CSIE DIC Lab 2023

Lab12. Cell-Based APR Training

Data Preparation

1. Extract LAB data from TA's directory:

   % tar xzvf ~dicta/lab12.tar.gz

2. The extracted LAB directory (**lab12**) contains:

   a. Innovus/ :APR working directory

   b. Netlist/ :Synthesized netlist

Command Line Mode

The following steps will guide you on how to run **Cadence® Innovus™ Implementation System (INNOVUS) 21.17** Timing/SI closure flow using command line mode. <u>The requirement of this lab is to complete the APR flow using both the command line mode and the GUI mode.</u>

Add I/O Pad to CORE module

1. Change to the directory: **Netlist**
2. Add I/O PAD module to CORE module:

   Open **CHIP_syn.v** and you can see the I/O PAD module is added at the beginning of the file. The top module name is **CHIP**.

   What is the input pad module name? ____________

   What is the output pad module name? ____________

   How many 1.8V core power pads are used? ____________

3. Open timing constraint file: **CHIP.sdc**
4. Change to the directory: **Innovus**, and open IO Pad Location file: **CHIP.io**

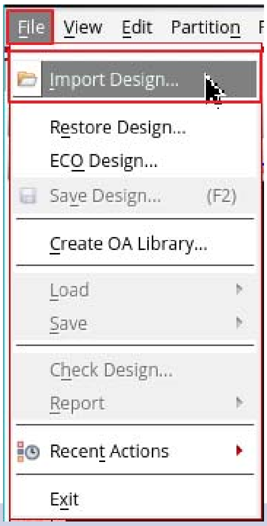
   Please see **Appendix A** for detail.

Reading Cell Library information and netlist for APR

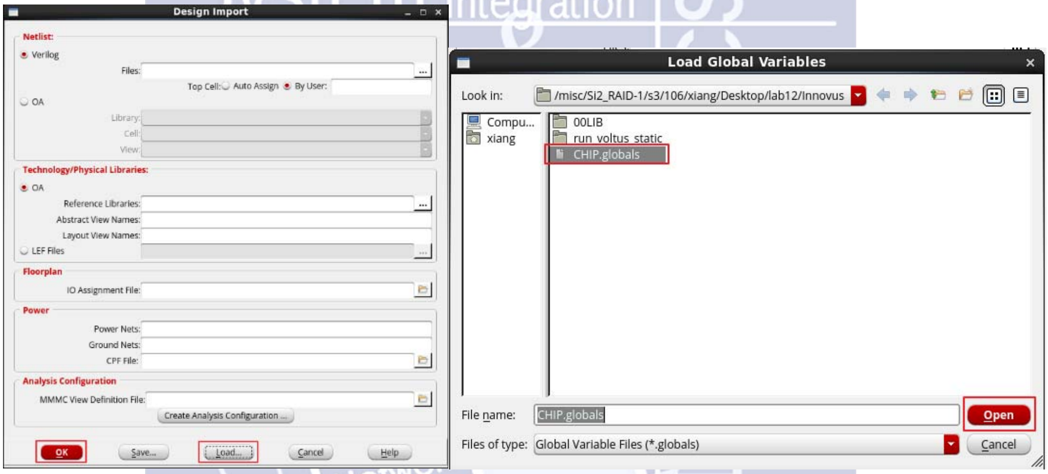
5. Start **INNOVUS** in **Innovus** directory:

   % innovus

6. In INNOVUS menu, open ***File->Import Design***



7. Load **CHIP.globals**, then click OK button



Look in: /misc/Si2_RAID-1/s3/106/xiang/Desktop/lab12/Innovus

File name: CHIP.globals

Files of type: Global Variable Files (*.globals)

8. In INNOVUS command prompt, execute the script **s0.cmd**:

```
innovus > source    s0.cmd
```

How many standard cell instances (stdCell insts) are in **CHIP**? ____________

Define Chip Floorplan

9. In INNOVUS command prompt execute script **s1.cmd**:

```
innovus > source    s1.cmd
```

10. In this design, Core Size: 500μm×500μm, Core to I/O Distance: 150μm.

Power Planning (Add Power Rings and Stripes)

11. In INNOVUS command prompt execute script **s2.cmd**:

```
innovus> source    s2.cmd
```

12. Power ring width: 9μm×7 (M5/M6)

13. Stripe width: 9µm (M5), and 9µm (M6).
14. Connect Pad core power pins.

Place Standard Cells and Add Tie-Hi/Tie-Lo Cells

15. In INNOVUS command prompt execute script **s3.cmd**:

```
innovus> source    s3.cmd
```

What is the placement density? ______

In-Placement Optimization and Connect Standard Cell Power Pins - Before Clock Tree Synthesis

16. In INNOVUS command prompt, execute the following timing analysis command

```
innovus> timeDesign    -preCTS
```

17. INNOVUS performs a trial route and estimates the interconnection RC effects.

What are the worst negative slack (WNS) and the total negative slack (TNS)?
WNS =__________, TNS= __________
How many real design rule violations (DRVs) are there?
Real DRV (max_cap, max_tran, max_fanout)=(_______, ________, _______)

18. Please see timing reports in **timingReports** directory, all slack values should be positive value in this file. Moreover, for detail timing path report, see **CHIP_preCTS_all.tarpt.gz**. DRVs report files: ***.cap.gz**, ***.fanout,gz**, and ***.tran.gz**.

```
% gvim    CHIP_preCTS_all.tar.gz    &
```

19. If the timing slack is negative, or there are DRVs, execute the following command

```
innovus> optDesign    -preCTS
```

What are the WNS and TNS after in-placement optimization (IPO)?
WNS =__________, TNS= __________
Real DRV (max_cap, max_tran, max_fanout)=(_______, ________, _______)
(The script for IPO before clock tree synthesis is **s4.cmd**.)

20. In INNOVUS command prompt execute script **s4.cmd**:

```
innovus> source    s4.cmd
```

Then the standard cell's power pins are connected.

Clock Tree Synthesis (CTS)

21. In INNOVUS command prompt execute script **s5.cmd**:

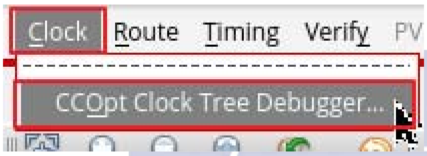
    innovus> source s5.cmd

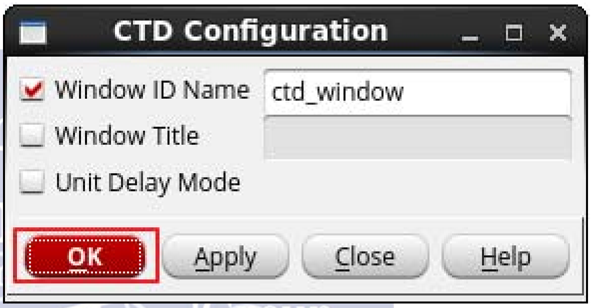
    Then the clock tree specification file (**CHIP.CCOPT.spec**) is automatically generated based on the timing constraints file (**CHIP.sdc**). After CTS, the information for the clock network can be found in Clock Tree Debugger.

22. Use Clock Tree Debugger to display the clock tree of the design

    a. In INNOVUS menu, open ***Clock -> CCOpt Clock Tree Debugger***

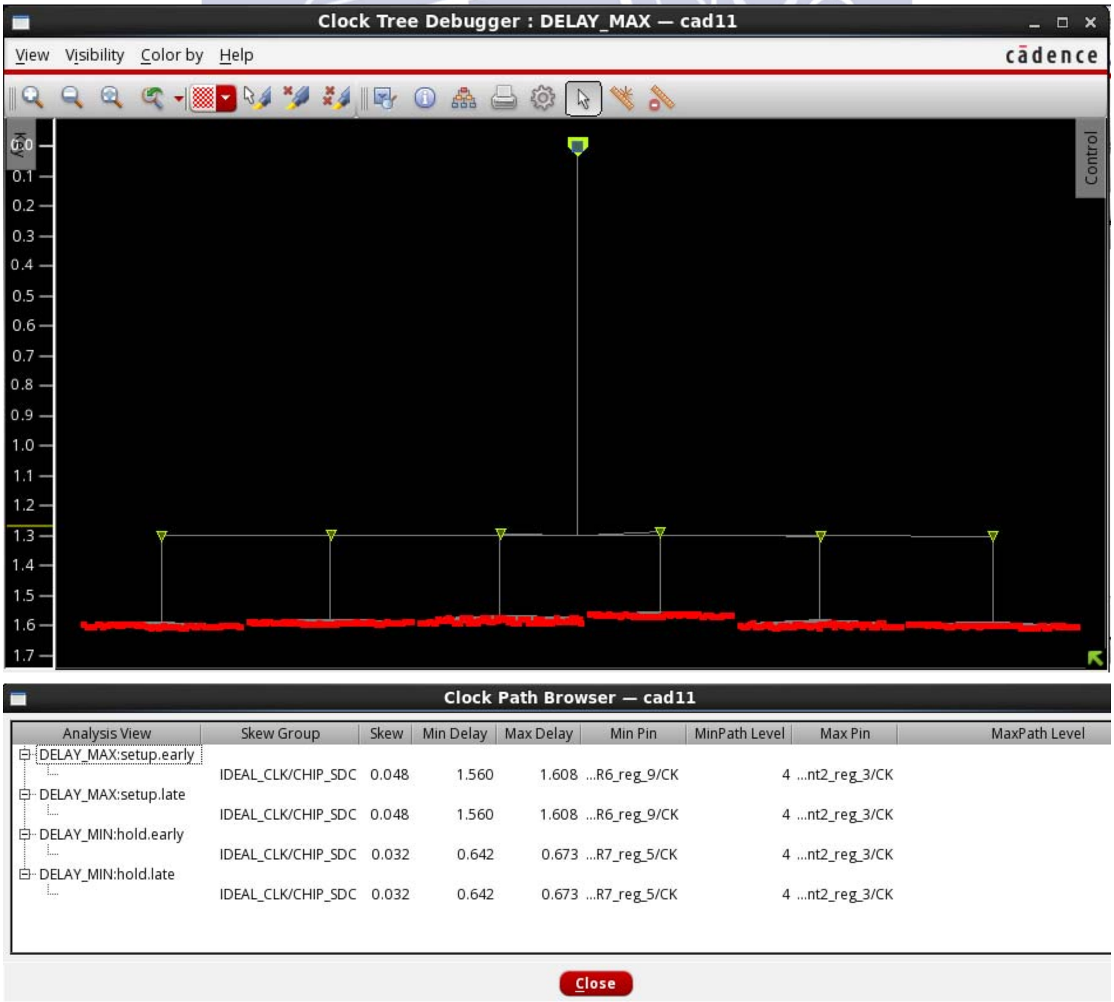
    b. Click **OK** button





☑ Window ID Name ctd_window
□ Window Title
□ Unit Delay Mode

    c. In Clock Tree Debugger menu, open ***View -> Enable clock path browser***



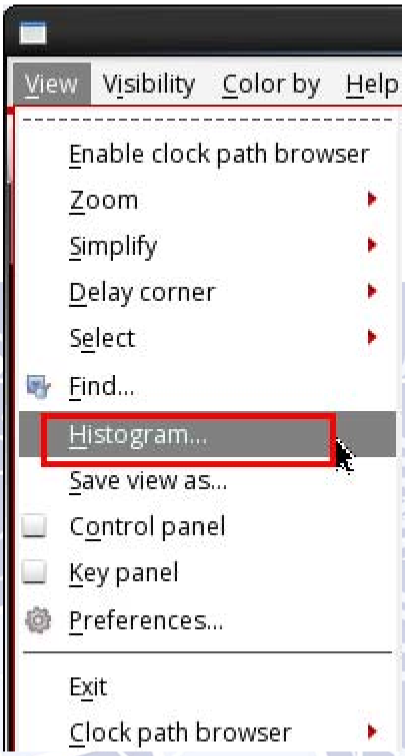
Analysis View	Skew Group	Skew	Min Delay	Max Delay	Min Pin	MinPath Level	Max Pin	MaxPath Level
DELAY_MAX:setup.early	IDEAL_CLK/CHIP_SDC	0.048	1.560	1.608	...R6_reg_9/CK	4	...nt2_reg_3/CK	
DELAY_MAX:setup.late	IDEAL_CLK/CHIP_SDC	0.048	1.560	1.608	...R6_reg_9/CK	4	...nt2_reg_3/CK	
DELAY_MIN:hold.early	IDEAL_CLK/CHIP_SDC	0.032	0.642	0.673	...R7_reg_5/CK	4	...nt2_reg_3/CK	
DELAY_MIN:hold.late	IDEAL_CLK/CHIP_SDC	0.032	0.642	0.673	...R7_reg_5/CK	4	...nt2_reg_3/CK	

The inserted clock buffers and their sink DFFs are shown. The clock skew for each skew group is also displayed.

What is the maximum clock skew of the design after CTS? ____________

23. Display the histogram of clock network delay

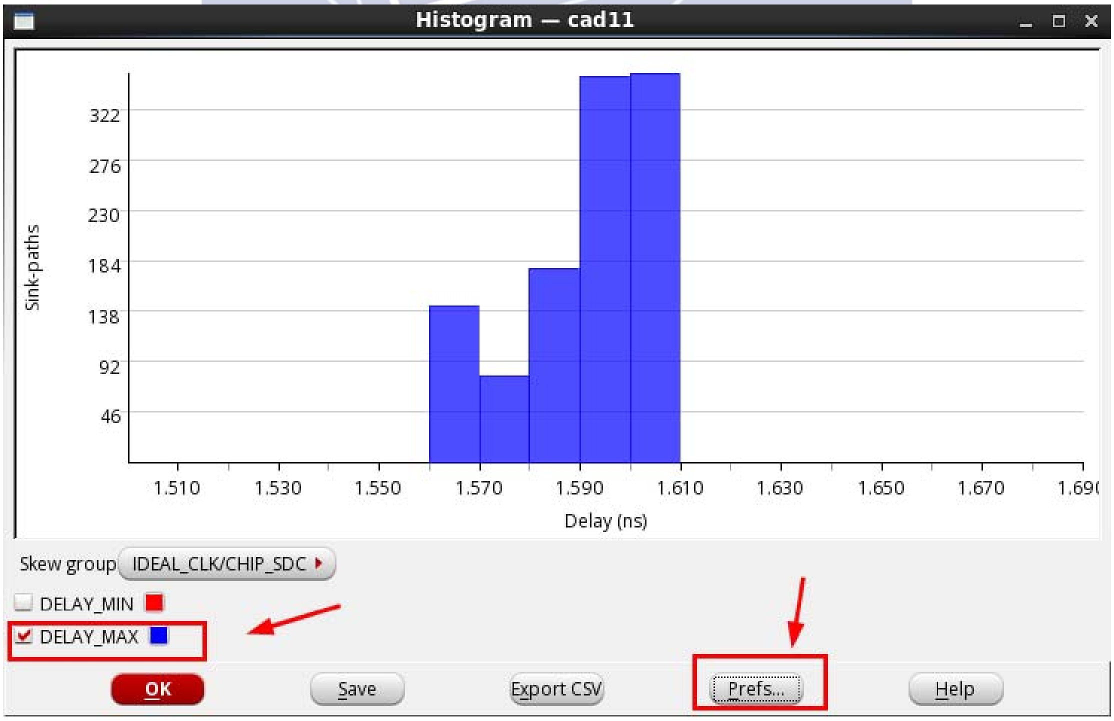
a. In Clock Tree Debugger menu, open ***View -> Histogram***



b. In Histogram window, select the following options

➢ ☑ **DELAY_MAX**

c. In Histogram window, open **Prefs…** (Set Preference)



Skew group IDEAL_CLK/CHIP_SDC

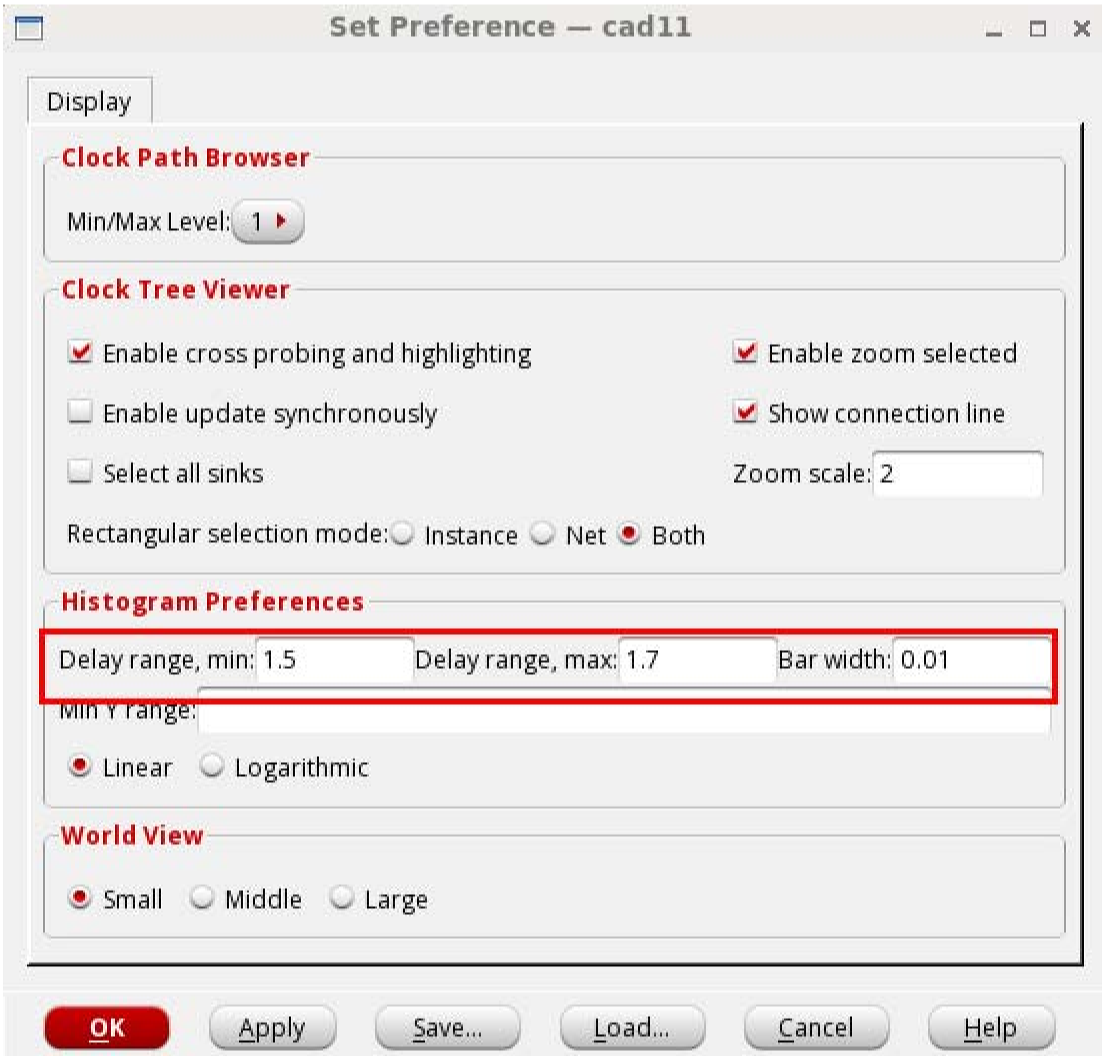
☐ DELAY_MIN 🟥

☑ DELAY_MAX 🟦

d. In Set Preference window, ***Histogram Preferences*** table

- Delay range, min: **1.5**
- Delay range, max: **1.7**
- Bar width: **0.01**

e. Click **OK** button



Set Preference — cad11

Display

Clock Path Browser

Min/Max Level: 1

Clock Tree Viewer

☑ Enable cross probing and highlighting

☑ Enable zoom selected

□ Enable update synchronously

☑ Show connection line

□ Select all sinks

Zoom scale: 2

Rectangular selection mode: ○ Instance ○ Net ● Both

Histogram Preferences

Delay range, min: 1.5 Delay range, max: 1.7 Bar width: 0.01

Min Y range:

● Linear ○ Logarithmic

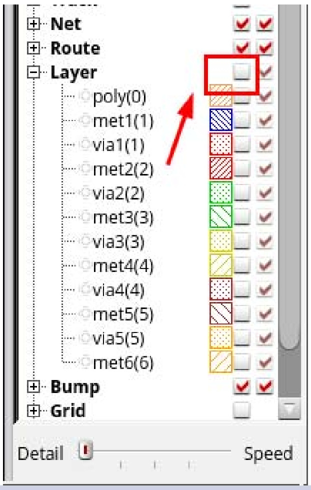
World View

● Small ○ Middle ○ Large

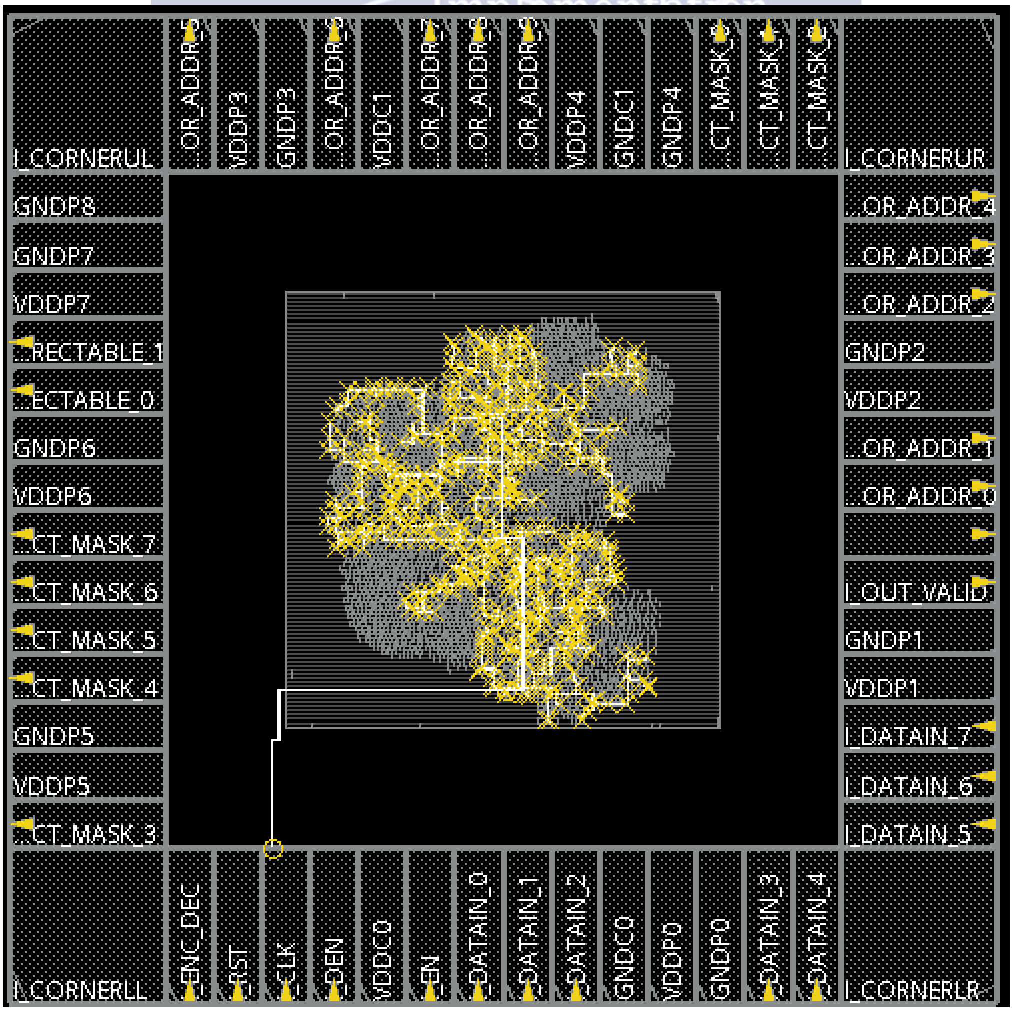
In the histogram, the distribution of the clock network delay is shown, and the maximum and minimum clock network delay can be found.

24. Display the clock tree in INNOVUS GUI window

a. In **Display Layer Control** menu, uncheck all metals and vias layers to see the clock tree more clearly.



b. In Clock Tree Debugger window, use the left mouse button and hold the Ctrl key to select all sinks DFFs of the clock tree. **You should click on the output line of each clock buffers to select the DFFs**. Subsequently, the clock tree will be displayed as follows:



c. In **Display Layer Control** menu, turn on the display of all metals and vias Layers.

In-Place Optimization (IPO)

- After Clock Tree Synthesis

25. In INNOVUS command prompt, execute the following command:

```
innovus> timeDesign    -postCTS
```

26. After clock tree synthesis, the clock skew will affect the timing of the design. INNOVUS performs a trial route and estimates the interconnection RC effects.

    What are the worst negative slack (WNS) and the total negative slack (TNS)?
    WNS =__________, TNS= __________
    How many real design rule violations (DRVs) are there?
    Real DRV (max_cap, max_tran, max_fanout)=(________, _________, ________)

27. After CTS, further timing optimization is performed to meet timing constraints if there has negative timing slack or DRVs in setup time analysis.

```
innovus> optDesign    -postCTS
```

    What are the worst negative slack (WNS) and the total negative slack (TNS)?
    WNS =__________, TNS= __________
    Real DRV (max_cap, max_tran, max_fanout)=(________, _________, ________)

28. After CTS, the clock skew may cause hold time violations. We need to perform hold time analysis.

```
innovus> timeDesign    -postCTS    -hold
```

    What are the worst negative slack (WNS) and the total negative slack (TNS)?
    WNS =__________, TNS= __________

29. If hold time slack is negative, you can fix them with the following command:

```
INNOVUS> optDesign    -postCTS    -hold
```

    What are the worst negative slack (WNS) and the total negative slack (TNS)?
    WNS =__________, TNS= __________

30. Please see timing reports in **timingReports** directory. For detail path report, see **CHIP_postCTS_all.tarpt.gz** (setup time check) and **CHIP_postCTS_all_hold.tarpt.gz** (hold time check). DRV violations report files: ***.cap.gz**, ***.fanout.gz**, and ***.tran.gz**.
    (The script for IPO after clock tree synthesis is **s6.cmd**.)

Add I/O Pad Filler Cells

31. In INNOVUS command prompt execute script **s7.cmd**:

```
innovus> source    s7.cmd
```

I/O PAD Fillers will be added and filled up the gap between I/O PADs. I/O PAD fillers need to be placed before detail routing.

SI-Prevention Detail Route (NanoRoute)

32. In INNOVUS command prompt execute script **s8.cmd**:

```
innovus> source    s8.cmd
```

33. Nanoroute can prevent cross talk effects (SI prevention) and fix antenna rule violations, and it also routes design according to the timing constraints.
34. After detail routing, in INNOVUS command prompt, execute the following command to check the routed design:

```
innovus> verifyConnectivity    -type all    -error 1000    -warning 50
```

If you see any violation, your routed design is not correct. (LVS error)

35. In INNOVUS command prompt, execute following command:

```
innovus> verify_drc
```

If you see any violation, your routed design is not correct. (DRC error)

(The script for NanoRoute is **s8.cmd**.)

In-Place Optimization (IPO)

- After Detailed Route

36. In INNOVUS command prompt, type the following timing analysis mode setting:

```
innovus> setAnalysisMode    -cppr both \
                  -clockGatingCheck true    -timeBorrowing true \
                  -useOutputPinCap true    -sequentialConstProp false \
                  -timingSelfLoopsNoSkew false    -enableMultipleDriveNet true \
                  -clkSrcPath true    -warn true    -usefulSkew true \
                  -analysisType onChipVariation -log true
```

(You can copy-paste these commands from **s9.cmd**)

```
innovus> timeDesign    -postRoute
```

What are the worst negative slack (WNS) and the total negative slack (TNS)?

WNS =__________, TNS= __________

Real DRV (max_cap, max_tran, max_fanout)=(_______, ________, _______)

37. After detail routing with NanoRoute, SI analysis (glitch analysis) is default turned on, and further timing optimization is performed to meet setup timing constraints. If you have negative setup timing slack, use the following command to perform timing optimization:

```
innovus> optDesign    -postRoute
```

What are the worst negative slack (WNS) and the total negative slack (TNS)?
WNS =__________, TNS= __________
Real DRV (max_cap, max_tran, max_fanout)=(_______, ________, _______)

38. The hold time violations are also needed to be verified after detail routing:

```
innovus> timeDesign    -postRoute    -hold
```

What are the worst negative slack (WNS) and the total negative slack (TNS)?
WNS =__________, TNS= __________
Real DRV (max_cap, max_tran, max_fanout)=(_______, ________, _______)

39. You can fix hold time violations if the timing slack is negative:

```
innovus> optDesign    -postRoute    -hold
```

What are the worst negative slack (WNS) and the total negative slack (TNS)?
WNS =__________, TNS= __________
Real DRV (max_cap, max_tran, max_fanout)=(_______, ________, _______)

40. Please see timing reports in **timingReports** directory. For a detail path report, see **CHIP_postRoute_all.tarpt.gz** (setup time check) and **CHIP_postRoute_all _hold.tarpt.gz** (hold time check). DRV violations report files: ***.cap.gz**, ***.fanout.gz**, ***.tran.gz**.
(The script for IPO after the detailed route is **s9.cmd**.)

Add CORE Filler Cells

41. In INNOVUS command prompt, execute script **s10.cmd**:

```
innovus> source    s10.cmd
```

42. Core filler cells are added to reduce well resistance and avoid CMOS latch-up problems.
43. I/O PAD fillers had already inserted before detail routing (NanoRoute), and before stream-out, core fillers are inserted in this step.
44. Core fillers will fill up all empty core placement region and make the design density becomes 100%. Thus it is not possible to have any further IPO after adding core filler cells.

RC Extraction, Delay Calculation, Stream Out, and Write Out Netlist

45. In INNOVUS command prompt, execute script **s11.cmd**:

```
innovus> source   s11.cmd
```

46. The routed design is output in GDS-II format for tape-out (**CHIP.gds**). Also, after 3D Full-Chip RC extraction (QRC), **CHIP.v** and **CHIP.sdf** are generated for post-layout gate-level simulation.
47. **CHIP_LVS.v** includes filler cells and is only for Calibre LVS comparisons.
48. Do not close the INNOVUS window now. You need it to display IR drop analysis results later.

Post-Layout Gate-Level Simulation

49. You need to open another terminal window to run this step.
50. Change to directory **Netlist**
51. Perform post-layout gate-level simulation of CHIP.v

    ```
    % ncverilog   -f   run.f
    ```

    You can also change the cycle time in TEST_CHIP.v to see the maximum clock rate of this design. (cycle time is defined as Tclk)
52. After post-layout gate-level simulation, **CHIP.tcf** is generated in this directory. This file will be used for power rail analysis.

Power Rail Analysis

53. Back to INNOVUS, in INNOVUS command prompt, execute script **s12.cmd:**

```
innovus> source   s12.cmd
```

54. In the first step (**libgen.tcl**) of power rail analysis, we need to generate the Port View database for all standard cells.

    A directory named **./run_voltus_static/fast_allcells.cl/** is generated.

55. In the second step (**power_analysis.tcl**), we need to run PowerMeter instance power analysis to calculate the power consumption of each instance of the design. Then the power database (**./run_voltus_static/power.db**) is generated.

    You can find the detail instance power information in **./run_voltus_static/ CHIP_static_power.rpt**. In the end of this file, what is the total power (including

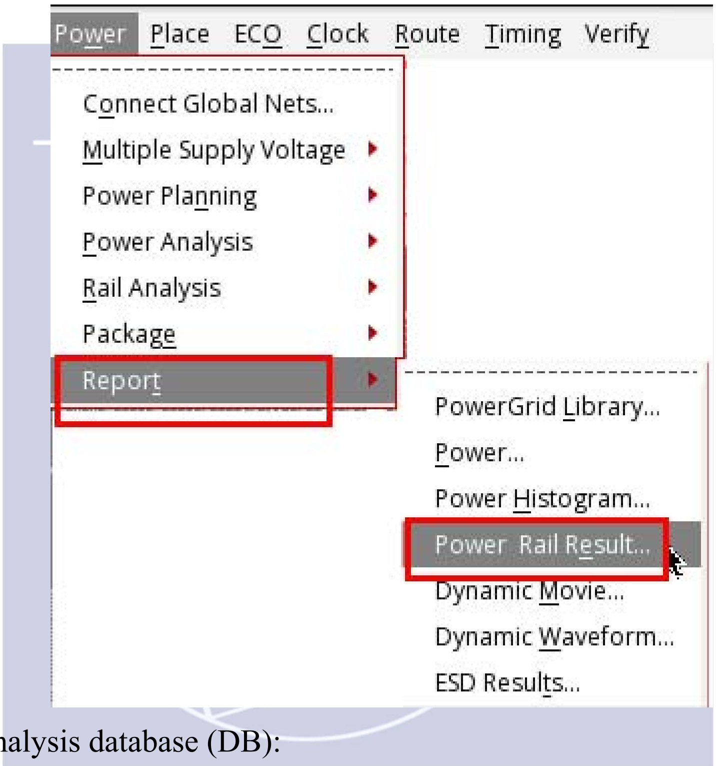
I/O Pads) of this design?

____

56. In the third step (**rail_analysis.tcl**), we need to run Voltus to analyze the IR-drop of the design. A directory named **./run_voltus_static/VDD_125C_avg_1** is generated. It contains the graphics IR-drop analysis results which can be displayed in INNOVUS.

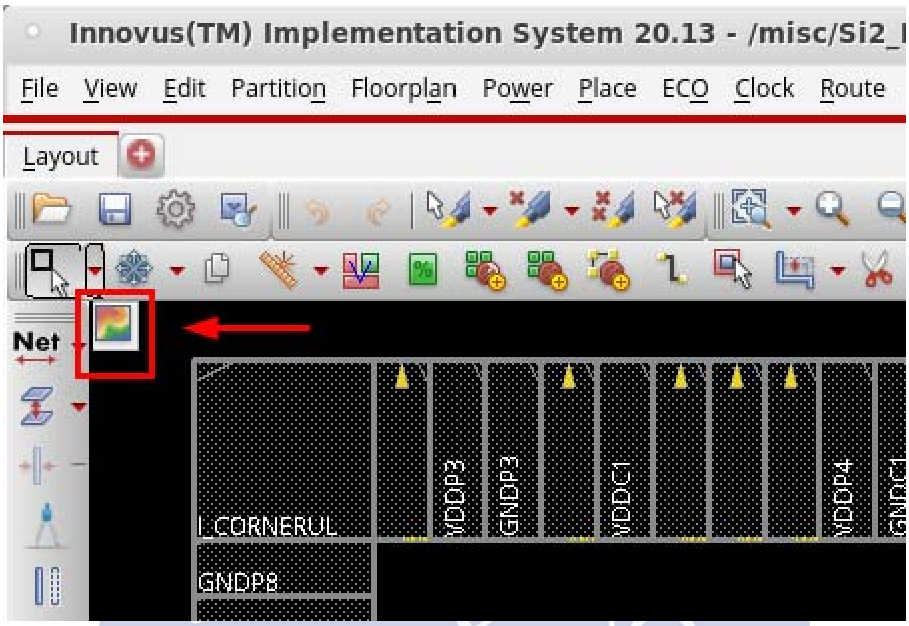
Display Power Rail Analysis

57. In INNOVUS menu, open ***Power -> Report -> Power & Rail Results…***



58. Load rail analysis database (DB):

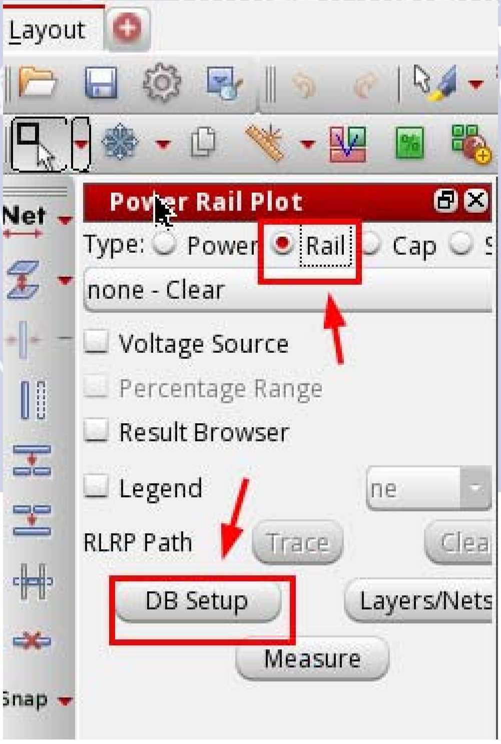
a. In layout window, you will see a hidden menu in the left-top area of the layout, as indicated in the following figure. Move the mouse cursor to this area, and click on this hidden menu to show Power Rail Plot window.



Innovus(TM) Implementation System 20.13 - /misc/Si2_I
File View Edit Partition Floorplan Power Place ECO Clock Route
Layout
Net
I_CORNERUL
GNDP8
VDDP3
GNDP3
VDDC1
VDDP4
GNDC1

b. In Power Rail Plot window, change Type to : ⦿Rail

c. Click **DB Setup** button



Type: ○ Power ● Rail ○ Cap ○ S

☐ Voltage Source

☐ Percentage Range

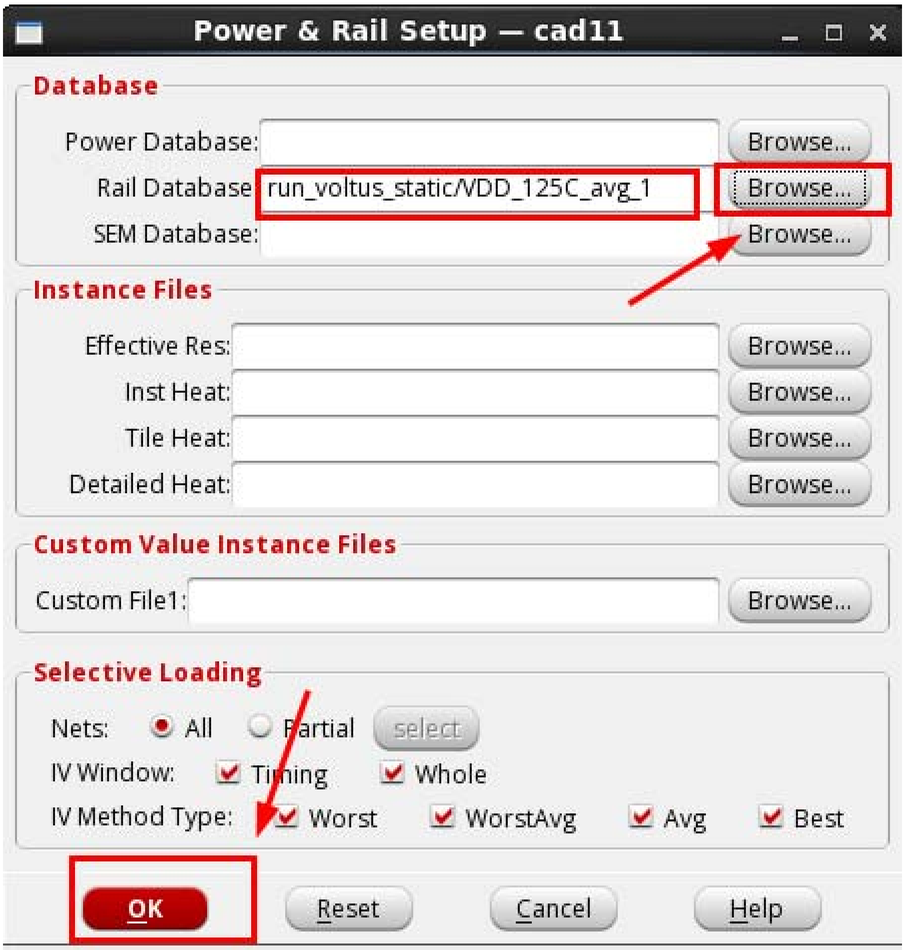
☐ Result Browser

☐ Legend

RLRP Path

d. In Power & Rail Setup window, click on Browse button, and select Rail Database directoy: **run_voltus_static/ VDD_125C_avg_1**

e. Click **OK** button to load Rail database.



Power & Rail Setup — cad11

Database

Power Database:
Rail Database: run_voltus_static/VDD_125C_avg_1
SEM Database:

Instance Files

Effective Res:
Inst Heat:
Tile Heat:
Detailed Heat:

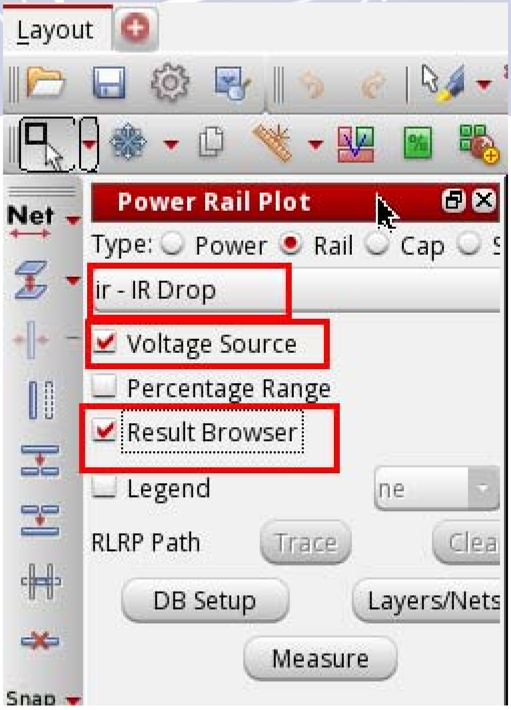
Custom Value Instance Files

Custom File1:

Selective Loading

Nets: ● All ○ Partial
IV Window: ☑ Timing ☑ Whole
IV Method Type: ☑ Worst ☑ WorstAvg ☑ Avg ☑ Best

59. In Power Rail Plot window
    - **a.** Change **“non - Clear”** to **“ir - IR Drop”**
    - **b.** Turn on: ⦿**Result Browser** and ⦿**Voltage Source**



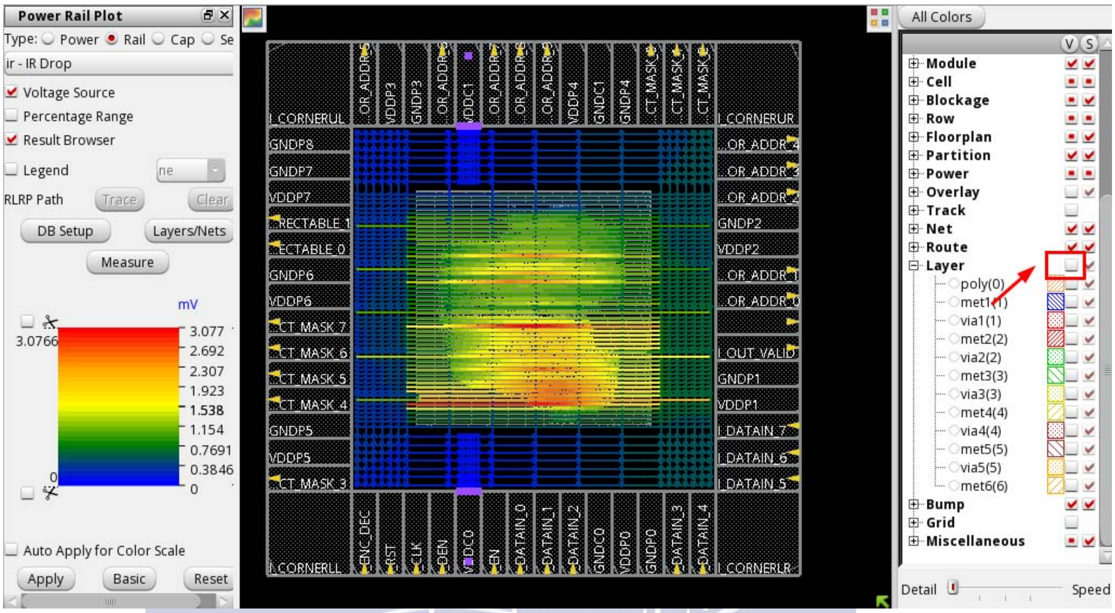
Power Rail Plot

Type: ○ Power ● Rail ○ Cap ○ S

ir - IR Drop

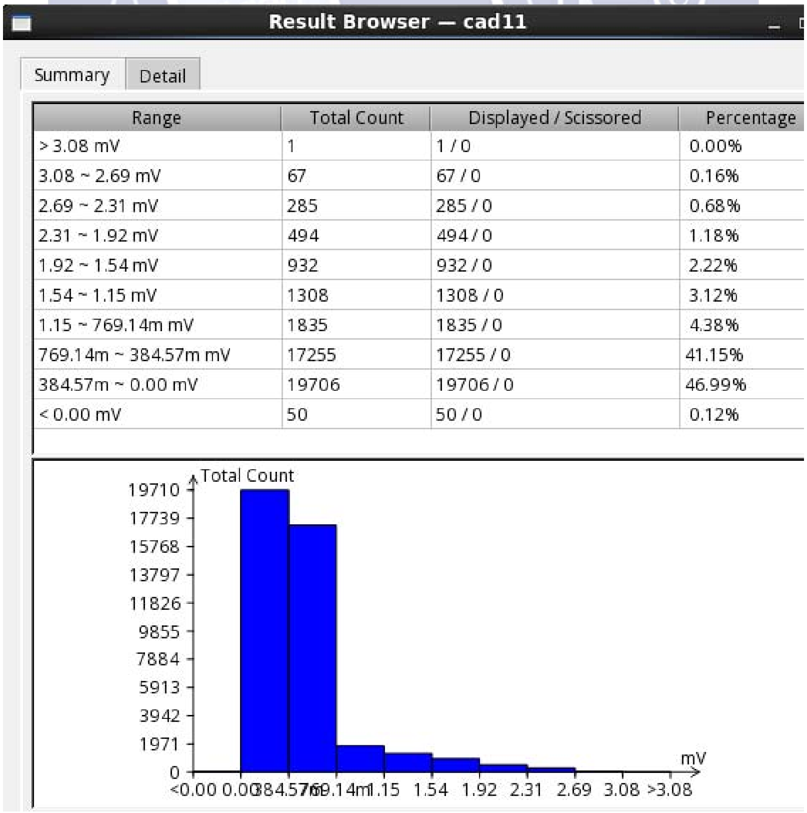
☑ Voltage Source
□ Percentage Range
☑ Result Browser
□ Legend

c. In **Display Layer Control** menu, uncheck all metals and vias **Layers** to see the IR drop result more clearly.



In Result Browser window, you can see the histogram of IR drop values, what is the overall data range for IR drop analysis?

____


Result Browser — cad11

Summary | Detail

Range	Total Count	Displayed / Scissored	Percentage
> 3.08 mV	1	1 / 0	0.00%
3.08 ~ 2.69 mV	67	67 / 0	0.16%
2.69 ~ 2.31 mV	285	285 / 0	0.68%
2.31 ~ 1.92 mV	494	494 / 0	1.18%
1.92 ~ 1.54 mV	932	932 / 0	2.22%
1.54 ~ 1.15 mV	1308	1308 / 0	3.12%
1.15 ~ 769.14m mV	1835	1835 / 0	4.38%
769.14m ~ 384.57m mV	17255	17255 / 0	41.15%
384.57m ~ 0.00 mV	19706	19706 / 0	46.99%
< 0.00 mV	50	50 / 0	0.12%

60. Close INNOVUS

innovus> exit

GUI Mode

The following steps will guide you on how to run **Cadence® Innovus™ Implementation System (INNOVUS) 21.17** Timing/SI closure flow using GUI. Each step in GUI mode has a corresponding one in command line mode. Please understand the meaning of each step.

Data Preparation

1. Remove the lab12 directory and extract LAB data from TA's directory again:

```
% rm   -rf   lab12
% tar   xzvf   ~dicta/lab12.tar.gz
```

Add I/O Pad to CORE module

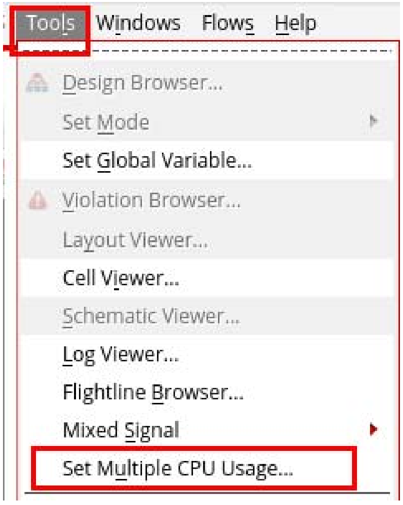
2. Change to the directory: **Netlist**
3. Add I/O PAD module to CORE module:
   Open **CHIP_syn.v** and you can see the I/O PAD module is added at the beginning of the file. The top module name is **CHIP**.
4. Open timing constraint file: **CHIP.sdc**
5. Change to the directory: **Innovus**, and open IO Pad Location file: **CHIP.io**
   Please see **Appendix A** for detail.

Reading Cell Library information and netlist for APR

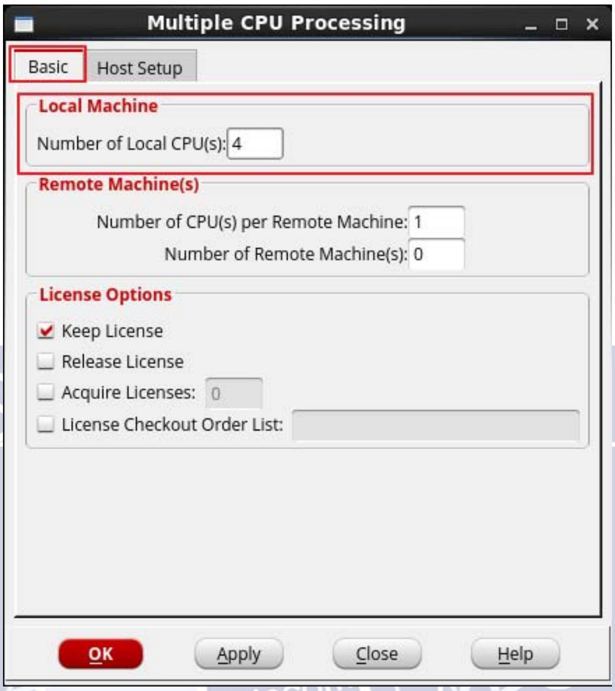
6. Start **INNOVUS** in **Innovus** directory:

```
% innovus
```

7. In INNOVUS menu, open ***Tools-> Set Multiple CPU Usages***



8. In the **Basic** tab, set the **Number of Local CPU(s)** to **4**. Then click OK to set up for the number of CPUs used by INNOVUS.



Local Machine

Number of Local CPU(s): 4

Remote Machine(s)

Number of CPU(s) per Remote Machine: 1

Number of Remote Machine(s): 0

License Options

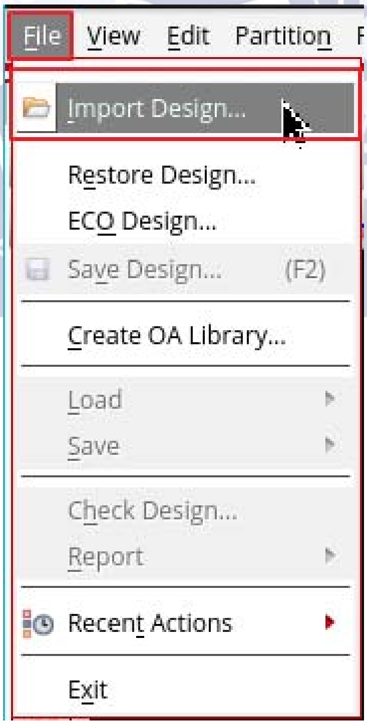
☑ Keep License

☐ Release License

☐ Acquire Licenses: 0

☐ License Checkout Order List:

9. In INNOVUS menu, open ***File -> Import Design***



10. Fill up the following field:

a. Netlist

- ➢ Files: **CHIP_syn.v**
- ➢ Top Cell: ⦿ By User : **CHIP**

b. Technology/Physical Libraries

- ⦿LEF Files: **00LIB/umc18_6lm_tech.lef**
  00LIB/umc18_6lm_core.lef
  00LIB/umc18_6lm_antenna.lef
  00LIB/umc18io3v5v_6lm.lef

Note: The order of those **.lef** files can not be interchanged.

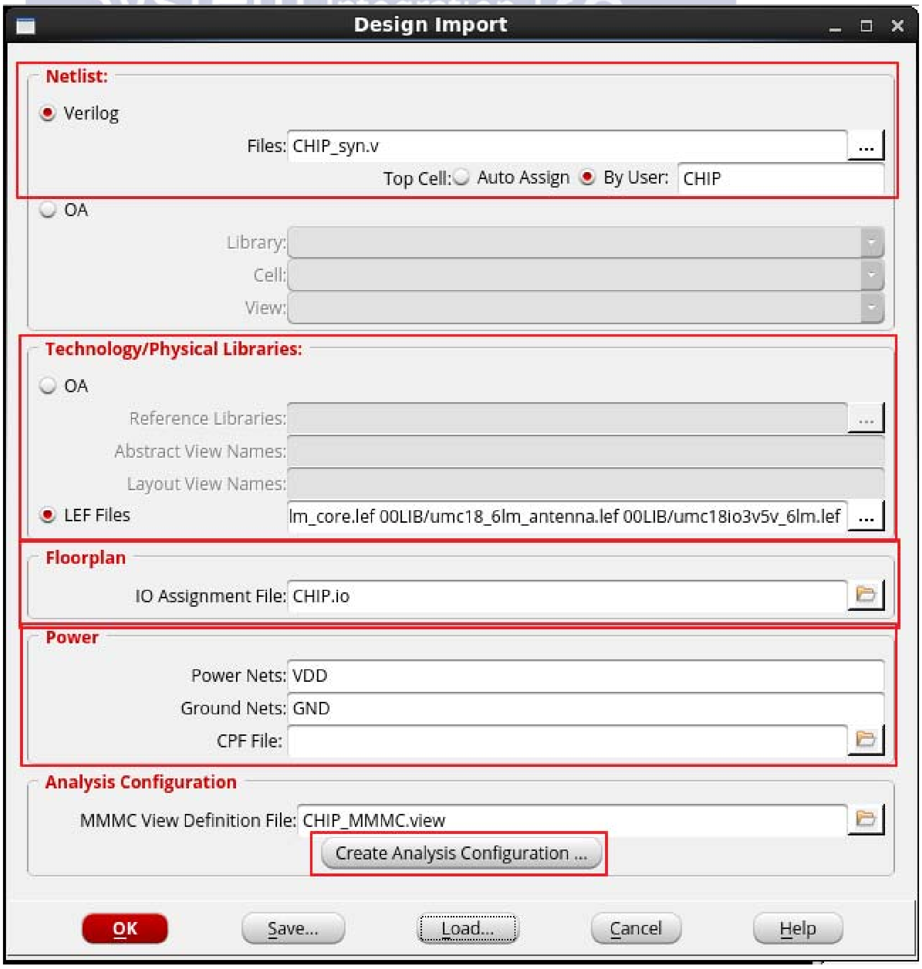
c. Floorplan

- IO Assignment Files: **CHIP.io**

d. Power

- Power Nets: **VDD**
- Ground Nets: **GND**

e. Click the **Create Analysis Configuration** button



Design Import

Netlist:
● Verilog
Files: CHIP_syn.v
Top Cell: ○ Auto Assign ● By User: CHIP
○ OA
Library:
Cell:
View:

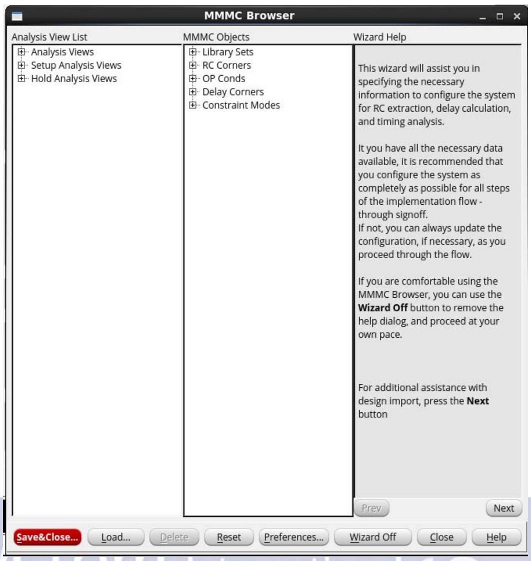
Technology/Physical Libraries:
○ OA
Reference Libraries:
Abstract View Names:
Layout View Names:
● LEF Files lm_core.lef 00LIB/umc18_6lm_antenna.lef 00LIB/umc18io3v5v_6lm.lef

Floorplan
IO Assignment File: CHIP.io

Power
Power Nets: VDD
Ground Nets: GND
CPF File:

Analysis Configuration
MMMC View Definition File: CHIP_MMMC.view

11. In **MMMC Browser**, we can configure for the MMMC objects.



This wizard will assist you in specifying the necessary information to configure the system for RC extraction, delay calculation, and timing analysis.

It you have all the necessary data available, it is recommended that you configure the system as completely as possible for all steps of the implementation flow - through signoff.
If not, you can always update the configuration, if necessary, as you proceed through the flow.

If you are comfortable using the MMMC Browser, you can use the **Wizard Off** button to remove the help dialog, and proceed at your own pace.

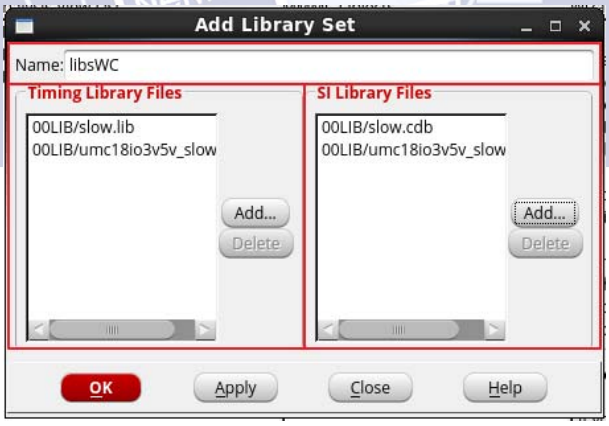
For additional assistance with design import, press the **Next** button

12. Create MMMC Objects

a. In MMMC Browser, double click on the **"Library Sets"**

b. Create a new Library Set with

- Name: **libsWC**
- Timing Library Files:
  00LIB/slow.lib
  00LIB/umc18io3v5v_slow.lib
- SI Library Files:
  00LIB/slow.cdb
  00LIB/umc18io3v5v_slow.cdb



Name: libsWC

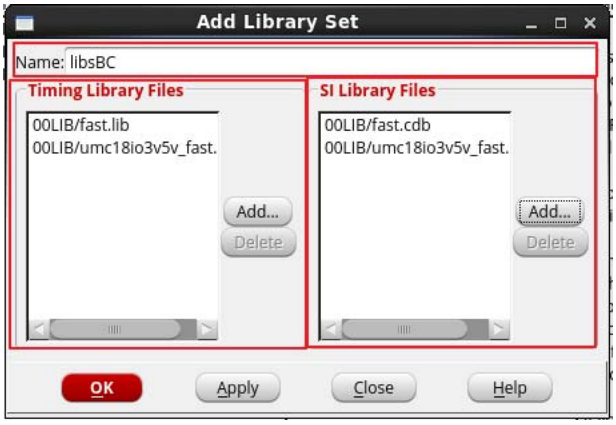
Timing Library Files:
00LIB/slow.lib
00LIB/umc18io3v5v_slow

SI Library Files:
00LIB/slow.cdb
00LIB/umc18io3v5v_slow

c. Create a new Library Set with

- Name: **libsBC**
- Timing Library Files:
  00LIB/fast.lib
  00LIB/umc18io3v5v_fast.lib
- SI Library Files:
  00LIB/fast.cdb

00LIB/umc18io3v5v_fast.cdb


Add Library Set

Name: libsBC

Timing Library Files

00LIB/fast.lib
00LIB/umc18io3v5v_fast.

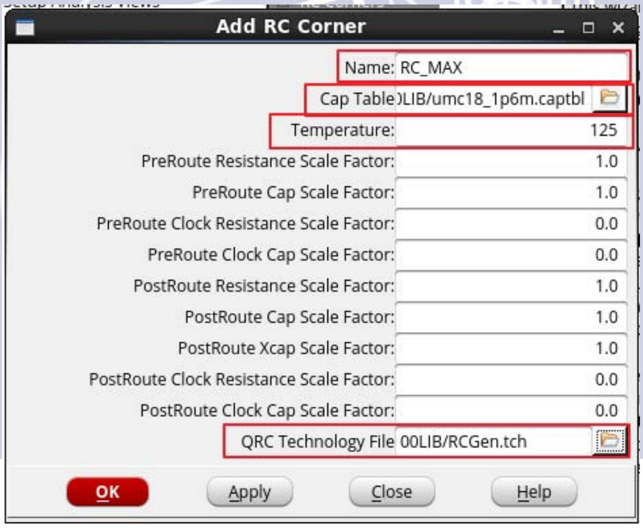
SI Library Files

00LIB/fast.cdb
00LIB/umc18io3v5v_fast.

d. In MMMC Browser, double click on **”RC Corners”**

e. Create a new RC Corner with

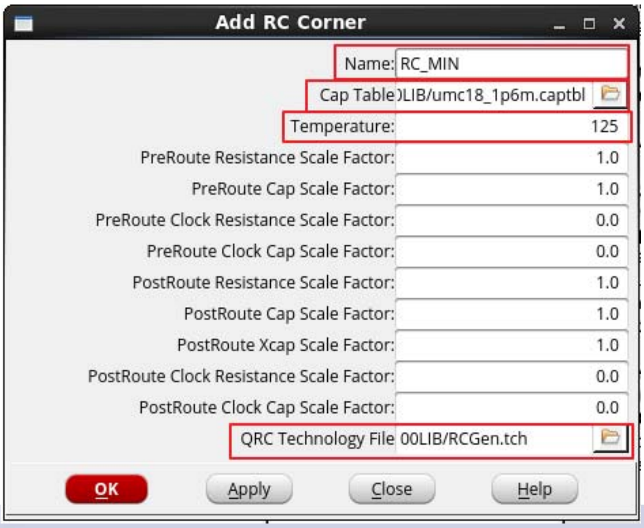
- Name: **RC_MAX**
- Cap Table: **00LIB/umc18_1p6m.captbl**
- Temperature: **125**
- QRC Technology File: **00LIB/RCGen.tch**


Add RC Corner

Name:	RC_MAX
Cap Table	)LIB/umc18_1p6m.captbl
Temperature:	125
PreRoute Resistance Scale Factor:	1.0
PreRoute Cap Scale Factor:	1.0
PreRoute Clock Resistance Scale Factor:	0.0
PreRoute Clock Cap Scale Factor:	0.0
PostRoute Resistance Scale Factor:	1.0
PostRoute Cap Scale Factor:	1.0
PostRoute Xcap Scale Factor:	1.0
PostRoute Clock Resistance Scale Factor:	0.0
PostRoute Clock Cap Scale Factor:	0.0
QRC Technology File	00LIB/RCGen.tch

f. Create a new RC Corner with

- Name: **RC_MIN**
- Cap Table: **00LIB/umc18_1p6m.captbl**
- Temperature: **0**
- QRC Technology File: **00LIB/RCGen.tch**



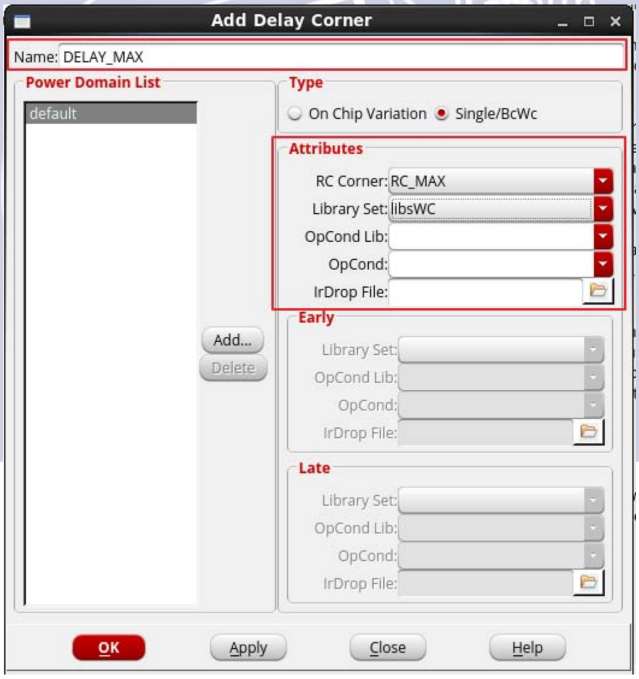
Add RC Corner

Name: RC_MIN
Cap Table)LIB/umc18_1p6m.captbl
Temperature: 125
PreRoute Resistance Scale Factor: 1.0
PreRoute Cap Scale Factor: 1.0
PreRoute Clock Resistance Scale Factor: 0.0
PreRoute Clock Cap Scale Factor: 0.0
PostRoute Resistance Scale Factor: 1.0
PostRoute Cap Scale Factor: 1.0
PostRoute Xcap Scale Factor: 1.0
PostRoute Clock Resistance Scale Factor: 0.0
PostRoute Clock Cap Scale Factor: 0.0
QRC Technology File 00LIB/RCGen.tch

g. In MMMC Browser, double click on ”**Delay Corners**”

h. Create a new Delay Corner with

- Name: **DELAY_MAX**
- RC Corner: **RC_MAX**
- Library Set: **libsWC**



Add Delay Corner

Name: DELAY_MAX

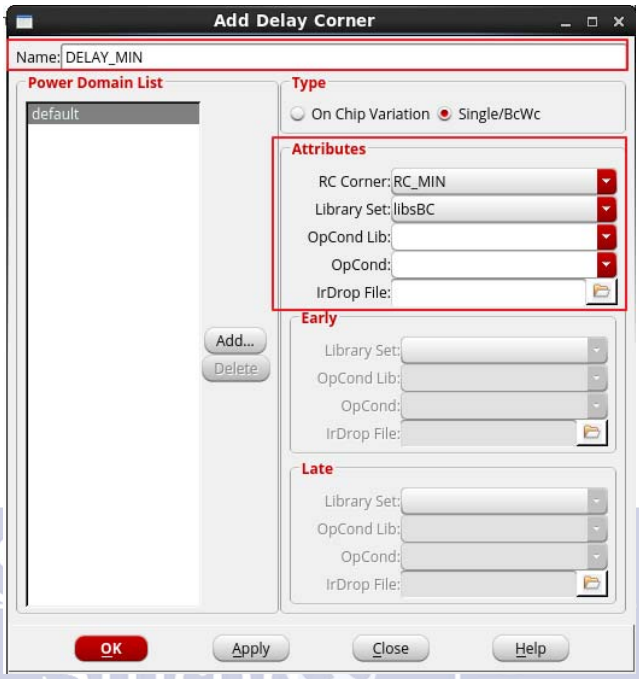
Power Domain List: default

Type: ○ On Chip Variation ● Single/BcWc

Attributes
RC Corner: RC_MAX
Library Set: libsWC
OpCond Lib:
OpCond:
IrDrop File:

i. Create a new Delay Corner with

- Name: **DELAY_MIN**
- RC Corner: **RC_MIN**
- Library Set: **libsBC**



Add Delay Corner

Name: DELAY_MIN

Power Domain List: default

Type: ○ On Chip Variation ● Single/BcWc

Attributes

RC Corner: RC_MIN

Library Set: libsBC

OpCond Lib:

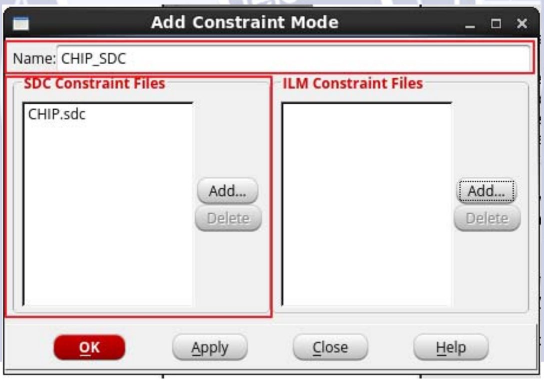
OpCond:

IrDrop File:

j. In MMMC Browser, double click on ”**Constraint Modes**”

k. Create a new Constraint Mode with

- Name: **CHIP_SDC**
- Add SDC Constraint Files: **CHIP.sdc**



Add Constraint Mode

Name: CHIP_SDC

SDC Constraint Files: CHIP.sdc

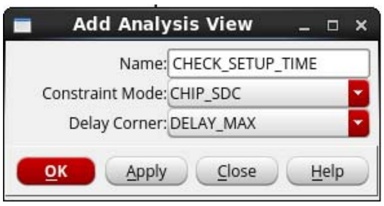
ILM Constraint Files:

13. Create analysis views and configure views for setup/hold time analysis

a. In MMMC Browser, double click on ”**Analysis Views**”

b. Create a new analysis view with

- Name: **CHECK_SETUP_TIME**
- Constraint Mode: **CHIP_SDC**
- Delay Corner: **DELAY_MAX**



Add Analysis View

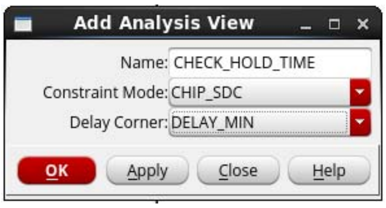
Name: CHECK_SETUP_TIME

Constraint Mode: CHIP_SDC

Delay Corner: DELAY_MAX

c. Create a new analysis view with

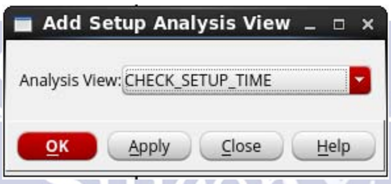
- Name: **CHECK_HOLD_TIME**
- Constraint Mode: **CHIP_SDC**
- Delay Corner: **DELAY_MIN**



Name: CHECK_HOLD_TIME
Constraint Mode: CHIP_SDC
Delay Corner: DELAY_MIN

d. In MMMC Browser, double click on **”Setup Analysis Views”**

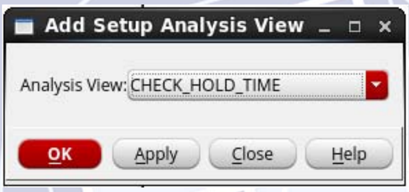
e. Add the **CHECK_SETUP_TIME** view to **Setup Analysis Views**



Analysis View: CHECK_SETUP_TIME

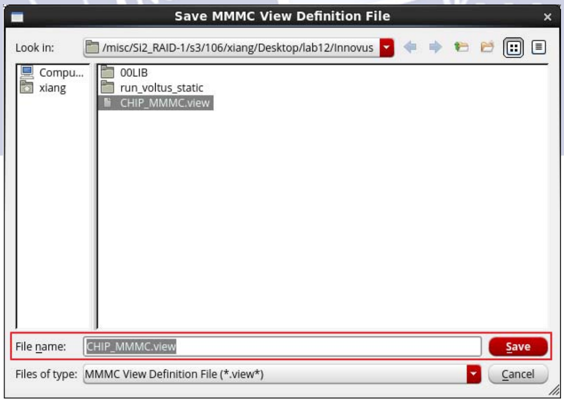
f. In MMMC Browser, double click on **”Hold Analysis Views”**

g. Add the **CHECK_HOLD_TIME** view to **Hold Analysis Views**



Analysis View: CHECK_HOLD_TIME

h. Click on Save & Close, and save the MMMC configuration as **CHIP_MMMC.view**



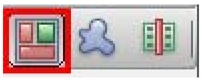
Look in: /misc/Si2_RAID-1/s3/106/xiang/Desktop/lab12/Innovus
File name: CHIP_MMMC.view
Files of type: MMMC View Definition File (*.view*)

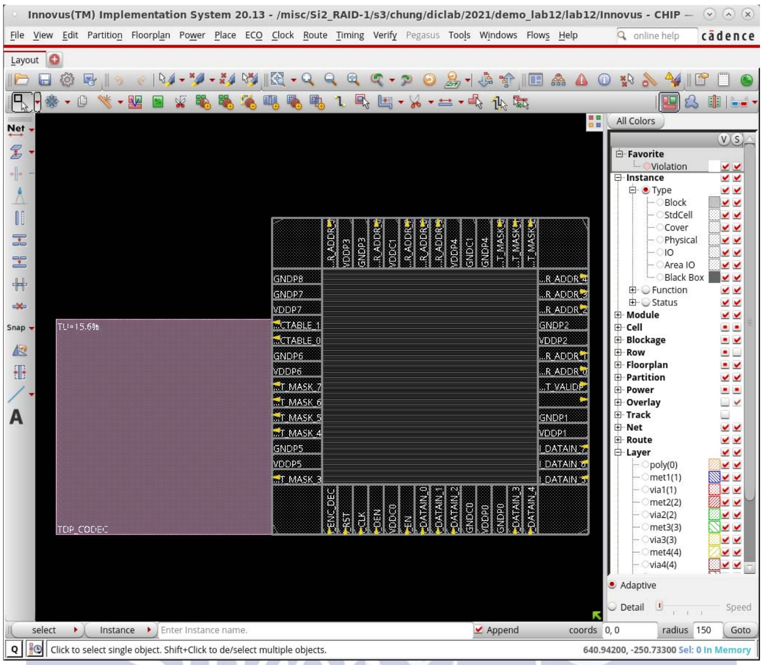
14. Save current design import settings:

a. Click **Save…** button

➢ File name: **CHIP.globals**

b. Click **OK** button

c. After design import, click  button and change to “Design View”



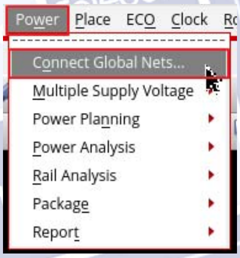
Innovus(TM) Implementation System 20.13 - /misc/Si2_RAID-1/s3/chung/diclab/2021/demo_lab12/lab12/Innovus - CHIP

Click to select single object. Shift+Click to de/select multiple objects.

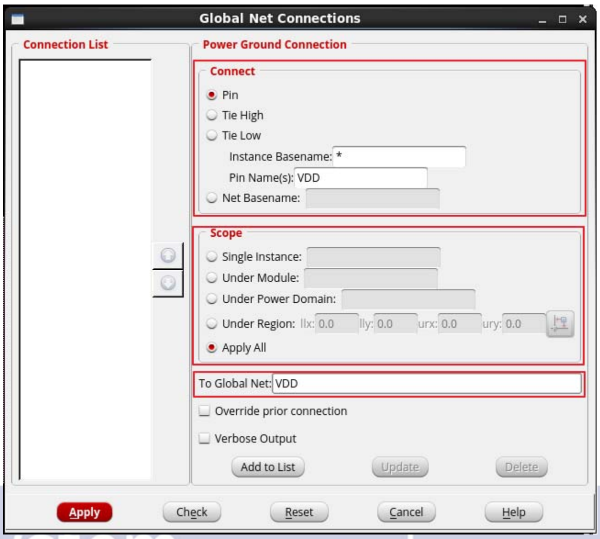
640.94200, -250.73300 Sel: 0 In Memory

Connect Global Power Nets

15. In INNOVUS menu, open ***Power -> Connections Global Nets***



16. Add all VDD pins to Connection List:
    - **a.** Connect ⦿ Pin
    - **b.** Pin Name(s): **VDD**
    - **c.** Scope ⦿ **Apply All**
    - **d.** To Global Nets: **VDD**
    - **e.** Click **Add to List** button



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

17. Add all VDD nets to Connection List:
    - **a.** Connect ⦿ Net Basename: **VDD**
    - **b.** Scope ⦿ **Apply All**
    - **c.** To Global Nets: **VDD**
    - **d.** Click **Add to List** button
18. Add all Tie High pins to Connection List:
    - **a.** Connect ⦿ **Tie High**
    - **b.** Pin Name(s): **VDD**
    - **c.** Scope ⦿ **Apply All**
    - **d.** To Global Nets: **VDD**
    - **e.** Click **Add to List** button
19. Add all GND pins to Connection List:
    - **a.** Connect ⦿ Pin
    - **b.** Pin Name(s): **GND**
    - **c.** Scope ⦿ **Apply All**
    - **d.** To Global Nets: **GND**
    - **e.** Click **Add to List** button
20. Add all GND nets to Connection List:
    - **a.** Connect ⦿ Net Basename: **GND**
    - **b.** Scope ⦿ **Apply All**
    - **c.** To Global Nets: **GND**
    - **d.** Click **Add to List** button
21. Add all Tie Low pins to Connection List:
    - **a.** Connect ⦿ **Tie Low**
    - **b.** Pin Name(s): **GND**
    - **c.** Scope ⦿ **Apply All**

d. To Global Nets: **GND**

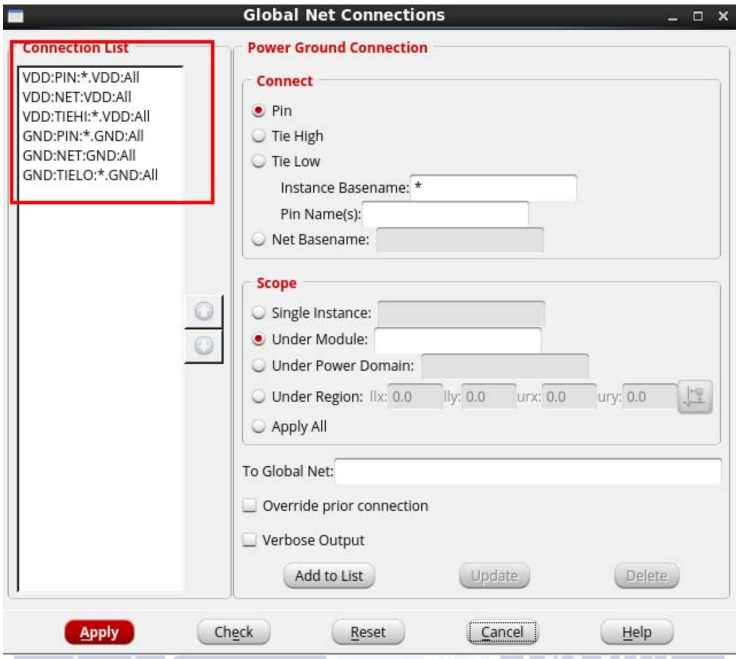
e. Click **Add to List** button

22. Apply the global connection list:

a. Click **Apply** button.

b. Click **Cancel** button to close the window

c. After setting the global net connections, the ***Connection List*** should look like the following figure.



Connection List:

VDD:PIN:*.VDD:All
VDD:NET:VDD:All
VDD:TIEHI:*.VDD:All
GND:PIN:*.GND:All
GND:NET:GND:All
GND:TIELO:*.GND:All

Connect: ● Pin ○ Tie High ○ Tie Low

Instance Basename: *

Pin Name(s):

○ 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:

☐ Override prior connection

☐ Verbose Output

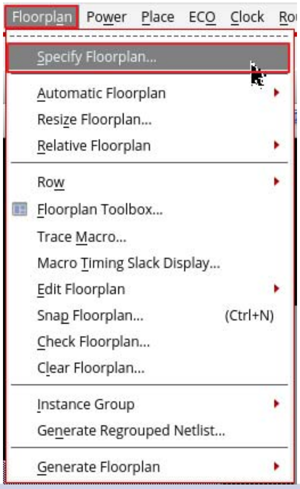
23. In INNOVUS command prompt, execute the following command:

innovus> setDesignMode -process 180

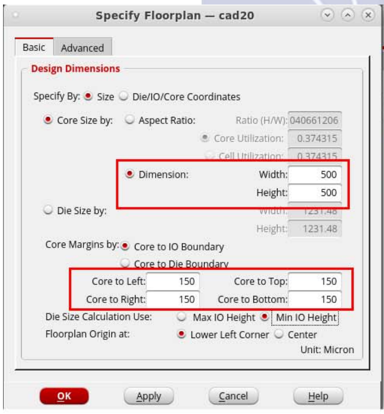
This command will setup up the default process technology information to **180nm** process, and the following RC extraction engine and cross talk analysis engine will change their default parameter specified for this process.

Define Chip Floorplan

24. In INNOVUS menu, open ***Floorplan -> Specify Floorplan***



25. Design Dimensions:
    a. Specify By ⦿ Size
    b. ⦿ Core Size by: ⦿ Dimension:
    c. Width: **500**
    d. Height: **500**
26. Core Margins by ⦿ Core to IO Boundary:
    a. Core to Left: **150**
    b. Core to Right: **150**
    c. Core to Top: **150**
    d. Core to Bottom: **150**
27. Apply the specify floorplan:
    a. Click **OK** button



Specify Floorplan — cad20

Basic | Advanced

Design Dimensions

Specify By: ● Size ○ Die/IO/Core Coordinates

● Core Size by: ○ Aspect Ratio: Ratio (H/W): 040661206

● Core Utilization: 0.374315

○ Cell Utilization: 0.374315

● Dimension: Width: 500 Height: 500

○ Die Size by: Width: 1231.48 Height: 1231.48

Core Margins by: ● Core to IO Boundary ○ Core to Die Boundary

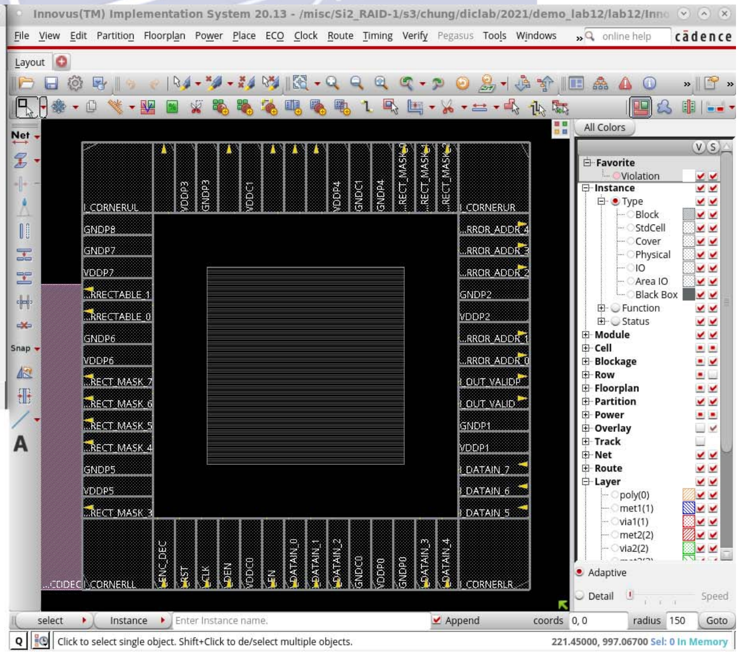
Core to Left: 150 Core to Top: 150

Core to Right: 150 Core to Bottom: 150

Die Size Calculation Use: ○ Max IO Height ● Min IO Height

Floorplan Origin at: ● Lower Left Corner ○ Center

Unit: Micron



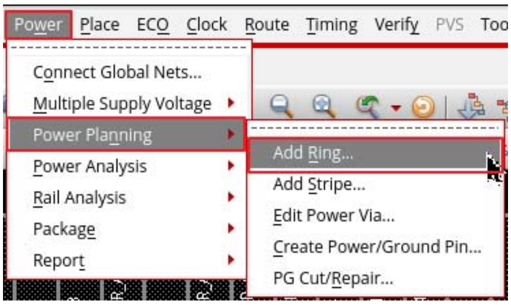
Innovus(TM) Implementation System 20.13 - /misc/Si2_RAID-1/s3/chung/diclab/2021/demo_lab12/lab12/Inno

Click to select single object. Shift+Click to de/select multiple objects.

221.45000, 997.06700 Sel: 0 In Memory

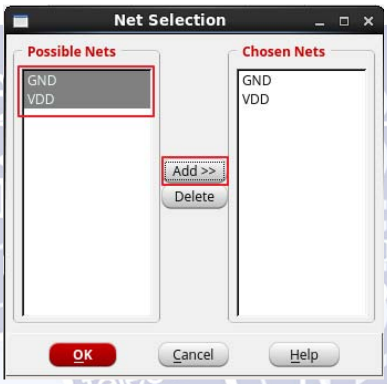
Power Planning (Add Power Rings)

28. In INNOVUS menu, open ***Power -> Power Planning -> Add Ring…***



29. In the **Basic** tab, fill the following field:

a. Add Nets: GND VDD



b. Ring Type: ⦿ Core ring(s) contouring

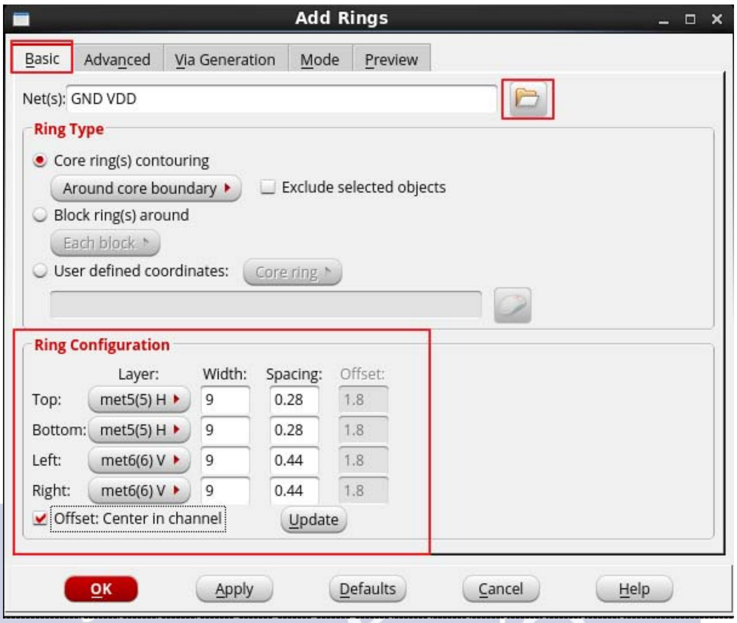
c. Specify metal layers and width

- ➢ Top: **met5(5) H** Width: **9**
- ➢ Bottom: **met5(5) H** Width: **9**
- ➢ Left: **met6(6) V** Width: **9**
- ➢ Right: **met6(6) V** Width: **9**

d. Click **Update** button, and it will automatically adjust the required spacing.

e. Specify power ring offset

- ➢ ⦿ Offset: Center in channel



Add Rings

Basic | Advanced | Via Generation | Mode | Preview

Net(s): GND VDD

Ring Type

● Core ring(s) contouring

Around core boundary ▸ □ Exclude selected objects

○ Block ring(s) around

Each block ▸

○ User defined coordinates: Core ring ▸

Ring Configuration

	Layer:	Width:	Spacing:	Offset:
Top:	met5(5) H ▸	9	0.28	1.8
Bottom:	met5(5) H ▸	9	0.28	1.8
Left:	met6(6) V ▸	9	0.44	1.8
Right:	met6(6) V ▸	9	0.44	1.8

☑ Offset: Center in channel   Update

OK   Apply   Defaults   Cancel   Help

30. In the **Advanced** tab, fill the following field:

   a. Configure wire group

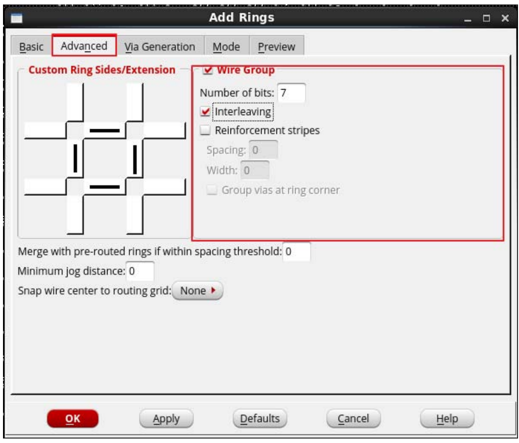
   - ☑ **Wire group**
   - Number of bits: **7**
   - ☑ **Interleaving**

31. Apply the power ring specification:

   a. Click **Apply** button.

   b. Check if the ring is correctly created. If not, click **undo** button ↶ (in INNOVUS toolbar) and repeat step 29 to step 30 again.

   c. Click **Cancel** button to close the **Add Rings** menu



Add Rings

Basic | Advanced | Via Generation | Mode | Preview

Custom Ring Sides/Extension

☑ Wire Group

Number of bits: 7

☑ Interleaving

□ Reinforcement stripes

Spacing: 0

Width: 0

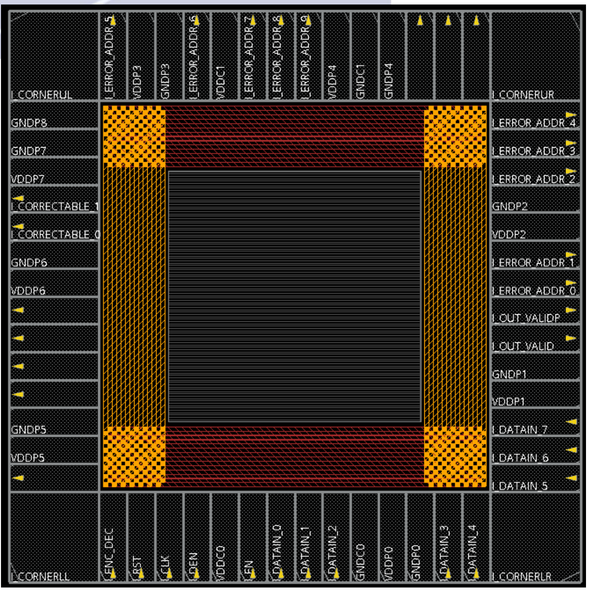
□ Group vias at ring corner

Merge with pre-routed rings if within spacing threshold: 0

Minimum jog distance: 0

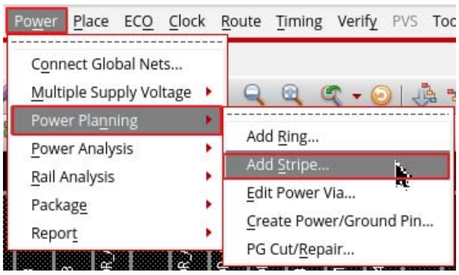
Snap wire center to routing grid: None ▸

OK   Apply   Defaults   Cancel   Help



Power Planning (Add Stripes)

32. In INNOVUS menu, open ***Power -> Power Planning -> Add Stripe…***



33. Create a global power network on metal 6:

a. Add Nets: **GND VDD**

b. Specify metal layer, width, direction and spacing

- ➢ Layer: **met6(6)**
- ➢ Direction: ⦿ **Vertical**
- ➢ Width: **9**
- ➢ Click **Update** button, and it will automatically adjust the required spacing.

c. Set Pattern

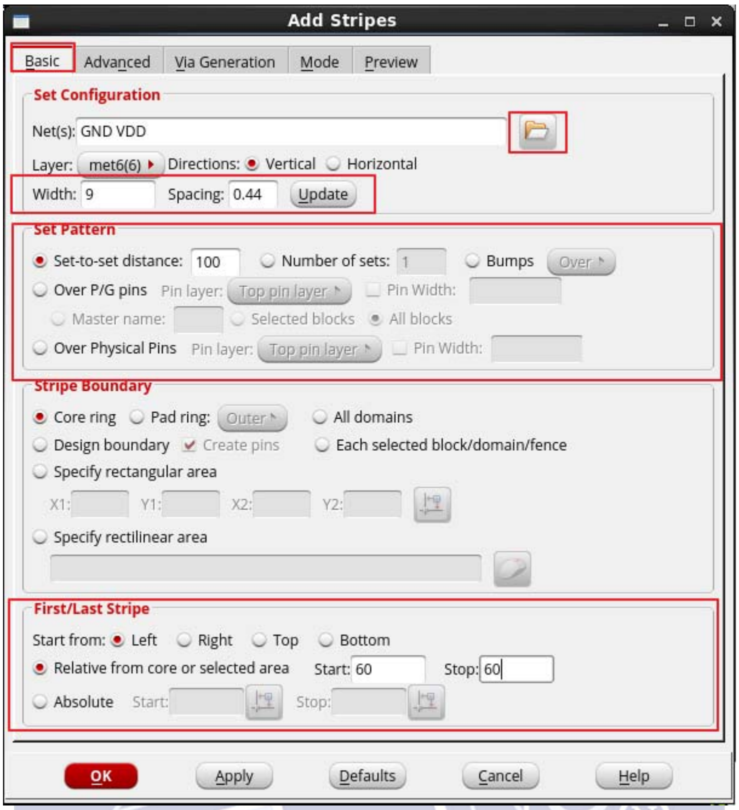
- ➢ Set-to-set distance: **100**

d. First/Last Stripe ⦿ Relative from the core or selected area

- ➢ Start: **60**
- ➢ Stop: **60**

e. Click **Apply** button

- ➢ Check if the stripes are correctly created. If not, click **undo** button ↶ (in INNOVUS toolbar) and repeat step 33.



Set Configuration

Net(s): GND VDD

Layer: met6(6) Directions: ● Vertical ○ Horizontal

Width: 9 Spacing: 0.44 Update

Set Pattern

● Set-to-set distance: 100 ○ Number of sets: 1 ○ Bumps Over

○ Over P/G pins Pin layer: Top pin layer □ Pin Width:

○ Master name: ○ Selected blocks ● All blocks

○ Over Physical Pins Pin layer: Top pin layer □ Pin Width:

Stripe Boundary

● Core ring ○ Pad ring: Outer ○ All domains

○ Design boundary ☑ Create pins ○ Each selected block/domain/fence

○ Specify rectangular area

X1: Y1: X2: Y2:

○ Specify rectilinear area

First/Last Stripe

Start from: ● Left ○ Right ○ Top ○ Bottom

● Relative from core or selected area Start: 60 Stop: 60

○ Absolute Start: Stop:

34. Create a global power network on metal 5:

a. Specify metal layer, width, direction and spacing

➢ Layer: **met5(5)**

➢ Direction: ⦿ **Horizontal**

➢ Width: **9**

➢ Click **Update** Button, and it will automatically adjust the required spacing.

b. Set Pattern

➢ Set-to-Set distance: **100**

c. First/Last Stripe ⦿ Relative from the core or selected area

➢ Start: **60**

➢ Stop: **60**

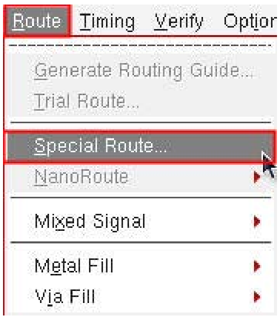
d. Click **Apply** button

➢ Check if the stripes are correctly created. If not, click **undo** button ↶ (in INNOVUS toolbar) and repeat step 34 again.

e. Click **Cancel** button to close the **Add Stripes** menu

Connect Core Power Pins

35. In INNOVUS menu, open ***Route -> Special Route***



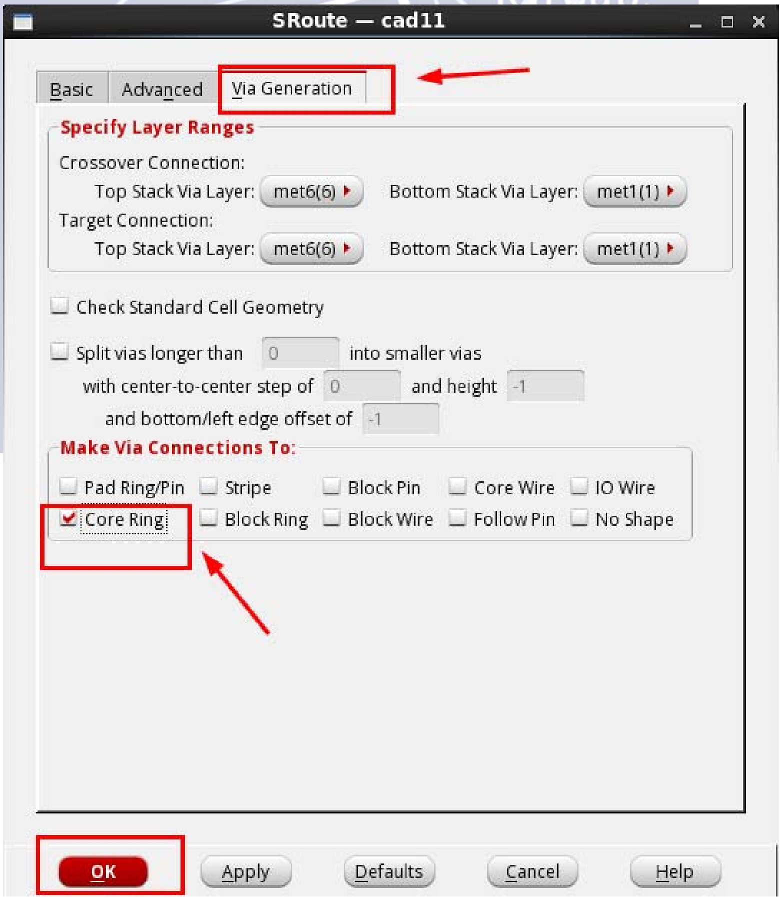
36. Connect core power:

a. Add Nets: **GND VDD**

b. SRoute (Special Net Routing):

- ➢ □ Block Pins
- ➢ □ Pad Rings
- ➢ □ Floating Stripes
- ➢ ☑ **Pad Pins**
- ➢ □ Follow Pins

c. In **Via Generation** table, Make via Connection To: **Core Ring**



SRoute — cad11

Basic | Advanced | Via Generation

Specify Layer Ranges

Crossover Connection:
Top Stack Via Layer: met6(6) Bottom Stack Via Layer: met1(1)

Target Connection:
Top Stack Via Layer: met6(6) Bottom Stack Via Layer: met1(1)

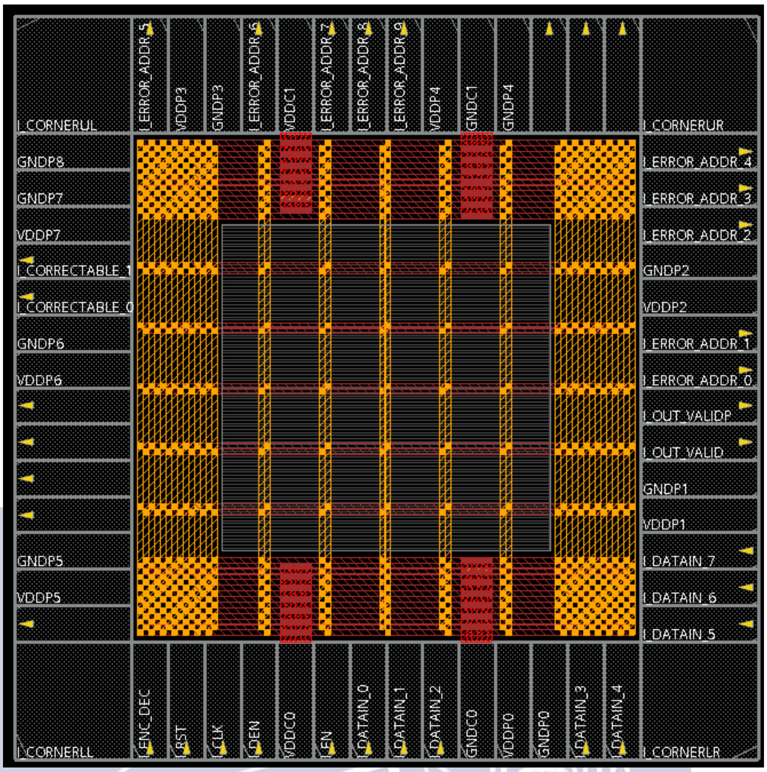
□ Check Standard Cell Geometry

□ Split vias longer than 0 into smaller vias
with center-to-center step of 0 and height -1
and bottom/left edge offset of -1

Make Via Connections To:

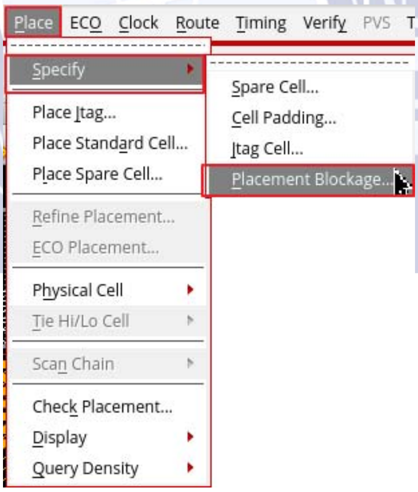
□ Pad Ring/Pin □ Stripe □ Block Pin □ Core Wire □ IO Wire
☑ Core Ring □ Block Ring □ Block Wire □ Follow Pin □ No Shape

d. Click **OK** button, and power rings will be connected to the core power pads.



Place Standard Cells

37. In INNOVUS menu, open ***Place -> Specify -> Placement Blockage***



38. Specify the placement blockage to avoid placing cells under M1/M2/M3 stripes

   a. Specify placement blockage

   ➢ ☑ **M1**

   ➢ ☑ **M2**

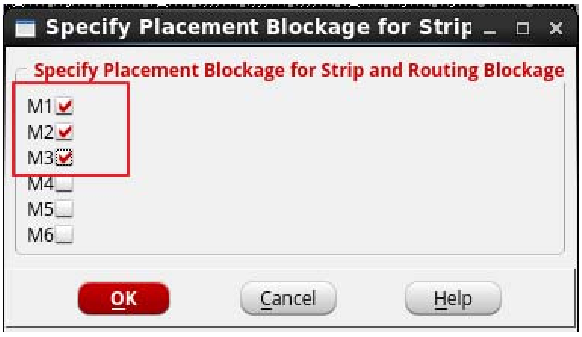
   ➢ ☑ **M3**

   ➢ □ M4

   ➢ □ M5

   ➢ □ M6

b. Click **OK** button

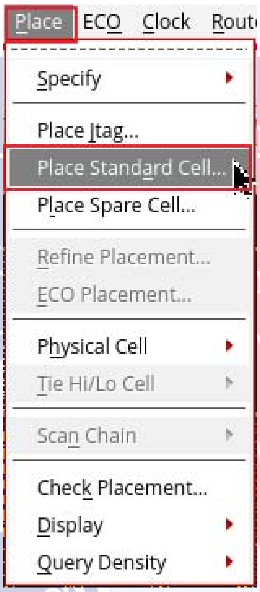


Specify Placement Blockage for Strip

Specify Placement Blockage for Strip and Routing Blockage

M1 ☑
M2 ☑
M3 ☑
M4 □
M5 □
M6 □

39. In INNOVUS menu, open ***Place -> Place Standard Cell***



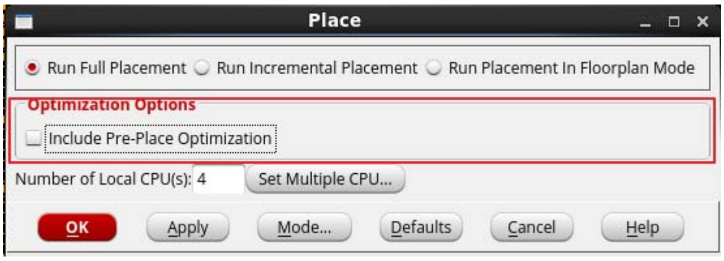
40. In **Place** menu, Setup those placement options:

a. ⦿ **Run Full Placement**

b. Optimization Options

- ➢ □ Include Pre-Place Optimization

c. Click **OK** button to start standard cell placement.



Place

● Run Full Placement ○ Run Incremental Placement ○ Run Placement In Floorplan Mode

Optimization Options

□ Include Pre-Place Optimization

Number of Local CPU(s): 4

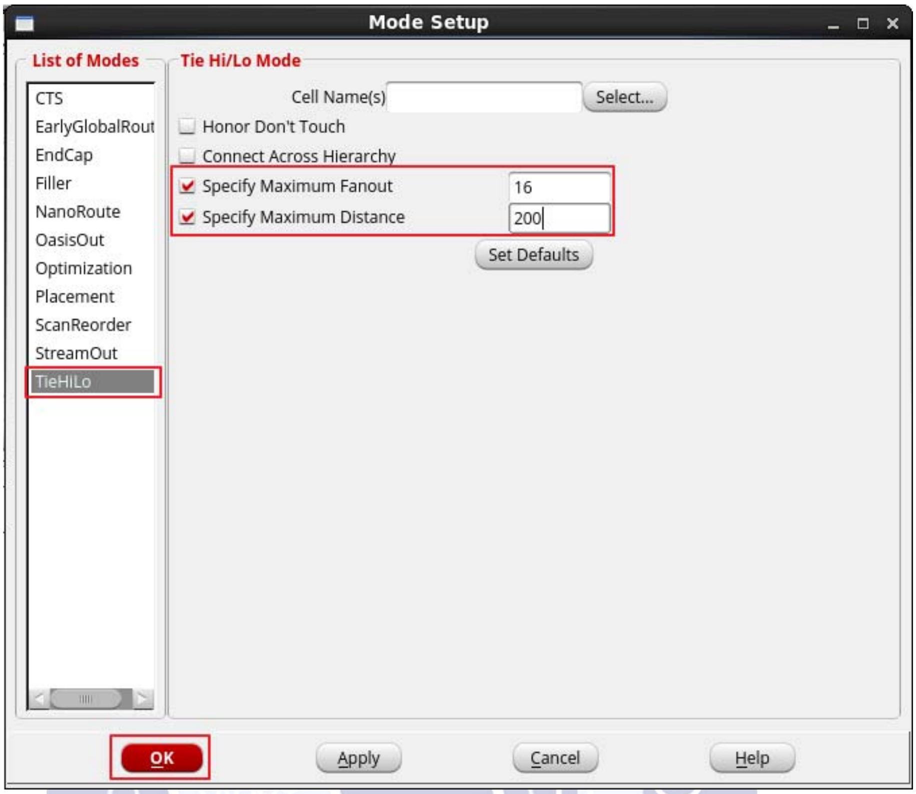
41. In INNOVUS menu, open ***Tools -> Set Mode -> Mode Setup***

a. In **TieHiLo** table, turn on the options

- ➢ □ Honor Don’t Touch
- ➢ □ Connect Across Hierarchy
- ➢ ☑ Specify Maximum Fanout : 16

- ☑ Specify Maximum Distance : 200

b. Click **OK** button


Mode Setup

List of Modes: CTS, EarlyGlobalRout, EndCap, Filler, NanoRoute, OasisOut, Optimization, Placement, ScanReorder, StreamOut, TieHiLo

Tie Hi/Lo Mode

Cell Name(s)

□ Honor Don't Touch

□ Connect Across Hierarchy

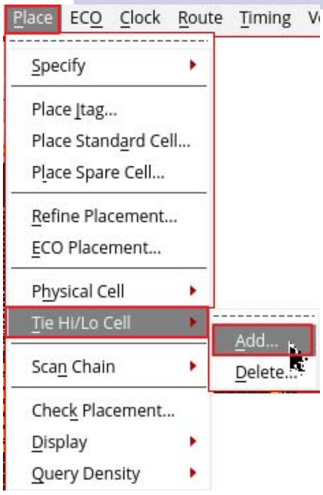
☑ Specify Maximum Fanout 16

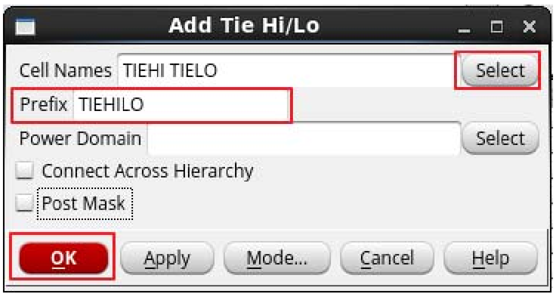
☑ Specify Maximum Distance 200

42. In INNOVUS menu, open ***Place -> Tie Hi/Lo Cells -> Add***

a. Setup the options:

- Cell Names: **TIEHI TIELO**
- Prefix: **TIEHILO**
- Click **OK** button




Add Tie Hi/Lo

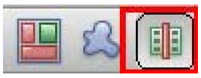
Cell Names TIEHI TIELO

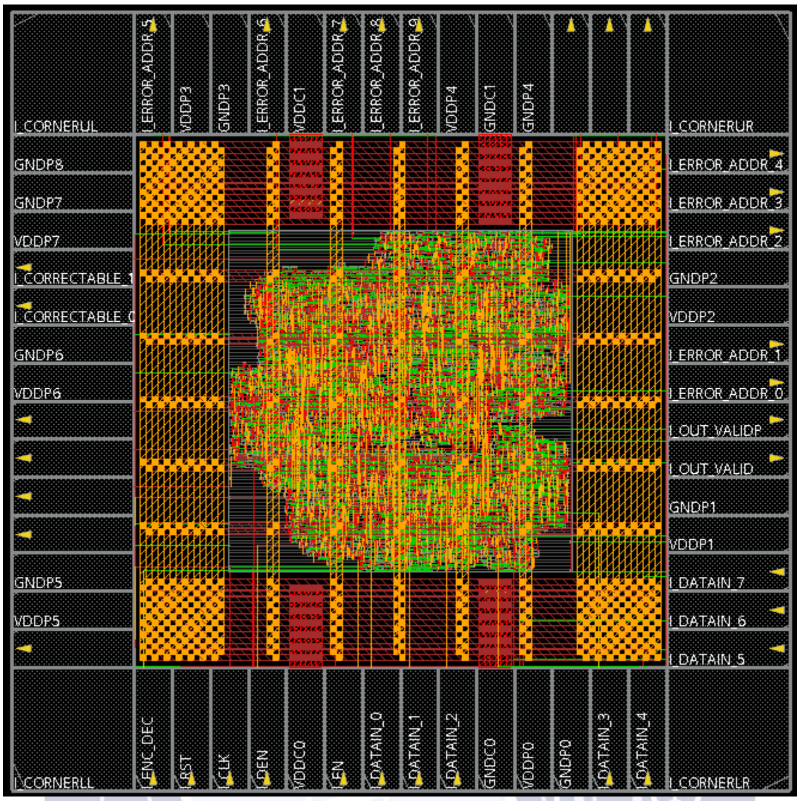
Prefix TIEHILO

Power Domain

□ Connect Across Hierarchy

□ Post Mask

43. After placement is done, click  button and change “Design View” to “Physical View”.

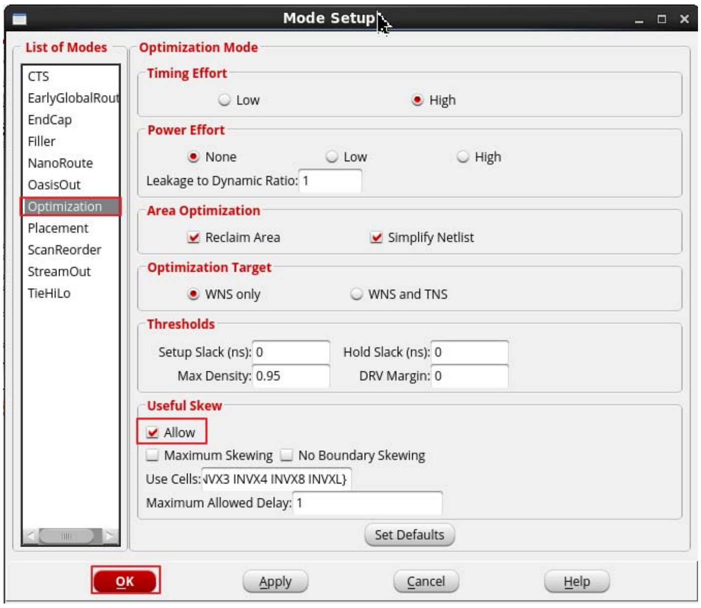


Design Optimization: preCTS

44. In INNOVUS menu, open ***Tools -> Set Mode -> Mode Setup***

a. In **Optimization** table, in **Useful Skew** option, turn on the options for **Allow**

b. Click **OK** button



Optimization Mode

Timing Effort
○ Low ● High

Power Effort
● None ○ Low ○ High
Leakage to Dynamic Ratio: 1

Area Optimization
☑ Reclaim Area ☑ Simplify Netlist

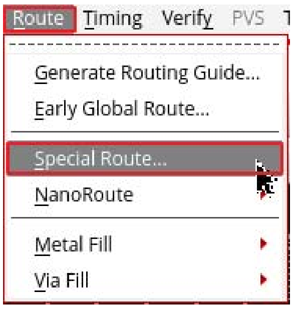
Optimization Target
● WNS only ○ WNS and TNS

Thresholds
Setup Slack (ns): 0 Hold Slack (ns): 0
Max Density: 0.95 DRV Margin: 0

Useful Skew
☑ Allow
□ Maximum Skewing □ No Boundary Skewing
Use Cells: NVX3 INVX4 INVX8 INVXL}
Maximum Allowed Delay: 1

Connect Standard Cell Power Pin

45. In INNOVUS menu, open ***Route -> Special Route***

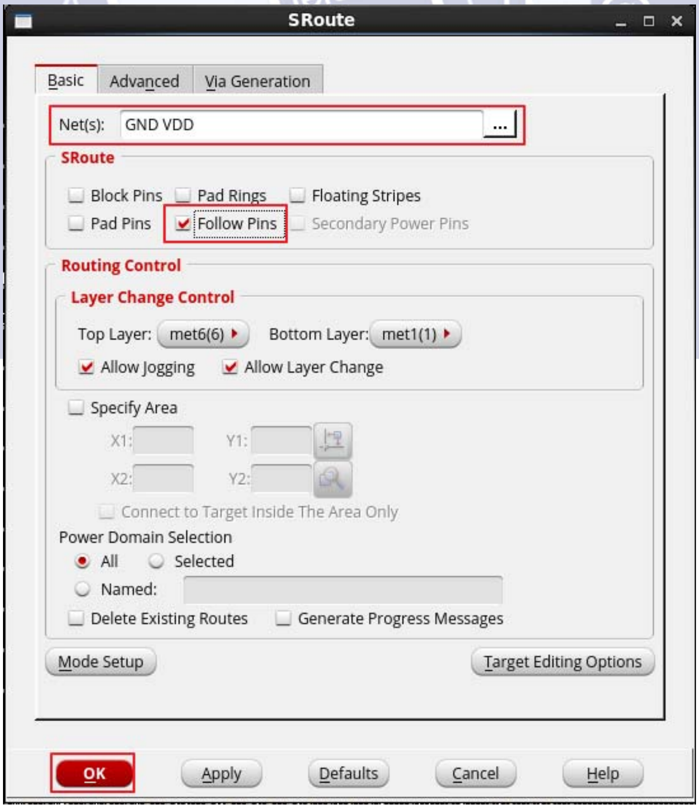


46. Connect standard cell power pin:

a. Add Nets: **GND VDD**

b. SRoute (Special Net Routing):

- ➢ □ Block Pins
- ➢ □ Pad Rings
- ➢ □ Floating Stripes
- ➢ □ Pad Pins
- ➢ ☑ **Follow Pins**



SRoute

Basic | Advanced | Via Generation

Net(s): GND VDD ...

SRoute

□ Block Pins □ Pad Rings □ Floating Stripes

□ Pad Pins ☑ Follow Pins □ Secondary Power Pins

Routing Control

Layer Change Control

Top Layer: met6(6) Bottom Layer: met1(1)

☑ Allow Jogging ☑ Allow Layer Change

□ Specify Area

X1: Y1:

X2: Y2:

□ Connect to Target Inside The Area Only

Power Domain Selection

● All ○ Selected

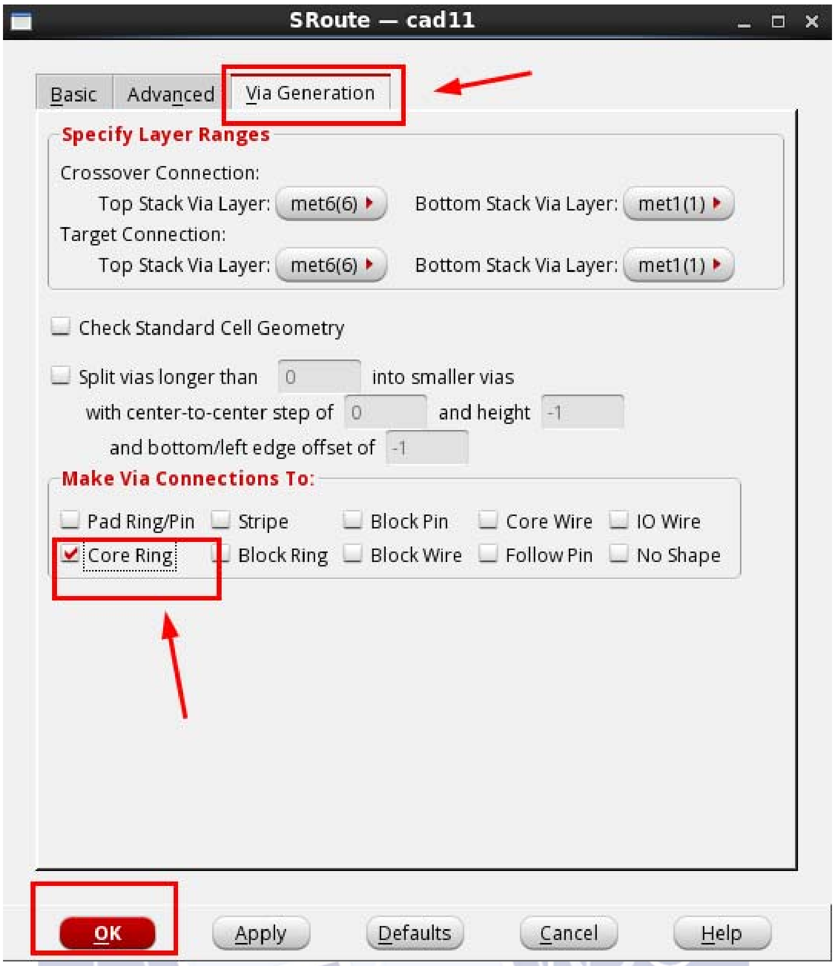
○ Named:

□ Delete Existing Routes □ Generate Progress Messages

Mode Setup | Target Editing Options

OK | Apply | Defaults | Cancel | Help

c. In **Via Generation** table, Make via Connection To: **Core Ring**



SRoute — cad11

Basic | Advanced | Via Generation

Specify Layer Ranges

Crossover Connection:

Top Stack Via Layer: met6(6) Bottom Stack Via Layer: met1(1)

Target Connection:

Top Stack Via Layer: met6(6) Bottom Stack Via Layer: met1(1)

☐ Check Standard Cell Geometry

☐ Split vias longer than 0 into smaller vias

with center-to-center step of 0 and height -1

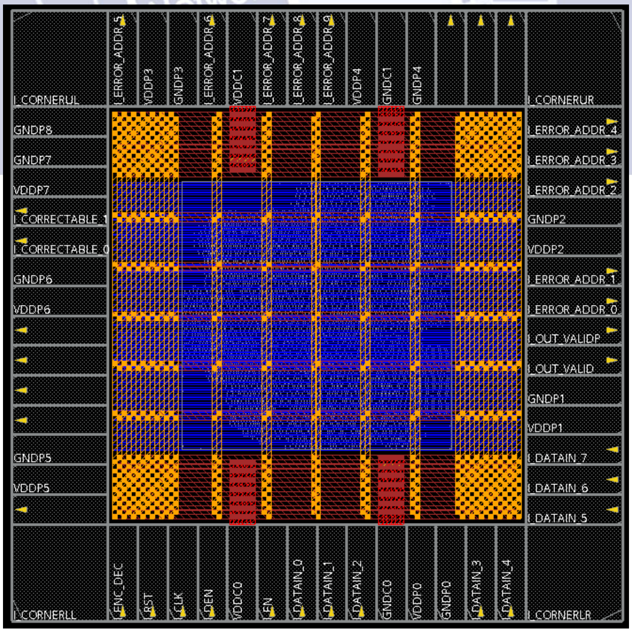
and bottom/left edge offset of -1

Make Via Connections To:

☐ Pad Ring/Pin ☐ Stripe ☐ Block Pin ☐ Core Wire ☐ IO Wire

☑ Core Ring ☐ Block Ring ☐ Block Wire ☐ Follow Pin ☐ No Shape

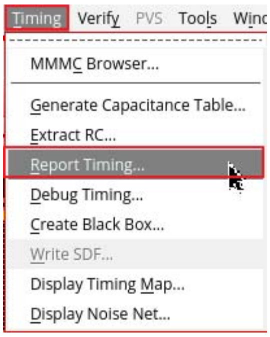
d. Click **OK** button



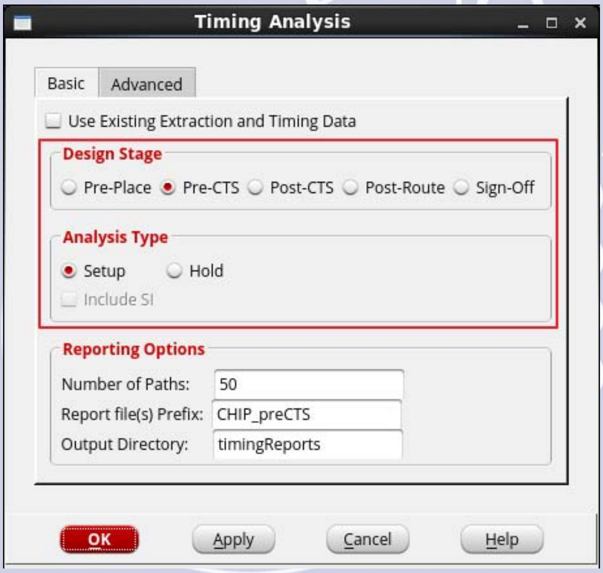
In-Place Optimization (IPO)

- Before Clock Tree Synthesis

47. In INNOVUS menu, open ***Timing -> Report Timing***



48. INNOVUS performs trial route, extract RC, estimates the interconnection RC effects and then perform delay calculation for timing analysis.
    a. Design Stage: ⦿ **Pre-CTS**
    b. Analysis Type: ⦿ **Setup**
    c. Click **OK** button



□ Use Existing Extraction and Timing Data

Design Stage
○ Pre-Place ● Pre-CTS ○ Post-CTS ○ Post-Route ○ Sign-Off

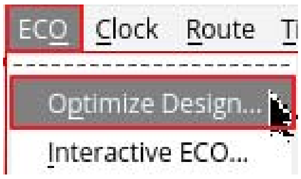
Analysis Type
● Setup ○ Hold
□ Include SI

Reporting Options
Number of Paths: 50
Report file(s) Prefix: CHIP_preCTS
Output Directory: timingReports

49. Timing analysis reports are stored in **timingReports/** directory, **CHIP_preCTS_all.tarpt.gz** shows the setup timing analysis results. Slack time should be a positive value in this file. Moreover, DRVs report files are ***.cap.gz**, ***.fanout.gz**, and ***.tran.gz**.

```
%gvim    CHIP_preCTS_all.tarpt.gz
```

50. If the timing slack is negative, or there has DRVs, open ***ECO -> Optimize Design*** in INNOVUS menu



51. Perform pre-CTS IPO

a. Design Stage: ⦿ **Pre-CTS**

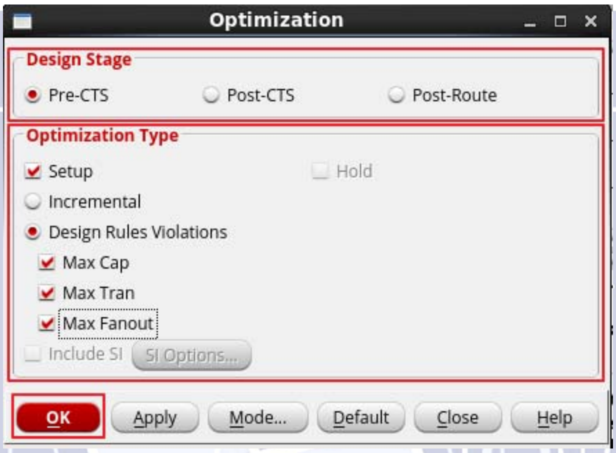
b. Optimization Type

- ➢ ☑ **Setup**
- ➢ ⦿ **Design Rule Violations**

  ☑ **Max Cap** ☑ **Max Tran** ☑ **Max Fanout**

c. Click **OK** button, INNOVUS will start to optimize your design.

d. If timing slack after pre-CTS IPO is still negative, change Optimization type to ⦿ **Incremental**. Then perform preCTS optimization iteratively until timing slack becomes positive. In preCTS stage, only the setup time is checked.



Optimization

Design Stage

● Pre-CTS ○ Post-CTS ○ Post-Route

Optimization Type

☑ Setup □ Hold

○ Incremental

● Design Rules Violations

☑ Max Cap

☑ Max Tran

☑ Max Fanout

□ Include SI

Clock Tree Synthesis (CTS)

52. Create clock tree specification file from the SDC constraint file

a. In INNOVUS command prompt, execute the following command:

```
innovus> create_ccopt_clock_tree_spec -file CHIP.CCOPT.spec \
-keep_all_sdc_clocks
```

53. Load clock tree specification file and then synthesize the clock tree

a. In INNOVUS command prompt, execute the following command:

```
innovus> source   CHIP.CCOPT.spec
```

b. In INNOVUS command prompt, execute the following command:

```
innovus> ccopt_design
```

54. Use Clock Tree Debugger to display the clock tree of the design

a. **Repeat step 22 to step 24** in command line mode **Clock Tree Synthesis (CTS)** section to display the clock tree synthesis result

In-Place Optimization (IPO)

- After Clock Tree Synthesis

55. In INNOVUS menu, open ***Timing -> Report Timing***. After clock tree synthesis, the clock network delay is not ideal anymore.

56. INNOVUS performs trial route, extract RC, estimates the interconnection RC effects and then perform delay calculation for timing analysis

    a. Design Stage: ⊙ **Post-CTS**

    b. Analysis Type: ☑ **Setup**

    c. Click **OK** button

57. After CTS, further timing optimization is performed to meet timing constraints if there has negative timing slack or DRVs. Open ***ECO -> Optimize Design*** in the INNOVUS menu.

58. Perform post-CTS IPO

    a. Design Stage: ⊙ **Post-CTS**

    b. Optimization Type:

    ➢ ☑ **Setup**

    ➢ ⊙ **Design Rule Violations**

    ☑ **Max Cap** ☑ **Max Tran** ☑ **Max Fanout**

    c. Click **OK** button

    d. If setup timing slack is still negative, change Optimization type to ⊙ **Incremental**. Then perform postCTS optimization iteratively until setup timing slack becomes positive.

59. Verify that if the hold time constraint is satisfied or not. Open ***Timing -> Report Timing*** in INNOVUS menu

    a. Design Stage: ⊙ **Post-CTS**

    b. Analysis Type: ☑ **Hold**

    c. Click **OK** button

60. If hold time slack is negative, open ***ECO -> Optimize Design*** in the INNOVUS menu.

61. Perform post-CTS IPO for hold time fixing

    ➢ Design Stage: ⊙ **Post-CTS**

    ➢ Optimization Type:

    ➢ ☑ **Hold**

    ○ Design Rule Violations

    ☑ Max Cap ☑ Max Tran ☑ Max Fanout

    ➢ Click **OK** button

    ➢ If hold timing slack is still negative, change Optimization type to ⊙

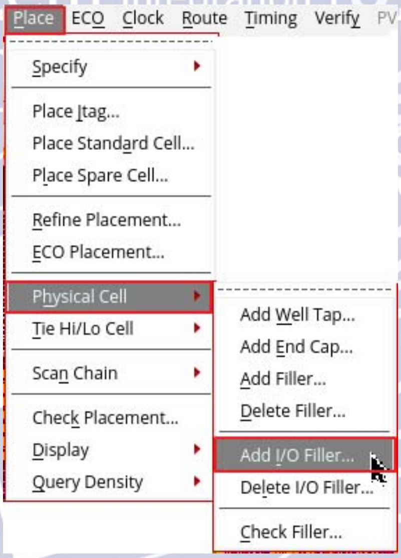
Incremental. Then perform postCTS optimization iteratively until hold timing slack becomes positive.

62. See timing analysis reports in **timingReports/** directory. For a detail path report, see **CHIP_postCTS_all.tarpt.gz** (setup time check) and **CHIP_postCTS_all_hold.tarpt.gz** (hold time check). DRV violations report files: ***.cap.gz**, ***.fanout.gz**, and ***.tran.gz**.

```
%gvim   CHIP_postCTS_all.tarpt.gz
%gvim   CHIP_postCTS_all_hold.tarpt.gz
```

Add I/O PAD Filler

63. In INNOVUS menu, open ***Place -> Physical Cell -> Add I/O Filler***



64. In Add IO Filler Menu:

a. Add PFILL cells

- Name: **PFILL**
- Prefix: **IOFILLER**
- □**Fill Any Gap**
- Side: **Top**
- Click **Apply** button
- Repeat this step for four sides (Top, Bottom, Left, and Right)

b. Add PFILL_9 cells

- Name: **PFILL_9**
- Prefix: **IOFILLER**
- □**Fill Any Gap**
- Side: **Top**
- Click **Apply** button

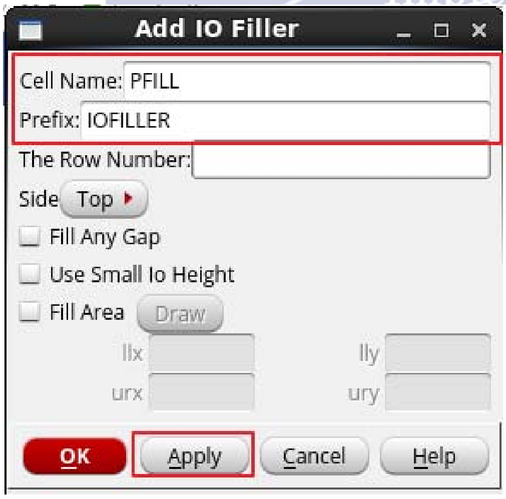
- Repeat this step for four sides (Top, Bottom, Left, and Right)

c. Add PFILL_1 cells

- Name: **PFILL_1**
- Prefix: **IOFILLER**
- □**Fill Any Gap**
- Side: **Top**
- Click **Apply** button
- Repeat this step for four sides (Top, Bottom, Left, and Right)

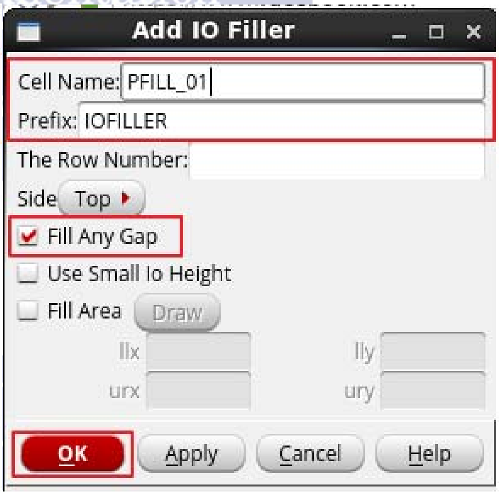
d. Add PFILL_01

- Name: **PFILL_01**
- Prefix: **IOFILLER**
- ☑**Fill Any Gap**
- Side: **Top**
- Click **Apply** button
- Repeat this step for four sides (Top, Bottom, Left, and Right)



Add IO Filler

Cell Name: PFILL
Prefix: IOFILLER
The Row Number:
Side Top
□ Fill Any Gap
□ Use Small Io Height
□ Fill Area



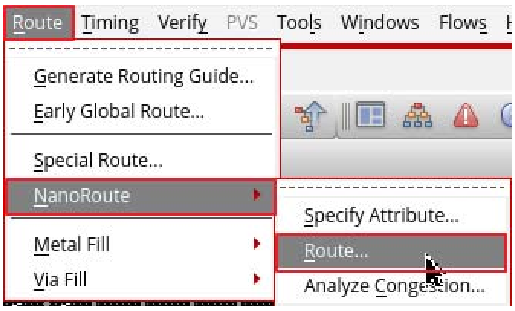
Add IO Filler

Cell Name: PFILL_01
Prefix: IOFILLER
The Row Number:
Side Top
☑ Fill Any Gap
□ Use Small Io Height
□ Fill Area

Note: PAD fillers must be added before the detailed route, or there may have some DRC/LVS violations after a detailed route.

SI-Prevention Detail Route (NanoRoute)

65. In INNOVUS menu, open ***Route -> NanoRoute -> Route***

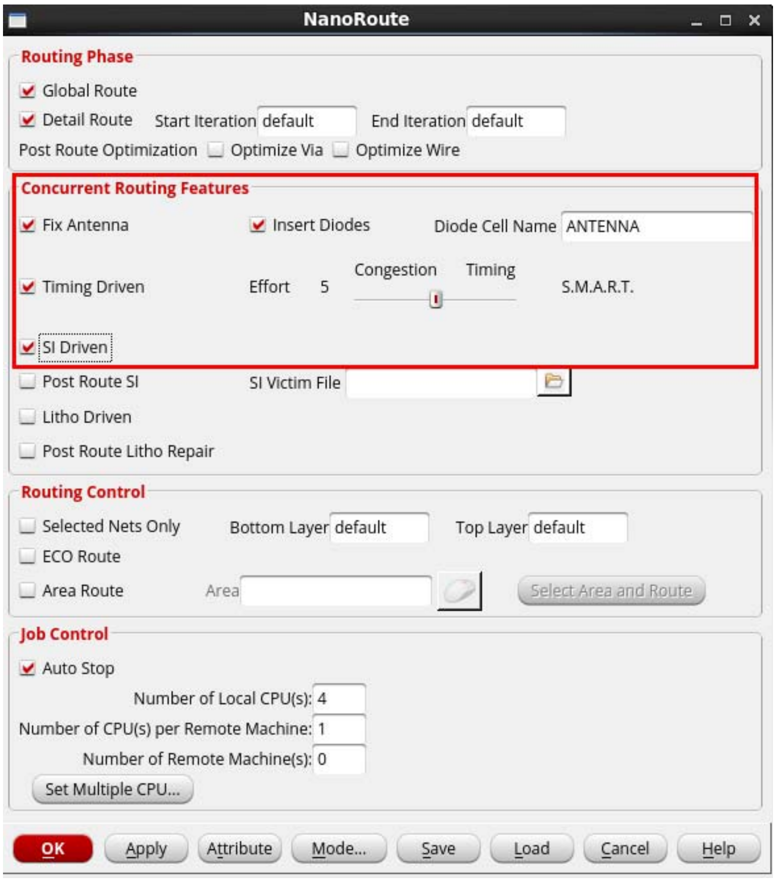


66. Nanoroute can prevent cross talk effects and fix antenna rule violations, and it also routes design to meet timing constraints.

a. Concurrent Routing Features

- ☑ **Fix Antenna** ☑ **Insert Diodes** Diode Cell Name: **ANTENNA**
- ☑ **Timing Driven**
- ☑ **SI Driven**

b. Click **OK** button



NanoRoute

Routing Phase

☑ Global Route

☑ Detail Route Start Iteration default End Iteration default

Post Route Optimization □ Optimize Via □ Optimize Wire

Concurrent Routing Features

☑ Fix Antenna ☑ Insert Diodes Diode Cell Name ANTENNA

☑ Timing Driven Effort 5 Congestion Timing S.M.A.R.T.

☑ SI Driven

□ Post Route SI SI Victim File

□ Litho Driven

□ Post Route Litho Repair

Routing Control

□ Selected Nets Only Bottom Layer default Top Layer default

□ ECO Route

□ Area Route Area Select Area and Route

Job Control

☑ Auto Stop

Number of Local CPU(s): 4

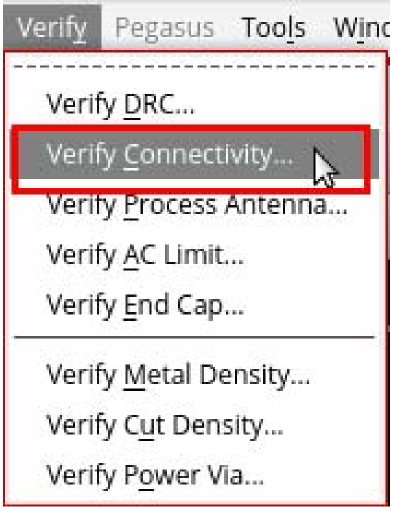
Number of CPU(s) per Remote Machine: 1

Number of Remote Machine(s): 0

Set Multiple CPU...

OK Apply Attribute Mode... Save Load Cancel Help

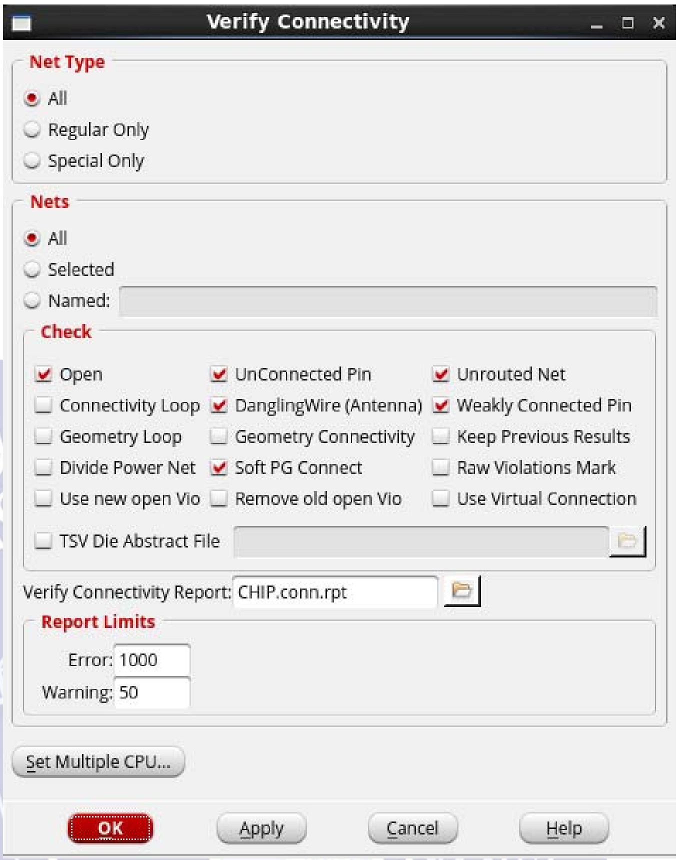
67. In INNOVUS menu, open ***Verify -> Verify Connectivity***



68. Check LVS error after detail routing

a. Click **OK** button

➢ If you see any violations, the routed design is not correct. (LVS error)



Verify Connectivity

Net Type
● All
○ Regular Only
○ Special Only

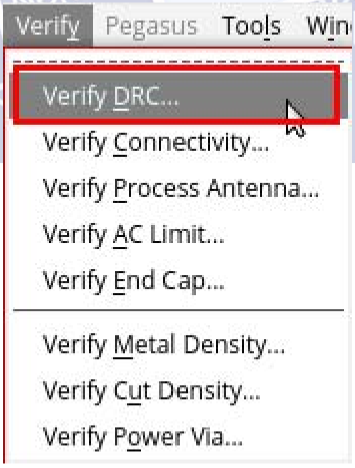
Nets
● All
○ Selected
○ Named:

Check
☑ Open ☑ UnConnected Pin ☑ Unrouted Net
□ Connectivity Loop ☑ DanglingWire (Antenna) ☑ Weakly Connected Pin
□ Geometry Loop □ Geometry Connectivity □ Keep Previous Results
□ Divide Power Net ☑ Soft PG Connect □ Raw Violations Mark
□ Use new open Vio □ Remove old open Vio □ Use Virtual Connection
□ TSV Die Abstract File

Verify Connectivity Report: CHIP.conn.rpt

Report Limits
Error: 1000
Warning: 50

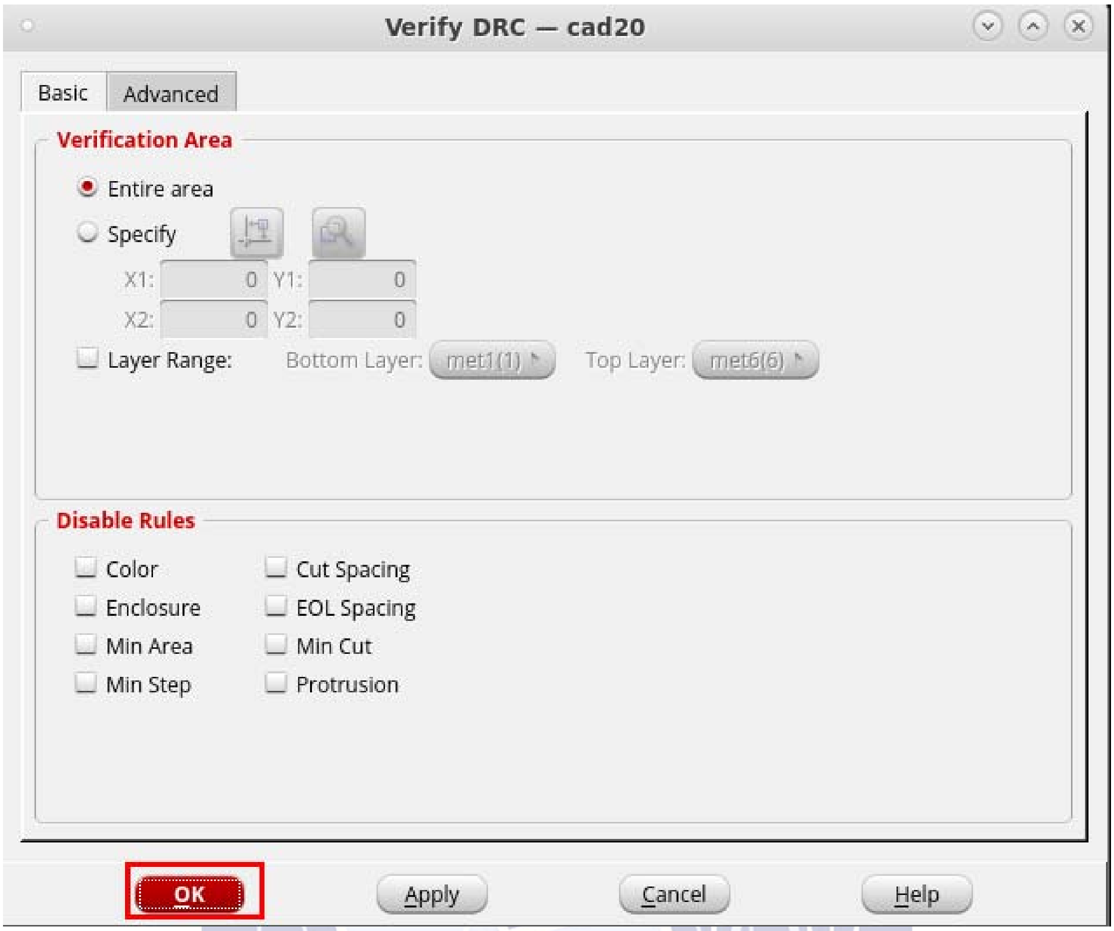
69. In INNOVUS menu, open ***Verify -> Verify DRC***



70. Check detail routing for DRC error

a. Click **OK** button

➢ If you see any violations, the routed design is not correct. (DRC error)



Verify DRC — cad20

Basic | Advanced

Verification Area

● Entire area

○ Specify

X1: 0 Y1: 0

X2: 0 Y2: 0

□ Layer Range: Bottom Layer: met1(1) Top Layer: met6(6)

Disable Rules

□ Color □ Cut Spacing

□ Enclosure □ EOL Spacing

□ Min Area □ Min Cut

□ Min Step □ Protrusion

OK | Apply | Cancel | Help

In-Place Optimization (IPO)

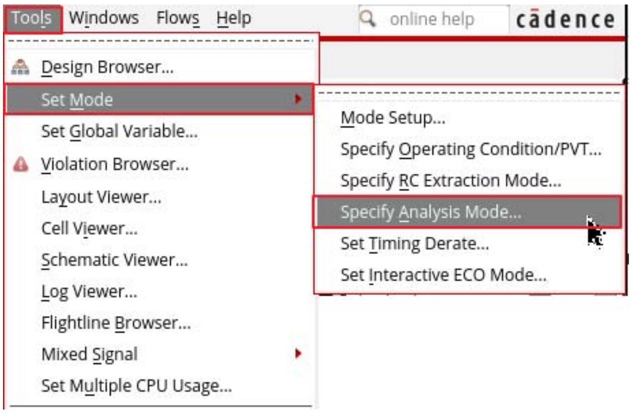
- After Detail Route

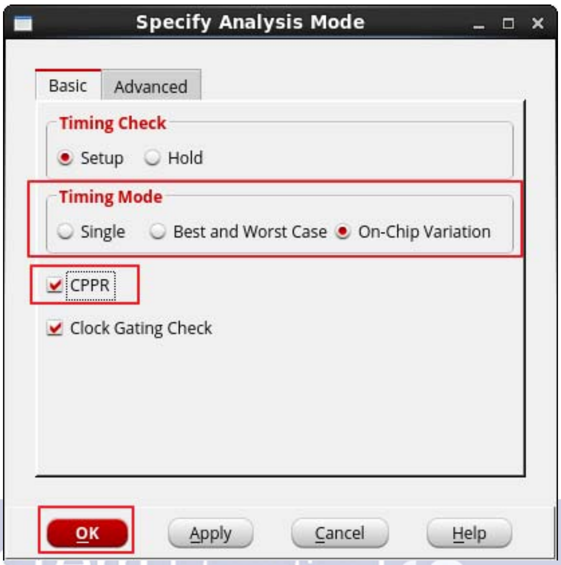
71. In INNOVUS menu, open ***Tools -> Set Mode -> Specify Analysis Mode***

a. Change Timing Mode to **On-Chip Variations**

b. ☑ **CPPR**

c. Click **OK** button





Specify Analysis Mode

Basic | Advanced

Timing Check: ● Setup ○ Hold

Timing Mode: ○ Single ○ Best and Worst Case ● On-Chip Variation

☑ CPPR

☑ Clock Gating Check

72. In INNOVUS menu, open ***Timing -> Report Timing***

73. INNOVUS extracts RC, estimates the interconnection RC effects and then perform delay calculation for timing analysis

    a. Design Stage: ⊙ **Post-Route**

    b. Analysis Type: ☑ **Setup**

    c. Click **OK** button

74. After detail routing with NanoRoute, SI aware analysis (glitch analysis) is enabled, and further timing optimization is performed to meet setup timing constraints. If there is negative setup timing slack or DRVs. Open ***ECO -> Optimize Design*** in the INNOVUS menu

75. Perform post-Route IPO

    a. Design Stage: ⊙ **Post-Route**

    b. Optimization Type

    - ➢ ☑ **Setup**
    - ➢ ⊙ **Design Rule Violations**

      ☑ **Max Cap** ☑ **Max Tran** ☑ **Max Fanout**

    c. ☑ **Include SI**

    d. Click **OK** button

    e. If setup timing slack is still negative, change Optimization type to ⊙ **Incremental**. Then perform postRoute optimization iteratively until setup timing slack becomes positive.

76. Verify if the hold time constraint is satisfied or not. Open ***Timing -> Report Timing*** in INNOVUS menu

    a. Design Stage: ⊙ **Post-Route**

    b. Analysis Type: ☑ **Hold**

    c. Click **OK** button

77. If hold time slack is negative, open ***ECO -> Optimize Design*** in the INNOVUS menu

78. Perform post-Route IPO for hold time fixing

   a. Design Stage: ⦿ **Post-Route**

   ➢ Optimization Type:

   ➢ ☑ **Hold**

   ➢ ⦿ **Design Rule Violations**

   ☑ **Max Cap** ☑ **Max Tran** ☑ **Max Fanout**

   b. ☑ **Include SI**

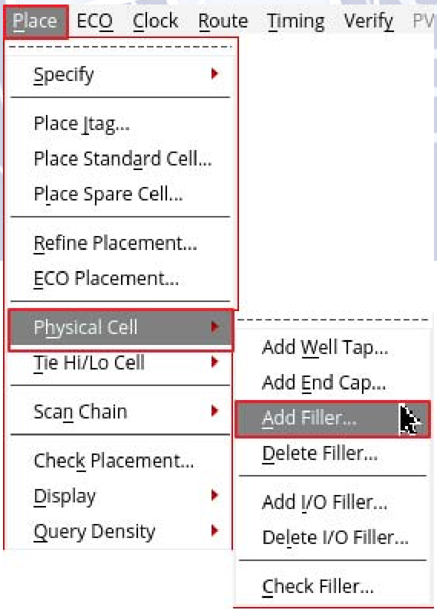
   c. Click **OK** button

   See timing reports in **timingReports/** directory. For a detail path report, see **CHIP_postRoute_all.tarpt.gz** (setup time check) and **CHIP_postRoute_all_hold.tarpt.gz** (hold time check). DRV violations report files: ***.cap.gz**, ***.fanout.gz**, and ***.tran.gz**.

```
%gvim   CHIP_postRoute_all.tarpt.gz
%gvim   CHIP_postRoute_all_hold.tarpt.gz
```

Add CORE Filler Cells

79. In INNOVUS menu, open ***Place -> Physical Cell -> Add Filler***



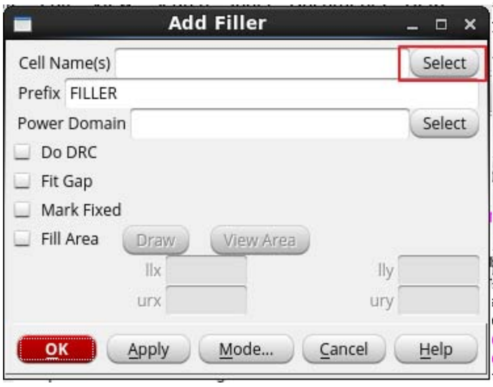
80. Core fillers can reduce well resistance and avoid CMOS latch-up problem, in **Add Filler** menu:

   a. Click **Select** button in Cell Names(s)

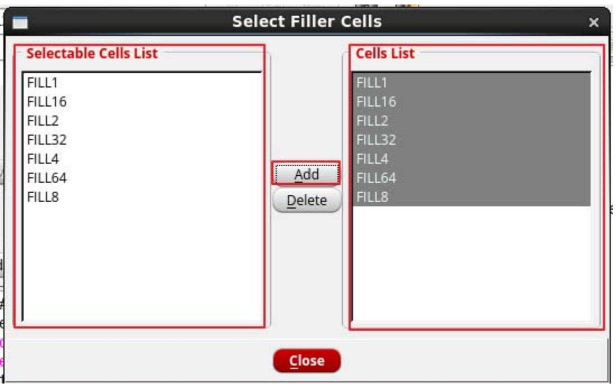
   b. In **Select Filler Cells** menu, select all core filler cells, then click **Add** button to add these cells to the Cell Lists.

c. Click **Close** button and return to **Add Filler** menu.

d. Click **OK** button to add core fillers.



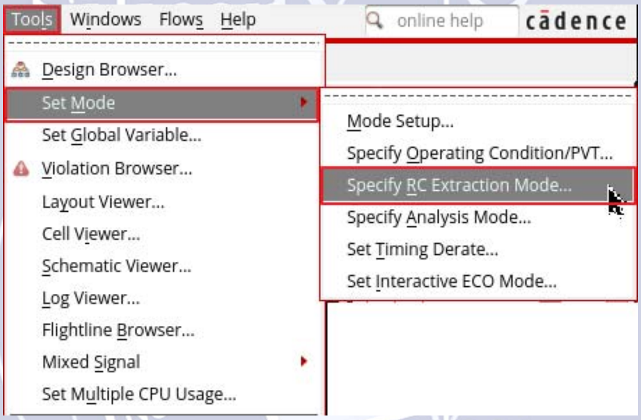
Prefix FILLER



Selectable Cells List	Cells List
FILL1	FILL1
FILL16	FILL16
FILL2	FILL2
FILL32	FILL32
FILL4	FILL4
FILL64	FILL64
FILL8	FILL8

Stream Out and Write Out Netlist

81. In INNOVUS menu, open ***Tools -> Set Mode -> Specify RC Extraction Mode***



a. ⦿**PostRoute**

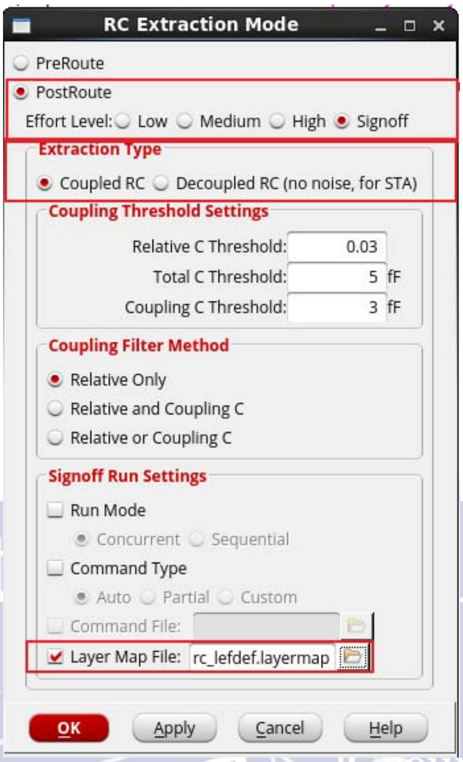
b. Effort Level: ⦿ **Signoff**

c. Extraction Type

⦿ **Coupled RC** ○ Decoupled RC (no noise, for STA)

d. ☑ **Layer Map File : 00LIB/qrc_lefdef.layermap**

e. Click **OK** button


RC Extraction Mode

○ PreRoute

● PostRoute

Effort Level: ○ Low ○ Medium ○ High ● Signoff

Extraction Type

● Coupled RC ○ Decoupled RC (no noise, for STA)

Coupling Threshold Settings

Relative C Threshold: 0.03

Total C Threshold: 5 fF

Coupling C Threshold: 3 fF

Coupling Filter Method

● Relative Only

○ Relative and Coupling C

○ Relative or Coupling C

Signoff Run Settings

□ Run Mode

● Concurrent ○ Sequential

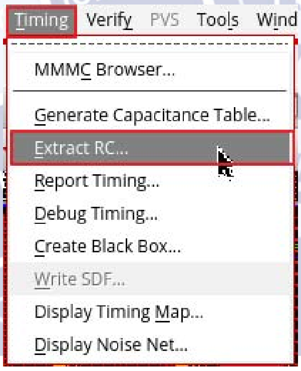
□ Command Type

● Auto ○ Partial ○ Custom

□ Command File:

☑ Layer Map File: rc_lefdef.layermap

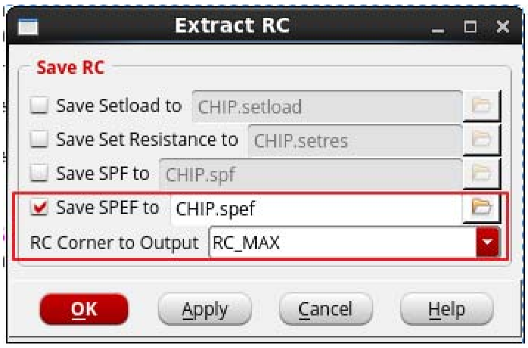
82. Perform 3D Full-Chip RC extraction. In INNOVUS menu, open ***Timing -> Extract RC***



a. ☑ Save SPEF to: **CHIP.spef**

b. RC Corner to Output: **RC_MAX**

c. Click **OK** button


Extract RC

Save RC

□ Save Setload to CHIP.setload

□ Save Set Resistance to CHIP.setres

□ Save SPF to CHIP.spf

☑ Save SPEF to CHIP.spef

RC Corner to Output RC_MAX

83. In INNOVUS command prompt, execute the following command to write out the SDF file for post-layout gate-level simulation:

```
innovus> write_sdf -edges check_edge -max_view CHECK_SETUP_TIME \
-min_view CHECK_HOLD_TIME -temperature 0.0::125 \
-voltage 1.98::1.62    CHIP.sdf
```

Note: the specified view names to ***max_view*** and ***min_view*** options should be defined in your MMMC configuration. Also, you can copy-paste this command from s11.cmd.

84. Save design netlist **CHIP.v** for post-layout gate-level simulation:

a. In INNOVUS command prompt, execute the following command:

```
innovus> saveNetlist    CHIP.v
```

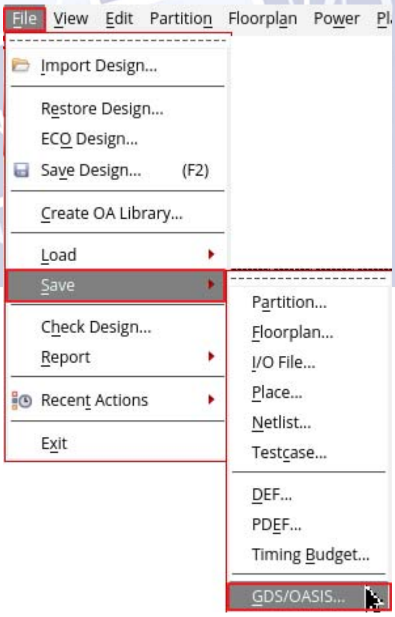
85. Save design netlist **CHIP_LVS.v** for Calibre LVS check only:

a. In INNOVUS command prompt, execute the following command:

```
innovus> saveNetlist -excludeLeafCell -includePhysicalInst CHIP_LVS.v
```

86. Save the routed design in GDSII format: **CHIP.gds** for Calibre DRC check:

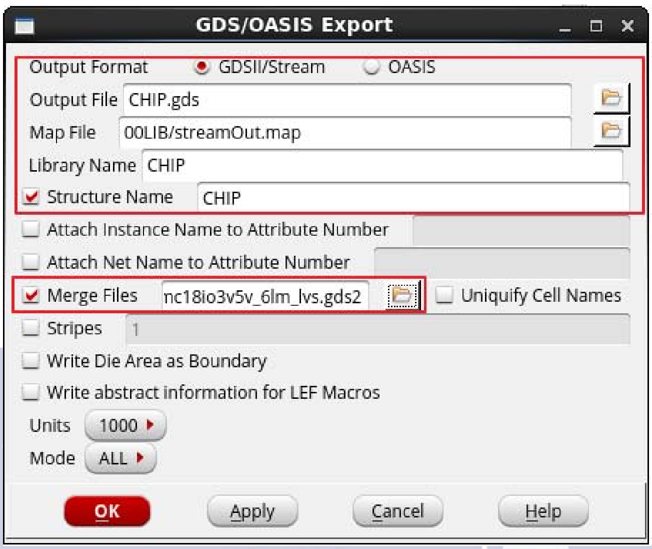
a. In INNOVUS menu, open ***File -> Save -> GDS/OASIS***



b. GDSII/OASIS Export Options

- ➢ Output Format: ⦿ GDSII/Stream
- ➢ Output File: **CHIP.gds**
- ➢ Map File: **00LIB/streamOut.map**
- ➢ Library Name: **CHIP**
- ➢ ☑ Structure Name: **CHIP**

- ☑Merge Files: **00LIB/umc18_lvs.gds2**
  00LIB/umc18io3v5v_6lm_lvs.gds2
- Click **OK** button



GDS/OASIS Export

Output Format ● GDSII/Stream ○ OASIS

Output File CHIP.gds

Map File 00LIB/streamOut.map

Library Name CHIP

☑ Structure Name CHIP

□ Attach Instance Name to Attribute Number

□ Attach Net Name to Attribute Number

☑ Merge Files nc18io3v5v_6lm_lvs.gds2 □ Uniquify Cell Names

□ Stripes 1

□ Write Die Area as Boundary

□ Write abstract information for LEF Macros

Units 1000

Mode ALL

Post-Layout Gate-Level Simulation

87. You need to open another terminal window to run this step.
88. Change to directory **Netlist**
89. Perform post-layout gate-level simulation of CHIP.v

    ```
    % ncverilog   -f   run.f
    ```

    You can also change the cycle time in TEST_CHIP.v to see the maximum clock rate of this design. (cycle time is defined as Tclk)

90. After post-layout gate-level simulation, **CHIP.tcf** is generated in this directory. This file will be used for power rail analysis.

Power Rail Analysis

91. **Repeat step 53 to step 59** in **command line mode** **Power Rail Analysis** section to perform power rail analysis with Voltus. Then you should display IR drop analysis result in INNOVUS.

Appendix A: IO Pad assignment

Syntax of CHIP.io:

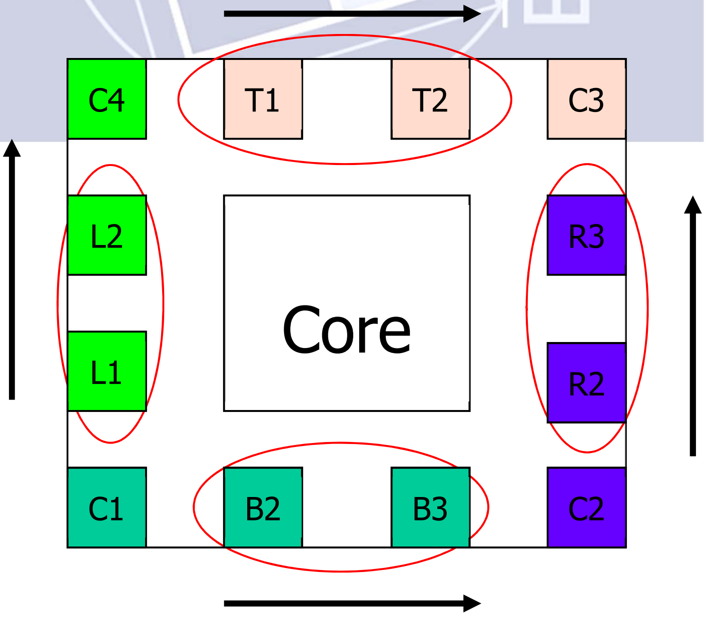
```
Version: 2
Pad: B2            S
Pad: B3            S

Pad: R2            E
Pad: R3            E         PFILL(If this pad is a pad filler)

Pad: T1            N
Pad: T2            N

Pad: L1            W
Pad: L2            W

Pad: C1            SW        PCORNER
Pad: C2            SE        PCORNER
Pad: C3            NE        PCORNER
Pad: C4            NW        PCORNER
```


C4
T1
T2
C3
L2
R3
Core
L1
R2
C1
B2
B3
C2