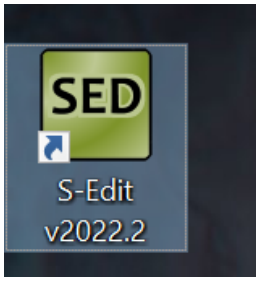
وَمَا أُوتِيتُمْ مِنَ الْعِلْمِ إِلَّا قَلِيلًا

Ain Shams University – Faculty of Engineering – ECE Dept. – Integrated Circuits Lab.
Dr. Hesham Omran

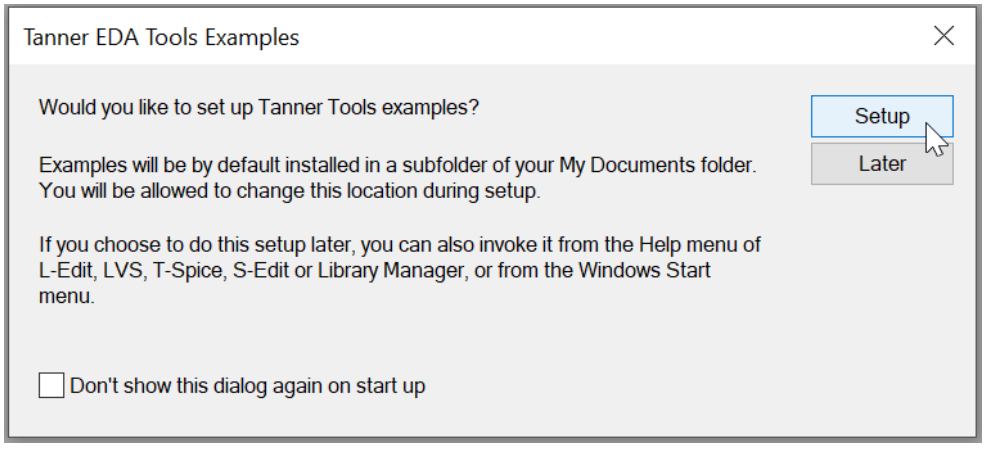
Analog/Mixed-Signal Simulation and Modeling
Course Project
Tanner Tutorial

Veriloga Design Entry and Simulation

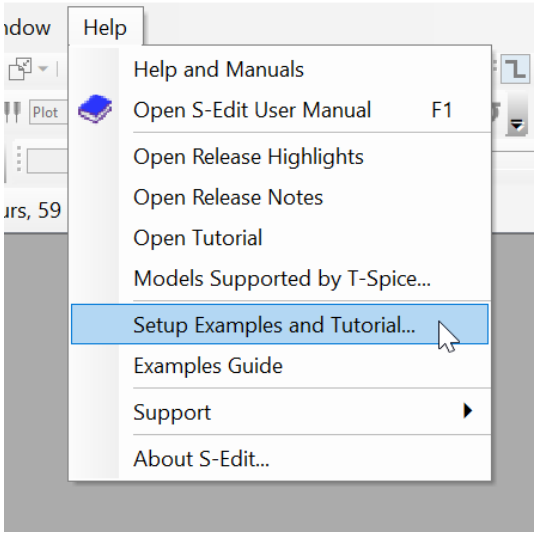
Start S-edit



Setup Examples and Tutorials.

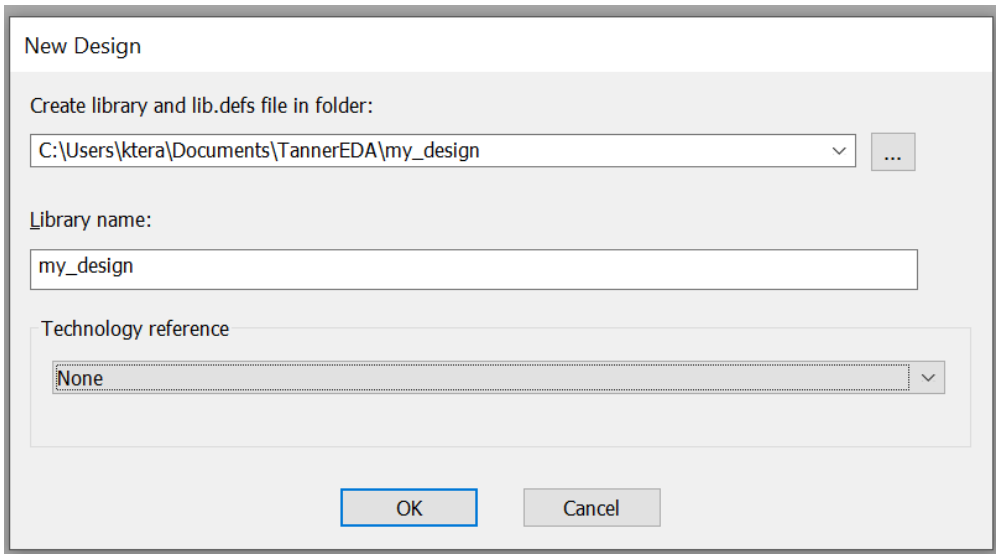


If you don’t see the previous window, you can get it from Help -> Setup Examples.

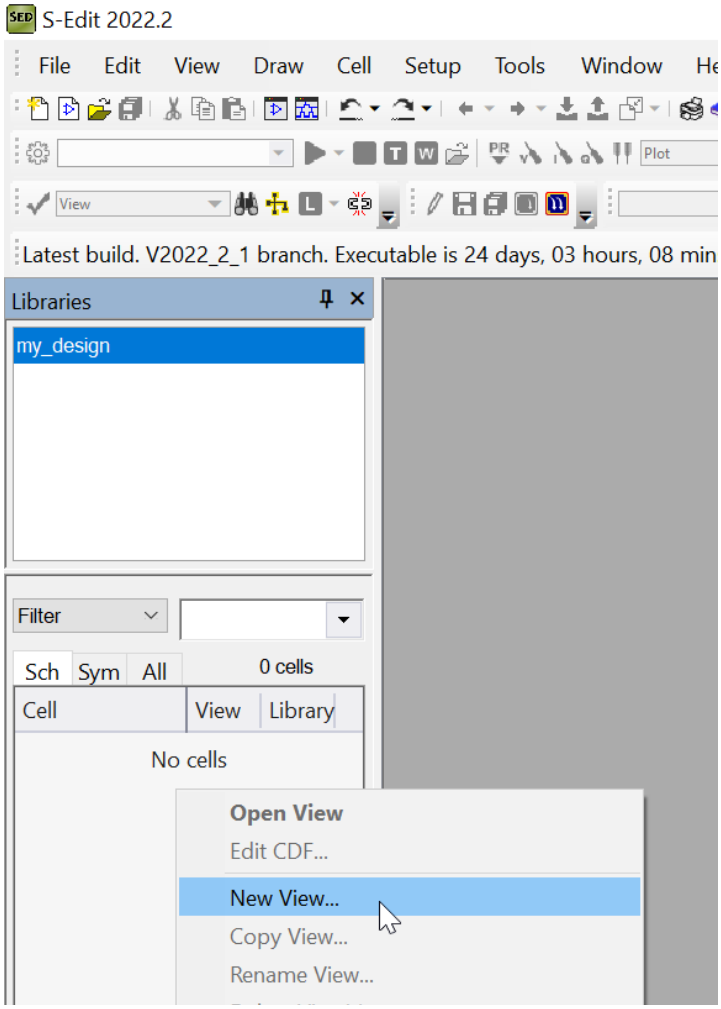


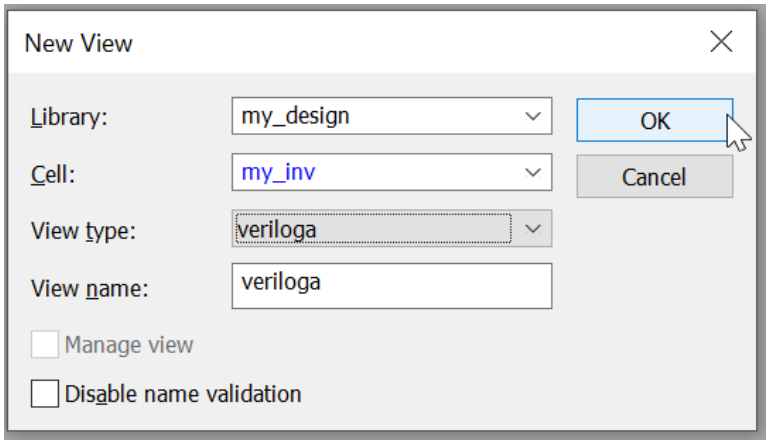
Note the path where the examples will be stored.
Ex: Documents\TannerEDA\TannerTools_v2022.2

Create a new design in a new folder from the menu bar: File -> New -> New Design.



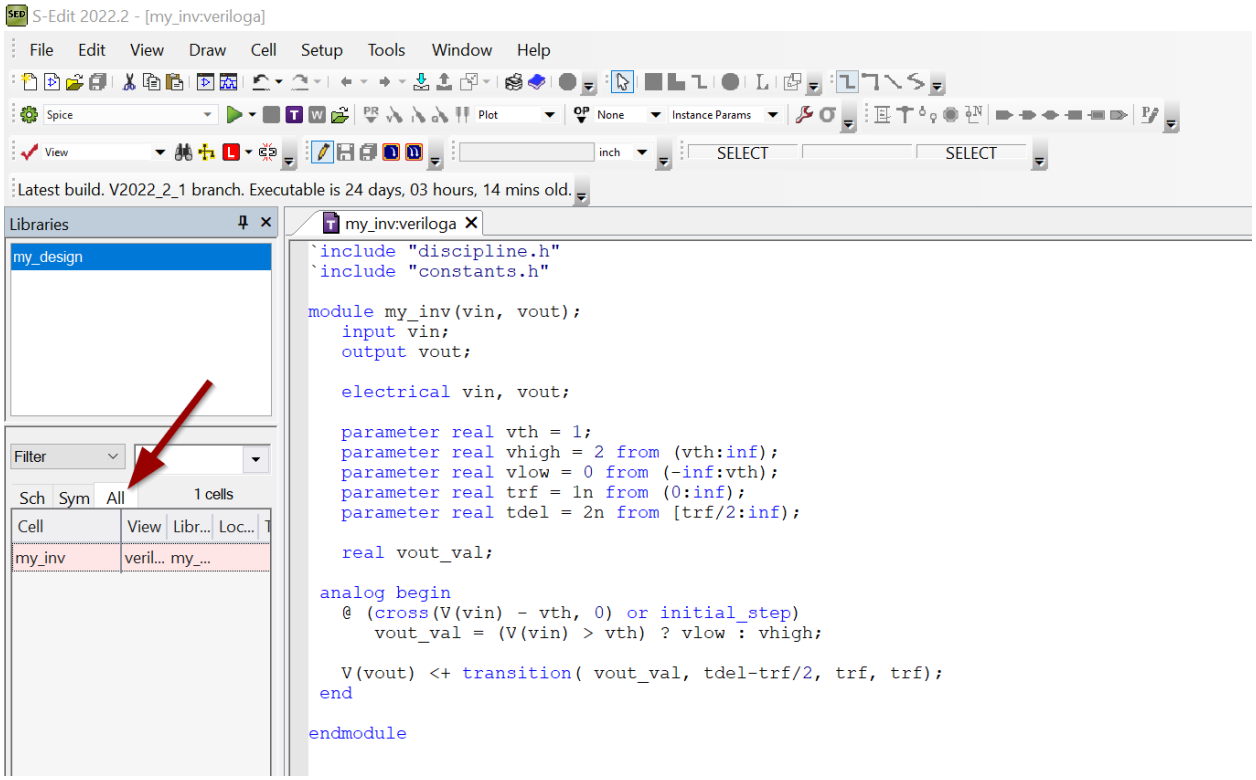
Create a new view (type: veriloga).





Create inverter veriloga description.

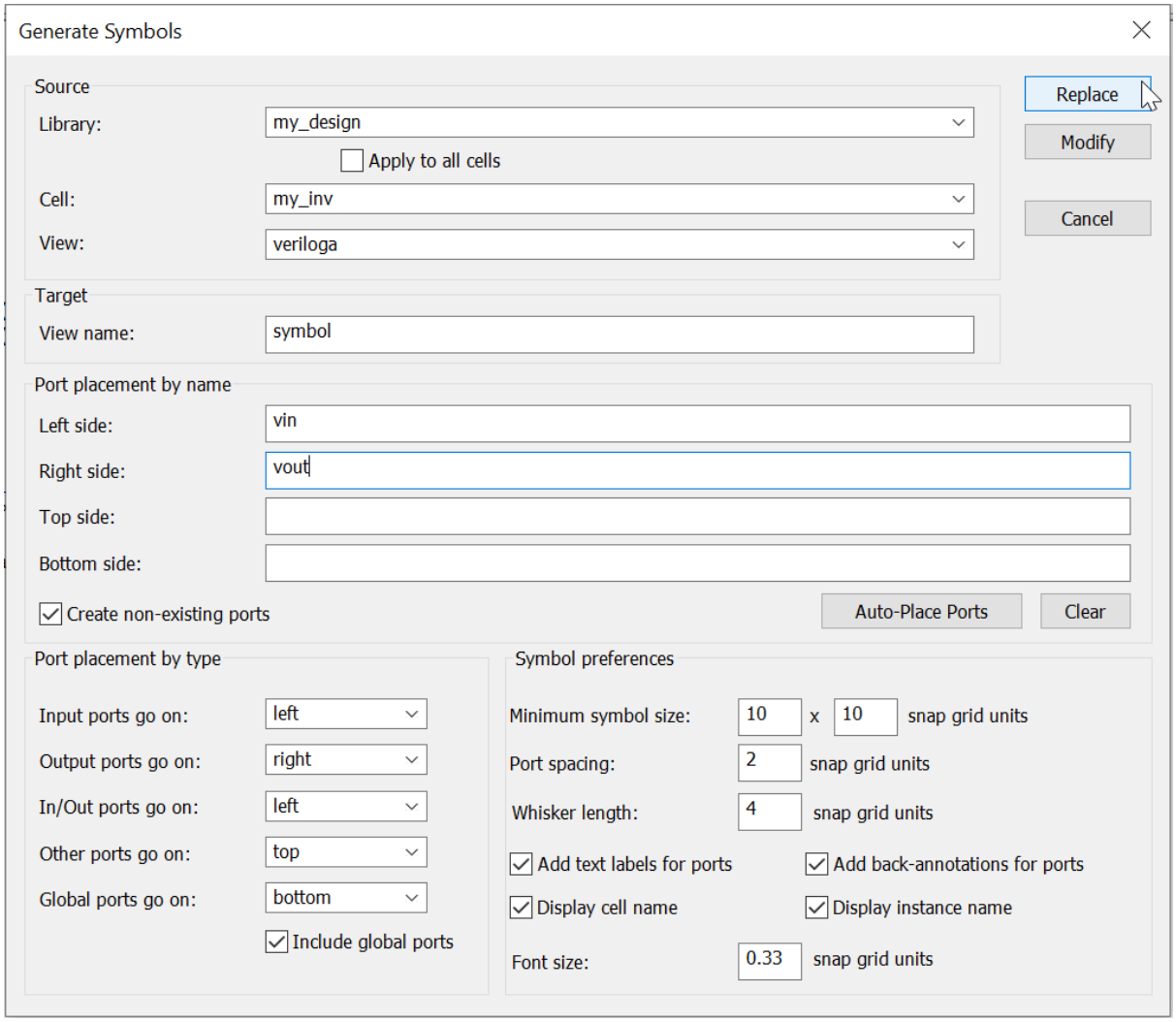
```
`include "discipline.h"
`include "constants.h"

module my_inv(vin, vout);
      input vin;
      output vout;

      electrical vin, vout;

      parameter real vth = 1;
      parameter real vhigh = 2 from (vth:inf);
      parameter real vlow = 0 from (-inf:vth);
      parameter real trf = 1n from (0:inf);
      parameter real tdel = 2n from [trf/2:inf);

      real vout_val;

 analog begin
      @ (cross(V(vin) - vth, 0) or initial_step)
            vout_val = (V(vin) > vth) ? vlow : vhigh;

      V(vout) <+ transition( vout_val, tdel-trf/2, trf, trf);
 end

endmodule
```

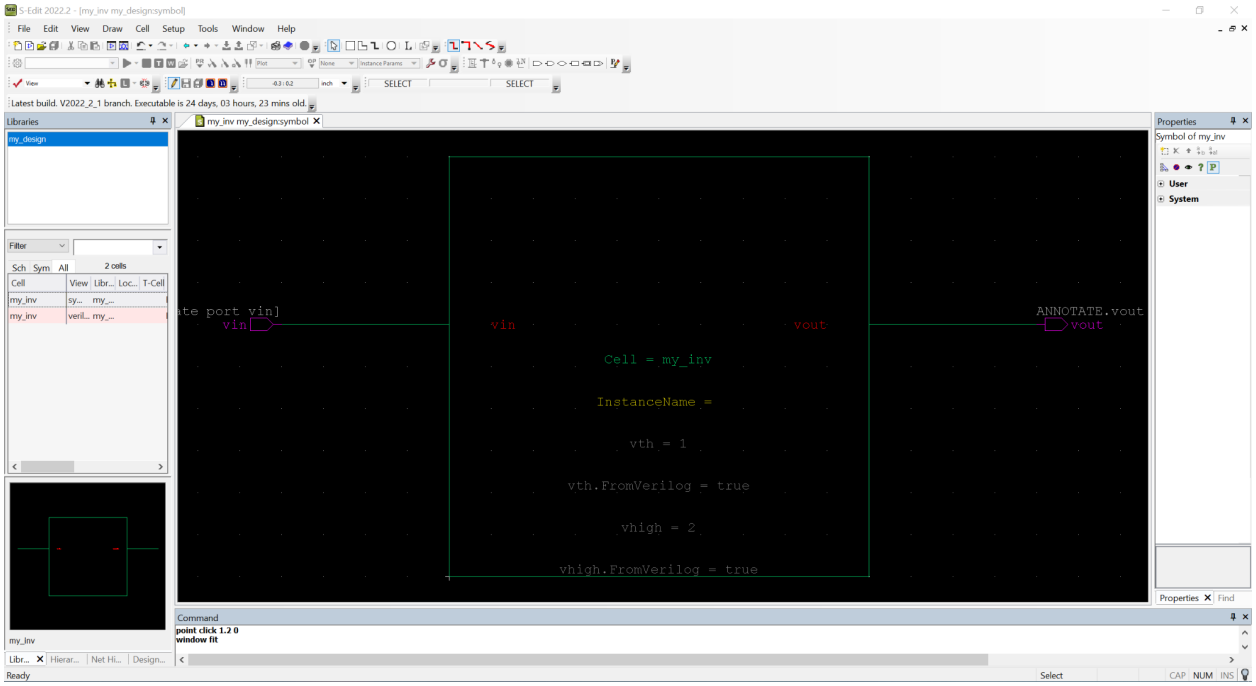


Save the veriloga view.

Create a symbol from the menu bar: Cell -> Generate symbol.

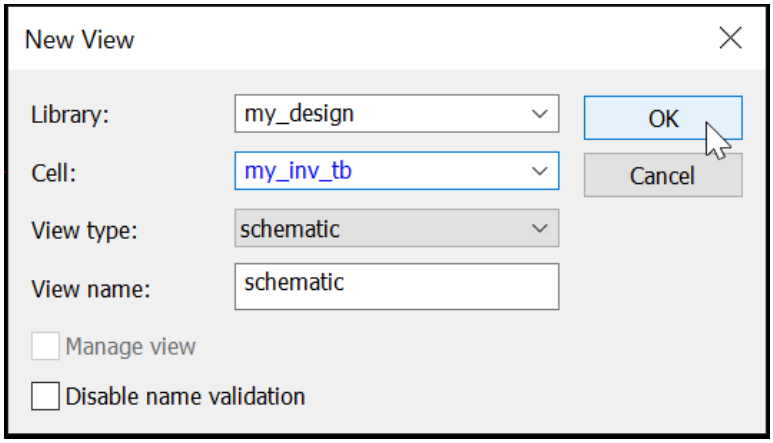
Note: Autoplace ports works for schematic views, but doesn't work for veriloga views.


Generate Symbols
Source
Library:
my_design
Apply to all cells
Cell:
my_inv
View:
veriloga
Replace
Modify
Cancel
Target
View name:
symbol
Port placement by name
Left side:
vin
Right side:
vout
Top side:
Bottom side:
Create non-existing ports
Auto-Place Ports
Clear
Port placement by type
Input ports go on:
left
Output ports go on:
right
In/Out ports go on:
left
Other ports go on:
top
Global ports go on:
bottom
Include global ports
Symbol preferences
Minimum symbol size:
10
x
10
snap grid units
Port spacing:
2
snap grid units
Whisker length:
4
snap grid units
Add text labels for ports
Add back-annotations for ports
Display cell name
Display instance name
Font size:
0.33
snap grid units

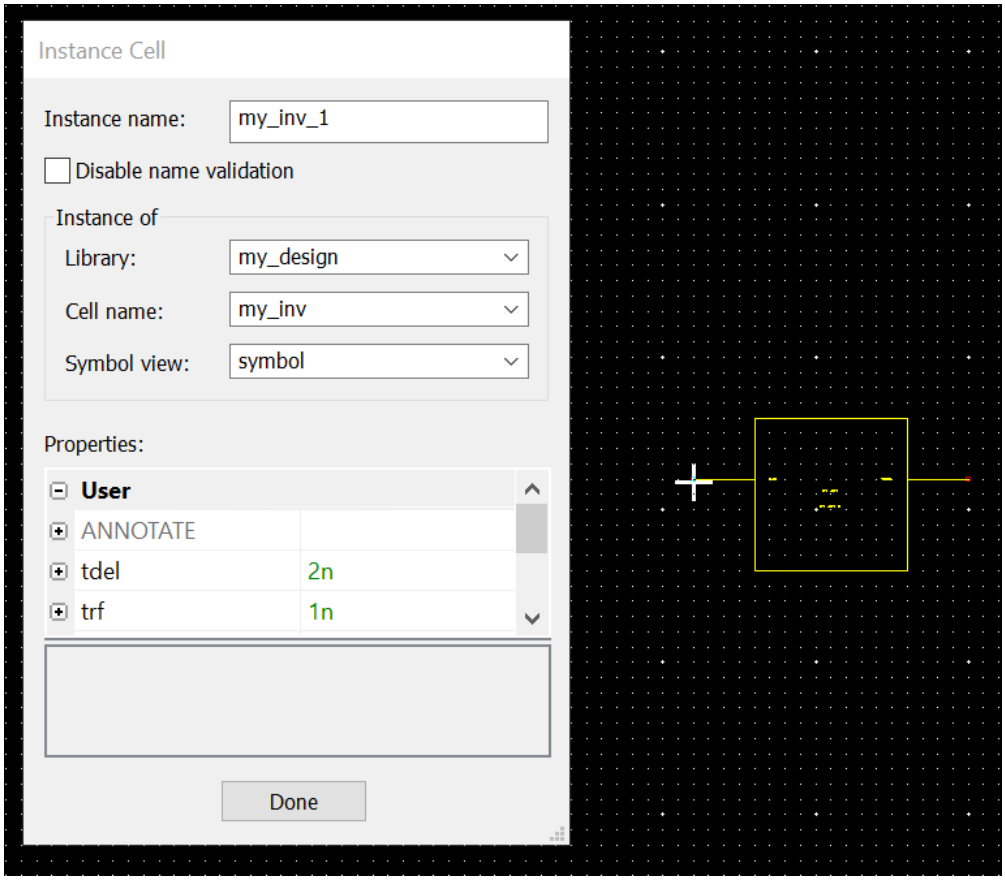


Save all.

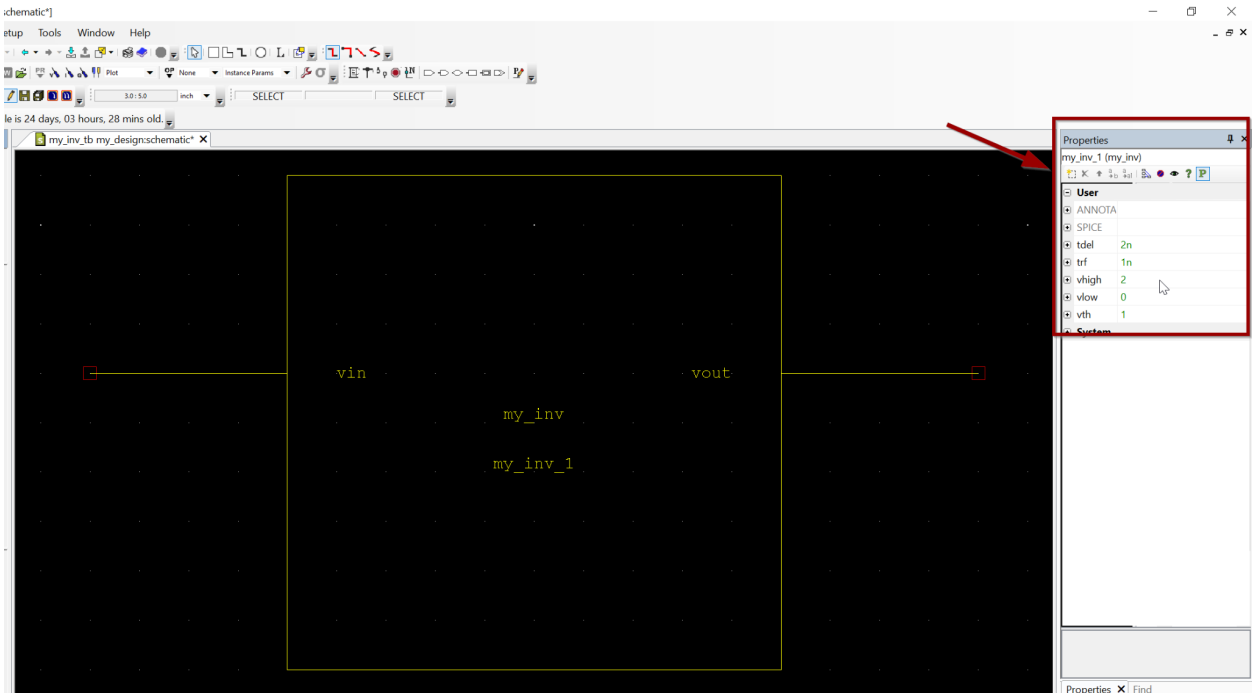
Create a new view for the testbench (type: schematic).



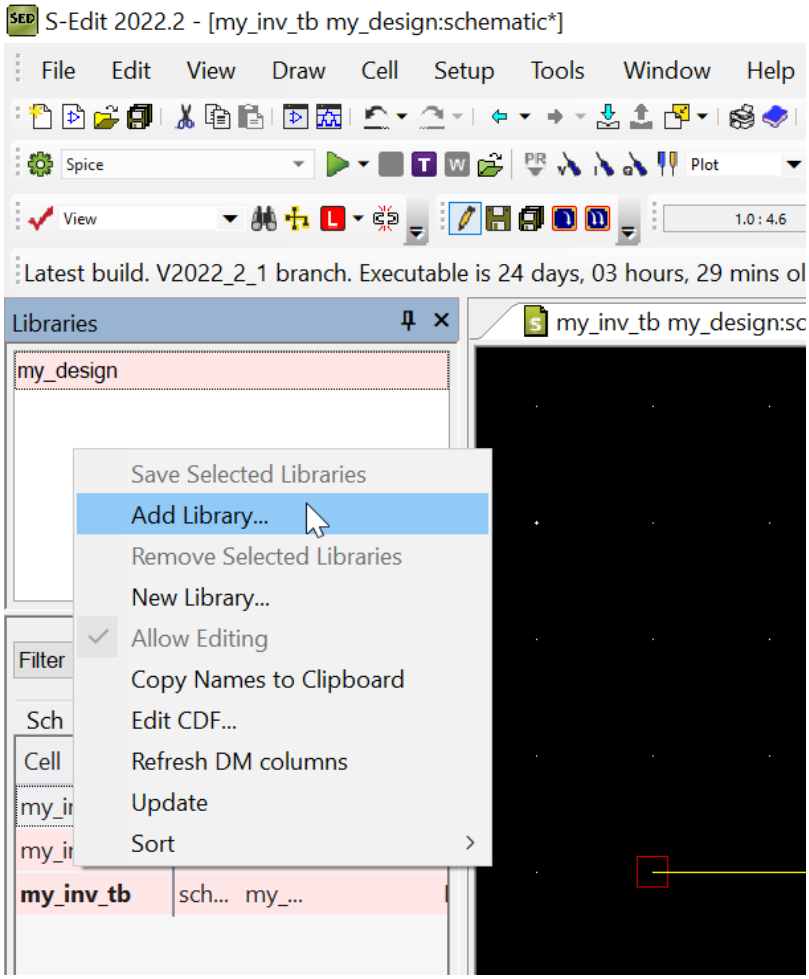
Use the hotkey ‘i’ to add an instance for your inverter.

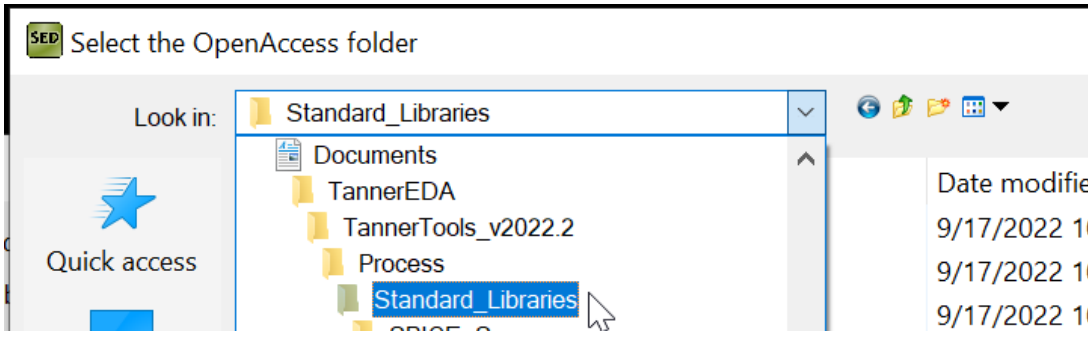


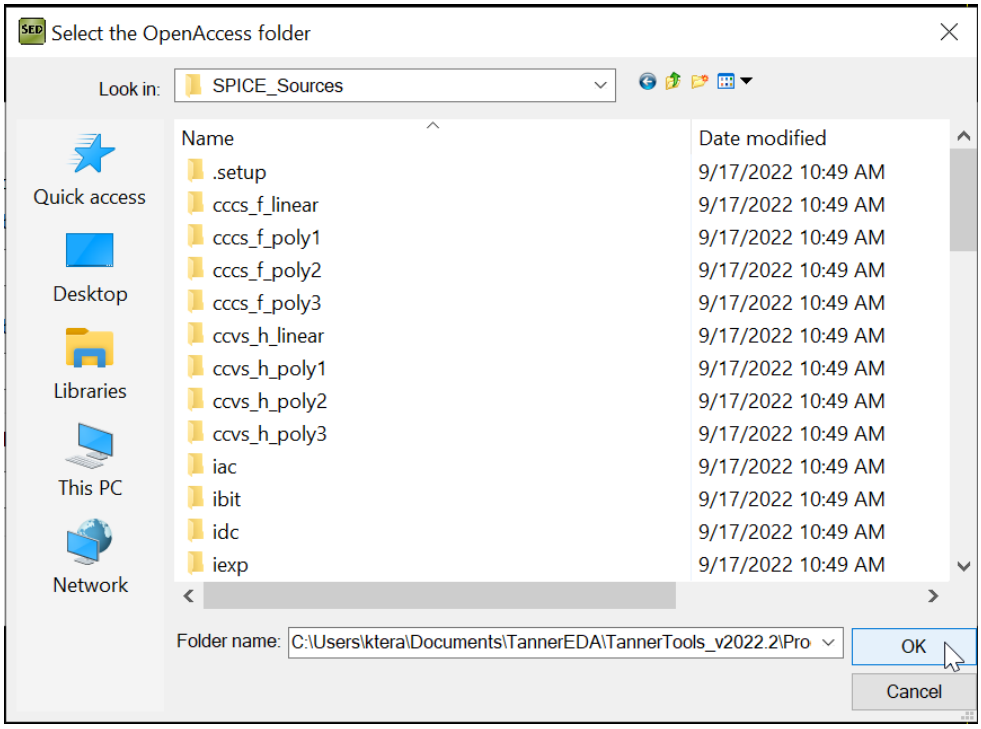
You can edit the properties on the right side.

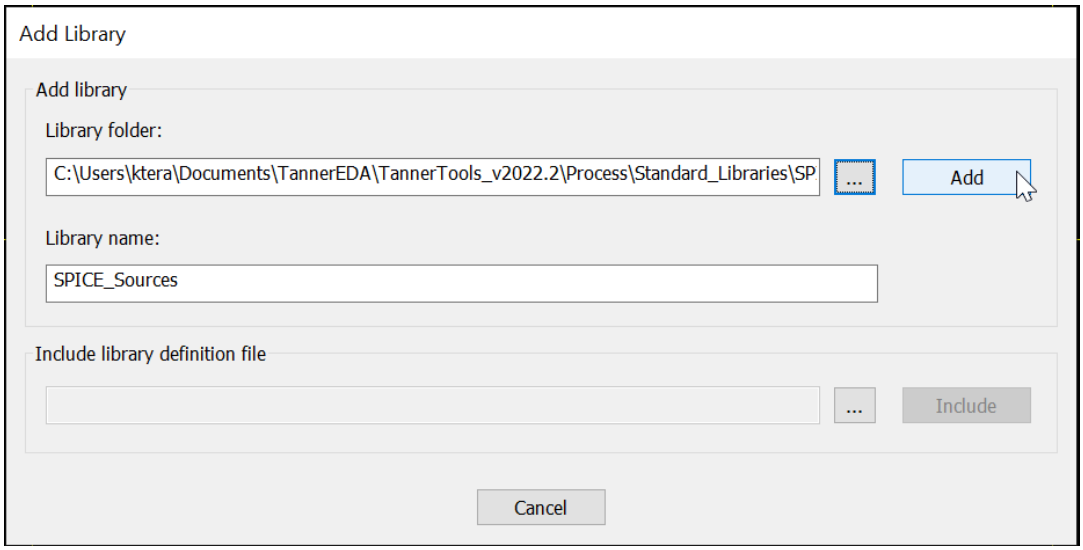


Add the standard analog libraries to complete your testbench.

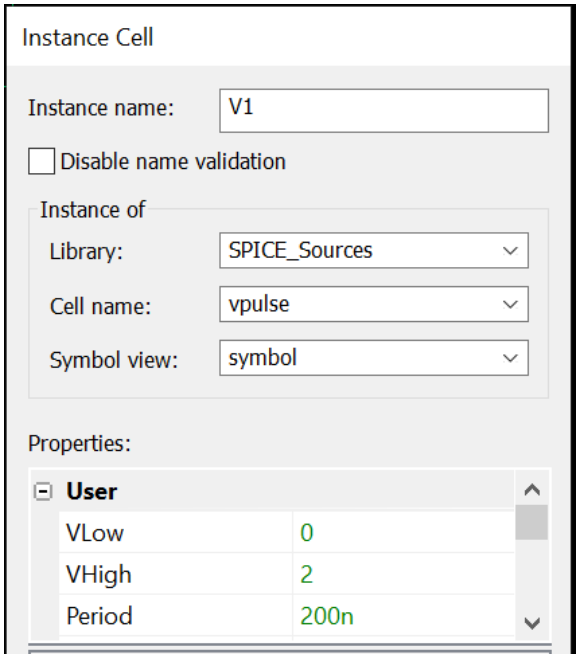




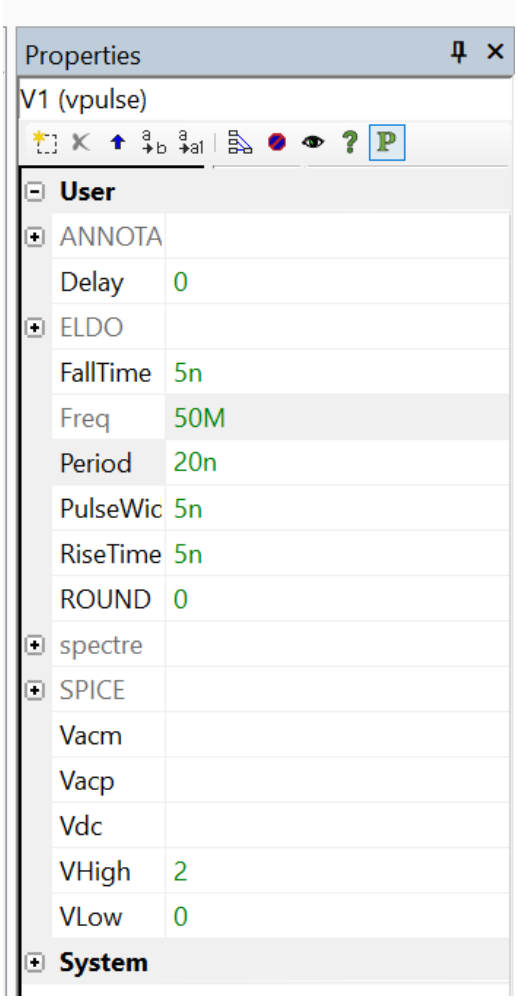




Add vpulse instance to the schematic from SPICE_Sources library.



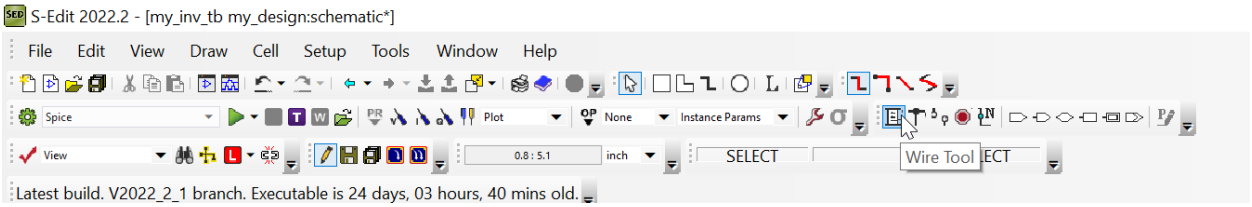
Edit Vpulse properties as below.



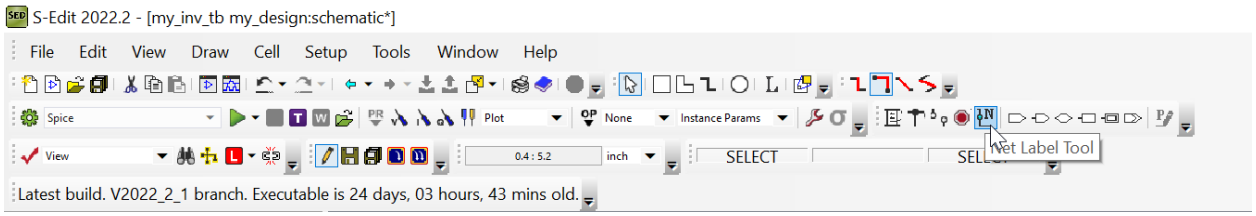
Add gnd and NoConnection from the Misc library.

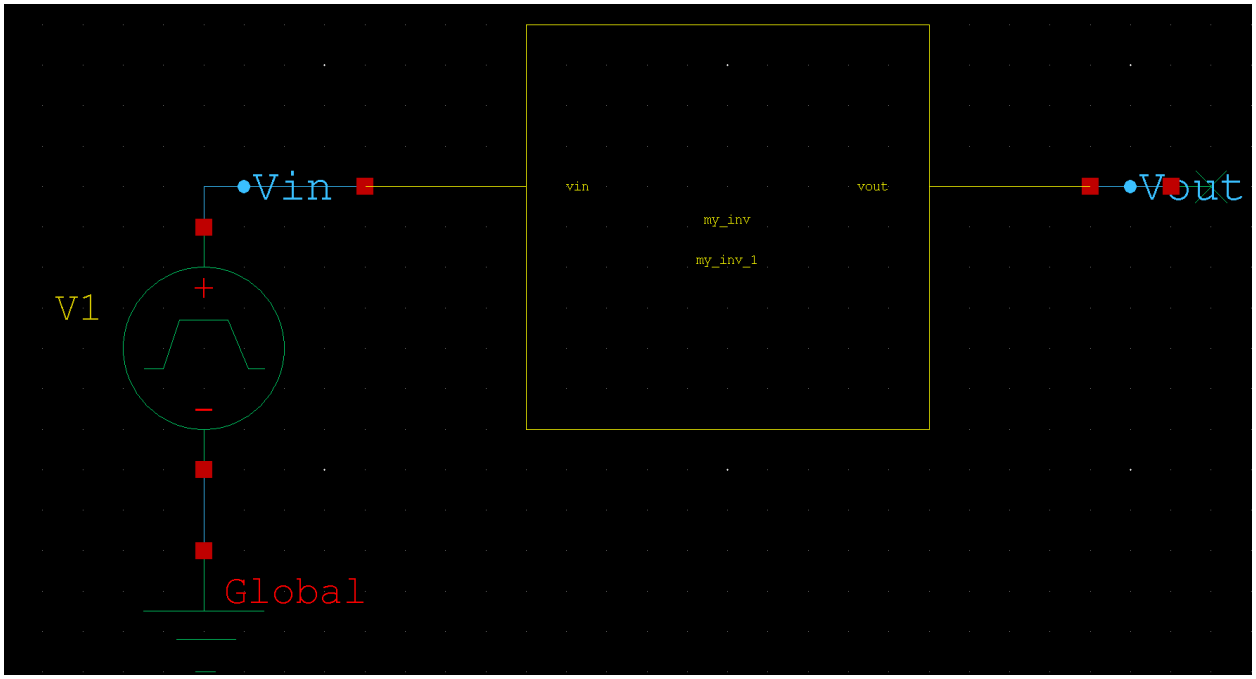
Hint: You can move elements in the schematic with the middle mouse button.

Use the wire tool to connect the elements.

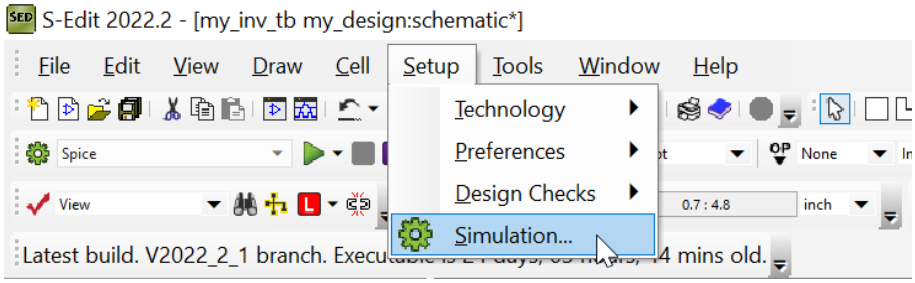


Add labels using the label tool.

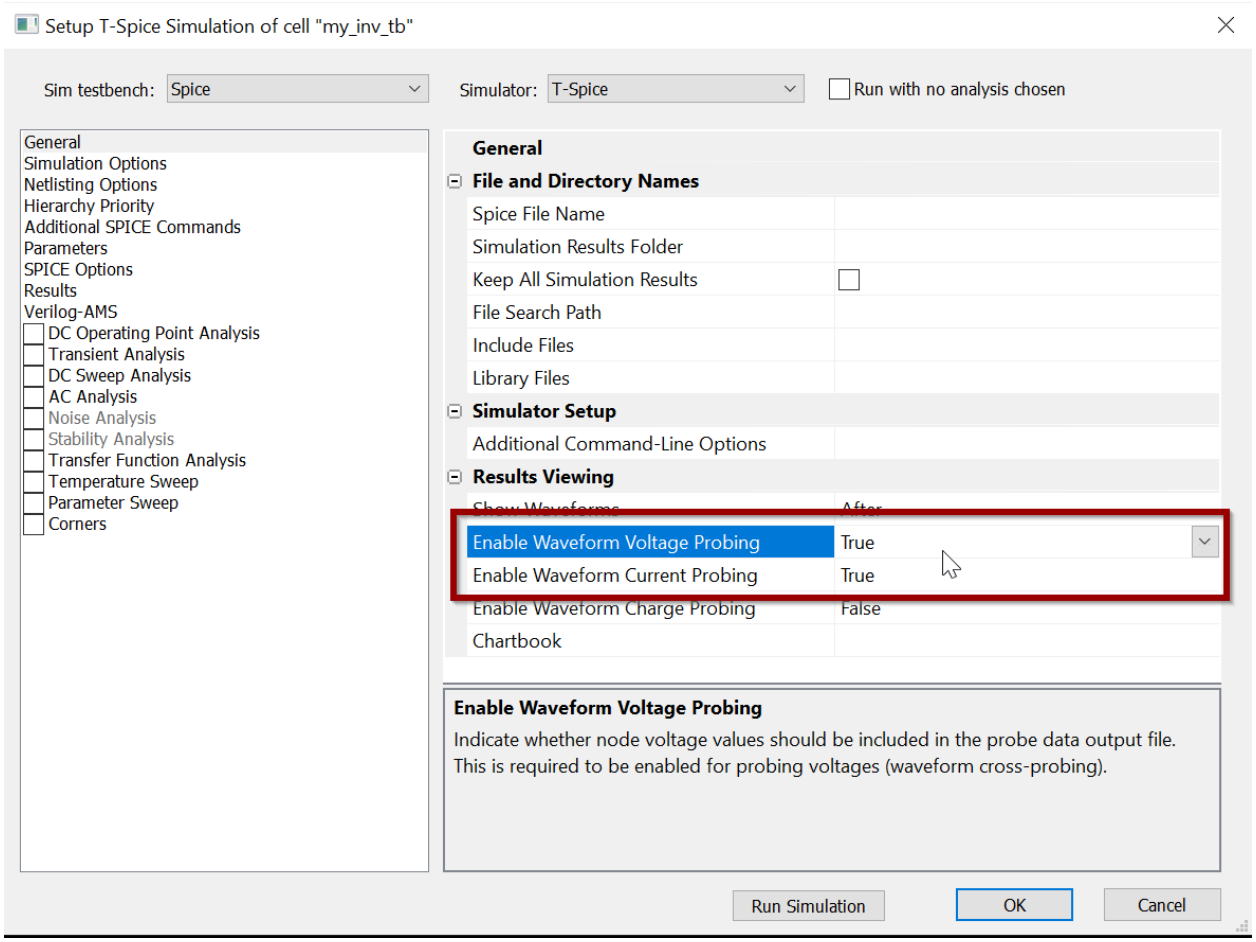




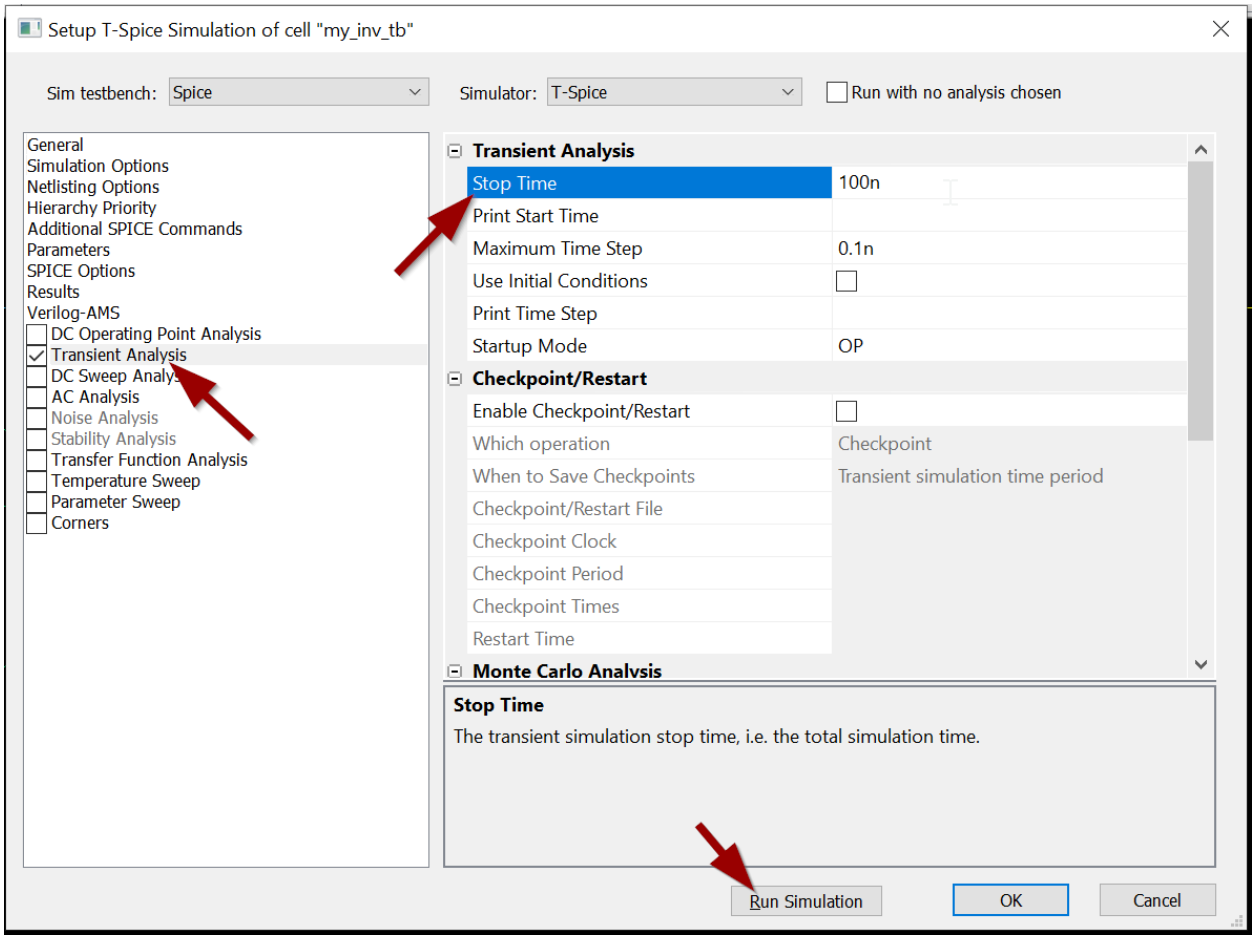
Setup the simulation.



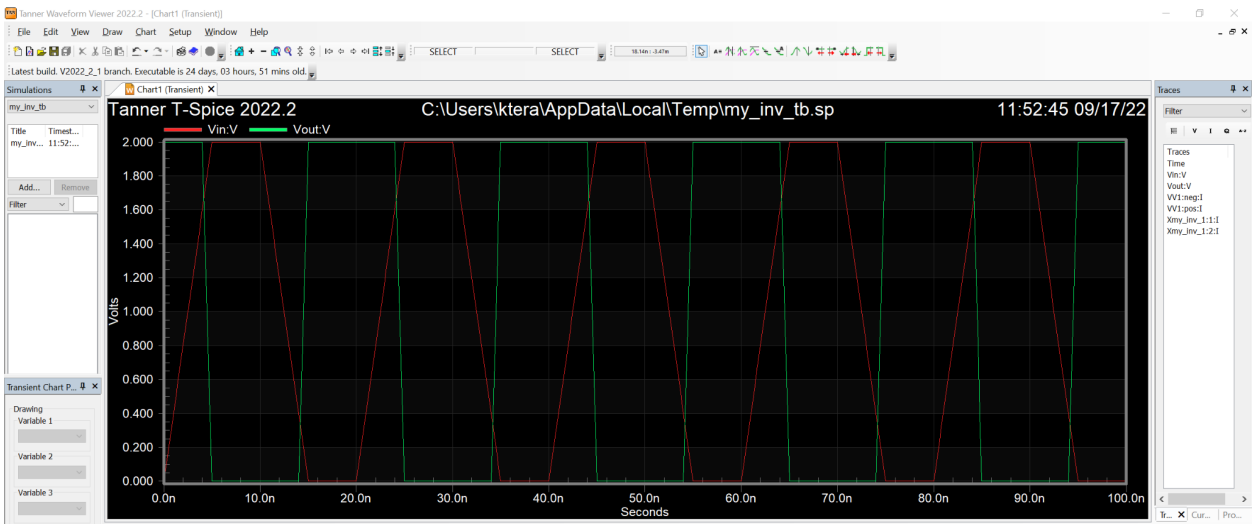
Set waveform probing to True.



Setup transient analysis and run simulation.



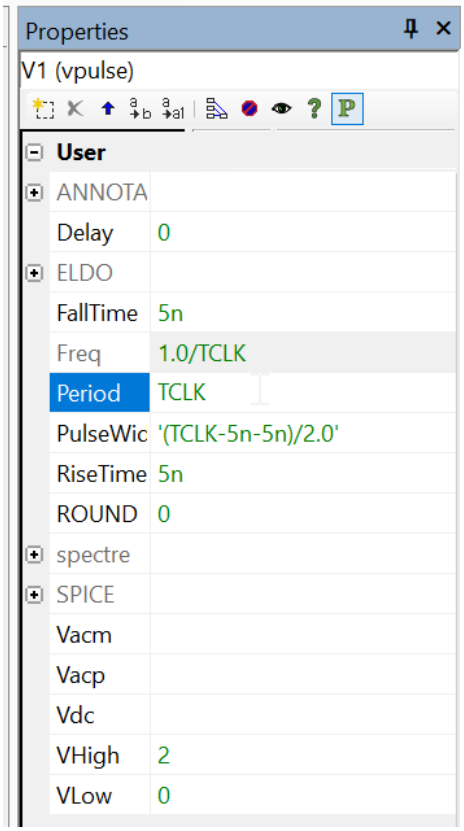
After the simulation is completed successfully, probe Vin and Vout.



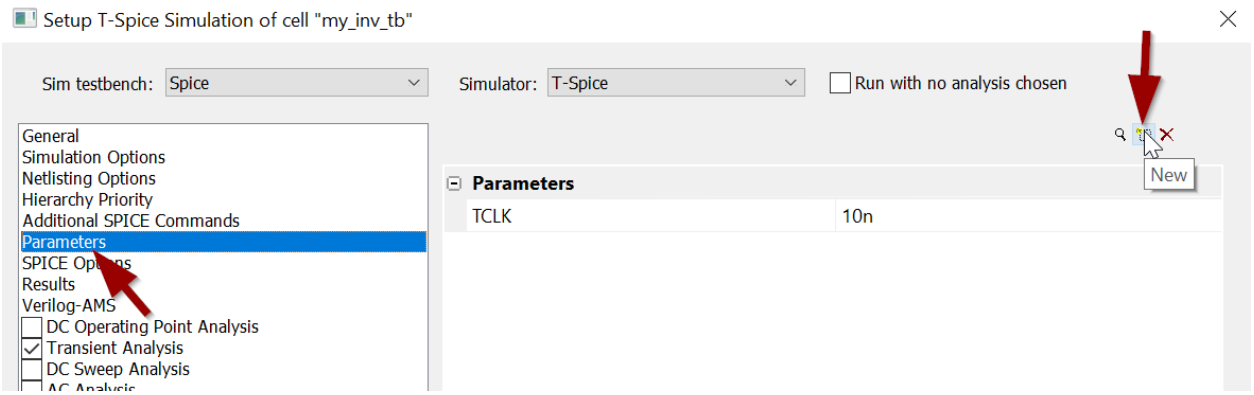
Parametric Sweeps and Waveform Calculator

Note: This step is just to practice using parametric sweeps and waveform calculator. There is no design perspective in this step.

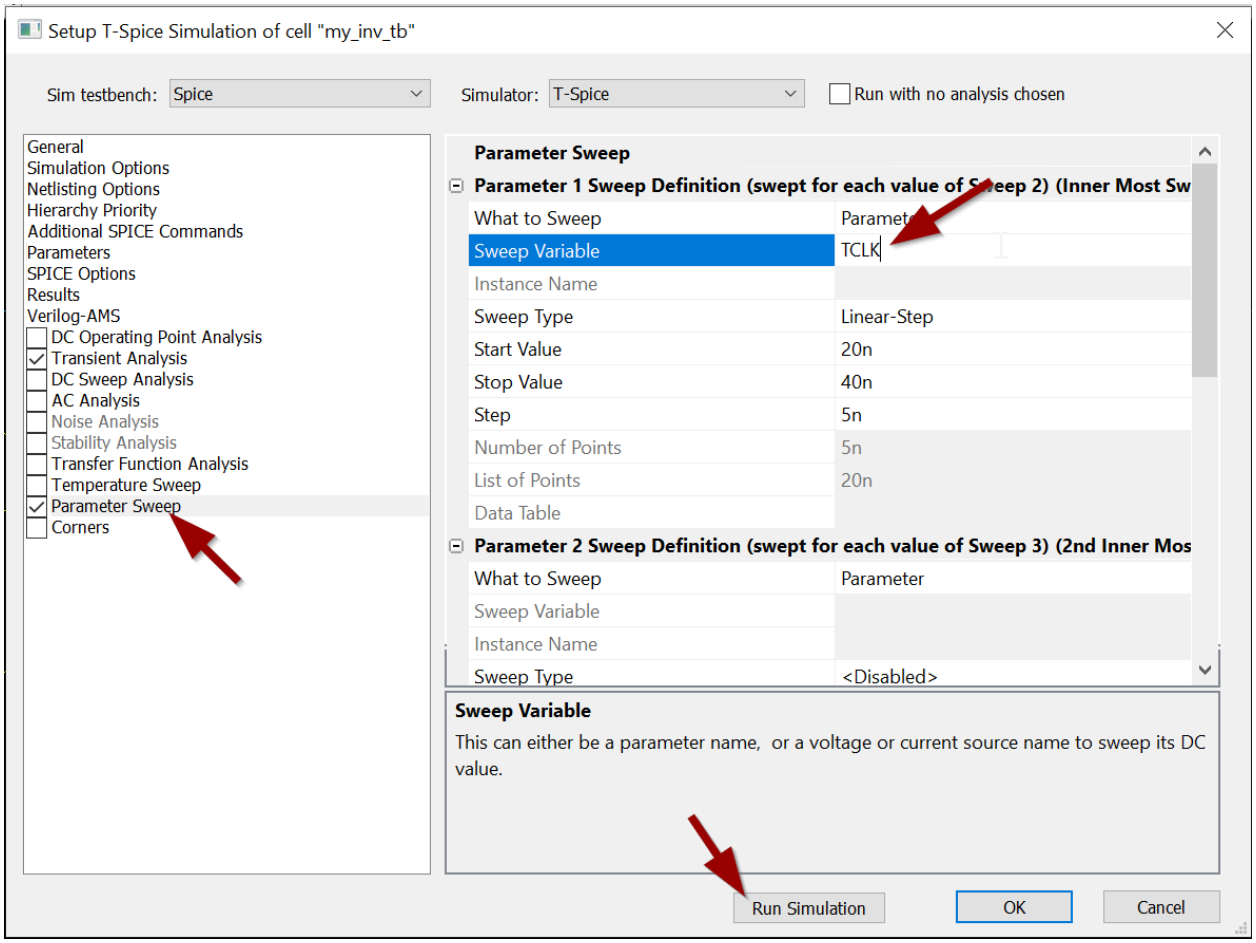
To do parametric sweep, define the period of Vpulse as a parameter.



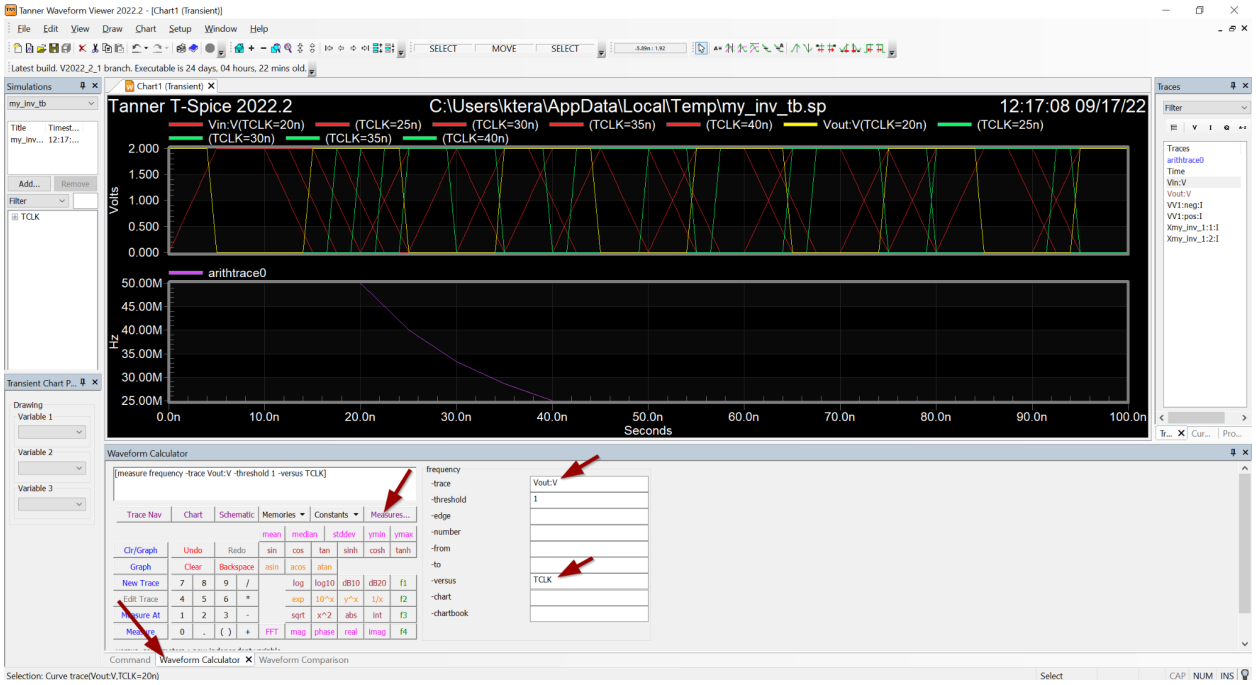
In the simulation setup, add a new parameter and give it the same name.



Define parametric sweep for the parameter you created and run the simulation.

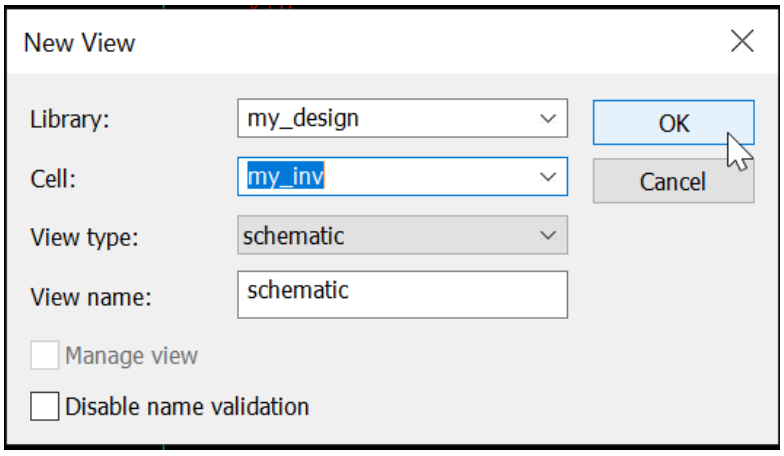


In the waveform viewer, measure the frequency and plot it vs TCLK as shown below.

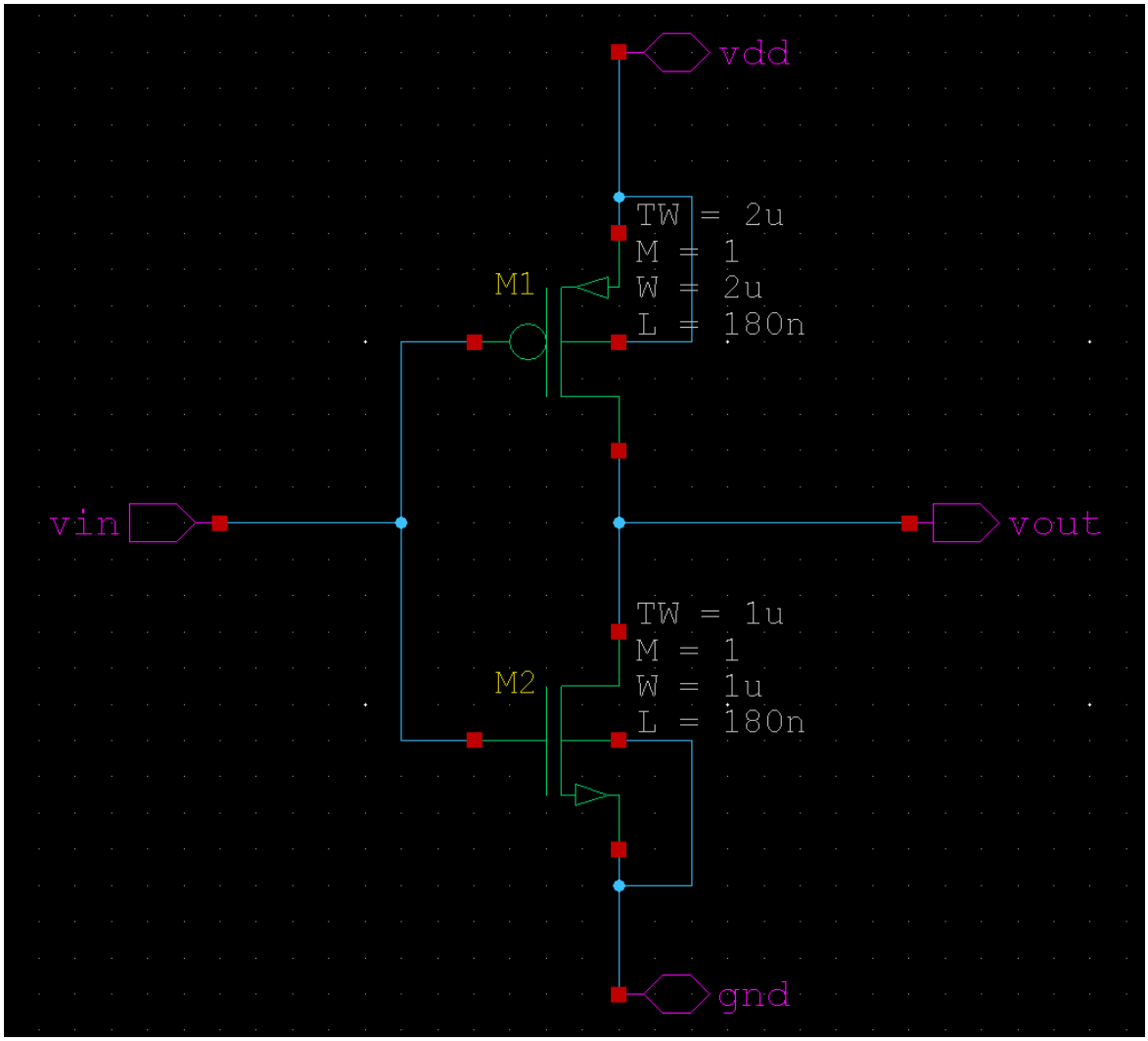


Transistor Level Simulation (Hierarchy Editor/Priority)

Create a schematic view for the inverter.

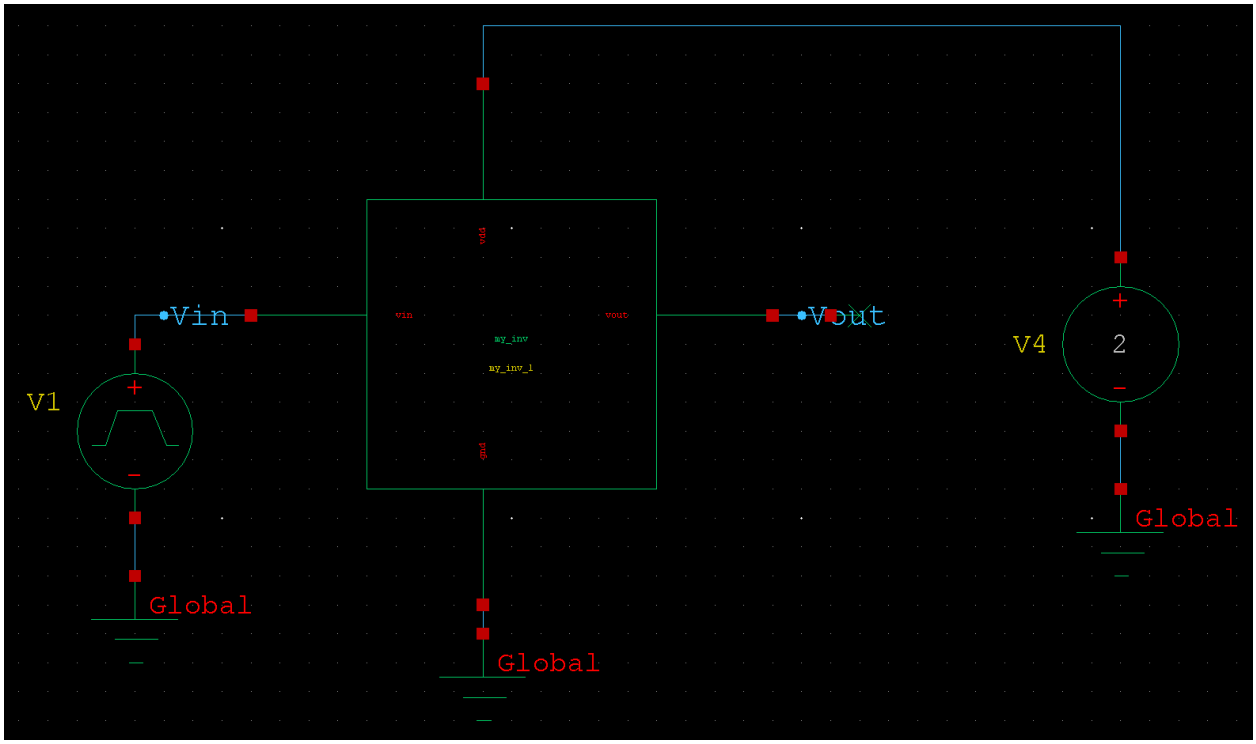


Create the schematic as shown below. Use nmos4 and pmos4 from the devices library. Set the model to nch and pch respectively (the model names come from the ee214b.sp SPICE model file). Use vdd and gnd from Misc library. Add vin and vout ports.



Modify the testbench to include a 2V DC voltage source. Modify the symbol to add pins for vdd and gnd. Add vdd and gnd pins to the veriloga model as well so that it matches the symbol and schematic (all views must be pin-accurate).

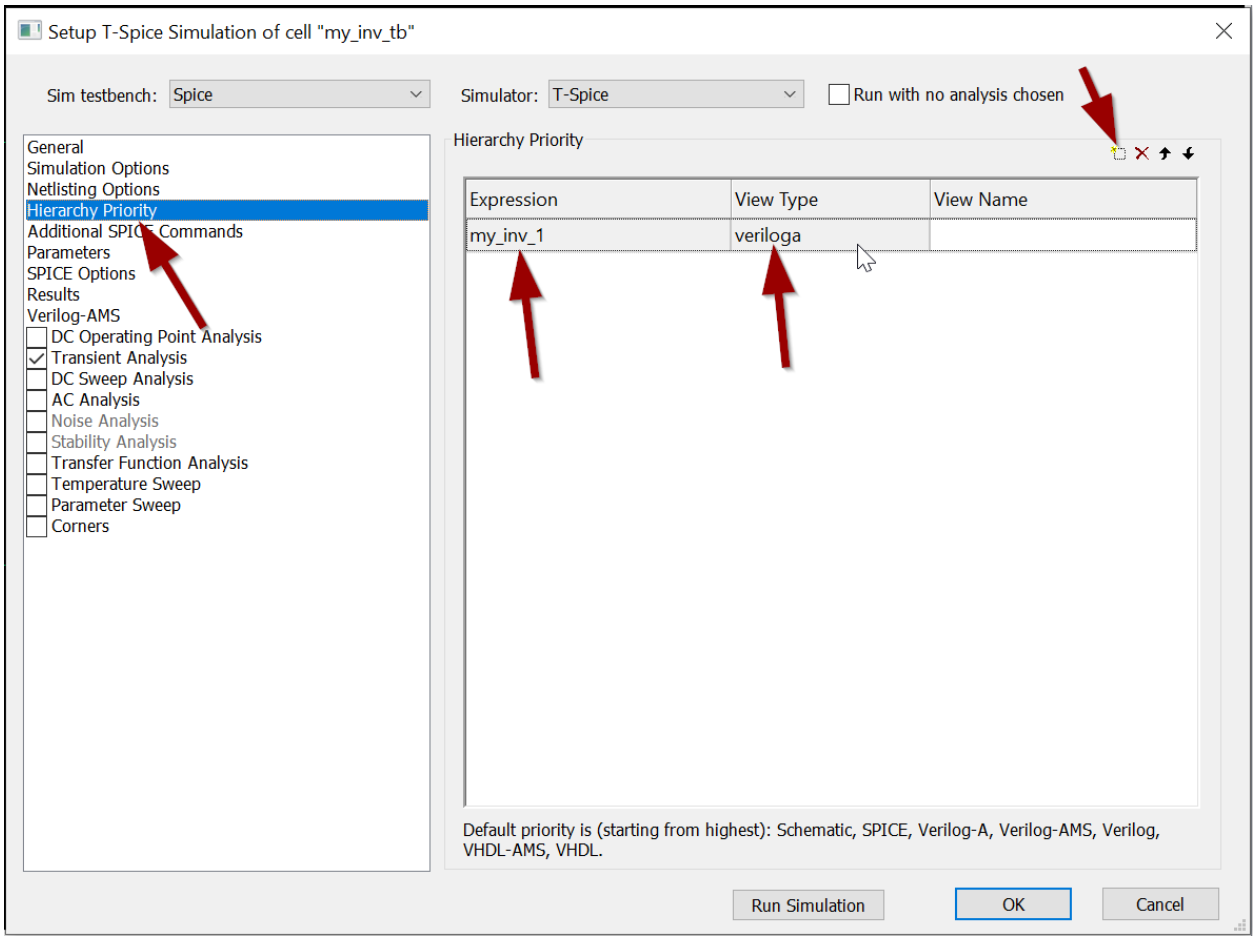
Note: We could have used global gnd and vdd instances to avoid creating new pins for the inverter, however, for some reason this doesn't work properly for v2022 of Tanner.



Note that we didn't include the model file yet. So if we run the simulation it should give an error as below.

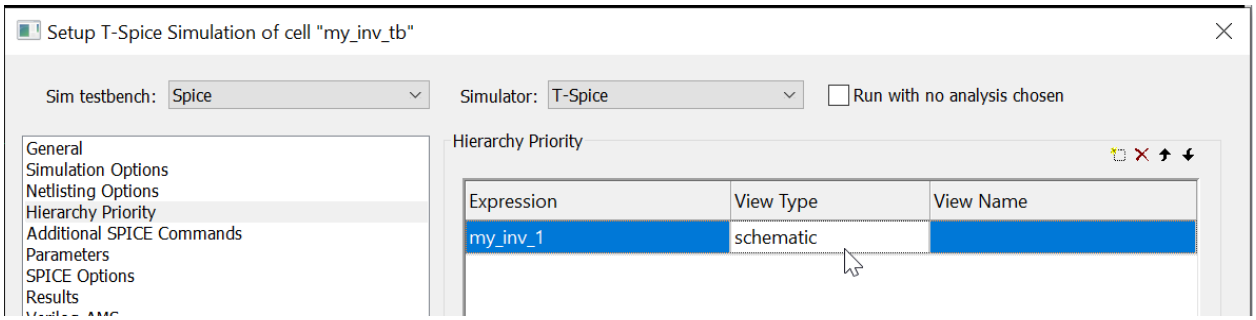
Fatal Error : "my_inv_tb.sp" line 35 Could not resolve model "nch" referenced by "Xmy_inv_1.MV4"
Fatal Error : "my_inv_tb.sp" line 36 Could not resolve model "pch" referenced by "Xmy_inv_1.MM1"

Note that the schematic view was selected by default because it has higher priority compared to the veriloga view. If we want to go back to the veriloga view, we need to change the priority as shown below.

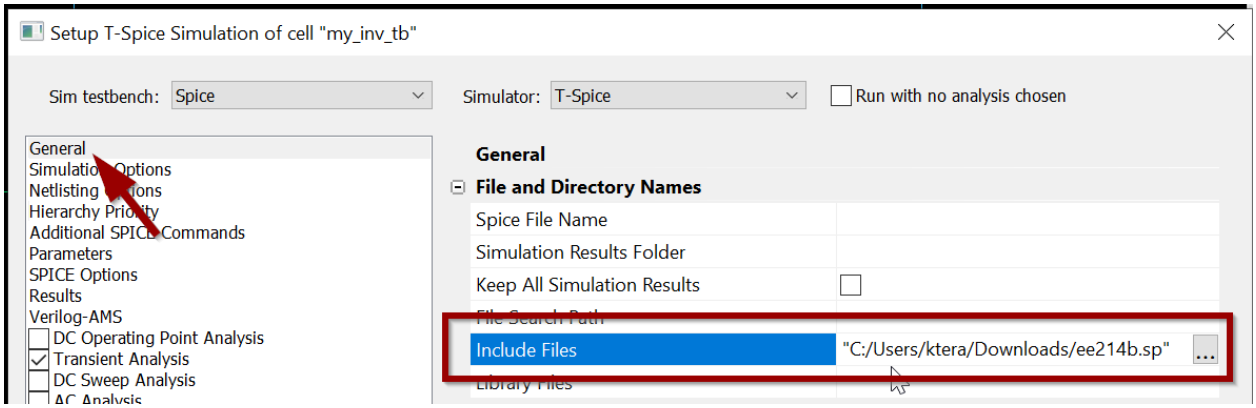


Run the simulation and make sure it now completes successfully.

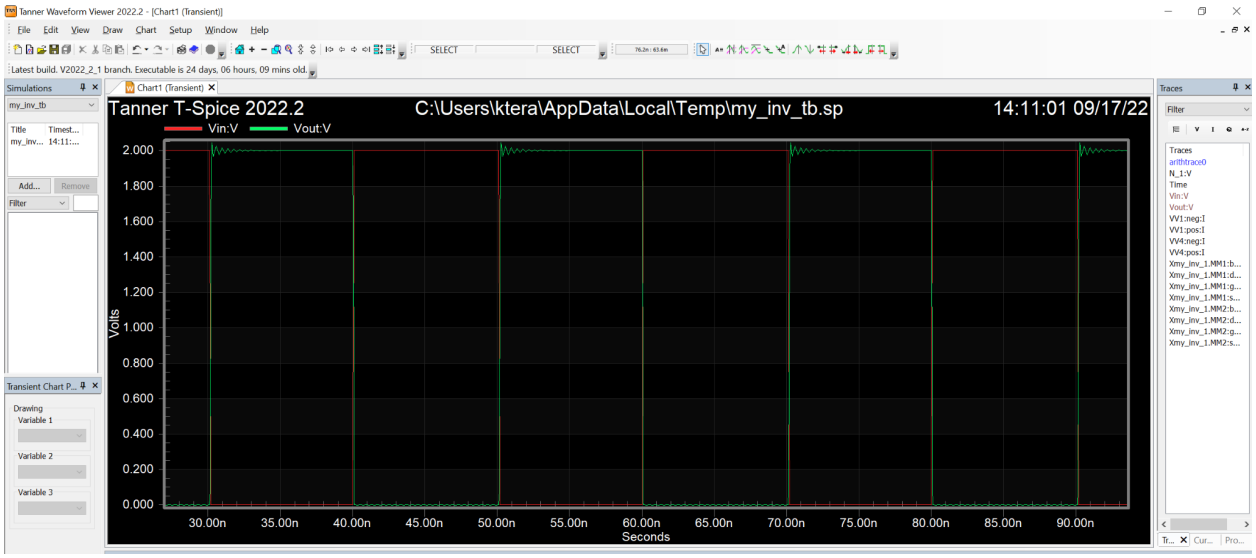
Next, switch to the schematic view.



Now add the model files.



Run the transistor level simulation.



Note the trapezoidal ringing in the output!