
LUND
UNIVERSITY

# Backend Tools

## Place and Route
## for Faraday 130nm

**Oskar Andersson**

# Import Design

- This step will take a while. Be patient and pick exactly the files specified in the guide.
- It is possible to save/load settings using the Save button.
  Do this to avoid browsing for all files multiple times.
  This applies both to Design Import and Create Analysis Configuration.

- It is important that all files are included and in the correct order.
  Look for possible errors in the command prompt.

- If you need to re-import your design the tool needs to be restarted.

# Import Design

- File -> Import Design

- You should enter:
  - Netlist + Top Cell
  - LEF files
  - IO file
  - Power
  - MMMC definition file (Use Create Analysis Configuration).
- More information on next slides.

# Import Design – LEF files

- Folder for backend files:
  - /usr/local-eit/cad2/far130/syn2012/

- LEFs - Physical information (Pins and Metal Layers):
  - header8m2t_V55.lef – Header LEF (Design rules)
  - fsc0l_d_generic_core.lef – Physical dimensions for Standard Cells
  - FSC0L_D_GENERIC_ANT_V55.8m2t.lef – Antenna information for Standard Cells
  - foc0l_a33_t33_generic_io.8m2t.lef – Physical dimension for I/O Cells
  - FOC0L_A33_T33_GENERIC_IO_ANT_V55.8m2t.lef – Antenna information for I/O Cells

# Import Design – Timing Files

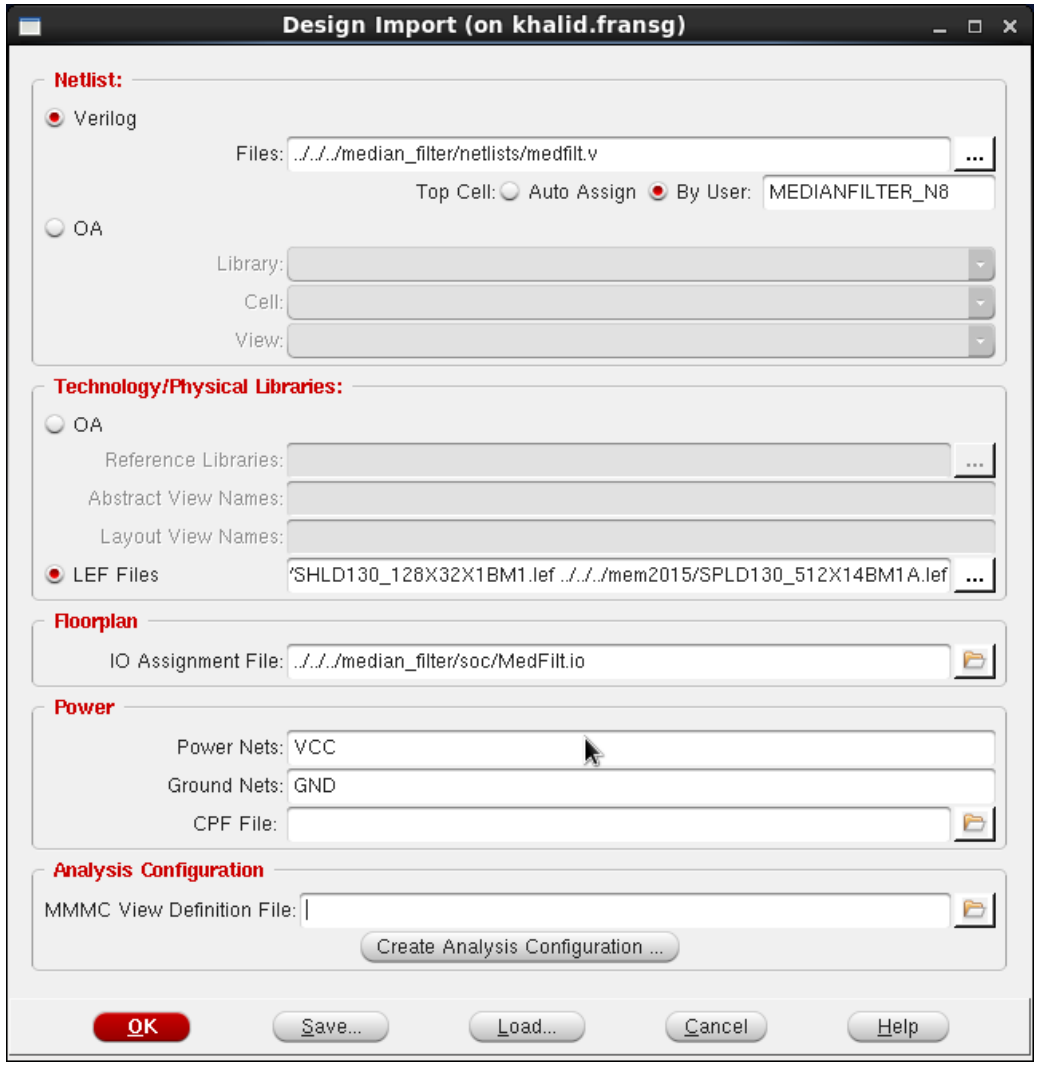
- Folder for backend files:
  - /usr/local-eit/cad2/far130/syn2012/
- Libs:
  - fsc0l_d_generic_core_xx.lib – Low Power Standard Cell Library
  - foc0l_a33_t33_generic_io_xx.lib – Low Power I/O Library
- Performance variations depending on environment: Manufacturing Process, Voltage, Temperature.
- MAX-timing = worst timing => Worst Case
  - ss1p08v125c (slow NMOS – slow PMOS) (1.08V) (125°C)
- Min-Timing = best timing => Best Case
  - ff1p32vm40c (fast NMOS – fast PMOS) (1.32V) (- 40°C)

# Import Design - Memories

- Memories are found in memory directory.
- LEFs:
  - SPLD130_512X14BM1A.lef
  - SHLD130_128X32X1BM1.lef
- Libs:
  - Best Case :
    - SPLD130_512X14BM1A_BC.lib
    - SHLD130_128X32X1BM1_BC.lib
  - Worst Case:
    - SPLD130_512X14BM1A_WC.lib
    - SHLD130_128X32X1BM1_WC.lib

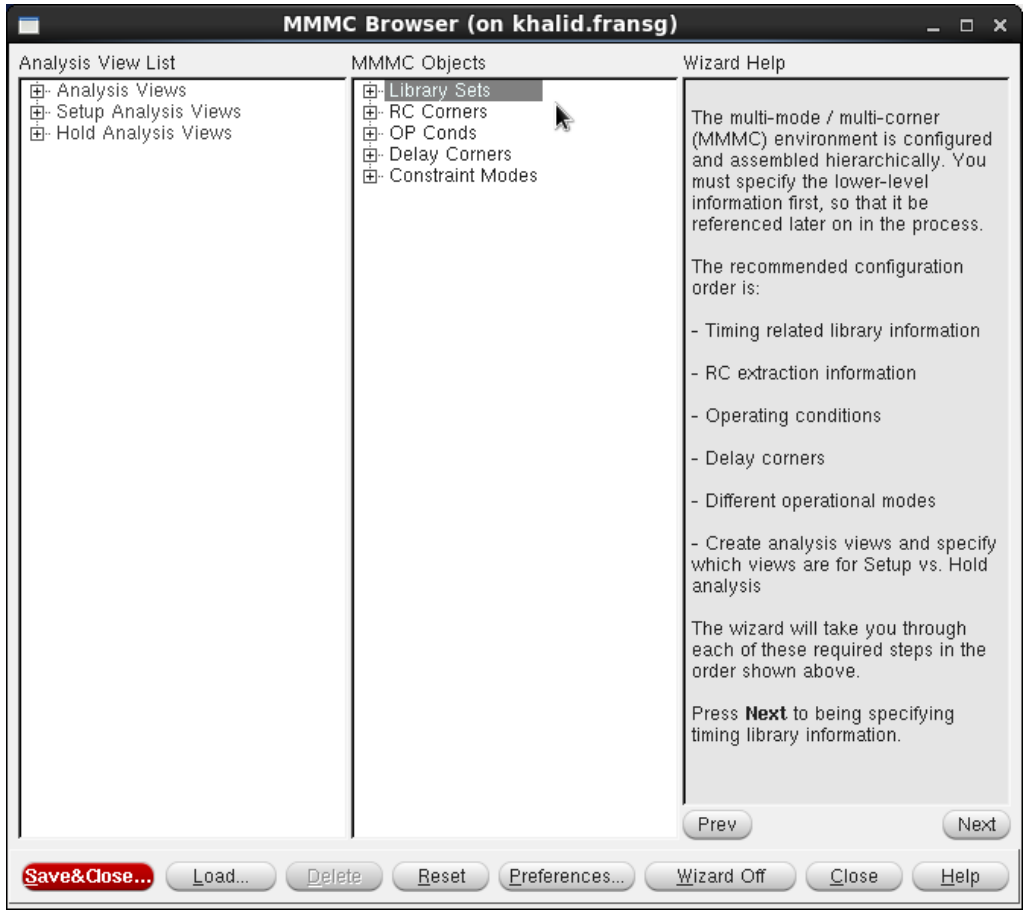
# Import Design

- After you have entered:
  - Netlist + Top Cell
  - LEF files
  - IO file
  - Power
- It should look something like this.
- First: Click Save.
- Afterwards: Click on Create Analysis Configuration



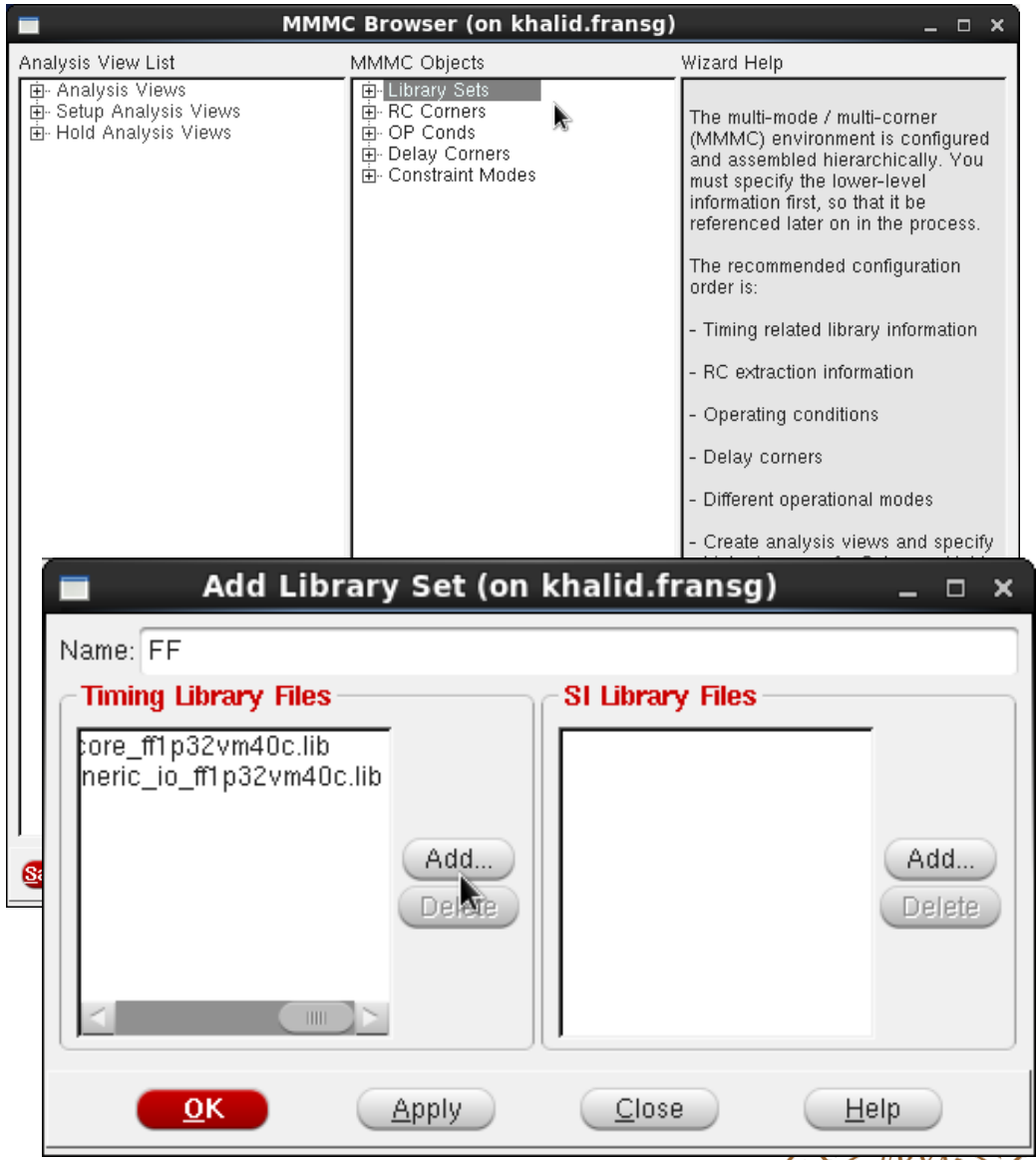

# Import Design - Create Analysis Configuration

- The empty configuration looks like this.
- Feel free to read the wizard about Multi-mode-multi-corner.
- The idea is to analyze the designed chip in multiple environments at different manufacturing process variations. To make sure the fabricated chip works in all cases.
- We will now pupulate elements in the MMMC viewer.



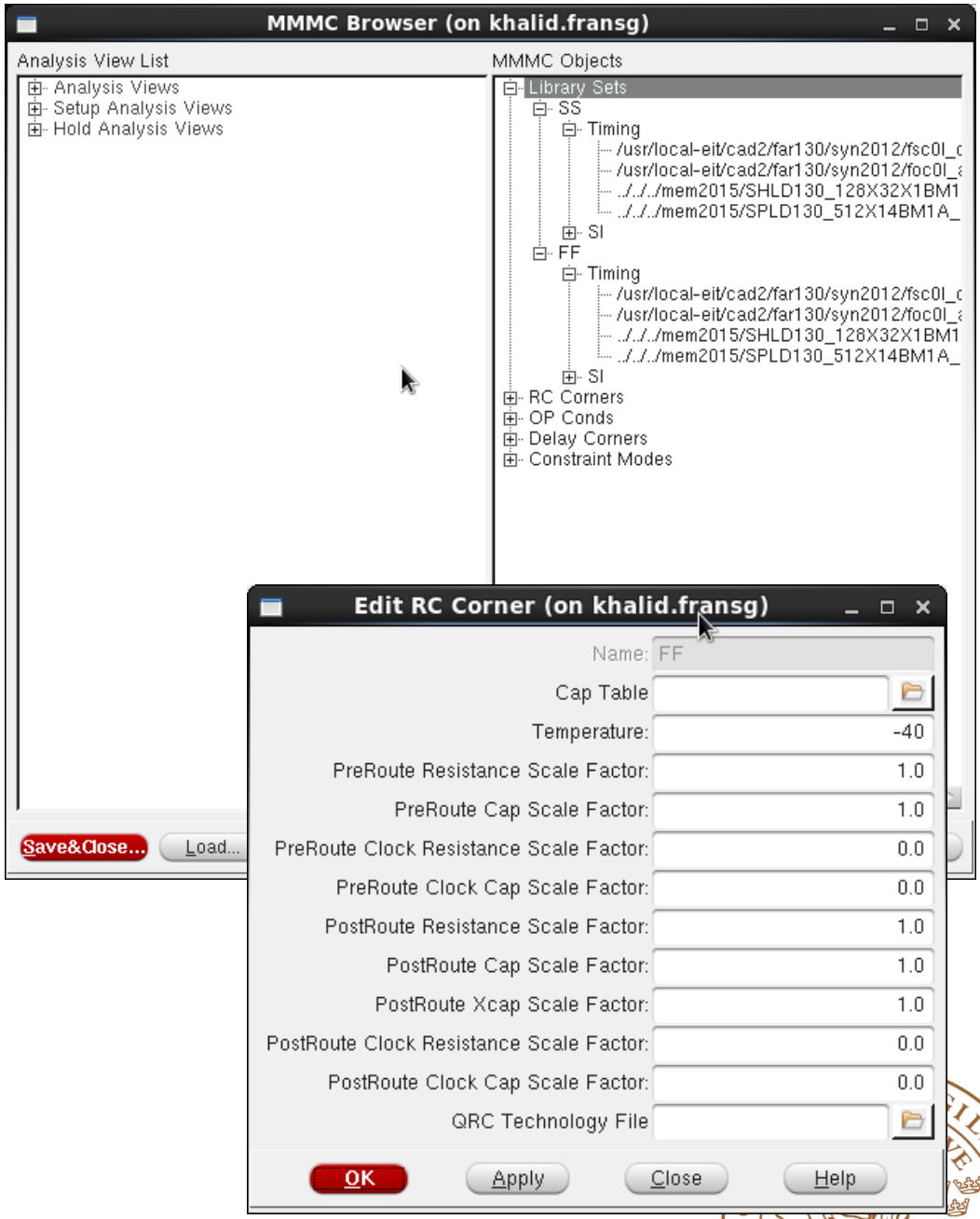
# Import Design - Create Analysis Configuration

- Start by right clicking on Library set and select New.
- Create one set FF for best case timing
  - Add the best case timing libraries.
  - Shown on Previous slide, don't forget the memory files (not shown in figure).
- Create a similar set SS for worst case timing.



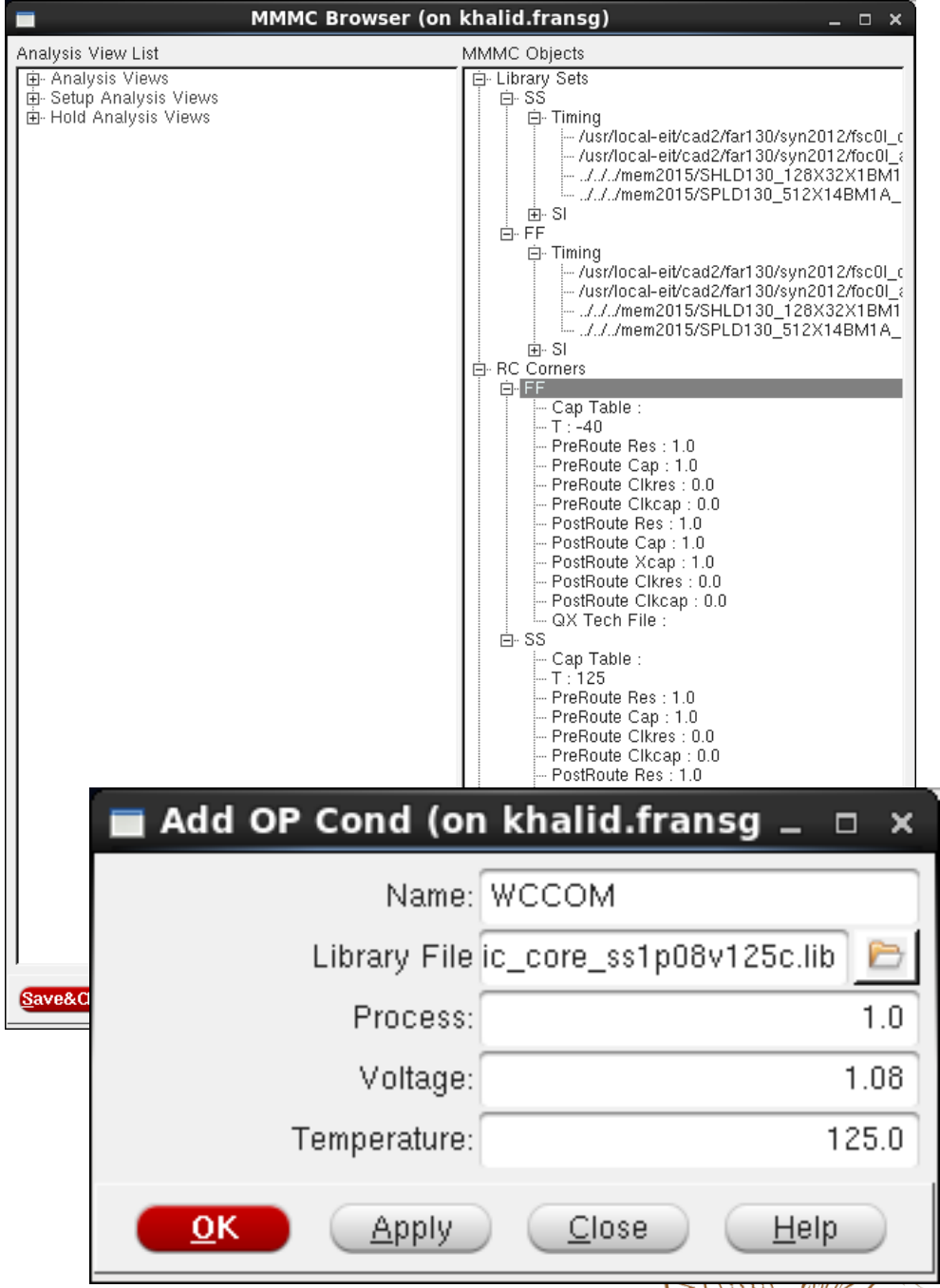
# Import Design - Create Analysis Configuration

- Continue to create two RC corners.
- Create one FF for best case timing
  - Change Temperature to -40°C.
- Use 125°C for Worst Case (SS).



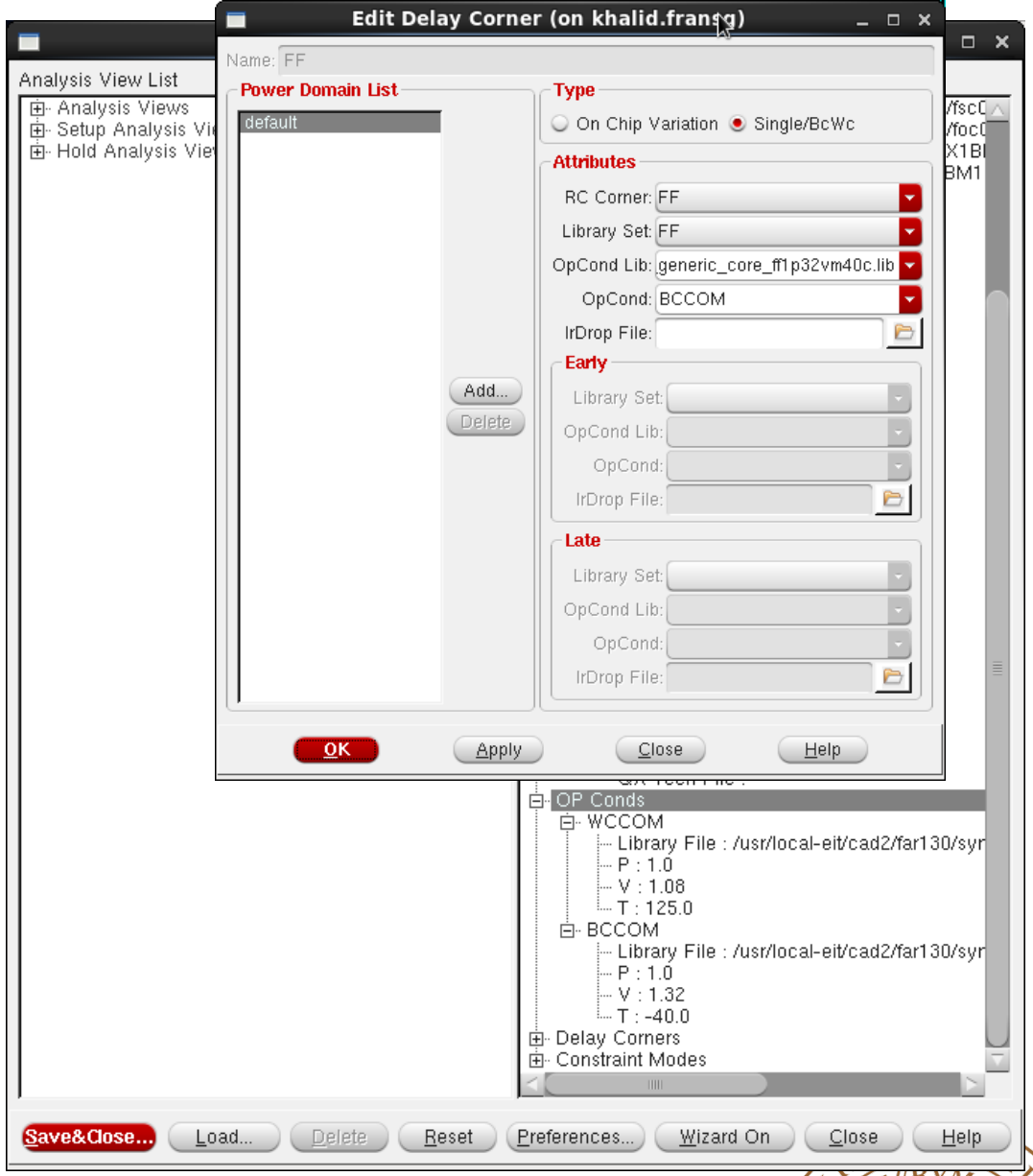
# Import Design - Create Analysis Configuration

- Create two OpConds.
- Name them:
  WCCOM – (SS) Worst Case
  BCCOM – (FF) Best Case
- These are operating conditions defined in the lib files, and therefore, named different.
- Use the Voltage and Temperature for the specific library.



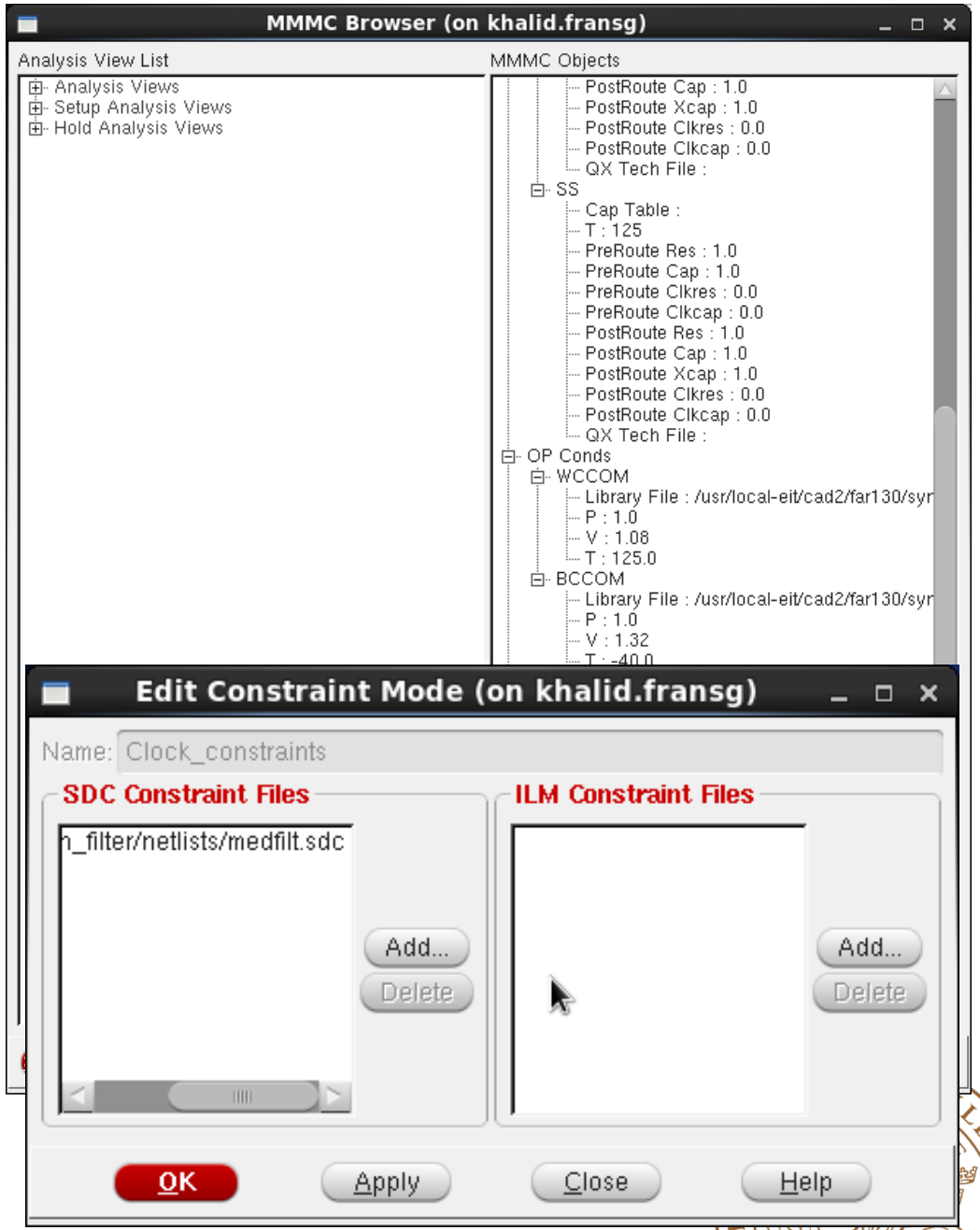
# Import Design - Create Analysis Configuration

- Add two delay corners. FF, and SS.
- Choose the existing RC corner and corresponding Library Set.
- Enter the OpCond and copy the Opcond Lib from the previous OpCond dialog.



# Import Design - Create Analysis Configuration

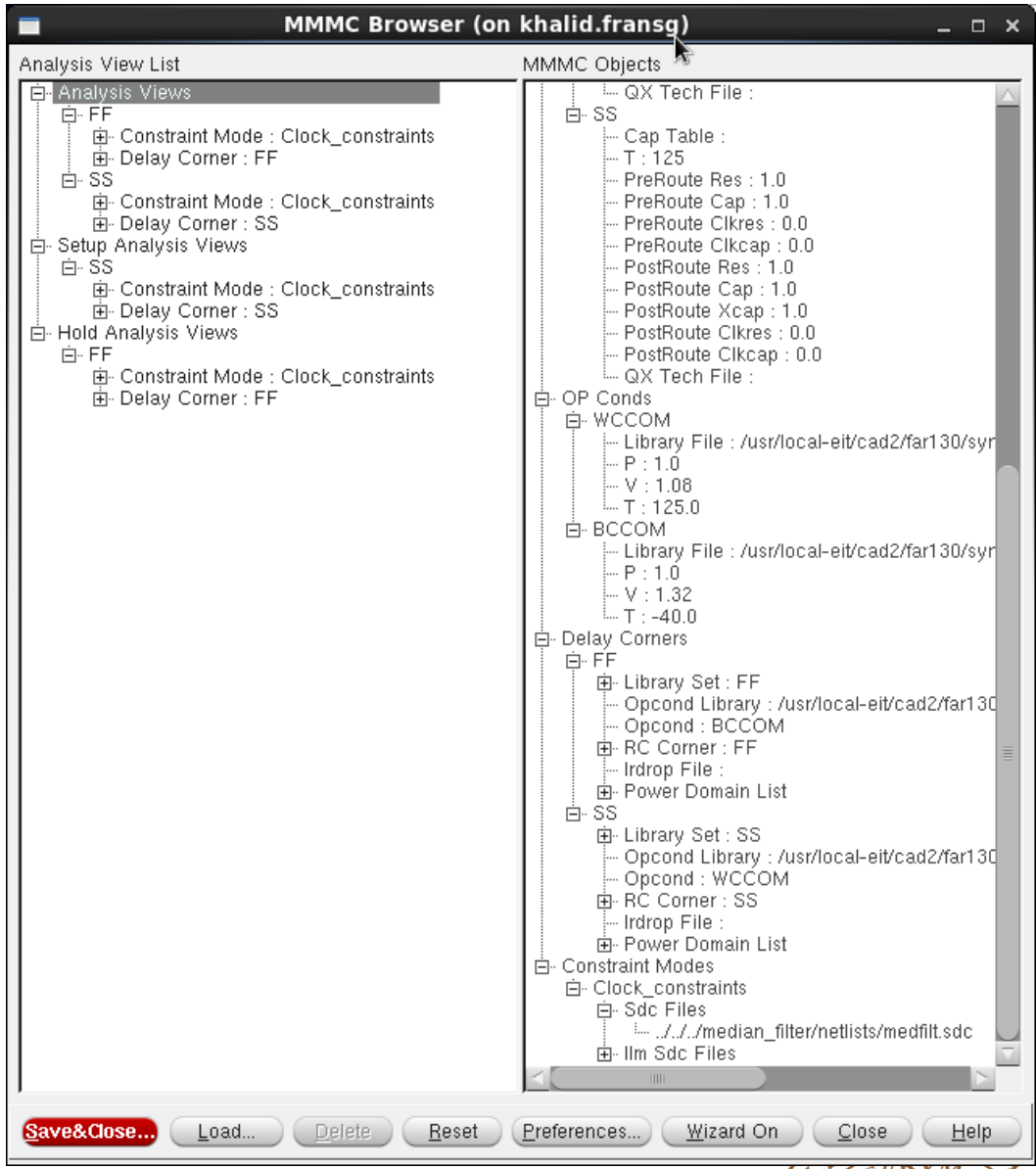
- Now add the clock constraints from the SDC file (created during synthesis).



# Import Design - Create Analysis Configuration

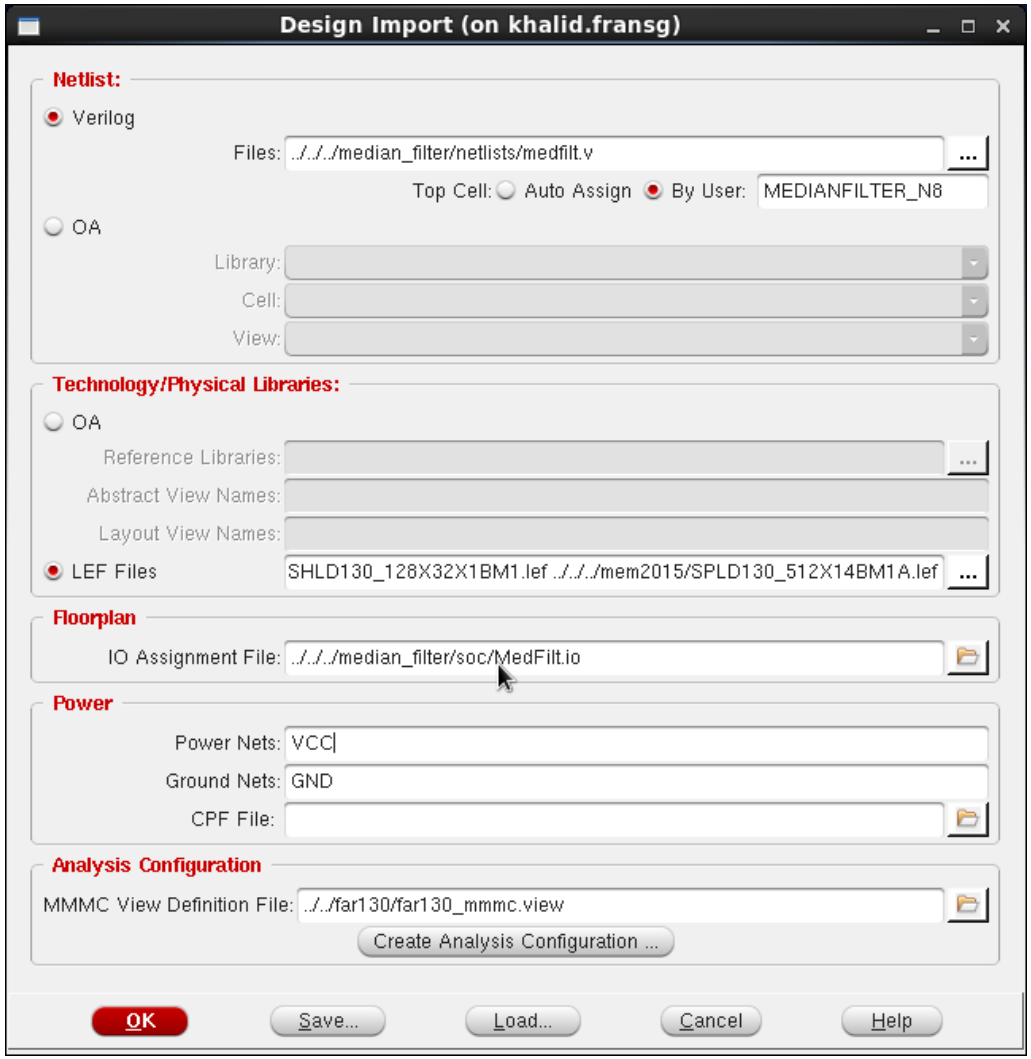
- Now it is time to combine all this information.
- Create two Analysis views using the existing Delay corner and Constraint Mode.
- Select to SS Analysis View as Setup Analysis View
- Select to FF Analysis View as Hold Analysis View

- The design should look like the screenshot.
- Do not forget to save the MMMC file.



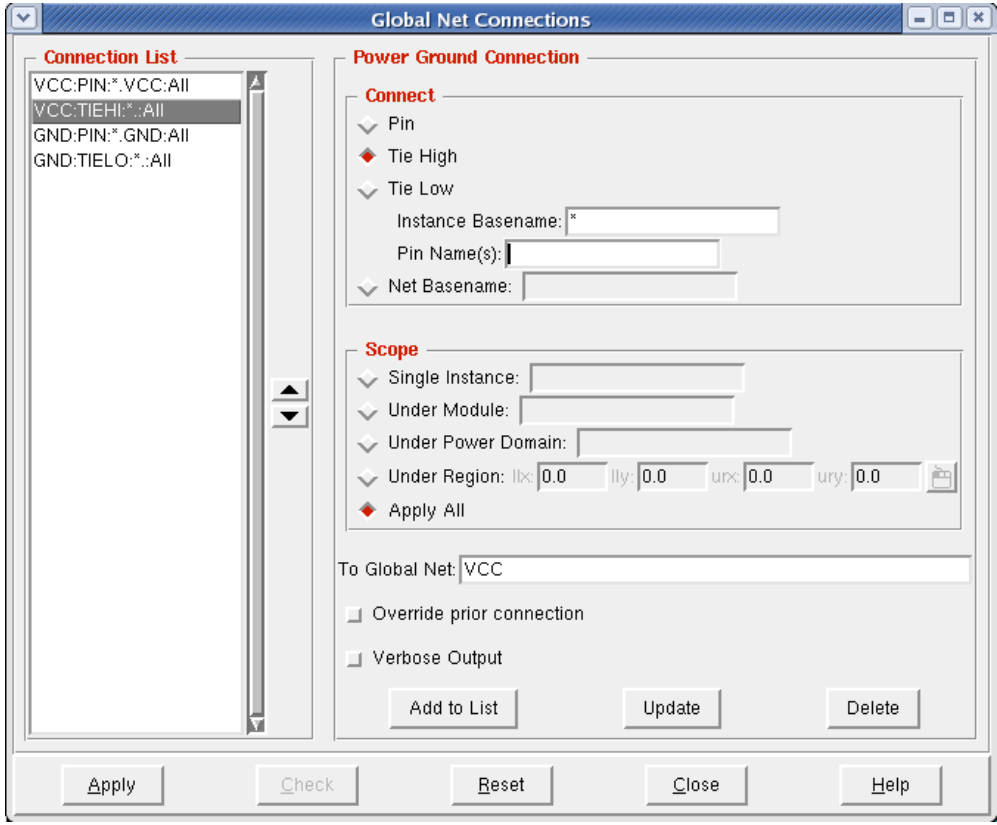
# Import Design

- It should now look something like this.
- First: Click Save.
- Afterwards: Click on OK
- Next time you can use Load instead, and skip browsing for all files.



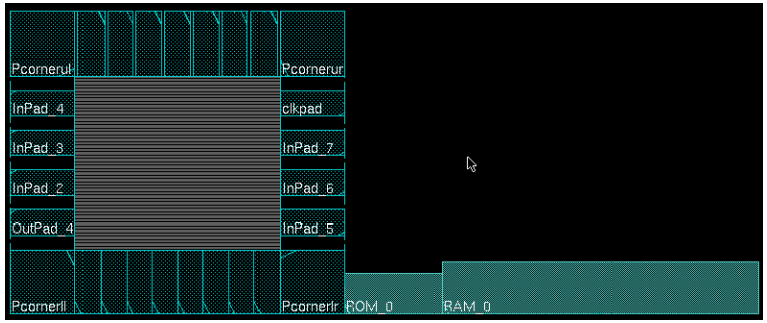
# Global Net Connect

- To connect the power networks:
  Power ->
  Connect Global Nets…
- Two types: Pin & TieHi/Low

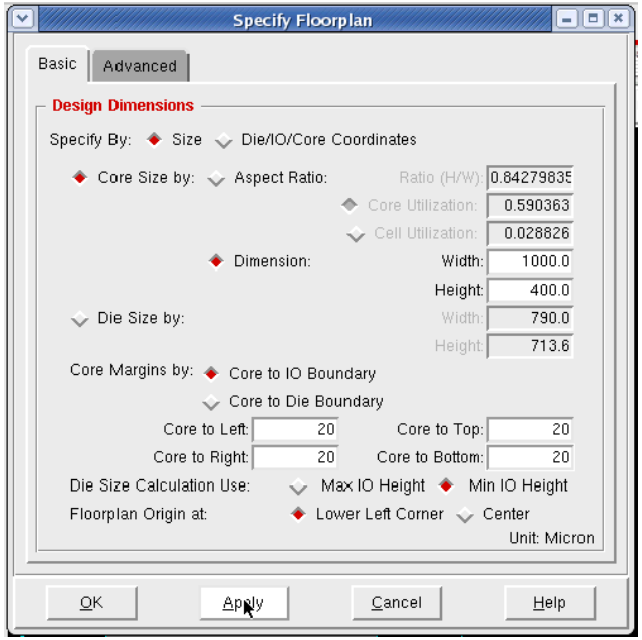



- Power network, e.g., VCC connect to TIEHI, Pin VCC.
- Ground network, e.g., GND connects to TIELO, Pin GND.
- Scope: Apply All

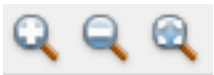
# Floorplan




- Resize floorplan to fit memories
- Floorplan -> Specify Floorplan

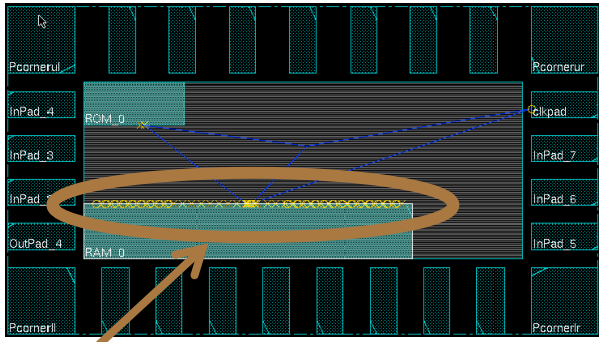



- The size of memories can be measured with the ruler tool. 
- To zoom use the zoom buttons:

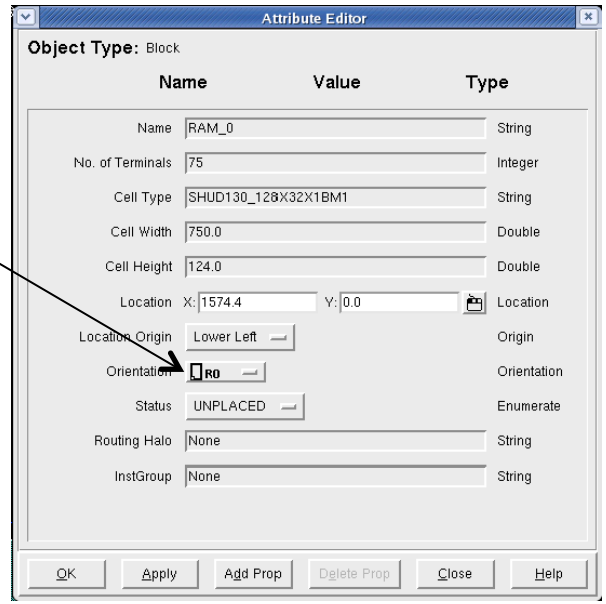


- Zoom in (z), Zoom out (shift+z), Fit to screen (f)
- Also right-click and drag a square to zoom in a to a desired area.

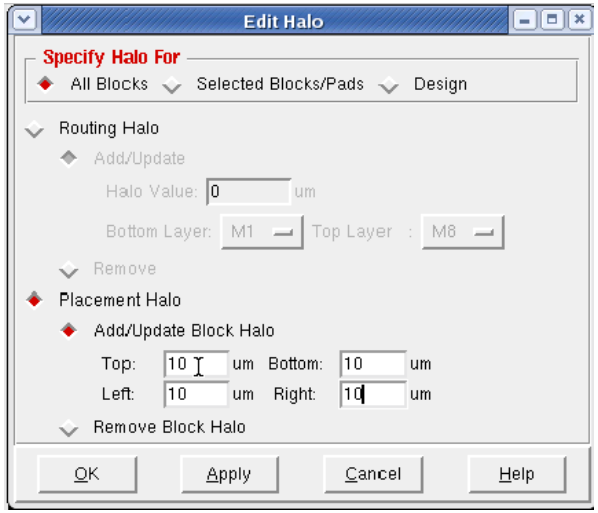
# Place and rotate memories



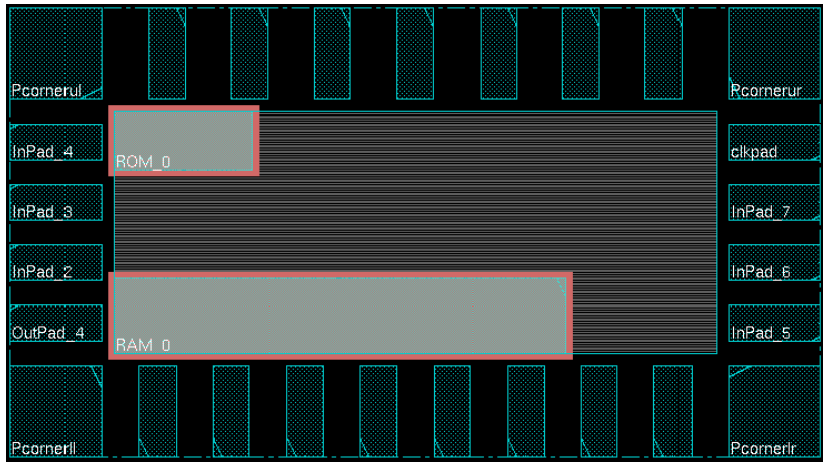

- Move memories by selecting movement tool  or press ”Shift+R”.
- Need to rotate memories to have pin connections inside the core.
- Rotate memories by edit properties for selected object by pressing ”q”.
- Orientation set to R180 for 180 degree rotation.
- Afterwards change to normal pointer by selecting the arrow next to  or press ”a”.



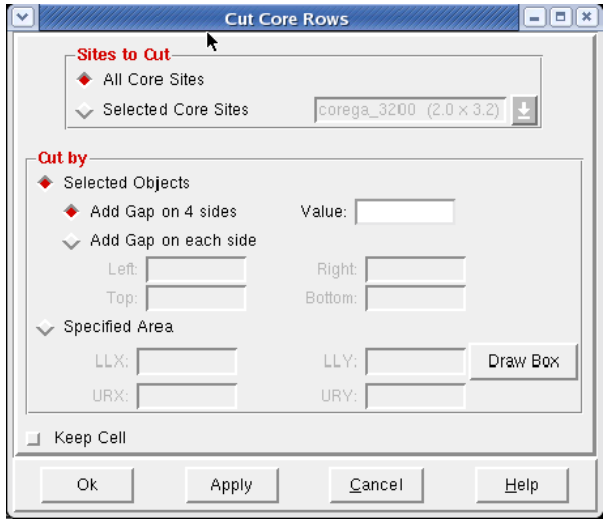

# Add Halo



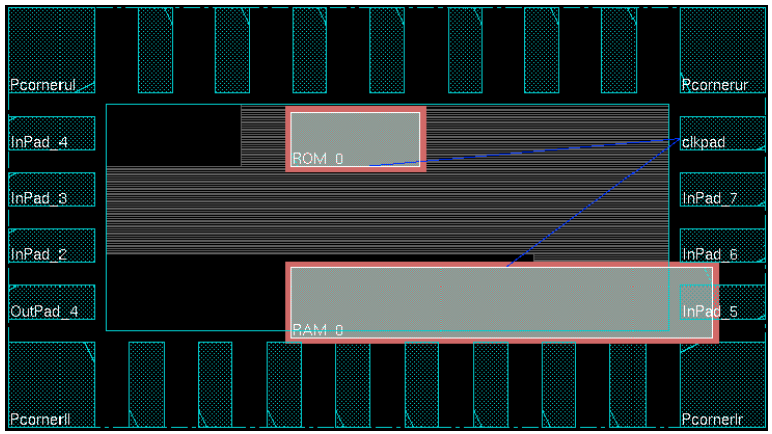

- Floorplan -> Edit Floorplan -> Edit Halo
- To create a ring around the memory macro, where no standard cells can be placed.
- Routing is still possible
- Be sure to specify a distance, e.g. 10 µm.



# Cut Rows

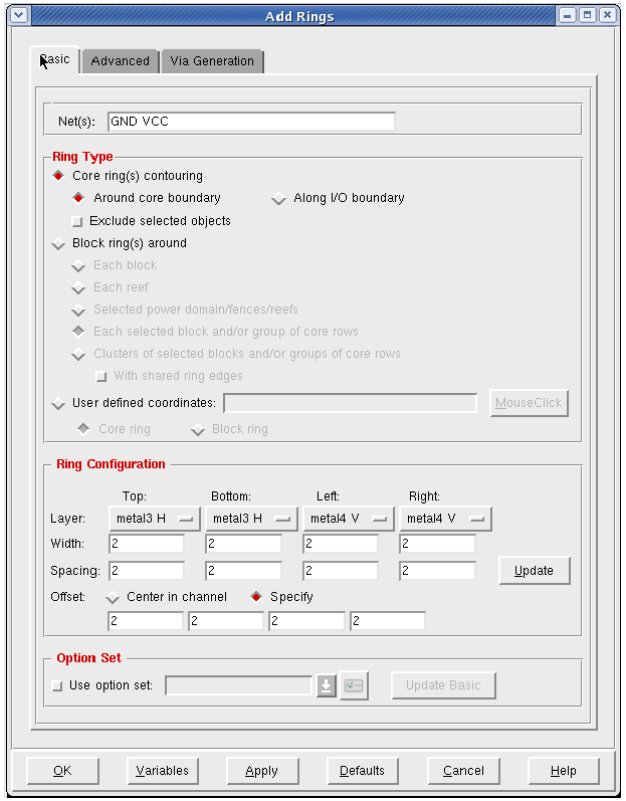



- Floorplan -> Row -> Cut Core Row
- Deletes core rows beneath memories.
- NOTE: Be sure to select memories before cutting.
- Now is a good time to save the design:
  File -> Save Design –
  Data Type: Encounter
- To restore:
  File -> Restore Design
- By moving memories, the cut rows are shown.
- Use undo to move back.



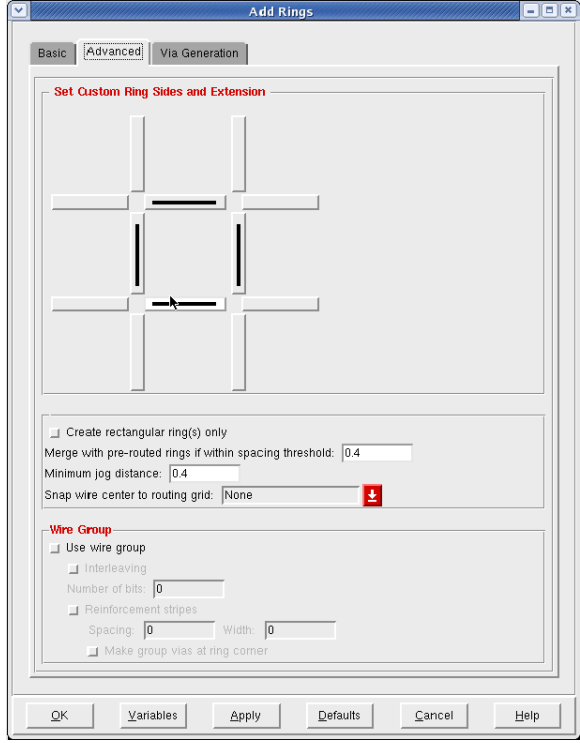
# Power rings

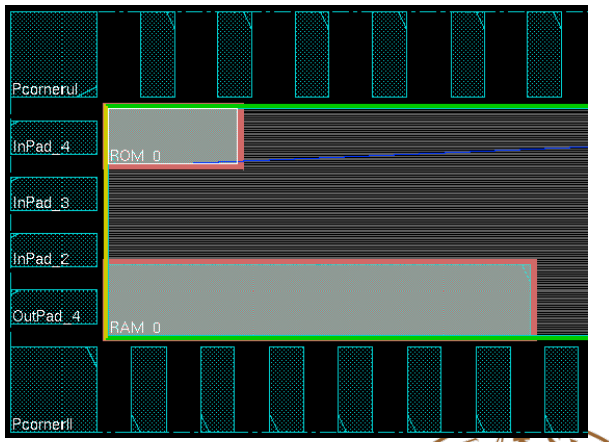
- Power -> Power Planning -> Add Rings
- To add Power rings around core specify Width: 2, Spacing: 2, Offset 2.
- Use metal3 for Horizontal wires and metal4 for Vertical wires.



# Power rings

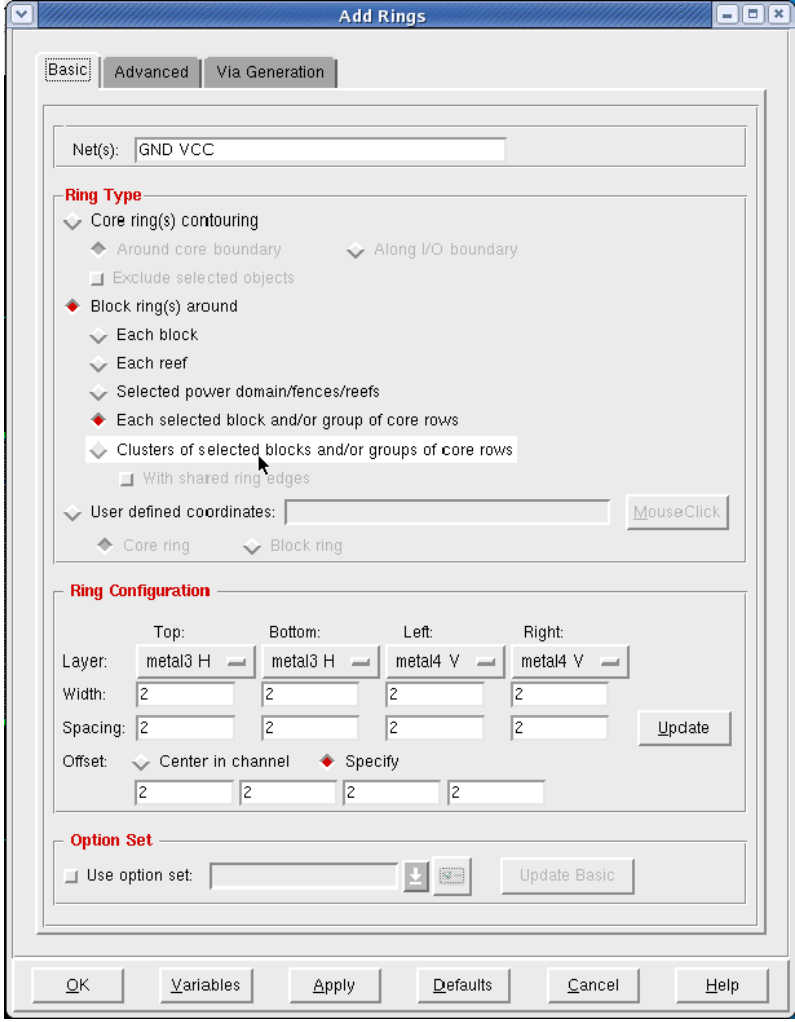
- Make sure that an entire ring is visible under the advanced tab.
- If applied correctly your design should look like this.



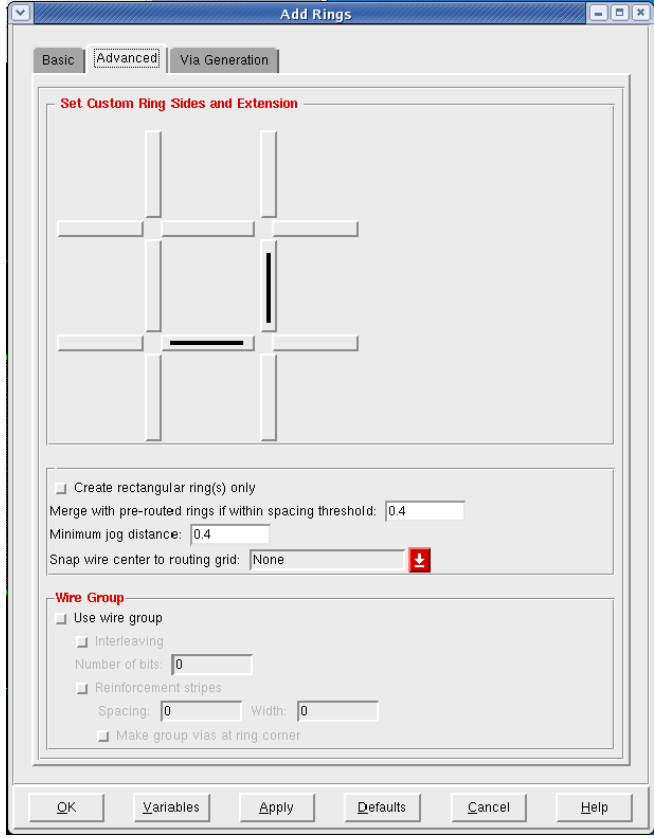
# Power rings

- Select the memory macros and select as in the figure.
- This will create a block ring around the memory block.
- Used to connect VDD and GND for memory.
- If memories are placed along the border of the die, some power-routing can be re-used.

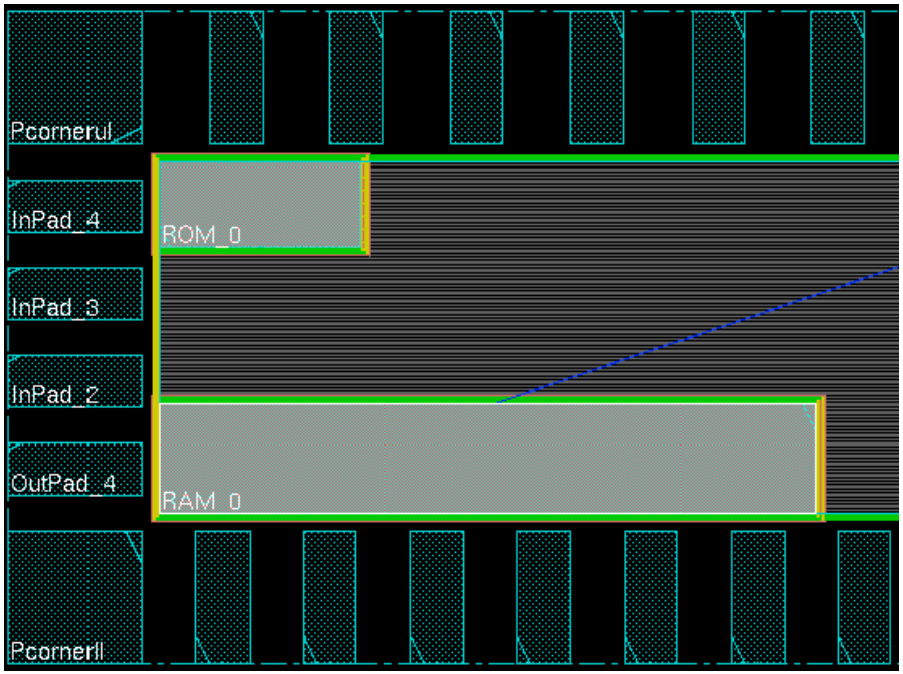



# Power rings

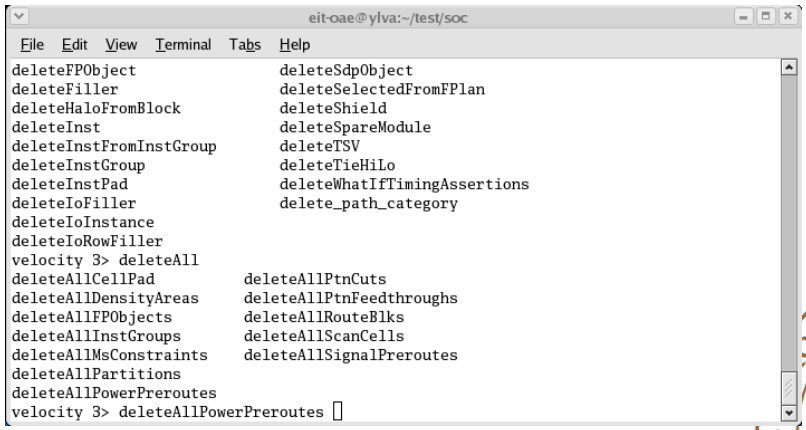
- For the upper memory no extra power routes are necessary for the top and left sides.



# Power rings

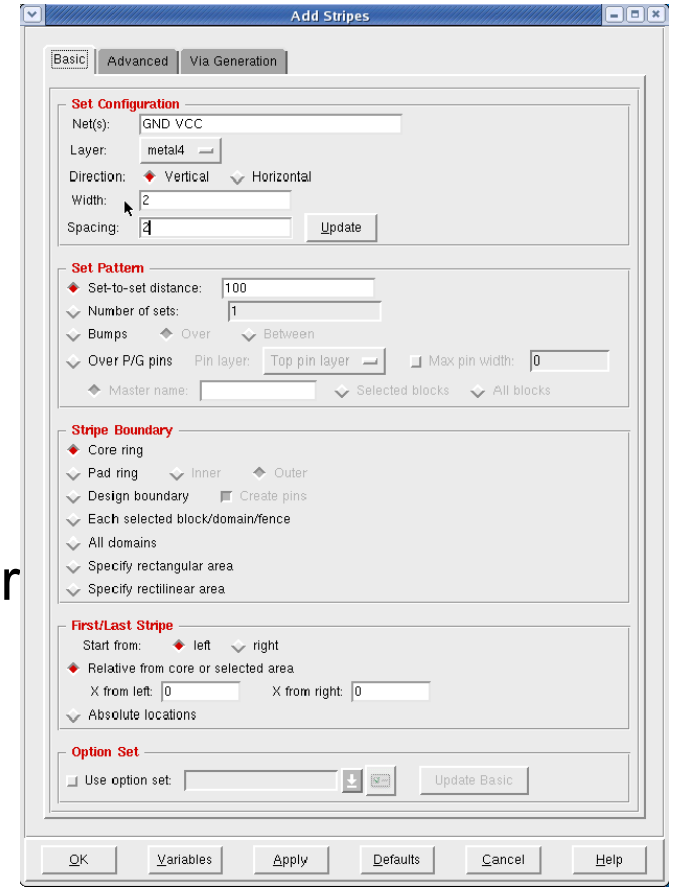


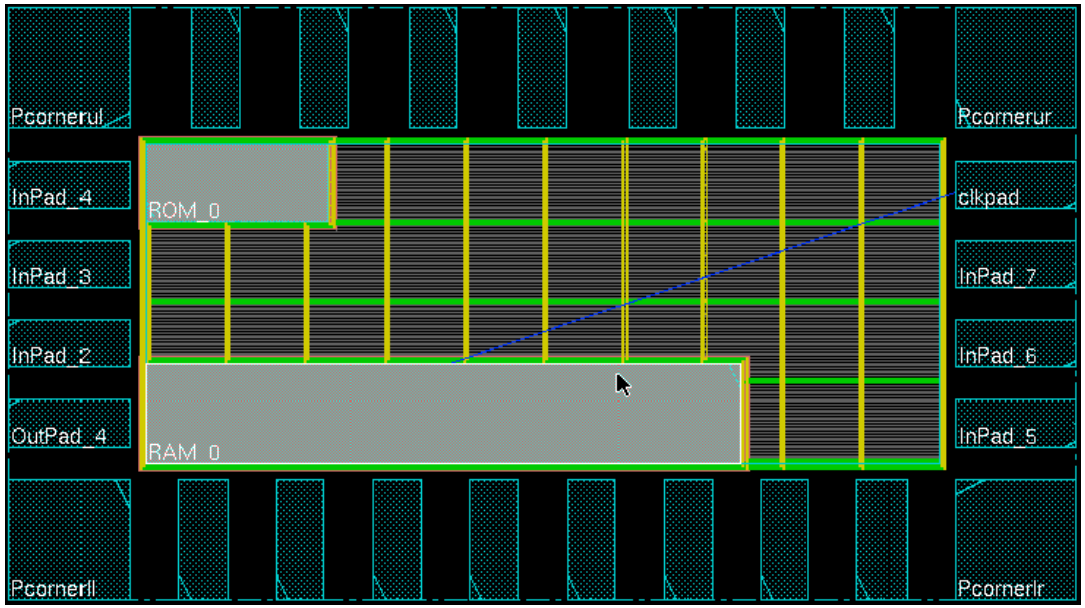

- If successfull the design should look like the picture.
- If not, type the command: ”deleteAllPowerPreroutes”, use Tab key to autocomplete.
- This command clears all power routing.



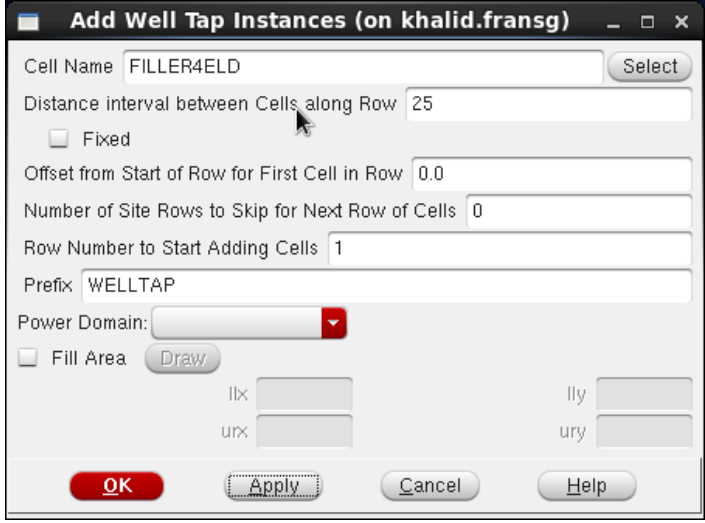
# Power stripes

- Power -> Power Planning -> Add Stripes
- Select metal4 for vertical and metal3 for horizontal



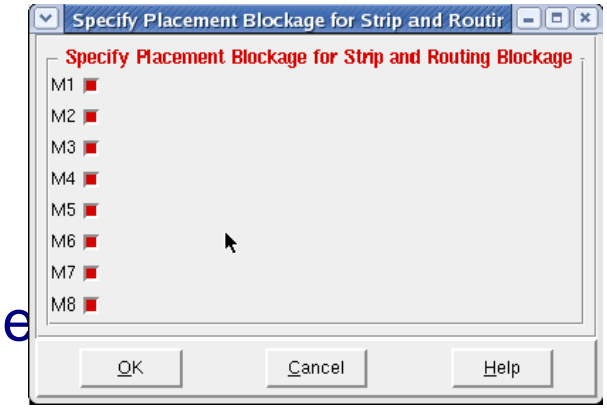
# Place well taps



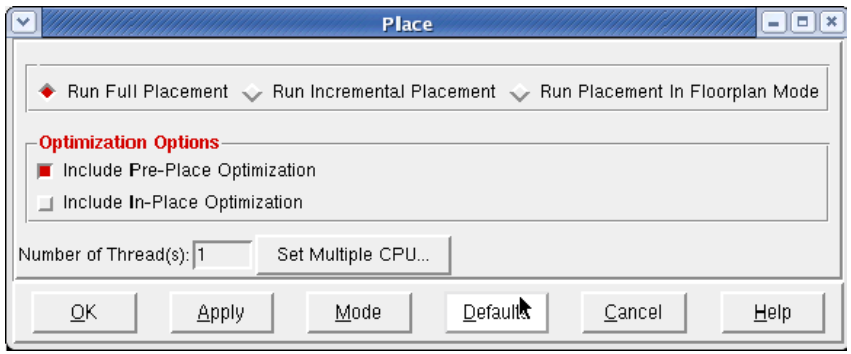

- Place -> Physical Cell -> Add well Tap

- Adds contacts for well and substrate.
- Use cell FILLER4ELD
- Use a spacing of 25µm.
- Prefix WELLTAP

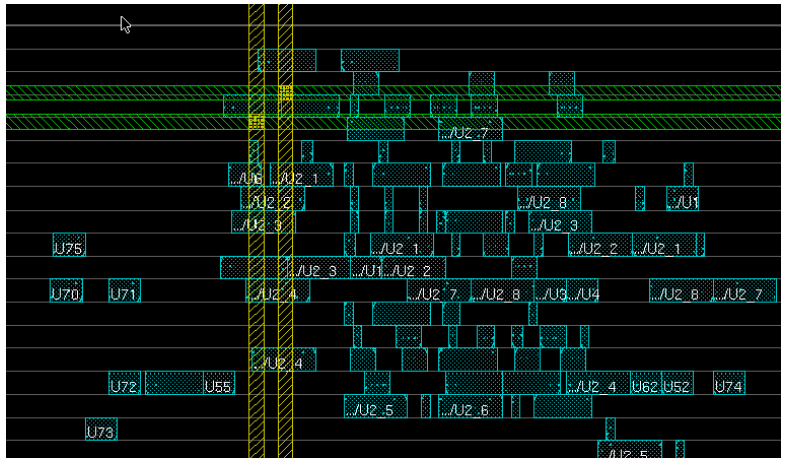
# Place standard cells

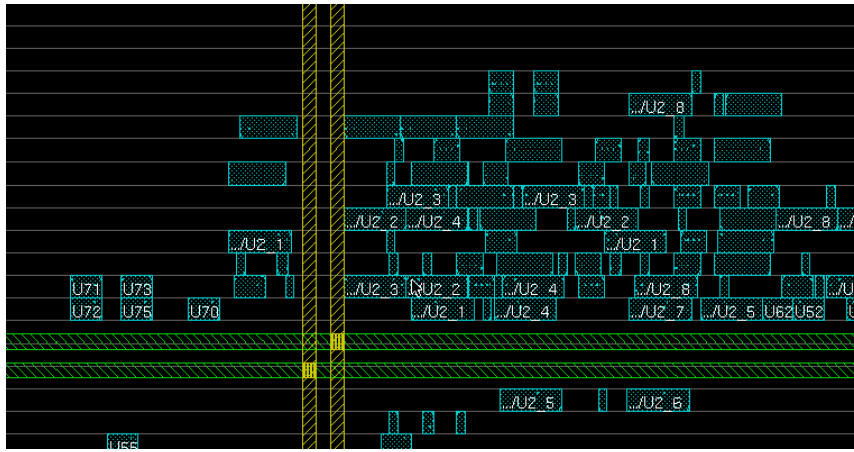



- Place -> Specify -> Placement Blockage
- Place -> Place Standard cells
- Change from ”Floorplan view” to ”Physical view” to see placed cells:

# Place standard cells

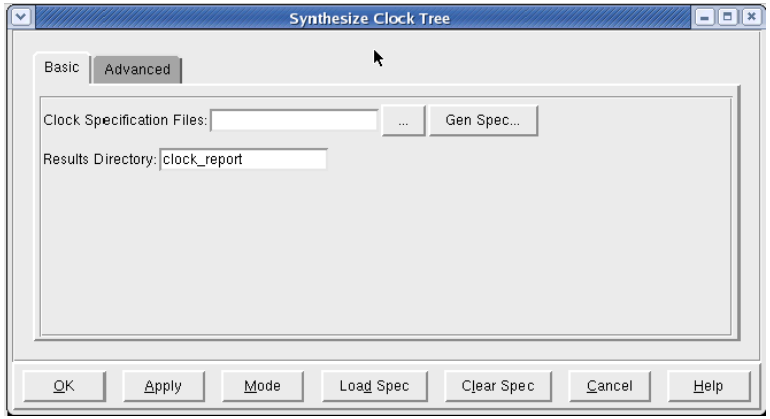


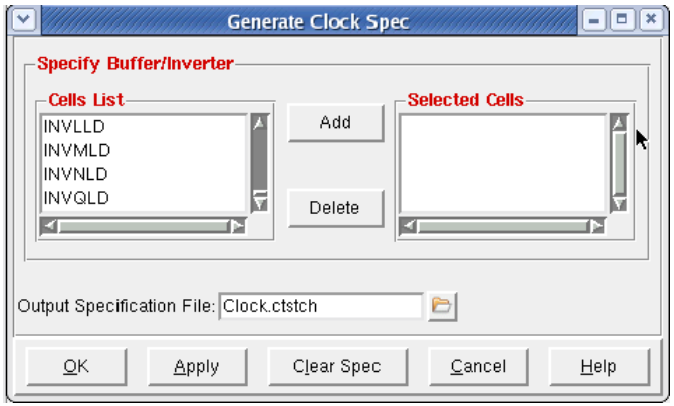




- Zoom in close to a crossing of two metal stripes in the middle of the die.
- Hide signal wires by expanding ”Net” in the right hand control ”Layer Control” and untick ”Net”.
- You should see that no cells are placed underneath metal stripes.
- Like in the lower figure.
- Now show the nets again.

# Design Clock

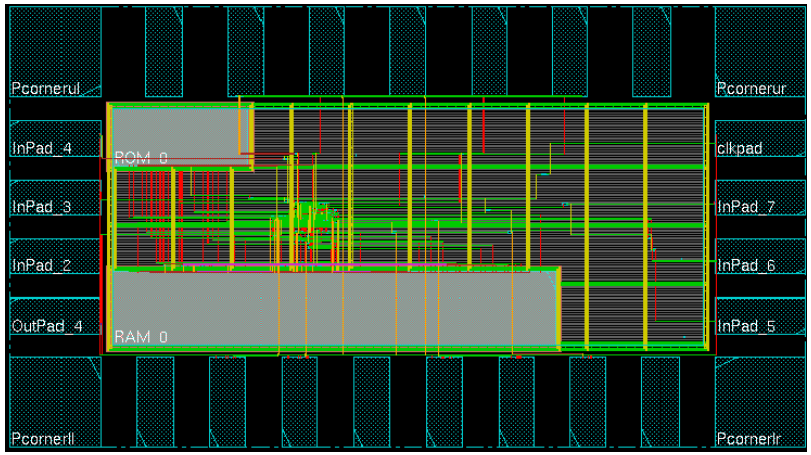
- Clock -> Synthesize Clock
- Click on Gen Spec and add all cells
- Second time use the (…) button to open your .CTSTCH file.



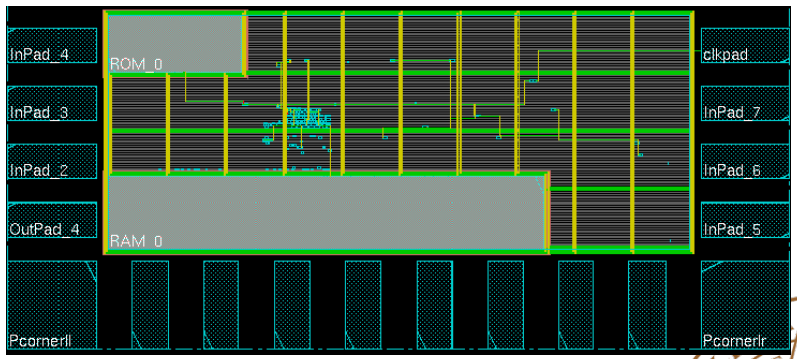



# Design Clock

- Now the synthesized clock with clock buffers and including a trial route of the remaining signals nets is shown.
- Type ”deleteTrialRoute” to delete the trial route and only show the clock net.



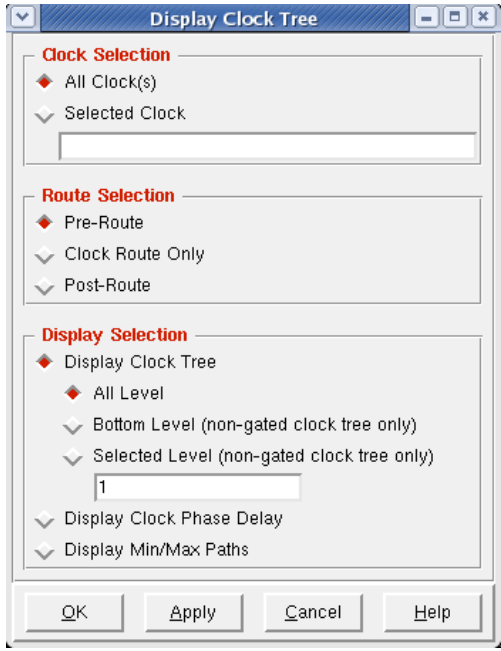

**With trial Route**

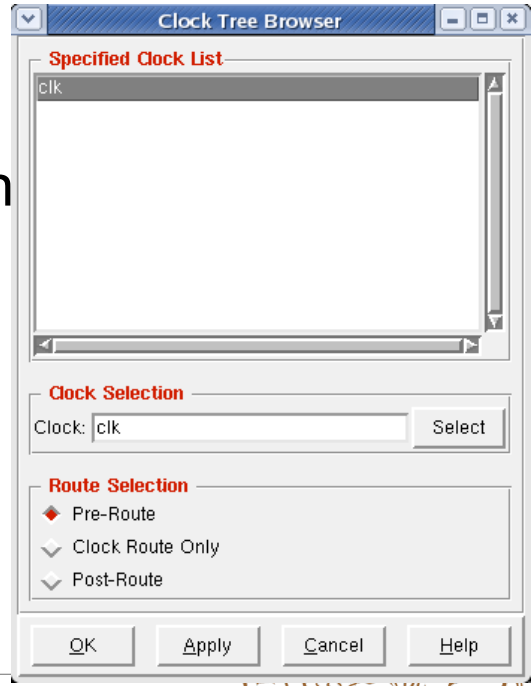



**Only clock net**

# Design Clock

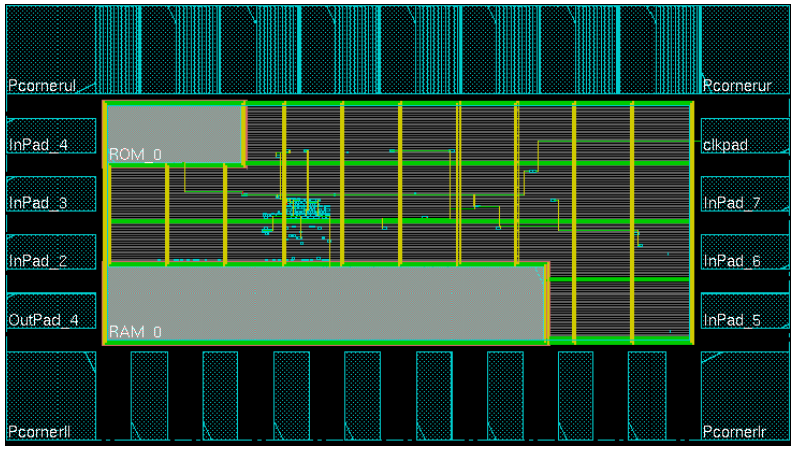
- It is possible to highlight the clock tree:
  Clock -> Display -> Display Clock Tree…
- Choose all clocks and All Level
- Clear it with
  Clock -> Display -> Clear Clock Tree Display.
- To see which buffers and inverters are used in the clock tree, use the clock tree Browser:
  Clock -> Browse Clock Tree



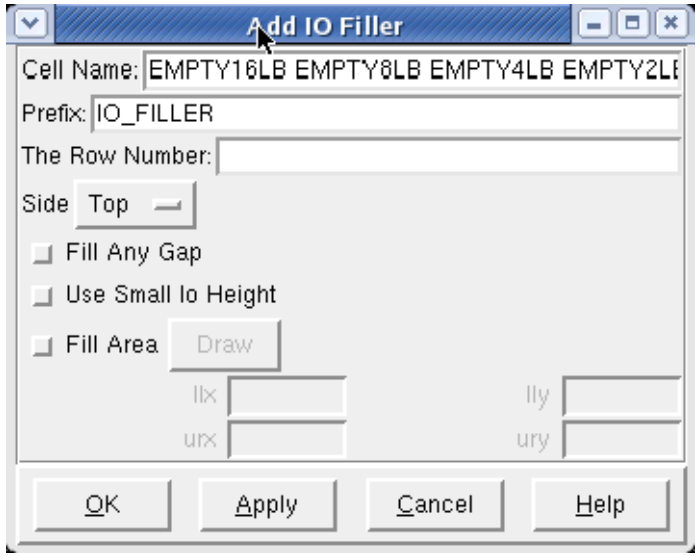



# IO Filler cells

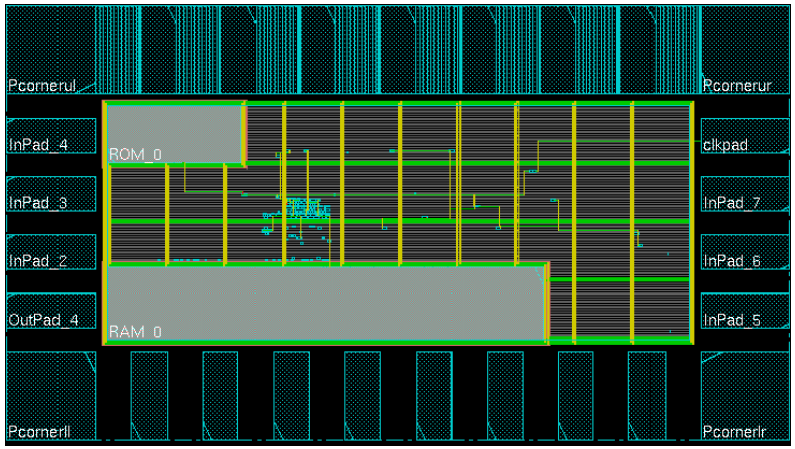
- Before placeing IO-fillers the pads need to be aligned on a 0.4µm x 0.4µm grid. As the width of minimum filler is 0.4µm.
- Select all pads in the top row except the right corner pad.
- Floorplan -> Edit Floorplan -> Space
- Enter a spacing value similar to the current spacing (Use ruler ”k” to measure).
- Choose ”Horizontal Spacing” and ”Fix Left”. For Right and Left side use ”Vertical spacing”.
- Verify with ruler that distance is a multiple of 0.4µm.



# IO Filler cells

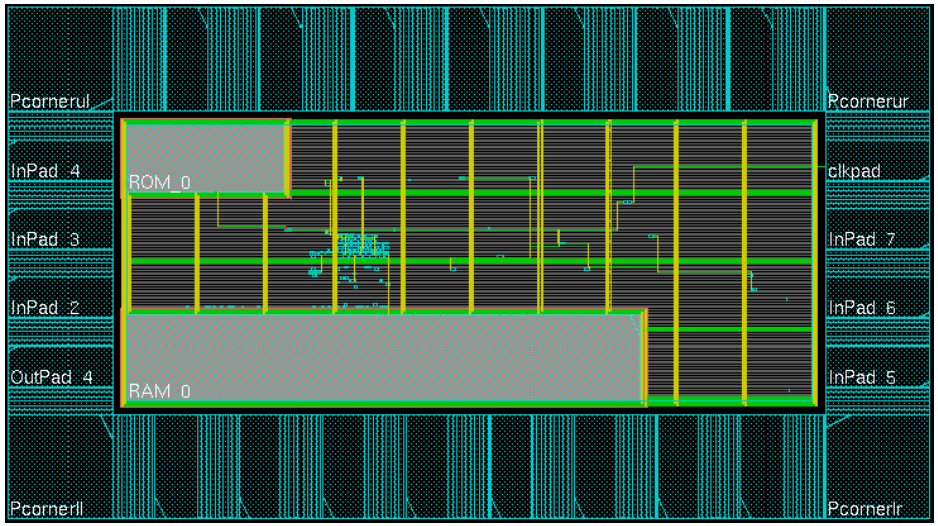



- Place -> Physical Cells -> Add I/O Filler
- Cells are named: EMPTY16LB EMPTY8LB EMPTY4LB EMPTY2LB EMPTY1LB
- Prefix: IO_FILLER
- Select which side to add to: Top/Bottom/Right/Left.
- The screen does not auto-refresh (press ”f”).
- Add to all sides.



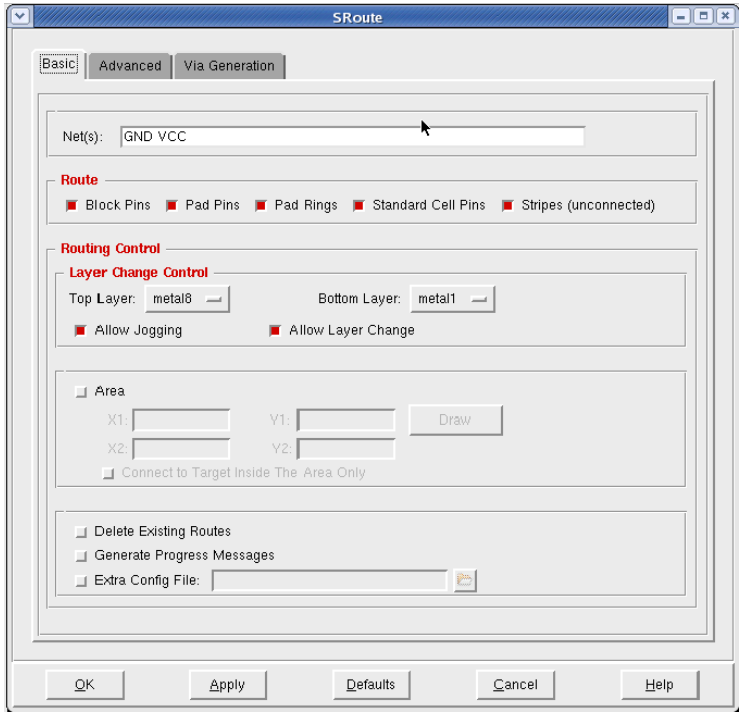
# IO Filler cells

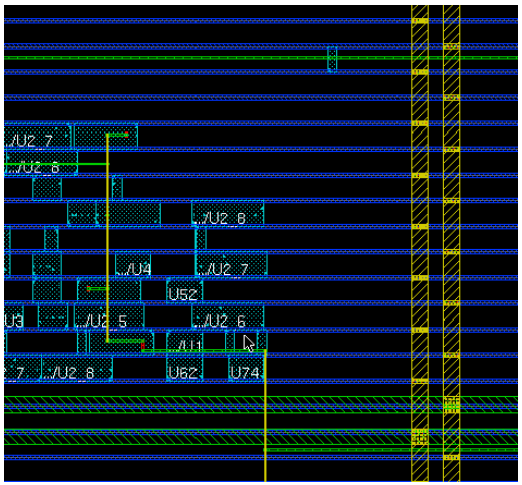
- If successfull design should look as in picture.

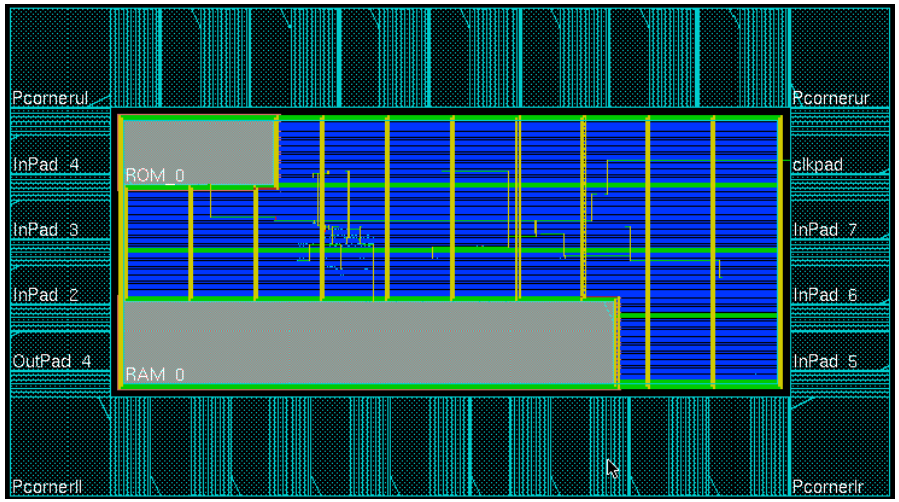


# Special Route

- Route -> Spceial Route
- Routes GND and VCC net for powering of standard cells.

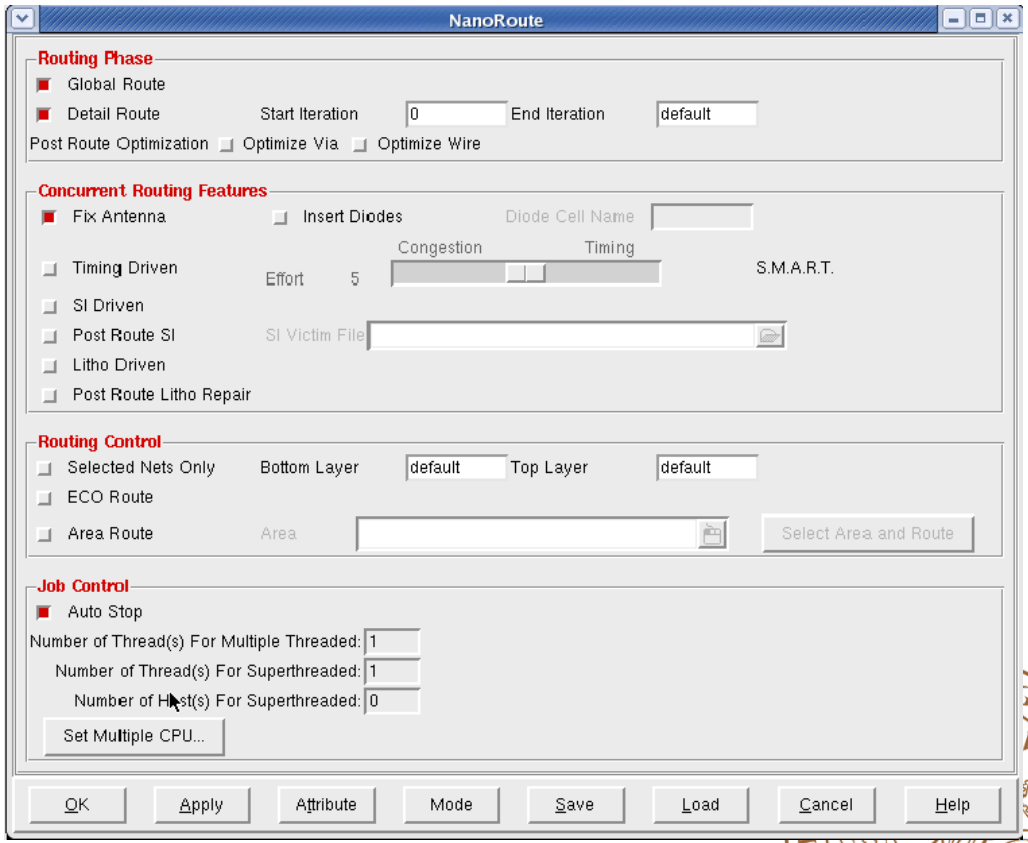




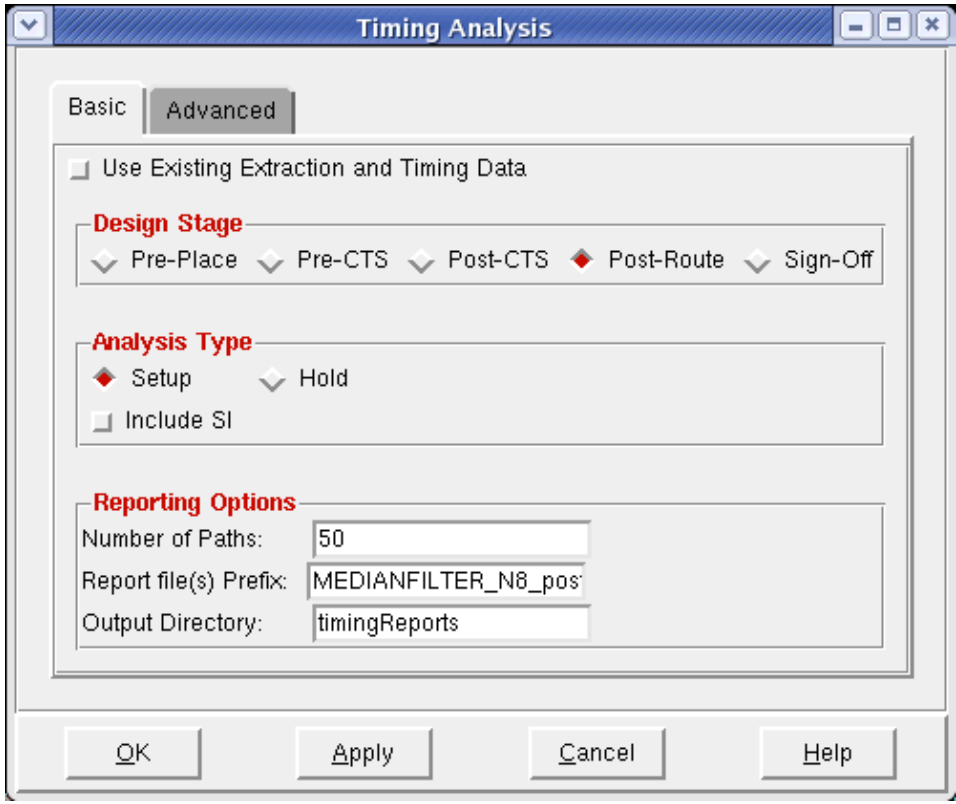
# Route normal nets

- Route -> Nanoroute -> Route
- Run with default options.



# Analyze Timing

- Before adding fillers we need to make sure that we meet timing, both setup and hold.
- Timing -> Report Timing
- Choose which design stage we are in, at this point Post-Route.
- Select both Setup and Hold, one after each other.



# Analyze Timing cont'd

- Make sure that we have no setup nor hold violations.
- WNS stands for Worst Negative Slack
- TNS stands for Total Negative Slack.
- If we do we need to run optimize timing, to solve this issue.

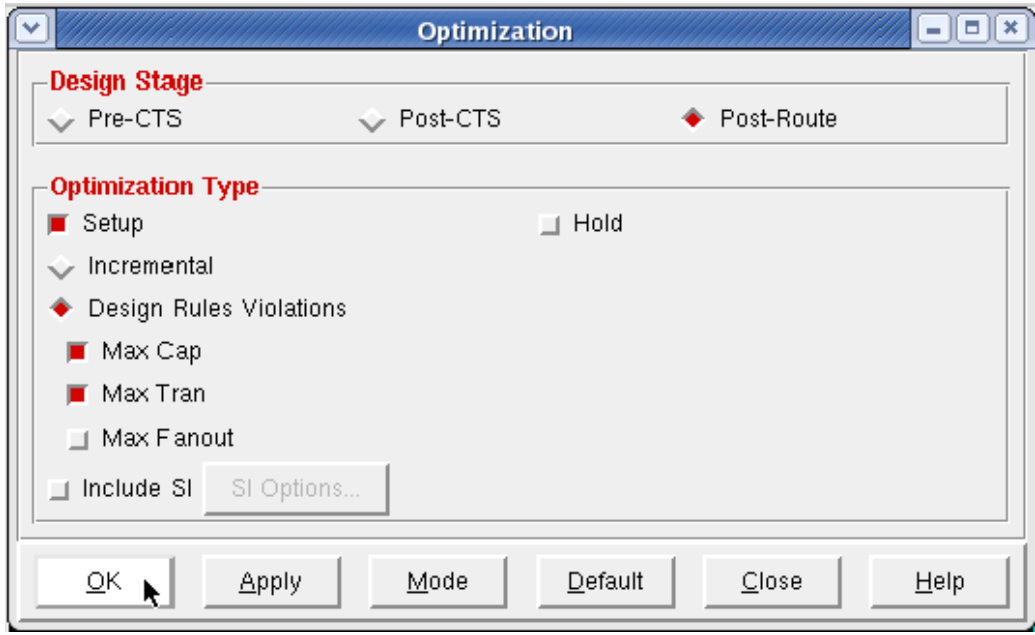
```
------------------------------------------------------------
     optDesign Final Summary
------------------------------------------------------------

+--------------------+---------+---------+---------+---------+---------+---------+
|     Setup mode     |   all   | reg2reg | in2reg  | reg2out | in2out  | clkgate |
+--------------------+---------+---------+---------+---------+---------+---------+
|           WNS (ns):|  5.242  |  5.242  | 15.707  |   N/A   |   N/A   |   N/A   |
|           TNS (ns):|  0.000  |  0.000  |  0.000  |   N/A   |   N/A   |   N/A   |
|    Violating Paths:|    0    |    0    |    0    |   N/A   |   N/A   |   N/A   |
|          All Paths:|   83    |   75    |    8    |   N/A   |   N/A   |   N/A   |
+--------------------+---------+---------+---------+---------+---------+---------+

+--------------------+---------+---------+---------+---------+---------+---------+
|     Hold mode      |   all   | reg2reg | in2reg  | reg2out | in2out  | clkgate |
+--------------------+---------+---------+---------+---------+---------+---------+
|           WNS (ns):| -0.019  | -0.019  |  0.074  |   N/A   |   N/A   |   N/A   |
|           TNS (ns):| -0.041  | -0.041  |  0.000  |   N/A   |   N/A   |   N/A   |
|    Violating Paths:|    4    |    4    |    0    |   N/A   |   N/A   |   N/A   |
|          All Paths:|   83    |   75    |    8    |   N/A   |   N/A   |   N/A   |
+--------------------+---------+---------+---------+---------+---------+---------+
```

# Fix Timing Violations

- Optimize -> Optimize Design
- Select your design stage, Post-Route.
- Select which violation you which to fix, such as Hold.




- Make sure to fix Max Cap, Max Tran and Max Fanout as well.
- TIP: Do not run all things at once.

# Fix Timing Violations cont'd

- Make sure that all violations are fixed.
- If not re-run optimize timing, and select the method to fix.
- If running only setup time make sure to run analyze timing for hold violations afterwards.

optDesign Final Summary

| Setup mode | all | reg2reg | in2reg | reg2out | in2out | clkgate |
|---|---|---|---|---|---|---|
| WNS (ns): | 5.049 | 5.049 | 15.699 | N/A | N/A | N/A |
| TNS (ns): | 0.000 | 0.000 | 0.000 | N/A | N/A | N/A |
| Violating Paths: | 0 | 0 | 0 | N/A | N/A | N/A |
| All Paths: | 83 | 75 | 8 | N/A | N/A | N/A |

| Hold mode | all | reg2reg | in2reg | reg2out | in2out | clkgate |
|---|---|---|---|---|---|---|
| WNS (ns): | 0.008 | 0.008 | 0.077 | N/A | N/A | N/A |
| TNS (ns): | 0.000 | 0.000 | 0.000 | N/A | N/A | N/A |
| Violating Paths: | 0 | 0 | 0 | N/A | N/A | N/A |
| All Paths: | 83 | 75 | 8 | N/A | N/A | N/A |

# Fix Timing Violations cont'd

- If no reg2out, in2reg paths are found, input delay and/ or output delay are missing.
- See synthesis slides.

optDesign Final Summary

| Setup mode | all | reg2reg | in2reg | reg2out | in2out | clkgate |
|---|---|---|---|---|---|---|
| WNS (ns): | 5.049 | 5.049 | 15.699 | N/A | N/A | N/A |
| TNS (ns): | 0.000 | 0.000 | 0.000 | N/A | N/A | N/A |
| Violating Paths: | 0 | 0 | 0 | N/A | N/A | N/A |
| All Paths: | 83 | 75 | 8 | N/A | N/A | N/A |

| Hold mode | all | reg2reg | in2reg | reg2out | in2out | clkgate |
|---|---|---|---|---|---|---|
| WNS (ns): | 0.008 | 0.008 | 0.077 | N/A | N/A | N/A |
| TNS (ns): | 0.000 | 0.000 | 0.000 | N/A | N/A | N/A |
| Violating Paths: | 0 | 0 | 0 | N/A | N/A | N/A |
| All Paths: | 83 | 75 | 8 | N/A | N/A | N/A |

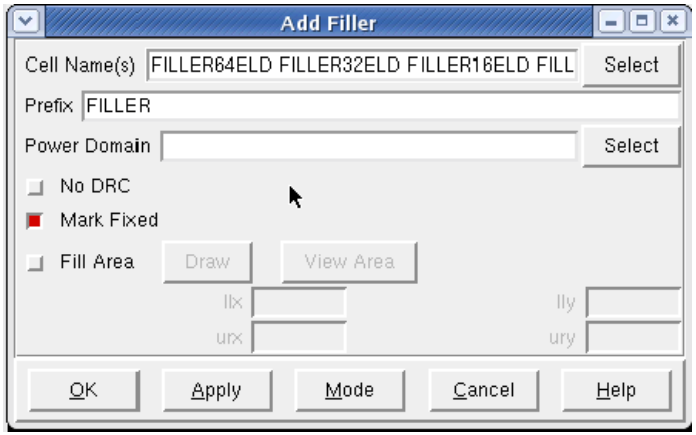
# Another note on Timing Violations

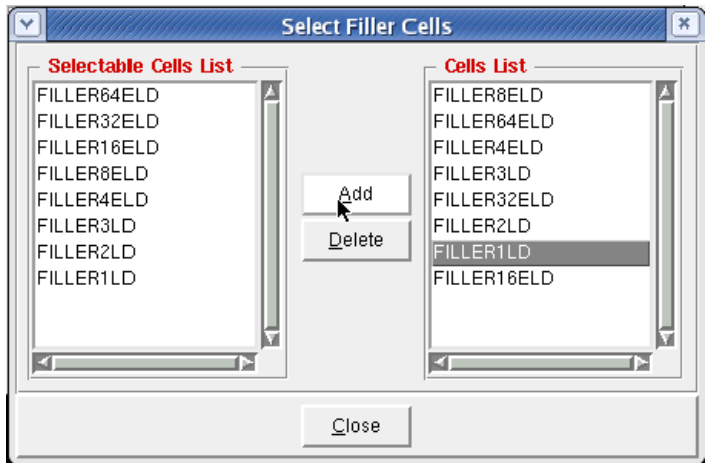
- If timing violations are still seen after mutliple runs with a larger design, what to do?

- Run Optimize Timing after each design stage, i.e:
  - After placement (before Clock Tree Synthesis) Pre-CTS
  - After Clock Tree Synthesis – Post-CTS
  - After Routing – Post-Route
- Try using incremental optimization for hold after the setup optimization.
- Reduce clock speed.
- Increase die size, i.e., have a lower die utilization.

# Add Filler cells

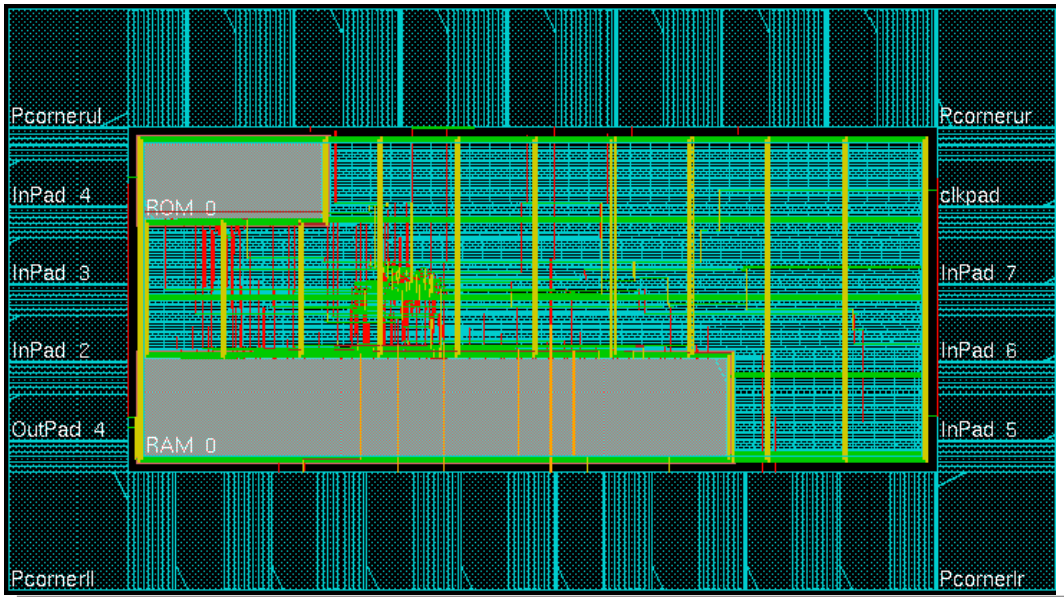
- Place -> Physical Cells -> Add Filler.
- Be sure to select the largest fillers first to use them when possible



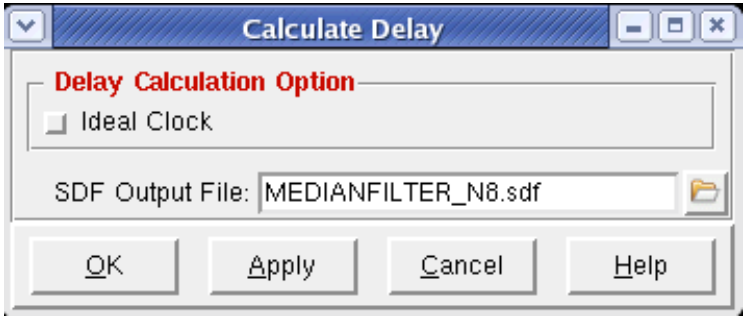

# Add Filler cells

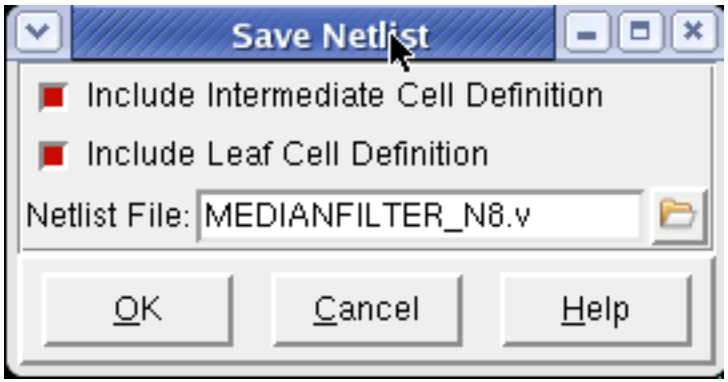
- Final design should look as in picture if Metal 1 is set to not visible in the drawer in the right hand side of Encounter.

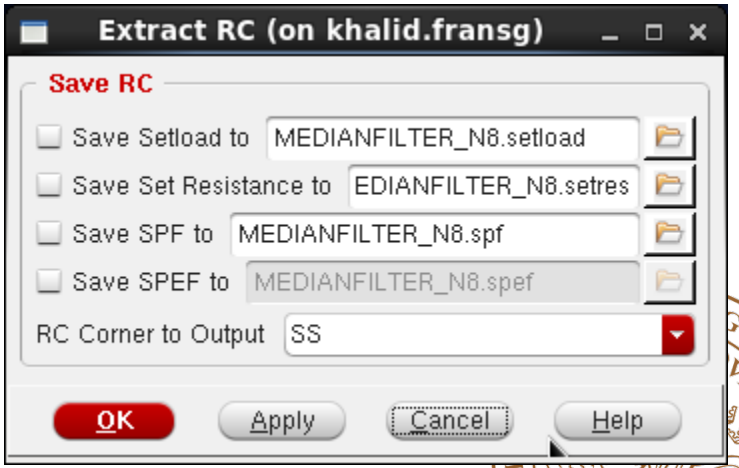


# Export Netlist and SDF

- To simulate your design in Modelsim with correct timing annotation, you need to export a netlist (your design) and SDF (Synopsys Delay Format, timing annotation).
- Timing -> Write SDF
- Be sure to untick Ideal Clock
- This dialog runs the ”write_sdf” command in the background. However, to avoid errors in ModelSim, use this ”write_sdf” command in your script:
  write_sdf -version 2.1 -interconn noutport file.sdf
- File -> Save -> Netlist
- It is also possible to save SPF, SPEF for further use in Power and Timing analysis.
- Timing -> Extract RC (Chose which RC corner to output.)



# Save Design and Restore Design

- To save and restore your design use:
- Save
  - File -> Save Design
- Restore
  - Design -> Restore Design

# Re-run entire placement

- When developing a chip, placement might be run many times and each indiviual step may take a long time, and a few hours is not uncommon.
- To avoid waiting for each step to finish running a script is easier.
- Enounter saves all commands entered in a file called **encounter.cmd** with an added digit for every run, i.e. **encounter.cmd23** if you you are running for the 23rd time in the same directory.
- **IMPORTANT!** Don't re-run this file directly as it contains every single change performed, including zooming in and out.
- Copy the file and remove unnecessary commands, e.g., zoom, fit.
- Type **source filename.cmd** to execute an encounter script.