Advanced VLSI Project
Bitcoin

Stage 4

Placement

שמות המגישים	אור שאול	עמרי אורן	סטאניסלאב קוגן	נועה פרקש
תעודת זהות	206491920	213409972	209379965	211820899

Working Path: /project/advvlsi/users/stanislavk2/ws/stage4

3.1 Warmup Questions- Placement:

[#P3.1_Q1]

The main differences between Floorplanning and Placement:

1. **Execution:** Floorplanning comes before Placement in the physical design work flow.

2. **Detail depth:** Floorplanning is more abstract level of design working on arrangement of blocks and I/O pads. On the other hand, placement operates on more deep level and arranging the individual cells within the blocks.

3. **Goals:** The main goal of the Floorplanning stage is to optimize the layout, determine the size and location of blocks and minimize the interconnect length. Placement stage goal is to minimize the total wire length connections between the individual cells with taking into account the timing, power and area constraints.

4. **Routing type:** During Floorplanning global routing paths between blocks are estimated, in the Placement stage cells are assigned specific locations in the blocks and local routing path is established.

5. **Algorithm:** Floorplanning algorithm uses methods of simulated annealing/ genetic algorithms to find an optimal placement of blocks. Placement algorithm uses techniques of force directed placement, partitioning and simulated annealing to optimize cell locations.

6.**Outputs:** The Floorplanning stage outputs a rough sketch of where each block and I/O pad will be placed, along with estimated routing paths between blocks. The placement stage outputs are more detailed layout, providing the exact location for each cell within the blocks, and the local routing paths within the blocks.

[#P3.1_Q2]

1. **Partitioning Algorithms:** Dividing the circuit into smaller parts or partitions, then place the cells in these partitions to reduce interconnect length.

2. **Simulated Annealing:** Probabilistic technique used for finding an approximated solution to an optimization problem. The first step is initial solution then, explores the solution space by making random changes, and probabilistically accepts changes that worsen the solution to escape local optima.

3. **Genetic Algorithms:** These are inspired by the process of natural selection and use techniques such as mutation and crossover to explore the solution space. Each potential solution (placement of cells) is considered as an individual in the population, and individuals are selected for reproduction to create offspring for the next generation based on their fitness (the objective function value).

4. **Analytical Placement:** Formulation of the placement problem as a mathematical

optimization problem. The objective function typically includes the wirelength and a density penalty to prevent overlaps.

5. **Force-Directed Placement:** The algorithm seeks a placement where the sum of all forces is minimized. Cells are modeled as objects exerting forces on each other. The 'force' is a function of the interconnect length or delay.

[#P3.1_Q3]

Simulated annealing is a probabilistic optimization technique used in the placement stage. The first stage is initial placement. The second stage is randomly swapping the positions of two cells. The logic is: If the new placement improves the objective (like reducing total wirelength), it's accepted. Else If it worsens the objective, it's accepted with a probability that decreases over time. The process is repeated until a termination condition is met.

3.3 Warmup Questions- Advanced STA:

[#P3.3_Q1]

When choosing between setup and hold timing violation in terms of which is more critical we often consider Setup violation more crucial. Setup violations happened when data does not arrive at a flip-flop before the clock edge, which can lead to incorrect data being latched. From the critical path that we get from the setup timing report we can limit the maximum operating frequency of the chip.
Hold violations, on the other hand, happened when data changes too soon after the clock edge, potentially causing incorrect data to be latched. However, with the advances in technology, hold violations have become less common. This is due to the fact that interconnect delays have become a significant portion of the total delay, which tends to increase the clock-to-Q delay of flip-flops, making hold violations less likely.

[#P3.3_Q2]

1. **Clock Tree Synthesis (CTS) Optimization:** As we learned in the course by making adjustments in the clock tree, we can reduce skew. This ensures that clock signals arrive at their destinations more synchronously, which can help alleviate setup violations.

2. **Optimizing Logic:** When setup violation happens we can look for additional optimizations of the logic. Reducing the logic levels on the violating path or restructuring the logic to achieve the same functionality with less delay can help.

3. **Constraints Adjusting:** As we saw in stage 1 we can adjust false paths and multi-cycle paths, which directly affect the timing analysis and optimization also, we can revisit and tune the timing constraints to be more accurate.

[#P3.3_Q3]

Hold violations can be created when the data signal at the input of a flip-flop changes too soon after the active edge of the clock. The data needs to stay stable for a required Hold time after the clock edge. Hold violations can be created from Clock skew, If the clock arrives at the data flip-flop later than at the capture flip-flop, it can cause the data signal to change before the required hold time has passed. Hold violations can happen also from the working

environment conditions like changes in voltages, manufacturing process that can affect the tpd time of the gates and cause hold violations. Another reason can occur from changes in the design in the placement or routing and providing solutions for setup violations can create Hold violations.

[#P3.3_Q4]

1. **Logic Optimization:** We learned in the course that adding buffers or extra logic cells in the data path can help increase the delay, ensuring data doesn't change too early.

2. **CTS:** Adjusting the clock tree to balance skew can help in synchronizing the data and clock signals better, alleviating hold violations.

3. **Gates Downsizing:** Downsizing a gate increases its delay, which can help eliminate hold violations this can be done by replacing faster cells in the design with slower ones, but this process can create setup violations so we need to be careful.

[#P3.4_Q4]

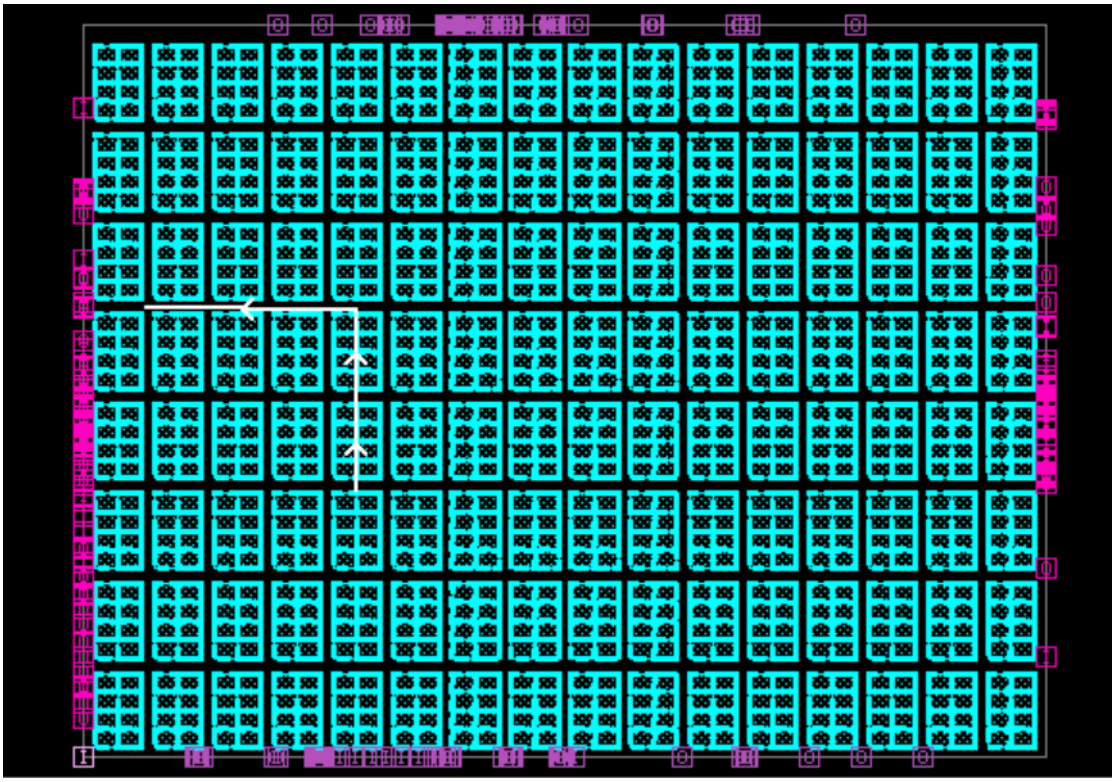
First problematic path:

```
Startpoint: bit_secure_0/sipo_slice_first/wr_tmp_reg (rising edge-triggered flip-flop clocked by lclk)
Endpoint: bit_secure_6/sipo_slice_first/wr_0_reg (rising edge-triggered flip-flop clocked by lclk)
Mode: FUNC
Corner: Typical
Scenario: FUNC_Typical
Path Group: lclk
Path Type: max

Point                                              Incr       Path
-----------------------------------------------------------------------------
clock lclk (rise edge)                             0.00       0.00
clock network delay (ideal)                        0.00       0.00

bit_secure_0/sipo_slice_first/wr_tmp_reg/CLK (SDFFASX1_RVT)
                                                   0.00       0.00 r
bit_secure_0/sipo_slice_first/wr_tmp_reg/QN (SDFFASX1_RVT)
                                                   0.13       0.13 r
place_optHFSBUF_400_89235/Y (NBUFFX8_RVT)          0.08       0.21 r
HFSINV_3087_47138/Y (INVX0_RVT)                    0.05       0.26 f
place_optHFSBUF_465_93191/Y (NBUFFX16_RVT)         0.07       0.33 f

HFSINV_375_47133/Y (INVX0_RVT)                     0.11       0.43 r
place_optHFSBUF_47_87670/Y (NBUFFX8_RVT)           0.08       0.51 r
bit_secure_6/sipo_slice_first/wr_0_reg/D (SDFFARX1_RVT)
                                                   0.02       0.53 r
data arrival time                                             0.53

clock lclk (rise edge)                             0.60       0.60
clock network delay (ideal)                        0.00       0.60
bit_secure_6/sipo_slice_first/wr_0_reg/CLK (SDFFARX1_RVT)
                                                   0.00       0.60 r
library setup time                                -0.12       0.48
data required time                                            0.48
-----------------------------------------------------------------------------
data required time                                            0.48
data arrival time                                            -0.53
-----------------------------------------------------------------------------
slack (VIOLATED)                                             -0.05
```

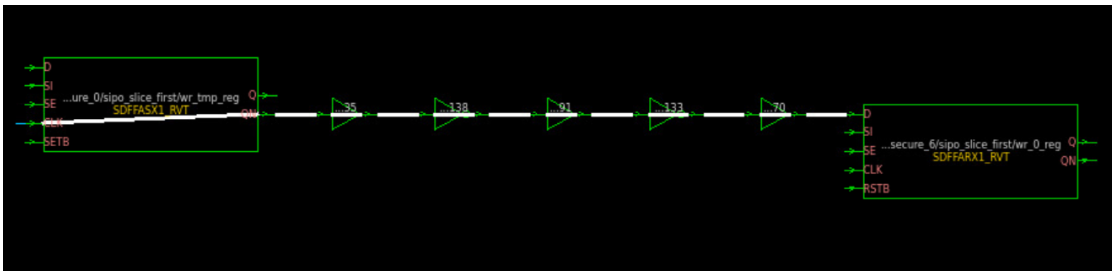
3.1

The report presents a Setup violation due to path type max which represents setup timing test.

3.3



3.4



3.5

We have setup time violation because the data arrival time (0.53 ns) is later than the required time (0.48 ns) at the destination flip-flop, resulting in negative slack of -0.05 ns.

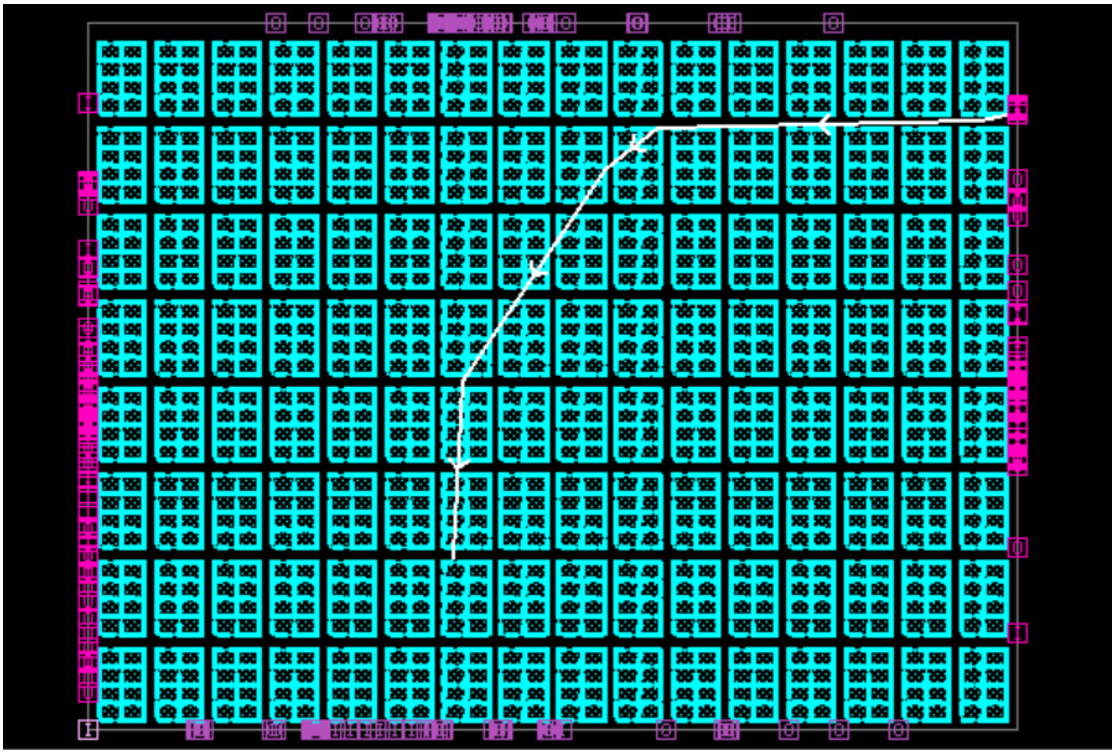
HFSINV_375_47133/Y(INVX0_RVT) is the cell with the highest delay in the path (0.11 ns).

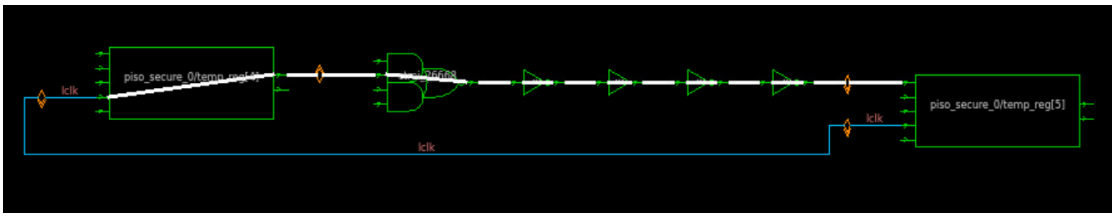
There are 8 cells in the path between the start and end points.

we can learn that The delays of the cells add up to create a delay in the entire path, causing the data arrival at the destination flip-flop to be later than required, leading to a setup violation.

Second path: (Setup)

```
Startpoint: piso_secure_0/temp_reg[4] (rising edge-triggered flip-flop clocked by lclk)
Endpoint: piso_secure_0/temp_reg[5] (rising edge-triggered flip-flop clocked by lclk)
Mode: FUNC
Corner: Typical
Scenario: FUNC_Typical
Path Group: lclk
Path Type: max

Point                                              Incr      Path
-----------------------------------------------------------------------
clock lclk (rise edge)                             0.00      0.00
clock network delay (ideal)                        0.00      0.00

piso_secure_0/temp_reg[4]/CLK (SDFFARX1_RVT)       0.00      0.00 r
piso_secure_0/temp_reg[4]/Q (SDFFARX1_RVT)         0.14      0.14 r
ctmi_26668/Y (AOI22X2_RVT)                         0.11      0.26 f
place_optHFSINV_8075_83981/Y (IBUFFX16_RVT)        0.09      0.35 r
place_optHFSBUF_7552_83980/Y (NBUFFX4_RVT)         0.12      0.47 r
place_optHFSINV_7515_83979/Y (IBUFFX16_RVT)        0.09      0.56 f

place_optHFSINV_2914_83978/Y (INVX8_RVT)           0.12      0.68 r
piso_secure_0/temp_reg[5]/D (SDFFARX1_RVT)         0.02      0.70 r
data arrival time                                            0.70

clock lclk (rise edge)                             0.60      0.60
clock network delay (ideal)                        0.00      0.60
piso_secure_0/temp_reg[5]/CLK (SDFFARX1_RVT)       0.00      0.60 r
library setup time                                -0.12      0.48
data required time                                           0.48
-----------------------------------------------------------------------
data required time                                           0.48

data required time                                          0.48
data arrival time                                          -0.70
-----------------------------------------------------------------------
slack (VIOLATED)                                           -0.23
```





3.5

We have setup time violation because the data arrival time (0.70 ns) is later than the required time (0.48 ns) at the destination flip-flop, resulting in negative slack of -0.23 ns.

HFSBUF_7552_83980/Y and HFSINV_2914_83978/Y are the cell with the highest delay in the path (0.12 ns).

There are 8 cells in the path between the start and end points.

we can learn that The delays of the cells add up to create a delay in the entire path, causing the data arrival at the destination flip-flop to be later than required, leading to a setup violation.

3.6 (for both paths):

There are several ways to fix the setup violation:

1. **decreasing the clock rate**, the timing requirements can be fixed by allowing more time for the signal to propagate through the path and reducing the option for setup violations.

2.**Adding an additional register** in the path creates an additional pipeline stage, breaking down the critical path into smaller segments and providing more time for the signal to reach the destination flip-flop.