Advanced VLSI Project – stage 4

Table of contents

1. Reminder

So far, we have covered three stages:

- Synthesis stage – converting our RTL into logical cells
- STA stage – analyzing the timing constraints of our design.
- Floorplan – planning how we would want our chip to be constructed in a general view.

In the current stage (Placement) we want to set every cell in its place while holding all the constraints including our general guidelines from the floorplan stage.

In this assignment we will also "turn back" to the STA stage with some practical practice of analyzing timing reports and handing in the problem we will encounter.

2. Setup

As we explained in previous stages of the project, we would like to copy the files for stage 4 from Elad's directory using the command:

```
cp -r /project/advvlsi/users/eladsimanian/ws/bitcoin/stage4 .
```

As explained in stage 3 document, you will work with a given NDM directories in this section too. For additional instruction on how to work with NDM directories please revisit the previous stage.

In this section we will work in two different contexts:

- Continue the flow of our chip implementation (Placement) – working with "bitcoin_stage_4.dlib" and the given TCL script.
- Analyzing timing report and fixing specific part of that design – working with "bitcoin_timing_1.dlib" (see further explanation below).

3. Stage 4: Placement

The Placement stage is a crucial step in the physical design of our chip. Once we have a general framework for our design in the Floorplan stage, the Placement stage involves placing the individual components – mainly the standard cells in such a way that would meet any design constraints, such as the timing, power, and area requirements of the design. The algorithm should try to achieve the best results within the allocated time (governed by number of iterations).

For your comfortable we marked all tasks needed for submission with:
[#P<Part_ID>_Q<Q_ID>] such as **[#P3.1_Q1]**

3.1 Warmup Questions - Placement

1. **[#P3.1_Q1]** What is the difference between Placement and Floorplanning in physical design?
2. **[#P3.1_Q2]** What are some of the common techniques used in Placement optimization?
3. **[#P3.1_Q3]** What is simulated annealing in the context of the placement stage? explain how the algorithm works, what are the hyperparameters included?

3.2 Placement implementation

1. Open the NDM lib “bitcoin_stage_4” (see section “Working with NDM libraries” in previous project document for details).
   This lib contains a similar snapshot of our design from the floorplan stage option 3 (un-fixed channel widths and heights).
2. Run the commands one by one in the given script.

3.3 Warmup Questions – Advanced STA

1.1. **[#P3.3_Q1]** Which type of timing violation can be more critical – setup or hold?
1.2. **[#P3.3_Q2]** Suggest 3 ways to fix a setup violation.
1.3. **[#P3.3_Q3]** How can a hold violation be created in a digital design.
1.4. **[#P3.3_Q4]** Suggest 3 ways to fix a hold violation.

3.4 Advanced STA

1. In Elad's directory you will find another tcl script named: "bitcoin_stage_4_sta.tcl". Run the commands one by one in the given script.
2. **[#P3.4_Q1]** Analyze the timing reports extracted and find 2 problematic paths. For each path:
   3.1. Note the path type (setup / hold).
   3.2. Add a screenshot of the report [Tip: get the screen shots from shell window and not from the gui window for better scaling]
   3.3. Add a screen shot of the path over the layout. Use the command from stage 1:
   - The -from and -to refer to the startpoint and endpoint of that timing path.

```
change_selection [get_timing_paths -from XXX -to XXX]
```

   3.4. Add a screen shot of the path as a schematic view.
   3.5. Explain the main problem of the path that causes the negative slack (Hints: which cells have the highest delay? How many cells are in the path? What can be inferred from the screenshots above?)
   3.6. Suggest 1-2 ways to fix the timing path.

4. Submission

1. If the team approves the design functionality, the team lead, which has the updated work area with all the teams' files working, should submit the assignment as follows.
2. Submit to Moodle a zip:
   Let's say you are group advvlsi_15 then:
   bitcoin_stage4_advvlsi_15.zip

   The zip must include a PDF file:
   bitcoin_stage4_advvlsi_15.pdf

   the pdf should include the following:
   2.1. The stage name and number (Stage 4 – Placement)
   2.2. Workarea path, such as: /project/advvlsi/users/$user/ws/bitcoin_project
   2.3. Team's info: members name, id, usernames (in micron servers), tau emails
   2.4. Answers for all questions and tasks marked with "[#P<>Q<>]"

Good Luck.

P.S the next stage will deal with CTS (Clock Tree Synthesis)!

Questions only

[#P3.3_Q1] Which type of timing violation can be more critical – setup or hold?

[#P3.1_Q2] What are some of the common techniques used in Placement optimization?

[#P3.1_Q3] What is simulated annealing in the context of the placement stage? explain how the algorithm works, what are the hyperparameters included?

[#P3.3_Q1] Which type of timing violation is worst – setup or hold?

[#P3.3_Q2] Suggest 3 ways to fix a setup violation?

[#P3.3_Q3] How can hold violation be created in a digital design?

[#P3.3_Q4] Suggest 3 ways to fix a hold violation?

[#P3.4_Q1] Analyze the timing reports extracted and find 2 problematic paths. For each path:

- Note the path type (setup / hold).
- Add a screenshot of the report [Tip: get from shell and not from gui for better scaling]
- Add a screen shot of the path over the layout. Use the command from stage 1:

```
change_selection [get_timing_paths -from XXX -to XXX]
```

- Add a screen shot of the path as a schematic view.
- Explain the main problem of the path that causes the negative slack (which cell has the highest delay? How many cells are in the path?)
- Suggest 1-2 ways to fix the timing path.