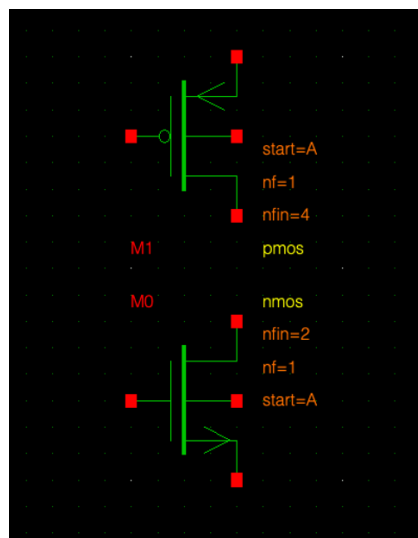
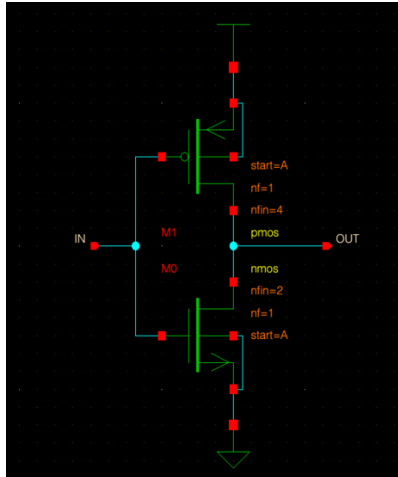


Glade training part 4 - lab

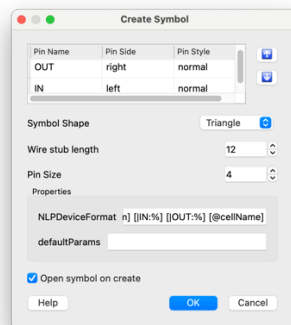
- For this lab, install the Xyce simulator available from <https://xyce.sandia.gov/downloads/sign-in.html>
- Start in the training directory
 - `cd glade_training/Part4`
- Make sure your PYTHONPATH includes the location of the FreePDK15 directory:
 - `PYTHONPATH=.:glade_training/Part4/FreePDK15:$GLADE_HOME:$PYTHONPATH`
- Open Glade from the cmd line or icon
- Open the library 'XyceLib'
- Open the library 'FreePDK15'
- Create a cellView in library FreePDK15 called 'inv' 'schematic'
- Place an instance 'pmos' 'symbol' from the FreePDK15 library in your schematic. Set the nfin property to 4 and add a property 'model' with value 'pfet'. Then place an instance of 'nmos' 'symbol' below it, adding the property 'model' with value 'nfet':



- Place an instance 'vdd' 'symbol' from library 'basic' above the pmos device, and an instance of 'gnd' 'symbol' below the nmos device. These symbols give a connection to the global nets 'vdd!' and 'gnd!'.
- Next place a pin 'IN' to the left and a pin 'OUT' to the right.
- Use the Create Wire command ('w' bindkey) to wire up the pins. You should end up with something like this:



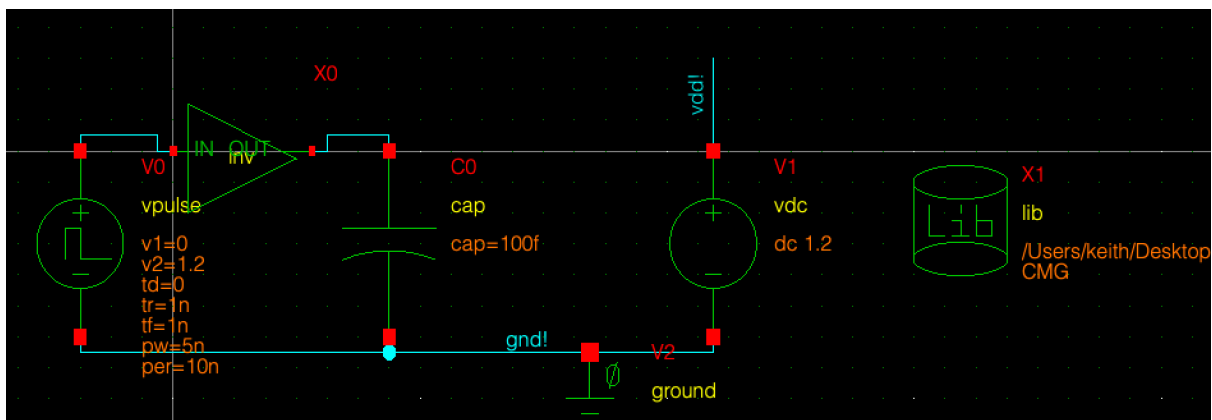
- Run File->Check&Save and make sure you don't have any errors.
- Run Create->Symbol and set the shape to 'triangle'.



You should get a symbol of an inverter, with a pin IN on the left and a pin OUT on the right. Don't worry if it's not perfect, you can fix that later.

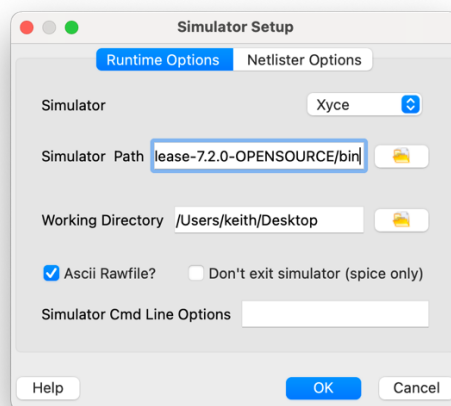
- Now create a cellView 'inv_test' 'schematic' in the FreePDK15 library.
- In it, create an instance 'inv' 'symbol' that you just created.
- Create an instance of XyceLib vpulse and cap.

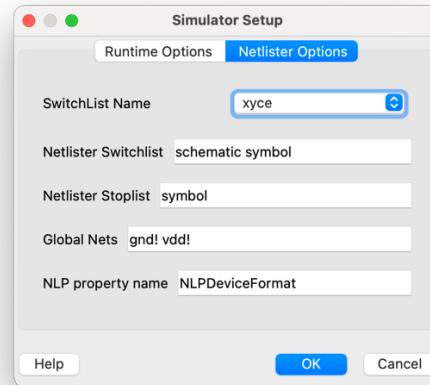
- Set the properties of the vpulse instance to v1=0, v2=1, td=0, tr=1n, tf=1n, pw=5n, per=10n
- Set the properties of the cap instance to cap = 0.1p
- Create an instance of 'vdc' and set the property dc to 1.
- Create an instance of 'lib' and set the path to your glade_training/FreePDK15/fet.lib file, and the section to CMG.
- Create an instance 'ground' and place it so it can attach to the ground net. This connects gnd! to the Spice global net 0.
- Wire it up and add labels for nets vdd! and gnd! as in the picture below:



Then File->Check&Save and make sure you get no errors.

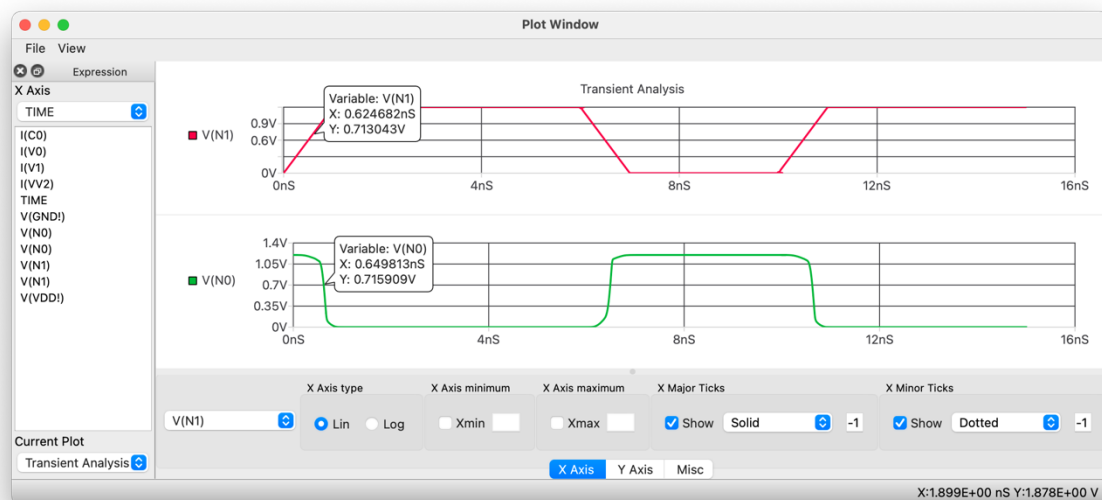
Open the simulate->setup dialog to set the simulator name and path:





The actual simulator path/name will depend where you installed it.

- Right click on the inv input and output nets and select 'add net voltage probe'. You should see probes added in the probe window.
- Now choose Simulate->Run Transient Analysis, set the initial timestep to 1n and the final time value to 20n. Click OK.
- You should see something like this:

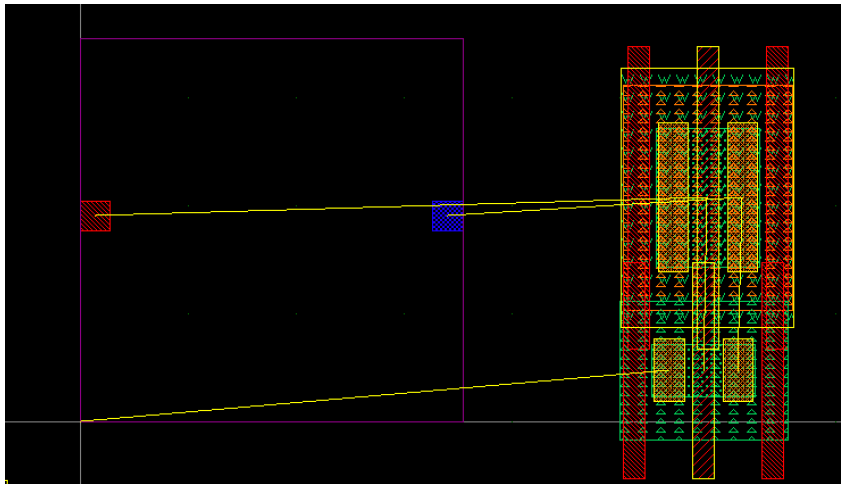


Lastly, let's try creating layout for our inverter.

- Open the 'inv' 'schematic' and select Layout->Gen Layout. Choose a pin layer for the input pin e.g. GATEA. Choose the side of the OUT pin to Right. Click OK.

- Use the Selection Options dialog to display connectivity (shift+E bindkey).

You should see something like this:



- Note how your input pin is on the GATEA layer to the left as specified.
- You can now move devices and see the connectivity as you move them.

This concludes the lab session for Part 4.