Cadence Tutorial

Colin Weltin-Wu

Step 1

Before anything you need to modify your `.bash_profile` file in you root directory. Open the file `~/.bash_profile` in your favorite editor, and it should look something like this:

```
# .bash_profile

# Get the aliases and functions
if [ -f ~/.bashrc ]; then
        . ~/.bashrc
fi

# User specific environment and startup programs

PATH=\$PATH:\$HOME/bin

export PATH
unset USERNAME
```

You must add the line `PATH=/usr/cad/mmsim/tools/bin:$PATH` before the `PATH` statement, so it looks like this:

```
# .bash_profile

# Get the aliases and functions
if [ -f ~/.bashrc ]; then
        . ~/.bashrc
fi

# User specific environment and startup programs
PATH=/usr/cad/mmsim/tools/bin:$PATH
PATH=$PATH:$HOME/bin

export PATH
unset USERNAME
```

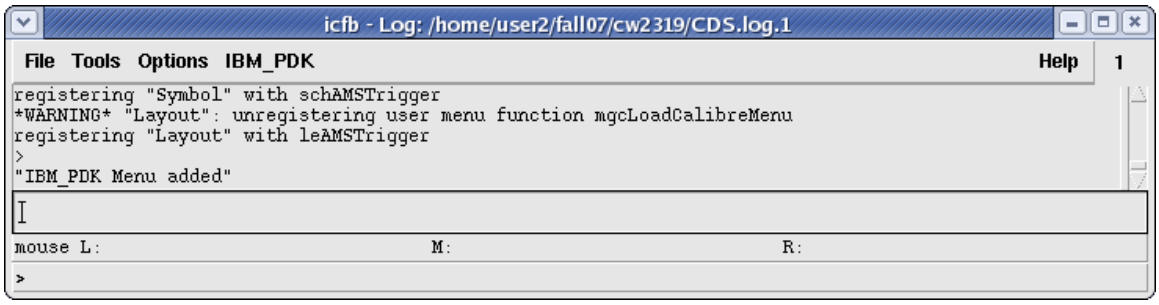
Save and exit. Check that the line is now there by typing `more .bash_profile` and the prompt.

Now create a Cadence directory for your files. You can name it "4312" for example, and create it with `mkdir 4312` at the prompt. Then change to that directory with `cd 4312`. Now

you have to set up some environment parameters. Run the following commands, replacing `<username>` with your login username:

```
cp ~tod/4312/display.drf ~<username>/4312/.
cp ~tod/4312/.cdsinit ~<username>/4312/.
cp ~tod/4312/cds.lib ~<username>/4312/.
cp ~tod/4312/.cdsenv ~<username>/.
```

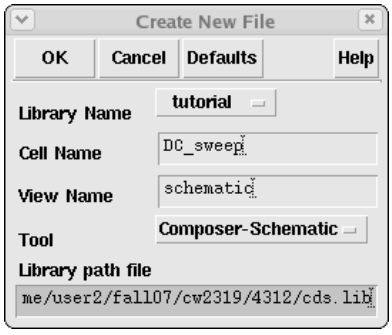
Now log out and log back in to make sure all the changes take effect. Once you're logged back in, `cd` to the Cadence directory and run cadence with `icfb &`. The `icfb` window is the main Cadence window and also serves as the console where error messages appear.



Step 2

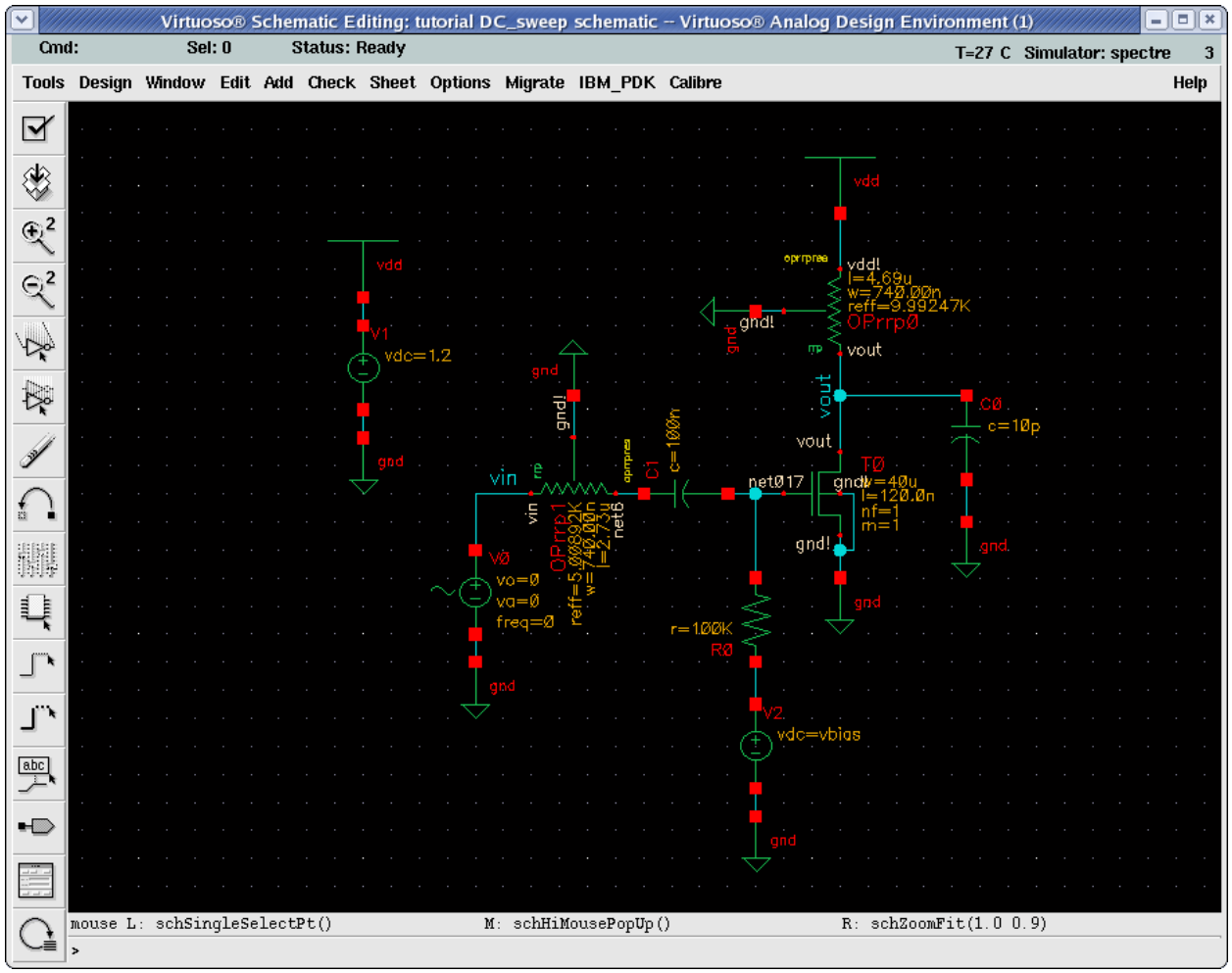
Once you have the `icfb` window, click `Tools -> Library Manager` to bring up the Library Manager. This is where all your designs are stored in a hierarchical fashion. To create a new project, go to `File -> New -> Library..` and name it anything you want, for example `tutorial`. When it asks for a techfile, select "Don't Need a Techfile".

Step 3

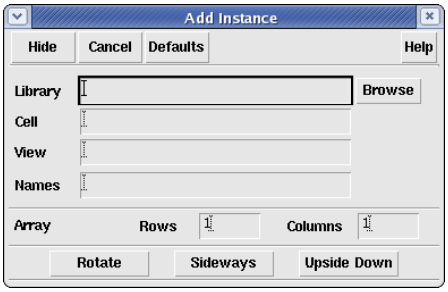


Now that we have a new library, let's put something in it. The first thing we want to simulate is a DC sweep, so go again to `File -> New -> Cell View..` and create a new Schematic view with cell name "DC_sweep", and make sure it's in the tutorial library. Once you click OK, a new virtuoso schematic editing window should come up.

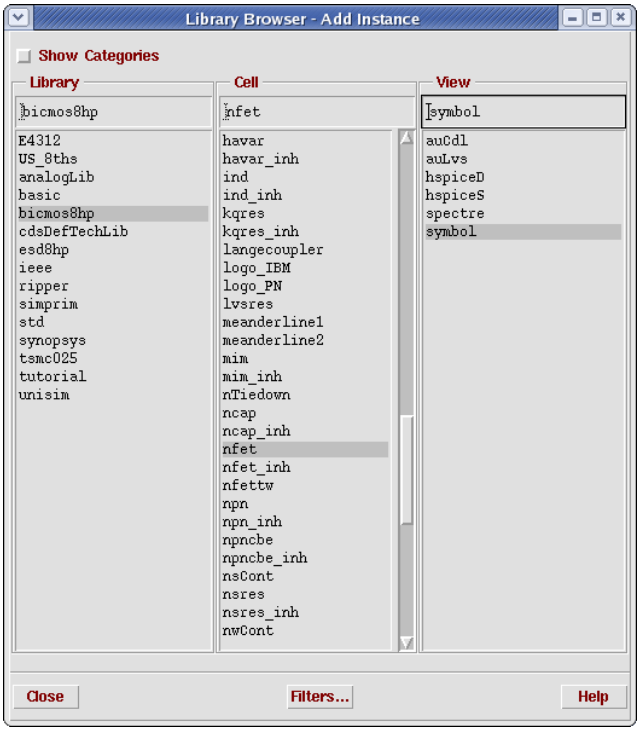
Step 4



Now we want to build a simple circuit to simulate, shown above. Cadence uses all three mouse buttons along many keyboard shortcuts. There is a contextual list of what each mouse button does at the bottom of the window. Let's put a MOSFET down first. Press `i` short for "instantiate" to bring up the instantiation dialog.



If you remember the parameters of the part you want to use you can type it directly into the fields, otherwise browse for an "nfet" symbol in the `bicmos8hp` library.



Note that the Instance window enlarges to show all the parameters associated with the item you have chosen to instantiate. Change the width of a single finger to `40u M` and leave everything else the same. Cadence accepts standard engineering suffixes of units to simplify data entry.

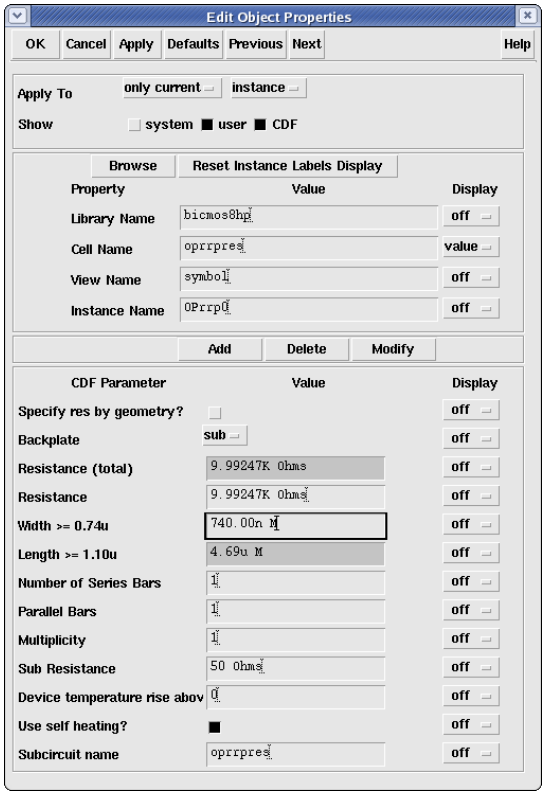
Now when you bring the mouse back over the Virtuoso window, a MOSFET is being dragged. Left click to place it down.

Step 5

Now instantiate a resistor `oprrpres` from the same library. Don't change anything, just instantiate it above the MOSFET. We will be building a CS amplifier. If you want to rotate a part while dragging, right-click. The third terminal of the resistor is the substrate, which adds a parasitic capacitance from the layout.

Now we show another way to specify the item parameters: by left-clicking to select an object, then pressing `q` to bring up the Object Properties dialog.

By unchecking `Specify res by Geometry?` you can set the resistance to 10k. Then click OK.

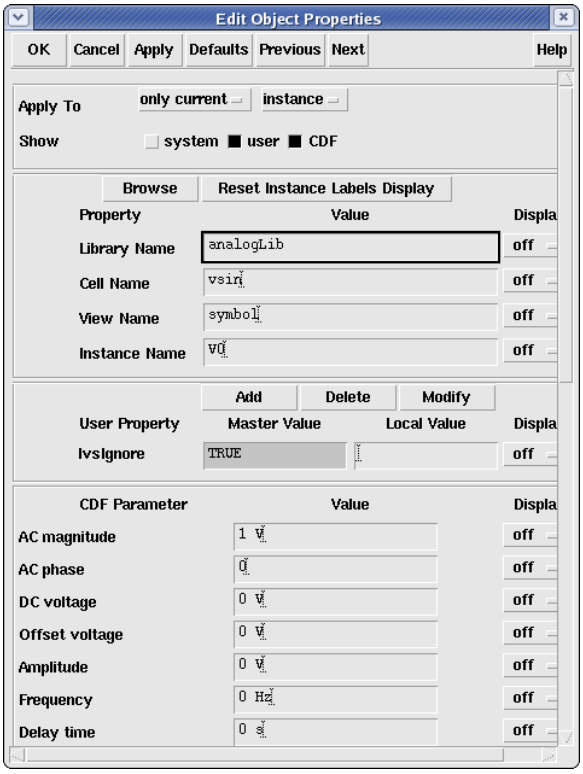


Once a cell is on the schematic, instead of instantiating a new one each time, you can simply copy existing cells by using the `c` key, then clicking the item you'd like to copy. Cadence has "sticky" keys, which means your command will work indefinitely unless you cancel out of it, or switch to a different command. Two useful keys are the `ESC` key, which gets out of whatever you're doing, and `CTL-d`, which deselects everything. (regions you may have inadvertently drag-selected)

Instantiate a gate resistor (the input source resistance) by copying and rotating the resistor, and set its value to 5kΩ. Don't be alarmed if the values change to something close but strange, Cadence rounds to the closest value possible within the constraints of layout, i.e. a resistor length of 9.2323μm is impossible so rounding may be required.

Step 6

Items such as ideal passive elements, voltage and current sources and the like are all in the `analogLib` library. Instantiate a DC power source with a `vdc` cell set to a `DC Voltage` of 1.2V. Instantiate an input voltage source `vsin` and set its parameters as shown, leaving all the others untouched for now:

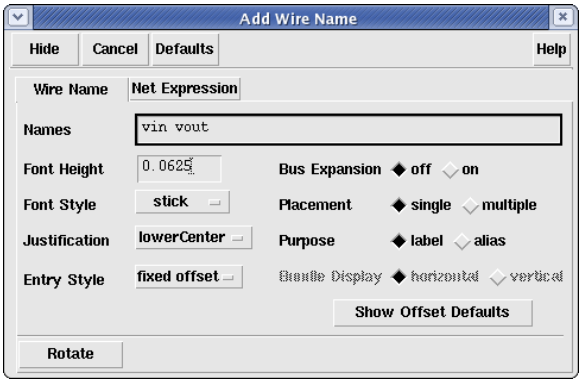


For the DC blocking cap and the load cap, use the ideal element `cap` in the `analogLib`, and set them to `100n` and `10p`, respectively. For the bias resistor, use an ideal resistor `res` also in `analogLib`, with value 100k.

Now instantiate the Vdd and Gnd symbols, which are called `vdd` and `gnd`, respectively. If you want to view what you've done, `f` will fit the window to the full schematic, `z` will let you zoom into a selected region, and `CTL-z` will zoom out of a selected region.

Step 7

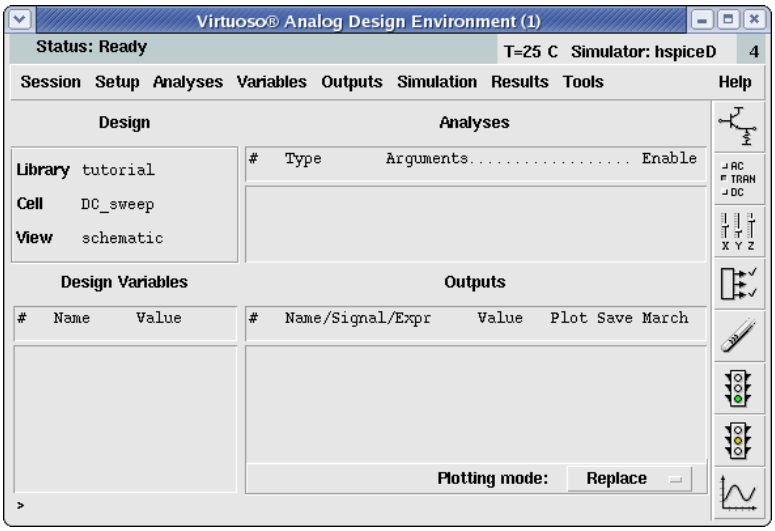
Now we want to wire it up. Press `w` to use the wire tool, and it's pretty easy. Cadence will try to snap to points that are attachable, or you can double-click to terminate a wire mid-route. Wire up the circuit, don't forget to wire up the substrate terminal of the MOSFET. When Cadence netlists the schematic to produce something to simulate, it will assign net names that aren't always readable. To specify wire names, press the `l` key (lowercase "L") to bring up the label dialog. List all the names you want to enter separated by a space, i.e.



Now click the wires you'd like to label. Your schematic should now be done and look like the picture in the beginning.

Now click the checkbox right under `Tools` to check and save the schematic. If there are any errors or warnings, Cadence will tell you. An Error will for sure cause the simulation to not run, whereas a Warning is less severe and depending on the particular case is allowable. This schematic should return nothing, because it should have zero of each.

Step 8



Now we want to run a DC simulation. Go to `Tools -> Analog Environment` to bring up the simulation window. The first thing we need to do is specify the simulation engine and the directory for the simulation. Go to `Setup -> Simulator/Directory/Host...` and choose the simulator `spectre` and the project directory `/tmp`. Once this finishes, we need to specify the model files for our devices. Do this by going to `Setup -> Model Libraries...` and then entering the following path for Model Library File:

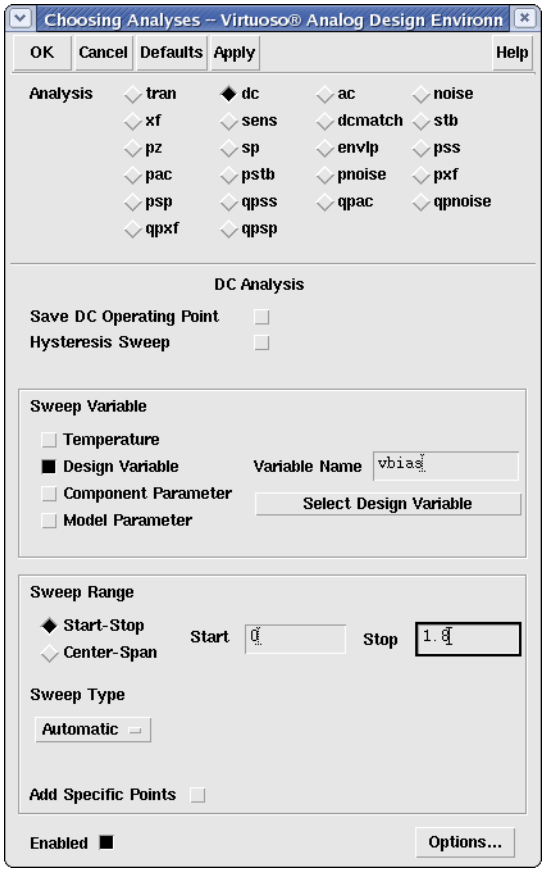
`/usr/tech/bicmos8hp/IBM_PDK/bicmos8hp/V1.1.0.0HP/Spectre/models/allModels.scs`

Then BE SURE to click "Add" to make the line appear above. Then click OK to exit.

Next we import the variables from the schematic. The easiest way to do this is have Cadence grab them from the schematic by going to `Variables -> Copy From Cellview` which should cause `vbias` to appear in the lower left window. Cadence needs all variables initialized to some value (even if it is a sweep variable and will change) so double-click `vbias`, enter `0` for its value, click `Change` then OK to exit.

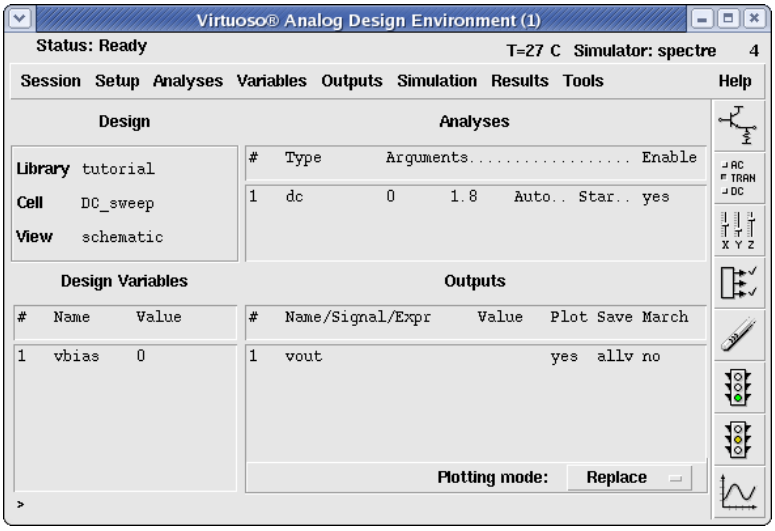
Step 9

Next go to `Analysis -> Choose` to setup our DC sweep. Configure it as shown in the picture, except you should sweep from 0 to 1.2V instead (I made a mistake):



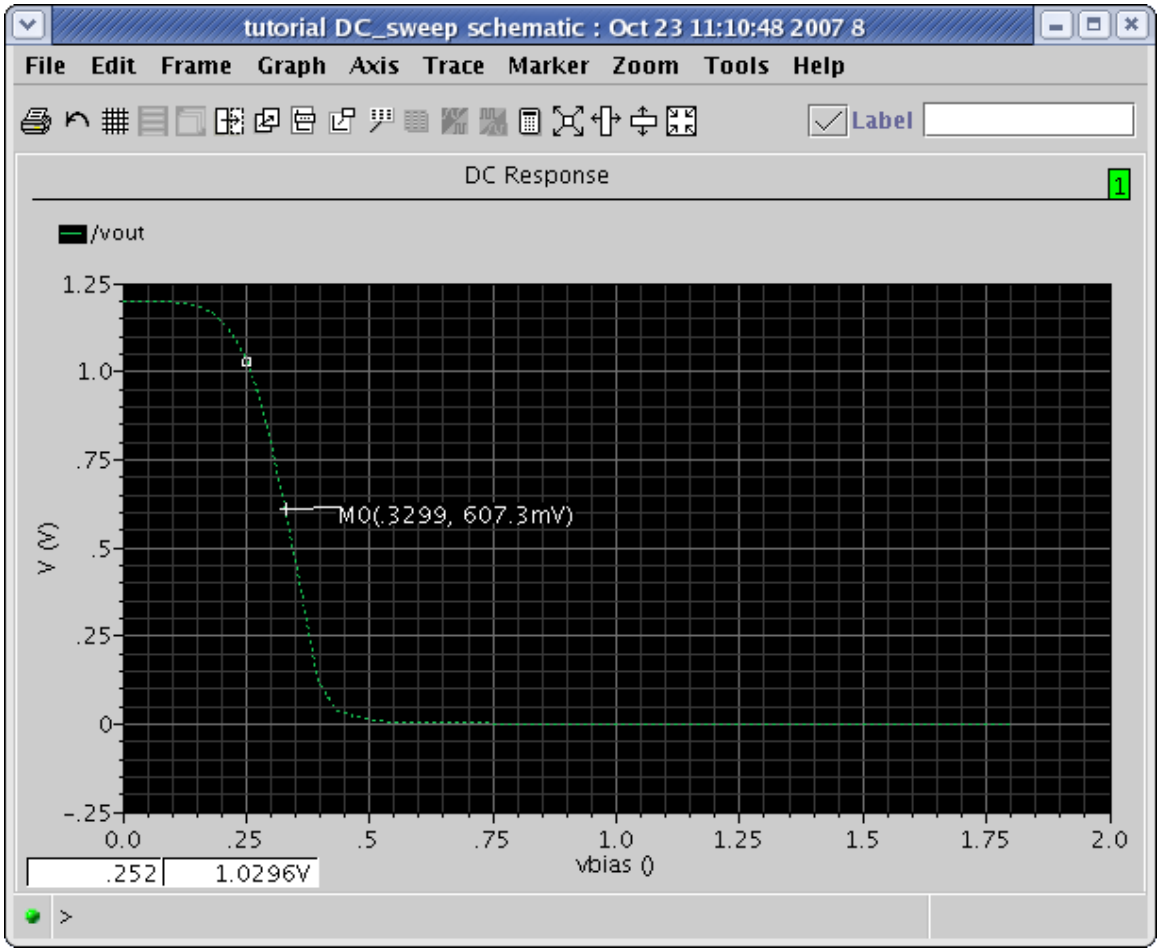
Now setup which variables you'd like to see plotted. There are again many ways to do this, but in the beginning it's easiest to just choose them on the schematic. Go to

`Outputs -> To Be Plotted -> Select On Schematic` then click `vout` in the schematic window. It should highlight. Press `ESC` otherwise you will continue to select nets, which may not be desirable. The simulator window should look like this:



Step 10

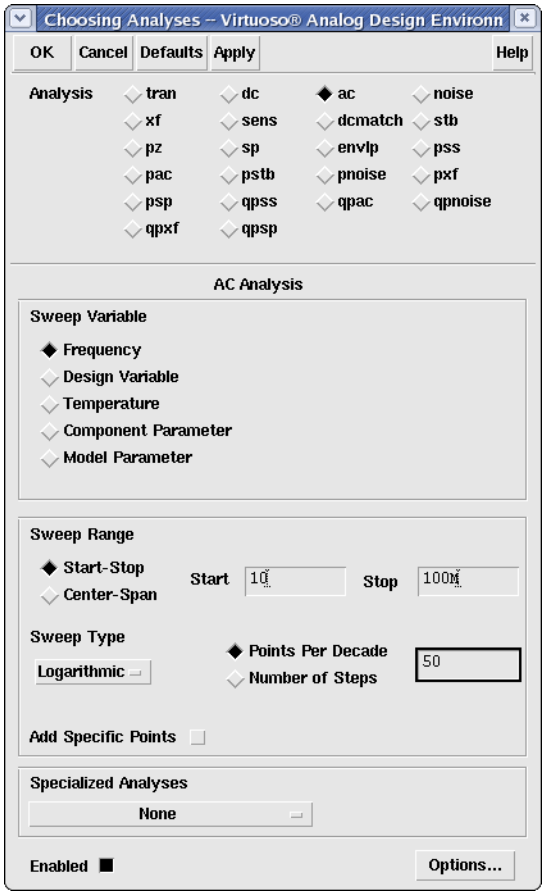
Now just click the third button from the bottom on the right, "Netlist and Run". A notification will pop up but just choose "Don't show again" and click OK, and spectre will run. You should get a plot window as so:



I placed a marker where we should set `vbias`, around 0.33V to get a DC output voltage of 0.6V which is the midpoint of Vdd. Once we've set the bias condition, we can do further analyses.

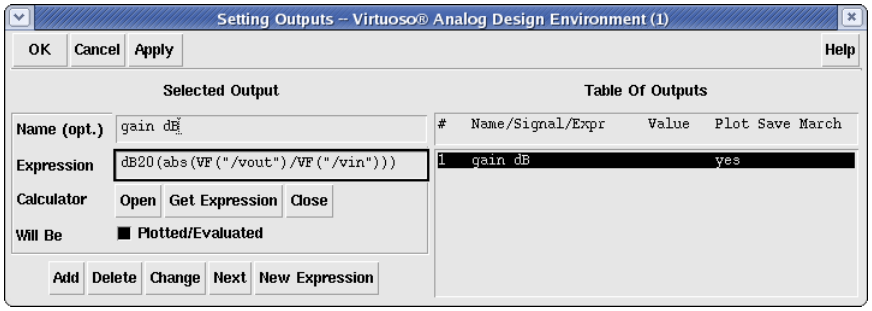
Step 11

Now we can analyze the AC performance of the amplifier. Go back to the simulation window, and first update `vbias` to 0.33V, which will bias the transistor in saturation. Now go back to `Analysis -> Choose...` and this time select `ac` analysis, and set it up like this:

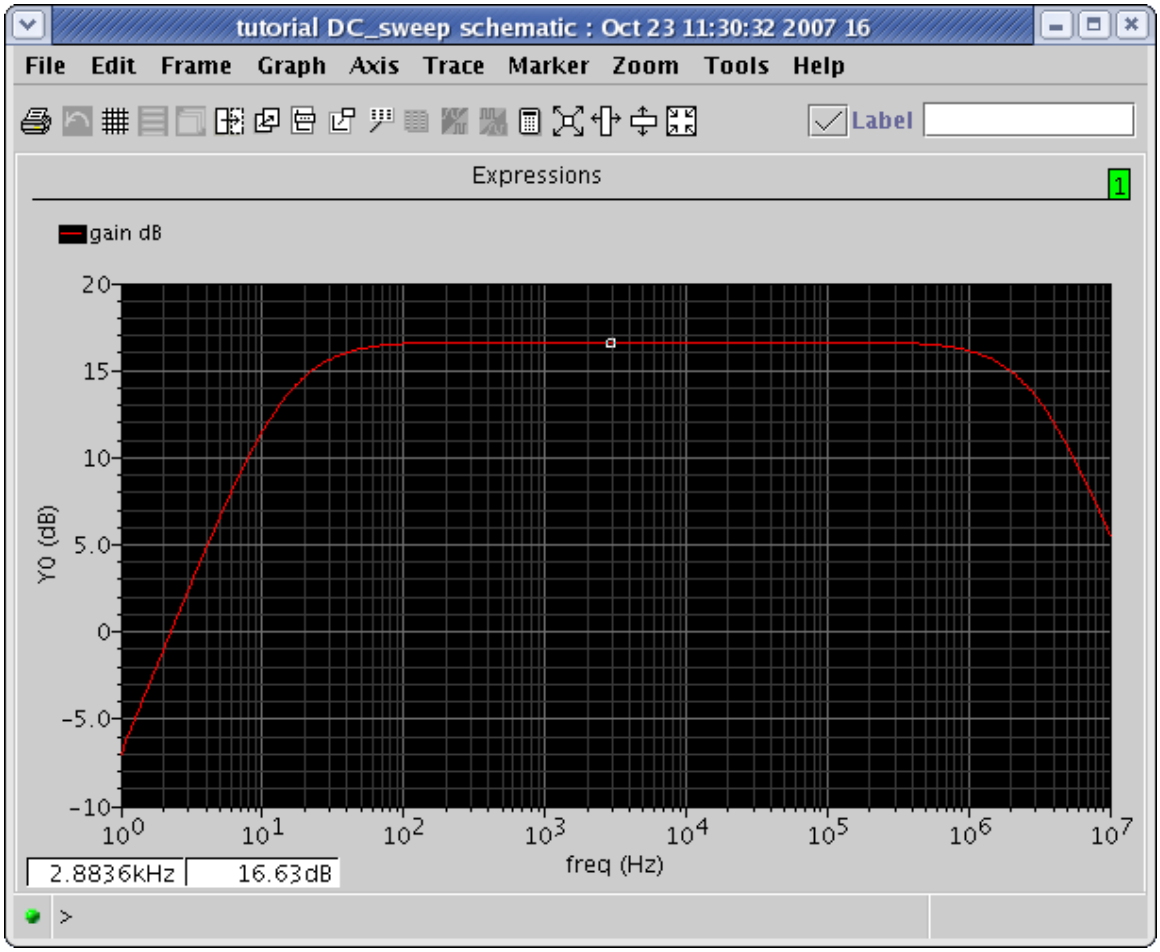


Double-click the DC analysis and in the lower-left corner de-select `enabled` so that only our AC analysis will run.

Now let's look at the gain in dB, so we double-click the `vout` under `Outputs` of the simulator window, and change it to this:



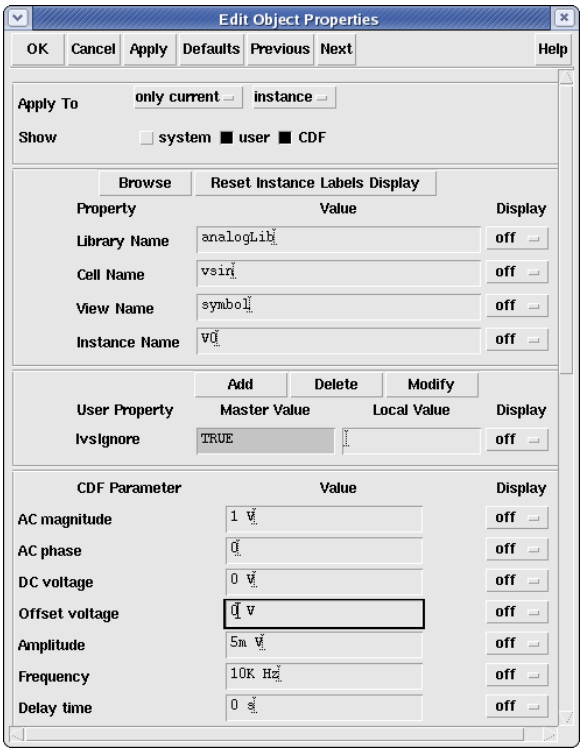
The syntax of the calculator takes some getting used to. Netlisting and running this should produce the gain plot like this:



The midband gain is 16.6dB, which isn't great but amplifies nonetheless.

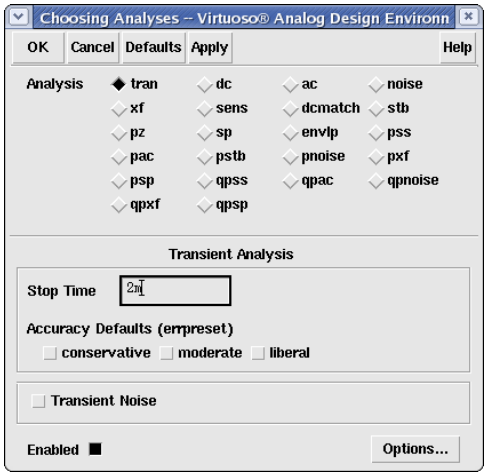
Step 12

We can also do a transient analysis. From the gain plot, a frequency of 10kHz is in-band for the amplifier. So we can go back to our `vsin` source (in Virtuoso) and change it like this:

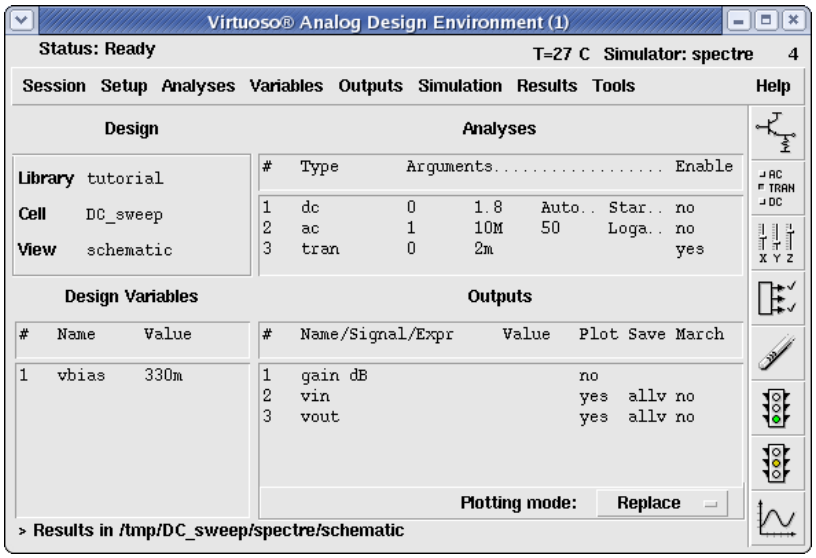


The first three parameters, `AC magnitude`, `AC phase`, and `DC voltage` are for AC analysis, whereas the latter three are corresponding parameters but for transient analyses. For the transient analysis we need to ensure we're still operating in small-signal (for the purpose of this example, but you can simulate anything you'd like) so I chose a peak amplitude of 5mV. Check and save the schematic to be sure the simulator updates, otherwise when you try to netlist and run you will get an unsuccessful warning in the `icfb` window.

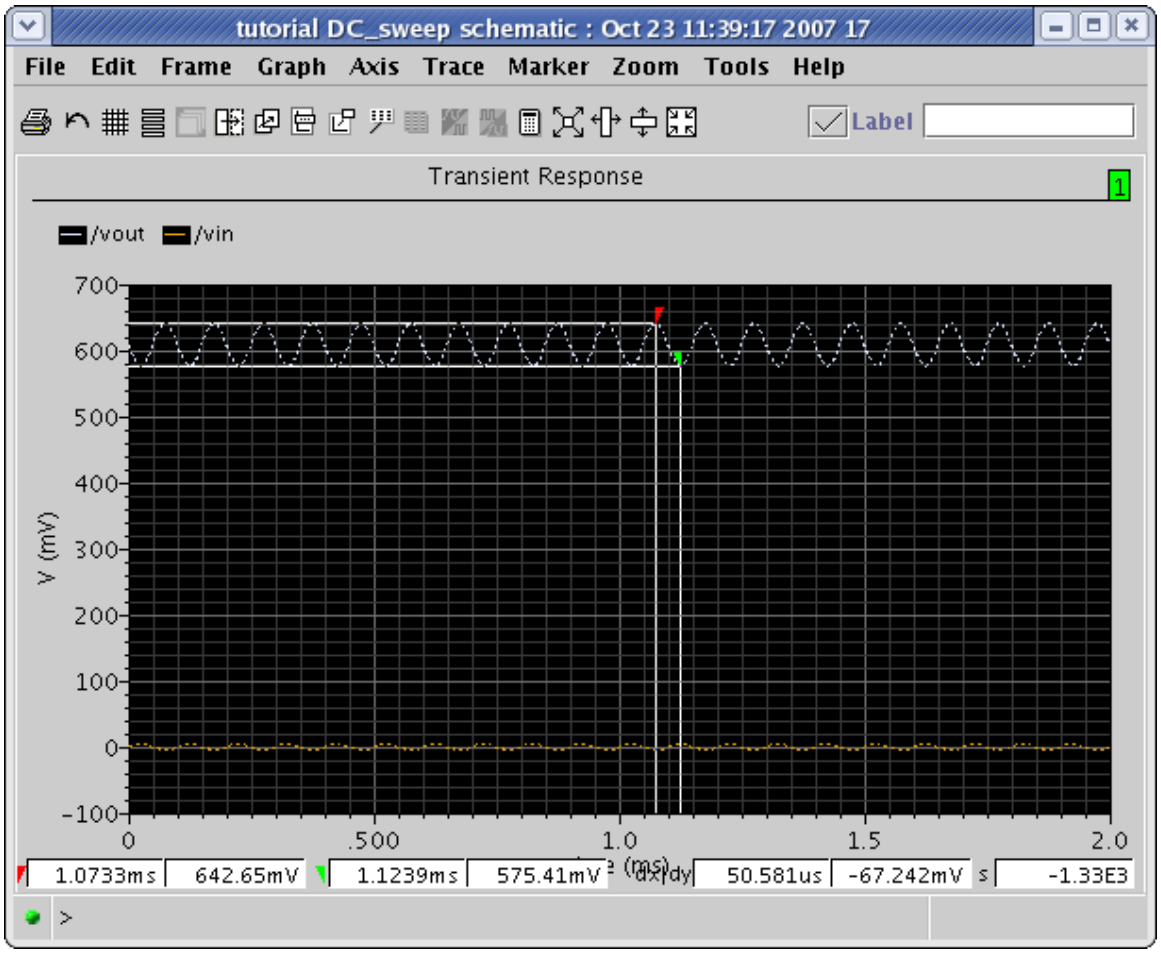
In the simulation setup, we now choose a transient analysis:



We should also add back the input and output waves to the `Outputs` window, and deactivate the unneeded Analyses and Outputs. The simulator should now look like this:



The simulation output is here.



I have marked off the peak-peak output, which is 67mV. The input peak-peak voltage is 10mV (5mV peak), so the gain magnitude is 6.7, which corresponds to the 16.6dB seen in the AC analysis.