Stage 6 – Routing

Advvlsi_02

Workarea path: /project/advvlsi/users/adamsalameh/ws/bitcoin/stage6

Adam Salameh 207534538 adamsalameh@mail.tau.ac.il

Nawaf Gadban 314702705 nawafgadban@mail.tau.ac.il

Anan Break 322663170 ananbreak@mail.tau.ac.il

Sara Abo Salah 208298562 saraabosalah23@mail.tau.ac.il

Lama Metanis 314857152 Lamametanis@mail.tau.ac.il

Warmup Questions:

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

The routing stage in physical design for integrated circuits determines the paths of electrical interconnects between components on the chip. It aims to optimize timing, ensure signal integrity, maximize area utilization, comply with design rules, and optimize power consumption. This critical step forms the network of wires that facilitate efficient communication and signal flow within the chip.

The routing stage comes after the placement stage, where the positions of various circuit components are determined. Once the components are placed, the routing stage is responsible for establishing the connections between them. These connections form the intricate network of wires that enable communication and signal flow within the chip.

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

The maze algorithm, also known as maze routing, is a popular routing algorithm used in the routing stage of physical design for integrated circuits. It is employed to determine the optimal paths for interconnects between components on a chip, navigating through obstacles and efficiently utilizing available routing resources.

The maze algorithm works as follows:

Grid Representation: The chip area is divided into a grid of cells, where each cell represents a potential routing track or segment. The cells are interconnected vertically and horizontally, forming a maze-like structure.

Source and Destination: The algorithm identifies the source and destination for the interconnect that needs to be routed. These correspond to the starting and ending points for the route.

Maze Creation: Initially, the entire grid is marked as "unvisited" or "free." However, obstacles such as pre-existing routes, components, or design constraints are marked as "blocked" or "occupied" cells.

Path Exploration: The algorithm begins exploring possible routes from the source cell. It searches for a feasible path by navigating through the maze, considering the neighboring cells and their connectivity.

Path Selection: The algorithm employs various strategies to determine the best path. It may use heuristics, such as minimizing wirelength, reducing congestion, or considering timing constraints. Different algorithms, such as Lee's algorithm or A* search, can be utilized for path selection.

Path Expansion: As the algorithm progresses, it expands the path by marking the cells as "visited" or "occupied" to avoid overlapping routes. It continues exploring and expanding until it reaches the destination cell.

Backtracking and Optimization: In case the algorithm encounters dead-ends or obstacles that prevent a direct route, it performs backtracking to explore alternative paths. This process continues until an optimal or acceptable path is found.

Completion and Output: Once the algorithm successfully connects the source and destination, the resulting path is output as the final routed interconnect. The routing information is then passed on to subsequent stages of physical design.

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

Crosstalk refers to unwanted signal coupling between adjacent wires in a circuit, causing signal distortion and noise interference. To minimize crosstalk, proper routing and spacing of wires, shielding and grounding techniques, twisted pair or differential signaling, proper termination, signal isolation, and simulation and analysis can be utilized. These approaches reduce the proximity between wires, cancel out electromagnetic interference, match impedance, isolate signals, and identify and address crosstalk issues during the design process. By implementing these measures, the impact of crosstalk can be minimized, ensuring reliable signal transmission in the circuit.

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

TOTAL NUMBER OF SHORTS:20

TOTAL NUMBER OF OPENS:2

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

To improve the implementation of our design:

- Intelligent Congestion Management: Develop intelligent congestion management techniques to handle routing congestion.
- Utilize Design Reusability: Promote design reusability by creating and utilizing pre-designed blocks or Intellectual Property (IP) libraries.
- Continuous Improvement and Analysis: Regularly analyze routing results and identify areas for improvement.
- Power-Aware Routing: Consider power optimization during routing. Implement power-aware routing techniques that minimize power consumption by optimizing wire lengths, reducing capacitance, and considering voltage drop constraints.
- Multi-Layer Routing: Make effective use of multiple routing layers. Assign critical signals or high-speed nets to dedicated routing layers to provide better separation, shielding, and reduced interference.