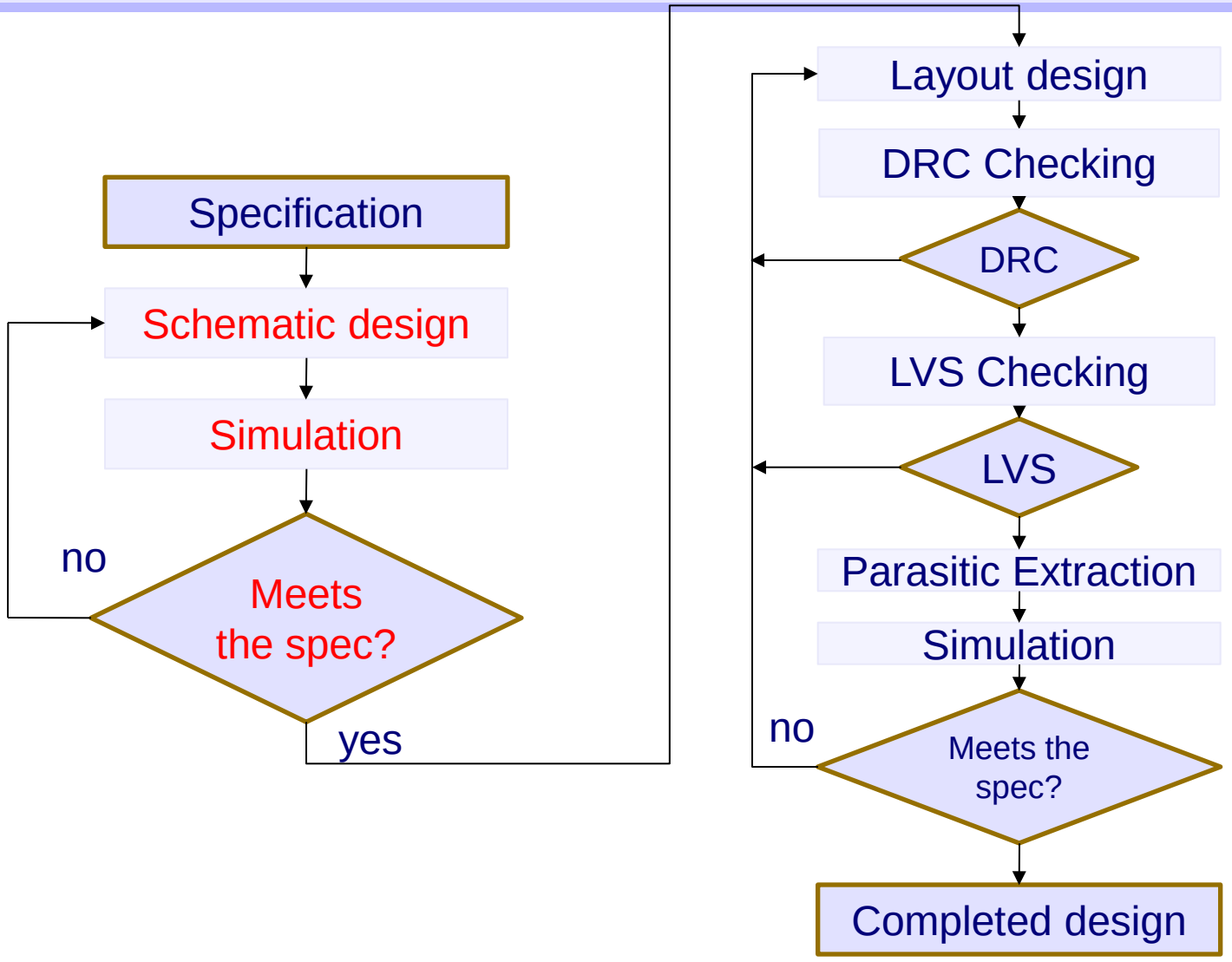
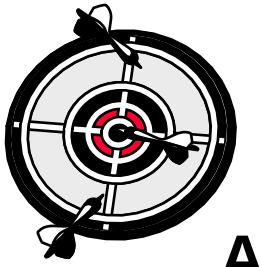
Symbols & Simulation

Dr Basel Halak

Custom Design Flow



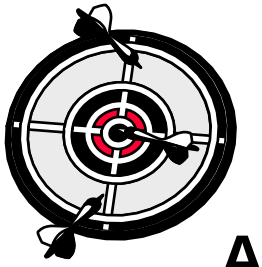
Learning Outcomes



After completing this unit, you should be able to:

- **Create a symbol from a schematic**
- **Set up the simulation environment**
- **Simulate your design**
- **Plot the results**
- **Use parameter passing and design variables**
- **Use parametric analysis**

Learning Outcomes



After completing this unit, you should be able to:

- **Create a symbol from a schematic**
- **Set up the simulation environment**
- **Simulate your design**
- **Plot the results**
- **Use parameter passing and design variables**
- **Use parametric analysis**

Set up the current working directory

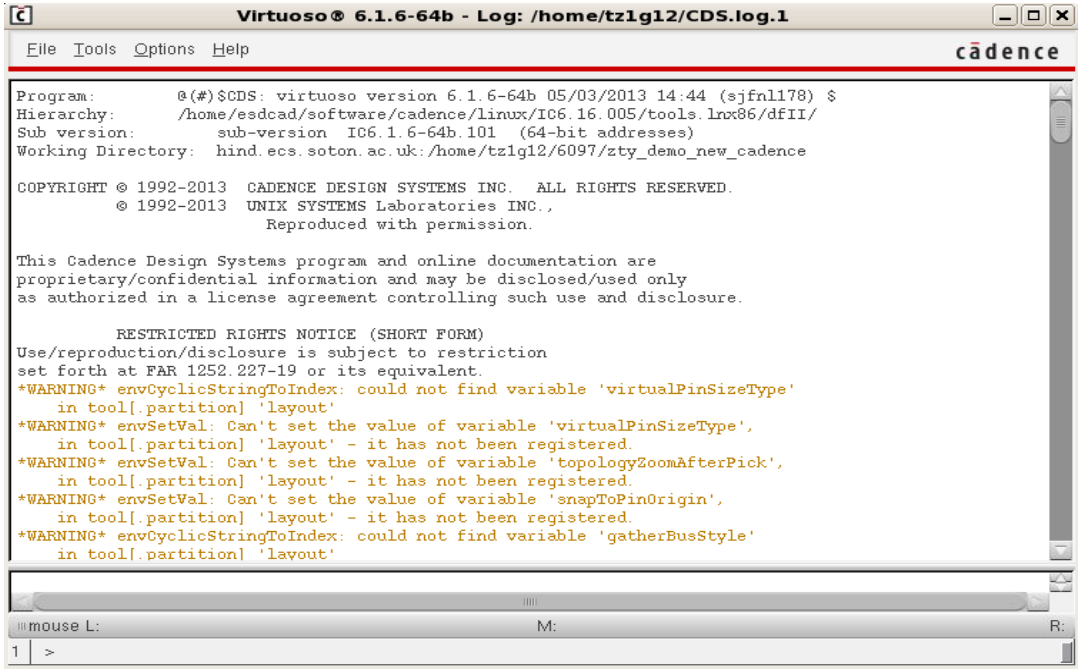
- **The first step is to set up the local working directory for the AMS 0.35μm process.**
- **open a new terminal and type the following:**

  tcsh

  source /opt/esdcad/scripts/ams_v400_tcshrc

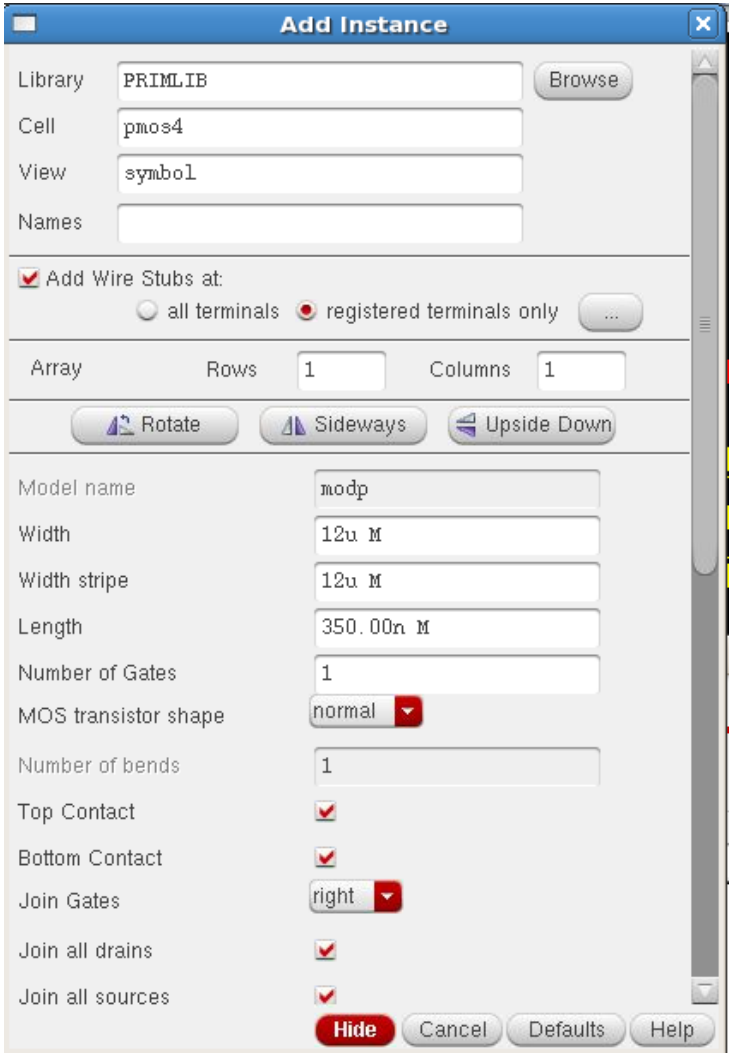
Run Cadence Design Manager

- **This will bring up the Cadence Design Manager:**



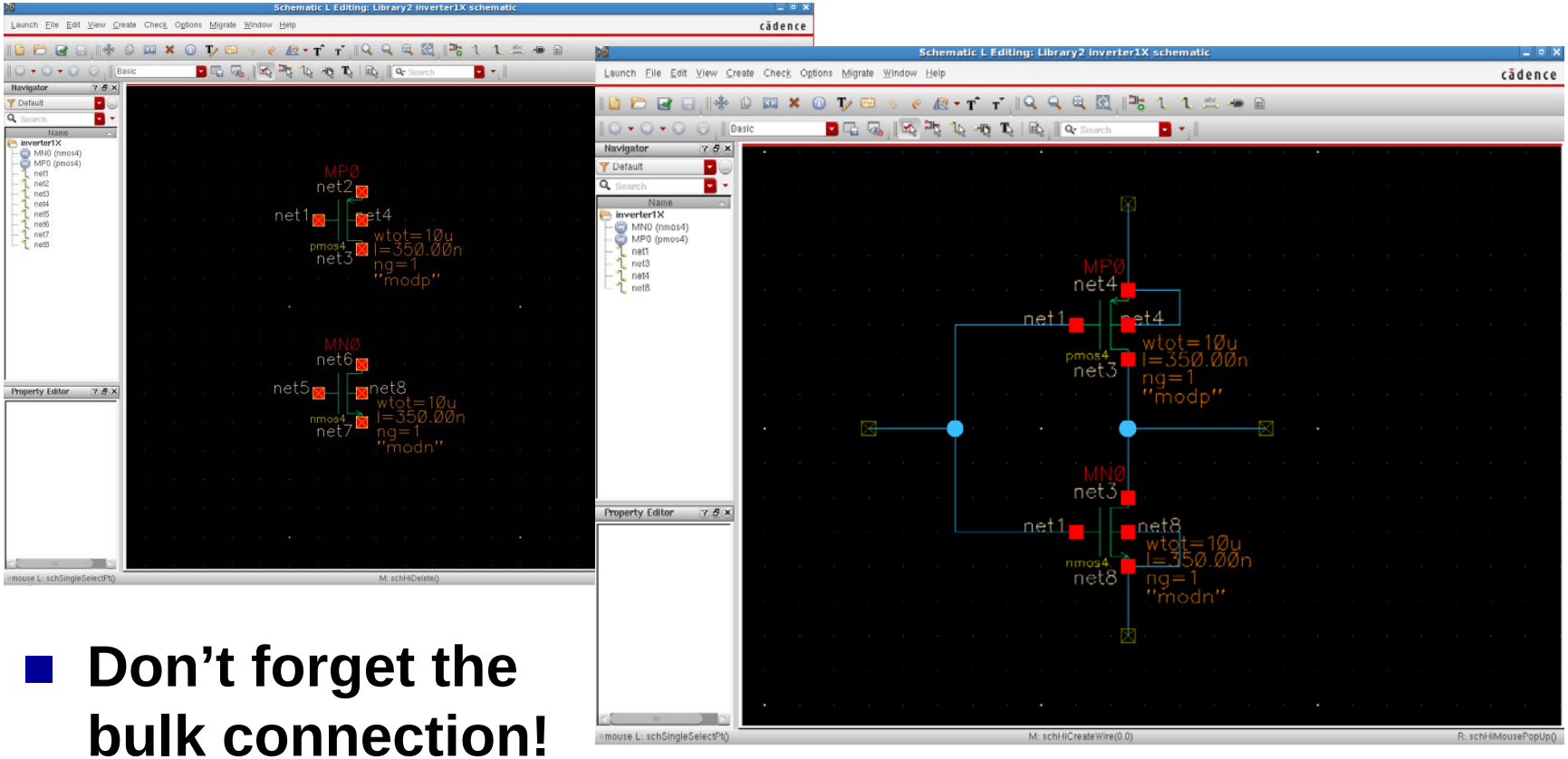
Create an inverter schematic

- Add a PMOS and NMOS transistor to a schematic from the PRIMLIB library with the following dimensions:
  - NMOS 4u/0.35u
  - PMOS 12u/0.35u



Create an inverter schematic

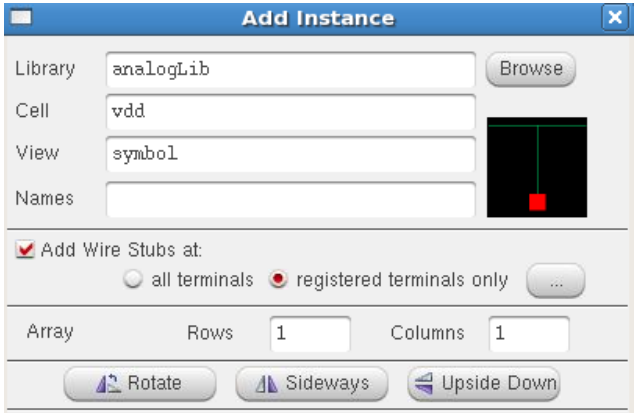
- Then wire them up using the add wire command

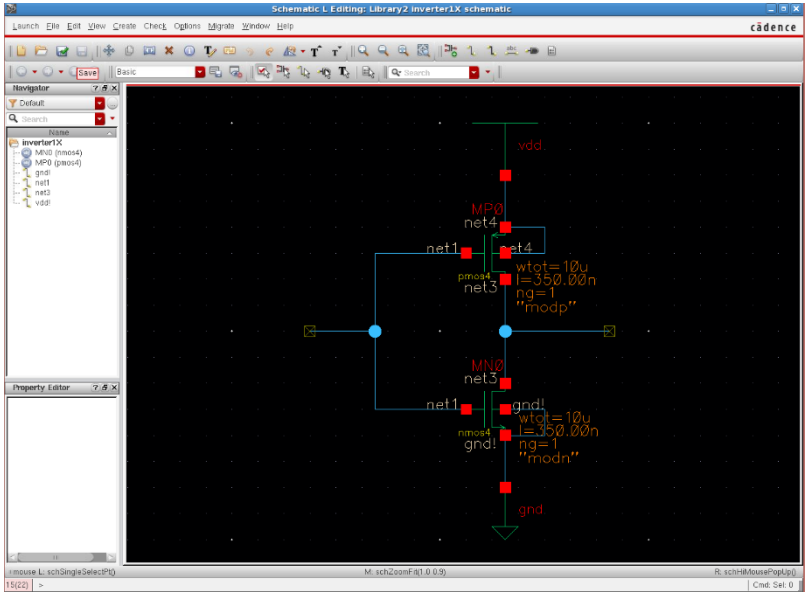


- Don’t forget the bulk connection!

Connecting things up…

- In Cadence we can add pins for inputs and outputs.
- For Vdd and Ground we use globals.
- Cadence includes symbols which are inherently global – e.g. vdd and gnd.

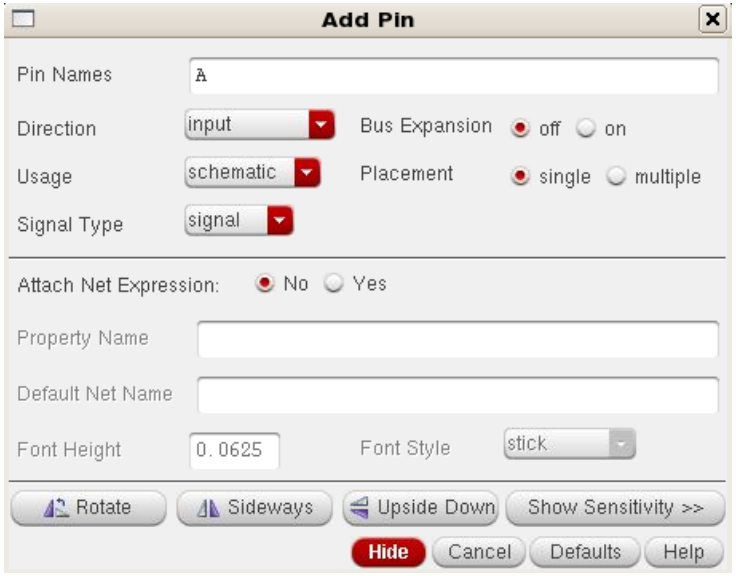


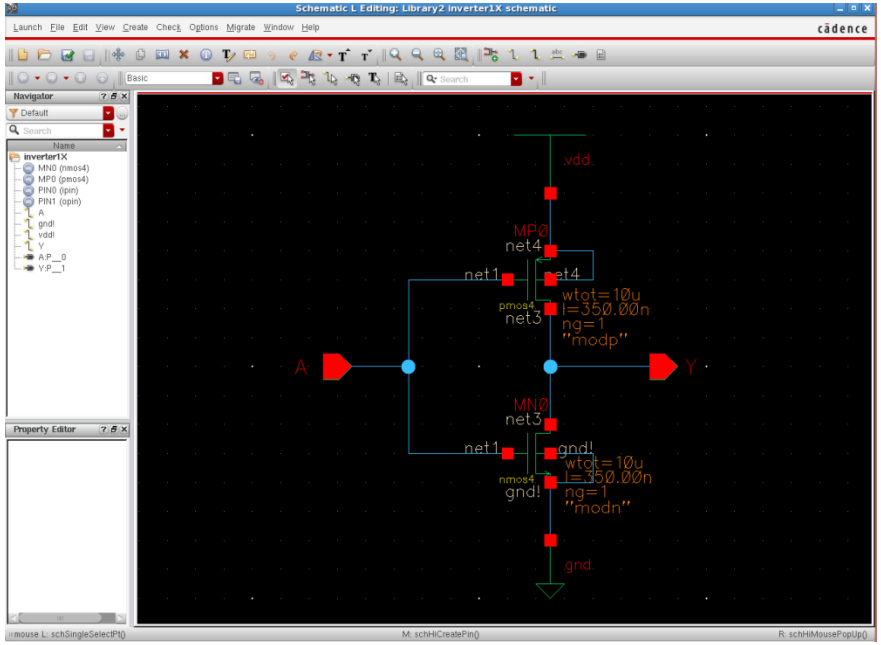


- Add these to your schematic

Adding Pins

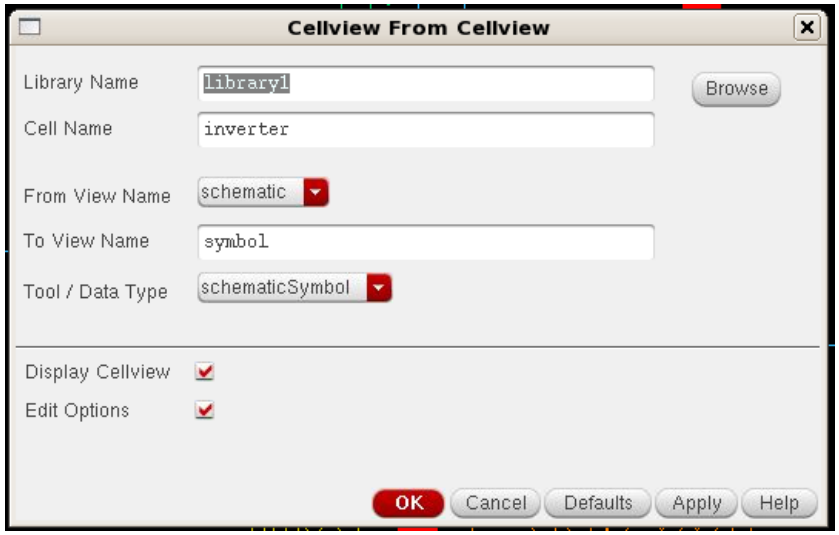
- Specific pins (inputs and outputs) are added using the Create -> Pins command
- Remember to specify the direction

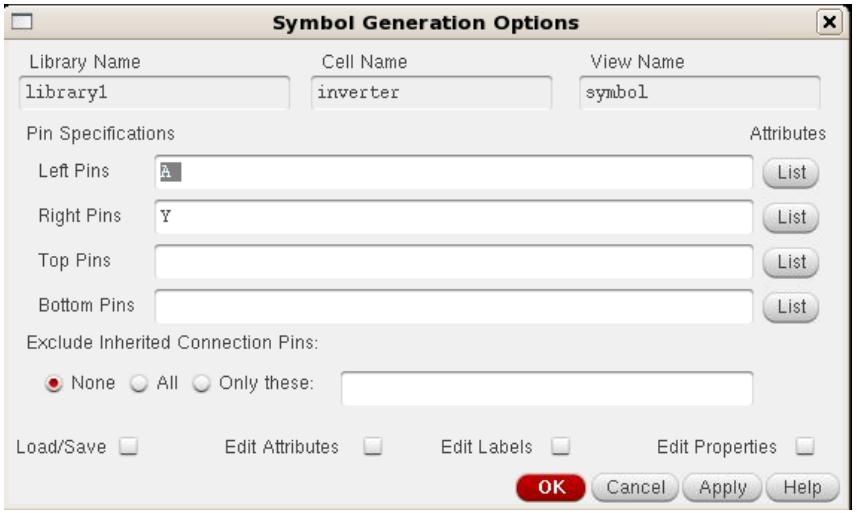




Creating a symbol from a schematic

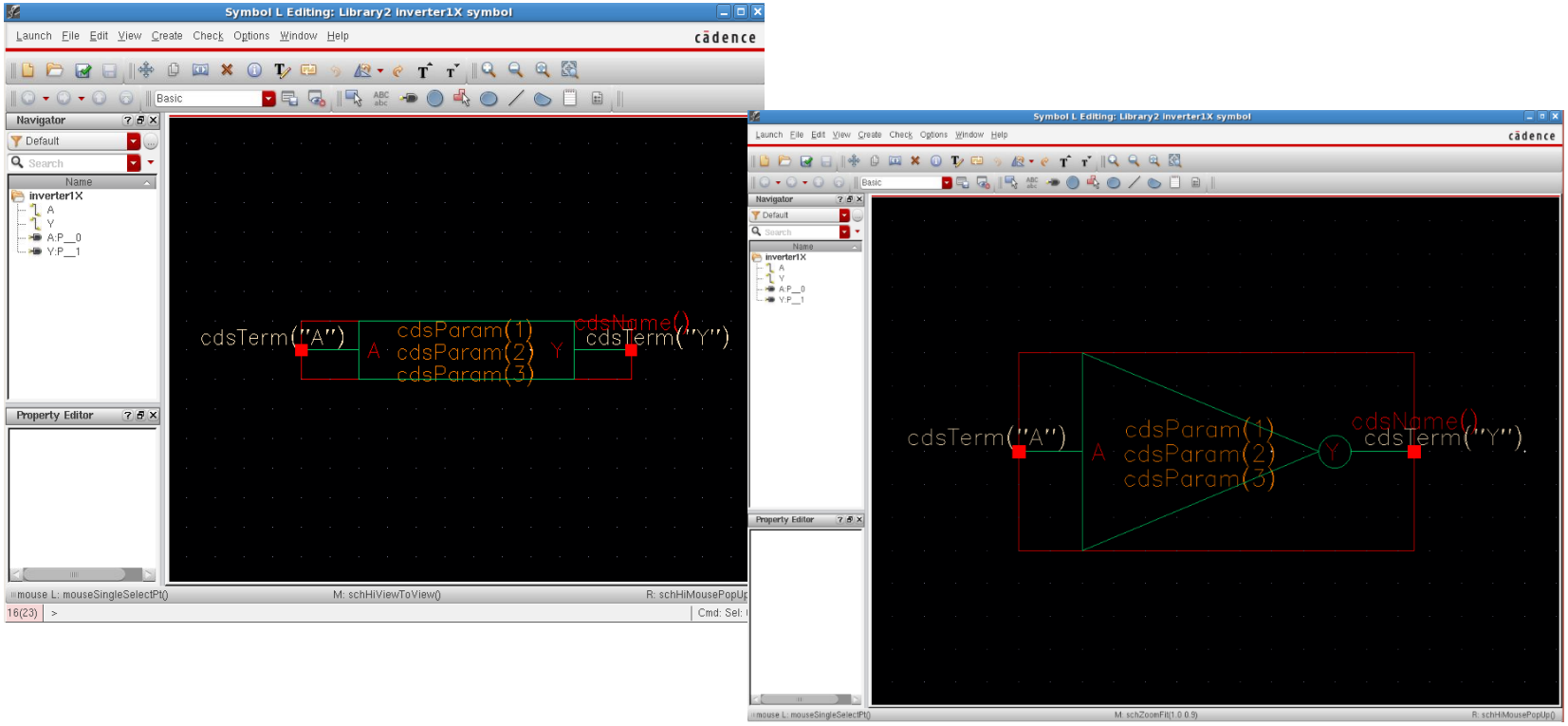
- To create a symbol from a schematic then we can use the create cell view -> from cell view command in the Create menu





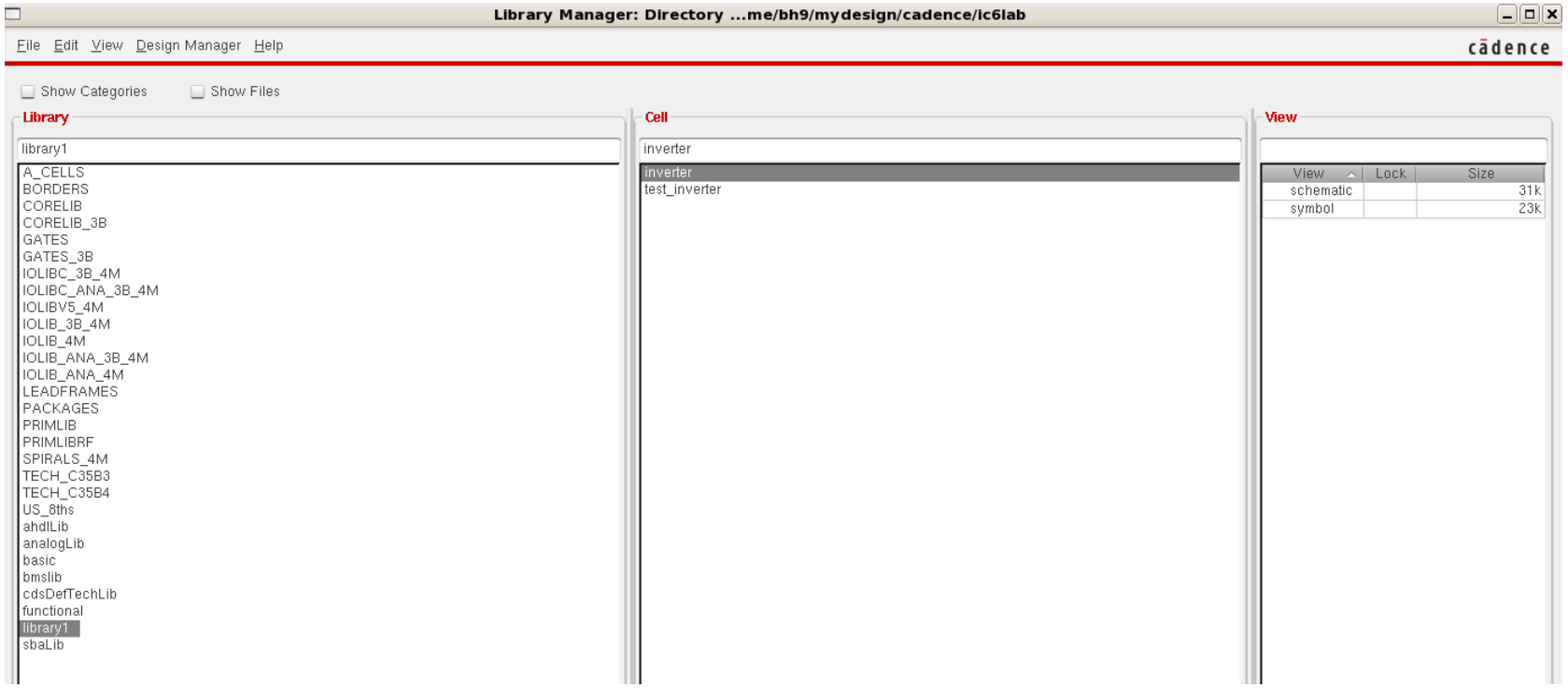
Symbol Creation

- The symbol will be automatically created as shown
- Use the graphics commands from Edit menu : Move, Delete, Stretch and add to draw the symbol



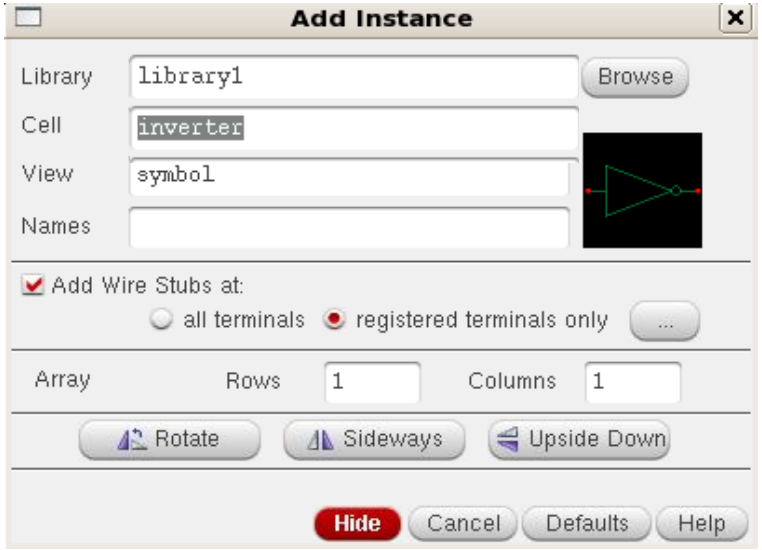
Symbol Creation

- Don’t forget to SAVE the symbol
- Close it and you will see the new cell view appear in library manager



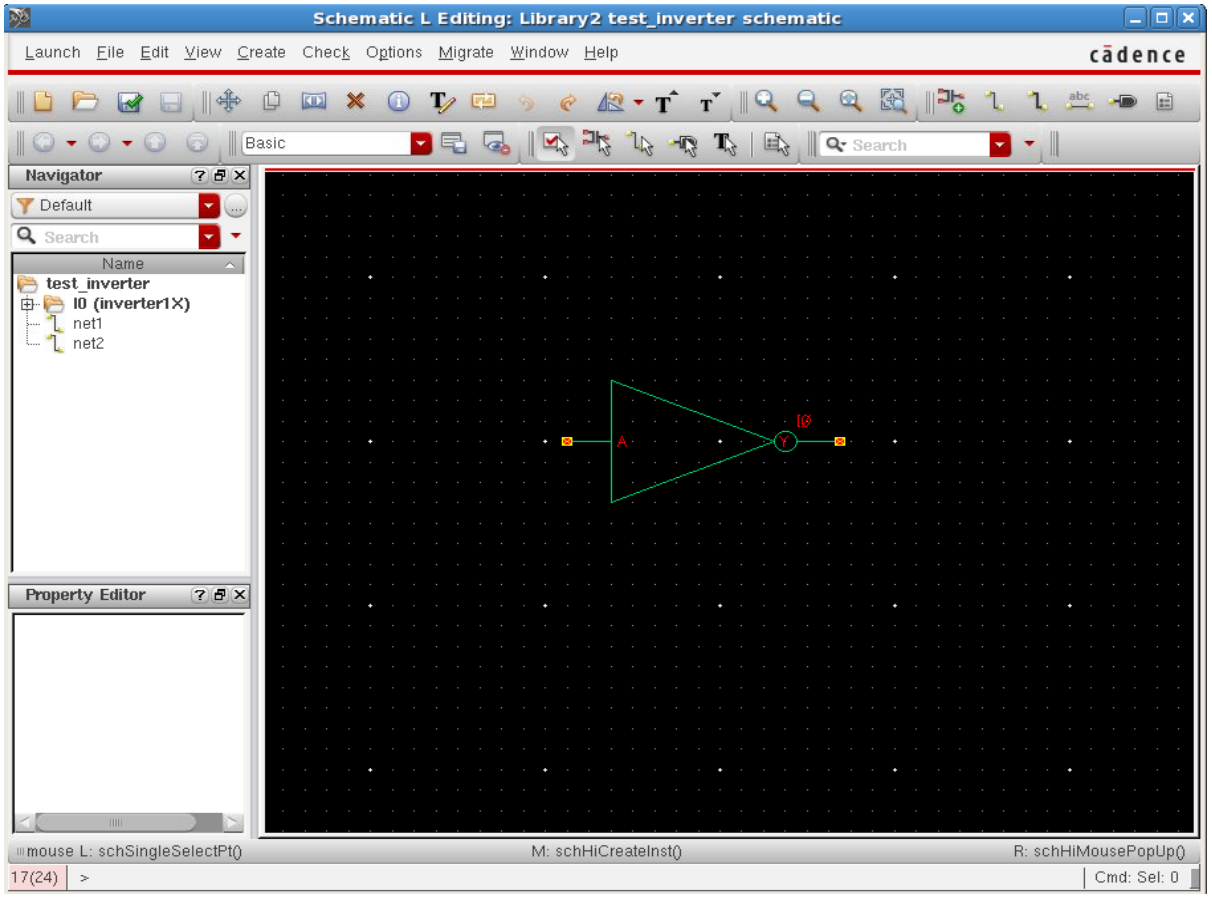
Using the symbol

- We can now use the new symbol in a hierarchical design or a test circuit
- From the library manager – create new cell view, schematic and call it ‘test_inverter’
- Use the Create-> instance command
- Place your new symbol



Placing the symbol on a schematic

- **We get the schematic looking like this:**



Analog test bench requirements

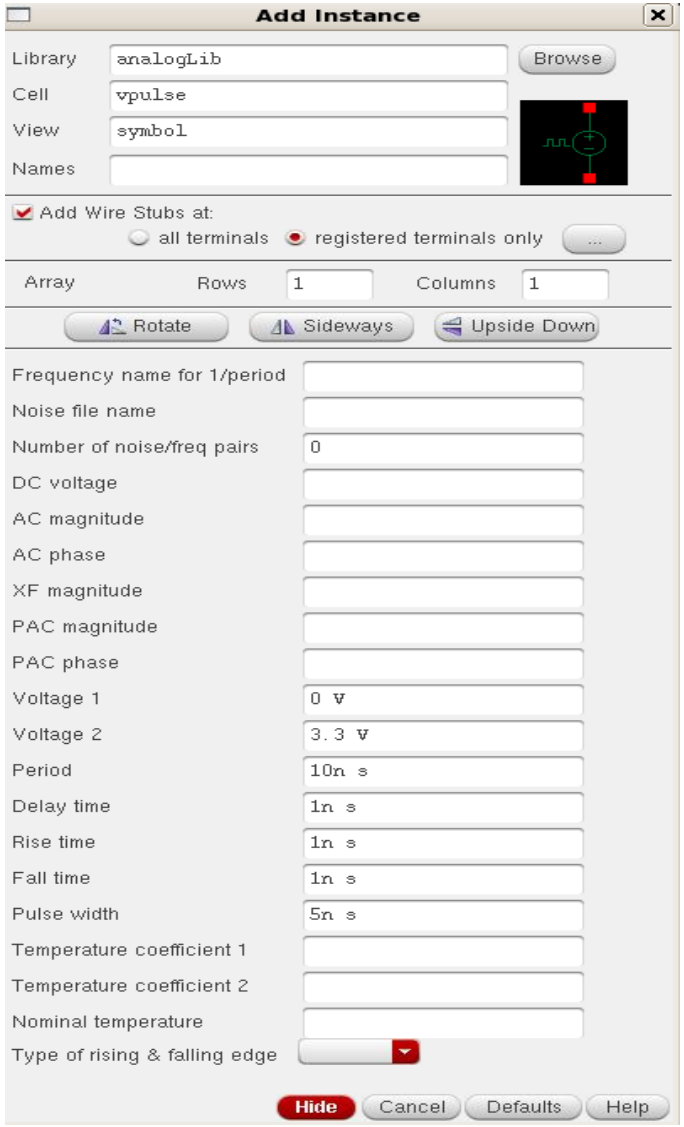
- **In an analogue test bench we need:**
- **Inputs**
  - sources (dc, transient, ac) depending on the analysis to be carried out
- **Outputs**
  - Measurement models
  - Loads
- **Supplies**
  - Vdd, Vpulse
  - Ground
  - gnd

test_inverter – test bench requirements

- **Inputs:**
  - pulse input source between 0 and 3.3V at about 100MHz
- **Outputs:**
  - capacitive load of about 1pF
- **Supplies:**
  - Vdd = 3.3V
- **Ground:**
  - gnd = 0V

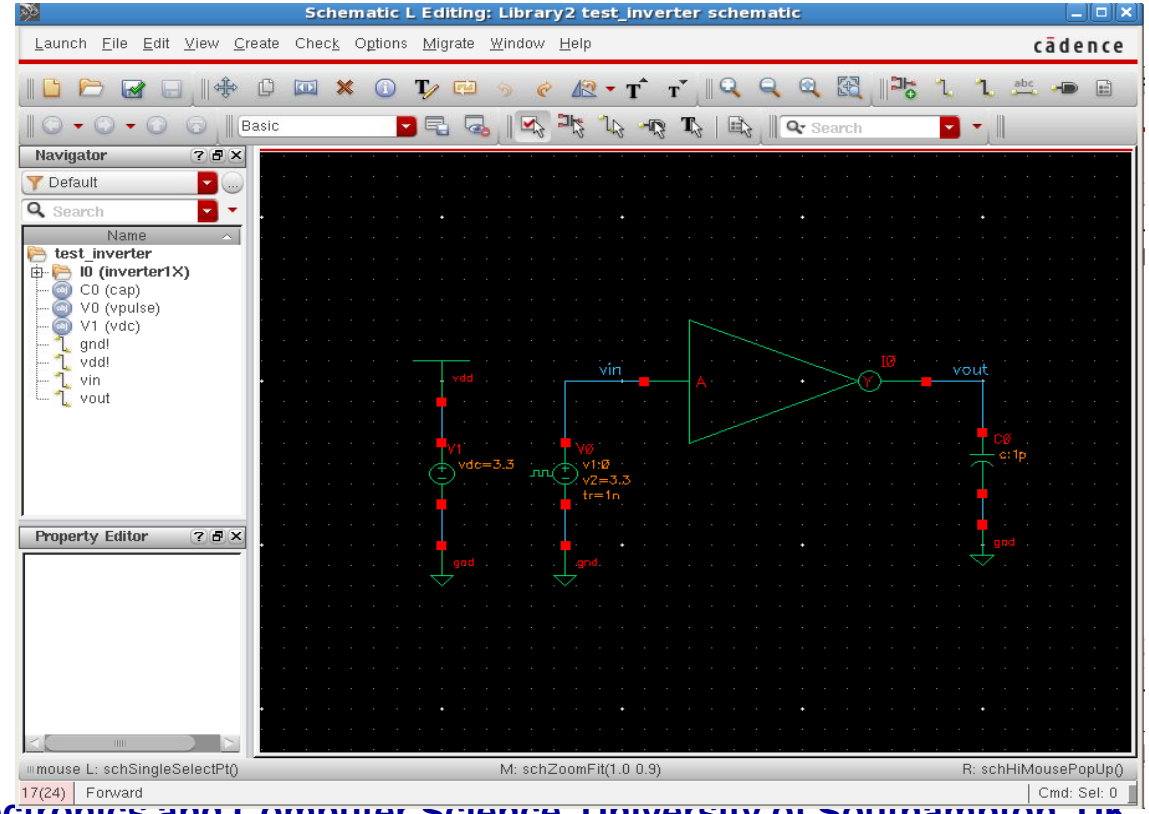
Specifying the vpulse instance

- **analogLib -> vpulse**
  - v1 = 0
  - v2 = 3.3
  - delay = 1n
  - rise = 1n
  - fall = 1n
  - pulse width = 5n
  - period = 10n

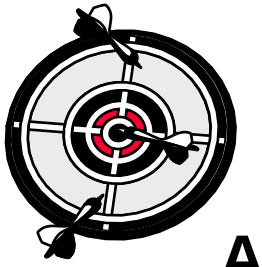


Complete the test circuit

- Wire up the source and load
- Give the wires meaningful names: you can name a wire by selecting it then write click and selcet Add name from the menu



Learning Outcomes



After completing this unit, you should be able to:

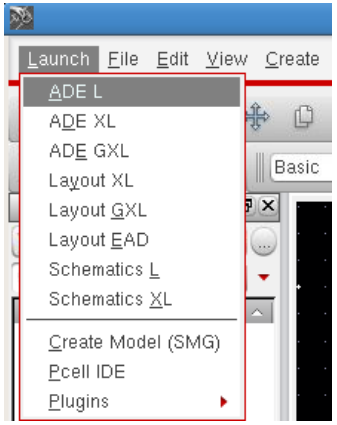
- **Create a symbol from a schematic**
- **Set up the simulation environment**
- **Simulate your design**
- **Plot the results**
- **Use parameter passing and design variables**
- **Use parametric analysis**

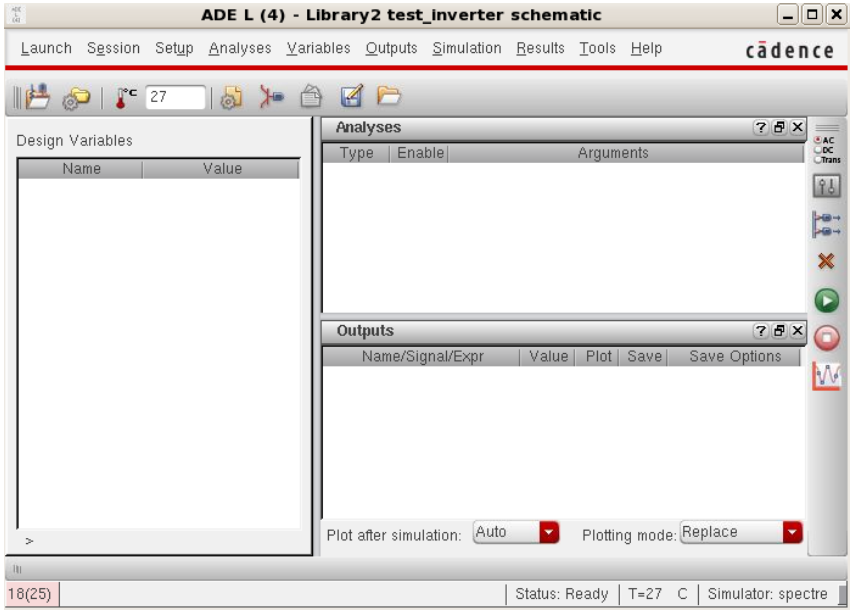
The Cadence simulator

- Now we want to simulate this circuit using the Cadence simulator: spectre
- The simulator environment in Cadence is called Analog Design Environment (ADE) and allows you to run simulations and plot results
- Although it looks pretty boring(!) it is actually very powerful

How to start Analog Design Environment

- Open your test circuit and go to Launch->ADE L.
- This starts the ADE interface

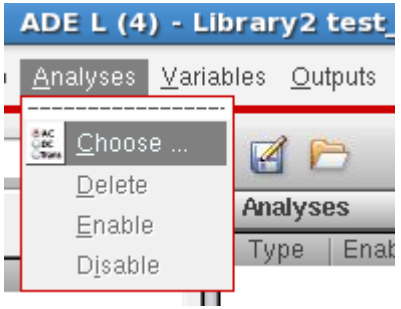


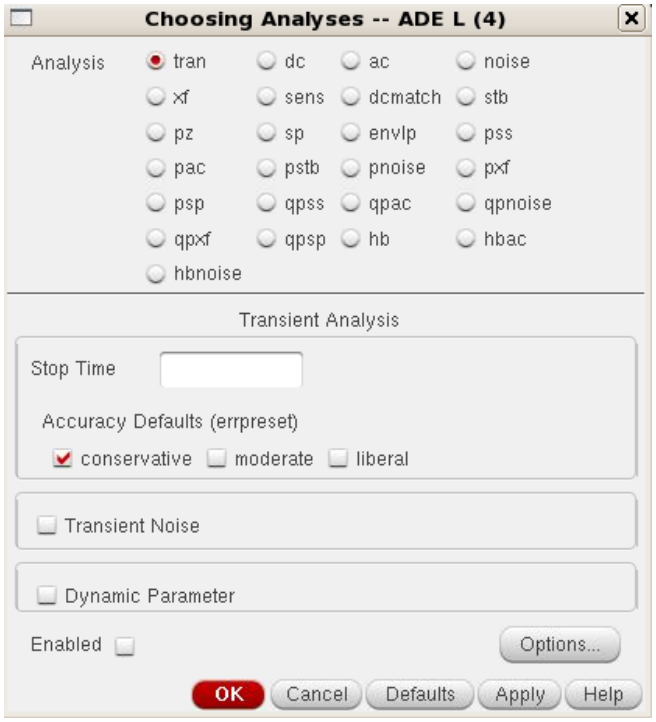


- You can see here we are using spectre simulator and that it has initialised the environment with our test circuit

Types of simulation

- We can choose from many different types of simulation
- Click the choose simulation button

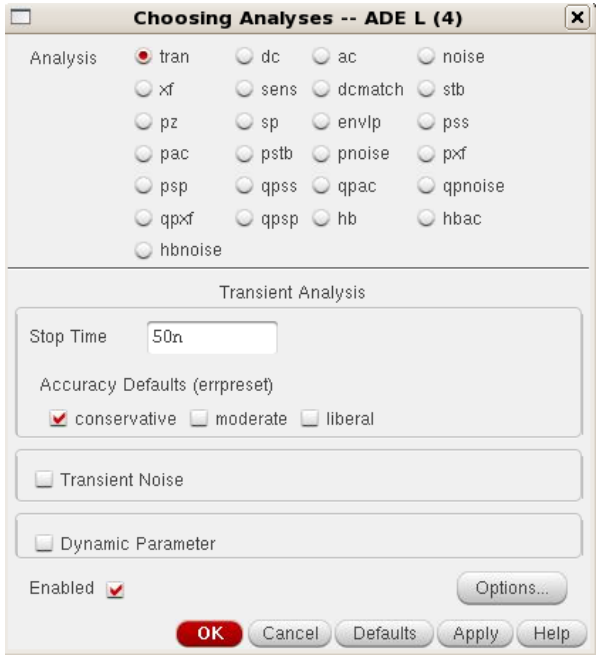




- First we will set up a transient (i.e. time domain) simulation

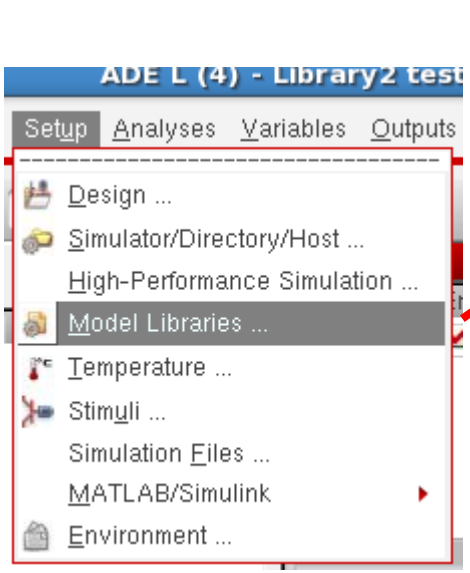
Transient simulation setup

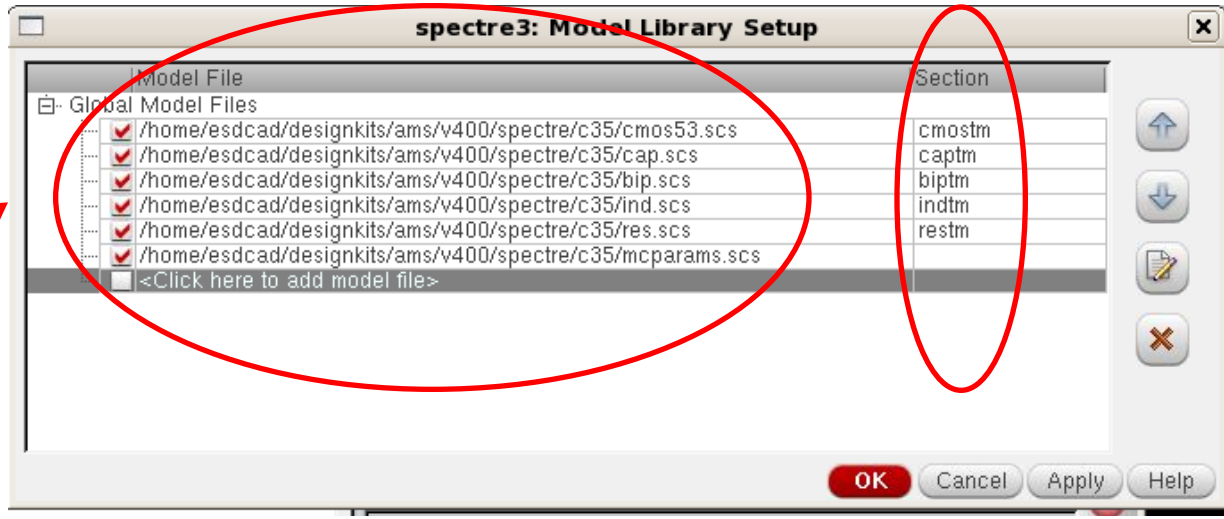
- **Different simulation accuracies are available**
  - Conservative is the most accurate but the slowest
- **In this example we will run the simulation for 5 cycles (50ns)**



Running the simulation

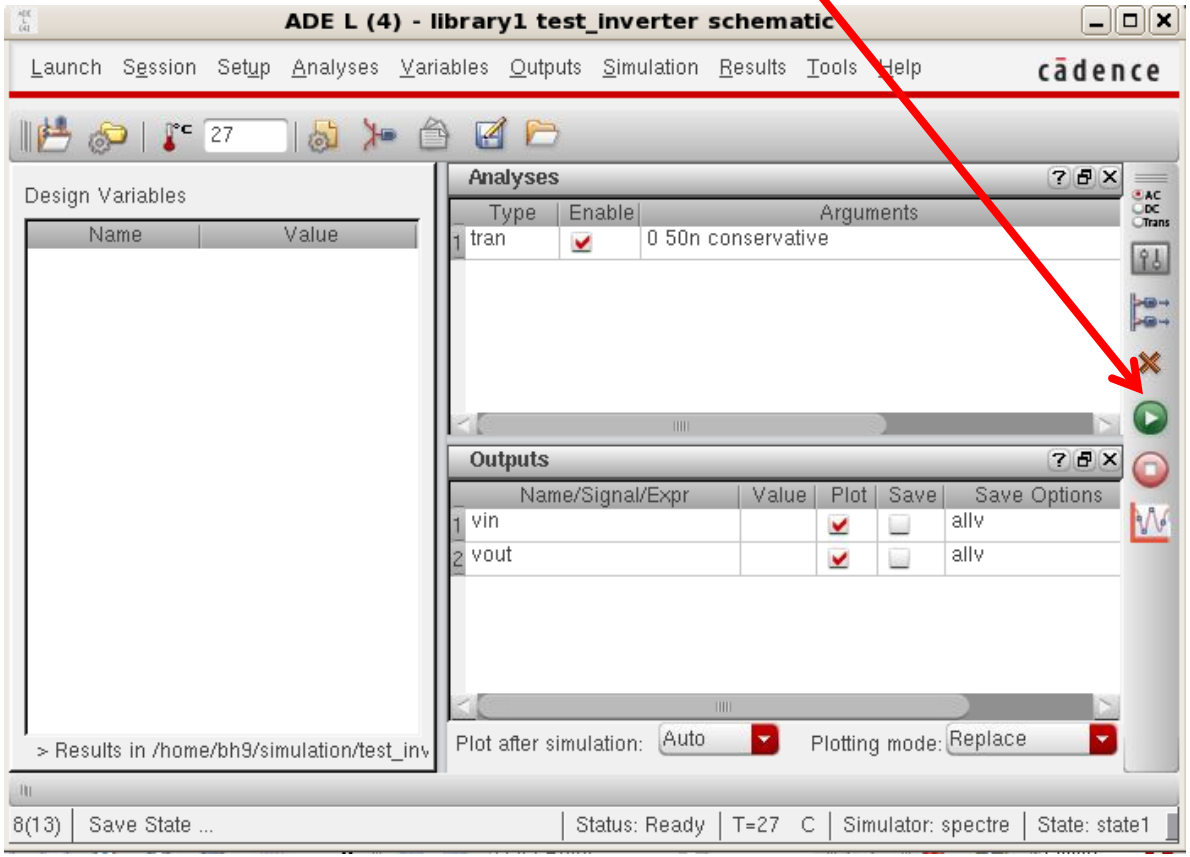
- **You need to add the Model Library before you run the simulation. Go to setup**





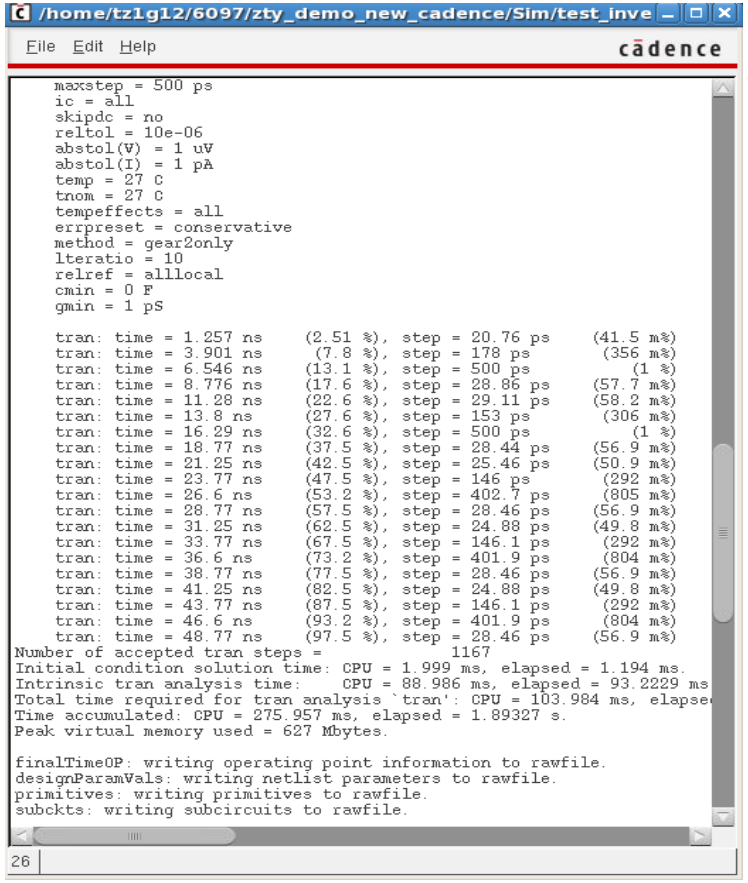
Running the simulation

- **Use the ‘Netlist and Run’ (green arrow) button to run the simulation**



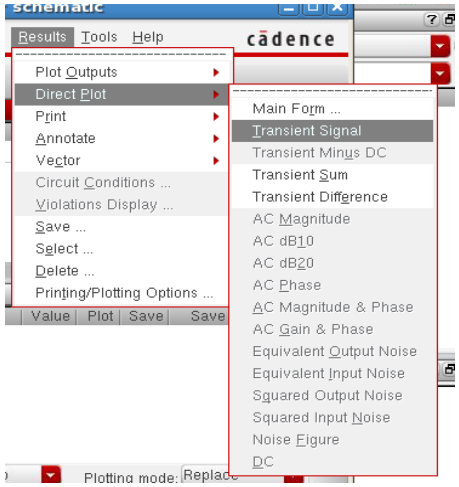
Running the simulation

- **A window pops up showing the progress of the simulation**



Plotting signals

- **There are many ways to plot a signal in ADE**
- **The easiest is with the Direct Plot option**
- **Choose Direct Plot -> Transient Signal**

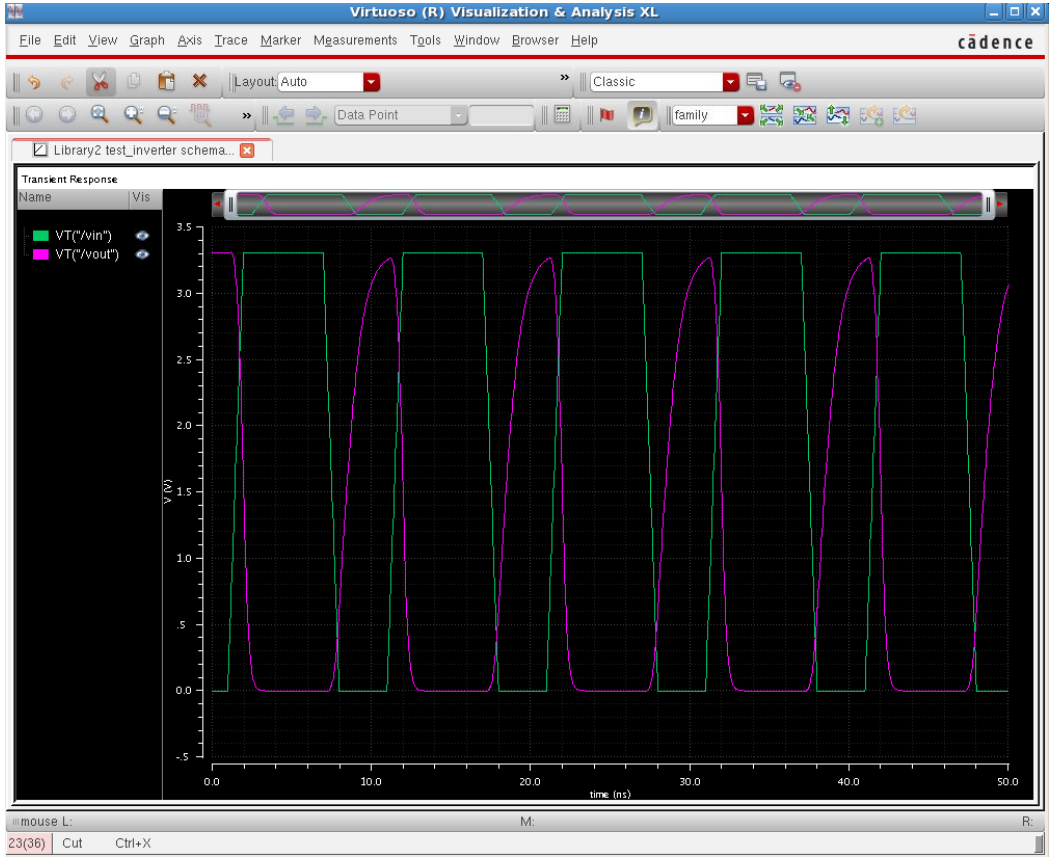


- **Go to the schematic and choose the signals**
  - Click a net for a voltage
  - Click a node for a current
- **Note the prompt at the bottom of the schematic**

> Select nodes or terminals, press <esc> to finish selection

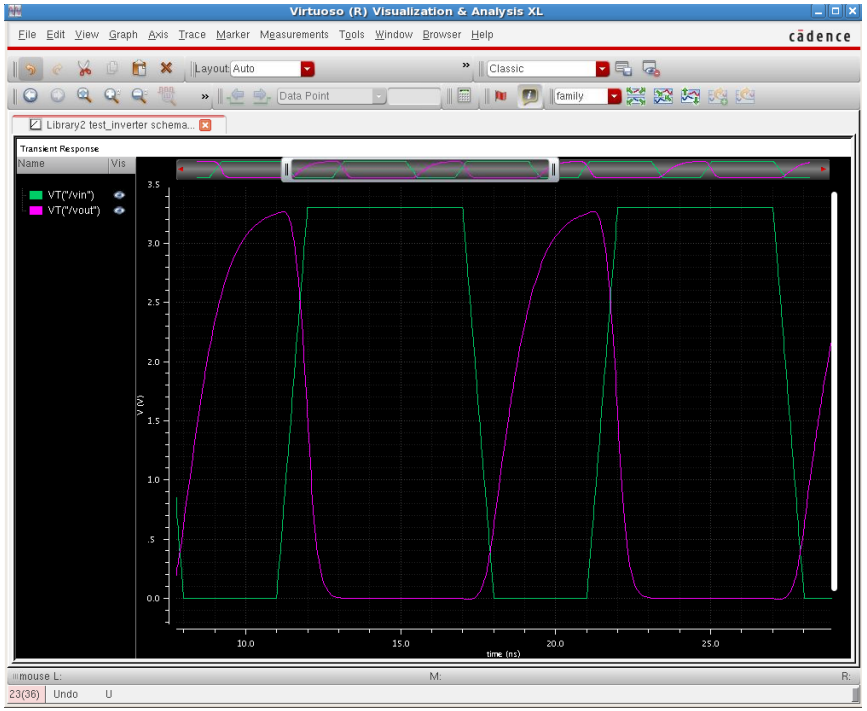
Plotting the signals

- **Select the input and output nets to plot the input and output voltages**

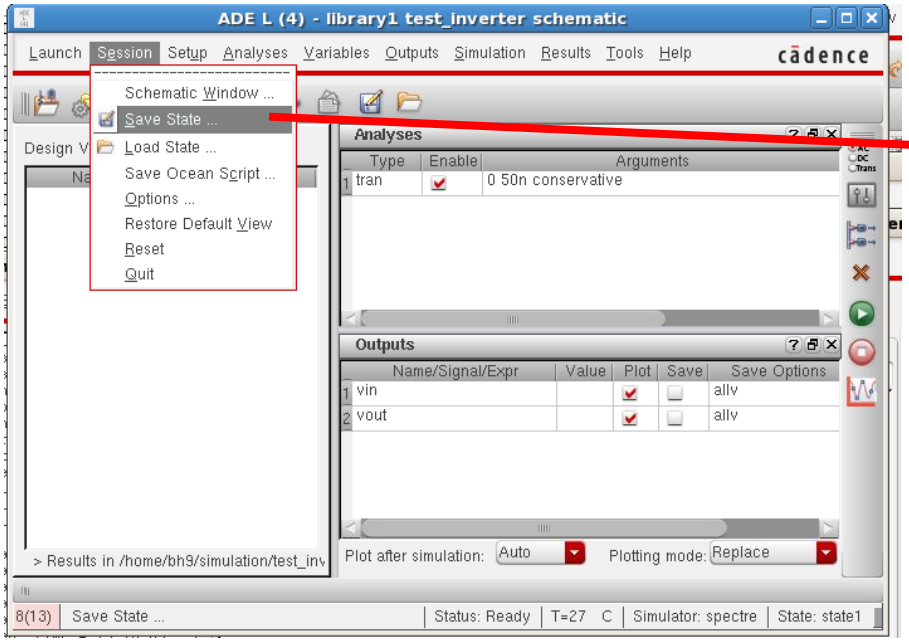


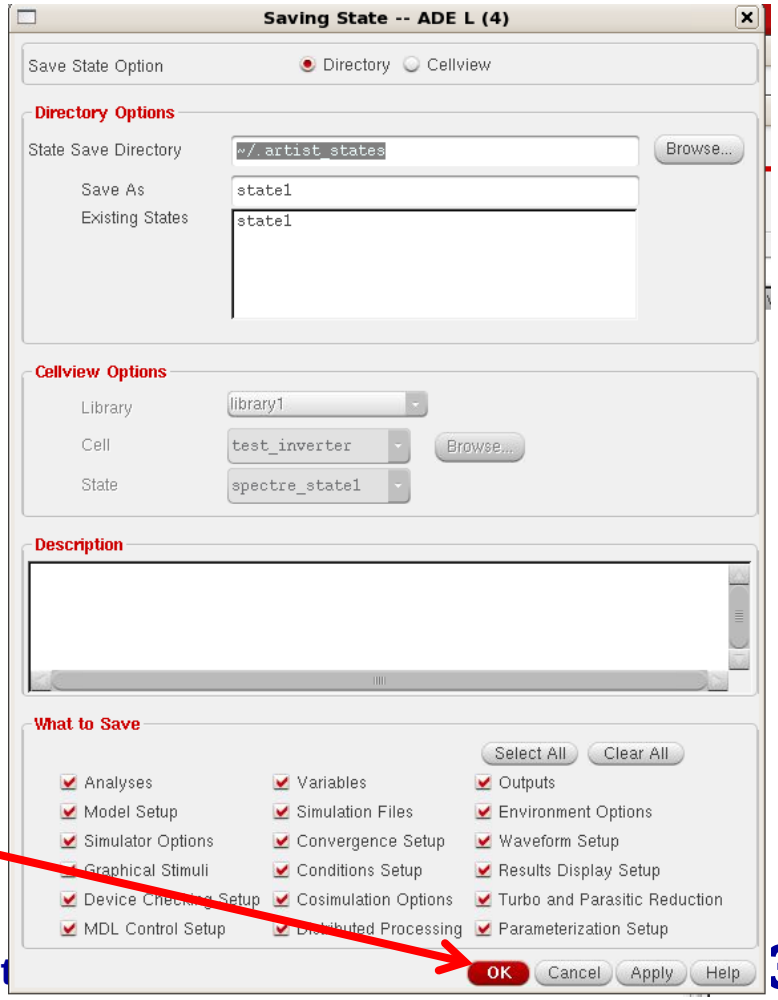
Customise the waveform display

- **Right click on a waveform to change properties**
  - Waveform name
  - Colour
- **Right click and drag to zoom in**
- **‘f’ to re-fit the graph**
- **‘H’ for horizontal marker, ‘V’ for vertical**
- **Notice slow rise time**



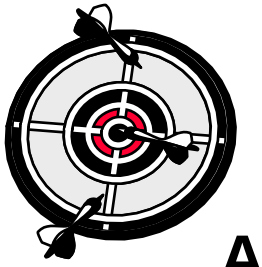
Save your simulation state





Click OK

Learning Outcomes



After completing this unit, you should be able to:

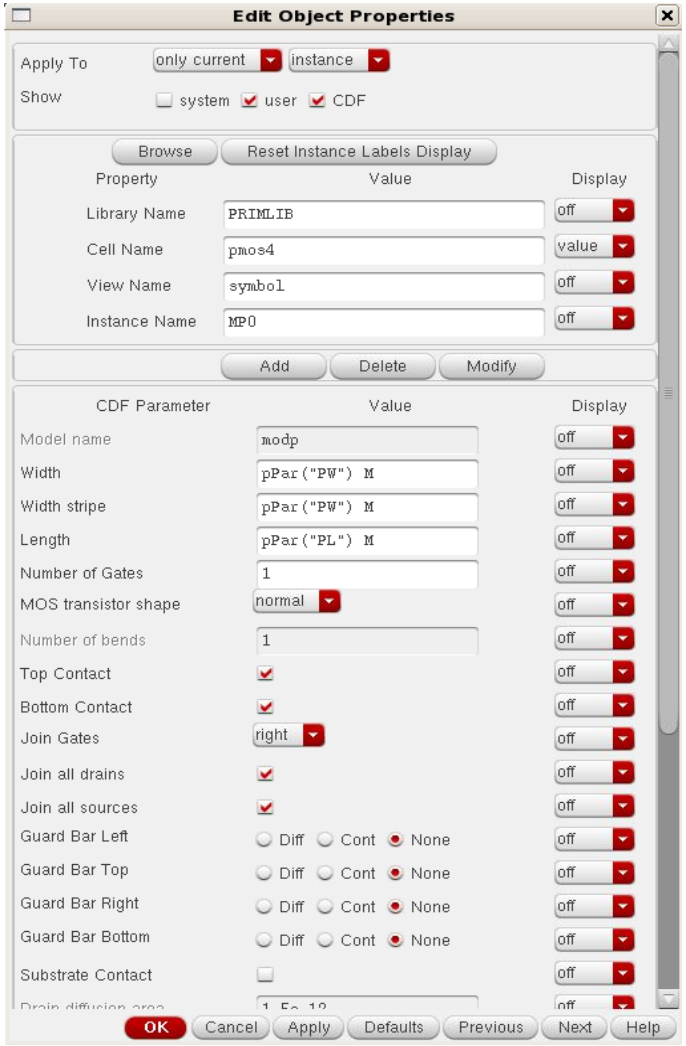
- **Create a symbol from a schematic**
- **Set up the simulation environment**
- **Simulate your design**
- **Plot the results**
- **Use parameter passing and design variables**
- **Use parametric analysis**

Lets optimise the rise time

- **We want to change the size of the PMOS and see the effect on the rise time**
- **What is the best way of doing this?**
- **Important points:**
  - Hierarchical parameter passing: from the schematic to the symbol
  - Design variables: schematic parameters which can be changed from ADE
  - Parametric simulations: run a set of simulations each with a different value for the design variable

Hierarchical parameter passing

- **Open the inverter schematic**
- **Change the PMOS width and length properties to:**
  - pPar(“PW”)
  - pPar(“PL”)
- **Change the NMOS width and length properties to:**
  - pPar(“NW”)
  - pPar(“NL”)

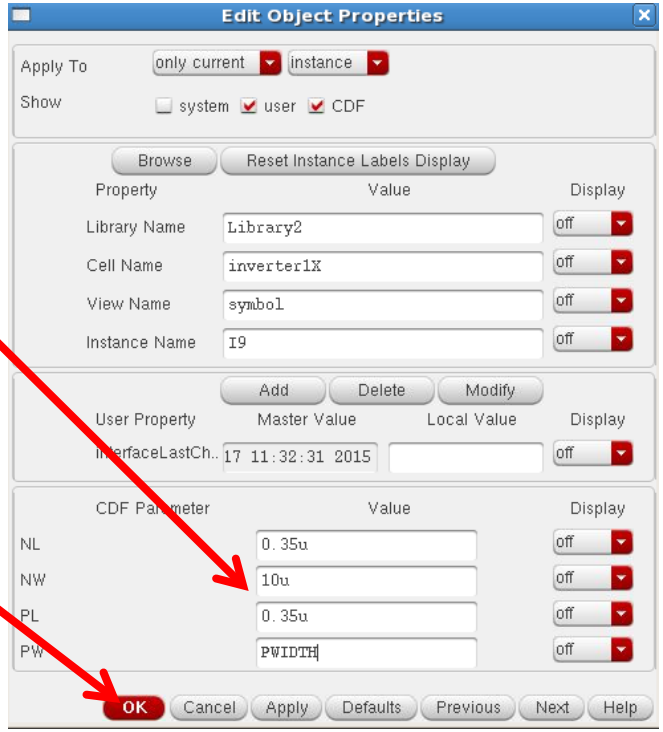


Hierarchical parameter passing

- Recreate the symbol
- This will create a symbol where you can set the widths and lengths for that symbol instance

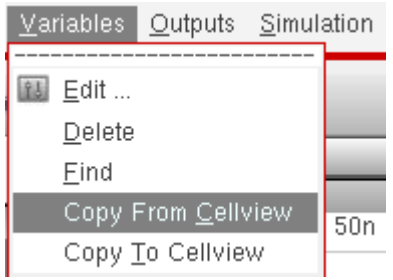
Hierarchical parameter passing

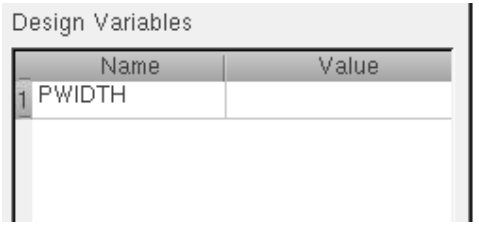
- Go back to the inverter test schematic
- Re-instantiate the new inverter symbol
- Change the properties of the new symbol as follows
  Put in ‘PWIDTH’ for the PMOS width and keep the others the same
- Check and save

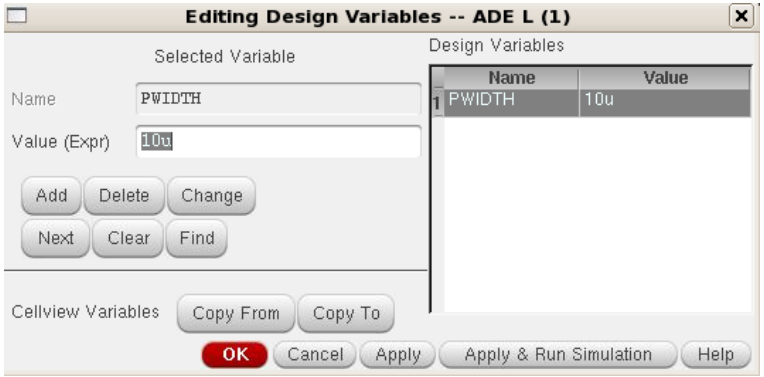


Importing design variables in to ADE

- Go back to the ADE form
- We need ADE to pick up our PWIDTH design variable from the schematic
- Go to Variables->Copy From Cellview
- Notice that PWIDTH appears in the design variables
- Now double click PWIDTH to bring up the design variable box
- Set PWDITH to 10um initially

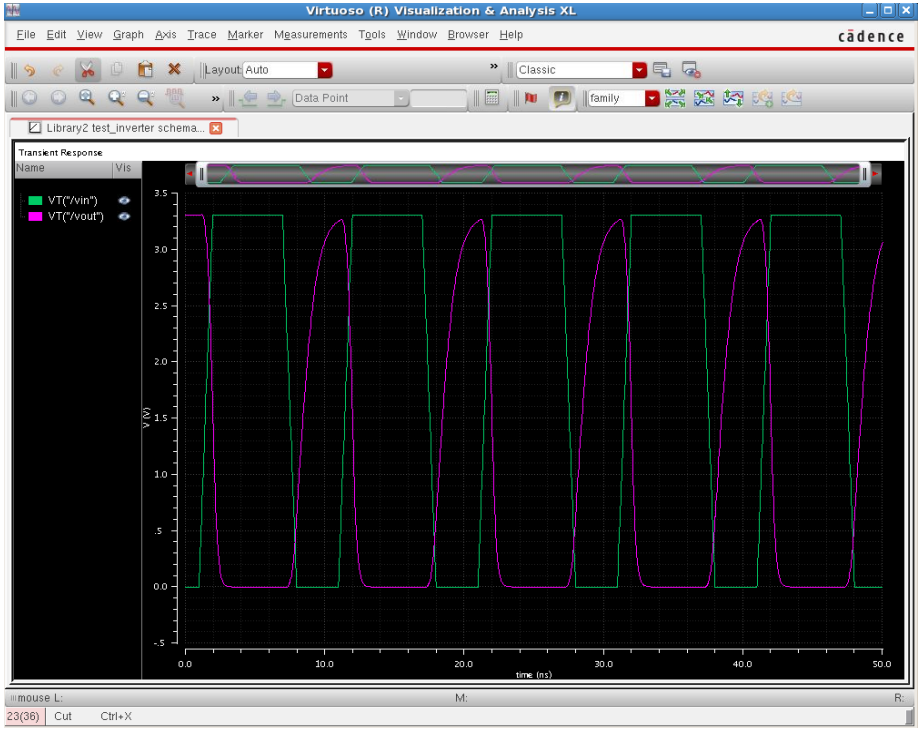






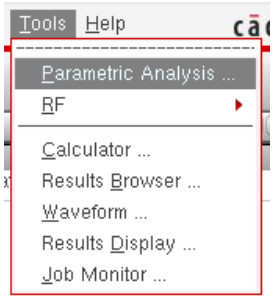
Importing design variables in to ADE

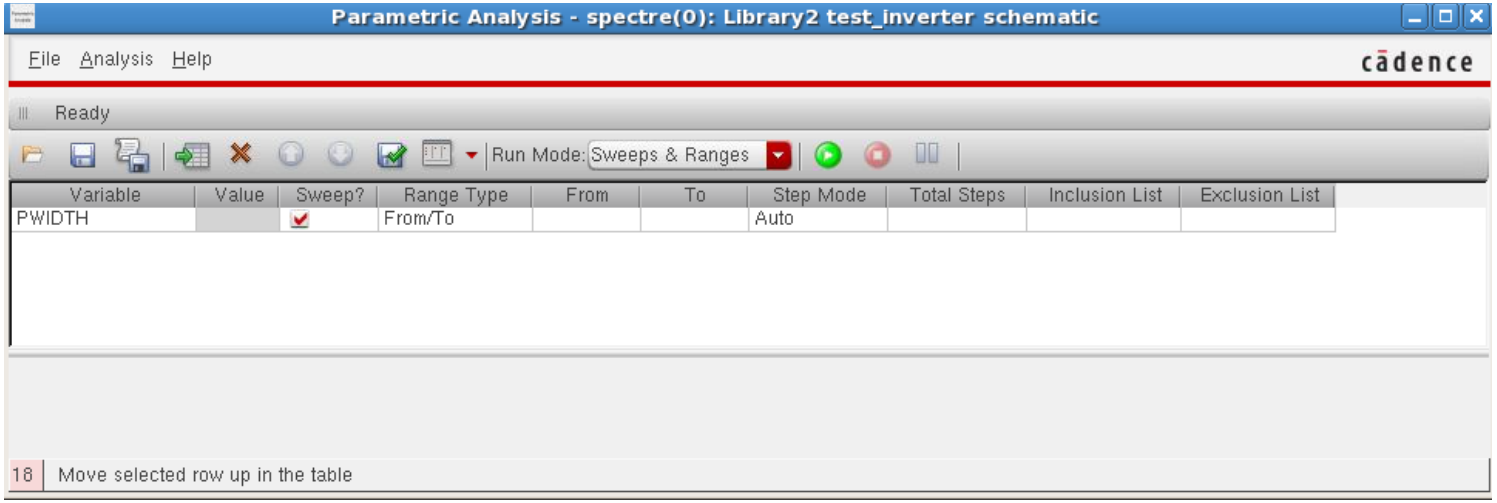
- Re-run the simulation, to check everything is ok
- Should get the same as before, but now we have the ADE set up and ready to do cool stuff



Running a parametric simulation

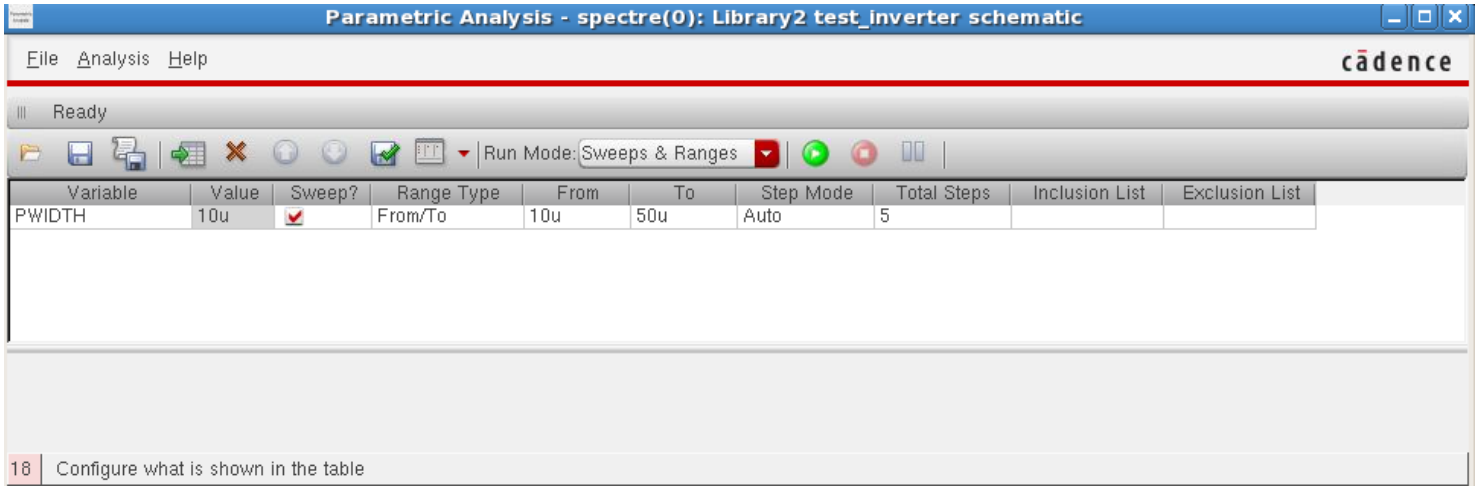
- **Parametric analysis allows you to run multiple simulations whilst varying design parameters**
- **We will run a parametric simulation during which PWIDTH will be varied**





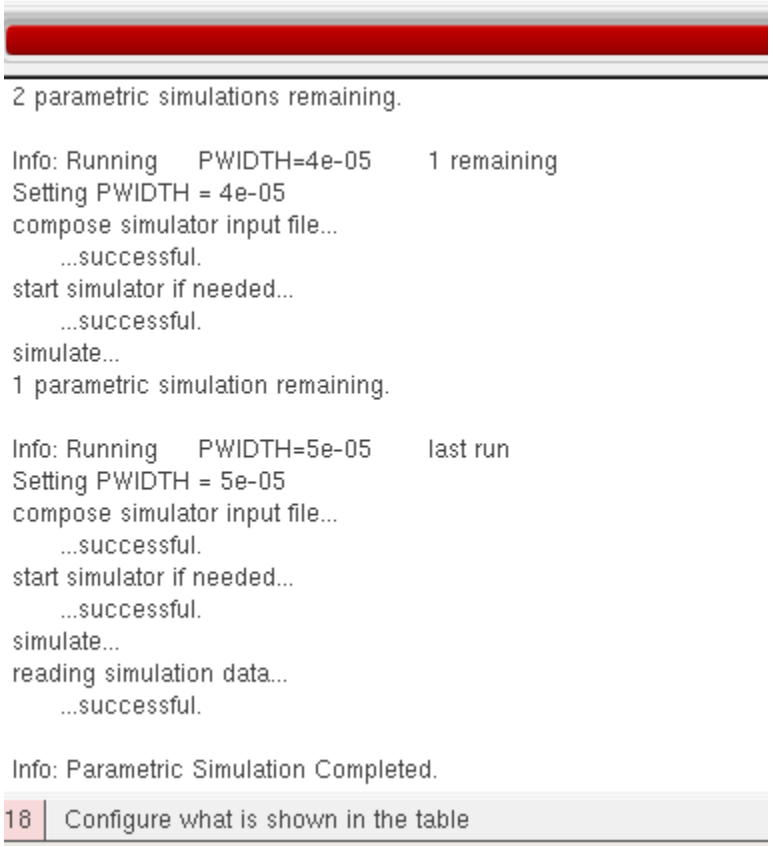
Setting up the parametric simulation

- **Put PWIDTH in the parameter box and choose to vary it from 10u to 50u in 5 steps**

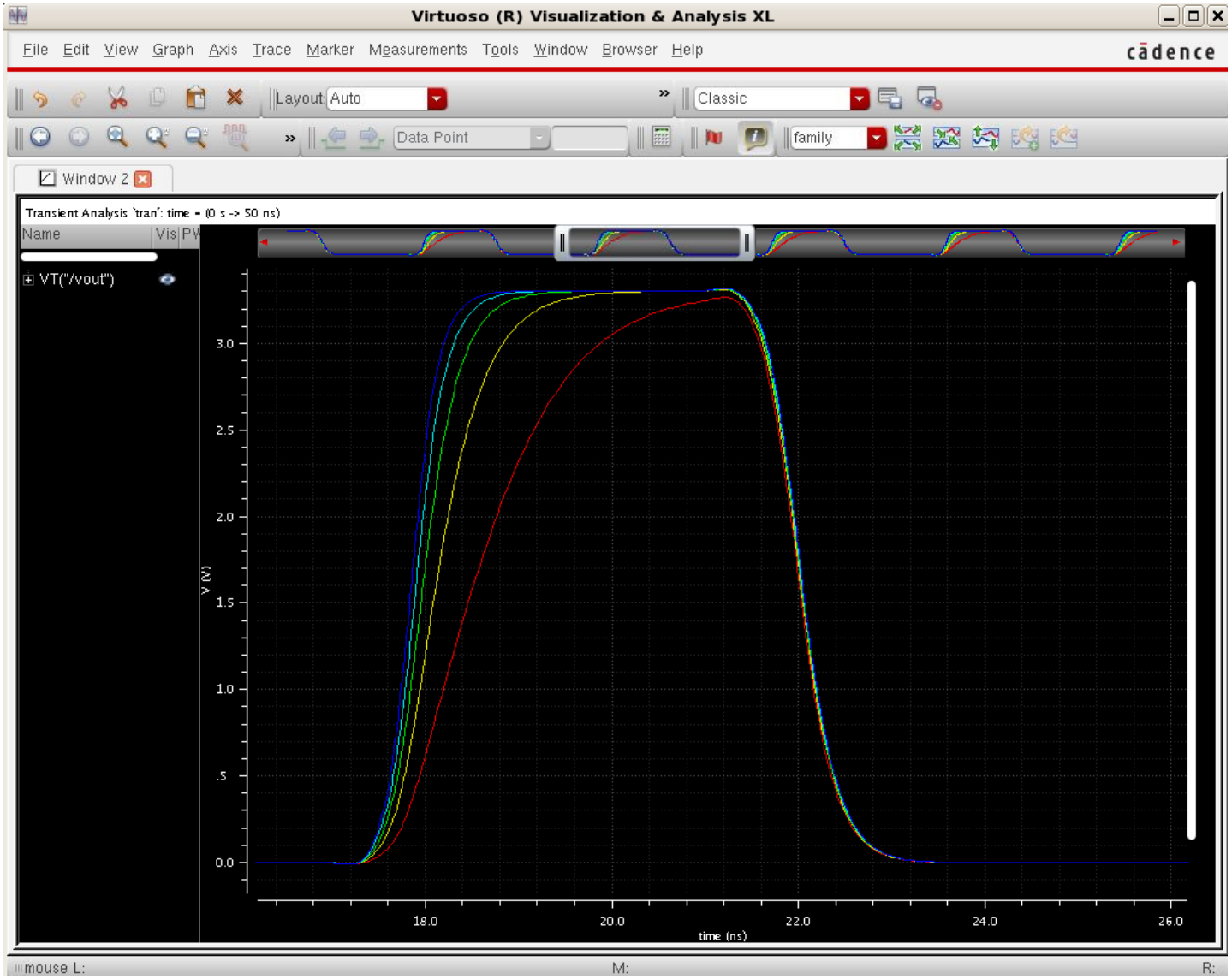


Run the parametric simulation

- Choose analysis -> start
- The log window will show progress of the runs
- Plot the output using the direct plot option as before
- Zoom into one cycle to examine the rise/fall times

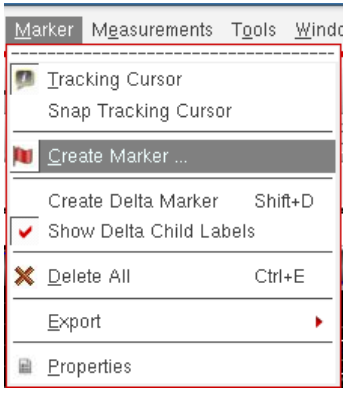


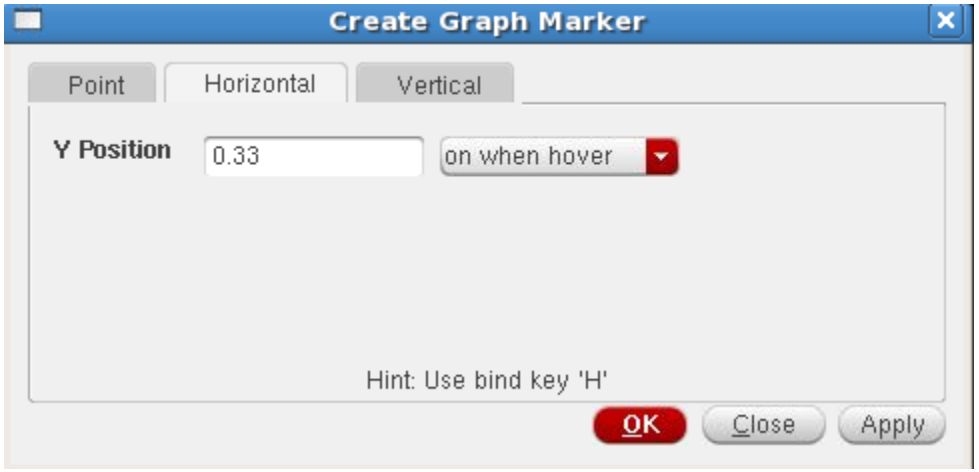
Parametric results



How to measure rise/fall time

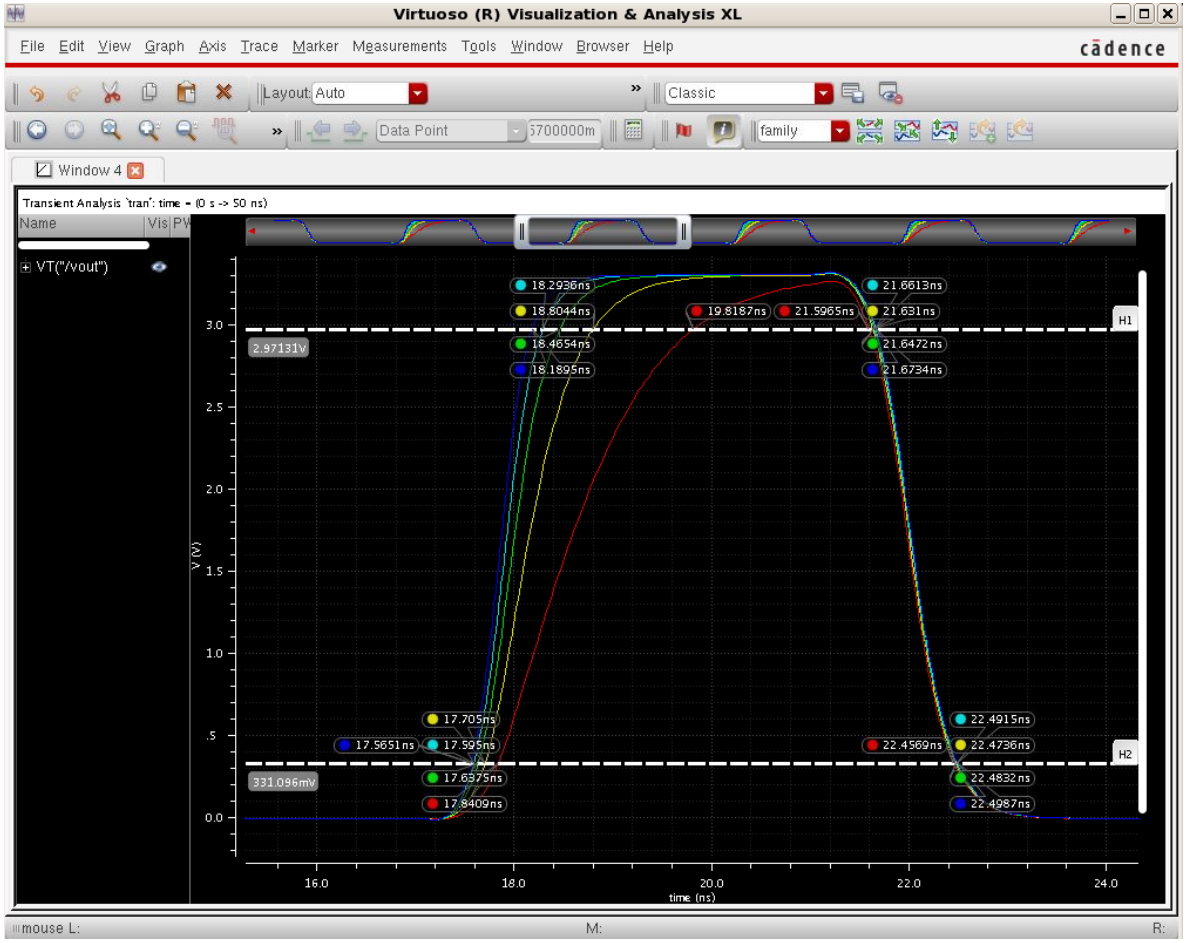
- **A few ways**
  - On an individual graph, use the marker facility
  - Use the calculator
  - Use horizontal markers
- **Trace marker**





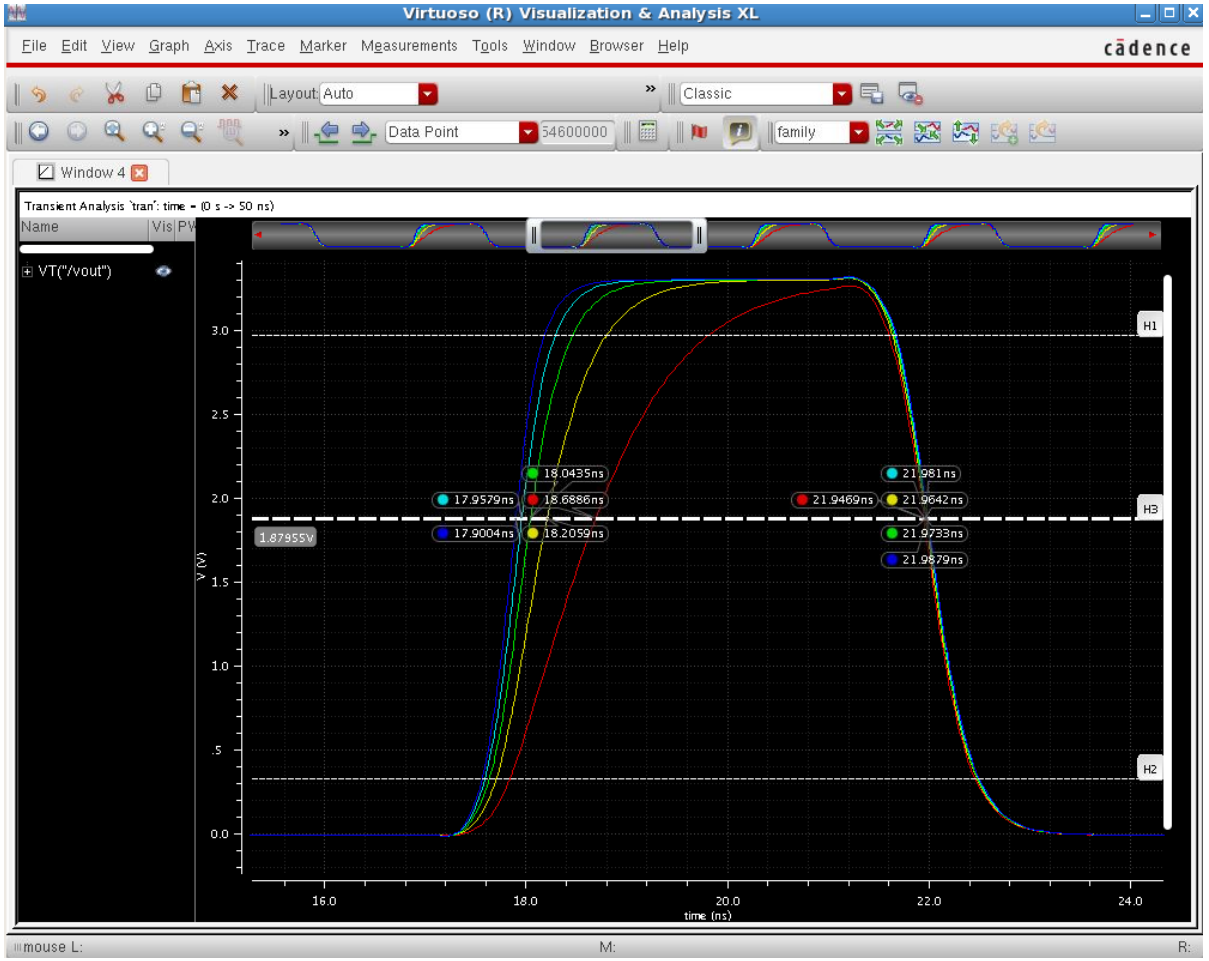
How to measure rise/fall time

- Place a trace marker at 0.33V and 2.97V and read off the times
- Rise time is 19.8-17.8ns =2ns



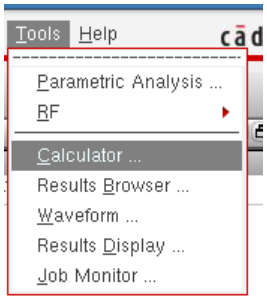
How to measure rise/fall time

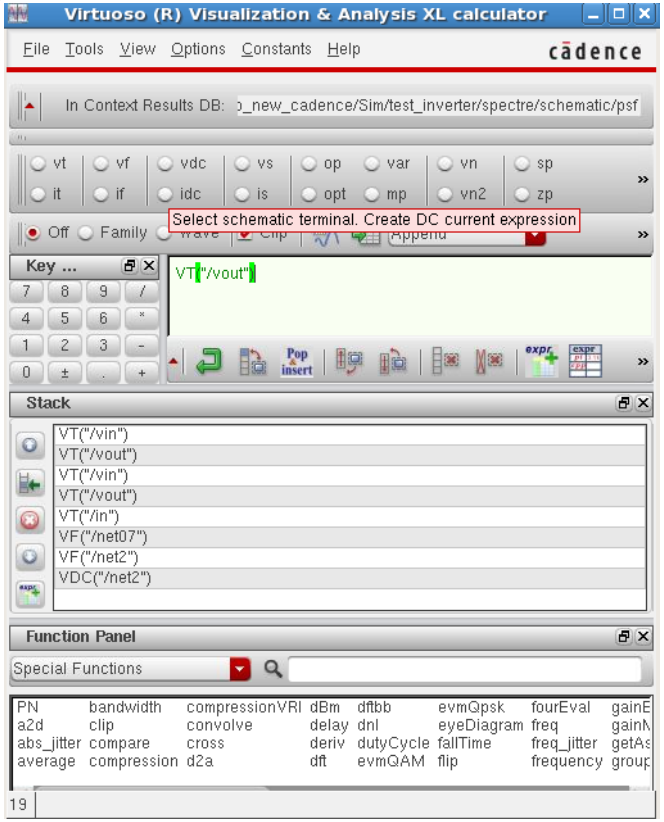
- Use horizontal markets ‘H’



How to measure rise/fall time

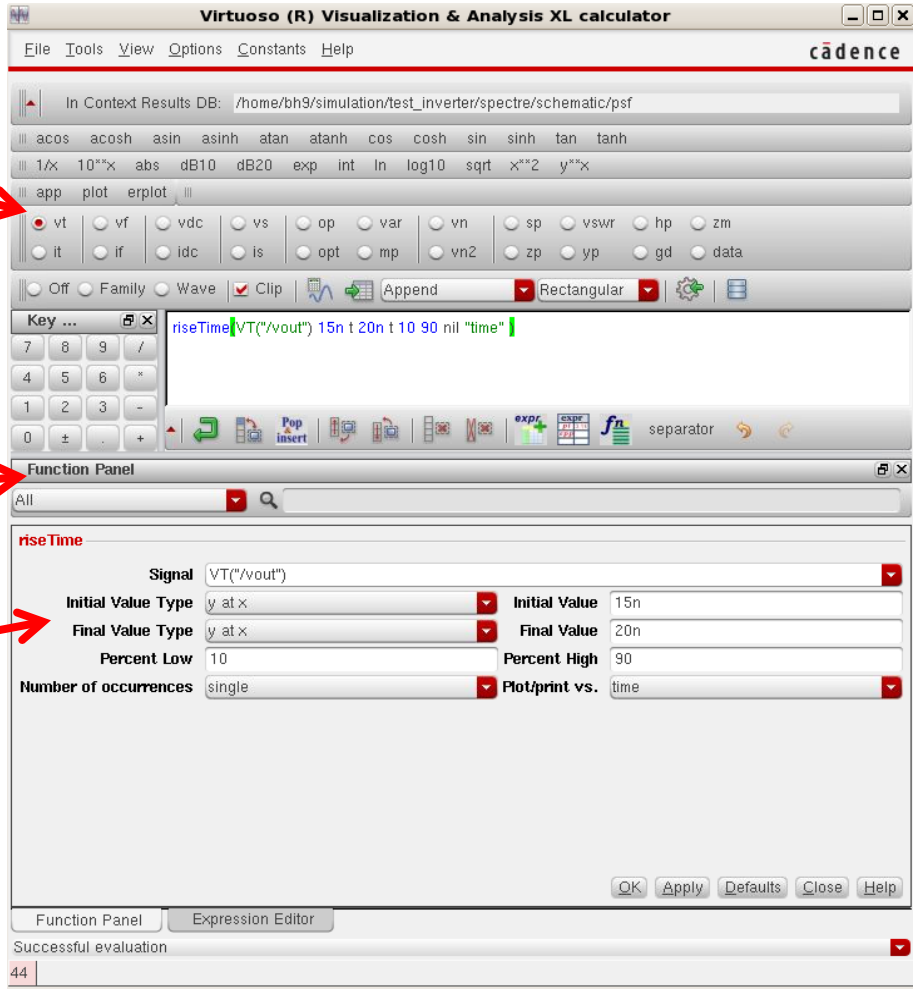
- **Use the calculator**
  - Very powerful collection of tools





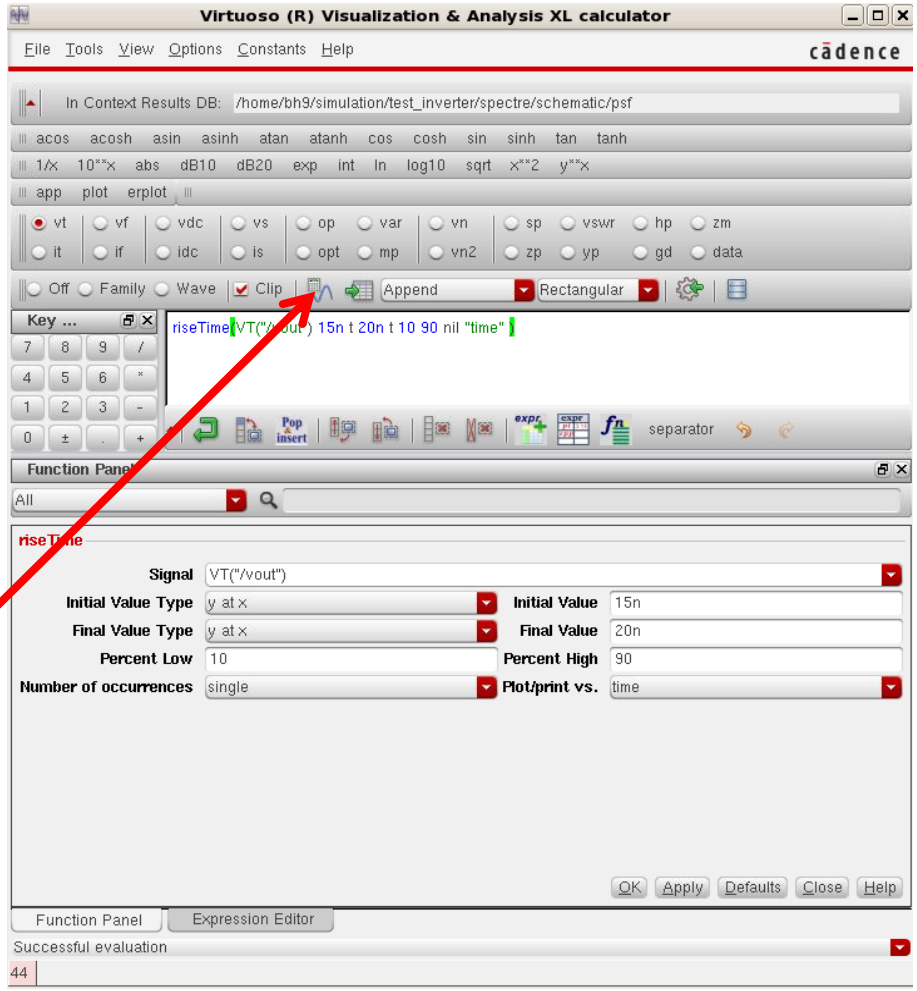
How to measure rise/fall time

- Choose vt
- Choose the net on the schematic window
- Select the riseTime from Function Panel
- Choose ‘y at x’
- Set initial and final x values (15ns, 20 ns)
- Press apply
- Press Evaluate



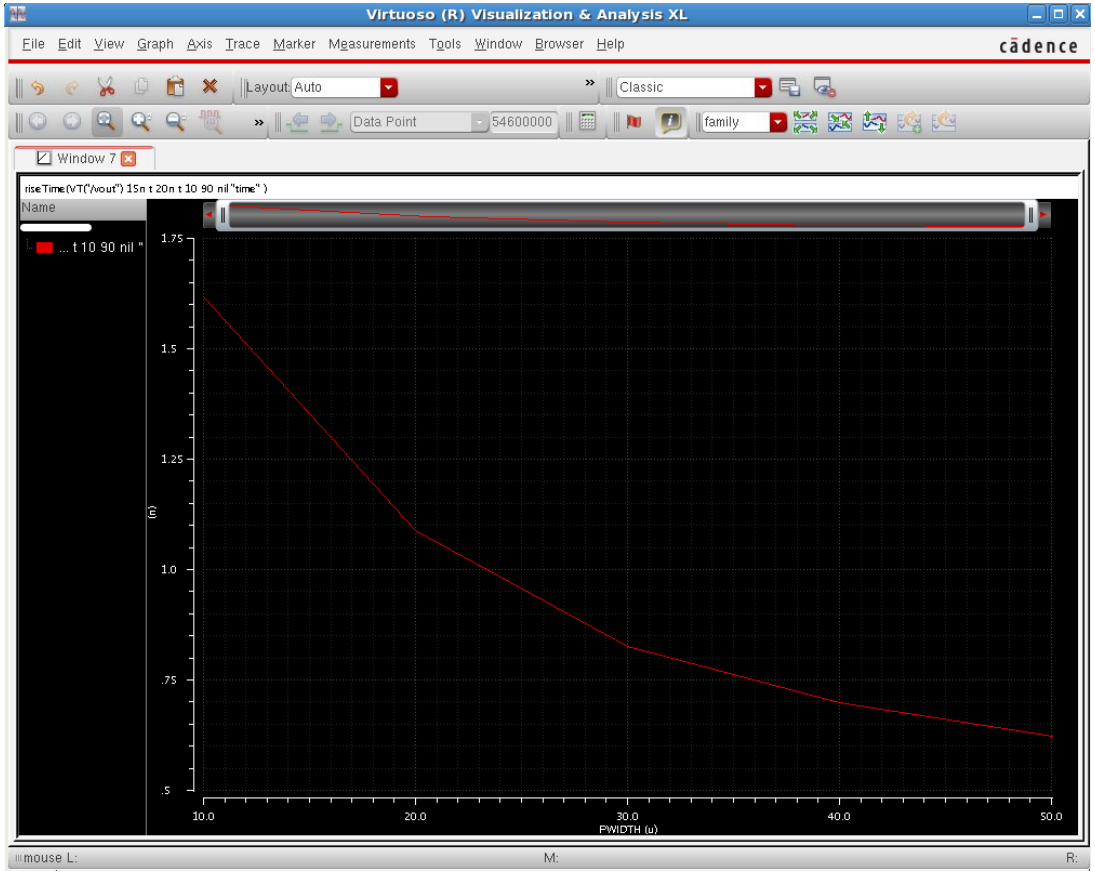
How to measure rise/fall time

- Choose vt
- Choose the net on the schematic window
- Select the riseTime from Function Panel
- Choose ‘y at x’
- Set initial and final x values (15ns, 20 ns)
- Press apply
- Press Evaluate



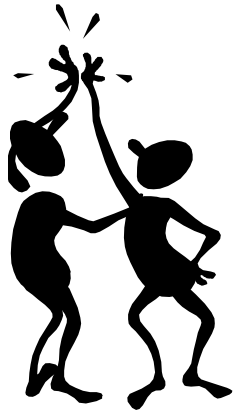
How to measure rise/fall time

- **Plots the rise time with different values of the PMOS width**



Summary

- In this lecture we created a symbol from a schematic
- We made a test schematic containing the symbol
- We learnt how to simulate the design
- We have seen how to plot and measure the results
- We have shown how to set up hierarchical parameter passing in your design
- We have used design variables and parametric simulations to show rise time variation against PMOS width



Lab: Simulation

Objective: Create a symbol and test circuit. Then simulate your test circuit and use parametric simulations and design variables