Layout Tutorial for Lab 3: Automated Design, Synthesis, and Layout

We won't be doing the layout of each cell ourselves. Instead we will be using "black box cells". The black boxes are provided by CMC and TSMC (CMC provides access to the materials and TSMCs fabrication facility). The black box cells are place holders (look them up in the library manager when you get the chance) and will be filled in by CMC when the design is submitted for fabrication (something we wont be doing). Instead we are going to use Encounter (a cadence tool) to do the following:

- Floor planning,
- Place and route,
- Power distribution
- Clock distribution.

Once we are done we will have a prototype layout for our Verilog module that can be sent to CMC for fabrication (once they replace the black boxes with the real cells).

The layout lab is a continuation of the synthesis work you did last week. Therefore, we will need the synthesised version of your signed serial multiplier. This example will use the synthesized counter from the previous tutorial.

Make sure you have a saved copy of your synthesized module (if necessary run design analyzer again). It is best to save a version of your synthesized module in a separate directory.

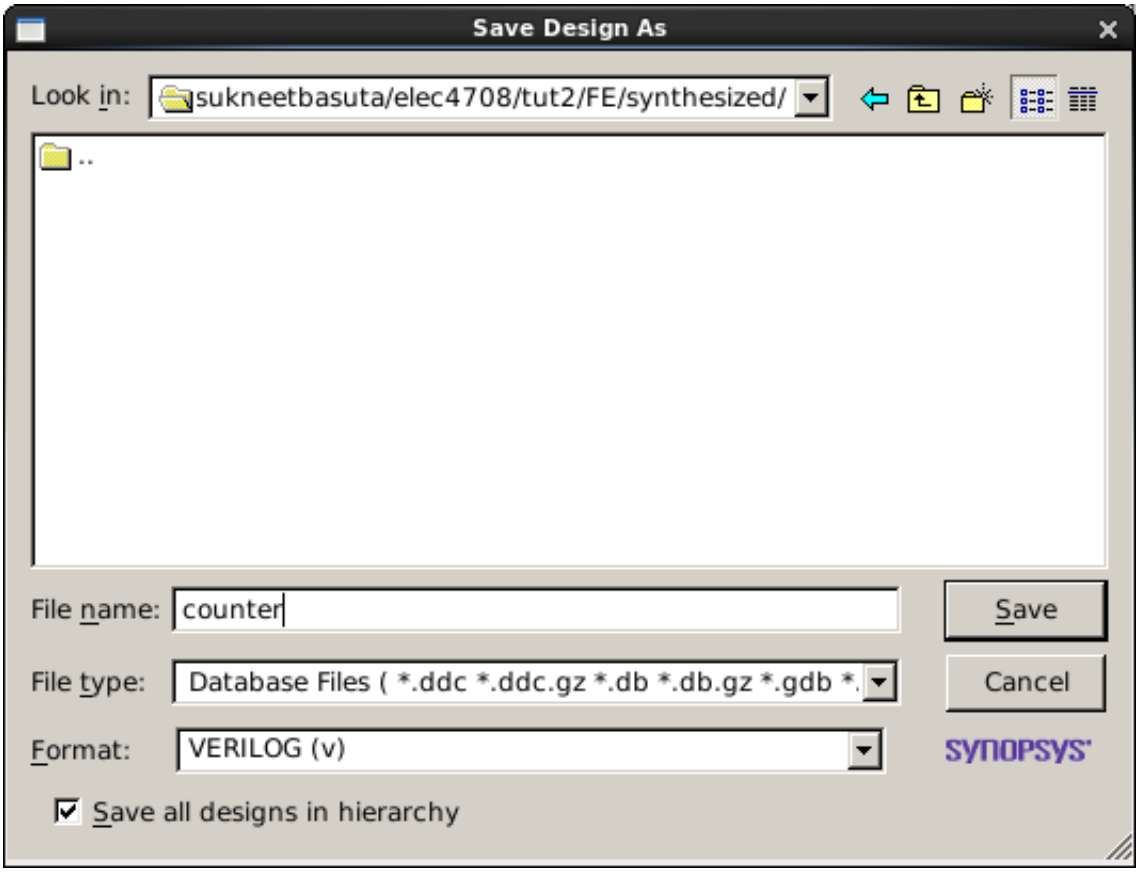
1. Open a terminal console, go to your lab2 directory, and create a constraint file. You may have to change "Clk" to the name of your Clock pin, and "Reset" to the name of your Reset pin.
   ```
   >cd elec4708s
   >cd lab2
   ```
```
>echo -e "set_time_unit –nanoseconds\ncreate_clock -name {Clk} -period 10.0
-waveform {0.0 5.0} [get_ports {Clk}]\nset_false_path -from [get_ports
{Reset}]" > constraint.sdc
```

2. Now create a Clock specification file. You may have to Change "Clk" to the name of your Clock pin.

```
>echo -e "AutoCTSRootPin Clk \nBuffer CKBD1 CKBD8 CKBD16 CKBD20 CKBD24 \nEND" > Clock.ctstch
```

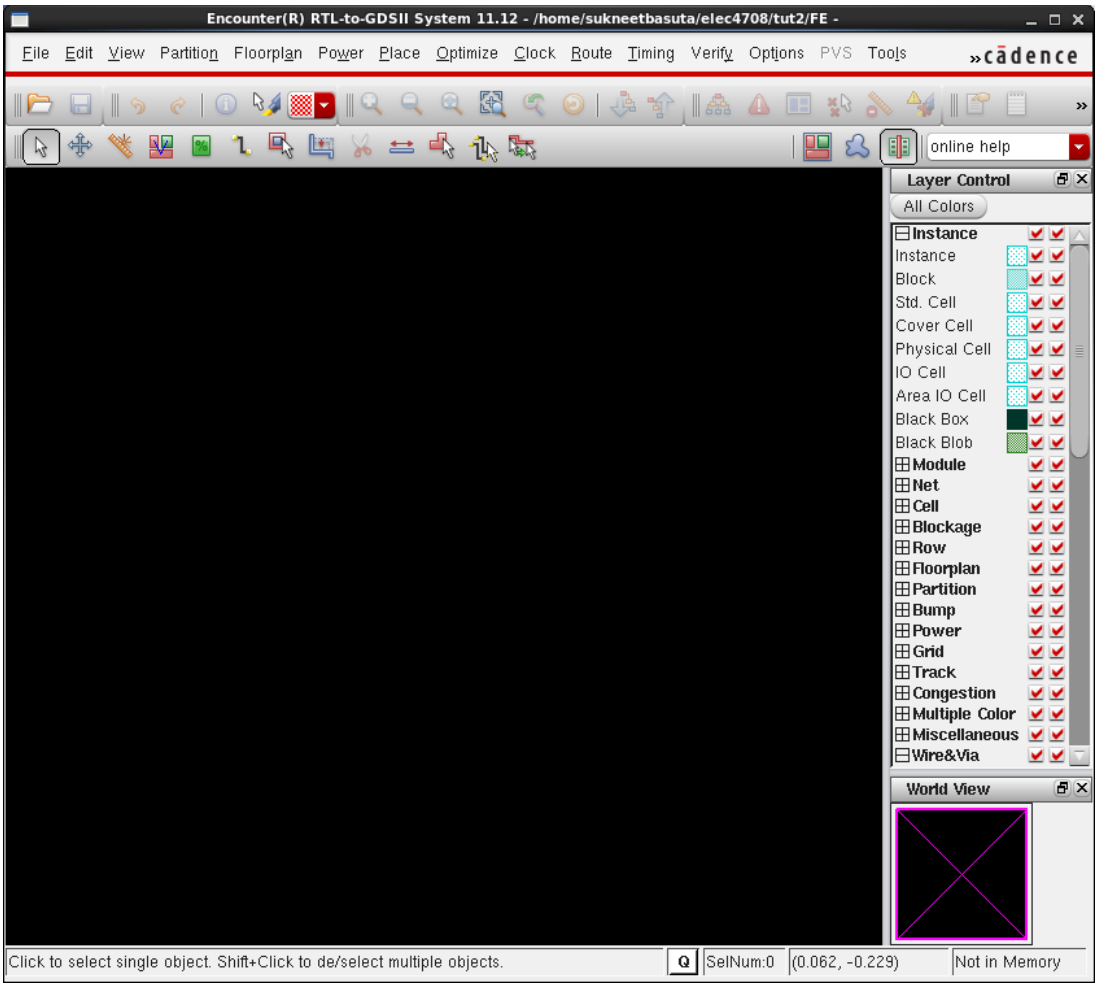
3. Now we want to save a copy of your synthesized module in a new directory but first we need to make a directory to save it to.
   ```
   >mkdir synthesized
   ```
   Now we need to run design analyzer as before (see tutorial for part 2 of lab 3. Make sure your in the right directory) and add a new step at the end as follows:

4. Go back to the top level of the hierarchy. Save the design using File-->Save As. Save the file in the "Synthesized" directory with file name "counter.v". Make sure you save it with the VERILOG format, and save the full hierarchy.



5. Now we can start encounter. In a terminal console make sure that you are in the lab2 directory and type

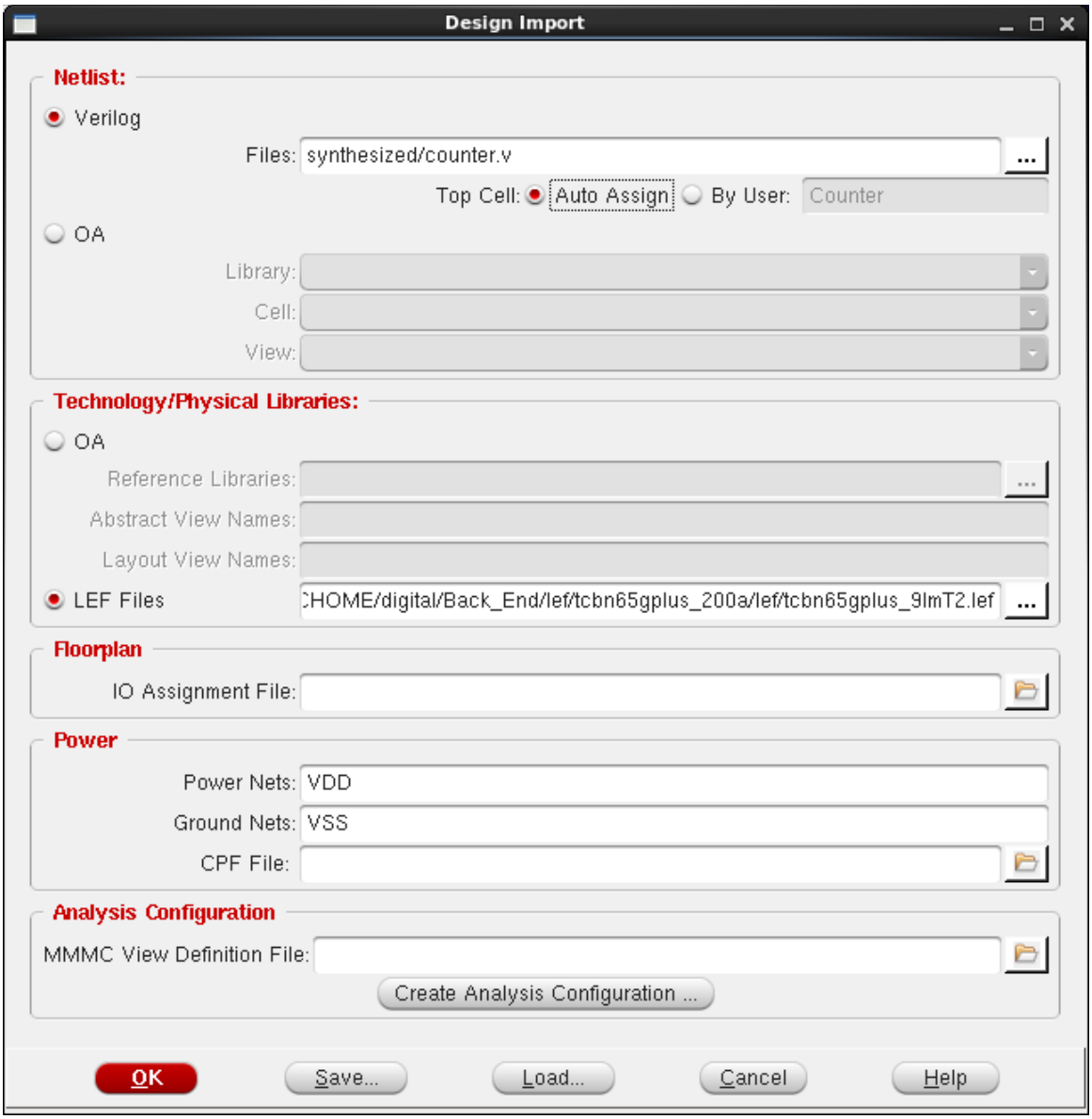
```
>encounter
```



6. In encounter we now need to import the design that we saved in the synthesized directory. Go to File -> Design Import, and fill in the pop-up window as shown below (modify the file locations to suit where your files are saved). Set VDD as the Power net, and VSS as the ground net.

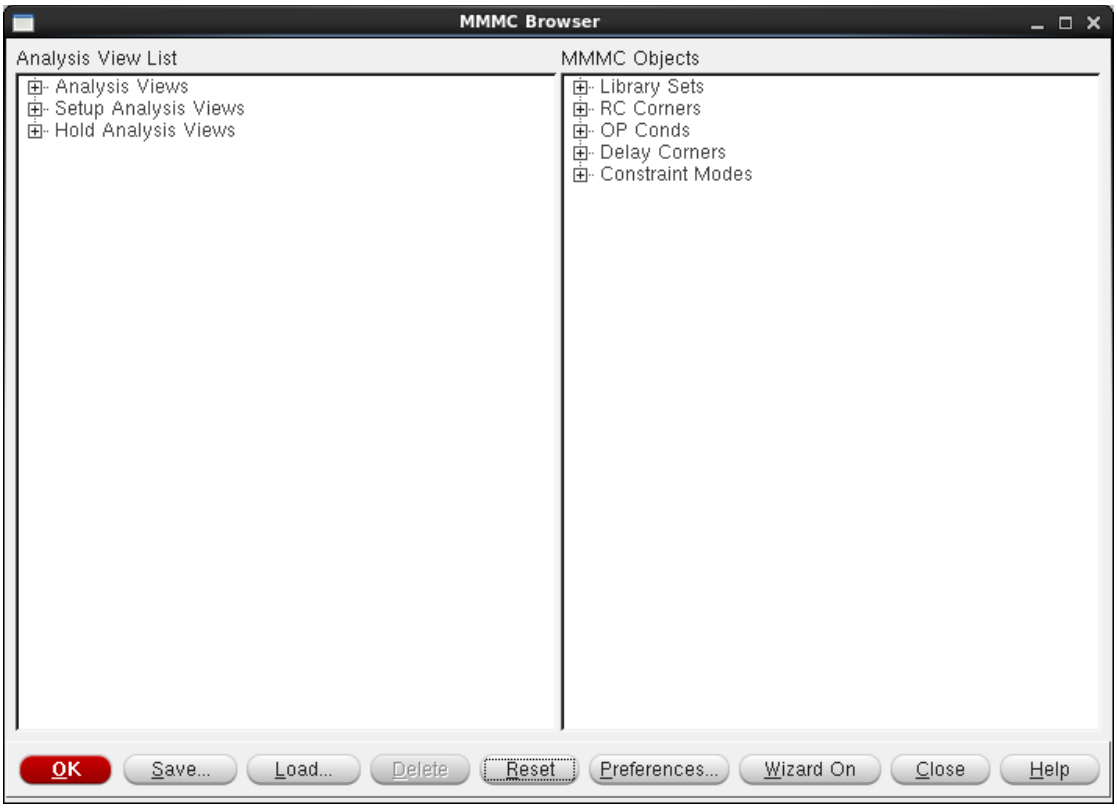
   Use the following path for the LEF Files:

   `/CMC/kits/tsmc_65nm_libs/tcbn65gplus_200a/TSMCHOME/digital/Back_End/lef/tcbn65gplus_200a/lef/tcbn65gplus_9lmT2.lef`

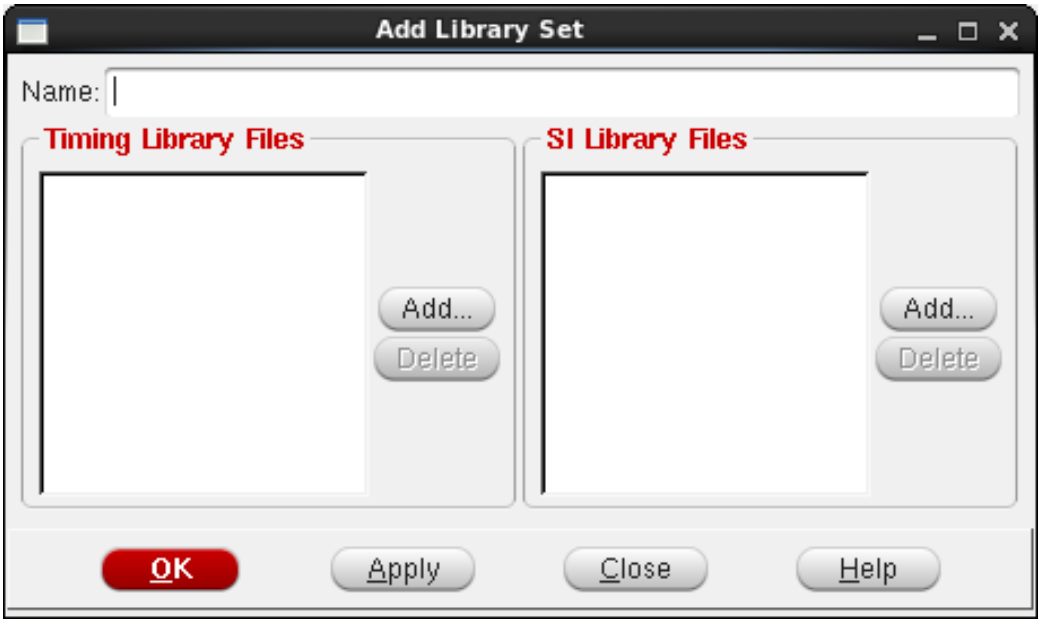


Hit OK.

7. Now lets specific the timing libraries. Select **Timing->Configure MMMC**. The MMMC browser window will popup.

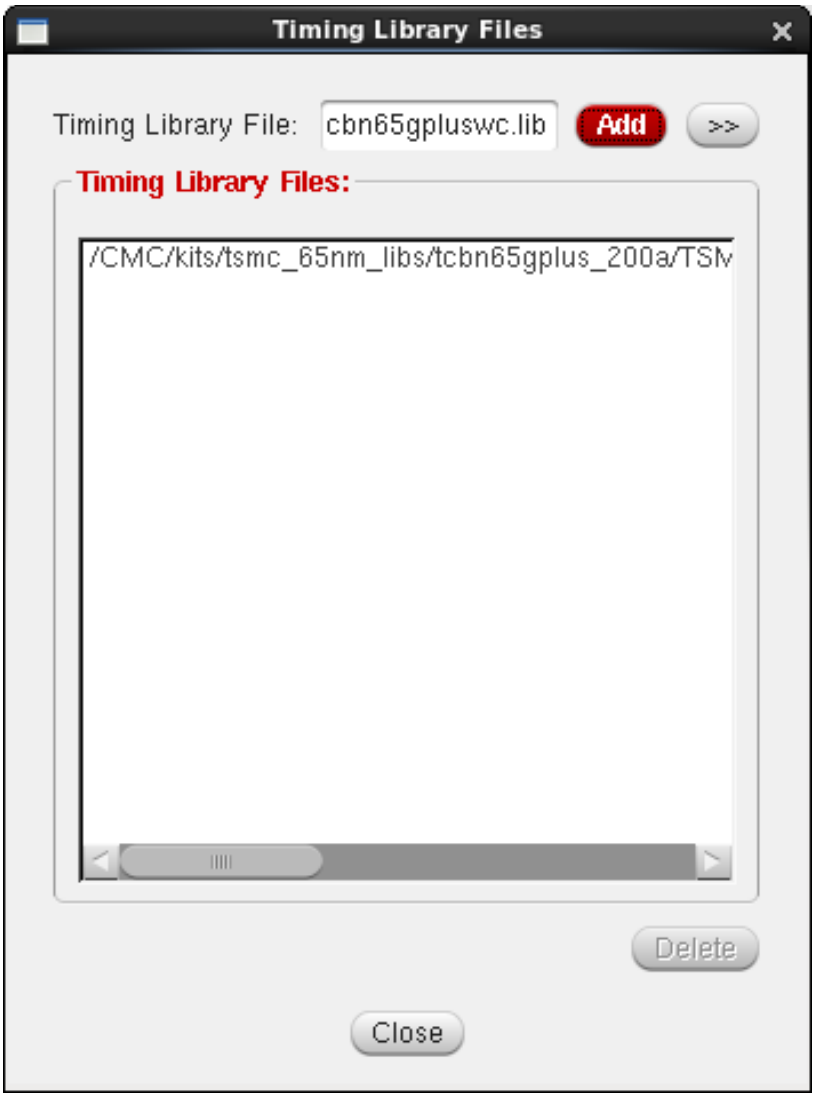


Double click on **"Library Sets"** at the top. A window asking for a library set should popup.



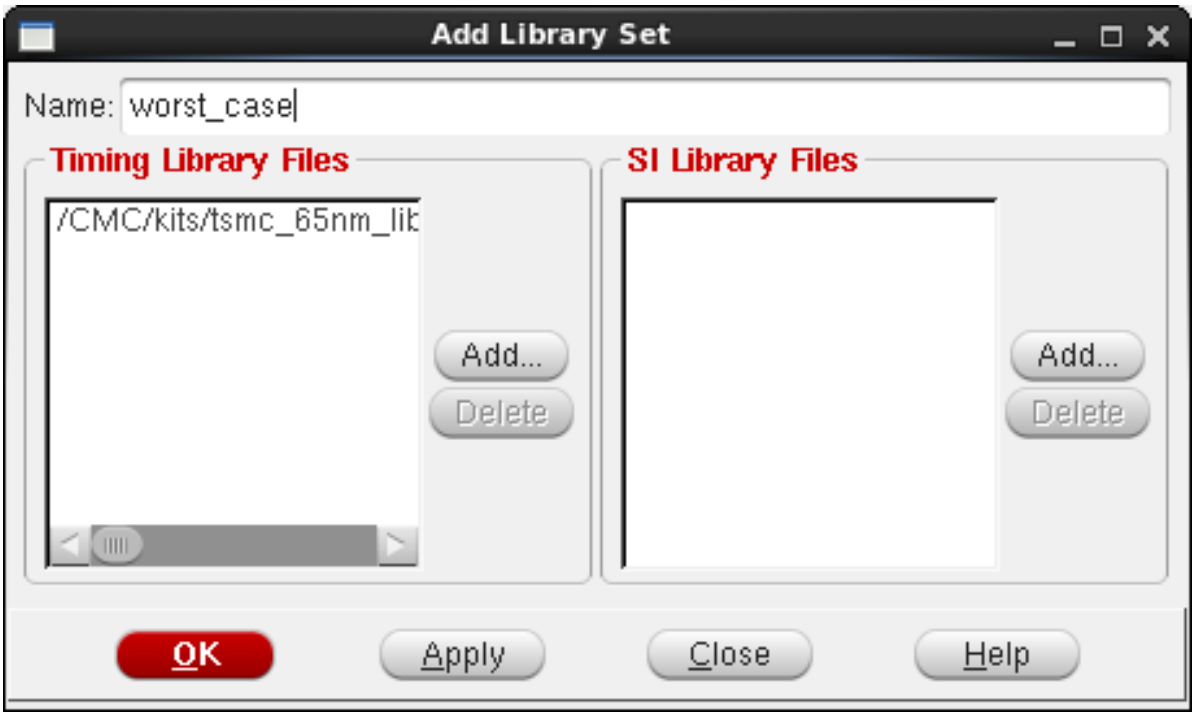
Hit the **"Add"** button under the Timing Library Files section. Add the following path:

`/CMC/kits/tsmc_65nm_libs/tcbn65gplus_200a/TSMCHOME/digital/Front_End/timing_power_noise/NLDM/tcbn65gplus_140b/tcbn65gpluswc.lib`



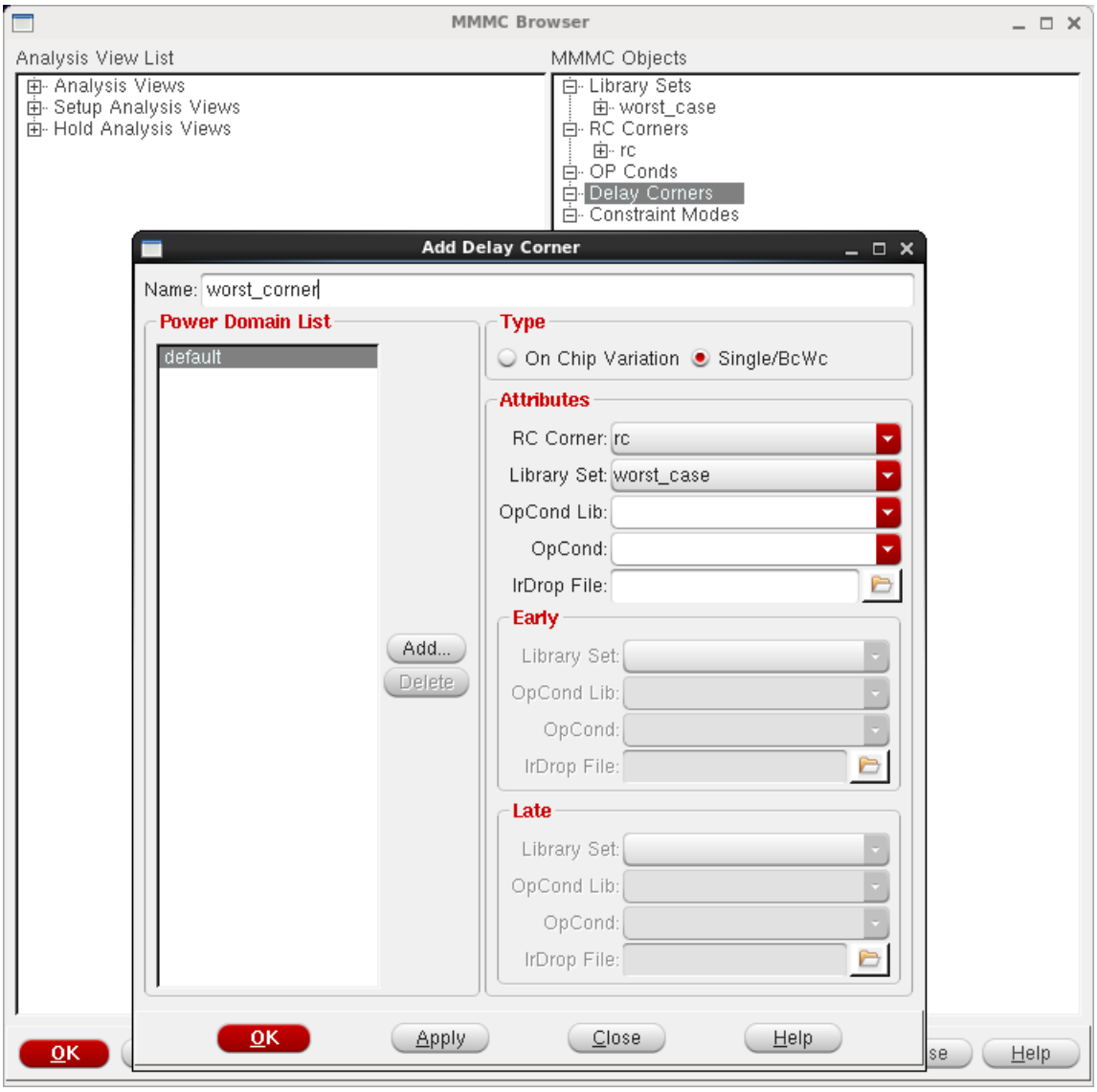
Add the library and hit close.

Give the library set a name (let's call it "worst_case" since we are using the worst case timing library) and Hit Ok.

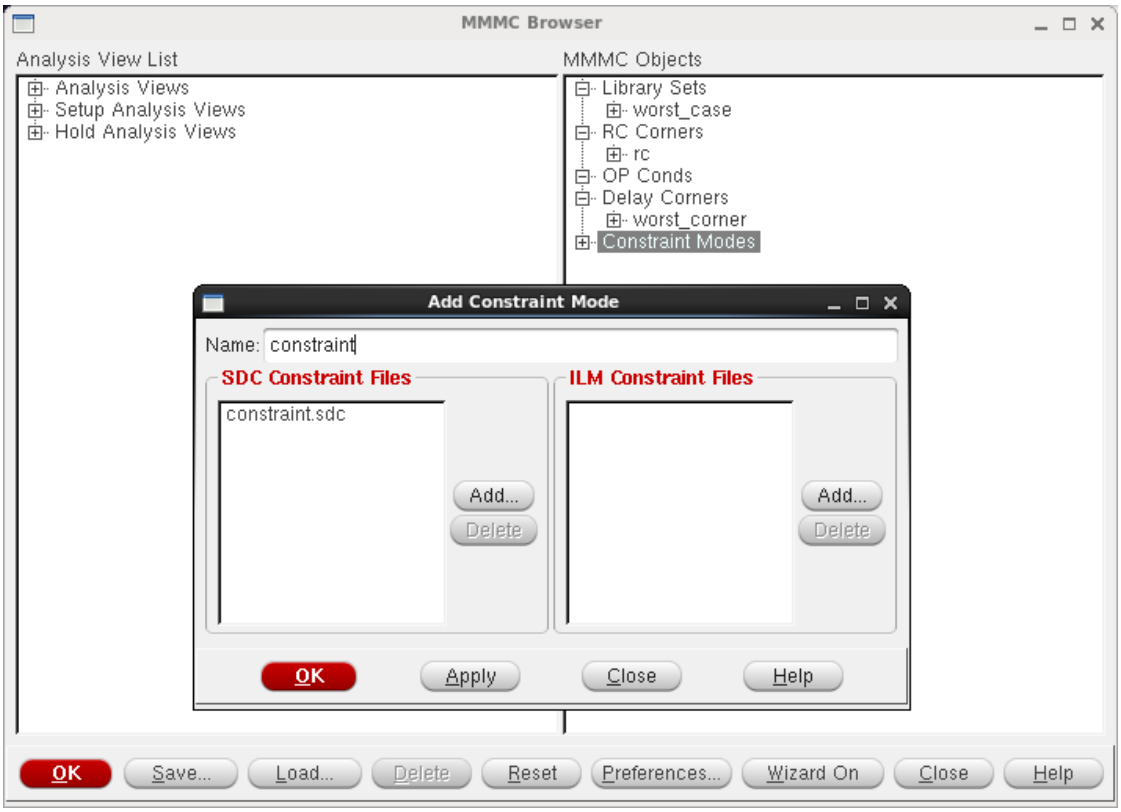


Next double click on **RC Corners**, set a Temperature of 25, and give it a name. Hit Ok.

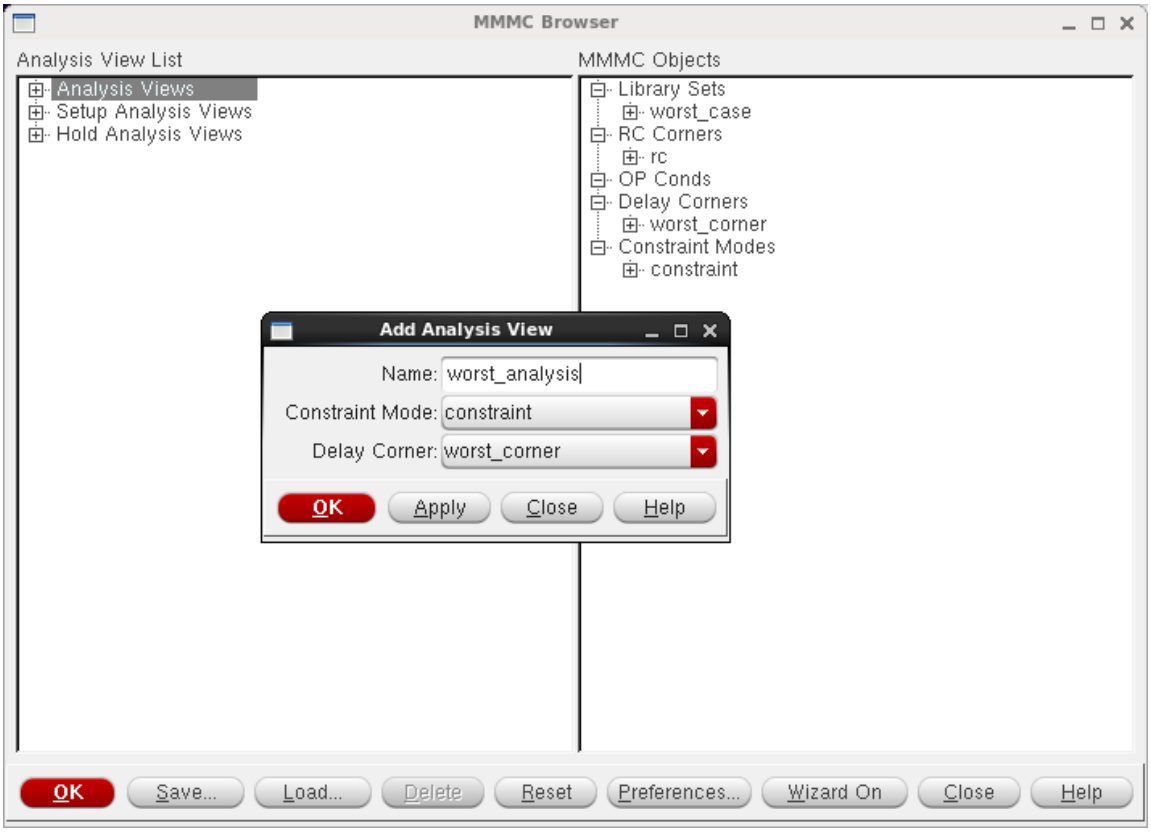
Double click on **Delay Corners**, set the RC Corner to the one you just defined and the library set to the library you defined earlier, and give it a name. Hit Ok.



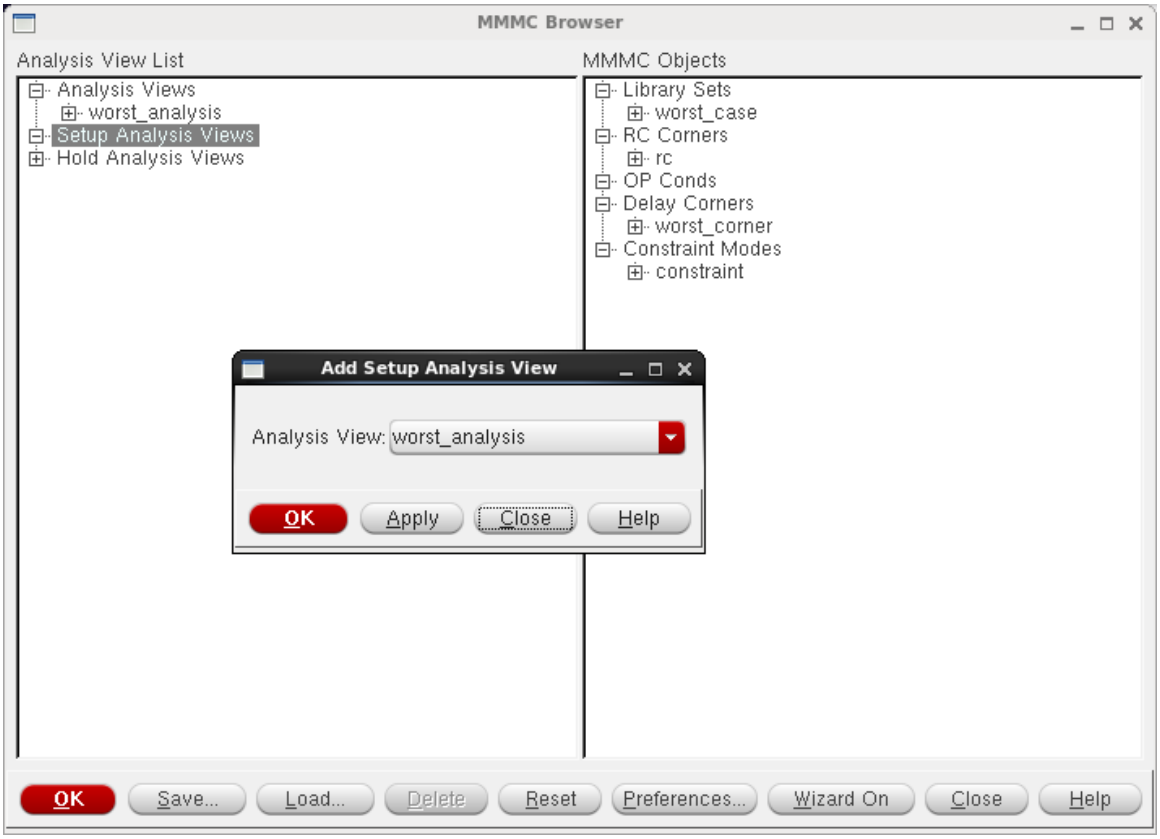
Double click on **Constraint Modes**, add the constraint file you created in Step 1, and give it a name. Hit Ok.



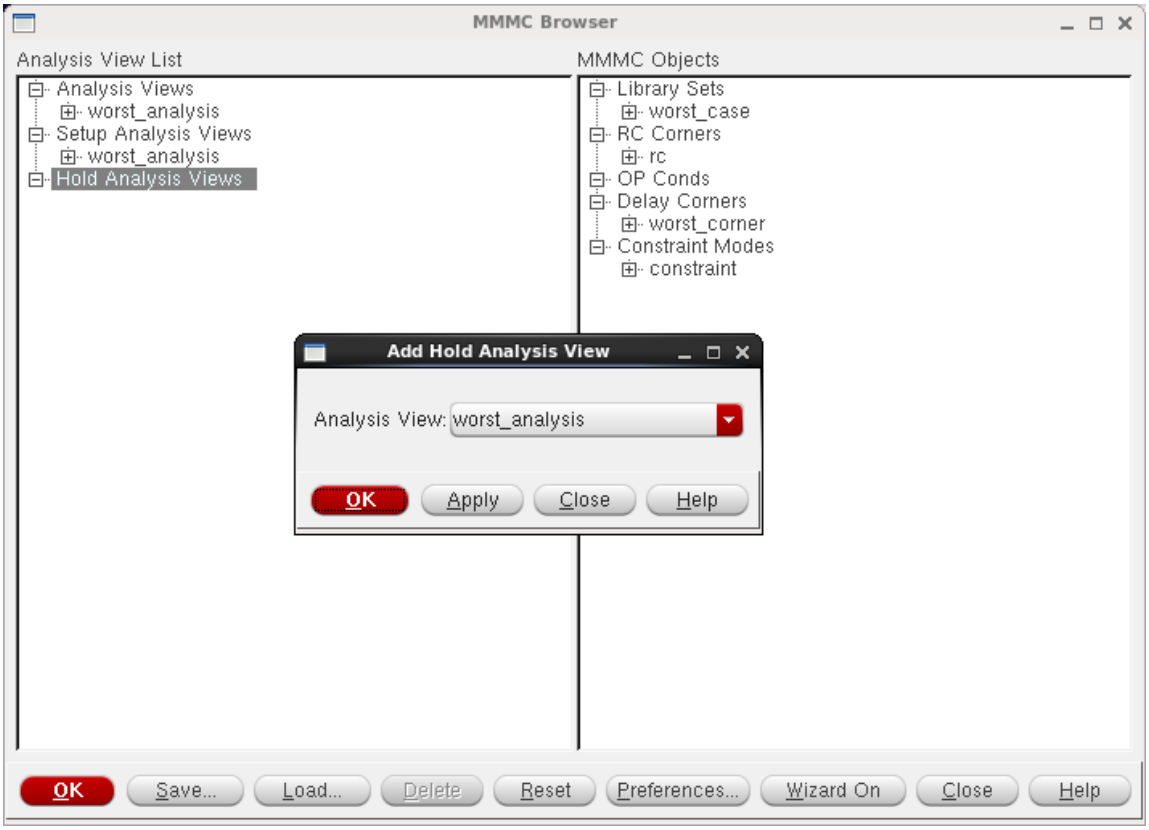
Double Click on **Analysis Views** and give it a name. Hit Ok.



Double click on **Setup Analysis Views** and select the analysis view you just created. Hit OK.

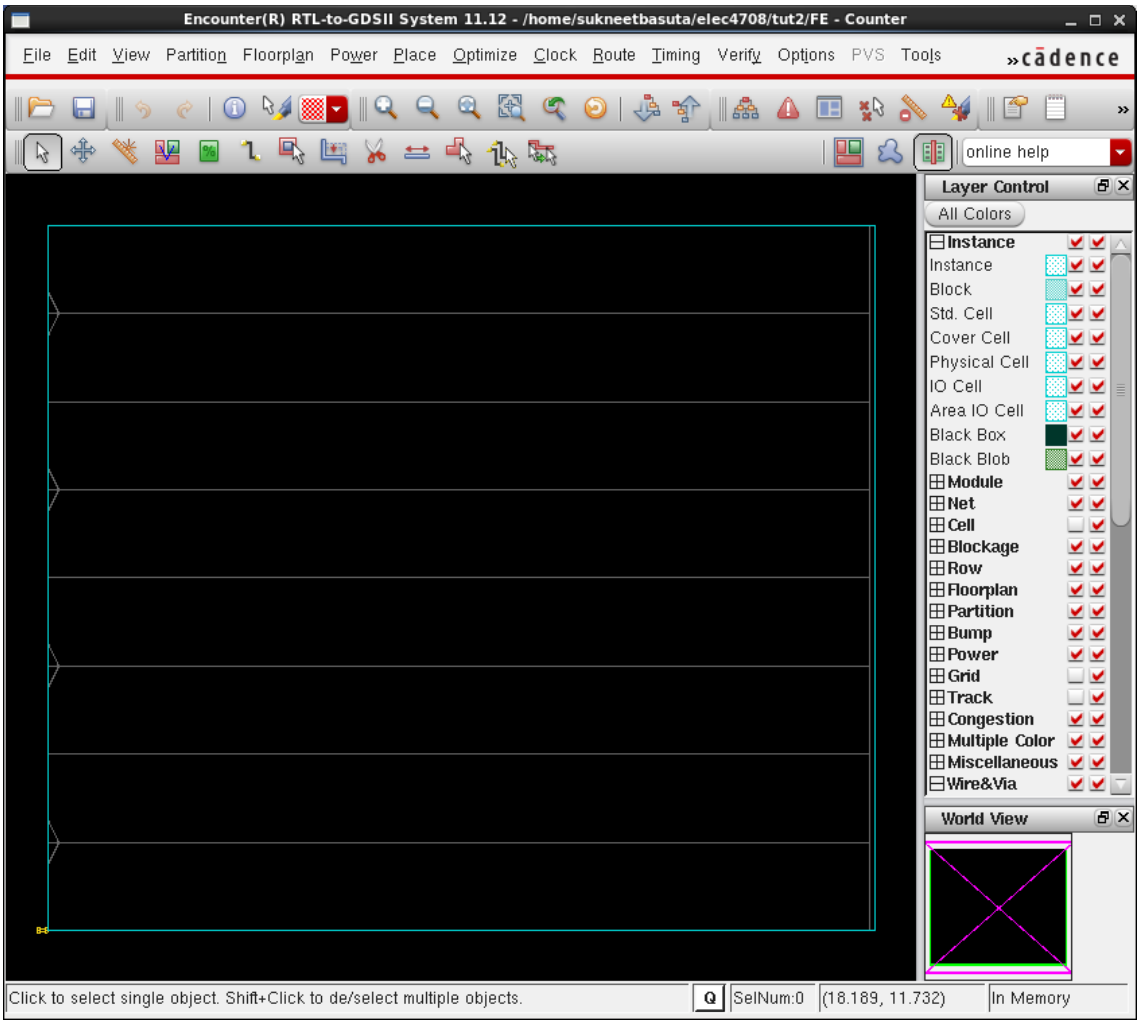


Double click on **Hold Analysis Views** and select the analysis view you just created. Hit OK.



Hit **OK** on the MMMC browser window.

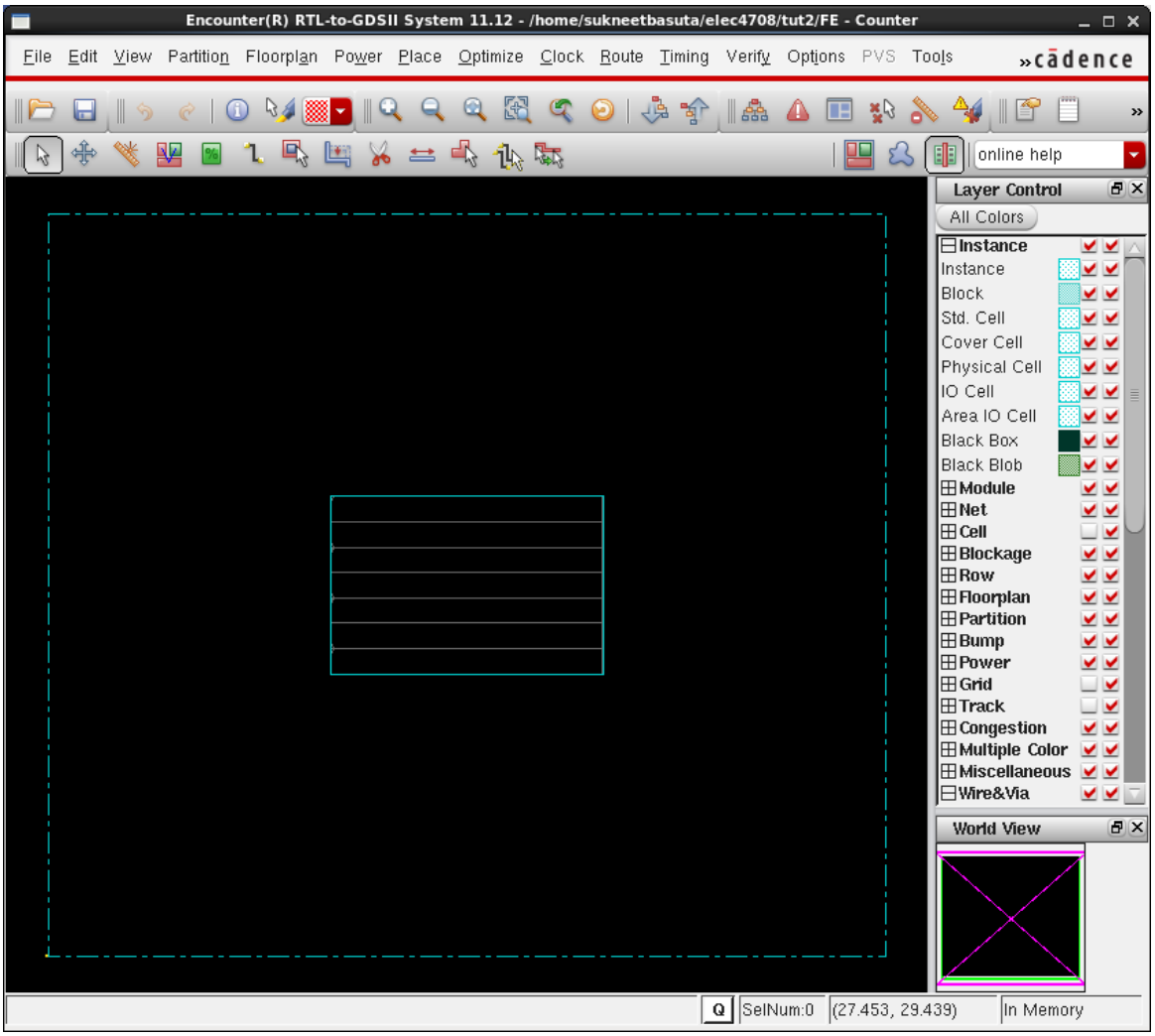
8. We can now see the initial floor plan (you may have to zoom out with **View->Fit** or with the Zoom fit  button on the top toolbar) and the module in the bottom left of the window (the counter is very small and doesn't take up much room in the module view).



9. The next step is to specify the floor plan. We need to leave enough space so that the router will be able to place all the metal interconnects, but not so much that we waste space. We also want to have a large enough space for any buffers that might be needed during optimization. A decent number for core utilization is somewhere between 0.5 and 0.7 (The default is most likely fine).

   We also need to decide how much space we want to leave around the core. Keep in mind that we want to add our power rings so we might need something like 15 - 50 microns of space (for the counter we will use 20 microns). From the encounter window select **Floorplan-> Specify Floorplan**.

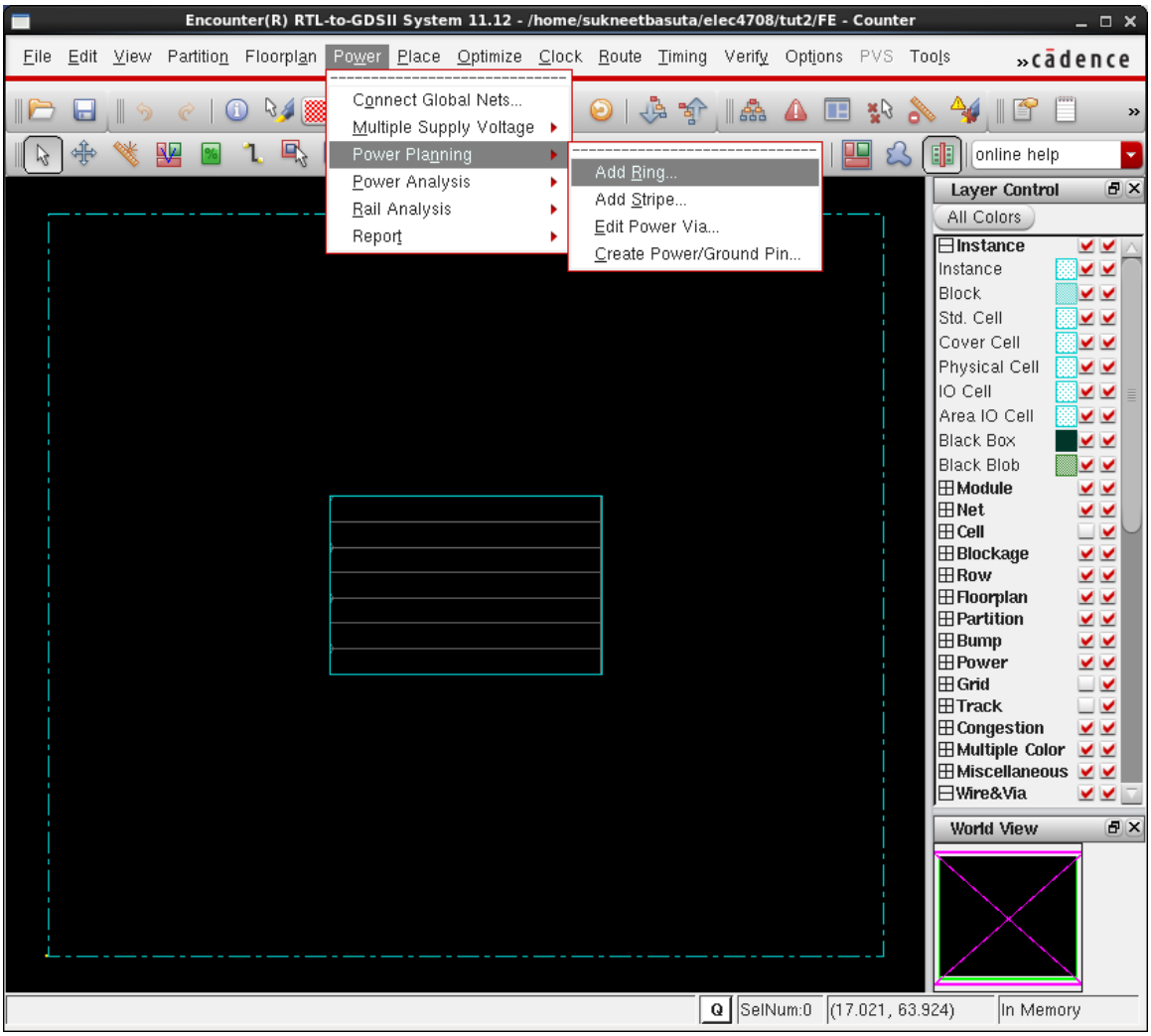
After you click “OK”, the encounter window should now look like this



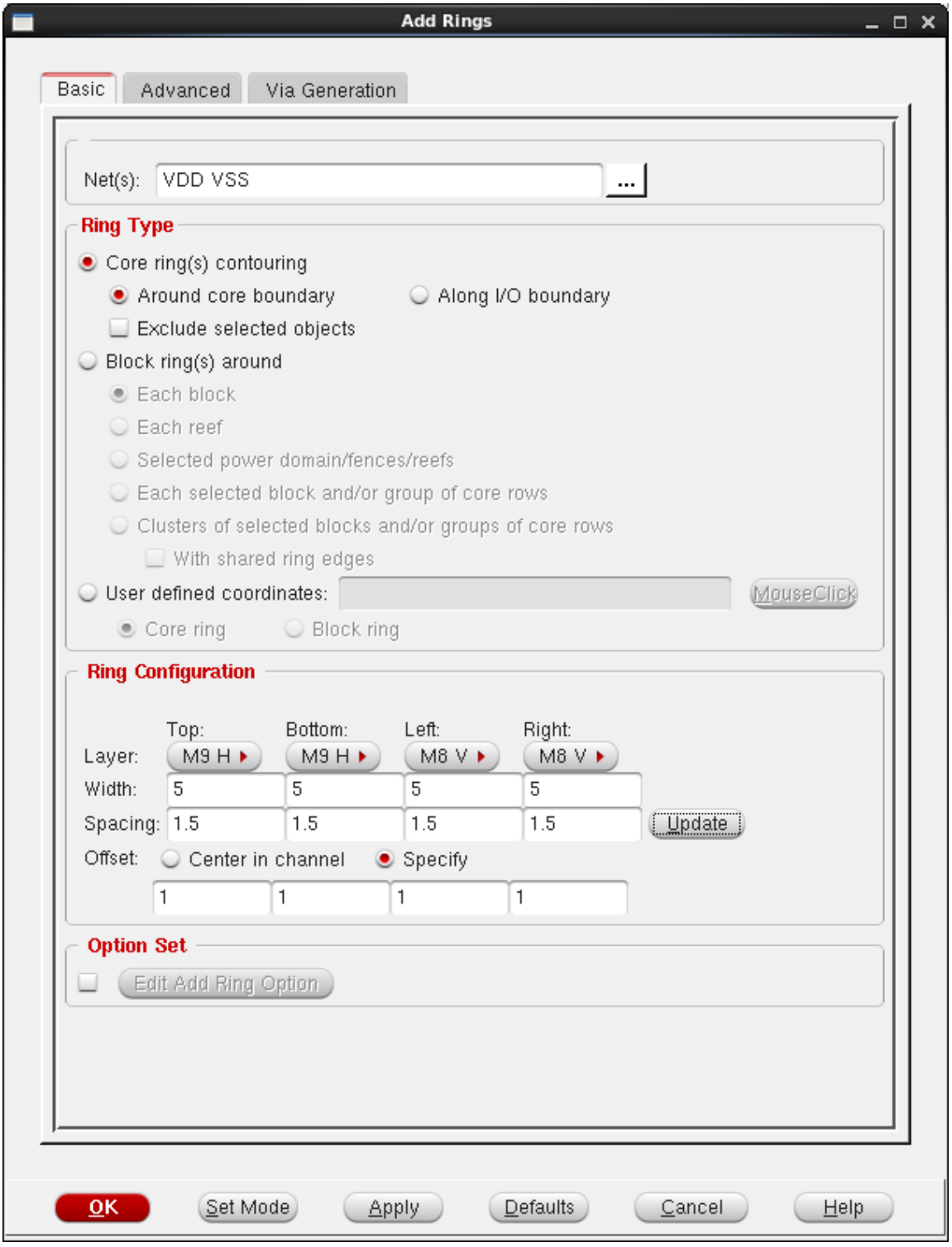
We can see that the chip has been divided into three sections. You can play around with some of the Floorplan numbers to get different sizes, size ratios, core boundary sizes, etc.

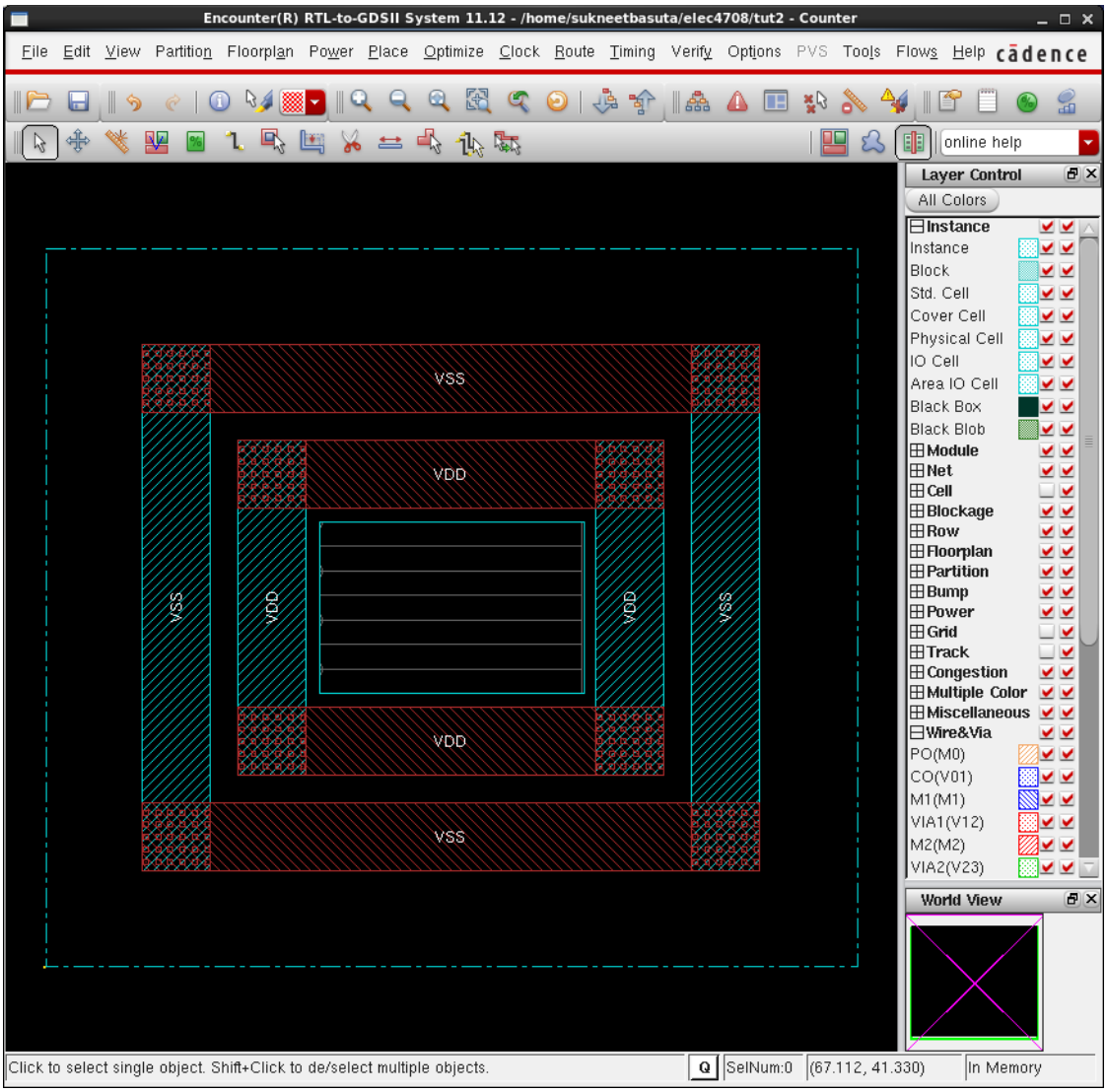
10. Now is probably a good time to save your work. It's also probably a good idea to save often. Select the **File -> Save Design**. Save your design with the Encounter Data Type (OA doesn't restore properly). You can restore a saved design by selecting **File-> Restore Design**. When you restore the design, you may have to select the physical view button  in the top toolbar. You can save your layout in your lab2 library with view layout so you can run simulations with it later in Cadence if you wish. If that doesn't work, select Encounter.

11. We can now add the power rings around the core. Select **Power->Power Planning -> Add Ring…**

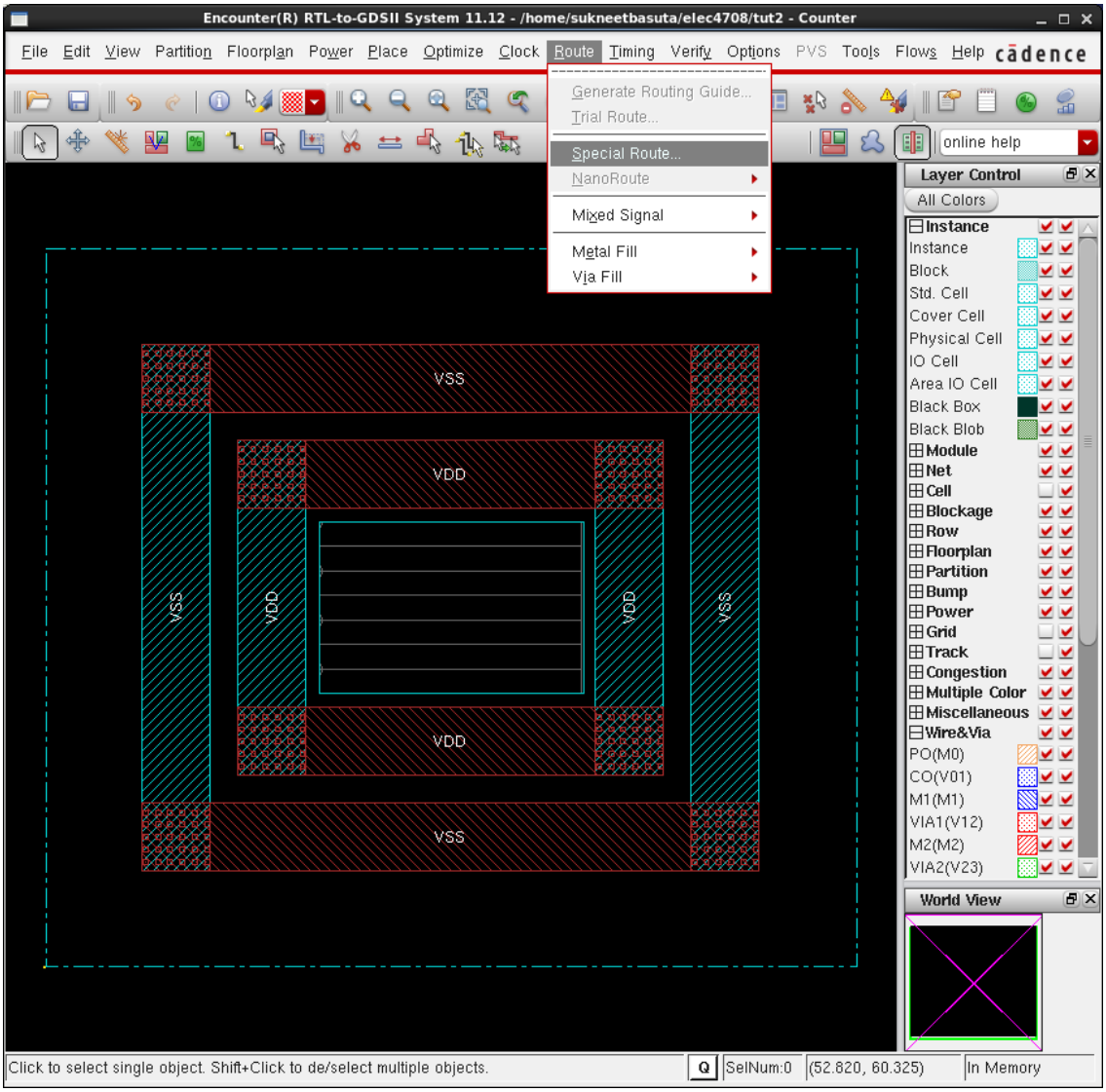


12. Select the VDD and VSS Nets by clicking on the "…" button. Make the power lines 5 microns wide and use Metal 9 for top and bottom and metal8 for left and right. Click on "Update". Set an offset of 1 micron. Click "OK". The width of the power lines is determined (somewhat) by the size of the chip. A much bigger design would need wider lines.

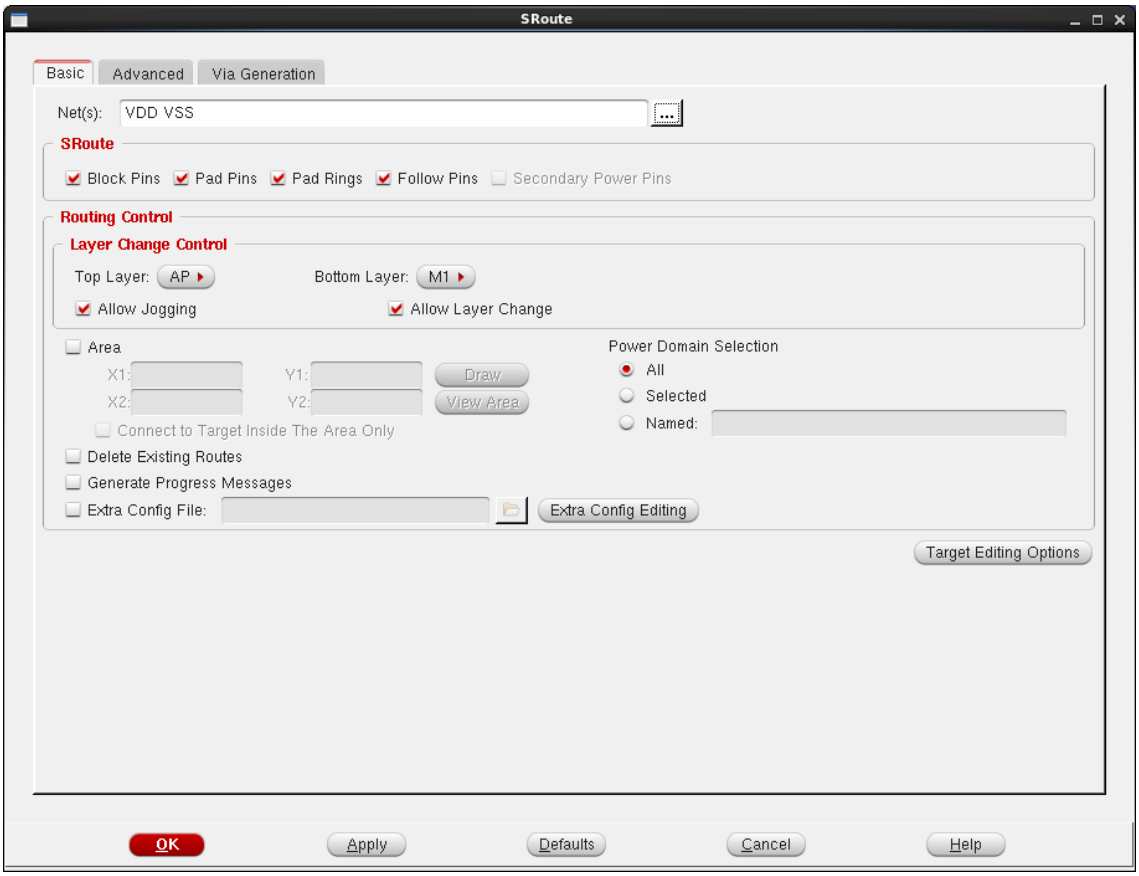




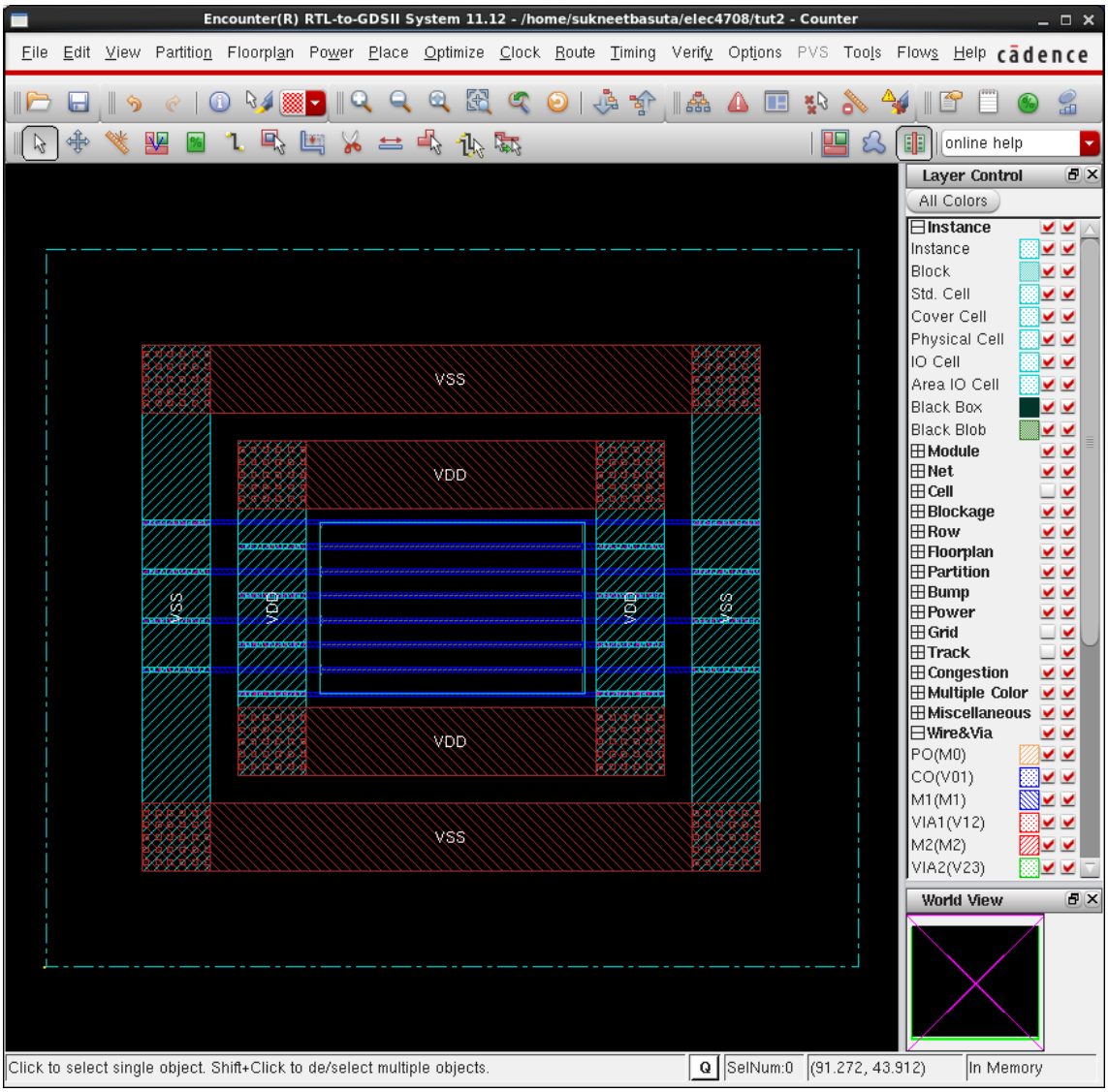
13. Now we need to add the power routing. Select **Route -> Special Route**



14. In the SRoute window, select the VDD and VSS nets by clicking on the "…" button. The rest of the defaults should be fine.



When it’s done, your layout should look similar to the layout below. There are alternating power stripes between the power rings.

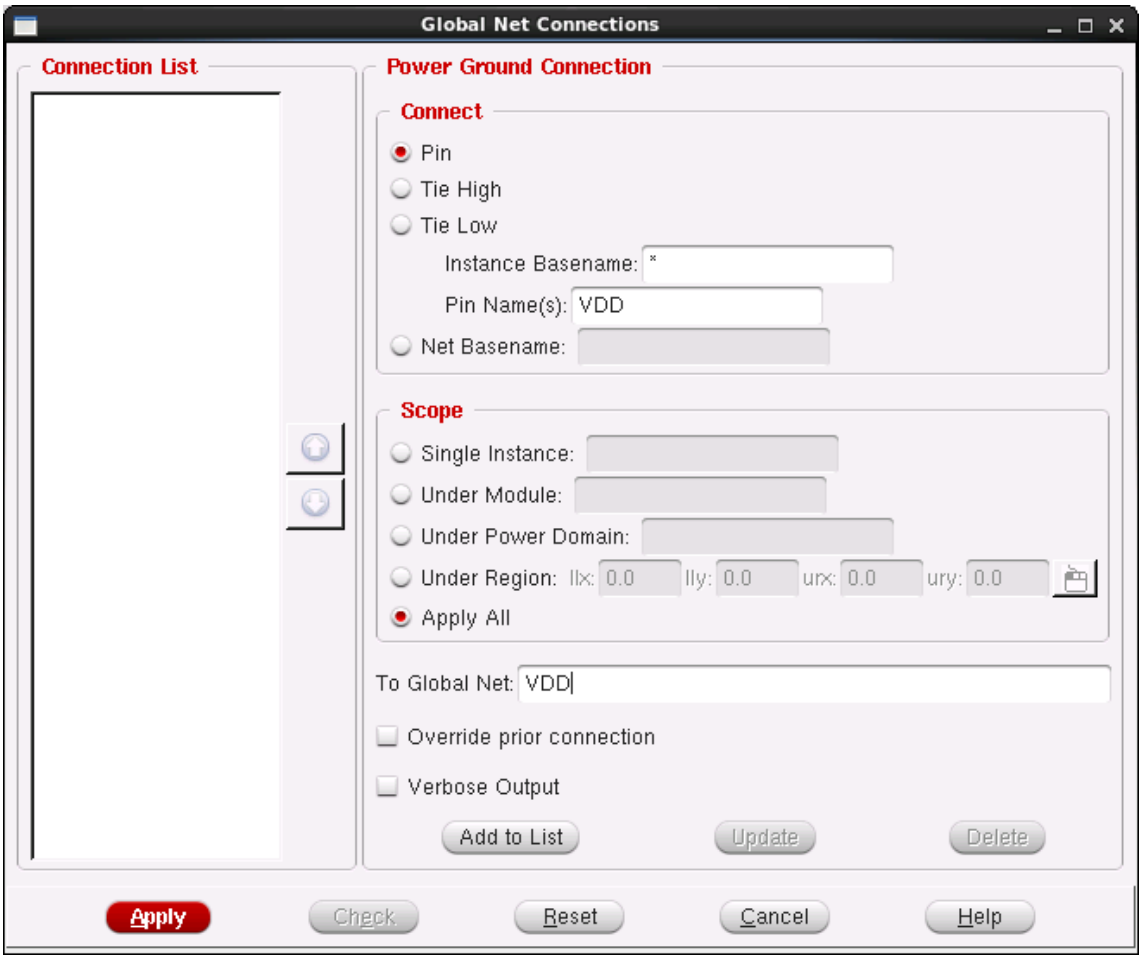


15. Now that we have routed our power wires, we need to define global net connections so that that power planning, power routing, detail routing, and power analysis function work correctly.

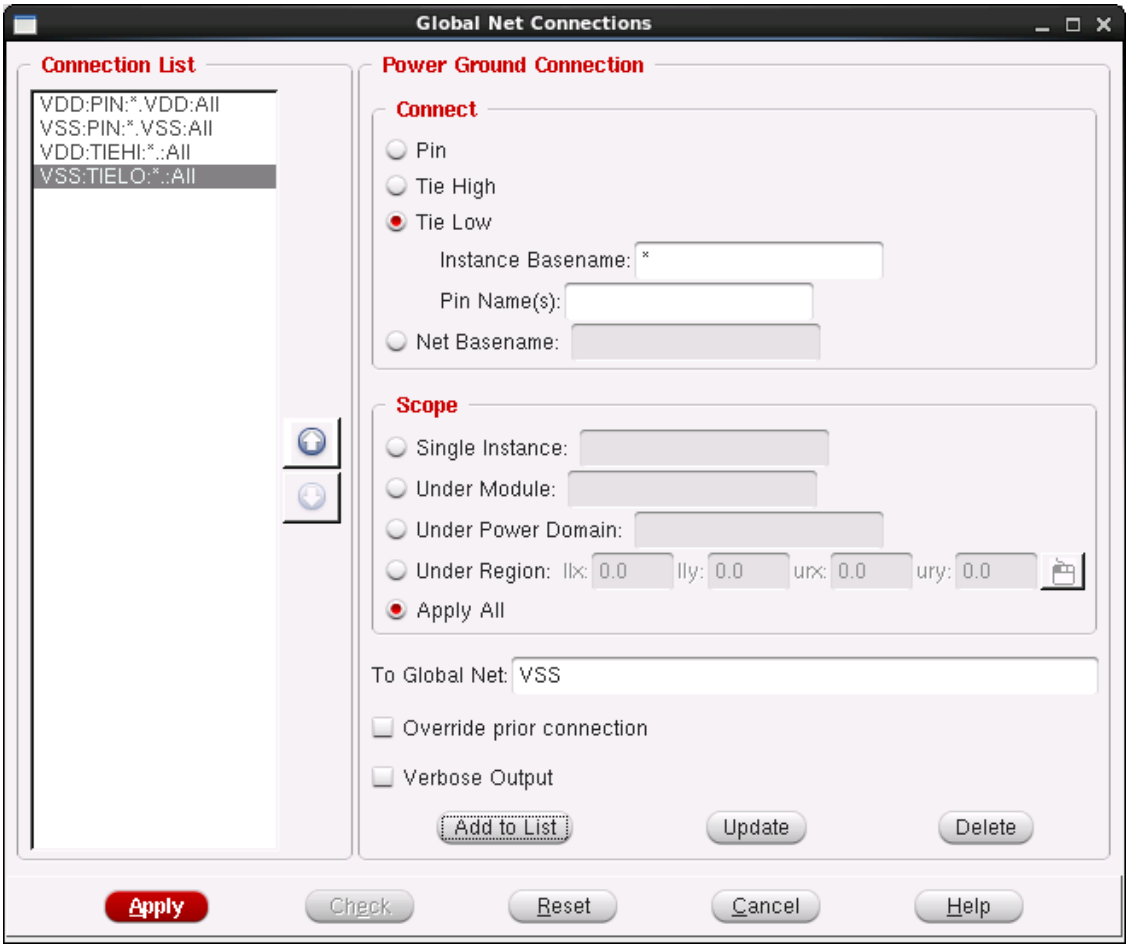
    Select **Power -> Connect Global Nets**. The Global Net Connections window will popup
    We want to add the following connections:

    - Pin VDD -> Global Net VDD
    - Pin VSS -> Global Net VSS
    - TIEHI -> Global Net VDD
    - TIELO -> Global Net VSS

    Fill in the VDD connection like the following and do the same for VSS:

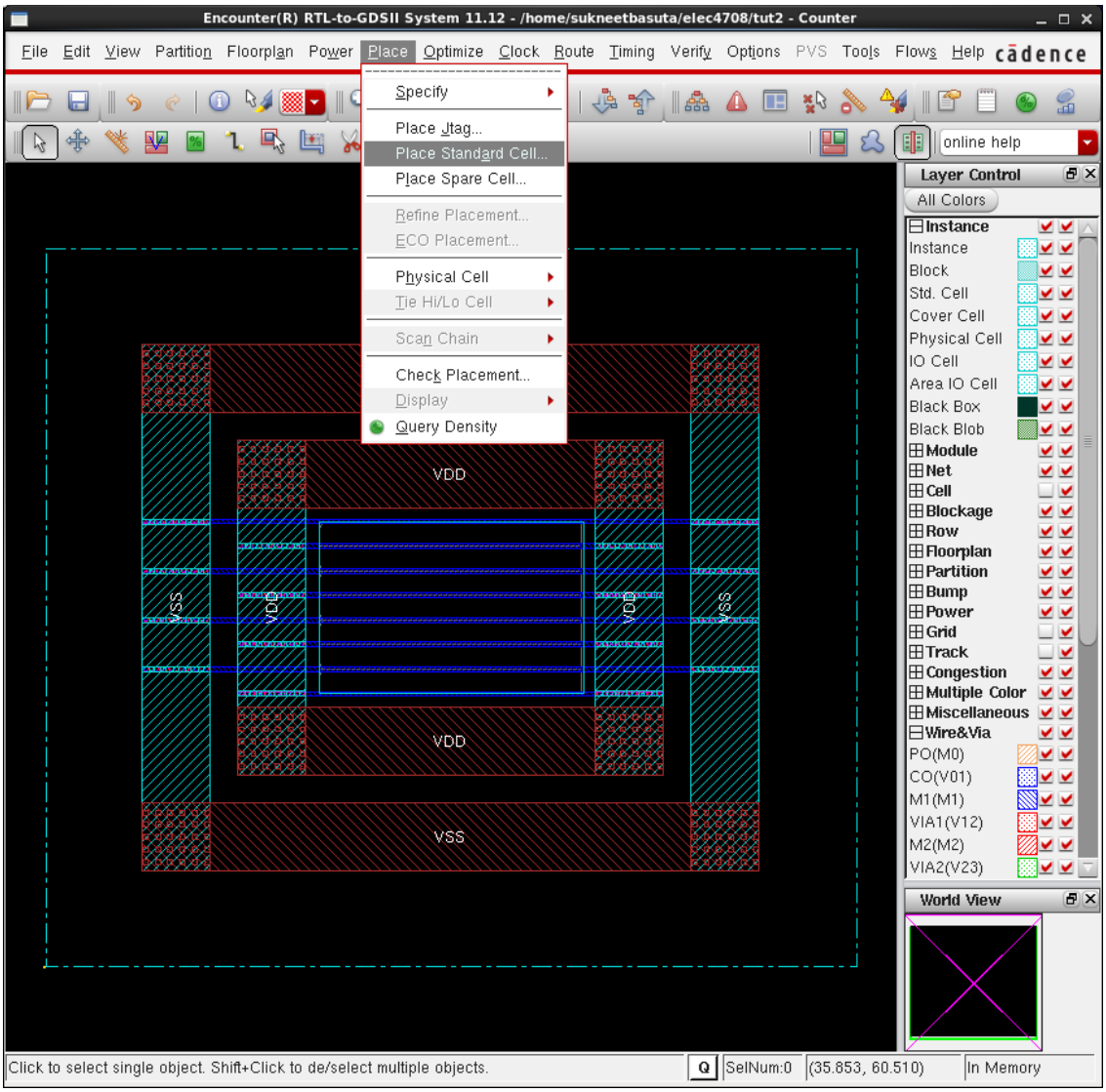

Global Net Connections
Connection List
Power Ground Connection
Connect
Pin
Tie High
Tie Low
Instance Basename: *
Pin Name(s): VDD
Net Basename:
Scope
Single Instance:
Under Module:
Under Power Domain:
Under Region: llx: 0.0 lly: 0.0 urx: 0.0 ury: 0.0
Apply All
To Global Net: VDD
Override prior connection
Verbose Output
Add to List
Update
Delete
Apply
Check
Reset
Cancel
Help

The TIEHI and TIELO connections are done like the following:

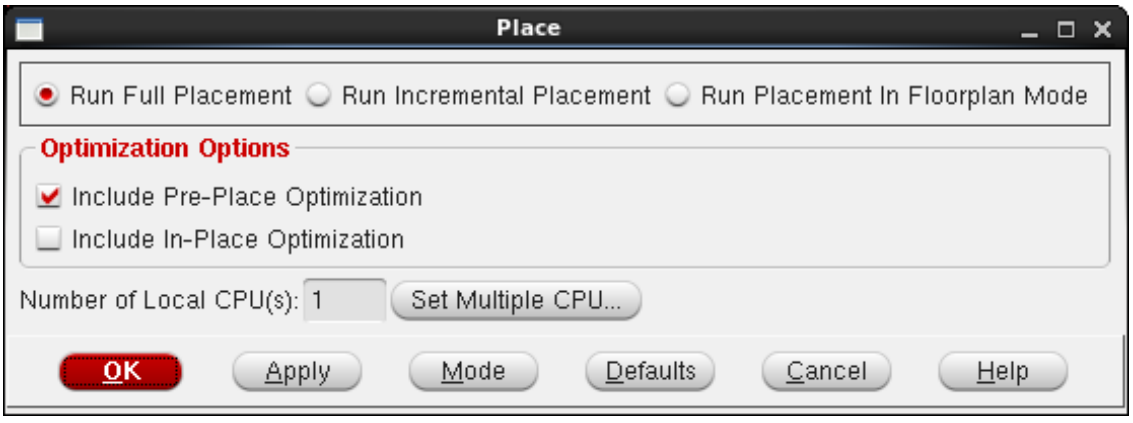


Once done, Hit Apply and close the window.

16. We can now place our cells. Select **Place -> Place Standard Cell…**

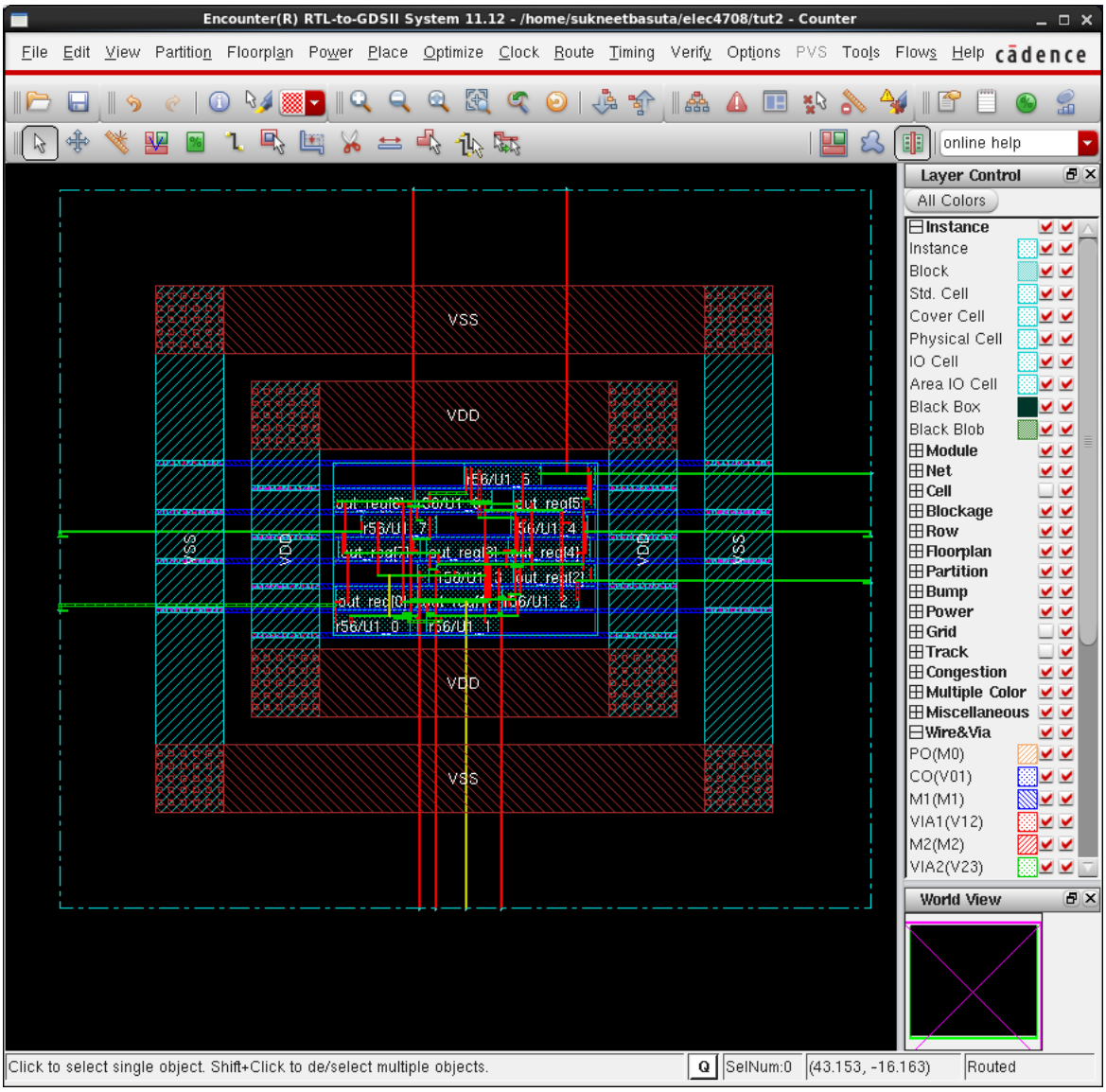


You should get the following window. The defaults are fine.

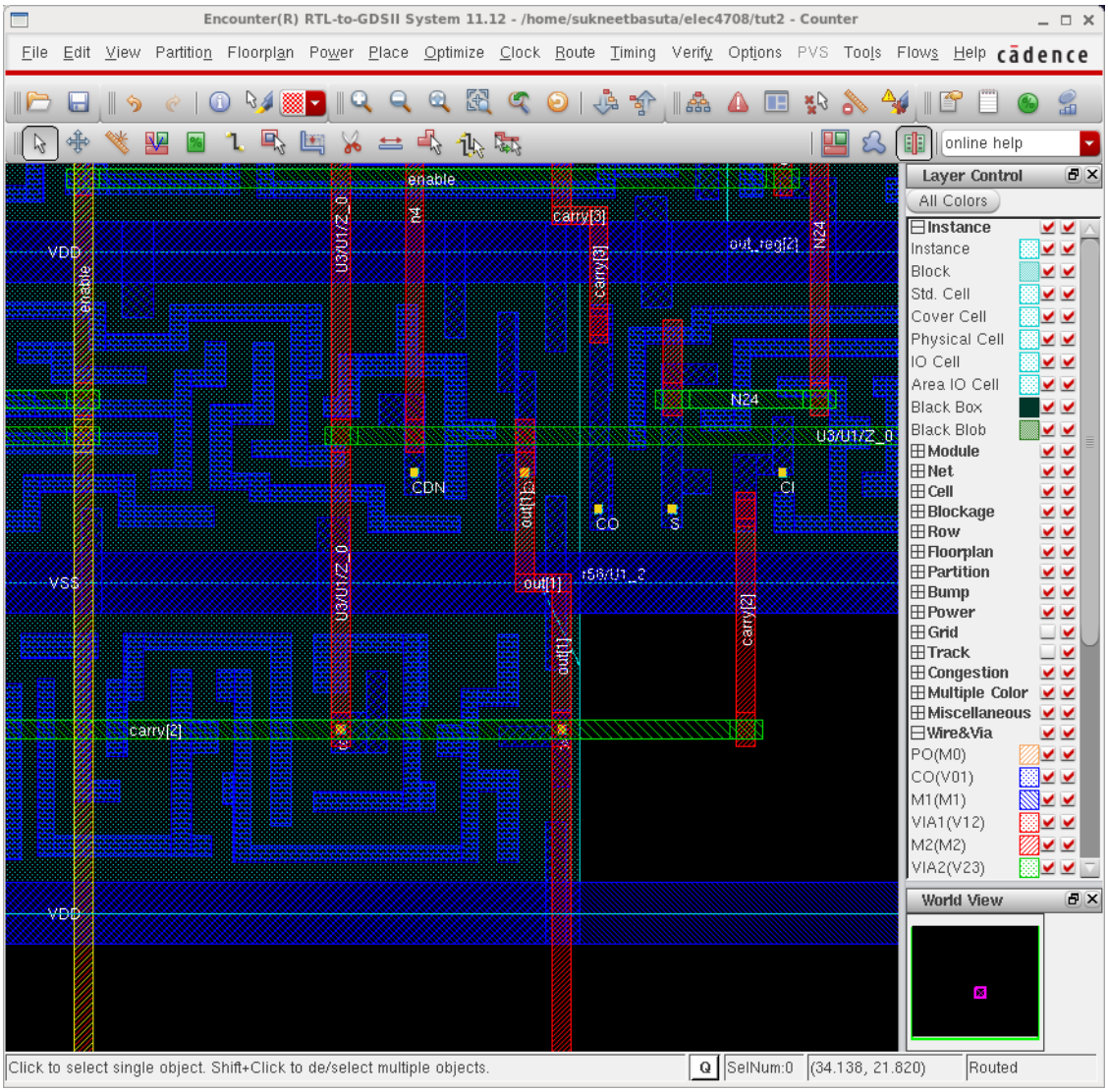


17. If you do not see the standard cells, you can zoom in and out with the mouse wheel, which will refresh the view. You can see that the input and out pins were added wherever routing was

convenient. If you want to specify the location of the pins, you can change them by selecting Edit->Pin Editor. We do not have specifications for the pin locations, so we will leave them where they are.

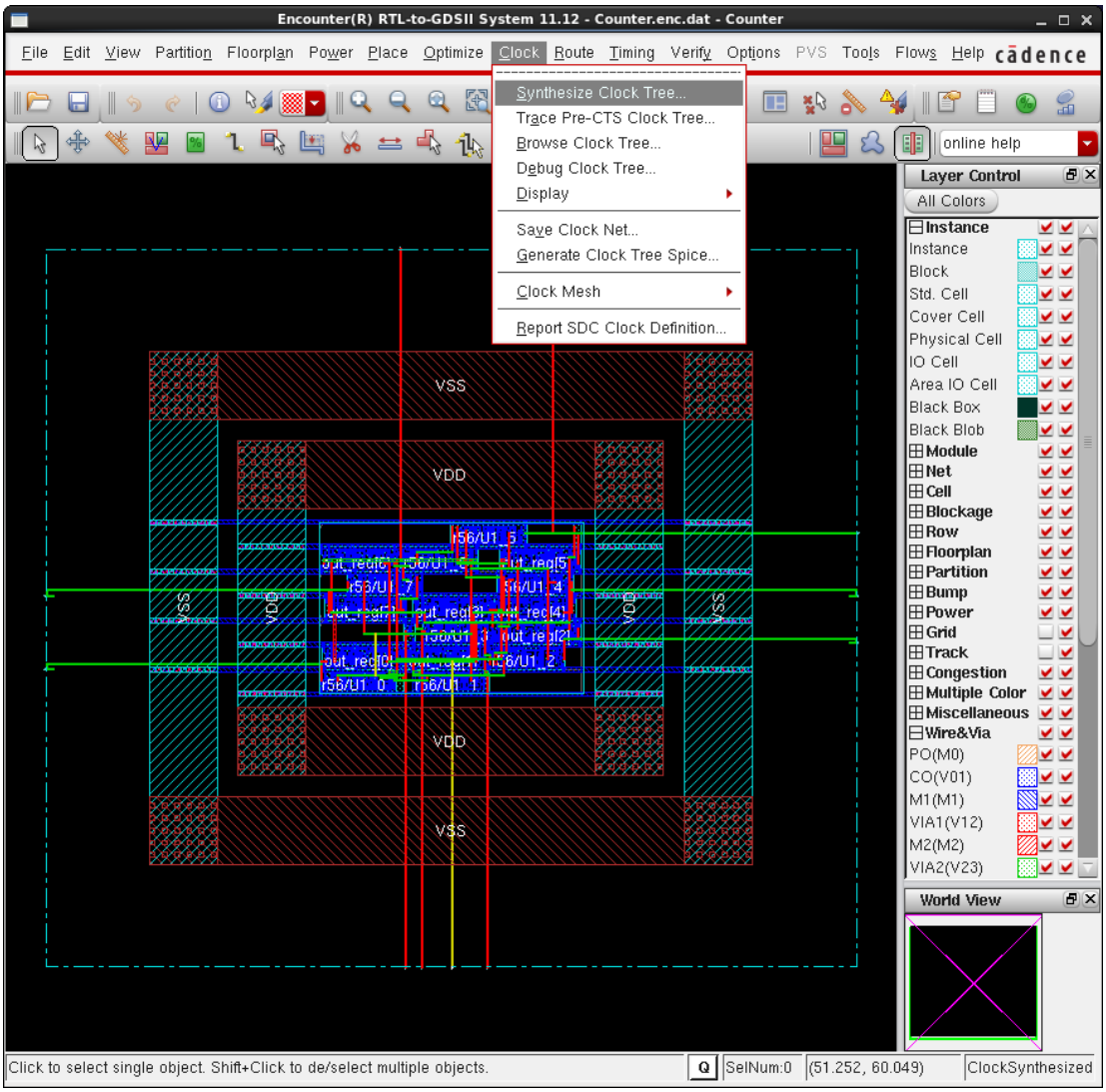


If you zoom in, you may notice that you cannot see inside the cells. If you Check the first checkbox for **Cell** under layer Control on the right, you can see inside the cells.

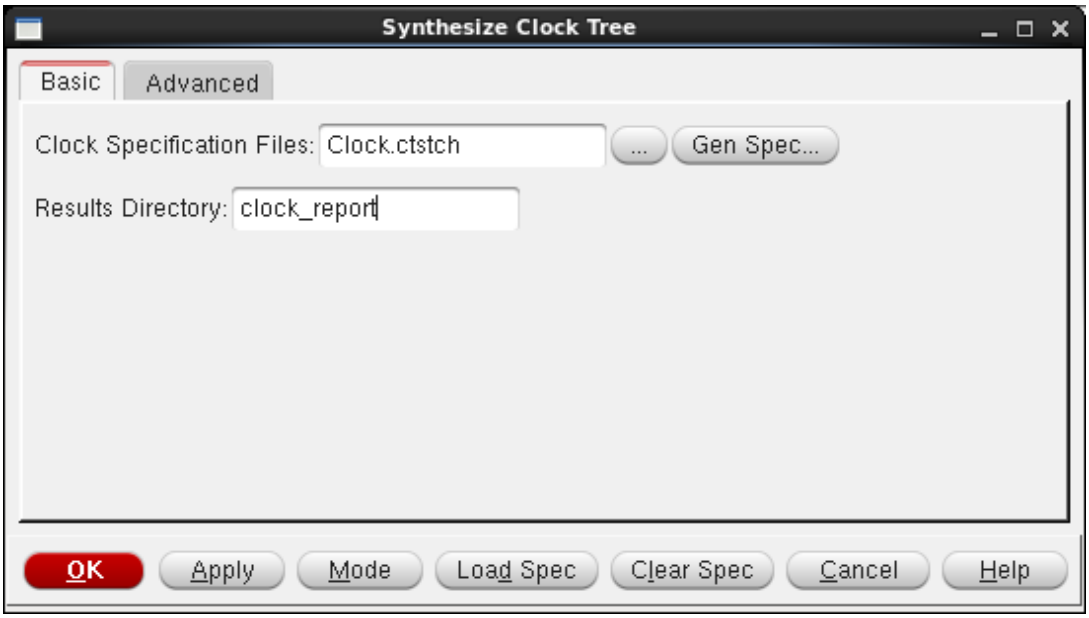


18. Now we need to insert a clock tree. We already created the specification file in Step 2. It is possible to generate the specification file by clicking on Gen Spec in the Synthesize Clock Tree window and selecting the proper buffers. However you still need to specify the Clk pin the generated file, so it is easiest to use the one in Step 2.

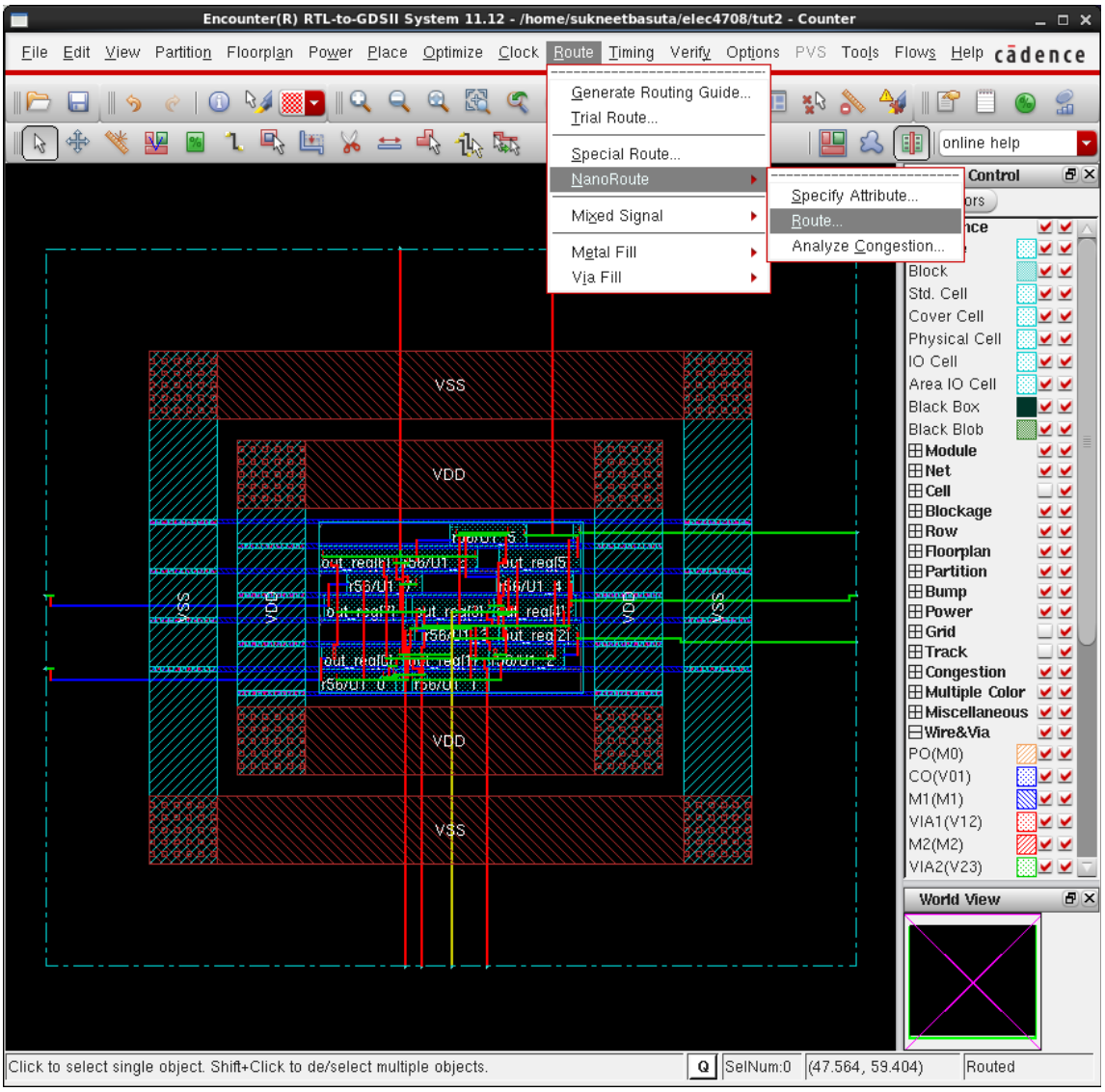
    Select **Clock-> Synthesize Clock Tree**.


Encounter(R) RTL-to-GDSII System 11.12 - Counter.enc.dat - Counter
File Edit View Partition Floorplan Power Place Optimize Clock Route Timing Verify Options PVS Tools Flows Help
cadence
Synthesize Clock Tree...
Trace Pre-CTS Clock Tree...
Browse Clock Tree...
Debug Clock Tree...
Display
Save Clock Net...
Generate Clock Tree Spice...
Clock Mesh
Report SDC Clock Definition...
online help
Layer Control
All Colors
Instance
Block
Std. Cell
Cover Cell
Physical Cell
IO Cell
Area IO Cell
Black Box
Black Blob
Module
Net
Cell
Blockage
Row
Floorplan
Partition
Bump
Power
Grid
Track
Congestion
Multiple Color
Miscellaneous
Wire&Via
PO(M0)
CO(V01)
M1(M1)
VIA1(V12)
M2(M2)
VIA2(V23)
World View
VSS
VDD
VDD
VSS
VDD
VSS
r56/U1_5
r53/U1_7
r56/U1_4
r56/U1_3
r56/U1_2
r56/U1_0
r56/U1_1
Click to select single object. Shift+Click to de/select multiple objects.
SelNum:0
(51.252, 60.049)
ClockSynthesized

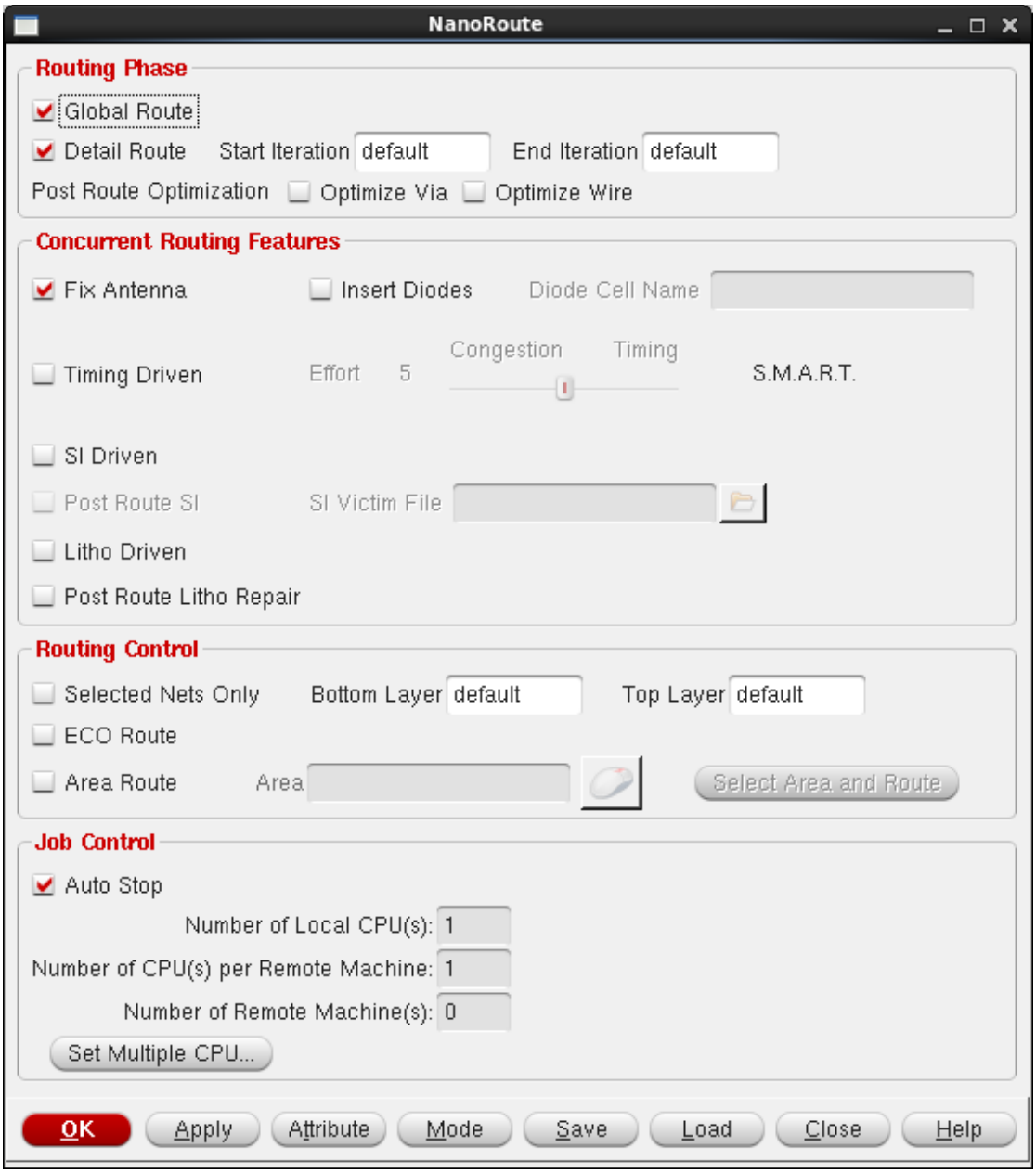
The Synthesize Clock Tree window should popup. Set the Clock Specification file to “Clock.ctstch”. The rest of the defaults should be fine. Hit OK and look at the terminal to determine if it was successful.



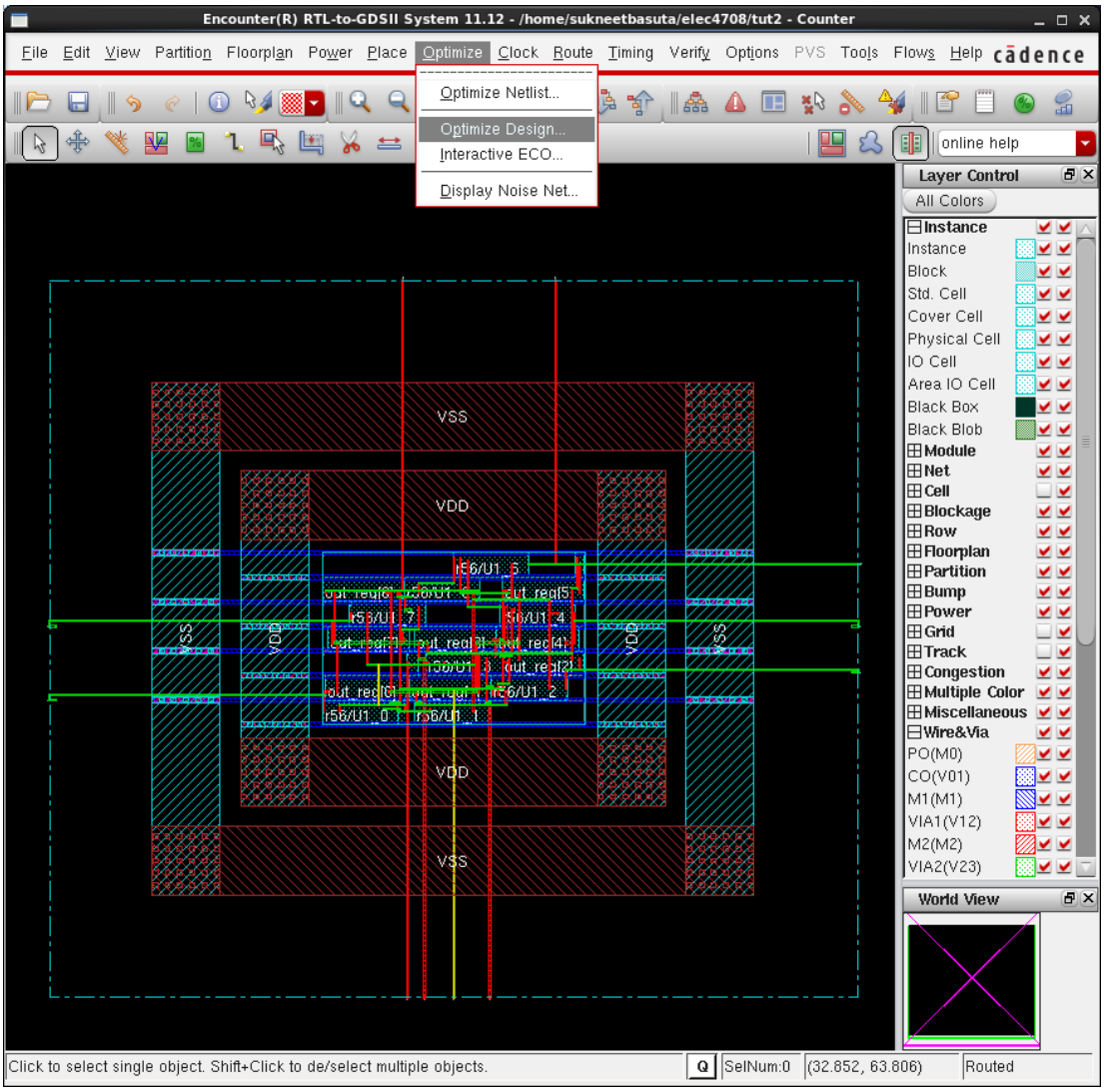
19. Placing the standard cells automatically routed the wires for the standard cells for us, but to ensure everything is routed lets use Nanoroute. Select **Route-> Nanoroute -> Route**



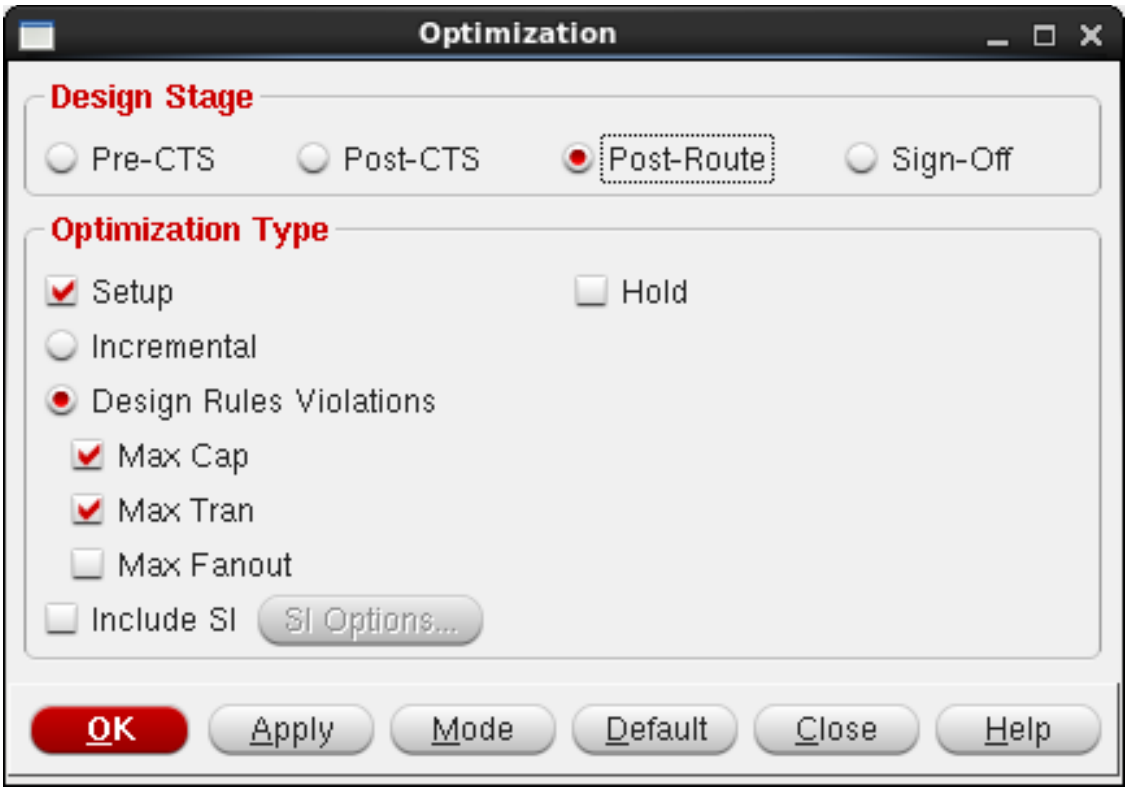
20. Use the default values for NanoRoute and hit OK. All your wires should be routed now.



21. We can now do a timing optimization. Select **Optimize -> Optimize Design**

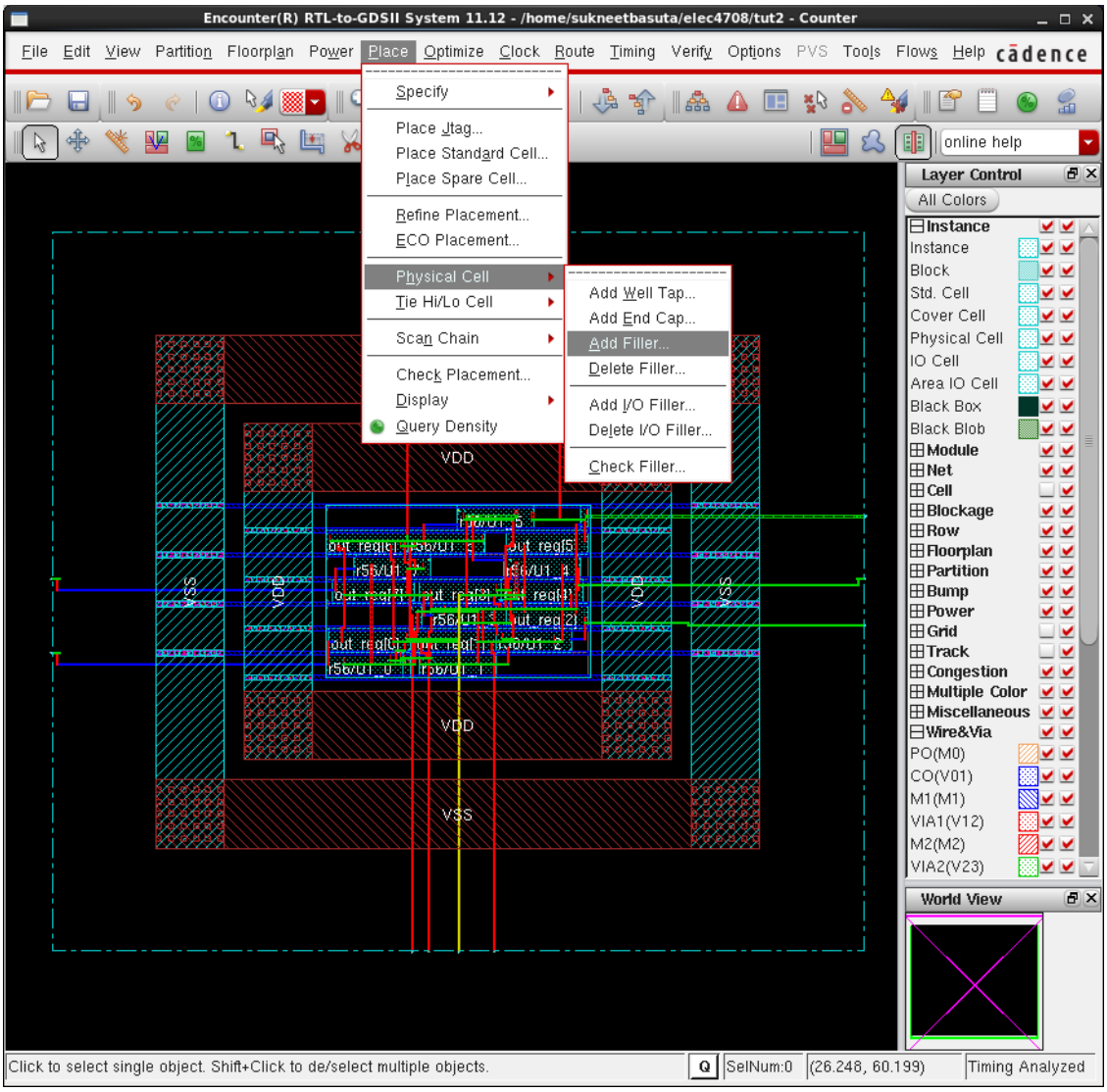


In the Optimization window, select “Post-Route” and Hit Ok.

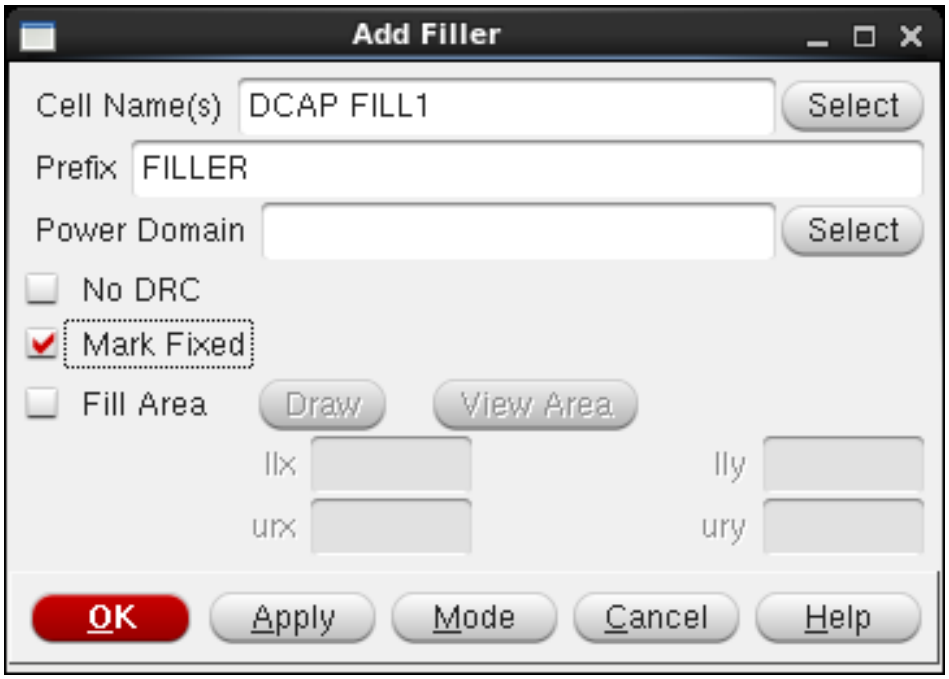


It will be difficult to see any changes in the layout so check the terminal window which should show a summery of changes

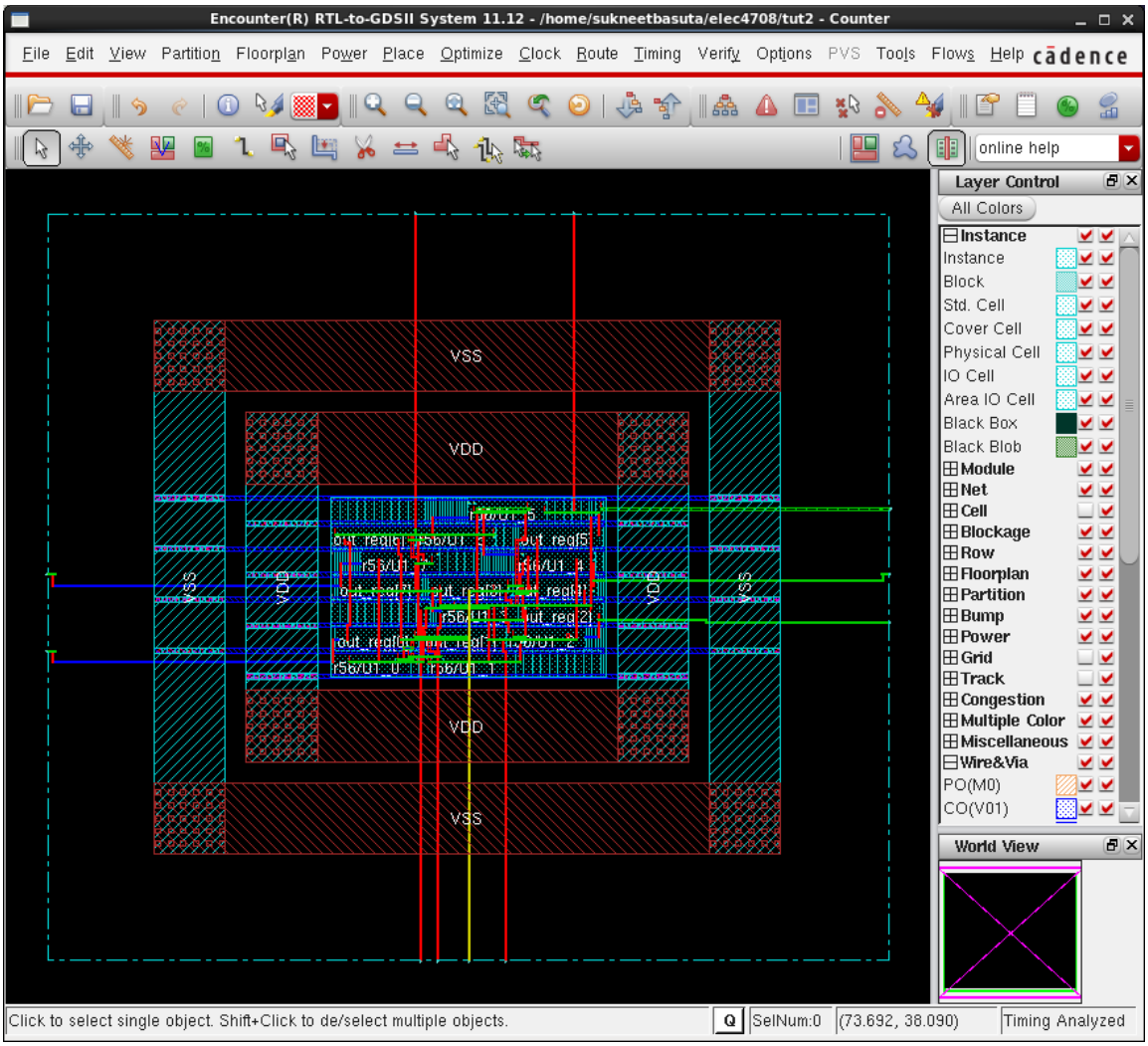
22. Now that we have routed all the wires and placed all the cells we require, we will add empty filler cells to the design. Select **Place -> Physical Cell -> Add Filler**



In the Add Filler window, select the DCAP and FILL1 cells and check of Mark Fixed. DCAP places a capacitor between VDD and VSS.



After you hit OK, you will see that all the empty spaces in the layout have been filled.

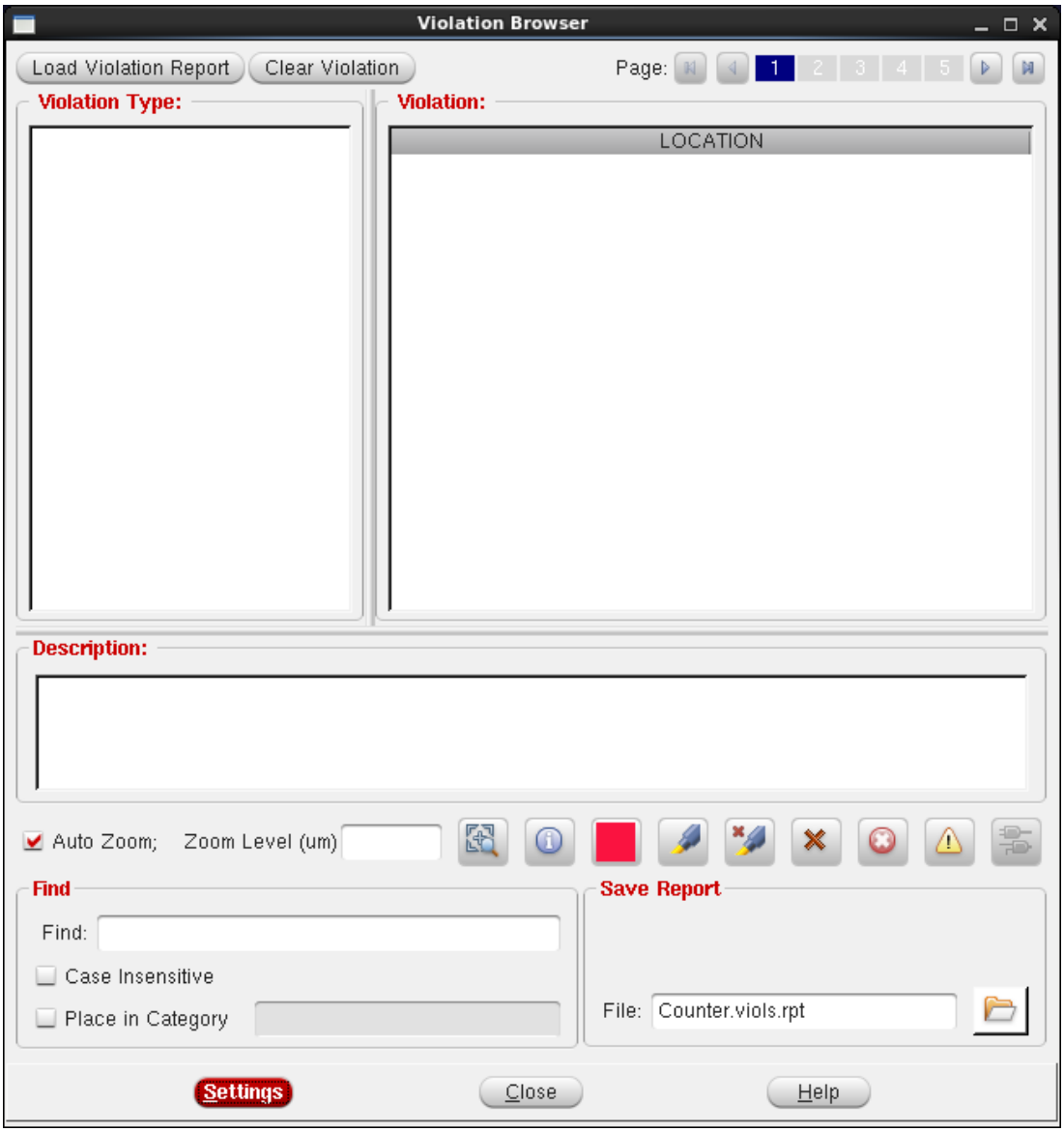


23. Now that the layout is complete, lets verify the layout. Let's verify the connectivity and geometry. Select **Verify -> Verify Geometry** and **Verify -> Verify Connectivity**. Use the default values for both cases. You will see the results in the terminal window.
    i.e.

```
  VERIFY GEOMETRY ..... Cells          :  0 Viols.
  VERIFY GEOMETRY ..... SameNet        :  0 Viols.
  VERIFY GEOMETRY ..... Wiring         :  0 Viols.
  VERIFY GEOMETRY ..... Antenna        :  0 Viols.
  VERIFY GEOMETRY ..... Sub-Area : 1 complete 0 Viols. 0 Wrngs.
VG: elapsed time: 0.00
Begin Summary ...
  Cells       : 0
  SameNet     : 0
  Wiring      : 0
  Antenna     : 0
  Short       : 0
```

```
  Overlap      : 0
End Summary

  Verification Complete : 0 Viols.  0 Wrngs.

**********End: VERIFY GEOMETRY**********

Begin Summary
  Found no problems or warnings.
End Summary

******** End: VERIFY CONNECTIVITY ********
```

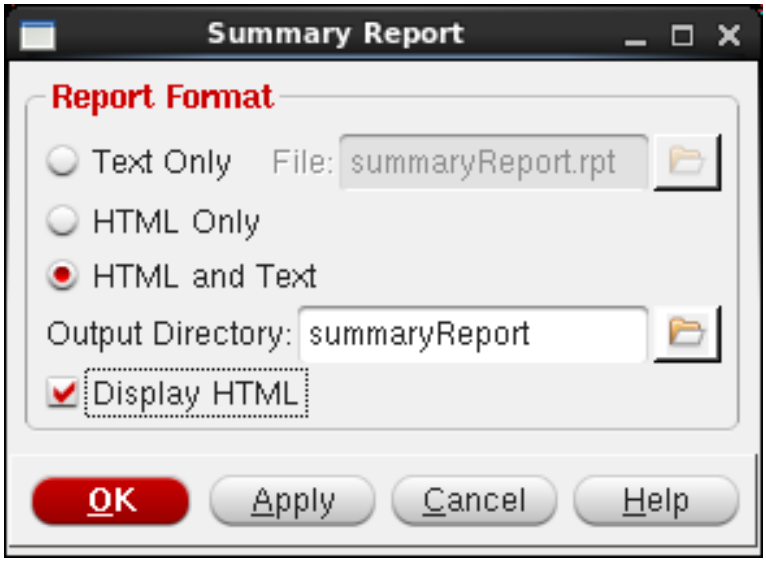
24. Once you have run both select **Tools->Violation Browser** and fix any errors that are displayed.



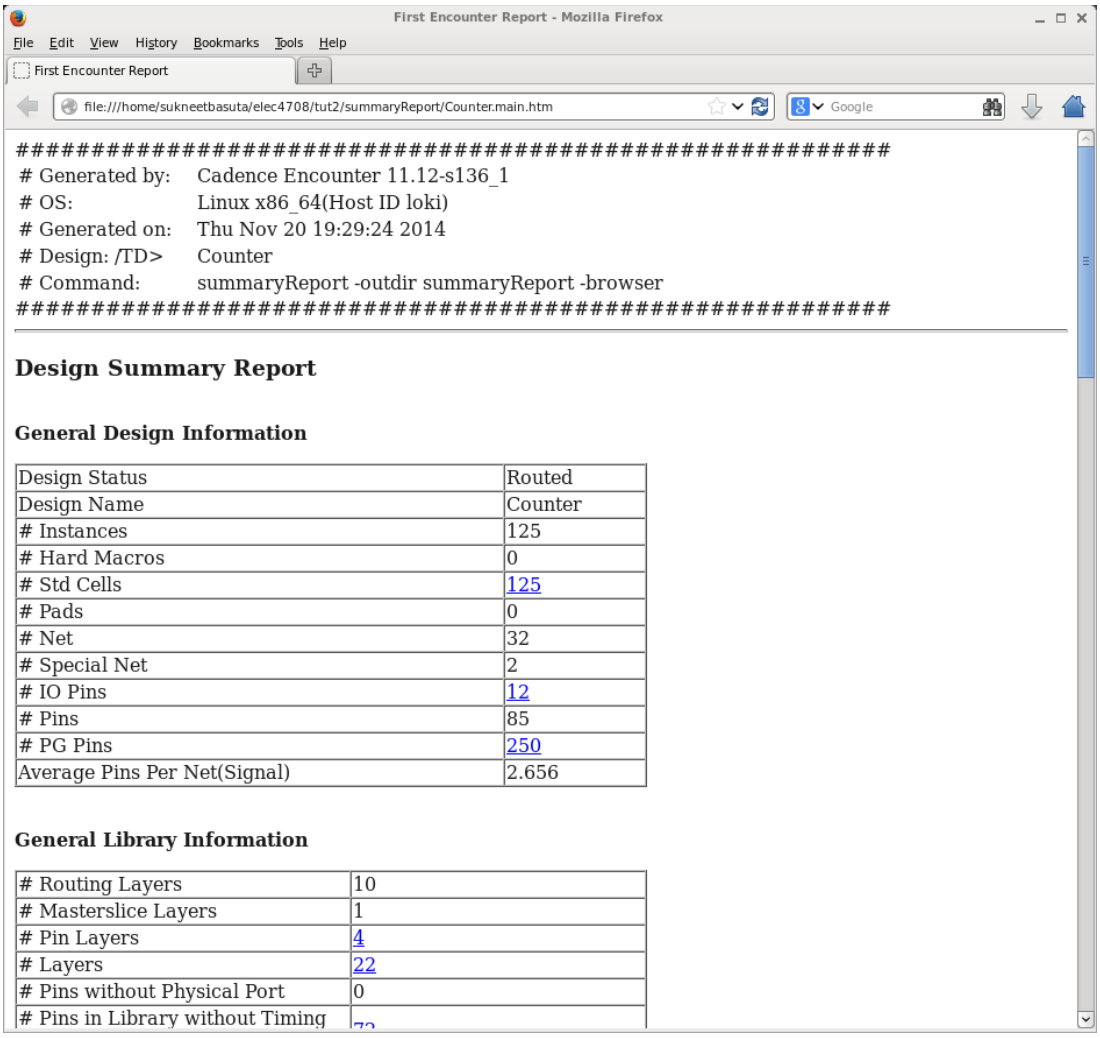
Hints:

- Some routing errors are due to insufficient space. A solution is to make the chip area larger

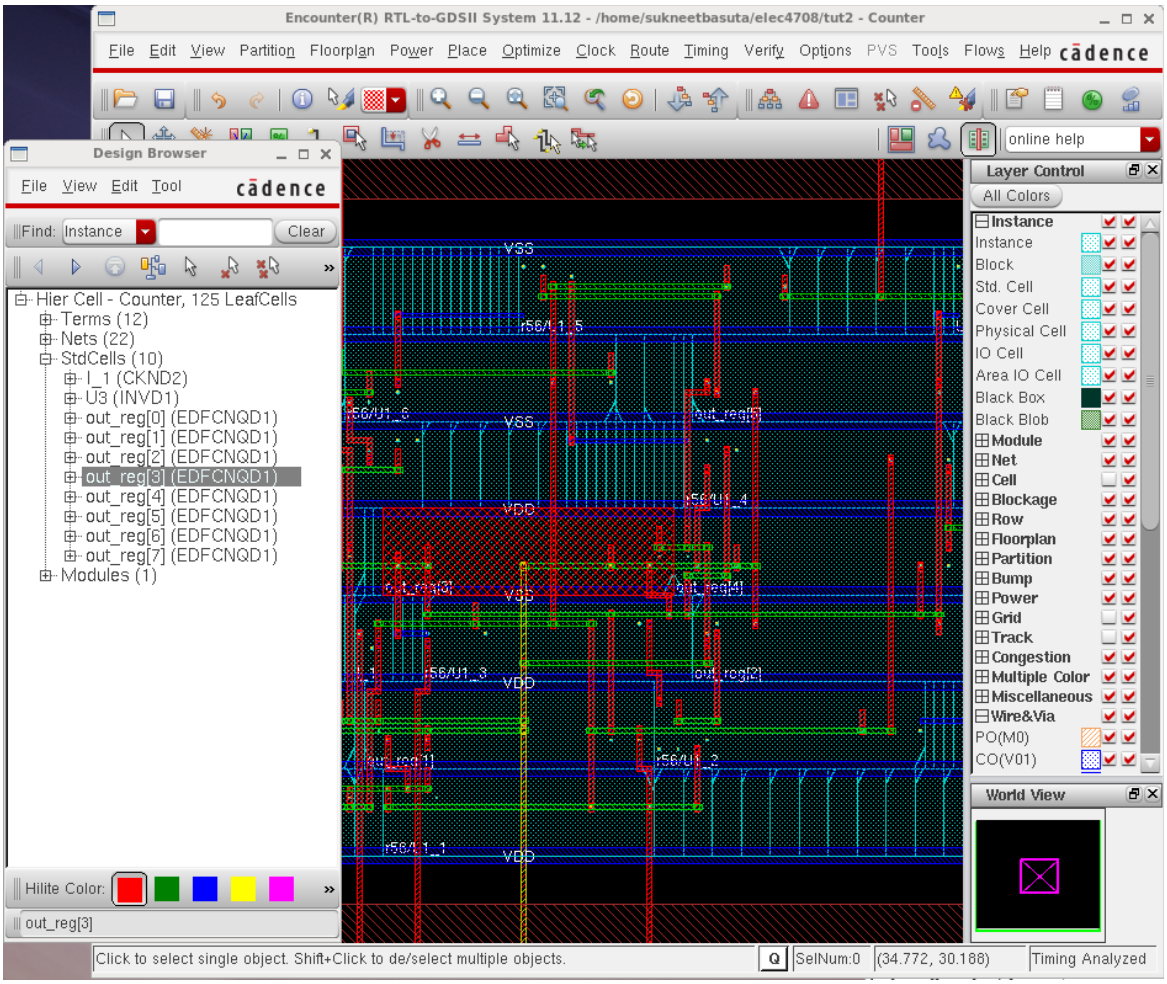
25. Now that the layout is done, we can get a summary report. Select **File -> Report -> Summary**
    In the Summary Report window, you can select a text and/or HTML report



The HTML report will be located at something like summaryReport/Counter.main.htm in your lab2 directory. The Text report will be located at summaryReport/Counter.main.ascii.



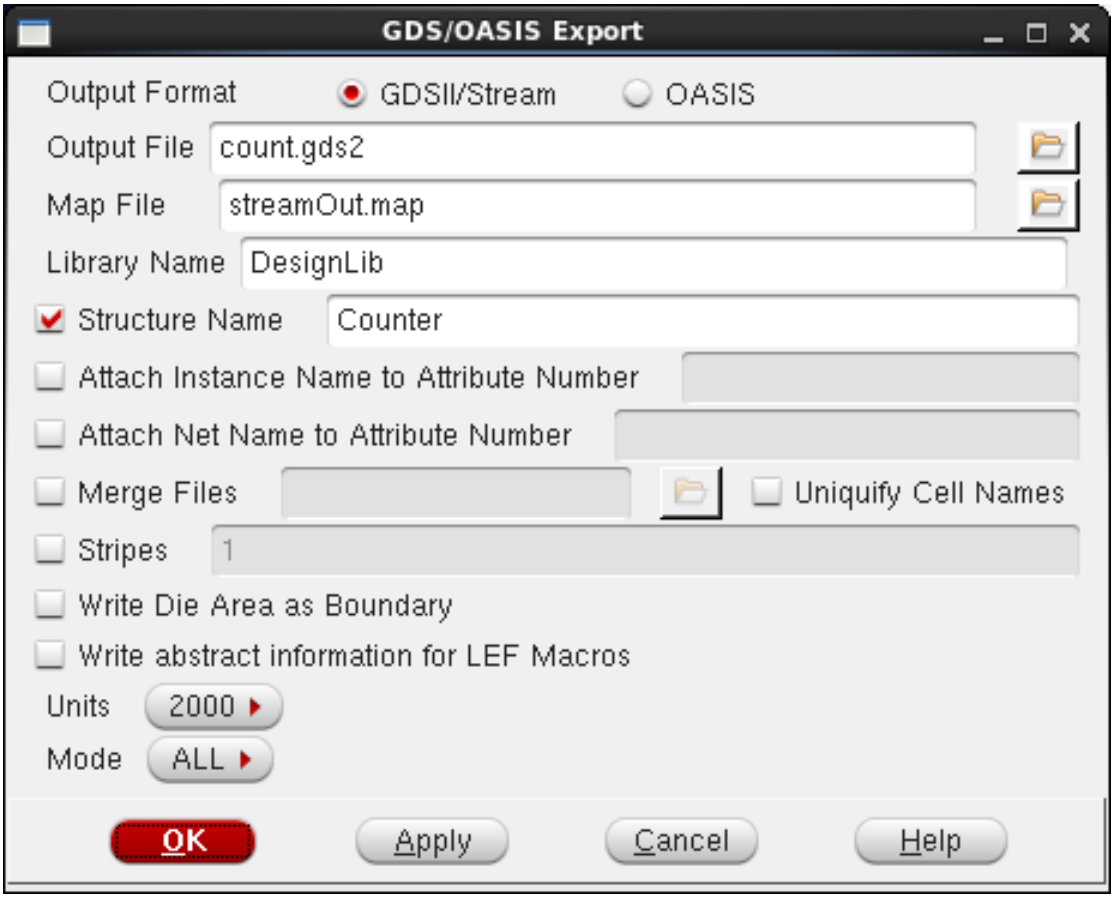
26. You can also use the design browser (located under the tools pull down menu) to locate the individual cells, module, nets, etc. To open the design browser, select **Tools-> Design Browser** The following picture shows the results browser and the selected module in red in the encounter window.



27. We have gone through a simplified design flow. There are many options that we haven't had a chance to explore. It might be a good idea to take a look at some of the available options. Like RC extraction or the various routing options.

    Generally the last step in a design flow is to send a completed design to the fabrication facility. To do this the design is exported to a GDS2 file. To generate a GDS2 file we need a map file that relates layer names to numbers. Select **File -> Save -> GDS/OASIS**.

    Name the output stream file anything you want (should be related to what the design is) with the gds2 extension. Make sure the map file is "streamOut.map" and the "Structure Name" is the name of your layout. click "OK".



28. Congratulations, you have now completed the tutorial for a simplified Encounter design flow.

Timing Report

- In the terminal window you opened encounter in, type
  `report_timing`
- The timing report will be printed in the console. The Arrival time is the worst case delay in ns. In the example below, the longest path delay is 0.826 ns.
  i.e.

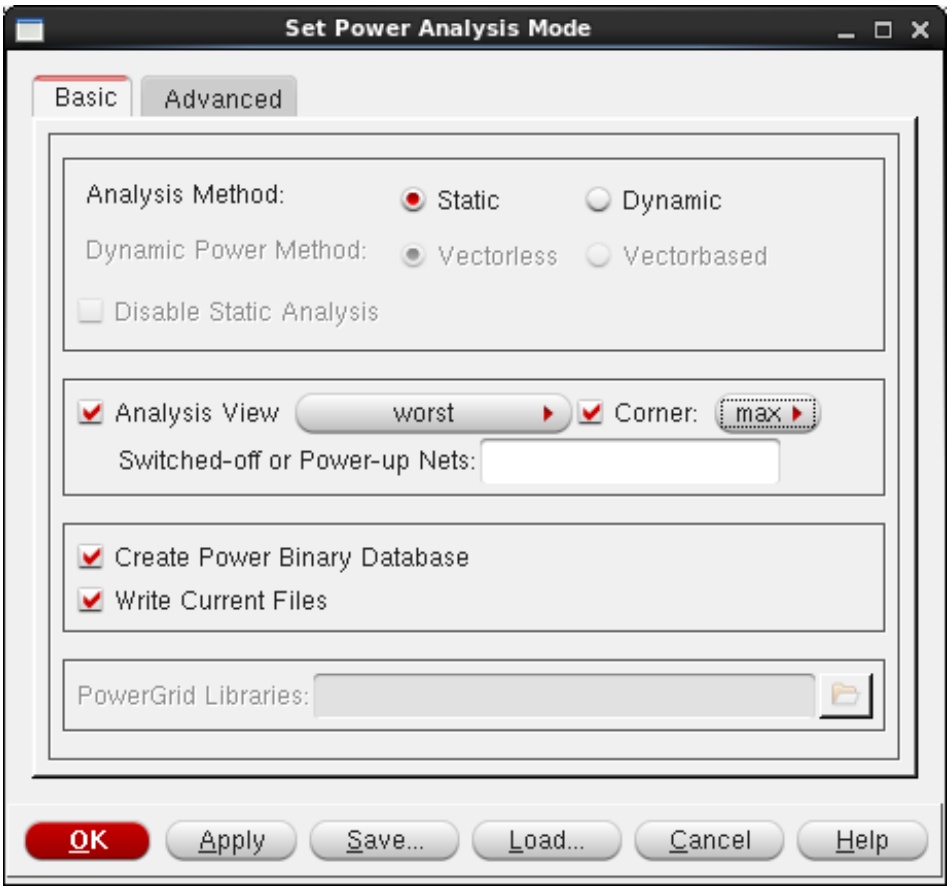
```
Analysis View: worst_view
Other End Arrival Time          0.032
- Setup                         0.113
+ Phase Shift                  10.000
= Required Time                 9.919
- Arrival Time                  0.826
= Slack Time                    9.093
     Clock Rise Edge                     0.000
     + Clock Network Latency (Prop)      0.032
     = Beginpoint Arrival Time           0.032
     +--------------------------------------------------------------------+
     |  Instance  |     Arc      |   Cell   | Delay | Arrival | Required |
     |            |              |          |       |  Time   |   Time   |
```

```
|-----------+--------------+----------+-------+---------+----------|
| out_reg[0] | CP ^         |          |       |   0.032 |    9.124 |
| out_reg[0] | CP ^ -> Q v  | EDFCNQD1 | 0.145 |   0.177 |    9.270 |
| r56/U1_0   | A v -> CO v  | FA1D1    | 0.141 |   0.318 |    9.411 |
| r56/U1_1   | CI v -> CO v | FA1D1    | 0.072 |   0.390 |    9.483 |
| r56/U1_2   | CI v -> CO v | FA1D1    | 0.071 |   0.461 |    9.554 |
| r56/U1_3   | CI v -> CO v | FA1D1    | 0.070 |   0.531 |    9.624 |
| r56/U1_4   | CI v -> CO v | FA1D1    | 0.071 |   0.602 |    9.695 |
| r56/U1_5   | CI v -> CO v | FA1D1    | 0.070 |   0.672 |    9.764 |
| r56/U1_6   | CI v -> CO v | FA1D1    | 0.068 |   0.740 |    9.833 |
| r56/U1_7   | CI v -> S ^  | FA1D1    | 0.086 |   0.826 |    9.919 |
| out_reg[7] | D ^          | EDFCNQD1 | 0.000 |   0.826 |    9.919 |
+-----------------------------------------------------------------------+
```

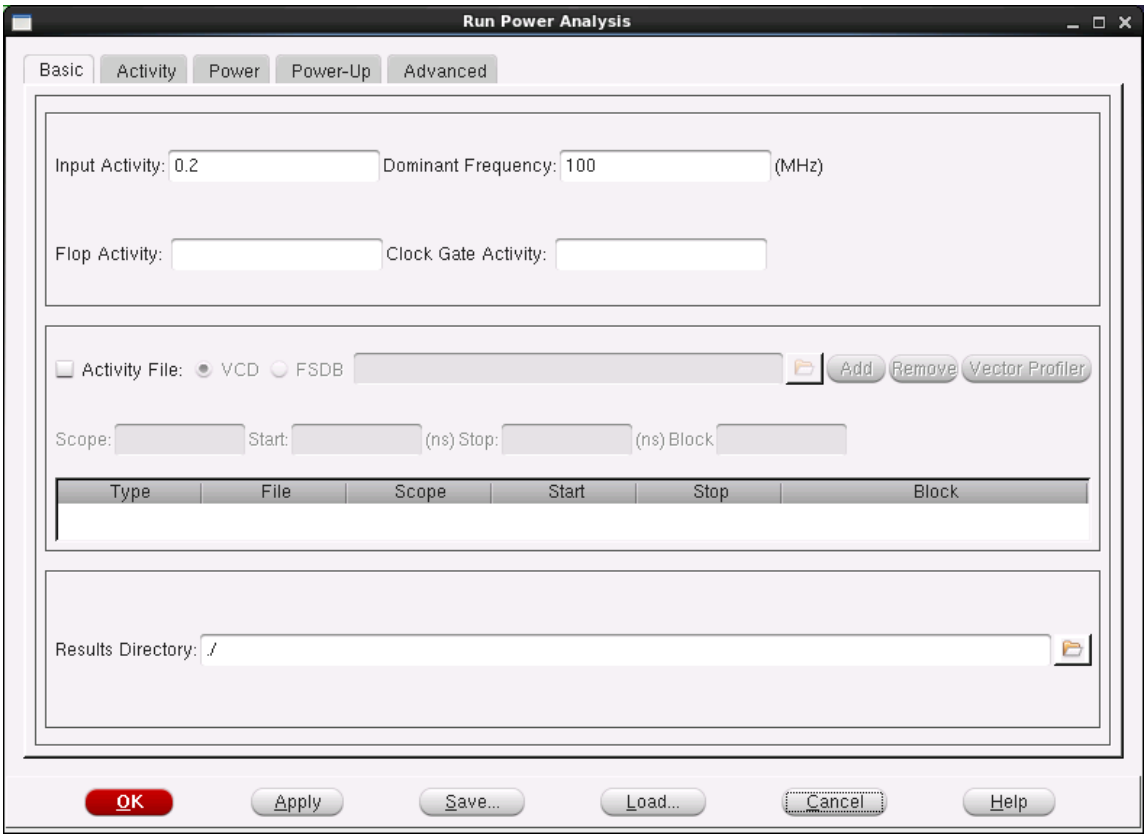
- If you wanted to the delays of other paths, you can do
  `report_timing –max_paths 5`
  where the number of max_paths prints n paths delays in descending order (n is 5 in the above command).

Power Report

- Select **Power -> Power Analysis -> Setup** to setup the power analysis. Select the analysis view you created in Step 7. User the max Corner.



- Select **Power -> Power Analysis -> Run**. The Run Power Analysis window will popup. Here you can specify the input activity and Clock Frequency (Dominant Frequency).



- Hit OK and the power consumption will be displayed in the terminal. Note that the power consumption is displayed in mW, and a V_{DD} of 0.9V is used by default.
  i.e.

```
Total Power
-----------------------------------------------------------------------
Total Internal Power:       0.0123       78.72%
Total Switching Power:     0.001449       9.267%
Total Leakage Power:     0.001879       12.02%
Total Power:      0.01563
-----------------------------------------------------------------------
```