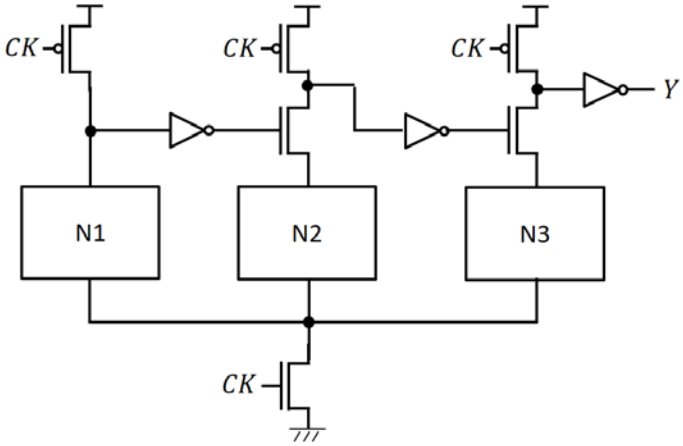


Now, we merge the PFETs into a single PFET as follows:



1) Will it work? If no, explain why. If yes, can you compare the size of the PFET in the second schematic and the sum of the sizes of the PFETs in the first schematic?

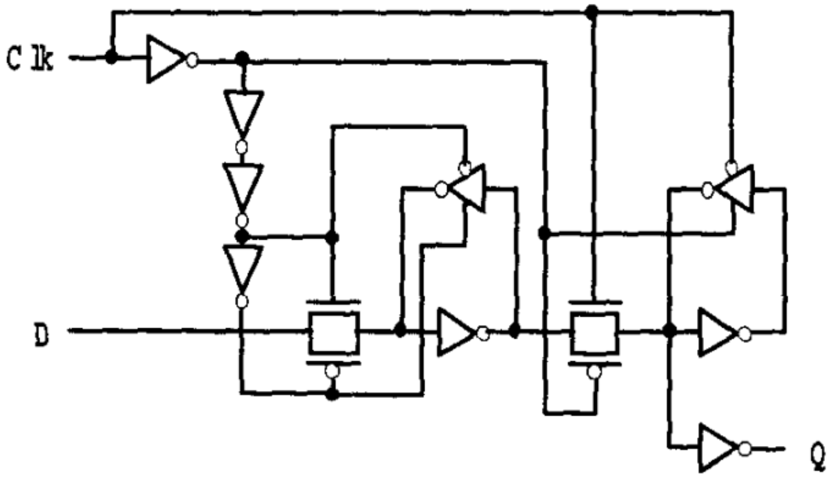
Now, let’s merge the NFETs into a single NFET as follows:



2) Will it work? If no, explain why. If yes, can you compare the size of the clock NFET in the third schematic and the sum of the sizes of the clock NFETs in the first schematic?

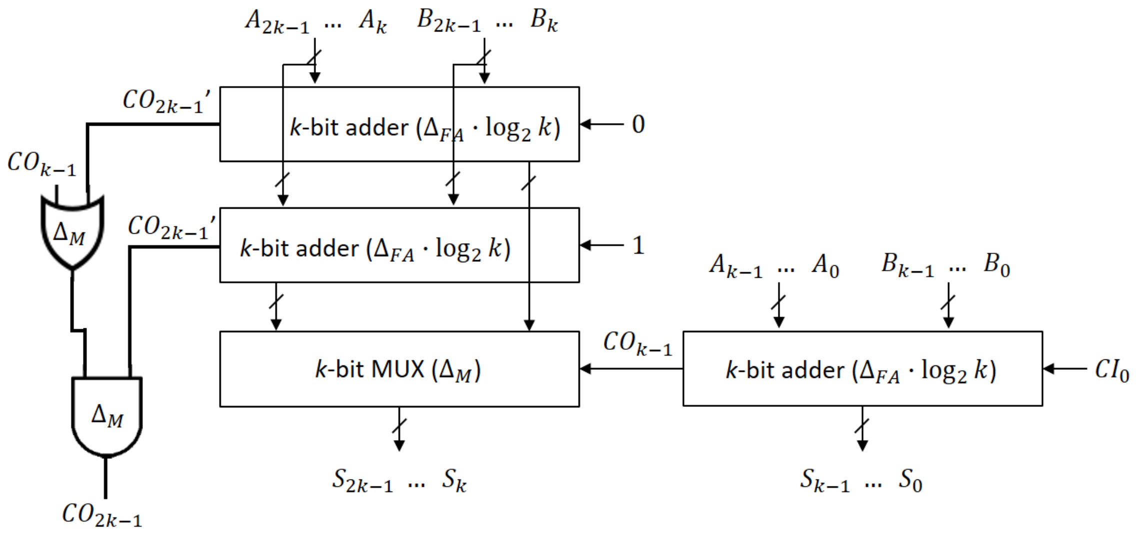
Problem #3 (Sequential Logic, 10 points)

The following shows a schematic of a D-F/F. Estimate the hold time constraint of the F/F (for example, “one inverter delay + one transmission gate delay”).



Problem #4 (Carry Select Adder, 10 points)

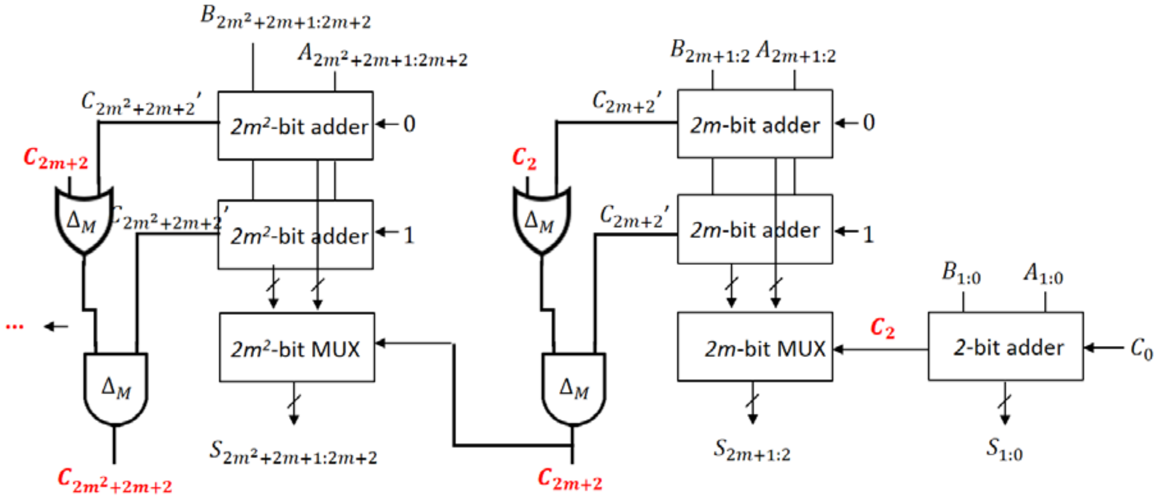
The following shows a schematic of a *2k*-bit carry select adder designed using *k*-bit parallel adders. The delay of a *k*-bit adder is $\Delta_{FA} \cdot \log_2 k$, the delay of a *k*-bit MUX is Δ_M, and the delay of a two-input AND (or OR) gate is Δ_M.



We are supposed to design an *N*-bit adder using carry select adders (# groups: $\frac{N}{k}$). Find k minimizing the delay of the *N*-bit adder (express the optimal k as a function of *N*, Δ_M, and Δ_{FA}). Notice that the worst-case delay occurs at C_N (the final carry out) or $S_{N-1:0}$ (the final sum). Just a small math hint: $\log_2 k = \frac{\ln k}{\ln 2}, \frac{d(\ln x)}{dx} = \frac{1}{x}$.

Problem #5 (Carry Select Adder, 10 points)

To radically improve the delay of a carry-select adder, we design an N-bit carry-select adder as follows:



We design the k-bit adder using a k-bit parallel adder where k is $2m^i$ (i is an integer greater than or equal to 0).

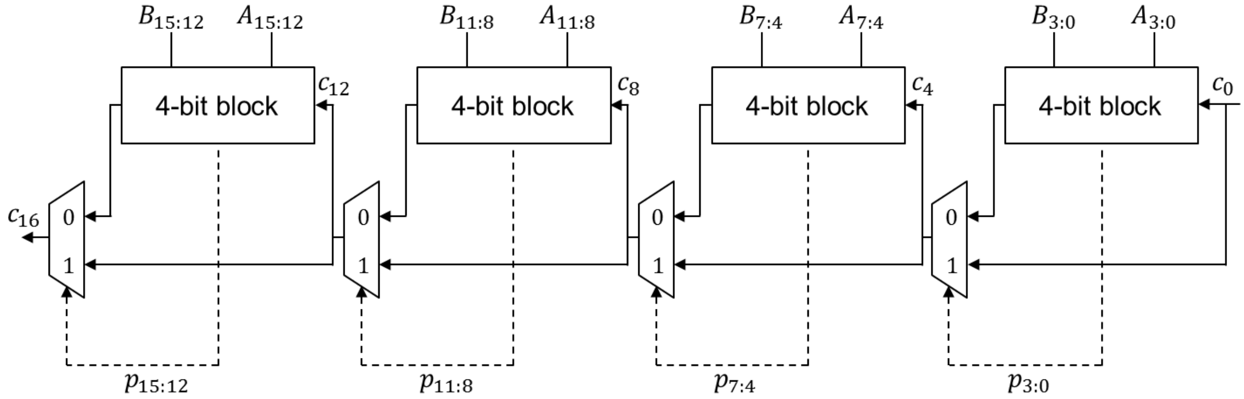
The following shows the delays of the components:

- k-bit adder: $\Delta_{FA} \cdot \log_2 k = \Delta_{FA} \cdot \log_2(2m^i) = \Delta_{FA} \cdot (1 + i \cdot \log_2 m)$
- k-bit MUX: Δ_M
- $\Delta_{FA} > 2 * \Delta_M$

Express the delay of the last carry out C_N using N, m, Δ_{FA}, and Δ_M.

Problem #6 (Carry Skip Adder, 20 points)

The following diagram shows a 16-bit carry-skip adder designed using 4-bit adders.



1) Show the details of the calculations in the carry skip adder for the following inputs (fill in the blanks).

$$A = 0101\ \ 0101\ \ 0101\ \ 0101$$

$$B = 1010\ \ 1010\ \ 1010\ \ 1010$$

$$C_0 = 1$$

Step 1)

[3:0] $g_0 =$ ___. $g_1 =$ ___. $g_2 =$ ___. $g_3 =$ ___. $p_0 =$ ___. $p_1 =$ ___. $p_2 =$ ___. $p_3 =$ ___.

[7:4] $g_4 =$ ___. $g_5 =$ ___. $g_6 =$ ___. $g_7 =$ ___. $p_4 =$ ___. $p_5 =$ ___. $p_6 =$ ___. $p_7 =$ ___.

[11:8] $g_8 =$ ___. $g_9 =$ ___. $g_{10} =$ ___. $g_{11} =$ ___. $p_8 =$ ___. $p_9 =$ ___. $p_{10} =$ ___. $p_{11} =$ ___.

[15:12] $g_{12} =$ ___. $g_{13} =$ ___. $g_{14} =$ ___. $g_{15} =$ ___. $p_{12} =$ ___. $p_{13} =$ ___. $p_{14} =$ ___.
$p_{15} =$ ___.

Step 2)

[3:0] $g_{1:0} =$ ___. $p_{1:0} =$ ___. $g_{2:0} =$ ___. $p_{2:0} =$ ___. $g_{3:0} =$ ___. $p_{3:0} =$ ___.

[7:4] $g_{5:4} =$ ___. $p_{5:4} =$ ___. $g_{6:4} =$ ___. $p_{6:4} =$ ___. $g_{7:4} =$ ___. $p_{7:4} =$ ___.

[11:8] $g_{9:8} =$ ___. $p_{9:8} =$ ___. $g_{10:8} =$ ___. $p_{10:8} =$ ___. $g_{11:8} =$ ___. $p_{11:8} =$ ___.

[15:12] $g_{13:12} =$ ___. $p_{13:12} =$ ___. $g_{14:12} =$ ___. $p_{14:12} =$ ___. $g_{15:12} =$ ___. $p_{15:12} =$ ___.

Step 3)

[3:0] $c_1 =$ ___. $c_2 =$ ___. $c_3 =$ ___.

$c_4 =$ ___.

Step 4)

[3:0] $s_0 =$ ___. $s_1 =$ ___. $s_2 =$ ___. $s_3 =$ ___.

[7:4] $c_5 =$ ___. $c_6 =$ ___. $c_7 =$ ___.

$c_8 =$ ___.

Step 5)

[7:4] $s_4 =$ ___. $s_5 =$ ___. $s_6 =$ ___. $s_7 =$ ___.

[11:8] $c_9 =$ ___. $c_{10} =$ ___. $c_{11} =$ ___.

$c_{12} =$ ___.

Step 6)

[11:8] $s_8 =$ ___. $s_9 =$ ___. $s_{10} =$ ___. $s_{11} =$ ___.

[15:12] $c_{13} =$ ___. $c_{14} =$ ___. $c_{15} =$ ___.

$c_{16} =$ ___.

Step 7)

[15:12] $s_{12} =$ ___. $s_{13} =$ ___. $s_{14} =$ ___. $s_{15} =$ ___.

2) Repeat it for the following inputs.

$$A = 0101\ \ 0101\ \ 0101\ \ 0101$$

$$B = 1010\ \ 1010\ \ 1110\ \ 1010$$

$$C_0 = 1$$

Step 1)

[3:0] $g_0 =$ ___. $g_1 =$ ___. $g_2 =$ ___. $g_3 =$ ___. $p_0 =$ ___. $p_1 =$ ___. $p_2 =$ ___. $p_3 =$ ___.

[7:4] $g_4 =$ ___. $g_5 =$ ___. $g_6 =$ ___. $g_7 =$ ___. $p_4 =$ ___. $p_5 =$ ___. $p_6 =$ ___. $p_7 =$ ___.

[11:8] $g_8 =$ ___. $g_9 =$ ___. $g_{10} =$ ___. $g_{11} =$ ___. $p_8 =$ ___. $p_9 =$ ___. $p_{10} =$ ___. $p_{11} =$ ___.

[15:12] $g_{12} =$ ___. $g_{13} =$ ___. $g_{14} =$ ___. $g_{15} =$ ___. $p_{12} =$ ___. $p_{13} =$ ___. $p_{14} =$ ___.
$p_{15} =$ ___.

Step 2)

[3:0] $g_{1:0} =$ ___. $p_{1:0} =$ ___. $g_{2:0} =$ ___. $p_{2:0} =$ ___. $g_{3:0} =$ ___. $p_{3:0} =$ ___.

[7:4] $g_{5:4} =$ ___. $p_{5:4} =$ ___. $g_{6:4} =$ ___. $p_{6:4} =$ ___. $g_{7:4} =$ ___. $p_{7:4} =$ ___.

[11:8] $g_{9:8} =$ ___. $p_{9:8} =$ ___. $g_{10:8} =$ ___. $p_{10:8} =$ ___. $g_{11:8} =$ ___. $p_{11:8} =$ ___.

[15:12] $g_{13:12} =$ ___. $p_{13:12} =$ ___. $g_{14:12} =$ ___. $p_{14:12} =$ ___. $g_{15:12} =$ ___. $p_{15:12} =$ ___.

Step 3)

[3:0] $c_1 =$ ___. $c_2 =$ ___. $c_3 =$ ___.

$c_4 =$ ___.

Step 4)

[3:0] $s_0 =$ ___. $s_1 =$ ___. $s_2 =$ ___. $s_3 =$ ___.

[7:4] $c_5 =$ ___. $c_6 =$ ___. $c_7 =$ ___.

$c_8 =$ ___.

Step 5)

[7:4] $s_4 =$ ___. $s_5 =$ ___. $s_6 =$ ___. $s_7 =$ ___.

[11:8] $c_9 =$ ___. $c_{10} =$ ___. $c_{11} =$ ___.

$c_{12} =$ ___.

Step 6)

[11:8] $s_8 =$ ___. $s_9 =$ ___. $s_{10} =$ ___. $s_{11} =$ ___.

[15:12] $c_{13} =$ ___. $c_{14} =$ ___. $c_{15} =$ ___.

$c_{16} =$ ___.

Step 7)

[15:12] $s_{12} =$ ___. $s_{13} =$ ___. $s_{14} =$ ___. $s_{15} =$ ___.

Problem #7 (Prefix Adder, 10 points)

Use the following delay values:

- AND, OR, XOR: Δ
- Two-level (sum-of-product) logic: 2Δ
- $g_i = a_i \cdot b_i$, $p_i = a_i \oplus b_i$

We are designing a 1024-bit Kogge-Stone adder.

Represent s_{885} hierarchically using group-generated and group-propagated carries ($g_{i:k}$, $p_{i:k}$) and c_0 (primary carry-in), then compute the delay to compute s_{885} assuming all the primary input signals are available at time 0 (10 points).

Problem #8 (Prefix Adder, 10 points)

Use the following delay values:

- AND, OR, XOR: Δ
- Two-level (sum-of-product) logic: 2Δ
- $g_i = a_i \cdot b_i, p_i = a_i \oplus b_i$

We are designing a 1024-bit adder, which is similar to the Kogge-Stone adder. However, we will design the adder as follows.

- Step 0: Compute g_i and p_i.
- Step 1: Instead of generating $g_{i:i-1}, p_{i:i-1}$ for each i by merging g_i, p_i and g_{i-1}, p_{i-1}, generate $g_{i:i-3}, p_{i:i-3}$ for each i (except $i = 0,1,2$. For i=1, merge g_1, p_1, g_0, p_0. For i=2, merge $g_2, p_2, \dots, g_0, p_0$) by merging $g_i, p_i, g_{i-1}, p_{i-1}, g_{i-2}, p_{i-2}, g_{i-3}, p_{i-3}$.
- Step 2: Generate $g_{i:i-15}, p_{i:i-15}$ for each i by merging $g_{i:i-3}, p_{i:i-3}, g_{i-4:i-7}, p_{i-4:i-7}, g_{i-8:i-11}, p_{i-8:i-11}, g_{i-12:i-15}, p_{i-12:i-15}$. Notice that this cannot be applied to $i = 0, \dots, 14$. However, you can generate $g_{i:0}, p_{i:0}$ for them properly.
- Step 3: Generate $g_{i:i-63}, p_{i:i-63}$ for each i.
- Repeat.

Represent s_{885} hierarchically using group-generated and group-propagated carries ($g_{i:k}$, $p_{i:k}$) and c_0 (primary carry-in), then compute the delay to compute s_{885} assuming all the primary input signals are available at time 0 (10 points).

Problem #9 (Carry Look-Ahead Adder, 40 points)

The max. fanout is 4. Use the following delay values:

- AND, OR, XOR: Δ
- Two-level (sum-of-product) logic: 2Δ
- $g_i = a_i \cdot b_i, p_i = a_i \oplus b_i$

We are designing a 2048-bit carry look-ahead adder.

1) Represent s_{885} hierarchically using group-generated and group-propagated carries ($g_{i:k}, p_{i:k}$) and c_0 (primary carry-in), then compute the delay to compute s_{885} assuming all the primary input signals are available at time 0 (10 points).

2) Represent s_{2019} hierarchically using group-generated and group-propagated carries ($g_{i:k}$, $p_{i:k}$) and c_0 (primary carry-in), then compute the delay to compute s_{2019} assuming all the primary input signals are available at time 0 (10 points).

3) Show details of the calculations in the 16-bit carry-lookahead adder for the following inputs (show g_i, p_i, group generation/propagation bits, carry signals, etc.) Draw some block diagrams too for the carry-lookahead units and show the values of the signals.

$$A = 0101\ \ 0101\ \ 0101\ \ 0101$$

$$B = 1010\ \ 1010\ \ 1010\ \ 1010$$

$$C_0 = 1$$

You can show the details like the one in Problem 6.

4) Repeat it for the following inputs.

$$A = 0101\ \ 0101\ \ 0101\ \ 0101$$

$$B = 1010\ \ 1010\ \ 1110\ \ 1010$$

$$C_0 = 1$$