INVERTER TUTORIAL

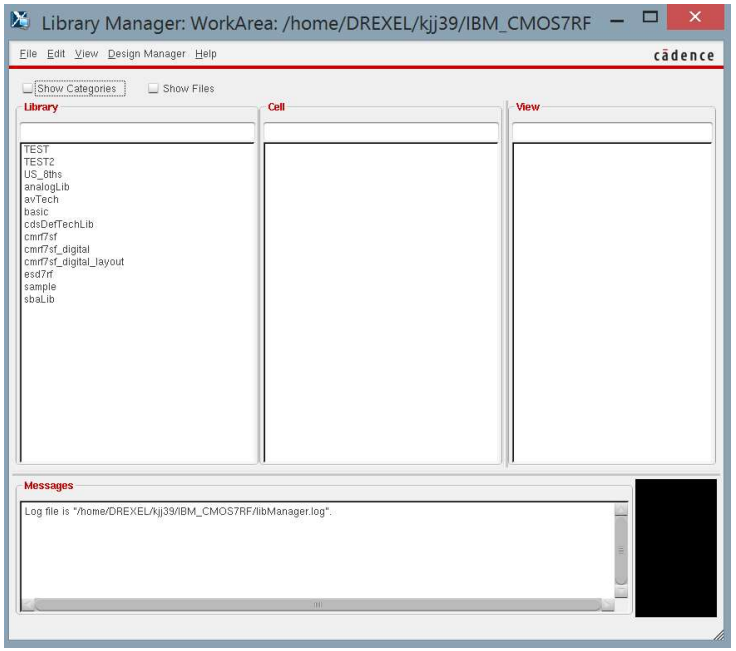
2 Using Cadence simulator

- Start cadence: type “**run_virt**” (or the name of the alias used in your .bashrc if the default run_virt was altered), then hit “enter”.

2.1 Panels in virtuoso

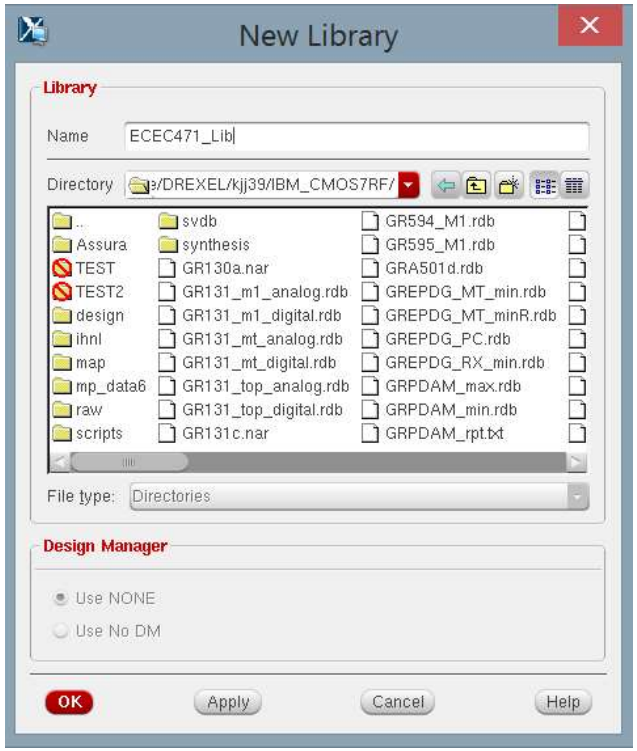
- CIW (Command interpreter window)

- In the CIW, select **Tools ->Library Manager…** The **“library manager”** will pop up.



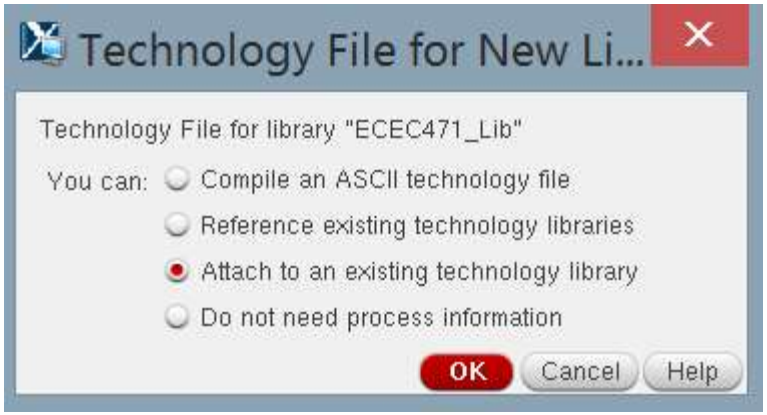
2.2 Create a new library

- In library Manager window, choose "**File-> New-> library…**"
- Enter the name of your library in the "New Library" window, such as: ECEC471_Lib or ECEC571_Lib
- Directory is the location where the library will be created.
- Hit "OK"



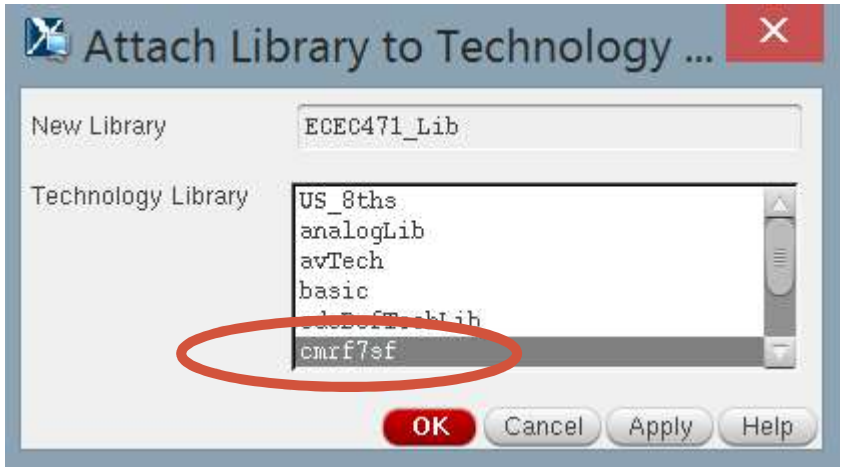
2.2 Create a new library

- After you hit “OK”, the **“Technology File for New Library”** will pop up.
- Select **“Attach to an existing technology library”.**
- Hit **“OK”.**



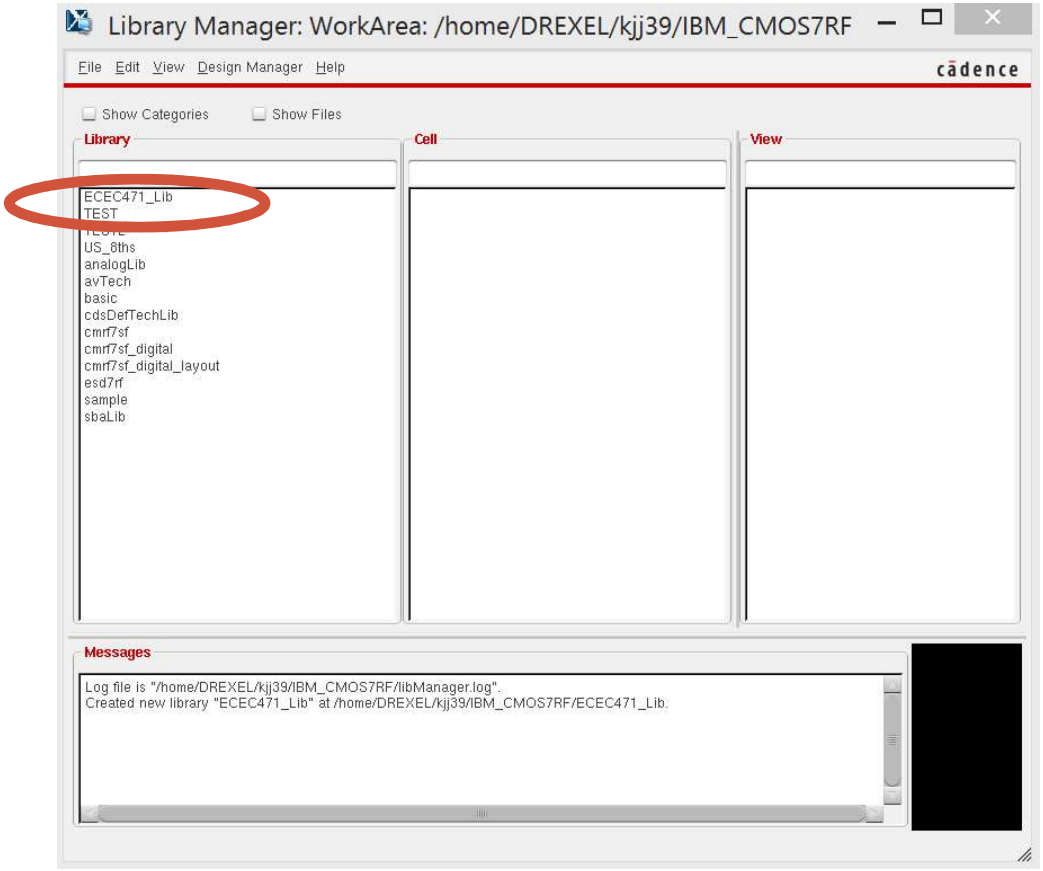
2.2 Create a new library

- After you hit “OK”, the **“Attach Library to Technology Library”** will pop up.
- Select **“cmrf7sf”.**
- Hit **“OK”.**



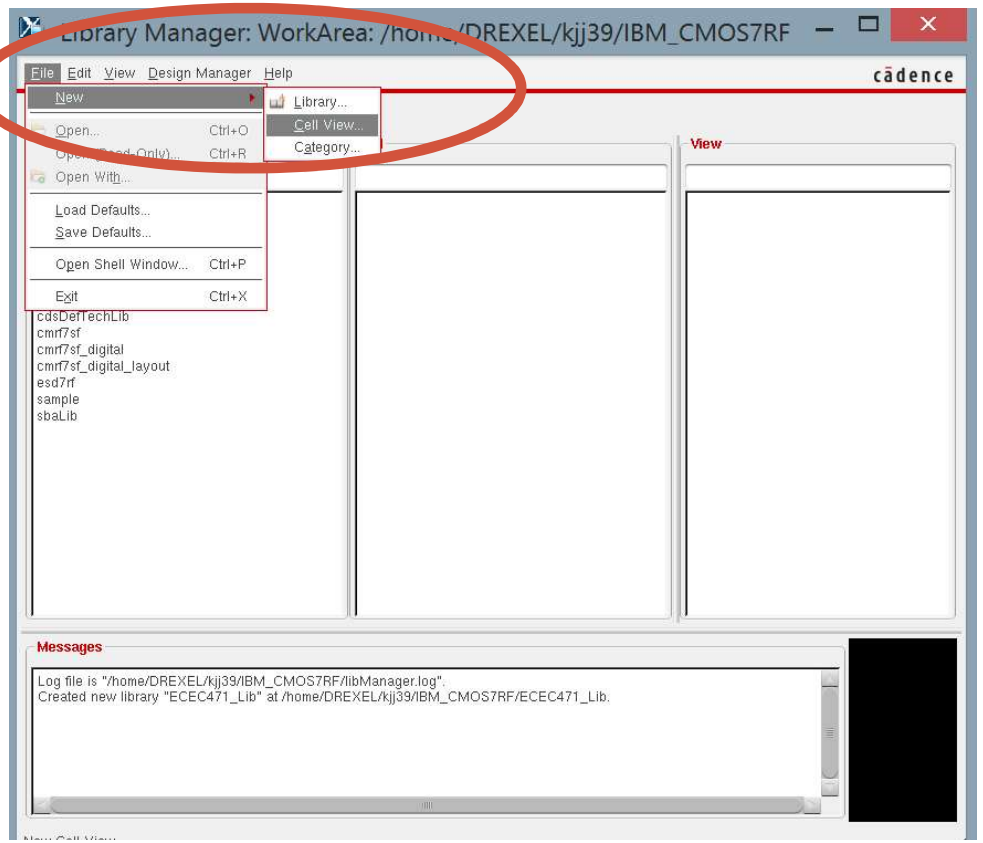
2.2 Create a new library

- The new library “ECEC471_Lib” is created and you can see it in the library manager.



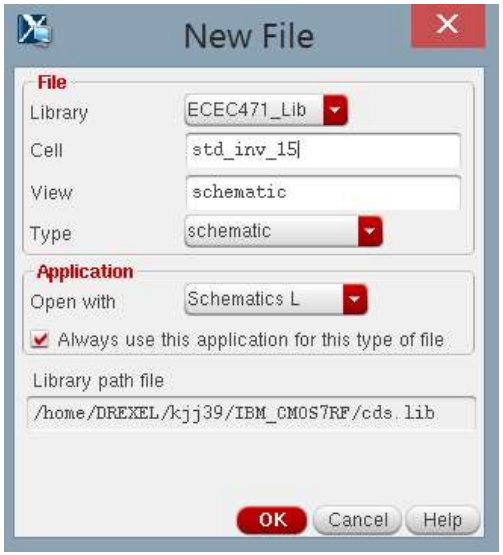
2.3 Create a new schematic view

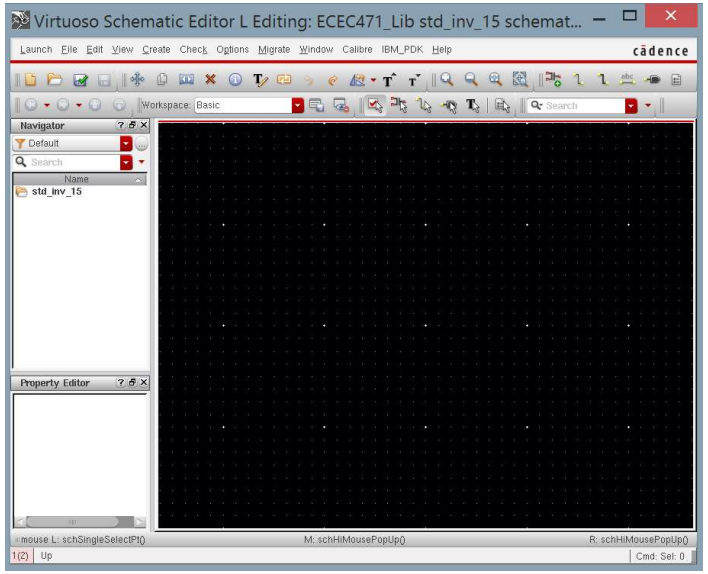
- In the “Library Manager”, select the newly created library ex. “ECEC471_Lib” in the “Library” column.
- Click on “**File-> New->Cellview…**”



2.3 Create a new schematic view

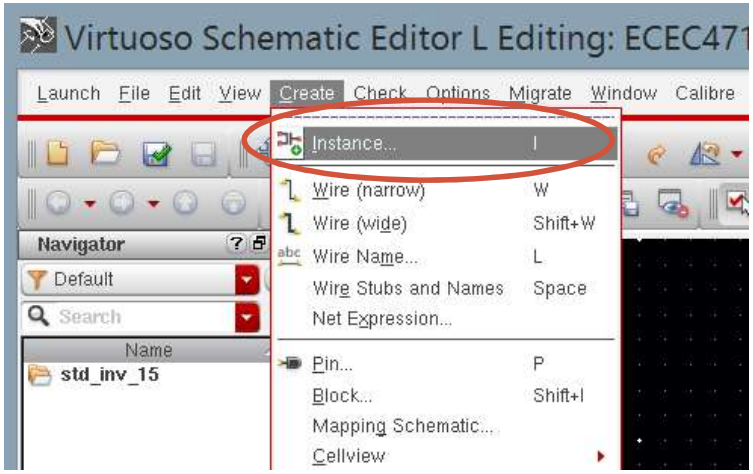
- The "**Create New File**" window will pop up.
- Library name: ECEC471_Lib
- Tool: Composer-Schematic
- Hit "OK"
- Schematic window pops up.
- Make sure Always use this application for this type of file is checked

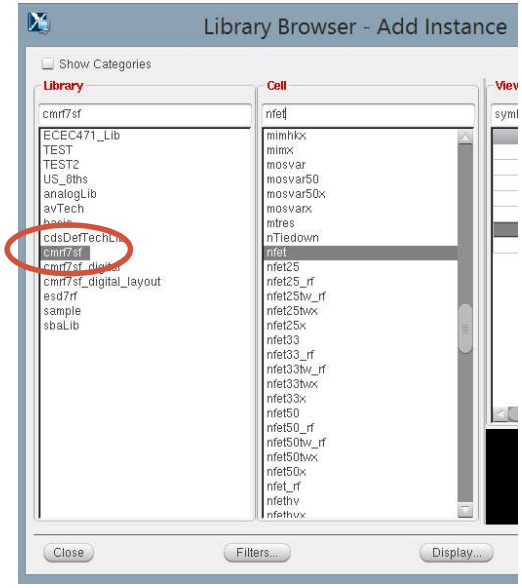




2.4 Create a schematic view of inverter

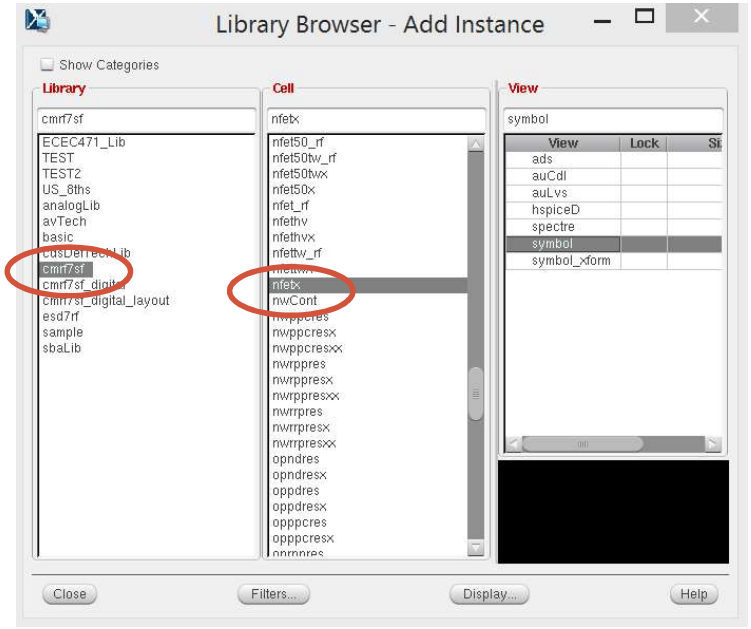
- Add instance:
  - To add an instance into your schematic, e.g. an NMOS device, activate the schematic window, click “Create-> Instance…”(or type i)
  - The “add instance” and “component Browser” dialogue boxes appear.
  - If the “Component Browser” does not appear, click on “browse” in the “Add instance” dialogue box to start it.

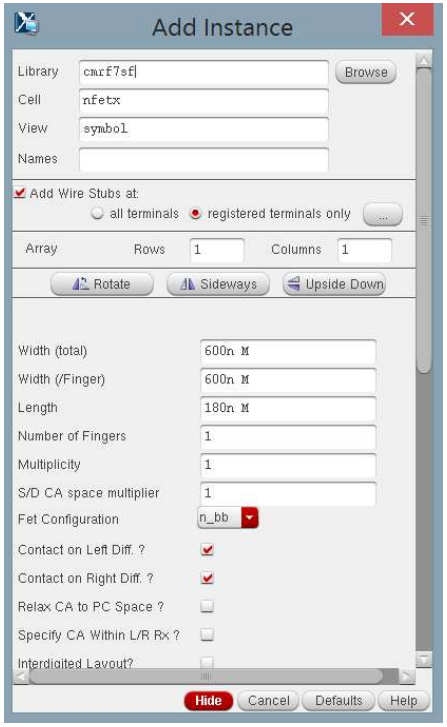




2.4 Create a schematic view of inverter

- In “Component Browser”:
  - Choose library “**cmrf7sf**”.
  - Click on **nfetx** within the cell list.
  - Place the transistor on the schematic editor



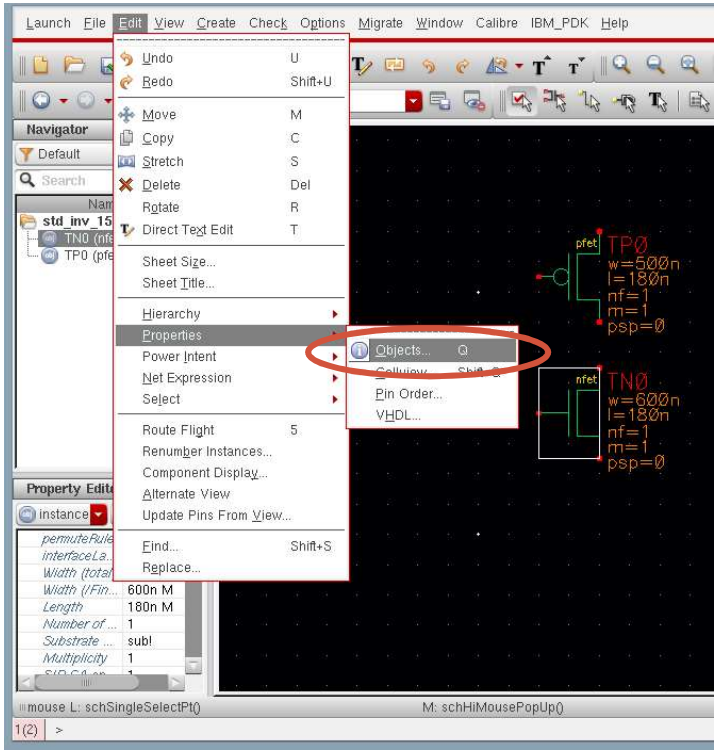


2.4 Create a schematic view of inverter

- To place this instance into your schematic:
  - Activate schematic window, click left button of your mouse.
  - Note that in Cadence schematic composers and layout editor, a command **WILL NOT** terminate until the user terminate it or start a new command. After you place the NMOS instance, you can see there is another NMOS instance ready to be place. To terminate the current operation(which is add instance), press **ESC**.

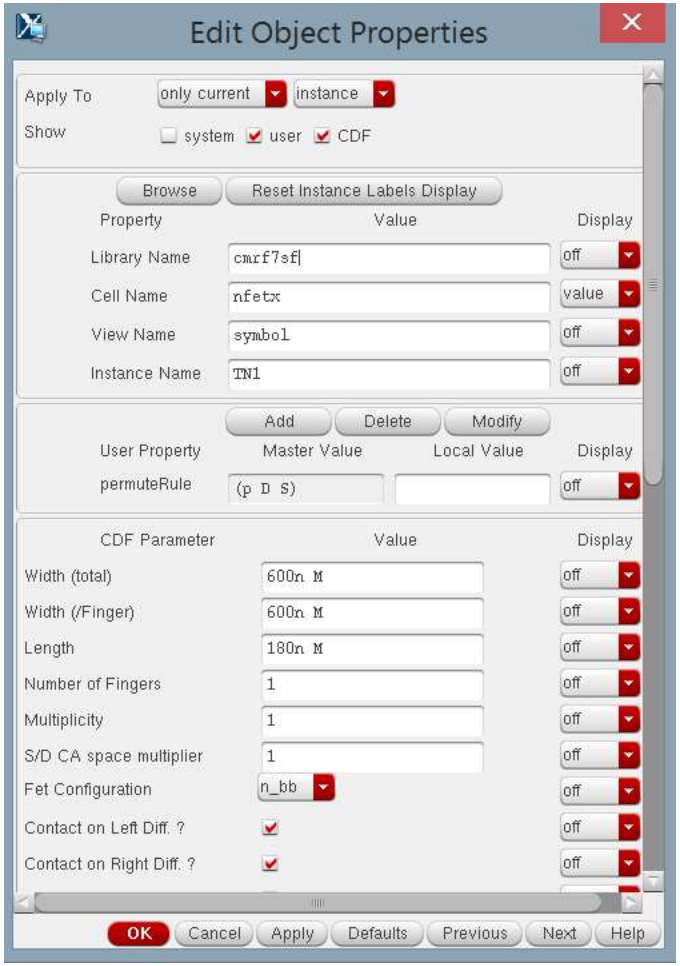
2.4 Create a schematic view of inverter

- Set properties of the instance
  - Activate your schematic window.
  - Select the object by clicking on it
  - Click Edit -> properties-> Objects…(or type q)



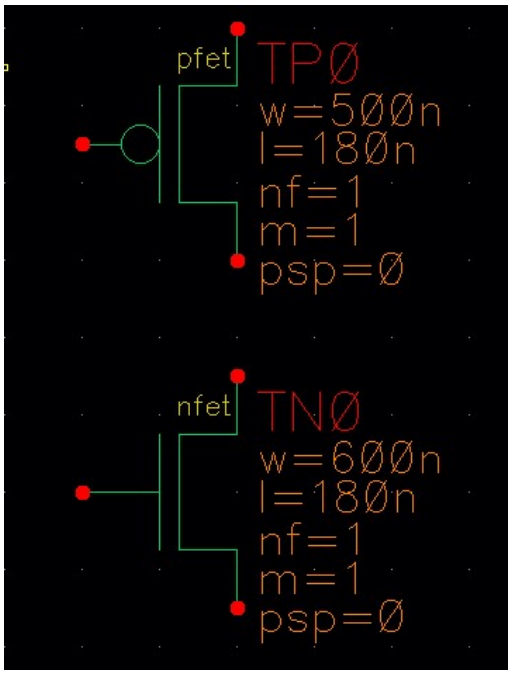
2.4 Create a schematic view of inverter

- The "**Edit Object Properties**" window will pop up.
- You can change the transistor properties including the length or width of the transistor.
- Leave width and length fields as the default



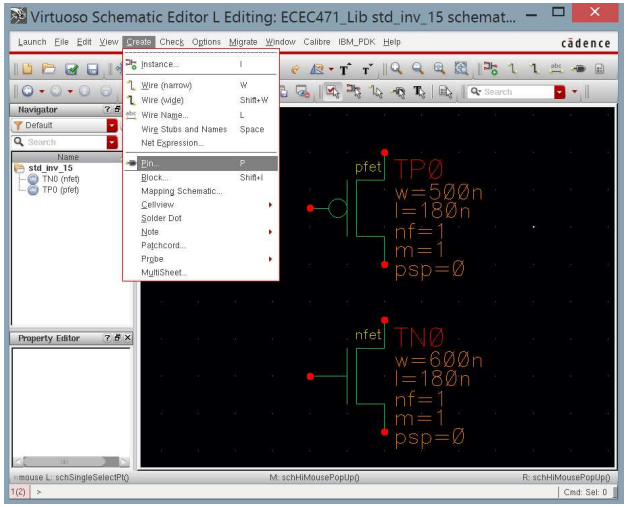
2.4 Create a schematic view of inverter

- Repeat steps for adding nfetx, but select pfet instead to add the PMOS to the inverter.
- After adding the PMOS and NMOS, the schematic should look like this:



2.4 Create a schematic view of inverter

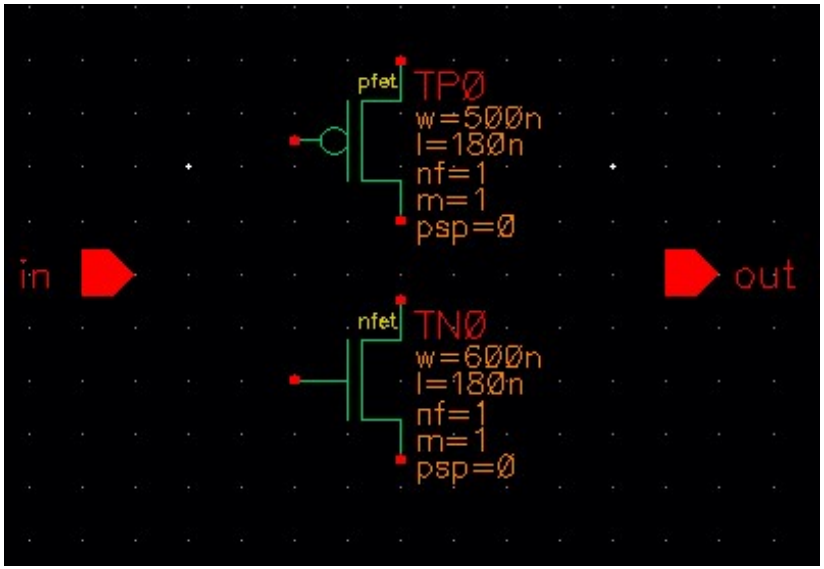
- Adding pins and labels:
  - To specify the input/output behavior of the circuit, input/output pins are added.
  - Click on "**Create->Pin…**"(or type p) in the schematic window.
  - The "Add Pin" dialogue box will appear
  - Choose **Direction**(e.g. input) and **enter the pin name**(e.g. in).
  - Click OK.
  - Place the pin in the schematic window the same way as place an instance.



2.4 Create a schematic view of inverter

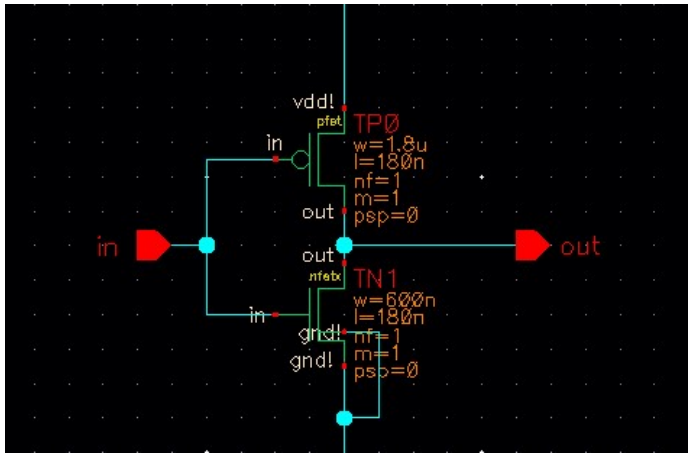
- Pin name: in Pin type: input
- Pin name: out Pin type: output

- The ! allows for symbol creation without the need for vdd/gnd pins
- After adding pins, the circuit looks like this:



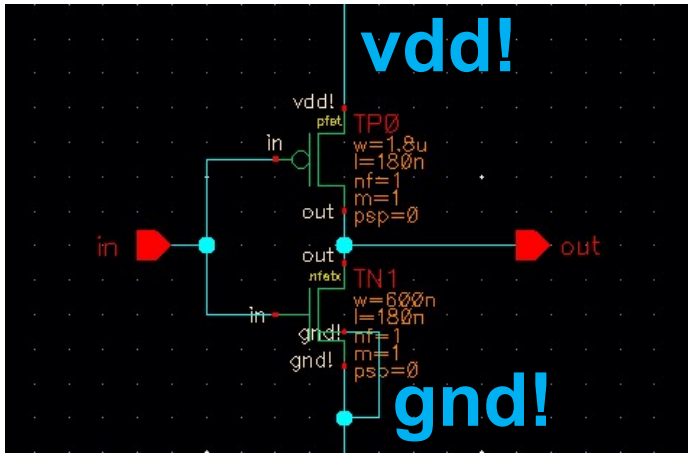
2.4 Create a schematic view of inverter

- Add wire:
  - Click on “Create ->Wire(narrow)” (or type w) in the schematic window.
  - The “Add Wire” dialogue box will appear. Hit “Hide” to go back to the schematic window.
  - A diamond shape will appear on the mouse arrow when the mouse arrow in on the port of an instance(red square) or a pin. Click once as the starting point of the wire, release the mouse, and click once at another port as the ending point of this piece of wire.
- Make sure the body of the nfetx is connected to gnd!



2.4 Create a schematic view of inverter

- Add wire name:
  - Click on “Create ->Wire Name” (or type l) in the schematic window.
  - The “Add Wire Name” dialogue box will appear. Hit “Hide” to go back to the schematic window.
  - Type vdd! and click on the wire location labeled vdd! in the picture to the right
  - Repeat for gnd!



2.4 Create a schematic view of inverter

- Check and save your design
  - Click on “File-> Check and Save”.
  - Check the CIW to see whether there is any errors in your design.

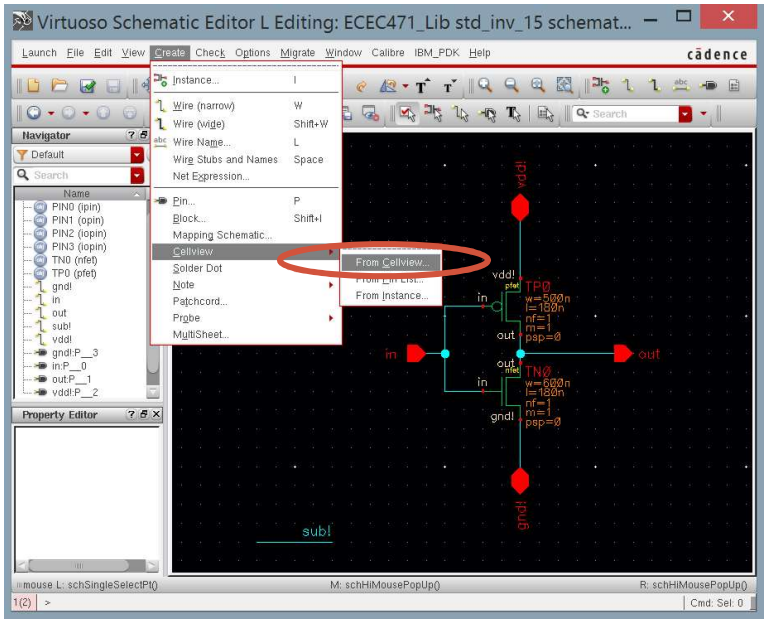
- Bindkeys:
  - i: Add instance
  - q: Edit properties
  - w: Add wire
  - p: Add a pin
  - l: label a wire
  - z: zoom in
  - Z: zoom out by 2X
  - f: fit the schematic in your schematic window
  - right mouse button: repeat last command

2.5 Create a symbol view of inverter

- Symbols are created for hierarchical design. At a higher level of abstraction, we would like to use a symbol to replace the details of a cell. Thus, a symbol of a cell should define all the inputs and outputs of that cell. Next, we introduce how to create a symbol from an existing schematic.

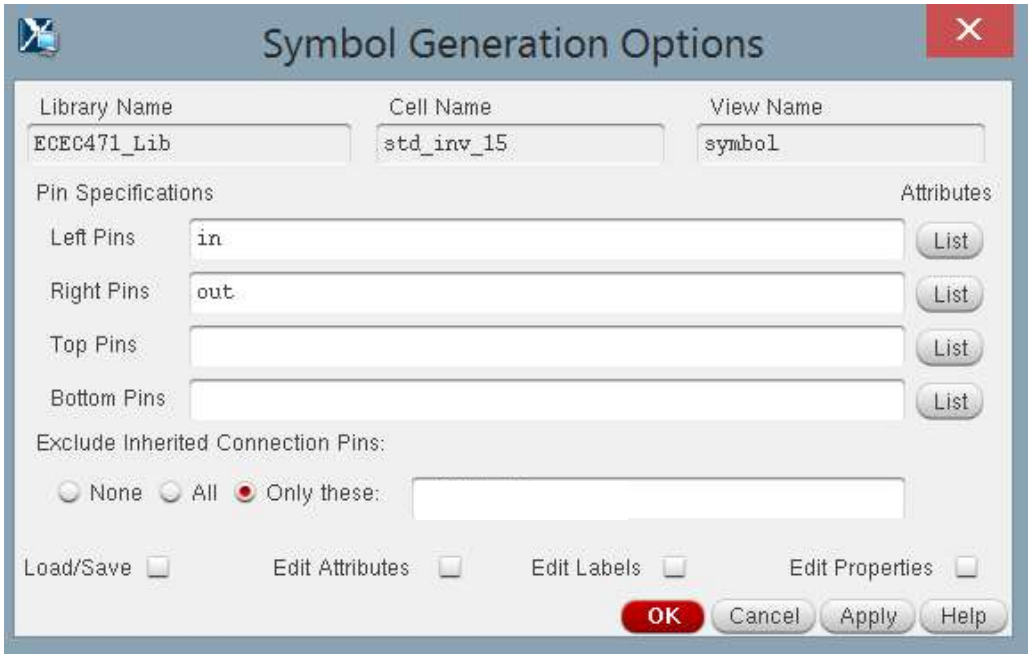
2.5 Create a symbol view of inverter

- Click on "**Create-> Create Cellview-> From Cellview**".
- The "**Cellview From Cellview**" window will pop up. We can use the default setup in this window to create the symbol.
- Hit "OK"



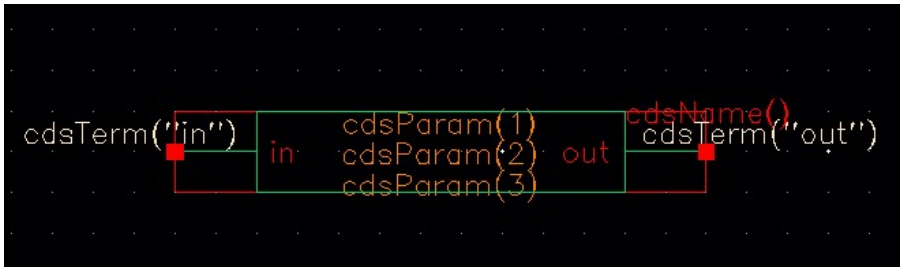
2.5 Create a symbol view of inverter

- The Symbol Generation Options should pop-up
- Ensure in is a Left Pin and out is a Right Pin
- Hit “OK”



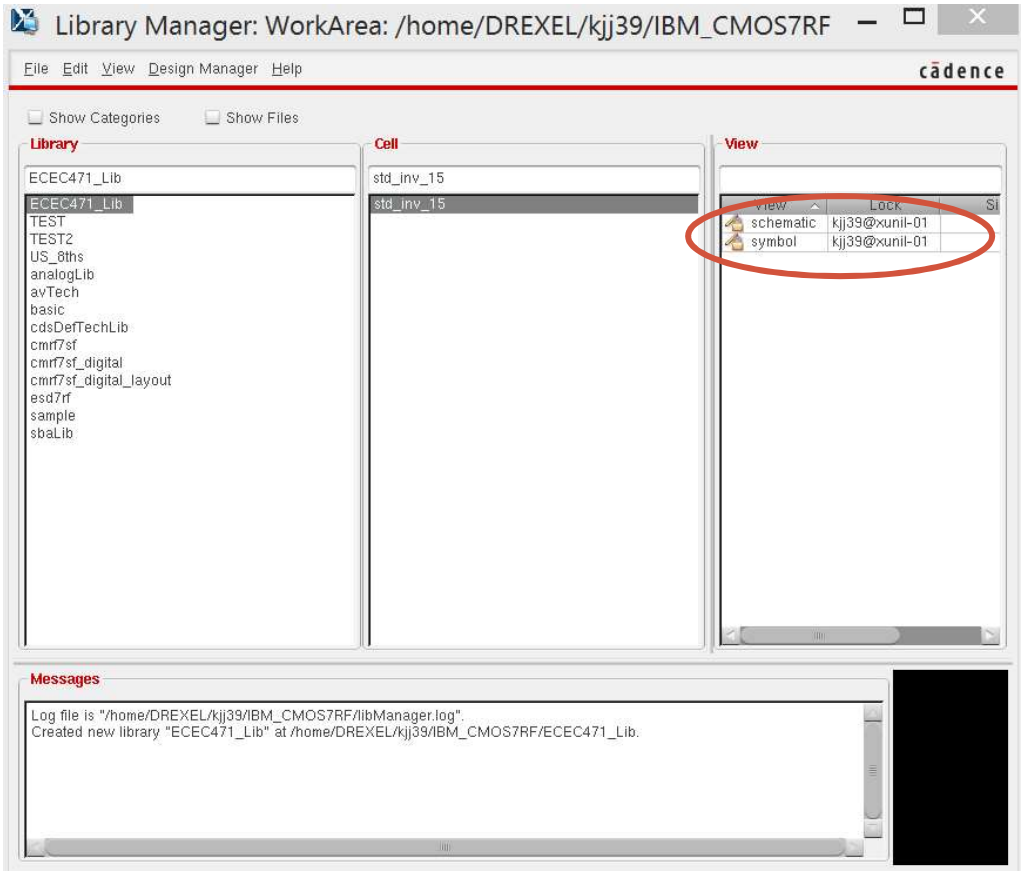
2.5 Create a symbol view of inverter

- Another window will pop up contains a default symbol picture. It has a red box that encloses the green colored inverter symbol. This red box defines the actual size a symbol will occupy, if you were to use this inverter in another design. You can change the size of this box. It is good custom to exactly fit the symbol within the red box. The red square dots indicate the pin connections. [@InstanceName] and [@PartName] are display variables, which you may delete or keep. The following picture shows the symbol. If you need to do some modifications to the symbol, remember to click “check and save”. The red circle is the quick button for “check and save”.



2.5 Create a symbol view of inverter

- Now if you go back to “Library Manager” window, the cell “std_inv_15” now has two views: schematic and symbol.



2.6 Simulation with spectre

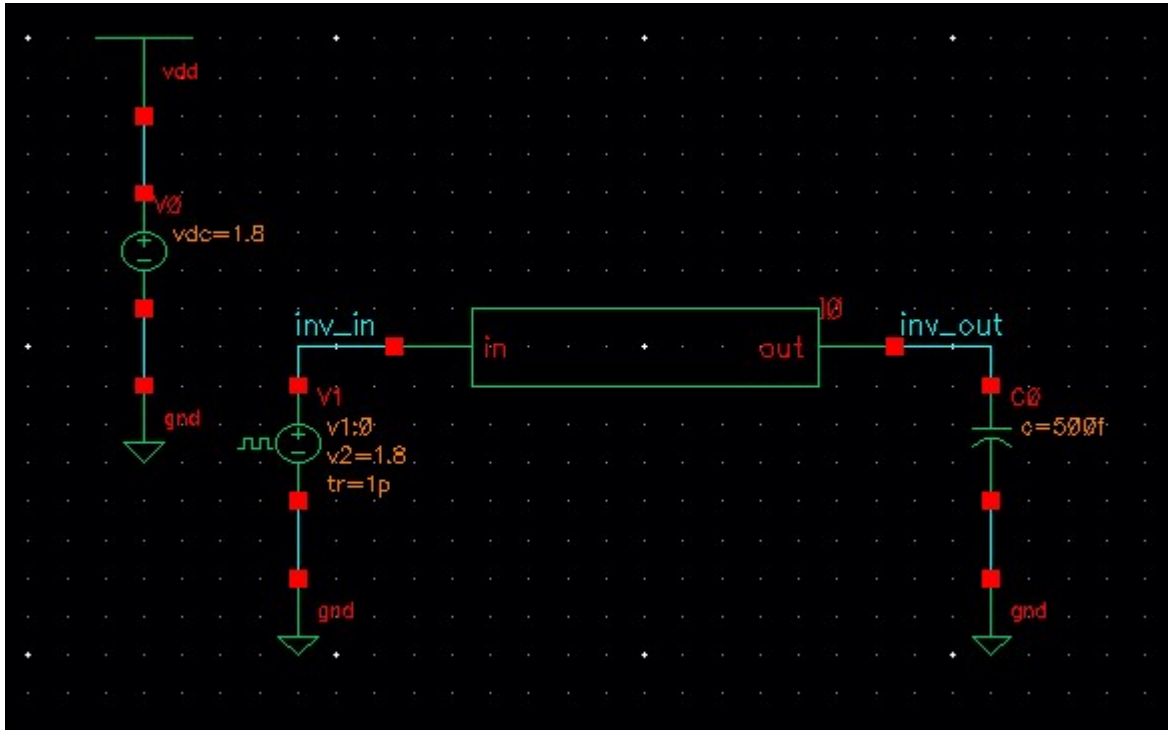
- Spectre is a Cadence version of the SPICE circuit simulator. The syntax of Spectre is compatible with SPICE simulation. By Comparison to Verilog-XL, Spectre lets you simulate transient behavior of your circuit at the transistor level. Next, we will perform transistor level simulations for the inverter schematic and observe its transient and DC behaviors.

2.6 Simulation with spectre

- Create a new cell "**sim_inv**" in the library you created (In library manager) with "schematic" view, using “File->New->CellView” on the library manager window.
- Add the following components into your "**sim_inv**“ schematic:
  - The symbol of “**std_inv_15**” from your own library.
  - The symbols for **VDD and GND** from “analogLib”.
  - The symbol for **vdc, vpulse** from “analogLib”.
  - The symbol for **cap** from “analogLib”.
- Connect the circuit.

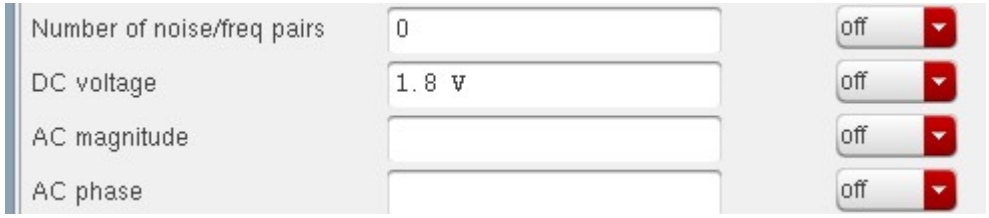
2.6 Simulation with spectre

- The circuit should look like this

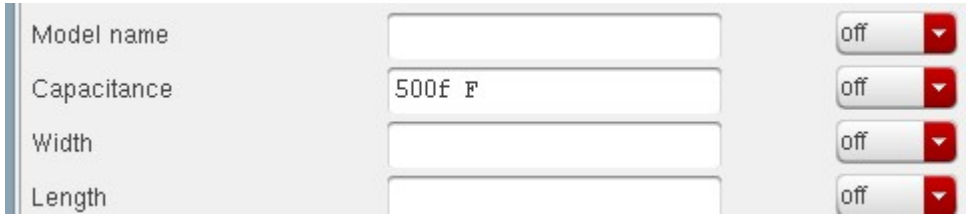


2.6 Simulation with spectre

- Edit the properties of the vdc to the following

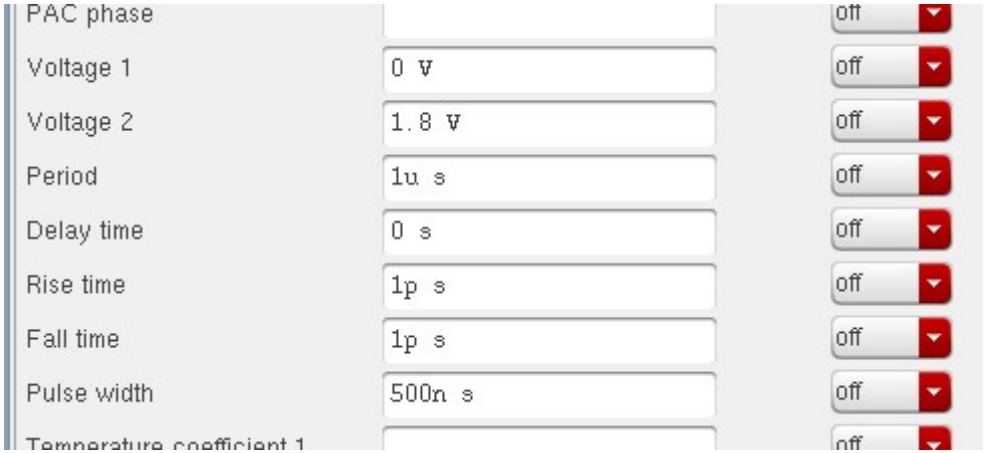


- Edit the capacitor attributes as follows



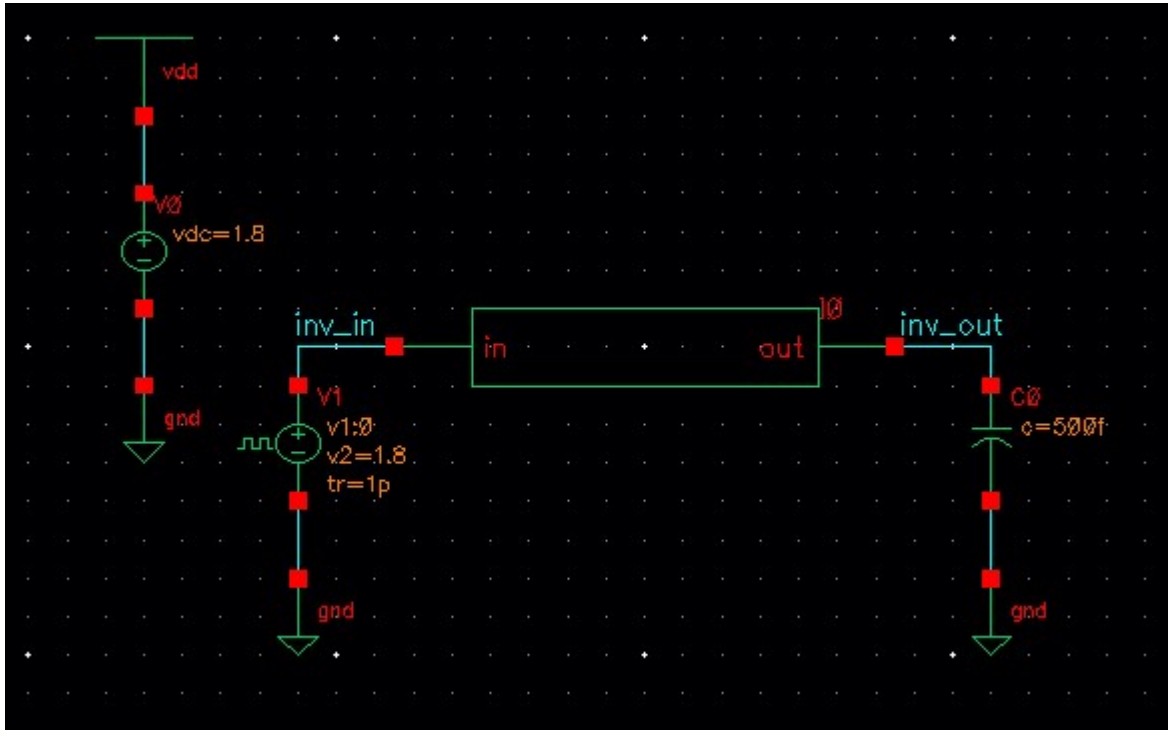
2.6 Simulation with spectre

- Edit the properties of the vpulse to the following



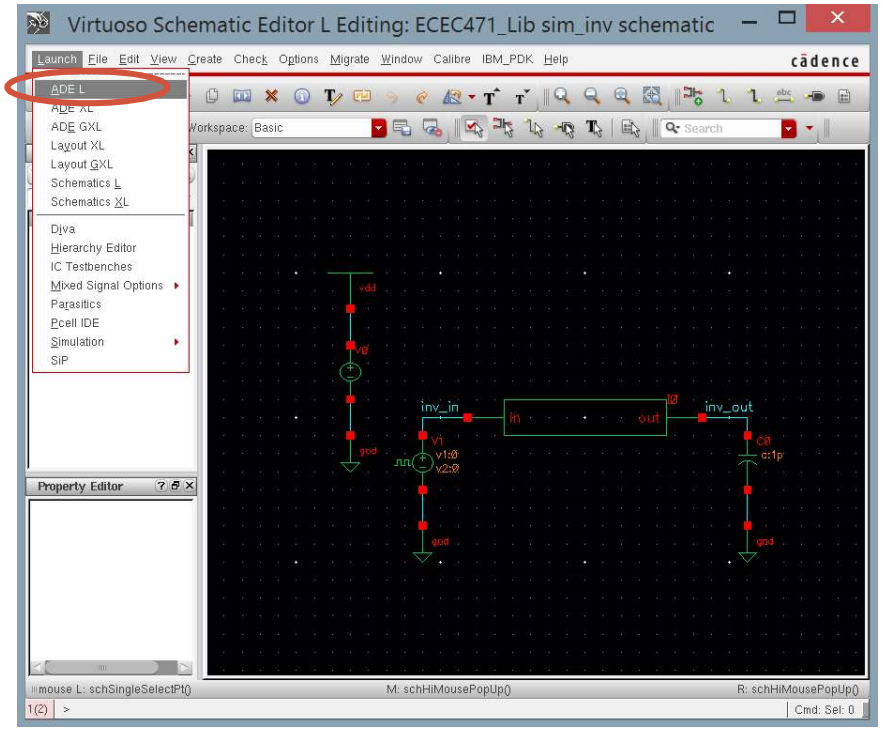
2.6 Simulation with spectre

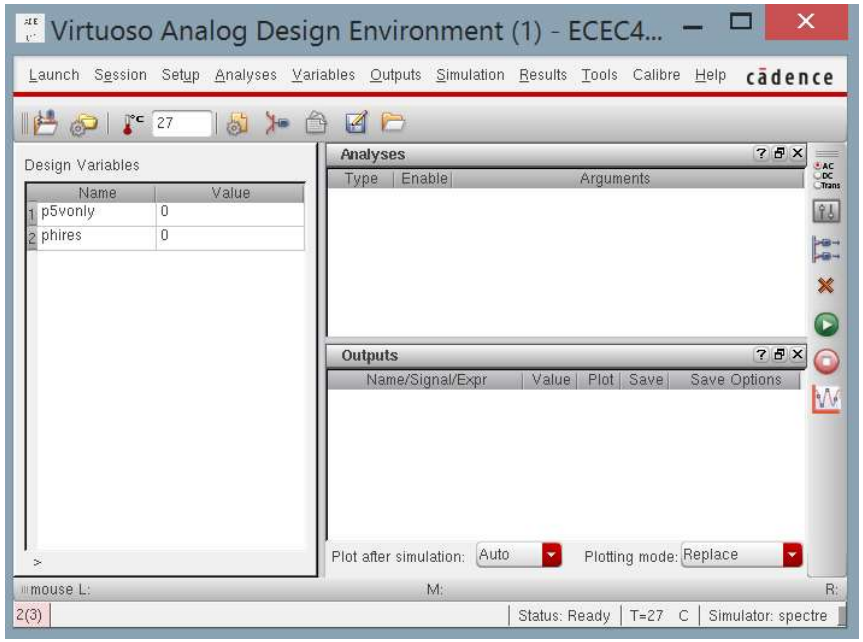
- Add a netname of inv_in to the wire connected to the input
- Add a netname of inv_out to the wire connected to the output



2.6 Simulation with spectre

- Simulation in Analog Environment
  - Activate the schematic window of sim_inv.
  - click on "**Launch->ADE L**".



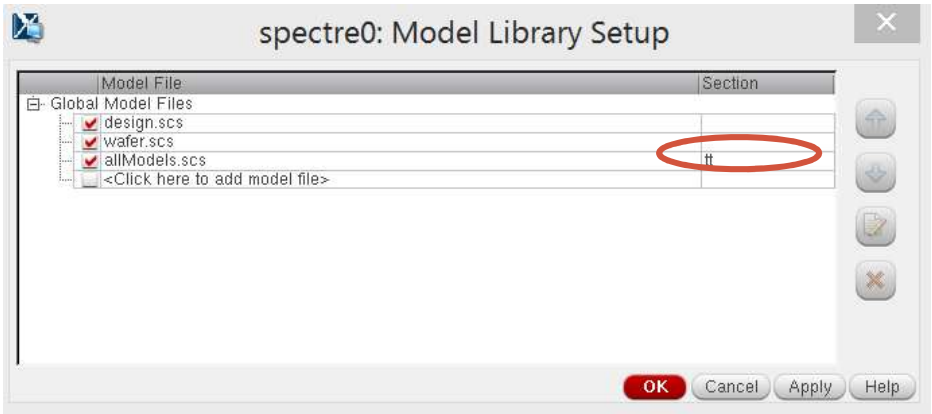


2.6 Simulation with spectre

- Verify simulation settings of the “Analog Design Environment”.
  - Click on “**Setup->Simulator/Directory/Host…**”, make sure the simulator is “**Spectre**”.
  - Click on "**Setup->Model Path**", make sure the right model files are included for simulation and that tt is chosen for allModels.scs
  - In case you do not see these options the .scs files listed below can be found in

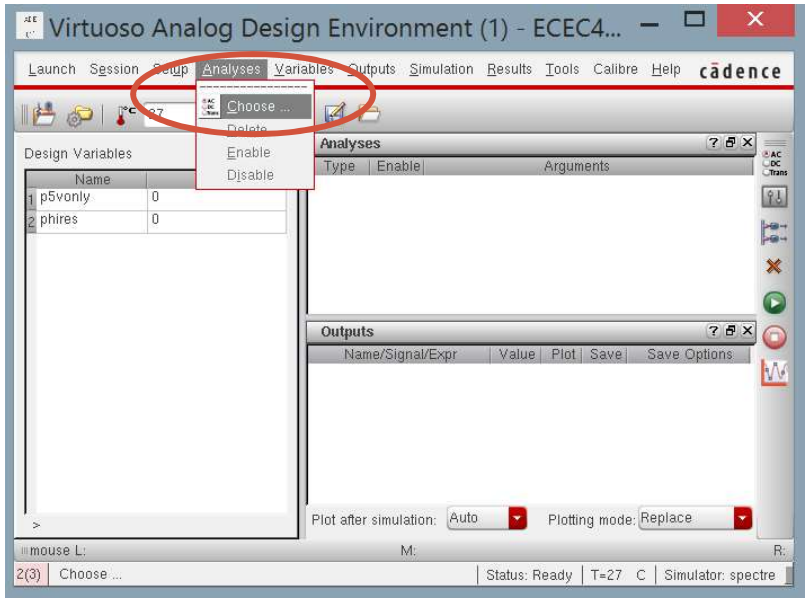
/home/class_data/ecec571-f2015/IBM_CMOS7RF_p18u/IBM_PDK/cmrf7sf/V2.0.1.0AM/Spectre/models

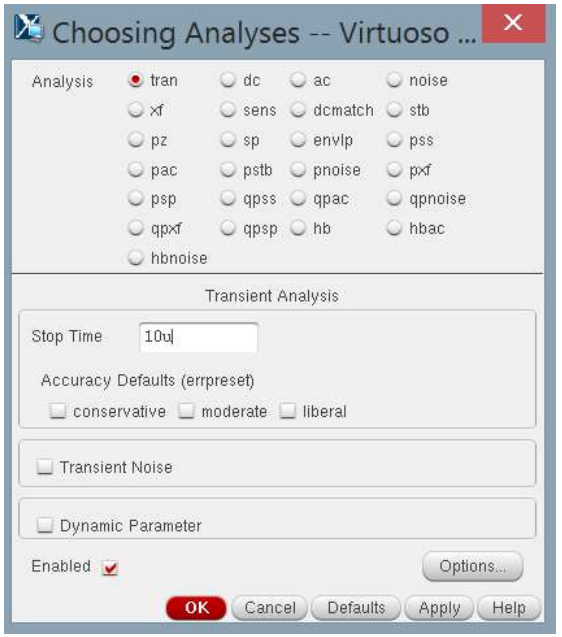
Note: Make sure that tt is selected for allModels.scs



2.7 Transient Simulation with spectre

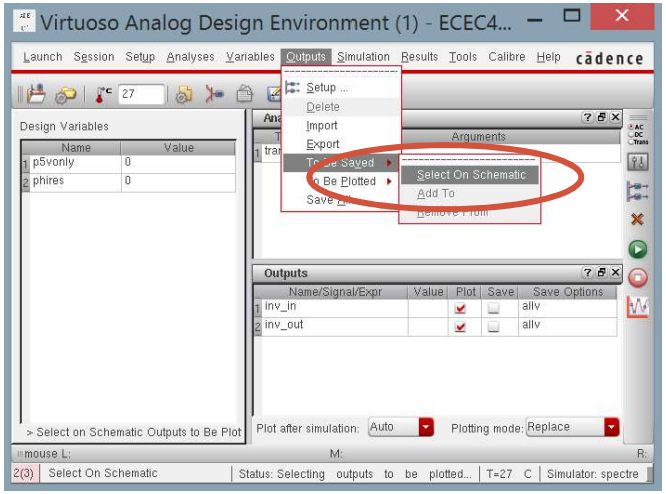
- Choose the simulation type.
- Click on “**Analyses->Choose…**”.
- "**tran**" is chosen as the default analysis. Write the “Stop Time” of approximately 10cycles (Based on what period you used for the vpulse earlier)
- Choose conservative as the Accuracy Defaults setting
- Click OK.

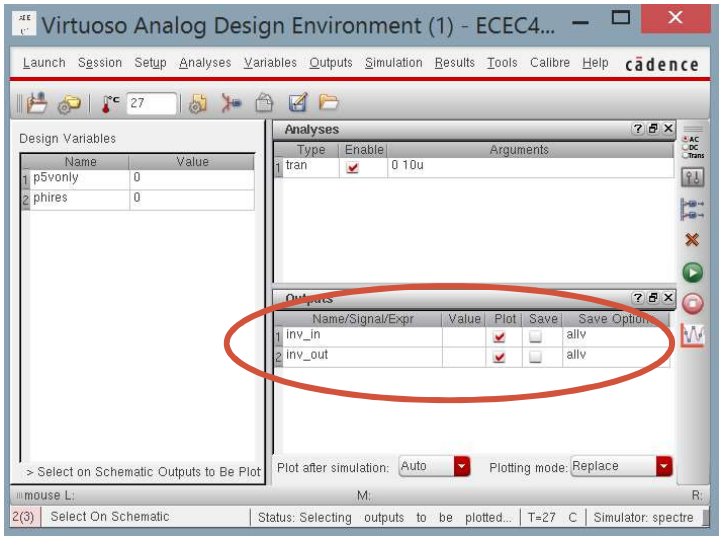




2.7 Transient Simulation with spectre

- Now, it is time to determine the nodes that will be observed.
- Click on "**Outputs->To Be Plotted->Select on Schematics**".
- This will let you choose the nodes to be observed. Focus the schematics window and click on the wires at the input and output of the inverter.



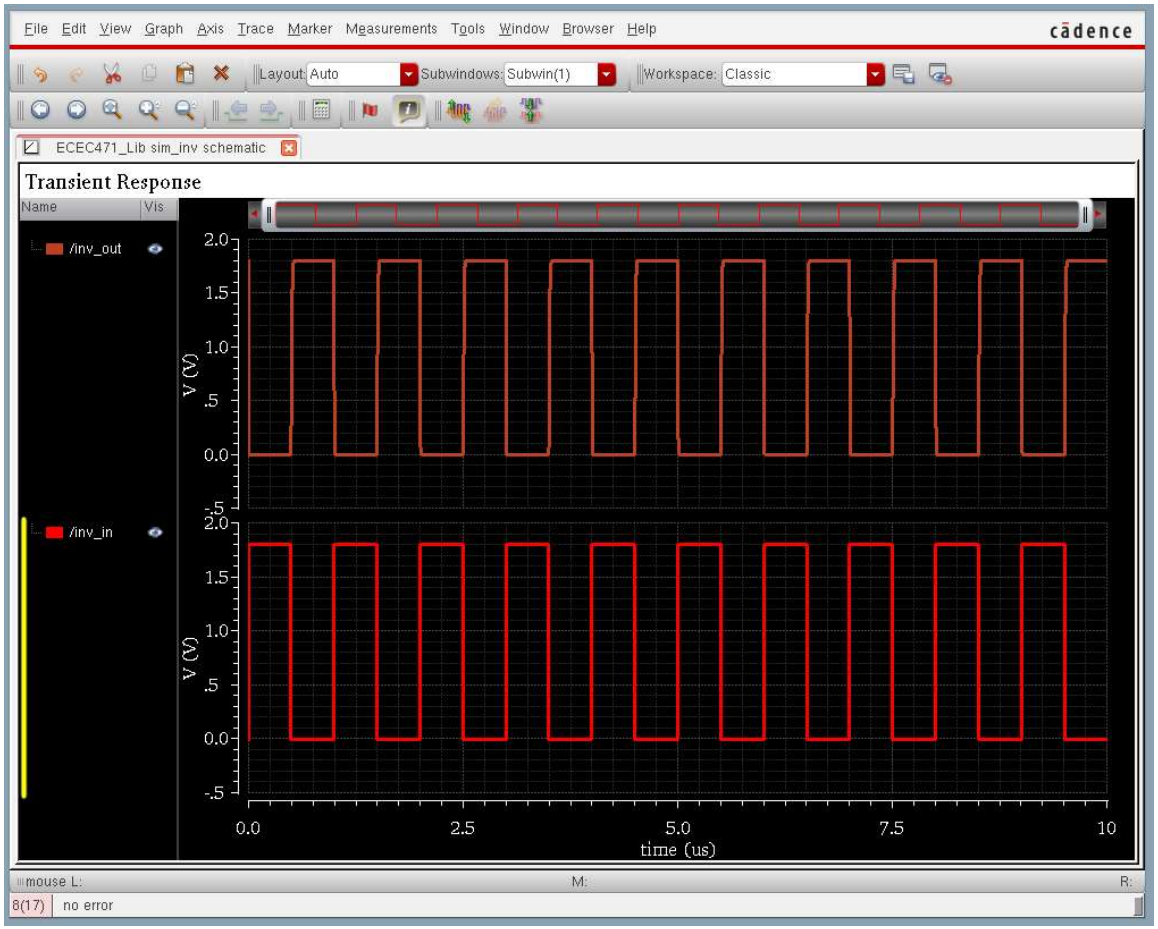


2.7 Transient Simulation with spectre

- There are 2 types of signals to be observed after the simulation: voltage and current.
- If you click on **wire**, you select that wire's **voltage** as the observed signal. If you click on the **node** like the input of the inverter, you select the **current** of that node as the observed signal. You see a circle on that node. **You will work on voltage here.**
- Once this is done, go to the Analog Design Environment Window and click "**Simulation->Netlist->Create**".
- The netlist of the schematic is extracted. Click "**Simulation->Netlist and Run**". Wait for a few seconds and after the simulation is finished, a waveform window will automatically appear.

2.7 Transient Simulation with spectre

- Simulation results

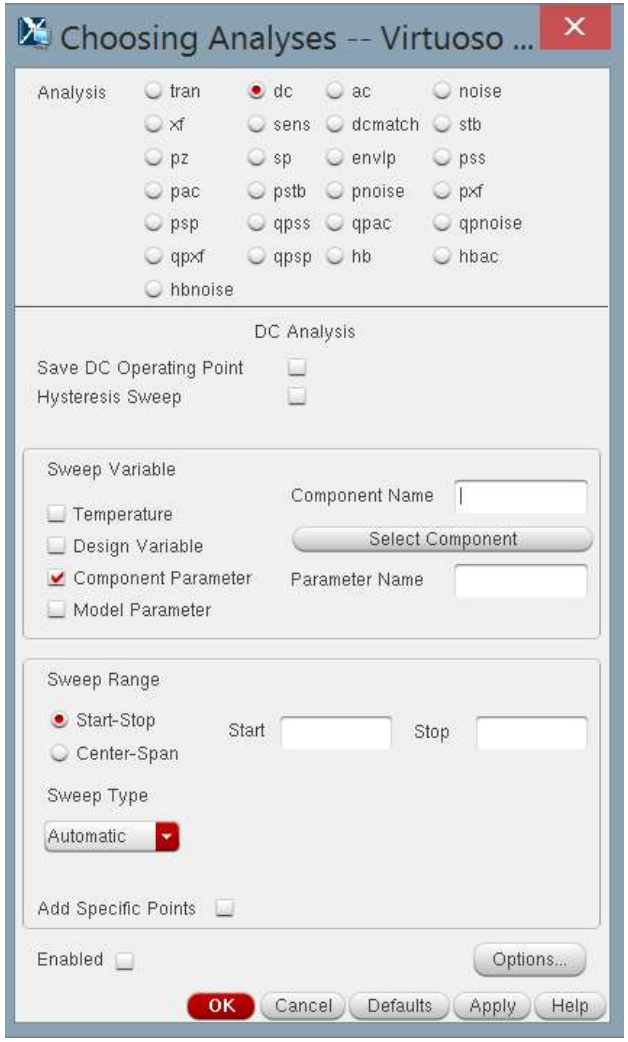


2.7 DC Simulation with spectre

- DC simulation is used to analyze the performance of the output of the inverter along with the changing its input.
- The purpose of doing DC simulation is find the point where **inv_out=inv_in=vdd/2** through adjusting the size of PMOS and NMOS. When reaching this point, the inverter is called a **symmetrical inverter**.

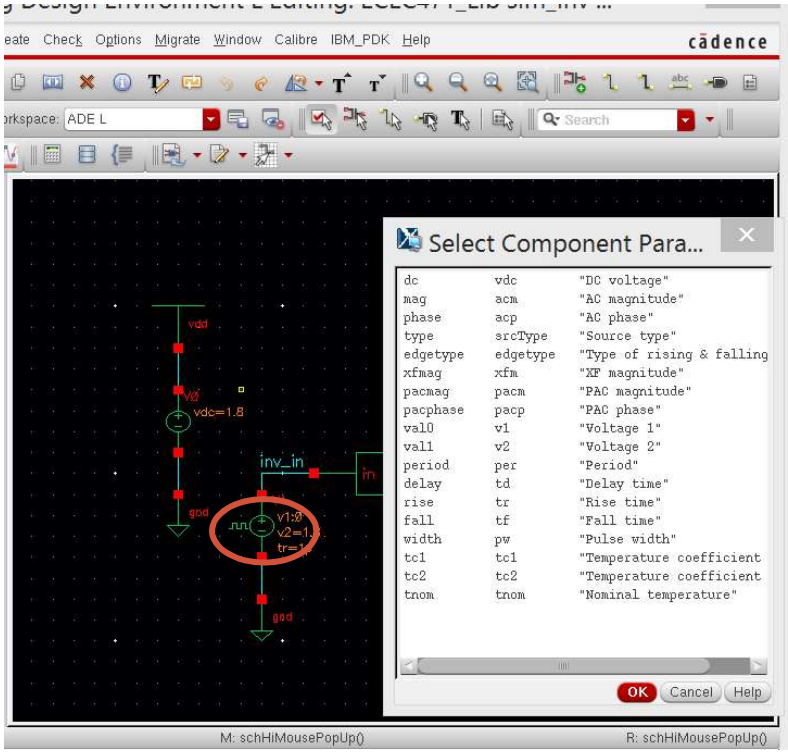
2.7 DC Simulation with spectre

- Go to the Analog Design Environment window, click on Analyses->Choose…, chosen “DC” as the simulation type.
- “Sweep Variable”, choose “**Component Parameter**”.



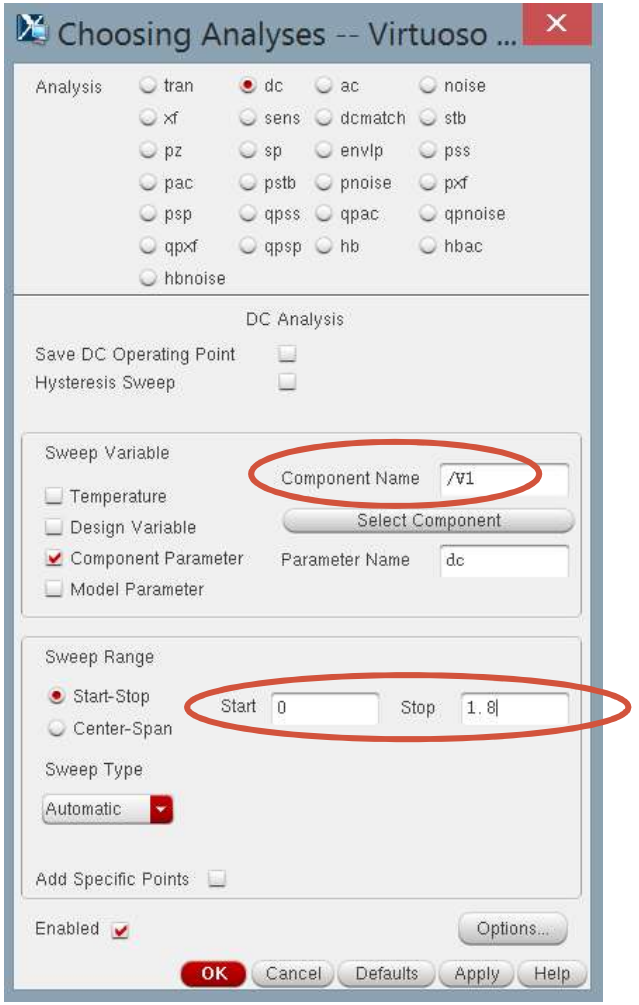
2.7 DC Simulation with spectre

- Click on "**Select Component**" will activate the schematic window.
- Click on the input voltage source, then the "**select component Parameter**" window will pop up.
- Choose "**DC**". Hit "OK".



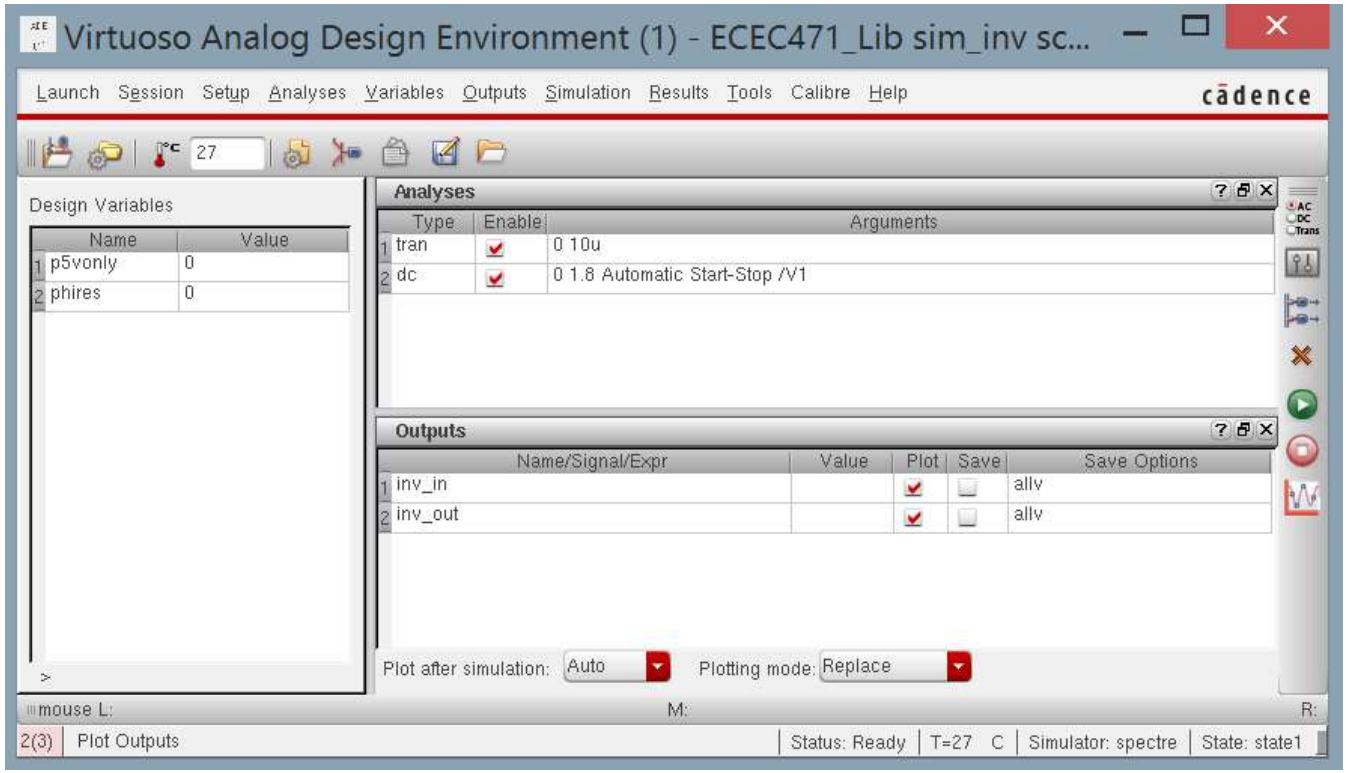
2.7 DC Simulation with spectre

- Now the “Component Name” is the input source of the inverter and the “Parameter Name” is “DC”.
- We also need to set the “Sweep Range”. Let the simulator the sweep the input voltage from “0v—1.8v”.



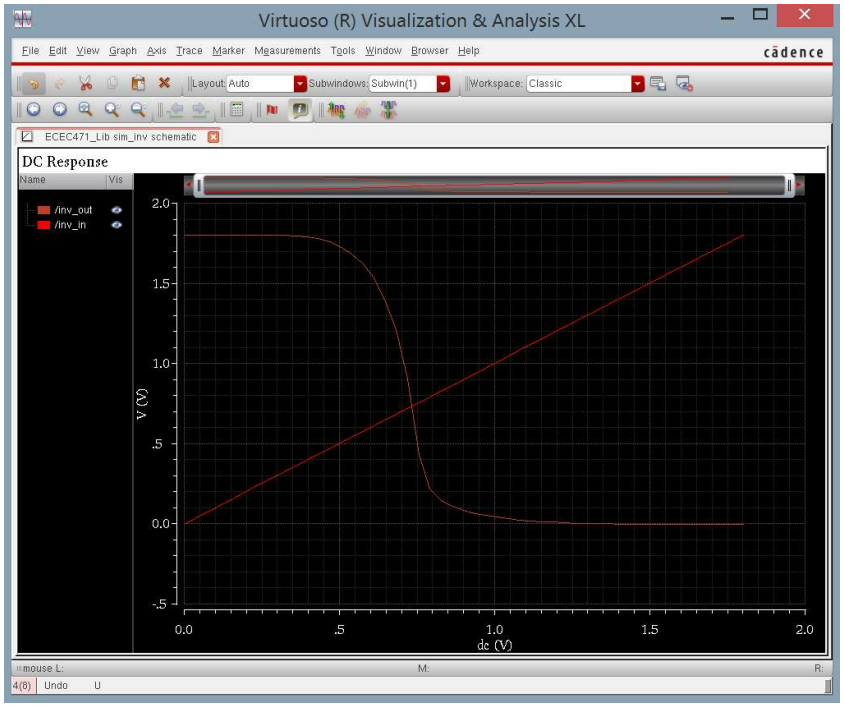
2.7 DC Simulation with spectre

- Now the Analog Design Environment window looks like this:



2.7 DC Simulation with spectre

- Once this is done, click "**Simulation->Netlist and Run**" to extract and run the netlist.
- Wait for a few seconds and after the simulation is finished, a waveform window will automatically appear.
- You can put a marker at the crossing point of input and output voltage of the inverter.

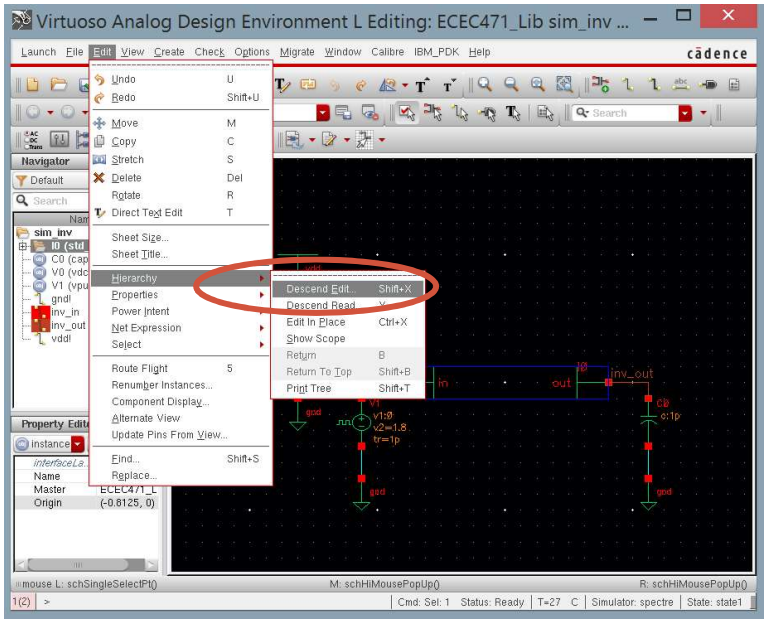


2.8 Find the symmetrical inverter

- In order to build a symmetrical inverter, we need to adjust the size of the MOSFET. Since only the ratio of PMOS width and NMOS width affect the performance of the inverter, here, we choose to only modify the width of the PMOS while keeping the width of the NMOS unchanged.
- How do we choose a proper number for the width of the PMOS?
- We will set the width of the PMOS as a variable, and sweep through a number of widths with simulations. Then we will pick the width that fits our requirements the best to be the desired width of the PMOS.

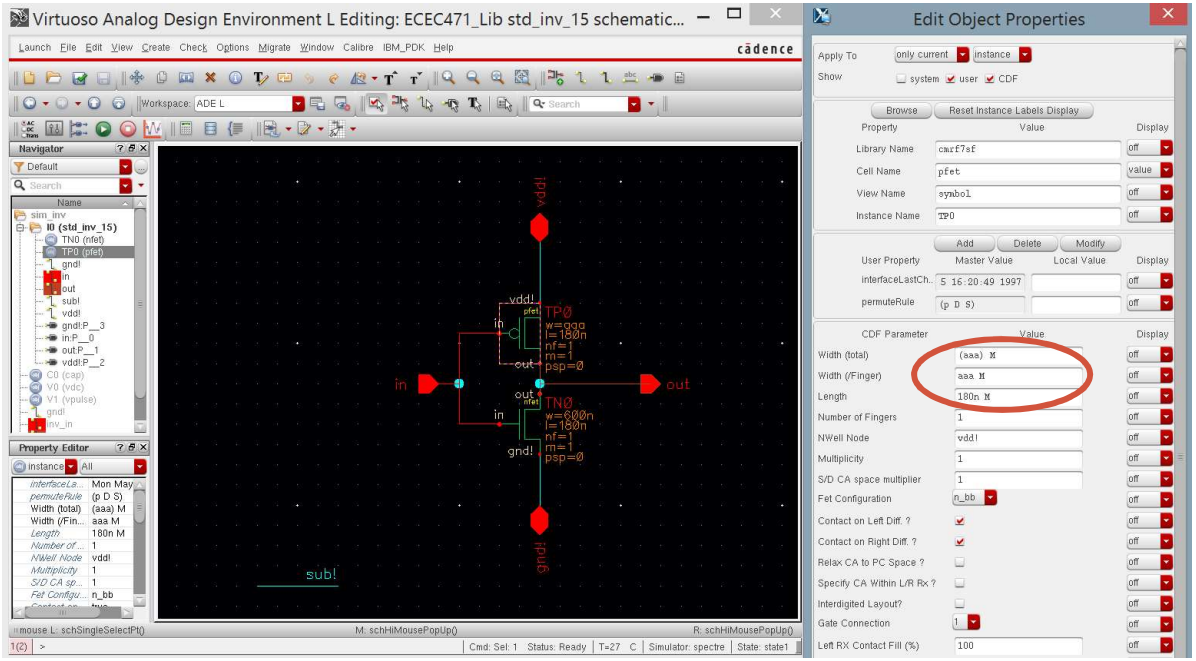
2.8 Find the symmetrical inverter

- Go back to the schematic window, select the inverter block, click on "**Edit->Hierarchy->Descend Edit…**".
- Then the "Descend" window pops up. Hit "OK" will descend you to the lower level schematic of this circuit, which is the schematic of the inverter we have just build.



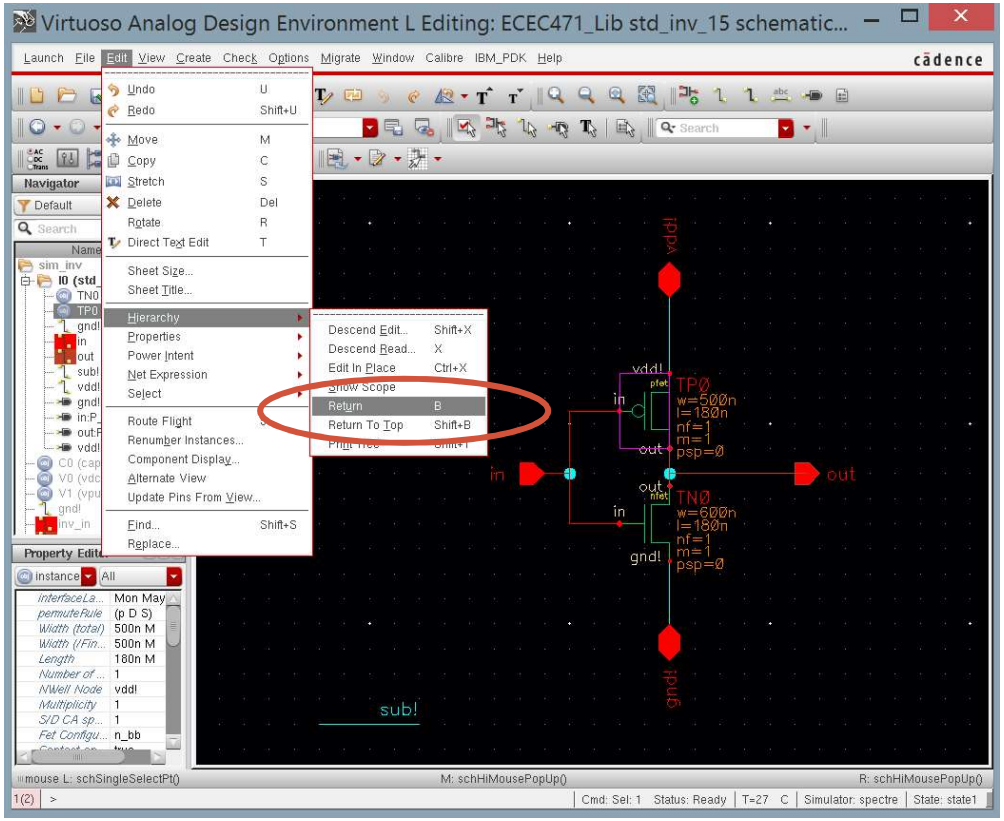
2.8 Find the symmetrical inverter

- Select the PMOS and hit “q” to edit its property.
- Hit “OK”.
- Hit “check and save” (every time you make modifications).



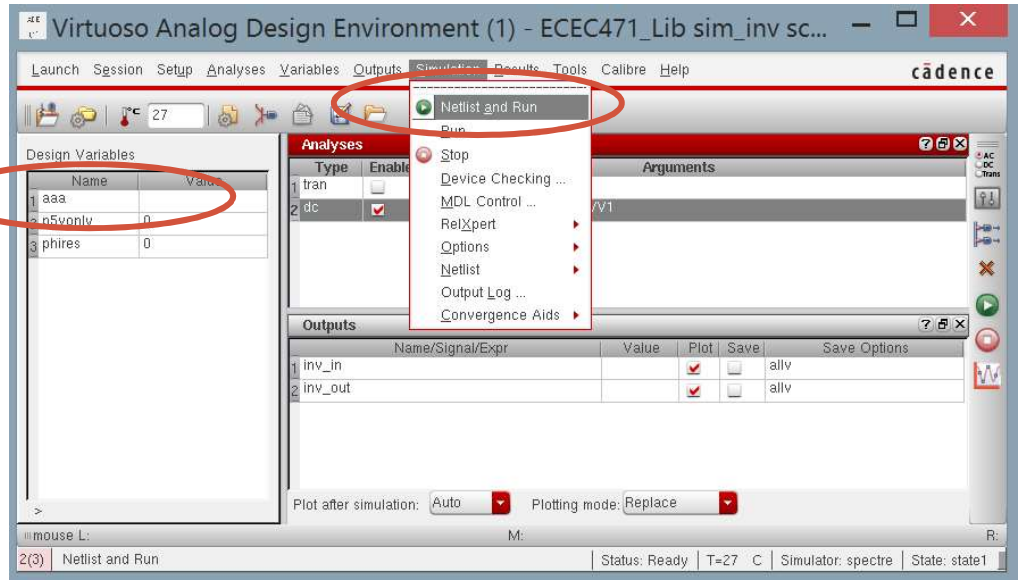
2.8 Find the symmetrical inverter

- Then hit “Edit->Hierarchy->Return” or “b” to return to the upper level. In the upper level, hit “check and save” again.



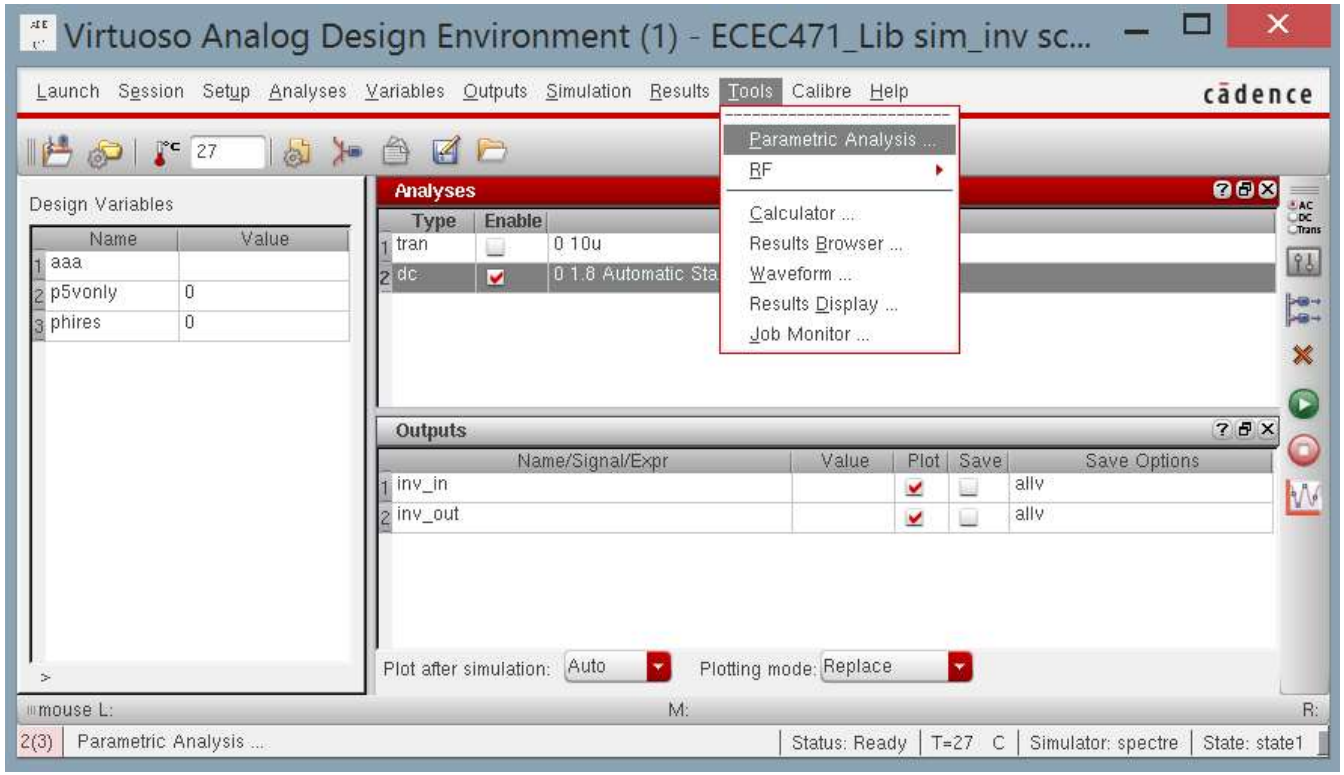
2.8 Find the symmetrical inverter

- Now we begin to do the simulation,
- Go to the “Analog Design Envionment” window, go to “Simulation->Netlist and Run”.
- After extracting the netlist, “aaa” is listed in the “design variables” table.



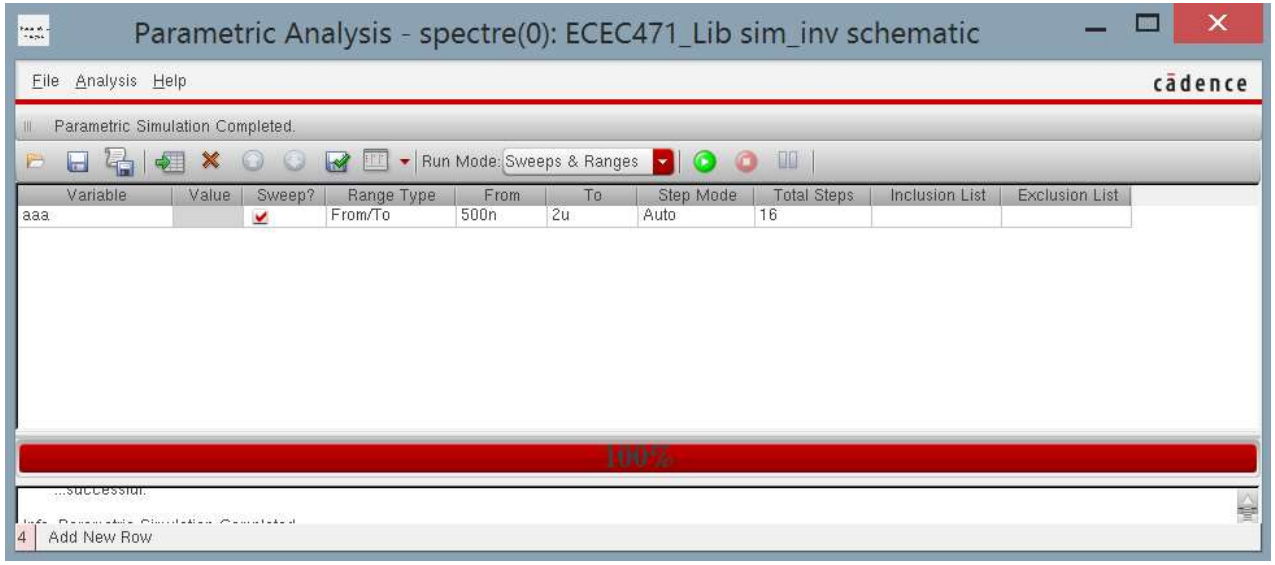
2.8 Find the symmetrical inverter

- Then hit “Tools->Parametric Analysis”.



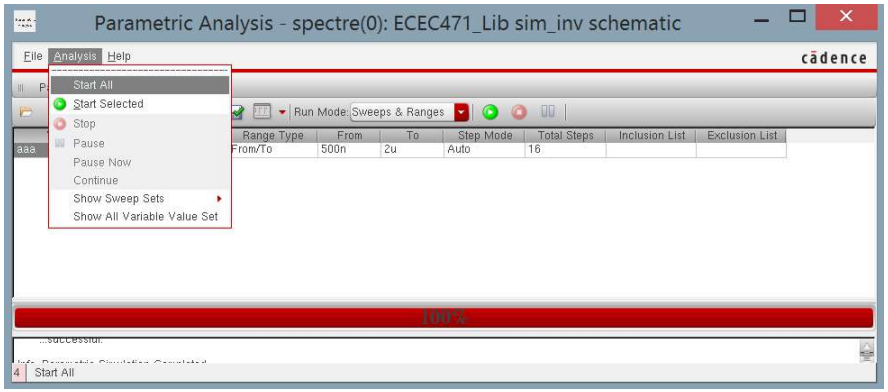
2.8 Find the symmetrical inverter

- In the “Parametric Analysis” window, fill in the “Variable Name”: the variable name of the width of PMOS (aaa as an example).
- Then fill in the sweep range. “from” should be equal or greater than 90n, which is required by the process. “to” is the final number that will be simulated with.



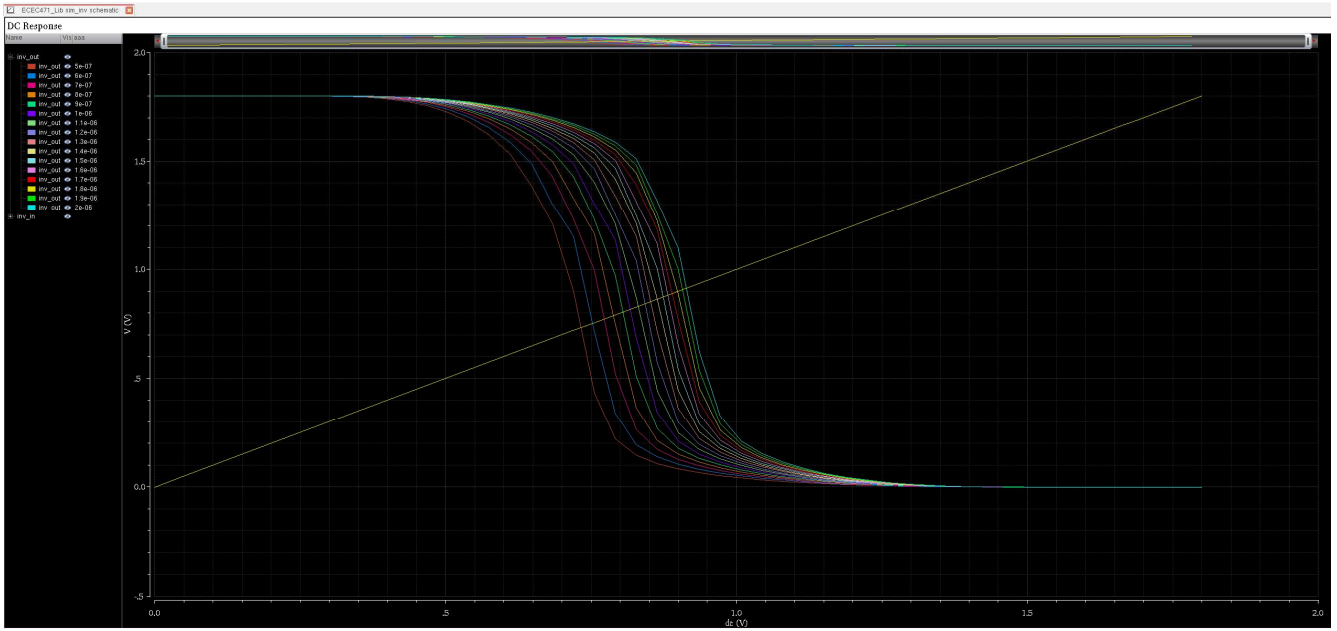
2.8 Find the symmetrical inverter

- Be careful about the “Total Steps”!!
- The width value used for simulation is calculated by:
  - **Width=From+(To-From)/Total Steps**
- Since the smallest incremental step of width is 10nm, you need to make sure:
  - **((To-From)/(10nm * n)) + 1 = number of steps (where n is an integer)**
- Keep in mind more steps = longer runtime and more congestion for analysis
- After you setup everything, hit “Analysis->Start”.



2.8 Find the symmetrical inverter

- The simulation results look similar to this:



- Then you can choose the best width of PMOS from these simulation results. If you think neither of them meets your requirements, then you can shrink the span between “From” and “To”, redo the parametric analysis.