EECS 312: Digital Integrated Circuits
Lab Project 1 – Introduction to Schematic Capture and Analog Circuit Simulation

Teacher: Robert Dick
GSI: Shengshuo Lu

Assigned: 5 September 2013
Due: 17 September 2013

1 Introduction

In this laboratory assignment, you will specify a CMOS inverter driving a resistive–capacitive (RC) load that using Cadence schematic capture software. You will then conduct analog simulation of the inverter.

1.1 Goal for Friday

Make sure you can get started using the CAD tools during Friday's help session. However, if you get confused about the technical details in the assignment after that, it is fine to "call it a day" and wait until after Tuesday's lecture before finishing the assignment. We are using the Friday help session to make sure nobody will get stuck with software or account configuration glitches but the technical ideas in the assignment won't be covered until Tuesday. However, if you feel confident about finishing the assignment after Mr. Lu's talk, give it a try!
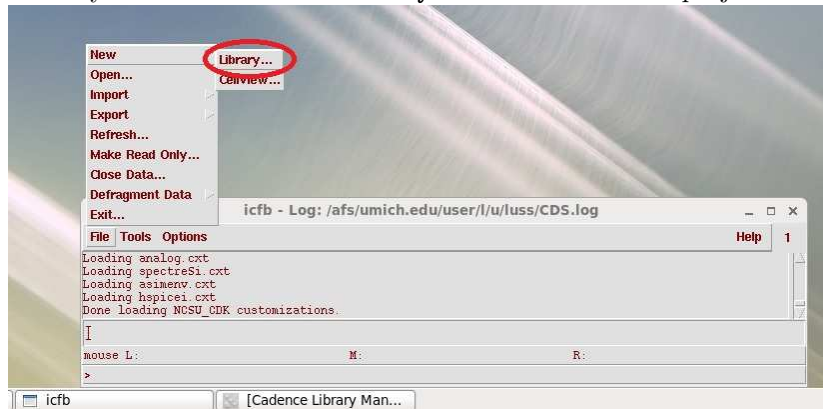
2 Tutorial

Before you design your own inverter, I will first walk you through the design of an NMOSFET with an RC load. You do not need to hand in anything from this section of the assignment. However, you will need to use these concepts when completing the assignment: the design of a (more complex) CMOS inverter.

1. Log into a Linux CAEN lab machine. If the machine is currently running Windows, restart it to boot into Linux. If you do not have a preferred window manager, use Gnome or KDE.

2. Start a shell, e.g., select "Applications → Accessories → Terminal".

3. Type "module load eecs312/f13". This sets all of the required environment variables.

4. Type "prep312"

   and set up the symbolic link, set up the kit directory, and move you into that kit directory (second line).
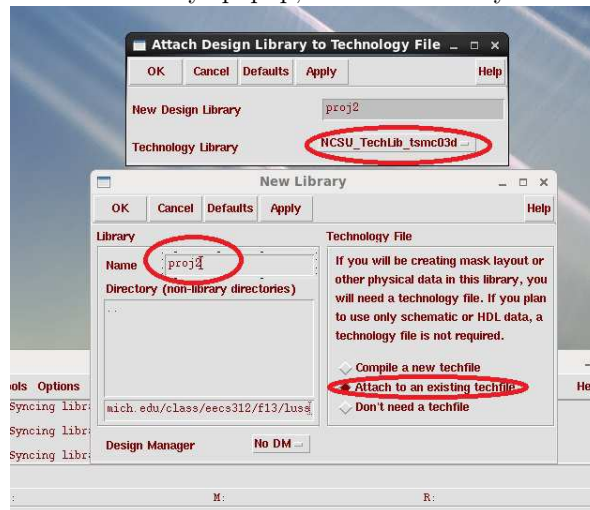
5. Start the Cadence software

   icfb

   If you need to execute shell commands while this is running, start another shell.
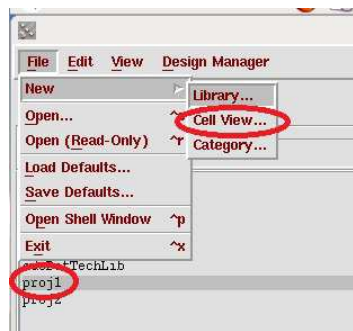
6. You will now create a new library and create a NMOS-based inverter within it. If the "Library Manager" did not open automatically, from the icfb software, start it "Tools → Library Manager".

7. Create a new library "File → New → Library". Give it the name "proj1". Click "OK".

8. In the "Technology File for New Library" popup, attach the library to an existing techfile, "NCSU_TechLib_tsmc03d".

9. From within the proj1 library window, create a new inverter. Make sure "proj1" is selected and then "File → New → Cell view". Name it "nmos-inverter". The Virtuoso schematic capture software will now start.

10. Enter the circuit shown in Figure 1. This is an NMOS-based inverter. If you want to understand the operation of the circuit, recall from class that an NMOSFET is on when its V_{gs} is higher than its threshold voltage.
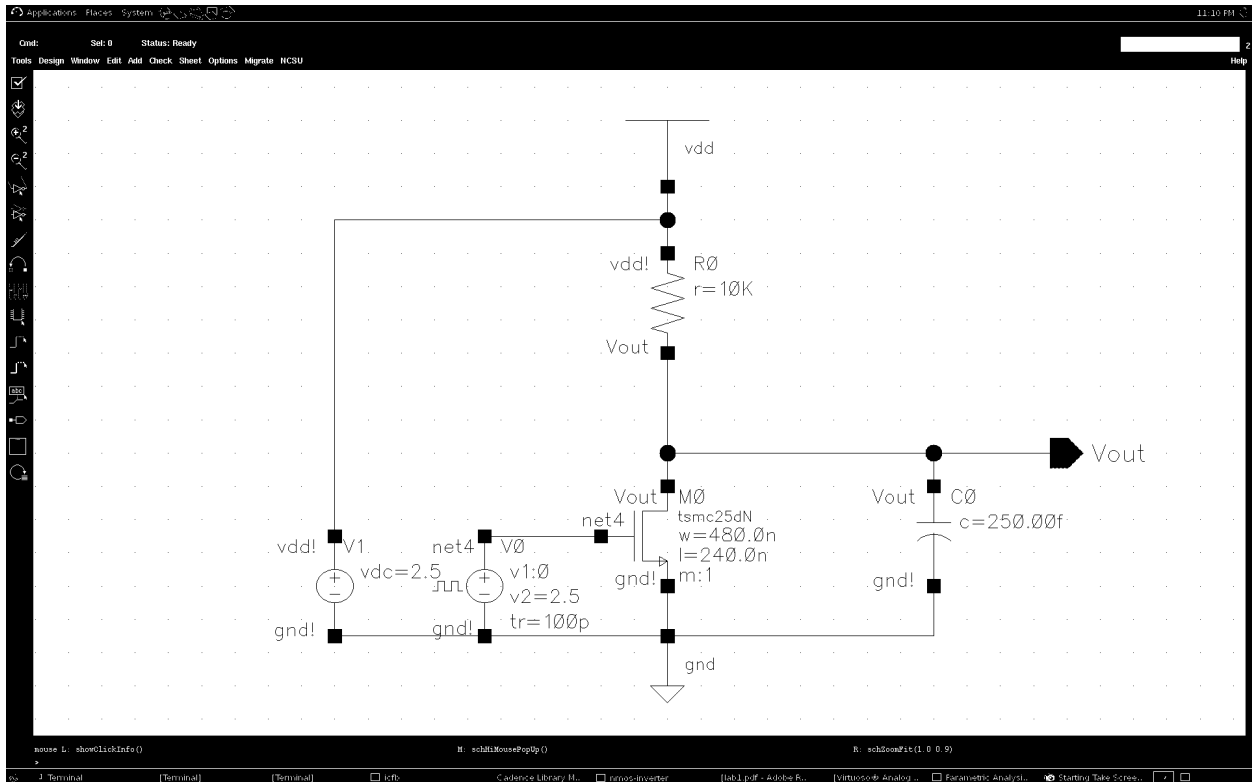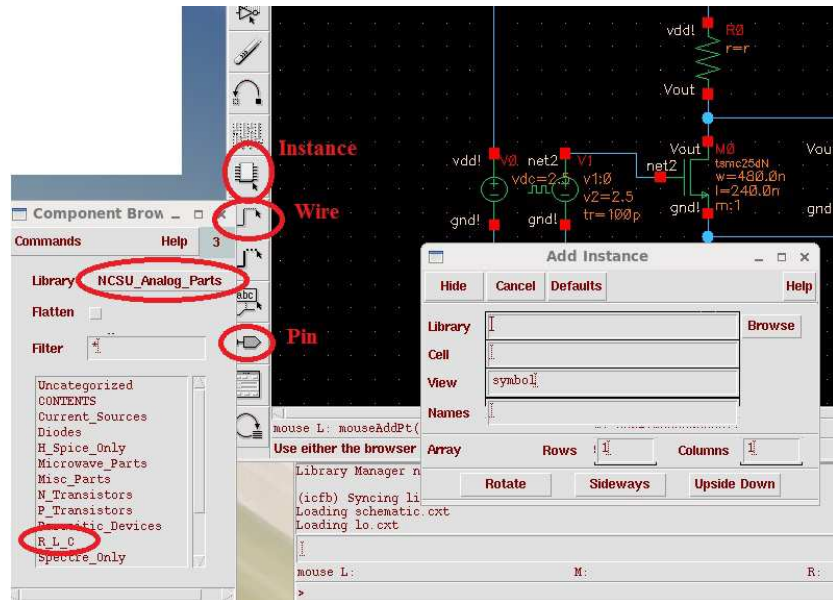
Figure 1: Schematic for NMOS with an RC load.

To do this, you will need to use the icons to the left of the window. Move the mouse pointer over them and read the tooltips.

(a) Place the resistor. Models for all the components needed are in the "NCSU_Analog_Parts" library. To instantiate the resistor, click the "Instance" icon on the left toolbar.

Table 1: Component Categories and Names

Part	Category	Component
Resistor	"R_L_C"	"res"
Capacitor	"R_L_C"	"cap"
NMOS Transistor	"N_Transistors"	"nmos"
Supply Voltage	"Supply_Nets"	"vdd"
Ground	"Supply_Nets"	"gnd"
DC Voltage Source	"Voltage_Sources"	"vdc"
Pulse Voltage Source	"Voltage_Sources"	"vpulse"

Table 2: Component Parameters

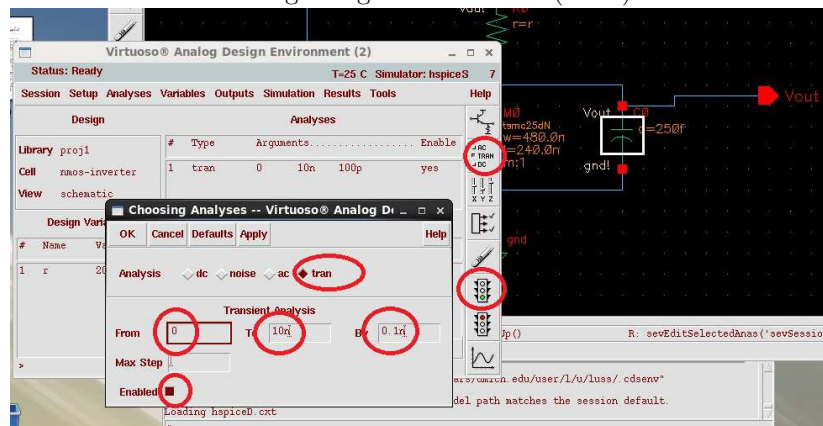| Part	Parameter (Units)	Value
Resistor	Resistance (Ohms)	"10K"
Capacitor	Capacitance (F)	"250f"
NMOS Transistor	Width (M)	"480n"
	Length (M)	"240n"
DC Voltage Source	DC Voltage (V)	"2.5"
Pulse Voltage Source	Voltage 1 (V)	"0"
	Voltage 2 (V)	"2.5"
	Delay (s)	"1n"
	Rise Time (s)	"100p"
	Fall Time (s)	"100p"
	Pulse Width (s)	"4n"
	Period (s)	"10n"

(b) Models for all the components needed are in the "NCSU_Analog_Parts" library. Select this library.

(c) Select the "R_L_C" category.

(d) Select the "res" component from the list and place it on the schematic. Press "ESC" to close the "Add Instance" window. Leave the default parameter values. These will be changed later.

(e) Add the six other components in a similar fashion. Table 1 lists the categories and components names.

(f) Add the output pin. Click on the "Pin" icon on the left toolbar. Set the pin name to "Vout", the direction to "Output", and place the pin on the schematic.

(g) Connect all the components as shown in Figure 1 using a wire, the "Wire (narrow)" icon on the left toolbar. Click the left mouse button (LMB) to start the wire, and double-click the LMB to complete it. While drawing a wire, a vertex or turn can be specified by single-clicking the LMB.

(h) Set the correct parameter values for each of the components. The properties of each component can be edited by selecting the component (single-click with the LMB) and pressing the "q" key. Table 2 lists the parameters that need to be changed. Mr. Lu and I will explain these parameters to you in the help session, and in Tuesday's lecture.

(i) Your schematic figure should now match the one show in Figure 1. If it does not, make the appropriate corrections.

(j) The final step is to save the design. Click the "Check and Save" button on the left toolbar. If successful, you will see a "Schematic check completed with no errors" message in the *icfb-Log* window. If there are errors or warnings, try to correct the underlying problem(s).

Note: It is advisable to "Check and Save" frequently, especially for larger circuits.

11. For simulations, you will always need to perform the following tasks:

   - Define and set up the analysis (either DC, AC, or Transient)
   - Define currents and/or voltages to be saved
   - Define any measurements, such as power consumption or delay

   In this lab, we will be conducting a transient (time-domain) analysis of the circuit, and observing its behavior as it responds to a varying input signal. In SPICE, there are three main simulation modes: DC, small signal AC, and Transient. In this class, we will be using mostly DC and Transient analyses. DC analysis determines what the circuit would eventually do if a particular set of inputs were applied and one waited a very long time (i.e., allow time to proceed toward infinity). It calculates the *steady-state* response of the circuit to the given inputs. Transient analysis determines the time-dependent response of a circuit to varying input signals.

12. Setup the Transient analysis. On the Virtuoso Schematic Editor, click on "Tools → Analog Environment". This will launch the Virtuoso Analog Design Environment (ADE) window.

13. On the righthand toolbar, click on the "Choose Analysis" icon.

14. In the window that appears, make sure that only the "tran" button is checked. Click "Choose analysis → Options". Enter "From 0n to 10n By 0.1n". Ignore the "Max Step" box. Click "OK".

15. In the ADE window, click on the "Run Simulation" icon on the righthand toolbar. Wait for the simulation to end. The *icfb-Log* window will display status messages regarding the simulation. If the simulation is unsuccessful, verify that the parameters for the "vpulse" component were set correctly.

16. View the results in the waveform viewer. In the ADE window, click on "Results → Direct Plot → Transient Signal". Go back to the schematic and single-click on the wire connected to the "Vout" pin and single-click on the wire connected to the gate of the "nmos" transistor. Press "ESC". A window with waveforms will popup.

17. The two waveforms will be overlapping. To see them individually, click on "Axis → Strips". You should see a waveform similar to Figure 2.

18. Perform a sweep analysis to show the role of the resistance in the circuit. A sweep analysis will run a simulation for each value of an instance that is being swept. The sweep is created by specifying a start and end value for a given parameter, along with a number of points or a value by which to increment. If possible, try to limit the number of points so that the simulation results do not consume a large amount of disk space. Please also keep in mind that this type of analysis should not be used to make a
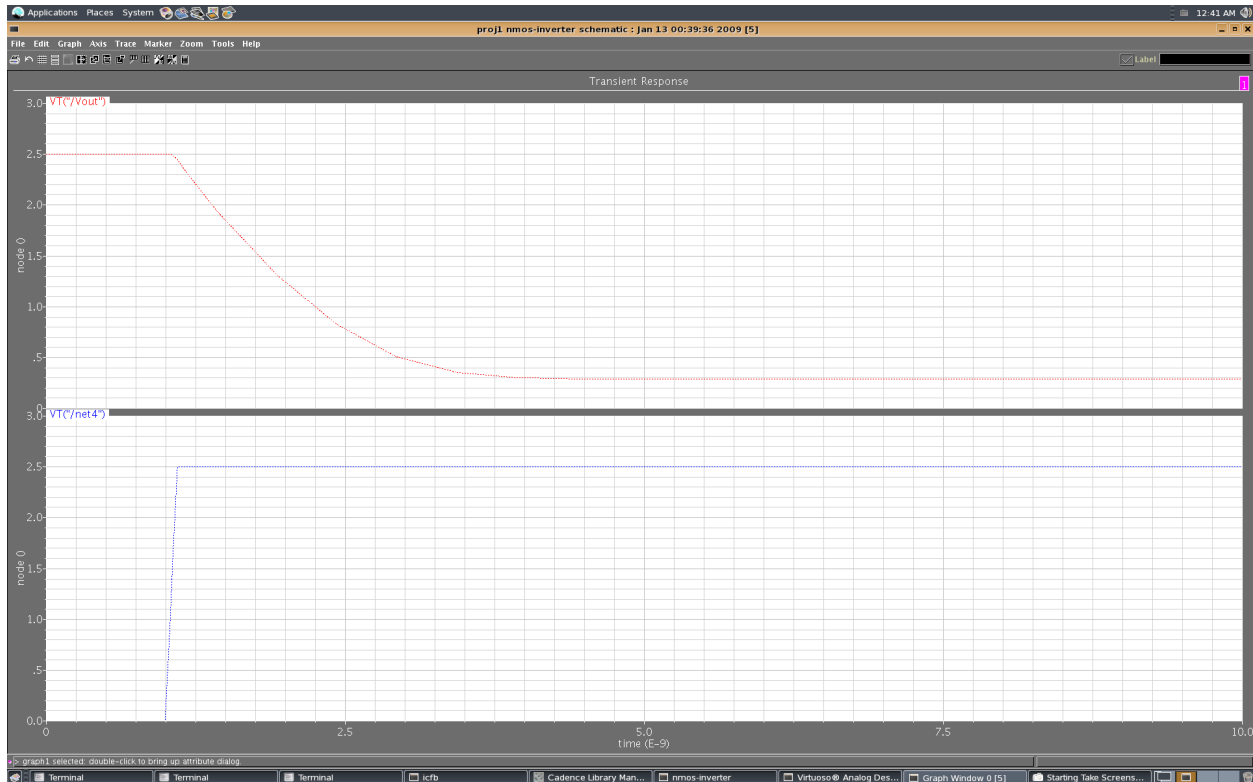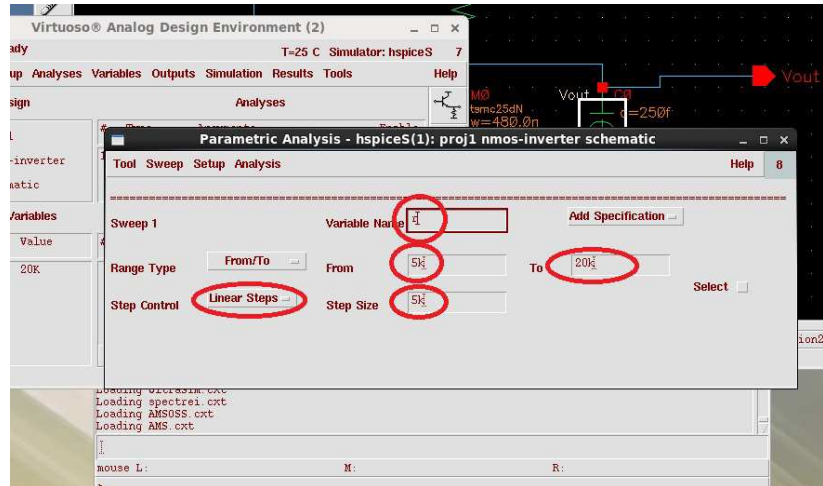
Figure 2: Input–output waveforms of the resistive load NMOS inverter.

voltage-transfer characteristic plot (VTC) or an I–V curve. For those, you will be using a DC analysis, which will solve for a steady-state operating point in your circuit.

You will be sweeping the value of the resistor to find out how the circuit behaves as the resistance changes.

19. Exit the waveform viewer and return to the schematic. Do not exit the ADE window. If you accidently you do exit it, you will need to repeat steps 12–14.

20. Change the resistance parameter for the resistor from "`10K`" to a variable, e.g. "`r`".

21. "`Check and Save`" the design. In the ADE window, click on "`Variables → Copy from Cellview`"

22. Click on "`Tools → Parametric Analysis`". Enter the "Variable Name" as "`r`" and specify the "Range Type" from "`5K`" to "`20K`". Change the "Step Control" from "`Auto`" to "`Linear Steps`" and set the step size to "`5K`".

6

23. In the *Parametric Analysis* window, click "`Analysis → Start`" to begin the parametric analysis.

24. Plot the results using instructions 16–17. There will be four different pairs of waveforms, one for each value of "`r`".

25. Separate windows or subplots can be created by selecting "`Trace → New Graph → (Move New SubWindow|Move New Window)`". The separate plot lines can be moved to the new windows simply by left clicking and dragging the plot line to the new window.

26. Relabel the title of each window to "Transient Analysis with Resistor Value of r", where r is the value of that specific sweep iteration. The axis and title can be changed by double clicking on the corresponding label. A new windows will appear. The labels can be changed by editing the "label" or "string field" for the axis and title respectively.

27. Markers can be added by clicking on "`Marker → Place → Vert Marker`" and the clicking on one of the waveforms. There are also "Cursors" available in the "`Trace`" menu. Play around with both of these and get comfortable using them. You'll be needing them extensively for future labs and your final project.

3 Your assignment

Design, enter, and simulate a CMOS inverter using a similar approach to that described in the tutorial. Use the same load and stimulus as described in the tutorial. Make the PMOSFET twice as wide as the NMOSFET. Please submit hardcopies of the deliverables *or* submit them via the website (Grades and report submission). You may use screenshots or print to files.

Deliverables:

1. The schematic for the inverter.

2. Time simulation timing diagram.

3. A brief write-up in the email indicating any significant design challenges or design tool software problems you overcame. This can be very short, or quite long, depending on the number of problems you encountered during the assignment.