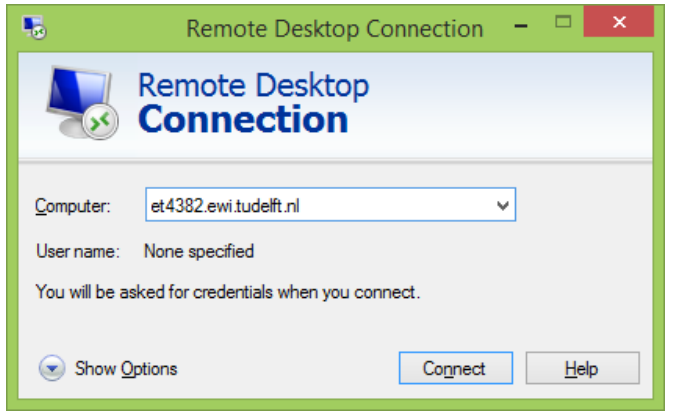
ET4382 Cadence Tutorial

In this course, you will use Cadence tools for assignments and the final project. This tutorial teaches you how to 1) connect to the Cadence server, 2) setup your Cadence environment, 3) perform schematic capture, and 4) simulate your circuit, by going through this flow for a CMOS inverter.

1. **Connect to Cadence Server**

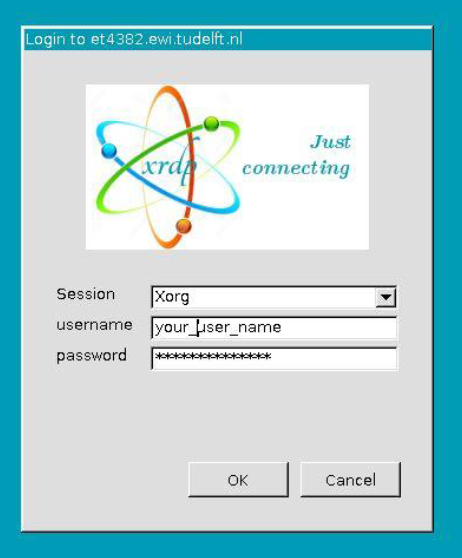
You need to be on TU Delft campus network for this to work. Once connected to the TU Delft campus network, launch Remote Desktop connection (bureaubladverbinding op afstand).



Log in to ET4382.ewi.tudelft.nl with your *netid* and *netid* password.

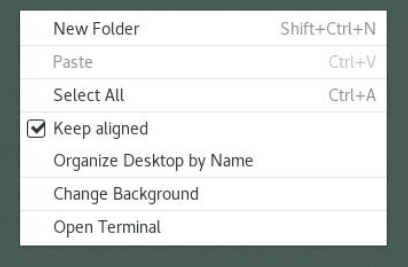
Username: *netid*

Password: your *netid* password.



1. **Setup Cadence Environment**

Once logged-in, right mouse click on desktop and launch terminal (click Open Terminal).



First, you set up your working directory where you will use Cadence for the entire semester. You only need to do this once. (Hit ENTER after each line.)

**cd ~**

**mkdir tsmcBCD**

**cd ~/tsmcBCD**

**/opt/ei/DK/tsmc/oa180/mini018BCDG2/216A/et4382/start**

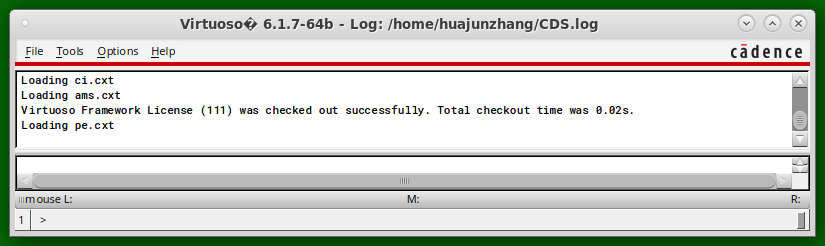
To start Cadence, type the following.

**cd ~/tsmcBCD**

**source sourceme**

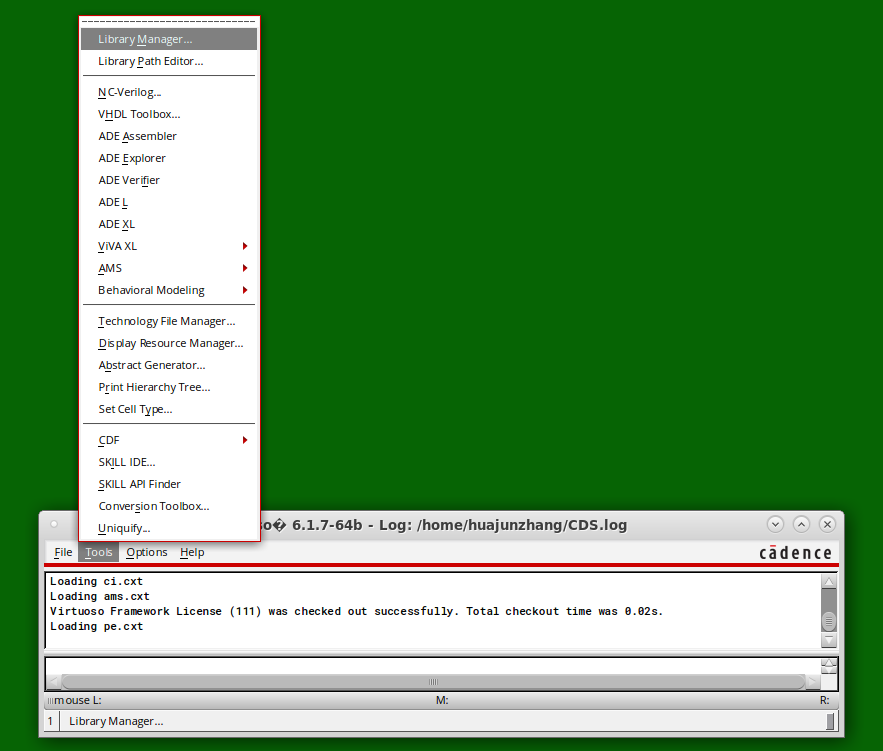
**virtuoso&**

You need to do this every time you start Cadence. After Cadence starts, you should see the CIW window:

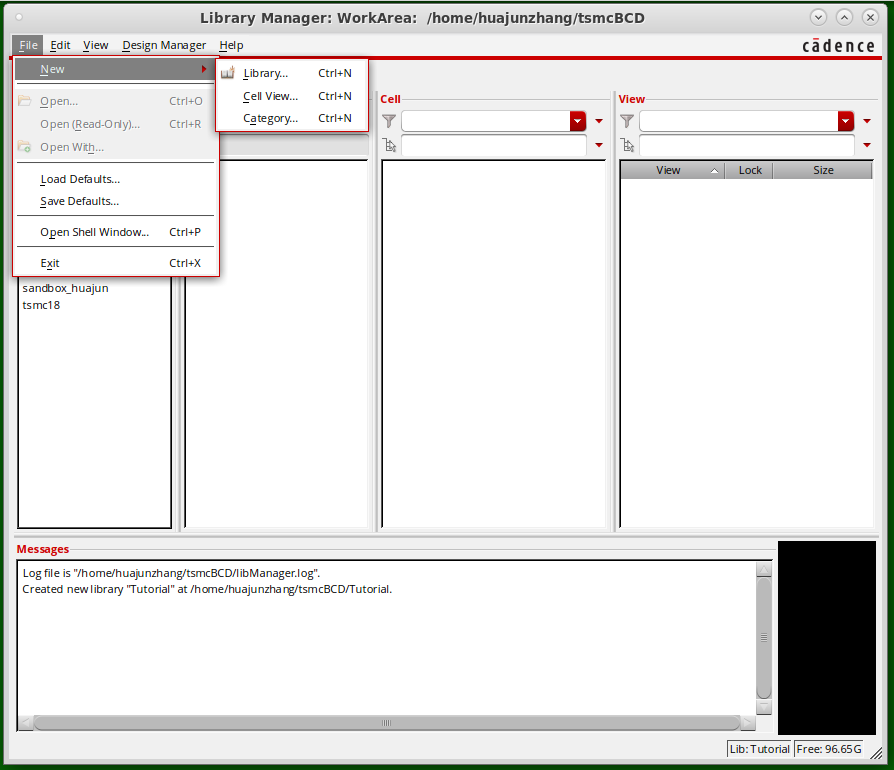


1. **Create a Library**

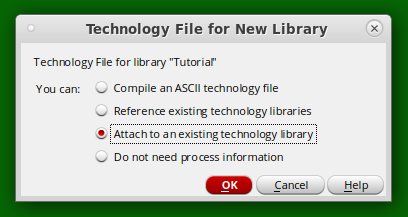
Your schematics must be placed under a library. Open library manager by clicking in CIW: Tools 🡪 Library Manager.



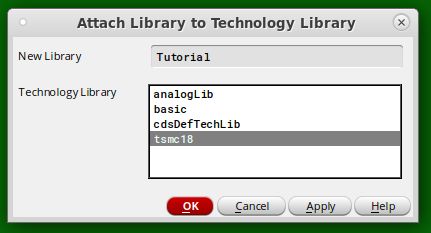
In Library Manager, click File 🡪 New 🡪 Library.



Fill in a name for the library and click OK. A small window called “Technology File for New Library” will pop up. Select “Attach to an existing technology library” and click OK.

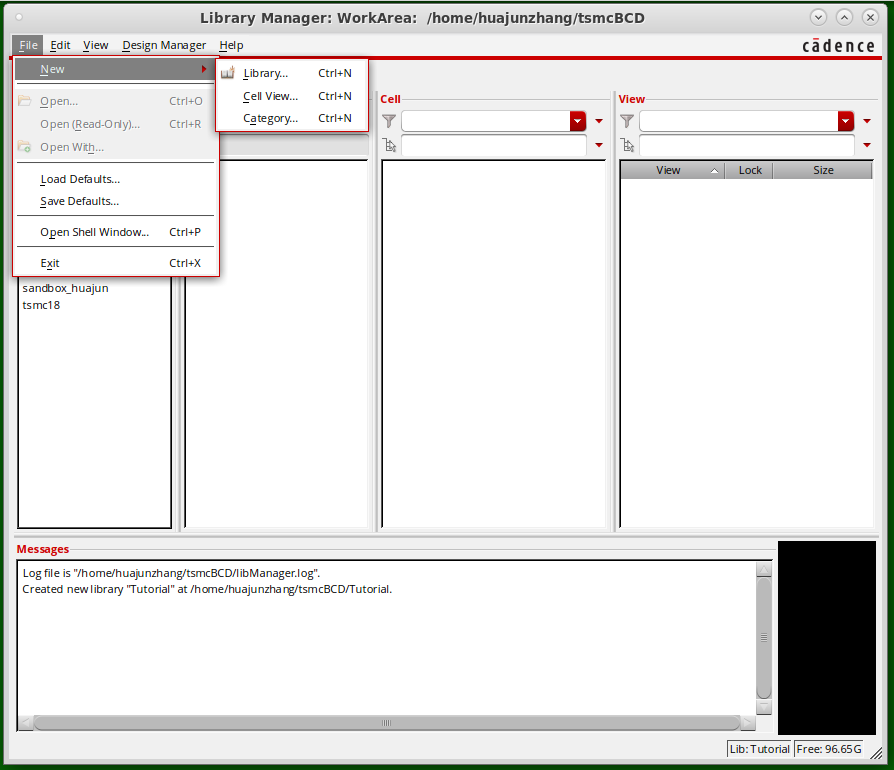


Then, select tsmc18 and click OK. The new library should appear in Library Manager.

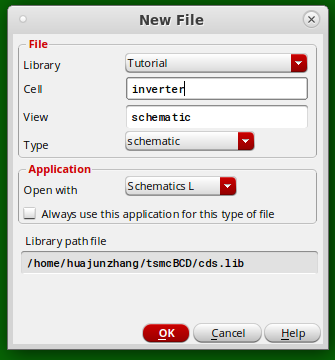


1. **Create a new schematic**

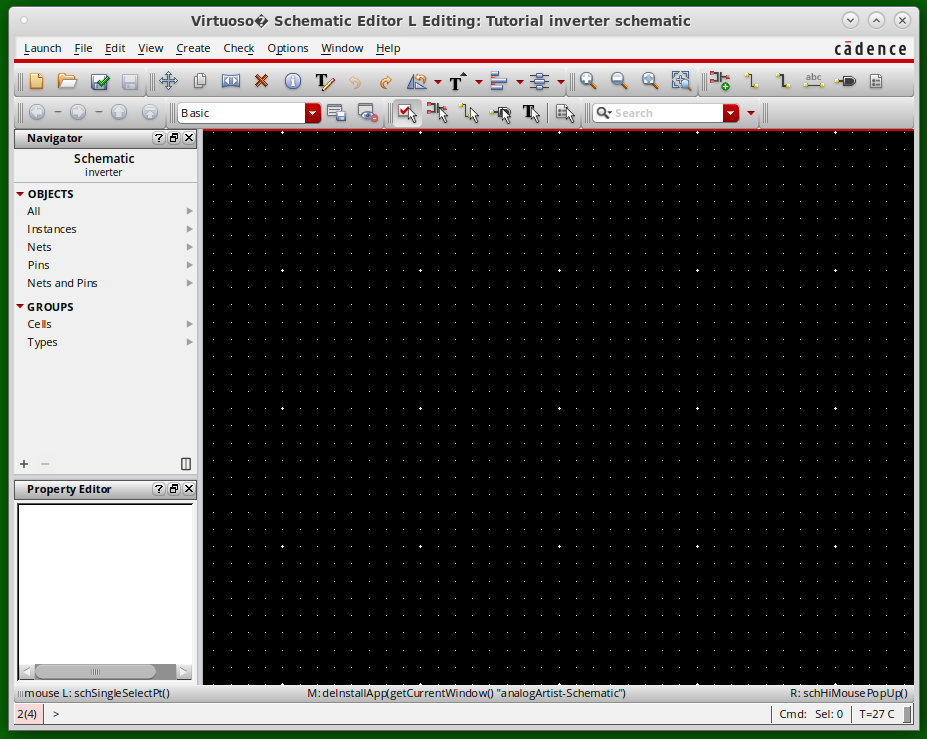
In this tutorial, we will create a schematic for an inverter. In Library Manager, click File 🡪 New 🡪 Cell View.



In the “New File” dialog, select your newly created library in “Library.” For “Cell,” fill in “inverter”. The “View” should be schematic. If not, select “schematic” under “Type”. The finished dialog is shown below.

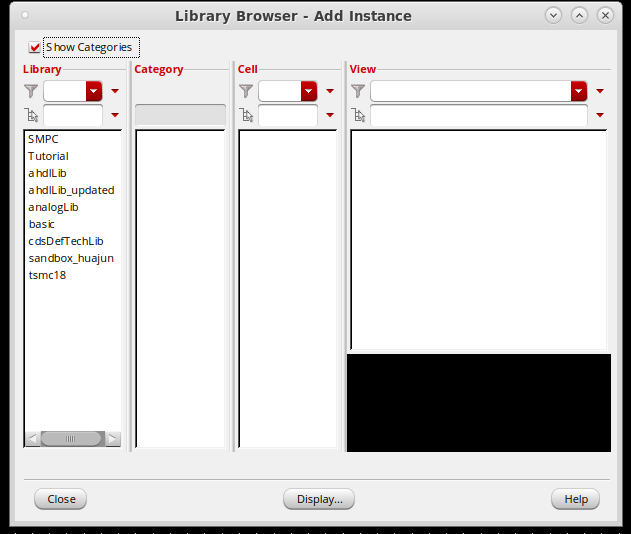


Click OK and a new window with a blank schematic will pop up. (If a dialog about license appears, click Yes and proceed.)

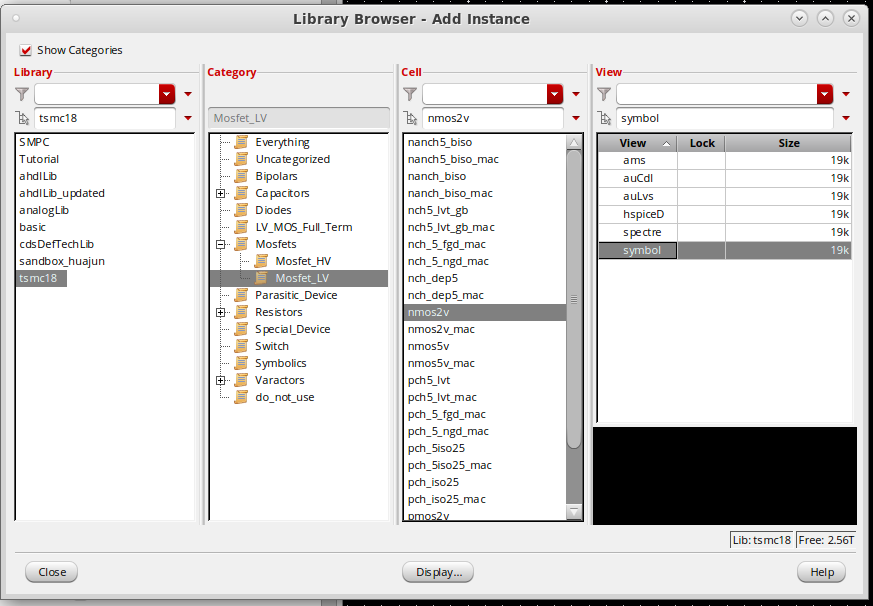


* 1. **Add Device Instances**

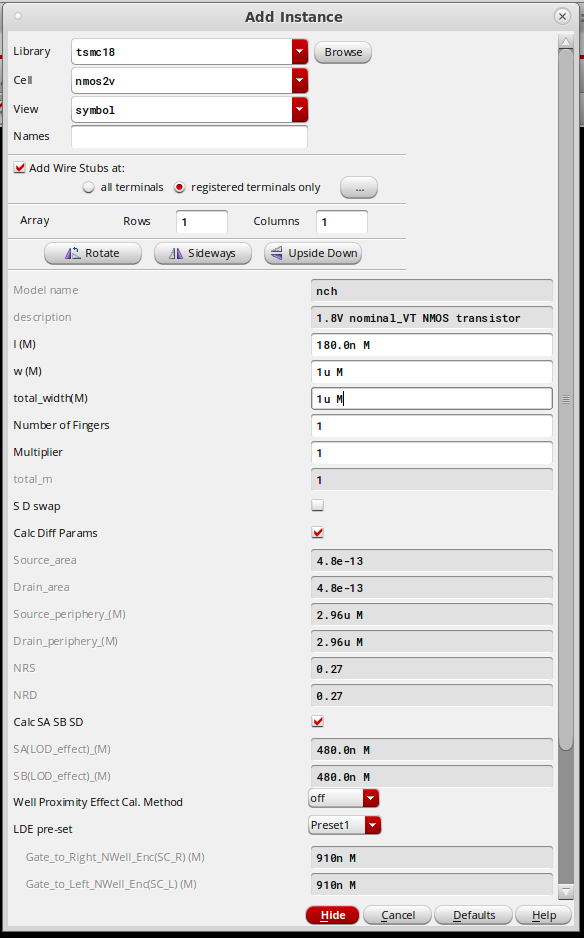
Make sure you are working in the schematic window by left-clicking a blank spot in the schematic. Create an instance using the shortcut “i” on the keyboard. A dialog called “Add Instance” will appear. Click “Browse.” A window similar to the library manager will appear (Library Browser). Check the “Show Categories” option.



Select “tsmc18” under library, Mosfets🡪Mosfet\_LV for Category and select nmos2v.



Click “Close” in the Library Browser and return to the “Add Instance” dialog. Here you can change the device parameters, including gate length, gate width, and the number of fingers, etc. Change gate width to 1um and click Hide (shortcut ESC).

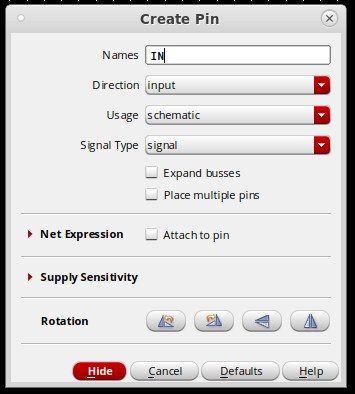


Use your mouse to place the nmos symbol onto the schematic (left-click). Before putting down the symbol, you can re-invoke the “Add Instance” dialog by shortcut F3 to modify parameters.

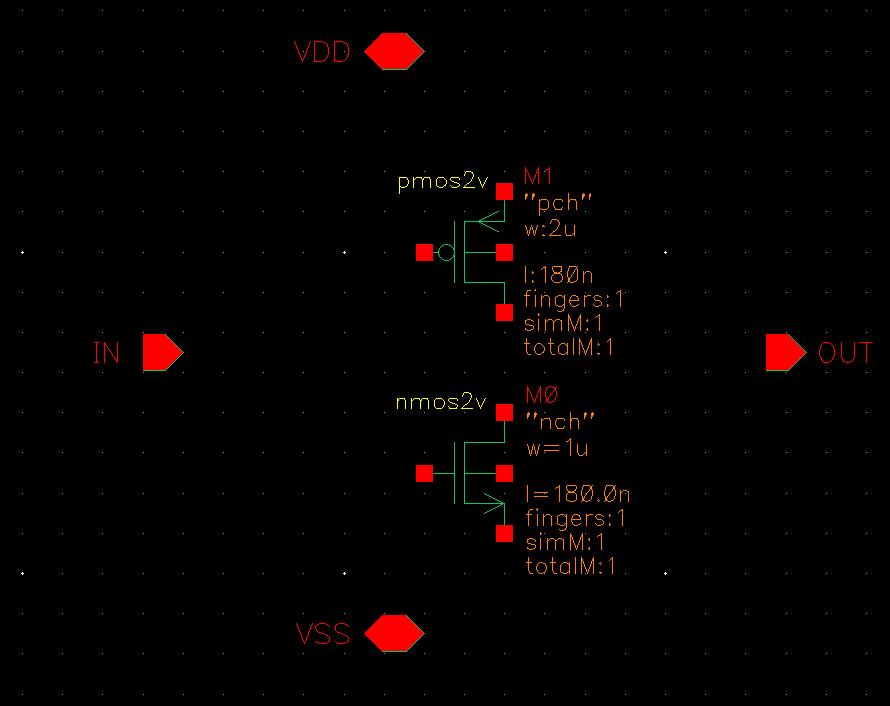
Add a pmos2v device (W=2u,L=180n) in a similar way.

* 1. **Add Pin**

It is good practice to keep a well-organized design hierarchy. For this, we first need to define the input and output pins. Create a pin using the shortcut “p”. Create a pin called “IN,” select “input” under Direction, and place it onto the schematic.

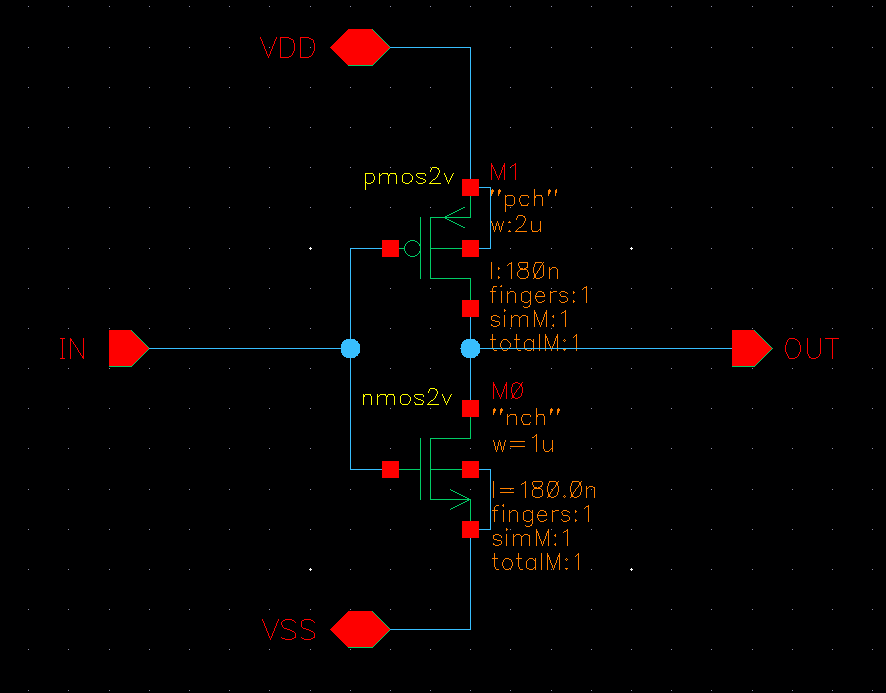


Similarly, create “OUT” for the output and “VDD” and “VSS” for the power supplies. For the supplies, select “InputOutput” under Direction.



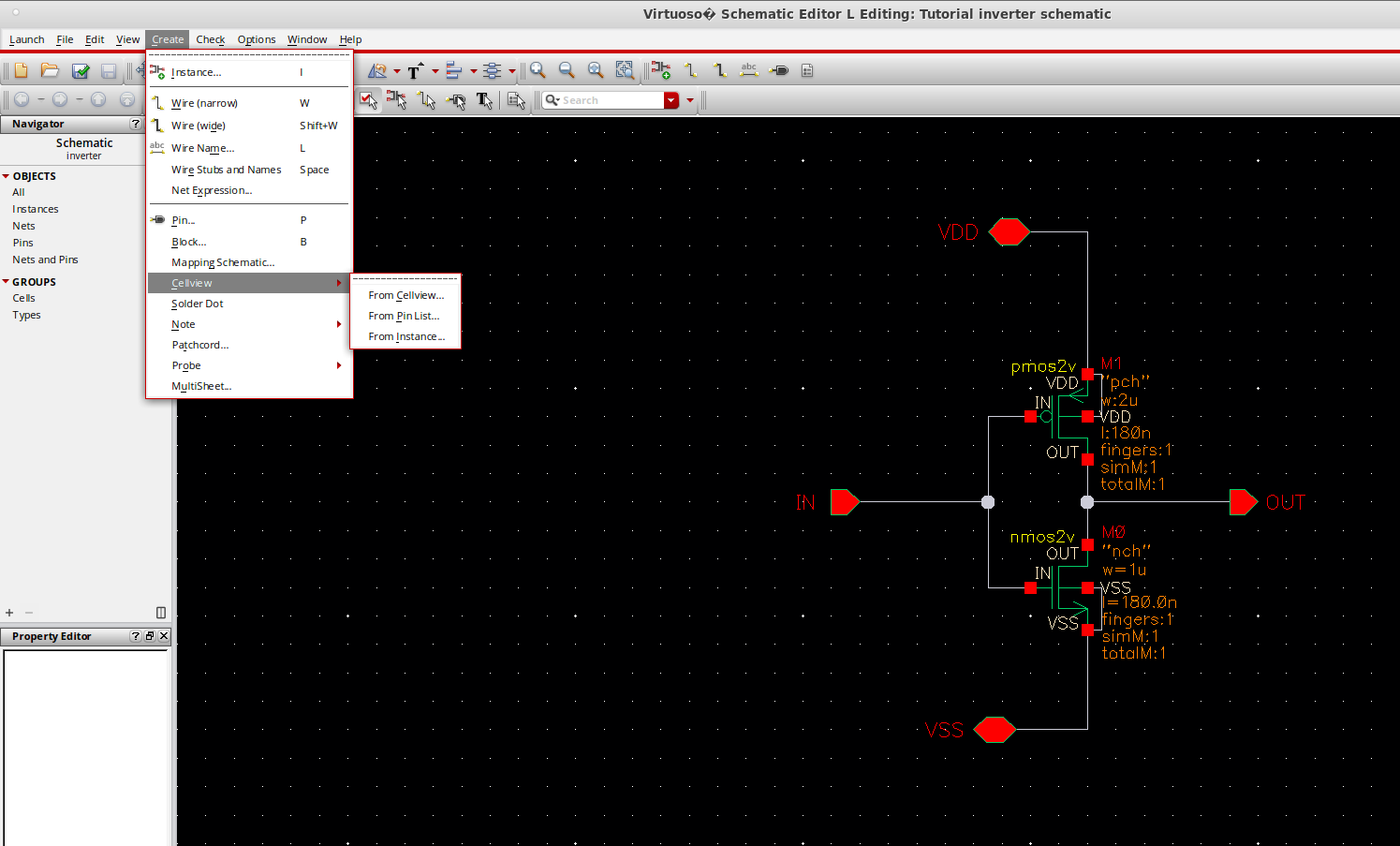
* 1. **Connect the Instances**

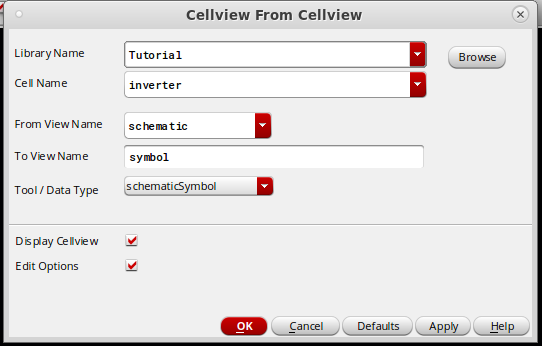
In your schematic, create wires using the shortcut “w.” Left-click on both the starting and ending points to finish a connection. You can also left-click to define intermediate points. If the wire terminates in blank space, you need to double-click. Connect the devices and pins as follows.



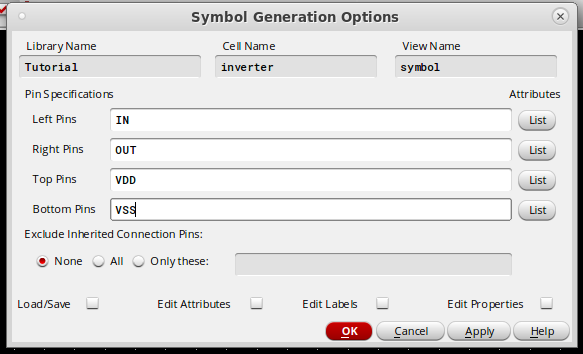
* 1. **Create a Symbol**

To use the inverter as a block in another schematic, a symbol is needed. In the schematic editor, go to Create 🡪 Cellview 🡪 From Cellview.

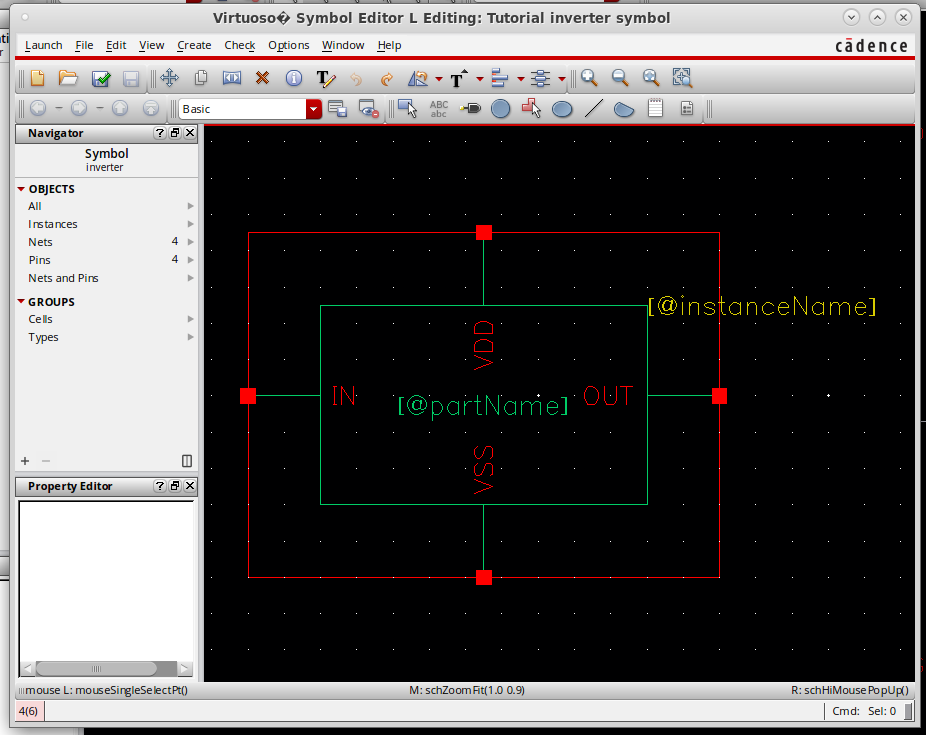
Click OK for the first dialog (below).



In Symbol Generation Options, move VSS from Top Pins to Bottom Pins and then click OK.



The symbol editor will appear. The VSS pin is placed at the bottom following our specifications. You can make the symbol more readable by creating lines and circles, etc.

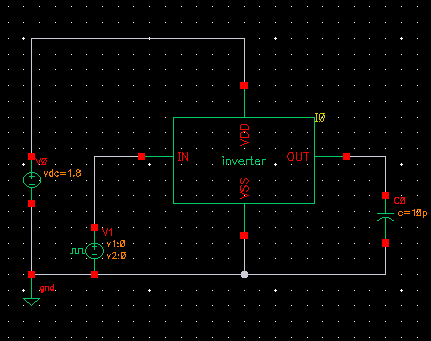


You can also move things around. To do so, press down the left key and drag your mouse to select. Then, hit shortcut “m” and left-click on the selected parts. To rotate, use shortcut “r” after you hit “m.” Left-click to place the parts to the new location. The same applies when moving things in the schematic editor.

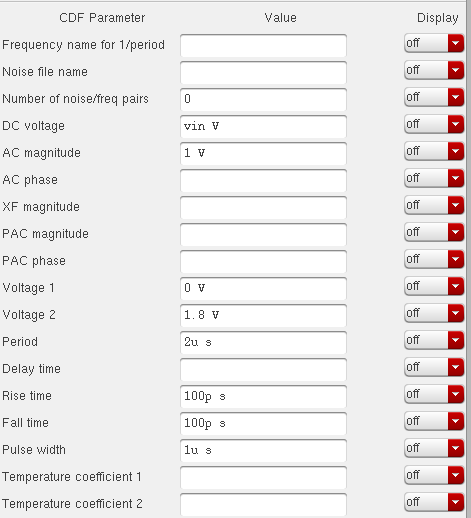
Save your symbol with File 🡪 Check and Save (Shortcut: Shift+X).

1. **Create a Testbench**

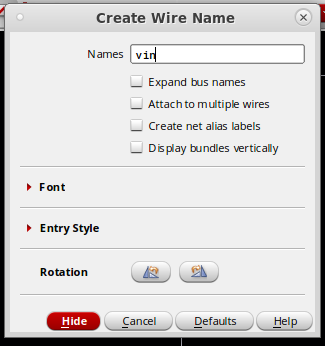
We create a new schematic for the inverter’s testbench for simulation. Create a new schematic and place an instance of the inverter (select Symbol under View in the Library Browser). Place a 10pF *cap* for the output load, a *vpulse* next to the input, a 1.8V *vdc* for VDD and *gnd* for VSS, all from Library *analogLib*, and wire them up. The finished testbench should be as below.

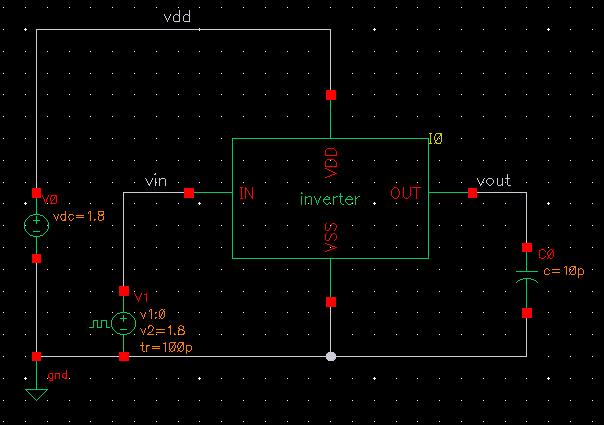


To define the input waveform, we need to add some specifications to the *vpulse* instance. Left-click on the instance and hit “q” on the keyboard. Fill in the dialog as below. DC voltage applies for DC simulations and also defines the operating point for AC simulation. AC magnitude is used for AC simulation. The parameters from *Voltage 1* to *Pulse width* apply for transient simulation only.



Circuit nodes (aka. nets) can be labeled to make the schematic, and simulation results more readable. Hit “L” on the keyboard (make sure your Caps Lock is OFF), fill in the name and place the label onto the corresponding net using your left mouse key.

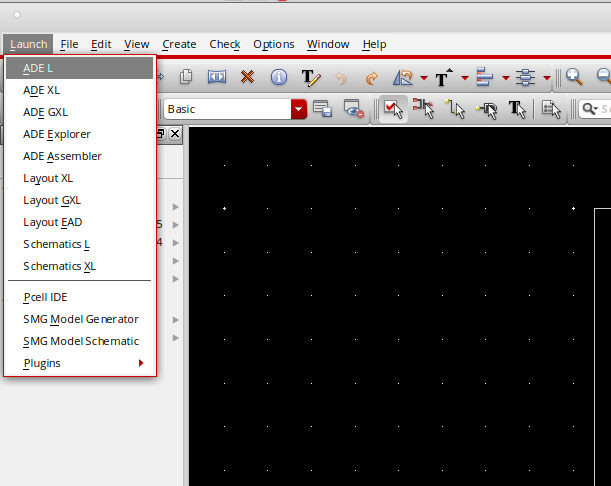




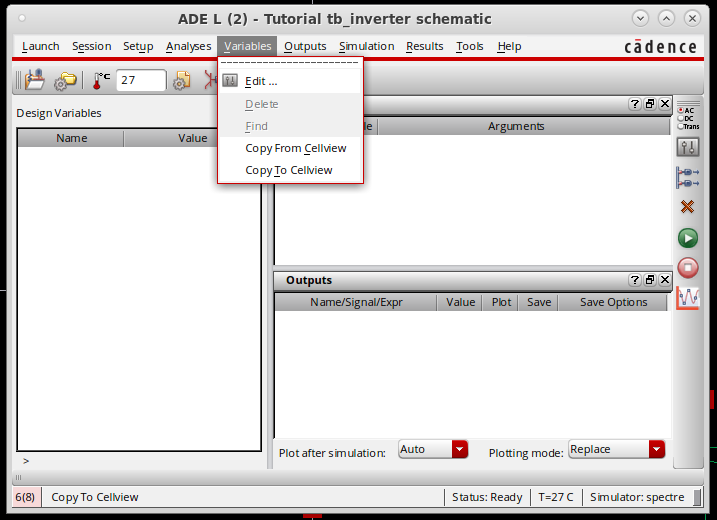
Save the testbench by File 🡪 Check and Save (or Shift+X).

1. **Setup Simulation**

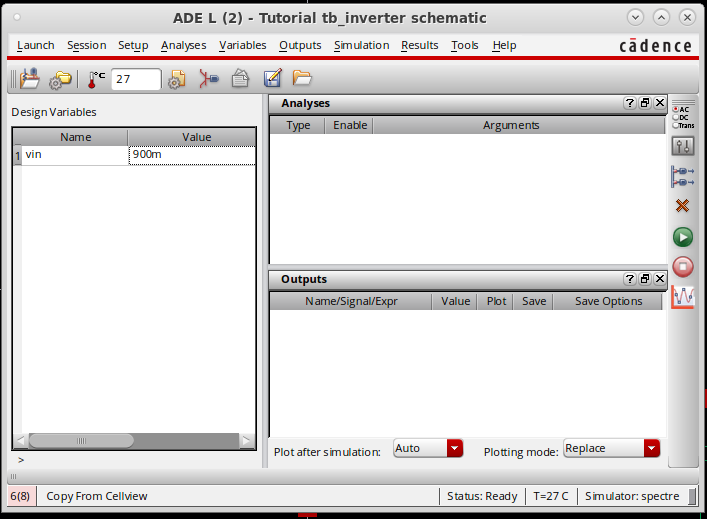
In the schematic editor, click Launch 🡪 ADE L. The ADE L window will appear. (If a dialog about license appears, click Yes and proceed.)



Simulation is conducted through this window. Click Variables 🡪 Copy from Cellview.



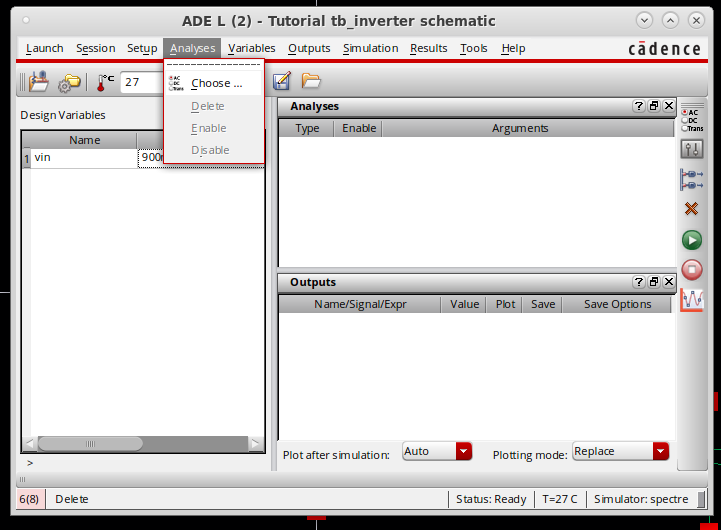
The variable “vin” will appear in the Design Variables section. Give a (default) value of 0.9 (or 900m).



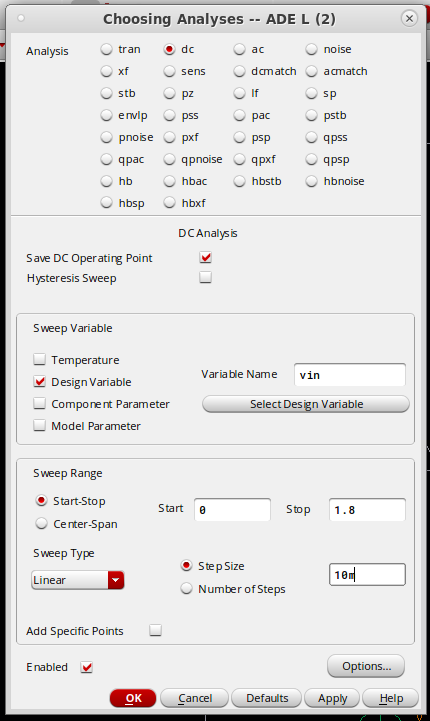
* 1. **DC Simulation**

Now we run a simulation to look at the input-output transfer curve of the inverter.

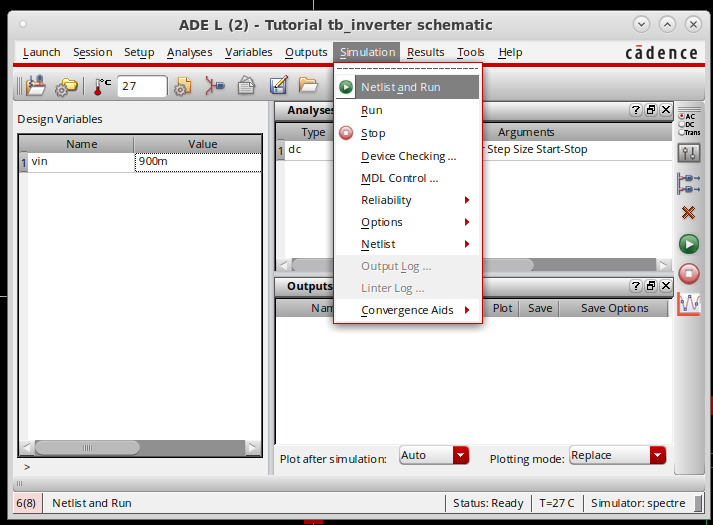
Go to Analyses 🡪 Choose…



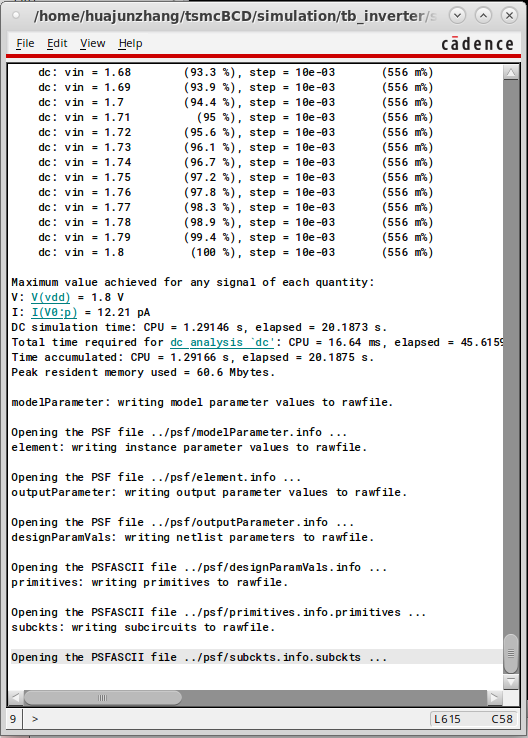
Select dc under Analysis. Check “Save DC Operating Point.” Under “Sweep Variable,” select “Design Variable.” Fill in “vin” for the Variable Name and give a sweep range from 0 to 1.8. Select Linear as the Sweep Type and give a step size of 10m. Click OK to finalize the setup.



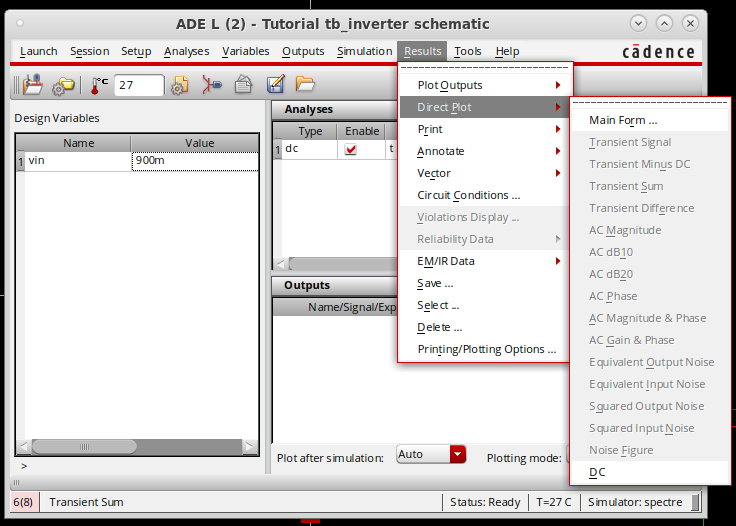
To run the simulation, click Simulation 🡪 Netlist and Run.



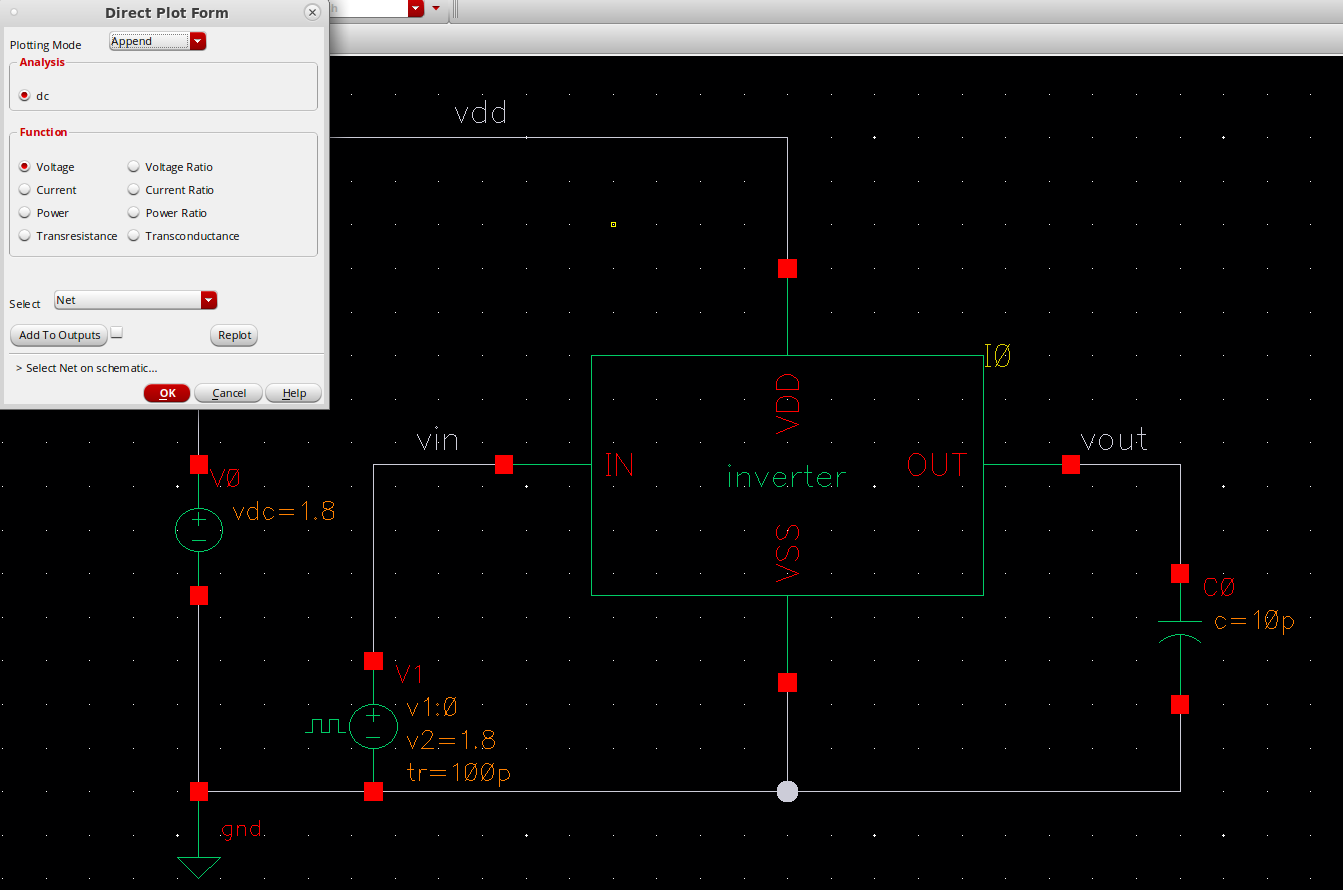
A simulation log pops up and if your setup is correct, the last part of the log should be as below.

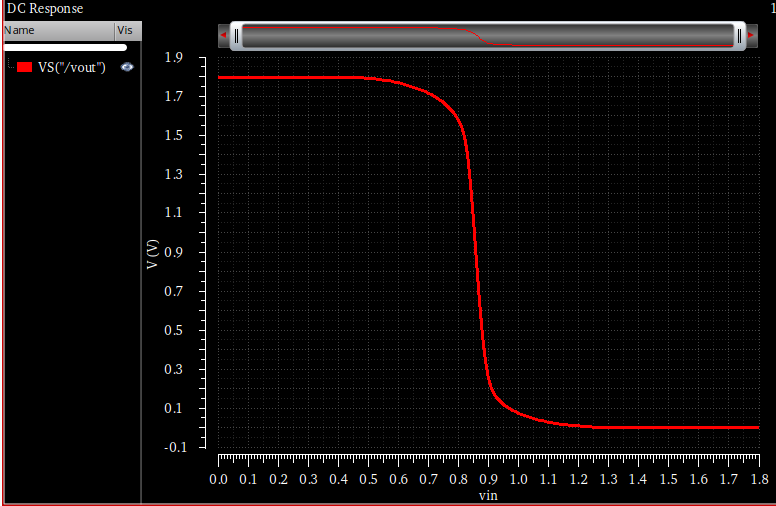


Now we can view the simulation result. Click Results 🡪 Direct Plot 🡪 Main Form…



Select Voltage under Function and click on the output net. The Virtuoso Visualization and Analysis (VIVA) window appears and shows the expected input-output transfer of an inverter. Notice that the transition point is slightly lower than 0.9V.

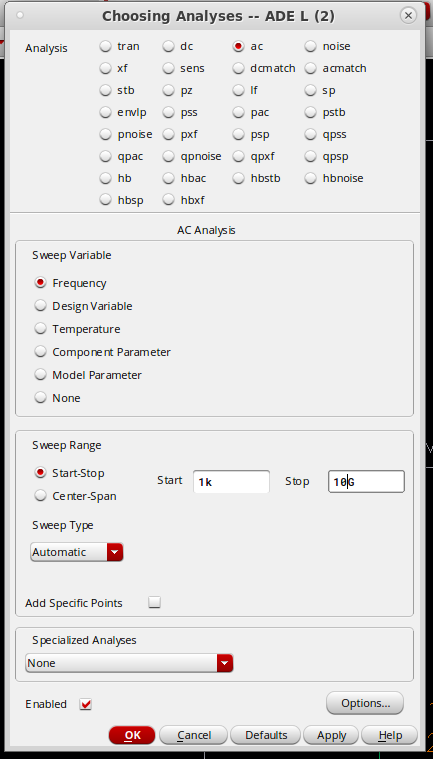




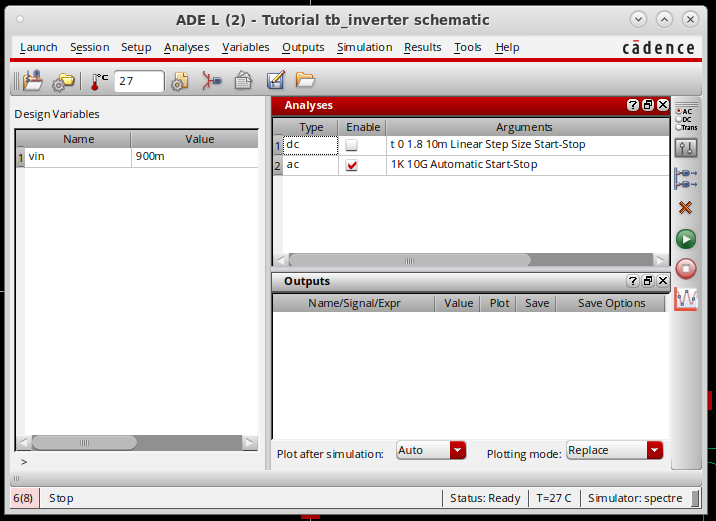
* 1. **AC Simulation**

We use ac simulations to find out the small-signal response of a circuit. Go to Analyses 🡪 Choose…

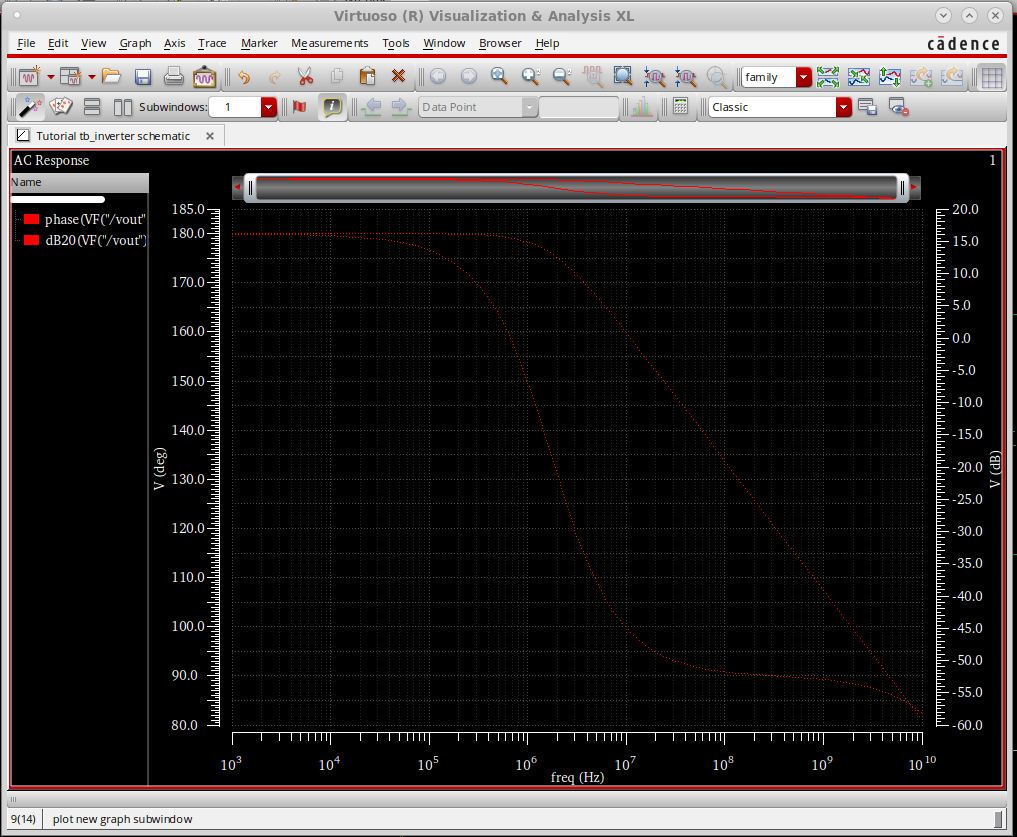
Select ac in Analysis. Set the frequency range from 1k to 10G and click OK.



Now an ac simulation will also appear in the list of Analyses, under the dc simulation. AC simulations can be run independently. Uncheck the Enable option for the dc simulation to run the ac simulation only.

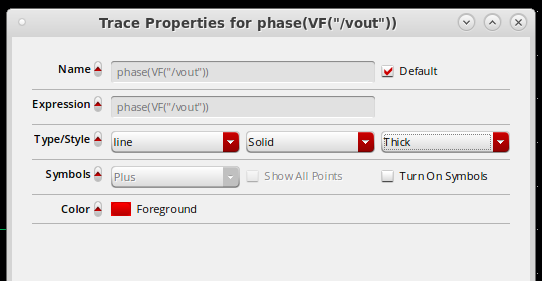


Run the ac simulation with Simulation 🡪 Netlist and Run. To view the frequency response, go to Results 🡪 Direct Plot 🡪 AC Magnitude and Phase. Select the output net with a left-click and hit ESC. The phase and magnitude response of the inverter should appear in VIVA.

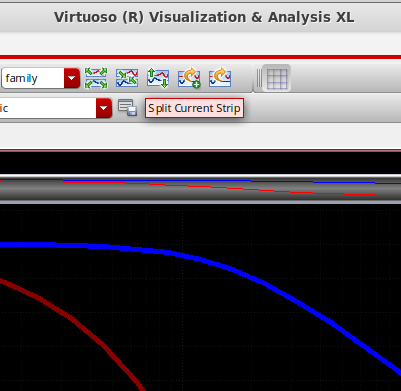


Note that this is the small-signal response around an input biasing point of 900mV, as specified by variable “vin.” These curves are too thin and hard to read.

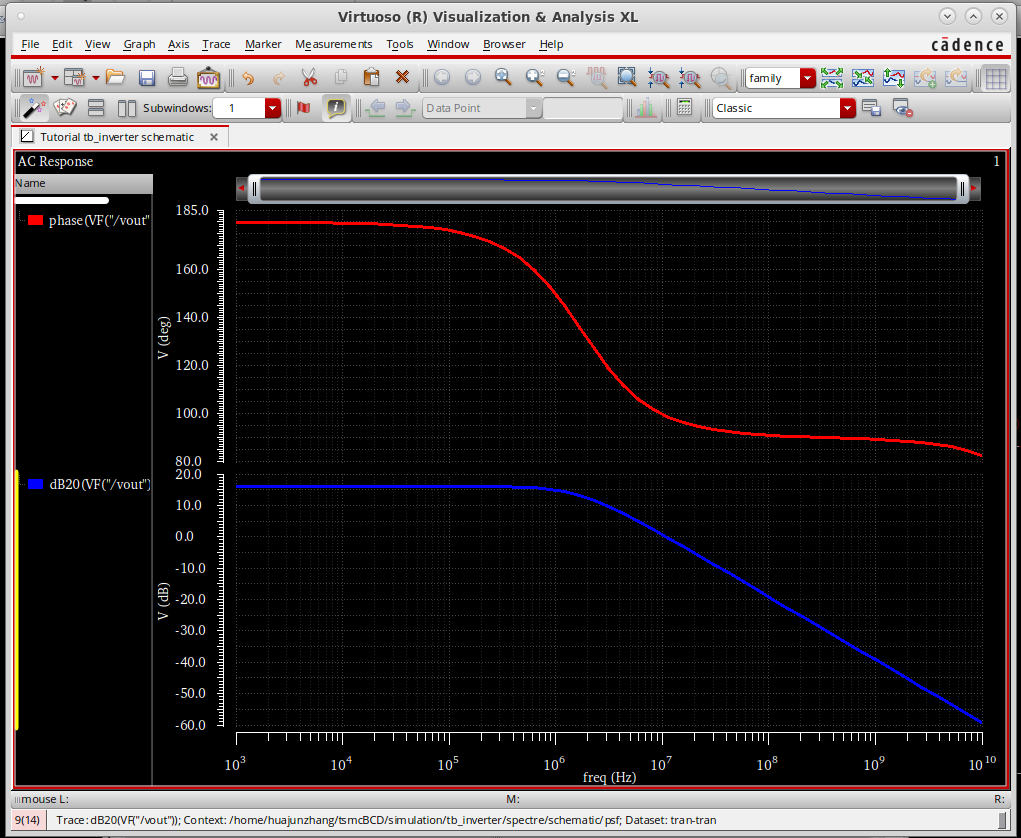
To change the appearance, click on one curve and press the shortcut “q”. Change the settings like below and click OK. Repeat for the other curve and choose a different color. **You should always make your plots legible in your assignment submissions.**



It is also possible to separate the curves. Select both curves by pressing down Shift and select them with your mouse. Click “Split Current Strip” at the top.



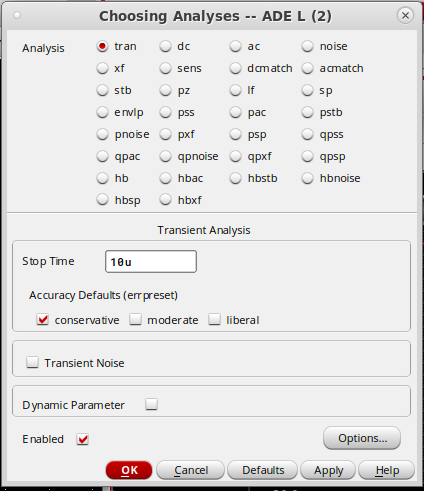
The result should look like below.



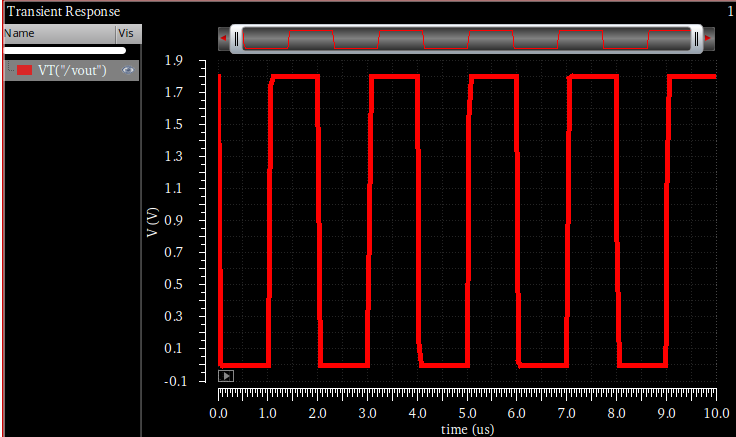
* 1. **Transient Simulation**

Similarly, set up the transient simulation under Analyses 🡪 Choose…

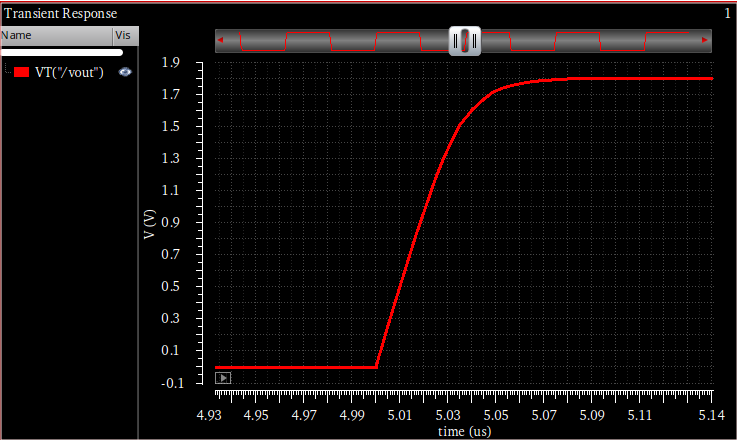
Fill in a stop time of 10u and select “conservative” under Accuracy Defaults. For analog circuit simulations, “conservative” is used most of the time. Click OK to finalize the settings.



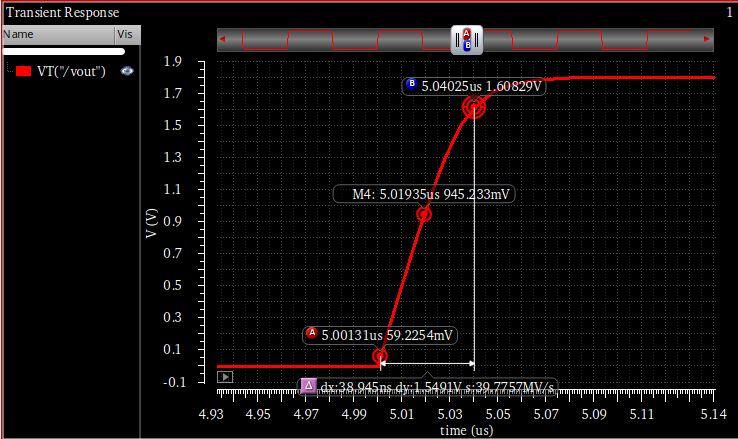
Run the transient simulation by, again, clicking Simulation 🡪 Netlist and Run. To view the results, go to Results 🡪 Direct Plot 🡪 Transient Signal. Select the output wire in Schematic Editor and hit ESC. An output waveform close to a square wave should show up in VIVA.



To observe the switching edges, zoom in by pressing down the right mouse key and dragging across the intended area. Alternatively, press down Shift and use your mouse wheel to zoom horizontally and press down Ctrl for the vertical direction.

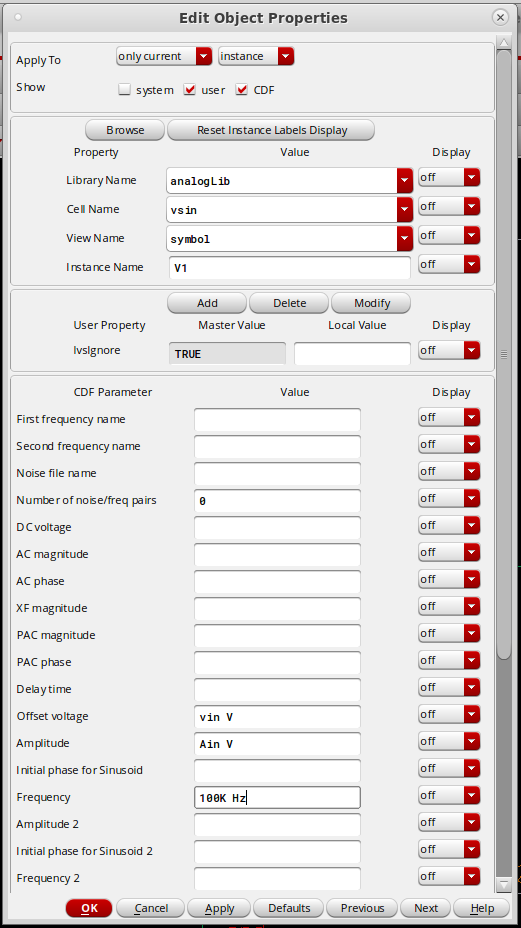


Measurement can be done by creating markers. Hover your cursor on the curve and press “M” on your keyboard to create a marker. Differential marker can then be created by pressing “A” for the first point and “B” for the second point.

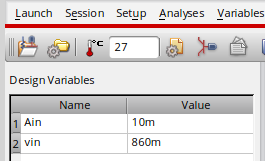


* 1. **Plotting FFT from transient simulations**

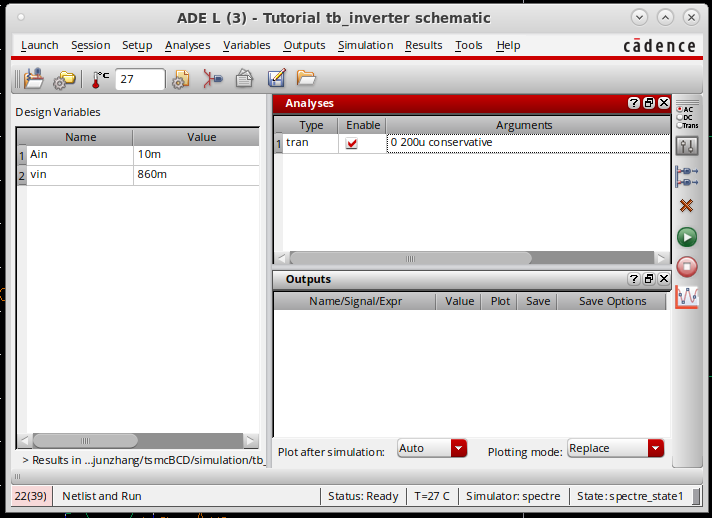
Next, we will replace the square wave input with a sine wave input and generate an output spectrum from a transient simulation. First, delete the input signal source by selecting it and press Delete on your keyboard. Add a *vsin* from *analogLib* as the new input source. Fill in the parameters as below.



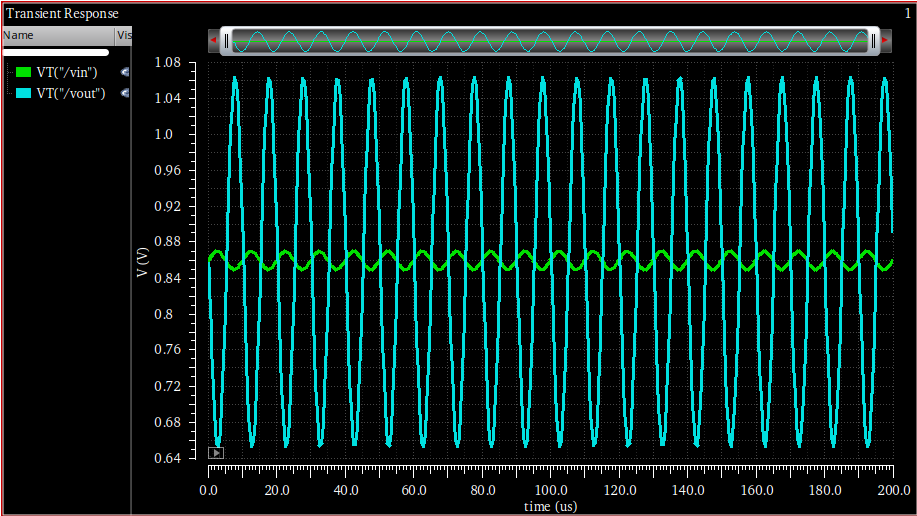
Go to the ADE L window and update the variable list with Variables 🡪 Copy From Cellview. In the list, fill in 10m for *Ain* and change *vin* to 860m. As a result, the input of the inverter is now a sine wave with a 10mV amplitude and 860mV DC offset.



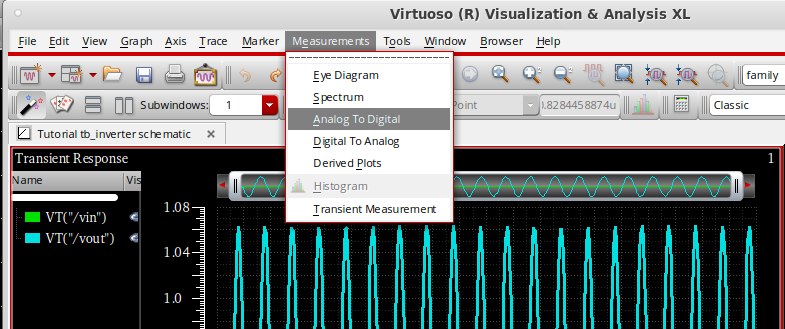
Increase the transient simulation time to 200us by double-clicking the transient analysis row under Analyses in ADE and changing the Stop Time to 200u. Your ADE should be as below.



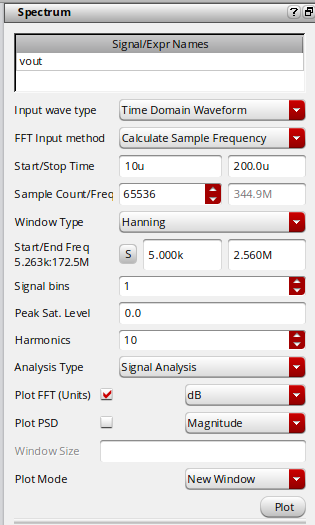
Rerun the transient simulation with the new settings by Simulation 🡪 Netlist and Run. Plot the transient waveform of input and output with Results 🡪 Direct Plot 🡪 Transient Signal. Click on *vin*, and then, *vout* on the schematic, and hit ESC after you have selected both. The input and output waveform should appear in VIVA. The inverter behaves as an amplifier for a small signal input biased at its tripping point (~860mV in this case).



To view the output spectrum, select the output trace in VIVA, go to Measurements 🡪 Spectrum.



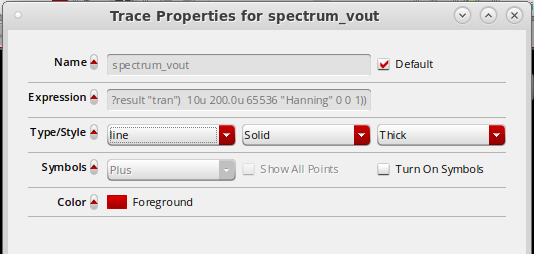
A small window will appear on the right. Fill in the form as below and click Plot.



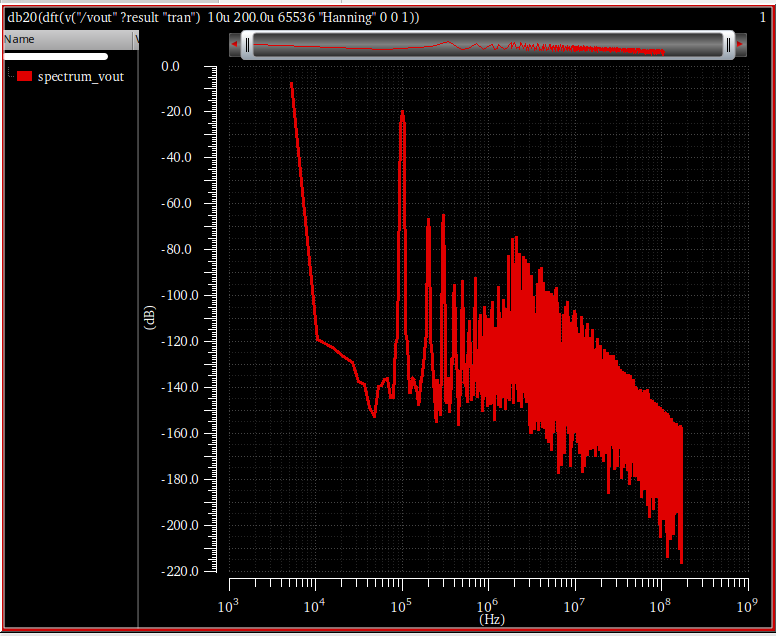
A new tab should appear in VIVA which plots the spectrum of *vout*. To change the x-axis to log scale. Click on the axis and press “q”, and check the “Log” option.



To make the plot more readable, click on the spectrum and press “q”, change “spectral” to “line” as below.



The result is as below.



**Miscellaneous Information**

* Cadence generates a lock file to allow only one user to edit a file at a given time in team projects. The lock is removed automatically when you close the file. If your Cadence session crashes, the lock file will not be removed as usual and you cannot edit the locked files after restarting Cadence. To fix this, in Terminal, *cd* to the directory where you start Cadence and execute the following commands to delete the lock files.

**source sourceme  
clsAdminTool  
> are .  
> quit**

* To change your default trace appearance in VIVA, open the file “.cdsinit” under the directory where you start Cadence with a text editor (e.g. *gedit*).

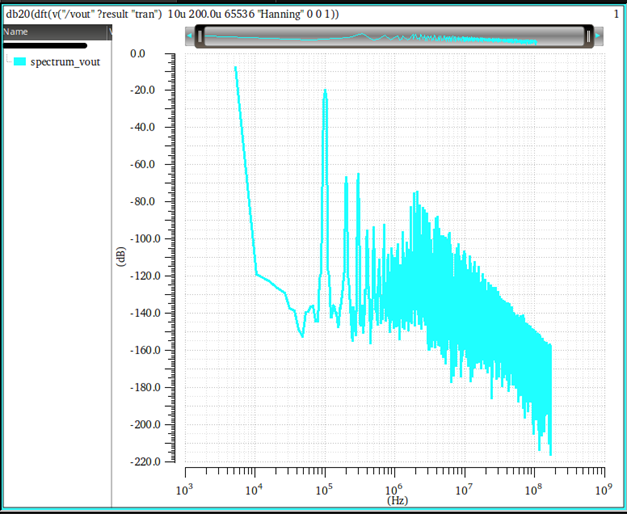
**gedit .cdsinit&**

Add the following lines to the file.

**envSetVal("asimenv.plotting" "useDisplayDrf" 'boolean nil)  
envSetVal("viva.trace" "lineStyle" 'string "solid")  
envSetVal("viva.trace" "lineThickness" 'string "thick")**

Save and close the file. Restart Cadence.

* In your assignment and project submissions, you should invert the colors for simulation plots to make a white background like the figure below. To do this, you can copy your image to Microsoft Paint and use the Invert Color feature. (See <https://www.youtube.com/watch?v=wrSxLPwM5lI>)



**Exercise**

Design a differential amplifier (shown below) using components from the *tsmc18* library to meet the specifications in Table 1. Capture the schematic in Cadence and verify its performance with simulations.

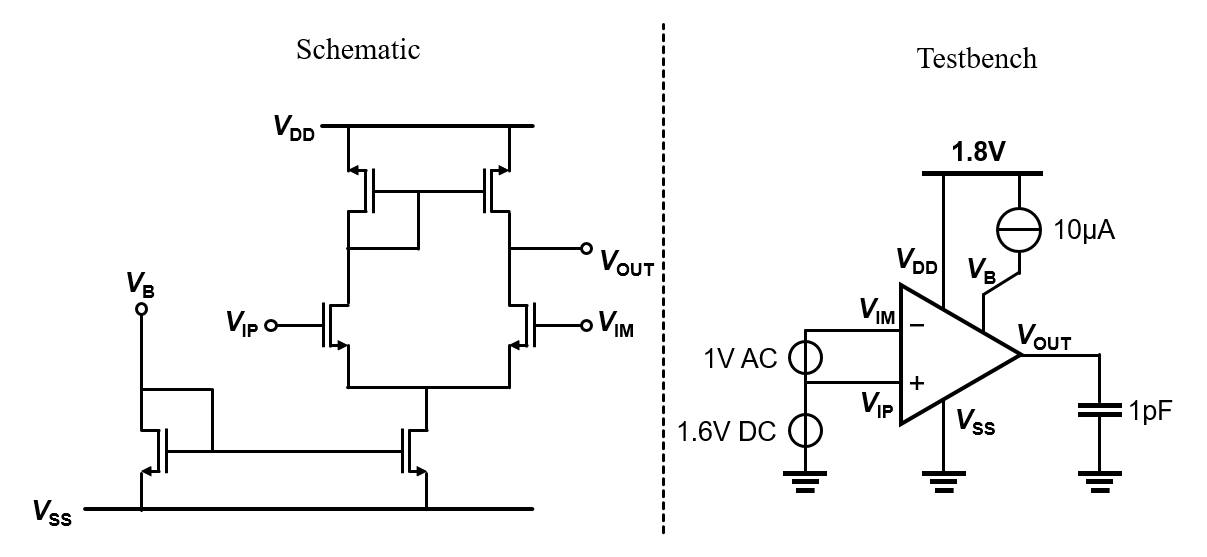


Table 1. Performance Specifications.

|  |  |
| --- | --- |
| **Specification** | **Value** |
| Supply Voltage | 1.8V |
| Input Common-Mode | 1.6V |
| Current Reference | 10μA |
| Load capacitance | 1pF |
| Unity-Gain Frequency | ≥100 MHz |
| DC Gain | 40 dB |
| Current Consumption | Minimize |

**Deliverables**

1. A legible figure for the schematic of your design, clearly showing device dimensions.
2. A legible figure for the testbench.
3. Some analysis explaining how your devices are sized.
4. A table summarizing simulated performances and comparing them with the specs in Table 1.
5. A legible plot from AC simulations. Add markers to the plot to highlight the unity-gain frequency and dc gain values.
6. A legible plot of the output waveform and FFT from a transient simulation. For this, apply a 100kHz 10mV (peak-to-peak) sine wave across the differential input. (Hint: Use *vsin* from *analogLib*.)