Lab 3 Digital Circuit Simulations ECE334S

Objective:

To design/simulate some CMOS logic gates, using SUE/HSPICE.

Preparation:

For the following questions, assume $V_{tn} = 0.43V, V_{tp} = -0.616V$, this was obtained from the $0.25\mu m$ CMOS process model file. Also assume that $\mu_n C_{ox} = 88\mu A/V^2$, $\mu_p C_{ox} = 30.67\mu A/V^2$, which was obtained by simulating a diode connected transistor and using the drain current to extract the value of the transistor transconductance parameters. For the rise and fall time calculations, you can approximate the transistors by their equivalent resistance.

P1) For the unit-sized inverter shown in figure 1, find analytically the approximate rise and fall times (10% to 90%) assuming the inverter derives a 0.3pF load capacitor.

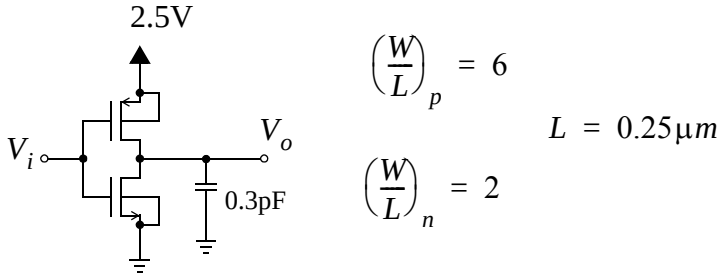

Figure 1: CMOS Inverter

P2) Draw the schematic of the CMOS logic gate that implements the function $Y = \overline{A + BC}$.

(a) Identify the input conditions for the worst case rise/fall times and the best case rise/ fall times.

(b) Size your transistors, such that for the worst case the gate achieves the same rise and fall times of the unit-sized inverter of P1 when your gate is driving a load capacitance of 1.5pF.

(c) Using the equivalent resistance to model the transistors and neglecting the diffusion capacitances, estimate the worst case rise and fall time of your gate and compare them to that of the unit-sized inverter in P1.

(d) Using the equivalent resistance to model the transistors and neglecting the diffusion capacitances, estimate the best case rise and fall time of your gate.

P3) Using a truth-table, briefly describe the operation of this flip-flop.

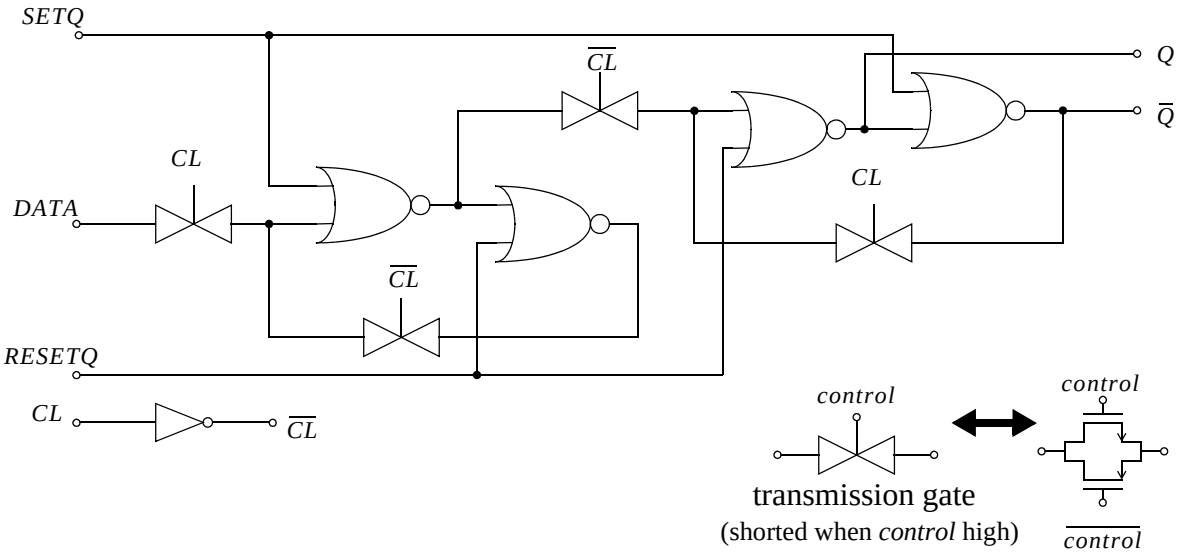

Figure 2: Logic diagram of "D-type" flip-flop[Ref. section 7.3.4 of textbook]

Lab Work

L1) Simulate the inverter of figure 1 using a 0.3pF load capacitance. Find the rise and fall times, and compare your simulation results with the analytic results.

L2) Simulate the CMOS gate you designed in P2 assuming it drives a load capacitance of 1.5pF. Find the worst/best case rise and fall times, and compare your simulation results with the analytic results. How does your gate's worst and best case rise/fall times compare to that of the inverter of part L1?

L3) Simulate the D-type flip-flop in figure 2 and show that it operates as expected. Use minimum length for all the transistors, and assume $W_n = 3\mu m$ and $W_p = 9\mu m$ for all the switches and the clock inverter. But size the NOR gates to achieve a worst case rise/fall times equal to that of the clock inverter.

You will need to increase the transient simulation time. To do that, create the netlist using SUE, then open it using any text editor, for example "nedit", and set the simulation time in the .tran command to **250nsec**. Use the command line to run hspice, by typing "hspice filename.sp" or "/local/bin/hspice filename.sp". When the

simulation ends, view the simulation results in NST, by loading the file “filename.tr0”. Divide your plot into 4 panels, and plot the following:

panel 1: CL, $\overline{CL}$
panel 2: SETQ, RESETQ
panel 3: DATA
panel 4: Q, $\overline{Q}$.

Hint: To demonstrate your flip-flop operation, it is recommended to use the following settings for your input signal sources:
CL: pulse(t_d=0ns, t_r=0.5ns, t_f=0.5ns, pulse-width=10ns, pertiod=20ns)
SETQ: pulse(t_d=55ns, t_r=0.5ns, t_f=0.5ns, pulse-width=40ns, pertiod=1000ns)
RESETQ: pulse(t_d=175ns, t_r=0.5ns, t_f=0.5ns, pulse-width=40ns, pertiod=1000ns)
DATA: pulse(t_d=5ns, t_r=0.5ns, t_f=0.5ns, pulse-width=20ns, pertiod=40ns).

Your input signals should look similar to figure 3.

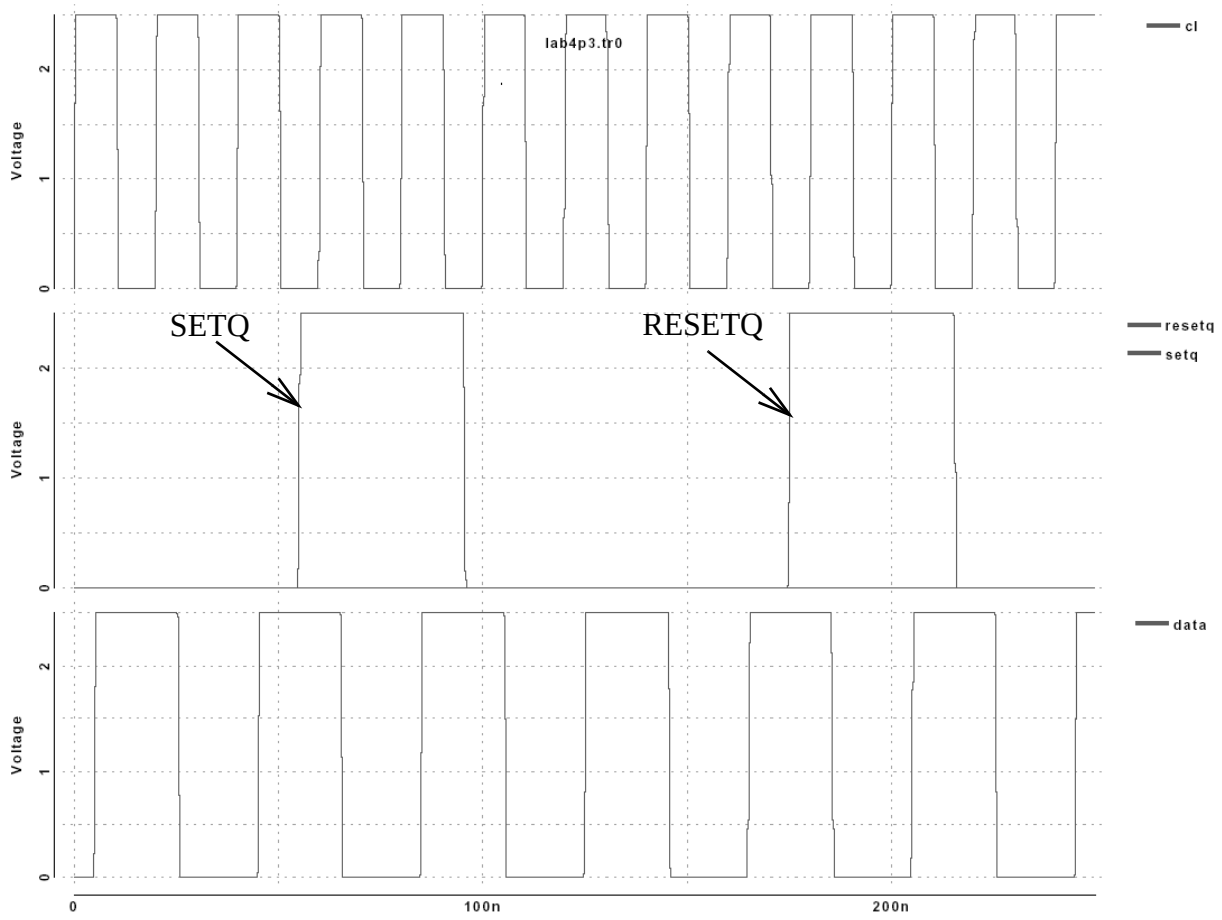

Figure 3: Input signals applied to verify the operation of the “D-type” flip-flop.

FAQ:

1. In part L1, the output waveform looks weird. It doesn't behave like an inverter in simulation.

Check the pulse width of your input source. Make sure enough time is given for the output rising or falling completely.

2. In part L3, which number should I change to 250nsec in the .tran command?

When you open the spice file, you should find the .tran command looks like,

.tran 10p 16n

The first parameter value (10 ps) specifies the step size, which instructs the simulator how often it should sample one simulation data, and the second parameter value (16 ns) specifies the end time of simulation. As a result, you should change the second parameter value to 250n.

3. What's the procedure to increase the transient simulation time in part L3?

The following procedure is suggested,

a) Type 'h' to generate the the SPICE file and simulate your design. It will open the NST viewer. Don't close it.

b) Open the generated SPICE file (e.g. lab3_L3.sp) with your favorite editor and change the simulation time to 250 nsec (refer to the previous question).

c) In the terminal (not the one you are running 'sue'), simulate your design by typing "hspice <filename>.sp" (e.g. hspice lab3_L3.sp). Wait until the simulation finishes. An output file named "<filename>.tr0" (e.g. lab3_L3.tr0) will be generated if simulation succeeds.

d) From the NST viewer window that is popped up in step (a), load the output file (e.g. lab3_L3.tr0) from FILE -> LOAD.