Ministry of Electronics and Information Technology

Sponsored Workshop on

OpenROAD for Low-Cost ASIC Design and Rapid Innovation

Organized by

Novel Integrated Electronics (NINE) Labs

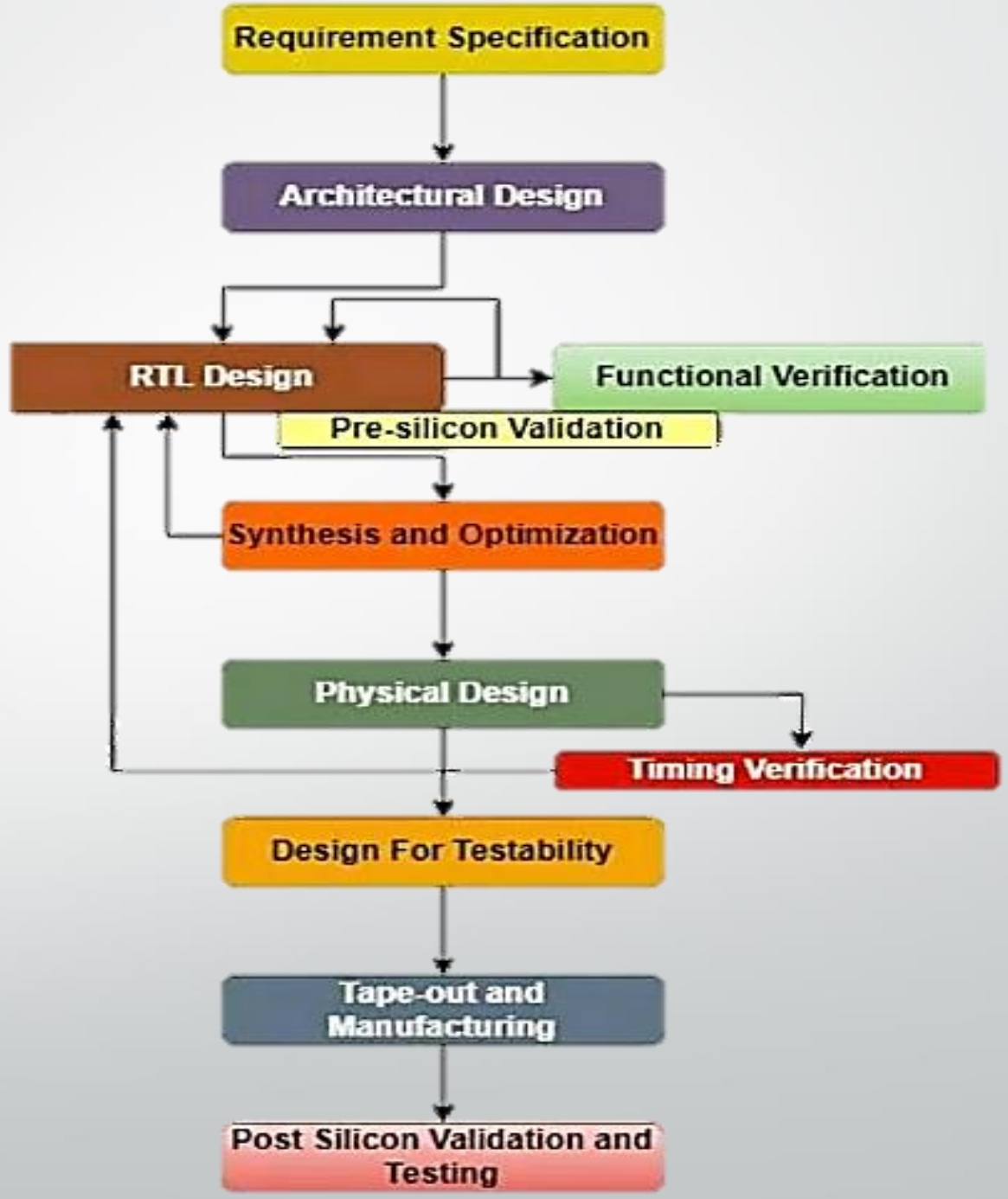
(Under MeitY, Govt. of India)

Indian Institute of Technology Guwahati

Guwahati, Assam, Pin 781039

Dates : 17 Jan – 24 Jan 2024

ASIC Design Flow



Synthesis

Synthesis

Synthesis is the process that generates a gate-level netlist for an IC design that has been defined using a Hardware Description Language (HDL)

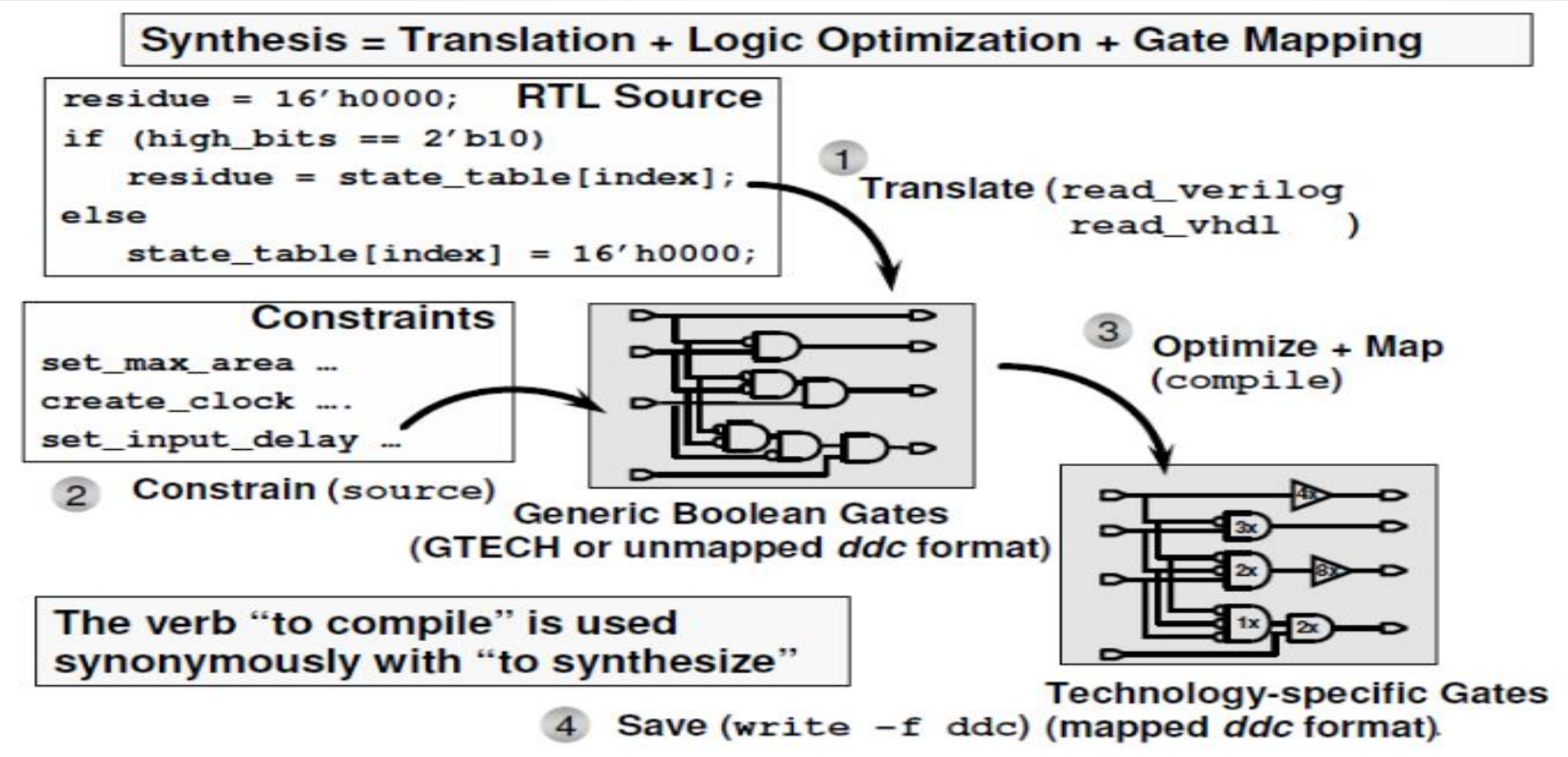
It has 3 stages namely

1 . Translate

2. Mapping

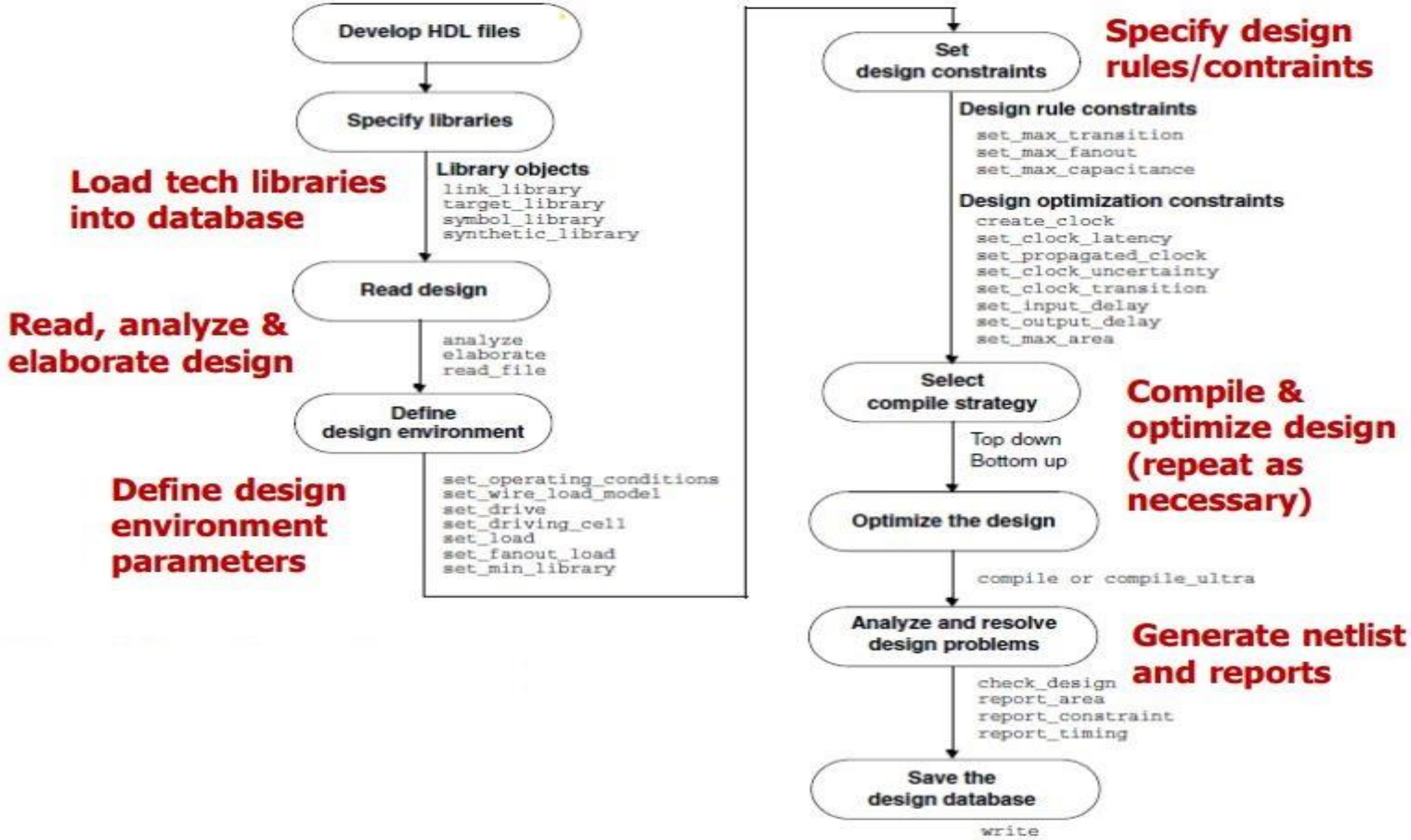
3. Optimize

Synthesis Transformations



Basic Synthesis Flow

- Develop HDL Files
- Specify Libraries
- Read Design
- Define Design Environment
- Set Design Constraints
- Optimize the Design
- Analyze and Resolve the Design Problems


Develop HDL files
Specify libraries
Load tech libraries into database
Library objects
link_library
target_library
symbol_library
synthetic_library
Read design
Read, analyze & elaborate design
analyze
elaborate
read_file
Define design environment
Define design environment parameters
set_operating_conditions
set_wire_load_model
set_drive
set_driving_cell
set_load
set_fanout_load
set_min_library
Set design constraints
Specify design rules/contraints
Design rule constraints
set_max_transition
set_max_fanout
set_max_capacitance
Design optimization constraints
create_clock
set_clock_latency
set_propagated_clock
set_clock_uncertainty
set_clock_transition
set_input_delay
set_output_delay
set_max_area
Select compile strategy
Compile & optimize design (repeat as necessary)
Top down
Bottom up
Optimize the design
compile or compile_ultra
Analyze and resolve design problems
Generate netlist and reports
check_design
report_area
report_constraint
report_timing
Save the design database
write

Input files required for Synthesis

- **Tech related:**
  - .tf- technology related information.
  - **.lib-timing info of standard cell & macros**
- **Design related:**
  - **RTL code. (.v or .sv or .vhdl)**
  - **SDC- Timing constraints.**
  - UPF- power intent of the design
  - Scan config- Scan related info like scan chain length, scan IO, which flops are to be considered in the scan chains.
- **For Physical aware:**
  - RC co-efficient file (tluplus).
  - LEF/FRAM- abstract view of the cell.
  - Floorplan DEF- locations of IO ports and macros.

Timing library file (.Lib)

```
cell(AND2X1) {
  /* gs005 */
  pg_pin(VDD) {
    voltage_name : V;
    pg_type : primary_power;
  }
  pg_pin(VSS) {
    voltage_name : VSS;
    pg_type : primary_ground;
  }
    scaling_factors : AND2X1_factors ;
    area : 7.445 ;
    leakage_power() {
   /* gs006 */
   related_pg_pin : VDD;
        when : "(IN1*!IN2')";
        value : 162490.000;
    }
    leakage_power() {
   /* gs006 */
   related_pg_pin : VDD;
        when : "(IN1*IN2')";
        value : 171100.000;
    }
    leakage_power() {
   /* gs006 */
   related_pg_pin : VDD;
        when : "(!IN1*!IN2)";
        value : 178960.000;
    }
    leakage_power() {
   /* gs006 */
   related_pg_pin : VDD;
        when : "(IN1*IN2)";
        value : 149790.000;
    }
```

- Liberty format , it has timing/logical libraries
- Contains timing and power info

Rise time, fall time, min trans, max trans

Cell info:
Cell name, pg pin name, area of the cell, leakage power.

Pin info:
Pin name, pin direction, internal power, capacitance, fanout

Physical library files (.lef)

Physical library contains the abstract view of layout for standard cells and macros, Lef file basically contains the

- Size of the cell(height and width)
- Symmetry of cell
- Pin name , direction , shape , layer
- Pin location

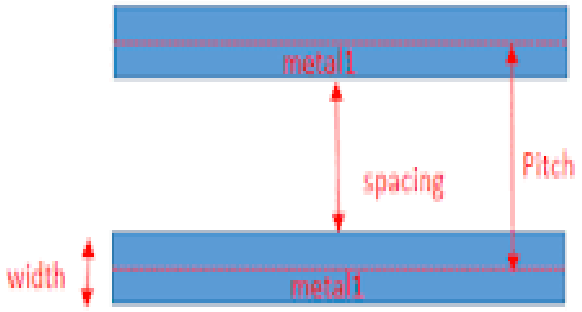
LEF file is provided by the standard cell vendor.

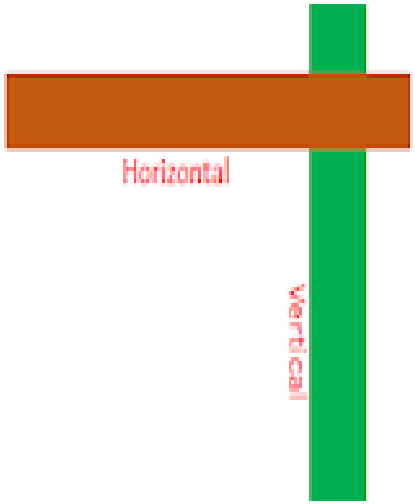
```
PIN B
  DIRECTION INPUT ;
  USE SIGNAL ;
  PORT
    LAYER metal1 ;
      RECT 0.4175 0.9175 0.6025 1.0525 ;
  END
```

Technology file

```
LAYER metal1
   TYPE ROUTING ;
   DIRECTION HORIZONTAL ;
   PITCH 0.19 ;
   WIDTH 0.065 ;
   SPACING 0.065 ;
   RESISTANCE RPERSQ 0.38 ;
END metal1

LAYER via1
   TYPE CUT ;
   SPACING 0.075 ;
   PROPERTY contactResistance 5.69 ;
END via1
```



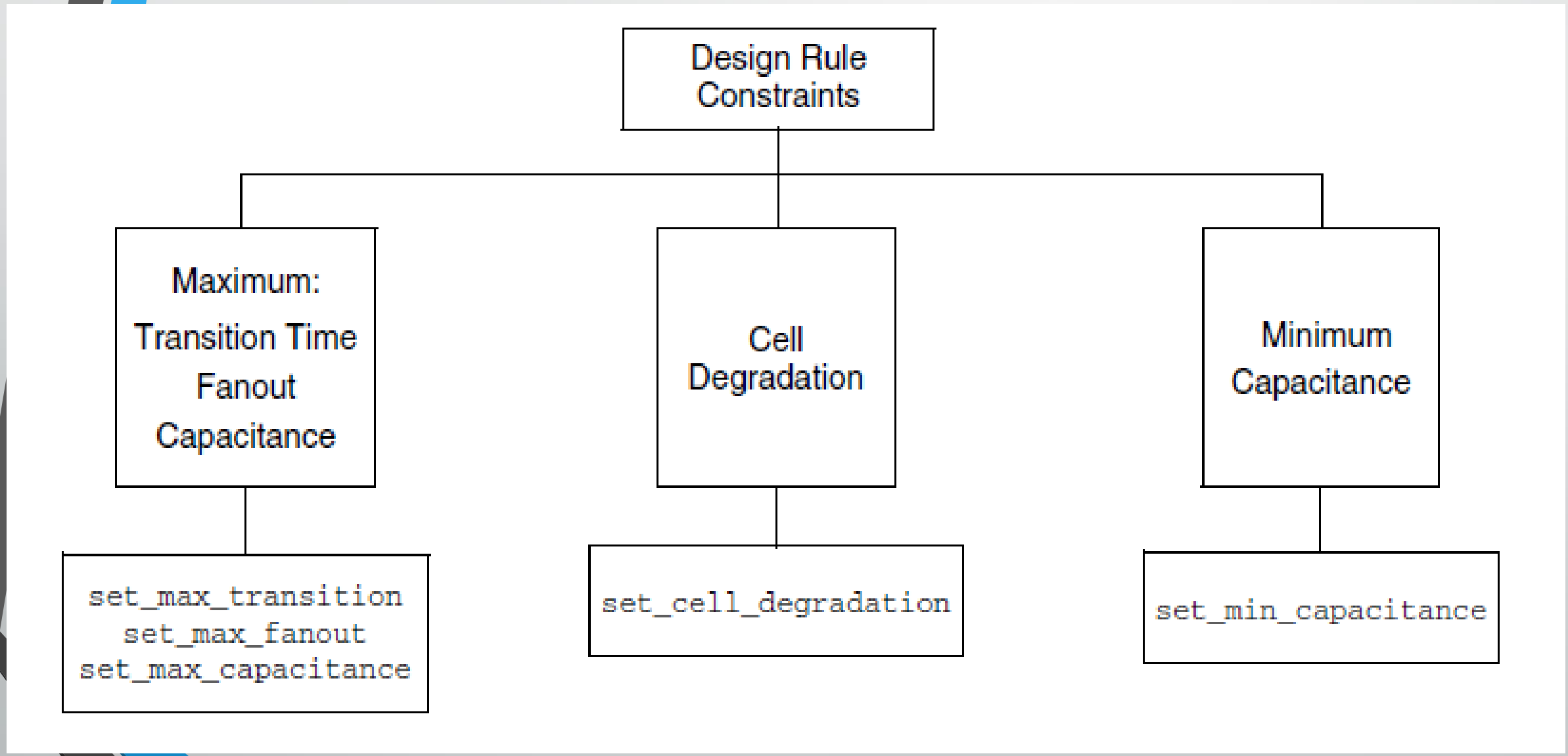


The technology file contains detailed information about all the metal layers, vias and their design rules. This file is in ASCII format and basically contains the following information.

- Layer name(poly/contact/metal1/via2)
- Types and directions of the metal
- Pitch
- Site information
- Width
- Spacing
- Resistance

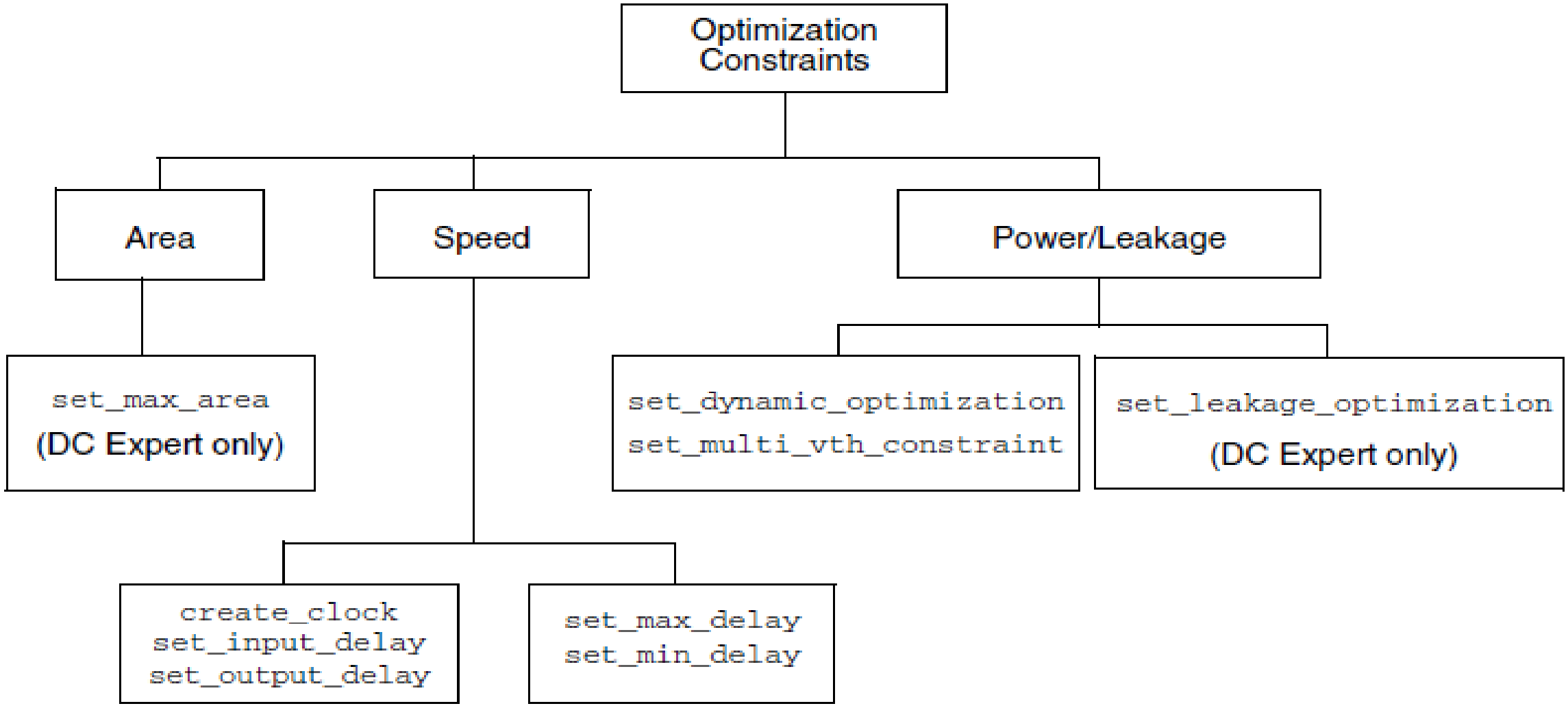
Design Rule Constraints:-

- The logic library defines these implicit constraints.
- These constraints are required for a design to function correctly.
- By default, design rule constraints have a higher priority than optimization constraints.


Design Rule Constraints
Maximum:
Transition Time
Fanout
Capacitance
Cell Degradation
Minimum Capacitance
set_max_transition
set_max_fanout
set_max_capacitance
set_cell_degradation
set_min_capacitance

Optimization Constraints

- Explicit constraints.
- Optimization constraints apply to the design on which you are working for the duration of the dc_shell session and represent the designs goals.
- During optimization, Design Compiler attempts to meet these goals, but no design rules are violated by the process.
- To optimize a design correctly, you must set realistic constraints

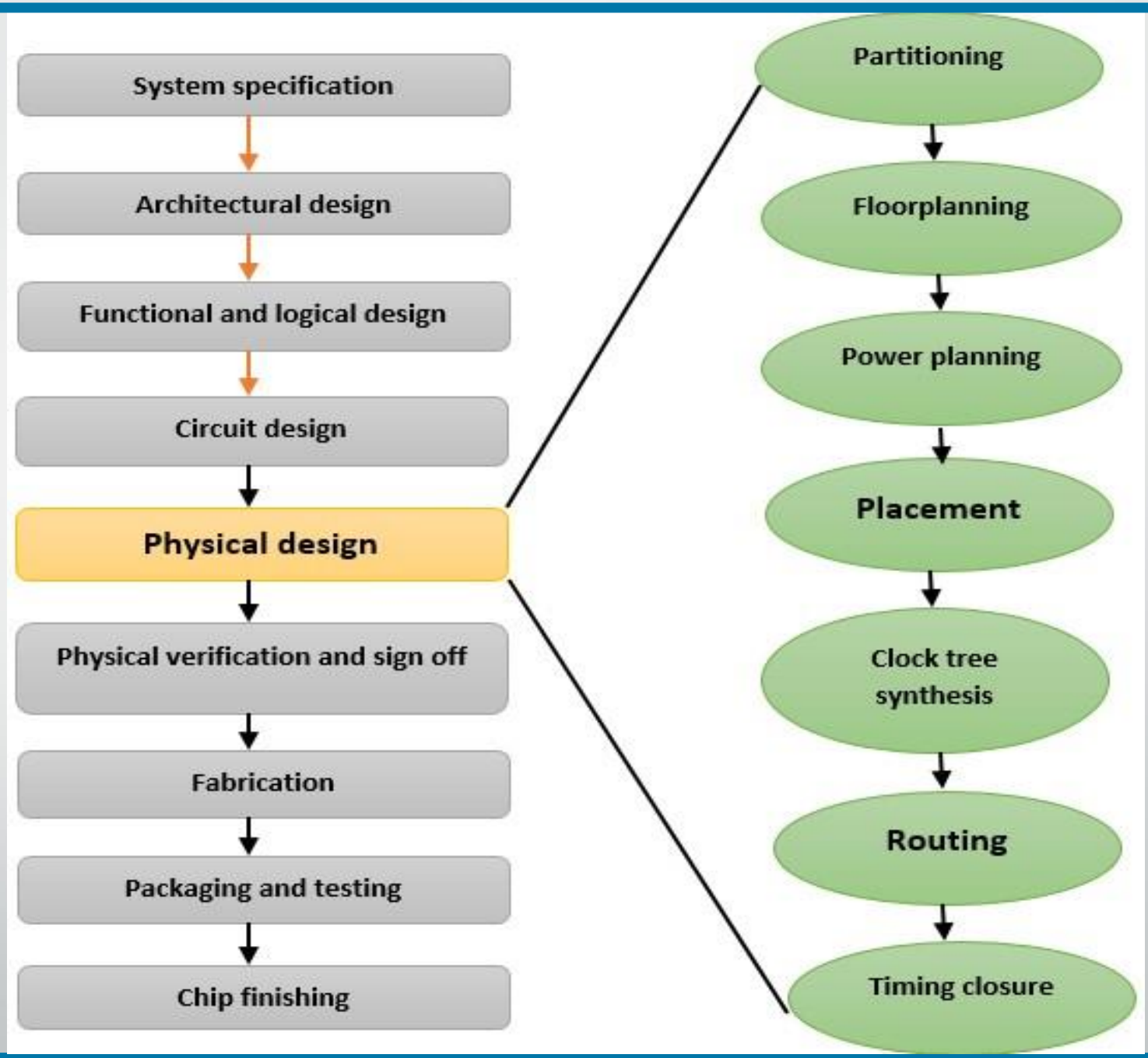

Optimization Constraints
Area
Speed
Power/Leakage
set_max_area
(DC Expert only)
set_dynamic_optimization
set_multi_vth_constraint
set_leakage_optimization
(DC Expert only)
create_clock
set_input_delay
set_output_delay
set_max_delay
set_min_delay

Outputs of Synthesis

- Netlist (.v)
- SDC (.sdc)
- UPF
- ScanDEF -> information of scan flops and their connectivity in a scan chain

Floorplanning

ASIC FLOW & PHYSICAL DESIGN FLOW



Introduction

- Floorplanning is a process of placing the Macros and I/O Ports in core area.
- Quality of floorplanning directly affects the following in a design:
  - Congestion and routing issues.
  - IR drop
  - Timing etc.

Goals

- We define the size and shape of your chip or block
- Place the IO pins/pads, macros and blockages in the core or chip area in order to effectively find the routing space between them.
- We reserve space for the placement of standard cells.

Inputs

1. NetlistTechnology (.v)
2. file (.techlef/.tf)
3. Timing Library files (.lib/.db)
4. Physical library (.lef)
5. Synopsys design constraints (.sdc)
6. Tlu+ or qrctech file

Sanity Checks-1

check_design :

This check analyzes the currently loaded netlist and reports the inconsistency if any. Netlist check mainly checks:

- Floating input pins and nets
- No direct connection between VDD and VSS
- Multidriven nets
- Combinational loops
- Unloaded outputs
- Mismatch pin count between instance and reference

Sanity Checks-2

check_library :

Performs consistency checks between the logical and physical library, across the logical library and within the physical library.

check_timing :

SDC file must be checked before start the design. Some of the common issues in SDC file are as follow.

- Unconstrained path
- Clock is reaching to all synchronous elements
- Multiclock driven registers
- Unconstrained endpoint
- Input/output delay missing for a port
- Slew or load constraint missing for a port

Floorplan Control Parameter

Aspect ratio:

- Aspect ratio will decide the size and shape of the chip.
- It is the ratio between horizontal routing resources to vertical routing resources (or) ratio of height and width.

Aspect ratio = width/height

Core utilization:-

- Utilization will define the area occupied by the standard cells, macros, and other cells.
- If core utilization is 0.8 (80%) that means 80% of the core area is used for placing the standard cells, macros, and other cells, and the remaining 20% is used for routing purposes.

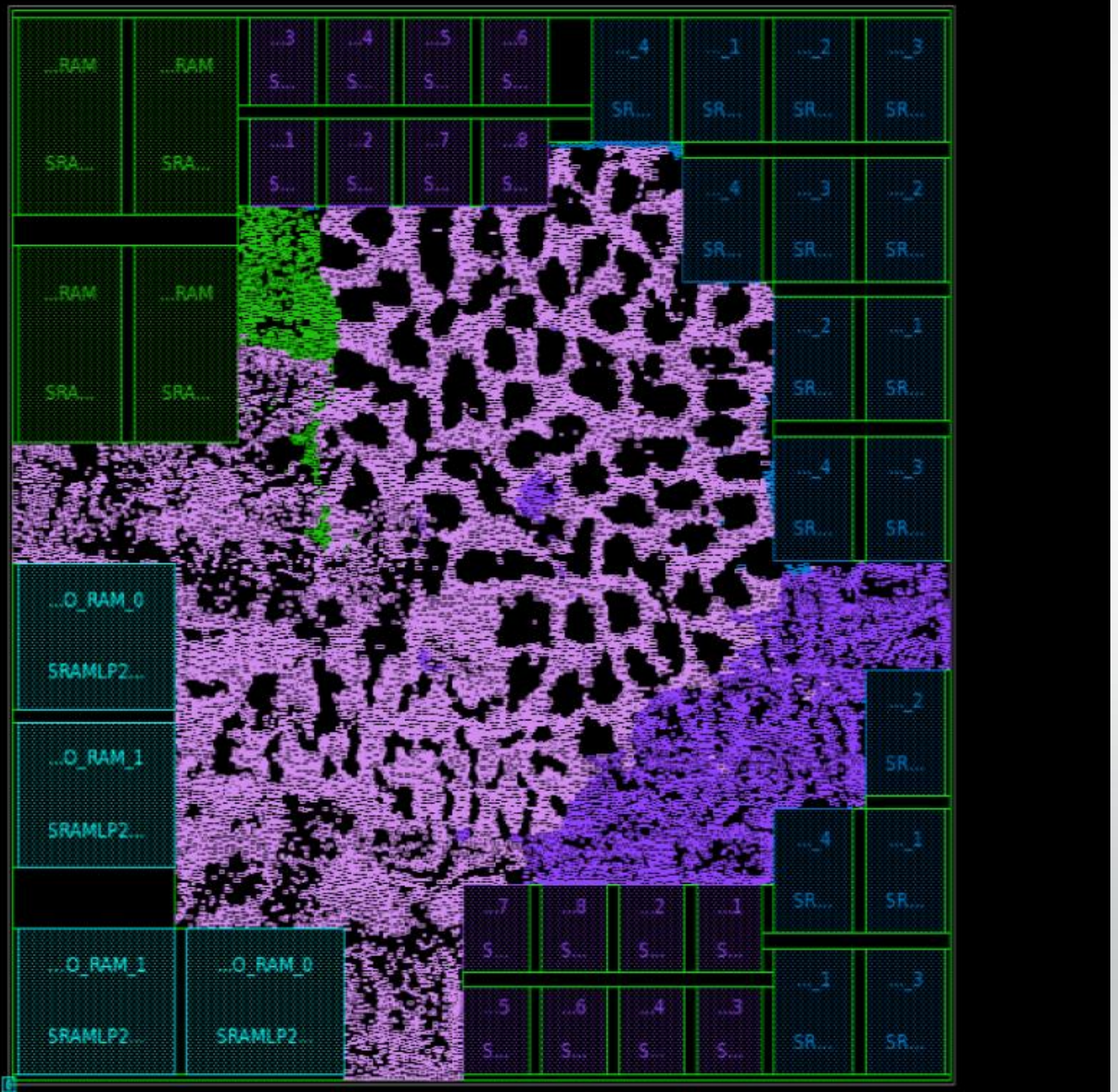
core utilization = (macros area + std cell area +pads area)/ total core area

Guidelines to place Macros

- Avoid crisscross connections between the macros using Fly-line analysis.
- Place the macros of same hierarchy together.
- Place macros near the boundary of the block.
- Provide a keep-out margin/ halo around all sides of the macros.
- Provide proper blockages as per the requirement.
- Proper channel spacing needs to be given between the macros

Channel spacing between macros =

(No. of pins x pitch)/(Available layers/2).



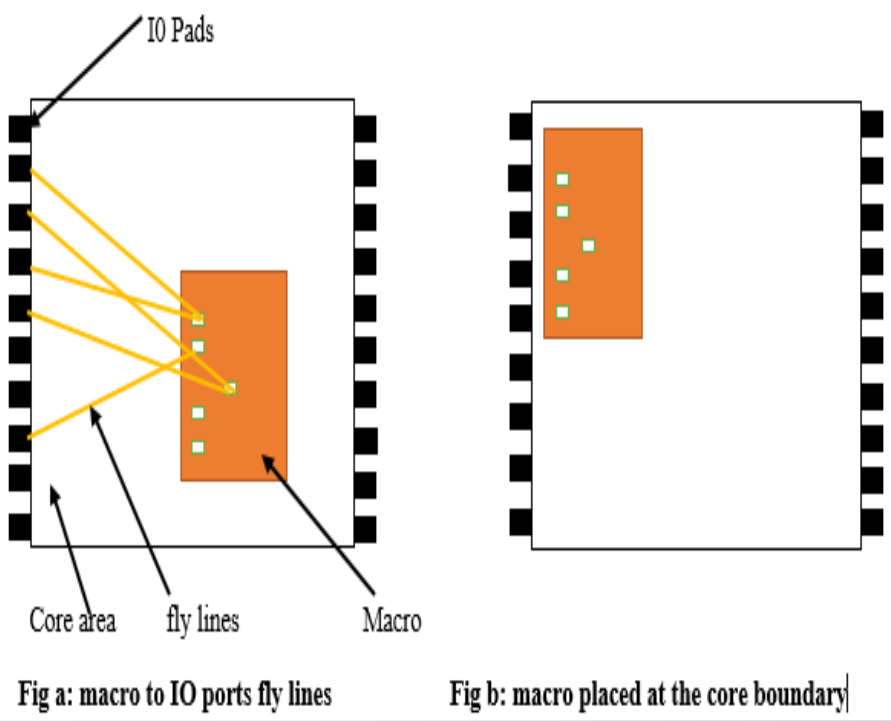


Fig a: macro to IO ports fly lines

Fig b: macro placed at the core boundary

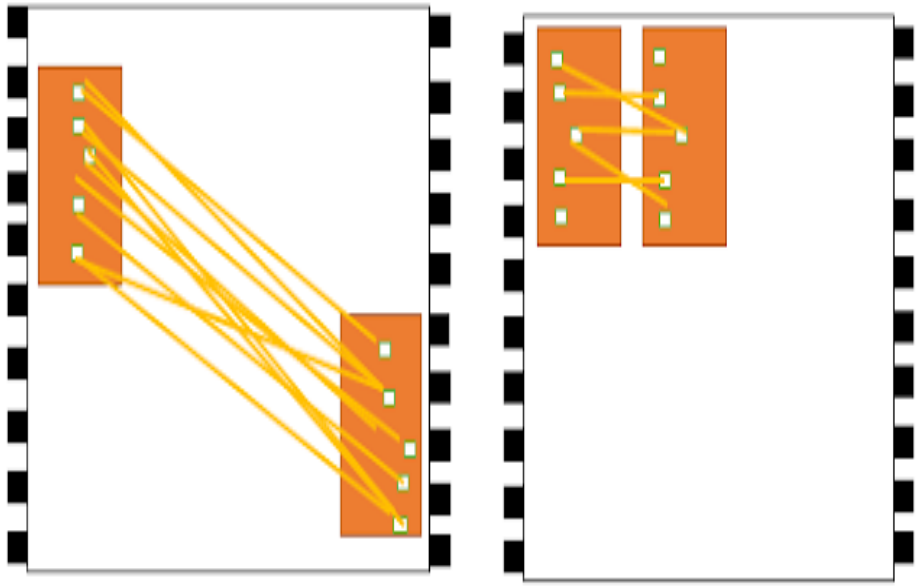


Fig a: macro to macro fly lines

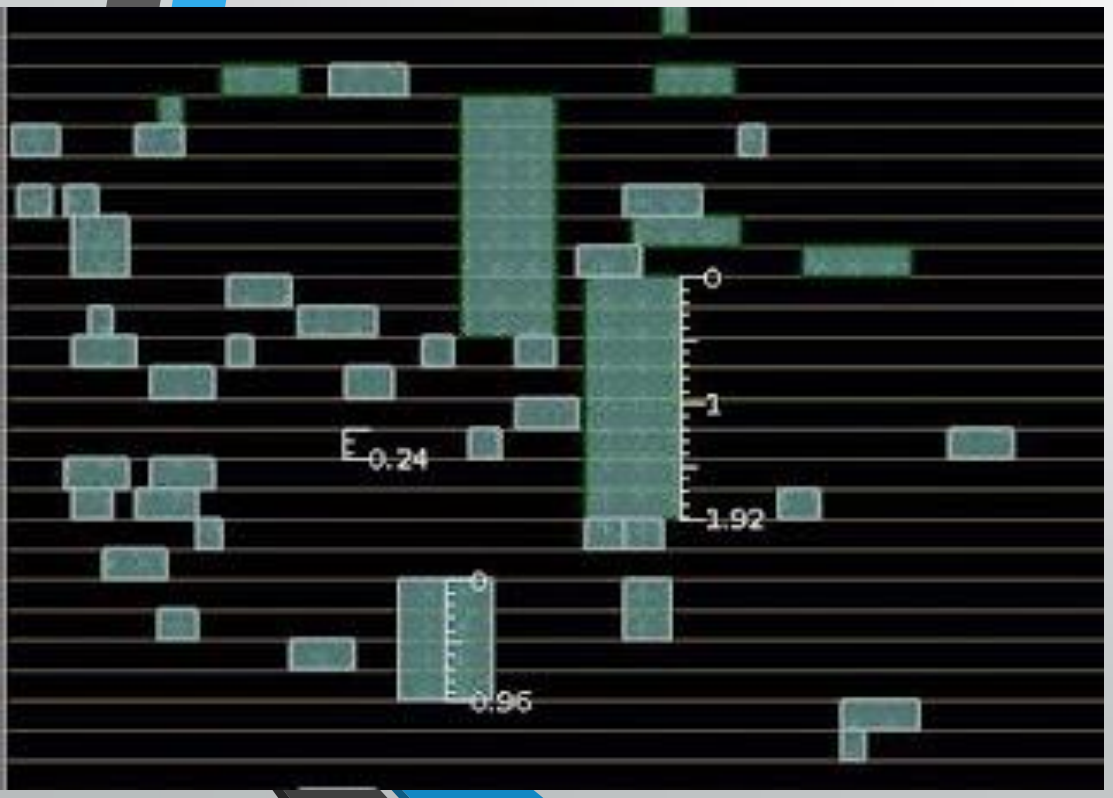
fig b: macros placed near to each other

Key terms related to floorplan

Blockages

1. Blockages are the specific location where the placing of cells is blocked.
2. Blockages guides the tool to not to place the standard cells, buffer and inverters at some particular area.
3. If the macros moved from one place to another place, blockages **will not move.**
4. Blockages are of three types.
    1. **Soft Blockage**
    2. **Hard Blockage**
    3. **Partial Blockage**

Key terms related to floorplan



Standard cell row:

- The area allotted for the standard cells on the core is divided into rows where standard cells are placed.
- The height of the row is equal to the height of the samllest standard cell and width varies.
- The height varies according to multiple standard cell row height. there may be double-height cells, triple-height cells, etc.
- The standard cells will sit in the row with proper orientation.

Key terms related to floorplan

Fly lines:

- Fly lines are a virtual connection between macros and macros to IO pads.
- This helps the designer about the logical connection between macros and pads.
- Fly lines act as guidelines to the designer to reduce the interconnect length and routing resources.
- fly lines are of two types:

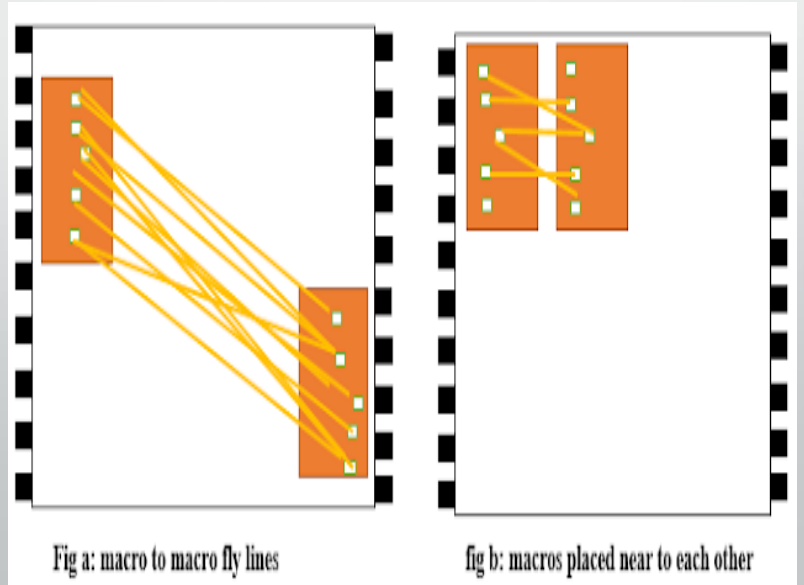


Fig a: macro to macro fly lines

fig b: macros placed near to each other

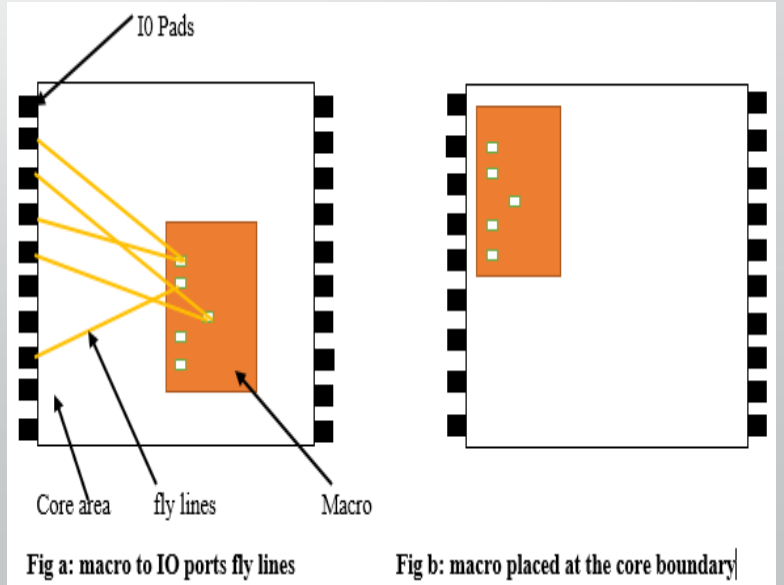


Fig a: macro to IO ports fly lines

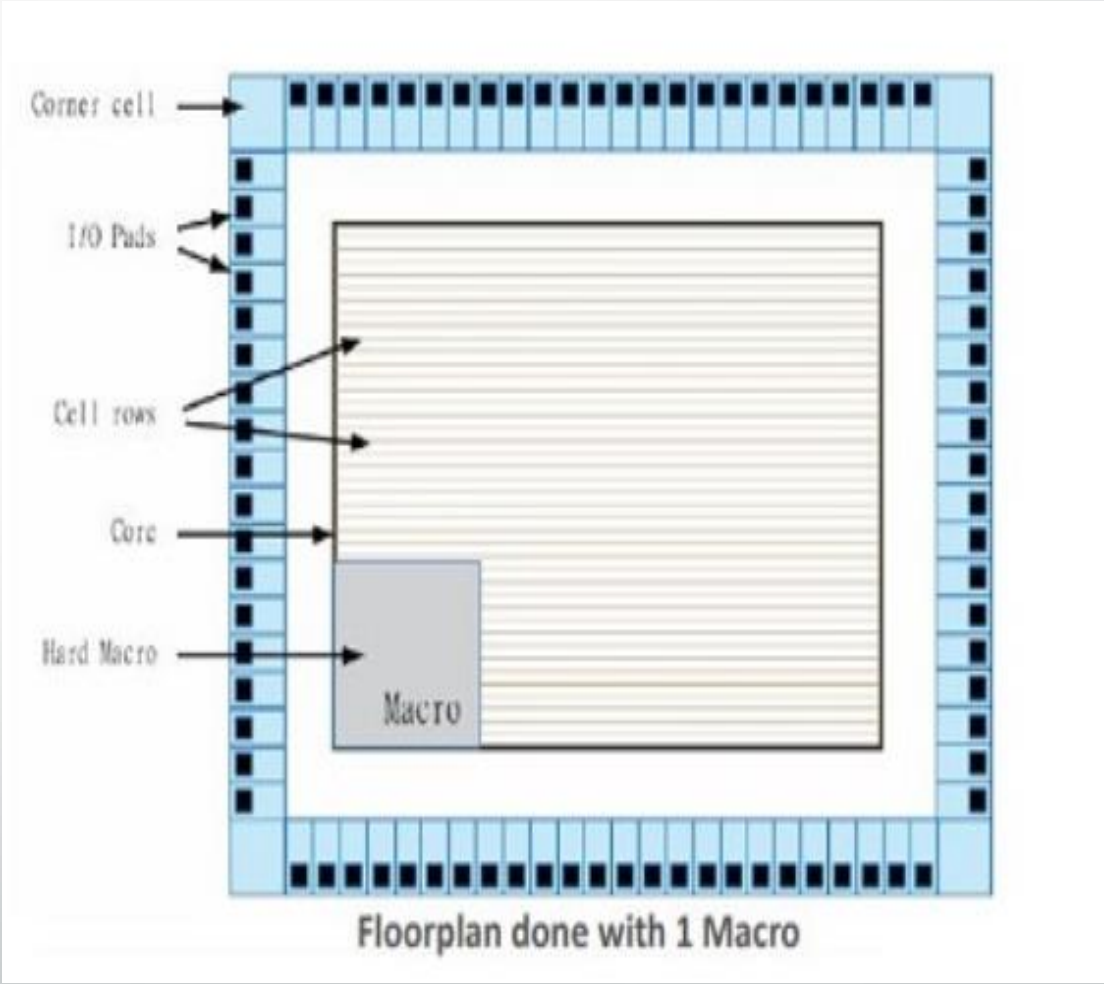
Fig b: macro placed at the core boundary

How to Qualify Floorplan

- No I/O ports short
- All I/O ports should be placed in routing grid
- All macros in placement grid
- No macros overlapping
- Add physical only cells (endcap/welltap).
- Check PG connections (For macros & pre-placed cells only)
- All the macros should be placed at the boundary
- There should not be any notches. If unavoidable, proper blockages has to be added
- Remove all unnecessary placement blockages & routing blockages (which might be put during floor-plan & pre-placing)

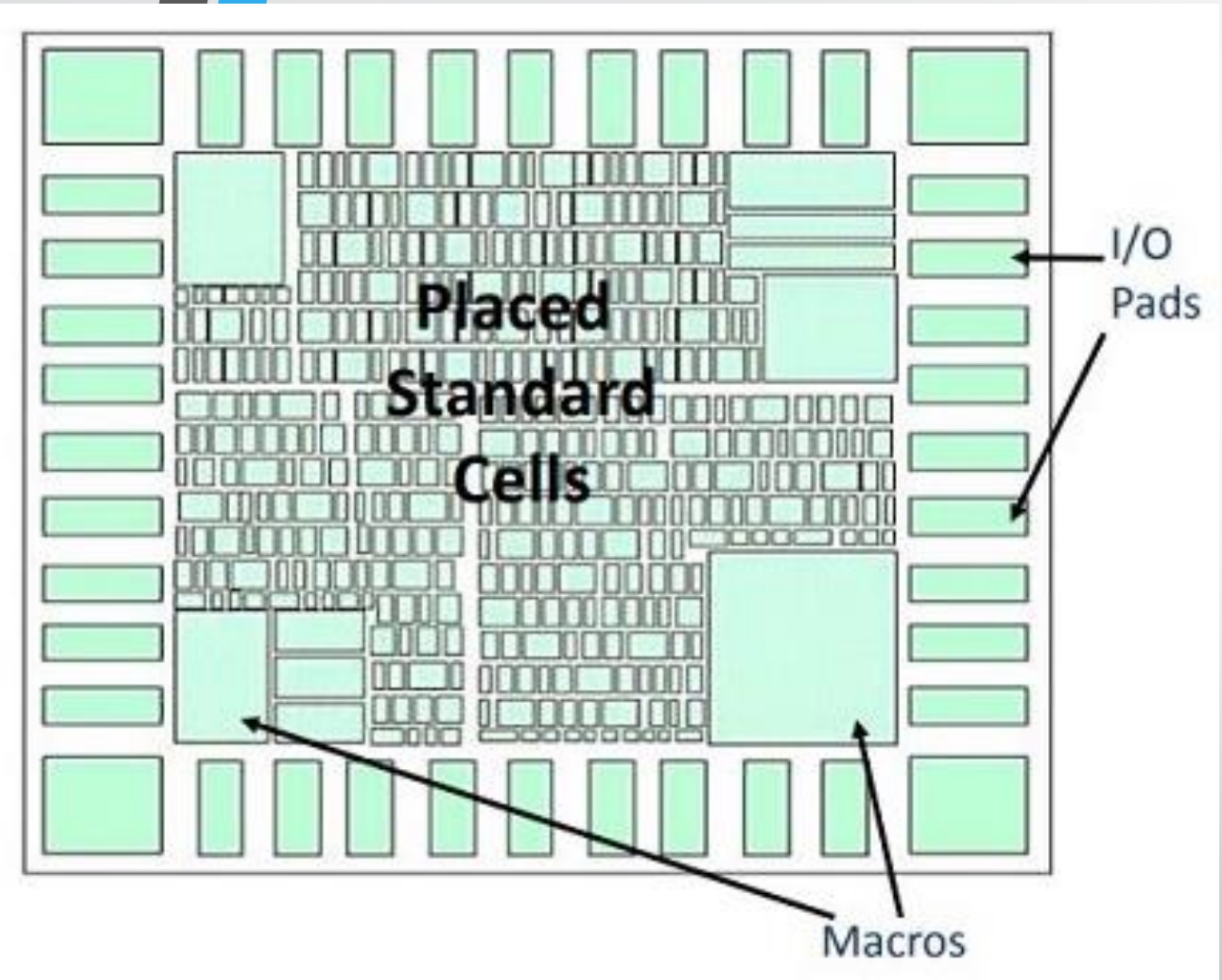
FLOORPLAN OUTPUTS

- IO ports placed
- Cell rows created
- Macro placement final
- Core boundary and area
- Pin position
- Floorplan def



Placement

Introduction

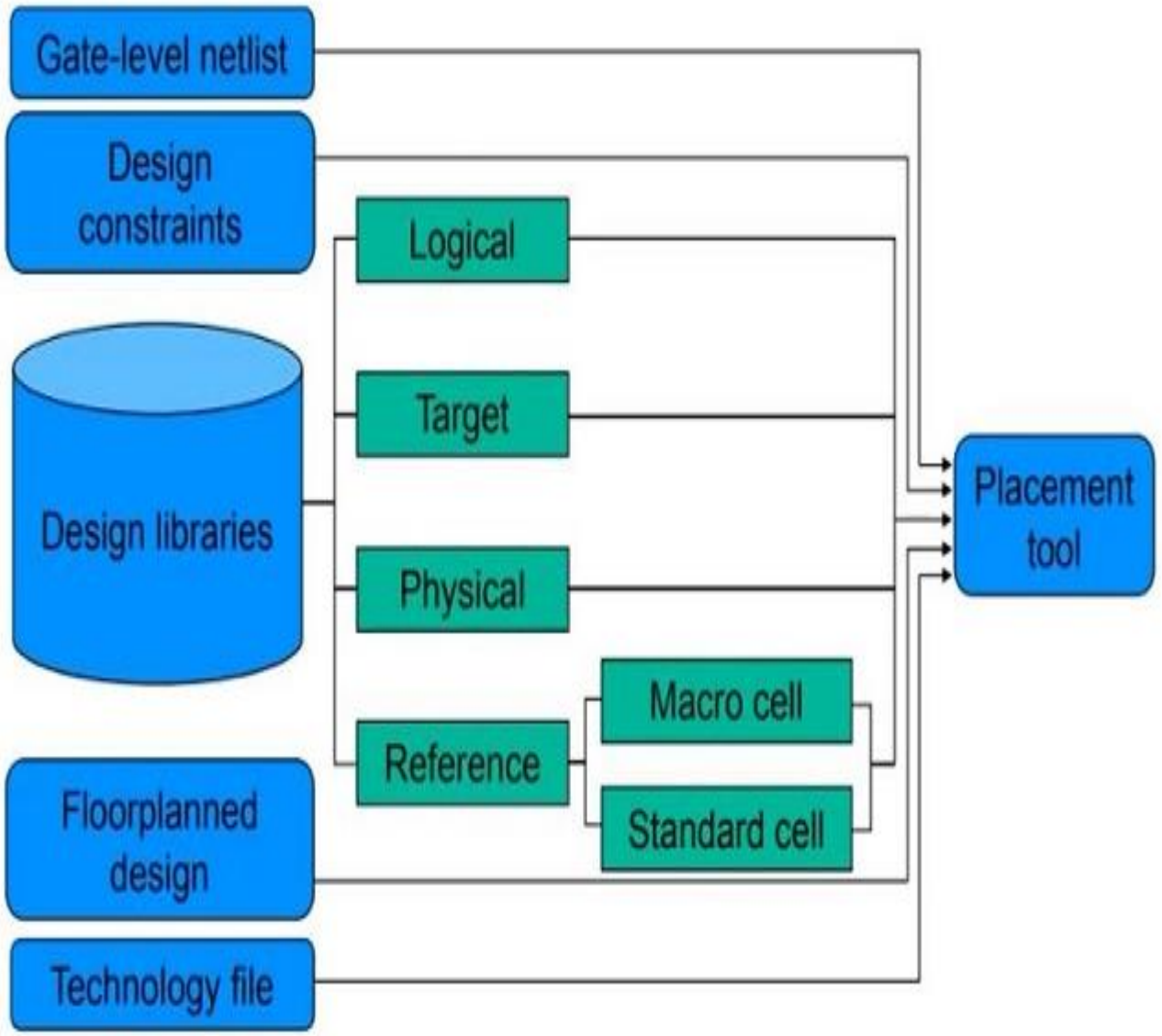


- Placement is the process of placing of all standard cells that are present in netlist by the tool into the core area.
- Placement does not just place the standard cells but it also optimizes the design while determining routability of the design.

Goals of Placement

- Timing, area and power optimization
- Routable Design
- Minimize congestion and congestion hotspots
- Minimum cell density, pin density
- No timing DRV'S

Inputs of Placement



- Netlist
- Physical libraries
- Floorplan DEF file
- Timing libraries
- RC corners
- SDC file
- UPF file (if the design has multiple voltage domains)

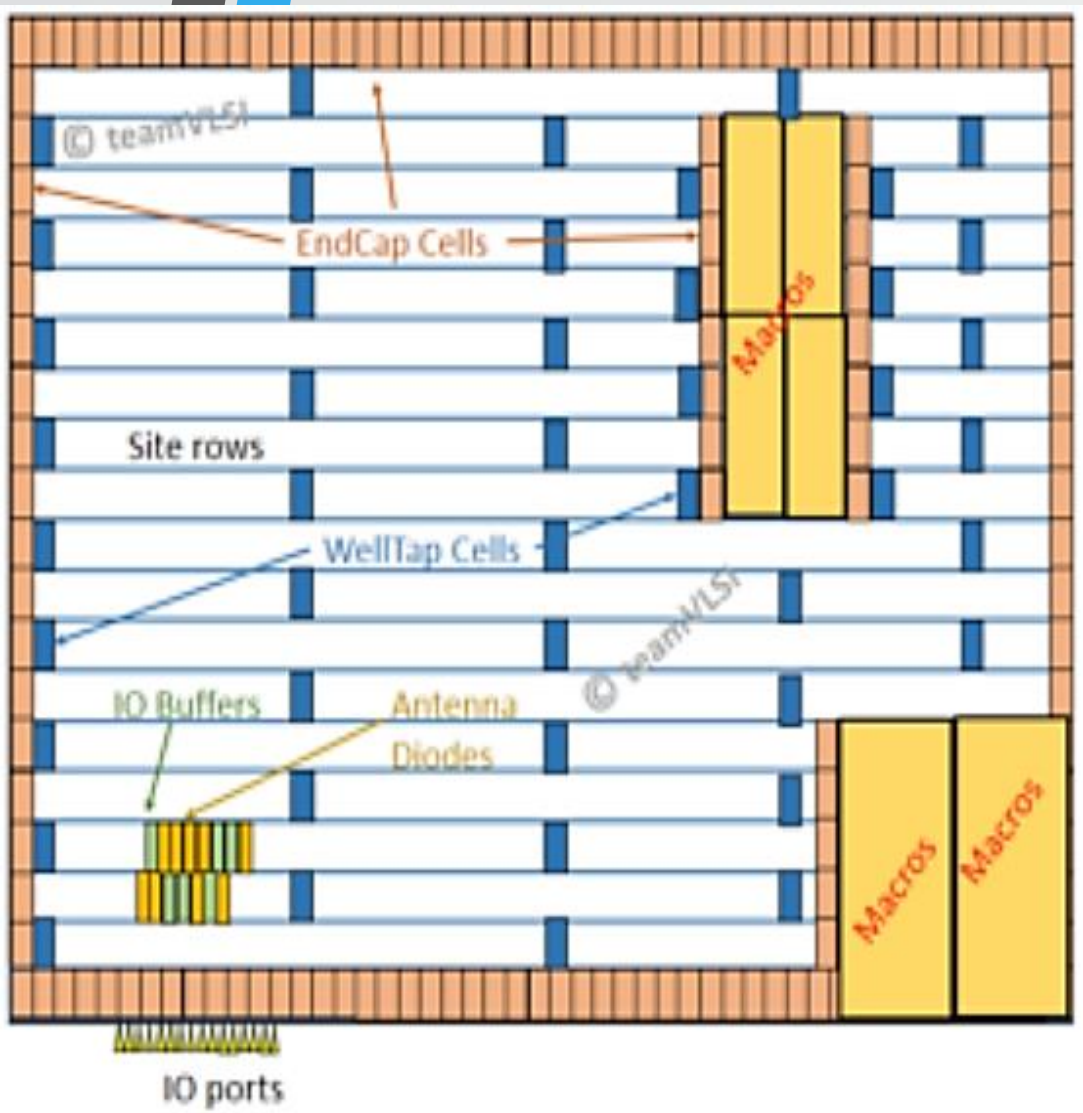
Checks before Placement

- Netlist should be clean.
- Floorplan DEF should have
  - Proper Pin placement
  - Macros and pre-placed cells in fix status.
  - Power planning free of DRCs
- Check for don't use, don't touch cells and routes should be applied properly.

Placement Steps

- Pre-placement
- Placement
- Post-placement

Pre – Placement

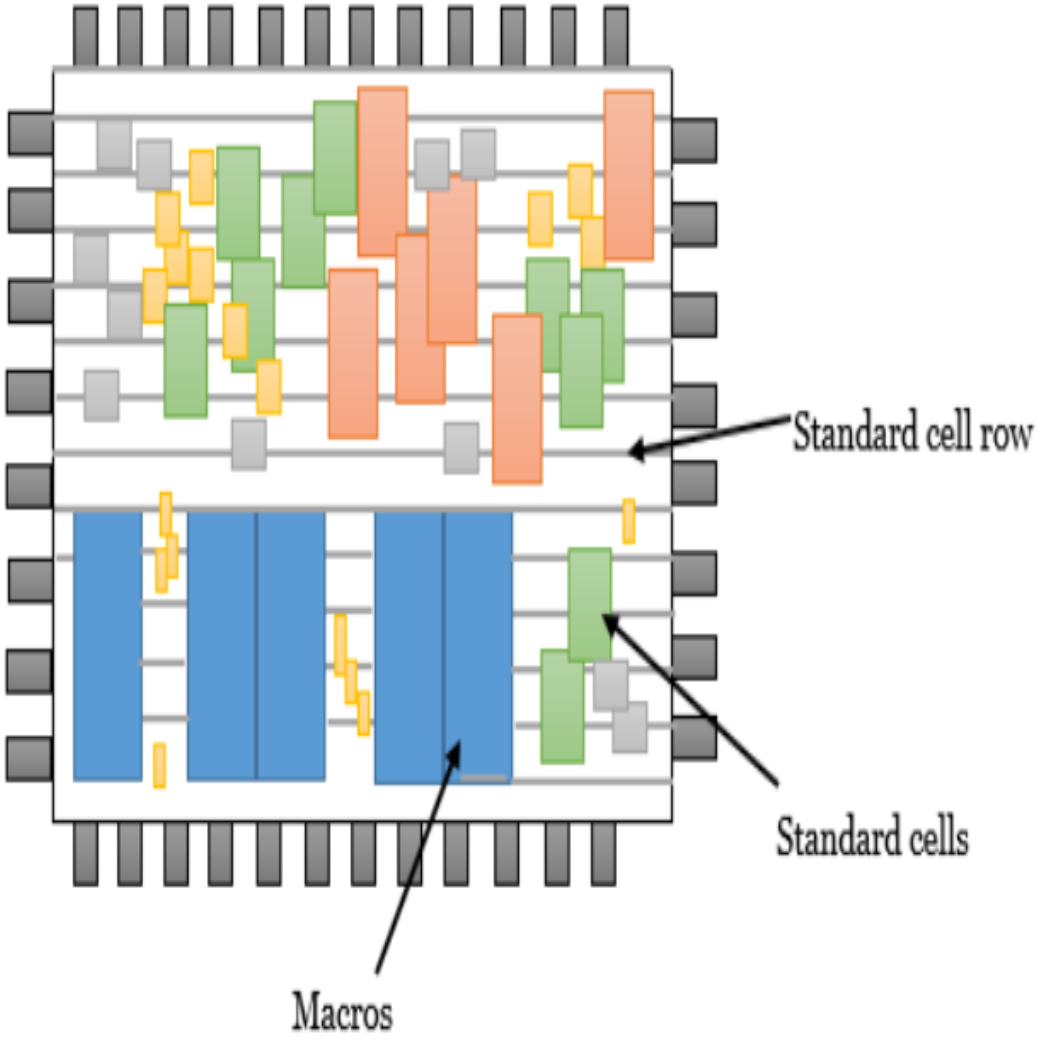


- During this step perform all the pre-placement checks
- If everything is clean then all the required pre-placement physical cells have to place (like Endcap, Well Tap, Spare cells, Decap cells etc.)

Placement

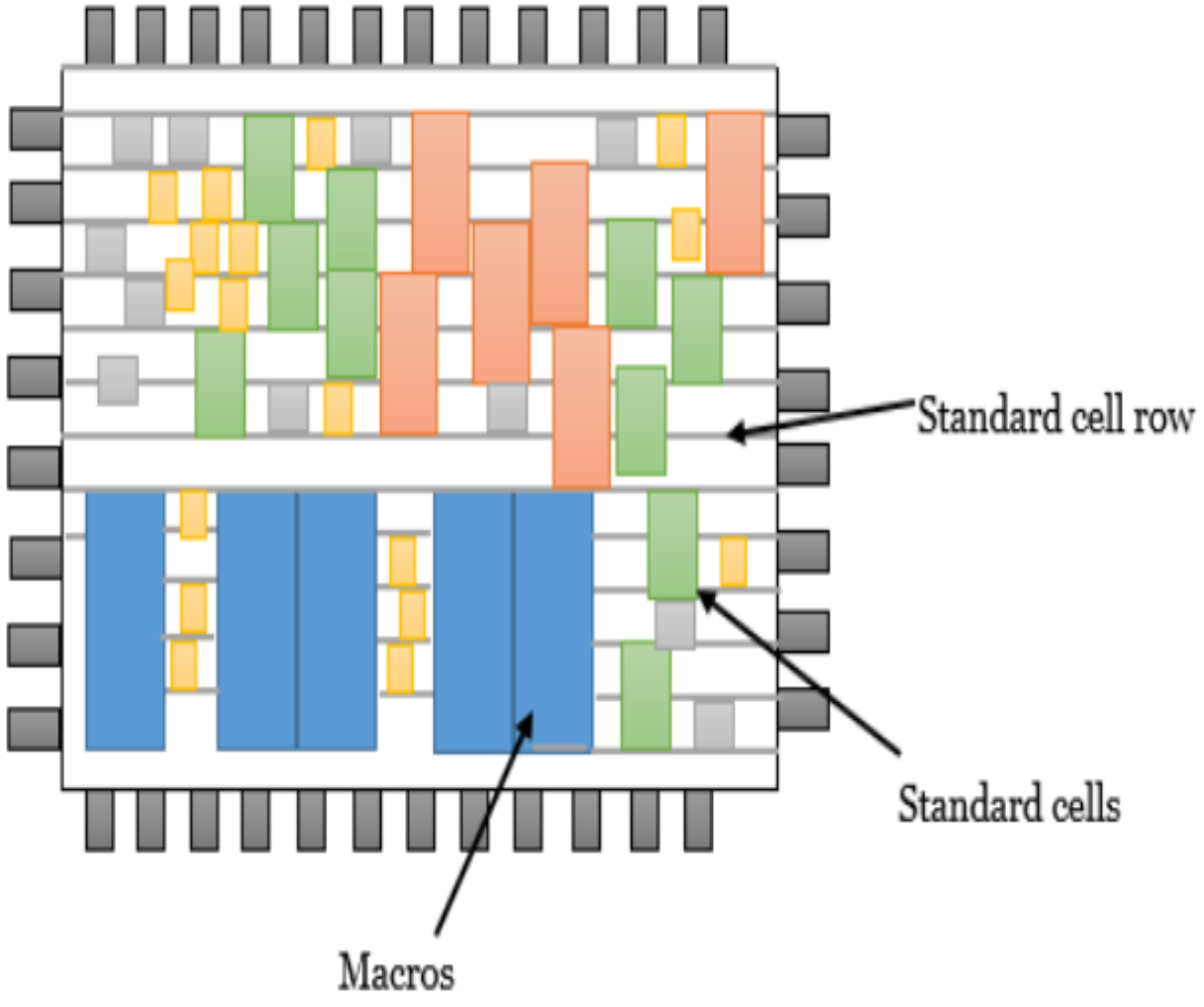
- During placement following steps are followed:
  - Initial Placement / Coarse Placement / Global Placement
  - Legalization
  - High Fanout Net Synthesis (HFNS)
  - Scan-Chain Reorder

Global Placement



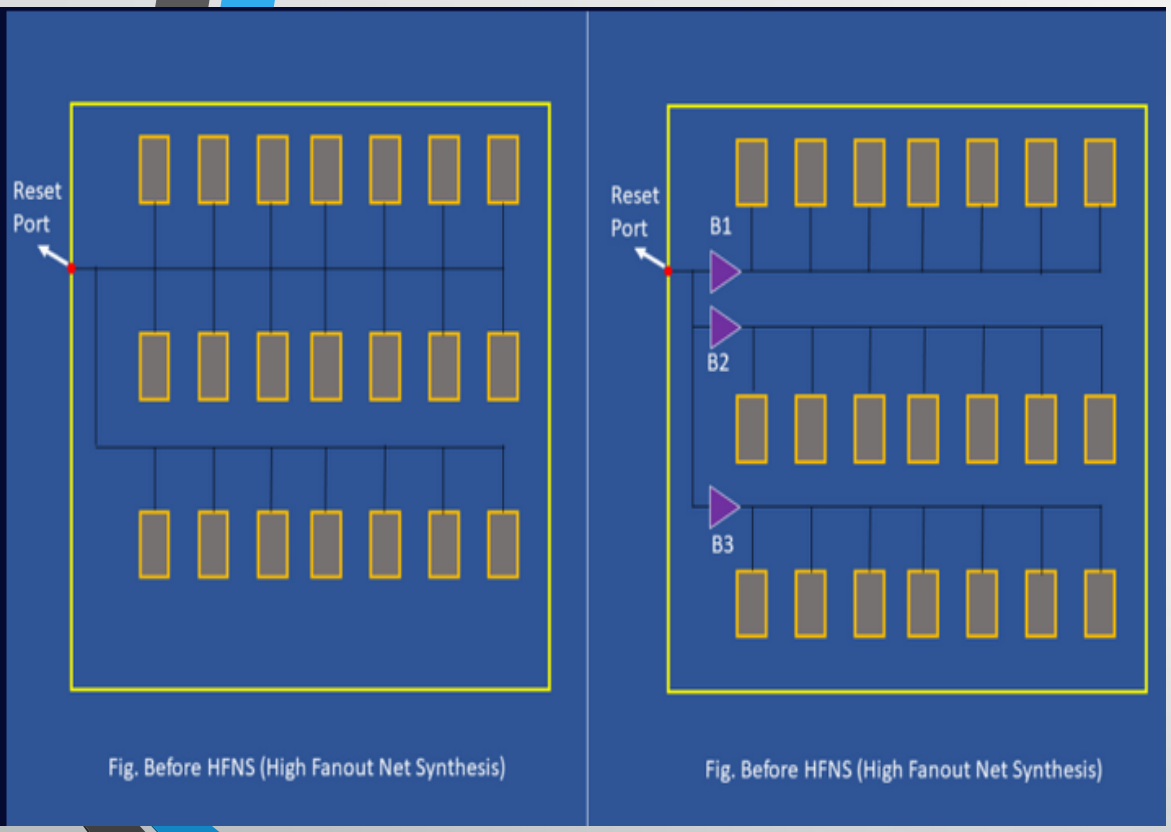
- To get approximate initial location according to timing, congestion and multi-voltage constraints
- Cells are not legally placed and are overlapping
- Standard cells must be in groups such that the number of connections between groups is minimum
- Coarse/ global placement is fast and accurate for initial timing and congestion analysis

Legalization



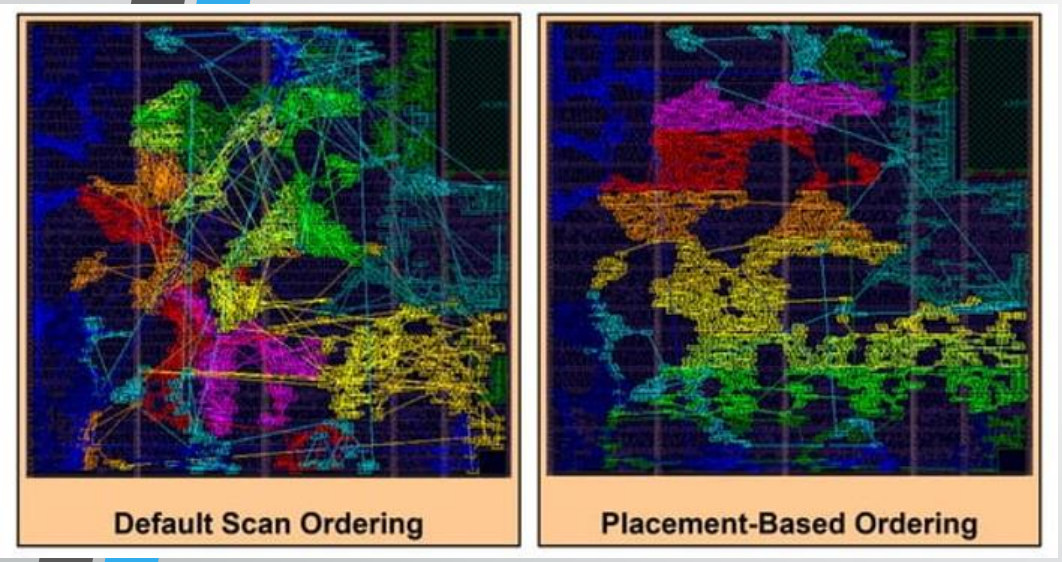
- To avoid cell overlapping
- To place cells at legalized locations by following DRC rules
- Placed macros are legally oriented with standard cell rows
- Placement Constraints followed
  - Placement Blockages
  - Placement Bounds
  - Density Constraints
  - Cell Spacing Constraints

High Fanout Net Synthesis (HFNS)



- Nets with sinks greater than the fanout limit is considered high fanout net
- Buffering of high fanout nets are done to balance the load
- Distribute the sinks on nets to different drivers

Scan Chain Reordering



- DFT tool flow makes a list of all the scan-able flops in the design and sorts them based on their hierarchy.
- Scan chains already present as groups of FFs that are serially connected through SI/SO pins
- Scan chain paths are active only during "test mode“ not during "functional mode“
- FFs are typically connected in alphanumeric order during synthesis, which is irrelevant for DFT, but not optimal for routing

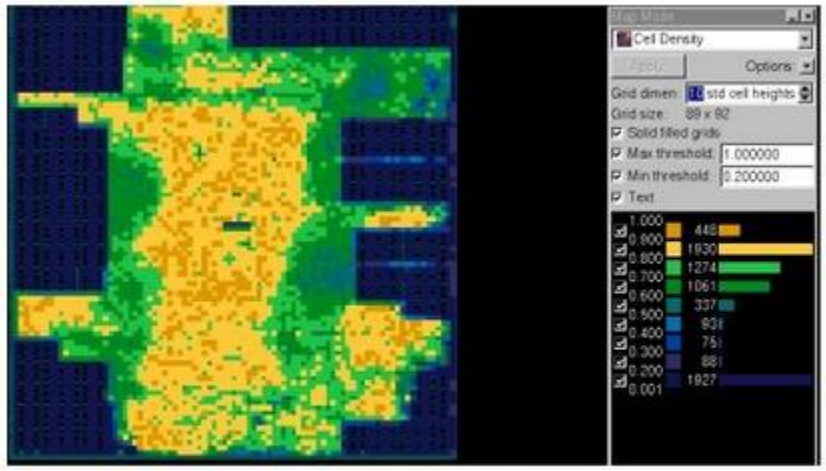
Post Placement

- During post placement tool tries to meet all design timing requirements like DRVs then WNS (Worst Negative Slack) and TNS (Total Negative slack) only for setup slack
- Finally tie cells also can be placed by the tool for the required input pins.
- Tie off cell instances provide connectivity between the Tie high and Tie-low logical inputs pins of the Netlist instances to Power and Ground
- Tie off cells are placed after the placement of Standard Cells
- After Placement, report Congestion, Utilization and Timing

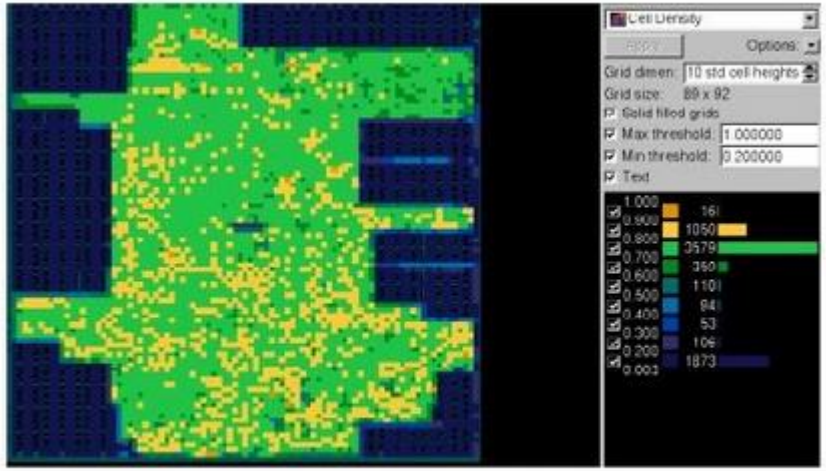
Congestion

- Types of Congestion
  - Routing Congestion Analysis
    - Congestion in general referred to Routing Congestion
    - Routing congestion is the difference between supplied and available tracks
    - A track is nothing but a routing resource which fills the entire Core
  - Placement Congestion Analysis
    - Placement Congestion is due to overlap of Standard Cells, it is called Overlapping rather than called as Congestion
    - Overlapping issue can be fixed by aligning cells to the Placement Grid by Legalization

Congestion Fixes



Before Fixing



After Fixing

- Add placement blockages in channels and around macro corners
- Review the macro placement
- Reduce local cell density using density screens
- Reordering scan chain to reduce congestion
- Congestion driven placement with high effort
- Continue the iterations until good congestion results
- Density screen is applied to limit the density of standard cells in an area to reduce congestion due high pin density

Clock Tree Synthesis

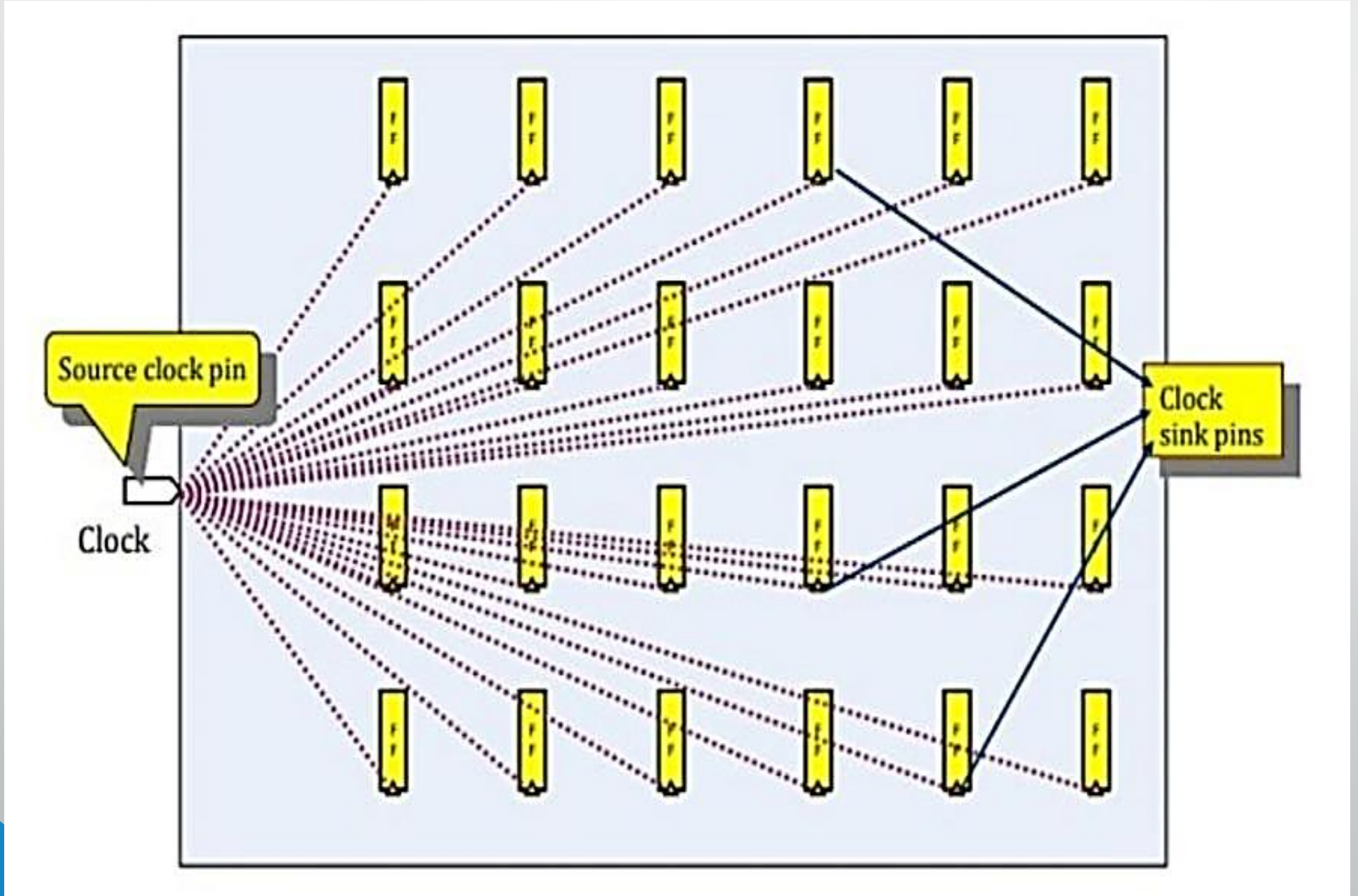
Introduction

- Process of distributing clock signals to clock pins of sequential elements
- Addition of buffers/inverters along the clock paths done to minimize skew and insertion delay .
- CTS QoR decides timing convergence & power.
- Efficient clock architecture, clock gating & clock tree implementation helps to reduce power.

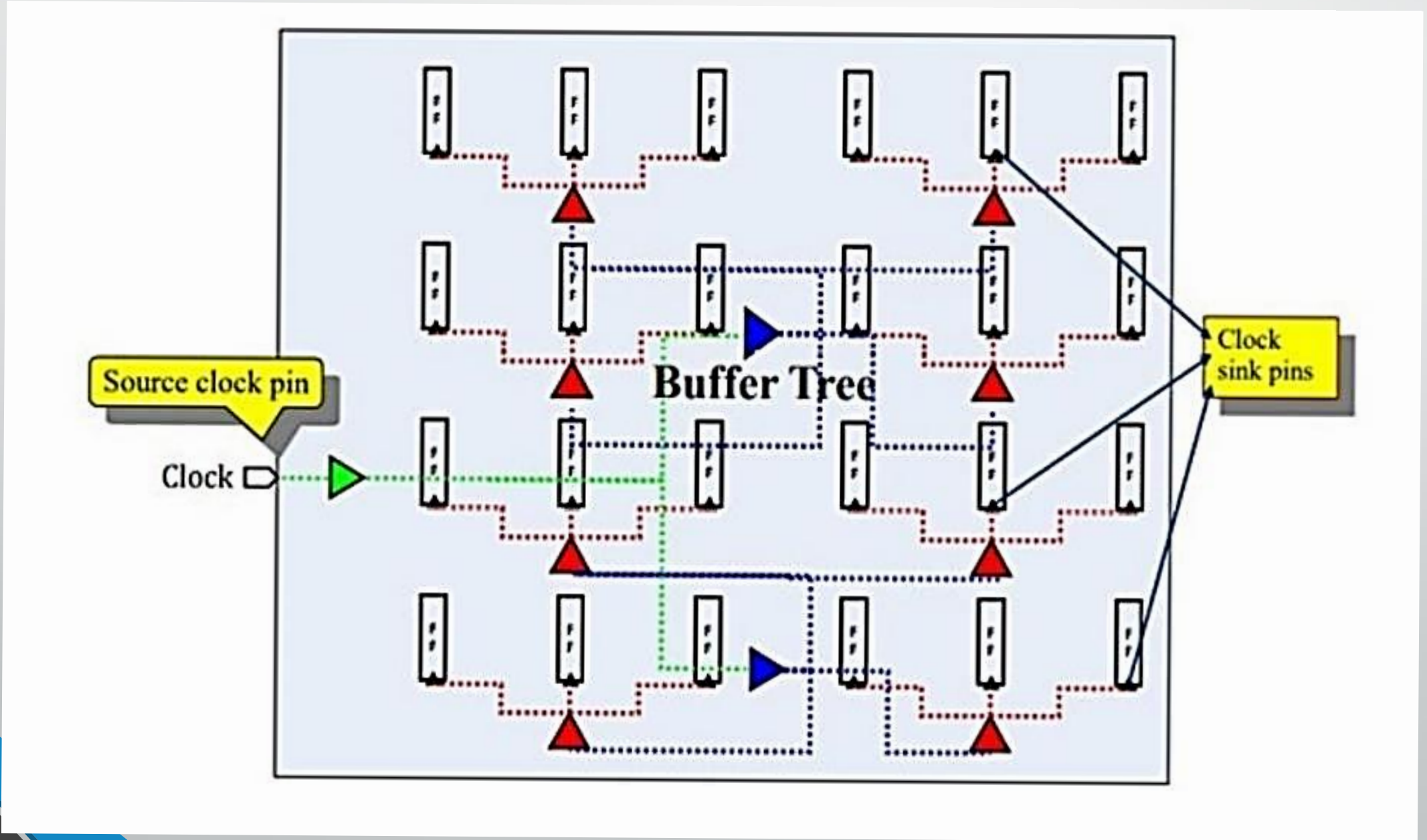
Checks to start CTS

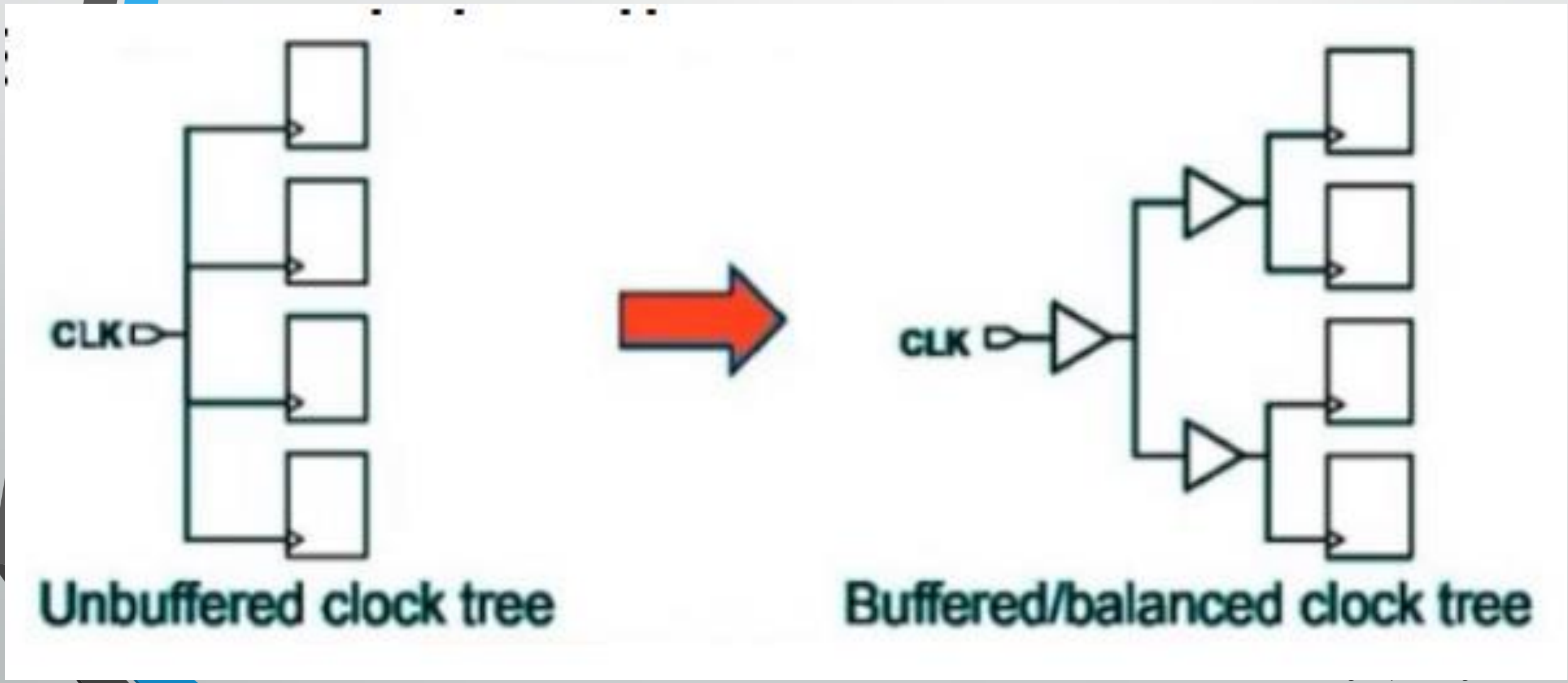
- Placement completed
- Power and Ground nets pre-routed
- Acceptable congestion
- Acceptable timing (0ns slack)
- No max cap/trans violations
- High fan-out nets such as scan enable, reset are synthesized with buffers.

Before CTS all Clock Pins are driven by a single Clock Source



After CTS the buffer tree is built to balance the loads and minimize the skew




CLK
CLK
Unbuffered clock tree
Buffered/balanced clock tree

Goals of CTS

- Minimum Skew and Insertion delays
- No clock tree DRC Violations
- No or Acceptable Timing Violations (Setup & Hold)
- No or Minimal Congestion at post CTS
- Acceptable power Numbers at post CTS
- Acceptable utilization jumps at post CTS

Inputs to CTS

- Placement def
- Target latency and skew if specify (SDC)
- Buffer or inverters for building the clock tree
- The source of clock and all the sinks where the clock is going to feed (all sink pins).
- Clock tree DRC (max Tran, max cap, max fan-out, max no. of buffer levels)
- NDR (Nondefault routing) rules (because clock nets are more prone to cross-talk effect)
- Routing metal layers used for clocks.

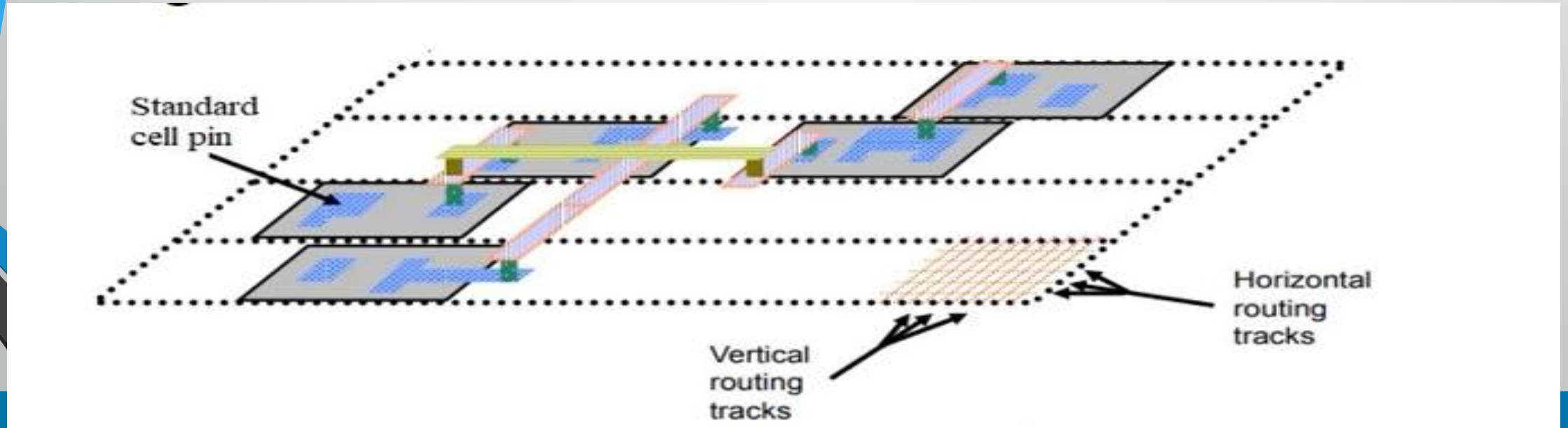
CTS Outputs

- Timing report
- Congestion report
- Skew report
- Insertion delay report
- CTS DEF file

ROUTING

ROUTING

- Making the signal connections between signal pins using metal layers are called Routing.
- Routing is the stage where CTS and optimization where exact paths for the interconnection of standard cells and macros and IO pins are determined.



Objectives And Goals

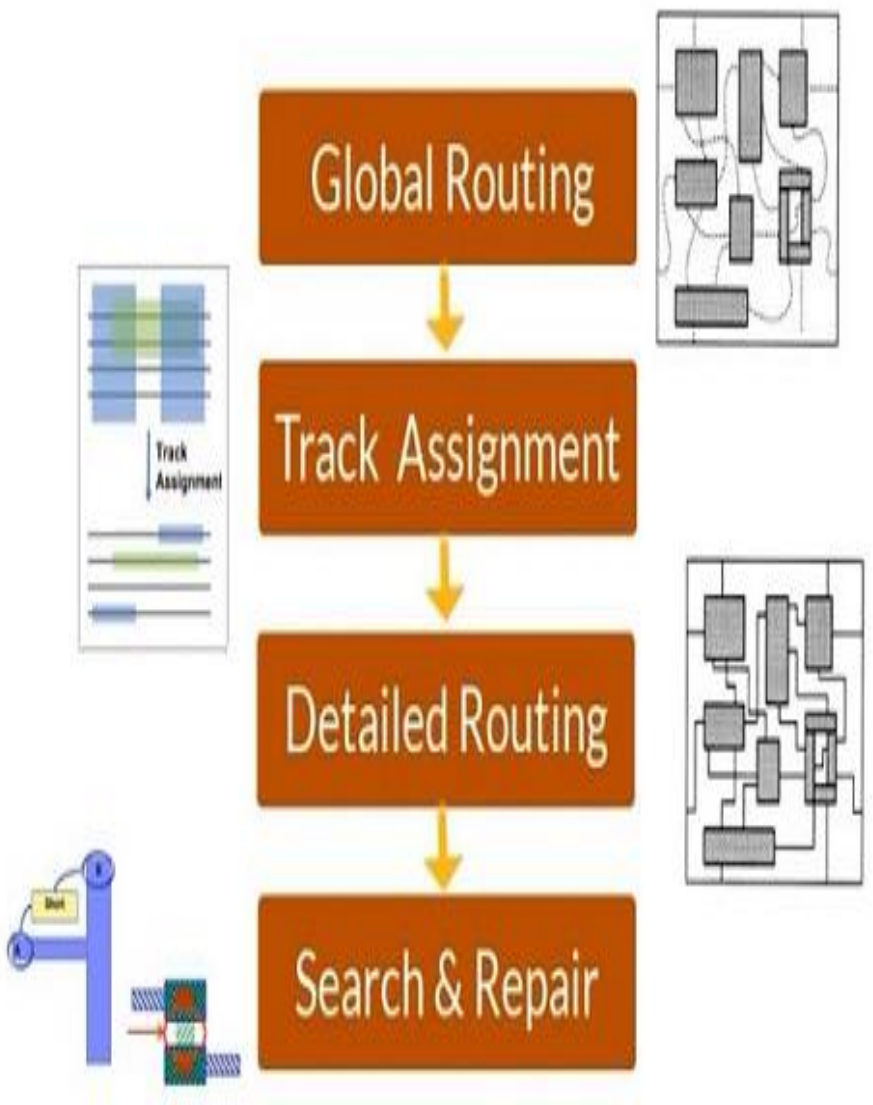
Routing Objectives:

- Skew Requirements.
- Open/Short Circuit Clean.
- Routed paths must meet setup and hold timing margin.
- DRVs max. Capacitance/Transition must be under the limit.
- Metal traces must meet foundry physical DRC requirements.

Goals of Routing:

- Minimize the total interconnect length.
- Minimize the critical path delay.
- Minimize the number of layer changes that the connection have to make.
- Complete the connections without increasing the total area of the block.
- Meeting the timing DRCs and obtaining a good Timing QOR.
- Minimizing the Congestion hotspot.
- Reduction in cross-talk noise.

Routing Flow

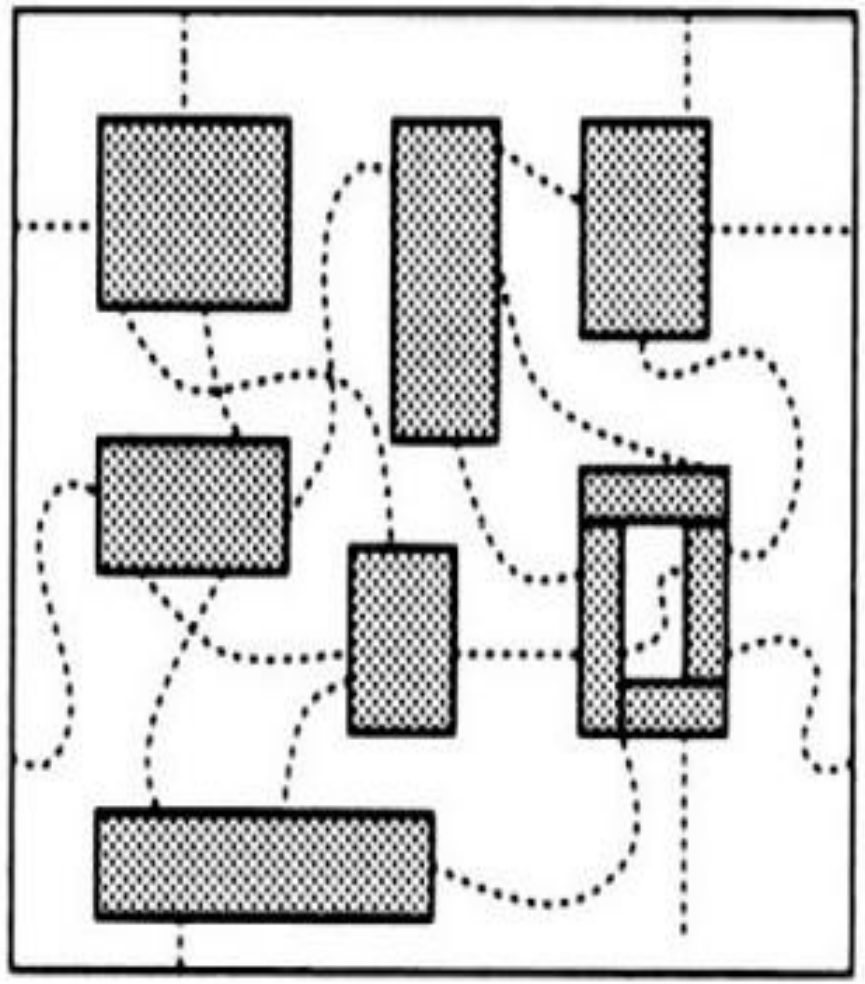


The different tasks that are following in the Routing stage are as follows.

- Global Routing
- Track Assignment
- Detail Routing
- Search and Repair

All clock/signal nets will be completely routed and should meet all timing, and most all DRC, requirements.

Global Routing

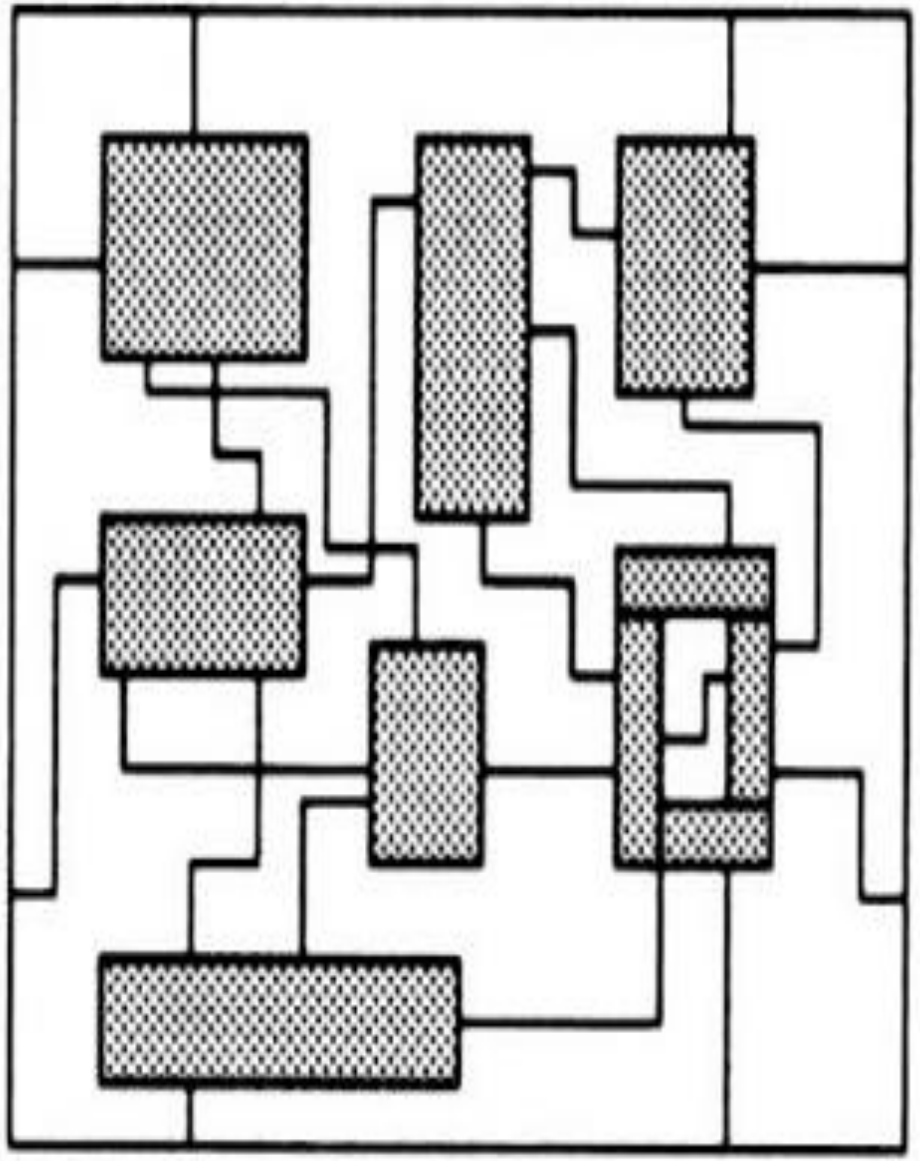


- Identifying routable path for the nets driving pins in a shortest distance.
- Does not consider DRC rules, which gives an overall view of routing and congested nets.
- Avoid routing areas and also long detours.
- Avoid routing over blockages.

Track Assignment

- Takes the Global Routed Layout and assigns each nets to the specific Tracks and layer geometry.
- It does not follow the physical DRC rules.
- It will do the Timing aware Track Assignment.
- It helps in Via Minimization.

Detailed Routing



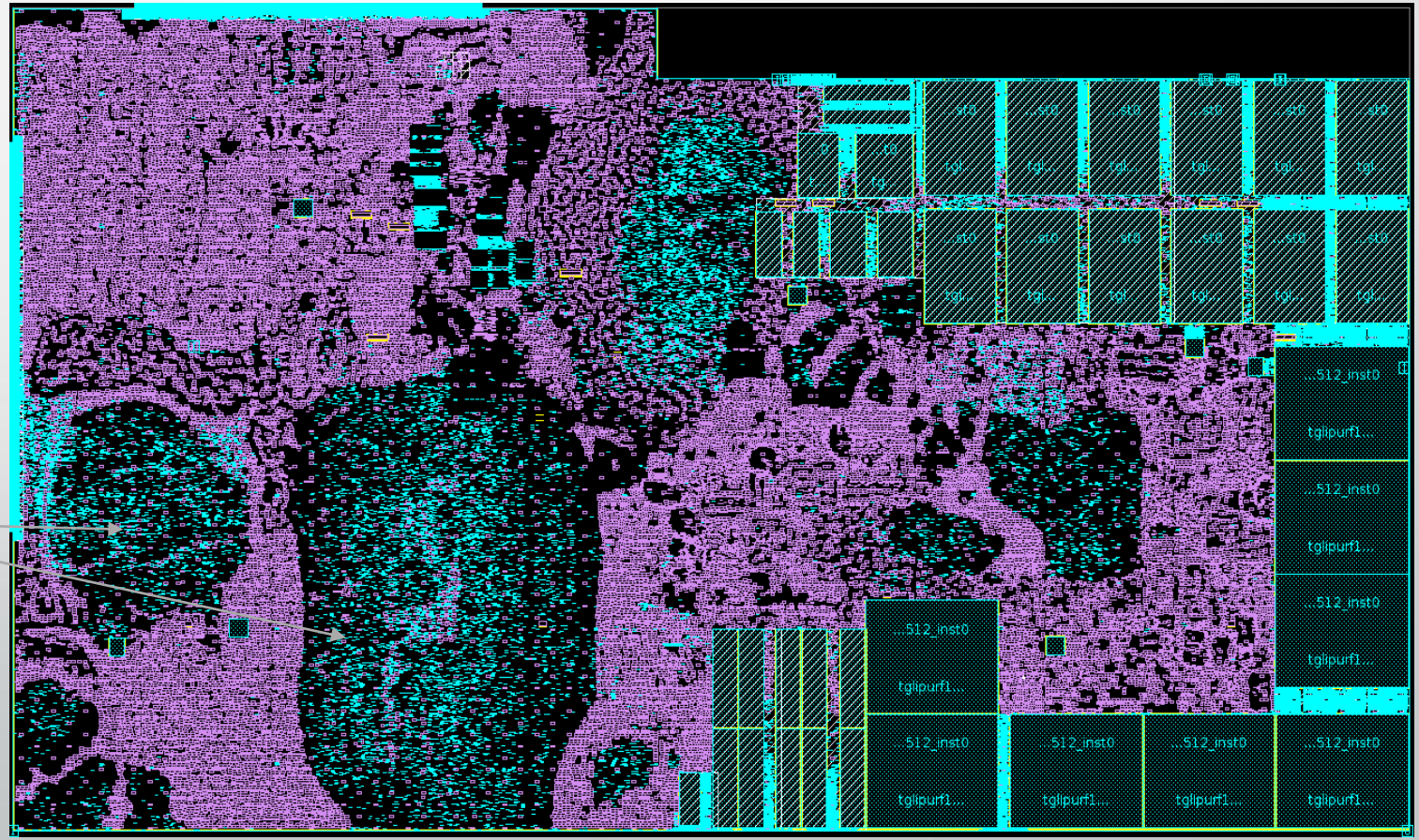
- Detailed routing follows up with the track routed net segments and performs the complete DRC aware and timing driven routing.
- It is the final routing for the design built after the CTS and the timing is freeze.
- Filler Cells are adding before Detailed Routing.
- Detail Routing is done after analyze the cause for congestion in the design, change floorplan etc.

Search And Repair

- The search and repair stage is performed during detailed routing after the first iteration.
- In search and repair , shorts and spacing violations are located and rerouting of affected areas to fix all the possible violation is executed.

Filler Cell Insertion

Non logical cells or filler cells are used to establish the continuity of the N- well and the implant layers on the standard cell rows.



Outputs Of Routing

- SPEF File
- GDS-II
- Netlist
- SDC

THANK YOU