UNIVERSITY OF CALIFORNIA
College of Engineering
Department of Electrical Engineering and Computer Sciences

Last modified on October 14, 2000 by Hanching Fuh (hfuh@eecs.berkeley.edu)

Borivoje Nikolic *Homework #5 Solutions* *EECS 141*

Problem #1

A two stage buffer is used to drive a metal wire of 1 cm. The first inverter is a minimum size with an input capacitance C_i=10 fF and a propagation delay t_{p0}=175 ps when loaded with an identical gate. The width of the metal wire is 3.6 µm. The sheet resistance of the metal is 0.08 Ω/ÿ, the capacitance value is 0.03 fF/µm^2 and the fringing field capacitance is 0.04 fF/µm.

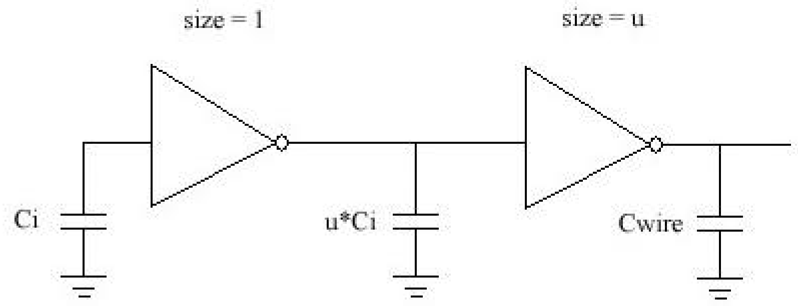
a. What is the propagation delay of the metal wire?

   R = 0.08 Ω/ÿ * (1cm/3.6µm) = 222 Ω
   C_{pp} = 0.03 fF/µm^2 * (3.6 µm) * (10000 µm) = 1.08 pF
   C_{fringe} = 2 * 0.04 fF/µm * 10000 µm = 0.80 pF
   $C_{wire} = C_{pp} + C_{fringe}$ = 1.88pF

   Assuming the lumped model,
   $t_p = 0.69 * R * C_{wire}$ = **288 ps**
   or assuming the distributed RC model,
   $t_p = 0.38 * R * C_{wire}$ = **159 ps**

b. Compute the optimal size of the second inverter as to minimize the total delay. What is this minimum delay through the buffer?

   The circuit given in this part of the problem looks like the following:

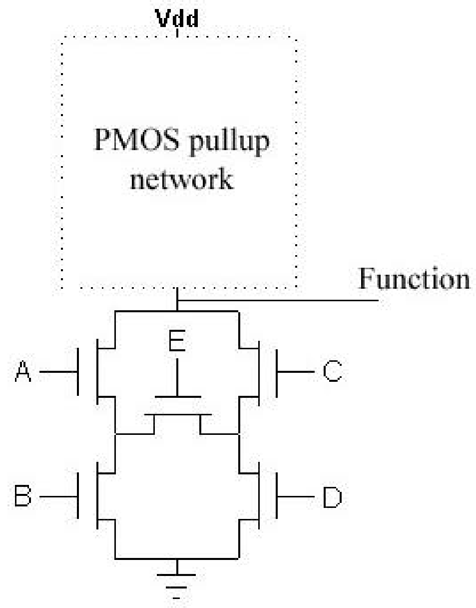


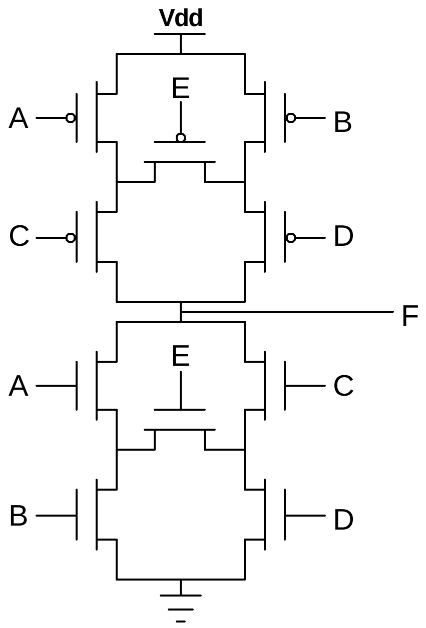
To find the minimum delay, we want
$t_{p0}/(u*C_i) + t_{p0}/C_{wire}$ = minimum

=> u = sqrt (C_{wire}/C_i) = 13.7

Thus, the second buffer should be 13.7X larger than the first buffer.
$t_{p,buffer} = 2 * u * t_{p0} = 4.8$ ns

Problem #2



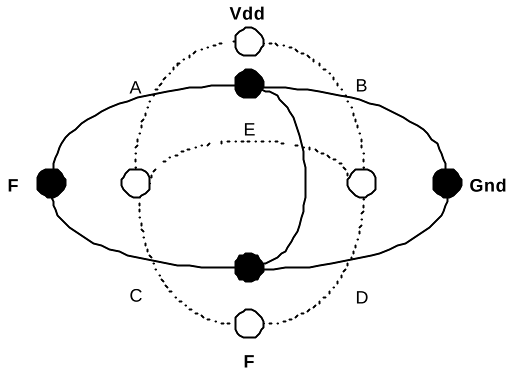


The figure to the left is a logic gate we want to implement in complementary CMOS. As you can see, the gate is already half drawn for you. We’ve given you the NMOS pull-down network. You’ll have to figure out the PMOS pull-up network!

a. What is the logic function of this gate? (F = ???)

$F = (AB + CD + AED + CEB)'$ or $F = A'C' + A'E'D' + B'D' + B'E'C'$
You can do this by considering the paths to ground and inverting or the paths to VDD.

b. Using the methods described in the text in the lecture notes, draw the Euler diagram for both PMOS and NMOS networks.



The Euler diagram to the left shows the NMOS network drawn in solid black (and rotated 90 degrees CCW) and the PMOS network drawn in white and with dotted lines.

c. Using your Euler path diagram as a guide, draw the transistor schematic for the PMOS network.

The PMOS network is shown above at the top of the problem.

Problem #3

An exclusive-OR (XOR) gate is a very important building block for many digital components (i.e. adders, etc.). We would like to implement it in complementary CMOS.

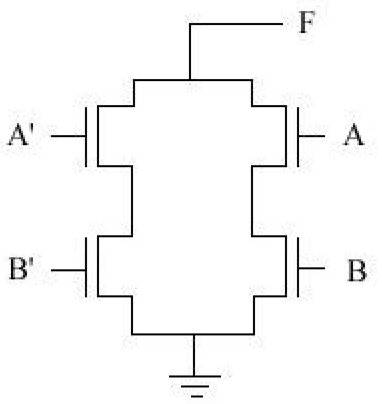
a. Write down the function table for an XOR and the function (F=???).

A	B	Out
0	0	0
0	1	1
1	0	1
1	1	0

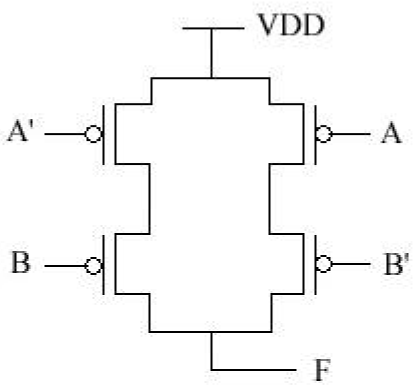
F = A’B + AB’ or F = (A’B’ +AB)’

b. Looking at the function table, implement EITHER the NMOS or PMOS network. You can even do both if you’d like!

NMOS implementation

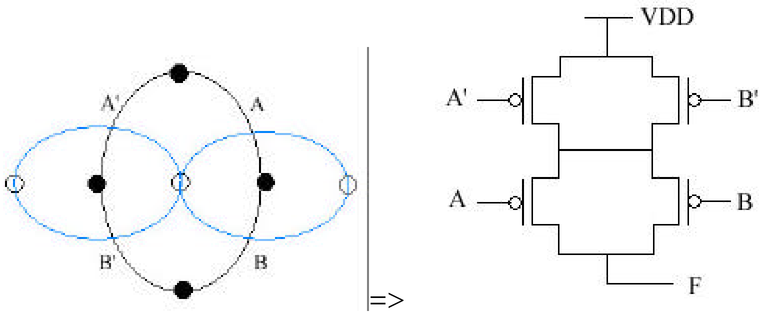


PMOS implementation

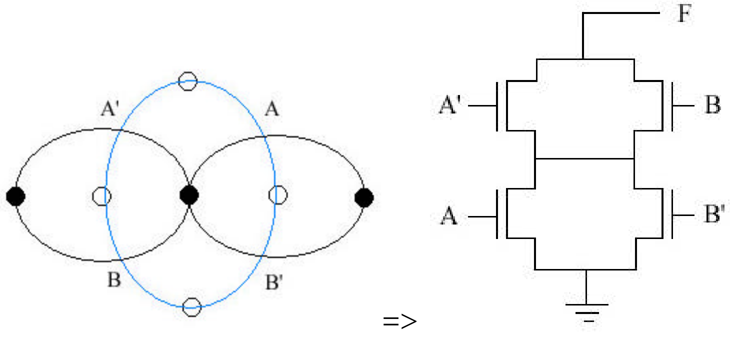


c. Using an Euler diagram, draw out the full complementary CMOS implementation of the XOR. Don’t forget any transistors! (As you can see, complementary CMOS is not an efficient way to implement the XOR. We will teach you better ways to do this soon!)

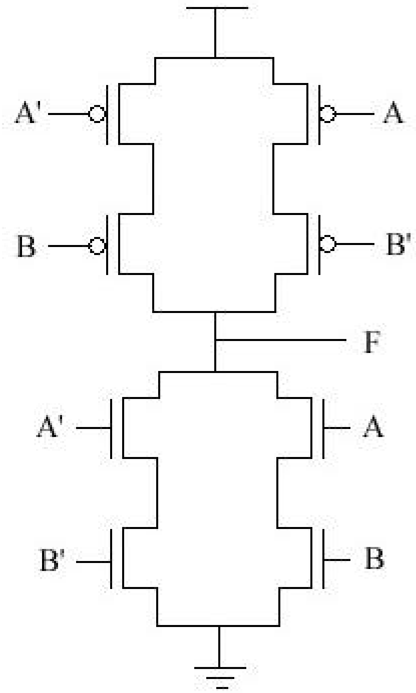
Euler diagram to get PMOS from NMOS:



Euler diagram to get NMOS from PMOS:



d. Notice, one path is unnecessary in the XOR. Get rid of it and redraw the XOR to reflect this change. Technically, this implementation is not complementary CMOS, but it is close!



The key to this is to note that in either implementation, the paths that include A’ and A or B’ and B are unnecessary since A’ ? A and B’ ? B at any point in time. Thus, the PMOS and NMOS networks in the final implementation are the “originals” of each implementation above. This is shown to the left. Technically, you should draw in the inverters to invert A and B, but we skip this since we assume you know to draw these. Hopefully you did, since the reader will probably dock you points for forgetting!

Problem #4

Suppose the EE 141 TAs are your future bosses at work. We are rather arbitrary people and since we are at a low point in the business cycle, we have decided to give all our employees (EE 141 students) a pop quiz. We have decided we would like you to design a three input NOR gate with the same average t_{phl} as a 2.5V, 0.25 µ technology inverter.

The inverter has W_{PMOS} = 2.5 μ and a symmetrical output swing. Both the inverter and the three input NOR will be driving a 50fF load (ignore other capacitances), but we want to implement the NOR with ratioed logic (Figure 6.24, section 6.2.2). The PMOS width in the NOR gate is the same as the inverter. You are given that all input combinations are equally probable and that you must calculate your answers using the unified model and parameters given in table 3.2 of the reader.

a. Calculation: What are the sizes of the NMOS transistors in the NOR gate to achieve the same average t_{phl} as the inverter?

First off, you need to calculate the W/L for the NMOS transistor with the information given. We know that V_M = 1.25V because of the symmetrical output swing. Thus,

$k_n' * (W/L)_n * ((V_M - V_{T,n})*V_{min,n} – V_{min,n}^2/2)*(1+?_n V_M)$
$= -k_p' * (W/L)_p * ((-V_M - V_{T,p})*V_{min,p} – V_{min,p}^2/2)*(1+?_p V_M)$

Note that the PMOS side has a few negative signs that differ from the NMOS side. This just accounts for the way we assign the source and drain terminals. Therefore, as an example, $V_{DS} = V_M$ for the NMOS, while $V_{DS} = -V_M$ for the PMOS. If you aren't convinced, plug in the numbers yourself!

115e-6 * $(W/L)_n$ * ((1.25V-0.43V)*0.63V-$(0.63V)^2$/2) * (1+0.06V^{-1}*1.25V)
= -(-30e-6) * (10) * ((-1.25V-(-0.4V))*(-0.85V)-$(-0.85V)^2$/2) * (1-0.1V*-1.25V)
You can solve and get **$(W/L)_n$ = 3.1**

You need to then solve for the propagation delay, t_{pHL}. We'll use equation 5.21 on page 166 of the reader.
t_{pHL} = 0.52 * $(C_L V_{DD})/((W/L)_n * k_n' * V_{DSATn} * (V_{DD} – V_{Tn} – V_{DSATn}/2))$ = **165 ps**

Finally, we can stop screwing around with the inverter and start fiddling with the three input pseudo-NMOS NOR. How might we size the transistors we ask? The difference between the pseudo-NMOS and the CMOS inverter in regards to timing is that there is a significant PMOS current that exists when the NMOS is on. This is the case for t_{pHL} in our NOR. Thus, we can modify equation 5.21 from the reader to get the following: t_{pHL} = 0.69 * (3/4) * $(C_L V_{DD})/(I_{DSATn} – I_{DSATp})$.

Essentially, the PMOS current counteracts the NMOS current when it is pulling down. We are finding the resistance at $3/4V_{DD}$, and using 0.69 $R_{eq}*C_L$ when we use this equation (5.21). This is approximate, but is by far the simplest approximation we can make for our crude hand calculations and still be close. Note, we are assuming that V_{end} for the t_{pHL} transition is = 1.25V (by definition).

We can calculate
I_{Dn} = 115e-6 * $(W/L)_n$ * ((2.5V-0.43V)*0.63V-$(0.63V)^2$/2) * (1+0.06V^{-1} * 1.875V)
= **$(W/L)_n$ * 0.141mA**

I_{Dp} = -(-30e-6) * (10) ((-2.5V-(-0.63V))*(-0.625)-$(-0.625)^2$/2)*(1- 0.1V^{-1} * (-0.625V))
= **0.31 mA**

Since we would like t_{pHL} for this gate to equal the t_{pHL} of the inverter, we need I_{Dn} – I_{Dp} = $I_{Dn,original}$ = 0.394 mA, where $I_{Dn,original}$ is the current at (3/4)V_{DD} in the inverter.

Thus, $(W/L)_n$ * 0.141mA – 0.31 mA = 0.394 mA

$(W/L)_n$ = 5. Of course, this is a “composite” (W/L). We need to average this over the switching possibilities.

Input A	Input B	Input C	Output	# NMOSs on
0	0	0	1	0
0	0	1	0	1
0	1	0	0	1
0	1	1	0	2
1	0	0	0	1
1	0	1	0	2
1	1	0	0	2
1	1	1	0	3

We gave you that all inputs are equally likely. Thus, we have 7 ways that t_{pHL} can occur. Three ways will occur with 1 NMOS transistors on, three will occur with 2 NMOS transistors, and one will occur with all 3 on. Thus, if we specify that all three pulldowns are the same size (a sensible thing to do unless you want to do a lot more calculation -> I don’t!), then to find the (W/L) of each NMOS:

3 occurrences * (W/L) + 3 occurrences * 2(W/L) + 1 occurrence * 3(W/L) = 5
$(W/L)_n$ = 2.92 for each NMOS. Wow, that’s a long calculation! (Longer than I expected when I created this problem!!!

b. How many truly “unique” ways to pull down the output are there? Write HSPICE code for your NOR gate, simulate the t_{phl} of these unique cases, and graph the results on the same axis. If you’re curious, you can simulate all the possibilities, but please don’t hand in a graph of all eight!

There are three unique ways to pull down the output, when one NMOS turns on, when two NMOS turn on, and when all three NMOS turn on. Barring second order effects, all three 1-input t_{pHL}’s should be the same and all three 2-input t_{pHL}’s should be the same.

Insert HSPICE code here… Maybe I’ll do it, maybe I won’t…

c. Would the t_{plh} of the NOR be slower, faster, or the same as the inverter? Why?

This is an acceptable answer (since it is what I thought at first): The t_{pLH} should be slower than the t_{pLH} of the inverter because of the extra parasitic capacitance present at the output node from the extra NMOS transistors.

Actually, this can be the right answer (from simulations) based on how you define t_p: t_{pLH} will be faster because it swings a smaller voltage (V_{OL} does not equal 0!).

So whether you get this problem right depends completely on how you explained it!

d. Calculate the V_{OH} and V_{OL} of the NOR gate. How many V_M's exist for the NOR gate? Please calculate the lowest V_M.

V_{OH} = 2.5V since when the input is low, it "looks" just like an inverter (all inputs will be low in this case).

To calculate V_{OL}, you must consider three cases, when 1, 2, and 3 NMOS transistors are on, respectively. Therefore, there are three V_{OL}'s for W/L = 1 x 2.92, 2 x 2.92, and 3 x 2.92. The approach, is as always, to set the currents $I_{Dn} = I_{Dp}$.

$k_n' * (W/L)_n * ((V_{DD}-V_{Tn})*V_{OL} - V_{OL}^2/2)*(1+?_n V_{OL}) =$
$k_p' * (W/L)_p * ((-V_{DD}-V_{Tp})*(V_{DSATp})-(V_{DSATp})^2/2)*(1+?_p(V_{OL}-V_{DD}))$

V_{OL} = 0.985V for W/L = 2.92 => This is actually wrong, since V_{OL} will be > than V_{DSAT} and therefore cannot be solved in the unified model (note: the V_{OL} present in the channel length modulation part is not good enough to get an accurate solution!). **V_{OL} = 1.21V** using the long channel equation. Basically, for that one part, ANY answer will be marked correct as long as you set it up correctly!

V_{OL} = 0.455V for W/L = 5.84
V_{OL} = 0.297V for W/L = 8.76

For the same reason that there are three V_{OL}'s, there will be **three** V_M's.

The lowest V_M will occur when all three NMOS transistors turn on and the equivalent W/L = 8.76

V_M = 1.51V
In case you're wondering, with a W/L = 2.92 (one NMOS on), V_M = 2.06V!

e. Problem 6.4: Given the choice between NOR or NAND logic, which one would you prefer for implementation in pseudo-NMOS. Why?

NOR logic would be more sensible to implement in pseudo-NMOS logic since you will have a pull-down network of parallel transistors as opposed to a chain of transistors in series. A design in NOR logic would therefore most likely be faster than the NAND logic.