Advanced VLSI

Bitcoin

Stage 6 : Routing

Ahmad foqara 322389982

Mohammad foqara 212193916

Neama hebi 212100705

Ward aboyonis 212021869

Work path: /project/advvlsi/users/ mohammadf1 /ws/bitcoin

Warmup Questions

[#P3.1_Q1] What is the purpose of the Routing stage in physical design?

In physical design, the routing step is essential for figuring out the connections between different parts or cells in an integrated circuit layout.

This phase entails:
1. Connectivity: Setting up particular wires or metal tracks to join various circuit components in accordance with the demands of the design.

2. Minimizing Area: In high-density systems, it is essential to reduce the area taken up by interconnections, which is made possible by effective routing.

3. Reducing Delay: By decreasing connection length and complexity, proper routing lowers signal latency.

4. Avoiding Congestion: By preventing the routing channels from packing an area too full, which may result in problems like higher resistance or capacitance, efficient routing helps to avoid congestion.

5. Compliance with Design rules: To guarantee functioning and manufacturability, routing must go by a number of design standards, including minimum width and spacing.

6. Power and Ground Distribution: In order to keep the circuit operating steadily, power and ground connections must also be distributed.

As a result, the routing stage is essential to the circuit's overall manufacturability, signal integrity, and intended electrical performance.

[#P3.1_Q2] What is the maze algorithm? How does it work?

In physical design, the maze algorithm is a frequently employed routing technique, especially when figuring out the best routes for interconnects on a device. It was initially created as a maze-solving algorithm that determines the optimal path between two places by examining potential pathways. Usually beginning with a grid that depicts the chip's routing region, the algorithm explores several routes using search methods such as Depth-First Search (DFS) or Breadth-First Search (BFS).

How the Maze Algorithm Works:

1. Grid Representation: The routing region of the chip is separated into a grid, with each cell denoting a potential wire position. Within this grid, the algorithm determines the source and destination points.

2. Path Exploration: From the source to the destination, the algorithm investigates potential routes. To locate suitable routes, it can employ a variety of search strategies, including DFS, BFS, and more complex techniques like the A* algorithm. These techniques aid in navigating the grid and avoiding impediments.

3. Heuristics and Optimization: Heuristics can be used to direct the search in complex designs, assisting in the discovery of routes that minimize wirelength, lessen congestion, and comply with design specifications. Because it prioritizes the most promising paths by combining the features of DFS and BFS with a cost function, A* is very helpful in this situation.

4. Constraints consideration: The program assesses every path in light of design guidelines and limitations including signal integrity, wirelength, and congestion. To make sure the routing paths satisfy these requirements, it iteratively investigates and modifies them.

5. Final Routing Solution: In physical design, the maze algorithm seeks to provide a routing solution that minimizes wirelength, lowers congestion, and preserves signal integrity while meeting design requirements. The routing for that interconnect is completed once a satisfactory or ideal path has been identified.

Applications: Physical Design in VLSI: To ensure effective and dependable communication between various components, the labyrinth method is frequently utilized in VLSI (Very-Large-Scale Integration) physical design for routing interconnects on chips.

[#P3.1_Q3] What is Cross talk? How can we minimize its effect?

Unwanted signal coupling between neighboring wires or interconnects, known as crosstalk, results in interference and deterioration of signal quality. It happens when electrical impulses on one wire change and cause voltages or currents to flow through adjacent wires. This can cause problems including noise, fluctuating delays, and inaccurate data interpretation. Crosstalk is a major issue in many domains, such as electronics, signal processing, and telecommunications. It can result from capacitive, inductive, or conductive connection between circuits.

Methods for Reducing Crosstalk:

1. Appropriate Routing and Spacing: Maintaining adequate distance between neighboring wires lowers crosstalk and coupling capacitance. Crosstalk effects can be further reduced by using thicker and broader wires.

2. Shielding and Isolation: Sensitive wires can be isolated from noise sources by sandwiching ground planes or shielding layers between them, which lowers capacitive and inductive crosstalk.

3. Crosstalk-Aware Routing: The coupling between neighboring wires can be reduced by using routing algorithms that take crosstalk constraints into account during the design phase.

4. Twisted-Pair Cables: By balancing out induced voltages from external magnetic fields, twisting wire pairs together can lessen inductive crosstalk.

5. Signal Encoding and Synchronization: You can reduce the effects of crosstalk and increase noise immunity by putting signal encoding systems like differential signaling or clock synchronization techniques into practice.

6. Appropriate Power and Ground Distribution: By lowering noise and voltage swings, a strong and evenly distributed power and ground network also lessens crosstalk.

7. Impedance Matching: Reflections and, hence, crosstalk can be minimized by making sure the impedance of the transmission line and the load impedance match.

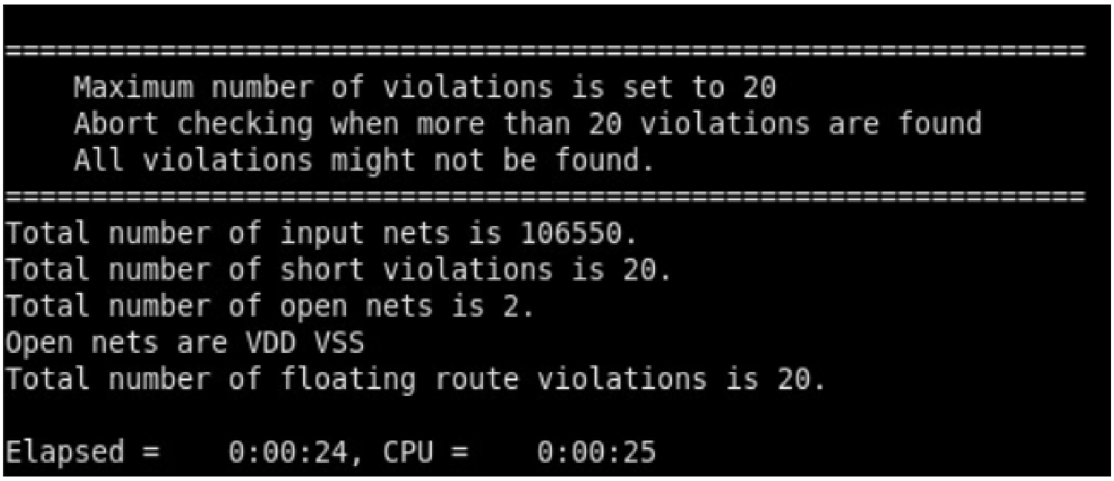
8. Design Rule Checks (DRC): During the physical design phase, comprehensive

DRC checks can detect possible crosstalk infractions and enable the required modifications to lessen the impact.

Crosstalk can be successfully reduced by combining these techniques, improving signal integrity and system performance in general.

[#P3.2_Q1] How many Short and opens were found in our design?

We found 20 violations, and two open circuits .



```
===============================================================
    Maximum number of violations is set to 20
    Abort checking when more than 20 violations are found
    All violations might not be found.
===============================================================
Total number of input nets is 106550.
Total number of short violations is 20.
Total number of open nets is 2.
Open nets are VDD VSS
Total number of floating route violations is 20.

Elapsed =    0:00:24, CPU =    0:00:25
```

[#P3.2_Q2] What would be your recommendations for improving the current implementation?

1. Intelligent Congestion Management: Create Strategies: To successfully manage routing congestion, put advanced congestion management strategies into practice. In order to prevent bottlenecks and guarantee continuous flow, this involves anticipating high-density regions and optimizing routing routes.
Utilize routing algorithms that actively manage traffic by modifying routes in response to real-time congestion data. This is known as congestion-aware routing.

2. Make Use of Design Reusability
Encourage reusability To expedite the design process, make use of Intellectual Property (IP) libraries or pre-made blocks. This method keeps designs consistent, cuts down on redundancy, and speeds up development.
Integration of IP Libraries: Use reusable design blocks to ensure that design rules are followed while enabling faster and more effective routing.

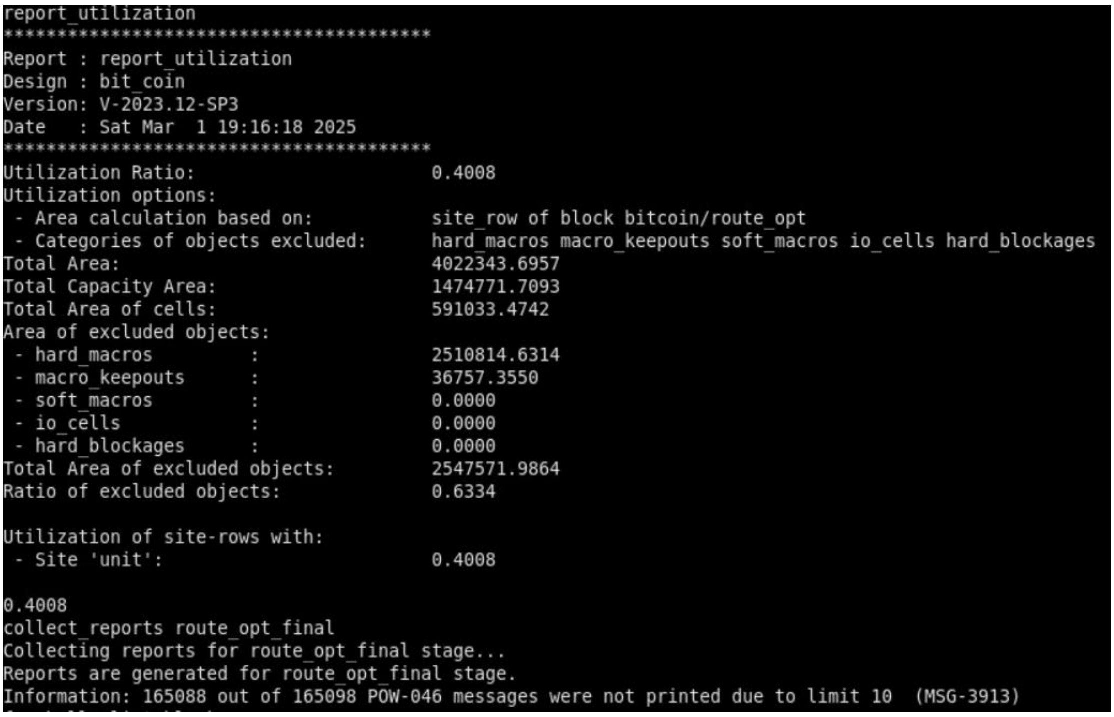
3. Ongoing Analysis and Improvement:
Frequent Analysis: To pinpoint areas that require improvement, regularly examine routing results. To improve routing algorithms and adjust to evolving design restrictions, use feedback loops and iterative reviews.
Iterative Refinement: To improve routing effectiveness and handle new issues, do frequent design reviews.

4. Power-Aware Routing: Reduce Power Usage: Utilize power-aware routing strategies to optimize wire lengths, lower capacitance, and control voltage drop limitations in order to reduce power usage.

Power Optimization Strategies: To guarantee effective power consumption and signal integrity, use strategies that take power limitations into consideration during routing.

5. Routing with multiple layers:
Efficient Use of Layers: To increase signal separation, shielding, and interference reduction, use several routing layers. For best results, allocate high-speed nets or important signals to specific layers.
Layer Assignment: To improve routing efficiency and preserve signal integrity, strategically assign layers according to design constraints and signal requirements.

To sum up, you may greatly improve the labyrinth routing algorithms in VLSI physical design by implementing intelligent congestion management, leveraging design reusability, concentrating on continuous development, optimizing for power, and effectively utilizing multi-layer routing. A more reliable and effective routing solution is the result of these tactics, which guarantee improved performance, less congestion, and effective power management.



```
report_utilization
****************************************
Report : report_utilization
Design : bit_coin
Version: V-2023.12-SP3
Date   : Sat Mar  1 19:16:18 2025
****************************************
Utilization Ratio:                      0.4008
Utilization options:
 - Area calculation based on:           site_row of block bitcoin/route_opt
 - Categories of objects excluded:      hard_macros macro_keepouts soft_macros io_cells hard_blockages
Total Area:                             4022343.6957
Total Capacity Area:                    1474771.7093
Total Area of cells:                    591033.4742
Area of excluded objects:
 - hard_macros        :                 2510814.6314
 - macro_keepouts     :                 36757.3550
 - soft_macros        :                 0.0000
 - io_cells           :                 0.0000
 - hard_blockages     :                 0.0000
Total Area of excluded objects:         2547571.9864
Ratio of excluded objects:              0.6334

Utilization of site-rows with:
 - Site 'unit':                         0.4008

0.4008
collect_reports route_opt_final
Collecting reports for route_opt_final stage...
Reports are generated for route_opt_final stage.
Information: 165088 out of 165098 POW-046 messages were not printed due to limit 10  (MSG-3913)
```