

# Training Course of SOC Encounter

---

REF:

- CIC Training Manual – Cell-Based IC Physical Design and Verification with SOC Encounter, July, 2006
- CIC Training Manual – Mixed-Signal IC Design Concepts, July, 2007

Speaker: T. –W. Tseng

# Outline

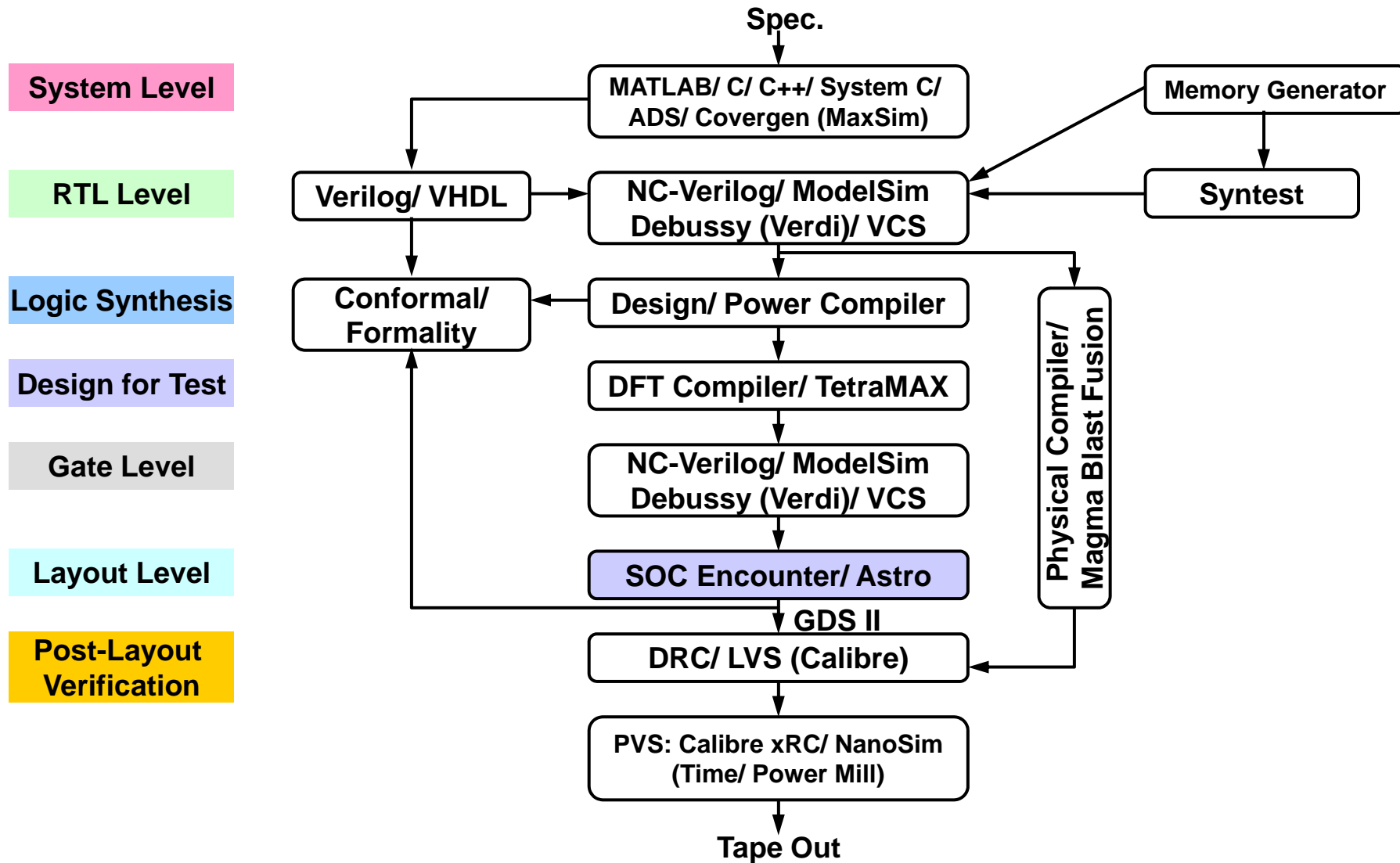
---

- ☐ Basic Concept of the Placement & Routing
- ☐ Auto Place and Route Using SOC Encounter
- ☐ Hard Block Abstraction Using Abstract Generator
- ☐ LAB

---

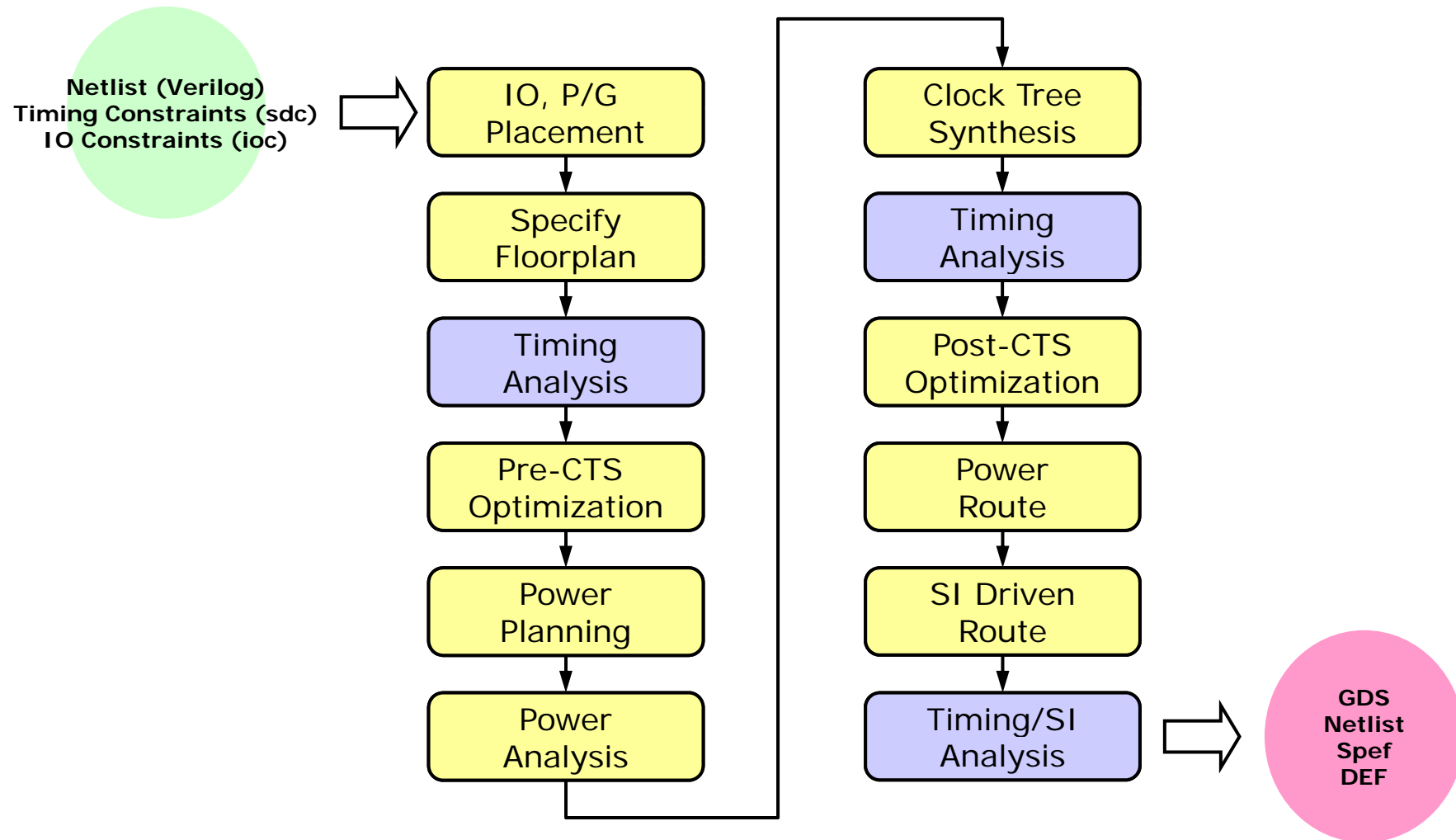
# Basic Concept of the Placement & Routing

# Cell-Based Design Flow



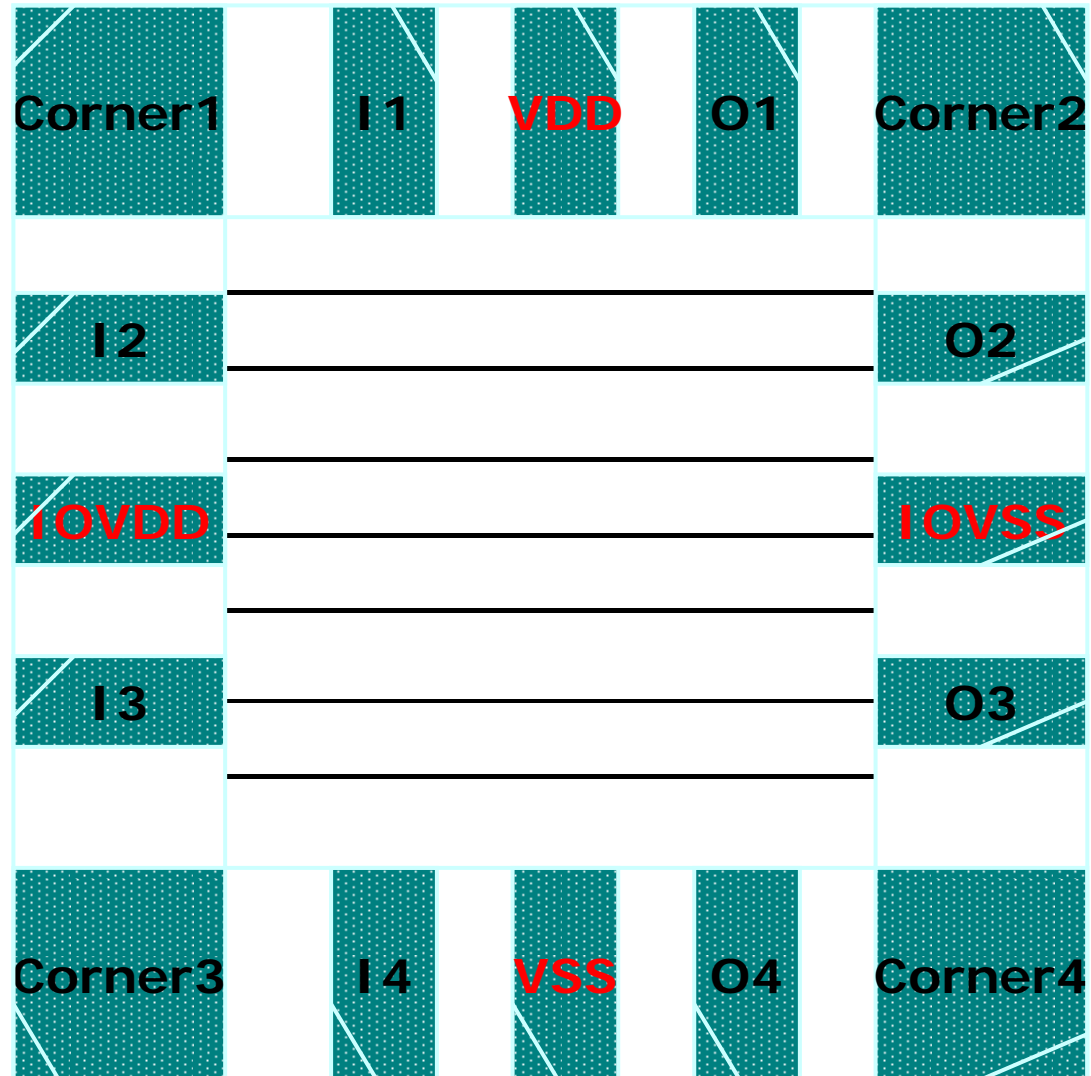


# SOC Encounter P&R Flow



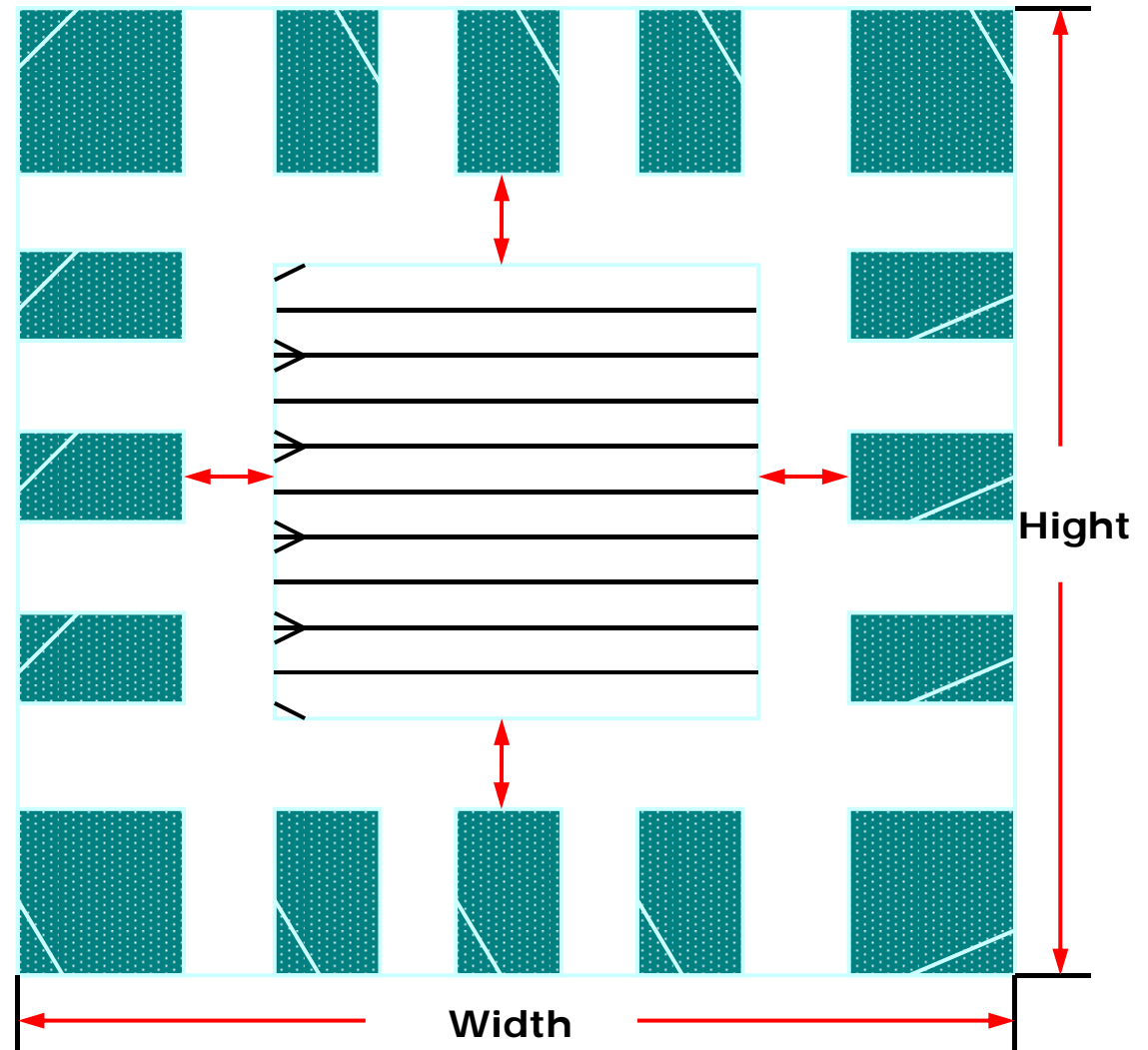
# IO, P/G Placement

- Determine the positions of the PADs
  - Functional IO PAD
  - Power/Ground PAD
  - Corner PAD
    - Just for the connection of PAD power rings



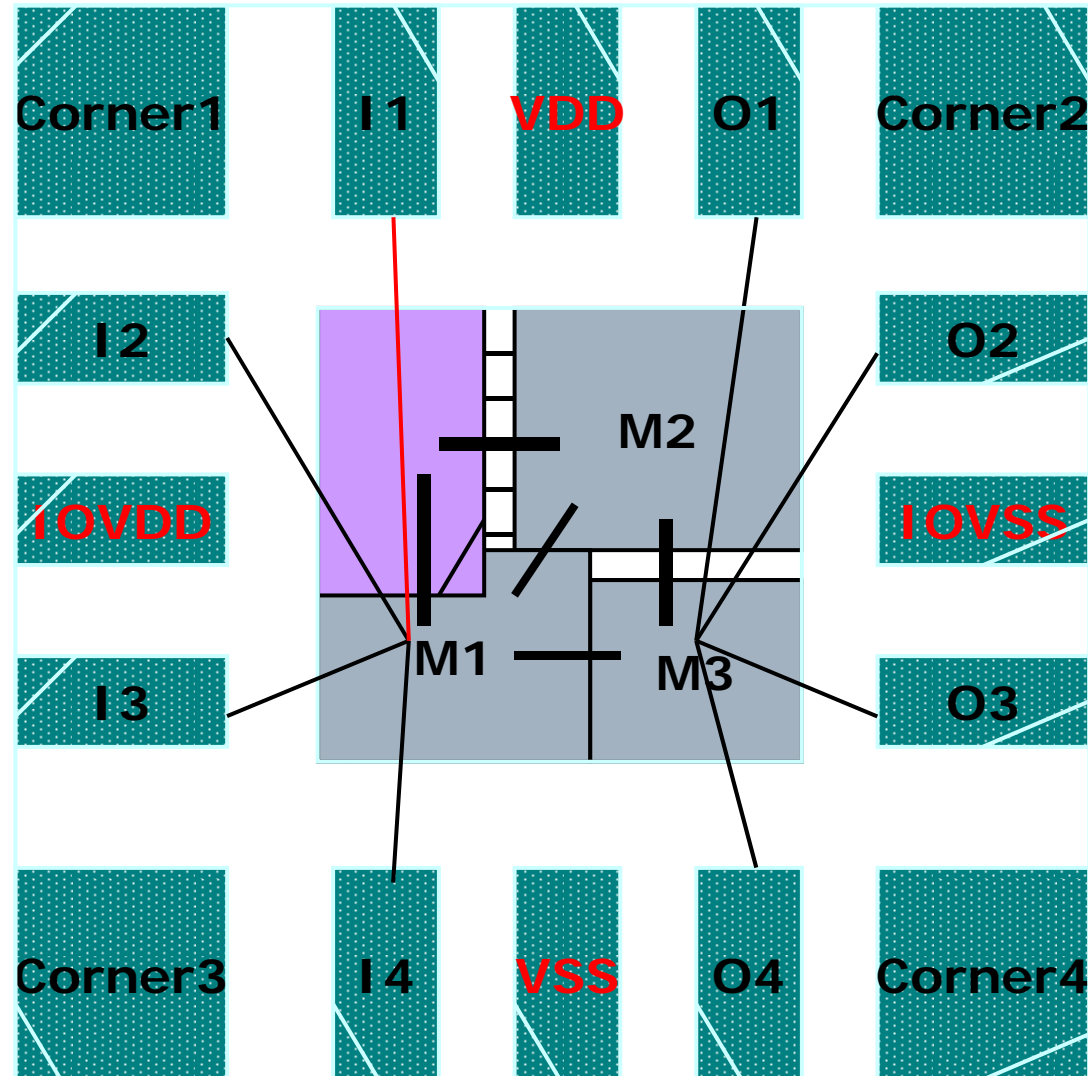
# Specify Floorplan

- Determine the aspect ratio of the Core and the gap between the PAD and Core
  - The Core Utilization is determined in this step
  - The final CHIP area is almost determined in this step



# Floorplan

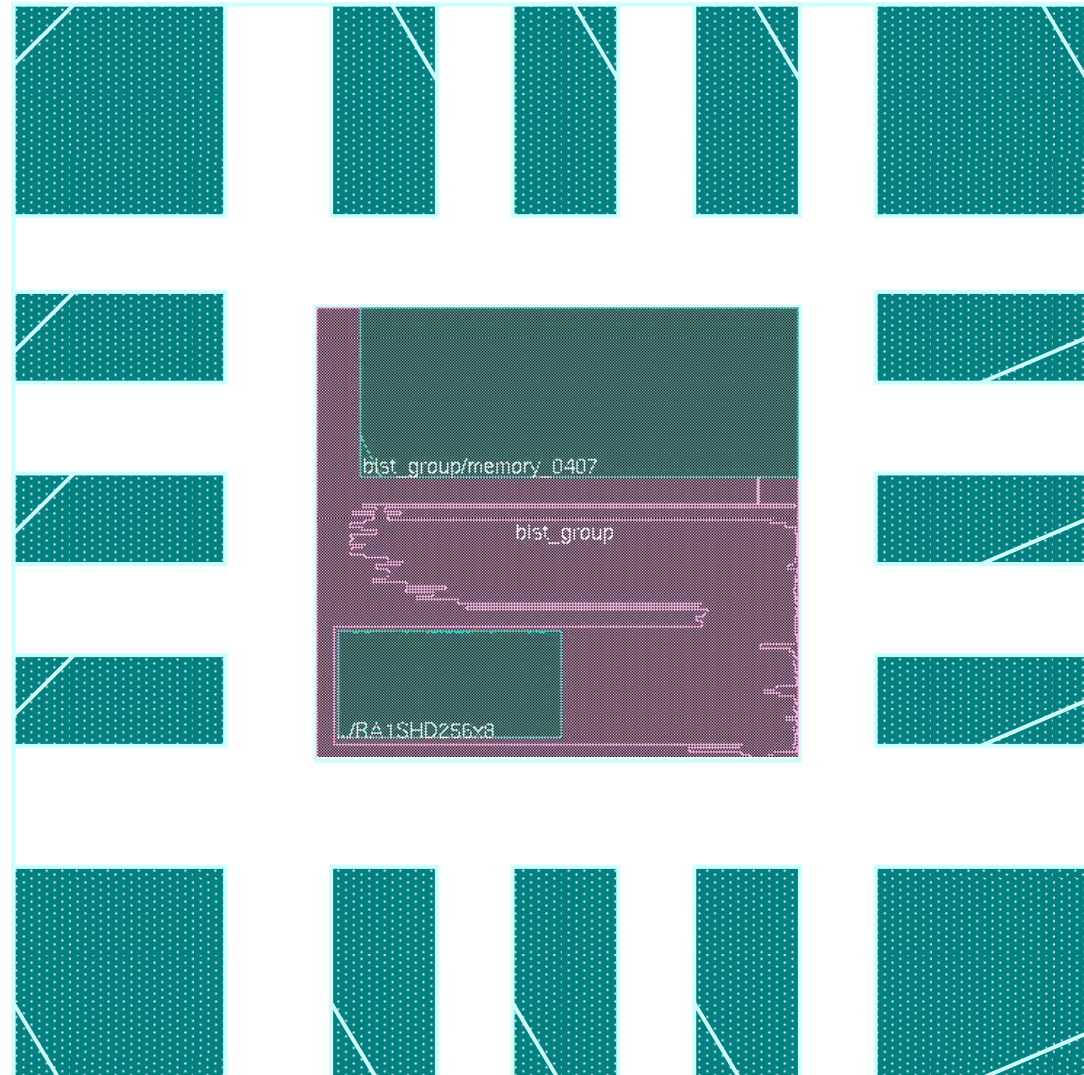
- Determine the related positions of Hard Blocks
- The performance is highly affected



# Amoeba Placement

---

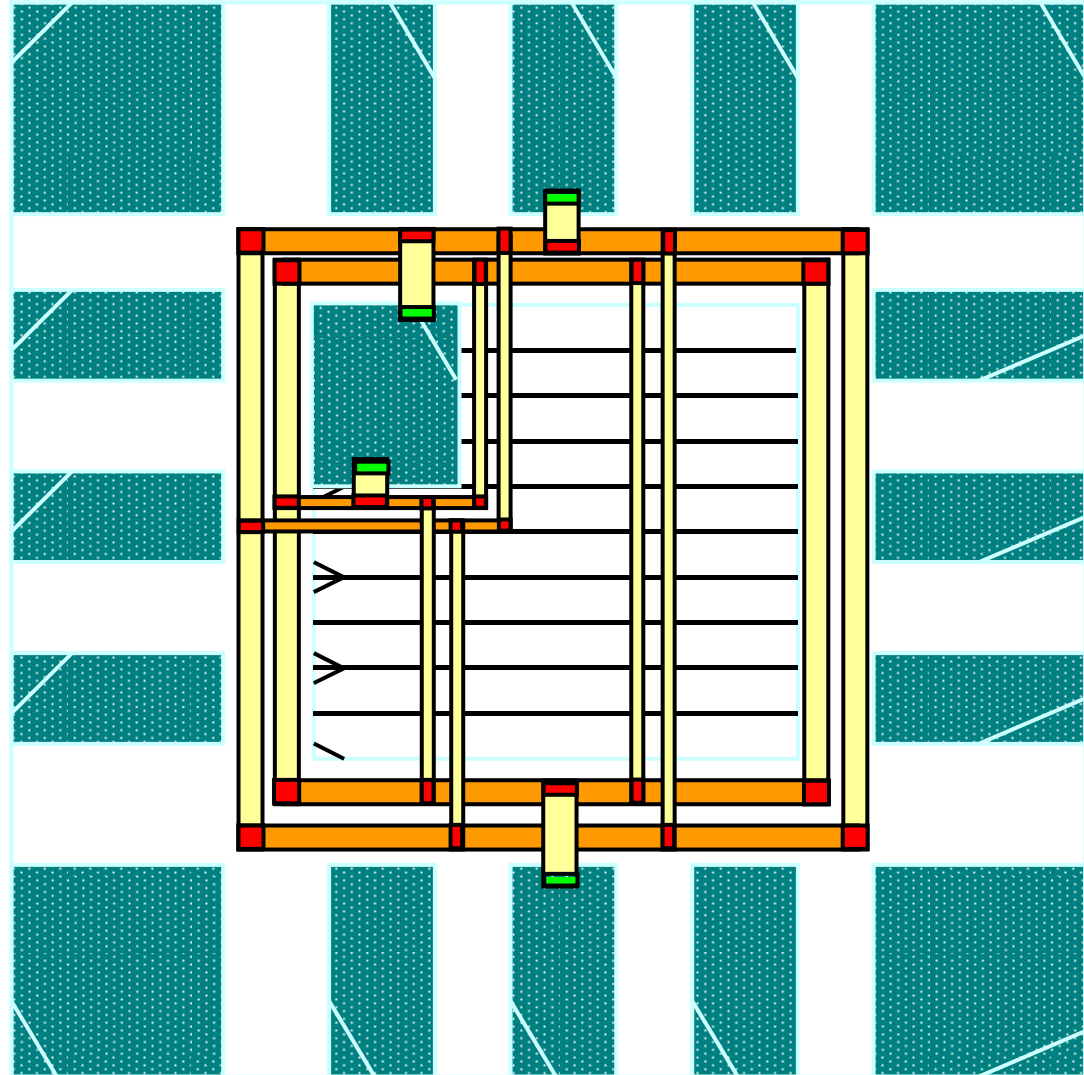
- Observe the result of cells and Hard Blocks placement



# Power Planning

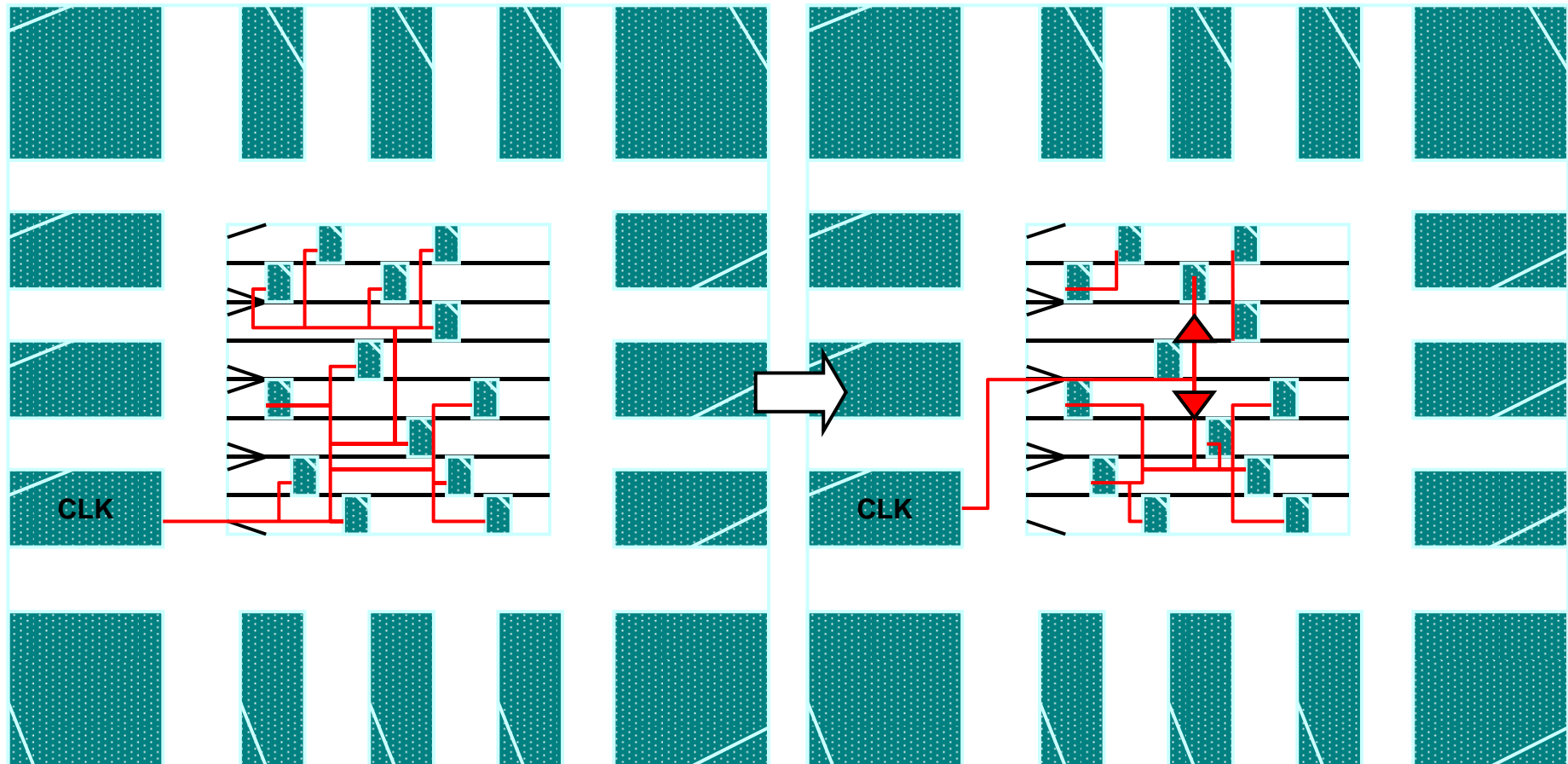
---

- Plan the power ring & power stripe
- IR-drop consideration



# Clock Tree Synthesis

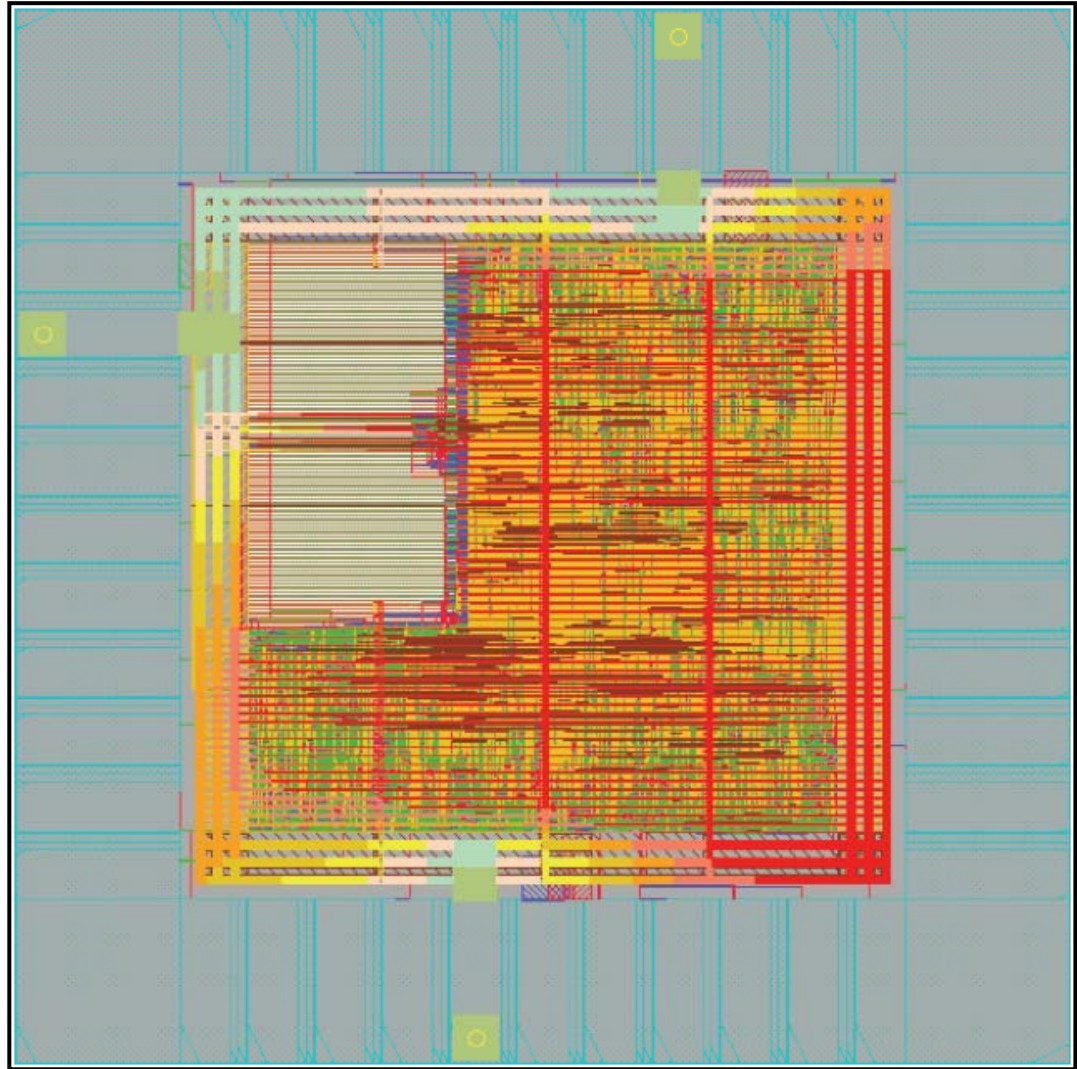
---



# Power Analysis

---

- IR-drop & electron migration

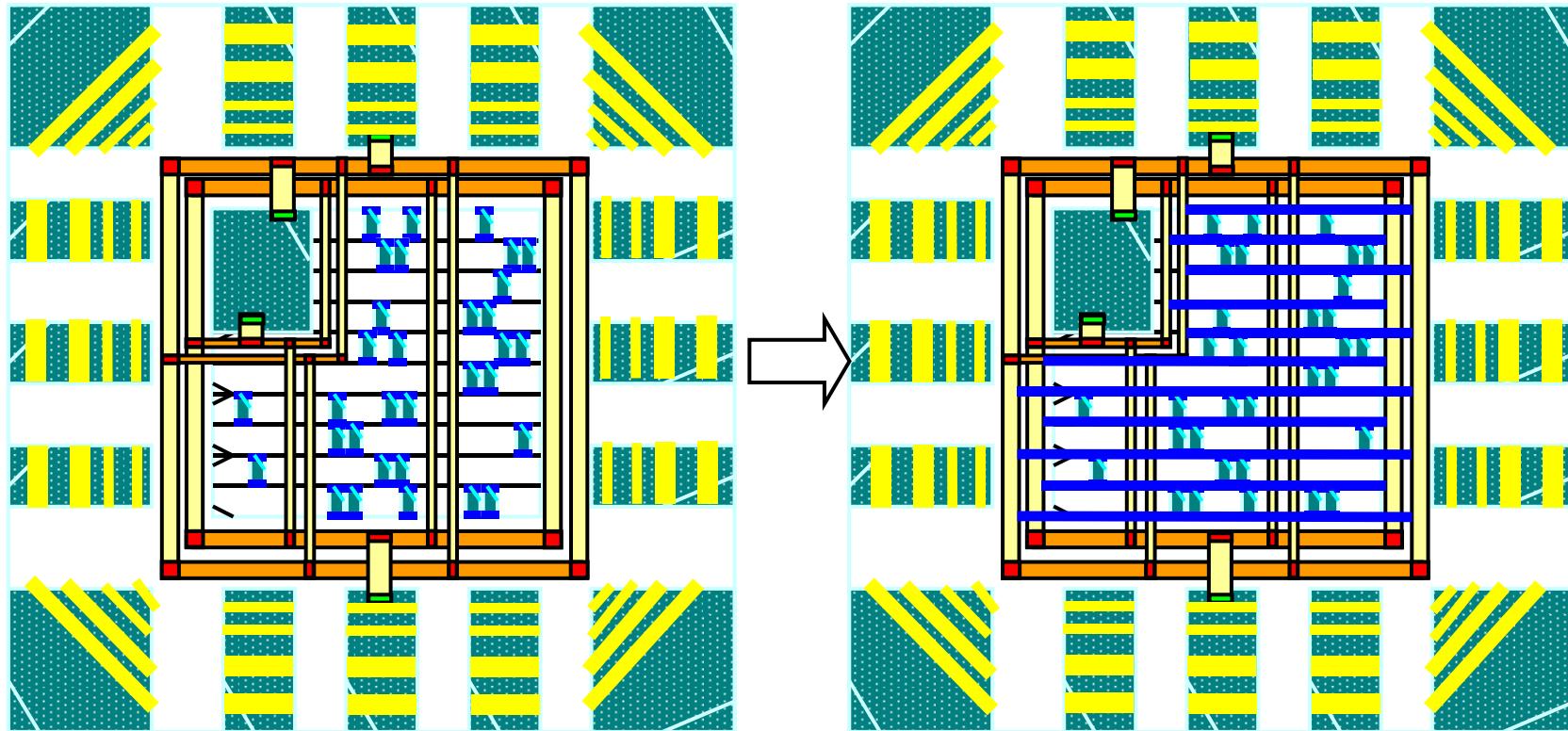




# Power Route

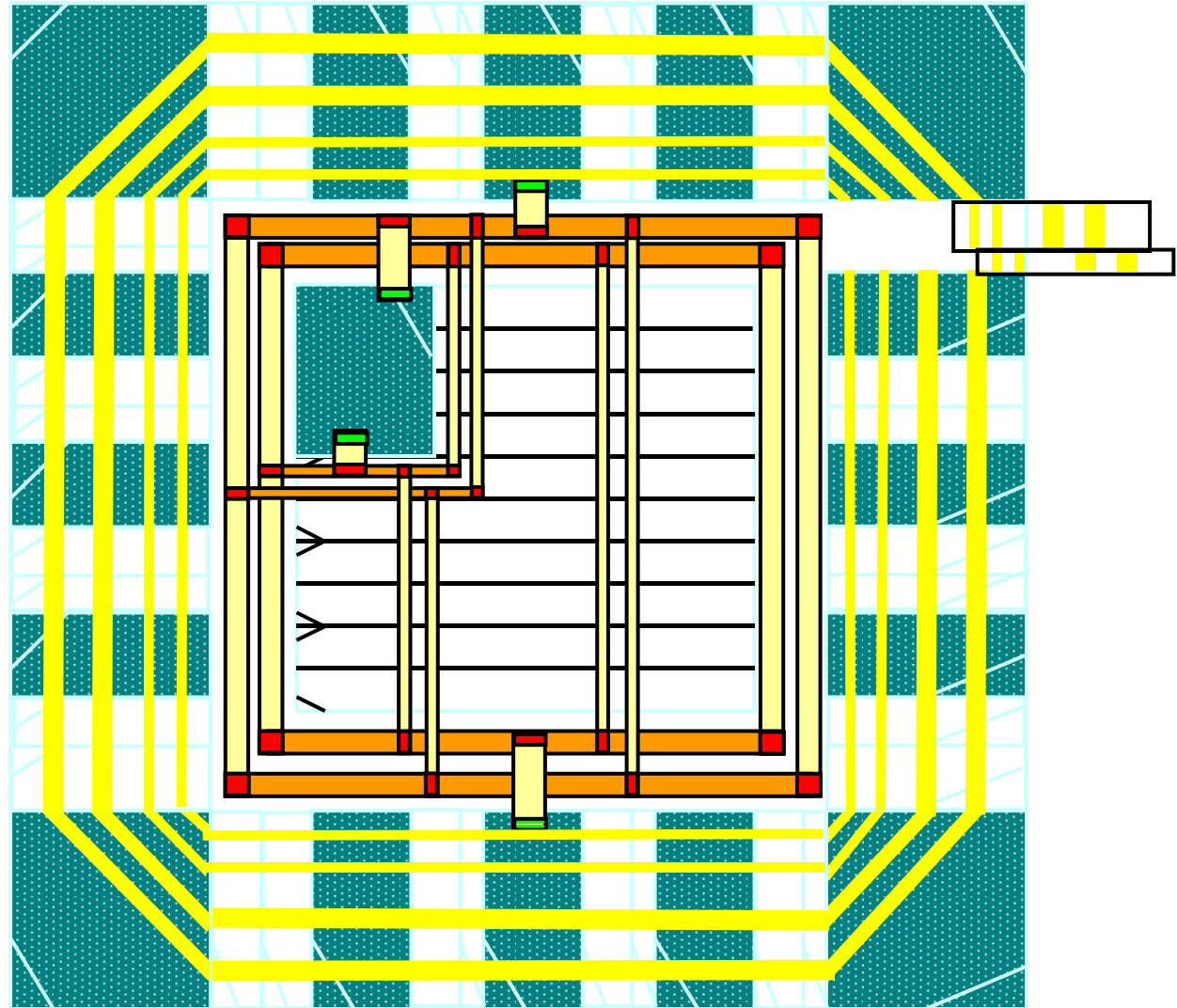
---

- Connect the power pins of standard cells to the global power lines



# Add IO Filler

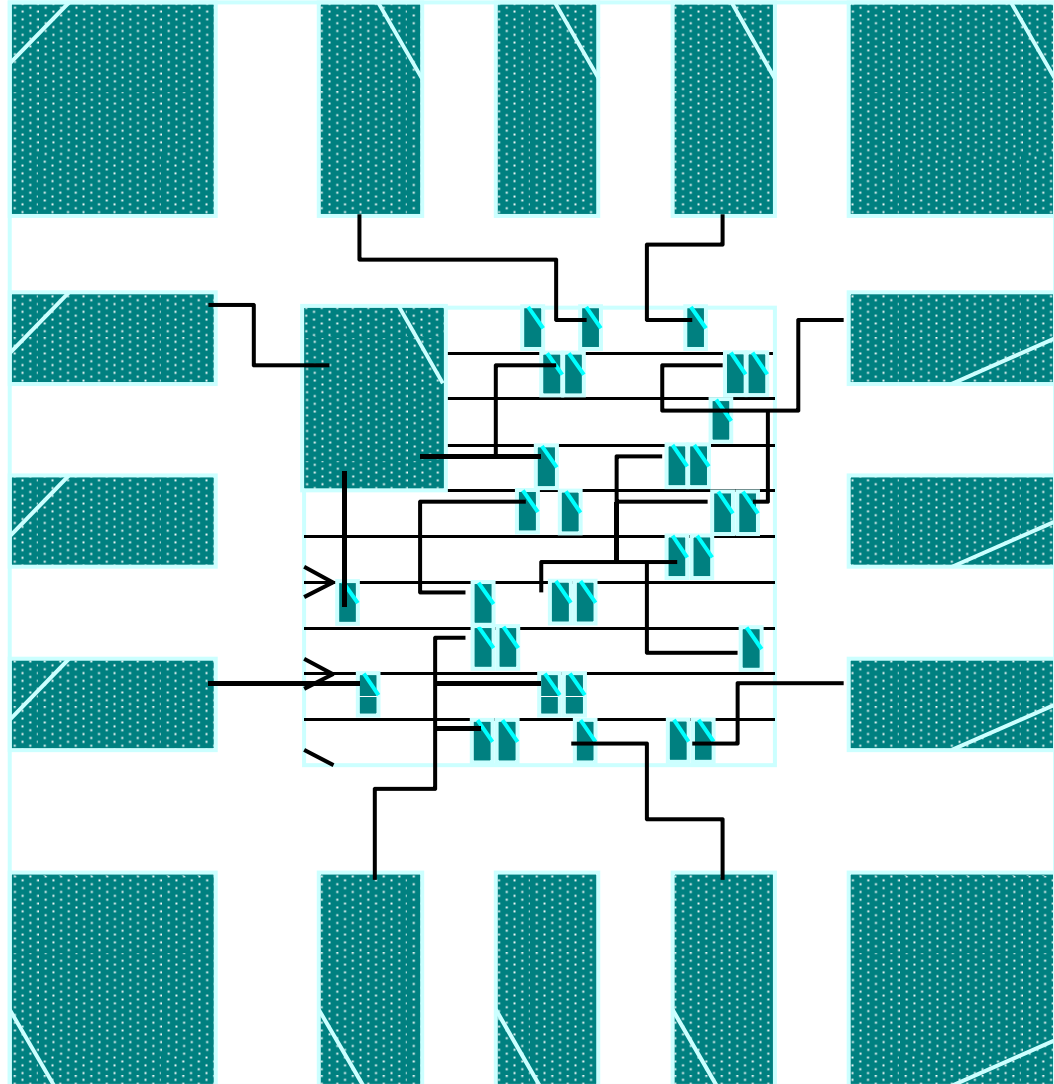
- Fill the gap between PADs
- Connect the PAD power rings



# Routing

---

- Construct the final interconnections



# Prepare Data

---

## ☐ Library

- Physical Library (LEF)
  - ☐ Information of technology, standard cells, Hard Blocks, and APR
- Timing Library (LIB)
  - ☐ Timing information of the standard cells and Hard Blocks
- Capacitance Table
  - ☐ For more accurate RC analysis
- Celtic Library
  - ☐ For crosstalk analysis
- FireIce/Voltage Storm Library
  - ☐ For RC extraction and power analysis





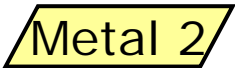
**Not Necessary !**

## ☐ User Data

- Gate-Level Netlist (Verilog)
- SDC Constraint (\*.sdc)
- IO Constraint (\*.ioc)

# LEF Format – Process Technology

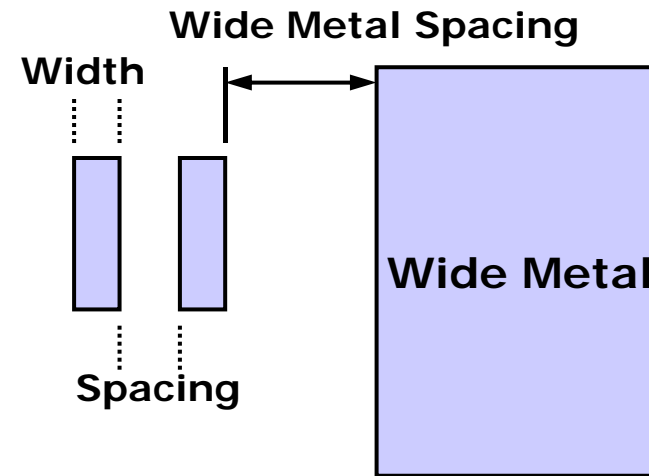
---

Layers	Design Rule	Parasitic
 POLY	Net Width	Resistance
 Contact	Net Spacing	Capacitance
 Metal 1	Area	
 Via1	Enclosure	
 Metal 2	Wide Metal Slot	
	Antenna	
	Current Density	

# LEF Format – Process Technology: Layer Define

---

```
Layer Metal1
  TYPE ROUTING;
  WIDTH 0.28;
  MAXWIDTH 8;
  AREA 0.202;
  SPACING 0.28;
  SPACING 0.6 RANGE 10.0 10000.0;
  PITCH 0.66;
  DIRECTION VERTICAL;
  THICKNESS 0.26;
  ANTENNACUMDIFFAREARATIO 5496;
  RESISTANCE RPERSQ 1.0e-01;
  CAPACITANCE CPERSQDIST 1.11e-04;
  EDGECAPACITANCE 9.1e-05;
END Metal1
```



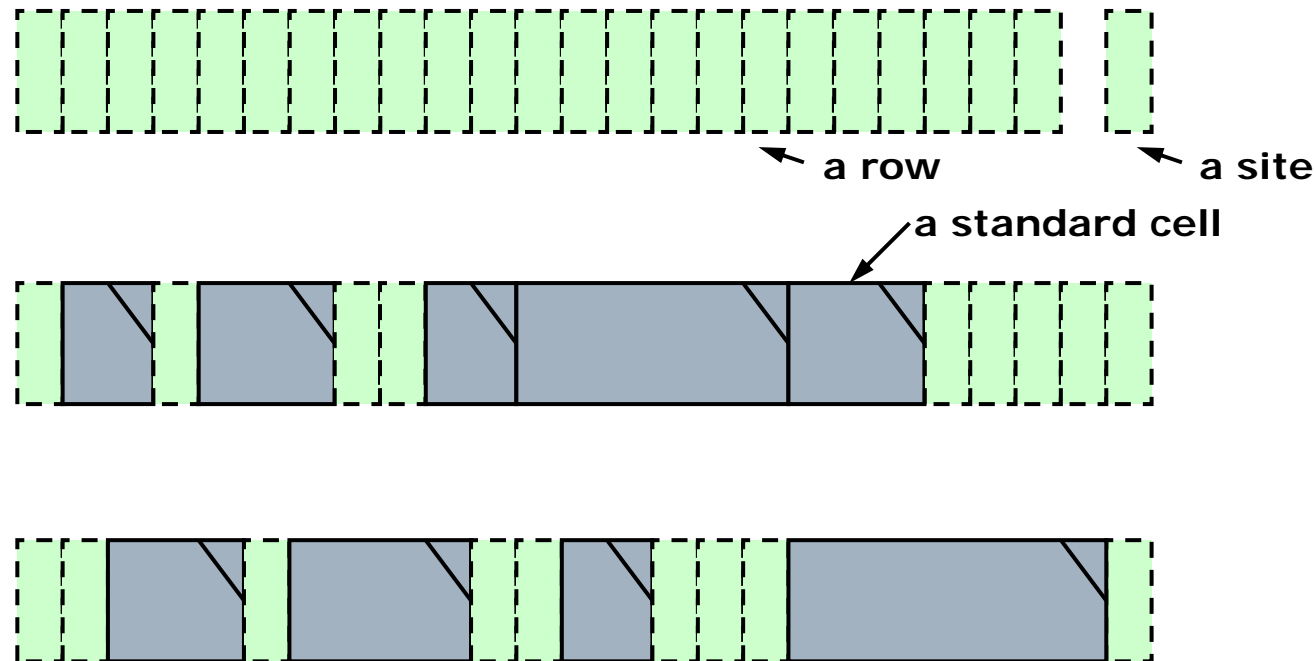
# LEF Format – APR Technology

---

- ☐ Unit
- ☐ Site
- ☐ Routing Pitch
- ☐ Default Direction
- ☐ Via Rule

# LEF Format – APR Technology: Site

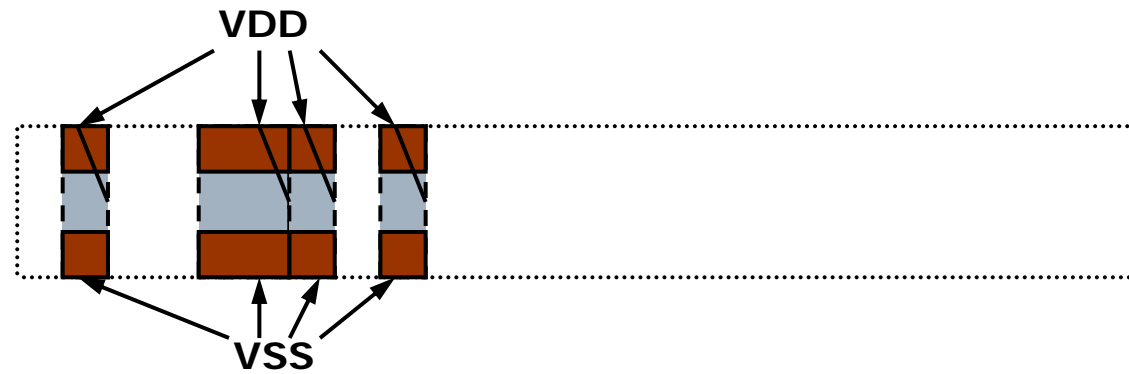
- ❑ The placement site gives the placement grid of a family of macros





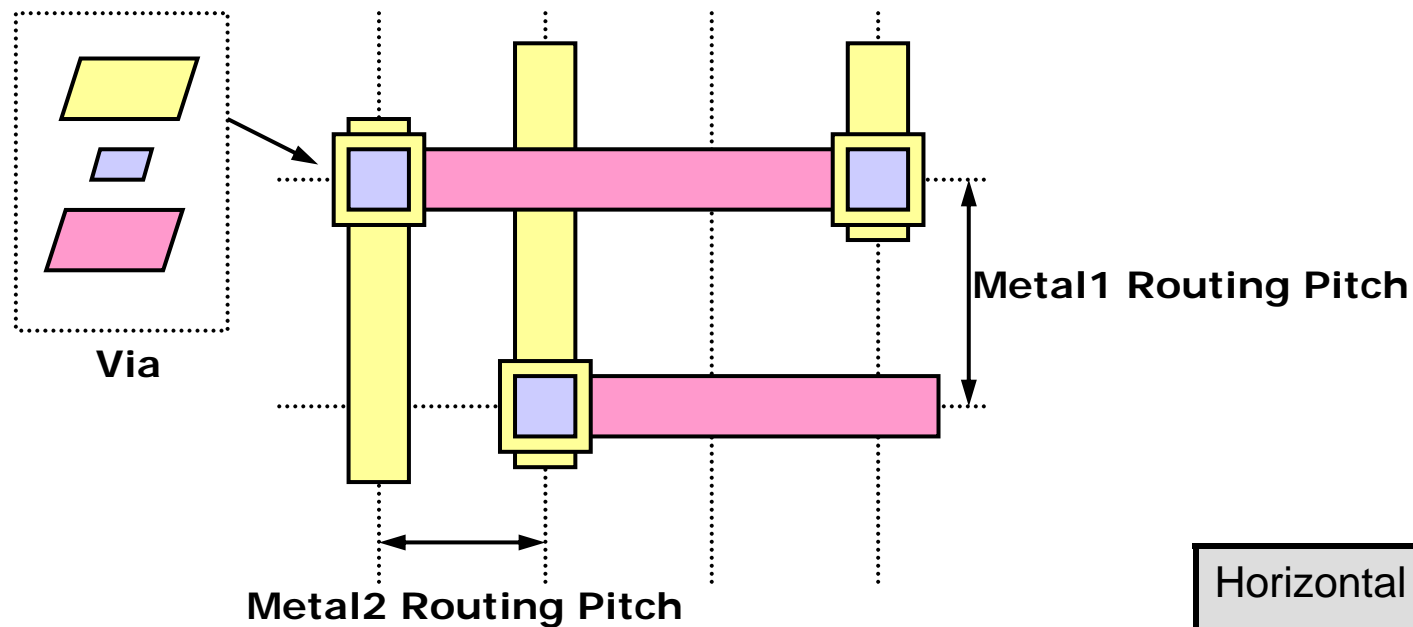
# Row Based PR

---



# LEF Format – APR Technology: Routing Pitch, Default Direction

---



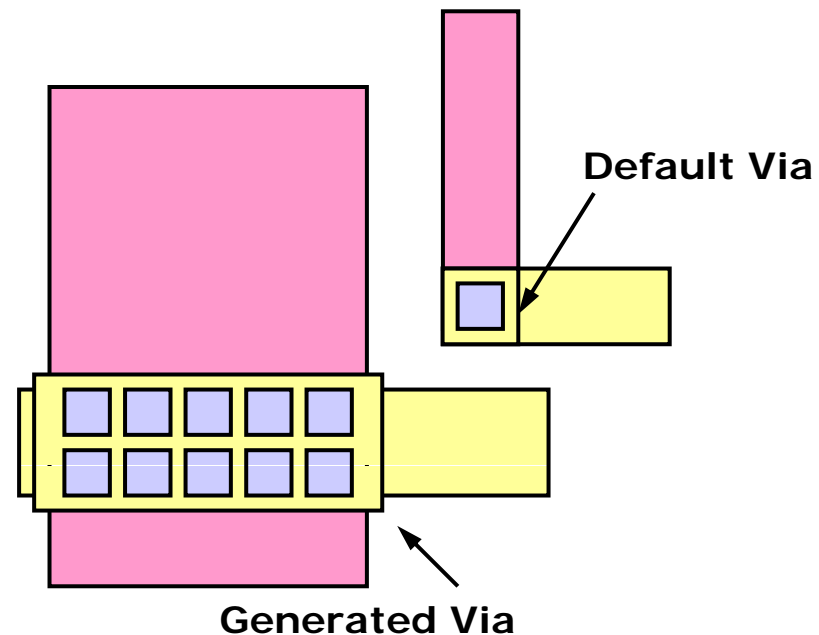
Horizontal Routing	Vertical Routing
Metal1	Metal2
Metal3	Metal4
Metal5	Metal6

# LEF Format – APR Technology: Via Generation

---

- ❑ To connect the wide metal, a via array is generated to reduce the via resistance
- ❑ Formulas for generating via arrays are defined

```
Layer Metal1
  Direction HORIZONTAL
  OVERHANG 0.2
Layer Metal2
  Direction VERTICAL
  OVERHANG 0.2
Layer Via1
  RECT -0.14 -0.14 0.14 0.14
  SPACING 0.56 BY 0.56
```

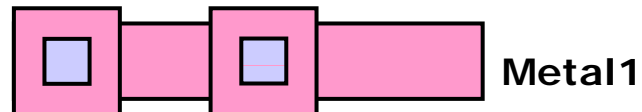


# LEF Format – APR Technology: Same Net Spacing

---

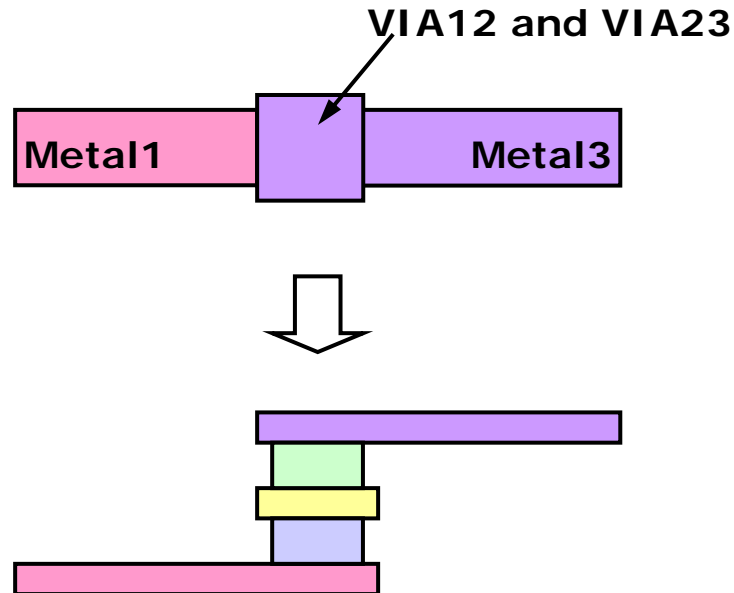
## SPACING

```
SAMENET Metal1 Metal1 0.23;  
SAMENET Metal2 Metal2 0.28 STACK;  
SAMENET Metal3 Metal3 0.28;  
SAMENET VIA12 VIA12 0.26;  
SAMENET VIA23 VIA23 0.26;  
SAMENET VIA12 VIA23 0.0 STACK;  
END SPACING
```



0.23

Same Net Spacing Rule



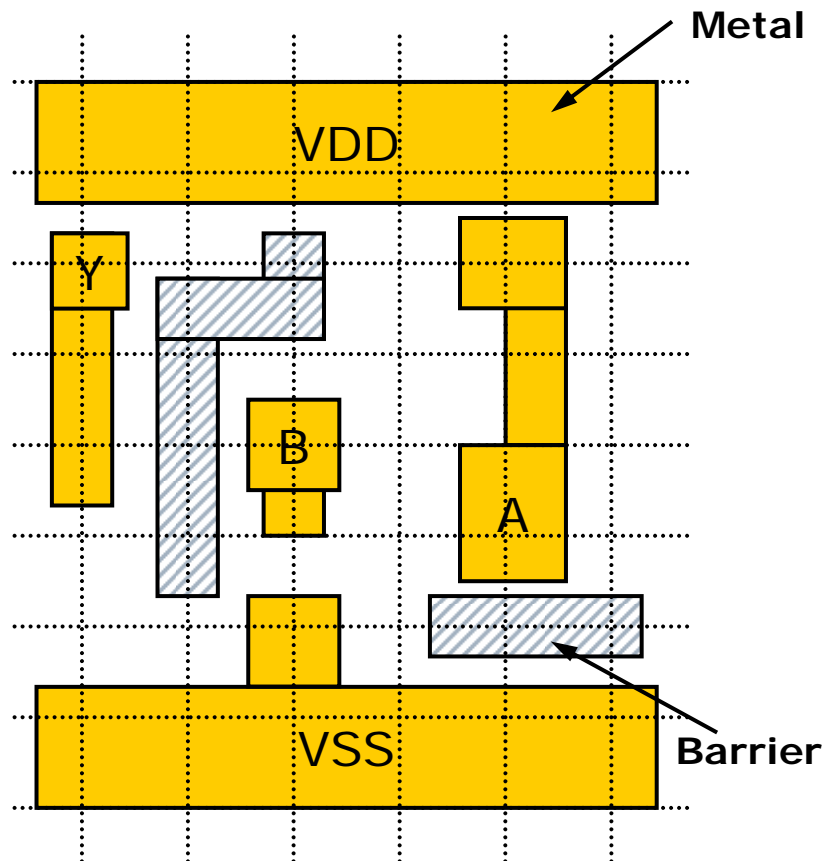
VIA12 and VIA23 allow stack

# LEF Format – APR Technology: Physical Macros

---

- Define physical data for
  - Standard cells
  - I/O pads
  - Memories
  - Other hard macros
- Describe abstract shape
  - Size
  - Class
  - Pins
  - Obstructions

# LEF Format – APR Technology: Physical Macros (Cont')



```
MACRO ADD1
  CLASS CORE;
  FOREIGN ADD1 0.0 0.0;
  ORIGIN 0.0 0.0;
  LEQ ADD;
  SIZE 19.8 BY 6.4;
  SYMMETRY x y;
  SITE coresite;
  PIN A
    DIRECTION INPUT;
  PORT
    LAYER Metal1;
    RECT 19.2 8.2 19.5 10.3;
    ....
  END
  END A
  ....
END ADD1
```

# LIB Format

---

- ❑ Operating condition
  - Slow, fast, typical
- ❑ Pin type
  - Input/output/inout
  - Function
  - Data/clock
  - Capacitance
- ❑ Path delay
- ❑ Timing constraint
  - Setup, hold, mpwh, mpwl, recovery

# Gate-Level Netlist

---

- ❑ If designing a chip, IO PADS, power PADS, and Corner PADS should be added before the netlist is imported
- ❑ Make sure that there is no “assign” statement and no “\*cell\*” cell name in the netlist



# SDC Constraint

---

- ☐ Clock constraints
- ☐ Input delay/ Input drive
- ☐ Output delay/ Output load
- ☐ False path
- ☐ Multi-cycle path

# I/O Constraint

**Version: 1**

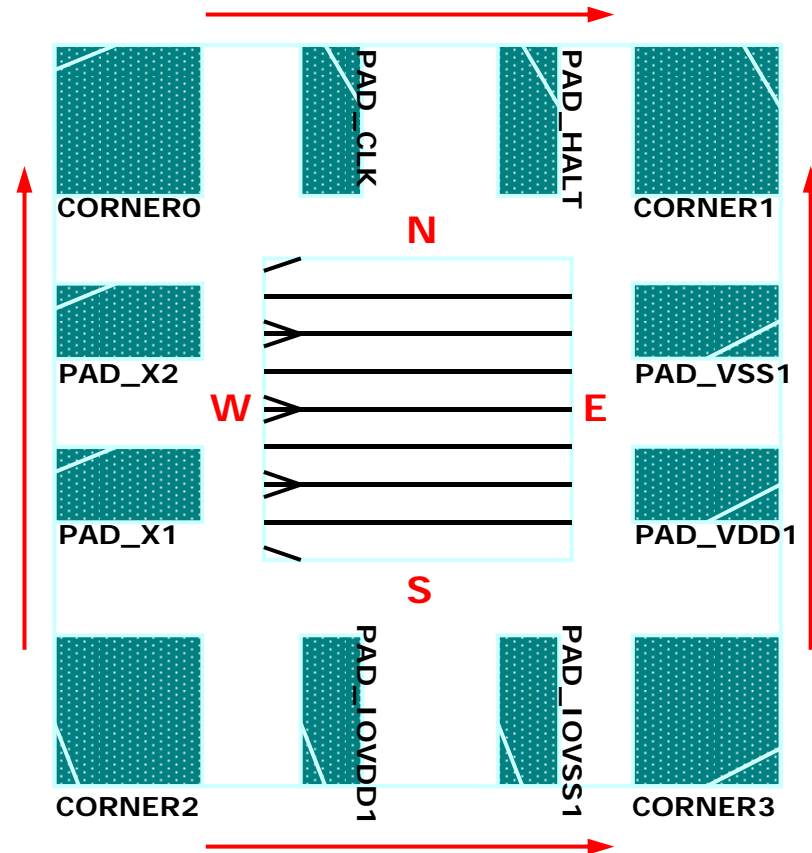
**Pad: CORNER0**      **NW PCORNERDGZ**  
**Pad: PAD\_CLK**      **N**  
**Pad: PAD\_HALT**      **N**

**Pad: CORNER1**      **NE PCORNERDGZ**  
**Pad: PAD\_X1**      **W**  
**Pad: PAD\_X2**      **W**

**Pad: CORNER2**      **SW PCORNERDGZ**  
**Pad: PAD\_IOVDD1**      **S PVDD2DGZ**  
**Pad: PAD\_IOVSS1**      **S PVSS2DGZ**

**Pad: CORNER3**      **SE PCORNERDGZ**  
**Pad: PAD\_VDD1**      **E PVDD1DGZ**  
**Pad: PAD\_VSS1**      **E PVSS1DGZ**

(\* .ioc File)



# How To Decide the NO. of Power/Ground PADS

---

- ❑ The following factors are considered:
  - SSO: Simultaneously Switch Outputs
  - SSN: The noise produced by SSO buffers
  - DI: Maximum NO. of copies for one specific kind of IO PAD switching from high to low simultaneously without making ground voltage level higher than 0.8 volt for one ground PAD
  - DF: Driving Factor,  $DF = 1/DI$
  - SDF: Sum of Driving Factor
- ❑ Suggestion in SSO case:
  - Required NO. of ground PADS = SDF
  - Required NO. of power PADS =  $SDF/1.1$

# SDF Example

---

IO Type	2mA	4mA	8mA	12mA	16mA	24mA
DF Value	0.02	0.03	0.09	0.18	0.3	0.56

- If a design has 20 PDB02DGZ (2mA) and 10 PDD16DGZ (16mA). Then,
- $SDF = 20 \times 0.02 + 10 \times 0.3 = 3.4$
- In SSO case,
  - NO. of VSS PAD =  $3.4 \rightarrow 4$
  - NO. of VDD PAD =  $3.4/1.1 = 3.09 \rightarrow 4$

# Tips to Reduce the Power/Ground Bounce

---


- ☐ Don't use stronger output buffers than what is necessary
- ☐ Use slew-rate controlled outputs
- ☐ Place power pad near the middle of the output buffer
- ☐ Place noise sensitive I/O pads away from SSO I/Os
- ☐ Place VDD and VSS pads next to clock input buffer

---

# Auto Place and Route Using SOC Encounter

# CHIP-Level Netlist

- If your gate-level netlist is generated by “CORE-level synthesis”, you should add the “CHIP-level module” in it

Ex:  CHIP.vg - KWrite

```
File Edit View Bookmarks Tools Settings Help
[Icons]

module CHIP ( I_clk, I_rst, I_ams, I_CSI, I_finj, I_si, I_se, I_scantest,
              I_bist_mode, I_mem_sel, I_WEN_S, I_CS_S, I_OE_S, I_DI_S, I_ADDR_S,
              O_cmd_done, O_BGO, O_state_addr, O_hold_state_addr, O_so, O_Q, I_tpclk,
              I_test_si2, O_test_so2 );
input [1:0] I_finj;
input [7:0] I_DI_S;
input [7:0] I_ADDR_S;
output [3:0] O_state_addr;
output [3:0] O_hold_state_addr;
output [7:0] O_Q;
input I_clk, I_rst, I_ams, I_CSI, I_si, I_se, I_scantest, I_bist_mode,
      I_mem_sel, I_WEN_S, I_CS_S, I_OE_S, I_tpclk, I_test_si2;
output O_cmd_done, O_BGO, O_so, O_test_so2;
wire clk, rst, ams, CSI, si, scantest, se, bist_mode, mem_sel, WEN_S, CS_S,
      OE_S, cmd_done, BGO, so;
wire [1:0] finj;
wire [7:0] DI_S;
wire [7:0] ADDR_S;
wire [3:0] state_addr;
wire [3:0] hold_state_addr;
wire [7:0] Q;

bist_group bist_group ( .clk(clk), .rst(rst), .ams(ams), .CSI(CSI), .finj(
    finj), .si(si), .scantest(scantest), .se(se), .bist_mode(bist_mode),
    .mem_sel(mem_sel), .WEN_S(WEN_S), .CS_S(CS_S), .OE_S(OE_S), .DI_S(DI_S),
    .ADDR_S(ADDR_S), .cmd_done(cmd_done), .BGO(BGO), .state_addr(state_addr),
    .hold_state_addr(hold_state_addr), .so(so), .Q(Q), .tpclk(tpclk), .test_si2(
    test_si2), .test_so2(test_so2));
PDIDGZ PAD_clk ( .PAD(I_clk), .C(clk) );
PDIDGZ PAD_rst ( .PAD(I_rst), .C(rst) );
PDIDGZ PAD_ams ( .PAD(I_ams), .C(ams) );
PDIDGZ PAD_CSI ( .PAD(I_CSI), .C(CSI) );
PDIDGZ PAD_finj0 ( .PAD(I_finj[0]), .C(finj[0]) );
PDIDGZ PAD_finj1 ( .PAD(I_finj[1]), .C(finj[1]) );
PDIDGZ PAD_si ( .PAD(I_si), .C(si) );
```

# CHIP-Level Netlist (Cont')

- If your design has a “Hard Block”, you should add an “empty module” for it
  - the module name should be the same as the “cell name” of the Hard Block

Ex:

```
module memory_0407 (O, clock, cen_in, oen_in, wen_in, A, D);  
  input    clock;  
  input [7:0] A;  
  input [7:0] D;  
  input    cen_in;  
  input    oen_in;  
  input    wen_in;  
  output [7:0] O;  
endmodule
```

(Module Declaration)

```
wire [7:0] D;  
wire [7:0] DI_T;  
wire [7:0] A;  
wire [7:0] ADDR_T;  
wire [7:0] Q2;  
wire [7:0] Q1;
```

(Module Reference)

Connected Wire Name in Verilog

```
RA1SHD256x8 RA1SHD256x8 ( .Q(Q1), .CLK(clk), .CEN(CEN1), .OEN(n168), .WEN(  
  WEN), .A(A), .D(D) );
```

```
memory_0407 memory_0407(.O(Q2), .clock(clk), .cen_in(CEN2), .oen_in(n168),  
  .wen_in(WEN), .A(A), .D(D));
```

Pin Name in SPICE



# CHIP-Level Timing Constraint

Ex:

```
#####  
# Created by Design Compiler write_sdc on Sun Jul 29 06:04:11 2007  
#####  
set_sdc_version 1.4  
  
create_clock -period 9 -waveform {0 4.5} [get_ports {clk}]  
set_input_delay 0.34 -clock "clk" [get_ports {ADDR_S[0]}]  
set_input_delay 0.34 -clock "clk" [get_ports {ADDR_S[1]}]  
set_input_delay 0.34 -clock "clk" [get_ports {ADDR_S[2]}]  
set_input_delay 0.34 -clock "clk" [get_ports {ADDR_S[3]}]  
set_input_delay 0.34 -clock "clk" [get_ports {ADDR_S[4]}]  
set_input_delay 0.34 -clock "clk" [get_ports {ADDR_S[5]}]  
set_input_delay 0.34 -clock "clk" [get_ports {ADDR_S[6]}]  
set_input_delay 0.34 -clock "clk" [get_ports {ADDR_S[7]}]  
set_input_delay 0.34 -clock "clk" [get_ports {DI_S[0]}]  
set_input_delay 0.34 -clock "clk" [get_ports {DI_S[1]}]  
set_input_delay 0.34 -clock "clk" [get_ports {DI_S[2]}]  
set_input_delay 0.34 -clock "clk" [get_ports {DI_S[3]}]
```



**CHIP-Level Clock Declaration**

**Set False Path to Your Test Pins**

**Set Parameters to the PAD IO**

```
# Created by Design Compiler write_sdc on Mon Jan 1 21:48:36 2007  
#####  
set_sdc_version 1.4  
current_design CHIP
```

```
create_clock [get_pins {PAD_clk/C}] -name CLK1 -period 20 -waveform {0  
10}
```

```
set_case_analysis 0 [get_ports {I_se}]  
set_max_fanout 50 [current_design]
```

```
set_false_path -from [get_ports {I_S1}]  
set_false_path -from [get_ports {I_se}]  
set_false_path -from [get_ports {I_scantest}]
```

```
set_input_delay 0.34 -clock [get_clocks {CLK1}] [get_ports {I_ADDR_S[0]}]  
set_input_delay 0.34 -clock [get_clocks {CLK1}] [get_ports {I_ADDR_S[1]}]  
set_input_delay 0.34 -clock [get_clocks {CLK1}] [get_ports {I_ADDR_S[2]}]  
set_input_delay 0.34 -clock [get_clocks {CLK1}] [get_ports {I_ADDR_S[3]}]  
set_input_delay 0.34 -clock [get_clocks {CLK1}] [get_ports {I_ADDR_S[4]}]  
set_input_delay 0.34 -clock [get_clocks {CLK1}] [get_ports {I_ADDR_S[5]}]  
set_input_delay 0.34 -clock [get_clocks {CLK1}] [get_ports {I_ADDR_S[6]}]
```

# Getting Started

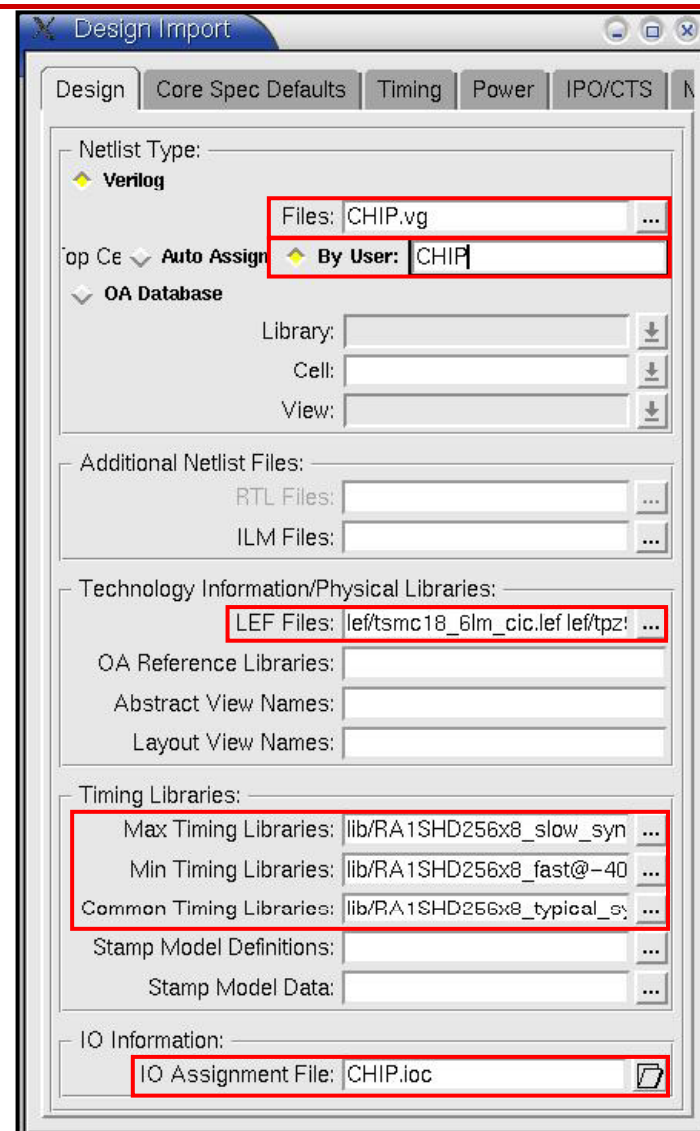
---

- ❑ *linux %> ssh -l "user name" cae18.ee.ncu.edu.tw* ← **Connect to Unix**
- ❑ *unix %> source /APP/cad/cadence/SOC/CIC/soc.csh*
- ❑ *unix %> encounter*

(Do not run in the background mode !!)

# Import Design <Design>

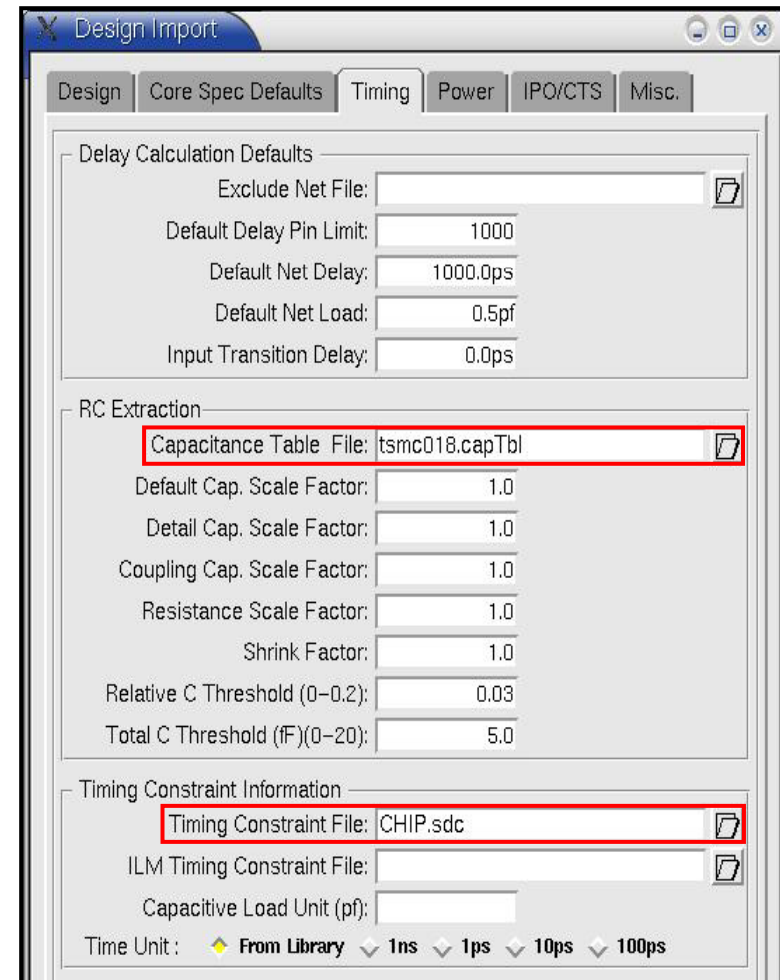
- ❑ **Design/Design Import**
- ❑ Verilog Files: your gate-level netlist
- ❑ Tot Cell
- ❑ LEF Files (\*.lef): including all the LEF files of cell libraries & hard blocks
- ❑ LIB Files (\*.lib):
  - Max Timing Libraries
  - Min Timing Libraries
  - Common Model Libraries
- ❑ IO Assignment File: \*.ioc



# Import Design <Timing>

---

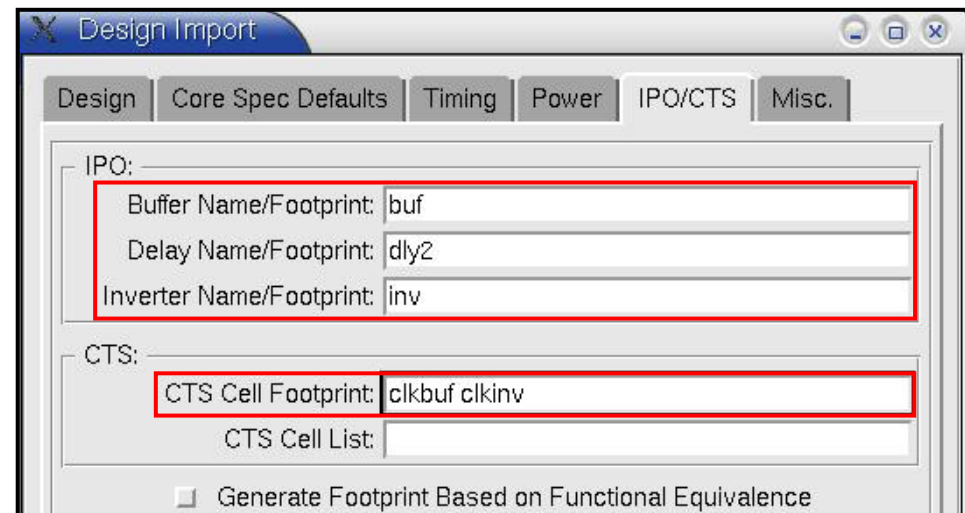
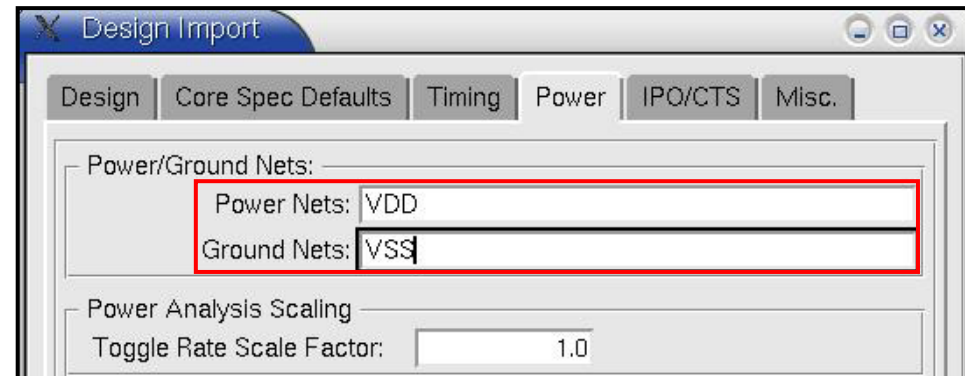
- ❑ Capacitance Table File
- ❑ Timing Constraint File: \*.sdc



# Import Design <Power> <IPO/CTS>

---

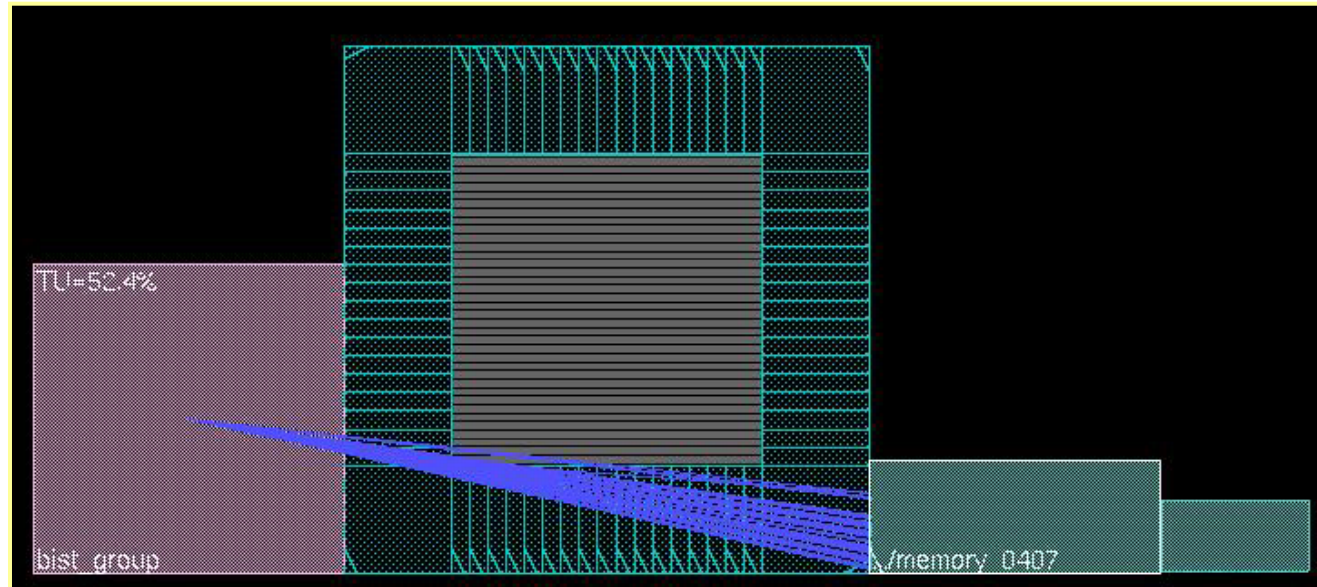
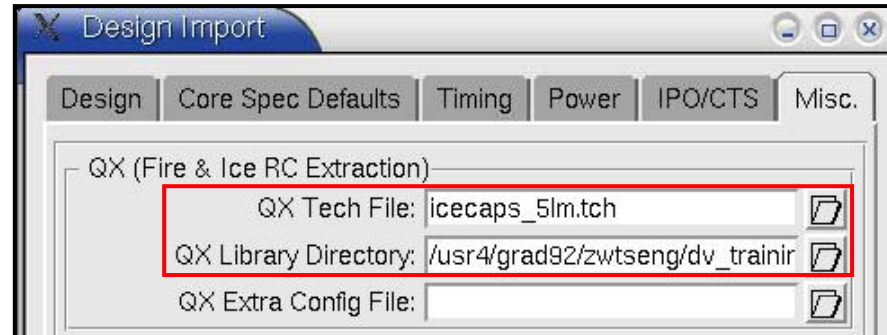
- ☐ Power Nets
- ☐ Ground Nets
- ☐ Footprints for In-Place Layout Optimization (IPO) and Clock Tree Synthesis (CTS)



# Import Design <Misc.>

---

- ❑ QX Tech File
- ❑ QX Library Directory

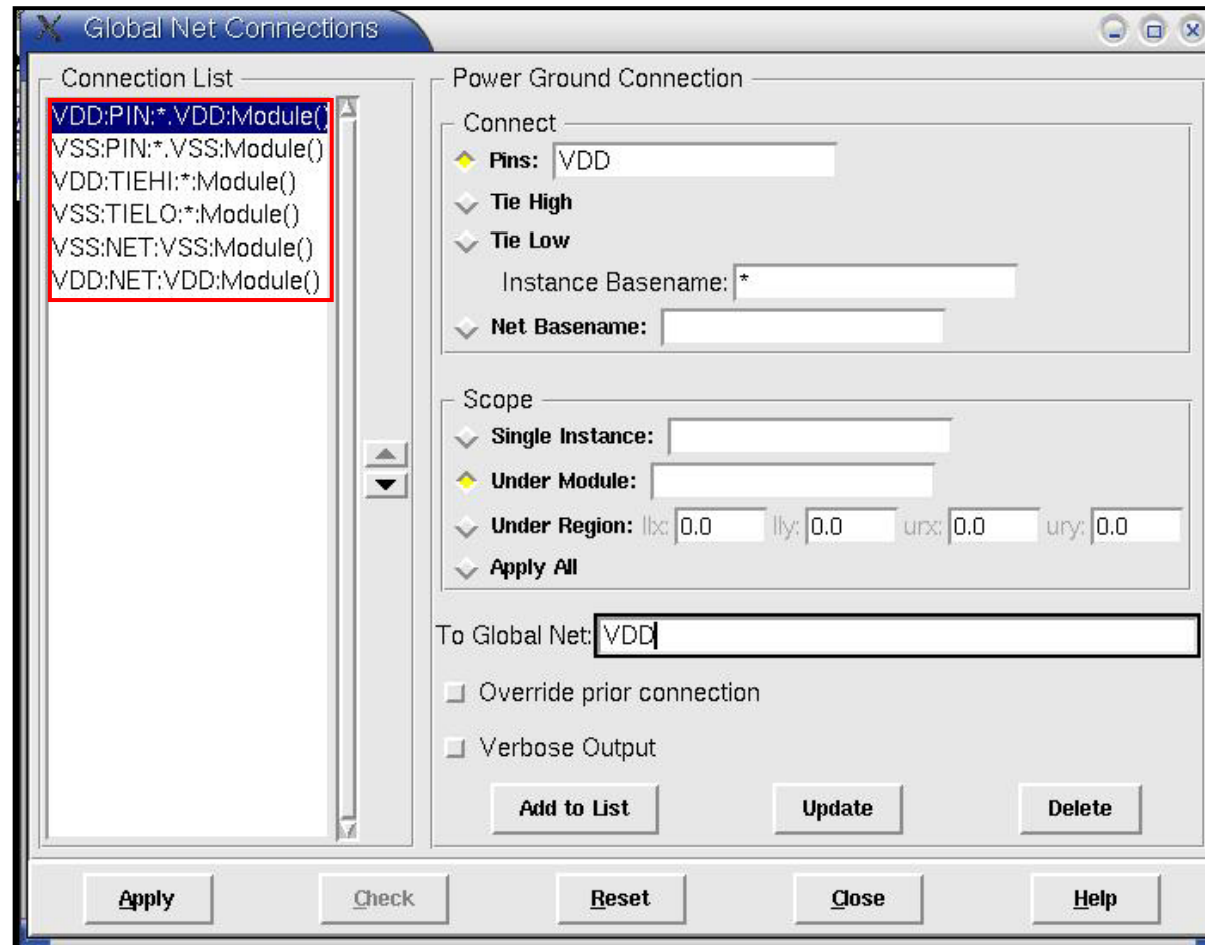


(Floorplan View)



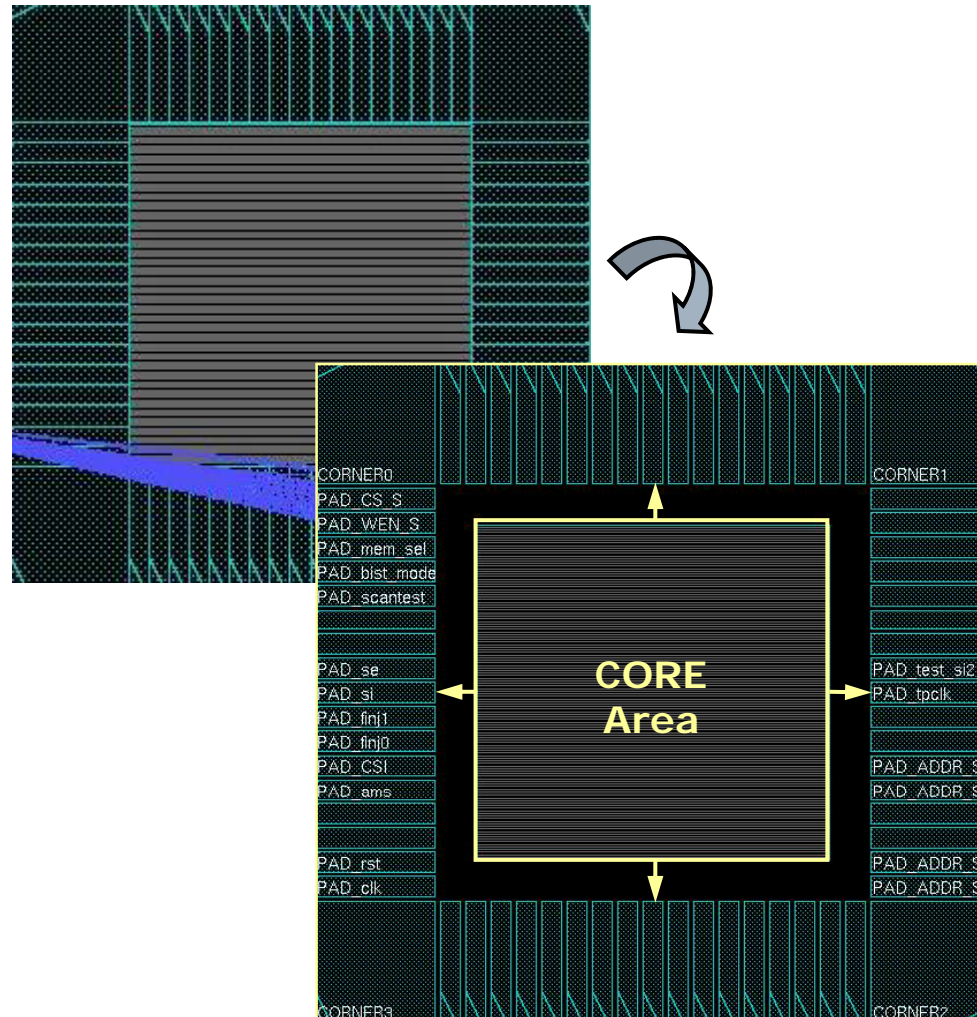
# Global Net Connection

## □ Floorplan/Global Net Connections



# Specify Floorplan

## □ Floorplan/Specify Floorplan



Specify Floorplan

Design Dimensions

Specify Dimensions by:

Size by:

Core Size by: Aspect Ratio: Ratio (H/W): 0.95097

Core Utilization: 0.524415

Std. Utilization: 0.117141

Width and Height: Core Height: 660. Core Width: 694.

Die Size by: Width and Height Die Height: 1290.52 Die Width: 1324.755

Core Margins by: Core to IO Boundary

Core to Die Boundary

Core to Left: 80.48 Core to Top: 80.0

Core to Right: 80.0 Core to Bottom: 80.28

Die Size Calculation Use: Max IO Height Min IO Height

Floorplan Origin at: Lower Left Corner Center

Die/IO/Core Coordinates:

Die LL:	Die UR:	IO LL:	IO UR:	Core LL:	Core UR:
0.0	0.0	235.0	235.0	315.48	315.28
	1324.755		1089.755		1009.755
	1290.52		1055.52		975.52

unit: micron

Standard Cell Rows

Double-back rows: Bottom row orient:

Row Spacing: 0.0 um For Every 2 Row

Site: tsm-site Row height:

OK Apply Cancel Help



# Specify Scan Chain

---

- ❑ `encounter %> specifyScanChain ScanChainName`
  - `start {ftname | instPinName}`
  - `start {ftname | instPinName}`
- ❑ `encounter %> scantrace`

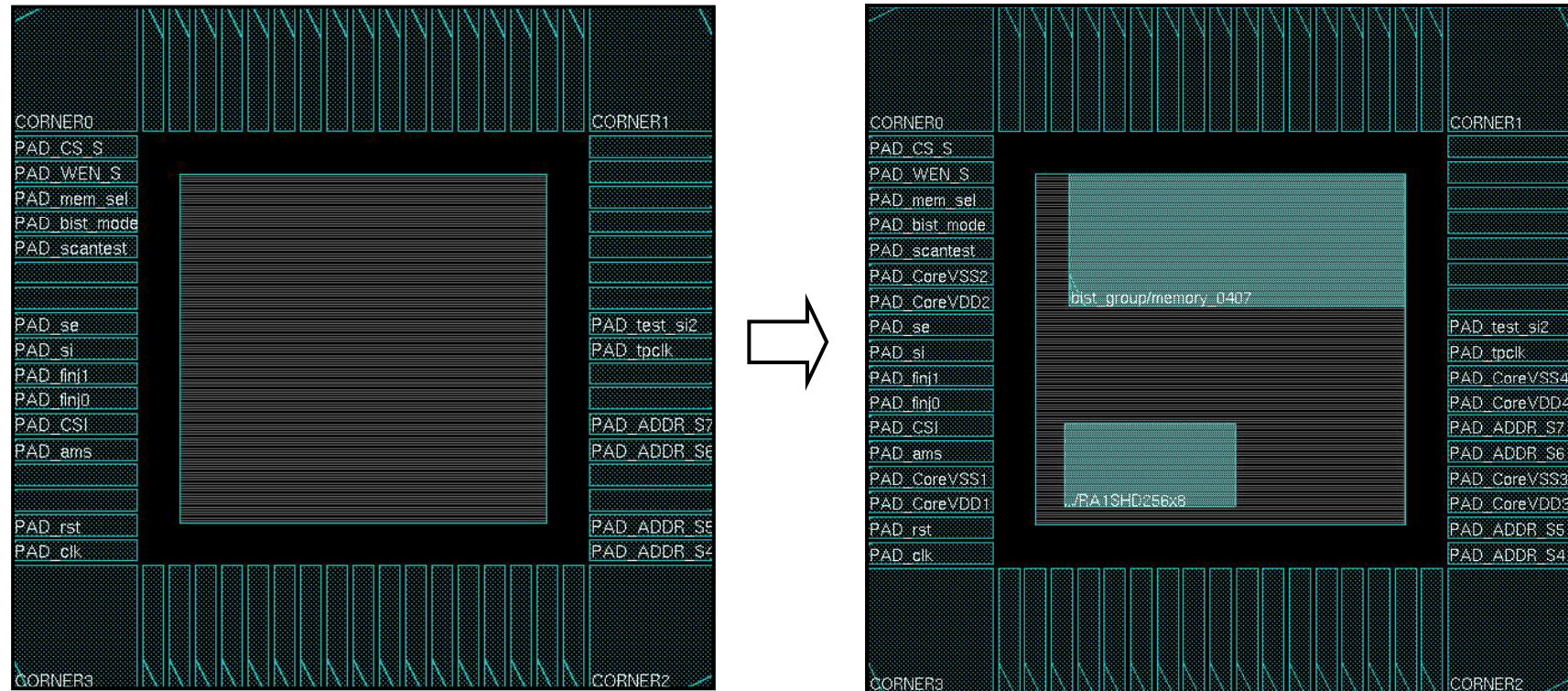
Ex:

```
encounter 2> specifyScanChain scan1 -start PAD_si/C -stop PAD_so/I
encounter 3> specifyScanChain scan2 -start PAD_test_si2/C -stop PAD_test_so2/I
encounter 4> scantrace

Tracing scan chain: scan1
Successfully traced scan group scan1 (99 elements: 97 scan bits).
Tracing scan chain: scan2
Successfully traced scan group scan2 (57 elements: 56 scan bits).
*** Scan Trace Summary:
Successfully traced scan group scan1 (99 elements: 97 scan bits).
Successfully traced scan group scan2 (57 elements: 56 scan bits).
Successfully traced 2 scan groups (total 156 elements: 153 scan bits).
INFO: Performed sanity check on scan group scan1 (+1 scan edge marked as fixed).
INFO: Passed sanity check on scan group scan2.
*** Scan Sanity Check Summary:
*** 1 scan group passed sanity check.
*** 1 scan group corrected sanity check (total +1 fixed scan edge ).
```

# Hard Block Placement

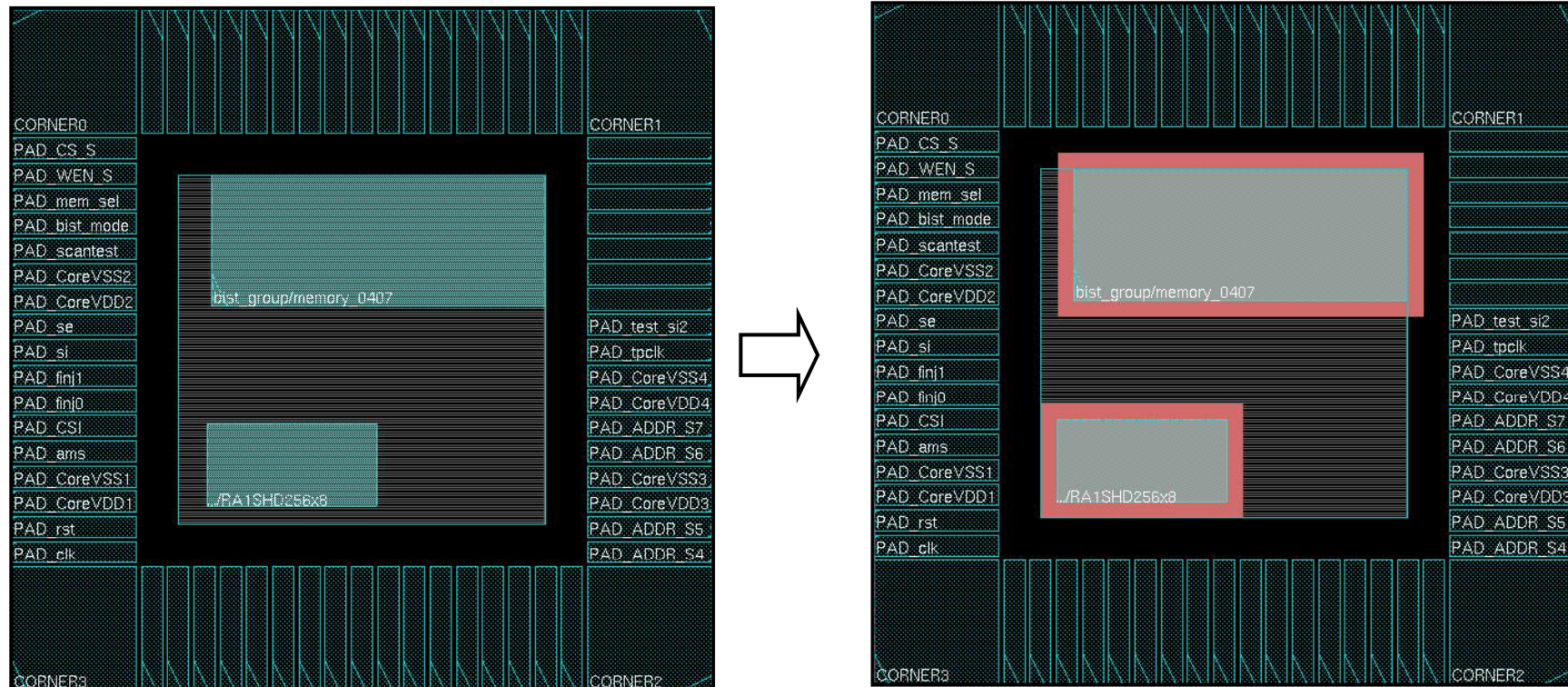
-  Move/Resize/Reshape floorplan object





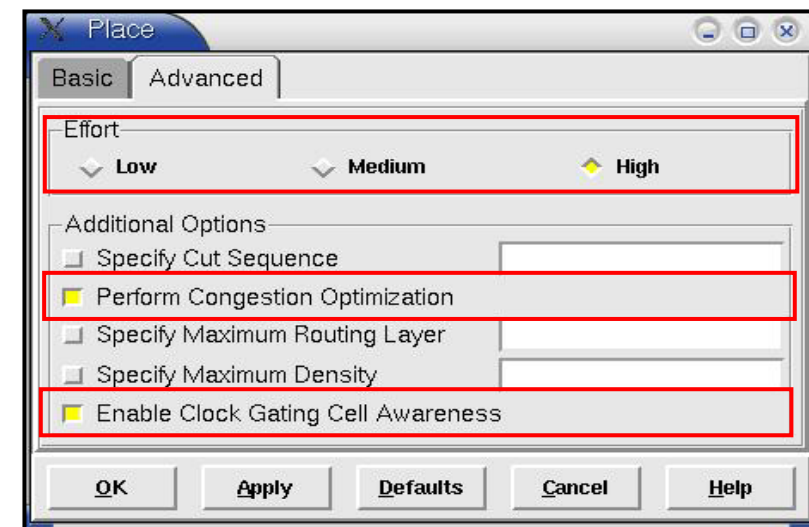
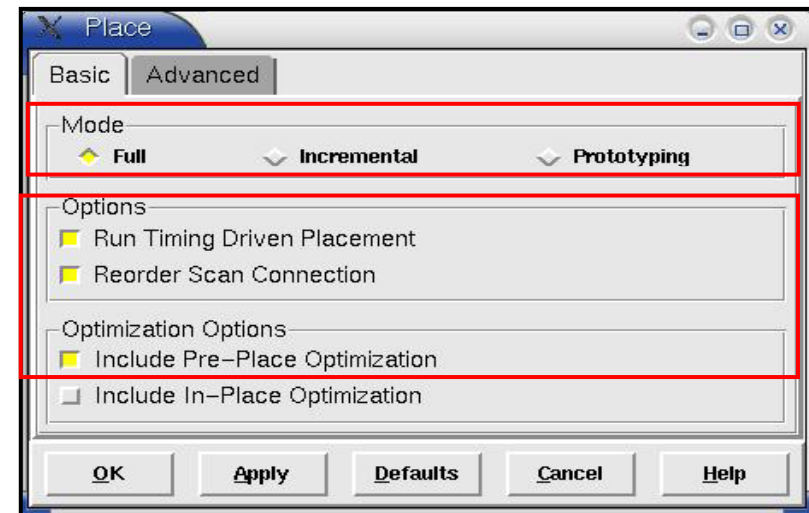
# Edit Block Halo

- ❑ *Floorplan/Edit Block Halo*
- ❑ Reserve space without standard cell placement



# Standard Cell Placement

## □ Place/Place



# Power Planning – Add Rings

## □ Floorplan/Custom Power Planning/Add Rings

Basic Advanced Via Generation

Net(s): VSS VDD

Ring Type

- Core ring(s) contouring:
  - ☒ Around core boundary ☐ Along I/O boundary
  - ☐ Exclude selected objects
- Block ring(s) around
  - ☒ Each block
  - ☐ Each reef
  - ☐ Selected power domain/fences/reefs
  - ☐ Each selected block and/or group of core rows
  - ☐ Clusters of selected blocks and/or groups of core rows
  - ☐ With shared ring edges
- User defined coordinates  MouseClick
- ☒ Core ring ☐ Block ring

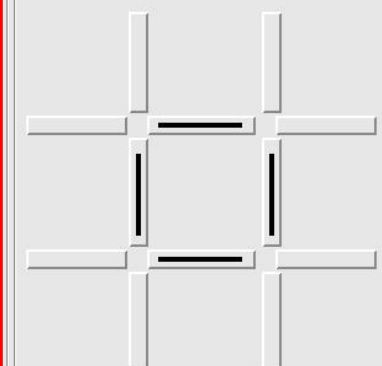
Ring Configuration

	Top:	Bottom:	Left:	Right:
Layer:	METAL5 H	METAL5 H	METAL4 V	METAL4 V
Width:	2	2	2	2
Spacing:	0.28	0.28	0.28	0.28
Offset:	<input checked="" type="checkbox"/> Center in channel <input type="checkbox"/> Specify			
	0.56	0.56	0.56	0.56

Update

Basic Advanced Via Generation

Set Custom Ring Sides and Extension



☐ Create rectangular ring(s) only

Merge with pre-routed rings if within spacing threshold: 0.56

Minimum jog distance: 0.56

Snap wire center to routing grid: None

Wire Group

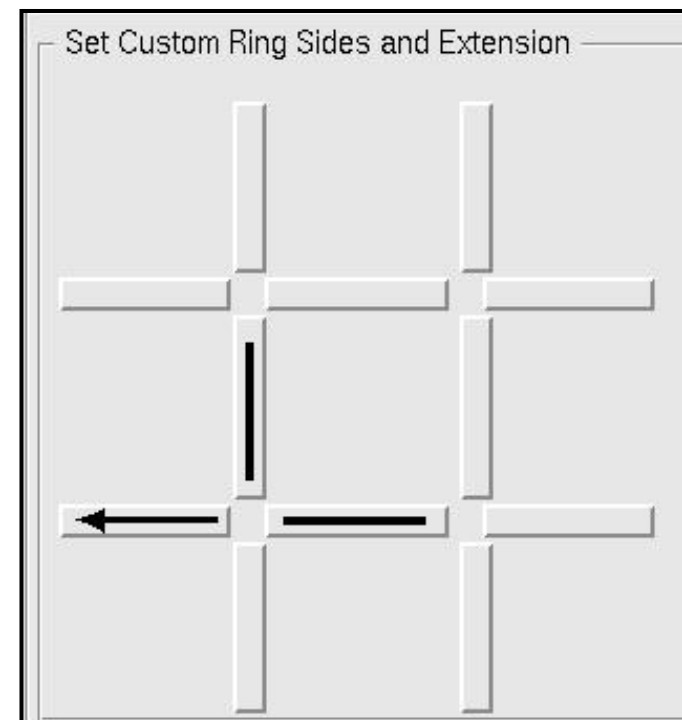
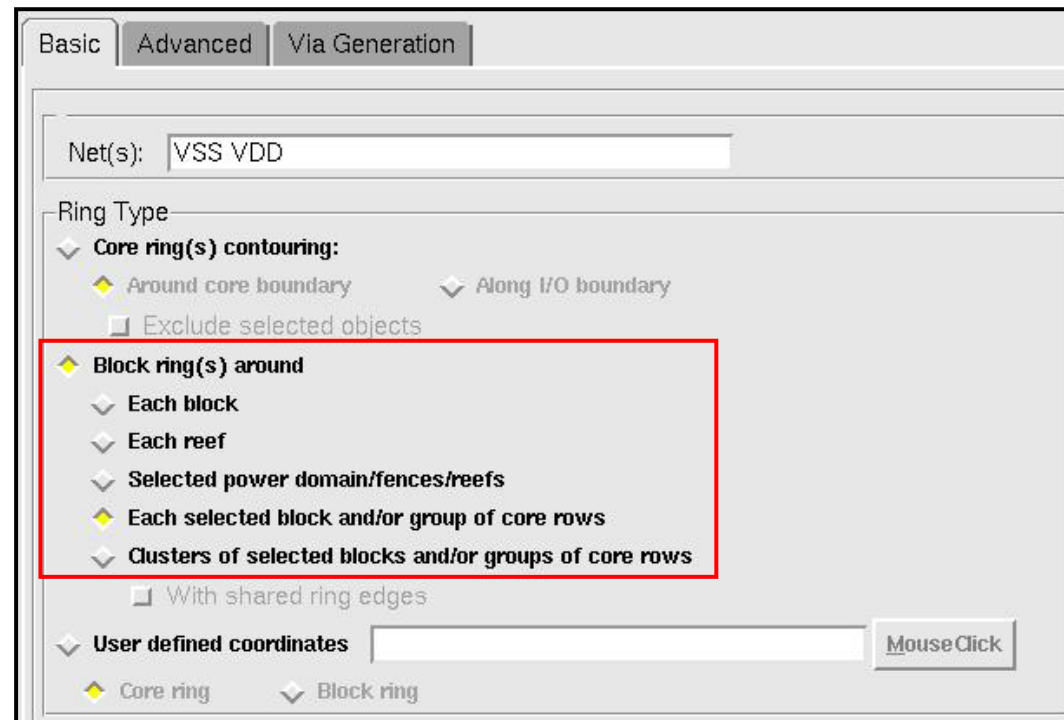
- ☒ Use wire group
- ☐ Interleaving
- Number of bits: 15

☐ Reinforcement stripes

Spacing: 0 Width: 0

# Power Planning – Add Block Rings

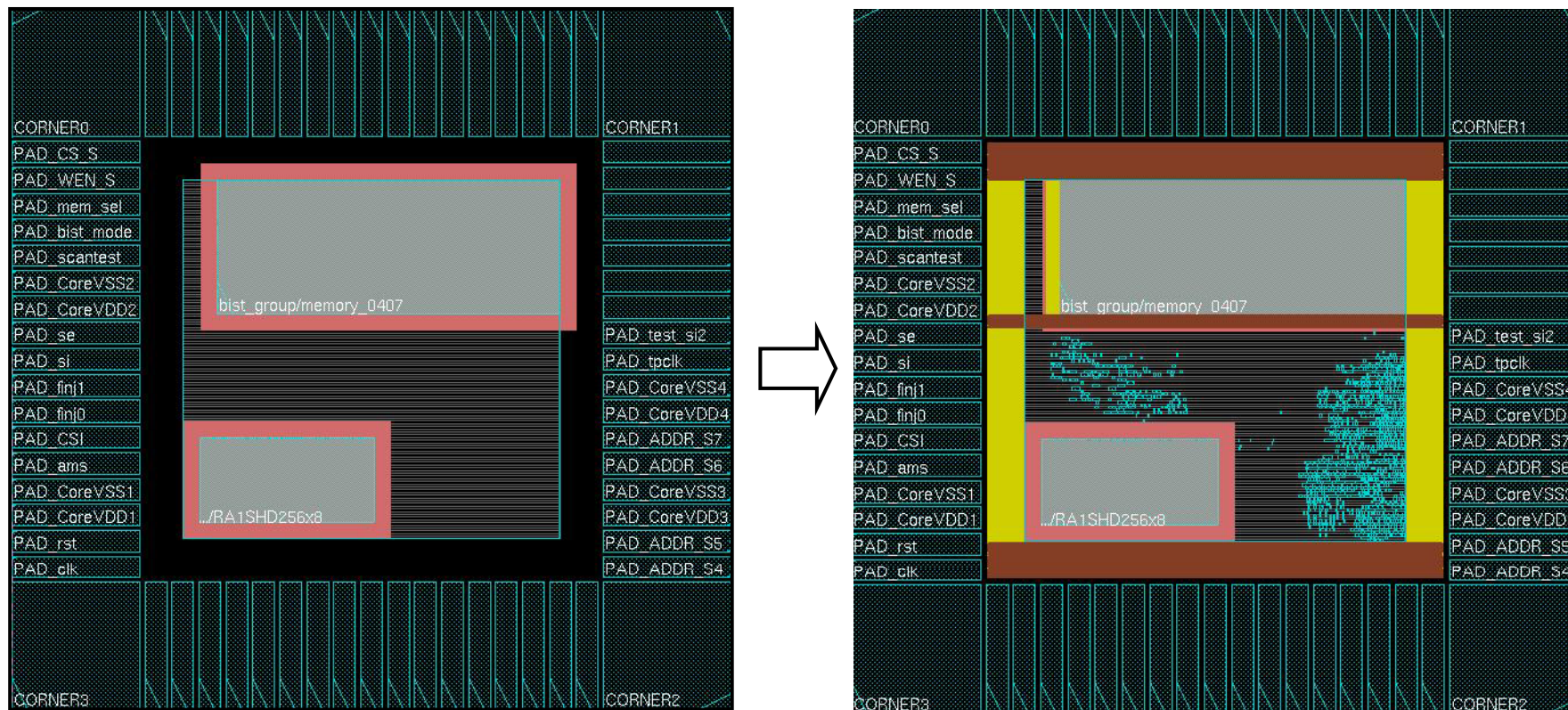
## □ *Floorplan/Custom Power Planning/Add Rings*





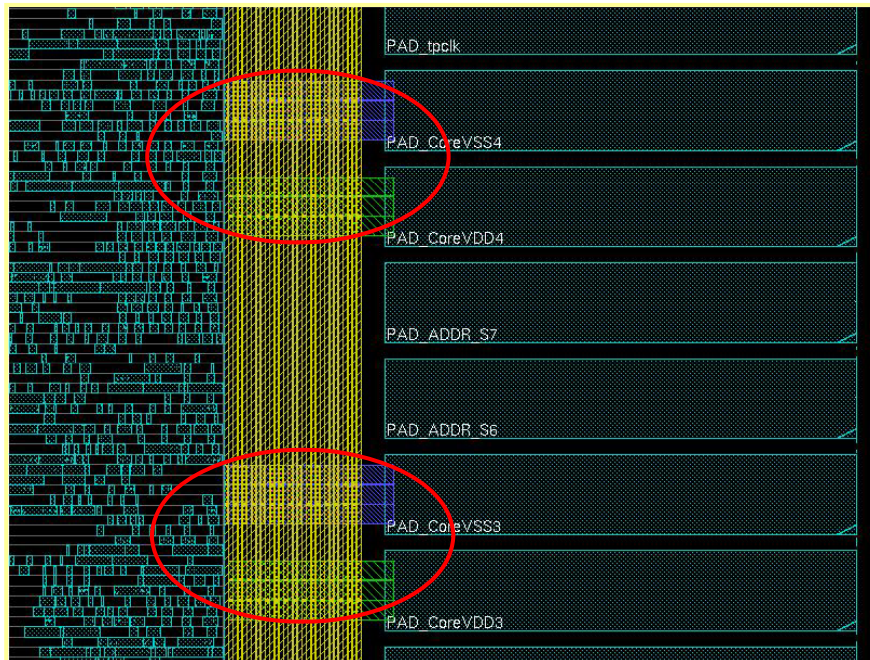
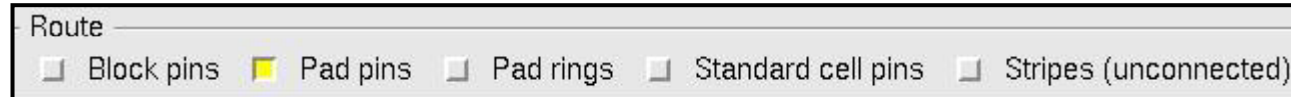
# Example for Power Rings

---



# PAD Pins

## ☐ Route/SRoute





# Power Planning – Add Stripes

## □ Floorplan/Custom Power Planning/Add Stripes

Basic | Advanced | Via Generation

Set Configuration

Net(s): VSS VDD

Layer: METAL4

Direction: ☒ Vertical ☐ Horizontal

Width: 1

Spacing: 0.28

Set Pattern

☒ Set-to-set distance: 100

Number of sets: 1

Bumps ☒ Over ☐ Between

Over P/G pins Pin layer: Top pin layer

Master name:  Selected blocks All blocks

Stripe Boundary

☒ Core ring

☐ Pad ring ☐ Inner ☒ Outer

☐ Design boundary ☐ Create pins

☐ Each selected block/domain/fence

☐ All domains

☐ Specify area

First / Last Stripe

Start from: ☐ left ☒ right

☒ Relative from core or selected area

X from left: 80 X from right: 20

☐ Absolute locations

Basic | Advanced | Via Generation

Stripe Breaking

☐ Omit stripes inside block rings

☐ Omit stripes over selected blocks/domains

Target Connection Control

Pad/Core ring connection

☐ Allow jogging

Block ring connection

☐ Allow jogging

☐ Merge with block rings if spacing less than: 0.56

Maximum length of same layer jog: 0.56

Layer Control for Target Connections

Pad/Core rings

Top limit: METAL4

Bottom limit: METAL4

Block rings

Top limit: METAL4

Bottom limit: METAL4

Wire Group

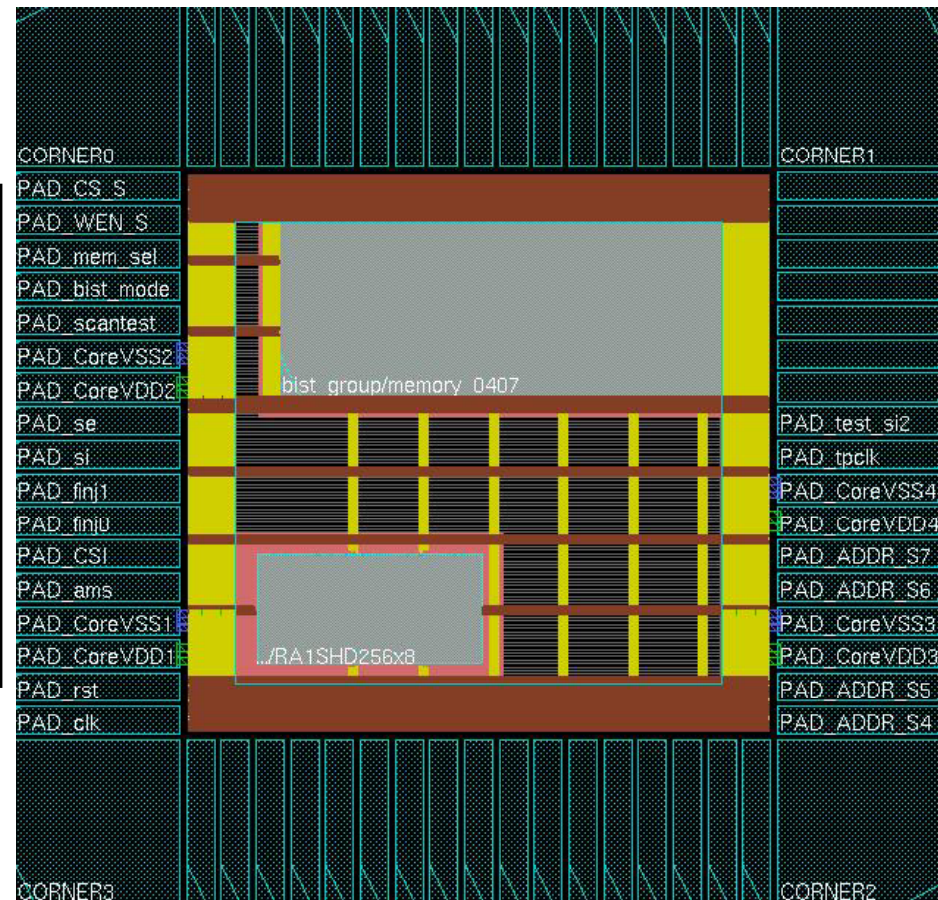
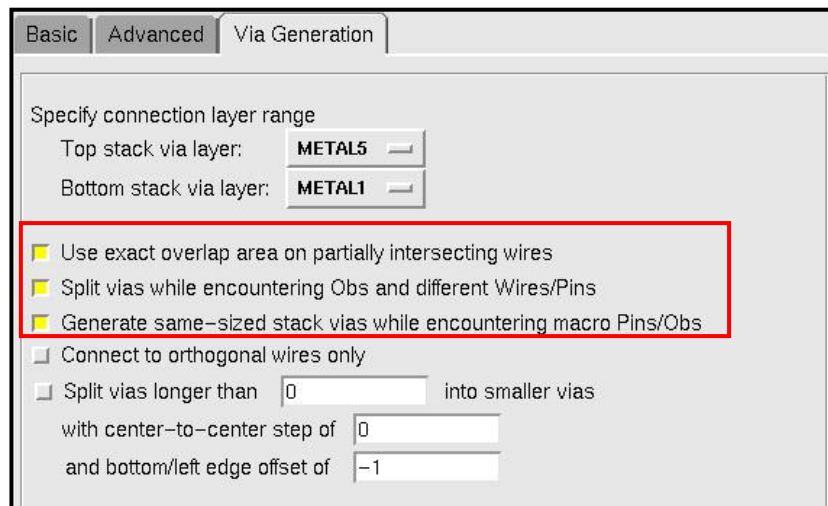
☐ Use wire group

☐ Interleaving

Number of bits: 5

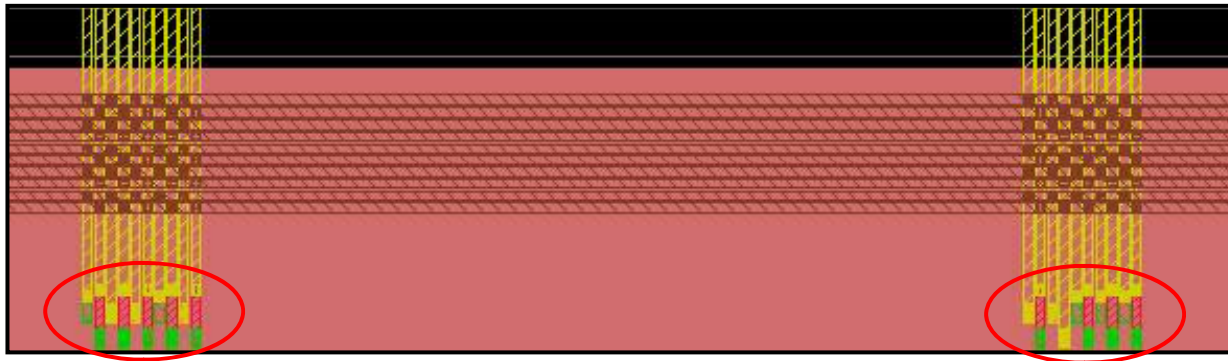
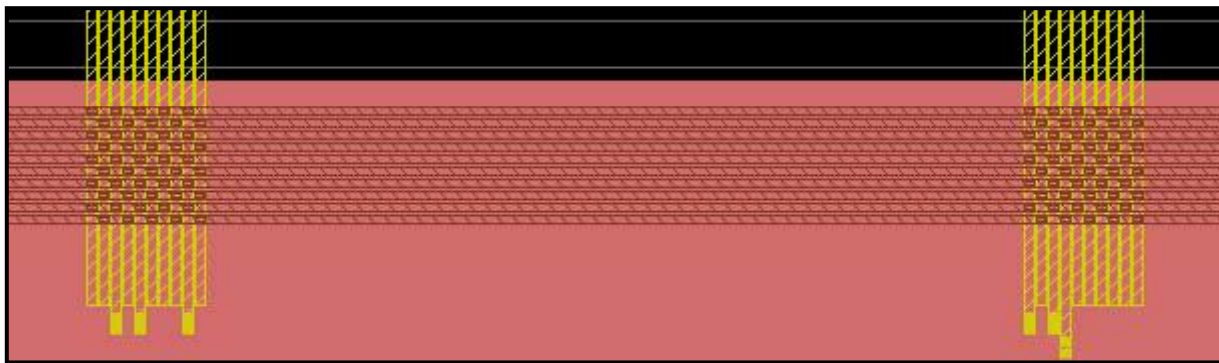
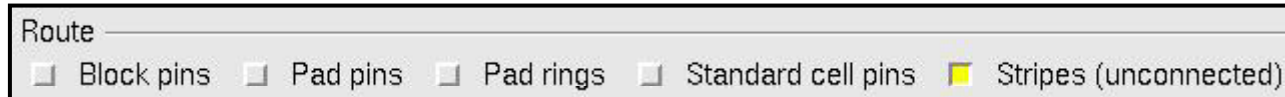
# Power Planning – Add Stripes (Cont')

Ex:



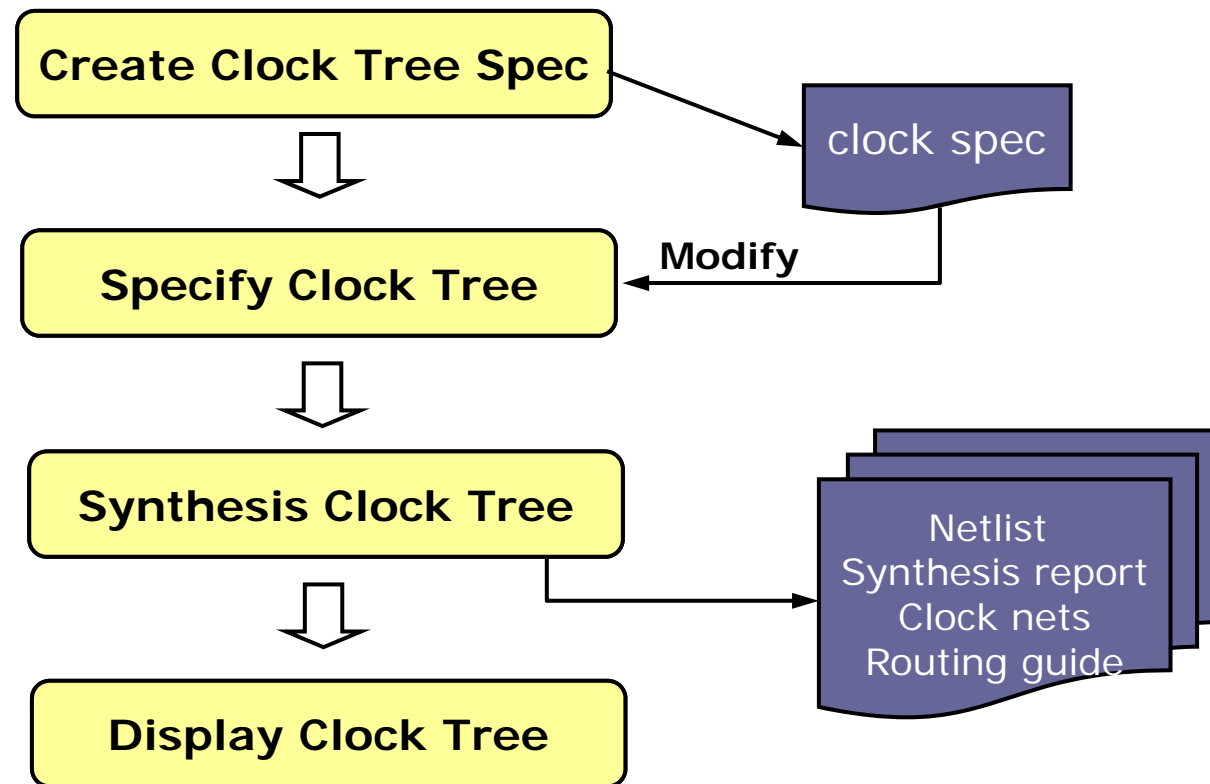
# Fix Un-Connected Stripes

## ☐ *Route/SRoute*



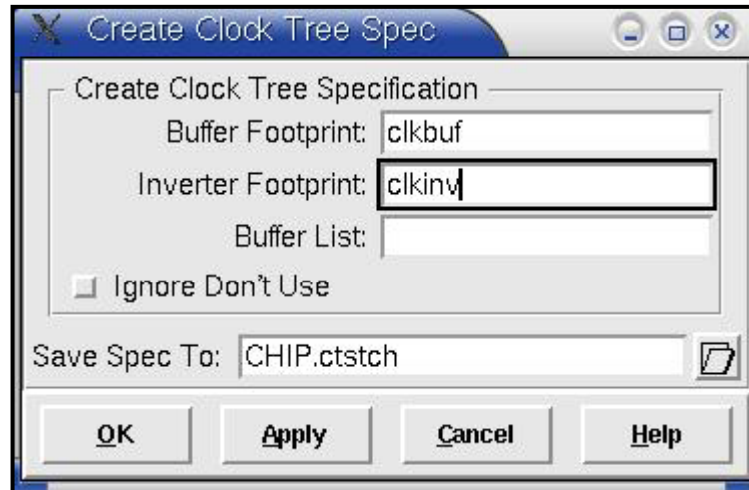
# Flow Clock Tree Synthesize

---

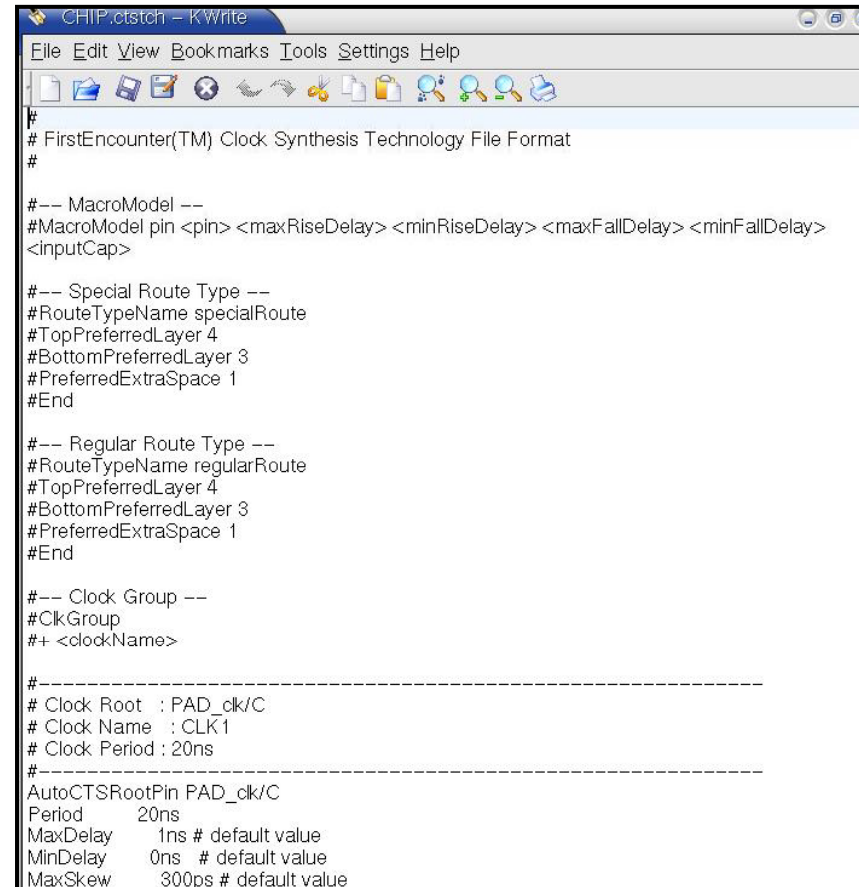


# Create/Specify/Synthesis Clock Tree Spec.

## ❑ Clock/Create Clock Tree Spec



## ❑ Clock/Specify Clock Tree



(Clock Spec.)

## ❑ Clock/Synthesis Clock Tree



# Example for CTS Report

---

```
Nr. of Subtrees      : 1
Nr. of Sinks        : 99
Nr. of Buffer       : 5
Nr. of Level (including gates) : 1
Max trig. edge delay at sink(F): bist_group/bist/m0/shift_reg_reg_10_/CKN 230.9(ps)
Min trig. edge delay at sink(R): bist_group/memory_0407/clock 176.8(ps)

                (Actual)      (Required)
Rise Phase Delay      : 176.8~217.5(ps)    0~1000(ps)
Fall Phase Delay      : 189.2~230.9(ps)    0~1000(ps)
Trig. Edge Skew       : 54.1(ps)          300(ps)
Rise Skew             : 40.7(ps)
Fall Skew             : 41.7(ps)
Max. Rise Buffer Tran  : 58.4(ps)          400(ps)
Max. Fall Buffer Tran  : 58.4(ps)          400(ps)
Max. Rise Sink Tran   : 177.9(ps)         400(ps)
Max. Fall Sink Tran   : 165.4(ps)         400(ps)
Min. Rise Buffer Tran  : 23.2(ps)          0(ps)
Min. Fall Buffer Tran  : 23.2(ps)          0(ps)
Min. Rise Sink Tran   : 97.9(ps)          0(ps)
Min. Fall Sink Tran   : 92.4(ps)          0(ps)

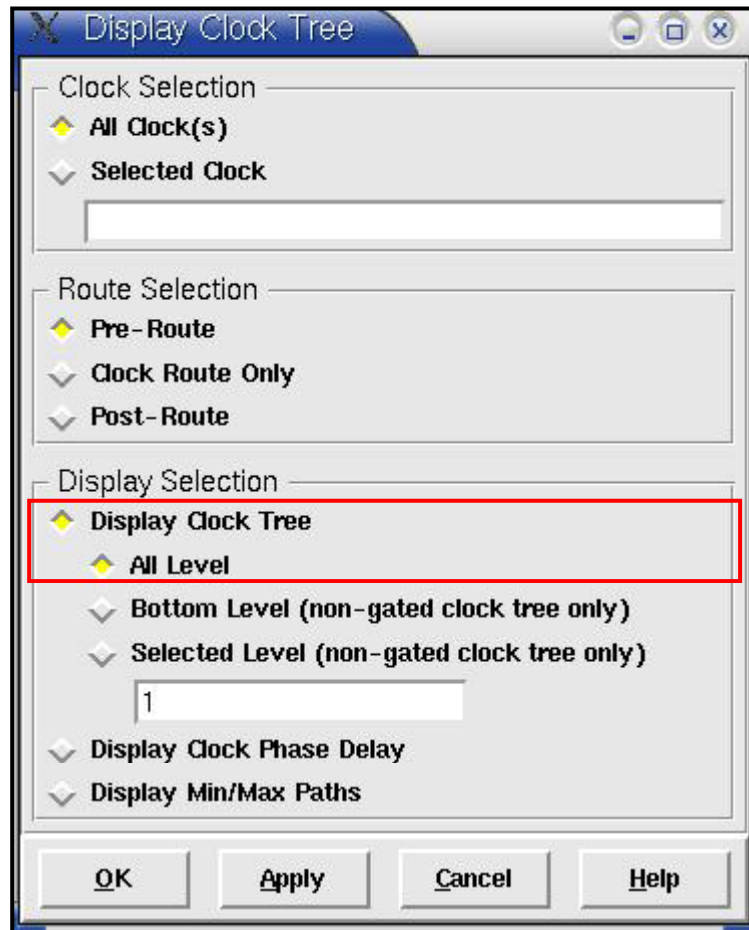
***** NO Max Transition Time Violation *****

***** NO Min Transition Time Violation *****

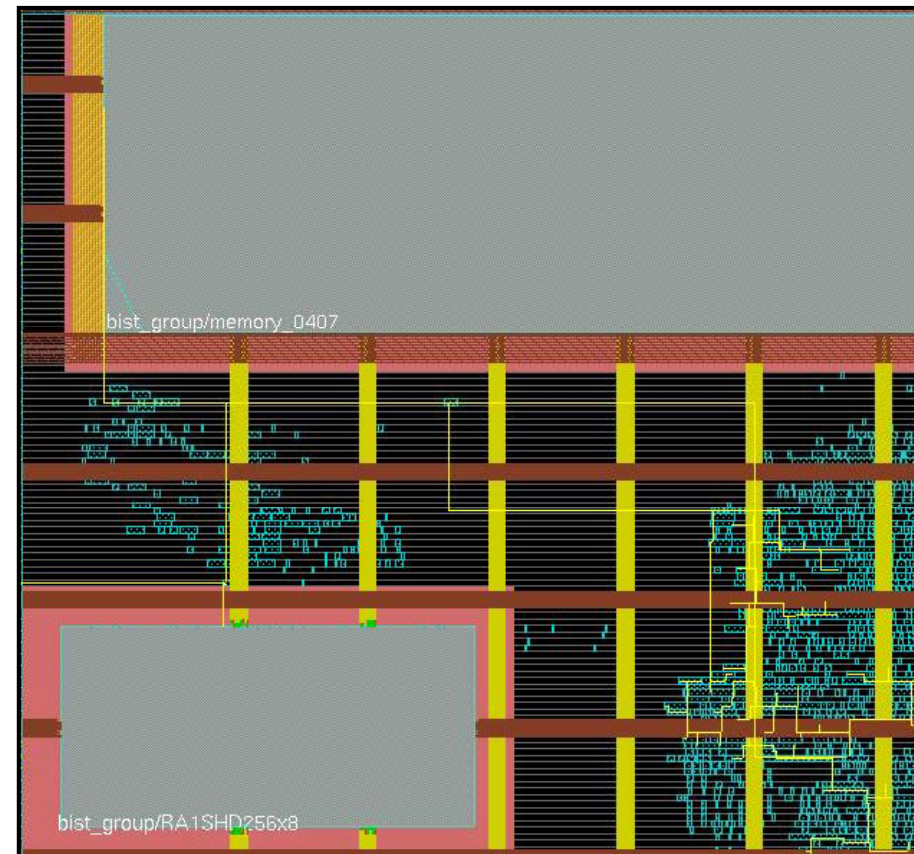
***** NO Max_Fanout Violation *****
```

# Display Clock Tree

## ☐ Clock/Display/Display Clock Tree

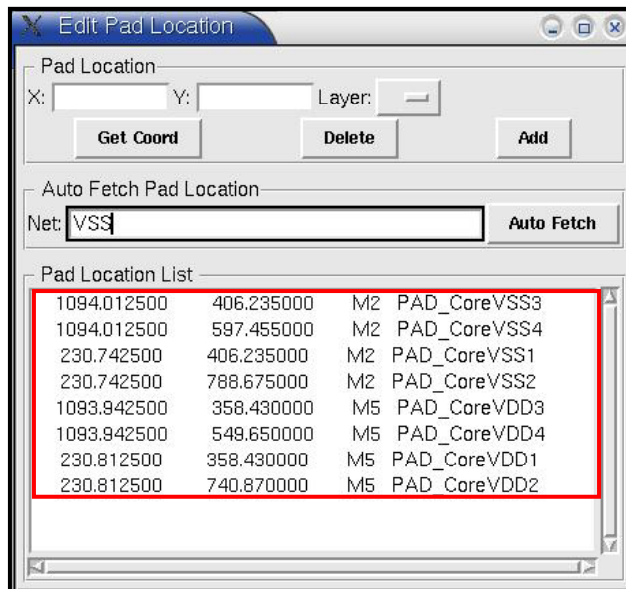


Ex:

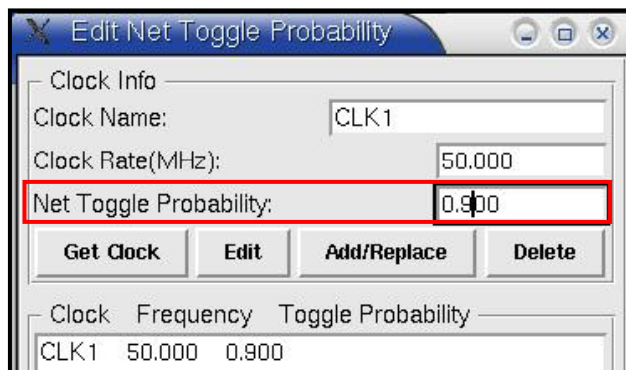


# Power Analysis

## □ Power/Edit Pad Location



## □ Power/Edit Net Toggle Probability



## □ Power/Power Analysis/Statistical

Ex:

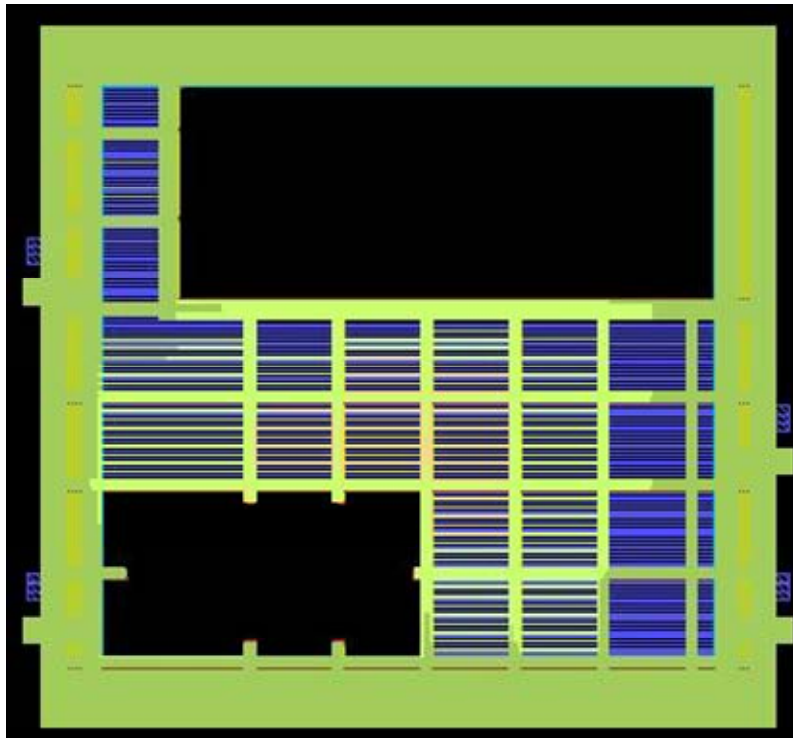
```
#####  
# The Power Analysis Report for VDD net #  
#####  
power supply: 1.98 v  
average power(default): 1.0470e+01 mw  
average switching power(default): 2.3212e+00 mw  
average internal power(default): 8.1442e+00 mw  
average leakage power(default): 4.7600e-03 mw  
average user specified power(default): 0.0000e+00 mw  
average power by clock domain category:  
clock domain(CLK1, 0.9) : 1.0282e+01 mw  
clock tree power : 4.1149e+00 mw  
non clock tree power : 6.1672e+00 mw  
unlock domain(0.2) : 1.8807e-01 mw  
average power by cell category:  
core: 7.3701e+00 mw  
block: 3.1001e+00 mw  
io: 0.0000e+00 mw  
average power(considered in rail analysis): 1.0470e+01 mw  
worst IR drop average analysis: 1.9718e-03 v  
number of nodes in rail network: 16645 nodes  
worst EM:  
"M1" 5.2000e-02 mA/u  
"M2" 0.0000e+00 mA/u  
"M3" 3.9142e-01 mA/u  
"M4" 3.9142e-01 mA/u  
"M5" 2.2647e-01 mA/u  
"V12" 1.4821e-02 mA/cut  
"V23" 1.4821e-02 mA/cut  
"V34" 9.7855e-02 mA/cut  
"V45" 2.8280e-02 mA/cut  
biggest toggled net: clk_L1_N0  
no. of terminal: 93  
total cap: 7.6044e+02 ff
```

(Power Analysis Report)

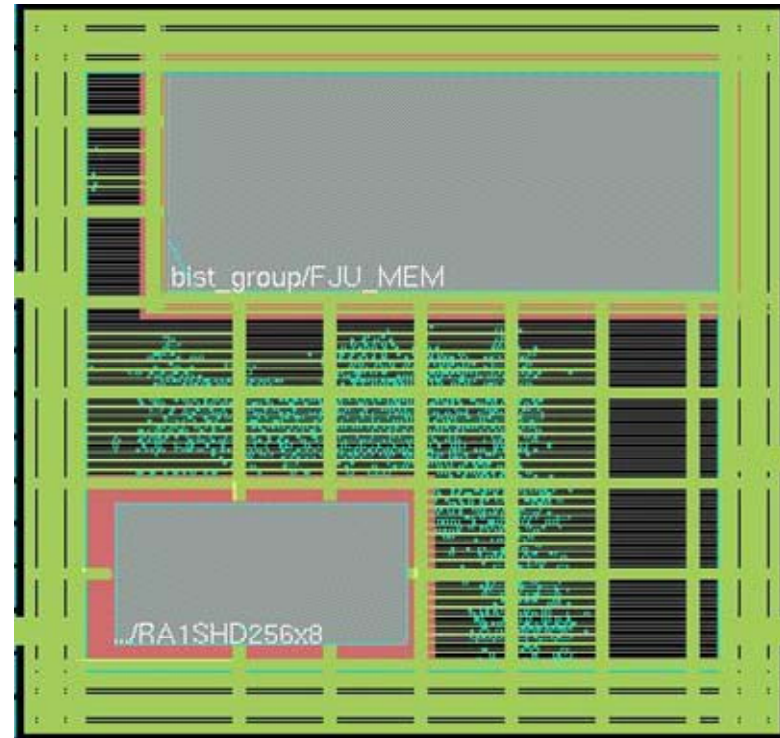


# Example for Rail Analysis of IR-Drop & EM

---



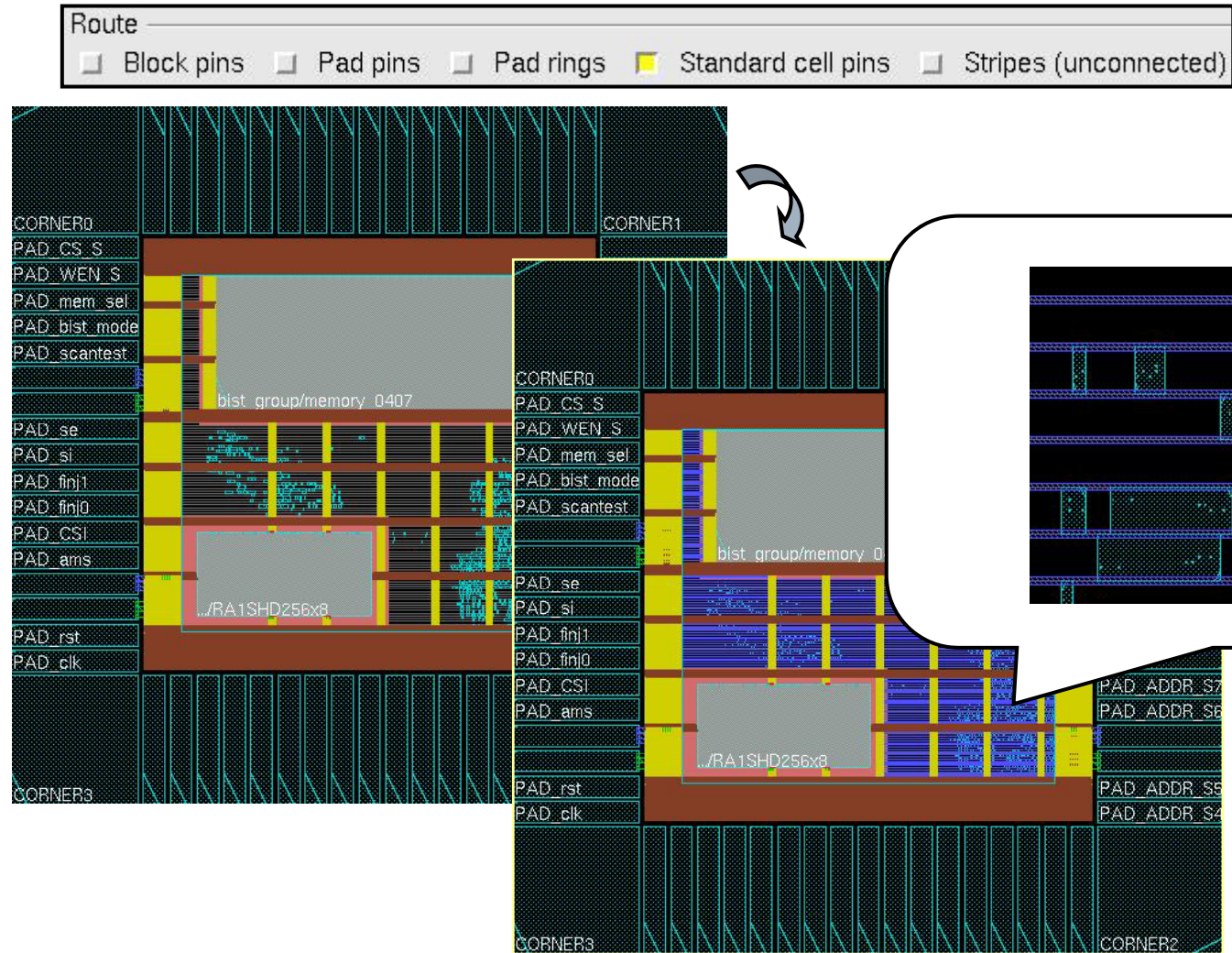
(IR-Drop)



(EM)

# Power Route

## □ Route/SRoute

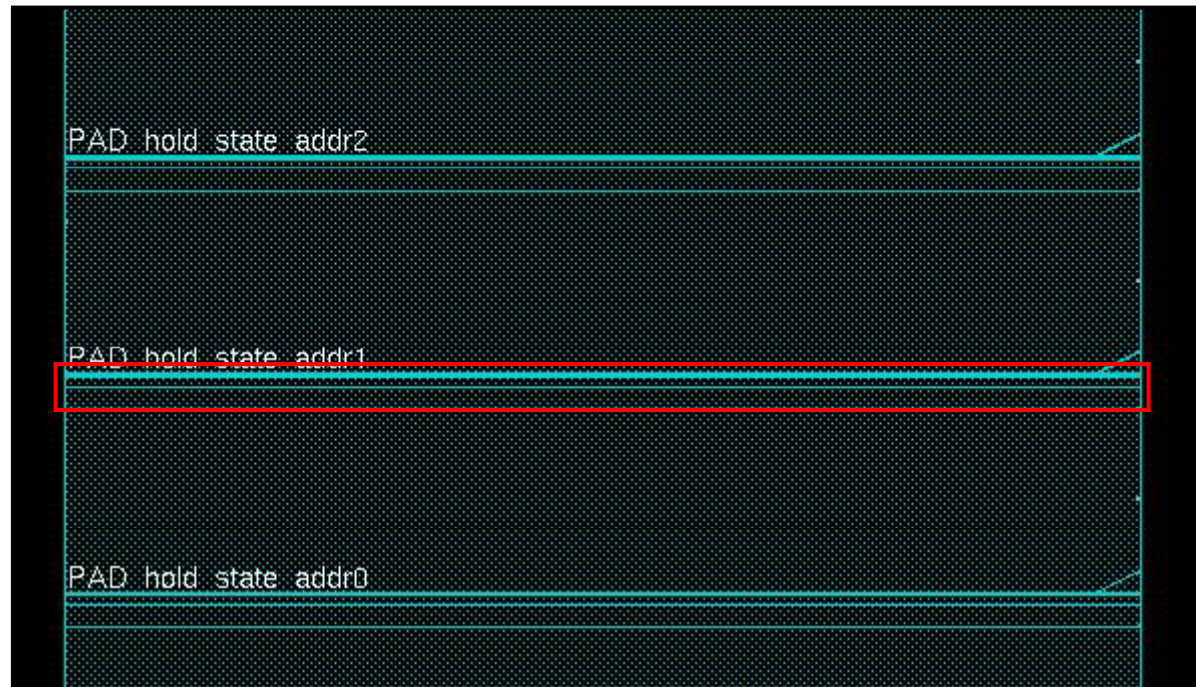




# IO Filler

---

- ❑ `encounter %> source addIoFiller.cmd`



# Nano Route

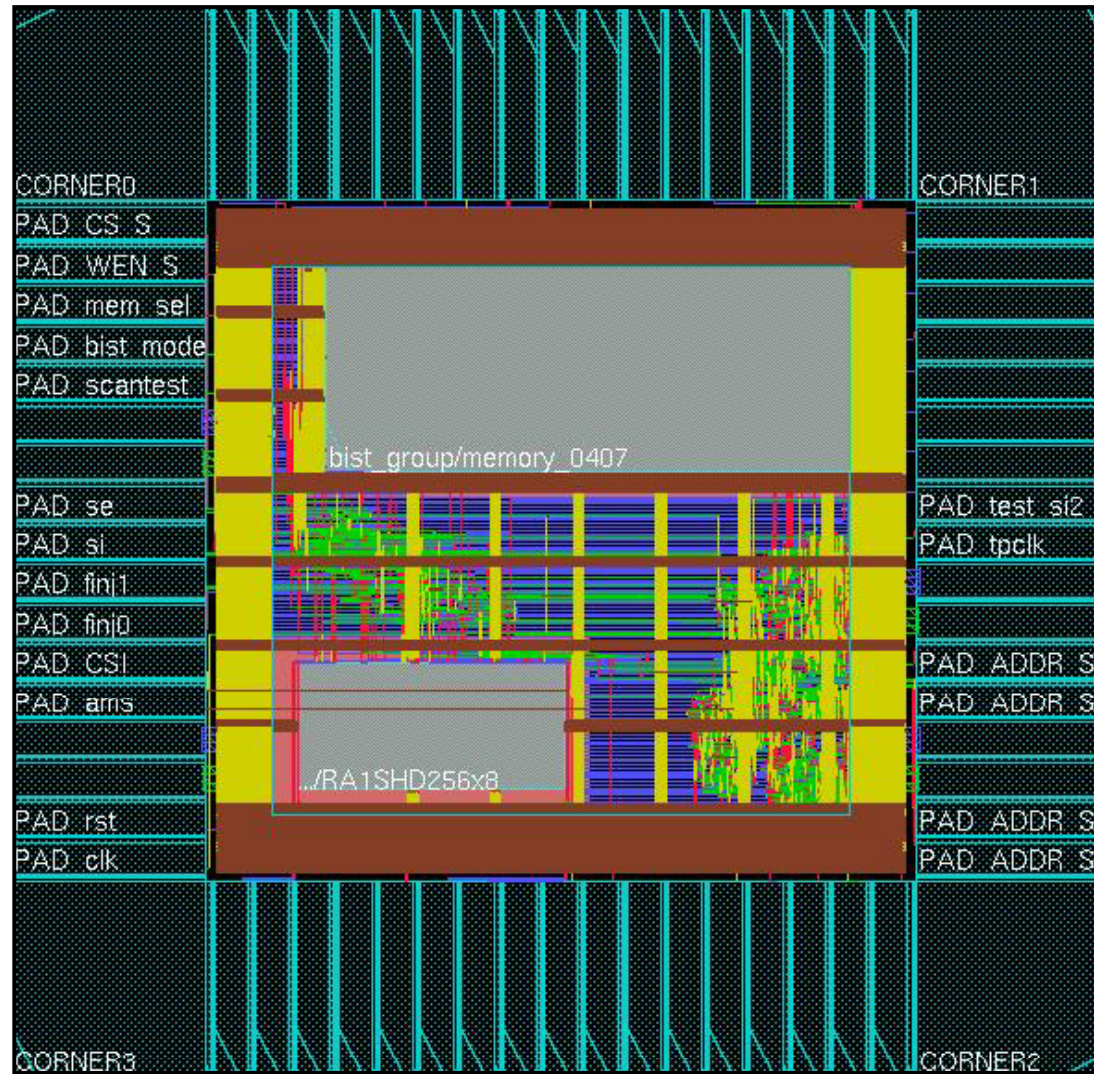
## ☐ *Route/NanoRoute*

The screenshot shows the 'Mode' tab of the NanoRoute/Attributes dialog. The 'Global Route' checkbox is selected. The 'Detail Route' checkbox is also selected, with 'Start Iteration' and 'End Iteration' both set to 'default'. The 'DFM' checkbox is unselected. Under 'Concurrent Routing Features', 'Timing Driven' and 'SI Driven' are both selected and highlighted with red boxes. 'Post Route SI' is unselected. The 'Insert Diodes' checkbox is unselected, with 'Diode Cell Name' and 'Fill Cells' fields. The 'Congestion' and 'Timing' sliders are visible, with 'Timing' set to 'S.M.A.R.T.'. The 'SI' slider is set to 'normal'. The 'SI Victim File' field is empty.

The screenshot shows the 'Net Attributes' tab of the NanoRoute/Attributes dialog. The 'Net Type(s)' dropdown is set to 'Clock Nets' and is highlighted with a red box. The 'Net Name(s)' field is empty. The 'Skip Antenna' checkbox is unselected, with 'TRUE' and 'FALSE' options. The 'Skip Routing' checkbox is unselected, with 'TRUE' and 'FALSE' options. The 'Avoid Detour' checkbox is selected and highlighted with a red box, with 'TRUE' and 'FALSE' options. The 'SI Prevention' checkbox is unselected, with 'TRUE' and 'FALSE' options. The 'SI Post Route Fix' checkbox is unselected, with 'TRUE' and 'FALSE' options. The 'Top Layer' and 'Bottom Layer' are both set to 'ASIS'. The 'Weight' is set to '10' and the 'Spacing' is set to '1', both highlighted with a red box. The 'Shield Net(s)' is set to 'ASIS'. The 'Nondefault Rule' is set to 'ASIS'. The 'Pattern' is set to 'ASIS'. The 'OK', 'Apply', 'Select', 'Cancel', and 'Help' buttons are at the bottom.

# Example for Nano Route

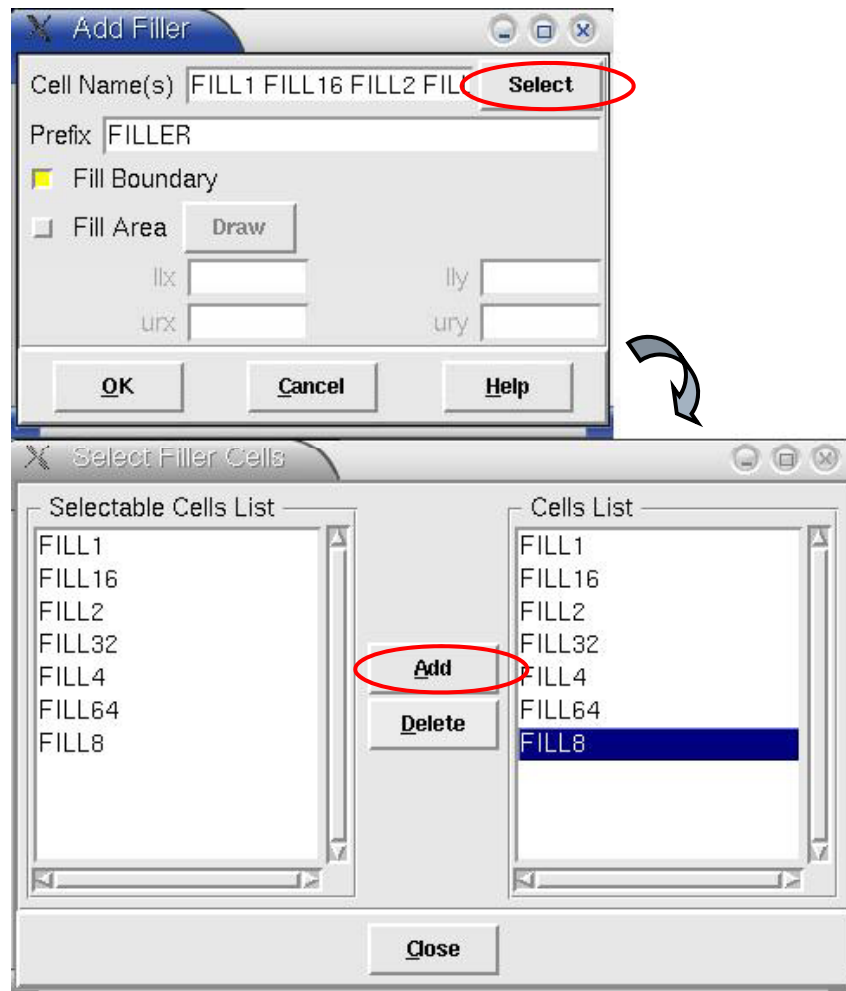
---



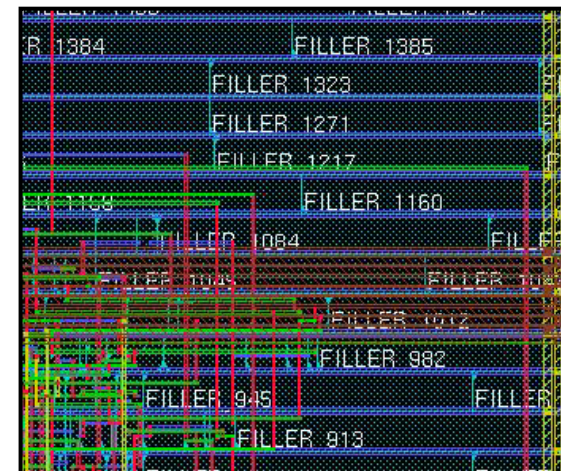


# Cell Filler

## □ Place/Filler/Add Filler



Ex:



# Save Design

---

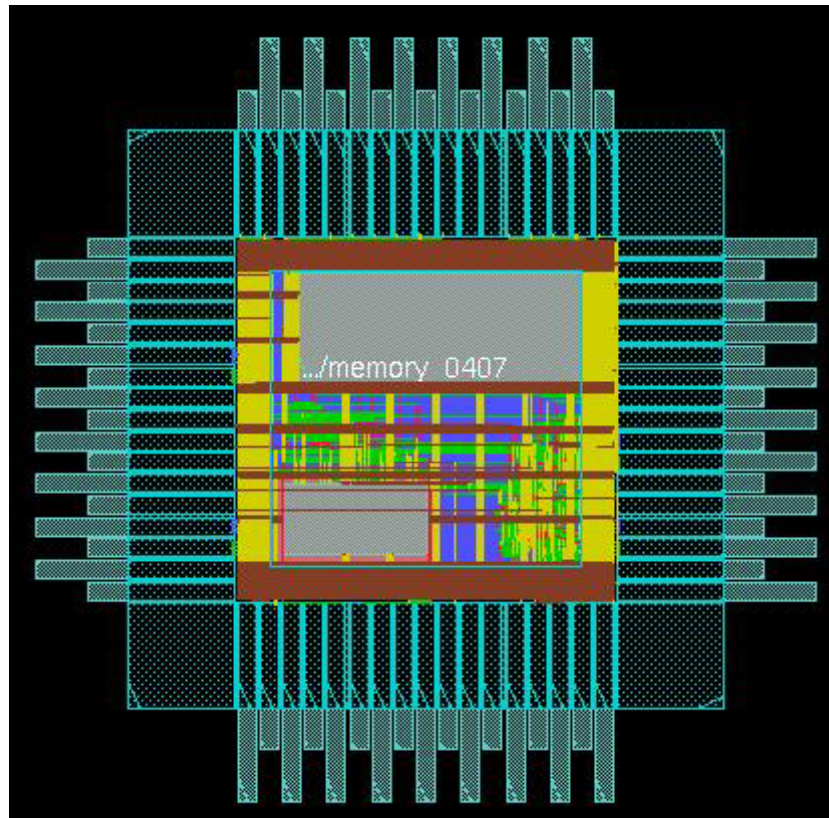
- ☐ *Design/Save/Netlist → \*.v*
- ☐ *Timing/Calculate Delay → \*.sdf*
- ☐ *Design/Save/DEF → \*.def*
  - *SELECT “Save Scan”*

# Bounding PAD

---

- ❑ `unix %> chmod 755 addbonding.pl`
- ❑ `unix %> /usr/bin/perl addbonding.pl CHIP.def`
- ❑ `encounter %> source bondPads.cmd`

Ex:





# Save GDSII

---

- *Design/Save/GDS → \*.gds*

---

# LAB