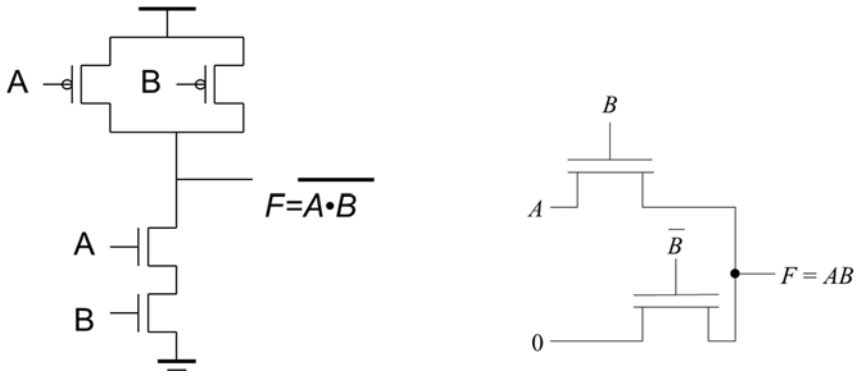
NAME_____

EN1600: Design and Implementation of VLSI Systems

Homework Assignment #1

Due **Wednesday, September 28, 2016, in class**

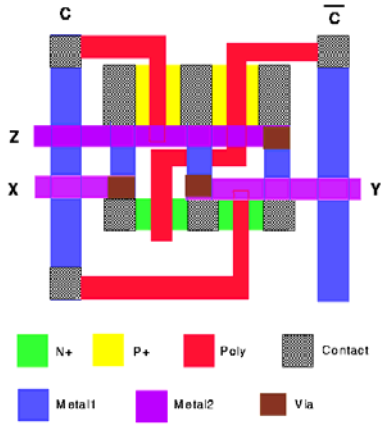
1) As we discussed in class, a NAND function can be implemented in CMOS using 4 transistors, 2 NMOS and 2 PMOS, as shown below in Figure 1(a). To generate a AND function would then require feeding the output signal *F* to an inverter. Now consider an alternative implementation of an AND function, as shown in Figure 1(b), which consists of 2 NMOS transistors. Is this a good circuit idea? Why or why not? If not, how would you suggest improving it?



(a) (b)

Figure 1.

2) Draw the transistor schematic representing the circuit below. Try to describe the Boolean function.



The remaining problems all involve schematic drawing, laying out and simulating some simple gates using Cadence.

For these problems, name your libraries according to the specific naming convention discussed in class. Specifically, each homework assignment should have its own self-contained library names with:
<your name>_<assignment>.
For example, for this assignment, I would save my library as "IrisBahar_Lab1". Use your name as it is listed on your Brown email address.

You will also need to hand in the printouts of the schematics, the SPECTRE simulation input/output waveforms, the results of the DRC and LVS run and the answers to the questions. Note that the waveforms can be printed out from the Virtuoso Visualization & Analysis (ViVA) tool by selecting **File > Save Image …** Please be sure you set options to print on a WHITE background (check "Replace background color with:" and choose white).

3) Using the Schematic Editor tool, build a transistor-level schematic of a CMOS inverter, 2-input NAND gate and 2-input NOR gate using the 4-terminal PMOS and NMOS symbols. Start with the tutorial "*Cadence Virtuoso Schematic Design and Circuit Simulation Tutorial*" from the course web page:
http://www.brown.edu/Departments/Engineering/Courses/engn1600/assignments.html
The transistor sizing should be as follows:

	W_p	L_p	W_n	L_n
Inverter	1 µm	160nm	500nm	160nm
NAND	1 µm	160nm	1 µm	160nm
NOR	2 µm	160nm	500nm	160nm

4) Using ADE L simulate your design (follow the tutorial for setting up the simulation parameters). Make sure you create a new testbench view for running the simulations:

   a) Using SPECTRE, simulate the three gates for all input combinations using a transient analysis.

   b) Use SPECTRE to plot the Voltage Transfer Characteristics (VTC) of the inverter (dc analysis). The SPECTRE tutorial talks about how to use DC analysis to perform a VTC simulation. What are the values for the switching threshold and the noise margins (NMH and NML) of the inverter?

   c) Use SPECTRE to plot the VTC of the NAND and NOR for three conditions: first input switching second input steady, second input switching and first input steady, both input switching at the same time. Which voltage should be applied at the input of the NAND and NOR gates in order to see a change in the output? Do the three voltage transfer characteristic curves look the same? Explain the results.

   d) Using SPECTRE simulate and plot the I_D vs. V_{ds} characteristics of an NMOS transistor sized as L=160nm and W=500nm. Vary V_{ds} from 0 to 1.1V with a step of 0.01V and V_{gs} from 0 to 1.1V with a step of 0.2V

5) Repeat part 5d, but this time change the size of the NMOS to L=50nm and W=150nm. Again vary V_{gs} from 0 to 1.1V in steps of 0.2V. Plot the I_D vs. V_{ds} characteristics of the resized NMOS. Do you see any difference in the I_D- V_{ds} plot between the new NMOS and the one in problem 5d)? Why?

6) Update the schematics for your three gates using the following size table:

	W_p	L_p	W_n	L_n
Inverter	300 nm	50nm	150 nm	50nm
NAND	300 nm	50nm	300 nm	50nm
NOR	600 nm	50nm	150 nm	50nm

7) Using the layout tool and the tutorial, draw a layout for an inverter with the new size for PMOS and NMOS. Make sure that your design is as compact as possible while following the design rules specified in the tutorial. You should have already completed this for the tutorial. Run DRC and LVS to check your design.

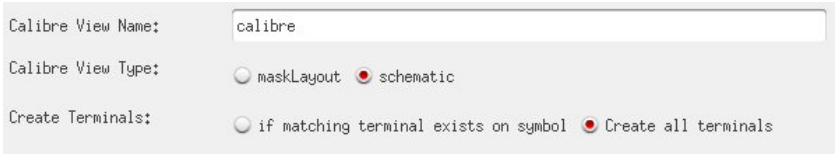
8) From the Layout view, generate a new netlist with parasitic extraction using Calibre PEX. This will create an additional view (calibre) in your Library. Use this view to simulate your gate again. Follow the tutorial for using Calibre PEX at the end of the assignment. Do you notice any difference with the previous set of simulations?

9) Repeat part 7) and 8), but this time for a 2-input NAND layout and a 2-input NOR layout using the sizes given from part 6). You do not need to simulate all possible input/output transitions, but you should activate all 4 input combinations.

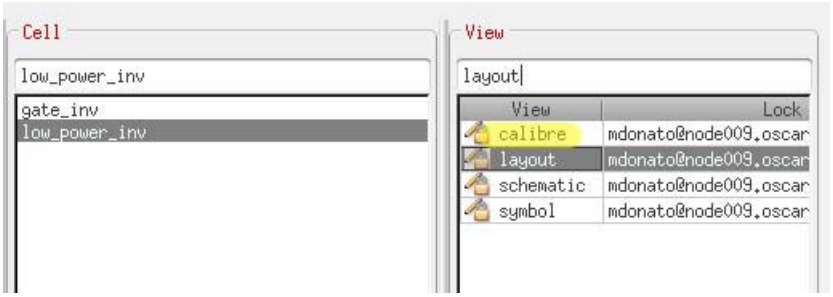
Use Calibre PEX for parasitic extraction and post-layout simulation

The schematic view is an ideal representation that does not take into account how the circuit layout is made. Using Calibre PEX, the tool for parasitic extraction, we can create a more accurate version of our circuit that includes parasitic capacitances due to the actual physical dimensions of the transistors and the routing between them.

From the layout view, select **Calibre > Setup > Calibre View …** and select the following settings:



Confirm the setup by pressing OK. Now, go to **Calibre > Run PEX** and when the Calibre Interactive Window pops up, press **Run PEX**. When the tool has finished extracting the parasitic components, the Calibre View Setup window will pop up again. Verify that the Output Library and the Schematic Library correspond to your own library (they should be the same) and press OK. This should generate an additional View of type *calibre* in your Cell.



In order to simulate the new circuit with parasitics, go to ADE L (the setup for the simulation is the same as before in part 4). Before running the simulation, go to **Setup > Environment** and add calibre before the word schematic in Switch View List as shown in the figure below. Confirm and run the simulation.