DOCUMENTATION OF ROUTING

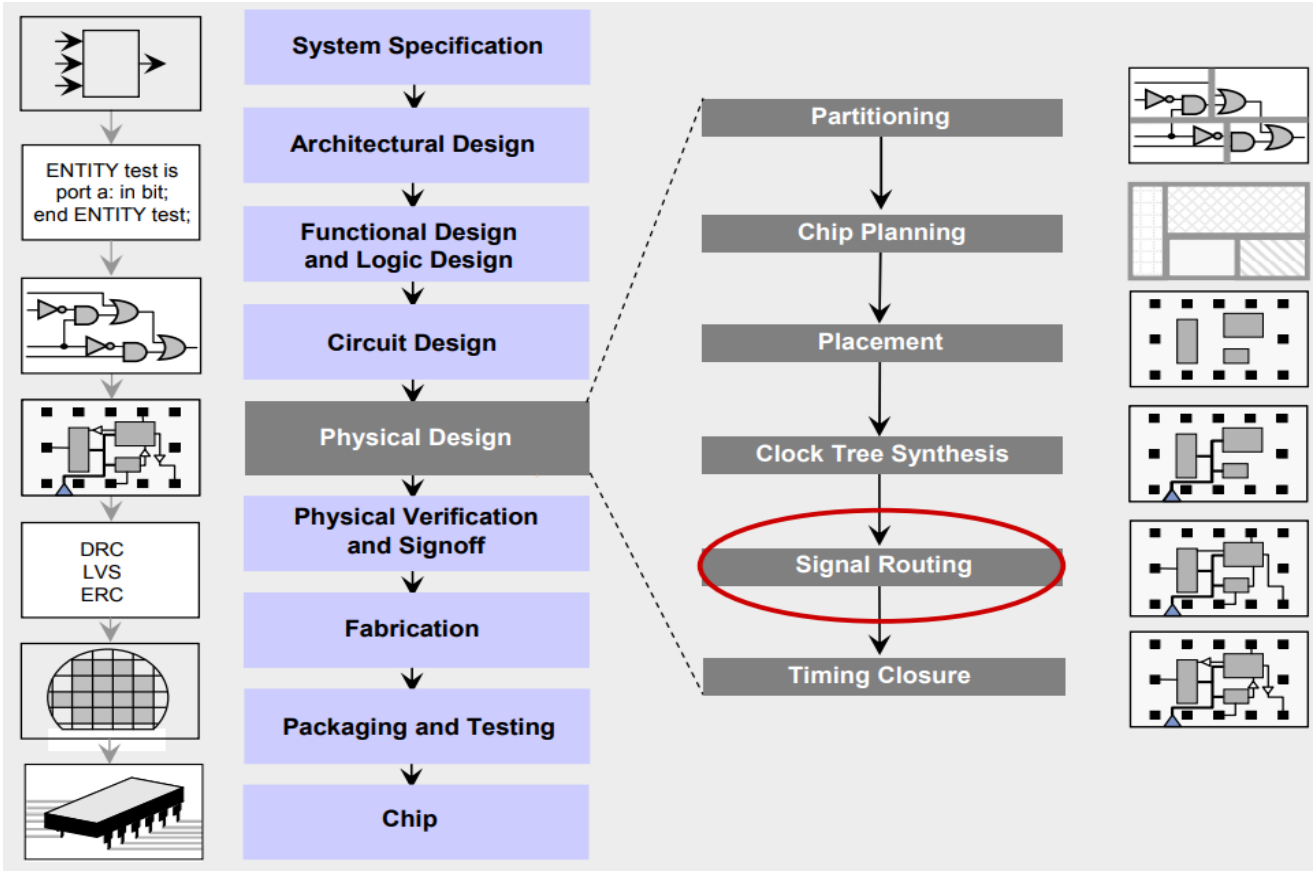
Date: 13-Feb-2025

Prepared by: Anik Balo

Contents

Routing

Routing is the process of creating physical connections based on logical connectivity. Signal pins are connected by routing metal interconnects. Routed metal paths must meet timing, clock skew, max trans/cap requirements and physical DRC requirements.



Routing Objective:

- Minimize the total wire length, vias, Routing area and congestion hotspots
- Complete the routing within the area of the design
- 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
- Layout geometries should meet current density specification

Inputs of routing:

- CTS optimized Database or Netlist, CTS DEF and SDC
- NDRs
- Timing libraries and Physical libraries
- Tech lef
- Captable or QRC tech file

Output of routing:

- Design with completed interconnection and geometric layout of the nets
- Netlist
- SDC
- GDS file
- Routing DEF file with on opens and shorts
- Timing and Congestion report
- SPEF

Checklist before routing:

- Placement, CTS and optimization should be completed
- PG nets should be pre-routed
- Timing DRC and QoR of post CTS should be acceptable
- High fanout nets(HFNS) should not be greater than the specified limit
- Check for blocked PG port
- Checks for overlapping cells in the design
- Estimated congestion – acceptable
- Check routability

Checklist after routing:

- **DRC (Design Rule Check)**: Ensure there are no design rule violations.
- **LVS (Layout vs. Schematic)**: Verify that the layout matches the schematic.
- **Timing Analysis**: Check for setup and hold time violations.
- **Antenna Effects**: Ensure there are no antenna violations.
- **Double Via Coverage**: Verify double via coverage for yield enhancement.
- **Signal Integrity**: Check for crosstalk and noise issues.
- **Power Analysis**: Ensure the power distribution network is robust.
- **Thermal Analysis**: Check for any potential thermal issues.
- **Shorts and Opens**: Ensure there are no shorts or open routes.
- **Manufacturability Checks**: Verify that the design is manufacturable without issues.

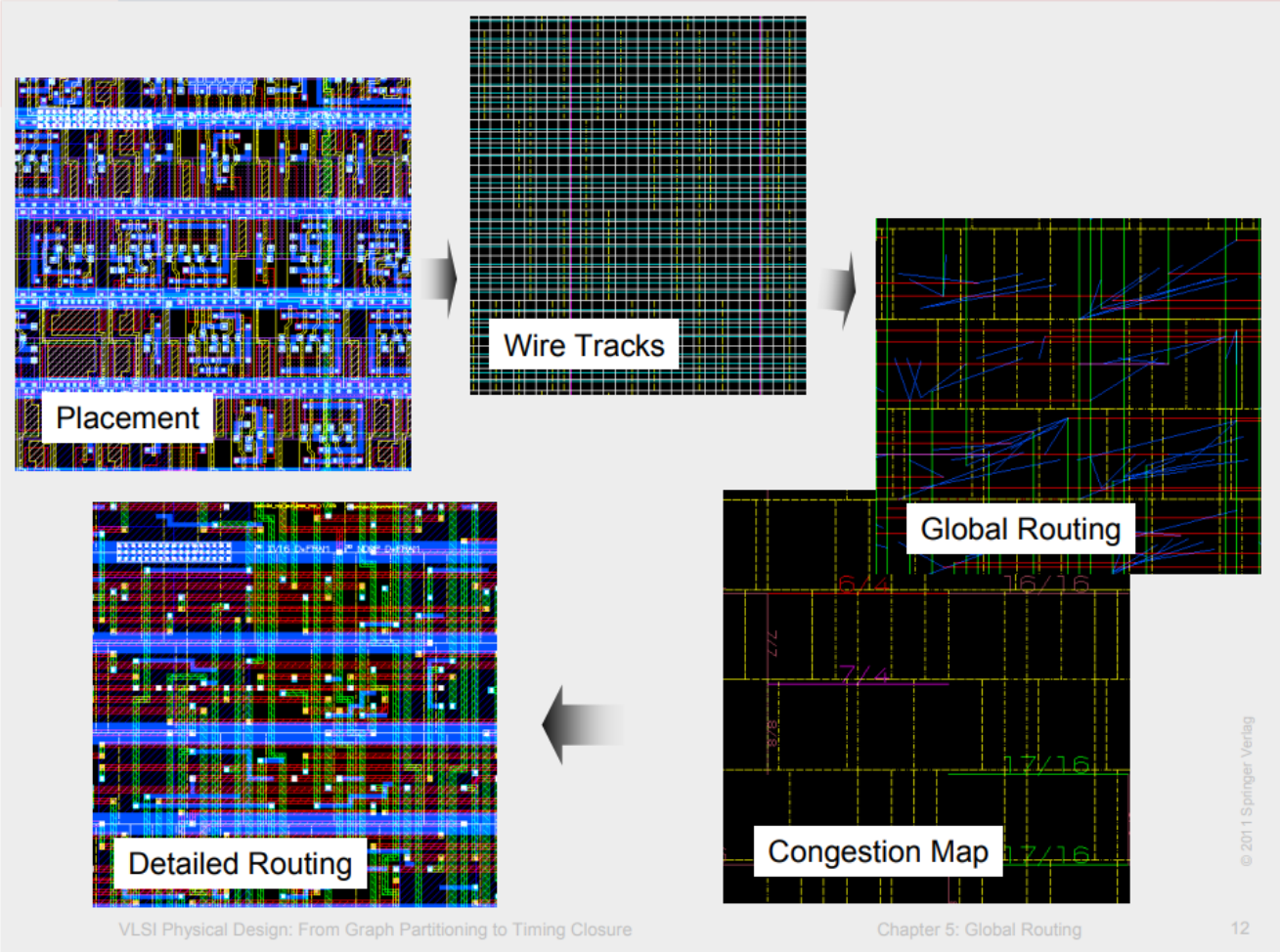
Types of routing:

Usually there are three types of routing available:

- Pre-Routing: Also knows as power routing which comes under power planning
- Clock Routing: It can be done while building Clock Tree in CTS stage
- Signal Routing: It is the stage after CTS

Stages of routing:

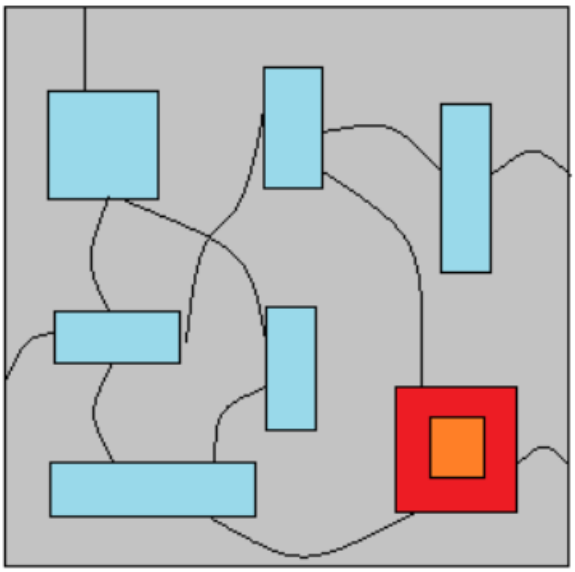
- **Global Routing**
- **Track assignment**
- **Detailed Routing**
- **Search and repair**



Global Routing:

In this stage, tool divides entire core area into global cells (gcells) and tries to find the shortest path from pin to pin as per the logical connectivity to route using gcells. While doing global routing tools tries to avoid routing over the routing blockages, and never touches the pre-routes (Power planning) and CTS nets.

- In placement stage tools identifies all routable path (scope of routing).
- Identifies shortest paths for the nets.
- For global routing tool doesn't consider DRCs (not DRC aware).
- In global routing it simply assigns metal layers to the nets.
- In this stage tool calculate congestion.
- For finding routing path tool majorly use Steiner tree and maze running algorithm.



Global Routing

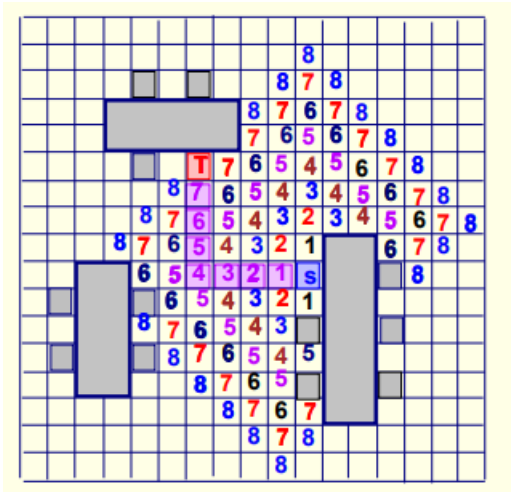


Fig: Maze routing algorithm

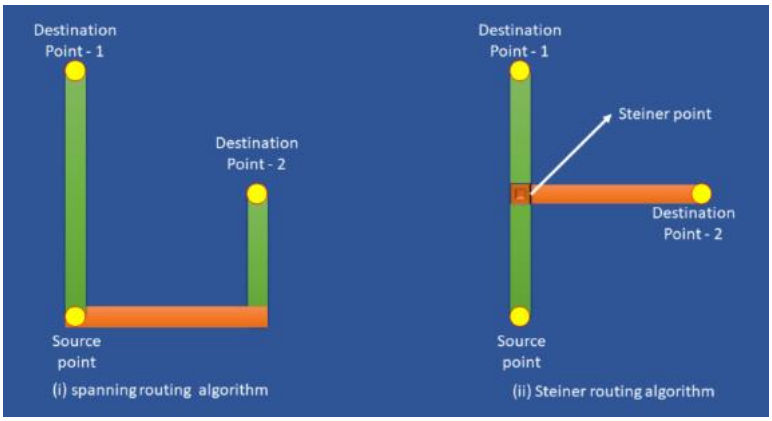


Fig: Steiner tree routing algorithm

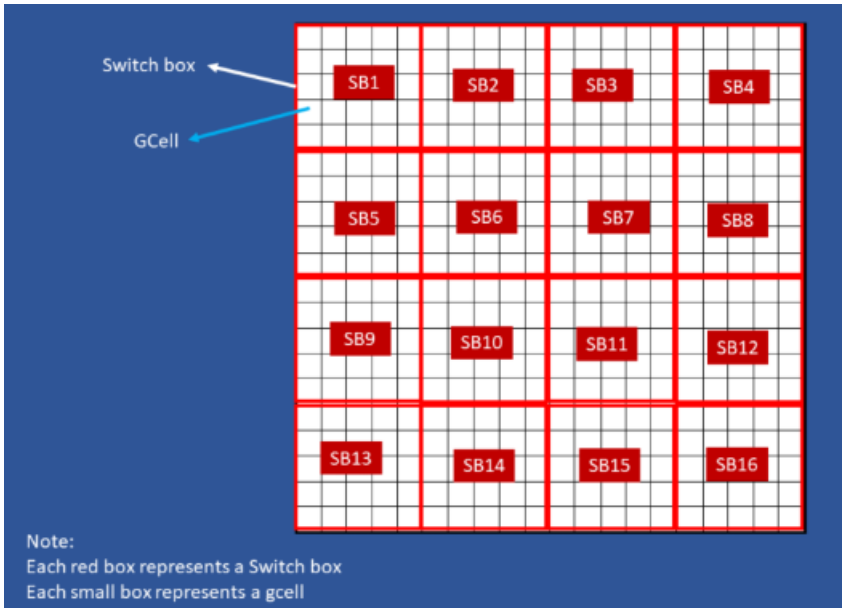
Track Assignment:

During this stage, tool replaces the global routes with actual metals, and those actual metals have to follow DRC rules. Now those actual metals will get real DRC violations, signal integrity (SI) and timing violations. Those violations will be fixed in the succeeding stages.

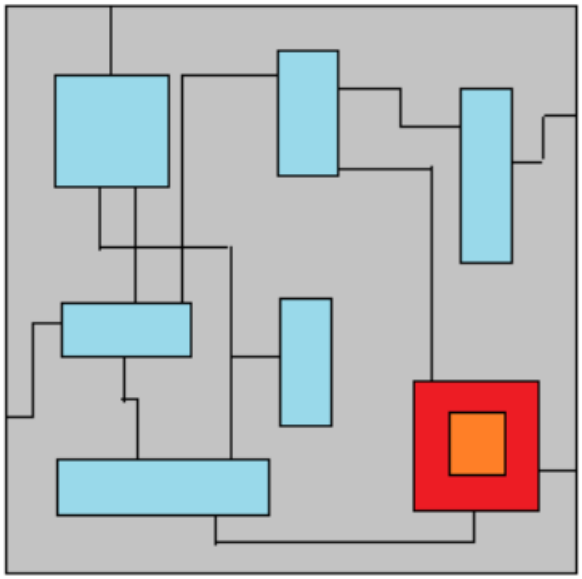
- Routing tracks are assigned for each global route.
- Tracks are assigned in vertical and horizontal direction.
- Replaces all global routes with actual metal layers.
- Rerouting all overlapped wires.
- There will be many DRC, Signal integrity and timing related violations.

Detailed Routing/Nano Routing:

During this stage, tool take care of routing while dealing DRC violations and improves the signal integrity(SI). The tool divides entire block into multiple switch boxes or sboxes. Each switch box will carry multiple gcells. The switch box boundaries are aligned to the gcells.



- It creates the actual via and metal connections.
- It specifies the specific tracks for the interconnection, each layer has its own routing grid, rules.
- Violations created during the track assignment stage are fixed in this stage.
- The main goal is to complete all of the required interconnect without leaving open/short/spacing violation while maintaining minimum total area, wire length and meet timing constraints.



Detail Routing

Detailed Routing Flow:

- **Track Assignment (TA)**
- **DRC fixing inside a Global Routing Cell (GRC)**
- **Iterate to achieve a solution (default ~20 iteration)**

Search and Repair:

- Performed during detailed routing after the first iteration.
- Locate all the violations and reroute the affected area to fix violations.

Timing-Driving Routing:

- Optimize critical paths
- Route critical nets first as most routing freedom at start.
- Use shortest paths possible
- Puts weighs on nets. Order of routing (priorities: e.g., Default Clock 50, others 2)
- Reduce resistance by widening the wire.
- For congested design need to set timing driven effort to low.

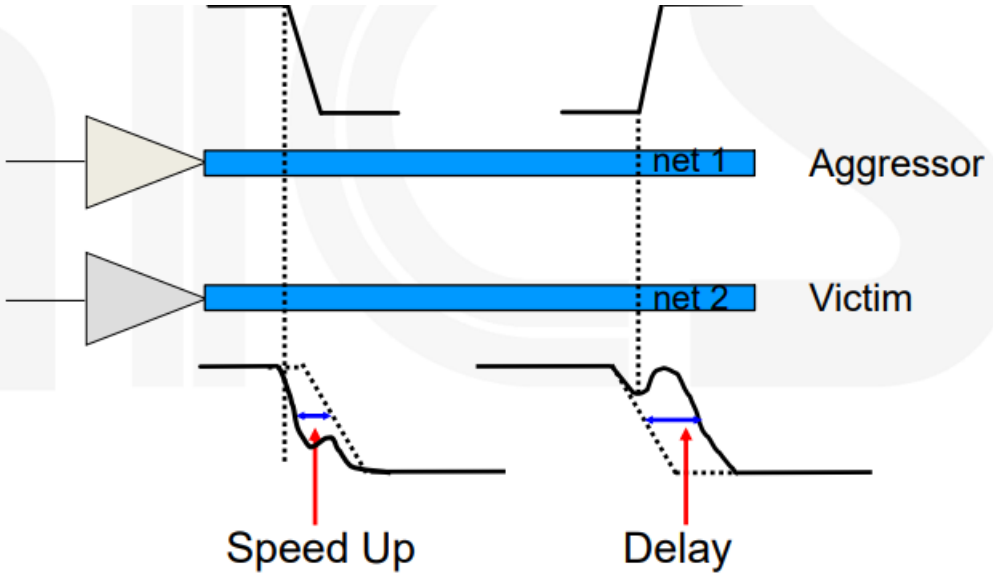
Signal Integrity (SI):

Signal Integrity during routing is synonymous with Crosstalk

- A switching signal may affect a neighboring net.
- The switching net is called the Aggressor.
- The affected net is called the Victim.

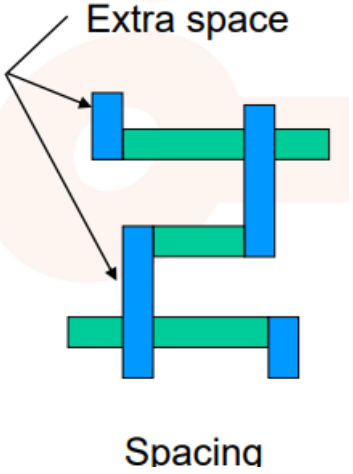
Two major effects duo to SI:

- **Signal slow down:**
  - When the aggressor and victim switch in opposite directions.
- **Signal speed up:**
  - When the aggressor and victim switch in the same direction.

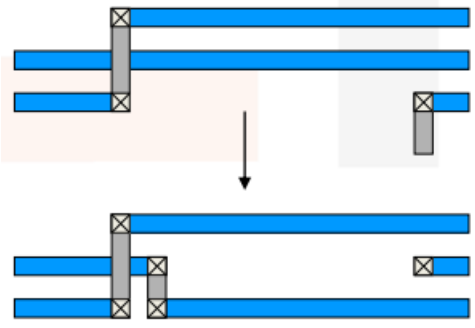


Signal Integrity (SI) Solutions:

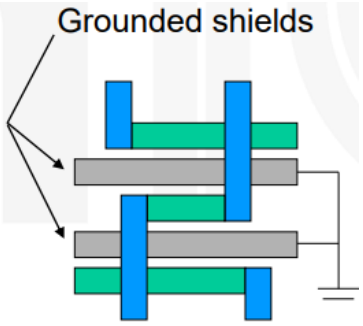
- **Limit length of parallel nets:**



- **Wire spreading:**

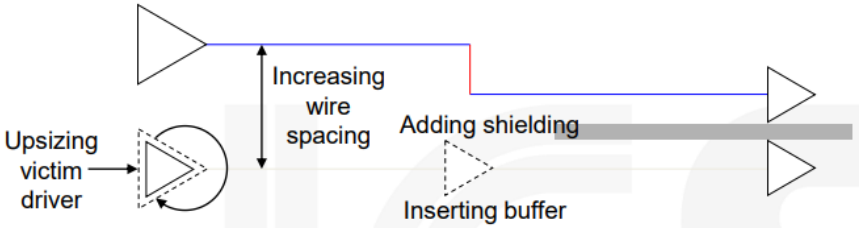


- **Shield special nets:**

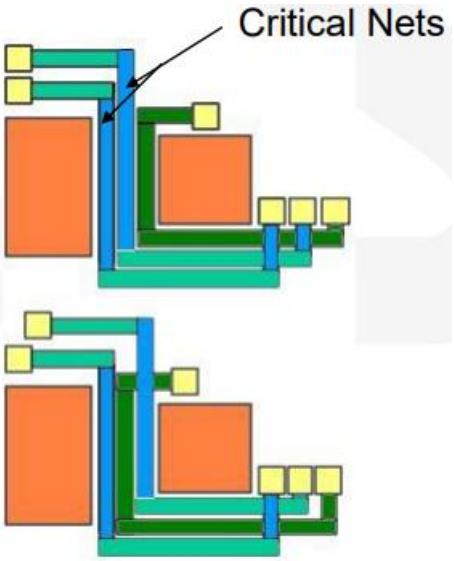


Shielding
Same layer (H)
Adjacent layers (V)

- **Upsize driver or buffer:**



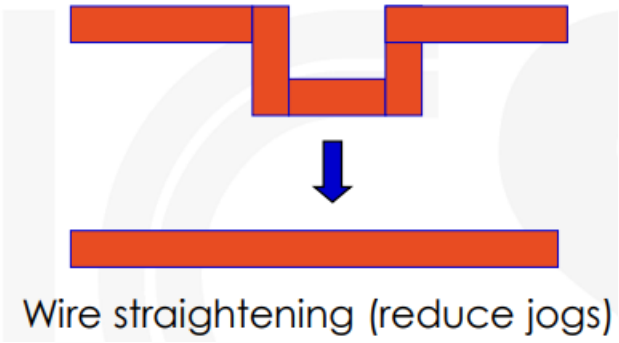
- **Net ordering:**

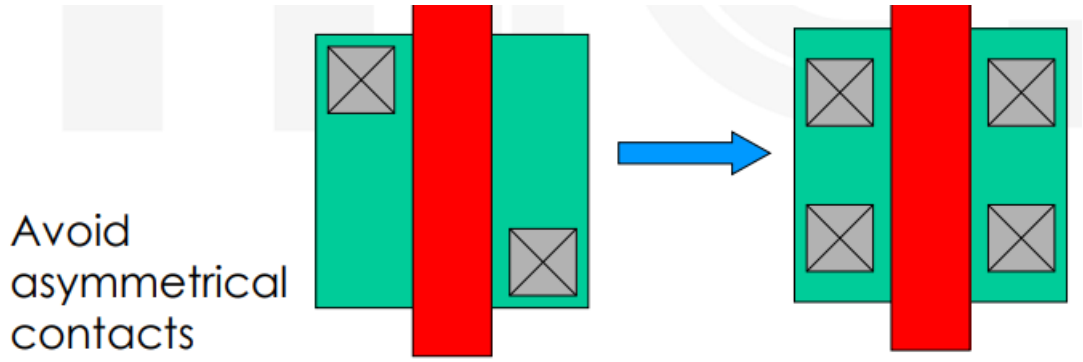


Design For Manufacturing (DFM):

During route, apply additional design for manufacturing (DFM) and/or design for yield (DFY) rules:

- Via reduction
- Redundant via insertion
- Wire straightening
- Wire spreading





DFM: Via Optimization:

Post-Route Via Optimization, includes:

- Incremental routing for the minimization of vias.
- Replacement of single vias with multi-cut vias.

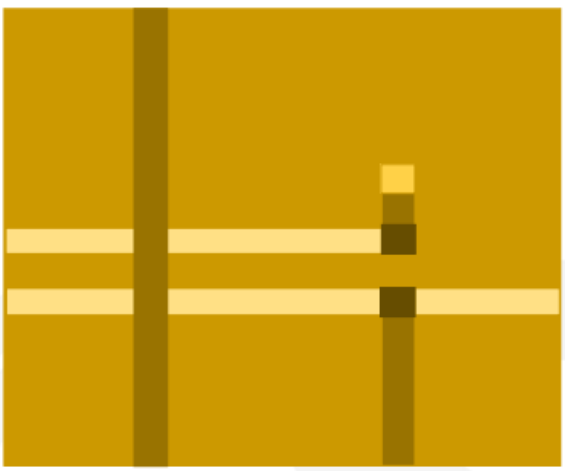
These operations are required for:

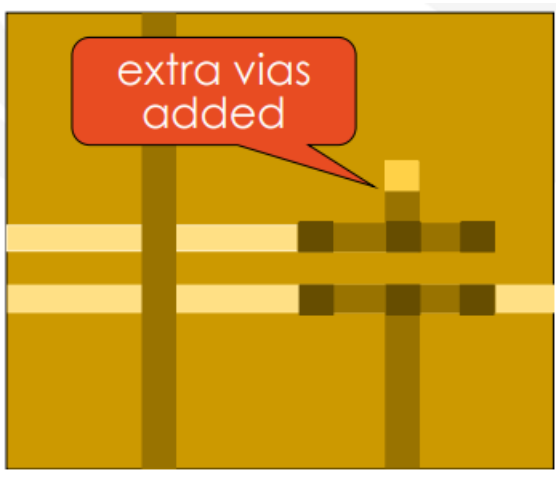
•Reliability:

- The ability to create reliable vias decreases with each process node. If a single via fails, it creates an open and the circuit is useless.

• Electromigration:

- Electromigration hazards are even more significant in vias, which are essentially long, narrow conductors.





DFM: Wire Spreading:

Wire spreading achieves:

- Lower capacitance and better signal integrity.
- Lower susceptibility to shorts or opens due to random particle defects.

Routing Command in different stages of innovus

Global routing in placement stage:
Command: route_early_global

Route in routing stage:
The commands for running a route with Nano Route are:
set_db route_design_with_timing_driven true
set_db route_design_with_si_driven true
route_design

Following detailed route wire optimization and timing optimization:
set_db route_design_with_timing_driven false
set_db route_design_detail_post_route_spread_wire true
set_db route_design_detail_use_multi_cut_via_effort high
route_design –wire_opt
set_db route_design_with_timing_driven true
opt_design -post_route -setup -hold

To achieve a high percentage of multi-cut vias:
set_db route_design_concurrent_minimize_via_count_effort high
set_db route_design_detail_use_multi_cut_via_effort high

To perform incremental routing (ECO routing):
set_db route_design_with_eco true
route_global_detail

To check the design after routing:
report_route // provide routing statistics
report_wires // provides wire statistics including wirelength
time_design –post_route //check timing after routing
check_drc // Run a DRC check

Different Command of routing stage for ICC2 tool:

Pre-route stage:
Create shield on the clock net to prevent cross talk:
create_shields $net_list
Optimize clock using the global route from the placement stage:

Clock_opt -from global_route_opt
Global route considering placement and CTS:
Route_global
Assign wire to the global route track defined in the technology:
Route_track

Route Stage:
Detail route:
Route_detail
Add multiple via (optional):
Add_redundant_vias

Post-route stage:
Remove routes with open or short connection:
Remove_route -detail_route -net $et)list
Route the removed net properly:
Route_eco
Optimize routing to solve DRC issues:
Route_detail -incremental true -initial_drc_from_input true
Optimize routing of the full design:
Route_detail -incremental true

Some Innovus routing attributes:

Detail route with timing optimization:
Route_design_with_timing_driven
Prevents crosstalk:
Route_design_with_si_driven
Spreading routing: (The router moves the wires without moving the vias)
Route_design_detail_post_route_spread_wire
Swaps single cut vias for multi-cut vias:
Route_design_detail_post_route_swap_via
Achieve high percentage of multi-cut vias:
Route_design_concurrent_minimize_via_count_effort
Route_design_detail_use_multi_cut_via_effort
Perform incremental routing:
Route_design_with_eco true
Avoid lithography problems during routing by avoiding certain routing patterns that might lead to the creation of lithography hotspots:
Route_design_with_litho_driven
Specifying iteration value for postroute optimization:
Route_design_detail_end_iteration
Specifies that pins are allowed as feedthrough:
Route_design_allow_pin_as_feedthru
Specifies the preferred trim metal grid for Nanorouterouter to use:
Route_design_with_trim_metal
Enclosed via geometries completely inside std cell pins:
Route_design_with_via_in_pin

Specifies the effort levels for adding via pillars:

Route_detail_post_route_via_piller_effort

Skip the prerouted nets:

set_route_attributes -nets @PREROUTED -skip_routing true

Route only the selected nets:

set_db route_design_selected_net_only true

Some usefull Innovus commands related to routing:

To Delete nets:

LUI : editDelete -net <net_name> -type Regular

CUI : delete_routes -net <net_name> -type Regular

To Select nets:

LUI : selectNet <net_name>

CUI: select_obj [get_db nets <net name>]

To delete routing for only those nets that have violations:

LUI : editDelete -net <*netName*> -regular_wire_with_drc

CUI: delete_routes -net <*netName*> -regular_wire_with_drc

To set top and bottom routing layer:

LUI: setDesignMode -bottomRoutingLayer <layer> and/or -topRoutingLayer <layer>

CUI: set_db design_bottom_routing_layer <layer>

set_db design_top_routing_layer <layer>

Routing blockages:

LUI: createRouteBlk -box {bbox coordinates list} -layer <layer>

CUI: create_route_blockage -rects {{x1 y1 x2 y2} ...} -layers <layer>

Shield nets with NanoRoute:

LUI: setAttribute -net <net_name> -shield_net <specialNetName> -shield_side <one_side/two_sides>

CUI: set_route_attributes -nets <net_name> -shield_nets <specialNetName> -shield_side <one_side/two_sides>

To shielding a net before nano-route:

LUI: setAttribute -net <net_name> -shield_net <specialNetName> -shield_side <one_side/two_sides>

CUI: set_route_attributes -nets <net_name> -shield_nets <specialNetName> -shield_side <one_side/two_sides>

To shield clock nets:

LUI: create_route_type -name <route_type_name> -shield_net <shield_net_name>
set_ccopt_property route_type <route_type_name> [-clock_tree <clock_tree_name>] [–net_type <top/trunk/leaf>]

CUI: create_route_type -name <route_type_name> -shield_net <shield_net_name>

Run standalone shielding.
LUI: createShield
CUI: route_add_shields

Reference:

Digital VLSI design lecture 9 Routing – DR. Adam Teman [digital_vlsi_design_lec_9_routing.pdf]
ROUTING - VLSI TALKS
Routing | vlsi4freshers
VLSI Physical Design: From Graph Partitioning to Timing Closure [VLSI Physical Design: From Graph Partitioning to Timing Closure]
How to delete and re-route specified nets with NanoRoute
Techniques for addressing the routing DRCs in Innovus
How to shield the nets with NanoRoute