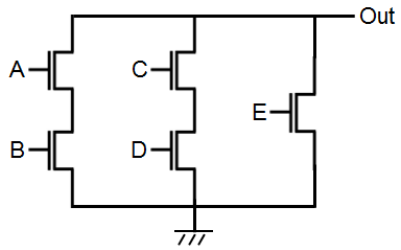
Homework Assignment 1
(Due Feb. 4th at the beginning of the class)

(1) [Static CMOS Circuit, **5 points**] Draw a transistor-level schematic for the following function. Use six NMOS and six PMOS transistors.

$$\boldsymbol{F = \overline{A \cdot B + A \cdot C + B \cdot D}}$$

(2) **[**Static CMOS Circuit, **5 points**] Draw a transistor-level schematic for the above function. Use the smallest number of transistors.

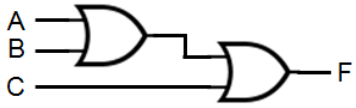
(3) **[**Static CMOS Circuit, **5 points**] An NMOS logic array for a function is shown below. Construct the PMOS circuit to complete the function.



(4) **[**Static CMOS Circuit, **5 points**] Draw a transistor-level schematic for the following function.

$$\boldsymbol{F = A}$$

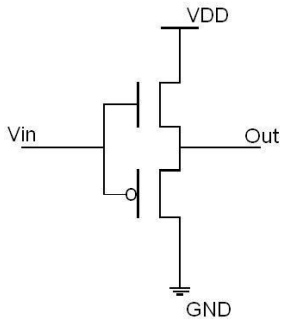
(5) **[**Logic Conversion, **5 points**] The two-input AND and OR gates use six transistors, but the two-input NAND and NOR gates use four transistors, so NAND and NOR gates are preferred to AND and OR gates in the design of CMOS circuits. Convert the following logic ($F = A + B + C$) into a new logic using only INV, two-input NAND, and two-input NOR gates. Use the smallest number of transistors.



(6) **[**Static CMOS Circuit, **5 points**] Draw a transistor-level schematic for a two-input MUX ($F = \bar{S} \cdot A + S \cdot B$).

	S	
	0	1
F	A	B

(7) [CMOS gates, **5 points**] What is the intended function of the circuit shown in the figure below? What is the output swing?



(8) [HSPICE simulation, **10 points**]

1. Create a work directory.

   > mkdir ee434_hw01

   > cd ee434_hw01

2. Download the following file into your directory.

   > wget http://www.eecs.wsu.edu/~ee434/Homework/hw01.tar.gz

3. Unzip it.

   > tar xvfz hw01.tar.gz

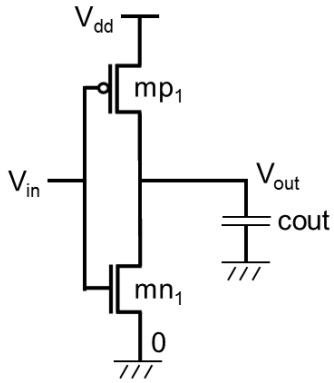
4. There are four files.

   - 45nm_PTM_HP_v2.1.pm: 45nm transistor models for SPICE simulation.
   - clr: run this to clear the files HSpice generates. (> ./clr)
   - inv.sp: HSpice netlist for an inverter.
   - synopsys.sh: a shell script to run HSpice.

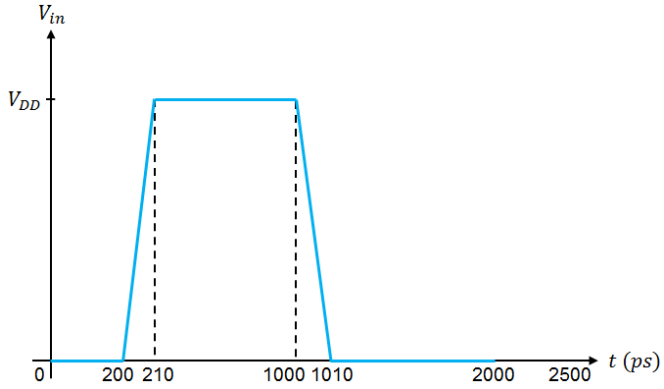
5. Source the .sh file to enable running HSpice. (You have to run this whenever you connect to an EECS server).

> source synopsys.sh

6. Open inv.sp and read the comments. You will be able to figure out the file format. The following visualizes the netlist.



7. The following visualizes the input signal (V_{DD}=1).



8. Run HSpice to simulate the given netlist.

> hspice inv.sp

9. You will see the following message.

```
>info:          ***** hspice job concluded
 lic: Release hspice token(s)
real 0.36
user 0.06
sys 0.02
```

10. Scroll up and see the outputs. If you cannot see the whole output messages, you can re-direct the output to a file and open it.

> hspice inv.sp > output.txt

> vi output.txt

11. The leftmost column is the simulation time, the middle one is V_{in}, and the rightmost one is V_{out}.
12. See the two "measure" statements in inv.sp. The first and the second statements measure the 80%-20% fall time and 20%-80% rise time, respectively.
13. See the output and find "tfall" and "trise". "tfall" is the fall time and "trise" is the rise time.

```
  2.4990e-09       0.000e+00        1.000e-00
  2.5000e-09       0.000e+00        1.000e-00
y
   tfall=  1.4672E-10  targ=  3.9465E-10   trig=  2.4793E-10
   trise=  1.4401E-10  targ=  1.1866E-09   trig=  1.0426E-09

          ***** job concluded
1****** HSPICE -- E-2010.12 32-BIT (Nov 20 2010) linux ******
```

14. In the above figure, the fall time is 146.7ps and the rise time is 144.0ps. They are balanced pretty well. See the width of the NMOS and PMOS transistors.
15. Change Wn (the width of the NMOS transistor) from 45nm to 90nm. Change Wp (the width of the PMOS transistor) from 70nm to 140nm. (so the inverter is 2x larger than the minimum-size inverter.) Re-run HSpice with these new widths and obtain rise and fall times.
16. [**Submit**] The rise time and the fall time.
17. Set Wn to 180nm and Wp to 280nm (INV_X4). Run HSpice and obtain new rise and fall times.
18. [**Submit**] The rise time and the fall time.
19. Set Wn to 90nm and Wp to 140nm. Set Cout to 20fF. Run HSpice and obtain new rise and fall times.
20. [**Submit**] The rise time and the fall time.
21. Wn=90nm. Wp=140nm. Cout=20fF. Change the rising and the falling transition times of the input signal from 10ps to 50ps, run HSpice, and obtain new rise and fall times.
22. [**Submit**] The rise time and the fall time.