Tutorial – Cadence Innovus

Dae Hyun Kim

EECS
Washington State University

Cadence Innovus

- Used for automatic physical design (P&R)

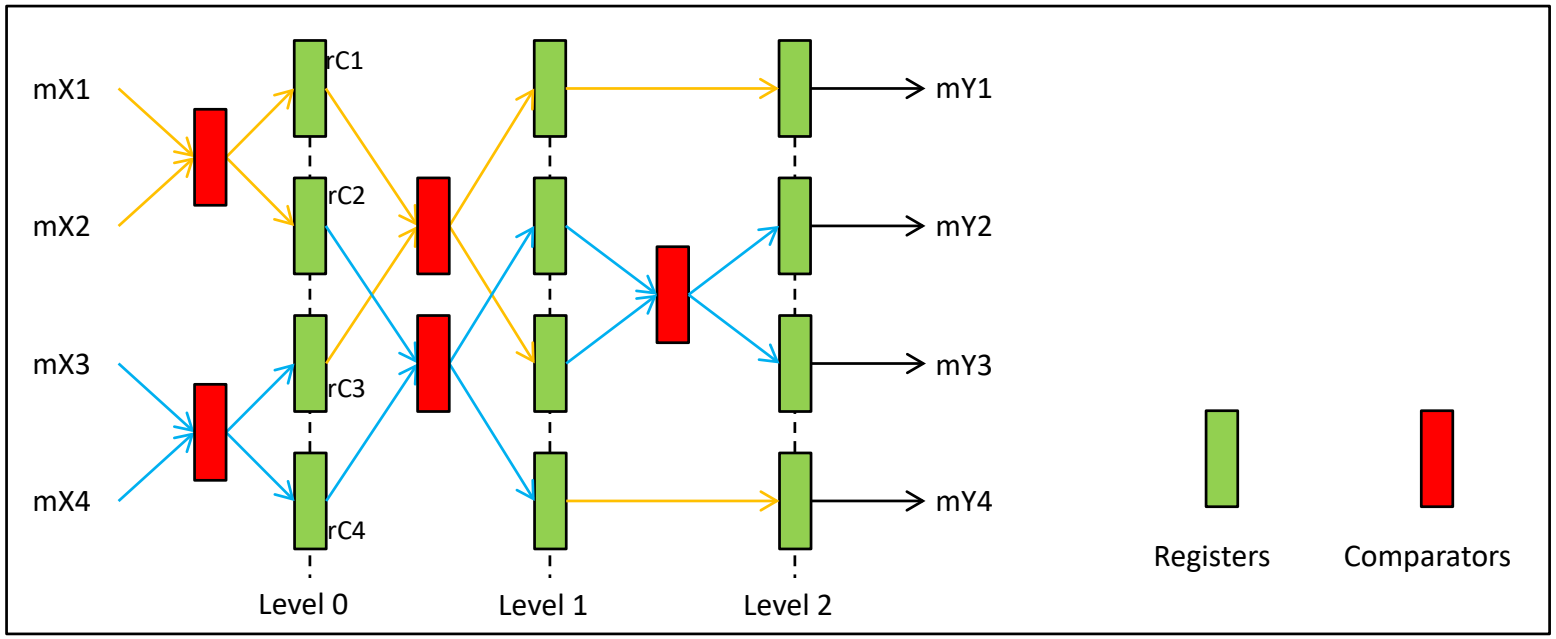
- Performs
  - Floorplanning
  - Placement
  - Clock-tree synthesis (CTS)
  - Routing
  - Optimization
  - Analysis (timing, power, ...)
  - ...

Tutorial

- Download the following file into your work directory.
  - wget http://eecs.wsu.edu/~ee434/Labs/tutorial innovus.zip
- Unzip it.
  - unzip tutorial_innovus.zip
- Source
  - source ictools_generic.sh
  - source cadence_innovus17.sh

Benchmark

- VQS64_4 (four-input 64-bit pipelined quick sort)
  - input [63:0] mX1, mX2, mX3, mX4; // four input data
  - input mCLK; // clock
  - output [63:0] mY1, mY2, mY3, mY4; // four output data

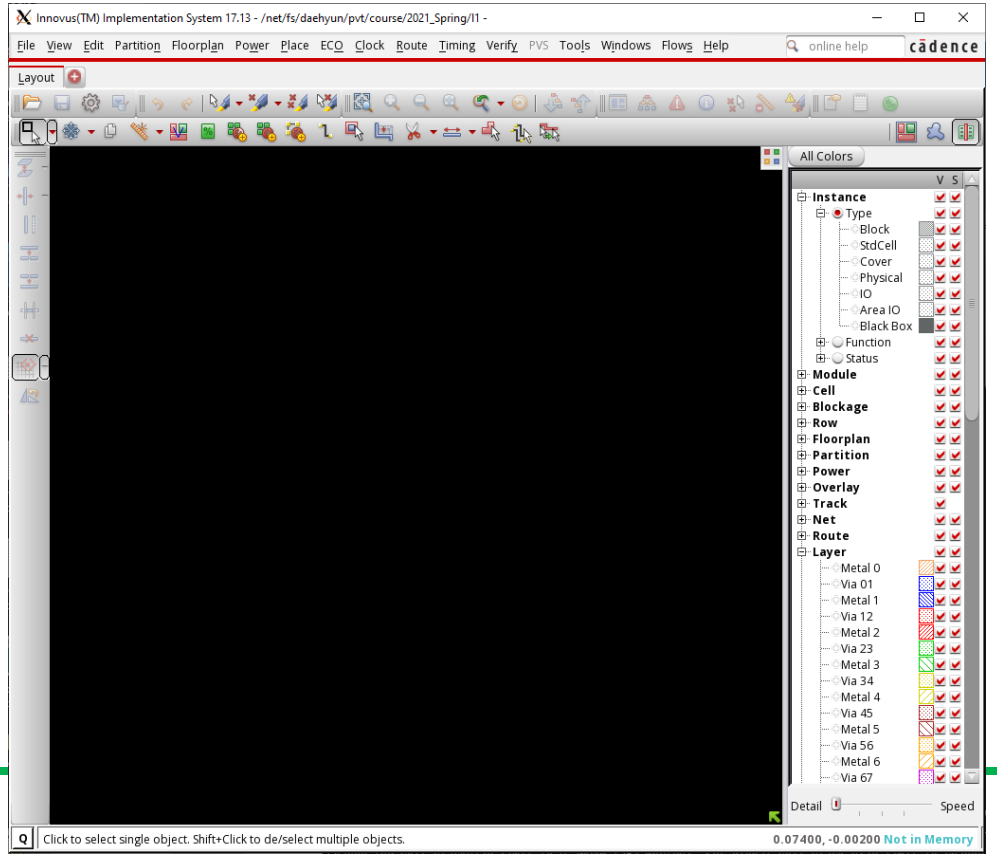


Physical

1. Chip outlining
2. P/G network design
3. Placement
4. Pre-CTS optimization
5. CTS
6. Post-CTS optimization
7. Routing
8. Post-routing optimization

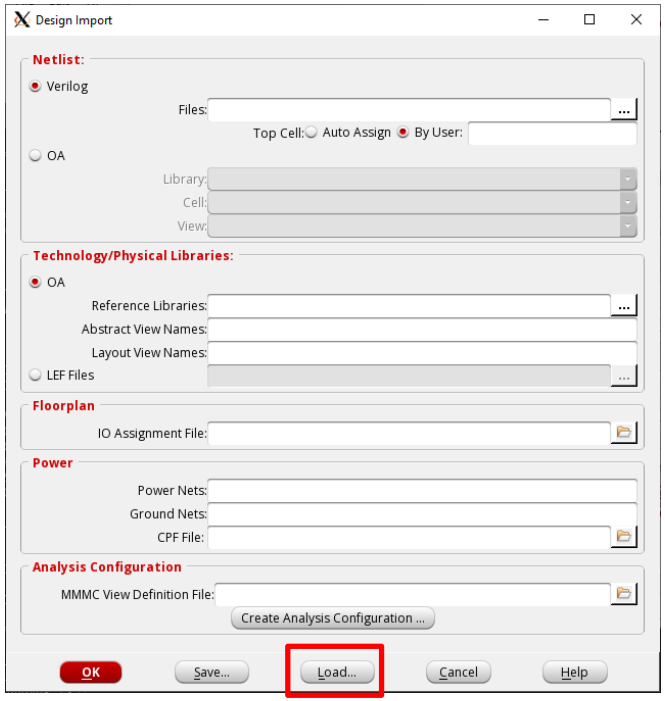
1. Chip Outlining

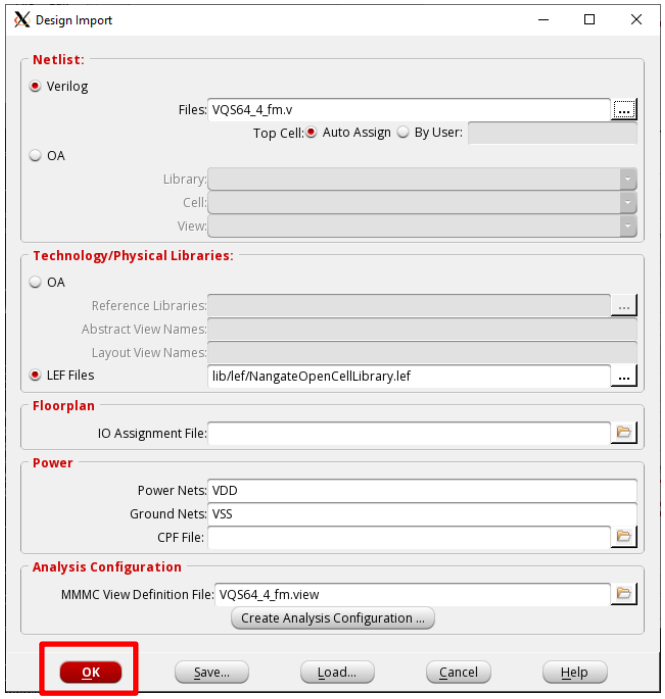
- Run Innovus.
  - innovus
- See the terminal window. It’s similar to Synopsys DC. You can use the GUI or the terminal to do something.



1. Chip Outlining

- Click “File” → “Import Design...”.
  - (The GUI might be a bit slow, so be patient.)
- In the “Design Import” window, click “Load…” and choose “VQS64_4_m.globals”. This will automatically fill up the settings. Then, click “OK”.



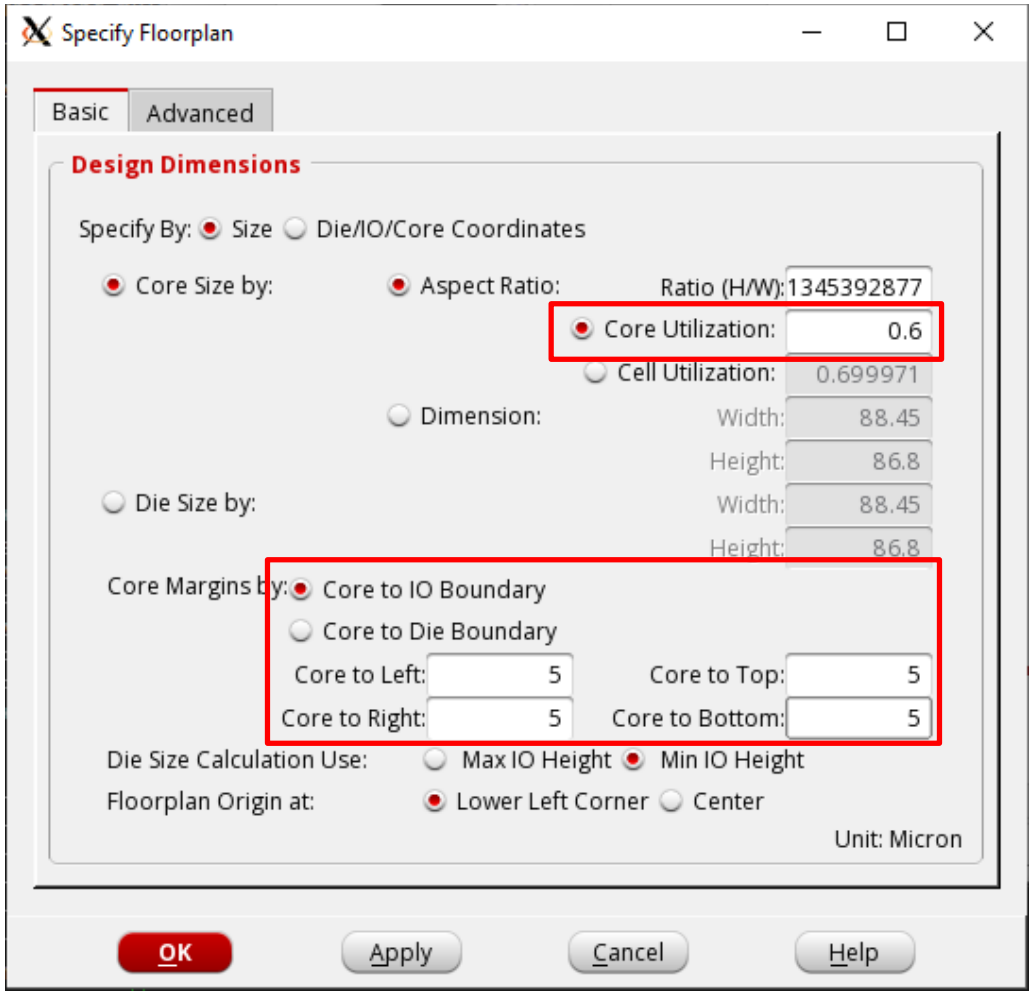


1. Chip Outlining

- See the terminal for Innovus messages. There might be some Error or Warning messages. You can ignore them.
- In the Innovus main window, press “f” to see the outline of the layout.
- Innovus automatically computes and prepares the layout area.

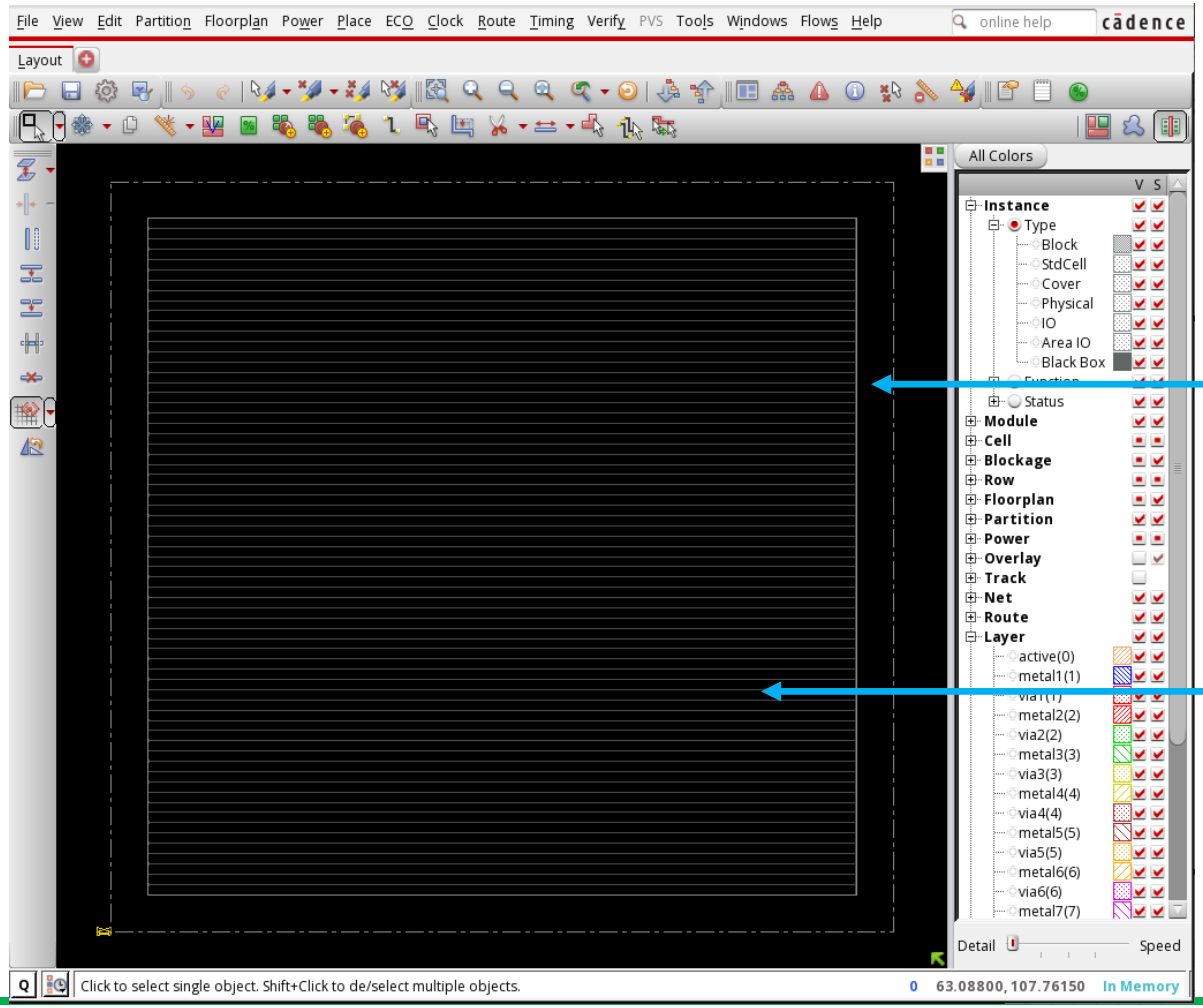
- Let’s modify the layout area.
- In the main window, click “Floorplan” → “Specify Floorplan…”.
- Set the core utilization to 0.6.
- Set the core-to-left, core-to-top, core-to-right, and core-to-bottom to 5.0.
- Then, click OK. (See the next page)

1. Chip Outlining



1. Chip Outlining

- Now, you will see the following window.



Power/Ground rings will be laid out in this area.

Std. cells will be placed in this core area.

How to Save the Design

- Let’s save the current design.

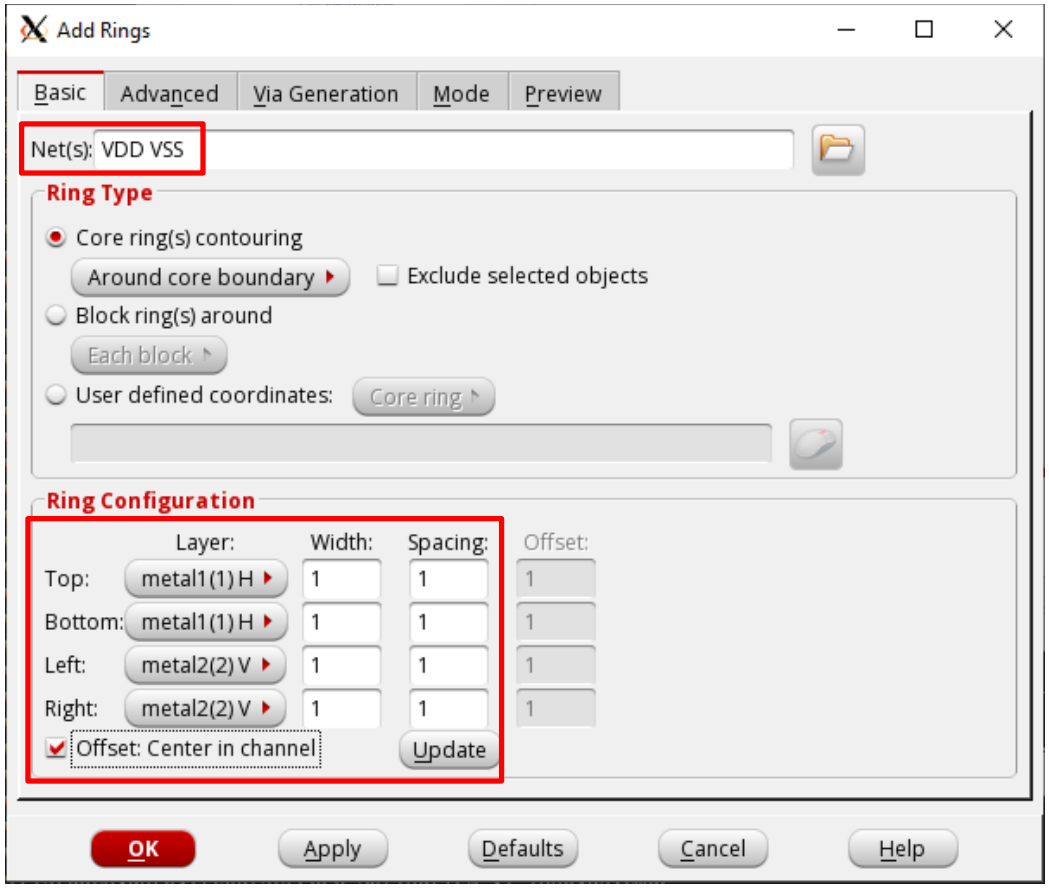
- In the terminal, run the following command to save the current design into “test_01_floorplan.enc”.

  innovus #> saveDesign test_01_floorplan.enc

- Later on, you can load the design as follows.
  - Run Innovus, click File → Restore Design → Data Type: Innovus → select the .enc file.

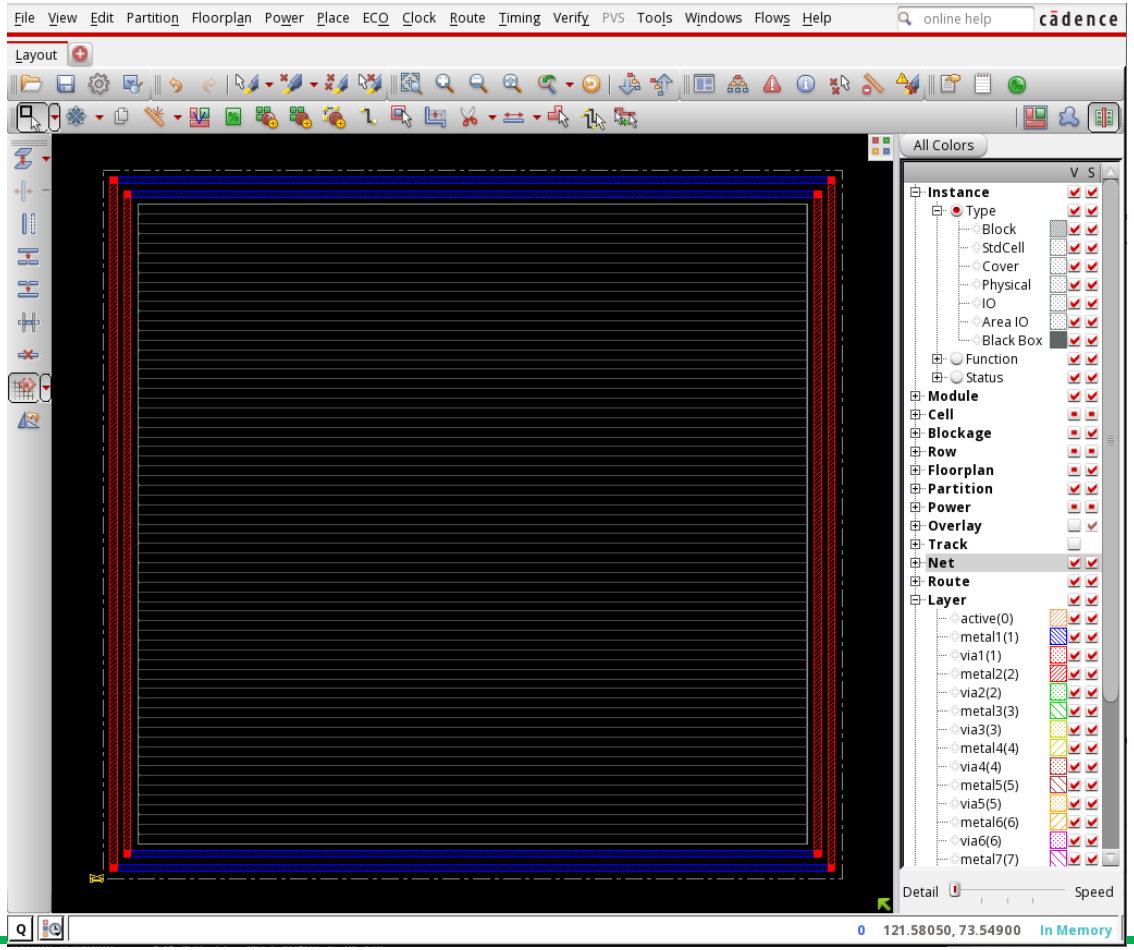
2. P/G Network Design

- Click “Power” → “Power Planning” → “Add Rings…”.



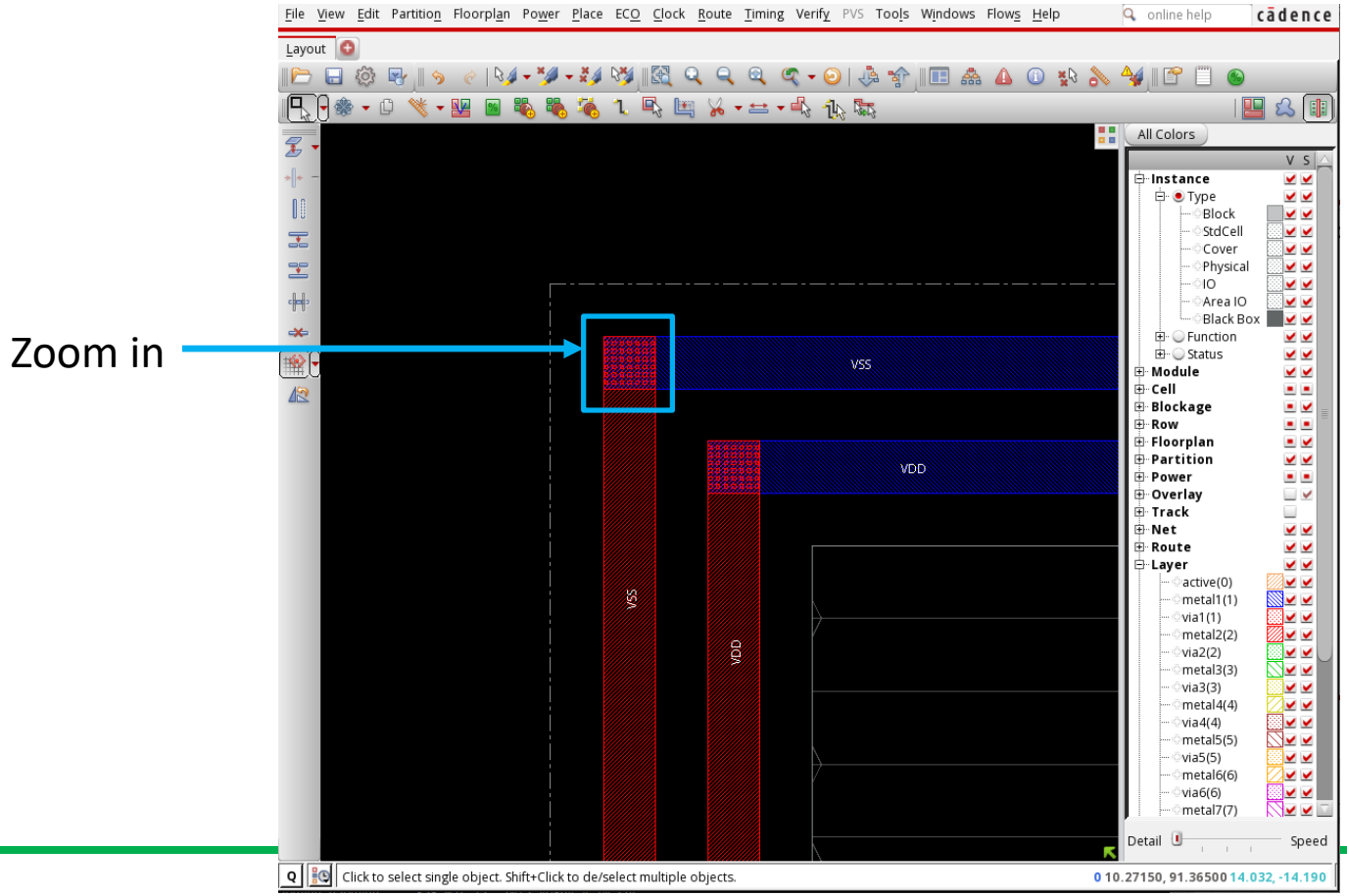
2. P/G Network Design

- Fill in the input boxes as shown in the previous page and click OK. Now you can see the power and ground rings.



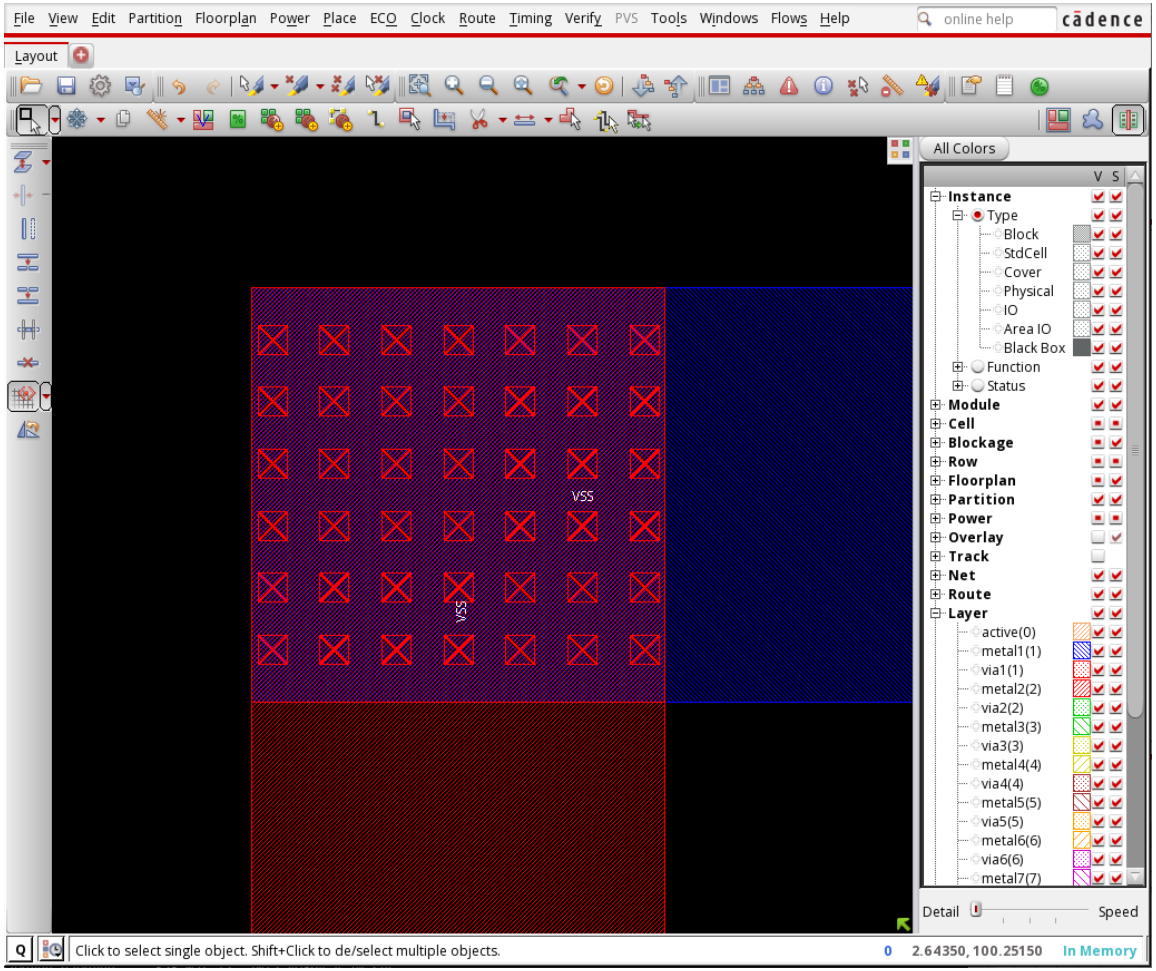
2. P/G Network Design

- Zoom in the top-left corner (Mouse right click – hold – drag – release). As shown below, the outer ring is VSS and the inner ring is VDD. Blue: Metal 1. Red: Metal 2. Zoom in the via arrays.



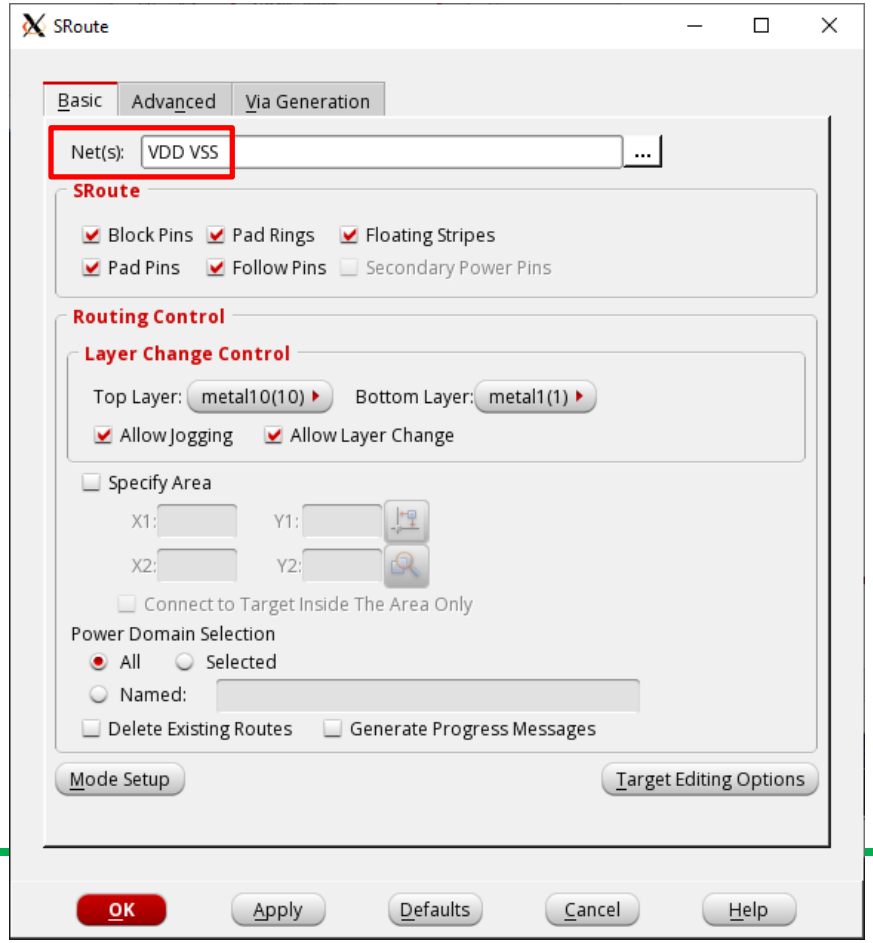
2. P/G Network Design

- The “X” squares are vias connecting the M1 and M2 wires.



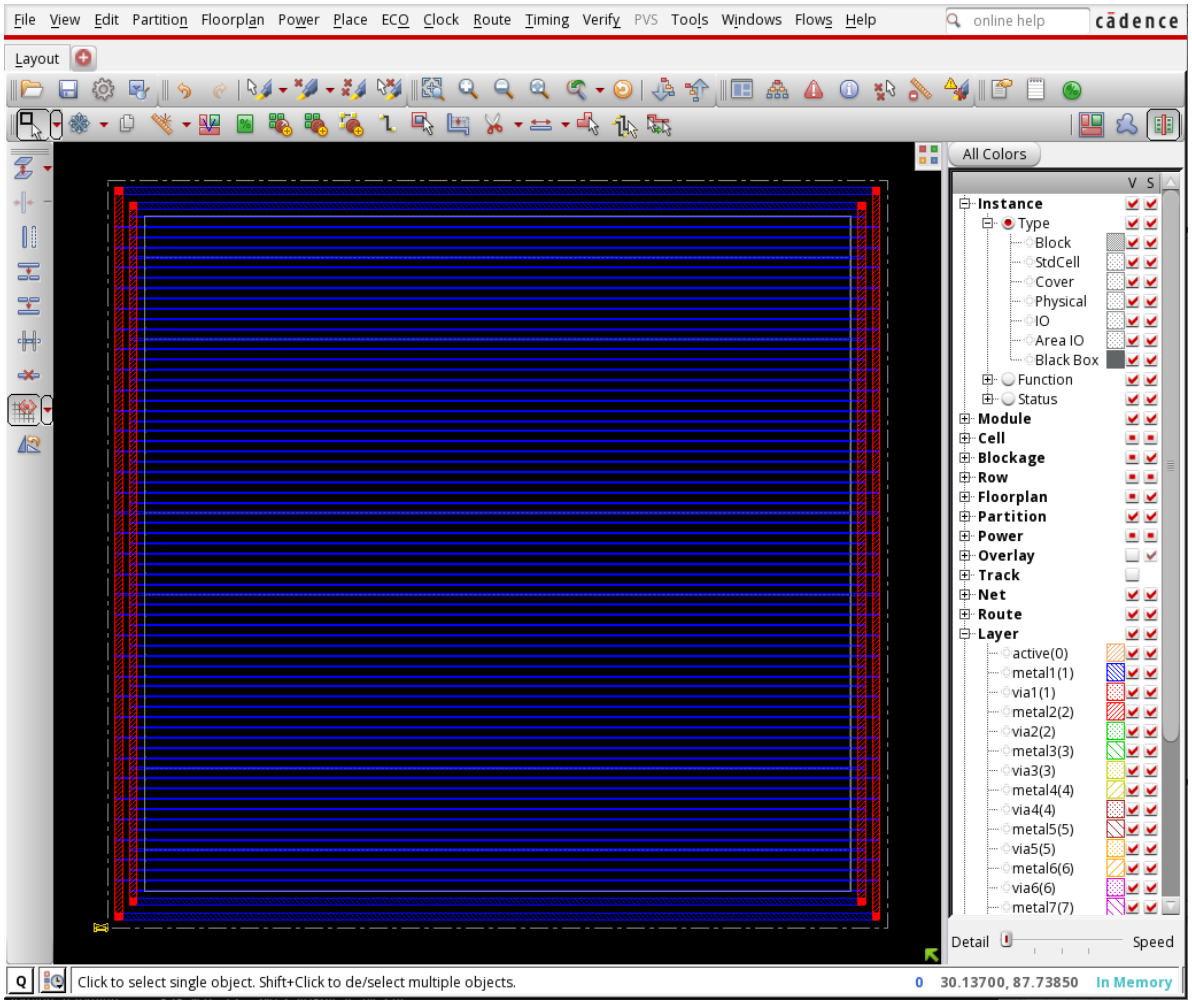
2. P/G Network Design

- Press “f” to zoom out to the full design.
- Now, we will draw power/ground stripes to connect the P/G rings to standard cells.
- Click “Route” → “Special Route…”.



2. P/G Network Design

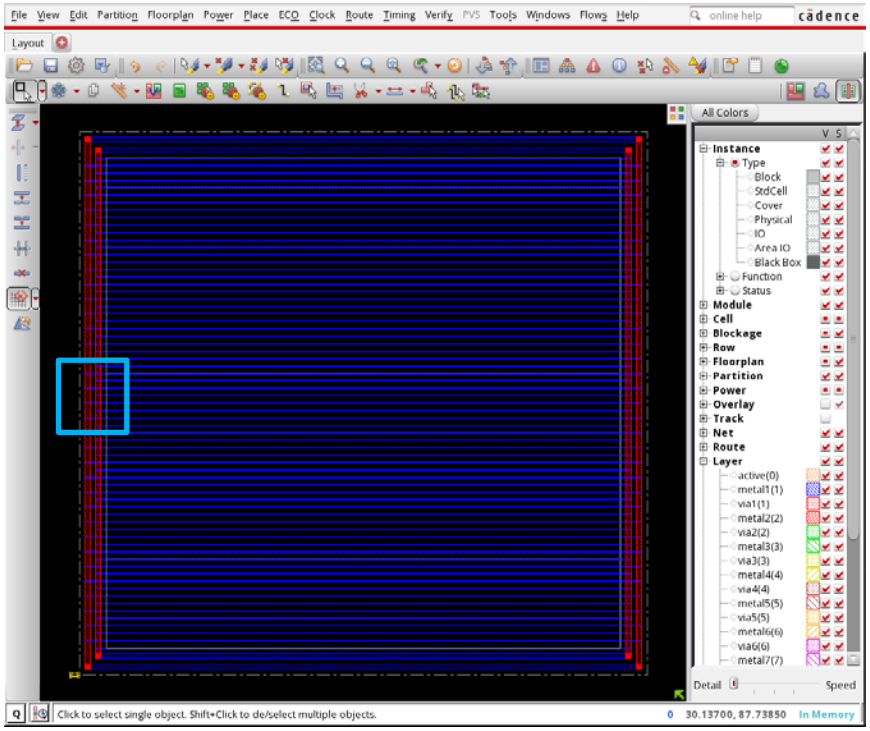
- P/G network



2. P/G Network Design

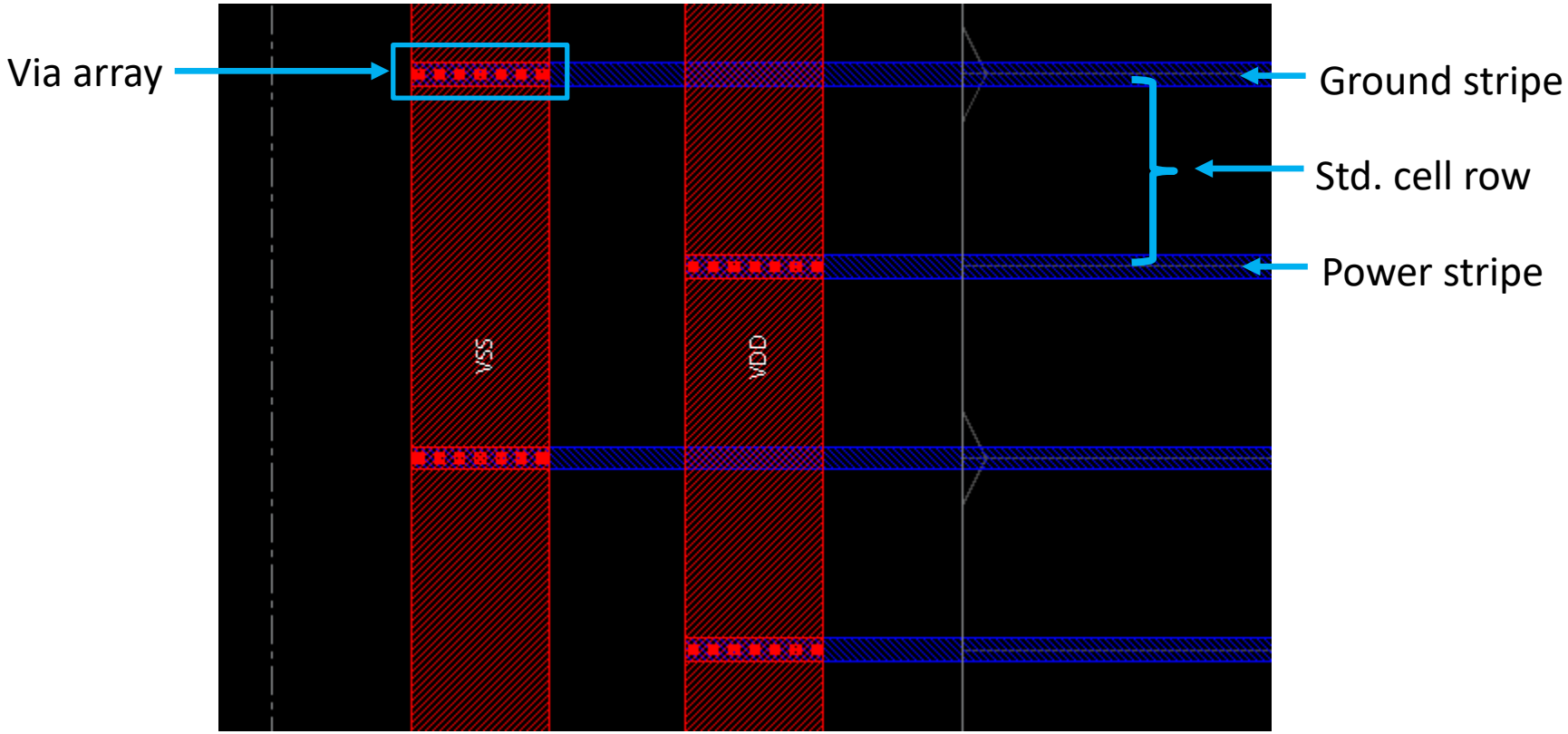
- saveDesign test_02_pg.enc

- Zoom in the following area.



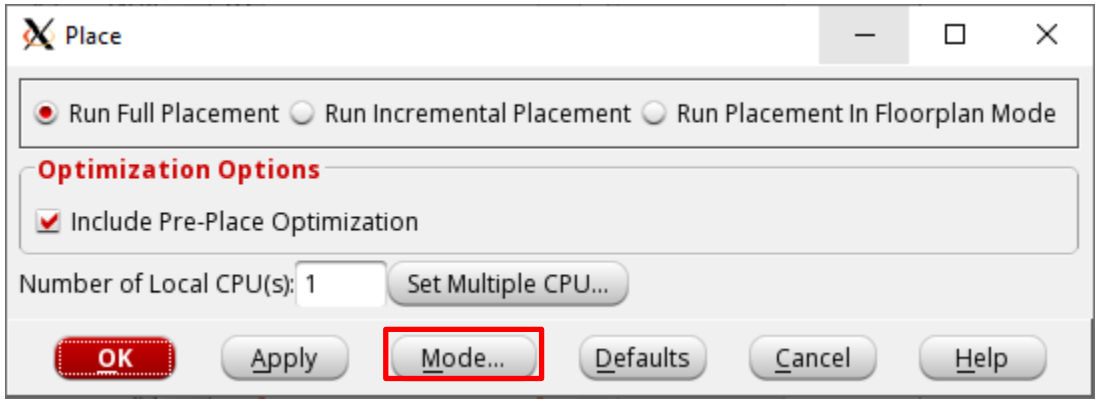
2. P/G Network Design

- As you see, the P/G stripes are alternating between VDD and VSS. See the vias.



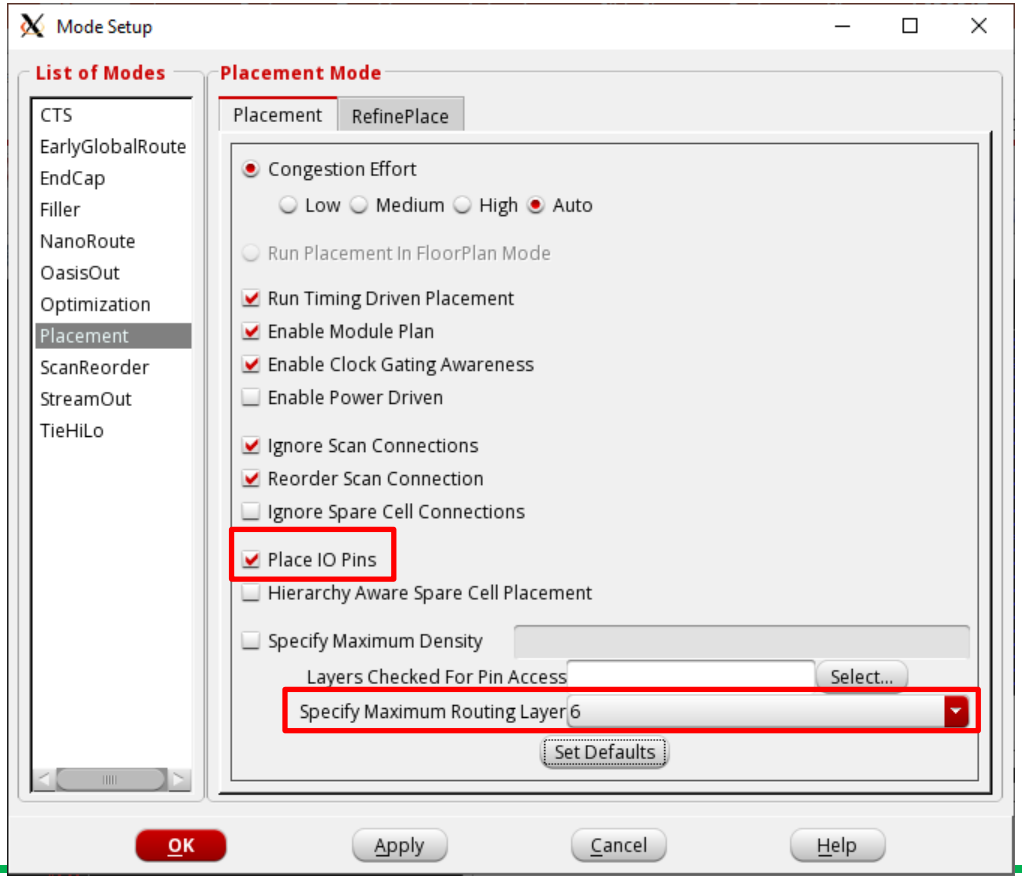
3. Placement

- Let’s place the instances (cells).
- In the main window, click “Place” → “Place Standard Cell”.
- In the “Place” window, click “Mode”.



3. Placement

- Turn on “Place IO Pins”. Set the “Specify Maximum Routing Layer” to 6. We will use only six metal layers. Click OK. In the “Place” window, click OK.

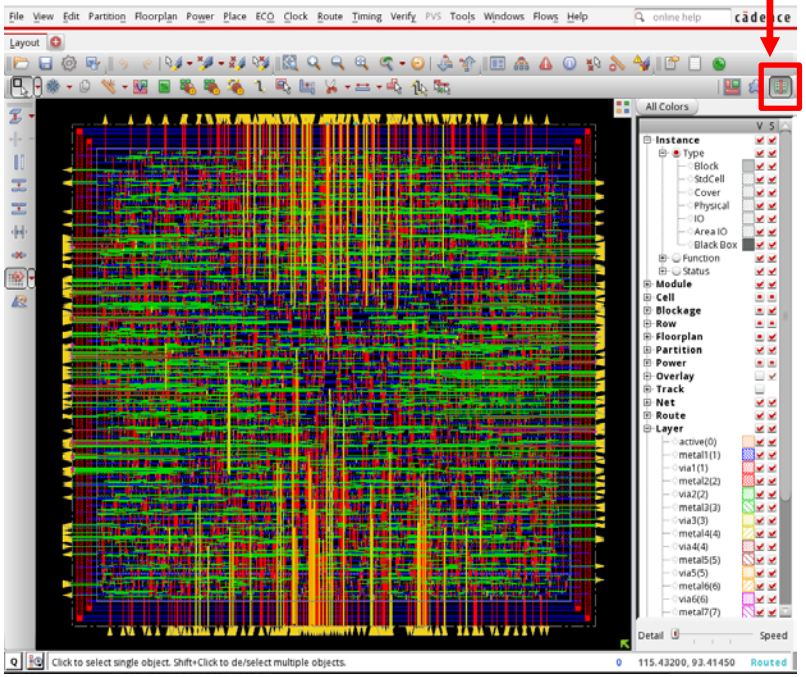


3. Placement

- It shows placement and trialRoute results. trialRoute is just a quick routing for an estimation of some design metrics.
- See the terminal. It shows some more information.
  - Total wire length: 32,540.21um

Click this button
if you don't see the layout.

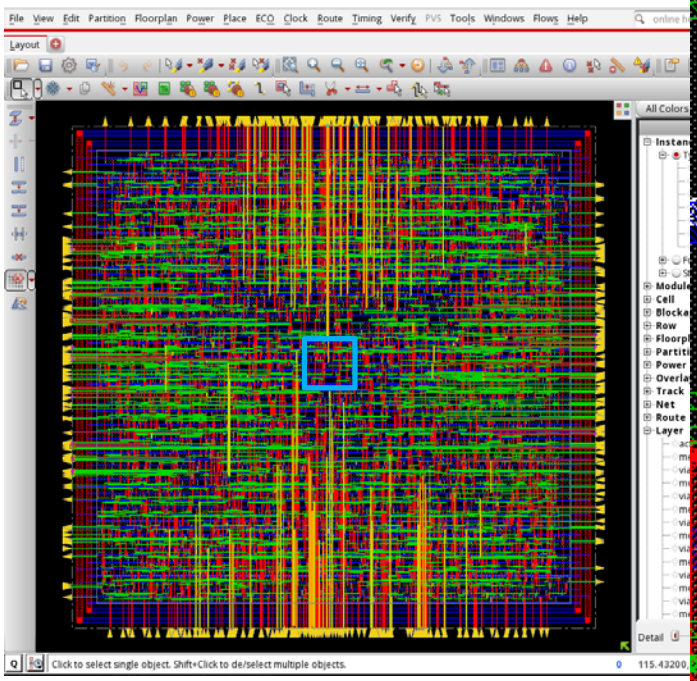
```
-------------------------------------------------------------
Layer1(metal1)(F) length: 0.000000e+00um, number of vias: 8752
Layer2(metal2)(V) length: 1.563008e+04um, number of vias: 12901
Layer3(metal3)(H) length: 1.402659e+04um, number of vias: 355
Layer4(metal4)(V) length: 1.713939e+03um, number of vias: 108
Layer5(metal5)(H) length: 8.258645e+02um, number of vias: 46
Layer6(metal6)(V) length: 3.437350e+02um, number of vias: 19
Total length: 3.254021e+04um, number of vias: 22181
-------------------------------------------------------------
```

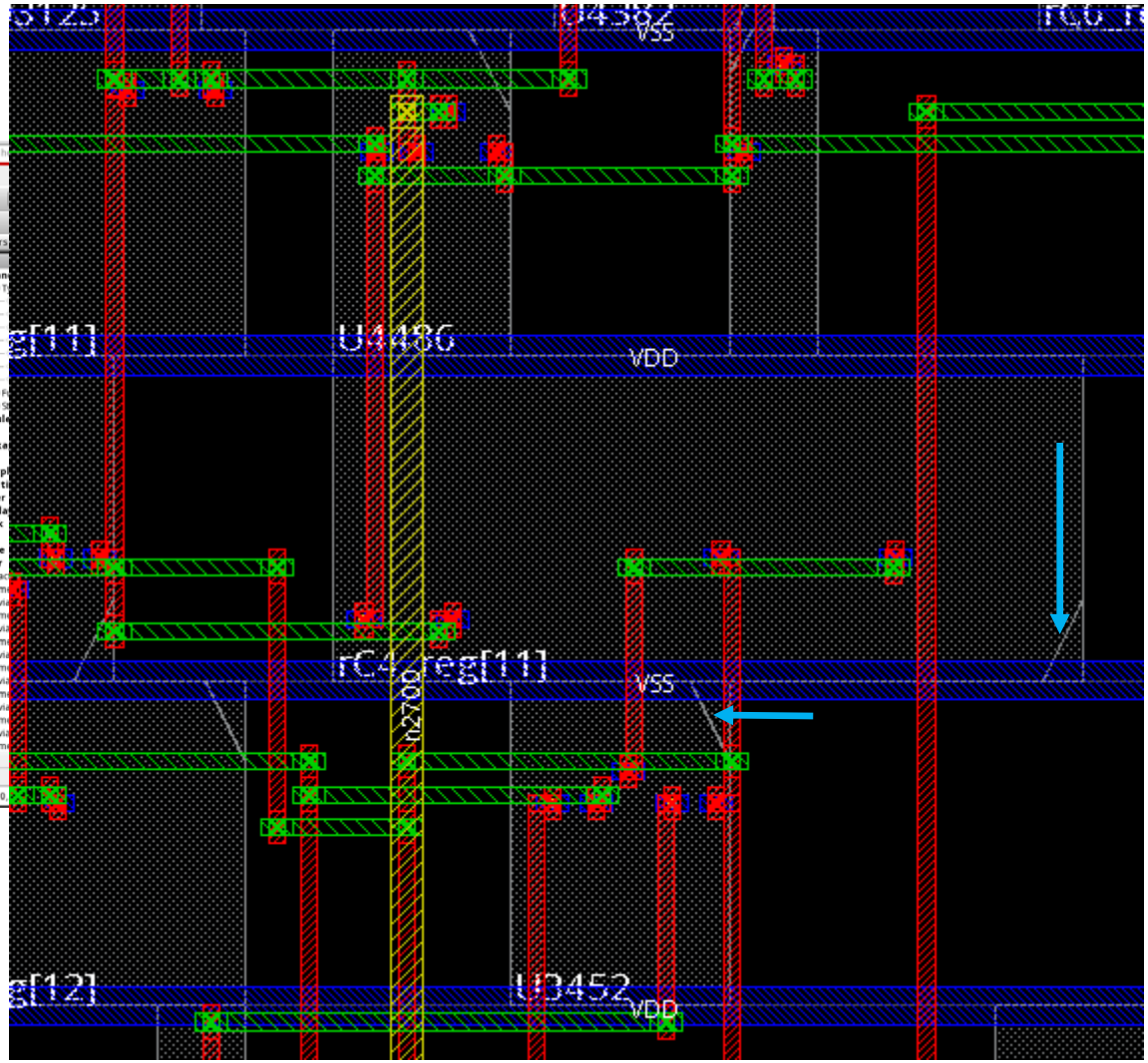


- Save it.
  - saveDesign test_03_pl.enc

3. Placement

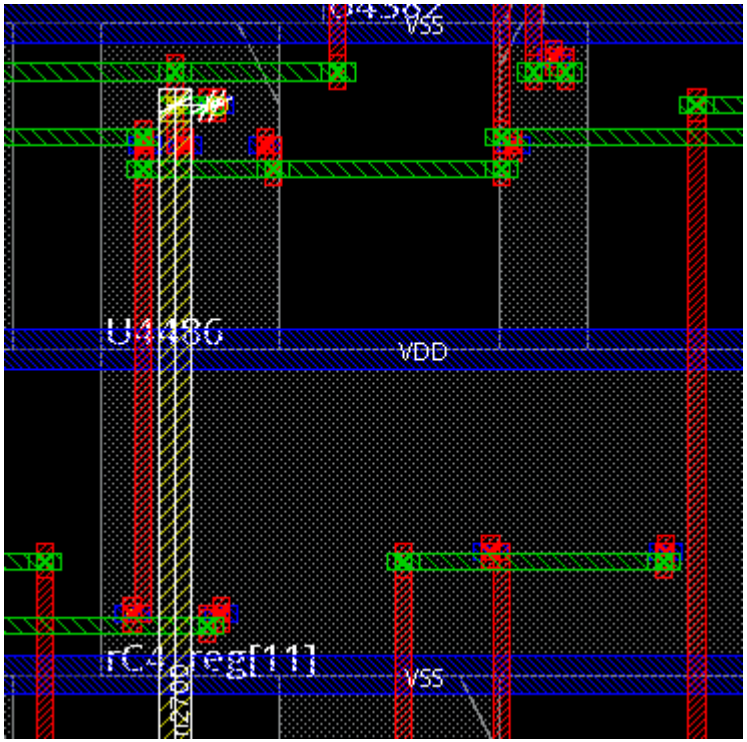
- Let's zoom in.

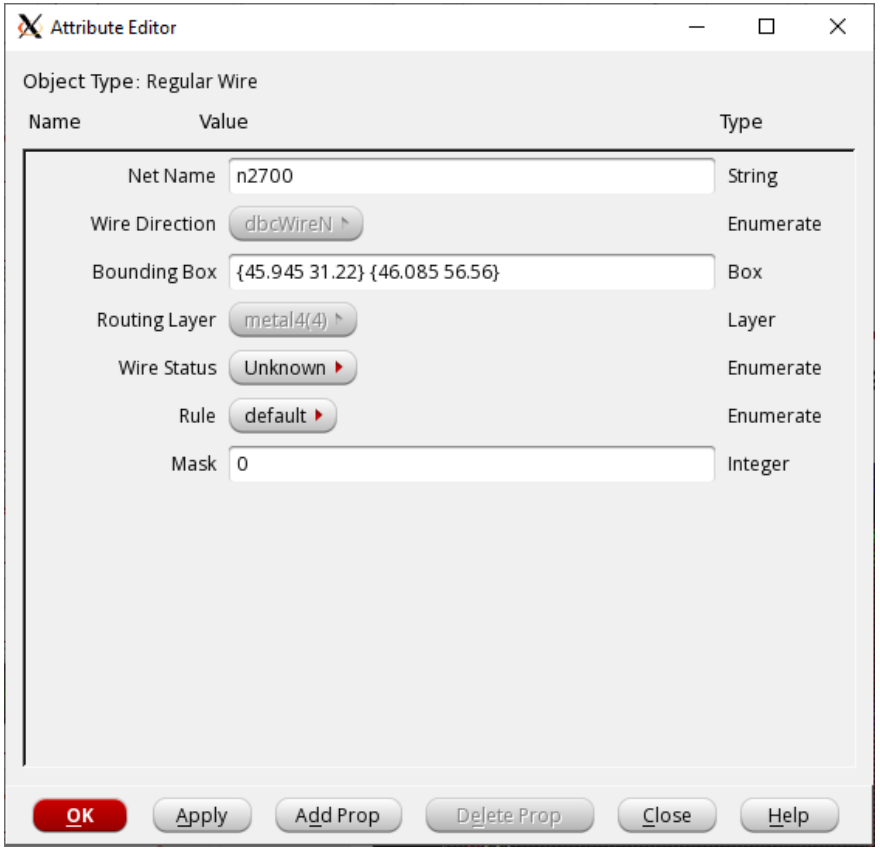




3. Placement

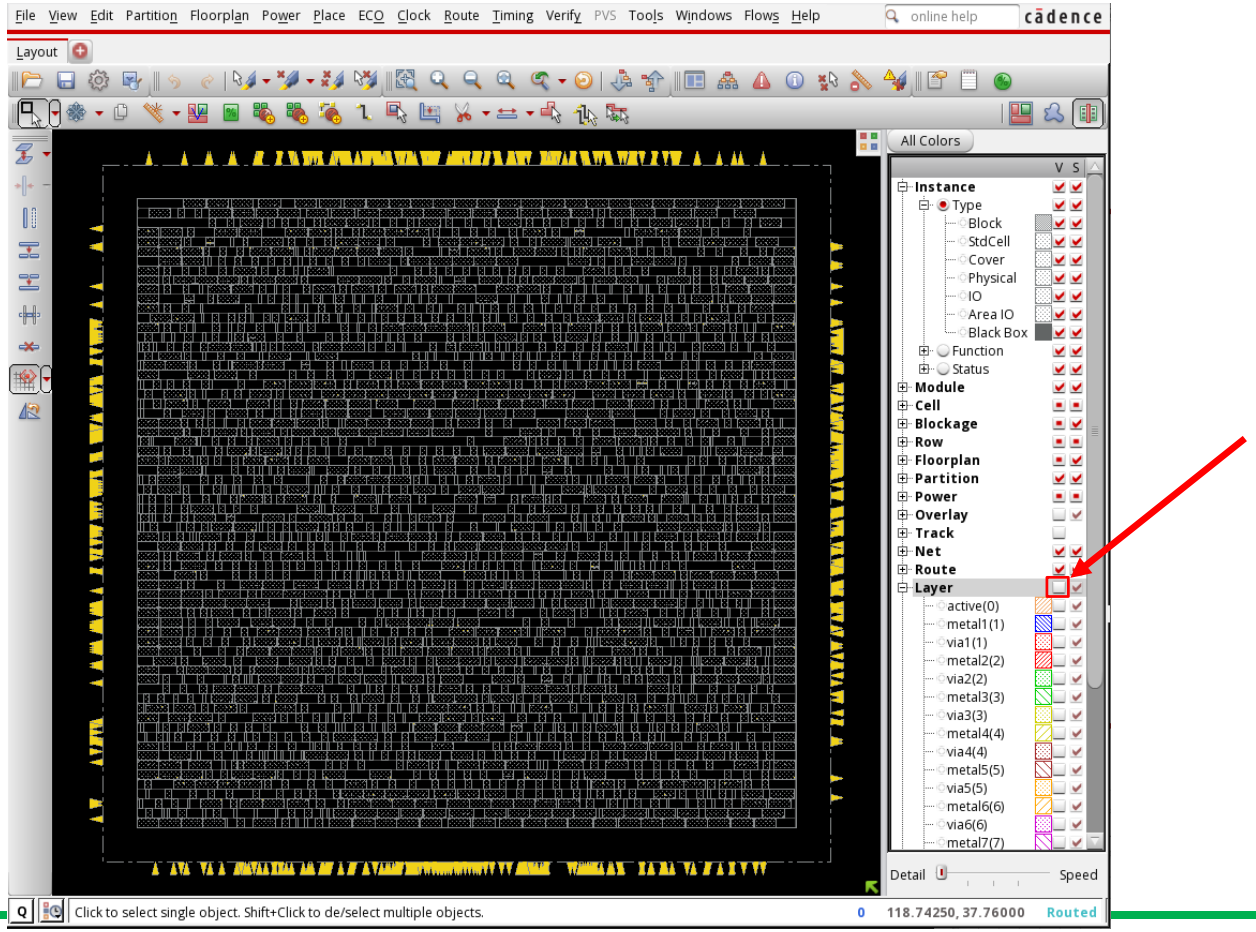
- Click a wire and press ‘q’. You will see a property window.





Visibility

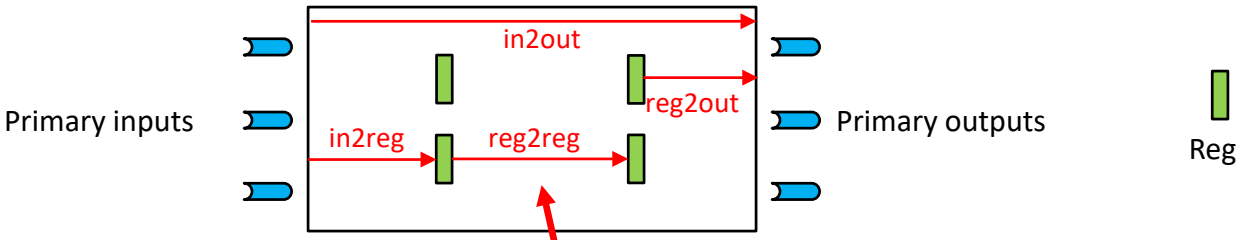
- Let’s see the placement result only.
- Turn off the following check-box to turn off the visibility of the wires.



Static Timing Analysis

- Run the following command to turn off SI-awareness.
  - innovus #> setDelayCalMode –siAware false
- Then, run the following command to analyze setup time.
  - innovus #> timeDesign –preCTS
  - preCTS means “before Clock-Tree-Synthesis (CTS)”. A clock tree is designed after placement.
- It will show the following summary:

Static Timing Analysis



Setup time analysis

Negative WNS

Design Rule Violations

Layout density

```
---------------------------------------------------------------
          timeDesign Summary
---------------------------------------------------------------

Setup views included:
 VView1

+--------------------+---------+---------+---------+
|     Setup mode     |   all   | reg2reg | default |
+--------------------+---------+---------+---------+
|           WNS (ns):| -2.965  | -1.791  | -2.965  |
|           TNS (ns):| -1252.1 |-528.427 |-723.700 |
|    Violating Paths:|   635   |   381   |   254   |
|          All Paths:|   768   |   512   |   256   |
+--------------------+---------+---------+---------+

+----------------+-------------------------------+------------------+
|                |              Real             |       Total      |
|      DRVs      +------------------+------------+------------------|
|                |  Nr nets(terms)  | Worst Vio  |  Nr nets(terms)  |
+----------------+------------------+------------+------------------+
|   max_cap      |     10 (10)      |   -0.244   |     10 (10)      |
|   max_tran     |    13 (1164)     |   -1.104   |    13 (1164)     |
|   max_fanout   |      0 (0)       |     0      |      0 (0)       |
|   max_length   |      0 (0)       |     0      |      0 (0)       |
+----------------+------------------+------------+------------------+

Density: 57.409%
Routing Overflow: 0.00% H and 0.00% V
```

Static Timing Analysis

- Run the following command to check the longest path.
  - innovus #> report_timing
  - The clock frequency is 1GHz.

```
Path 1: VIOLATED Setup Check with Pin rC2_reg[3]/CK
Endpoint:   rC2_reg[3]/D (v) checked with  leading edge of 'myCLK'
Beginpoint: mX2[0]       (v) triggered by  leading edge of '@'
Path Groups: {myCLK}
Analysis View: VView1
Other End Arrival Time          0.000
- Setup                         0.090   <-- F/F setup time (90ps)
+ Phase Shift                   1.000   <-- Clock period
= Required Time                 0.910   <-- RT
- Arrival Time                  3.875   <-- AT
= Slack Time                   -2.965   <-- Slack (=RT – AT)
     Clock Rise Edge                     0.000
     + Input Delay                       0.000
     = Beginpoint Arrival Time           0.000
     +-------------------------------------------------------------------+
     |  Instance  |     Arc     |   Cell    | Delay | Arrival | Required |
     |            |             |           |       |  Time   |   Time   |
```

4. Pre-CTS Optimization

- Now, since the design violates the timing constraints, let's optimize it. (Notice that we can still try to optimize it to reduce power even if it satisfies the timing constraints.)

- Run the following command to optimize the design before CTS.
  - innovus #> optDesign –preCTS

- (This will take some time, up to several minutes depending on the machine you are working with).

- After Pre-CTS optimization is done, you will see the following result:

4. Pre-CTS Optimization

- Pre-CTS optimization

Positive WNS!!!

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

Setup views included:
 VView1

+--------------------+---------+---------+---------+
|     Setup mode     |   all   | reg2reg | default |
+--------------------+---------+---------+---------+
|           WNS (ns):|  0.001  |  0.003  |  0.001  |
|           TNS (ns):|  0.000  |  0.000  |  0.000  |
|    Violating Paths:|    0    |    0    |    0    |
|          All Paths:|   768   |   512   |   256   |
+--------------------+---------+---------+---------+

+----------------+-------------------------------+------------------+
|                |              Real             |       Total      |
|    DRVs        +------------------+------------+------------------|
|                |  Nr nets(terms)  | Worst Vio  |  Nr nets(terms)  |
+----------------+------------------+------------+------------------+
|   max_cap      |      0 (0)       |   0.000    |      0 (0)       |
|   max_tran     |      0 (0)       |   0.000    |      0 (0)       |
|   max_fanout   |      0 (0)       |     0      |      0 (0)       |
|   max_length   |      0 (0)       |     0      |      0 (0)       |
+----------------+------------------+------------+------------------+

Density: 61.022%
Routing Overflow: 0.00% H and 0.00% V
```

The density increased from 57% to 61%.

4. Pre-CTS Optimization

- saveDesign test_04_prectsopt.enc

5. Clock Tree Synthesis (CTS)

- Run the following command to run CTS.
  - innovus #> create_ccopt_clock_tree_spec
  - innovus #> get_ccopt_clock_trees *
    - myCLK (You will see this.)
  - innovus #> set_ccopt_property target_max_trans 0.05
    - Max. transition time at a clock pin is 50ps.
  - innovus #> set_ccopt_property target_skew 0.02
    - Clock skew is 20ps.
  - innovus #> ccopt_design

5. Clock Tree Synthesis (CTS)

- saveDesign test_05_cts.enc

Timing Analysis

- Run the following command to check timing.
  - timeDesign –postCTS

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

Setup views included:
 VView1

+--------------------+---------+---------+---------+
|     Setup mode     |   all   | reg2reg | default |
+--------------------+---------+---------+---------+
|           WNS (ns):|  0.001  |  0.003  |  0.001  |
|           TNS (ns):|  0.000  |  0.000  |  0.000  |
|    Violating Paths:|    0    |    0    |    0    |
|          All Paths:|   768   |   512   |   256   |
+--------------------+---------+---------+---------+

+----------------+-----------------------------------+------------------+
|                |              Real                 |       Total      |
|    DRVs        +------------------+----------------+------------------|
|                |  Nr nets(terms)  | Worst Vio      |  Nr nets(terms)  |
+----------------+------------------+----------------+------------------+
|   max_cap      |      0 (0)       |   0.000        |      0 (0)       |
|   max_tran     |      0 (0)       |   0.000        |      0 (0)       |
|   max_fanout   |      0 (0)       |     0          |      0 (0)       |
|   max_length   |      0 (0)       |     0          |      0 (0)       |
+----------------+------------------+----------------+------------------+

Density: 61.022%
Routing Overflow: 0.00% H and 0.00% V
```

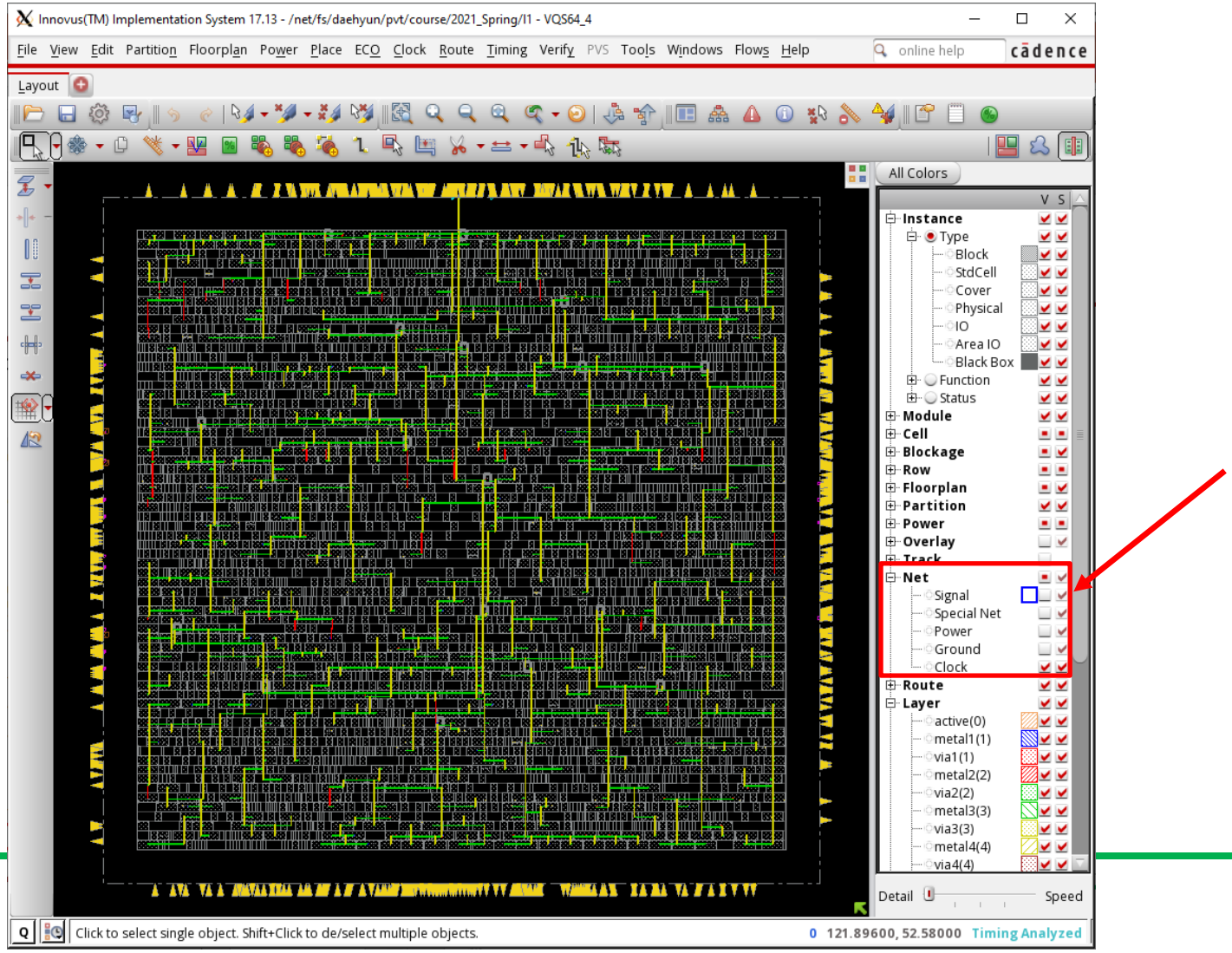
Before CTS

```
------------------------------------------------------------
          timeDesign Summary
------------------------------------------------------------

Setup views included:
 VView1

+--------------------+---------+---------+---------+
|     Setup mode     |   all   | reg2reg | default |
+--------------------+---------+---------+---------+
|           WNS (ns):|  0.000  |  0.000  |  0.011  |
|           TNS (ns):|  0.000  |  0.000  |  0.000  |
|    Violating Paths:|    0    |    0    |    0    |
|          All Paths:|   768   |   512   |   256   |
+--------------------+---------+---------+---------+

+----------------+-----------------------------------+------------------+
|                |              Real                 |       Total      |
|    DRVs        +------------------+----------------+------------------|
|                |  Nr nets(terms)  | Worst Vio      |  Nr nets(terms)  |
+----------------+------------------+----------------+------------------+
|   max_cap      |      0 (0)       |   0.000        |      0 (0)       |
|   max_tran     |      0 (0)       |   0.000        |      0 (0)       |
|   max_fanout   |      0 (0)       |     0          |      0 (0)       |
|   max_length   |      0 (0)       |     0          |      0 (0)       |
+----------------+------------------+----------------+------------------+

Density: 61.239%
Routing Overflow: 0.00% H and 0.00% V
```

After CTS

5. Clock Tree Synthesis (CTS)

- How to show the clock tree



6. Post-CTS Optimization

- Although we already satisfied the timing without any further optimization after CTS, we will run post-CTS optimization.
  - innovus #> optDesign –postCTS

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

Setup views included:
 VView1

+--------------------+---------+---------+---------+
|     Setup mode     |   all   | reg2reg | default |
+--------------------+---------+---------+---------+
|           WNS (ns):|  0.000  |  0.000  |  0.010  |
|           TNS (ns):|  0.000  |  0.000  |  0.000  |
|    Violating Paths:|    0    |    0    |    0    |
|          All Paths:|   768   |   512   |   256   |
+--------------------+---------+---------+---------+

+----------------+--------------------------------+------------------+
|                |              Real              |       Total      |
|    DRVs        +------------------+-------------+------------------|
|                |  Nr nets(terms)  | Worst Vio   |  Nr nets(terms)  |
+----------------+------------------+-------------+------------------+
|   max_cap      |      0 (0)       |   0.000     |      0 (0)       |
|   max_tran     |      0 (0)       |   0.000     |      0 (0)       |
|   max_fanout   |      0 (0)       |     0       |      0 (0)       |
|   max_length   |      0 (0)       |     0       |      0 (0)       |
+----------------+------------------+-------------+------------------+

Density: 61.239%
Routing Overflow: 0.00% H and 0.00% V
```

6. Post-CTS Optimization

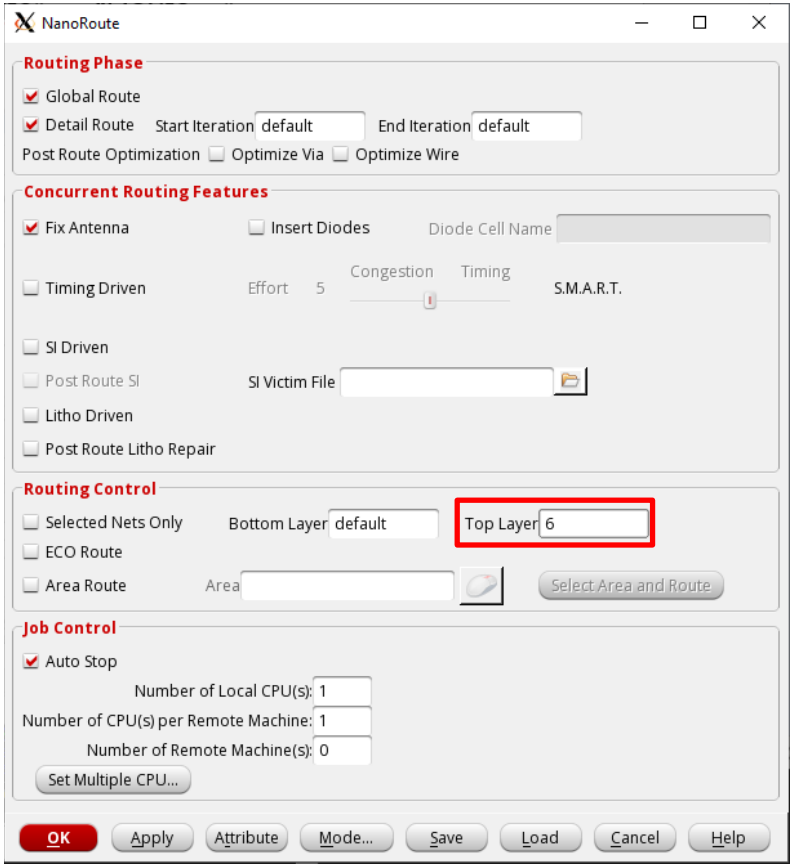
- saveDesign test_06_postctsopt.enc

- So far, we have done
  - Placement
  - CTS

- Now we will route the nets.

7. Routing

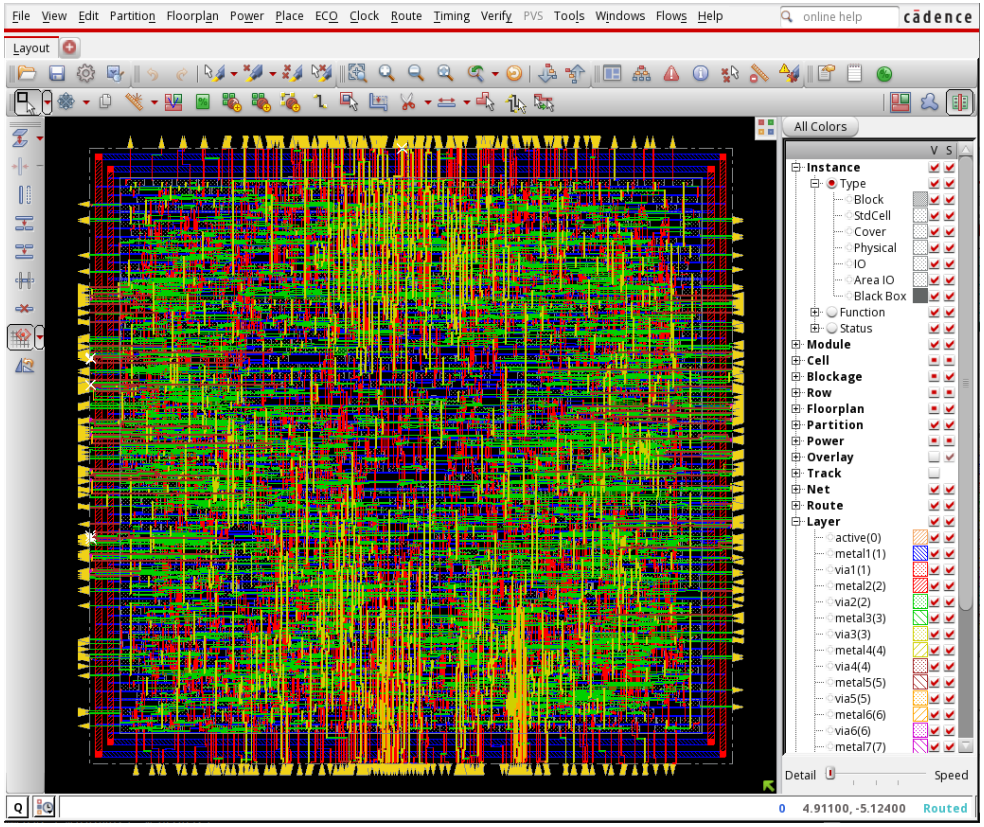
- Click “Route” → “NanoRoute” → “Route…”.
- Make sure that the top layer is “6”. If not, set it to 6.
- Click OK.



7. Routing

- Routing result.
- See the log.
  - WL: 35,172um

```
#Post Route wire spread is done.
#Total number of nets with non-default rule or having
#Total wire length = 35172 um.
#Total half perimeter of net bounding box = 31277 um.
#Total wire length on LAYER metal1 = 1730 um.
#Total wire length on LAYER metal2 = 14515 um.
#Total wire length on LAYER metal3 = 13908 um.
#Total wire length on LAYER metal4 = 3849 um.
#Total wire length on LAYER metal5 = 998 um.
#Total wire length on LAYER metal6 = 173 um.
#Total wire length on LAYER metal7 = 0 um.
#Total wire length on LAYER metal8 = 0 um.
#Total wire length on LAYER metal9 = 0 um.
#Total wire length on LAYER metal10 = 0 um.
#Total number of vias = 20085
#Up-Via Summary (total 20085):
```



Timing Analysis

- Run the following command to check timing.
  - timeDesign –postRoute

```
------------------------------------------------------------
          timeDesign Summary
------------------------------------------------------------

Setup views included:
 VView1

+--------------------+---------+---------+---------+
|     Setup mode     |   all   | reg2reg | default |
+--------------------+---------+---------+---------+
|           WNS (ns):|  0.014  |  0.019  |  0.014  |
|           TNS (ns):|  0.000  |  0.000  |  0.000  |
|    Violating Paths:|    0    |    0    |    0    |
|          All Paths:|   768   |   512   |   256   |
+--------------------+---------+---------+---------+

+----------------+-------------------------------+------------------+
|                |              Real             |       Total      |
|    DRVs        +------------------+------------+------------------|
|                |  Nr nets(terms)  | Worst Vio  |  Nr nets(terms)  |
+----------------+------------------+------------+------------------+
|   max_cap      |      0 (0)       |   0.000    |      0 (0)       |
|   max_tran     |      0 (0)       |   0.000    |      0 (0)       |
|   max_fanout   |      0 (0)       |     0      |      0 (0)       |
|   max_length   |      0 (0)       |     0      |      0 (0)       |
+----------------+------------------+------------+------------------+

Density: 61.239%
```

8. Post-Route Optimization

- Although we've already satisfied the timing without any further optimization after routing, we will run post-routing optimization.
- Well, let's get PPA before that.
  - innovus #> report_area

```
innovus 17> report_area
Depth  Name       #Inst  Area (um^2)
-------------------------------------
0      VQS64_4    2670   5483.058
```

  - innovus #> report_power

```
*       Power Units = 1mW
*
*       Time Units = 1e-09 secs
*
*       report_power
*
-------------------------------------------------------------


Total Power
-------------------------------------------------------------
Total Internal Power:        6.34652051            65.4303%
Total Switching Power:       3.23971289            33.4002%
Total Leakage Power:         0.11343796             1.1695%
Total Power:                 9.69967147
-------------------------------------------------------------
```

- Now, let's optimize the design.
  - innovus #> optDesign –postRoute

8. Post-Route Optimization

```
------------------------------------------------------------
     optDesign Final Non-SI Timing Summary
------------------------------------------------------------

Setup views included:
 VView1

+--------------------+---------+---------+---------+
|     Setup mode     |   all   | reg2reg | default |
+--------------------+---------+---------+---------+
|           WNS (ns):|  0.013  |  0.019  |  0.013  |
|           TNS (ns):|  0.000  |  0.000  |  0.000  |
|    Violating Paths:|    0    |    0    |    0    |
|          All Paths:|   768   |   512   |   256   |
+--------------------+---------+---------+---------+

+----------------+-------------------------------+------------------+
|                |             Real              |       Total      |
|    DRVs        +------------------+------------+------------------|
|                |  Nr nets(terms)  | Worst Vio  |  Nr nets(terms)  |
+----------------+------------------+------------+------------------+
|   max_cap      |      0 (0)       |   0.000    |      0 (0)       |
|   max_tran     |      0 (0)       |   0.000    |      0 (0)       |
|   max_fanout   |      0 (0)       |     0      |      0 (0)       |
|   max_length   |      0 (0)       |     0      |      0 (0)       |
+----------------+------------------+------------+------------------+

Density: 61.239%
```

Before postRoute opt.

```
------------------------------------------------------------
          timeDesign Summary
------------------------------------------------------------

Setup views included:
 VView1

+--------------------+---------+---------+---------+
|     Setup mode     |   all   | reg2reg | default |
+--------------------+---------+---------+---------+
|           WNS (ns):|  0.014  |  0.019  |  0.014  |
|           TNS (ns):|  0.000  |  0.000  |  0.000  |
|    Violating Paths:|    0    |    0    |    0    |
|          All Paths:|   768   |   512   |   256   |
+--------------------+---------+---------+---------+

+----------------+-------------------------------+------------------+
|                |             Real              |       Total      |
|    DRVs        +------------------+------------+------------------|
|                |  Nr nets(terms)  | Worst Vio  |  Nr nets(terms)  |
+----------------+------------------+------------+------------------+
|   max_cap      |      0 (0)       |   0.000    |      0 (0)       |
|   max_tran     |      0 (0)       |   0.000    |      0 (0)       |
|   max_fanout   |      0 (0)       |     0      |      0 (0)       |
|   max_length   |      0 (0)       |     0      |      0 (0)       |
+----------------+------------------+------------+------------------+

Density: 61.239%
```

After postRoute opt.

8. Post-Route Optimization

- Area and power (before)

```
innovus 17> report_area
Depth  Name       #Inst  Area (um^2)
--------------------------------------
0      VQS64_4    2670   5483.058
```

```
*       Power Units = 1mW
*
*       Time Units = 1e-09 secs
*
*       report_power
*
-------------------------------------------------------------


Total Power
-------------------------------------------------------------
Total Internal Power:        6.34652051            65.4303%
Total Switching Power:       3.23971289            33.4002%
Total Leakage Power:         0.11343796             1.1695%
Total Power:                 9.69967147
-------------------------------------------------------------
```

- Area and power (after)

```
innovus 17> report_area
Depth  Name       #Inst  Area (um^2)
--------------------------------------
0      VQS64_4    2670   5483.058
```

```
Total Power
-------------------------------------------------------------
Total Internal Power:        6.34654717            65.4311%
Total Switching Power:       3.23959843            33.3994%
Total Leakage Power:         0.11343796             1.1695%
Total Power:                 9.69958366
-------------------------------------------------------------
```

8. Post-Route Optimization

- saveDesign test_08_postrouteopt.enc

- Done

9. Verification

9. Verification

- In the main menu, Verify → Very Geometry. Click OK.

```
innovus 28>  *** Starting Verify Geometry (MEM: 1346.7) ***

**WARN: (IMPVFG-257):   verifyGeometry command is replaced by verify_drc command. It still works in this release but will be removed in
future release. Please update your script to use the new command.
  VERIFY GEOMETRY ..... Starting Verification
  VERIFY GEOMETRY ..... Initializing
  VERIFY GEOMETRY ..... Deleting Existing Violations
  VERIFY GEOMETRY ..... Creating Sub-Areas
                  ..... bin size: 2160
  VERIFY GEOMETRY ..... SubArea : 1 of 1
**WARN: (IMPVFG-47):    This warning message means the PG pin of macro/macros is not connected to relevant PG net in the design. If we q
uery the particular PG pin 'net:NULL' will be displayed in the Innovus GUI.

  VERIFY GEOMETRY ..... Cells          :  0 Viols.
  VERIFY GEOMETRY ..... SameNet        :  0 Viols.
  VERIFY GEOMETRY ..... Wiring         :  6 Viols.
  VERIFY GEOMETRY ..... Antenna        :  0 Viols.
  VERIFY GEOMETRY ..... Sub-Area : 1 complete 6 Viols. 0 Wrngs.
VG: elapsed time: 1.00
Begin Summary ...
  Cells       : 0
  SameNet     : 0
  Wiring      : 3
  Antenna     : 0
  Short       : 3
  Overlap     : 0
End Summary
```

9. Verification

- In the main menu, Verify → Very Connectivity. Click OK.

```
innovus 28> innovus 28> VERIFY_CONNECTIVITY use new engine.

******** Start: VERIFY CONNECTIVITY ********
Start Time: Sun Mar 29 15:08:03 2020

Design Name: VQS64_4
Database Units: 2000
Design Boundary: (0.0000, 0.0000) (107.1000, 102.4800)
Error Limit = 1000; Warning Limit = 50
Check all nets

Begin Summary
  Found no problems or warnings.
End Summary

End Time: Sun Mar 29 15:08:03 2020
Time Elapsed: 0:00:00.0

******** End: VERIFY CONNECTIVITY ********
  Verification Complete : 0 Viols.  0 Wrngs.
```