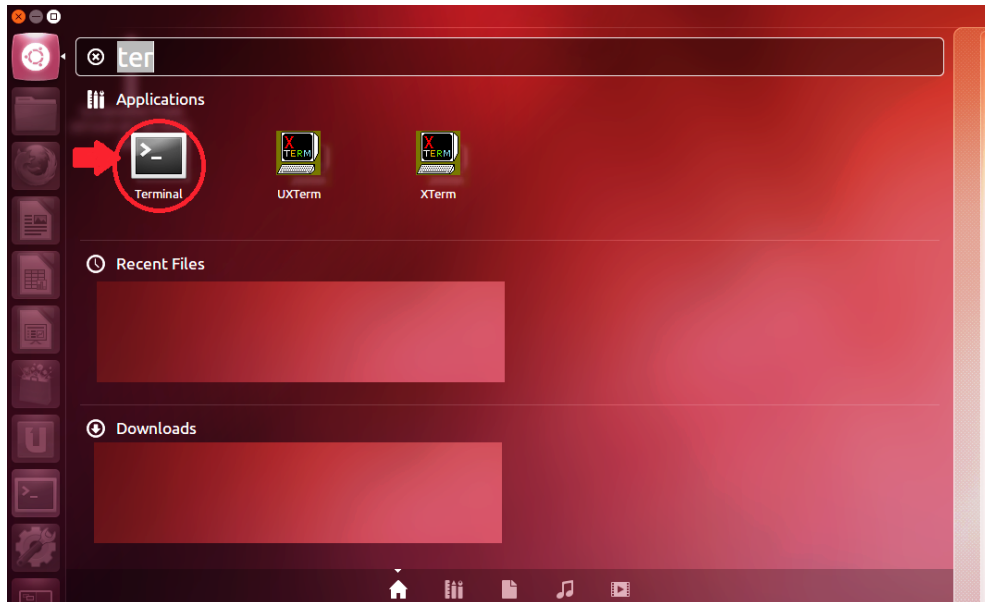


Cadence (version 6.1) Tutorial for Linux Environment

1. Setting up your Linux environment

1.1. Open a terminal



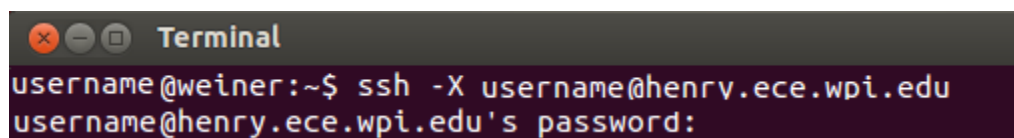
1.2. Log on to henry/db

Enter

```
ssh -X username@henry.ece.wpi.edu
```

or

```
ssh -X username@db.ece.wpi.edu
```



Then enter your 'ece password'.

1.3. Set up directories

To keep things manageable, you should do all your work in a separate directory. The first time you run cadence, create a separate directory cadence:

```
mkdir cadence61
```

You should always be in this directory when you start cadence. Go into the directory
`cd cadence61`

You need to copy some files to set up software environment variables and other simulation stuff.
Do this with

```
cp /opt/cadence/NCSU_CDK/newuser/*. * .  
cp /opt/cadence/NCSU_CDK/newuser/*.* .
```

(NOTE: there is a space after "cp" and after the final * before the period on each line)

Make another directory to store the lab schematics
`mkdir CD4007lib`

Now make yet another directory to store homework schematics
`mkdir 16umlib`

Be sure to remember these names, as you will need to use them when creating new libraries from inside 'virtuoso' in the next part of the lab.

You also need to make another directory to store your own spice models:

```
mkdir models  
cd models  
mkdir spectre
```

Spectre is Cadence's version of the SPICE circuit simulator.

2. Starting Virtuoso and Creating your libraries

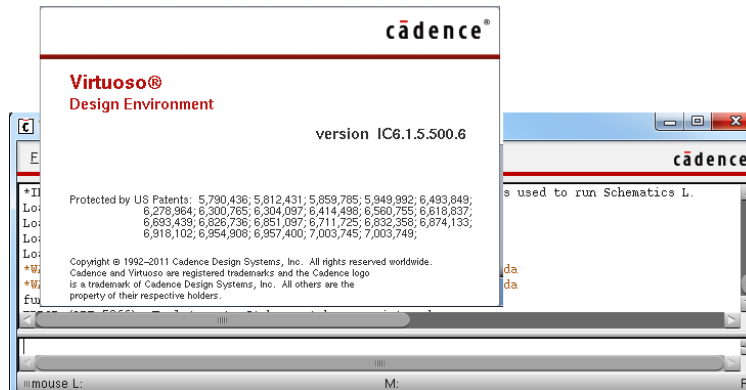
2.1. Start cadence

Be sure you're in the cadence61 directory before starting. You may need to

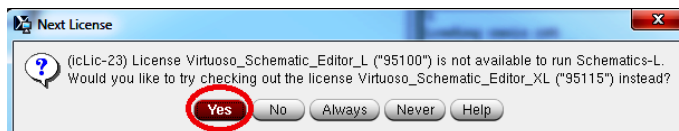
```
cd ~/cadence61
```

to get back to the cadence61 directory. Once you're sure you're in the cadence directory, to start the software, type:

```
virtuoso
```



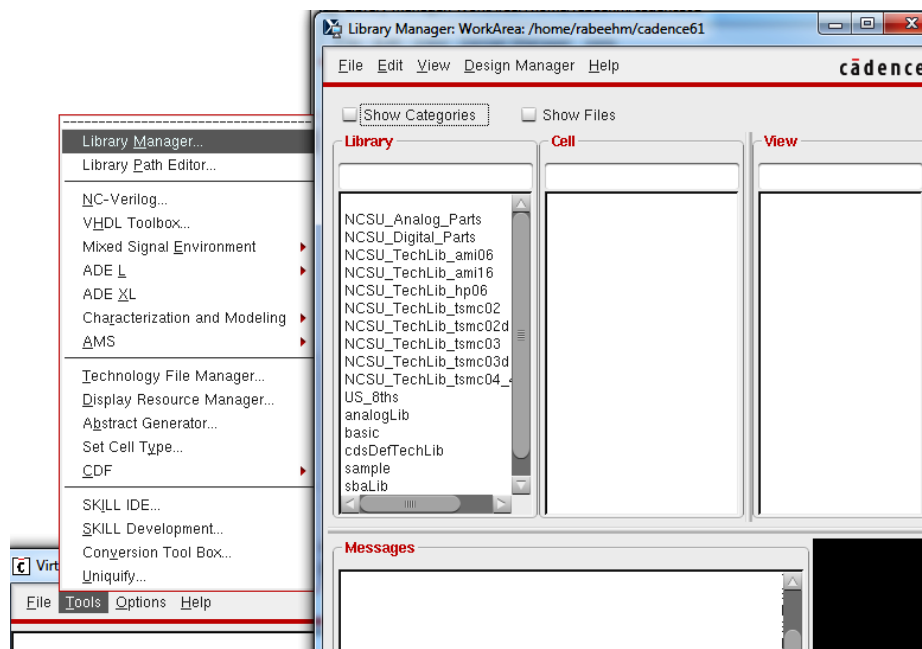
Some "What's New" and/or "help" windows may come up; click on whatever's necessary to make them go away. On "New License" menu, click on YES!



2.2. Create libraries for your design

Using the pull-down menu, select

Tools > Library Manager

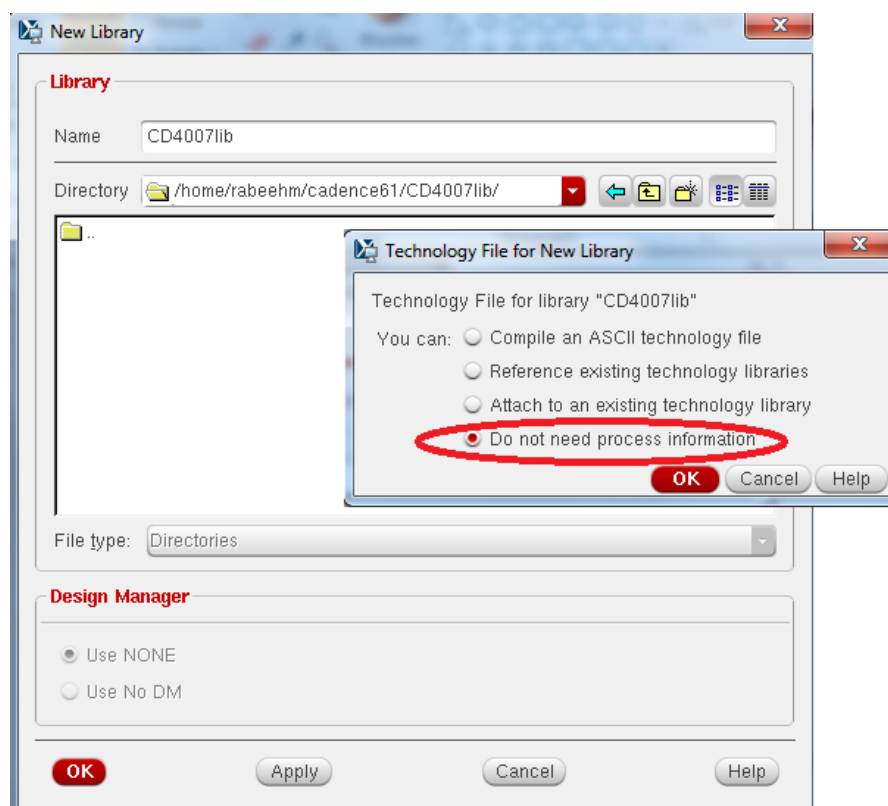


Using the pull-down menu in Library Manager, select

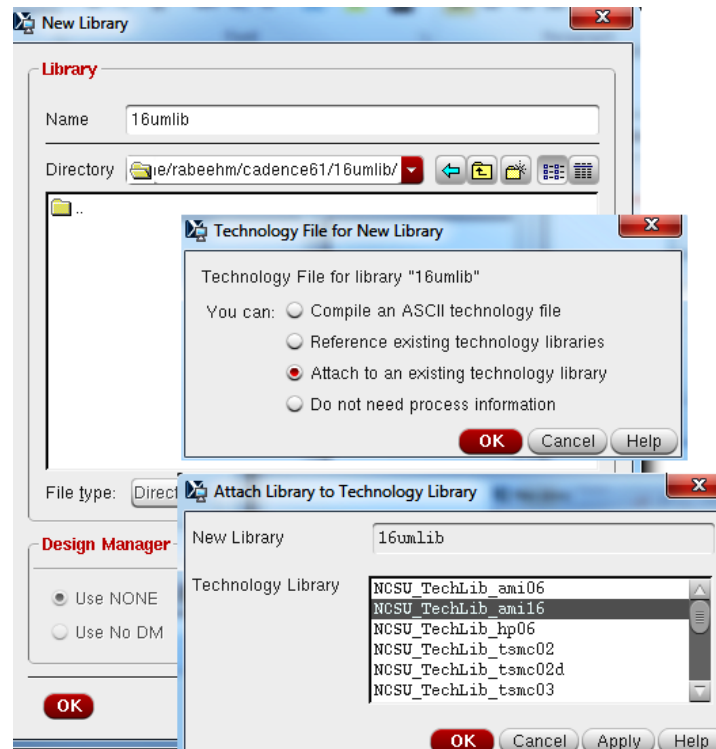
File > New>Library

A Create Library window will appear (see below). Choose a name for your library and type it in the box for Name: and set the directory path. In this example, the name is 'CD4007lib'. **The library name needs to be the same as the directory you created in part 1.3.**

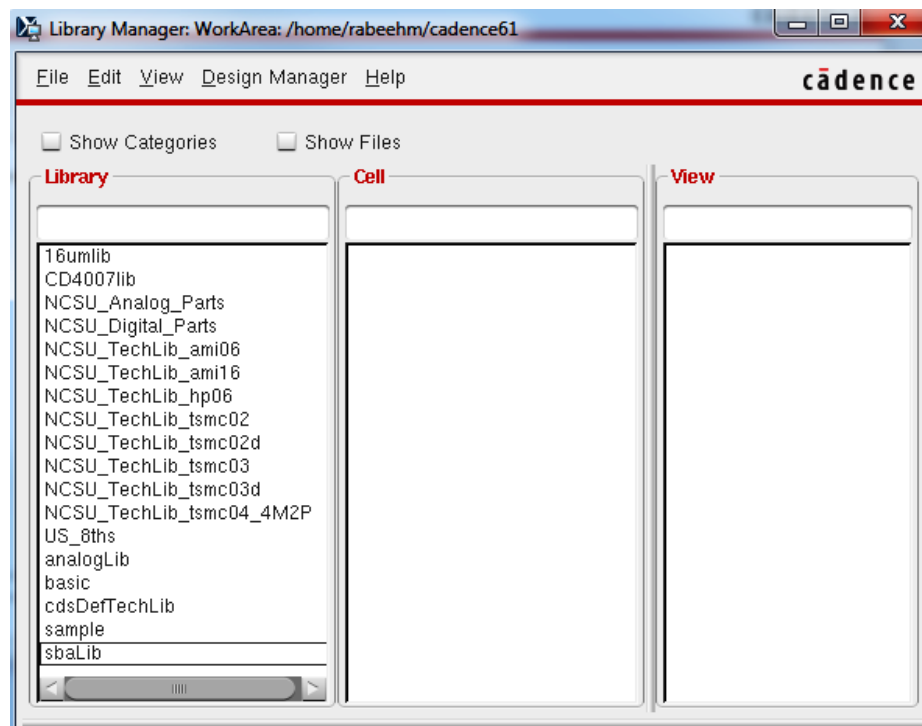
In the 'Technology file for new library' section of the Create Library window, click “Do not need process information”. No tech library is needed because the parameters of the CD4007 MOSFETs will be used for lab circuits.



Now create another library using the '16umlib' directory you created. This time in the 'Technology file for new library' section of the Create Library window, click 'Attach to existing technology library' and select 'NCSU_Techlib_ami16'. The technology file contains information specific to the process your IC would be fabricated in. Click OK.



After you click OK in the Create Library window, it should disappear. Now your library manager window should look like this:



And your library should be all ready for you to draw a schematic!

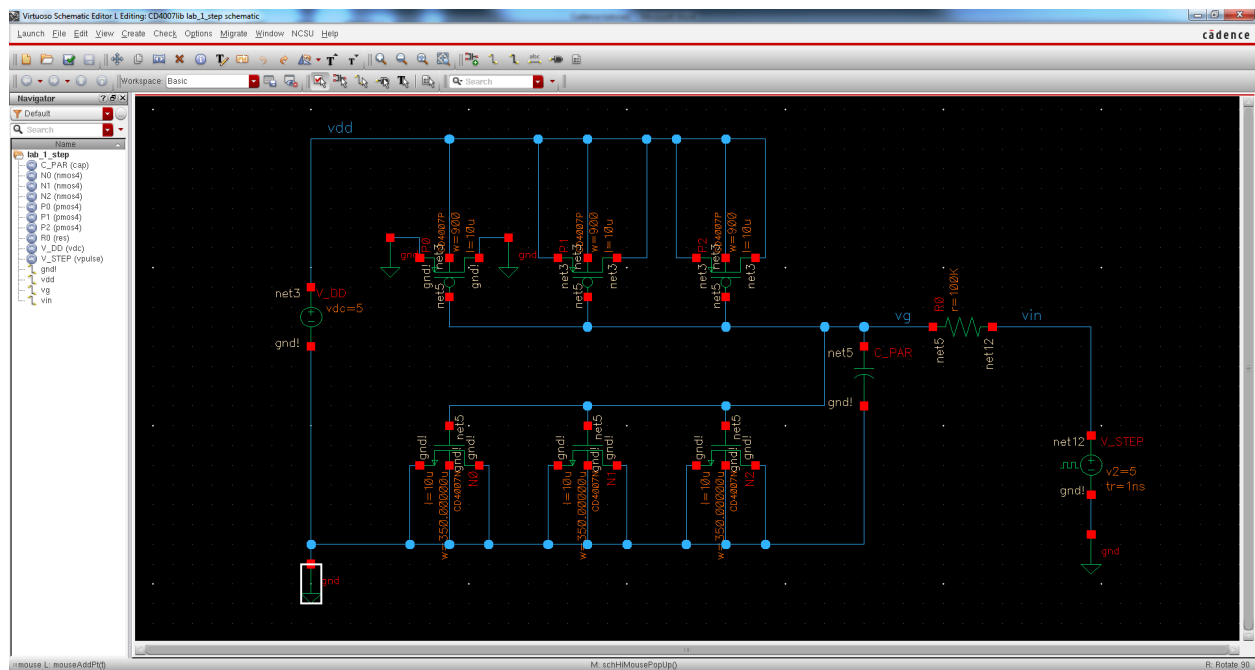
3. Schematic Entry, NMOS

Now you've set up your libraries and it's time to make a cell. Cells represent each of the pieces of the design you are working on. Each represents a circuit or sub-circuit, which can be shown in several different ways, such as schematic or symbol. Schematic and symbol are two types of views that can be created to represent the circuit.

A schematic view is an electrical representation of the circuit and might contain components (such as MOSFETs, resistors, capacitors, etc.) or symbols for other sub-circuits.

A symbol view is an abstraction that represents the circuit as a simpler symbol (such as the triangle for an op-amp) that can then be used in other schematics or circuit designs.

For our cell we will first create a schematic view that represents the MOSFET gate capacitance measurement circuit in Lab 1. Eventually the cell schematic will look like this:

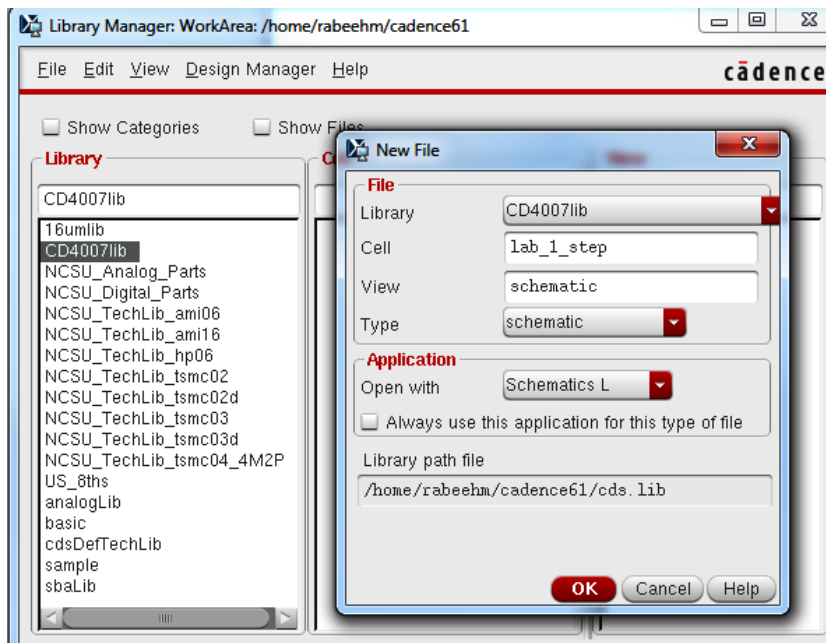


3.1. Creating a Cell

In the Library Manager window, click on the library name (CD4007lib) you just created. Using the pull-down menu, select

File > New > Cell View

This will allow you to define the cell name, the type of view, and the software tool to be associated with the cell view.



Click your cursor into the Cell Name box and type the name of the cell you want to create. Give it a descriptive name, maybe something like

lab_1_step

so you can remember what it does when you choose from your library of cells in the future.

For View Name, make sure to choose "schematic". (Usually we will start the IC design process with schematic representations of the circuit. This is followed by a symbol view for use in simulation. After some iteration with simulation, we should have a schematic of a working circuit. Then you can create a layout view of the cell and proceed with physical design.)

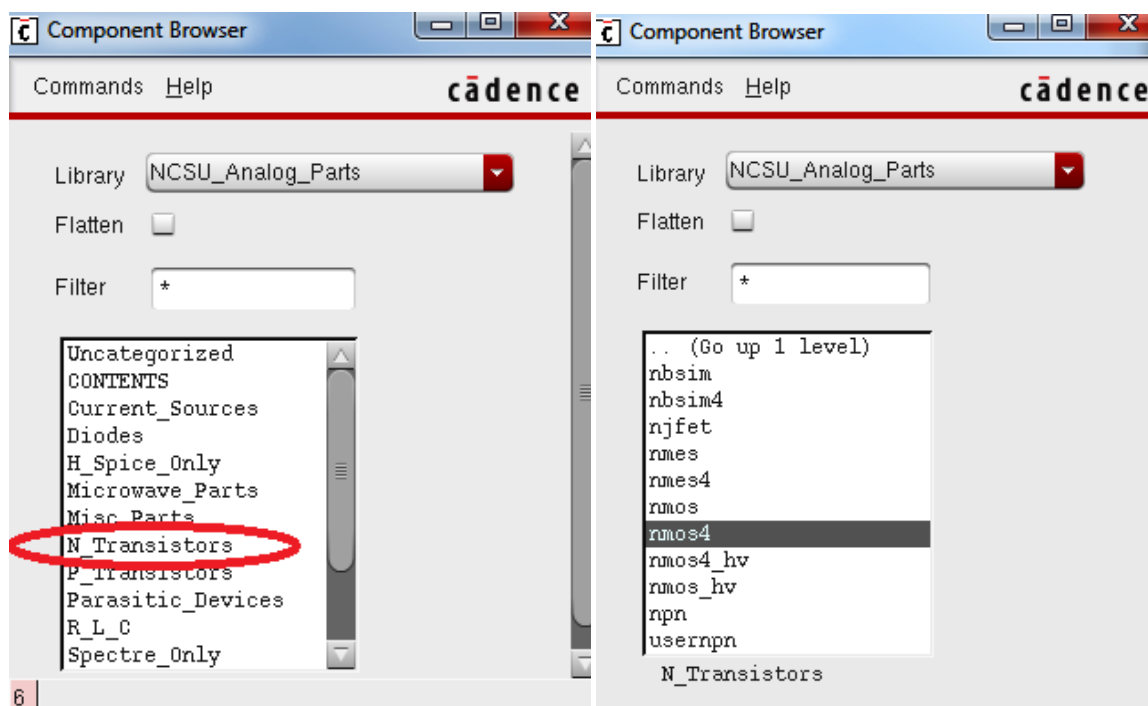
When you click on the OK box, a Virtuoso Schematic Editor window will open. Now it's time to draw the schematic.

3.2. Add MOSFET components

To add a component, you choose it from a "menu" of available components in the component browser. You can get to the component browser in one of these ways:

- From the pull down menu: select Create > Instance
- From the keyboard: hit "i"

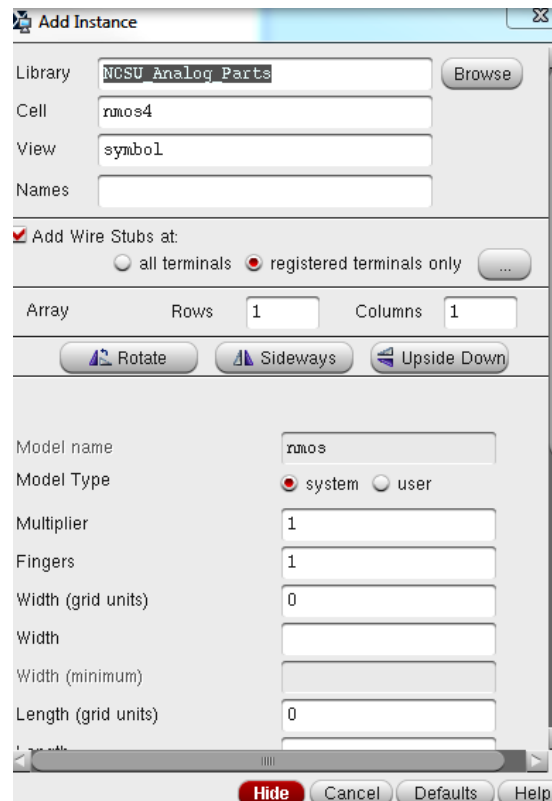
Doing any one of these will open a Component Browser Window, and an Add Instance window (see below). To choose from analog parts, go to the Component Browser Window and choose the **NCSU_Analog_Parts** library. Select the **N_Transistors** entry from the available contents and the component of **nmos4**.



When you move the cursor into the schematic window, you will see the outline of a MOSFET. You should also get an "Add Instance" box with (among other things) three buttons: "Rotate", "Sideways", and "Upside Down".

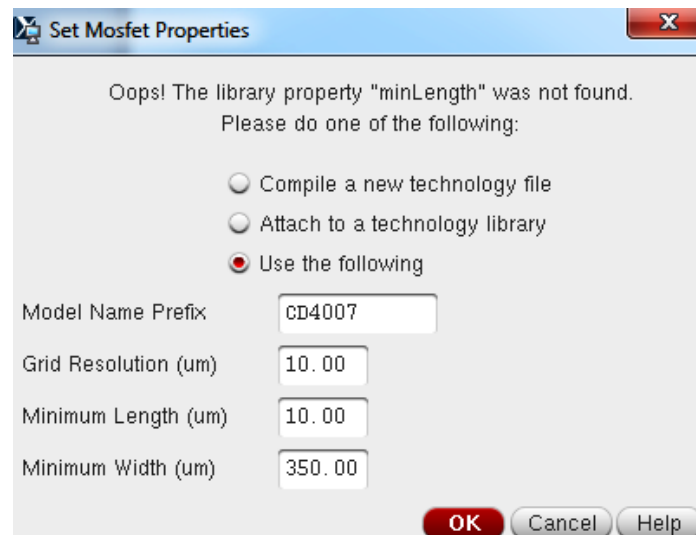
Clicking on these allows you to orient the MOSFET symbol before placing it on the schematic. Try clicking these to get the MOSFET oriented the way you want. When it's facing the right way, click the left mouse button to place the MOSFET. Moving the cursor will continue to show

the MOSFET outline; if you keep clicking you will place more MOSFETs. Just place one for now; hit the Esc key to get out of component placement mode.



3.3. Modifying NMOS part

When you first select the NMOS part, you will have to specify the properties of the transistor as shown below. Here we specify the dimensions of the CD4007 NMOS transistor.



Click OK when you're done. You should now be able to place the MOSFET in the schematic window.

Now that you've done something, save your design using

Design > Save

from the pulldown menu. It's a good idea to save your design often in case there's a software or network crash.

Now that you've correctly specified the NMOS device, make two more copies to represent the total of three NMOS devices in the CD4007/MC14007 MOSFET array.

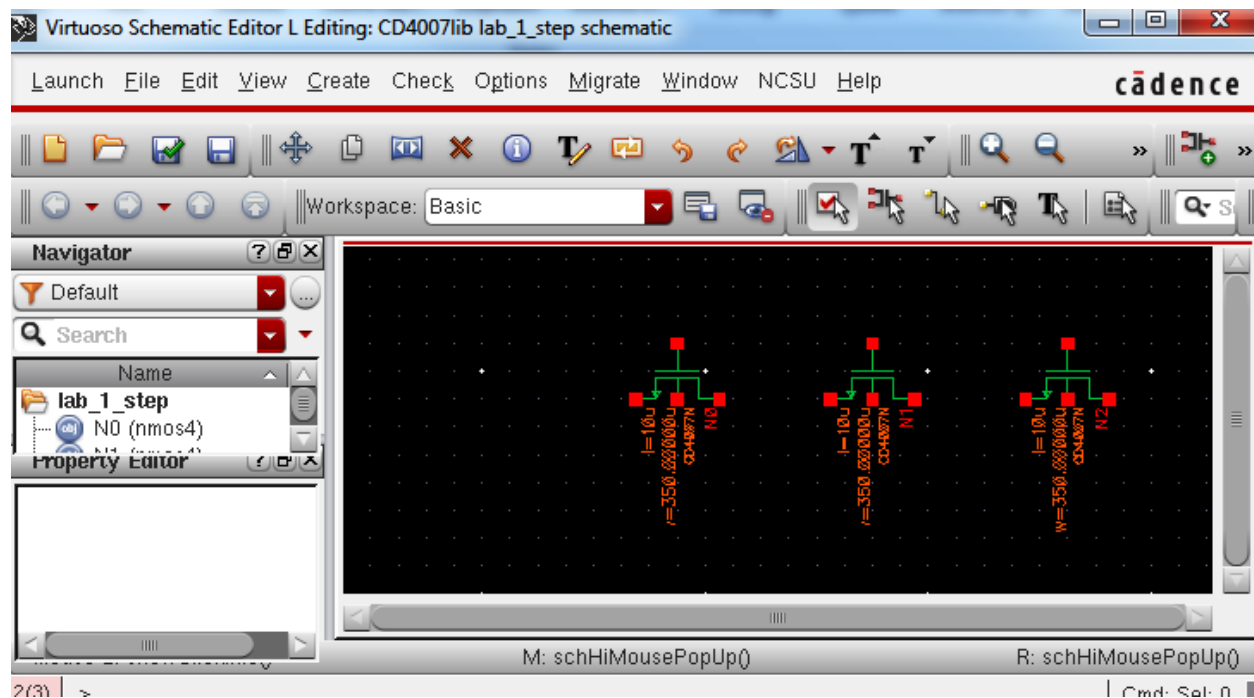
You can get to the copy function in one of three ways:

- From the pull down menu: select Edit > Copy
- From the keyboard: hit "c"
- From the buttons: the "Copy" button



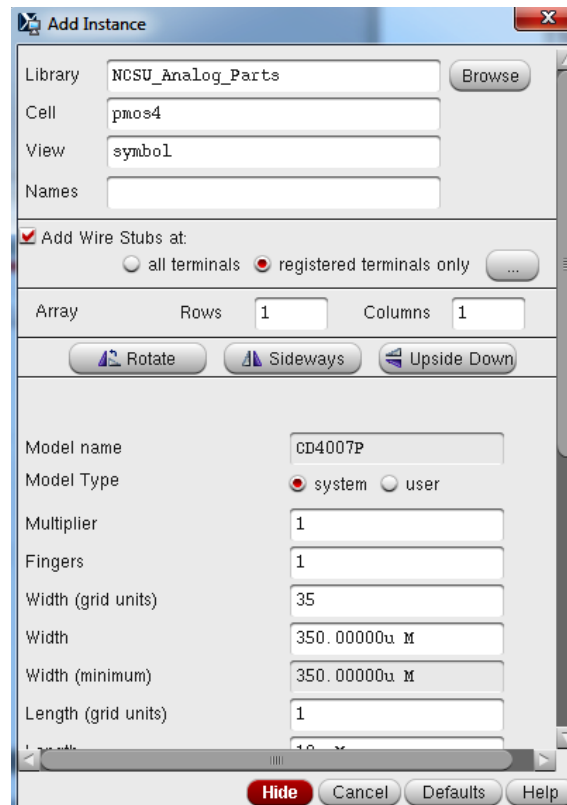
Once you've opened the Copy function, it takes two clicks to copy something. First click on the item you want to copy, and then click where you want the copy to be placed. Keep clicking to copy. When you're done, hit ESC so you don't keep copying things.

After making the copies, your schematic window should look something like this:



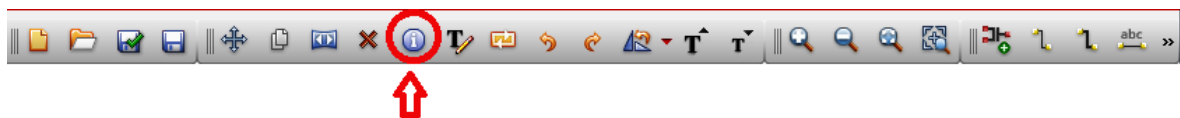
4. Schematic Entry, PMOS part

Now you need to add the PMOS transistors. From the **NCSU_Analog_Parts** library, choose **pmos4** from the **P_Transistors** menu. As you did with the NMOS transistor, orient and place the symbol in your schematic window.



The width of the PMOS transistor is different from the NMOS, so this value needs to be changed. Click on the component to select it. Once the component is selected, you can get to the component definition in one of three ways:

- From the pull down menu: select Edit > Properties > Object
- From the keyboard: hit "q"
- From the buttons: the "Property" button



You will be presented with the "Edit Object Properties" window (see below), offering a dizzying array of properties for the MOSFET.

Now change the following in the Edit Object Properties window

- Change the Width to 900.00u
- Change the Length to 10u.

Edit Object Properties

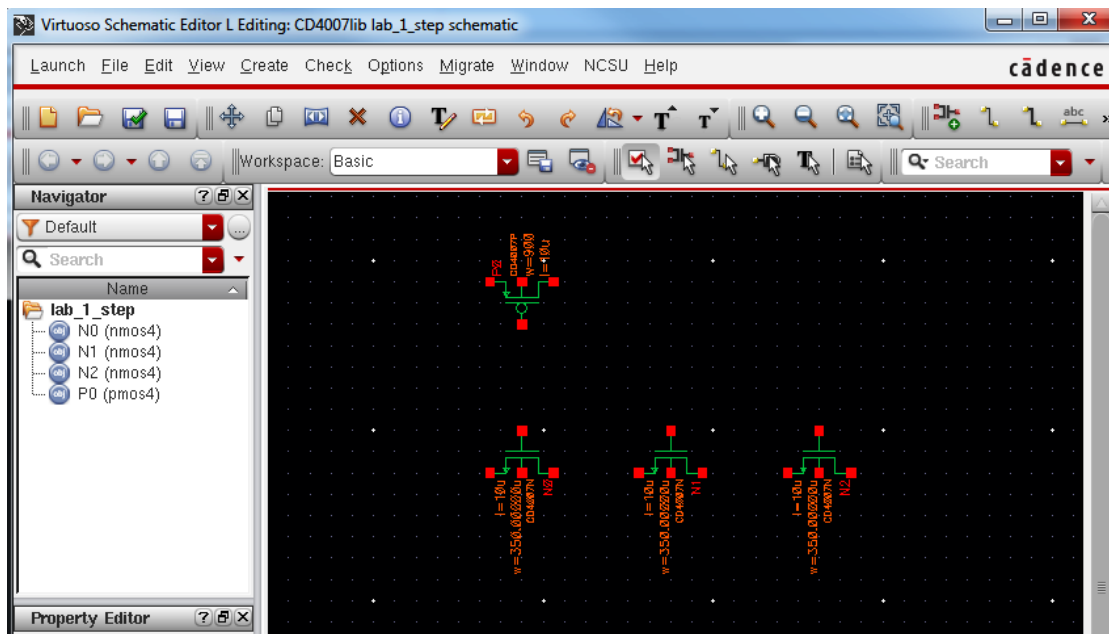
Apply To:

Show: ☐ system ☒ user ☒ CDF

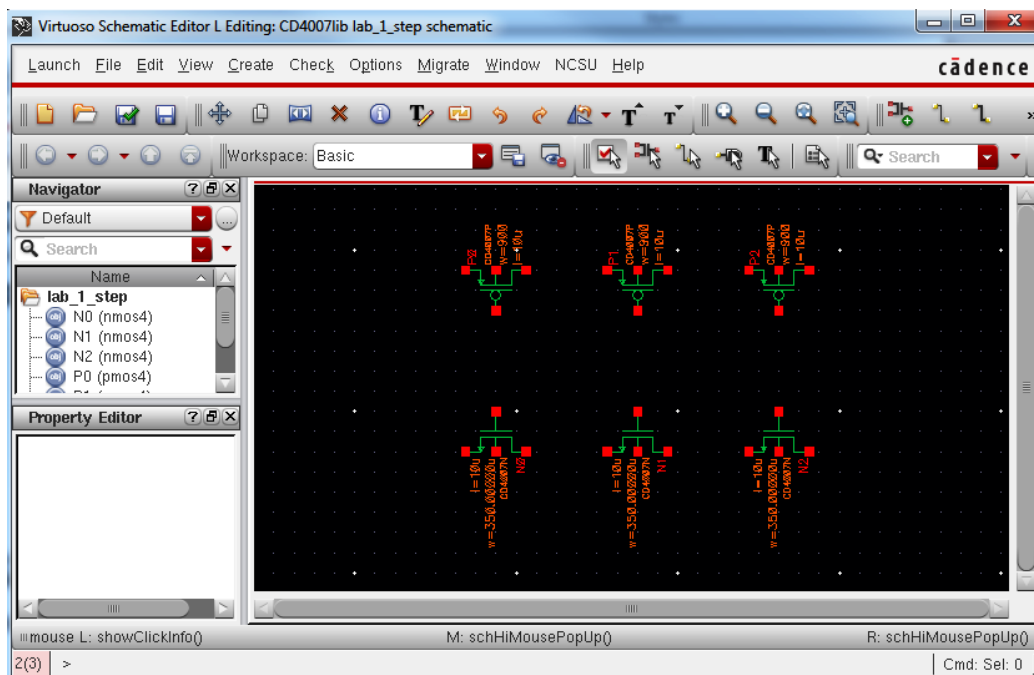
Property	Value	Display
Library Name	NCSU_Analog_Parts	<input type="button" value="off"/>
Cell Name	pmos4	<input type="button" value="off"/>
View Name	symbol	<input type="button" value="off"/>
Instance Name	P0	<input type="button" value="off"/>

CDF Parameter	Value	Display
Model name	CD4007P	<input type="button" value="off"/>
Model Type	<input checked="" type="radio"/> system <input type="radio"/> user	<input type="button" value="off"/>
Multiplier	1	<input type="button" value="off"/>
Fingers	1	<input type="button" value="off"/>
Width (grid units)	90	<input type="button" value="off"/>
Width	900.0000u M	<input type="button" value="off"/>
Width (minimum)	350.0000u M	<input type="button" value="off"/>
Length (grid units)	1	<input type="button" value="off"/>
Length	10u M	<input type="button" value="off"/>
Length (minimum)	10u M	<input type="button" value="off"/>
Drain diffusion area	2.25e-08	<input type="button" value="off"/>
Source diffusion area	2.25e-08	<input type="button" value="off"/>
Drain diffusion perimeter	1.85u M	<input type="button" value="off"/>
Source diffusion perimeter	1.85u M	<input type="button" value="off"/>
Drain diffusion res squares		<input type="button" value="off"/>
Source diffusion res squares		<input type="button" value="off"/>
Virtuoso-XL layout cell		<input type="button" value="off"/>
Drain diffusion length		<input type="button" value="off"/>
Source diffusion length		<input type="button" value="off"/>
Temp rise from ambient		<input type="button" value="off"/>

Click OK when you're done. You should see the new property values next to the MOSFET in the schematic window.



Copy the PMOS transistor as you did with the NMOS before, and your schematic should now look like this:



5. R,C, Sources

Now that all the MOSFETs are in place, the next step is to add the other components in the circuit, and wire them up to complete the schematic.

5.1. Add DC power

- The circuit requires two voltage sources: a DC source for the +5V power supply and a pulse source to provide the square wave input. We'll do the DC power supply first.

Get to the **Create > Instance** Browser Window again. To choose from analog parts, go to the Component Browser Window and choose the **NCSU_Analog_Parts** library. If you are still in the N_Transistors menu, you will need to click the

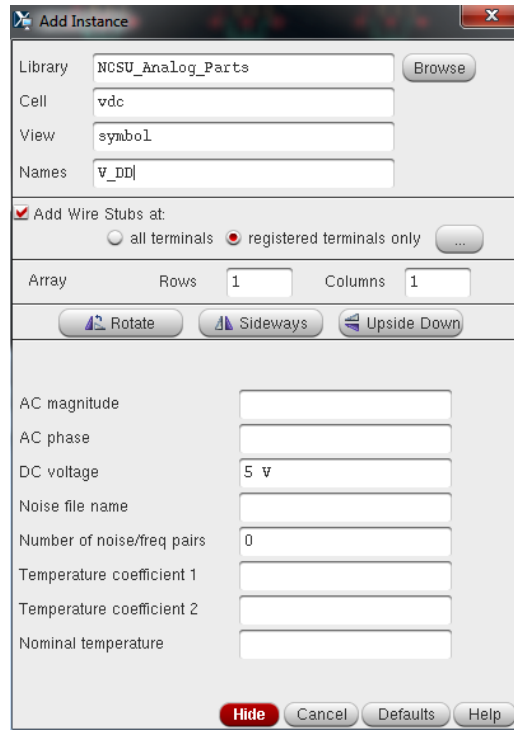
.. (Go up 1 level)

selection to make the Voltage_Sources menu available.

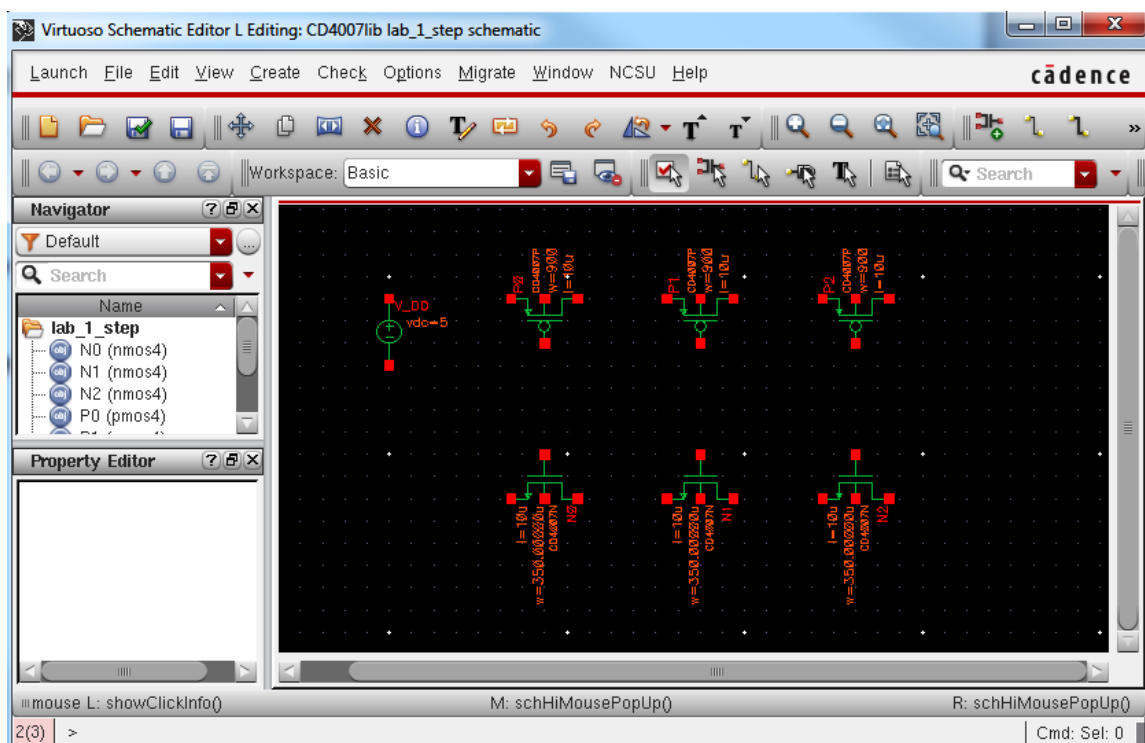
- Select the Voltage_sources entry from the available contents; you will then get a choice of different types of supplies. Choose 'vdc' and place one source in the schematic window. Remember to hit the Esc key to avoid sprinkling voltage sources all over your schematic.
- Now you should name the voltage sources. The software unimaginatively names the sources V0, V1, etc. To provide a little more insight, you should give the sources meaningful names.

You will be presented with the "Add Instance" window (see below), offering a dizzying array of properties for the voltage source. Change the 'Instance name' and 'DC Voltage' appropriately. Consider V_DD for the name of the source. For the voltage, use +5V.

- After changing the name and voltage, click 'Hide' to make the window go away. You should see the new values in the schematic.



And your schematic should now look like this:



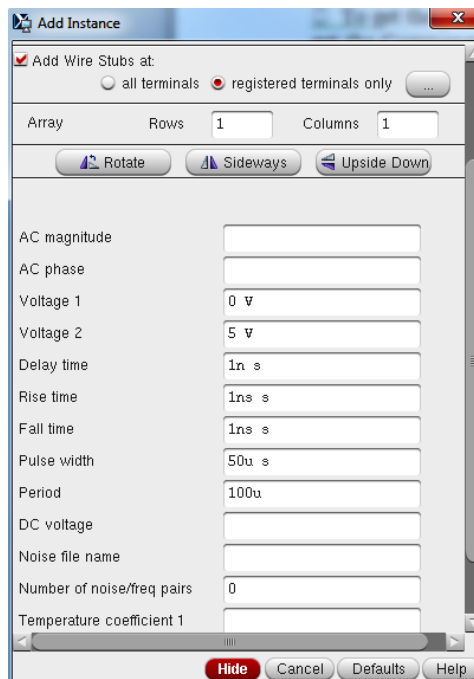
5.2. Add square wave input

To get the pulse voltage source for the square wave input, go back to the **Create > Instance** to get the Component Browser. Select the Voltage_sources entry from the available contents; you will then get a choice of different types of supplies. This time, choose 'vpulse' and place one source in the schematic window. Remember to hit the Esc key to avoid sprinkling voltage sources all over your schematic.

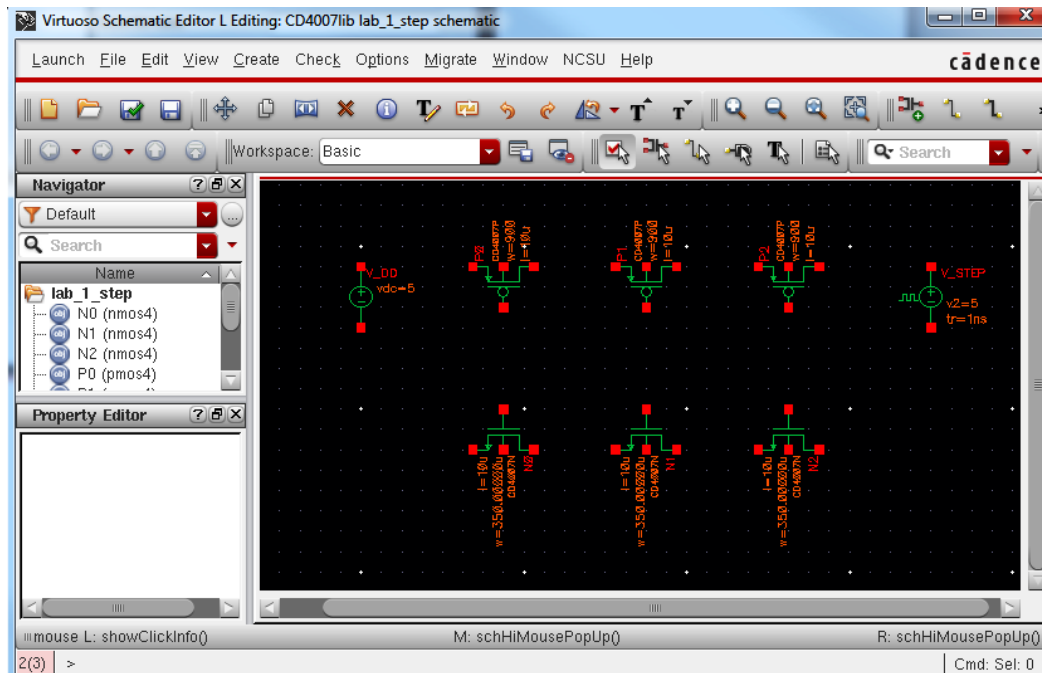
Repeat the selection and naming process for the pulse source. Consider a name like V_STEP.

Note that the pulse source requires many more parameters to specify the waveform:

- "Voltage 1" and "Voltage 2" are the two levels of the square wave. Enter these as 0V and +5V to match the waveform you applied from the function generator in lab.
- "Delay time" is the delay from the start of the transient simulation at time $t=0$ to the first pulse edge. Enter this as 1ns.
- "Rise time" and "Fall time" are the rise and fall times of the pulse waveform in the simulator which represents the output of the function generator. Strictly speaking, we should measure the rise and fall times of the function generator and use those here. However, as long as the function generator is much faster than the rise and fall times of the RC circuit, this isn't critical. Enter these as 1ns.
- "Pulse width" specifies the amount of time the waveform is at the "Voltage 2" level; "Period" specifies the total amount of time for one cycle of the waveform. To make a 50% duty cycle square wave, Pulse width should be half the Period. Since a 10 kHz frequency signal has a period of $1/100 \text{ kHz} = 100\mu \text{ s}$, enter these as 50 $\mu \text{ s}$ for Pulse width and 100 $\mu \text{ s}$ for Period.

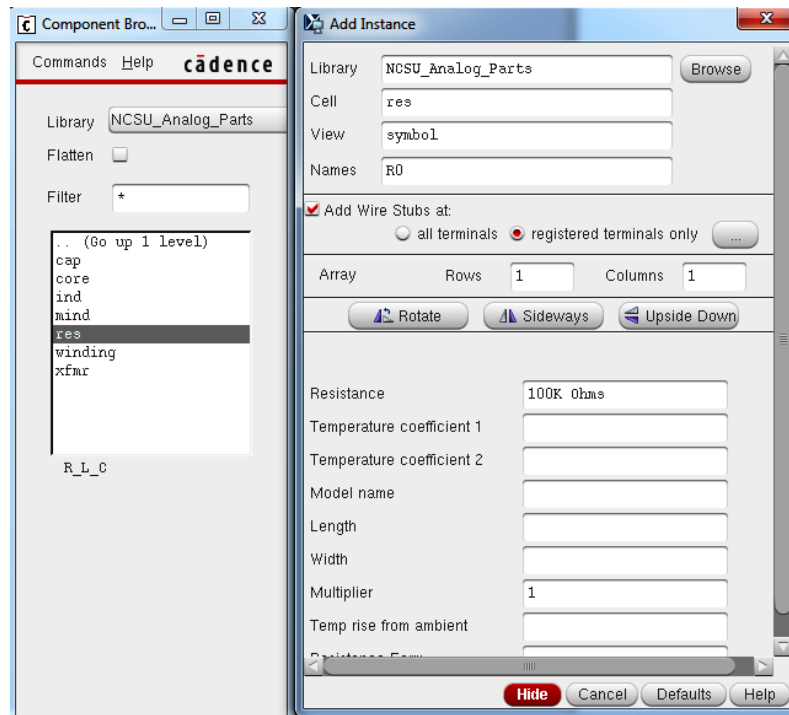


And your schematic should now look like:



5.3. Add 100 k Ω series resistor

Using the Create > Instance menu again, choose
res
from the
R_L_C
menu of the **NCSU_Analog_Parts** library. In the Add Instance form, change the "Resistance"
Value to 100K Ohms.



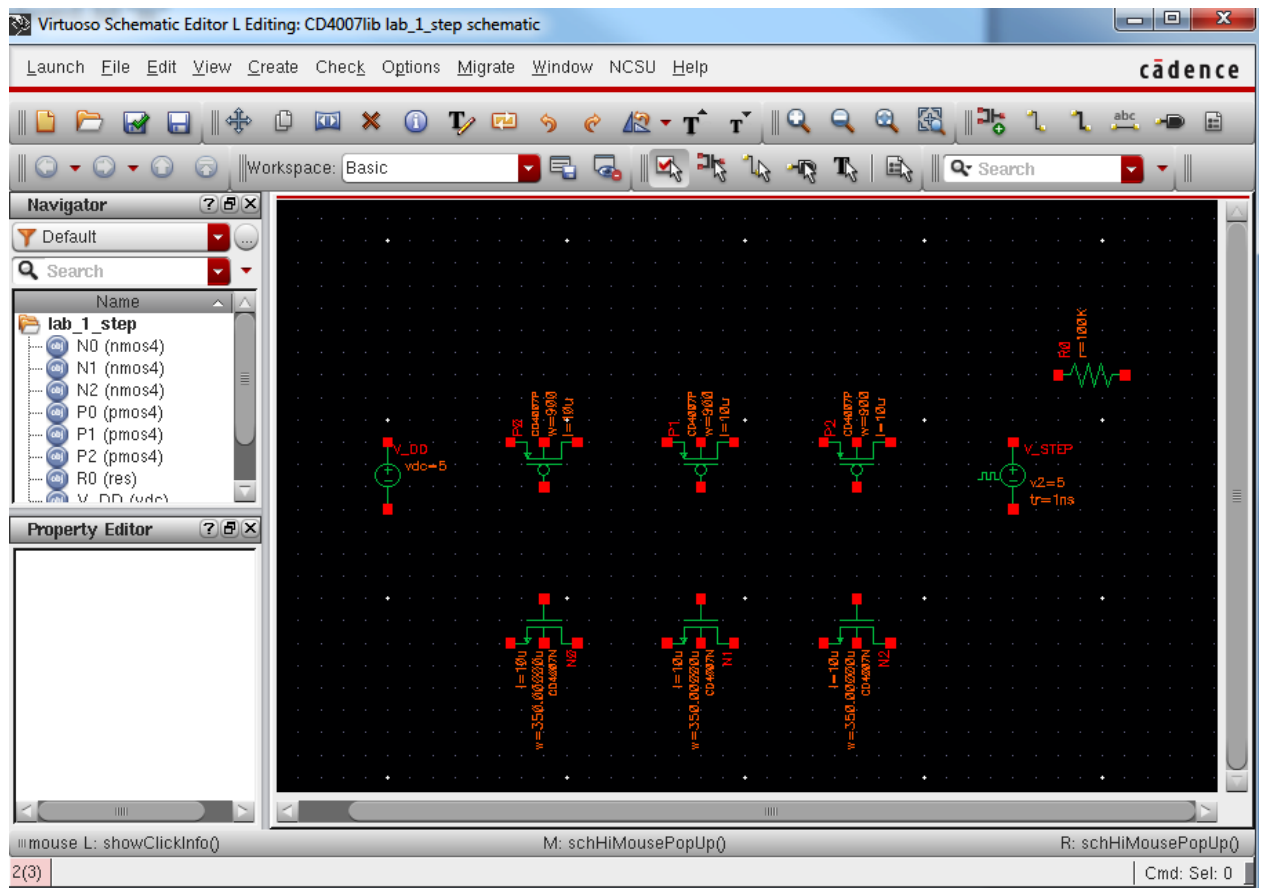
You may want to move or rotate a part after you've placed it. To move a part, use

- From the pull down menu: select Edit > Move
- From the keyboard: hit "m"

To rotate a part, use

- From the pull down menu: select Edit > Rotate
- From the keyboard: hit "r"

Now your schematic should look like



5.4. Add Capacitor to model parasitics

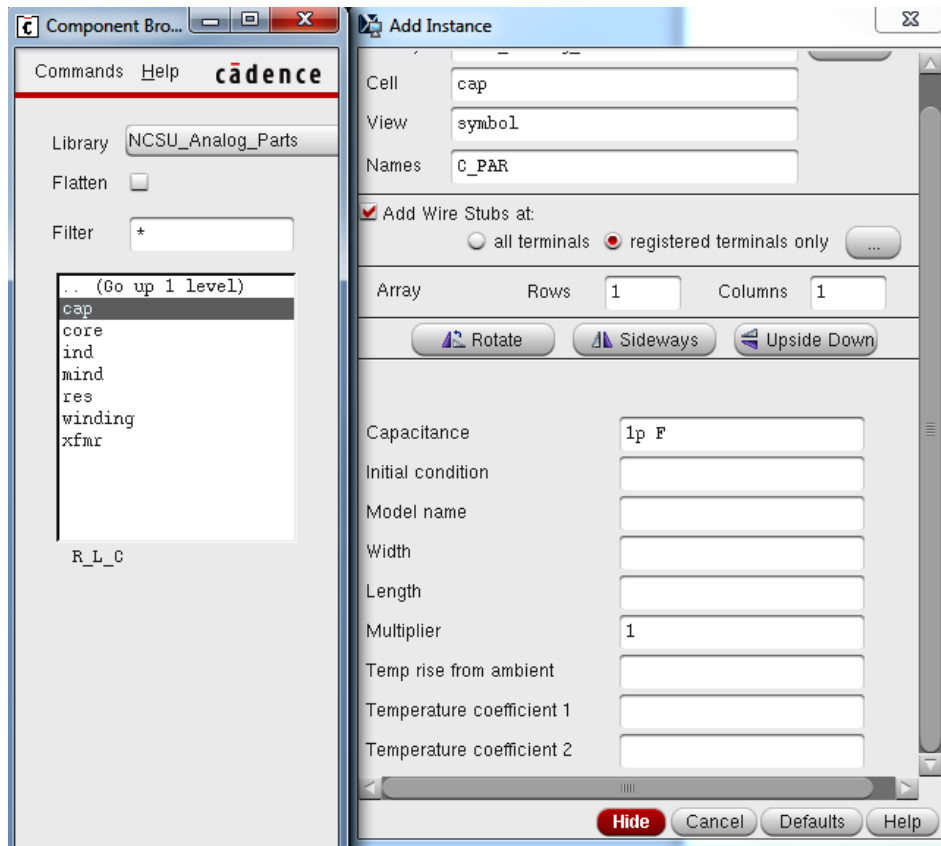
The MOSFET model has parasitic capacitances included; however to model the parasitic capacitance of the breadboard and scope we need to include an explicit capacitor in the schematic.

Using the Create > Instance menu again, choose

cap
from the

R_L_C

menu of the **NCSU_Analog_Parts** library. In the Add Instance form, enter "C_PAR" for the component name to remind you that it represents PARasitics. For now, leave the Capacitance entry at its default value of 1pF.

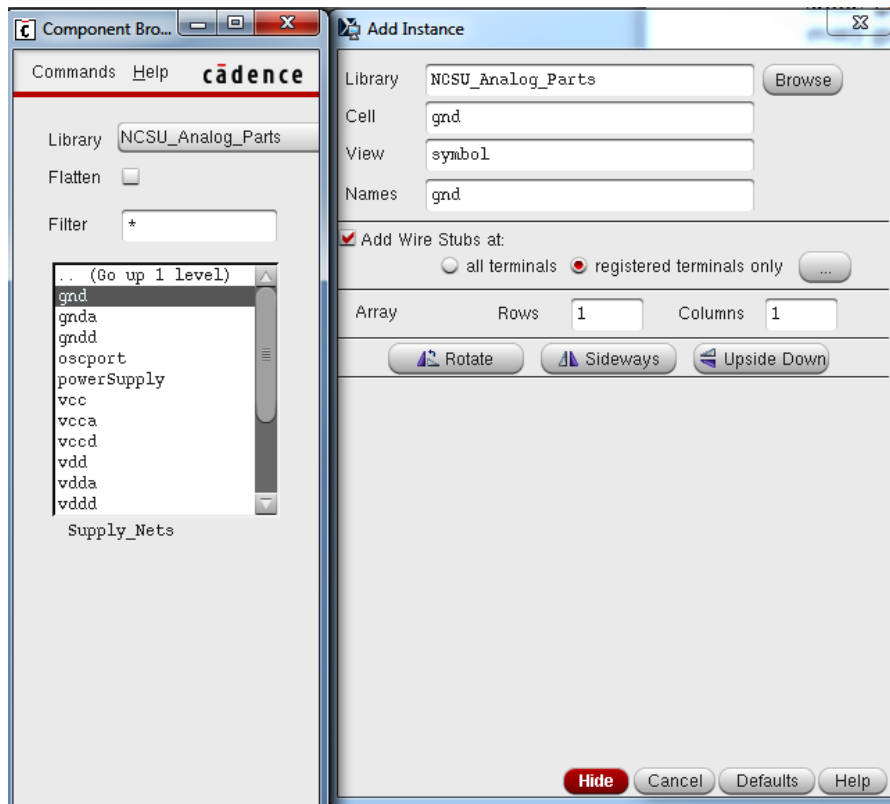


5.5. Add ground reference node

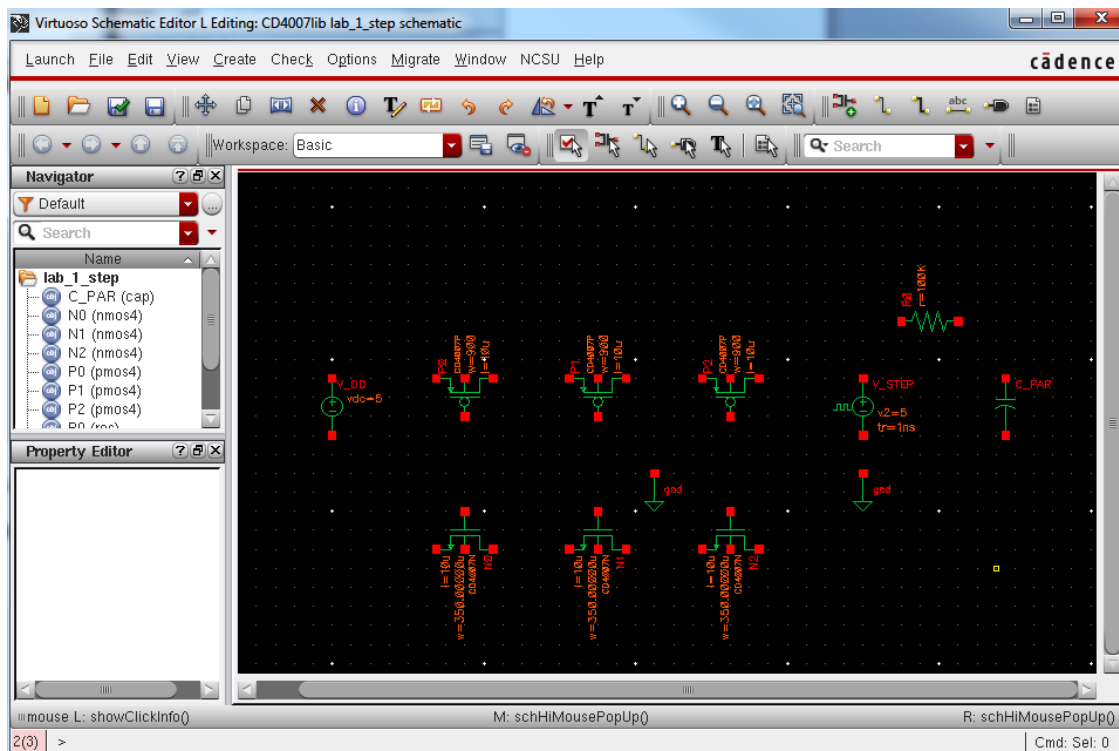
Next, go to the **NCSU_Analog_Parts** library and select the **Supply_nets** entry. Select and add some '**gnd**' symbols to the schematic. Note that you could just place one **gnd** symbol and wire every grounded terminal to that '**gnd**' symbol. However, this results in a very busy schematic with lots of wires. Usually, you will end up with a more readable schematic if you use plenty of **gnd** symbols wherever a terminal is connected to ground. Again, you may need to click the

.. (Go up 1 level)

selection to make the **Supply_nets** menu available.



Now your schematic should look like

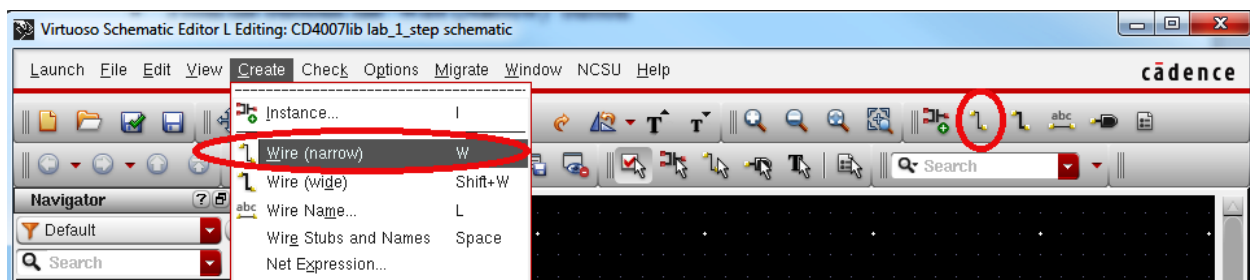


And finally all necessary components are placed!

5.6. Wire it up

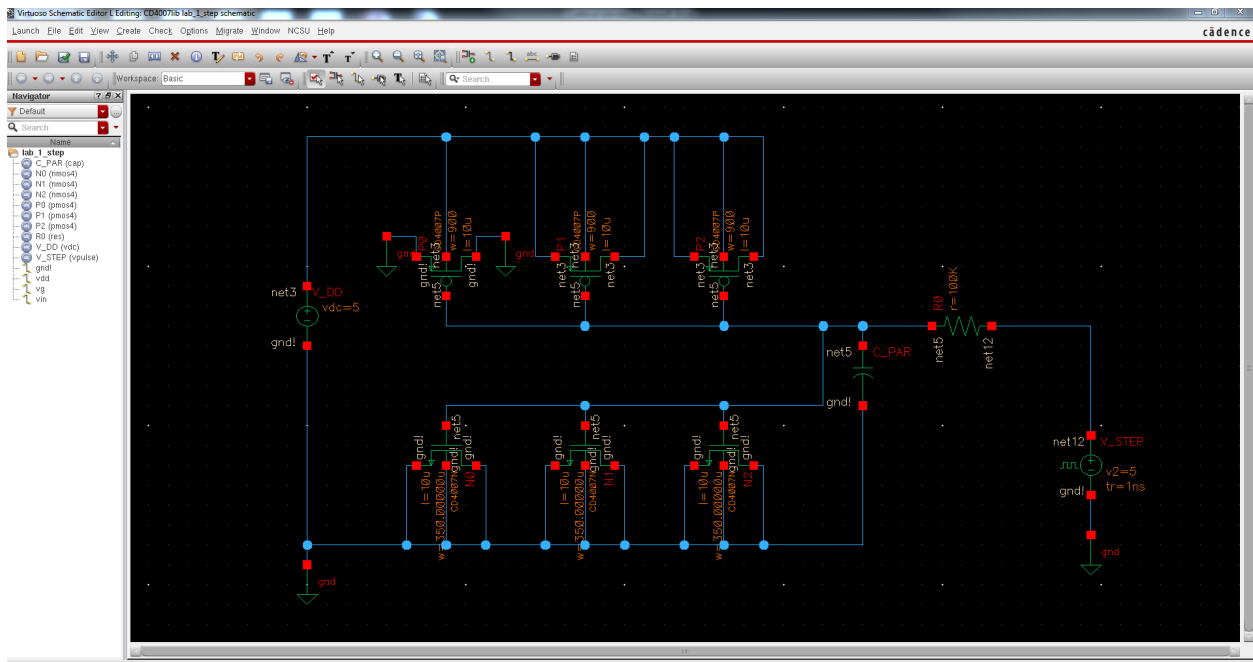
You can get to the wiring tool in one of three ways:

- From the pull down menu: select Create > Wire (narrow)
- From the keyboard: hit "w"
- From the buttons: the "Wire (Narrow)" button



Click the left mouse button to place ends of the wire. Connection points (terminals) are indicated with red squares on the boundary of each symbol. Connect wires until all terminals are connected appropriately (don't forget the substrate terminal!). Hit the Esc key to get out of the wire placement mode and back into the selection mode.

Now your schematic should look like



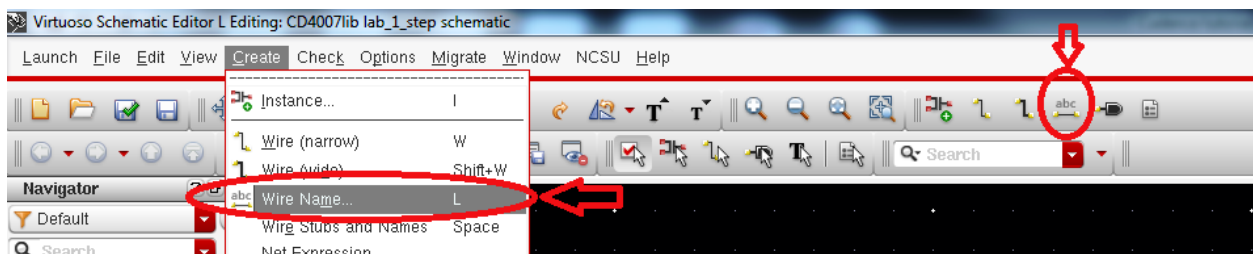
5.7. Name signals

In complicated designs, it's a good idea to give the nodes names. Although this circuit isn't complicated, we'll do it in this circuit to get in some practice.

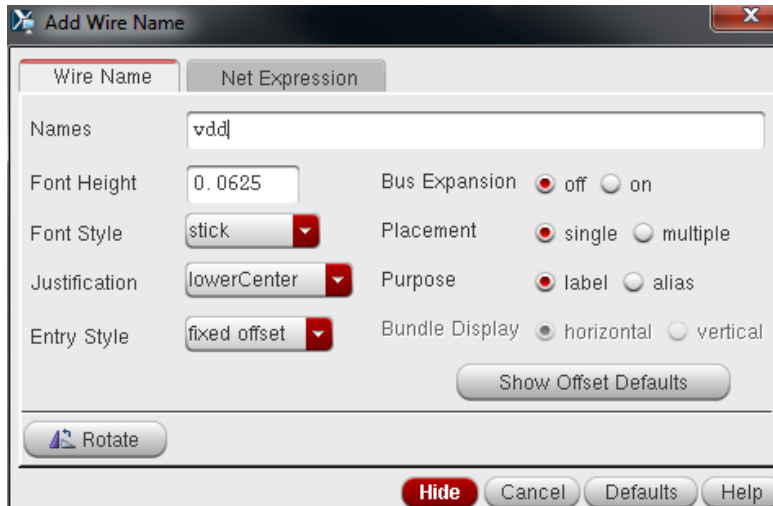
First select the wire to be named. Click on the wire to select it; you'll know the wire is selected when it's highlighted in white.

Let's name the node where all six gates are tied together "vg". You can get to the wire name tool in one of three ways:

- From the pull down menu: select Create > Wire Name
- From the keyboard: hit "l" (lower case L)
- From the buttons: the "Wire Name" button



The Add Wire Name form opens up; type in the name and hit return. The name appears in the schematic window; move the mouse to place it near the wire you're naming and click the left mouse button to place the name.



Repeat to assign the name

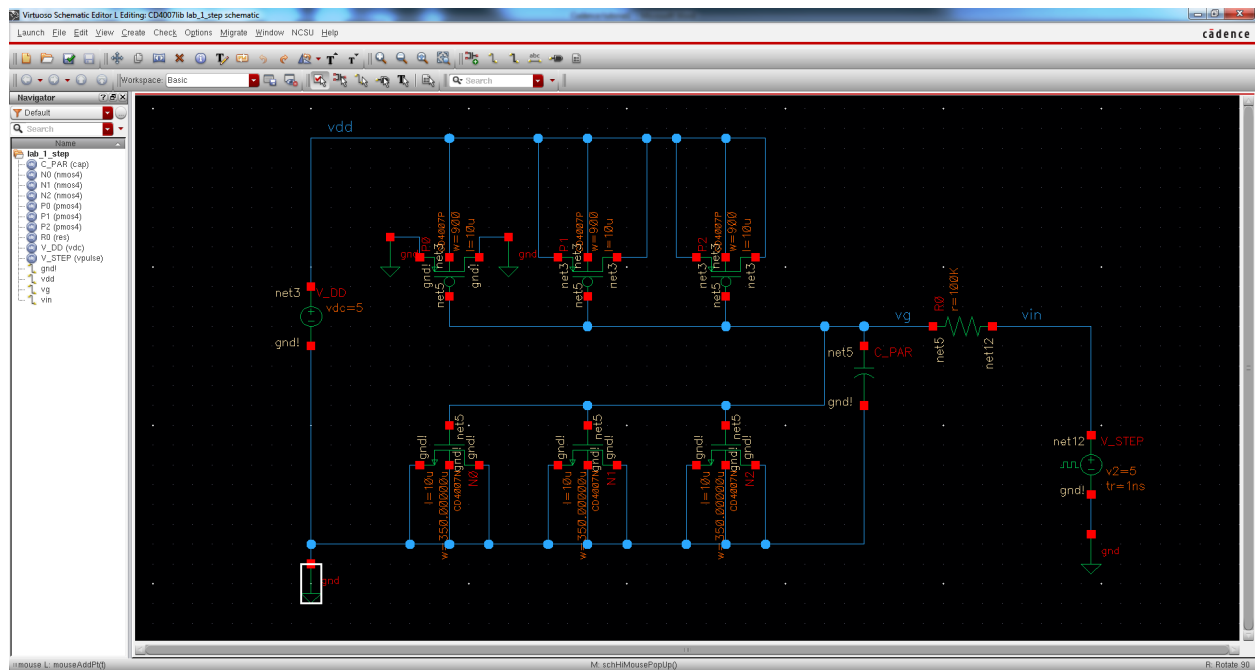
`vin`

to the input node (the output of the pulse generator), and

`vdd`

to the VDD = +5V supply rail.

Now your schematic should look like this:



Check and **Save** your schematic, and now it's time to simulate!

6. Model parameters, NMOS and PMOS

For your simulation results to match the lab measurements, you need to have the correct model parameters in the simulator. You need to **create your own model parameter file**, which is done in this part.

The model file needs to be in the directory '**cadence61/models/spectre**' that you created earlier. Using any unix text editor you're comfortable with (could be **emacs** from the unix terminal window), create a file called

CD4007N.m

which has EXACTLY the following format:

```
.MODEL CD4007N NMOS ( LEVEL=3 &
TOX=mytox &
)
```

For the PMOS devices, create a file called

CD4007P.m

which has EXACTLY the following format:

```
.MODEL CD4007P PMOS ( LEVEL=3 &  
TOX=mytox &  
)
```

Caution!!! Cutting and pasting the above text may not work. You can use (Ctrl-y) to paste it in emacs.

Note: In your eventual model file, there will be a numerical value in place of the variable name

mytox

One huge advantage of cadence's Spectre simulator is that you can define variables in model files that can be adjusted based on simulation outputs to get the model parameters "just right" so that simulated and measured data agree.

Check and save your design and you're ready to simulate.

7. Transient simulation

Here you simulate the time domain response of the circuit, using transient analysis in SPICE.

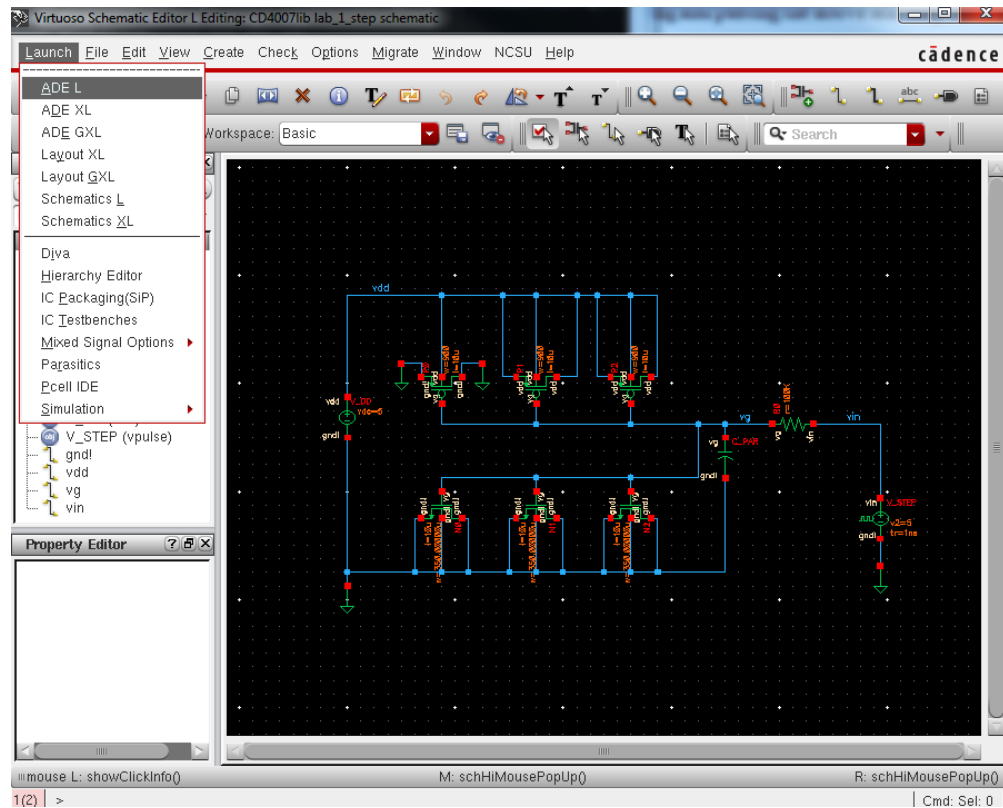
The resulting waveform should look similar (sort of) to what you measured in lab, but the simulation results will not match your lab results yet, since the model parameters will be different. That's OK; the point now is just to get the simulation working. You will change the model parameters in the next part.

7.1. Open simulation tool

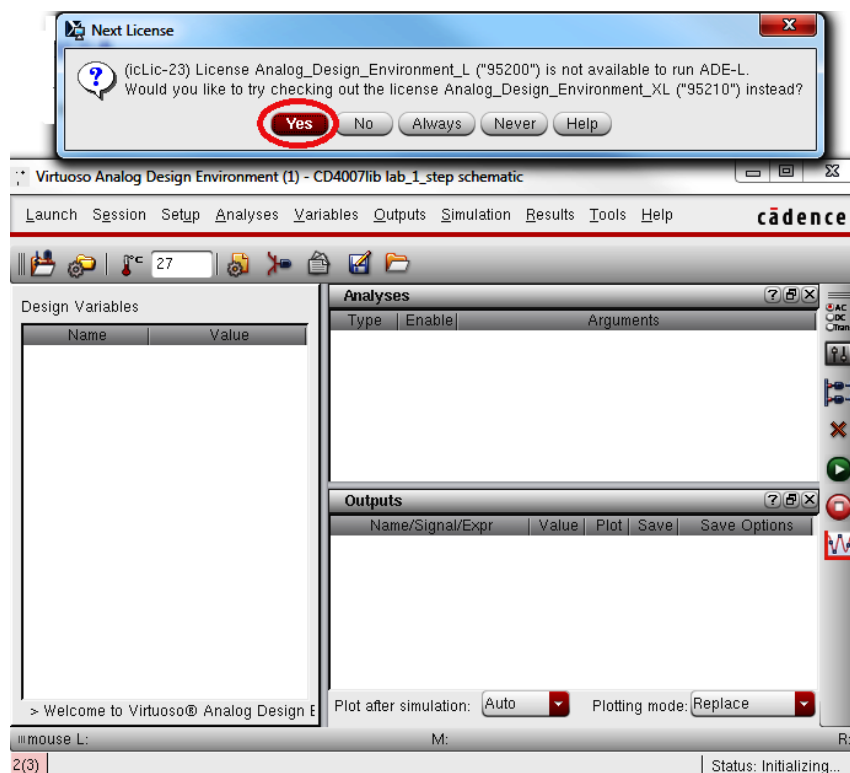
To open the analog artist simulation tool, go to

Launch > ADE L

This will open the Analog Design Environment tool.



. On “Next License” menu, click on YES!



7.2. Specify Transient analysis and time range

Go to the pull-down menu

Analyses > Choose,

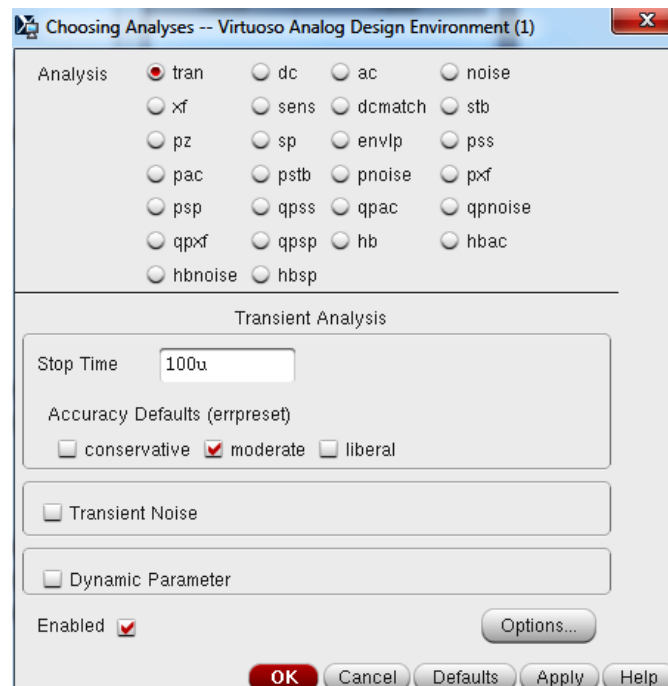
and select

tran

in the Choosing Analyses window. We want to sweep time over at least one full cycle of the square wave to duplicate in simulation the measurement made in lab. So enter

100u

in the "Stop Time" box. Also, click on "Moderate" for the Accuracy Defaults which determines the behavior of the numerical algorithms used in solving the Transient Analysis equations. Click on OK when you're done.

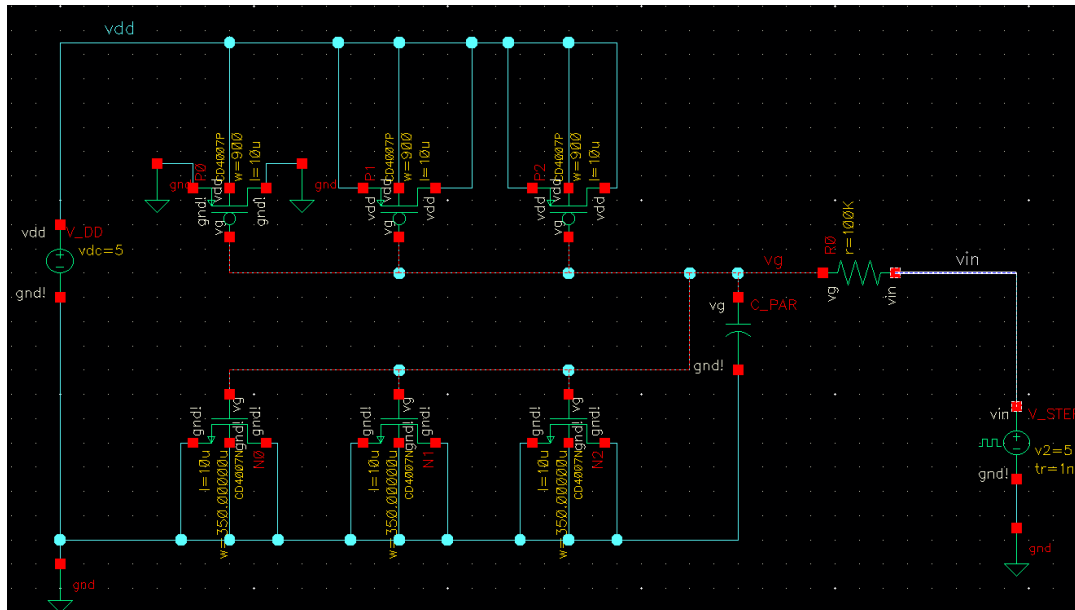


7.3. Select outputs plot

We want to plot the gate voltage 'vg' as a function of **time**, to measure the **rise and fall times** and compare the simulated values to what we measured. From the pull-down menu in the Analog Design Environment window, choose

Outputs > To Be Plotted > Select On Schematic.

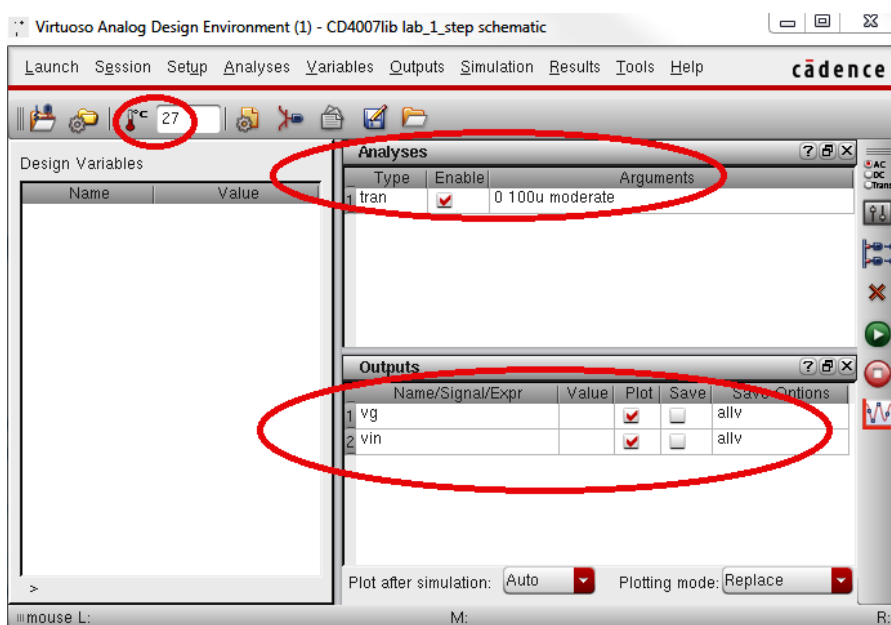
Then click on the **vg** wire; when you do the wire should be **highlighted** in a striped pattern (see below). This means the voltage at that node will be plotted. Also click on **vin**, just for comparison. The schematic should now look like:



Return to the Analog Design Environment window. You should see that the Type of Analysis is indicated as

tran

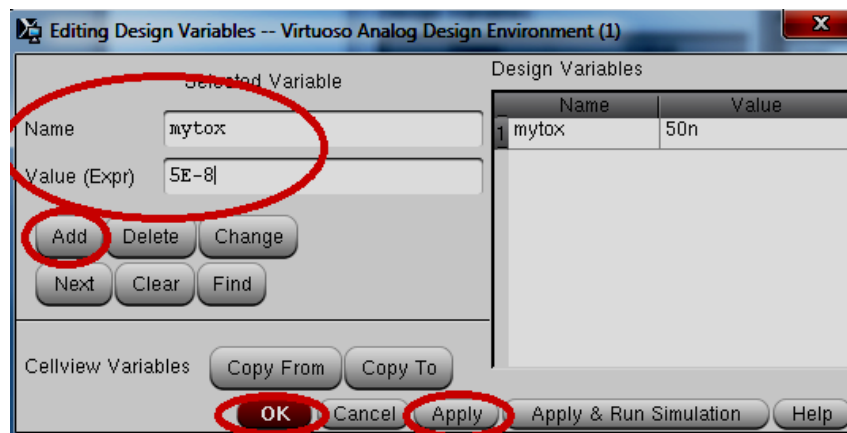
with Arguments **0** and **100u** (indicating the **start and stop times** for the simulation). Also, under the Outputs section, you should see that **vin** and **vg** are selected to be plotted.



Before doing the simulation, you need to define a value for the variable '**mytox**' so the simulator has a number to work with. From the **Variables > Edit ...** pull-down menu, enter

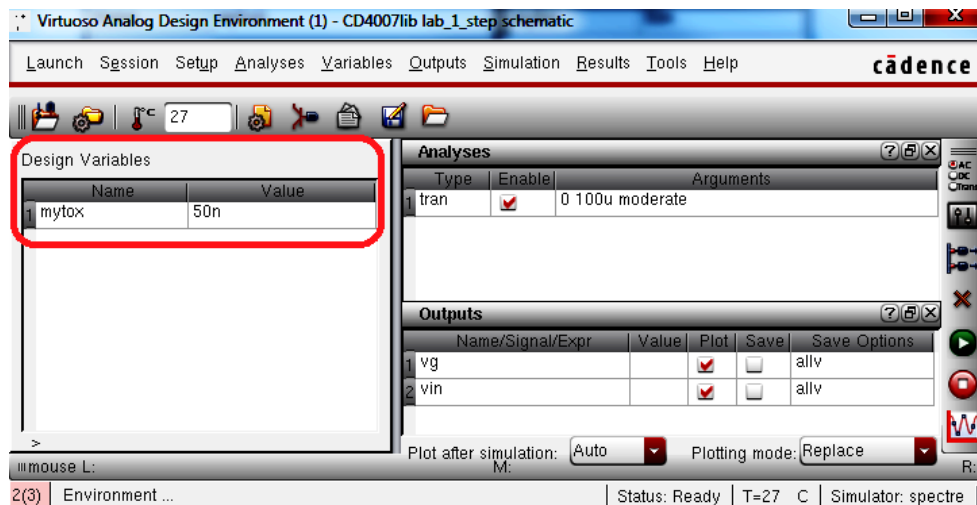
mytox

as the Name and a Value (Expr) of something reasonable, say 5E-8, just to get something to simulate.



Then click 'Add' and 'Apply' to enter this value in the Table of Design Variables. Then click "OK" to get back to the Analog Design Environment to run your simulation.

The Environment window should now show the newly defined mytox variable:



7.4. Setup the simulator

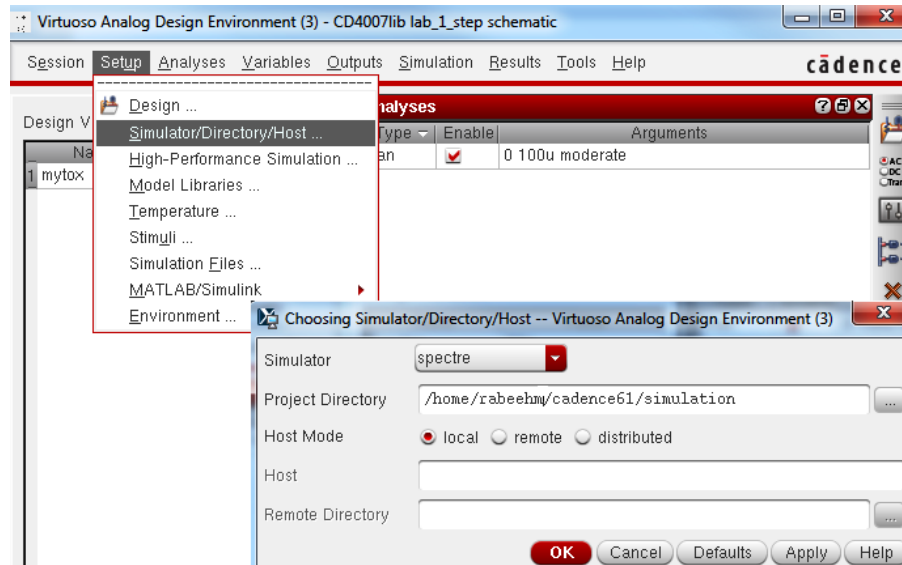
In Virtuoso Analog Design Environment, click on Setup => Simulator/Directory/Host.

- Choose spectre as the Simulator (if not default).

- The default Project Directory should be “. /simulation”. This points to a directory named

“simulation” inside of the directory you launched Cadence from.

This is where your simulations files and results will be saved.



7.5. Setup Model files

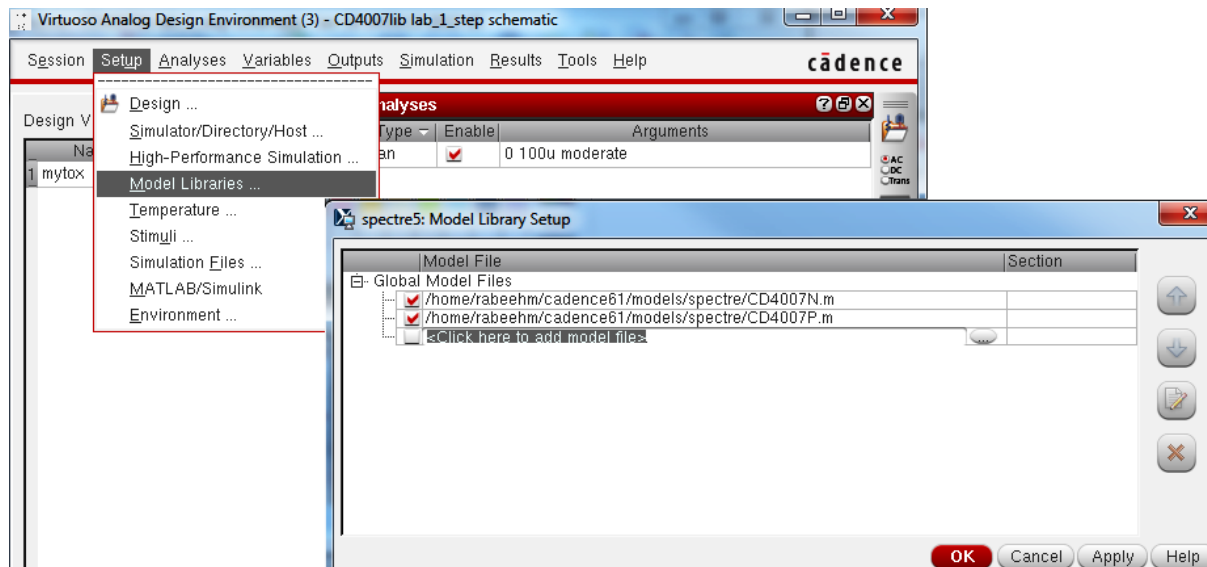
In Virtuoso Analog Design Environment, click on Setup => Model Libraries...

You can add the necessary Model Library Files by click the bottom “Model Library File” box

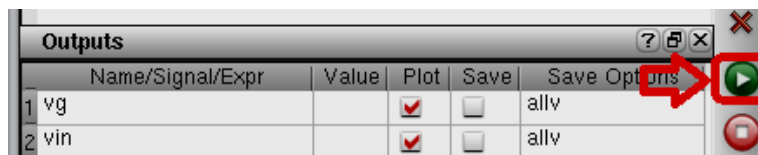
Click the “Add” button and then add two model files as:

“/home/username/cadence61/models/spectre/CD4007N.m”

“/home/username/cadence61/models/spectre/CD4007P.m”



You are ready to simulate! Click on the green light and stand back ...

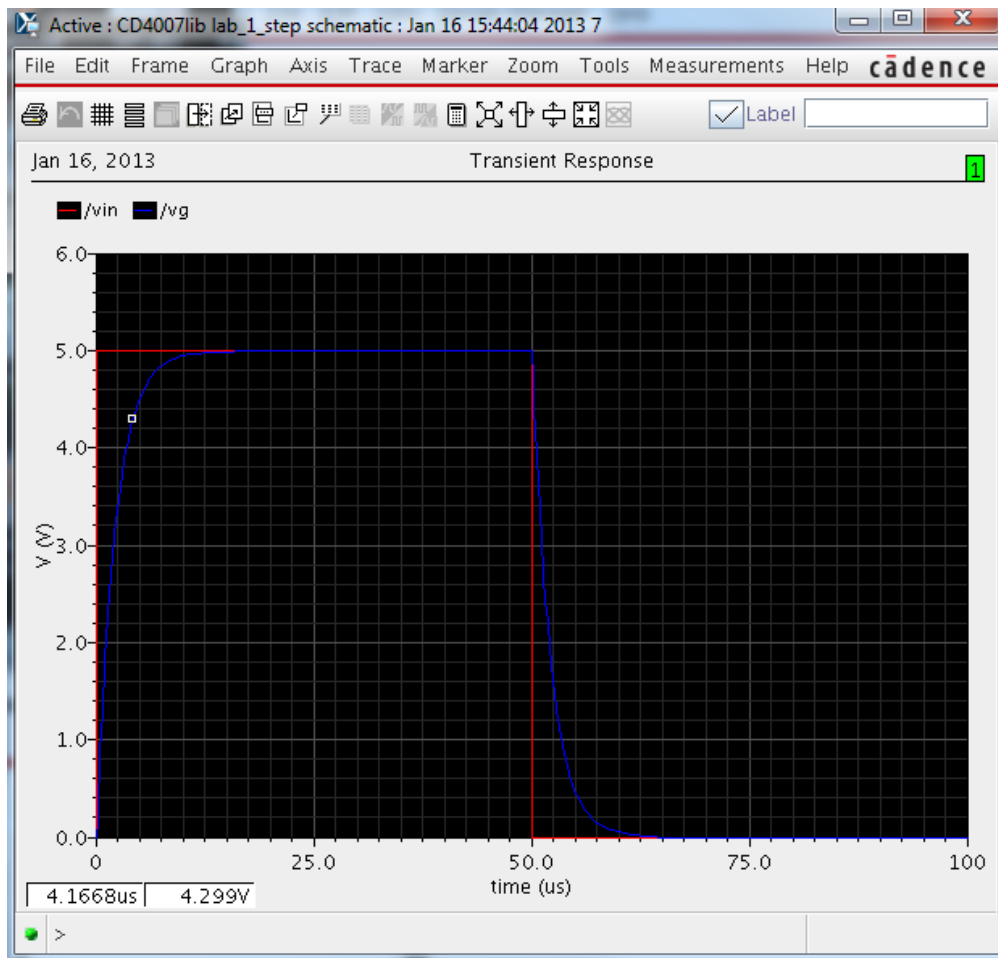


7.6. Start the simulation

In the Analog Design Environment window, you could also choose

Simulation > Run

to start the simulation. A warning window may pop up. Just click 'yes'. Then, after much thought on the part of the simulator, you will get an output waveform plot that looks something like the following:



The general shape is sort of OK, but the rise and fall times are not the same as what you measured because

1. the value of the parasitic capacitance, 1pF, is too low, and
2. the value of **tox** was chosen somewhat arbitrarily.

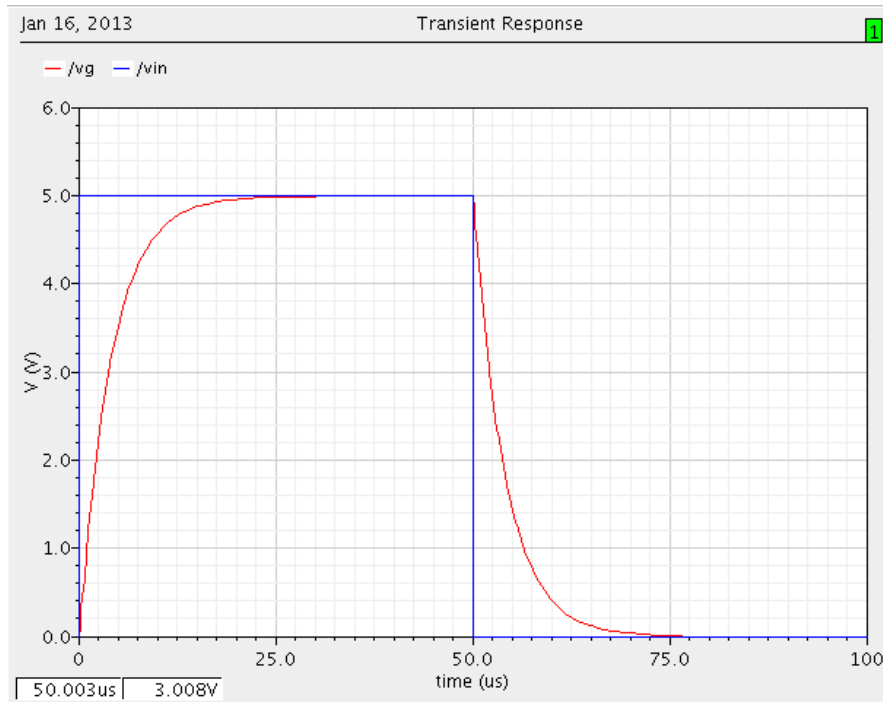
You'll fix that in the next part.

7.7. Parameter Adjustment

The goal now is to get the simulated rise and fall times as close as possible to your measured values. To start, you need to change the value of the parasitic capacitance from 1pF to the value you measured in the lab. Next change the **mytox** value to the value you calculated in W1-7 in the lab.

You will probably need to adjust the value of the **mytox** parameters to get reasonable agreement between your measured data and the simulated rise and fall times. The nice thing about having the mytox variable is that you can adjust the parameter value in the Analog Design Environment, without having to edit the model file and restarting the Environment every time.

Now the waveform plot should show a reasonable rise time:



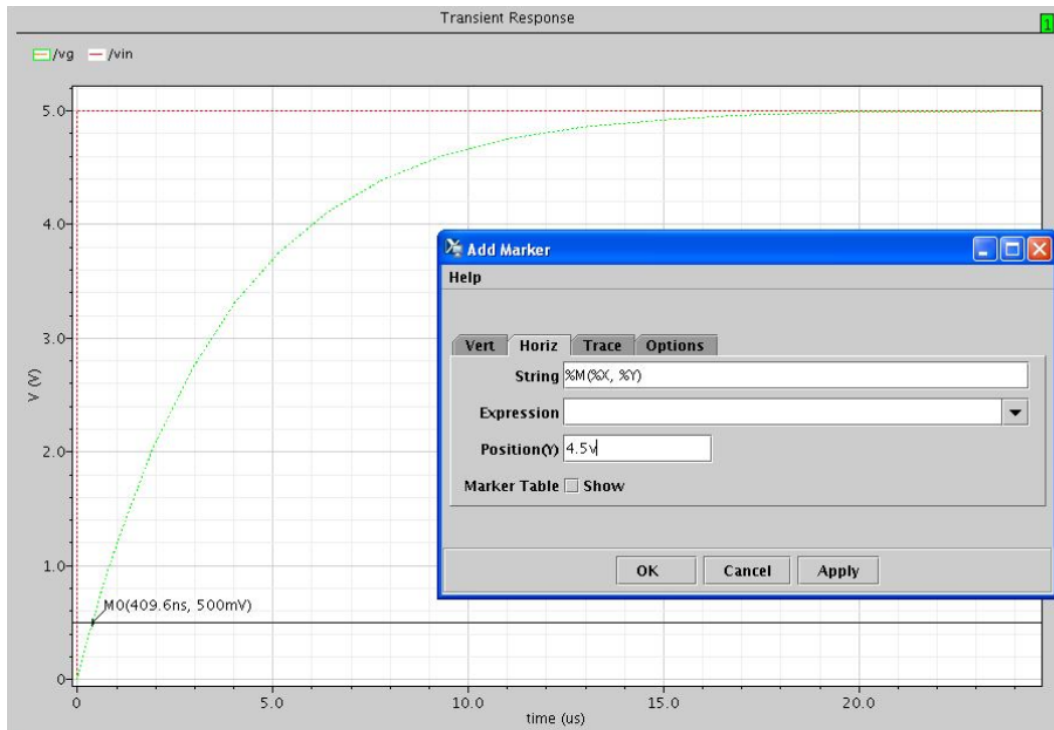
Using the Zoom tool, look more closely at the rising edge.

To measure the rise time of the gate voltage, you need to add a few markers to the plot. To add a marker use the top menus and choose

Marker > Add

You're looking for time values, but you know the voltage values that correspond to 10% and 90% of the waveform, so you want to use a horizontal trace. Click on the "Horiz" tab and add markers corresponding to the 10% and 90% voltages (0.5v and 4.5v for this case).

The default label for the markers shows the voltage as well as the corresponding time for that voltage value. The rise time is thus the difference between these two times.



In this case, the difference (delta) on the x-axis (time) is 8.23us. This number can be compared with the measured results from the lab and mytox adjusted accordingly until agreement is obtained.

7.8. Saving the plot as an Image

When all is said and done, you'll want some way to prove that your work was successful. The easiest way to do this is to save the graphs as an image that you can then put into your lab report. To do this choose

File > Save as Image

Browse to a folder on your user drive where you want to save the picture and click **save**.

Hooray!