Cell-Based IC Physical Design and Verification
- SOC Encounter

張年翔　　CIC2004/07

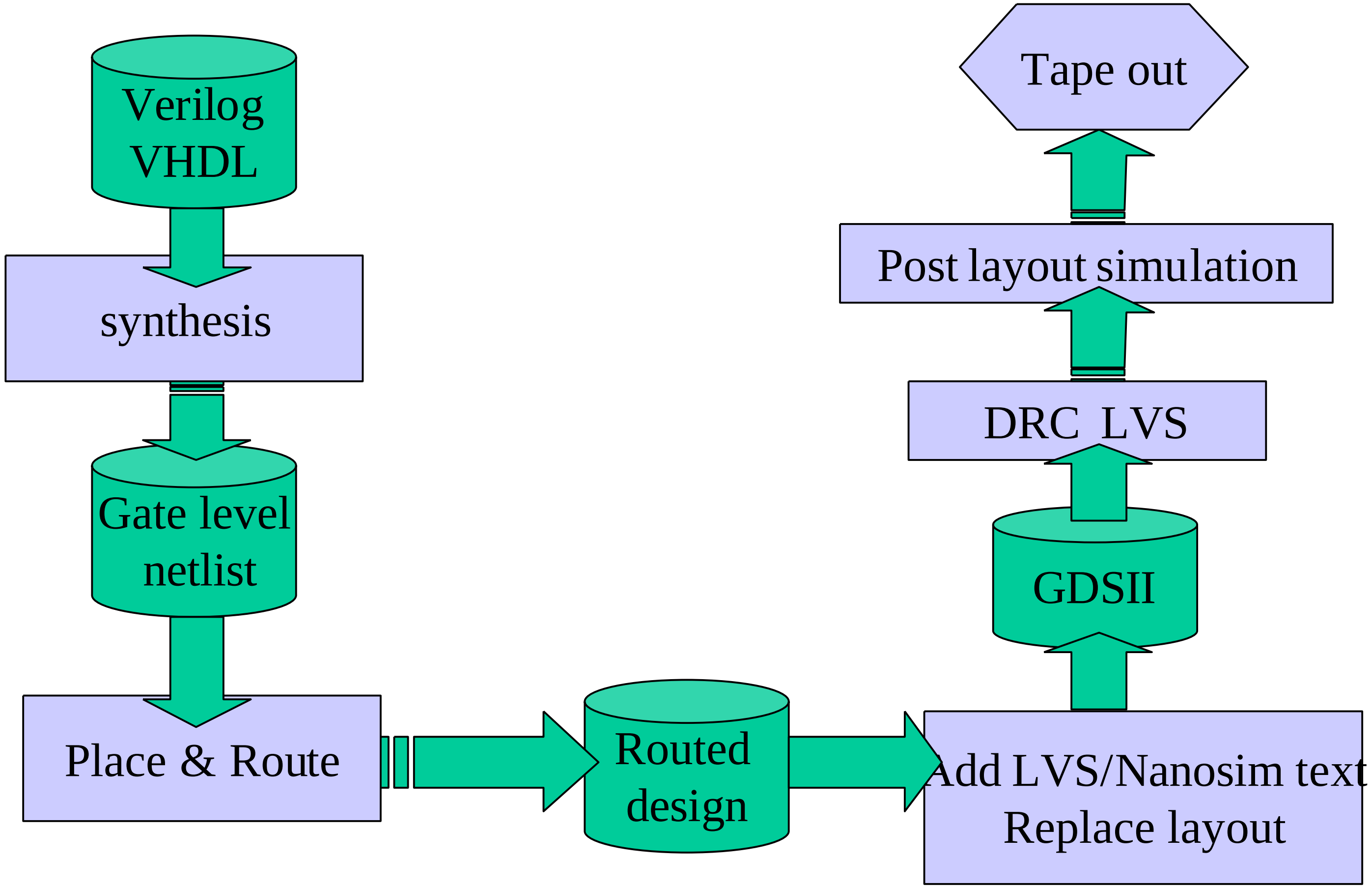
Class Schedule

- Day1
  - Design Flow Over View
  - Prepare Data
  - Getting Started
  - Importing Design
  - Specify Floorplan
  - Power Planning
  - Placement
  - Synthesize Clock Tree
- Day2
  - Timing Analysis
  - Trial Route
  - Power Analysis
  - SRoute
  - NanoRoute
  - Fill Filler
  - Output Data
  - DRC
  - LVS
  - extraction/nanosim

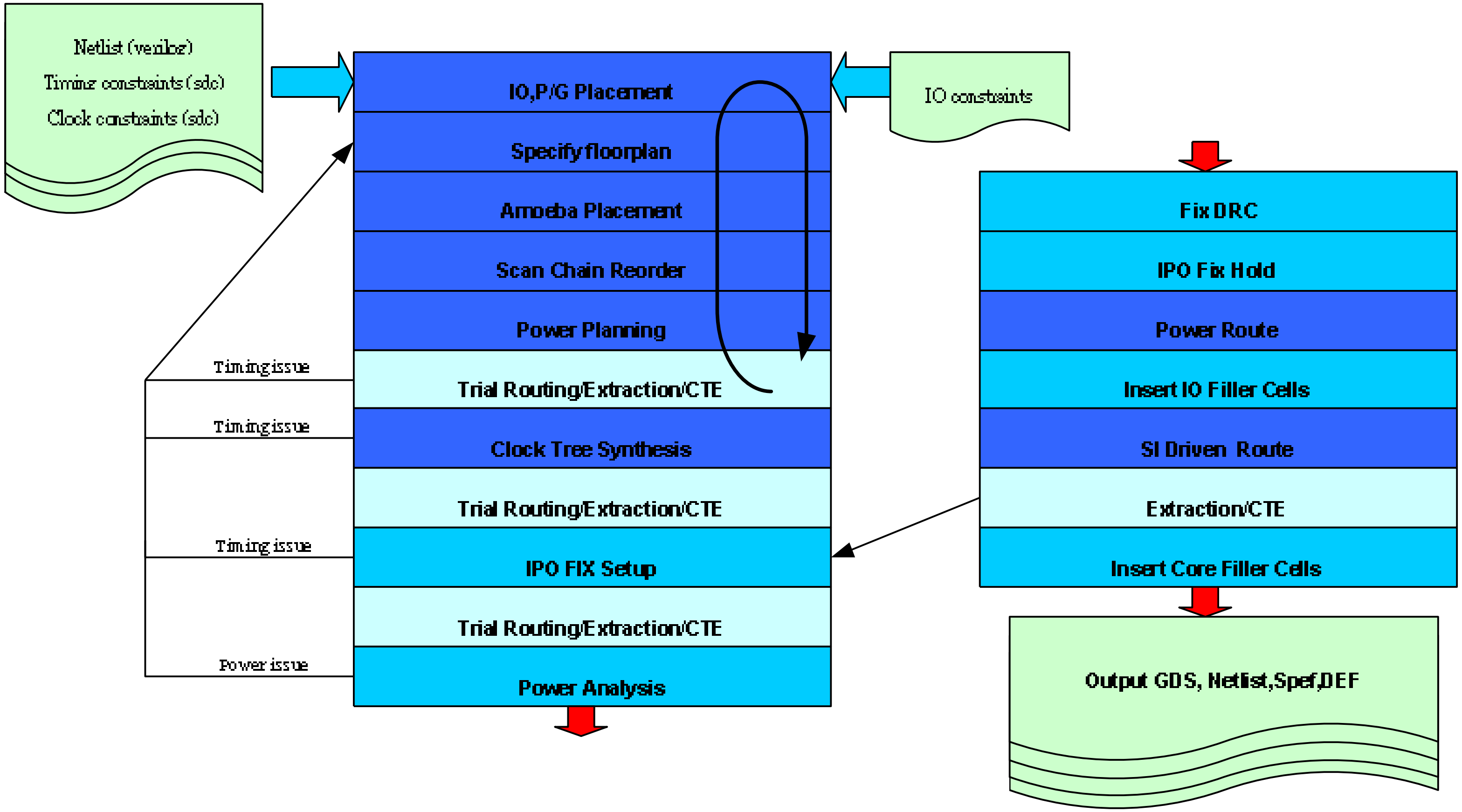
Chapter1

Cell-Based Physical Design – SOC Encounter 3.2

Cell-Based Design Flow



SOC Encounter P&R flow



Netlist (verilog)
Timing constraints (sdc)
Clock constraints (sdc)

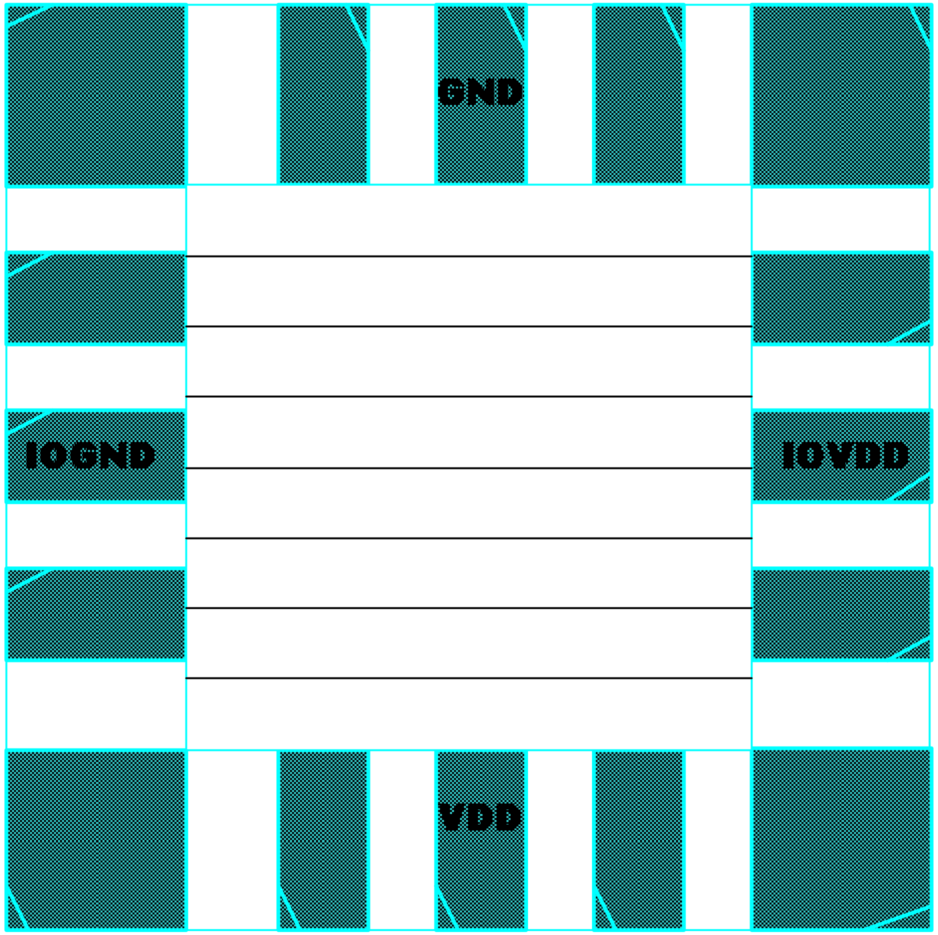
IO constraints

- IO,P/G Placement
- Specify floorplan
- Amoeba Placement
- Scan Chain Reorder
- Power Planning
- Trial Routing/Extraction/CTE
- Clock Tree Synthesis
- Trial Routing/Extraction/CTE
- IPO FIX Setup
- Trial Routing/Extraction/CTE
- Power Analysis

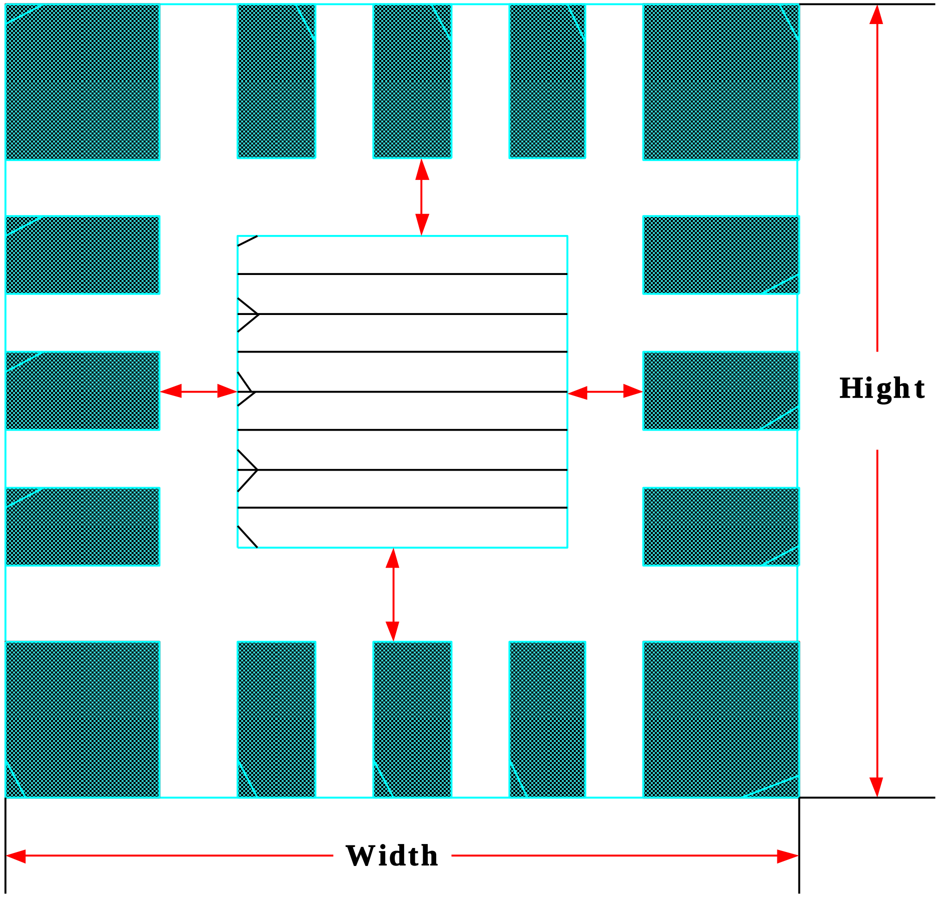
- Fix DRC
- IPO Fix Hold
- Power Route
- Insert IO Filler Cells
- SI Driven Route
- Extraction/CTE
- Insert Core Filler Cells

Output GDS, Netlist,Spef,DEF

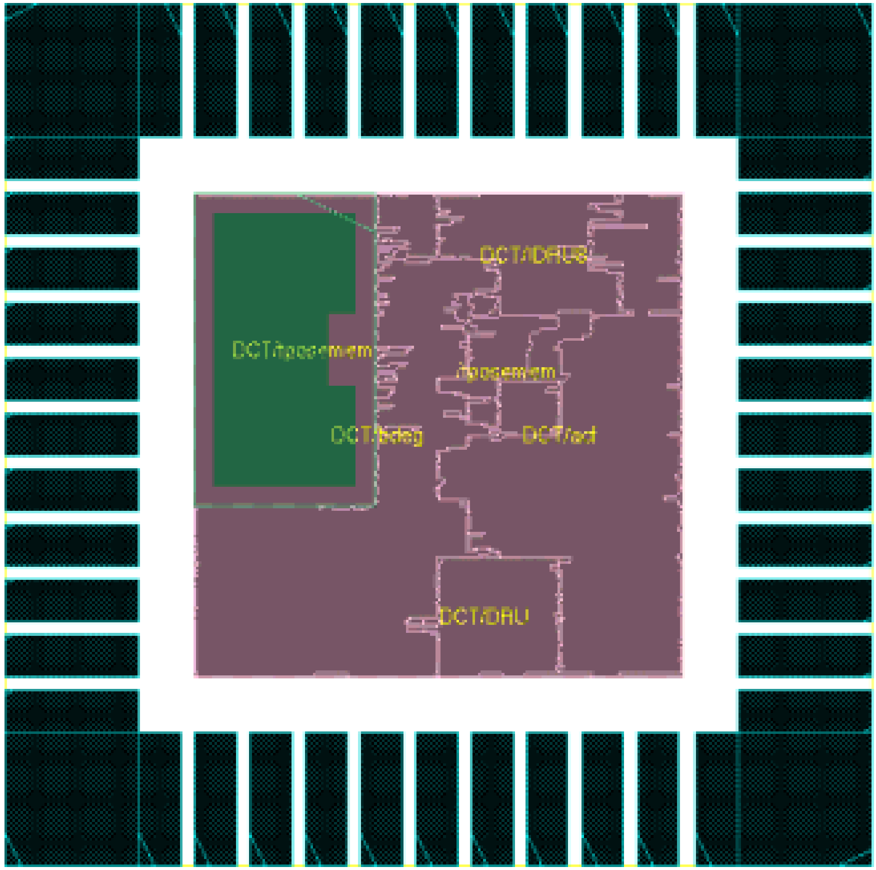
IO, P/G Placement



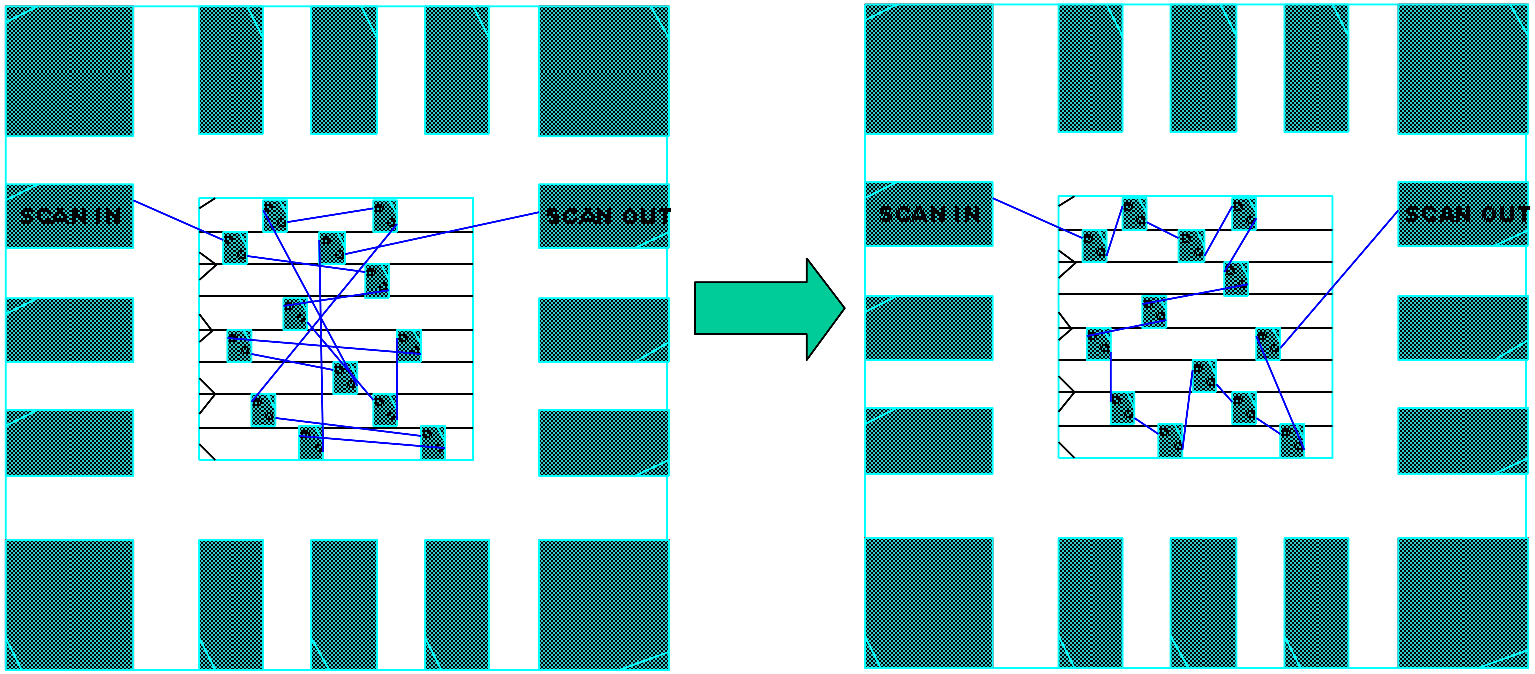
Specify Floorplan



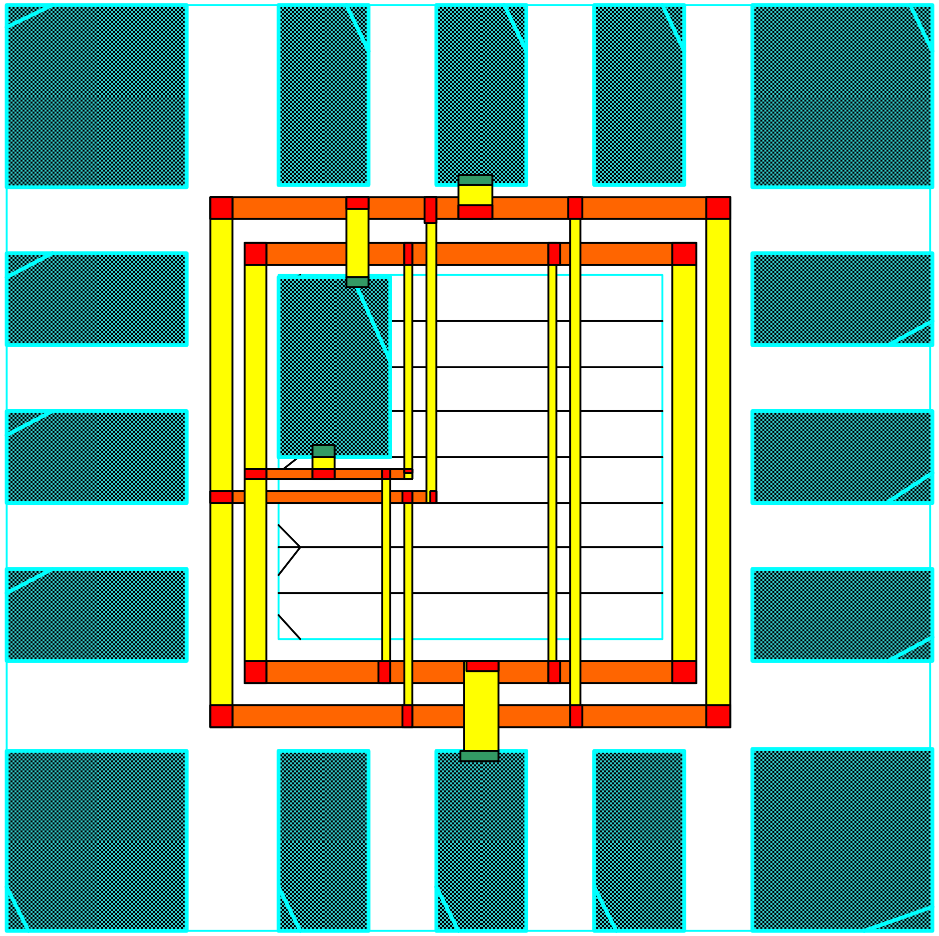
Amoeba Placement



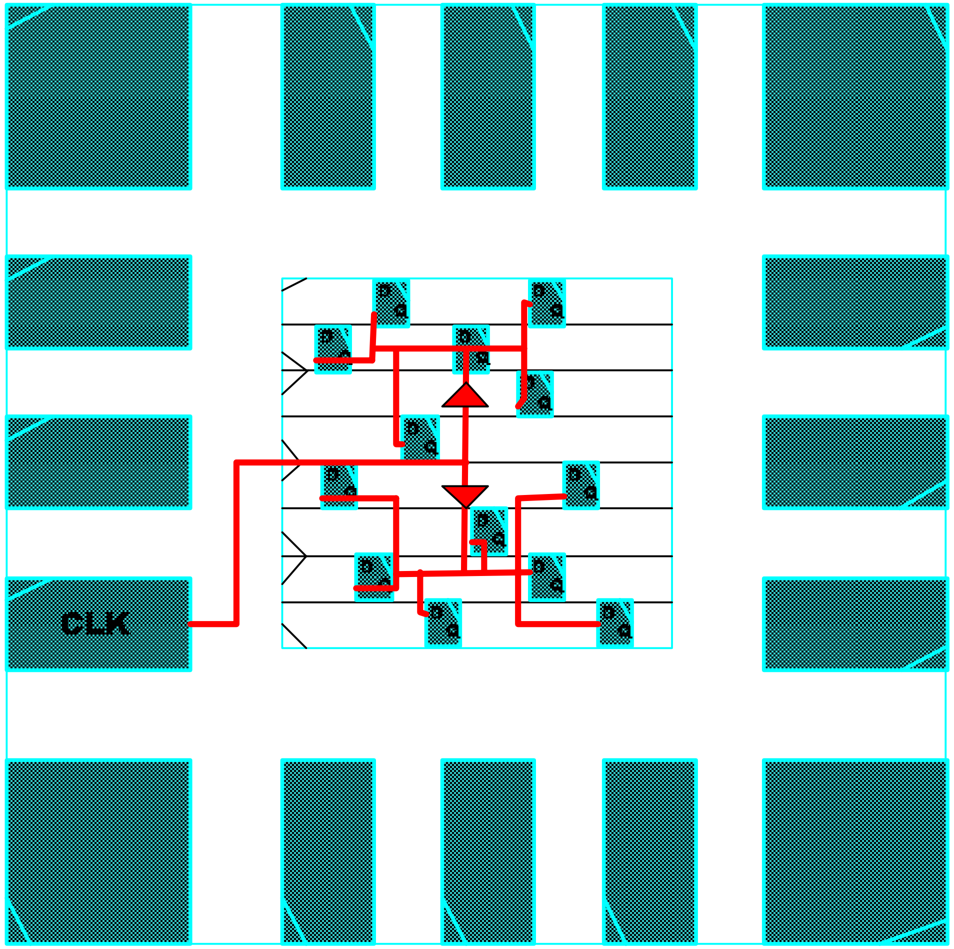
Scan Chain Reorder



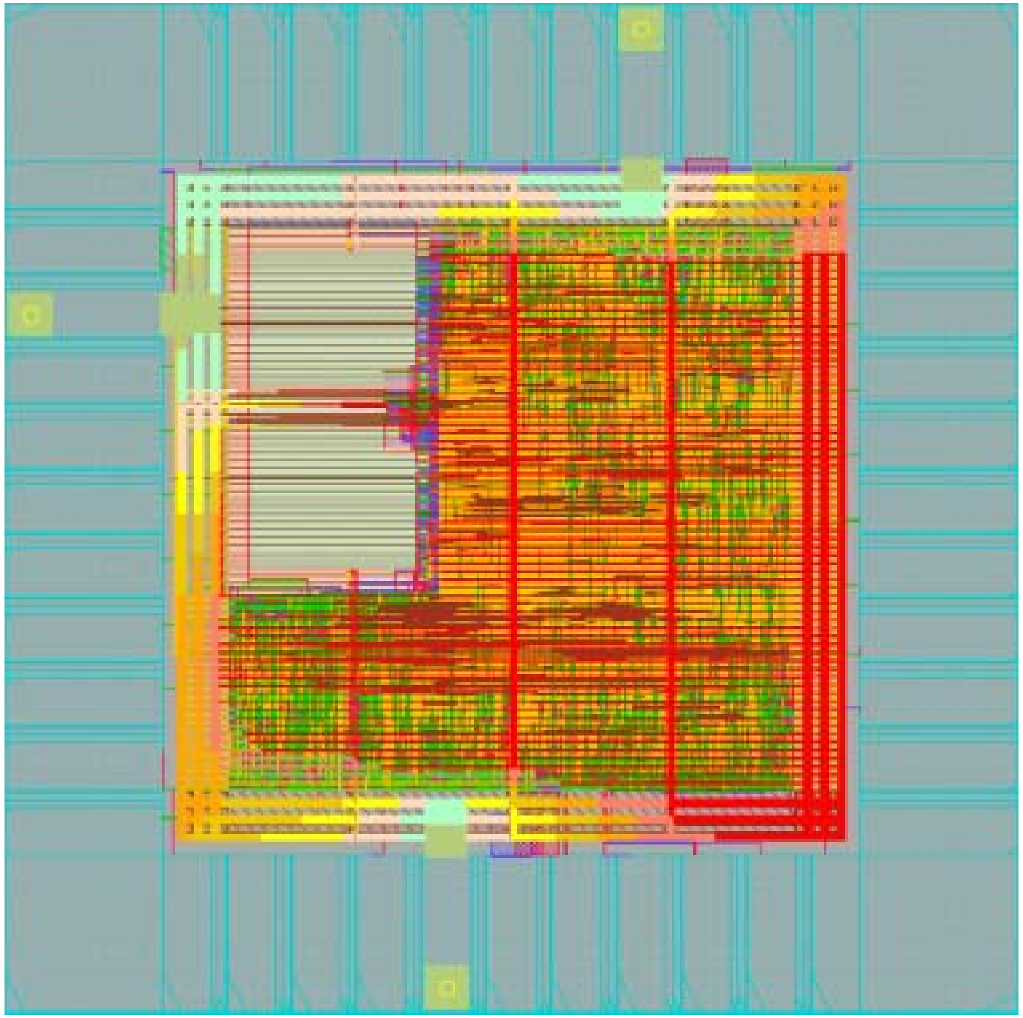
Power Planning



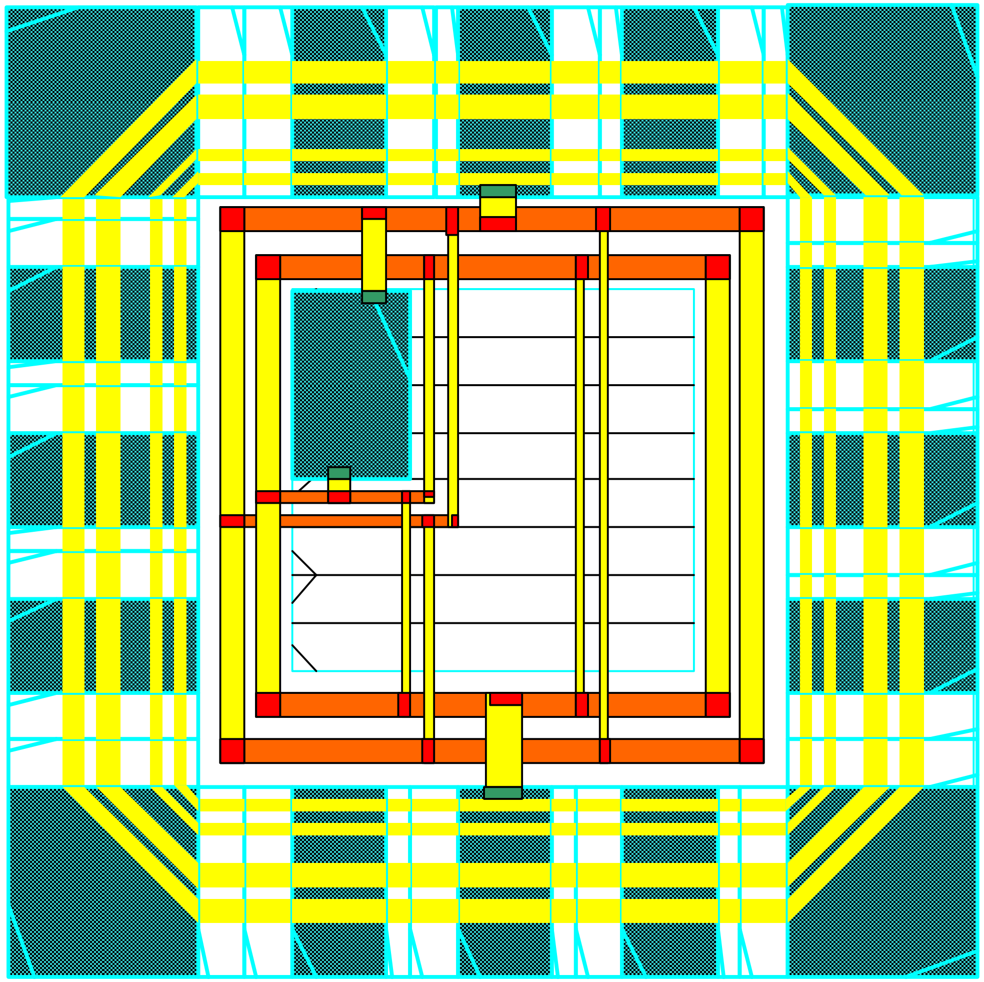
Clock Tree Synthesis



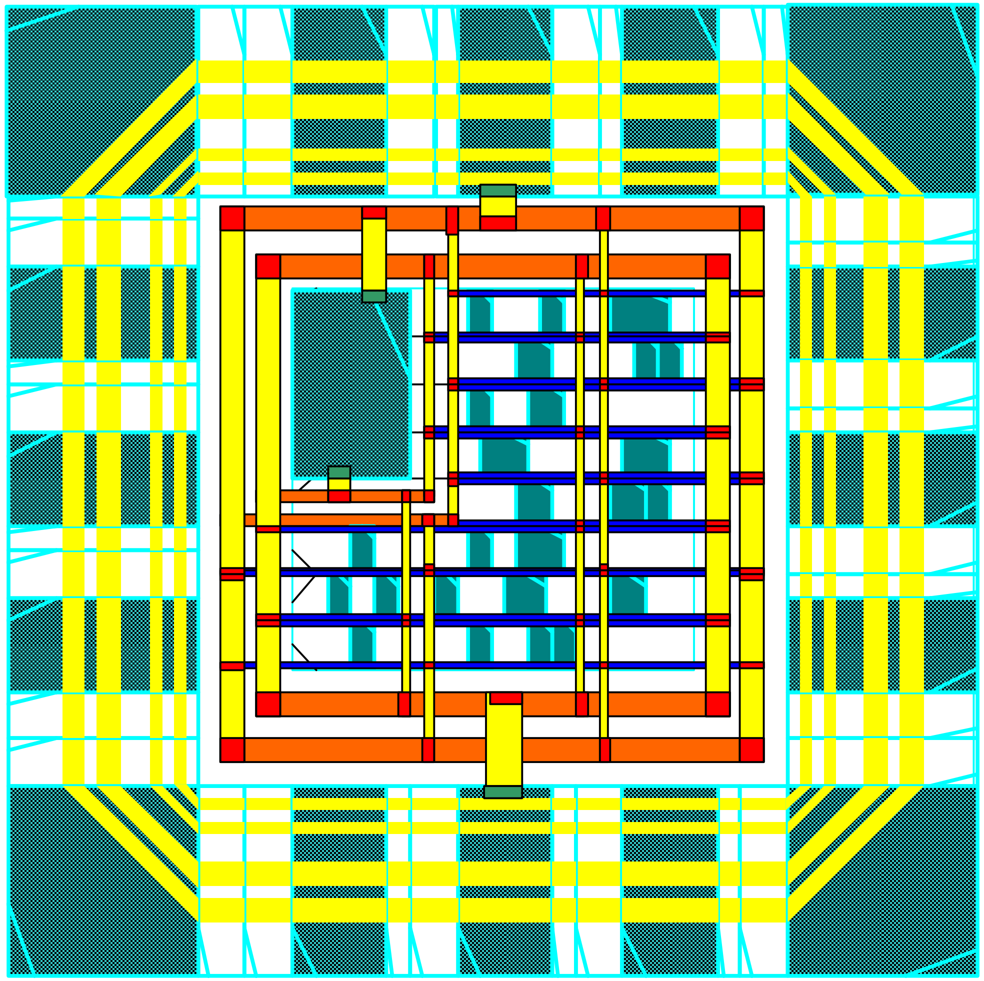
Power Analysis



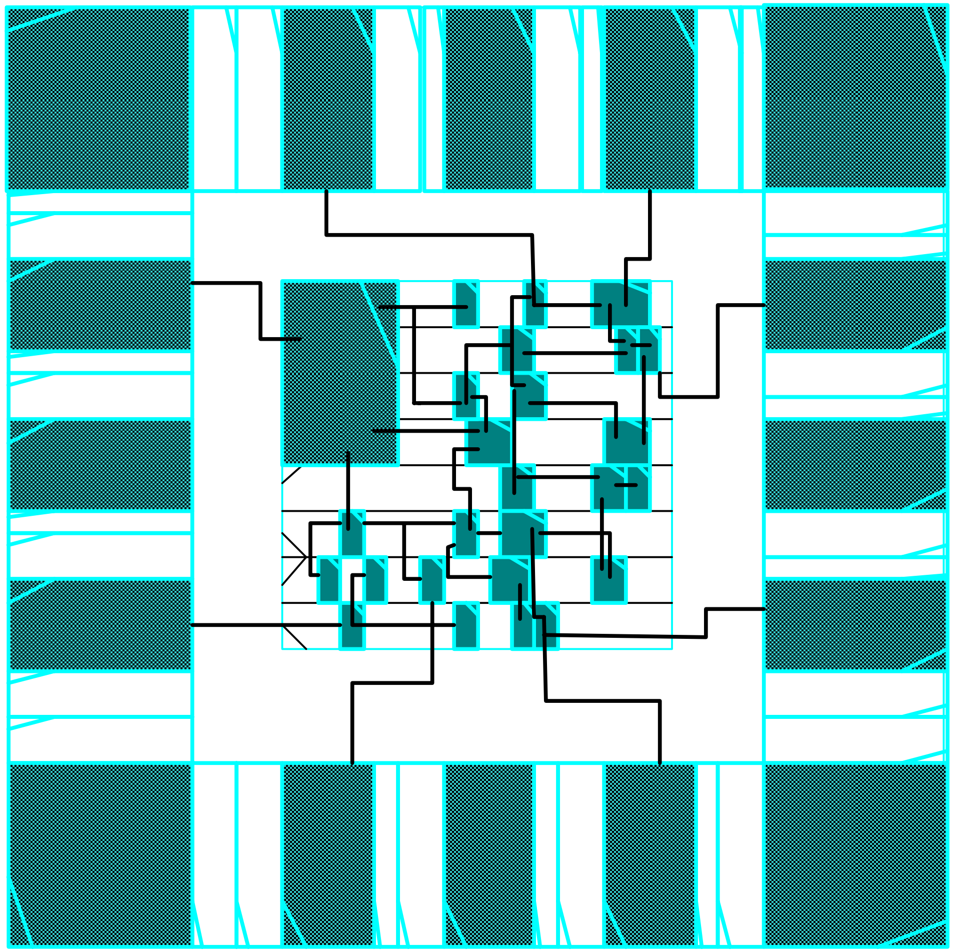
Add Filler



Power Route



Routing



Prepare Data

- ◆ Gate-Level netlist (verilog)
- ◆ Physical Library (LEF)
- ◆ Timing Library (LIB)
- ◆ Timing constraints (sdc)
- ◆ IO constraint

Preparing Data : 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.
  - Use the synthesis command below to remove assign statement.

    ```
    set_boundary_optimization
    ```

  - Use the synthesis commands below to remove “*cell*” cell name

    ```
    define_name_rules name_rule –map {{\\*cell\\* cell”}}
    change_names –hierarchy –output name_rule
    ```

Prepare Data : LEF
-- Process Technology

Layers

POLY

◆Contact

Metal1

◆ Via1

Metal2

Design Rule

Net width
Net spacing
Area
Enclosure
Wide metal
slot
Antenna
Current density

Parasitic

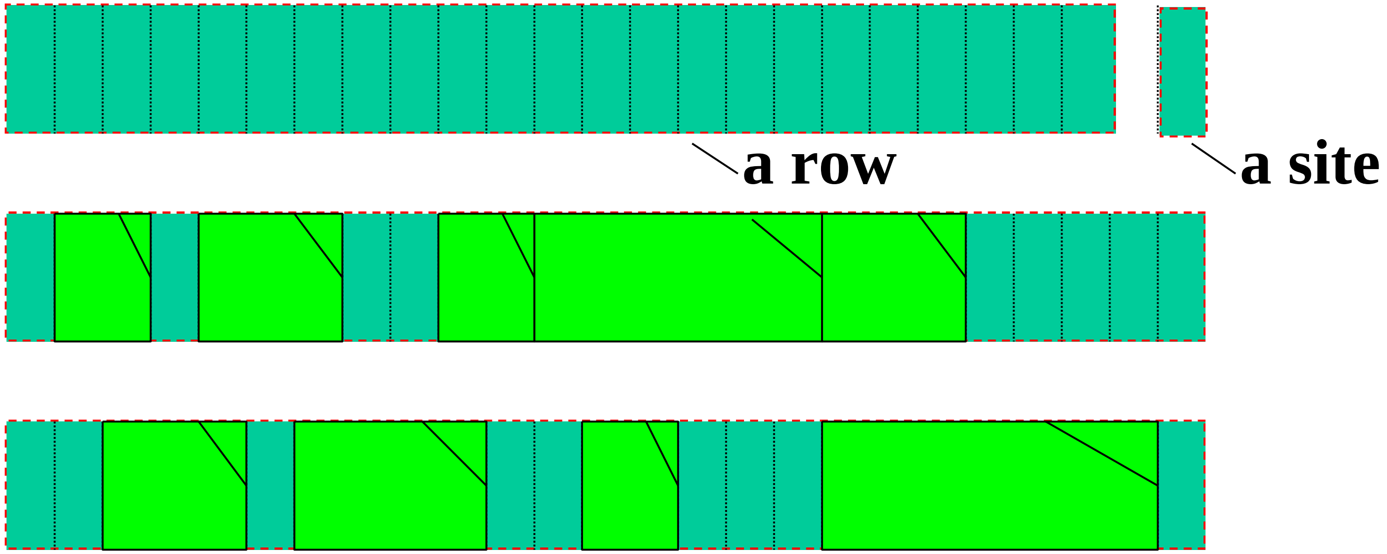
Resistance
Capacitance

Prepare Data: LEF
-- APR technology

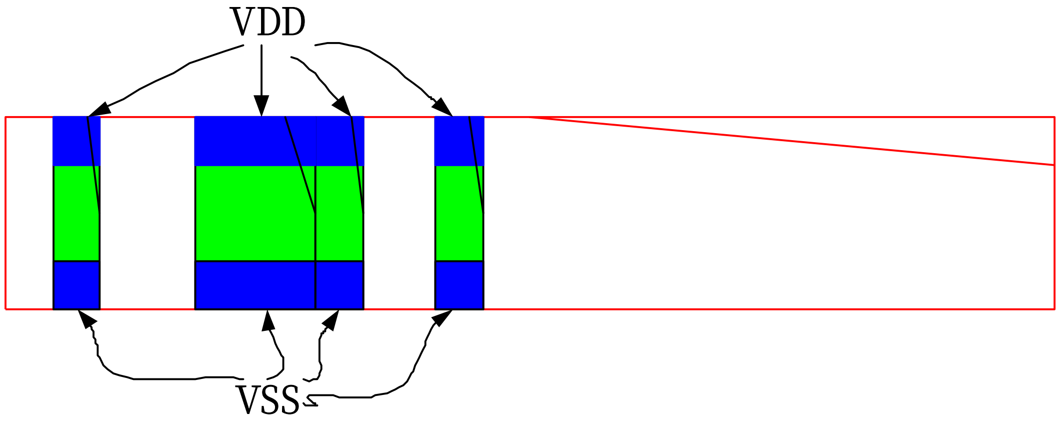
- Unit
- Site
- Routing pitch
- Default direction
- Via generate
- Via stack

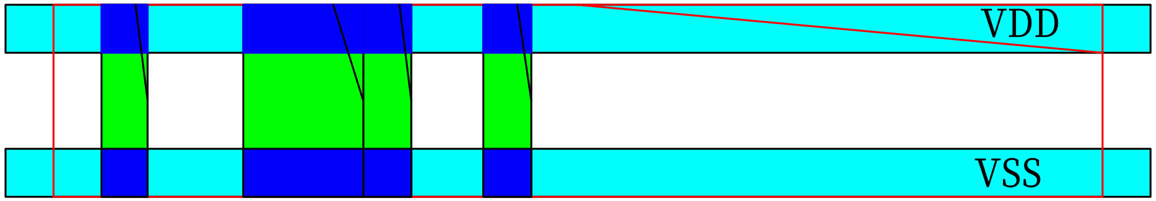
Prepare Data: LEF
-- APR technology : SITE

- The Placement site give the placement grid of a family of macros

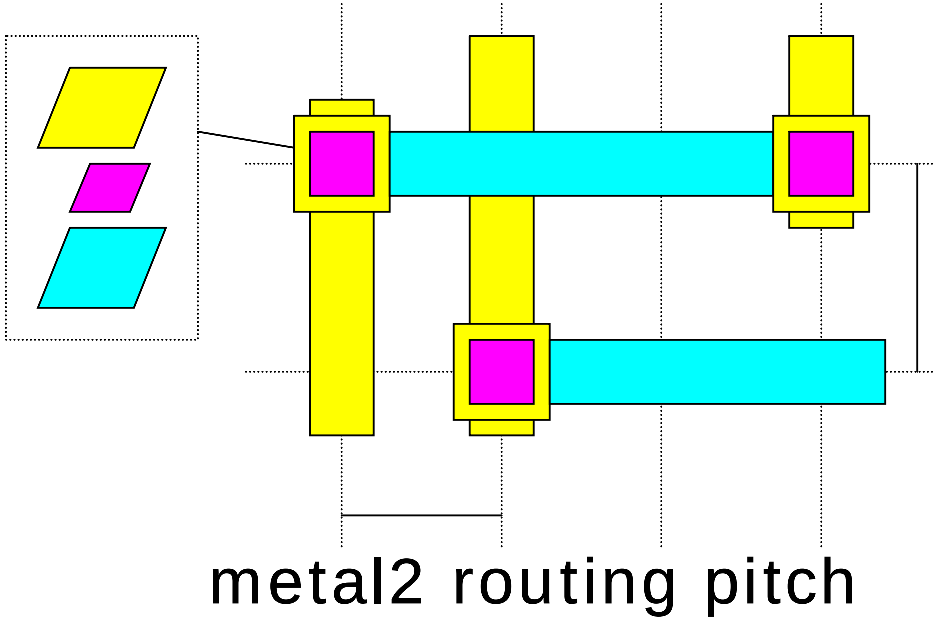


Row Based PR





Prepare Data: LEF
-- APR technology :
routing pitch , default direction

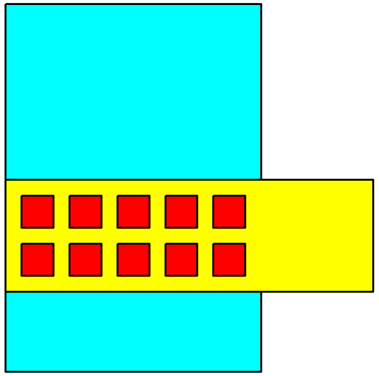


Horizontal routing	Vertical routing
Metal1 Metal3 Metal5	Metal2 Metal4 Metal6

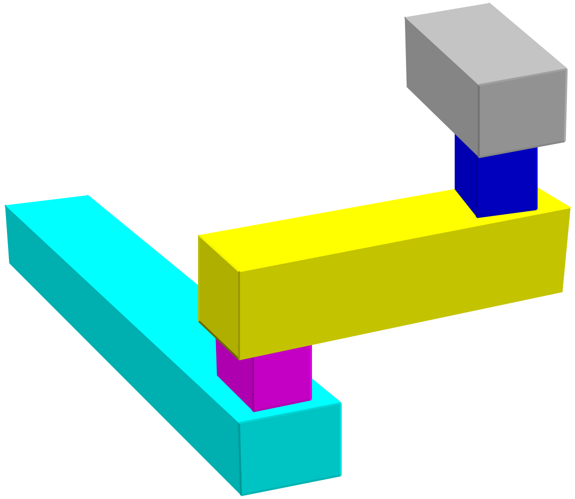
Prepare Data: LEF
-- APR technology : via generate

- To connect wide metal , create a via array to reduce via resistance
- Defines formulas for generating via arrays

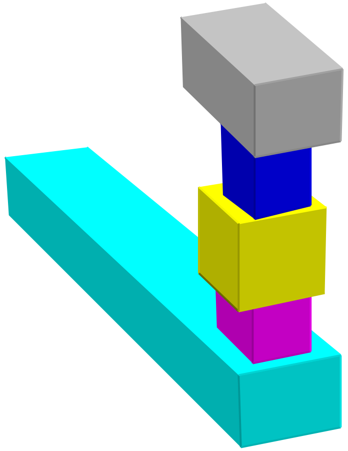
```
Layer Metal1
   Direction HORIZONTAL
   OVERHANG 0.2
Layer Metal2
   Direction VERTICAL
   OVERHANG 0.2
Layer Via1
   RECT –0.14 –0.14 0.14 0.14
   SPACING 0.56 BY 0.56
```



Prepare Data: LEF
-- APR technology : via stack



Without via stack



With via stack

◆ Higher density routing

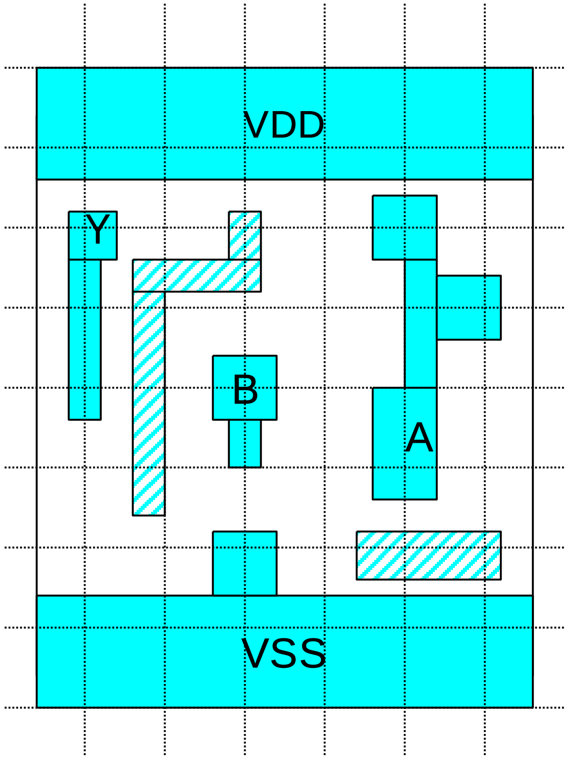
◆ Easier usage of upper layer

◆ Must Follow minimum area rule

Prepare Data: LEF
-- APR technology : Physical Macros

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

Prepare Data: LEF
-- APR technology : Physical Macros cont.



```
MACRO ADD1
   CLASS CORE ;
   FOREIGN ADD1 0.0 0.0 ;
   ORIGEN 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
   OBS
      ......
   END
END ADD1
```

Prepare Data: LIB

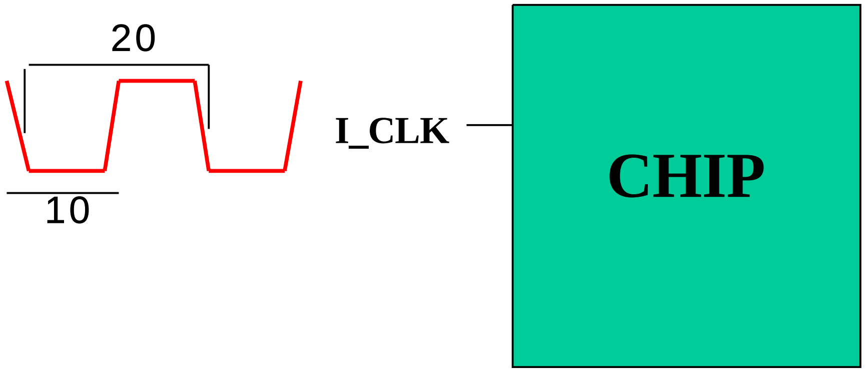
- Operating condition
  - slow, fast, typical
- Pin type
  - input/output/inout
  - function
  - data/clock
  - capacitance
- Path delay
- Timing constraint
  - setup, hold, mpwh, mpwl, recovery

Prepare Data: Timing constraint

- ◆ Create clock
- ◆ Input delay
- ◆ Output delay
- ◆ Input drive
- ◆ Output loading

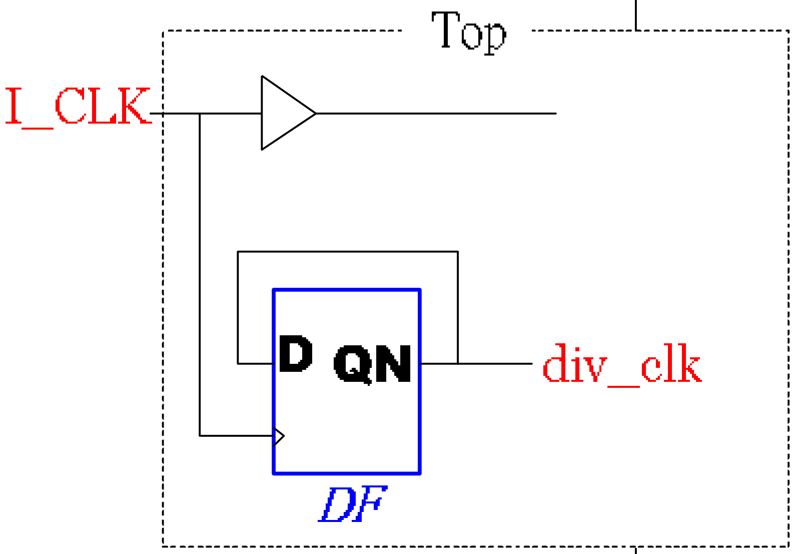
Prepare Data: Timing constraint -- Create Clock

```
create_clock  [-name clock_name]
              -period period_value
              [-waveform edge_list]
              [clock_source_list]
```



create_clock –name CLK1 –period 20 –waveform {0 10} [get_ports I_CLK]

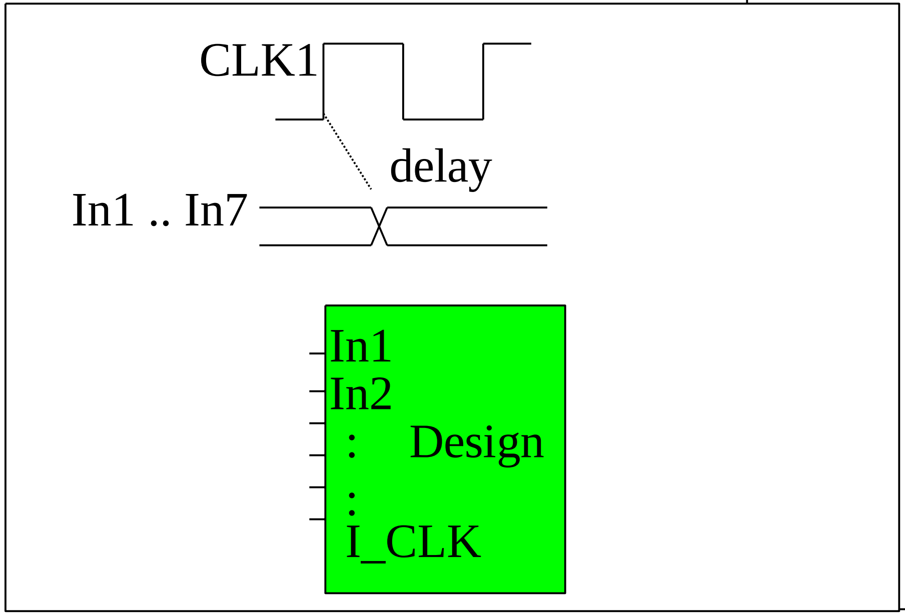
Prepare Data: Timing constraint -- create_generated_clock



```
create_generated_clock  -add  -master_clock
                        [-name clock_name]
                        -source master_clock_root
                        [-multiply_by mult]
                        [-divide_by div]
                        [-duty_cycle dc]
                        [-invert]
                        [-edges edge_list]
                        [-edge_shift edge_shift_list]
                        clock_root_list
```

create_generated_clock –name CLK2 –source [get_ports I_CLK] –divide_by 2 [get_pins DF/QN]

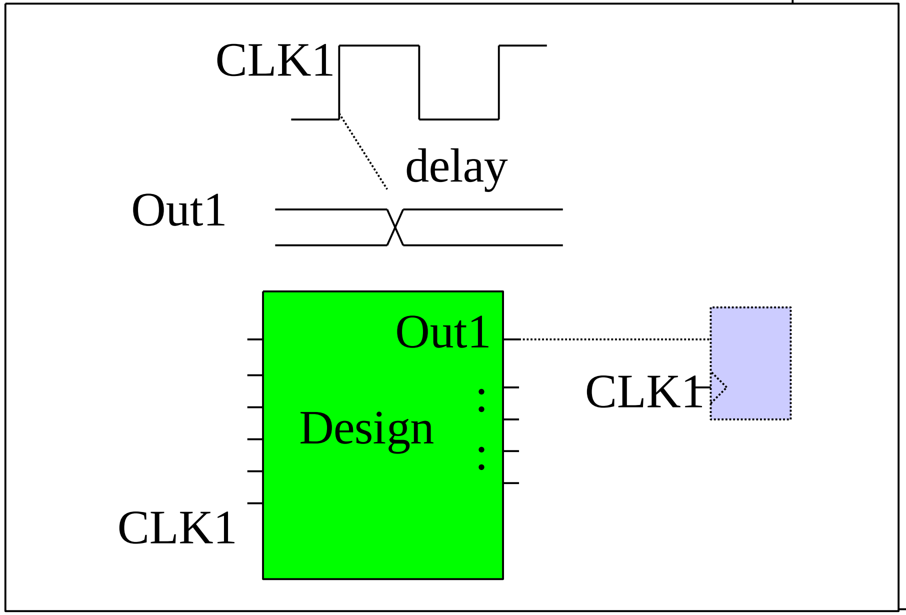
Prepare Data: Timing constraint --set_input_delay



```
set_input_delay  delay_value
                 [-min]
                 [-max]
                 [-rise]
                 [-fall]
                 [-clock clock_name]
                 [-clock_fall]
                 [-add_delay]
                 [-network_latency_included]
                 [-source_latency_included]
                 port_pin_list
```

set_input_delay 1 –clock [get_clocks {CLK1}] [getports {In1}]

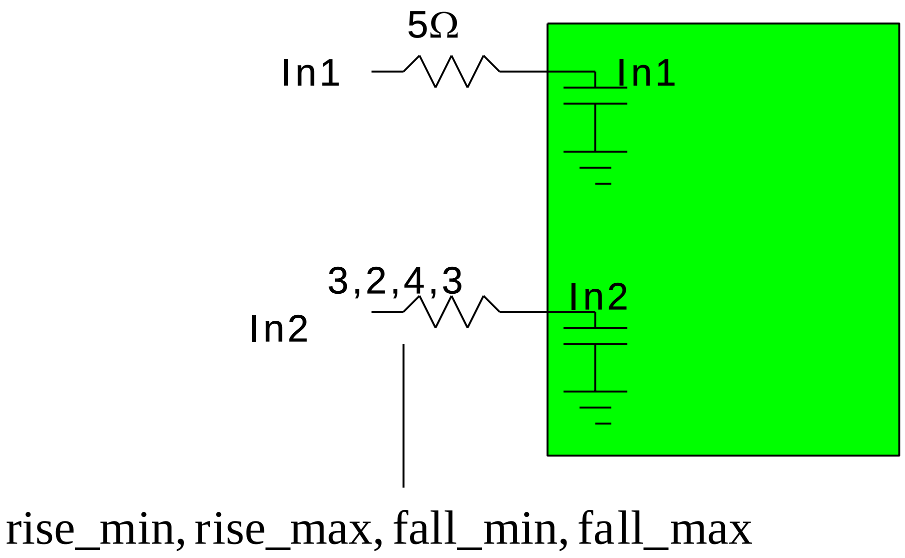
Prepare Data: Timing constraint --set_output_delay



```
set_output_delay  delay_value
                  [-min]
                  [-max]
                  [-rise]
                  [-fall]
                  [-clock clock_name]
                  [-add_delay]
                  [-network_latency_included]
                  [-source_latency_included]
                  port_pin_list
```

set_output_delay 1 –clock [get_clocks {CLK1}] [getports {Out1}]

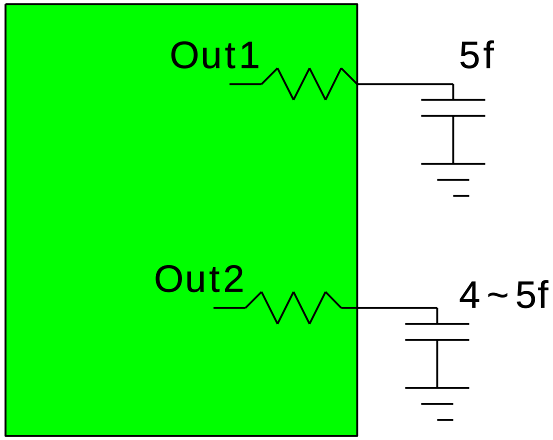
Prepare Data: Timing constraint --set_drive



```
set_drive  [-min]
           [-max]
           [-rise]
           [-fall]
           drive_strength
           port_list
```

set_drive 1 [get_ports {In1}]

Prepare Data: Timing constraint --set_load



```
set_load  [-min]
          [-max]
          [-pin_load]
          [-wire_load]
          load_value
          port_list
```

set_load 1 [get_ports {Out1}]

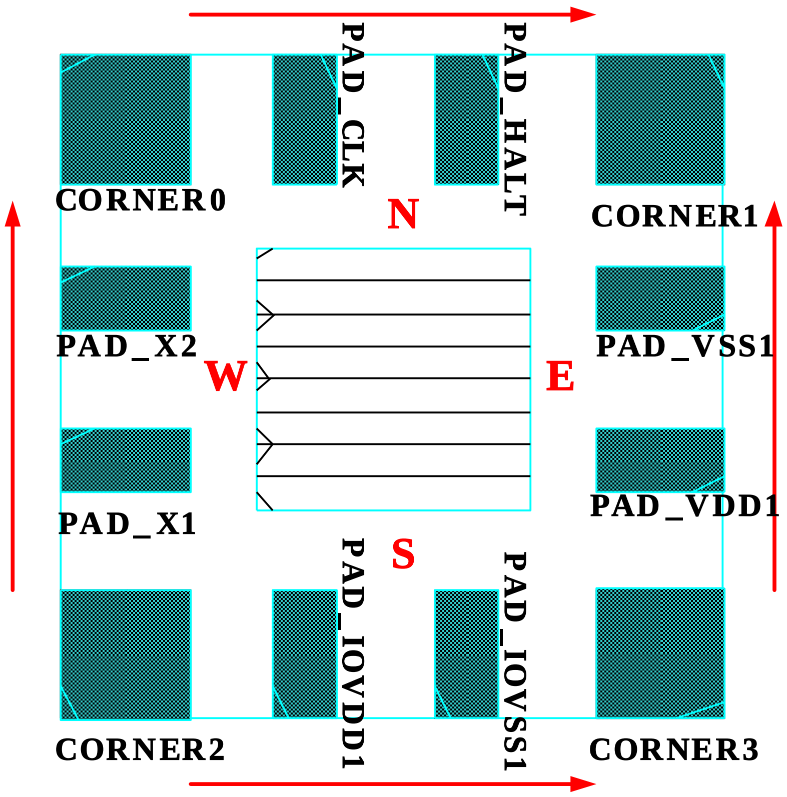
Prepare Data: IO constraint

- ◆Create an I/O assignment file manualy using the following template:

```
Version: 1
MicronPerUserUnit: value
Pin: pinName side |corner
Pad: padInstanceName side|corner [cellName]
Offset: length
Skip:length
Spacing: length
Keepclear: side offset1 offset2
```

Prepare Data: IO constraint cont.

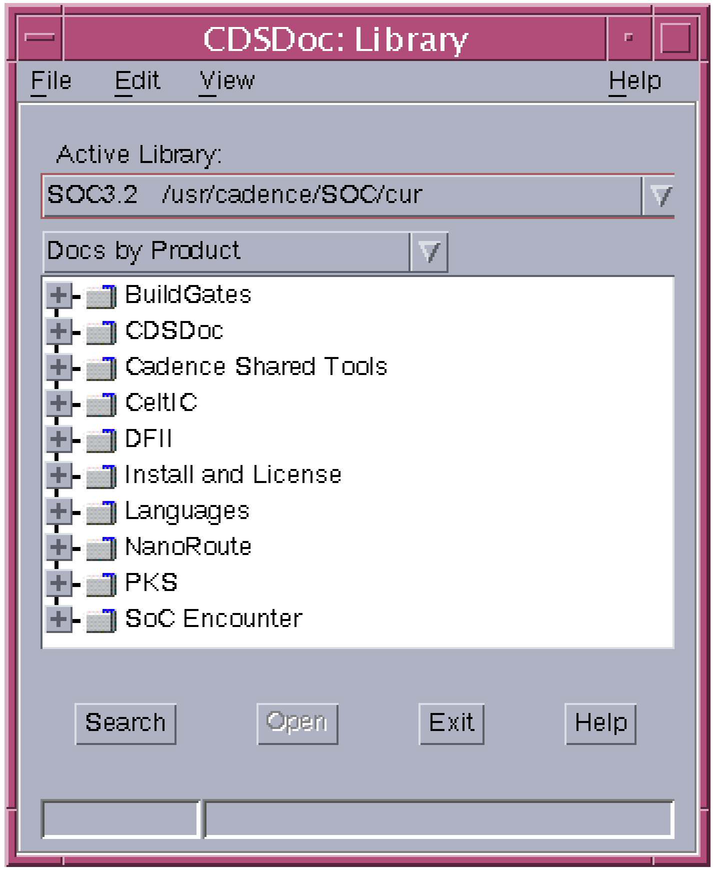
```
Version: 1
Pad: CORNER0 NW
Pad: PAD_CLK        N
Pad: PAD_HALT       N

Pad: CORNER1        NE
Pad: PAD_X1         W
Pad: PAD_X2         W

Pad: CORNER2        SW
Pad: PAD_IOVDD1     S
Pad: PAD_IOVSS1     S

Pad: CORNER3        SE
Pad: PAD_VDD1       E
Pad: PAD_VSS1       E
```



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
- ◆ Consider using double bonding on the same power pad to reduce inductance

Cadence On-Line document



unix% */usr/cadence/SOC/cur/tools/bin/cdsdoc &*

unix% */usr/cadence/IC/cur/tools/bin/cdsdoc &*

unix% */usr/cadence/LDV/cur/tools/bin/cdsdoc &*

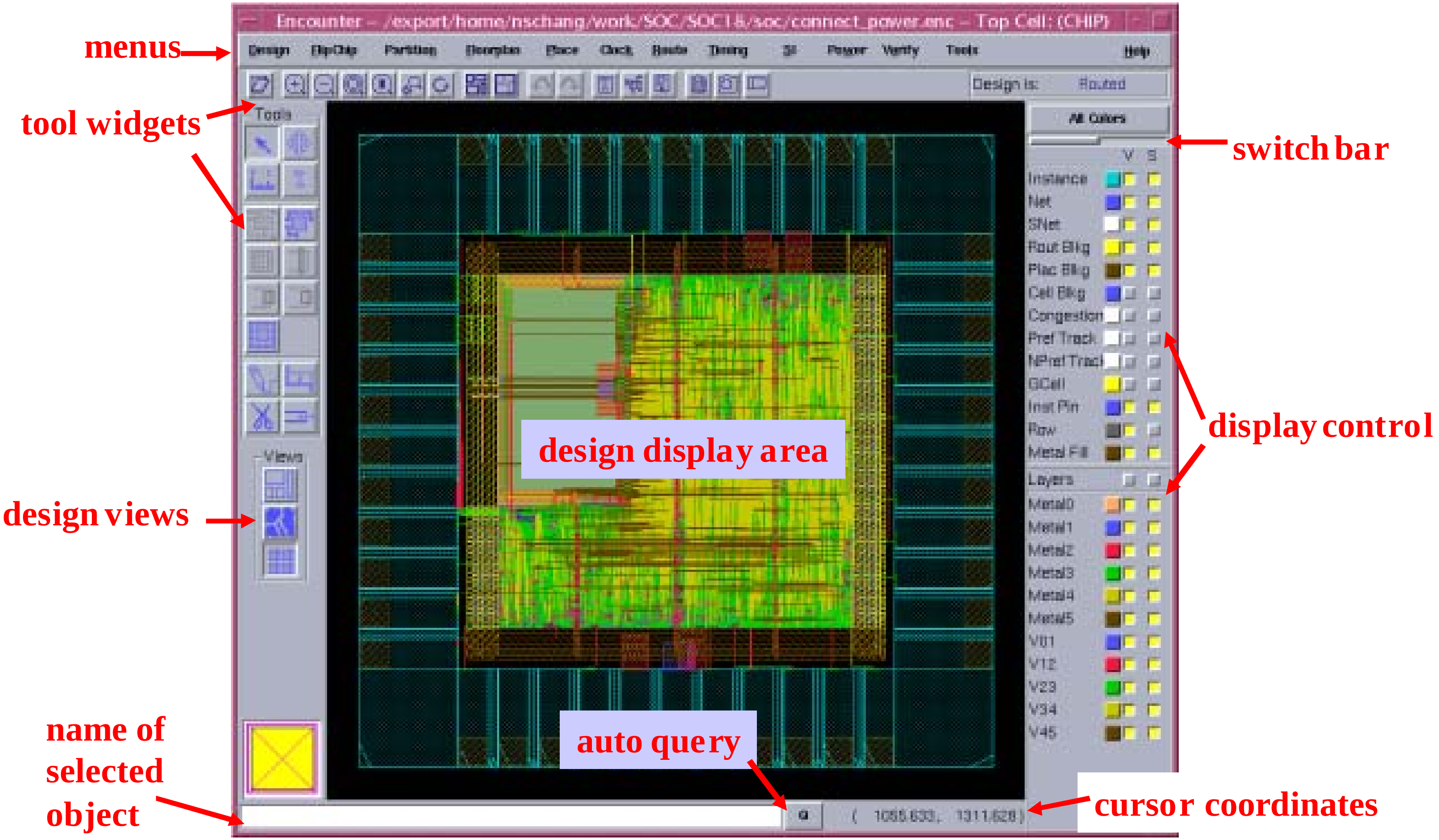
…..

- html browser must be installed
- do not set the proxy in html browser

Getting Started

- Source the encounter environment:
  unix% *source /usr/cadence/cic_setup/soc.csh*
- Invoke soc encounter in 64 bit mode:
  unix% *encounter -64*
- Do not run in background mode. Because the terminal become the interface of command input while running soc encounter.
- The Encounter reads the following initialization files:
  - $ENCOUNTER/etc/enc.tcl
  - ./enc.tcl
  - ./enc.pref.tcl
- Log file:
  - encounter.log*
  - encounter.cmd*

GUI



menus

tool widgets

switch bar

display control

design display area

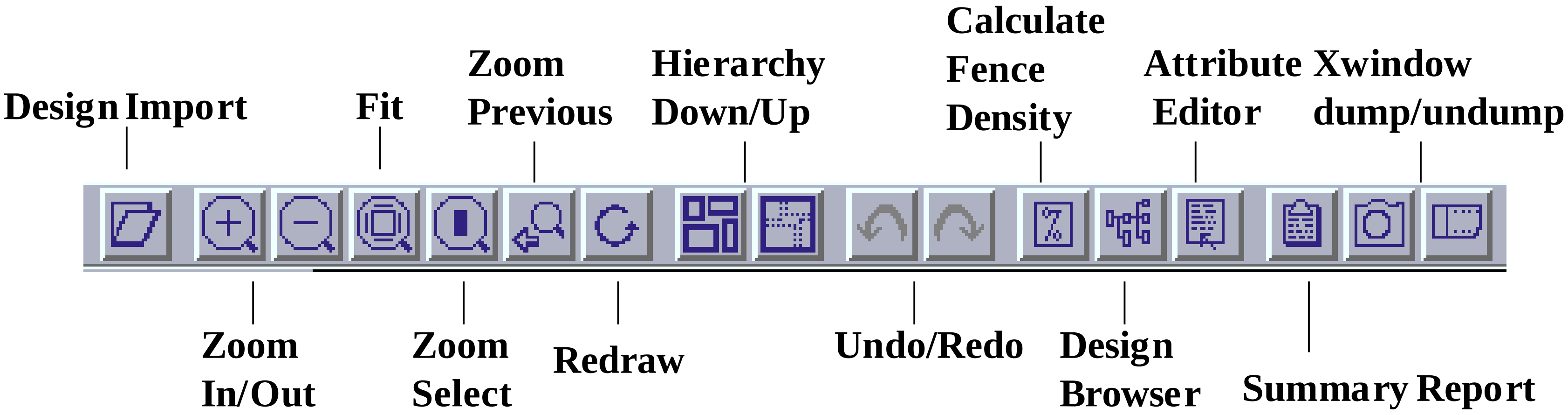
design views

auto query

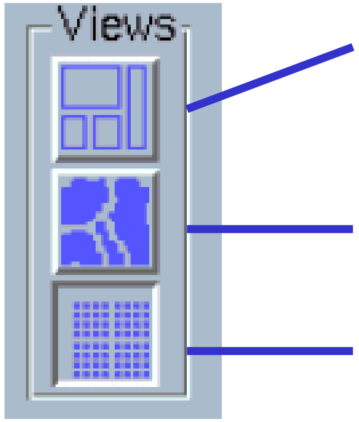
name of selected object

cursor coordinates

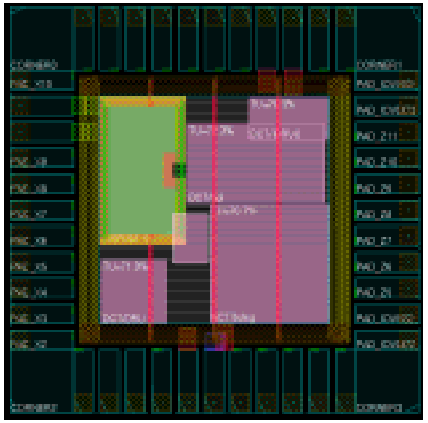
Tool Wedgits

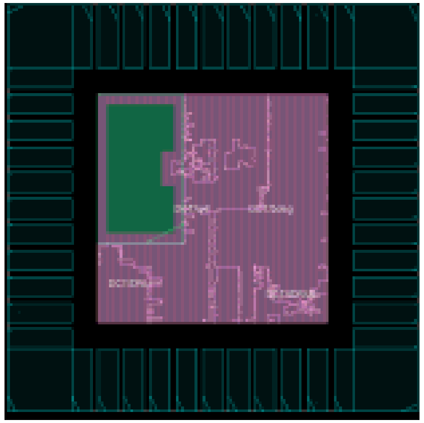


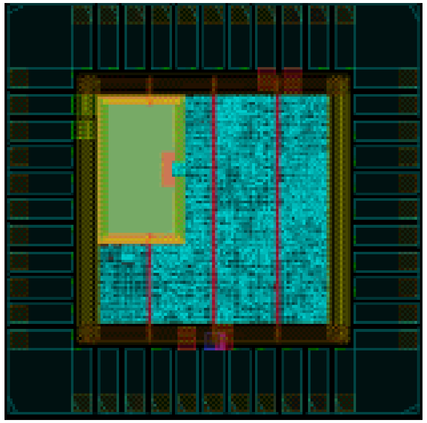
Design Views



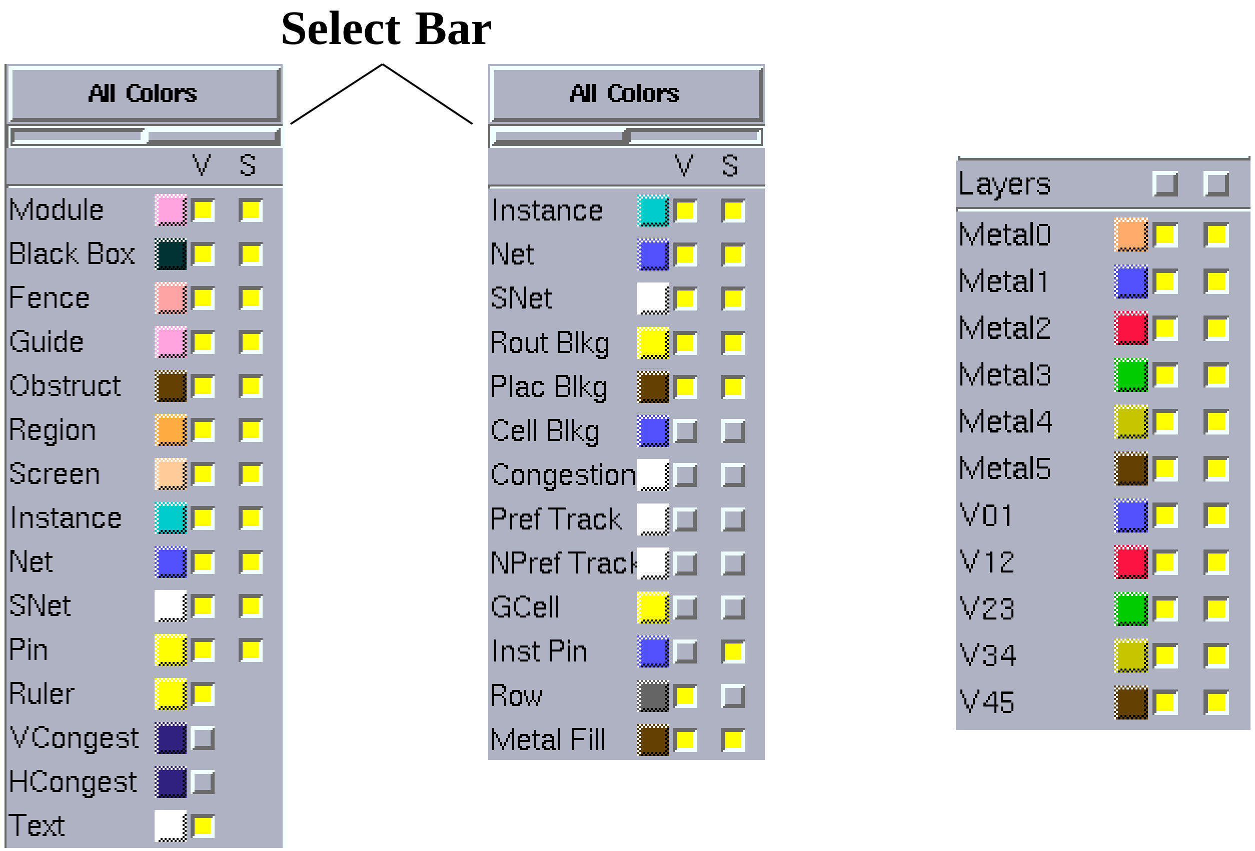
- FloorplanView
  - displays the hierarchical module and block guides,connection flight lines and floorplan objects
- Amoeba View
  - display the outline of modules after placement
- Placement View
  - display the detailed placements of cells, blocks.







Display Control



Select Bar

		V	S
Module	🟪	☑	☑
Black Box	⬛	☑	☑
Fence	🟥	☑	☑
Guide	🟪	☑	☑
Obstruct	🟫	☑	☑
Region	🟧	☑	☑
Screen	🟧	☑	☑
Instance	🟦	☑	☑
Net	🟦	☑	☑
SNet	⬜	☑	☑
Pin	🟨	☑	☑
Ruler	🟨	☑	
VCongest	🟦	□	
HCongest	🟦	□	
Text	⬜	☑	

		V	S
Instance	🟦	☑	☑
Net	🟦	☑	☑
SNet	⬜	☑	☑
Rout Blkg	🟨	☑	☑
Plac Blkg	🟫	☑	☑
Cell Blkg	🟦	□	□
Congestion	⬜	□	□
Pref Track	⬜	□	□
NPref Track	⬜	□	□
GCell	🟨	□	□
Inst Pin	🟦	□	☑
Row	⬛	☑	□
Metal Fill	🟫	☑	☑

Layers		□	□
Metal0	🟧	☑	☑
Metal1	🟦	☑	☑
Metal2	🟥	☑	☑
Metal3	🟩	☑	☑
Metal4	🟨	☑	☑
Metal5	🟫	☑	☑
V01	🟦	☑	☑
V12	🟥	☑	☑
V23	🟩	☑	☑
V34	🟨	☑	☑
V45	🟫	☑	☑

Common Used Bindkeys

Key	Action
q	Edit attribute
f	Fits display
z	Zoom in
Z	Zoom out
Arrows	pans design area in the direction of the arrow
Delete	Removes the last ruler
Esc	Removes all rulers

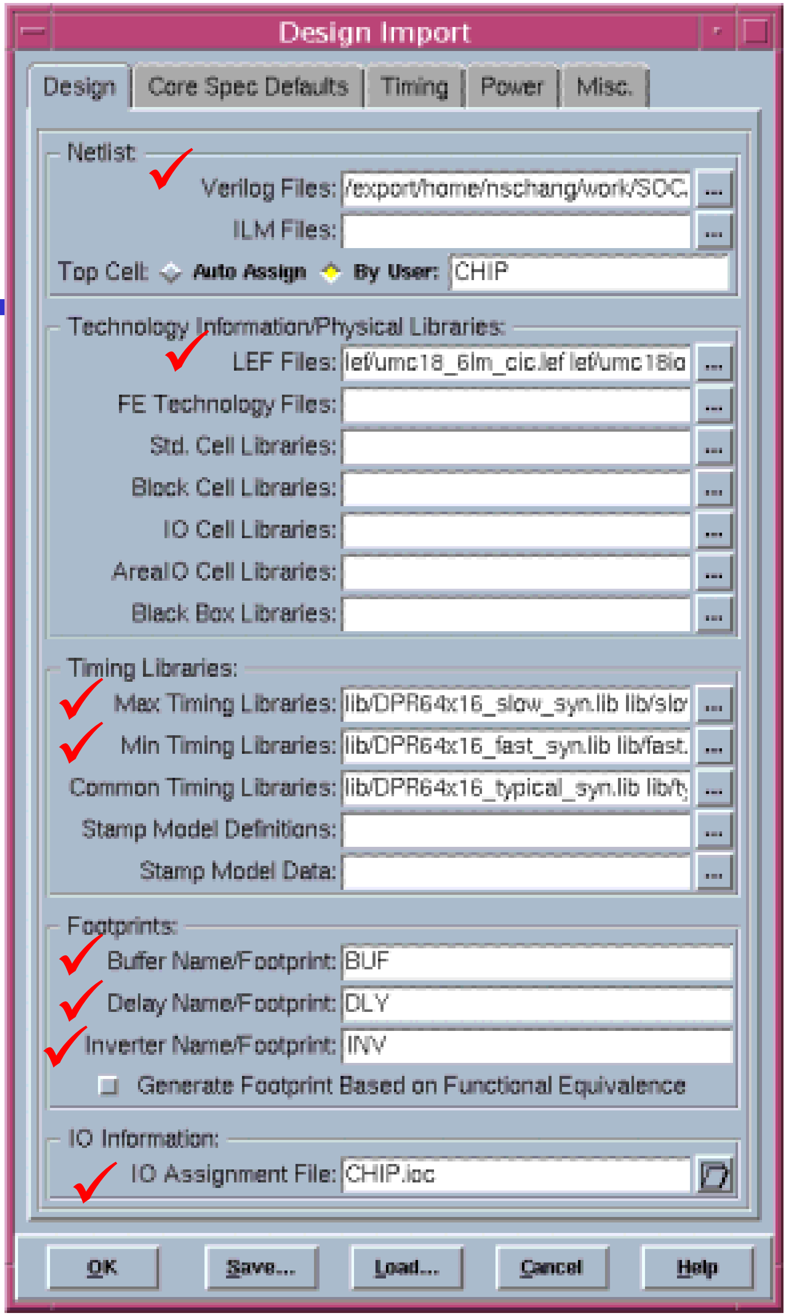
Key	Action
d	popup Delete
e	popup Edit
T	editTrim
0-9	View layer [0-9]
h	hierarchy up
H	hierarchy down

Looking for more bindkey:
Design->Preference, Binding Key

Import Design

Design→Design Import…

- Max Timing Libraries
  - containing worst-case conditions for setup-time analysis
- Min Timing Libraries
  - containing best-case conditions for hold-time analysis
- Common Timing Libraries
  - used in both setup and hold analysis
- IO Assignment File:
  - get a IO assignment template: ***Design→Save→I/O File…***



Design Import

Netlist:

✓ Verilog Files: /export/home/nschang/work/SOC.

ILM Files:

Top Cell: ○ Auto Assign ● By User: CHIP

Technology Information/Physical Libraries:

✓ LEF Files: lef/umc18_6lm_cic.lef lef/umc18io

FE Technology Files:

Std. Cell Libraries:

Block Cell Libraries:

IO Cell Libraries:

AreaIO Cell Libraries:

Black Box Libraries:

Timing Libraries:

✓ Max Timing Libraries: lib/DPR64x16_slow_syn.lib lib/slo

✓ Min Timing Libraries: lib/DPR64x16_fast_syn.lib lib/fast.

Common Timing Libraries: lib/DPR64x16_typical_syn.lib lib/ty

Stamp Model Definitions:

Stamp Model Data:

Footprints:

✓ Buffer Name/Footprint: BUF

✓ Delay Name/Footprint: DLY

✓ Inverter Name/Footprint: INV

□ Generate Footprint Based on Functional Equivalence

IO Information:

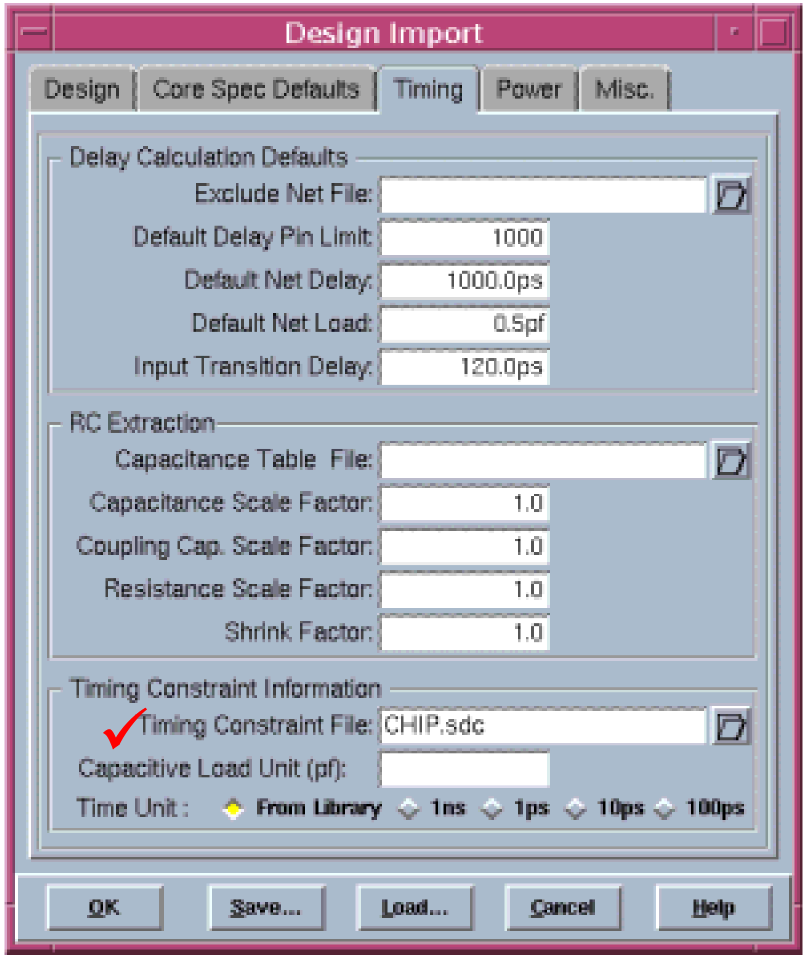
✓ IO Assignment File: CHIP.ioc

Import Design cont.

- Buffer Name/Footprint:
  - specifies the buffer cell family to be inserted or swapped.
  - required to run IPO and TD placement.
- Delay Name/Footprint:
  - required to run a fix hold time violation
- Inverter Name/Footprint:
  - required to run IPO and TD placement.
- Get footprint of library cells by:
  - ***Timing→Report→Cell Footprint***

Footprint Example:
For Cells:
BUFXL
BUFX1
BUFX2
BUFX3
BUFX4
BUFX8
BUFX12
BUFX16
BUFX20
Footprint : buf

Import Design -- Timing



Delay Calculation Defaults

Exclude Net File:

Default Delay Pin Limit: 1000

Default Net Delay: 1000.0ps

Default Net Load: 0.5pf

Input Transition Delay: 120.0ps

RC Extraction

Capacitance Table File:

Capacitance Scale Factor: 1.0

Coupling Cap. Scale Factor: 1.0

Resistance Scale Factor: 1.0

Shrink Factor: 1.0

Timing Constraint Information

✓Timing Constraint File: CHIP.sdc

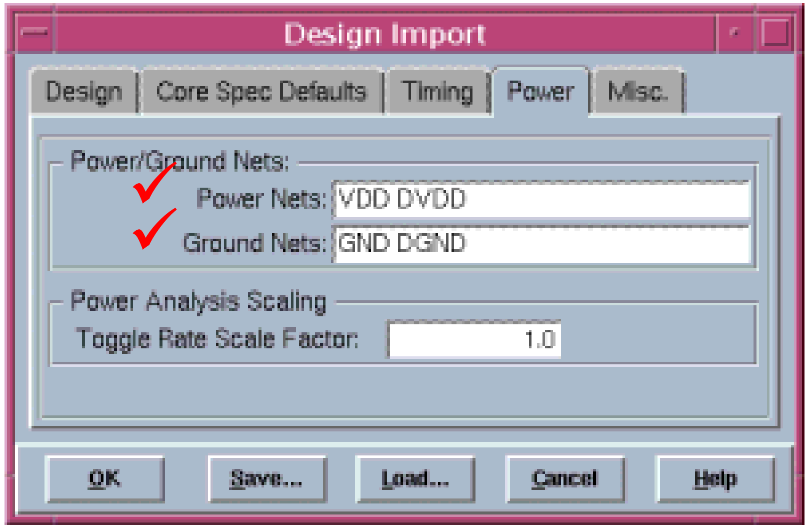
Capacitive Load Unit (pf):

Time Unit : ◆ From Library ◇ 1ns ◇ 1ps ◇ 10ps ◇ 100ps

- Default Delay Pin Limit:
  - Nets with terminal counts greater than the specified value are assigned the default net delay and net load entries.
- Default Net Delay:
  - Set the delay values for a net that meets the pin limit default.
- Default Net Load:
  - Set the load for a net that meets the pin limit default.
- Input Transition Delay:
  - Set the Primary inputs and clock nets.

Import Design -- Power

- Specify the names of Power Nets and Ground Nets



Power/Ground Nets:

✓ Power Nets: VDD DVDD

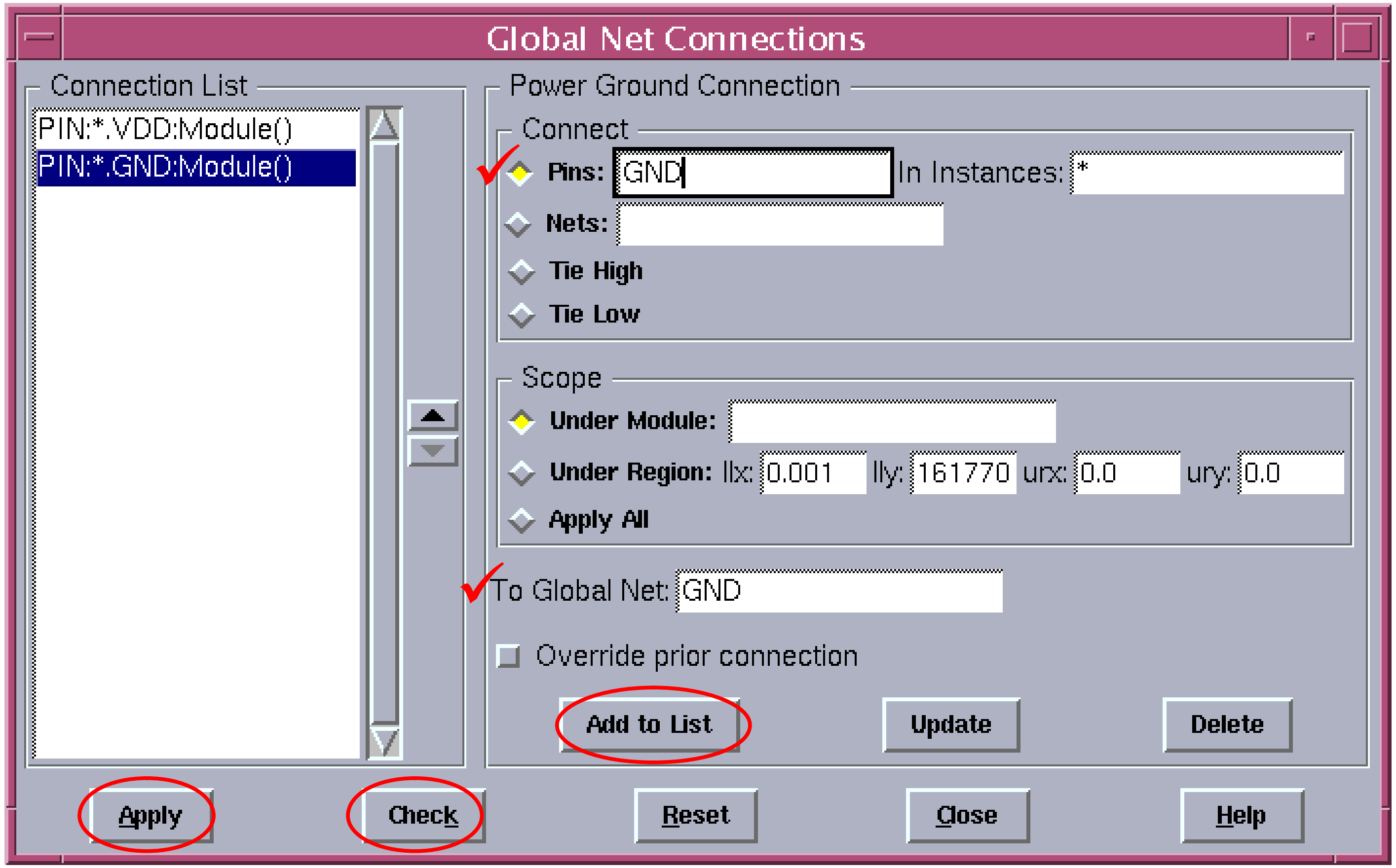
✓ Ground Nets: GND DGND

Power Analysis Scaling

Toggle Rate Scale Factor: 1.0

Global Net Connection

Floorplan→ Gloval Net Connections…



Global Net Connections

Connection List

PIN:*.VDD:Module()

PIN:*.GND:Module()

Power Ground Connection

Connect

● Pins: GND In Instances: *

○ Nets:

○ Tie High

○ Tie Low

Scope

● Under Module:

○ Under Region: llx: 0.001 lly: 161770 urx: 0.0 ury: 0.0

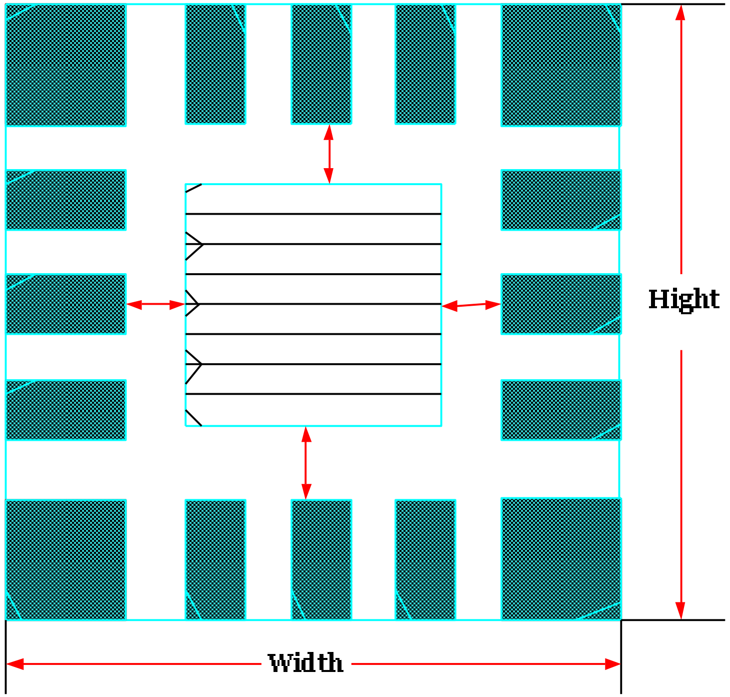
○ Apply All

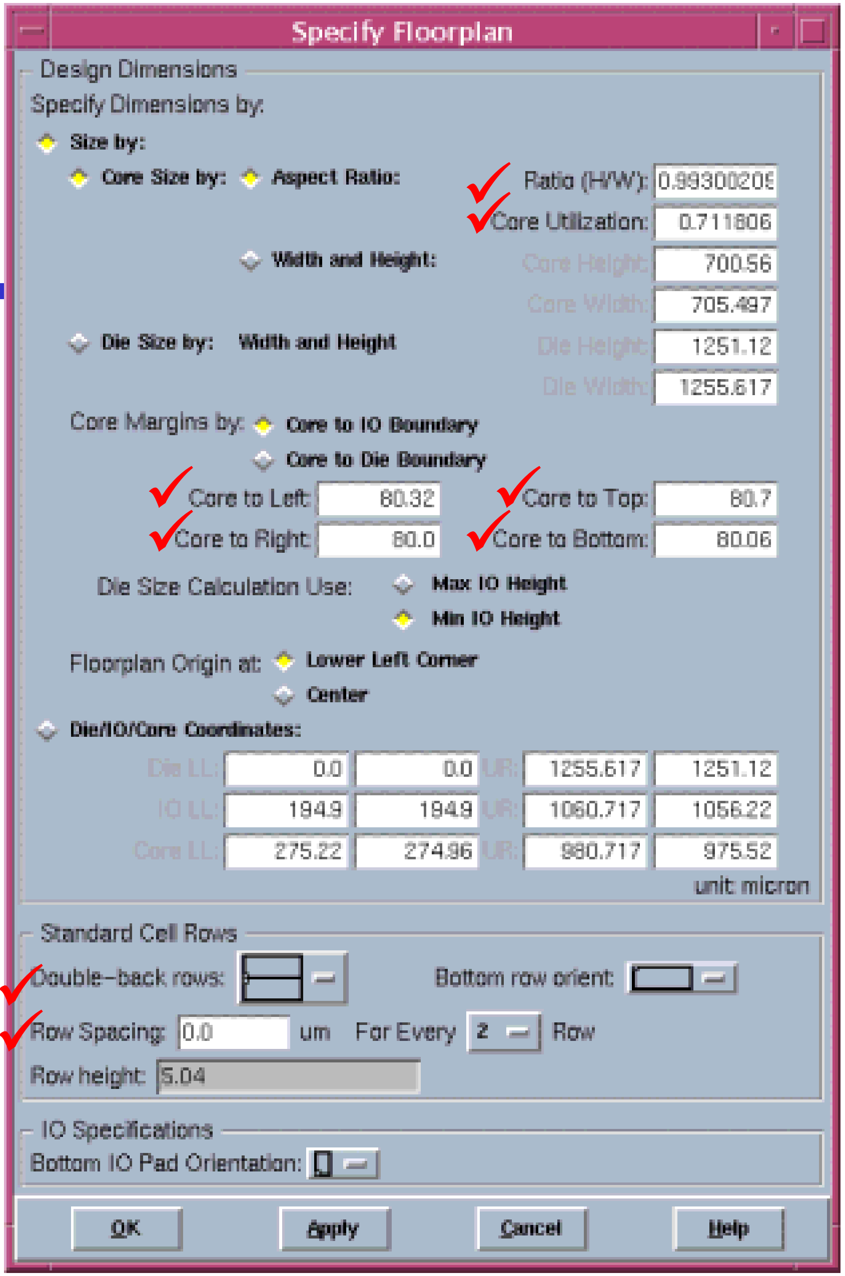
To Global Net: GND

□ Override prior connection

Specify Floorplan

Floorplan→Specify Floorplan …





Design Dimensions

Specify Dimensions by:

● Size by:

● Core Size by: ● Aspect Ratio: ✓ Ratio (H/W): 0.99300209 ✓ Core Utilization: 0.711806

○ Width and Height: Core Height: 700.56 Core Width: 705.497

○ Die Size by: Width and Height Die Height: 1251.12 Die Width: 1255.617

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

✓ Core to Left: 80.32 ✓ Core to Top: 80.7

✓ Core to Right: 80.0 ✓ Core to Bottom: 80.06

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

Floorplan Origin at: ● Lower Left Corner ○ Center

○ Die/IO/Core Coordinates:

Die LL:	0.0	0.0	UR:	1255.617	1251.12
IO LL:	194.9	194.9	UR:	1060.717	1056.22
Core LL:	275.22	274.96	UR:	980.717	975.52

unit: micron

Standard Cell Rows

✓ Double-back rows: Bottom row orient:

✓ Row Spacing: 0.0 um For Every 2 Row

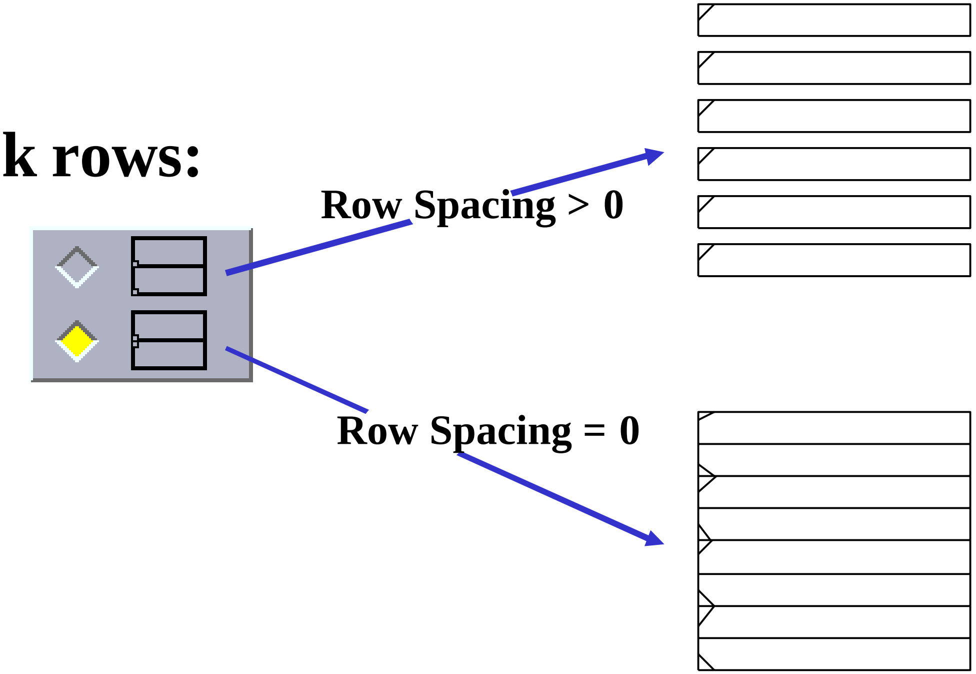
Row height: 5.04

IO Specifications

Bottom IO Pad Orientation:

Specify Floorplan – Doube back rows

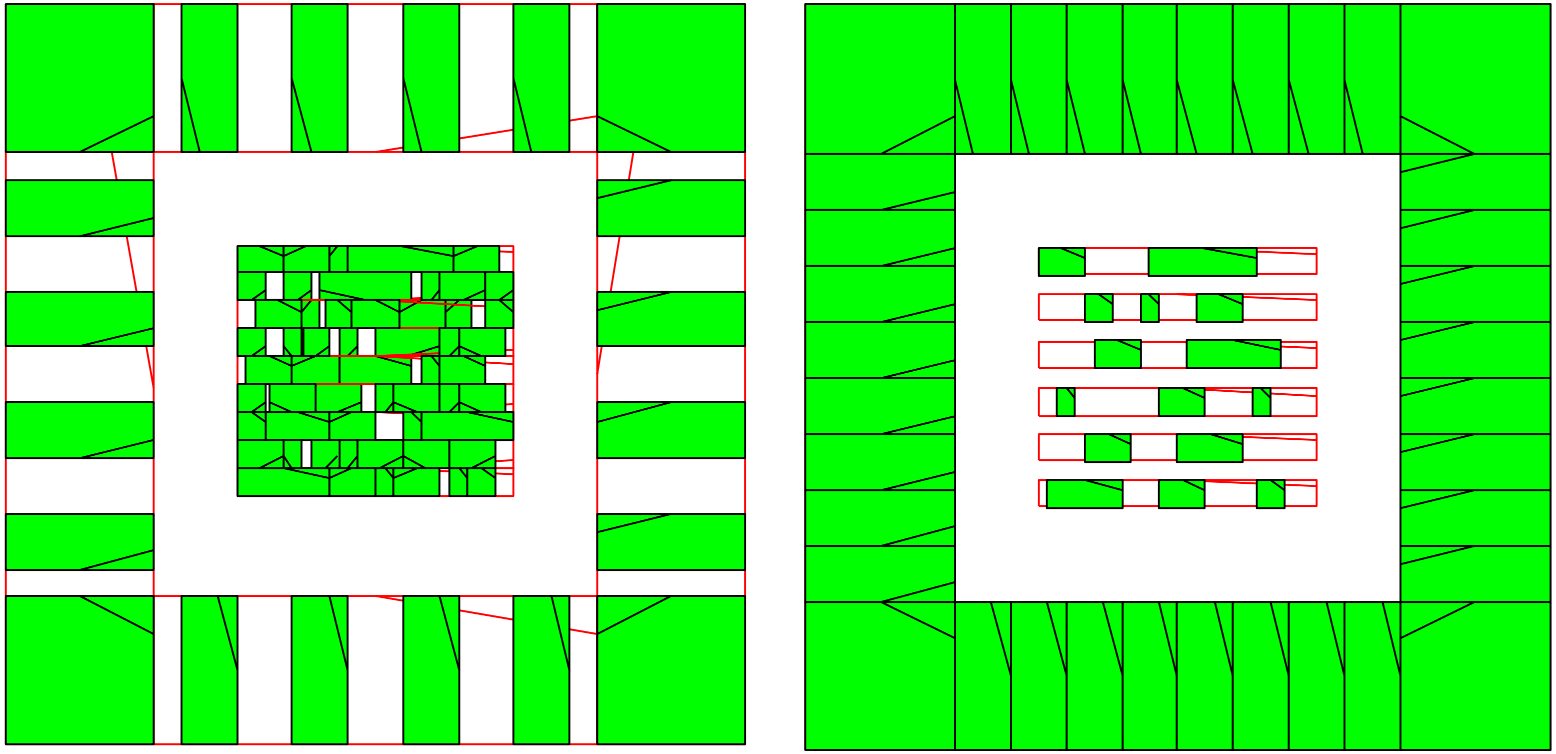
Double-back rows:



Row Spacing > 0

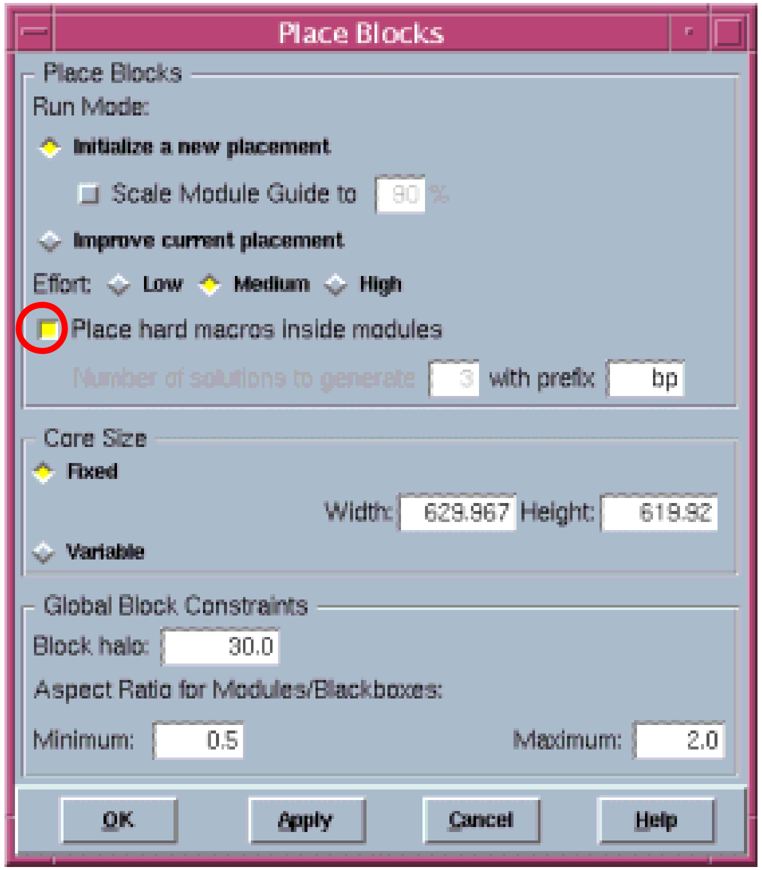
Row Spacing = 0

Core Limit, I/O Limnt



Place Blocks

Floorplan→Place Blocks/Modules→Place …



Place Blocks

Run Mode:

◆ Initialize a new placement

□ Scale Module Guide to 90 %

◇ Improve current placement

Effort: ◇ Low ◆ Medium ◇ High

■ Place hard macros inside modules

Number of solutions to generate 3 with prefix bp

Core Size

◆ Fixed

Width: 629.967 Height: 619.92

◇ Variable

Global Block Constraints

Block halo: 30.0

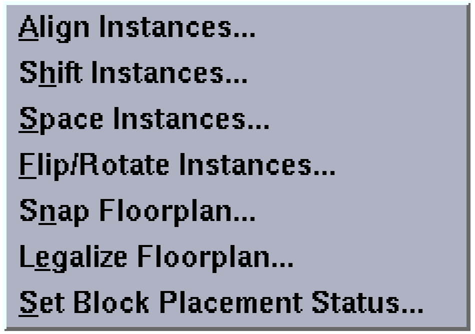
Aspect Ratio for Modules/Blackboxes:

Minimum: 0.5 Maximum: 2.0

- automatic place blocks (blackboxes and partitions) and hard macros at the top-level design.
- Block halo
  - Specifies the minimum amount of space around blocks that is preserved for routing.

Manually Place Block

- ✥ Move/Resize/Reshape floorplan object.
- Use functions in : ***Floorplan→Edit Floorplan*** to edit floorplan.

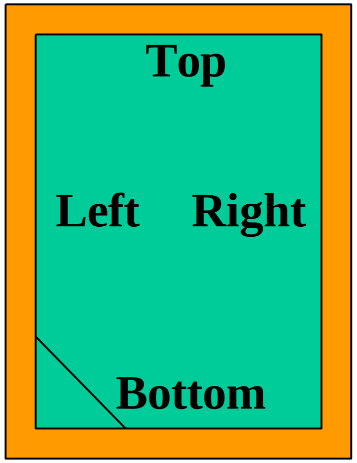


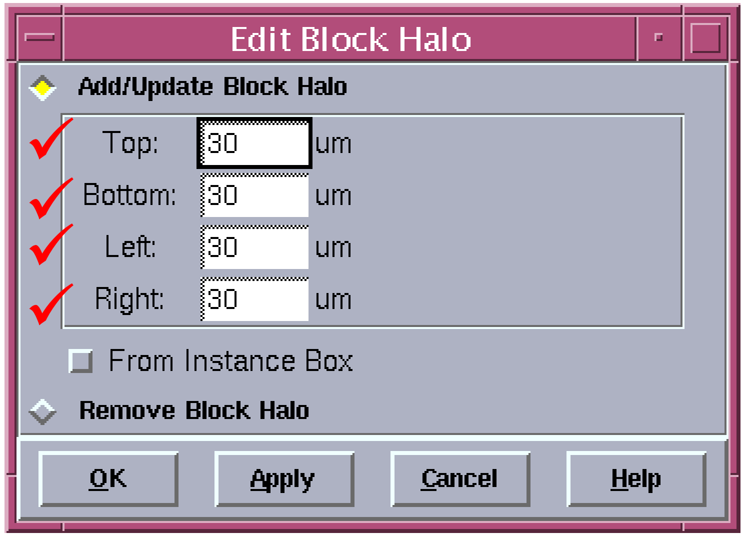
- Set placement status of all pre-placed block to ***preplaced*** in order to avoid these blocks be moved by amoebaPlace later.
- ***Floorplan→Edit Floorplan→Set Block Placement Status…***

Add Halo To Block

Floorplan→Edit Block Halo…

- Prevent the placement of blocks and standard cells in order to reduce congestion around a block.





◆ Add/Update Block Halo

✓ Top: 30 um

✓ Bottom: 30 um

✓ Left: 30 um

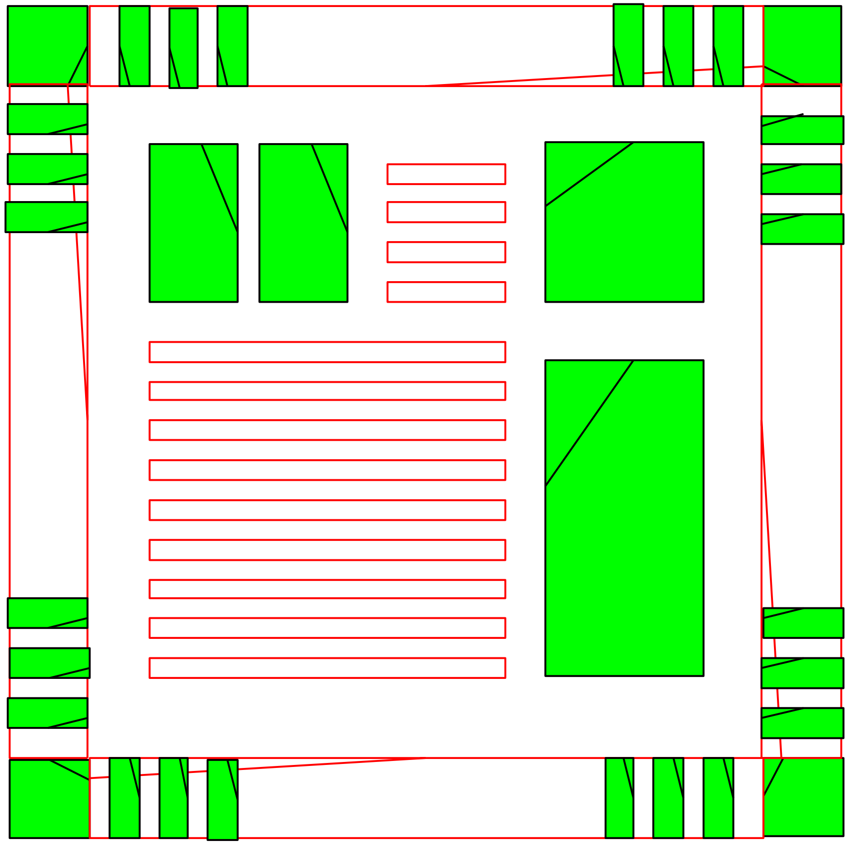
✓ Right: 30 um

□ From Instance Box

◇ Remove Block Halo

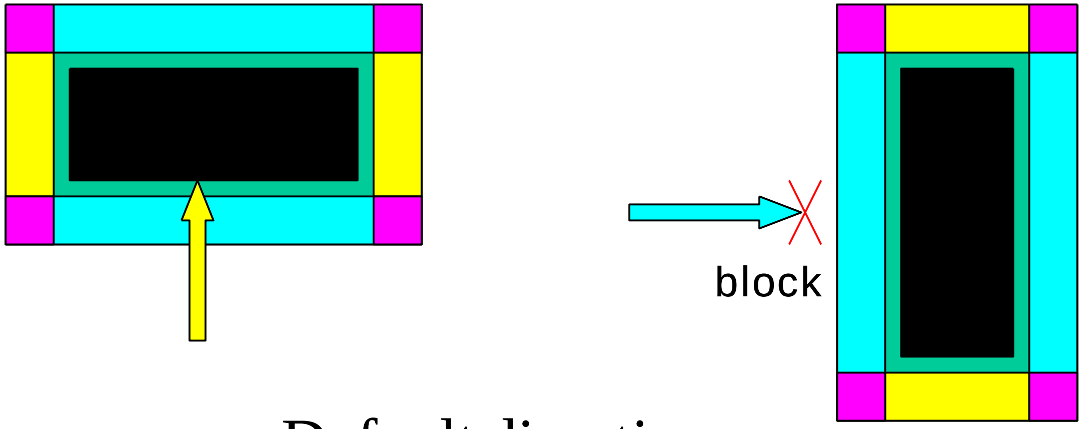
Block Placement

- Flow step
  - I/O pre-placed
  - Run quick block placement
  - Throw away standard cell placement
  - Manually fit blocks
- Block place issue
  - power issue
  - noise issue
  - route issue



Block Placement

- Preserve enough power pad
- Create power rings around block
- Follow default routing direction rule
- Reserve a rounded core row area for placer

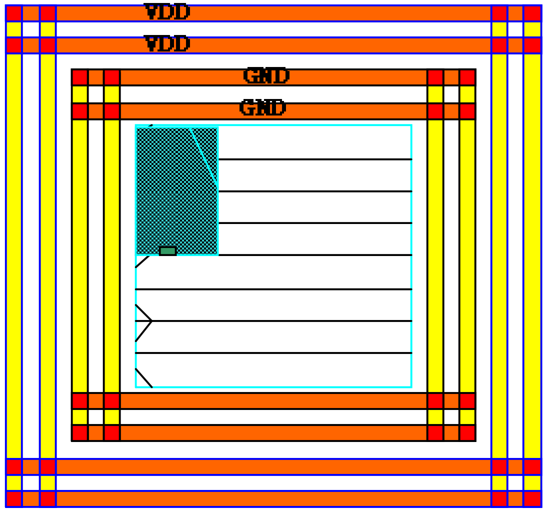


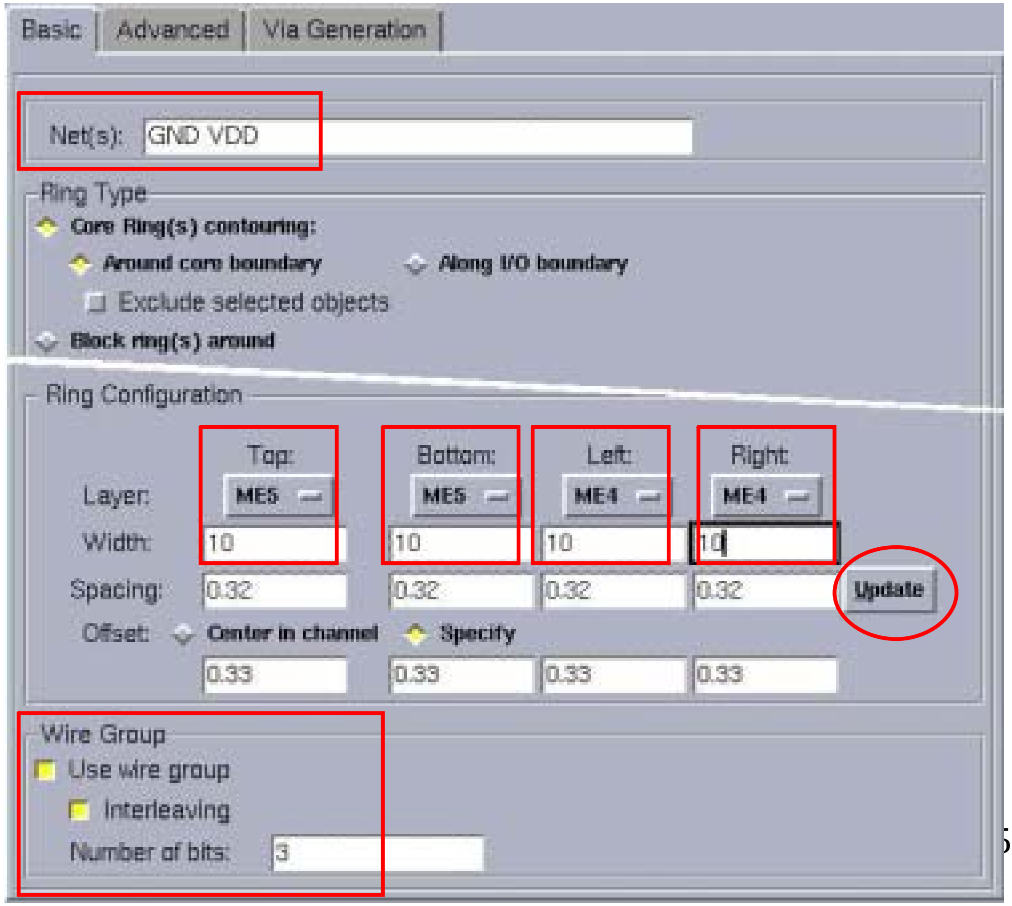
Default direction

Power Planning: Add Rings

Floorplan→Power Planning→AddRings

Use wire group to avoid slot DRC error





Net(s): GND VDD

Ring Type

◆ Core Ring(s) contouring:

◆ Around core boundary ◇ Along I/O boundary

□ Exclude selected objects

◇ Block ring(s) around

Ring Configuration

	Top:	Bottom:	Left:	Right:
Layer:	ME5	ME5	ME4	ME4
Width:	10	10	10	10
Spacing:	0.32	0.32	0.32	0.32
Offset: ◇ Center in channel ◆ Specify	0.33	0.33	0.33	0.33

Wire Group

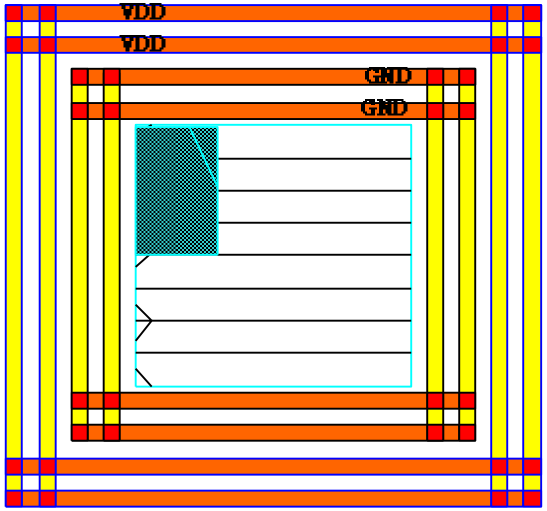
■ Use wire group

■ Interleaving

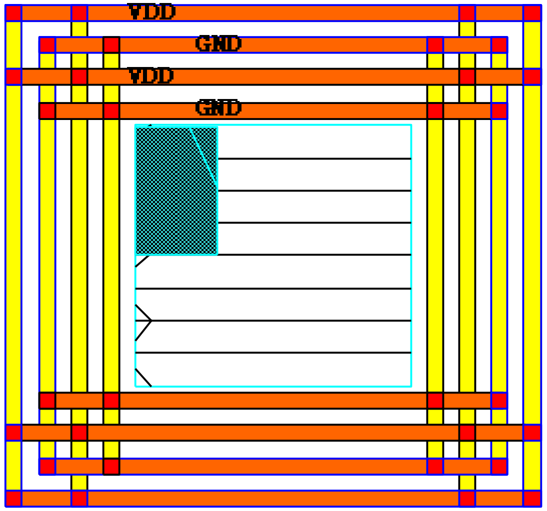
Number of bits: 3

Power Planning: Wire Group

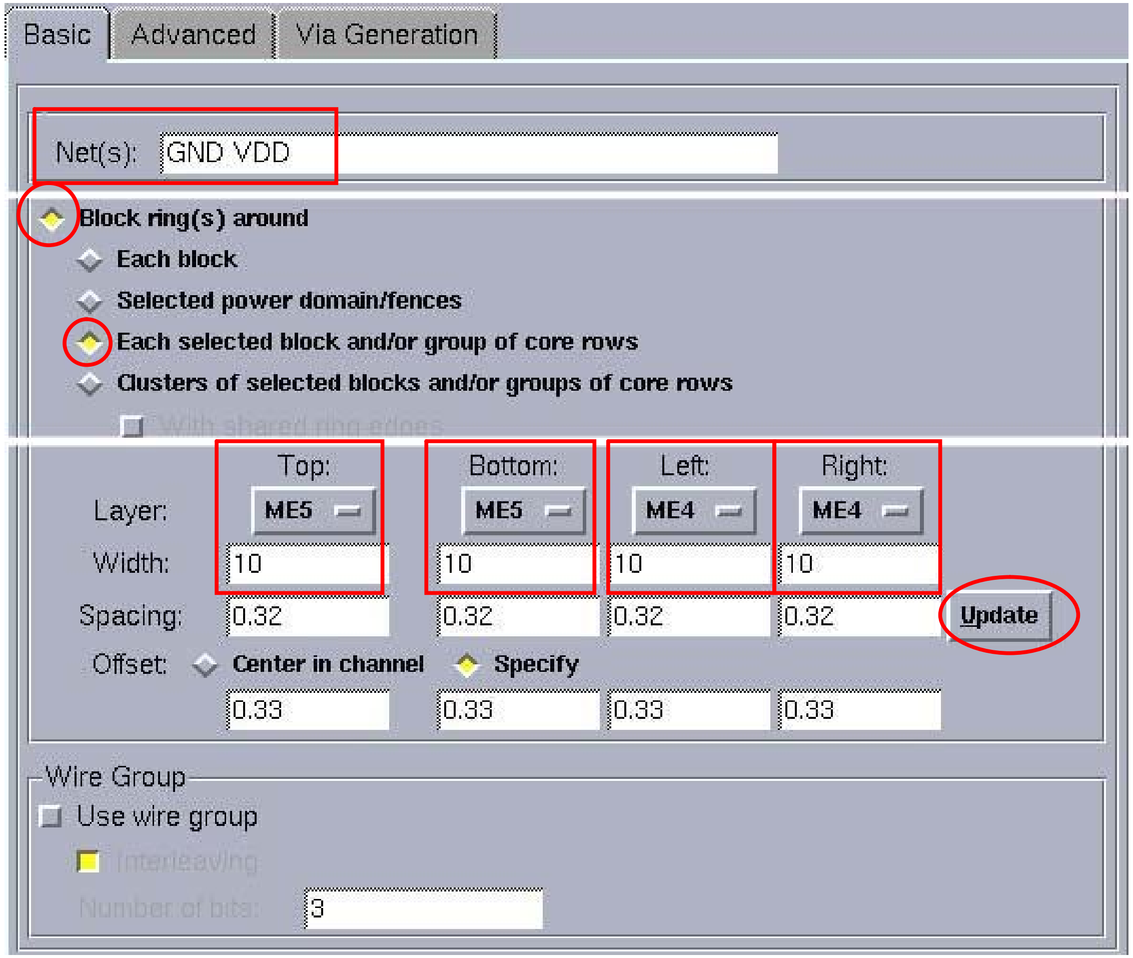
- ✓Use wire group
  no interleaving
- ✓number of bits = 2



- ✓Use wire group
- ✓interleaving
- ✓number of bits = 2



Power Planning: Block Ring



Net(s): GND VDD

◆ Block ring(s) around

◇ Each block

◇ Selected power domain/fences

◆ Each selected block and/or group of core rows

◇ Clusters of selected blocks and/or groups of core rows

□ With shared ring edges

	Top:	Bottom:	Left:	Right:
Layer:	ME5	ME5	ME4	ME4
Width:	10	10	10	10
Spacing:	0.32	0.32	0.32	0.32
Offset: ◇ Center in channel ◆ Specify	0.33	0.33	0.33	0.33

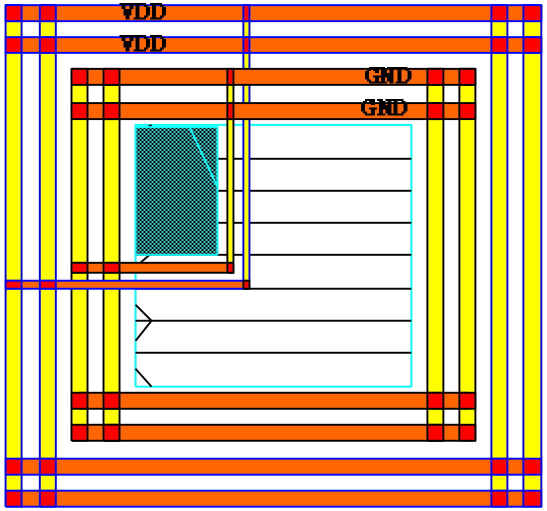
Wire Group

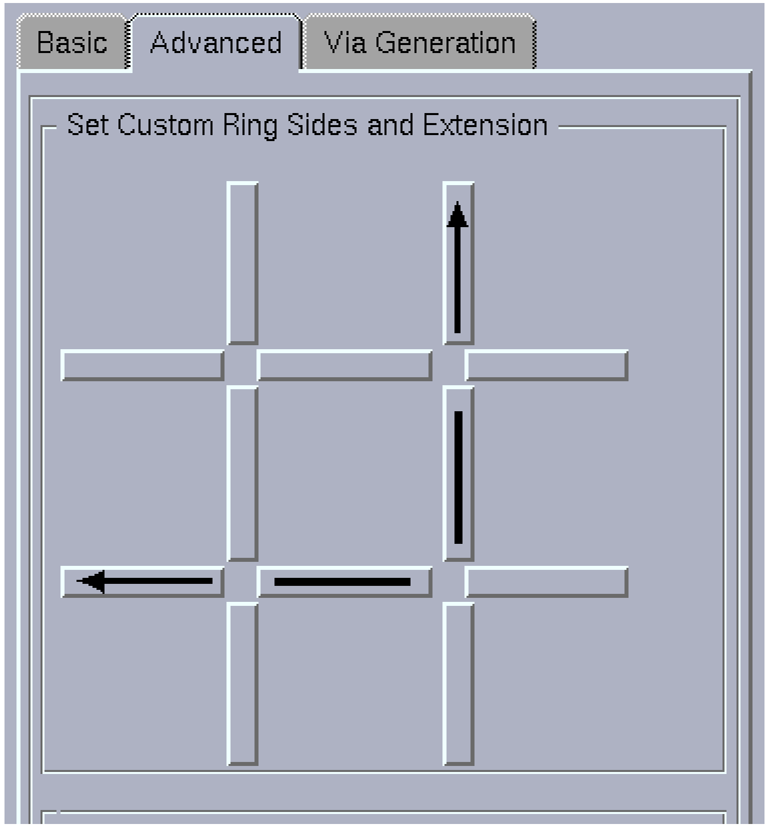
□ Use wire group

☑ Interleaving

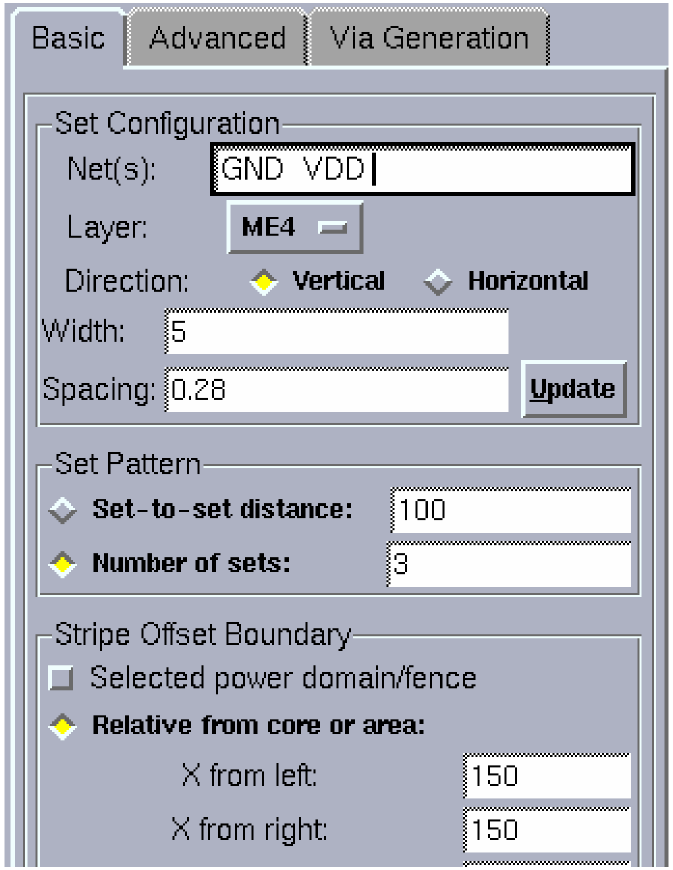
Number of bits: 3

Power Planning: Block Ring cont.





Power Planning: Add Stripes



Set Configuration

Net(s): GND VDD

Layer: ME4

Direction: ● Vertical ○ Horizontal

Width: 5

Spacing: 0.28

Set Pattern

○ Set-to-set distance: 100

● Number of sets: 3

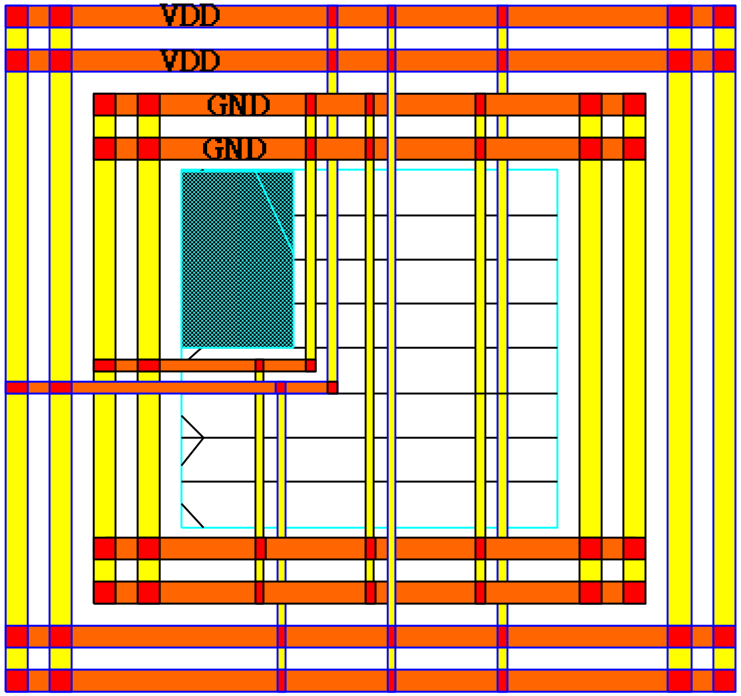
Stripe Offset Boundary

□ Selected power domain/fence

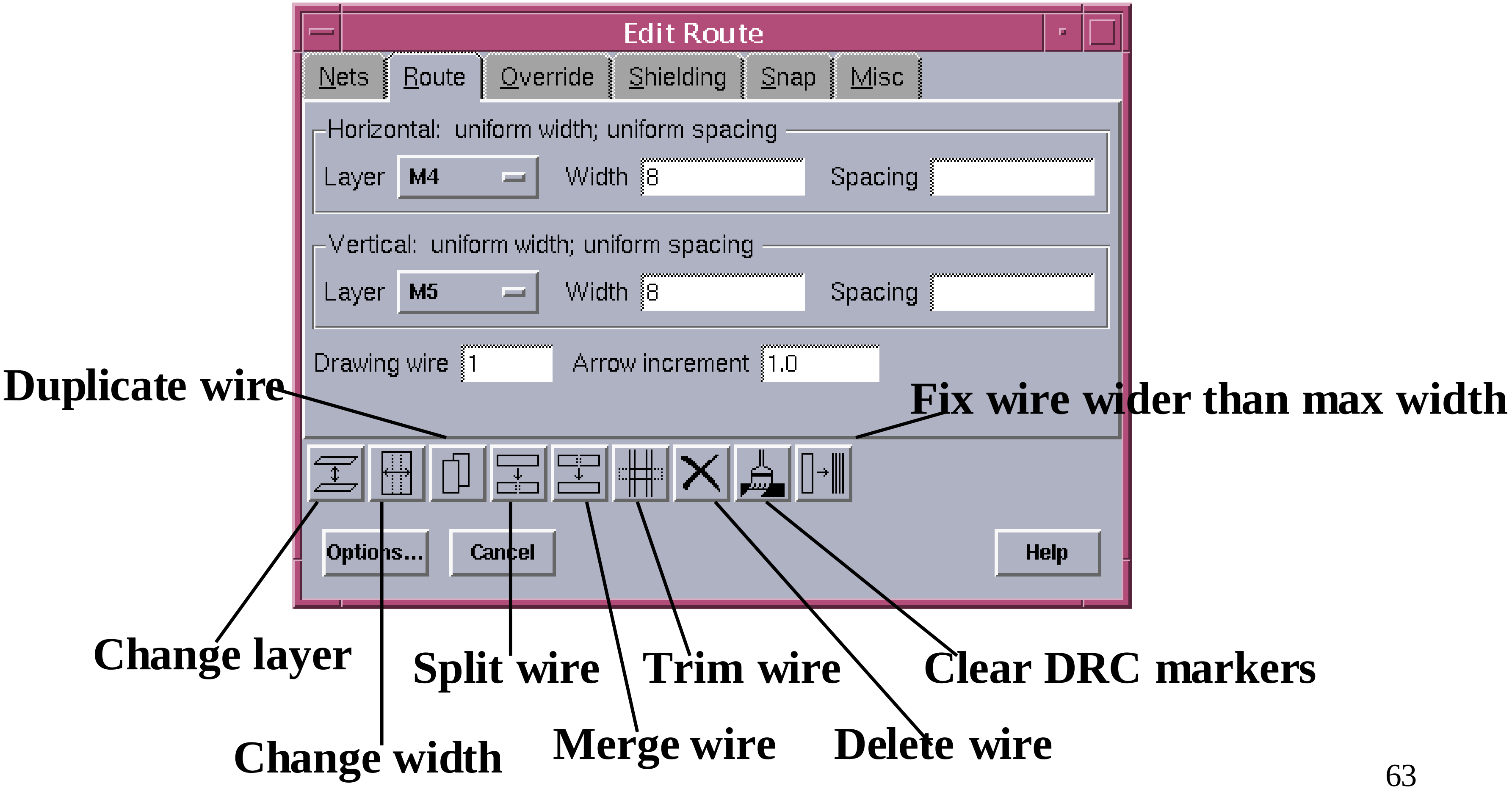
● Relative from core or area:

X from left: 150

X from right: 150



Edit Route



Edit Route

Horizontal: uniform width; uniform spacing

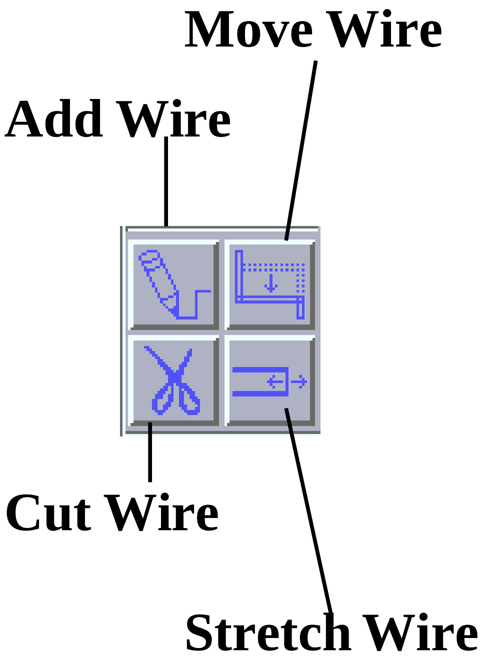
Layer M4 Width 8 Spacing

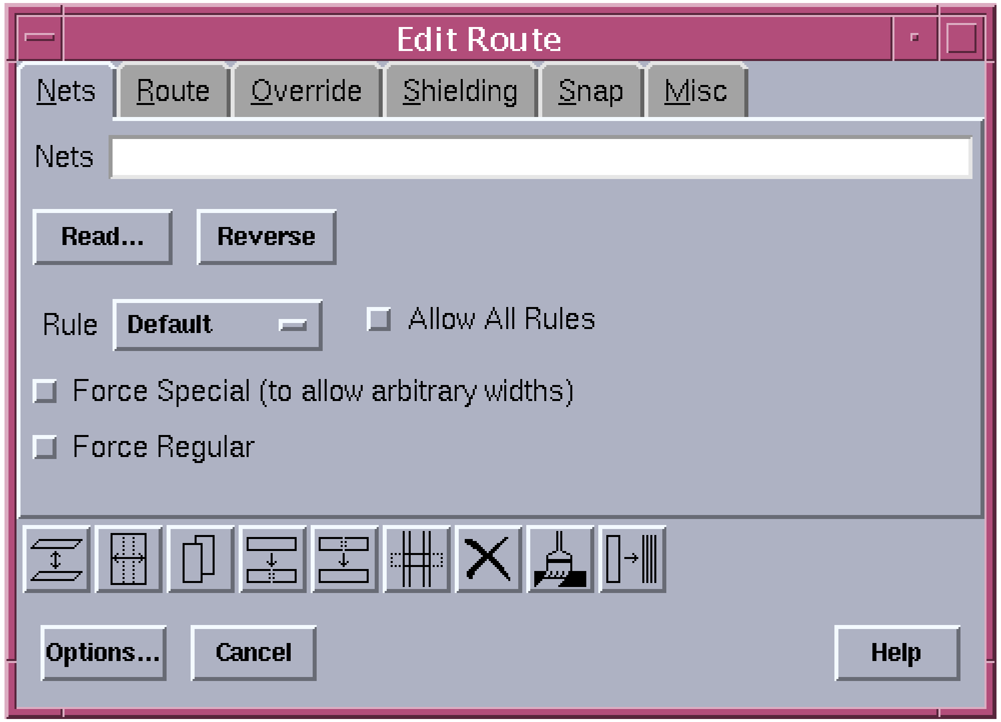
Vertical: uniform width; uniform spacing

Layer M5 Width 8 Spacing

Drawing wire 1 Arrow increment 1.0

Edit Route cont.





Nets

Rule Default

□ Allow All Rules

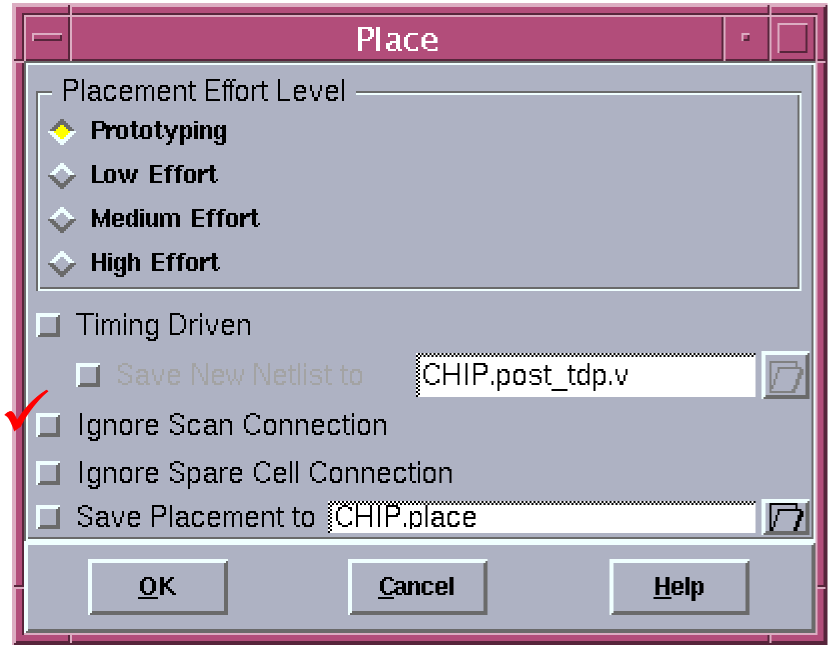
□ Force Special (to allow arbitrary widths)

□ Force Regular

Placement

Place→Place…

- Prototyping : Runs quickly, but components may not be placed at legal location.
- Timing Driven:
  - Build timing graph before place.
  - meeting setup timing constraints with routability.
  - Limited IPO by upsizeing/downsizing instances.
- Ignore Scan Connection
  - nets connected to either the scan-in or scan-out are ignored.
- Check placement after placed
  - ***place→Check Placement***



Place

Placement Effort Level

● Prototyping

○ Low Effort

○ Medium Effort

○ High Effort

□ Timing Driven

□ Save New Netlist to CHIP.post_tdp.v

✓ □ Ignore Scan Connection

□ Ignore Spare Cell Connection

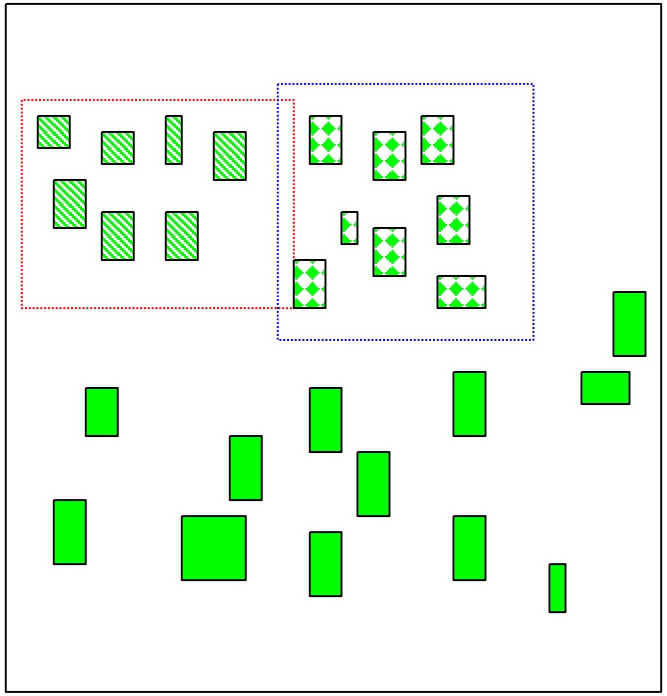
□ Save Placement to CHIP.place

Floorplan Purposes

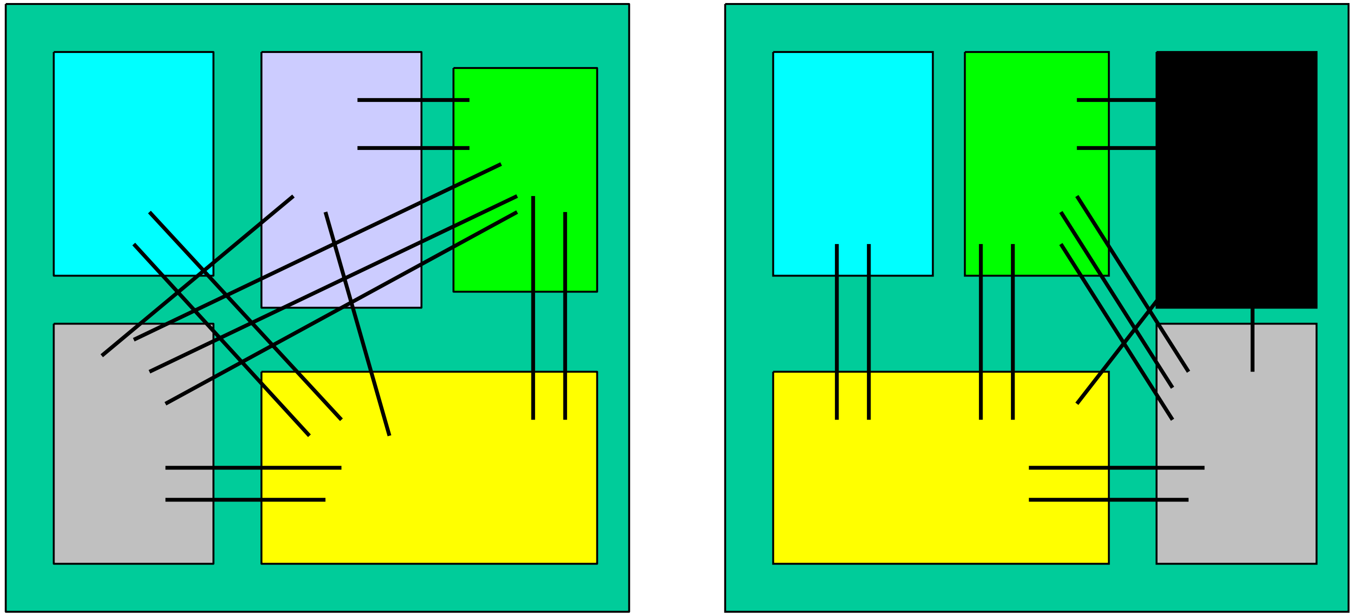
- Develop early physical layout to ensure design objective can be archived
  - Minimum area for low cost
  - Minimum congestion for design routable
  - Estimate parasitic for delay calculation
  - Analysis power for reliability
- gain early visibility into implementation issues

Guide , Region, Fence

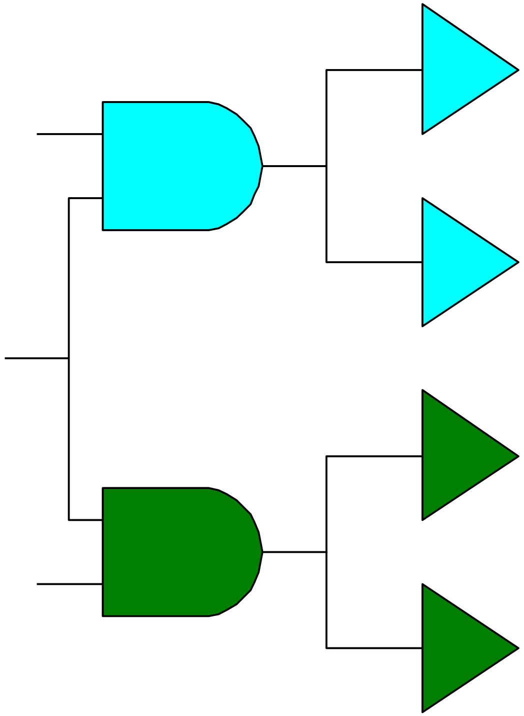
- Placement constraint
- Create guide for timing issue
- A critical path should not through two different modules
- The more region, the more complicated floorplanning



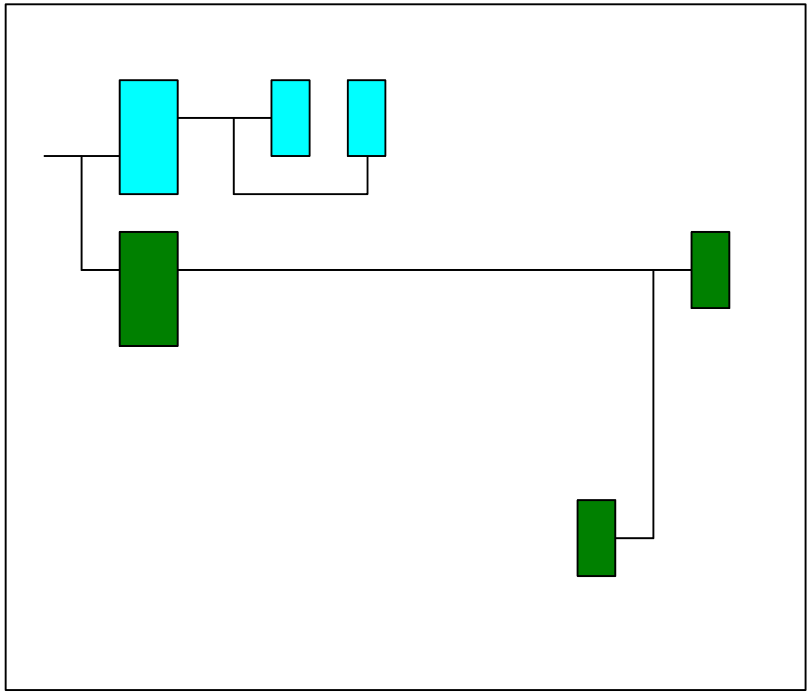
Difference Floorplan
Difference Performance



Wire Load After Placement



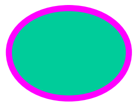
Logical



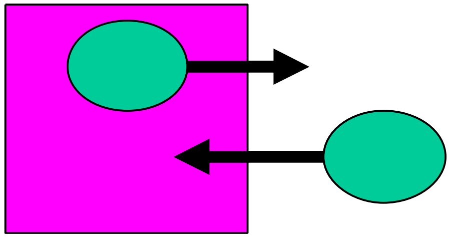
wire load after placement

Module Constraint

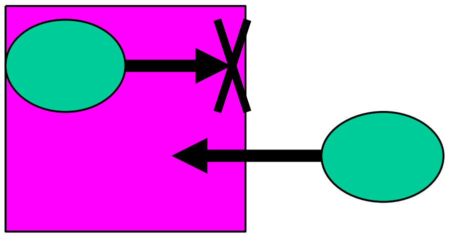
- ◆ Soft Guide
- ◆ Guide
- ◆ Region
- ◆ Fence



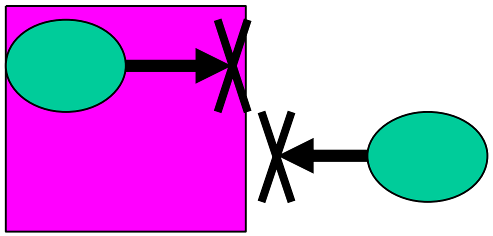
Soft Guide



Guide



Region



Fence

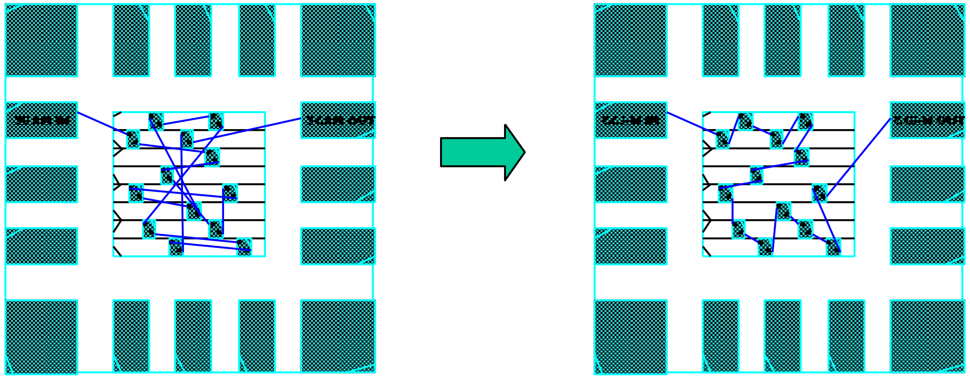
Specify Scan Chain

encounter > ***specifyScanChain scanChainName***
–start {ftname | instPinName}
– stop {ftname | instPinName}

- Specifies a scan chain in a design. The actual tracing of the scan chain is performed by the ***scanTrace*** or ***scanReorder*** command
- ftname
  - The design input/output pin name
- instPinName
  - The design instance input/output pin name

Scan Chain Reorder

Place→Reorder Scan

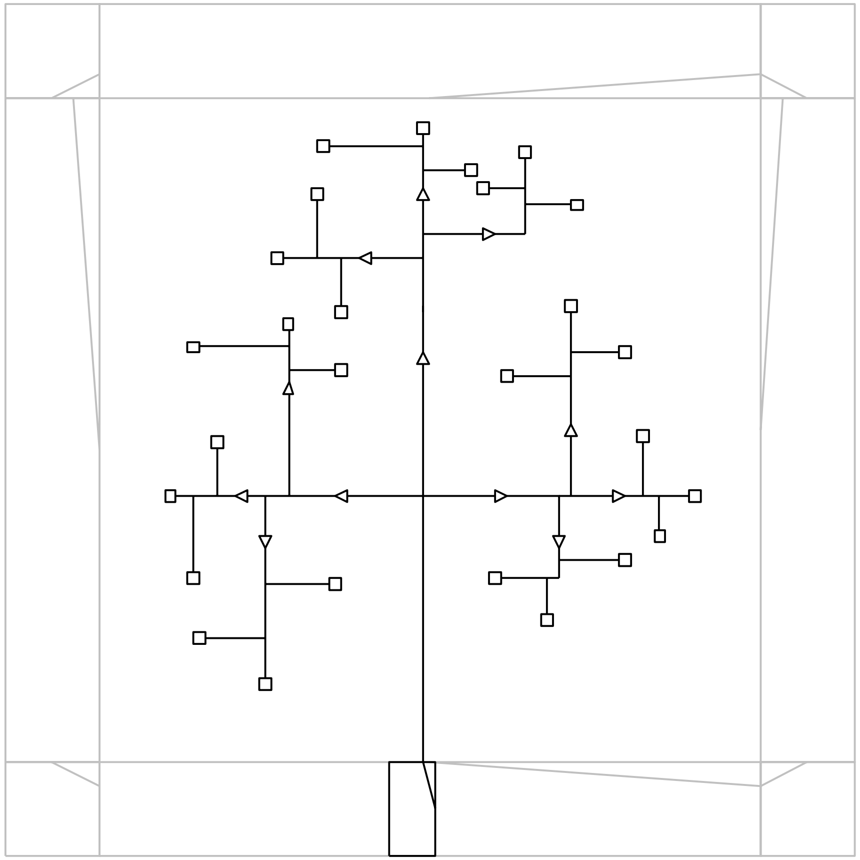


- No Skip
  - ➢ Buffers and inverters remain after the scan chain reorder
- Skip Buffer
  - ➢ Ignores buffers in the scan chain.
- Skit Two Pin Cell
  - ➢ Ignores buffers and inverters in the scan chain

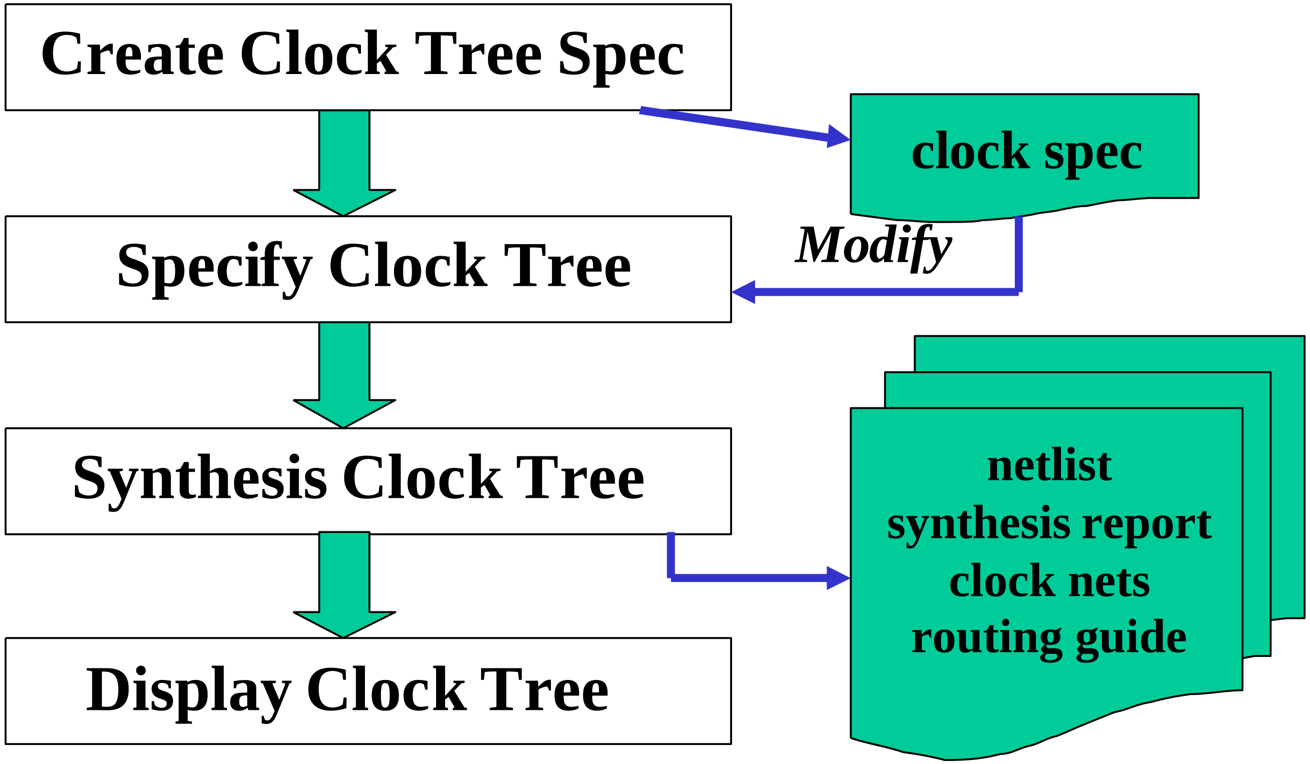
Clock Problem

- Clock problem
  - Heavy clock net loading
  - Long clock insertion delay
  - Clock skew
  - Skew across clocks
  - Clock to signal coupling effect
  - Clock is power hungry
  - Electromigration on clock net
- Solutions of these problems may be conflict
- Clock is one of the most important treasure in a chip, do not take it as other use.

Clock Tree Topology

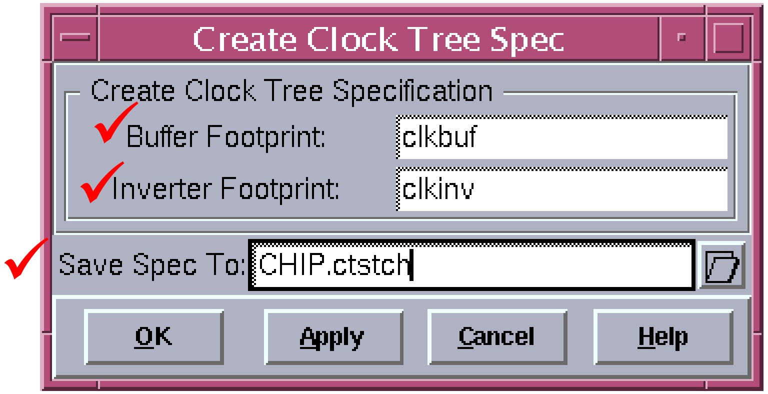


Synthesize Clock Tree



Create Clock Tree Spec.

Clock→Create Clock Tree Spec



Create Clock Tree Specification

✓Buffer Footprint: clkbuf

✓Inverter Footprint: clkinv

✓Save Spec To: CHIP.ctstch

CTS

- CTS traces the clock starting from a root pin, and stops at:
  - A clock pin
  - A D-input pin
  - An instance without a timing arc
  - A user-specified leaf pin or excluded pin
- Write a CTS spec. template:
  - ***specifyClockTree -template***

CTS spec.

- A CTS spec. contain the following information.
  - Timing constraint file (optional)
  - Naming attributes (optional)
  - Macro model data (optional)
  - Clock grouping data (optional)
  - Attributes used by NanoRoute routing solution (optional)
  - Requirement for manual CTS or utomatic,gated CTS

CTS spec.
--Naming Attributes Section

- ***TimingConstraintFile*** *filename*
  - define a timing constraint file for use during CTS
- ***NameDelimiter*** *delimiter*
  - name delimiter used when inserting buffers and updating clock root and net names.
  - ***NameDelimiter #*** ➔ create names clk##L3#I2
  - default ➔ clk__L3_I2
- ***UseSingleDelim*** *YES|NO*
  - *YES* ➔ clk_L3_I2
  - NO ➔ clk__L3_I2 (default)

CTS Spec.
-- NanoRoute Attribute Section

- ***RouteTypeName*** *name*

  RouteTypeName CK1

  ……

  END

- ***NonDefaultRule*** *ruleName*
  - Specify LEF NONDEFAULTRULE to be used
- ***PreferredExtraSpace*** *[0-3]*
  - add space around clock wires
- ***Shielding*** *PGNetName*
  - Defines the power and ground net names

CTS Spec.
-- Macro Model Data Section
-- Clock Grouping Section

- ***MacroModel***
  - ***MacroModel port*** *R64x16/clk 90ps 80ps 90ps 80ps 17pf*
  - ***MacroModel pin*** *ram1/clk 90ps 80ps 90ps 80ps 17pf*
  - delay_and_capacitance_value:
    maxRise minRise maxFall minFall inputCap
- ***ClkGroup***
  - Specifies tow or more clock domains for which you want CTS to balance the skew.
  - ***ClkGroup***
    +clockRootPinName1
    +clockRootPinName2
    …..

CTS Spec.
--Manually Define Clock Tree Topology

ClockNetName *netName*

LevelNumber *number*

- ➢ Specify the clock tree level number

LevelSpec *levelNumber numberOfBuffers bufferType*

- ➢ *levelNumber*
  - ✓ Specify the level number in the clock tree
- ➢ *numberOfBuffer*
  - ✓ the total number of buffers CTS should allow on the specified level
- ➢ Example:

  LevelSpec 1 2 CLKBUFX2

  LevelSpec 2 2 CLKBUFX2

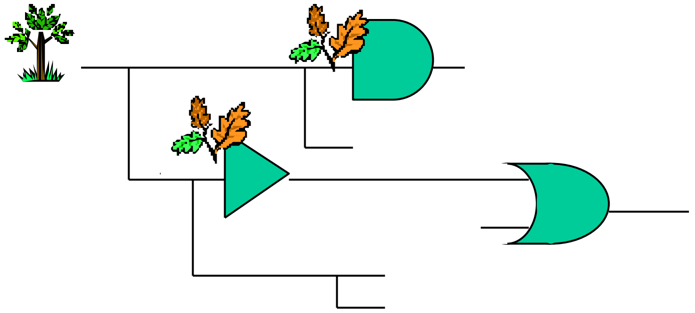
End

CTS Spec.
-- Automatic Gated CTS Section

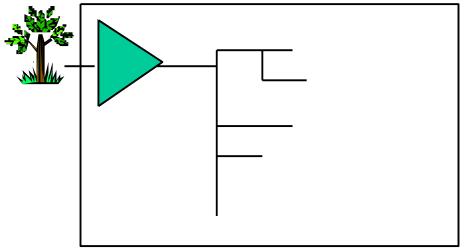
- ***AutoCTSRootPin*** *clockRootPinName*
- ***MaxDelay*** *number{ns|ps}*
- ***MinDelay*** *number{ns|ps}*
- ***SinkMaxTran*** *number{ns|ps}*
  - maximum input transition time for sinks(clock pins)
- ***BufMaxTran*** *number{ns|ps}*
  - maximum input transition time for buffers (defalut 400)
- ***MaxSkew*** *number{ns|ps}*

CTS Spec.
-- Automatic Gated CTS Section cont.

- ***NoGating*** *{rising|falling|NO}*
  - *rising* : stops tracing through a gate(include buffers and inverters) and treats the gate as a rising-edge-triggered flip-flop clock pin.
  - *falling*: stops tracing through a gate(include buffers and inverters) and treats the gate as a falling-edge-triggered flip-flop clock pin.
  - *No*: Allows CTS to trace through clock gating logic. (default)



- ***AddDriverCell*** *driver_cell_name*
  - Place a driver cell at the cloest possible location to the clock port location .



CTS Spec. -- Automatic Gated CTS Section cont.

- ***MaxDepth*** *number*
- ***RouteType*** *routeTypeName*
- ***RouteClkNet*** *YES|NO*
  - Specifies whether CTS routes clock nets.
- **PostOpt** *YES|NO*
  - whether CTS resizes buffers of inverters , refines placement,and corrects routing for signal and clock wires.
  - default YES
- ***Buffer*** *cell1 cell2 cell3 …*
  - Specifies the names of buffer cells to use during CTS.

CTS Spec.
-- Automatic Gated CTS Section cont.

- ***LeafPin***

  + *pinName rising|falling*

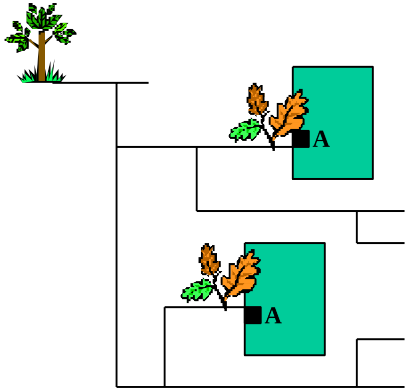
  + ……

  - ➢ Mark the pin as a "leaf" pin for non-clock-type instances.
  - ➢ LeafPin

    + instance1/A rising

    + instance2/A rising

    ……



- ***LeafPort***

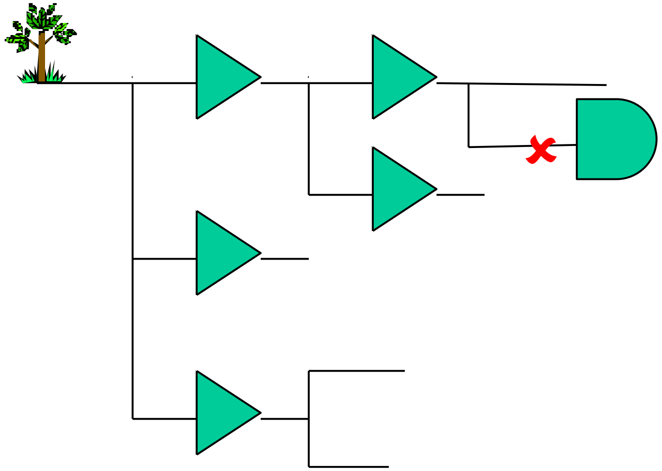
  + *portName rising|faling*

  + ……

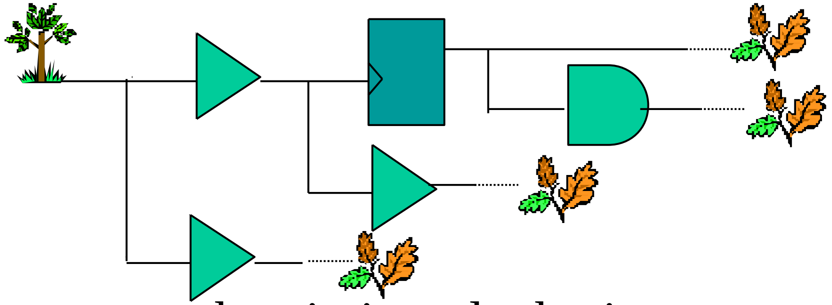
  - ➢ Mark the port as a "leaf" port for non-clock-type instances

CTS Spec. -- Automatic Gated CTS Section cont.

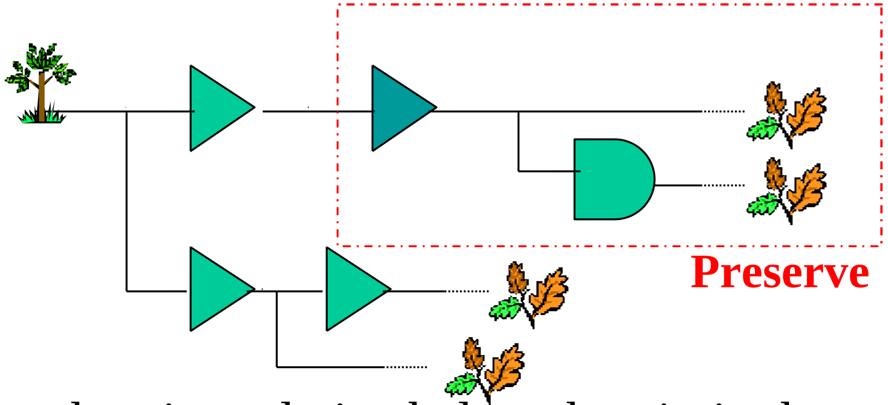
- ***ExcludedPin***
  + ***pinName***
  + ***…..***
- ***ExcludedPort***
  + ***portName***
  + ***……***
    - ➢ Treats the port as a non-leaf port, and prevents tracing and skew analysis of the pin.



CTS Spec.
-- Automatic Gated CTS Section cont.



- ***ThroughPin***
  - + *pinName*
  - + …..
    - ➢ Traces through the pin, even if the pin is a clock pin



- ***PreservePin***
  - + *inputPinName*
  - + …….
    - ➢ Preserve the netlist for the pin and pins below the pin in the clock tree.

CTS Spec.
-- Automatic Gated CTS Section cont.

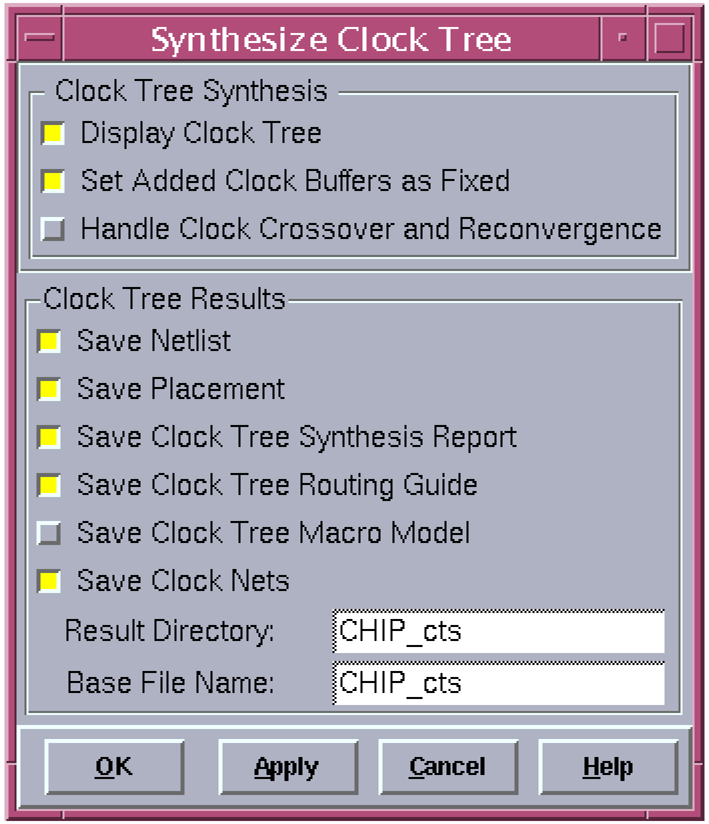
- ***DefaultMaxCap*** *capvalue*
  - ➢ CTS adheres to the following priority when using maximum capacitance value:
    - ✓ MaxCap statements in the clock tree specification file
    - ✓ DefaultMaxCap statement in the clock tree specification file
    - ✓ Maximum capacitance values in the SDC file
    - ✓ maximum capacitance values in the .lib file
- **MaxCap**

  + *bufferName1 capValue1{pf|ff}*
  + *bufferName2 capValue2{pf|ff}*
  + …..

  - ➢ Buffer should be inserted if the given capacitance value is exceeded

Synthesize Clock Tree

Clock→Synthesize Clock Tree



Synthesize Clock Tree

Clock Tree Synthesis

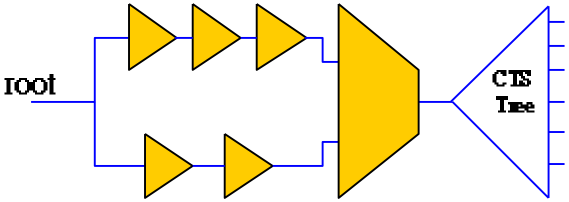
- ☑ Display Clock Tree
- ☑ Set Added Clock Buffers as Fixed
- ☐ Handle Clock Crossover and Reconvergence

Clock Tree Results

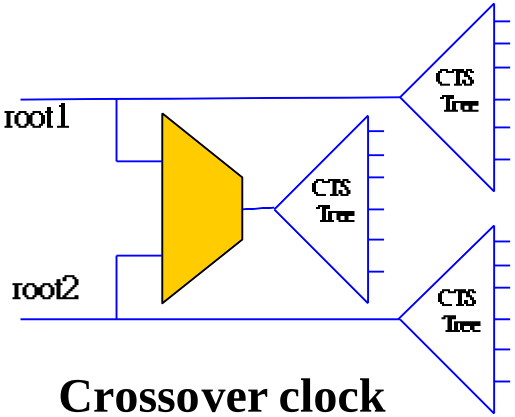
- ☑ Save Netlist
- ☑ Save Placement
- ☑ Save Clock Tree Synthesis Report
- ☑ Save Clock Tree Routing Guide
- ☐ Save Clock Tree Macro Model
- ☑ Save Clock Nets

Result Directory: CHIP_cts

Base File Name: CHIP_cts



Reconvergence clock



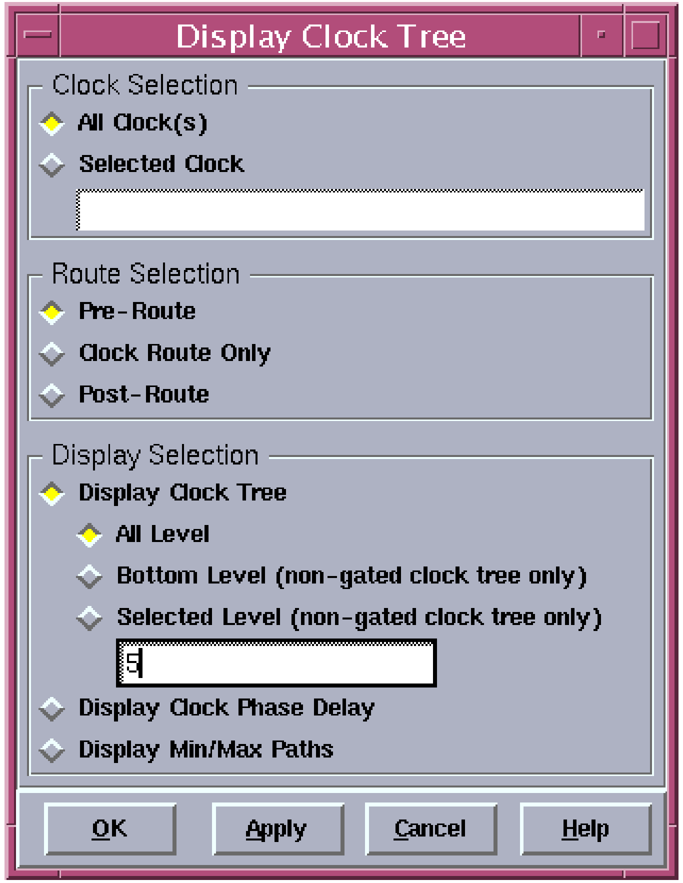
Crossover clock

Clock Synthesis report

- Summary report and detail report
  - number of sub trees
  - rise/fall insertion delay
  - trigger edge skew
  - rise/fall skew
  - buffer and clock pin transition time
  - detailed delay ranges for all buffers add to clocks
- Clock nets
  - Saves the generated clock nets
  - used to guide clock net routing
- Clock routing guide
  - Saves the clock tree routing data
  - used as preroute guide while running Trial Route

Display Clock Tree

Clock→Display→Display Clock Tree…



Display Clock Tree

Clock Selection

- ● All Clock(s)
- ○ Selected Clock

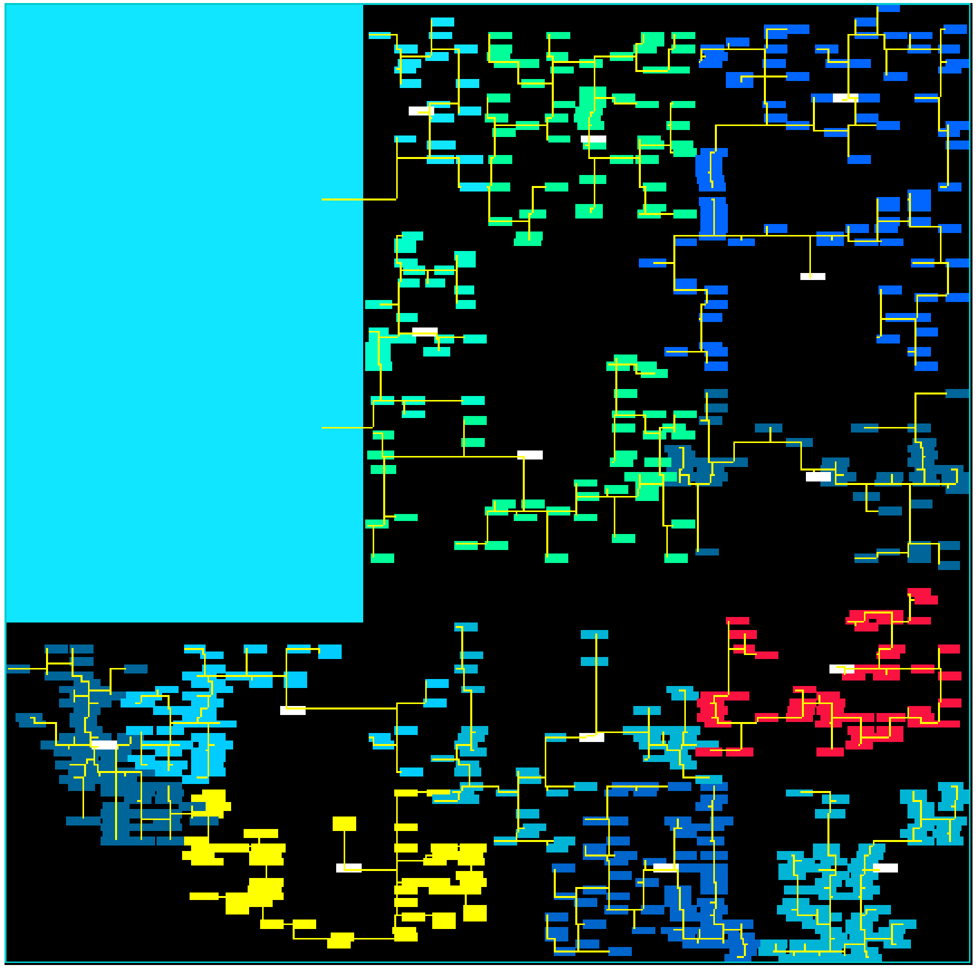
Route Selection

- ● Pre-Route
- ○ Clock Route Only
- ○ Post-Route

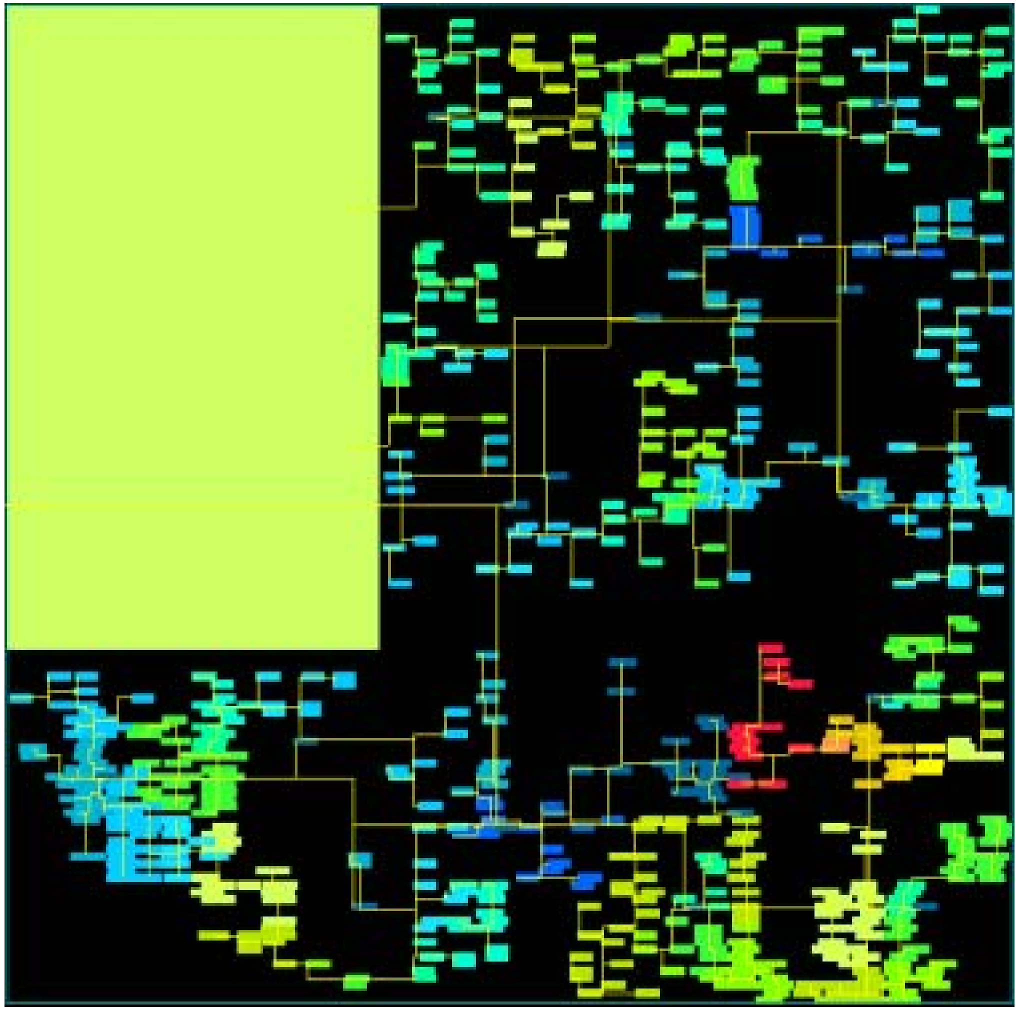
Display Selection

- ● Display Clock Tree
  - ● All Level
  - ○ Bottom Level (non-gated clock tree only)
  - ○ Selected Level (non-gated clock tree only)
  - 5
- ○ Display Clock Phase Delay
- ○ Display Min/Max Paths

Display Clock Tree
-- by level

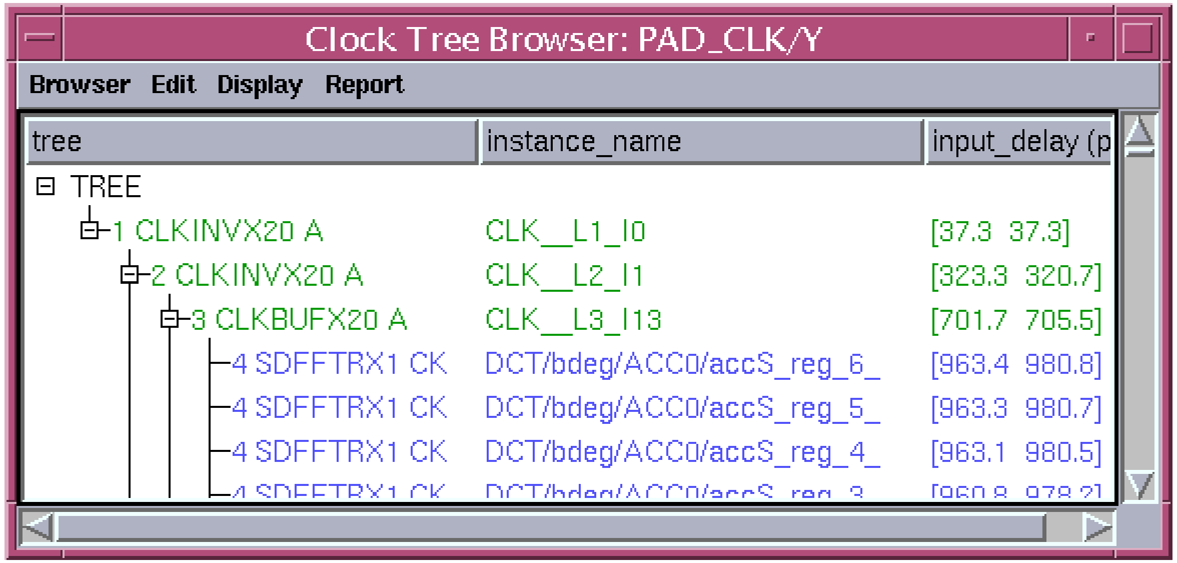


Display Clock Tree --by phase delay



Clock Tree Browser

Clock→Clock Tree Brower



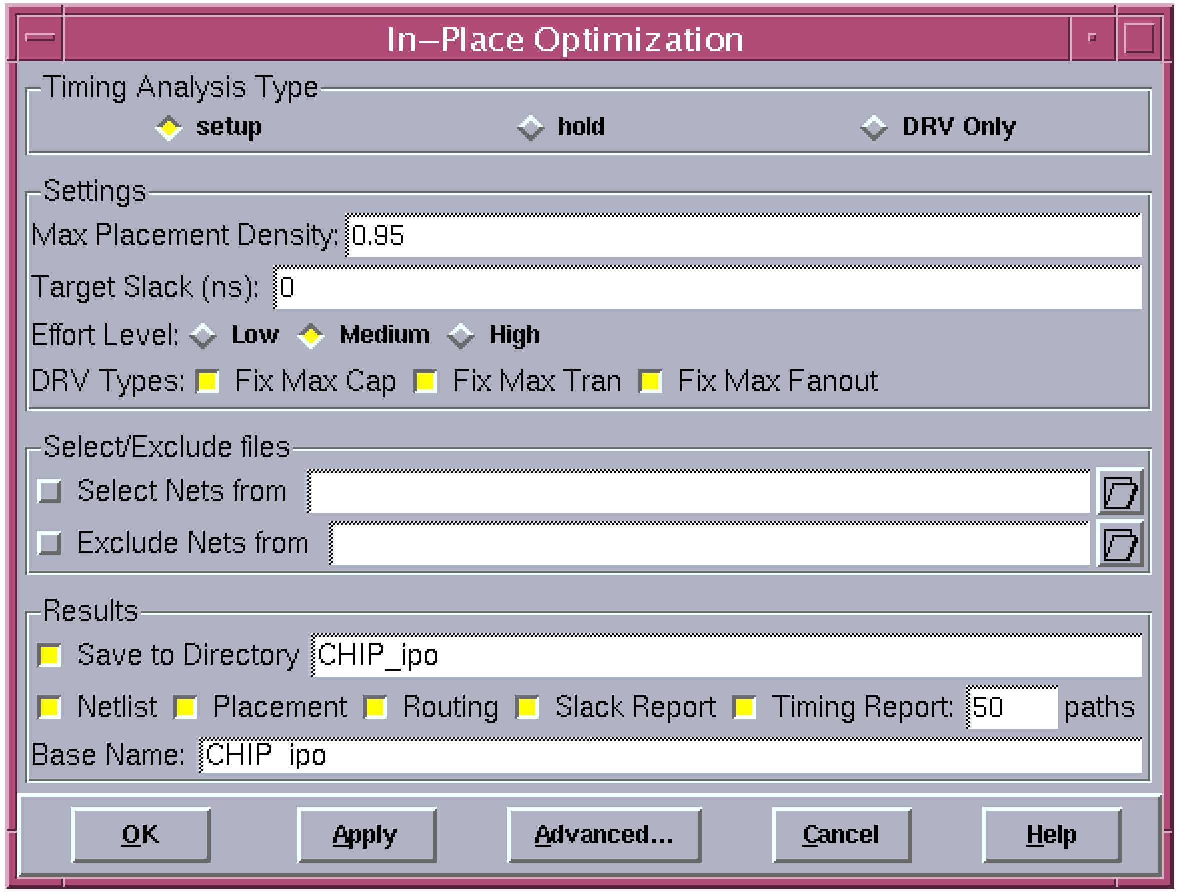
tree	instance_name	input_delay (p
⊟ TREE		
⊟ 1 CLKINVX20 A	CLK__L1_I0	[37.3 37.3]
⊟ 2 CLKINVX20 A	CLK__L2_I1	[323.3 320.7]
⊟ 3 CLKBUFX20 A	CLK__L3_I13	[701.7 705.5]
4 SDFFTRX1 CK	DCT/bdeg/ACC0/accS_reg_6_	[963.4 980.8]
4 SDFFTRX1 CK	DCT/bdeg/ACC0/accS_reg_5_	[963.3 980.7]
4 SDFFTRX1 CK	DCT/bdeg/ACC0/accS_reg_4_	[963.1 980.5]
4 SDFFTRX1 CK	DCT/bdeg/ACC0/accS_reg_3_	[960.8 978.2]

- ◆ Display trig edge, rise/fall delay, rise/fall skew, input delay, input tran of each cell.
- ◆ Resize/Delete leaf cell or clock buffer
- ◆ Reconnect clock tree

In-Place Optimization

Timing→In-Place Optimization…

- ◆ IPO
  - ➢ setup time
  - ➢ hold time
  - ➢ DRV (Design Rule Violation)



Timing Analysis Type

● setup ○ hold ○ DRV Only

Settings

Max Placement Density: 0.95

Target Slack (ns): 0

Effort Level: ○ Low ● Medium ○ High

DRV Types: ☑ Fix Max Cap ☑ Fix Max Tran ☑ Fix Max Fanout

Select/Exclude files

□ Select Nets from

□ Exclude Nets from

Results

☑ Save to Directory CHIP_ipo

☑ Netlist ☑ Placement ☑ Routing ☑ Slack Report ☑ Timing Report: 50 paths

Base Name: CHIP_ipo

Congestion Optimization

encounter > ***congOpt***
[*-nrIterInCongOpt nrIter*]
[*-maxCPUTimeInCongOpt time*]

- Reduces congestion after placement in an iterative way.
- Parameters
  - nrIterInCongOpt nrIter
    - ✓ Specifies the total number of iteration in congestion optimization. (default 1)
  - maxCPUTimeInCongOpt
    - ✓ specifies the maximum CPU time in congestion optimization,in hours.

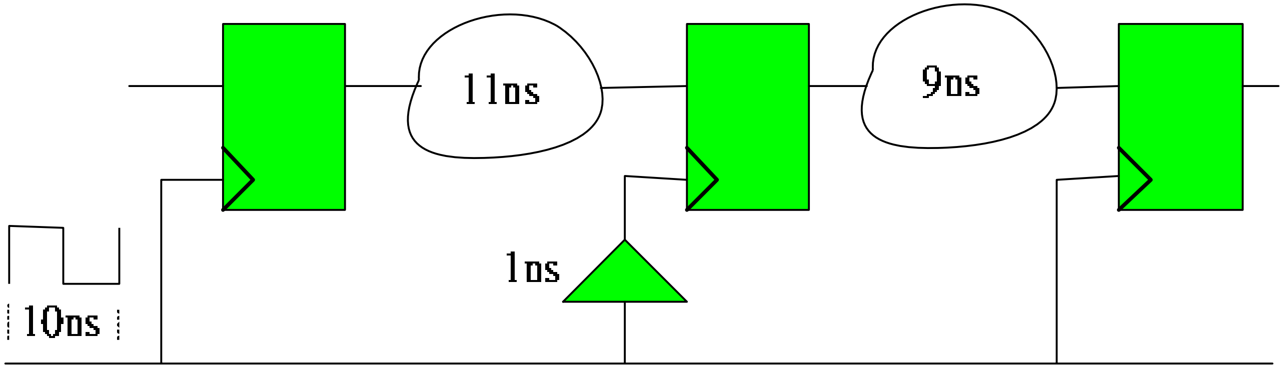
Balance Slew

```
encounter >  balanceSlew
                 [ -selNetFile selNetFileName ]
                 [ -excNetFile excNetFileName ]
```

- Speeds up or slows down the transition time if it is greater or less than the specified maximum transition time.
- Parameters
  - selNetFile selNetFileName
    - ✓ Specifies th file that contains the hierarchical net names that are excluded from the IPO pperation
  - excNetFile excNetFileName
    - ✓ Specifies the file that contains the hierarchical net (path) names for the IPO operations. Only these net names are considered.

Useful Skew

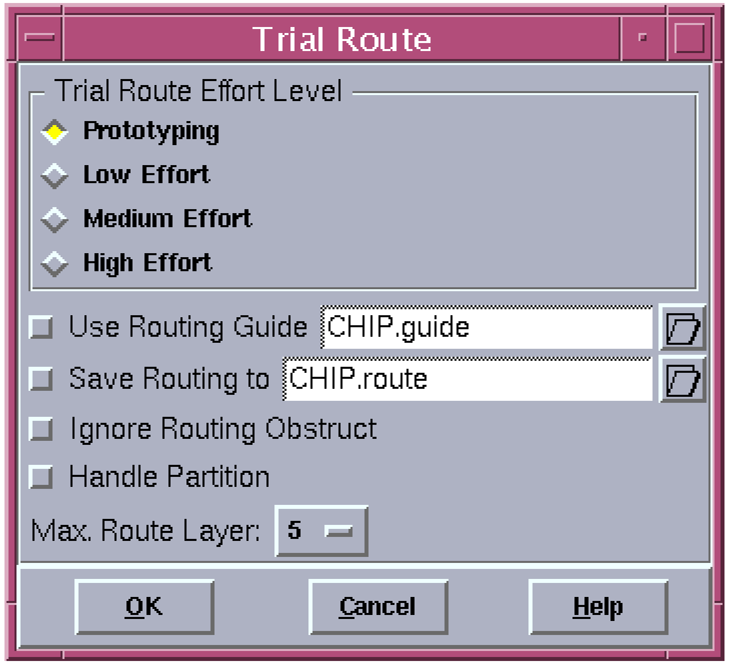
```
encounter >  setAnalysisMode -usefulSkew
encounter >  skewClock
encounter >  optCritPath
```



balanced clock

Trial Route

- perform quick routing for congestion and parasitics estimation
- Prototyping:
  - Quickly to gauge the feasibility of netlist.
  - components in design might no be routed at legal location



Trial Route

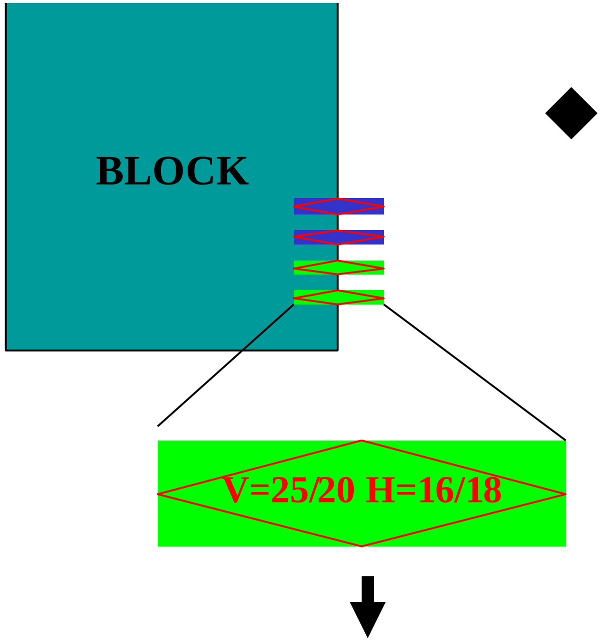
Trial Route Effort Level

● Prototyping
○ Low Effort
○ Medium Effort
○ High Effort

□ Use Routing Guide CHIP.guide
□ Save Routing to CHIP.route
□ Ignore Routing Obstruct
□ Handle Partition

Max. Route Layer: 5

Trial Route Congestion Marker



- visually check the congestion statistics.
- dump congestion area:
  - ***dumpCongesArea -all file_name***

The vertical (V) overflow is 25/20 (25 tracks are required , but only 20 tracks are available) .
The Horizontal (H) overflow is 16/18 (16 tracks are required , and18 tracks are available) .

Trial Route Congestion Marker cont.

Level	Color	Overflow Value
1	● Blue	One more track required
2	● Green	Two more track required
3	● Yellow	Three more track required
4	● Red	Four more track required
5	● Magenta	Five more track required
6 and higher	● Grey to White	Six or more track required

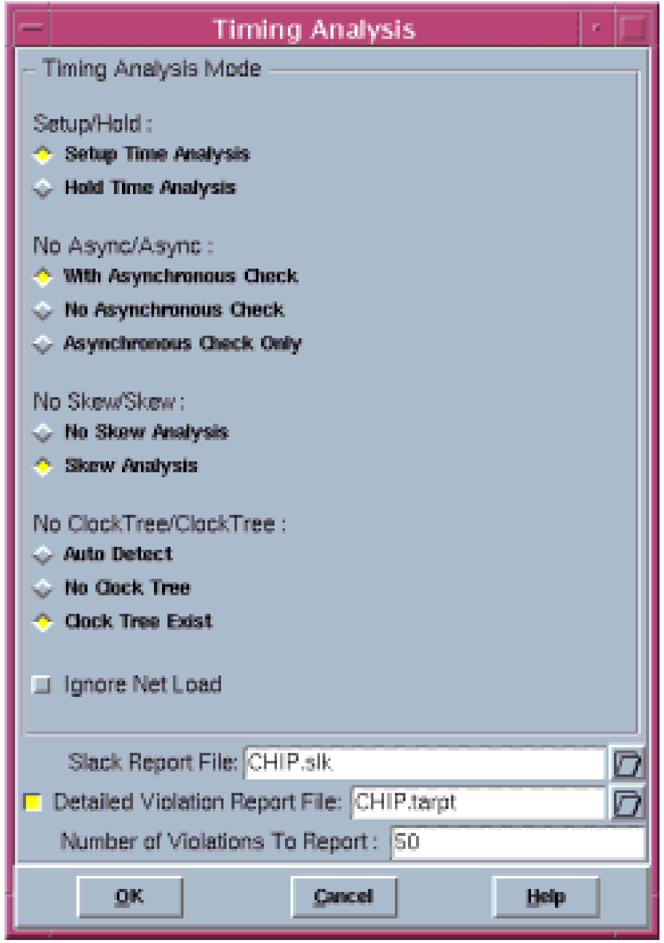
Timing Analysis

Timing→Specify Analysis Condition→Specify RC Extraction Mode …

Timing→Extract RC…

Timing→Timing Analysis…

- No Async/Async:
  - recovery, removal check
- No Skew/Skew:
  - check with/without clock skew constraint



Timing Analysis

Timing Analysis Mode

Setup/Hold :

◆ Setup Time Analysis

◇ Hold Time Analysis

No Async/Async :

◆ With Asynchronous Check

◇ No Asynchronous Check

◇ Asynchronous Check Only

No Skew/Skew :

◇ No Skew Analysis

◆ Skew Analysis

No ClockTree/ClockTree :

◇ Auto Detect

◇ No Clock Tree

◆ Clock Tree Exist

□ Ignore Net Load

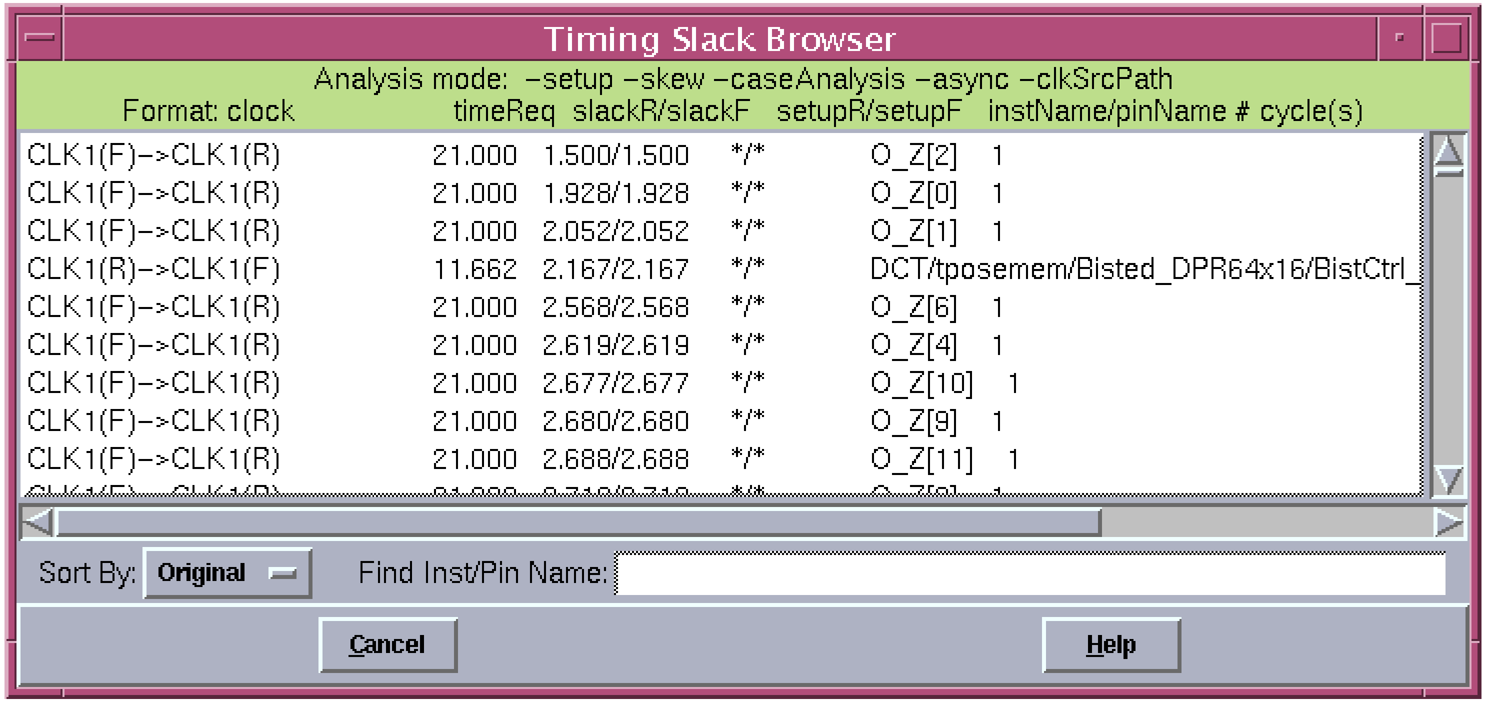
Slack Report File: CHIP.slk

■ Detailed Violation Report File: CHIP.tarpt

Number of Violations To Report : 50

Slack Browser

Timing→Timing Debug→Slack Browser…



Timing Slack Browser

```
Analysis mode:  -setup -skew -caseAnalysis -async -clkSrcPath
Format: clock                 timeReq  slackR/slackF   setupR/setupF   instName/pinName # cycle(s)
CLK1(F)->CLK1(R)          21.000   1.500/1.500    */*          O_Z[2]    1
CLK1(F)->CLK1(R)          21.000   1.928/1.928    */*          O_Z[0]    1
CLK1(F)->CLK1(R)          21.000   2.052/2.052    */*          O_Z[1]    1
CLK1(R)->CLK1(F)          11.662   2.167/2.167    */*          DCT/tposemem/Bisted_DPR64x16/BistCtrl_
CLK1(F)->CLK1(R)          21.000   2.568/2.568    */*          O_Z[6]    1
CLK1(F)->CLK1(R)          21.000   2.619/2.619    */*          O_Z[4]    1
CLK1(F)->CLK1(R)          21.000   2.677/2.677    */*          O_Z[10]    1
CLK1(F)->CLK1(R)          21.000   2.680/2.680    */*          O_Z[9]    1
CLK1(F)->CLK1(R)          21.000   2.688/2.688    */*          O_Z[11]    1
```

Sort By: Original

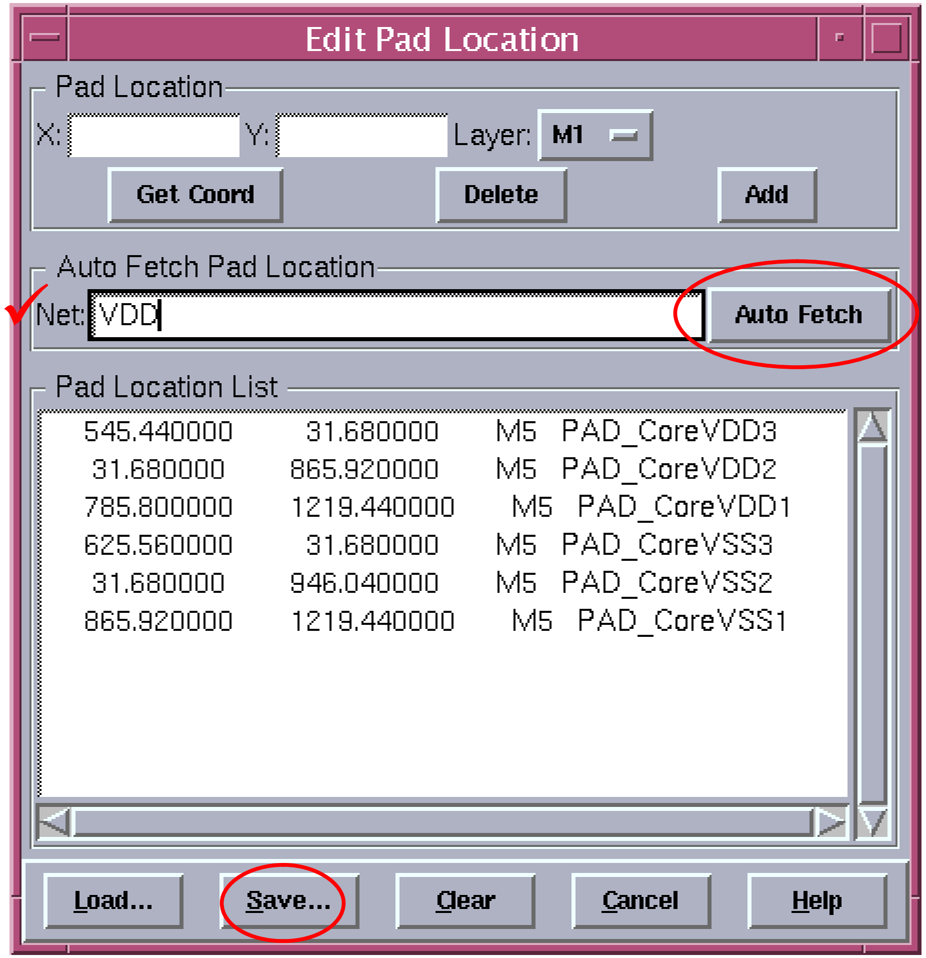
Find Inst/Pin Name:

Power Analysis

Timing→Extract RC…

Power→Edit Pad Location…

Power→Edit Net Toggle Probability…



Edit Pad Location

Pad Location

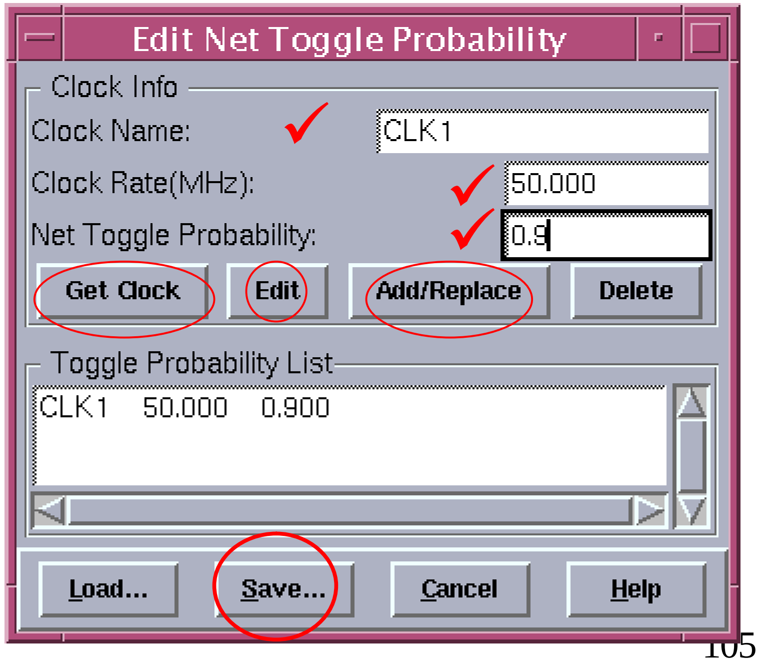
X: Y: Layer: M1

Auto Fetch Pad Location

✓ Net: VDD

Pad Location List

```
545.440000    31.680000     M5  PAD_CoreVDD3
 31.680000    865.920000    M5  PAD_CoreVDD2
785.800000    1219.440000   M5  PAD_CoreVDD1
625.560000    31.680000     M5  PAD_CoreVSS3
 31.680000    946.040000    M5  PAD_CoreVSS2
865.920000    1219.440000   M5  PAD_CoreVSS1
```



Edit Net Toggle Probability

Clock Info

Clock Name: ✓ CLK1

Clock Rate(MHz): ✓ 50.000

Net Toggle Probability: ✓ 0.9

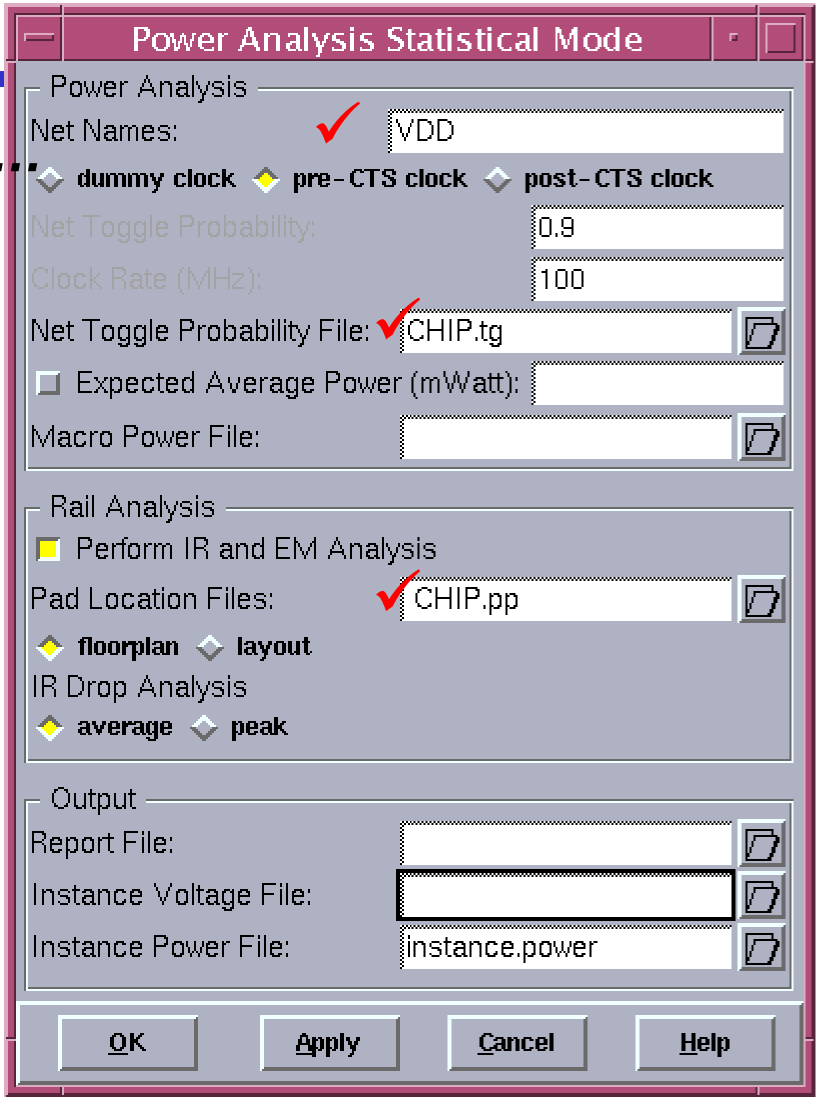
Toggle Probability List

```
CLK1   50.000   0.900
```

Statistical Power Analysis

Power→Power Analysis→Statistical …

- analysis report:
  - A power graph
  - report contains
    - ✓ average power usage
    - ✓ worst IR drop
    - ✓ worst EM violation
  - instance power file
  - instance voltage file
  - boundary voltage file



Power Analysis Statistical Mode

Power Analysis

Net Names: ✓ VDD

○ dummy clock ● pre-CTS clock ○ post-CTS clock

Net Toggle Probability: 0.9

Clock Rate (MHz): 100

Net Toggle Probability File: ✓ CHIP.tg

☐ Expected Average Power (mWatt):

Macro Power File:

Rail Analysis

☑ Perform IR and EM Analysis

Pad Location Files: ✓ CHIP.pp

● floorplan ○ layout

IR Drop Analysis

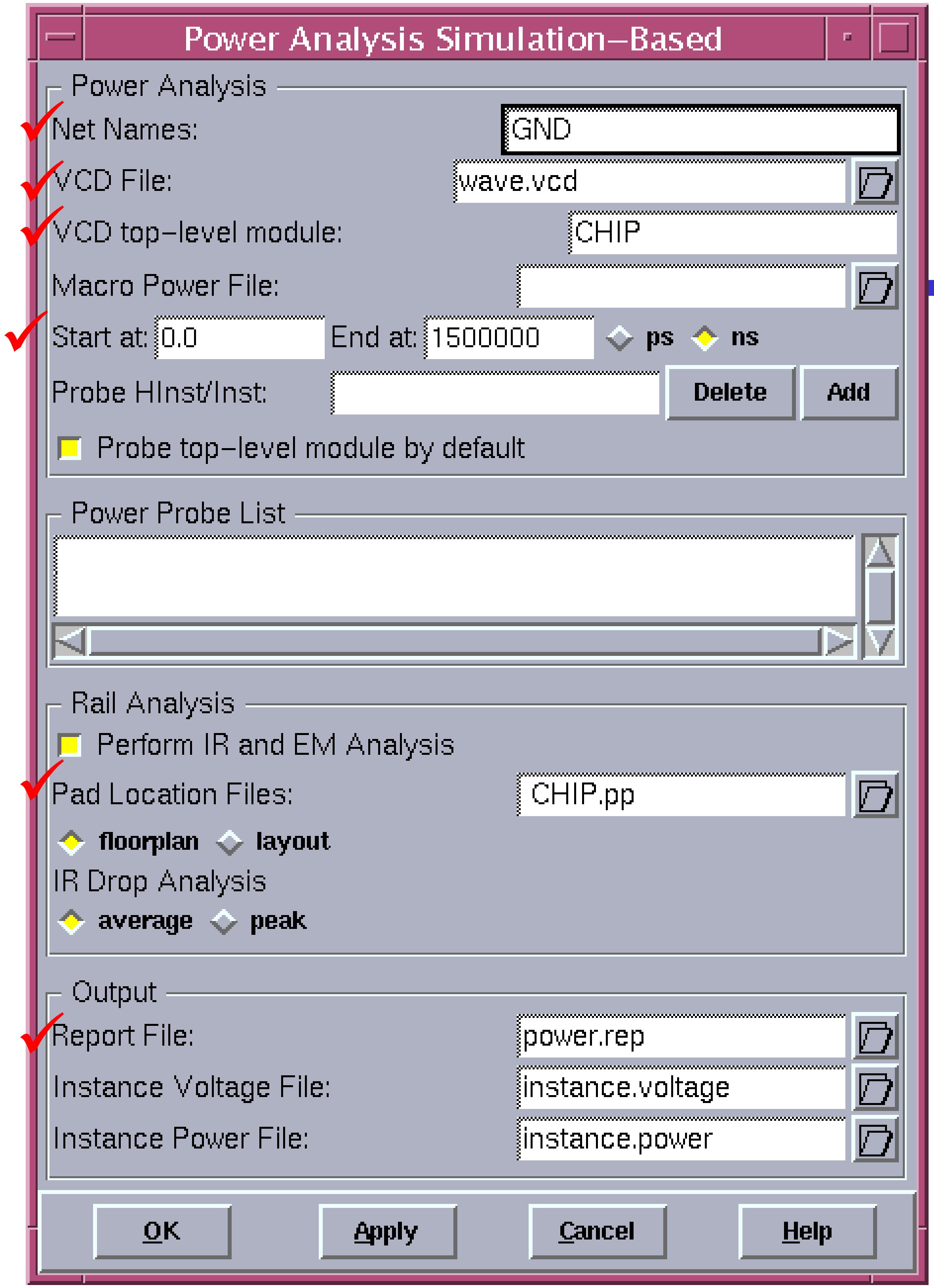
● average ○ peak

Output

Report File:

Instance Voltage File:

Instance Power File: instance.power



Power Analysis Simulation–Based

Power Analysis

✓ Net Names: GND

✓ VCD File: wave.vcd

✓ VCD top–level module: CHIP

Macro Power File:

✓ Start at: 0.0 End at: 1500000 ○ ps ● ns

Probe HInst/Inst:

☑ Probe top–level module by default

Power Probe List

Rail Analysis

☑ Perform IR and EM Analysis

✓ Pad Location Files: CHIP.pp

● floorplan ○ layout

IR Drop Analysis

● average ○ peak

Output

✓ Report File: power.rep

Instance Voltage File: instance.voltage

Instance Power File: instance.power

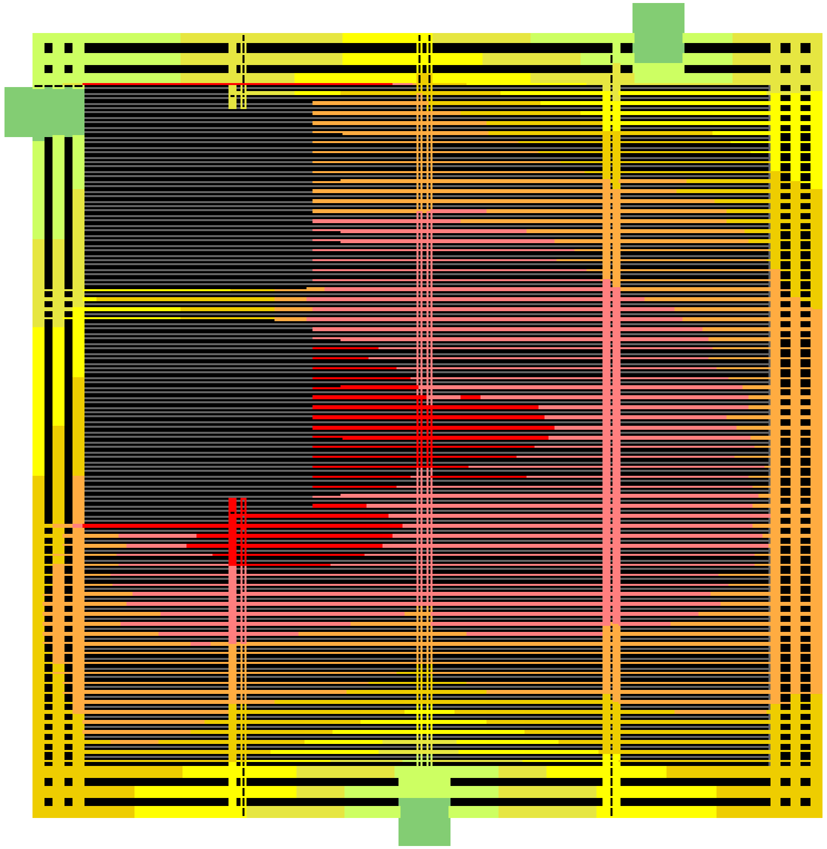
Simlation-Based Power Analysis

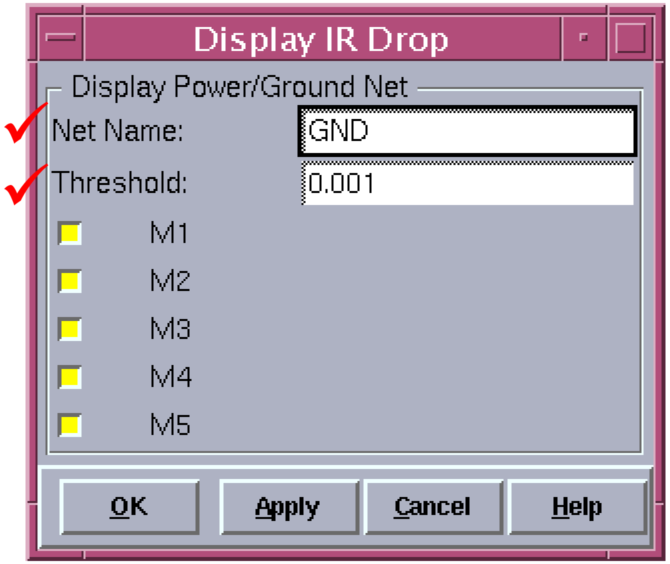
Power→Power Analysis→ Simulation-Based

- save netlist for simulation
  - ***Design→Save→Netlist…***
- simulation and dump vcd file.
  - $dumpvars;
  - $dumpfile(“wave.vcd”);
- Input vcd file for power analysis

Display IR Drop

Power→Display→Display IR Drop…





Display IR Drop

Display Power/Ground Net

✓ Net Name: GND

✓ Threshold: 0.001

☐ M1

☐ M2

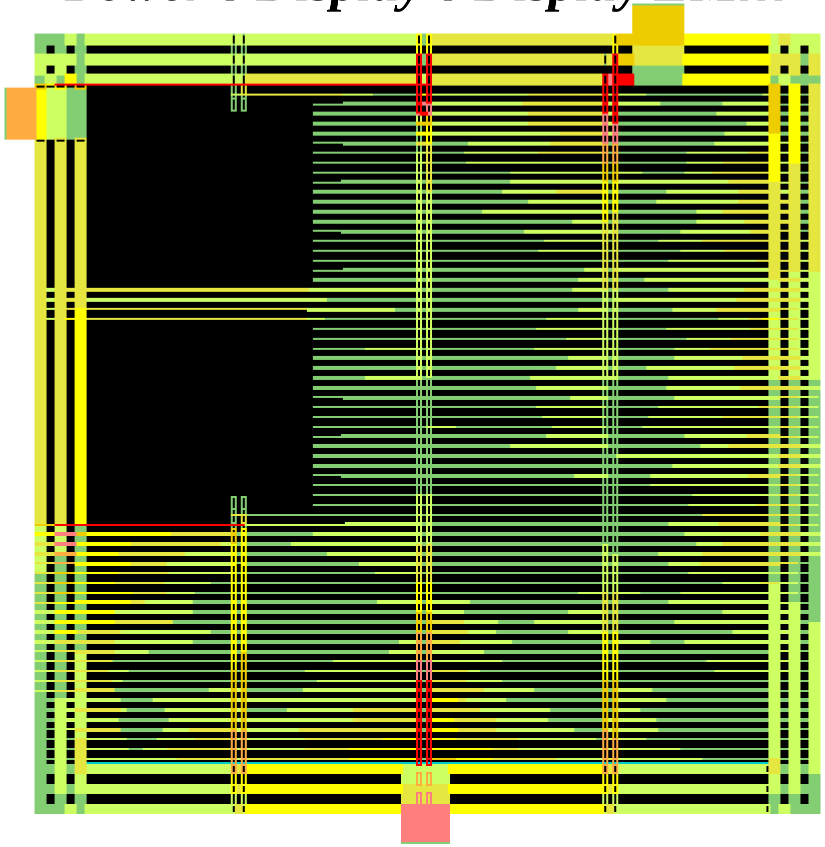
☐ M3

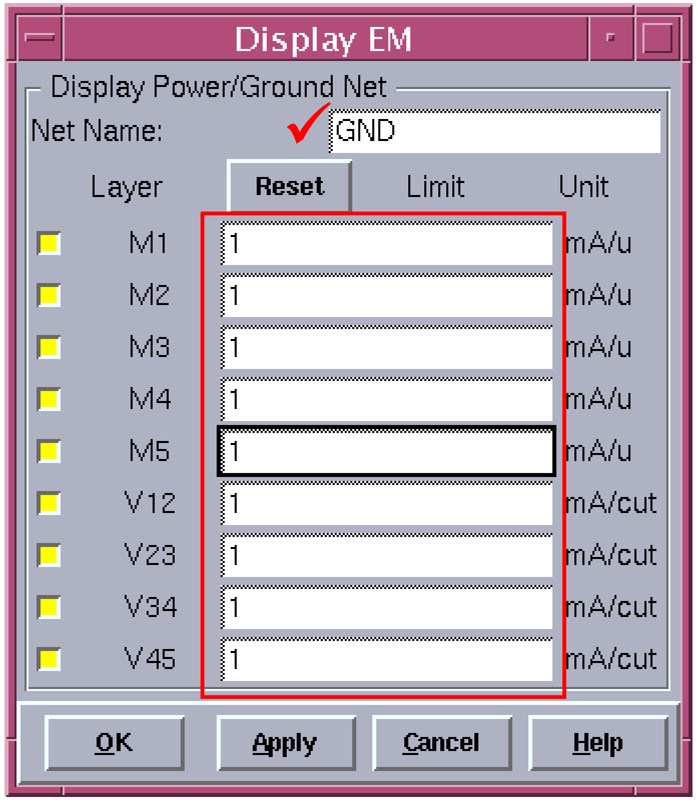
☐ M4

☐ M5

Display Electron Migration

Power→Display→Display EM…





Display EM

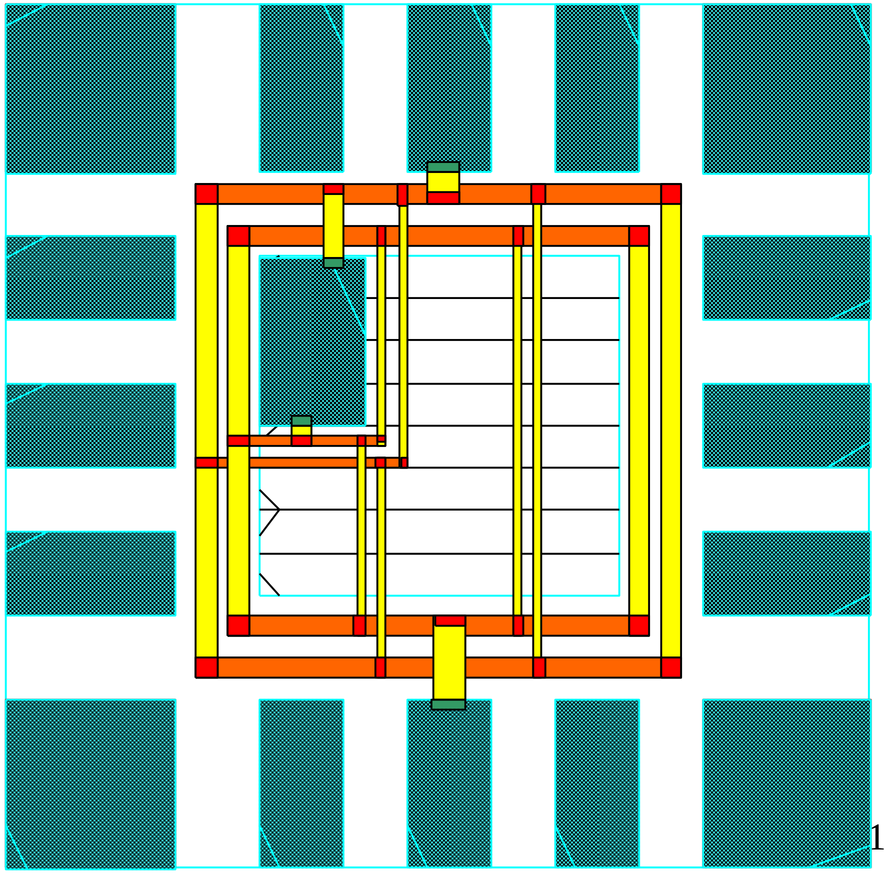
Display Power/Ground Net

Net Name: ✓ GND

	Layer	Reset	Limit	Unit
☑	M1		1	mA/u
☑	M2		1	mA/u
☑	M3		1	mA/u
☑	M4		1	mA/u
☑	M5		1	mA/u
☑	V12		1	mA/cut
☑	V23		1	mA/cut
☑	V34		1	mA/cut
☑	V45		1	mA/cut

SRoute

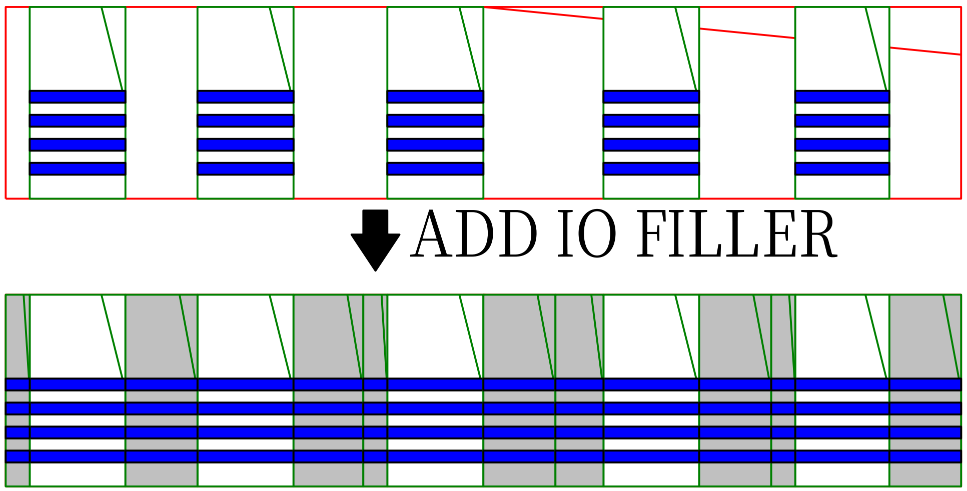
- Route Special Net (power/ground net)
  - Block pins
  - Pad pins
  - Pad rings
  - Standard cell pins
  - Stripes (unconnected)



Add IO filler

```
addIoFiller –cell PFILL –prefix IOFILLER
addIoFiller –cell PFILL_9 –prefix IOFILLER
addIoFiller –cell PFILL_1 –prefix IOFILLER
addIoFiller –cell PFILL_01 –prefix IOFILLER -fillAnyGap
```

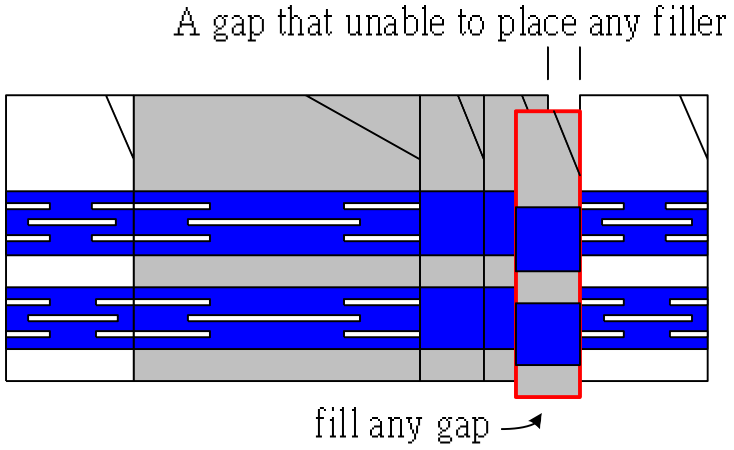
- Connect io pad power bus by inserting IO filler.
- Add from wider filler to narrower filler.



ADD IO FILLER

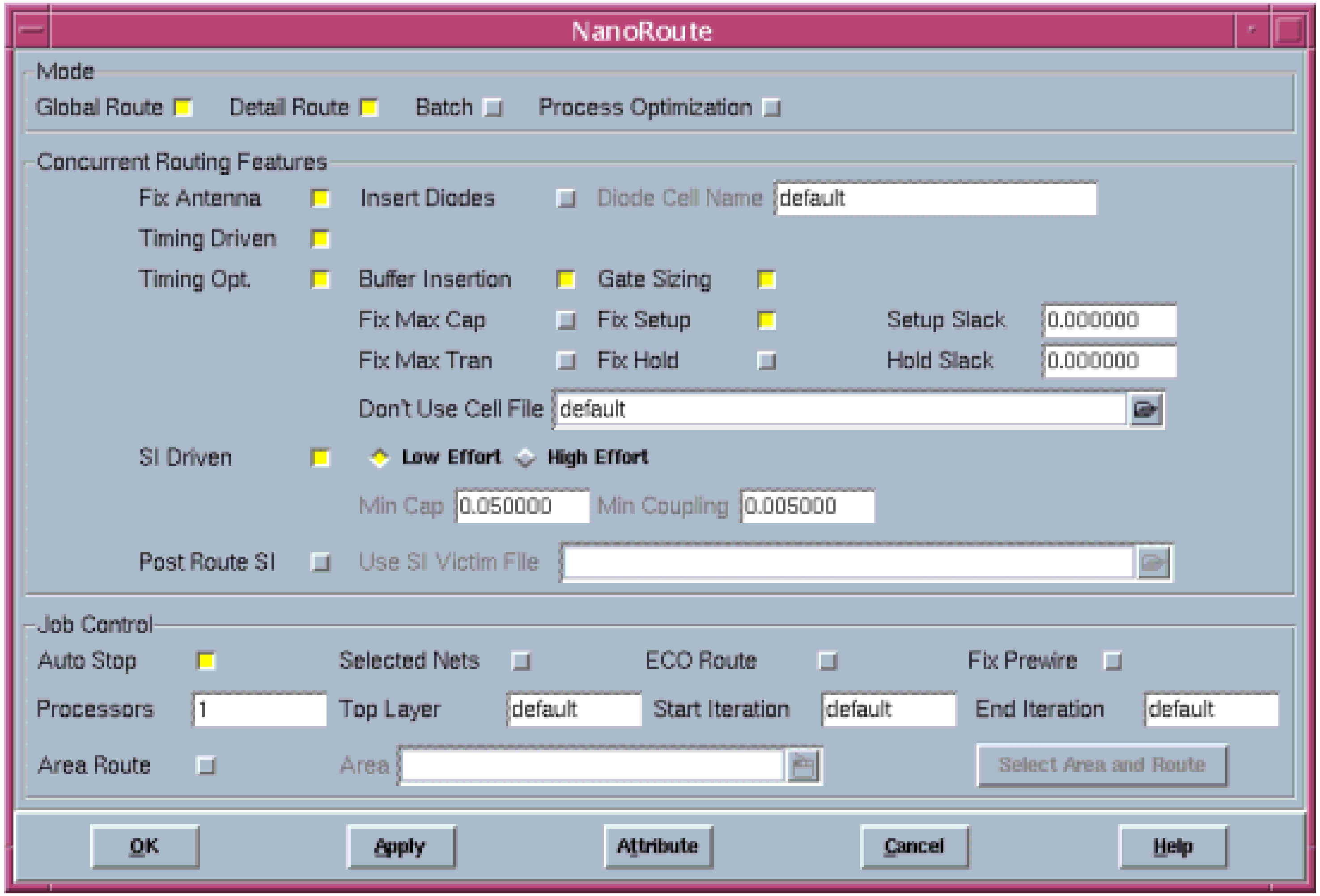
Add IO filler cont.

- ◆ In order to avoid DRC error
  - ➢ The sequence of placing fillers must be from wider fillers to narrower ones.
  - ➢ Only the smallest filler can use -fillAnyGap option.



NanoRoute

Route→NanoRoute



NanoRoute

Mode

Global Route ☑ Detail Route ☑ Batch □ Process Optimization □

Concurrent Routing Features

Fix Antenna ☑ Insert Diodes □ Diode Cell Name default

Timing Driven ☑

Timing Opt. ☑ Buffer Insertion ☑ Gate Sizing ☑

Fix Max Cap □ Fix Setup ☑ Setup Slack 0.000000

Fix Max Tran □ Fix Hold □ Hold Slack 0.000000

Don't Use Cell File default

SI Driven ☑ ● Low Effort ○ High Effort

Min Cap 0.050000 Min Coupling 0.005000

Post Route SI □ Use SI Victim File

Job Control

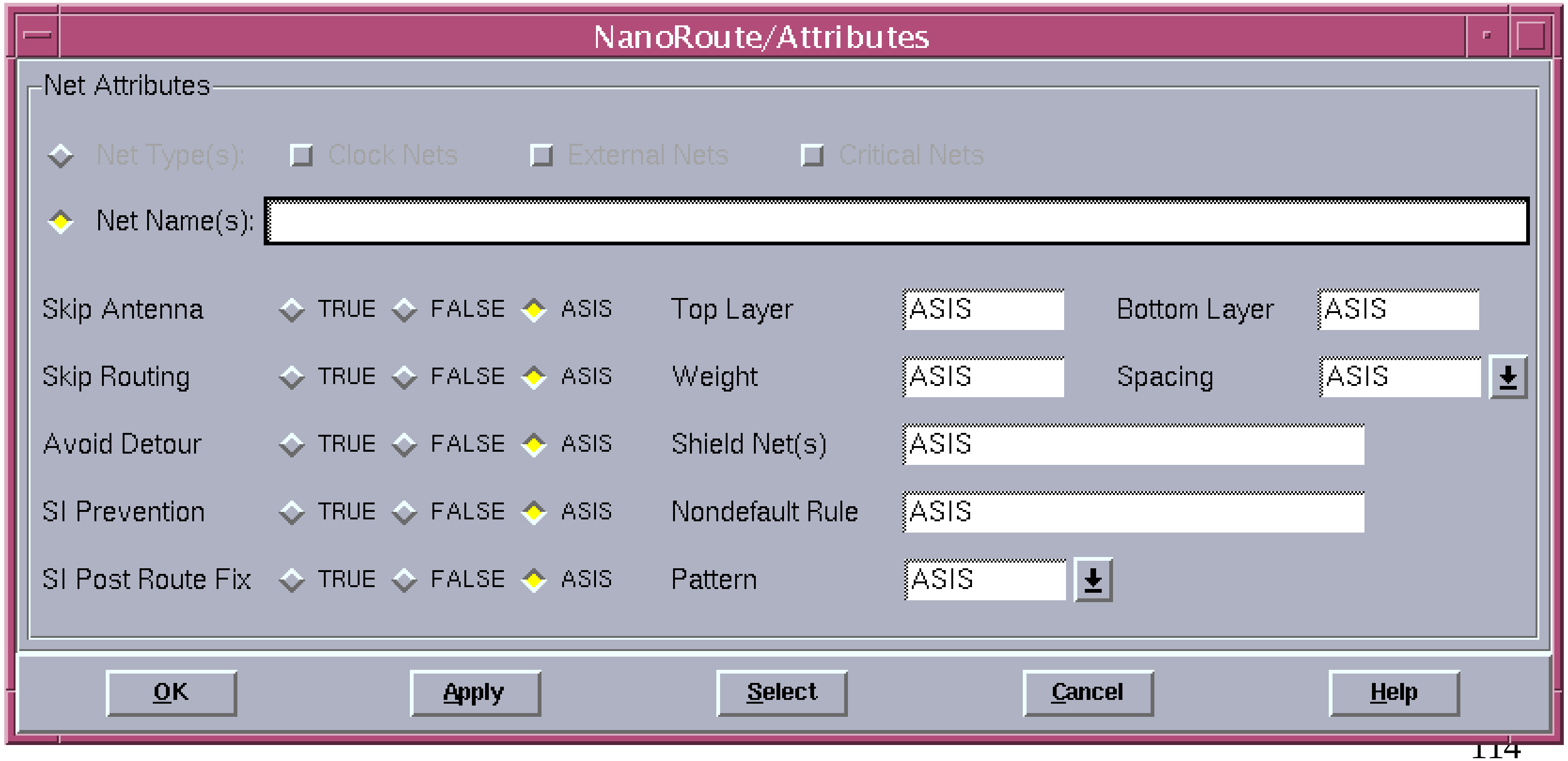
Auto Stop ☑ Selected Nets □ ECO Route □ Fix Prewire □

Processors 1 Top Layer default Start Iteration default End Iteration default

Area Route □ Area

NanoRoute Attributes

Route→NanoRoute/Attributes



NanoRoute/Attributes

Net Attributes

○ Net Type(s): □ Clock Nets □ External Nets □ Critical Nets

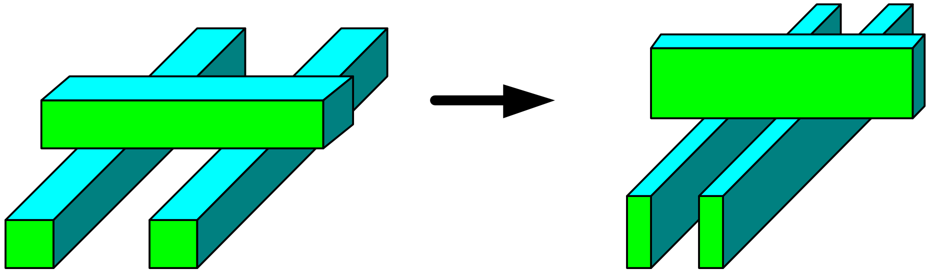
● Net Name(s):

Skip Antenna	○ TRUE	○ FALSE	● ASIS	Top Layer	ASIS	Bottom Layer	ASIS
Skip Routing	○ TRUE	○ FALSE	● ASIS	Weight	ASIS	Spacing	ASIS
Avoid Detour	○ TRUE	○ FALSE	● ASIS	Shield Net(s)	ASIS		
SI Prevention	○ TRUE	○ FALSE	● ASIS	Nondefault Rule	ASIS		
SI Post Route Fix	○ TRUE	○ FALSE	● ASIS	Pattern	ASIS		

Crosstalk

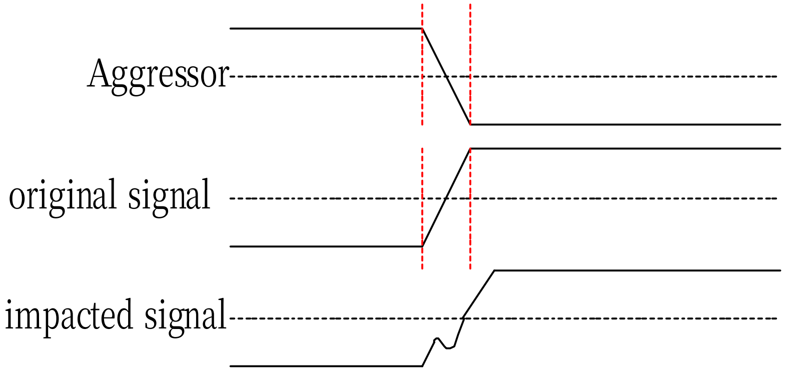
Crosstalk problem are getting more serious in 0.25um and below for:

- Smaller pitches
- Greater height/width ratio
- Higher design frequency

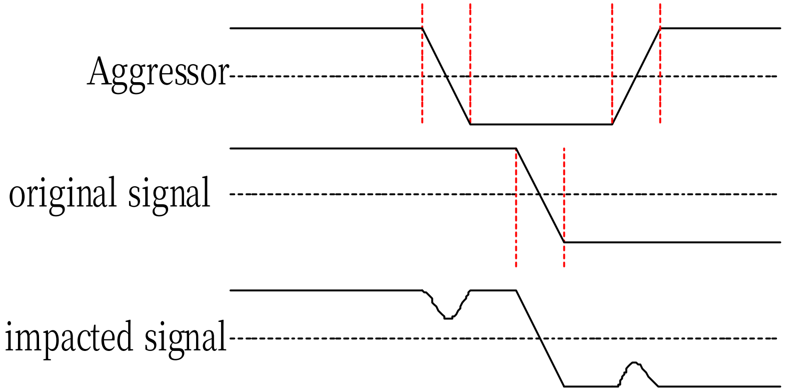


Crosstalk Problem

- Delay problem

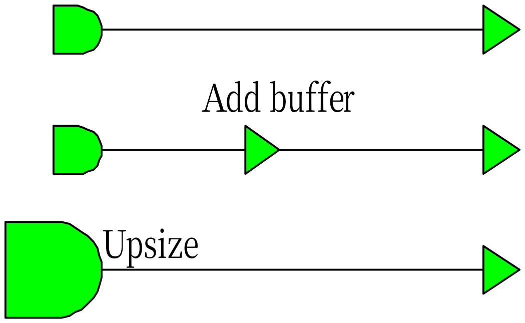


- Noise problem

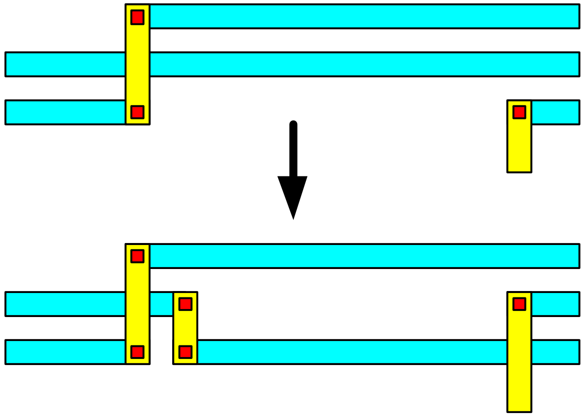


Crosstalk Prevention

- Placement solution
  - Insert buffer in lines
  - Upsize driver
  - Congestion optimization



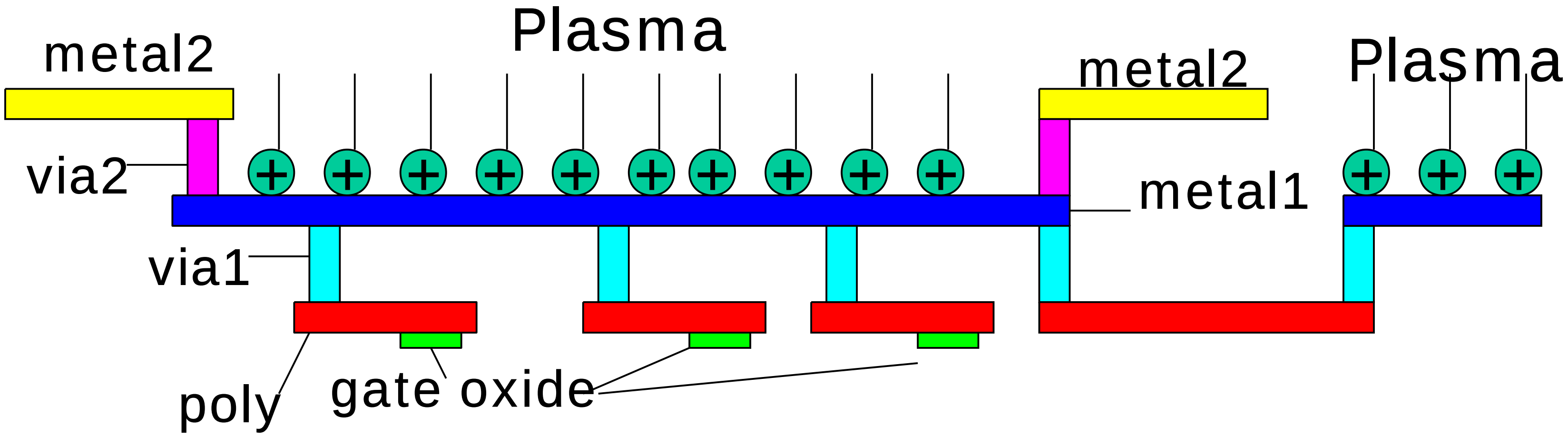
- Routing solution
  - Limit length of parallel nets
  - Wider routing grid
  - Shield special nets



Antenna Effect

- In a chip manufacturing process, Metal is initially deposited so it covers the entire chip.
- Then, the unneeded portions of the metal are removed by etching, typically in plasma(charged particles).
- The exposed metal collect charge from plasma and form voltage potential.
- If the voltage potential across the gate oxide becomes large enough, the current can damage the gate oxide.

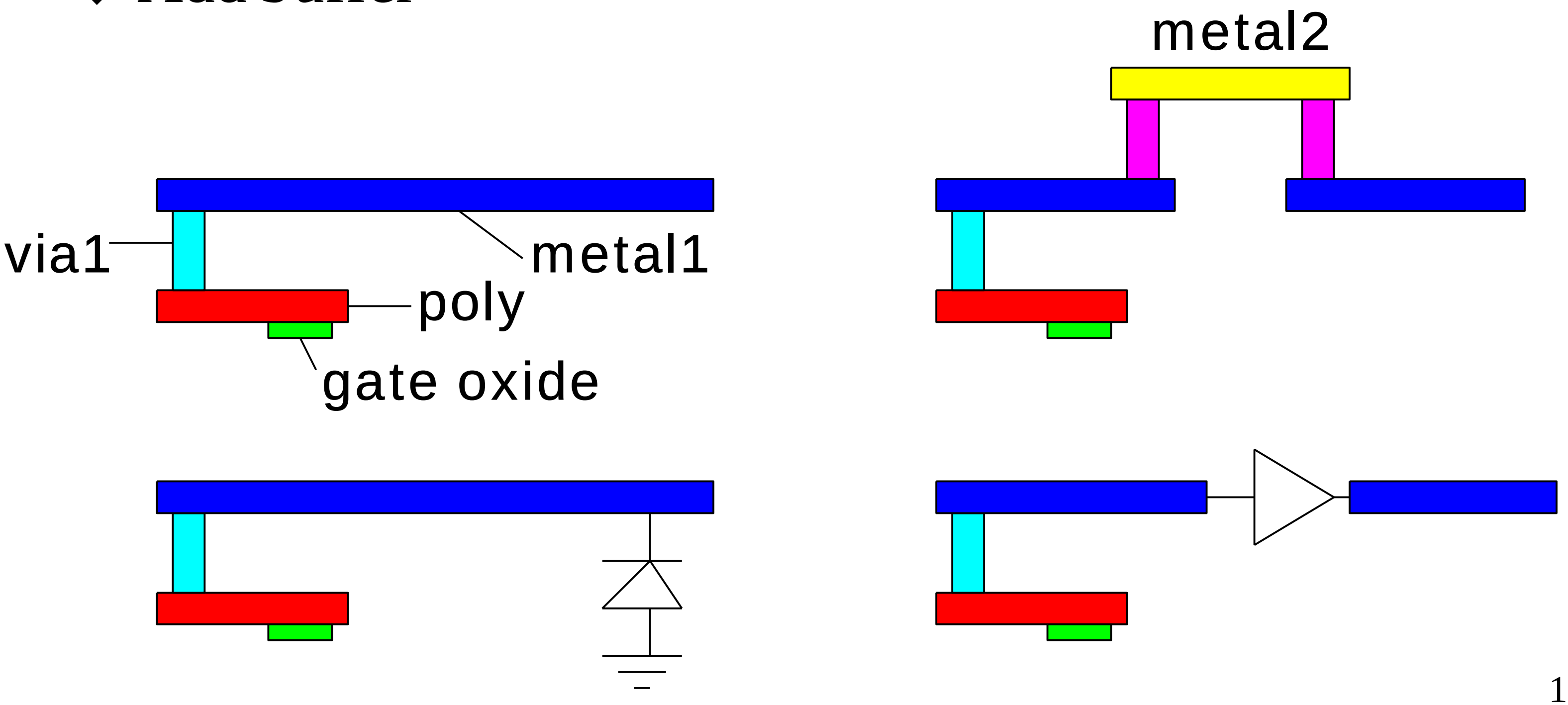
Antenna Ratio



$$\text{Antenna Ratio} = \frac{\text{Area of process antennas on a node}}{\text{Area of gates to the node}}$$

Antenna Problem Repair

- Add jumper
- Add antenna cell (diode)
- Add buffer

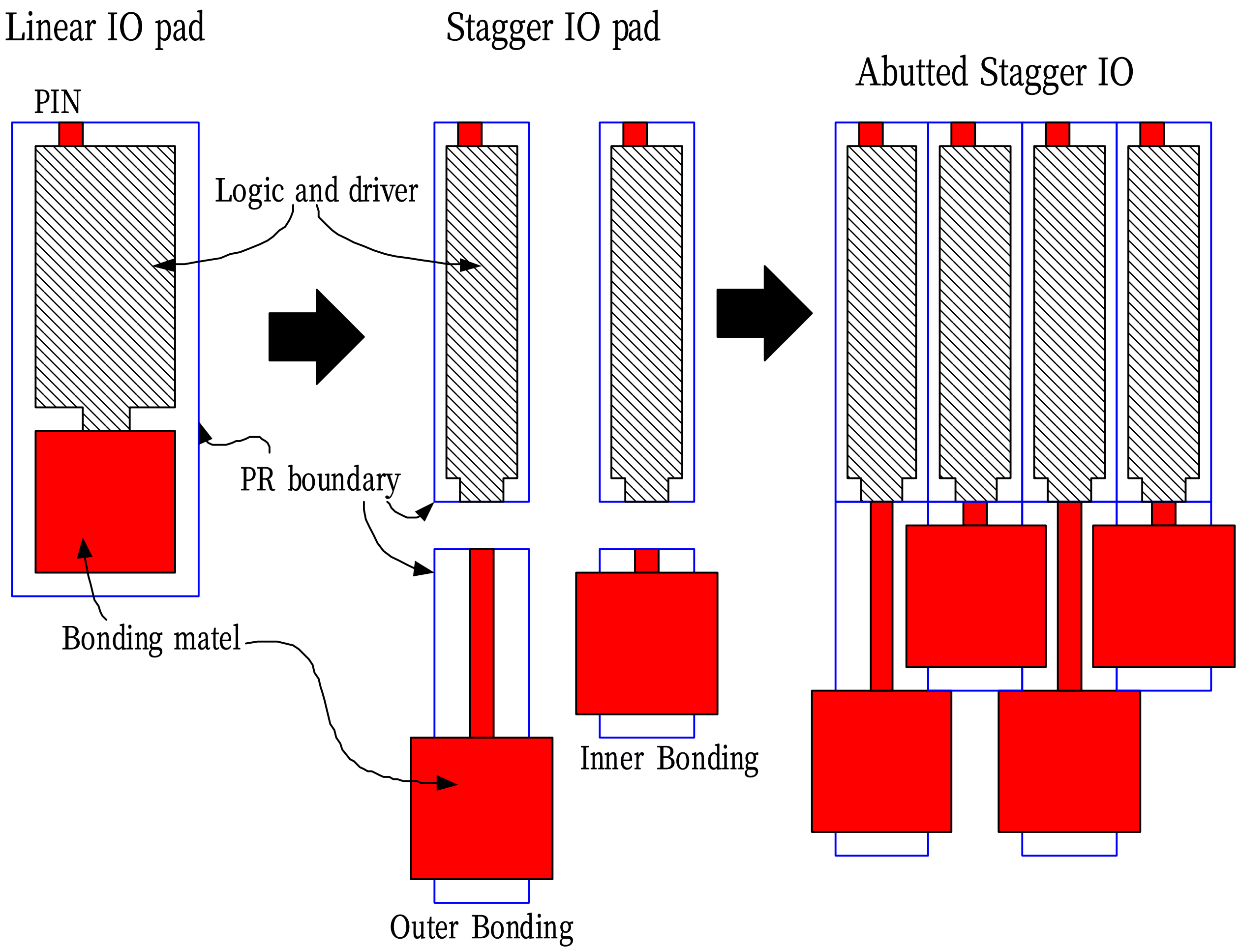


Add Core Filler

Place→Filler→Add Filler…

- Connect the NWELL/PWELL layer in core rows.
- Insert Well contact.
- Add from wider filler to narrower filler.

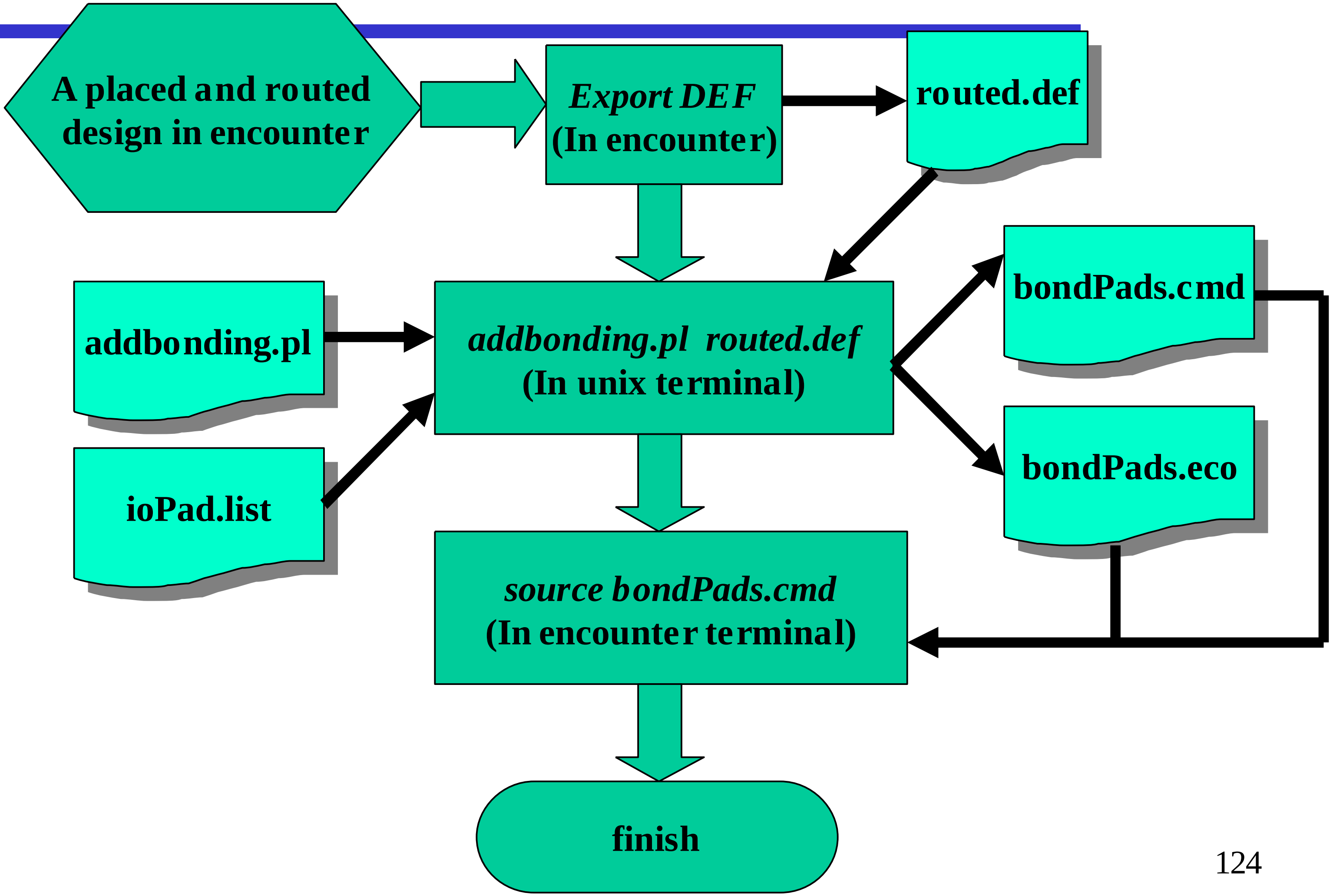
Add bonding pads (stagger IO pads only)



Add bonding pads (stagger IO pads only)

- For the limitation of bonding wire technique , the stagger IO pads are used in order to reduce IO pad width.
- We have to add the bonding pads after APR is finished if stagger IO pads is used. But SE does not provide a built-in function for add bonding pads, CIC reaches this purpose by the way of importing DEF.
- CIC provides a perl script to calculate the bonding pad location. The full flow is described in next page

Add bonding pads flow (stagger IO pads only)



A placed and routed design in encounter

Export DEF
(In encounter)

routed.def

addbonding.pl

ioPad.list

addbonding.pl routed.def
(In unix terminal)

bondPads.cmd

bondPads.eco

source bondPads.cmd
(In encounter terminal)

finish

Output Data

Design→Save→GDS…

Design→Save->Netlist…

Design→Save->DEF

◆ Export GDS for DRC,LVS,LPE,and tape out.

◆ Export Netlist for LVS and simulation.

◆ Export DEF for reordered scan chain.

Chapter2

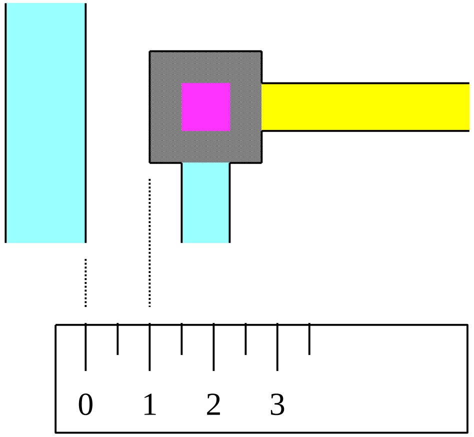
Post-Layout Verification – DRC/ERC/LVS/LPE

Post-Layout Verification Overview

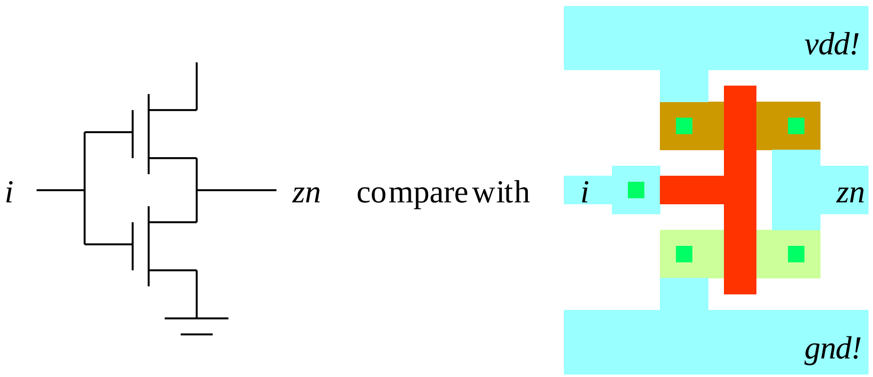
- Post-Layout Verification do the following things :
  - DRC (Design Rule Check)
  - ERC (Electrical Rule Check)
  - LVS (Layout versus Schematic)
  - LPE/PRE (Layout Parasitic Extraction / Parasitic Resistance Extraction) and Post-Layout Simulation.

Post-Layout Verification Overview cont.

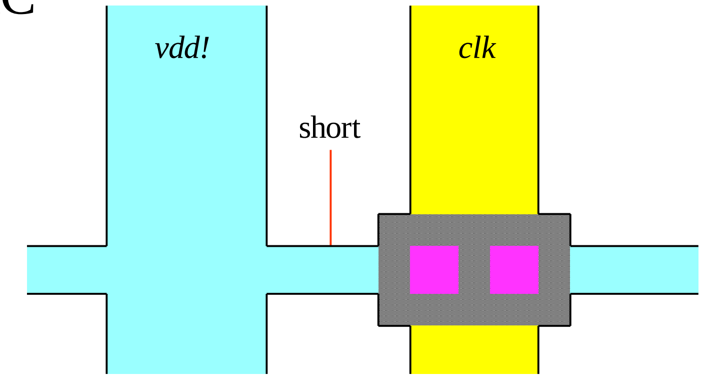
DRC



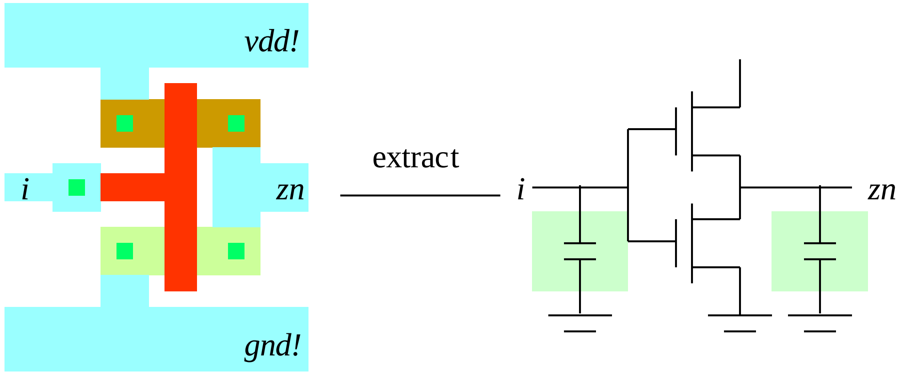
LVS



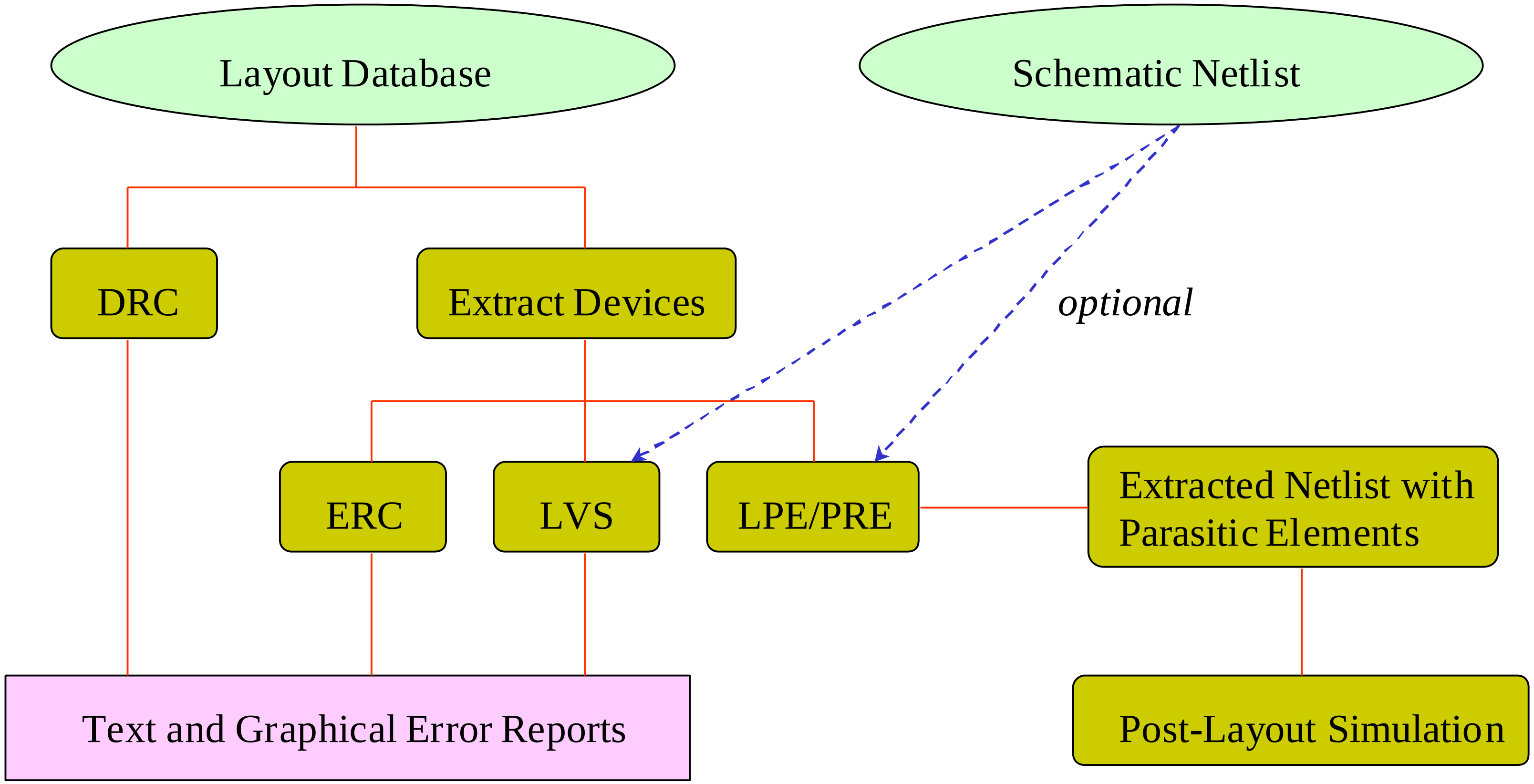
ERC



LPE/PRE



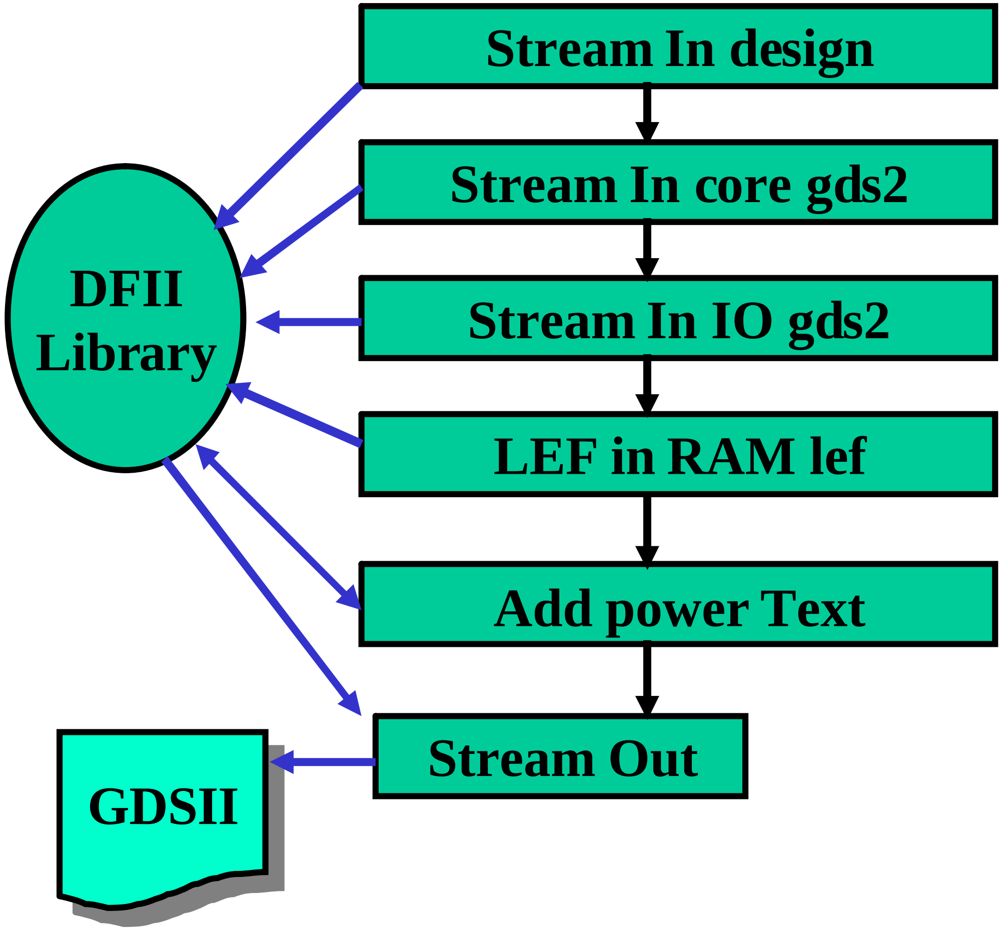
Post-Layout Verification Overview



DRC flow

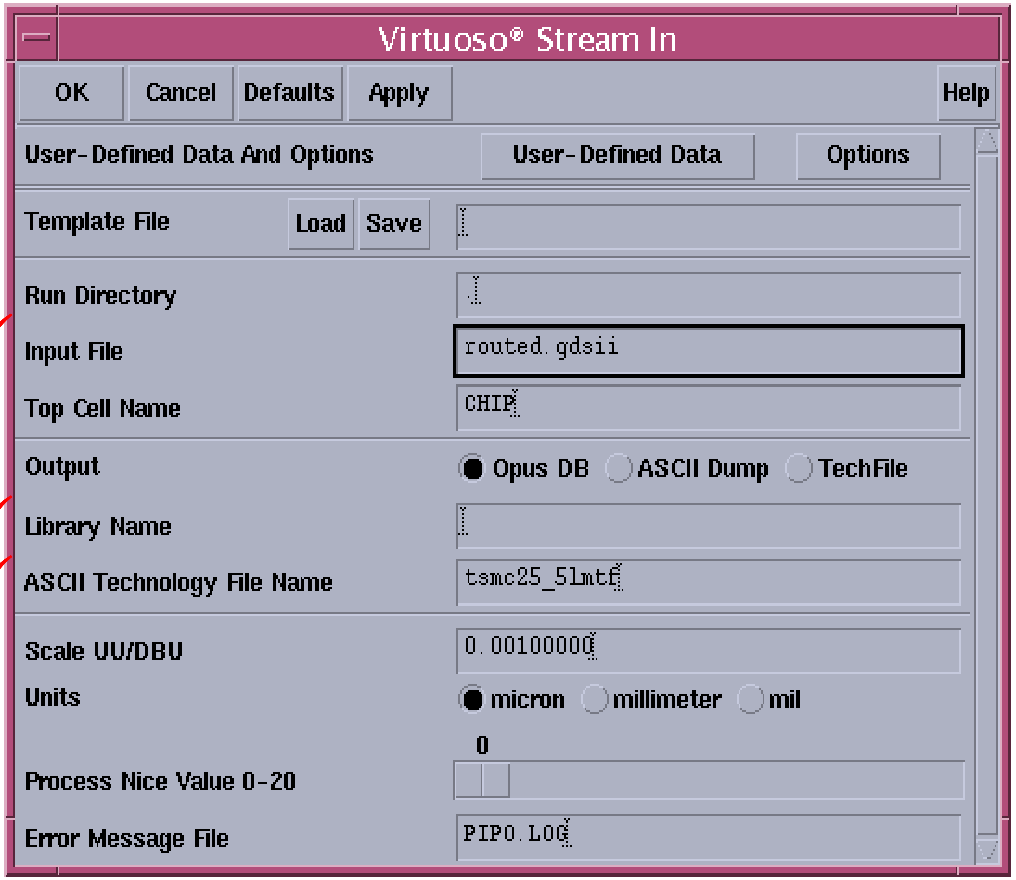
- Prepare Layout
  - stream in gds2
  - add power pad text
  - stream out gds2
- Prepare command file
- run DRC
- View DRC error (DRC summary/RVE)

Prepare Layout



Prepare Layout: Stream In GDSII

- Require:
  - technology file
  - display.drf
- *File->import->stream*



User-Defined Data And Options

Template File:

Run Directory: .

✓ Input File: routed.gdsii

Top Cell Name: CHIP

Output: ● Opus DB ○ ASCII Dump ○ TechFile

✓ Library Name:

✓ ASCII Technology File Name: tsmc25_5lmtf

Scale UU/DBU: 0.00100000

Units: ● micron ○ millimeter ○ mil

Process Nice Value 0-20: 0

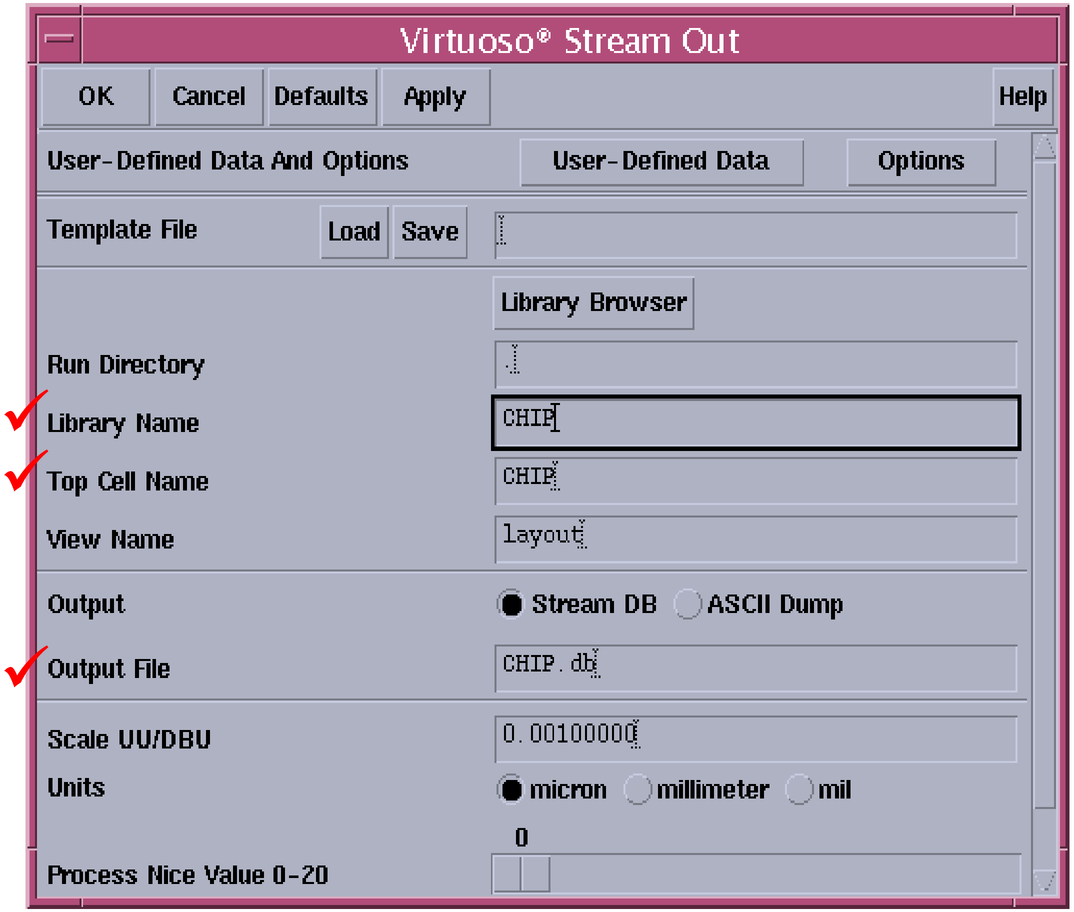
Error Message File: PIPO.LOG

Prepare Layout: Add Power Text

- Add power text for LVS and Nanosim
- For UMC18/artisan library
  - Add text DVDD for IO power pad
  - Add text DGND for IO ground pad
  - Add text VDD for core power pad
  - Add text GND for core ground pad

Prepare Layout: Stream Out GDSII

- *File->Export->stream..*



User-Defined Data And Options

Template File

Run Directory .

✓ Library Name CHIP

✓ Top Cell Name CHIP

View Name layout

Output ● Stream DB ○ ASCII Dump

✓ Output File CHIP.db

Scale UU/DBU 0.00100000

Units ● micron ○ millimeter ○ mil

Process Nice Value 0-20 0

Prepare command file

- ◆ Prepare DRC Command file:
  - ➢ **0.18** *(CBDK018_UMC_Artisan)* Calibre
    - ✓ 180nm_layers.cal
    - ✓ G-DF-IXEMODE_RCMOS18-1.8V-3.3V-1P6M-MMC-Calibre-drc-2.2-p1

Prepare Calibre Command file

◆ Edit runset file

```
LAYOUT PATH “CHIP.gds2”
LAYOUT PRIMARY “CHIP”
LAYOUT SYSTEM GDSII
…
…
…
DRC SELECT CHECK
    NW.W.1
    NW.W.2
    …
DRC UNSELECT CHECK
    NW.S.1Y
    NW.S.2Y
    …
DRC ICSTATION YES
INCLUDE “Calibre-drc-cur”
```

Submit Calibre Job

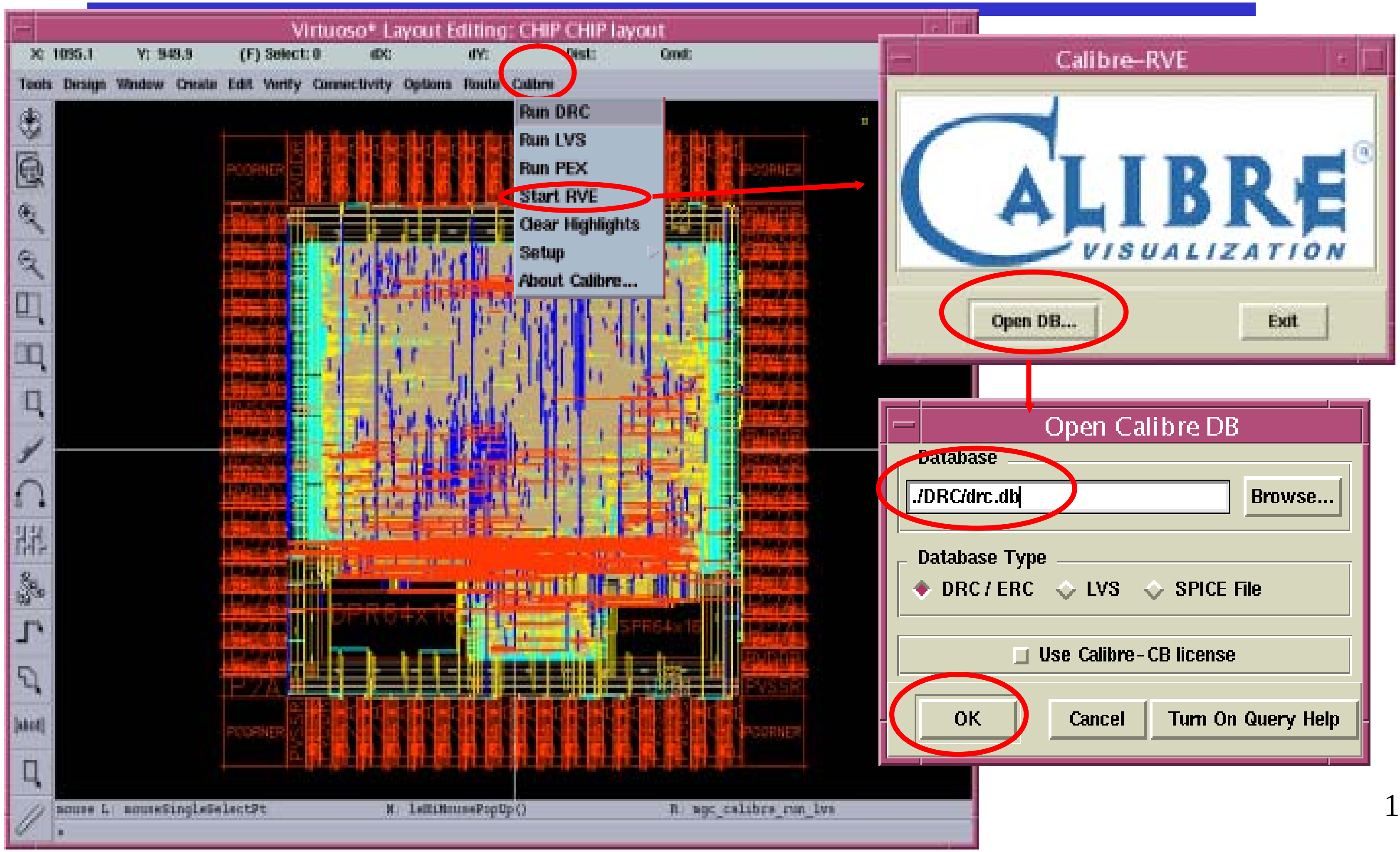
- Submit Calibre Job
  - calibre –drc umc18DRC.cal
- Result log
  - CHIP.drc.summary (ASCII result)
  - CHIP.drc.results (Graphic result)

Using Calibre RVE

◆ Add in .cdsinit

```
setSkillPath(“.  ~/  /usr/memtor/Calibre_ss/cur/shared/pkgs/icb/tools/queryskl”)
load(“calibre.skl”)
```

Using Calibre RVE



Database

./DRC/drc.db

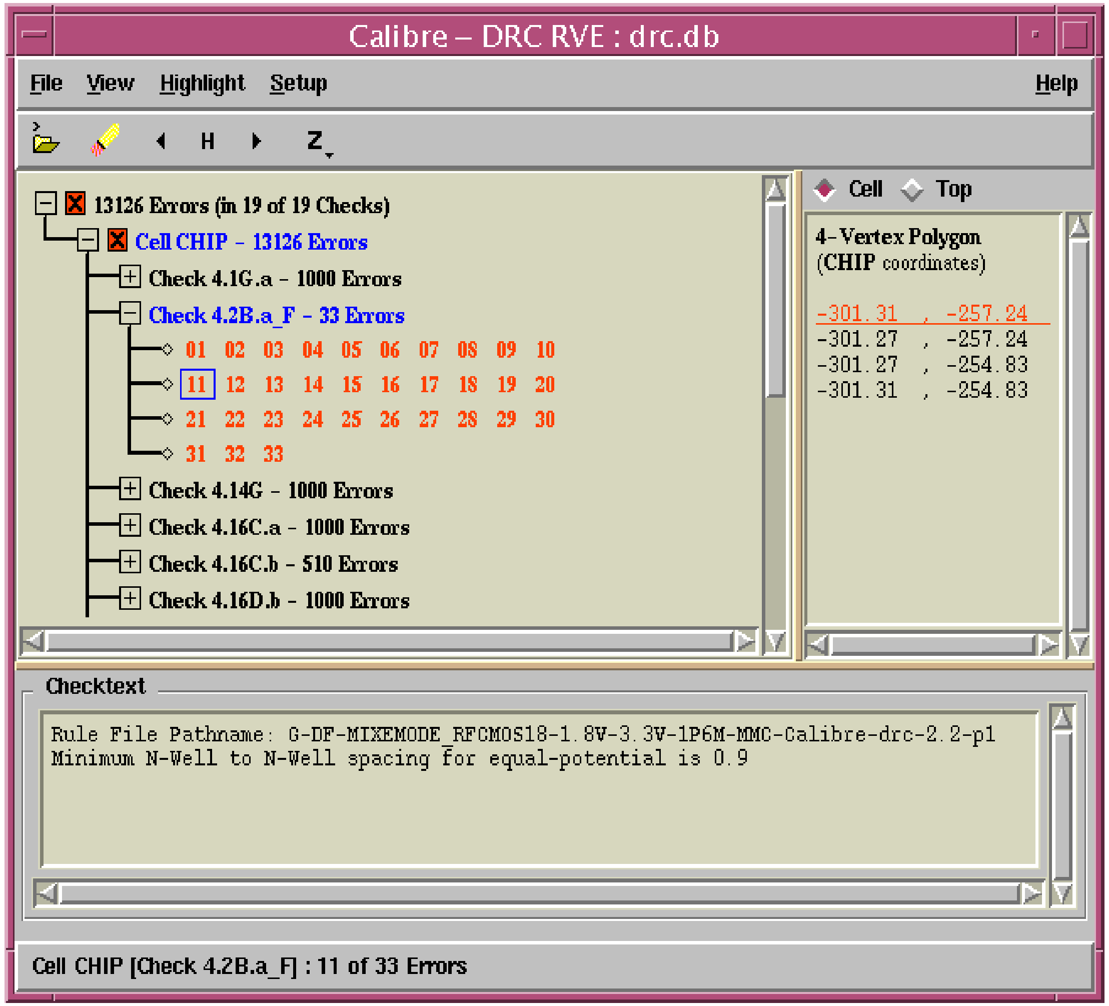
Database Type

◆ DRC / ERC ◇ LVS ◇ SPICE File

□ Use Calibre-CB license

Virtuoso® Layout Editing: CHIP CHIP layout
X: 1095.1 Y: 949.9 (F) Select: 0 dX: dY: Dist: Cmd:
Tools Design Window Create Edit Verify Connectivity Options Route Calibre
Run DRC
Run LVS
Run PEX
Start RVE
Clear Highlights
Setup
About Calibre...
mouse L: mouseSingleSelectPt M: leHiMousePopUp() R: mgc_calibre_run_lvs
Calibre-RVE
CALIBRE VISUALIZATION
Open DB...
Exit
Open Calibre DB
Browse...
OK
Cancel
Turn On Query Help

Using Calibre RVE



- ☒ 13126 Errors (in 19 of 19 Checks)
  - ☒ Cell CHIP – 13126 Errors
    - ⊞ Check 4.1G.a – 1000 Errors
    - ⊟ Check 4.2B.a_F – 33 Errors
      - 01 02 03 04 05 06 07 08 09 10
      - 11 12 13 14 15 16 17 18 19 20
      - 21 22 23 24 25 26 27 28 29 30
      - 31 32 33
    - ⊞ Check 4.14G – 1000 Errors
    - ⊞ Check 4.16C.a – 1000 Errors
    - ⊞ Check 4.16C.b – 510 Errors
    - ⊞ Check 4.16D.b – 1000 Errors

4-Vertex Polygon
(CHIP coordinates)

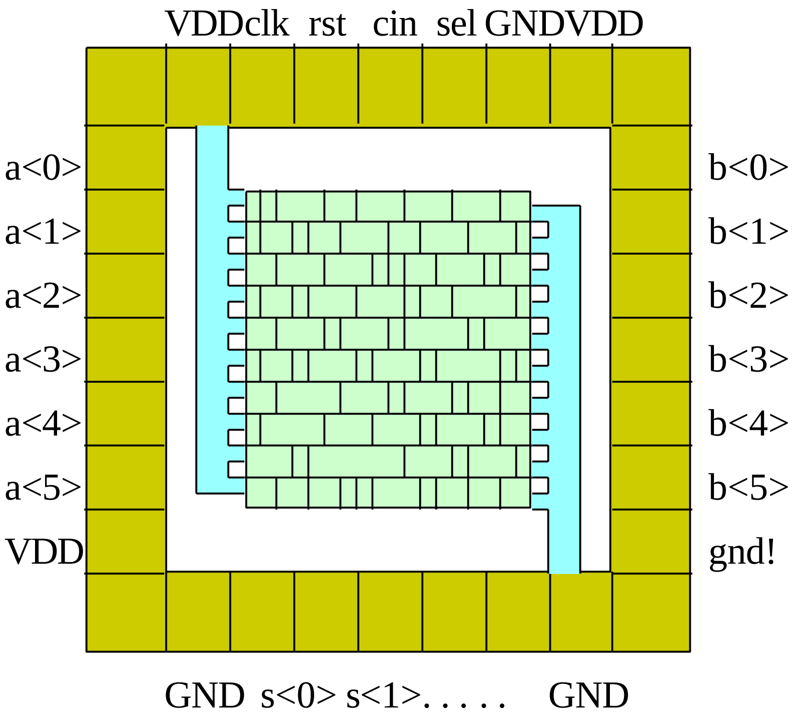
```
-301.31  ,  -257.24
-301.27  ,  -257.24
-301.27  ,  -254.83
-301.31  ,  -254.83
```

```
Rule File Pathname: G-DF-MIXEMODE_RFCMOS18-1.8V-3.3V-1P6M-MMC-Calibre-drc-2.2-p1
Minimum N-Well to N-Well spacing for equal-potential is 0.9
```

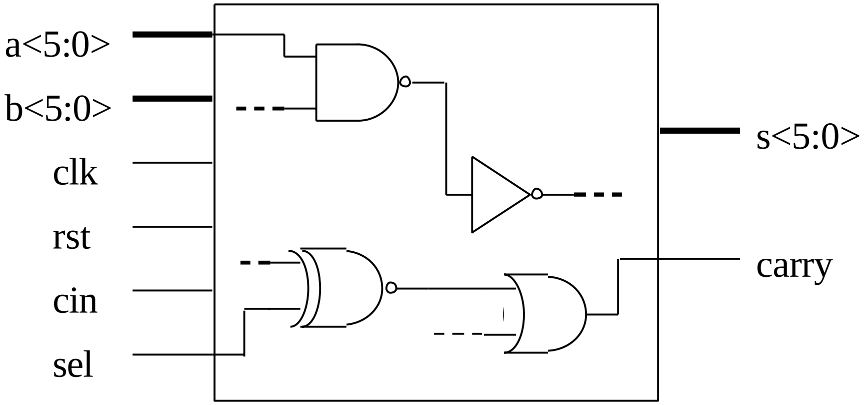
Cell CHIP [Check 4.2B.a_F] : 11 of 33 Errors

LVS Overview

Layout Data

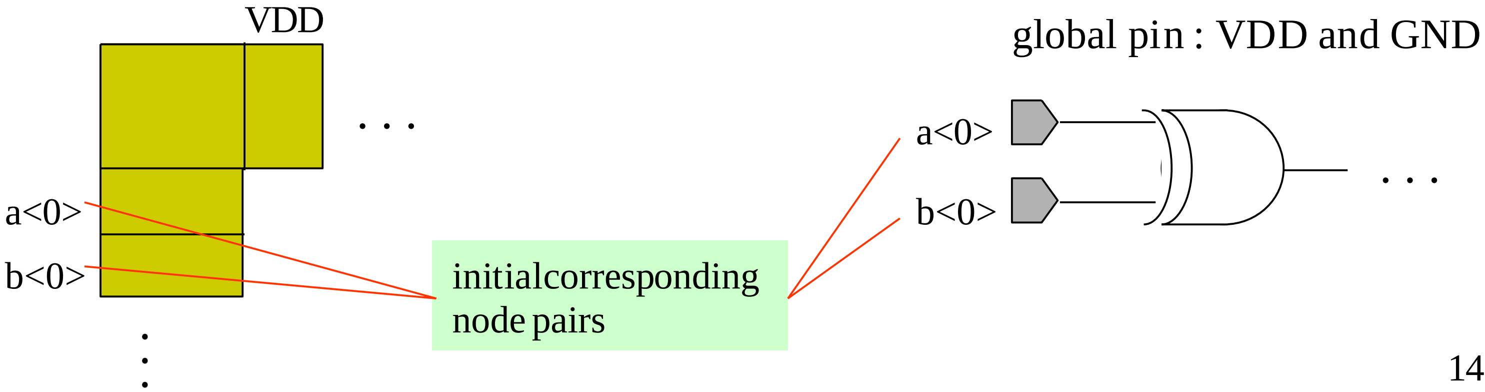


Schematic Netlist



Initial Correspondence Points

- ◆ Initial correspondence points establish a starting place for layout and schematic comparison.
- ◆ Create initial correspondence node pairs by
  - ➢ adding text strings on layout database.
  - ➢ all pins in the top of schematic netlist will be treated as an initial corresponding node if calibre finds a text string in layout which matches the node name in schematic.



VDD

global pin : VDD and GND

a<0>

b<0>

initialcorresponding node pairs

a<0>

b<0>

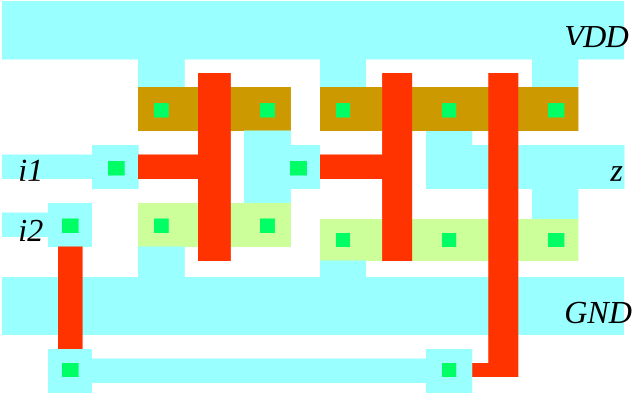
Black-Box LVS

Calibre black-box LVS

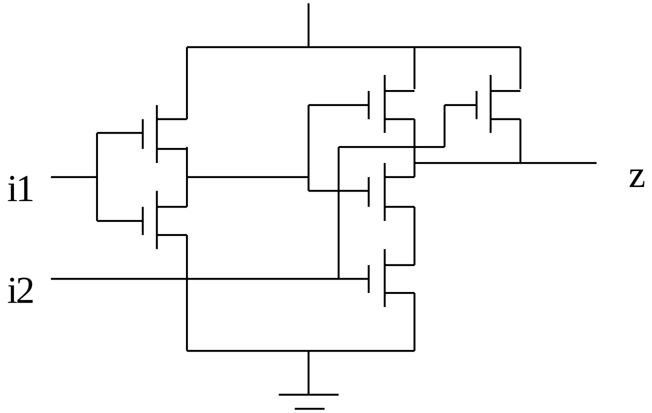
- One type of hierarchical LVS.
- Black-box LVS treats every library cell as a *black box*.
- Black-box LVS checks only the interconnections between library cells in your design, but not cell inside.
- You need not know the detail layout of every library cells.
- Reduce CPU time.

Black-Box LVS vs. Transistor-Level LVS

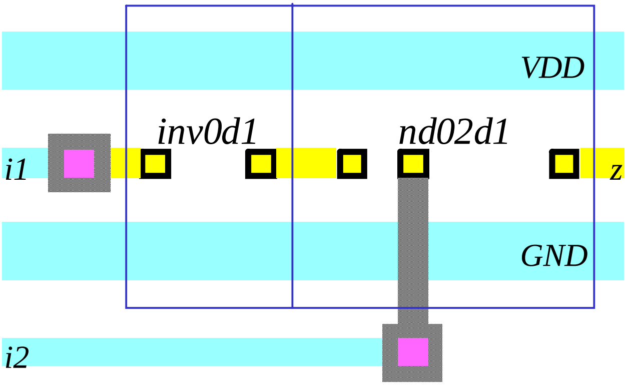
Transistor Level LVS



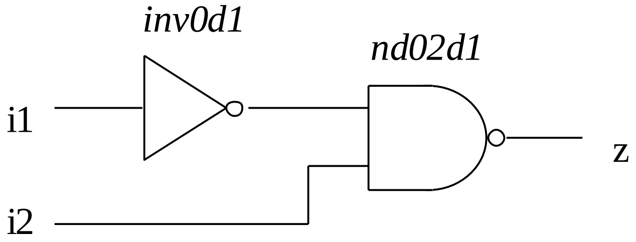
vs.



Black-Box LVS



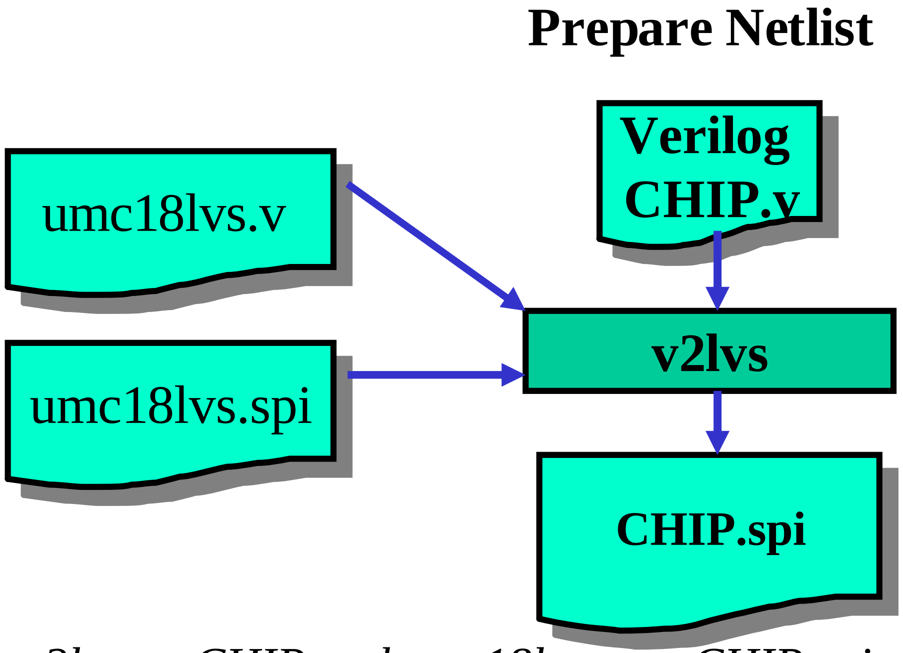
vs.



LVS flow

- Prepare Layout
  - The same as DRC Prepare Layout
- Prepare Netlist
  - v2lvs
- Prepare calibre command file
- run calibre LVS
- View LVS error (LVS summary/RVE)

Prepare Netlist for Calibre LVS



- *v2lvs –v CHIP.v –l umc18lvs.v –o CHIP.spi –s umc18lvs.spi –s1 VDD –s0 GND*

If a macro DRAM64x16 is used

- *v2lvs –v CHIP.v –l umc18lvs.v –l DRAM64x16.v –o CHIP.spi –s umc18lvs.spi –s DRAM64x16.spi –s1 VDD –s0 GND*

CIC Supported Files (0.18)

- CIC supports the following files in our cell library design kit.
  - Calibre LVS runset file

    umc18LVS.cal
  - Calibre LVS rule file

    G-DF-MIXEDMODE_RFCMOS18-1.8V_3.3V-1P6M-MMC-CALIBRE-LVS-1.2-P3.txt
  - Black-box LVS relative files
    - ✓ pseudo spice file

      umc18LVS.spi
    - ✓ **pseudo verilog** file

      umc18LVS.v

Black Box related file

- Pseudo spice file

```
.GLOBAL VDD VSS
.SUBCKT AN2D1 Z A1 A2 VDD GND
.ENDS
…
```

- Pseudo verilog file

```
module AN2D1 (Z, A1, A2);
    output Z;
    input A1;
    input A2;
endmodule
…
```

Prepare command file for Calibre LVS

◆ Edit Calibre LVS runset

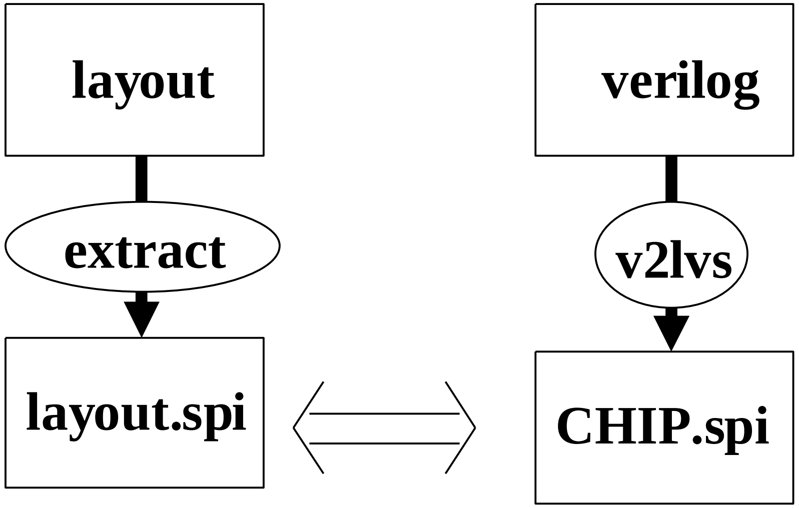
```
LAYOUT PATH “CHIP.calibre.gds”
LAYOUT PIMARY “CHIP”
LAYOUT SYSTEM GDSII
SOURCE PATH “CHIP.spi”
SOURCE PRIMARY “CHIP”
…
…
INCLUDE “/calibre/LVS/Calibre-lvs-cur”
```

◆ Edit Calibre LVS rule file

```
…
…
LVS BOX PVSSC
LVS BOX PVSSR
LVS BOX DRAM64x4s
```

Submit Calibre LVS

- *calibre –lvs –spice layout.spi –hier –auto umc18LVS.cal > lvs.log*



Check Calibre LVS Summary

- ◆ OVERALL COMPAISON RESULTS
- ◆ CELL SUMMARY
- ◆ INFORMATION AND WARNINGS
- ◆ Initial Correspondence Points

Check Calibre LVS Summary

OVERALL COMPAISON RESULTS

OVERALL COMPARISON RESULTS

```
               #       ###################       _   _
              #        #                 #       *   *
         #   #         #     CORRECT     #         |
          # #          #                 #       \___/
           #           ###################
```

Check Calibre LVS Summary

CELL SUMMARY

```
******************************************************
                    CELL  SUMMARY
******************************************************

      Result        Layout                  Source
    -----------   -----------           --------------
     CORRECT        CHIP                     CHIP
```

Check Calibre LVS Summary

INFORMATION AND WARNINGS

```
*******************************************************************
                INFORMATION AND WARNINGS
*******************************************************************

              Matched    Matched   Unmatched   Unmatched   Component
               Layout     Source      Layout      Source        Type
             -------    -------   ---------   ---------   ---------
Nets:          11525      11525           0           0

Instances:         1          1           0           0   ADDFHX1
                  54         54           0           0   ADDFHX4
                  79         79           0           0   ADDFX2
                 542        542           0           0   AND2X1
              ……         ……          ..          ..   …………
                   8          8           0           0   XOR3X2
             -------    -------   ---------   ---------   ---------
Total Inst:    10682      10682           0           0
```

Check Calibre LVS Summary

Initial Correspondence Points

o Initial Correspondence Points:

Nets: DVDD VDD DGND GND I_X[2] I_X[3] I_X[4] I_X[5] I_X[6] I_X[7] I_X[8] I_X[9] I_X[10] I_X[11] O_SCAN_OUT O_Z[0] O_Z[1] O_Z[2] O_Z[3] I_HALT I_RESET_ I_DoDCT I_RamBistE I_CLK I_SCAN_IN I_SCAN_EN I_X[0] O_Z[4] I_X[1] O_Z[5] O_Z[6] O_Z[7] O_Z[8] O_Z[9] O_Z[10] O_Z[11]

Check Calibre LVS Log

- TEXT OBJECT FOR CONNECTIVITY EXTRACTION
- PORTS
- Extraction Errors and Warnings for cell “CHIP”

Check Calibre LVS Log

TEXT OBJECT FOR CONNECTIVITY EXTRACTION

```
-----------------------------------------------------------------------------
    TEXT OBJECTS FOR CONNECTIVITY EXTRACTION
-----------------------------------------------------------------------------
O_Z[0] (523.447,31.68) 105 CHIP          O_Z[1] (598.068,31.68) 105 CHIP
O_Z[2] (821.931,31.68) 105 CHIP          O_Z[3] (896.553,31.68) 105 CHIP
O_Z[4] (971.175,31.68) 105 CHIP          O_Z[5] (1164.455,372.964) 105 CHIP
O_Z[6] (1164.455,446.966) 105 CHIP     O_Z[7] (1164.455,520.968) 105 CHIP
O_Z[8] (1164.455,594.97) 105 CHIP       O_Z[9] (1164.455,668.972) 105 CHIP
O_Z[10] (1164.455,742.974) 105 CHIP    O_Z[11] (1164.455,816.976) 105 CHIP
……
……
```

Check Calibre LVS Log

PORTS

```
-------------------------------------------------------------------------------
                              PORTS
-------------------------------------------------------------------------------
O_Z[0] (523.447,31.68) 105 CHIP        O_Z[1] (598.068,31.68) 105 CHIP
O_Z[2] (821.931,31.68) 105 CHIP        O_Z[3] (896.553,31.68) 105 CHIP
O_Z[4] (971.175,31.68) 105 CHIP        O_Z[5] (1164.455,372.964) 105 CHIP
……
……
```

Check Calibre LVS Log

Extraction Errors and Warnings for cell “CHIP”

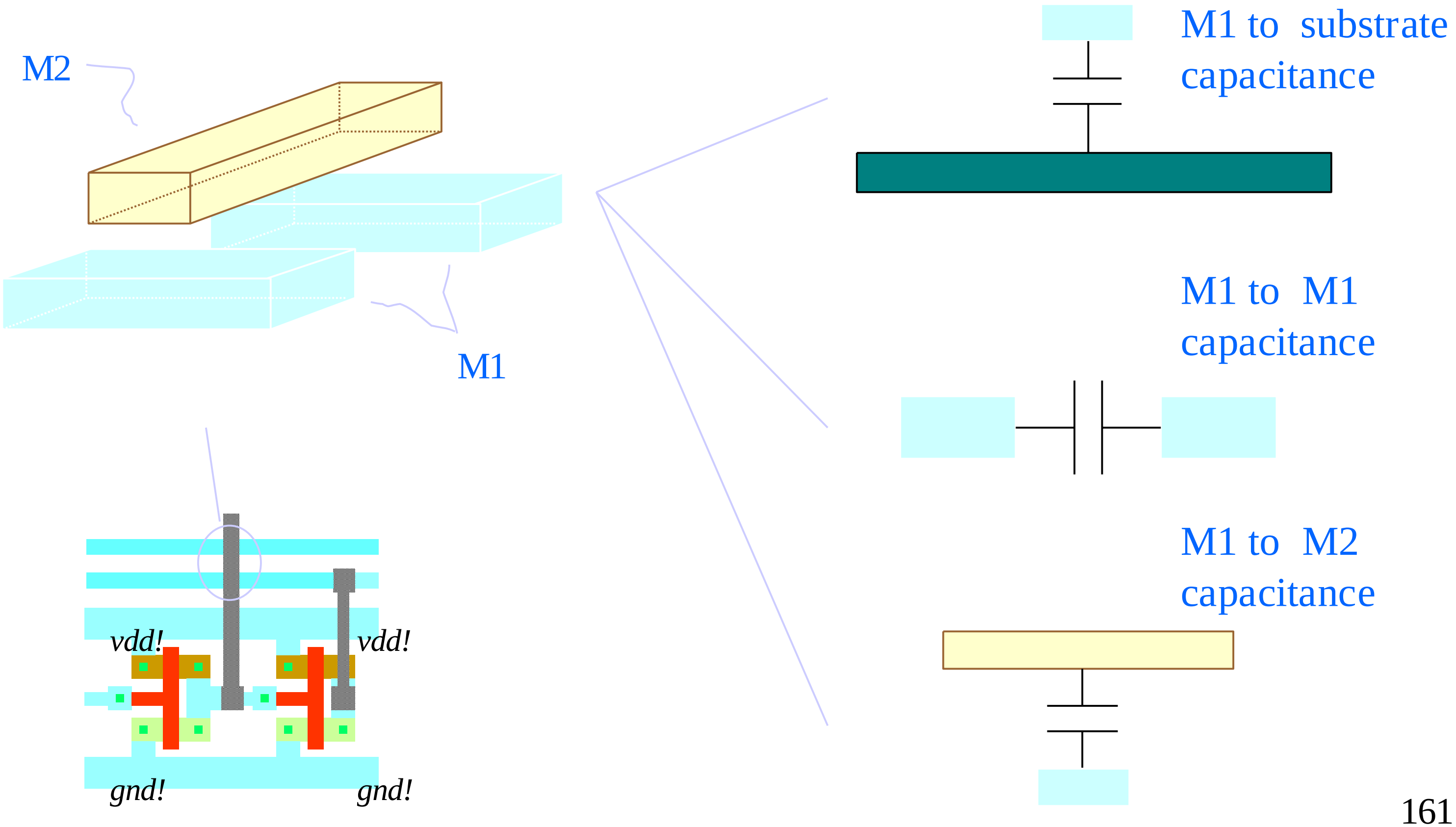
```
Extraction Errors and Warnings for cell "CHIP"
----------------------------------------------

WARNING:  Short circuit - Different names on one net:
          Net Id:  18
           (1)  name  "GND"  at location  (330.301,216.95)  on layer  102 "M2_TEXT"
           (2)  name  "GND"  at location  (673.2,29.1)  on layer  101 "M1_TEXT"
           (3)  name  "VDD"  at location  (748.1,31.5)  on layer  101 "M1_TEXT"
           (4)  name  "VDD"  at location  (208.93,274.56)  on layer  101 "M1_TEXT"
           The name "VDD" was assigned to the net.
```

Chapter3

Post-Layout Timing Analysis -- Nanosim

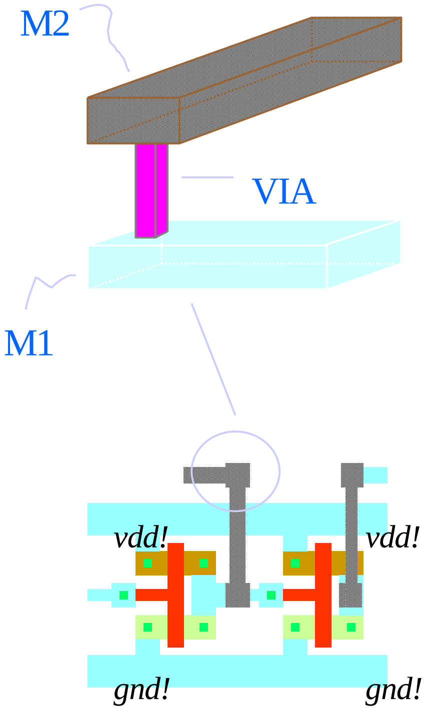
What Introduce After Place&Route?

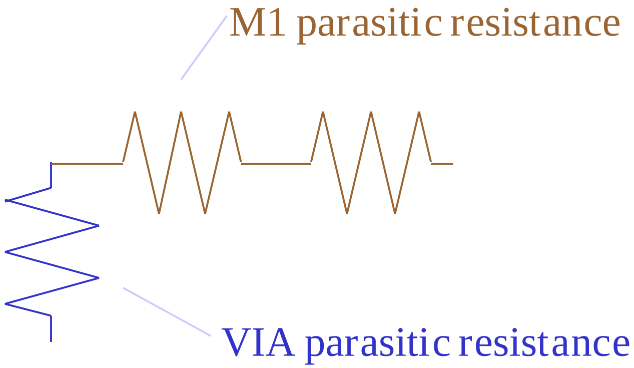
◆ Interconnection wire's parasitic capacitance.



What Introduce After Place&Route?

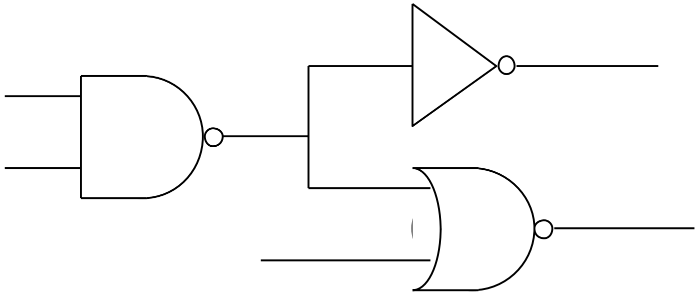
- Interconnection wires' parasitic resistance.

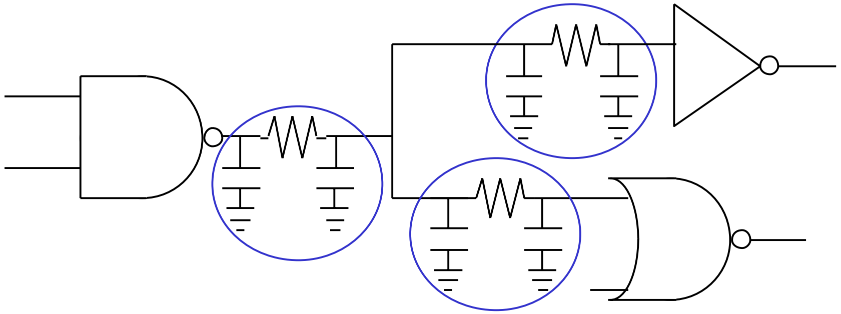




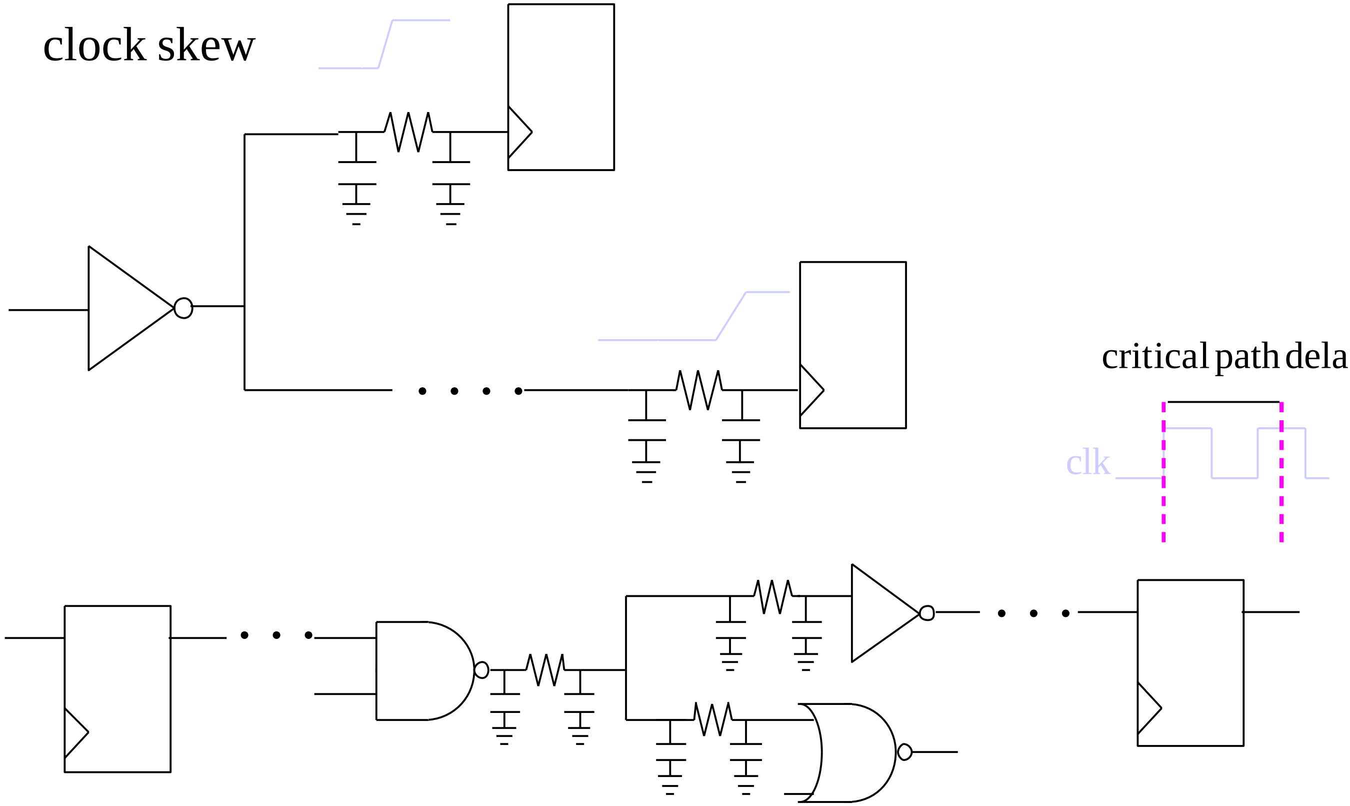
Pre-Layout And Post-Layout Design

- ◆ A pre-layout design (before P&R) and a post-layout design (after P&R)

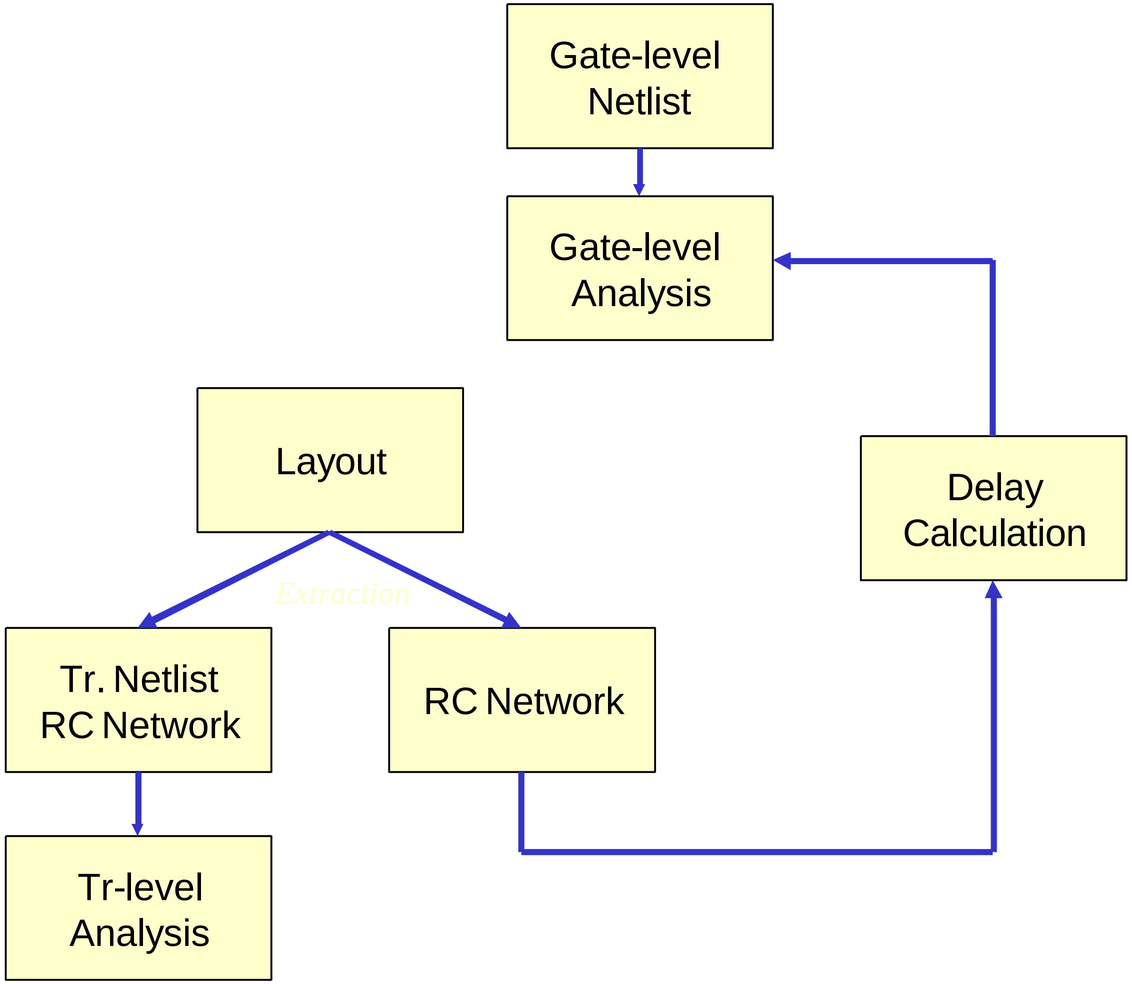




Why Post-Layout Simulation?



Post-layout Timing Analysis Flow



Gate-level Netlist

Gate-level Analysis

Layout

Extraction

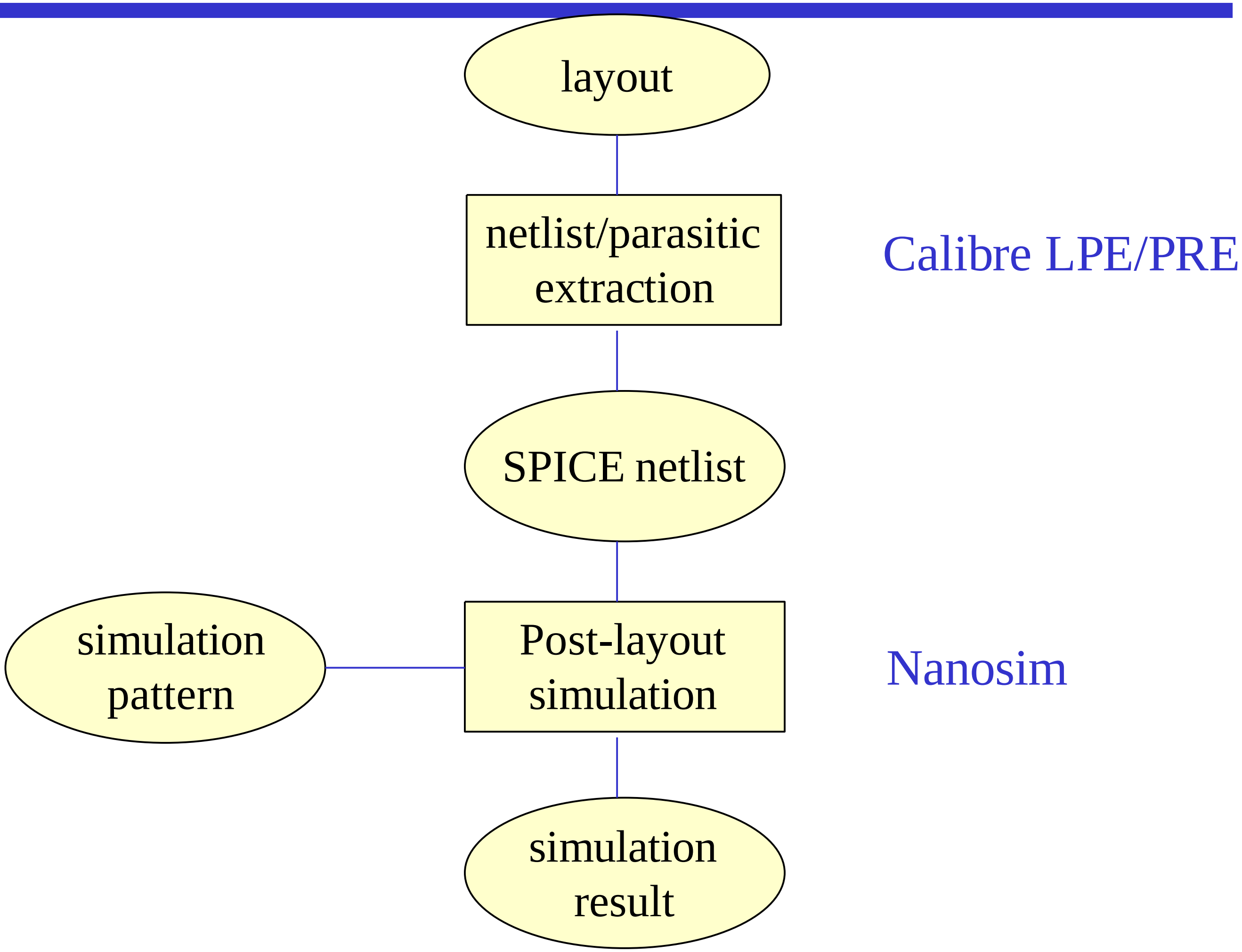
Tr. Netlist RC Network

RC Network

Delay Calculation

Tr-level Analysis

Transistor-level Post-layout Simulation



What is Nanosim

- *Nanosim* is a *transistor- level timing* simulation tool for digital and mixed signal CMOS and BiCMOS designs.
- *Nanosim* handles voltage simulation and timing check.
- Simulation is event driven, targeting between SPICE (*circuit simulator*) and Verilog (*logic simulator*).

Prepare for Post-Layout Simulation

- Apply for a CIC account
  - http://www.cic.org.tw ⇒ 工作站帳號申請.
  - fill in your personal data and your request.
- Install *identd* program
  - this program is used to identify yourself when you log into CIC's account from remote machine.
- Put your DB file to CIC's account

Replace Layout / LPE

- **Qentry**

```
-M  {LPE}
-tech  {UMC18 | TSMC18 | TSMC25 | TSMC35}
-f  GDSII
-T  Top_cell_name
-s  Ram_spce_filename
-t {ra1sd | ra1sh | ra2sd | ra2sh | rf2sh |
    t18ra1sh | t18ra2sh | t18rf1sh | t18rf2sh | t18rodsh|
    18ra1sh_1 | 18ra1sh_2 | 18ra2sh}
-c  {UMC18 | TSMC18 | TSMC25 | TSMC35}
-i  {UMC18 | TSMC18 | TSMC25 | TSMC35}
-o  Netlist_file_name
```

- Example:
  - ➢

```
Qentry -M LPE -tech UMC18 -f CHIP.gds -T CHIP
 -s RAM1.spec -t 18ra2sh -s RAM2.spec  -t 18ra1sh_1
 -s RAM3.spec -t 18ra1sh_2 -c UMC18 -i UMC18 -o CHIP.netlist
```

- Use **`Qstat`** to check the status of your job.
- The result is stored in **“`result_#`”** directory.

Replace/LPE

- INPUT
  - gds2
  - ram spec
- OUTPUT
  - output netlist
  - TOP_CELL.NAME
  - nodename
  - spice.header
  - nanosim.run
  - log files for strem in, stream out, lpe

Running Nanosim

- **Qentry**

  –M {NANOSIM}
  ~~–n {CHIP.io}~~
  –nspice CHIP.netlist spice.header
  –nvec CHIP.vec
  –m Top_cell_name
  –c {CHIP.cfg}
  –z {CHIP.tech.z}
  –o Output_file_name
  –out fsdb
  –t Total_simulation_time

- Example:
  - **Qentry –M NANOSIM –nspice CHIP.netlist spice.header –nvec CHIP.vec –m CHIP –c CHIP.cfg –z CHIP.tech.z –o UMC18 –t 100**
- Use Qstat to check the status of your job.
- The result is stored in “result_#” directory.

Spice Header File

- Spice Header File → Modify PVT
  - `.lib 'l18u18v.012' L18U_BJD`
  - `.lib 'l18u18v.012' L18U18V_TT`
  - `.lib 'l18u33v_g2.011' l18u33v_tt`
  - `*epic tech="voltage 3.3“`
  - `*epic tech="temperature 100"`

Generate Nanosim Simulation Pattern

◆ Input simulation pattern --- *vec* format

```
type vec
signal   CLOCK,START,IN[7:0]
;    time       clock   start     in<7:0>
radix               1       1          44
io                  i       i          ii
high 3.3
low 0.0
       25           0       0          xx
       50           1       0          xx
       75           0       0          xx
. . . . .
```

Generate Nanosim Simulation Pattern

- ◆ Input simulation pattern --- *nsvt* format

```
type nsvt
signal   CLOCK,START,IN[7:0]
;              clock   start     in[7:0]
radix              1       1          44
io                 i       i          ii
period 25
high 3.3
low 0.0
                   0       0          xx
                   1       0          xx
                   0       0          xx
. . . . .
```

Generate Nanosim Simulation Pattern

- ◆ You can generate Nanosim simulation pattern from Verilog-XL stimulus.

Verilog *test bench* file

```
integer outf;
initial begin
  outf = $fopen("input.dat");
  . . . . .
  $fclose(outf);
  $finish;
  end

always @(sys_clock or start or in)
     $fdisplay(outf,"%t %b %b %h",$time,sys_clock,start,in);
. . . . .
```

Nanosim Configuration File

Example *Nanosim_configuration* file

```
bus_notation [ : ]
print_node_logic ADRS[0]
print_node_logic CLK
print_node_logic DATA[0]
. . . . . .
report_node_power VDD
set_node_gnd DGND
set_node_gnd GND
set_node_v DVDD 3.3
set_node_v VDD 1.8
```

nodename file

```
ADRS[0]
ADRS[1]
. . . . . .
CLK
DATA[0]
. . . . . .
```

View Simulation Result --- nWave

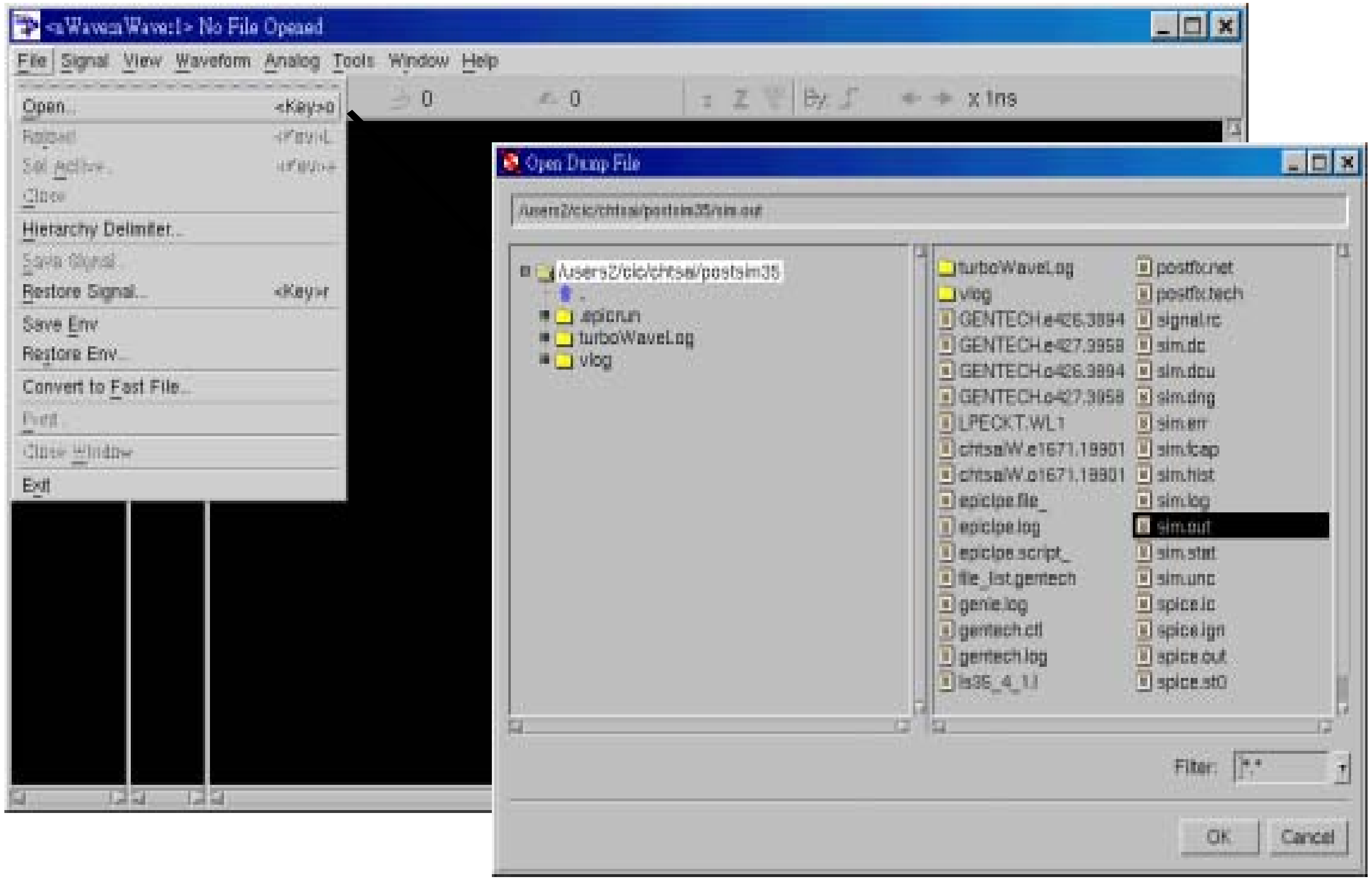
- *NOVAS nWave*
  - a waveform viewer which support Timemill output waveform format.
- Environment setup

```
unix% source /usr/debussy/CIC/debussy.csh
```

- Starting *nWave*

```
unix% nWave &
```

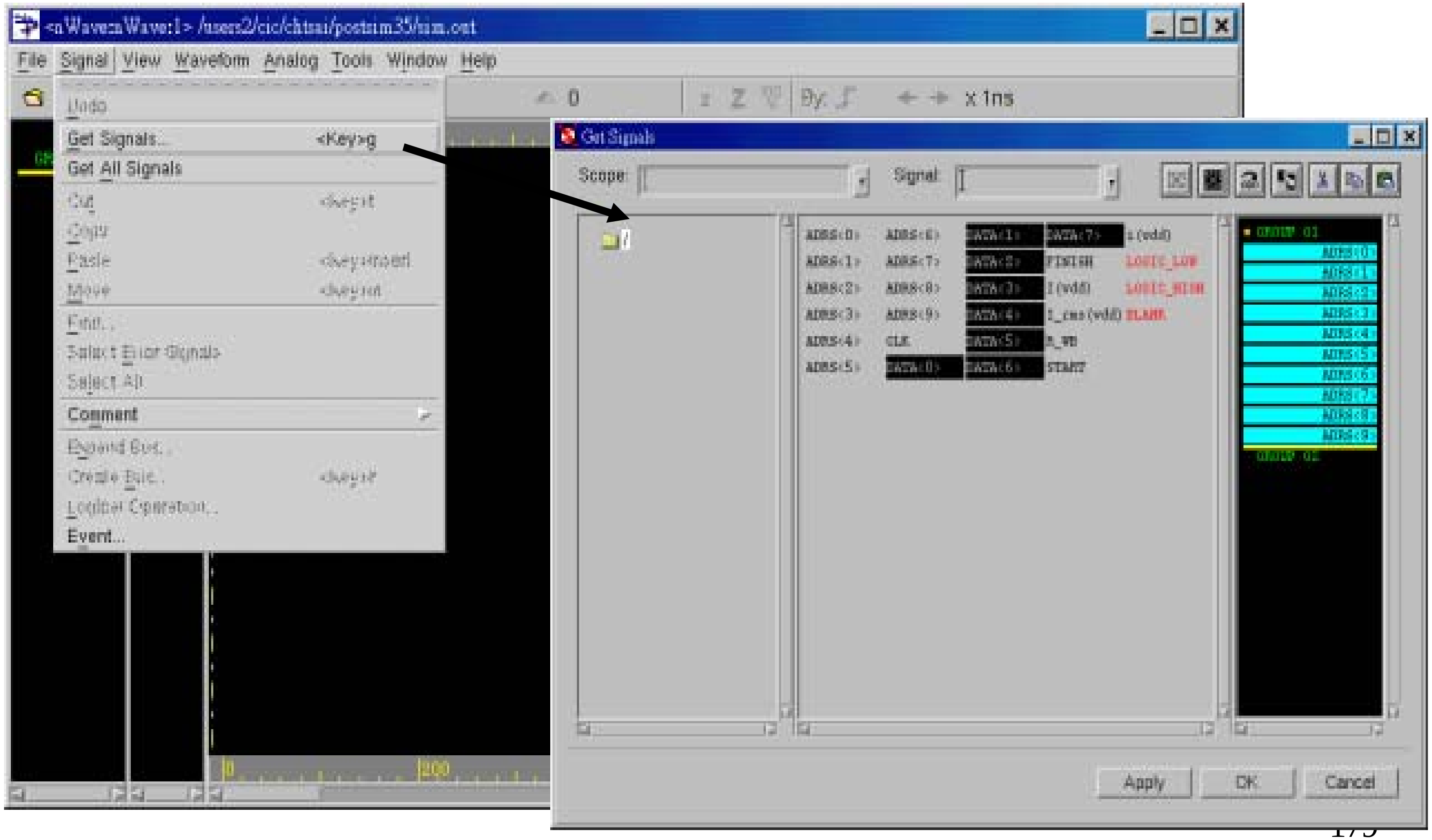
Load Simulation Result --- nWave



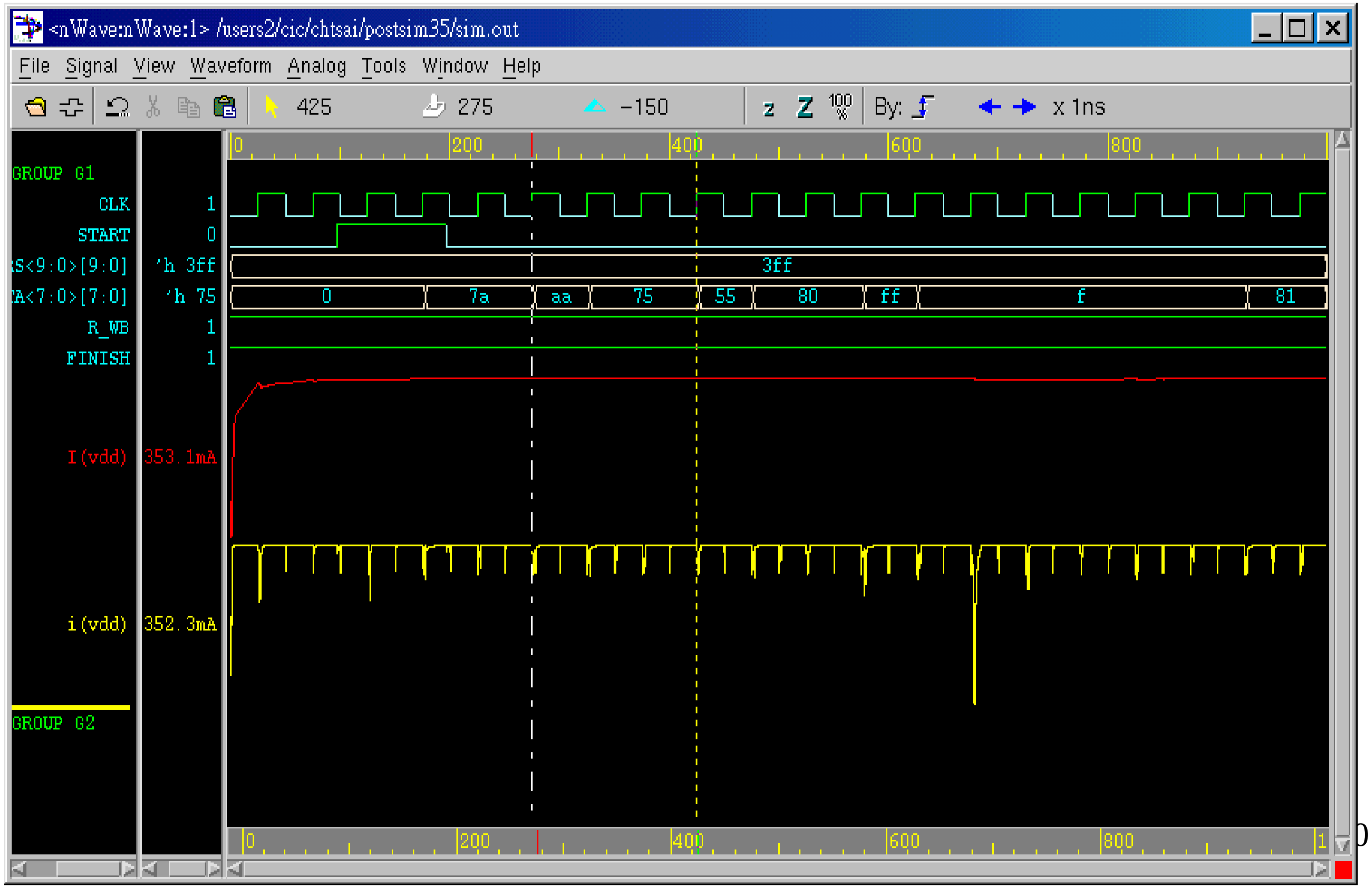
/users2/cic/chtsai/postsim35/sim.out

Filter: *.*

Select Signals --- nWave



Check Simulation Result --- nWave



Power Analysis Result

- The power analysis result is stored in Nanosim simulation log (*xxx.log*) file

```
. . . . . .
Current information calculated over the intervals:

     0.00000e+00 -  1.00010e+03 ns

Node: VDD
    Average current      : -3.53355e+05 uA
    RMS current          :  3.53388e+05 uA

    Current peak #1      : -4.54061e+05 uA  at    6.78400e+02 ns
    Current peak #2      : -4.34973e+05 uA  at    4.00000e-01 ns
    Current peak #3      : -3.88048e+05 uA  at    2.59000e+01 ns
    Current peak #4      : -3.87280e+05 uA  at    1.27500e+02 ns
    Current peak #5      : -3.84302e+05 uA  at    5.77800e+02 ns
. . . . . .
```