VDF Group Project: Part I

Objective

The objective of this project is to gain a hands-on experience on the state-of-the-art CAD tools and understanding various trade-offs that are involved in design and verification of simple digital circuits. The focus should not only be on running of the tools, but also on analysis/interpretation of the results.

Read the following instructions carefully. Marks may be deducted if the instructions are not followed.

Plagiarism/Cheating

Academic dishonesty policies of IIIT Delhi apply.
https://www.iiitd.ac.in/sites/default/files/docs/education/AcademicDishonesty.pdf
Checks will be in place to detect copying/plagiarism/cheating

There will be no WARNING. Penalties will be imposed on the whole group in case of suspicion.
You may refer to the internet. But do not copy code/result/analysis from internet.
Do not copy from other group(s) or share your answers.
Even copying the FORMAT used by any other group will invite penalty for both the copier and the copied groups.

Choice of Problem

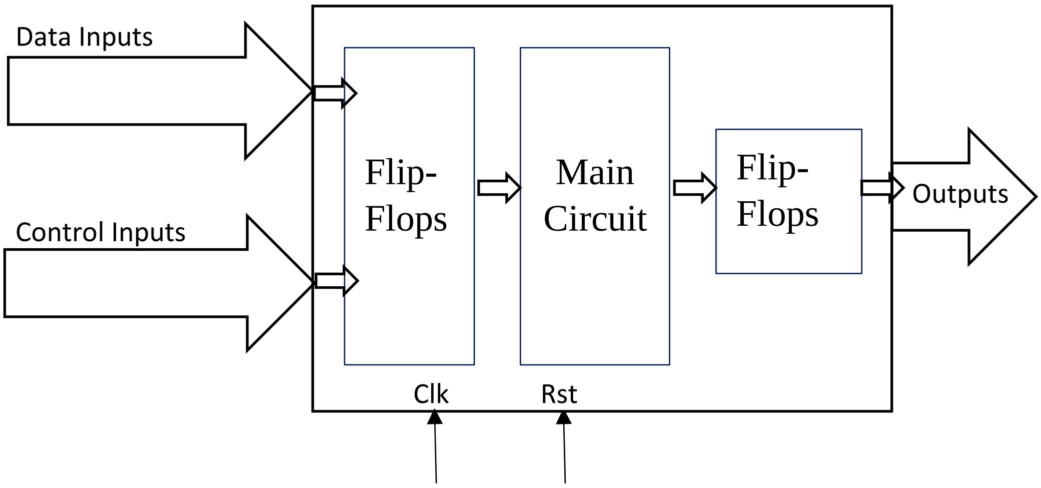
- Assignments will be done in groups that have already been formed.
- You can manually code the RTL or use any HLS tool. Even when you use an HLS tool, you will be expected to understand the generated code. **Do not copy RTL from the internet**.
- Keep focus on the objectives mentioned above and try fulfilling these objectives in the best possible manner. Evaluation will be based on how good these objectives are fulfilled by the students. **Note that during evaluation not only the results will be looked, but your interpretation/analysis of the results will also be taken into account.**

Problem Description

Following points are valid for all the problems:

- A specification with the details of inputs, outputs, functional behaviour, constraints, tools employed, etc. needs to be developed by the students, based on the design requirement.
- Clock and Reset should be used in all the problems. The design should be a synchronous design.
- All the data and control ports should be registered appropriately. Input ports should be connected to Flip Flops only (i.e., not to any combinational device directly). See figure below. Exceptions are Clock/Reset signals.
- Output ports should also be connected to Flip Flops only (i.e., not to any combinational device directly). See figure below. Exceptions could be output ports that generate Clock/Reset signals.

- The general structure of your design will be as shown in the following figure:
    - Clock (Clk) and Reset (Rst) signals should be used as required by the design.



Choice of Tools

There are sufficient licenses for the Cadence and Synopsys Digital-flow tools.

You can use opensource tools also.

Students are free to use any of the above tool sets.

EXCUSE of licenses/machines not being available will NOT be entertained.

The key to timely submission is to start early and complete your assignment a week before the completion date.

Design Steps

Following design steps need to be completed.

1. Make a detailed specification for your problem.
   It should mention Inputs and Output. It should describe functionality in plain English or state diagram or flowchart or pseudo-code (in whatever way you think is most appropriate).
   Explain all your assumptions.

2. Write Verilog code for the design meeting the above specification.
   Develop Testbench that simulates the above Verilog design and compares the results with the given specification.
   Three different testbenches need to be written, with different code-coverages. Simulation needs to be carried out for all the three testbenches and coverage computed for all three testbenches.
   You need to study the impact of test-vectors on code-coverage.
   Put your final analyses in the report.

3. Synthesize the design using three different constraints. Three different constraints should be as follows:
    a. Synthesize for minimum area, keeping timing constraints highly relaxed.
    b. Synthesize for best timing, keep timing constraints tight. Keep making the timing constraints tighter unless you observe a negative slack. The timing analysis should show slight negative slack for this constraint.
    c. Synthesize for timing constraints that is between (a) and (b) above.

    You need to study the impact of changing the constraints on the QoR.

    Also try to understand what portion of the code in RTL corresponds to which all instances in the netlist. Put your explanation in the final report.

4. Do formal equivalence checking of the generated netlists in 3 (a)-(c) and the Verilog code. Manually make a change in the netlist (let us call it bad-netlist) to make it non-equivalent with the RTL. Study the failure for this netlist and analyse the result.
    Note down different error messages that are reported by the formal equivalence tool. Try to make bad-netlist in several different ways and list out different types of error messages reported by Formal Equivalence Tool.

5. Perform Static Timing Analysis (STA) of the netlists generated in 3 (a)-(c)
    Use the same constraint file that was employed in 3(c) for performing STA of all three netlists generated in 3 (a)-(c).
    Analyze and interpret the slack obtained in all the three netlists.
    Analyze the timing report generated for different timing paths by the tools. Explain different fields that appear in the timing report and how are they computed.

6. Test insertion:
    Insert a single scan chain in the synthesized designs using RTL Compiler or Design Compiler
    Save the netlist that has scan-chain inserted.
    Explain the purpose of all the new entities that appear in the scan-chain inserted netlist.
    With the scan-chain inserted netlist, explain how the netlist will be used to detect failures.

    Report area and timing for scan-inserted netlists and compare with the result for netlist that has no scan chain.
    Explain the reason for difference in QoR for both timing and area.
    Take a detailed timing report of a same path in the original netlist and the scan-chain inserted netlist. Explain the difference in QoR by observing the effect of scan-insertion on different fields of the timing report.

Submission Requirements

Two files need to be submitted by each group (no individual submission):

1. One PDF file
2. One Zipped/tarred (archived) files

What does PDF file contain and what does archive folder contains is described below.

You should also keep your setup ready for a demo throughout the VDF course. Marks will be allotted for comments/documentation. You may also be asked to give a demo by TA/Instructor based on their discretion.

For each of the design steps mentioned above, following information should be supplied with the submission.

The explanation mentioned in the tasks above should be written in the PDF file. Additionally, following information must be mentioned.

Step1: Make a detailed specification for your problem

1. Specification of the design and assumptions used (in PDF file)

Step 2: Simulation and coverage analysis

1. Verilog code and three testbenches for the design (in same PDF file as above)
2. Simulation results showing a few waveforms for each of the three testbenches (screenshot should be saved in a PDF file)
3. Code coverage report for all three testbenches (in same PDF file as above)
4. Analysis of the results of simulation and coverage (in same PDF file as above)
5. Verilog code, logfiles of simulation and code-coverage, commands that were run (in archived directory)

Step 3: Synthesis

1. 3 different constraint files (in same PDF file as above)
2. Report three different QoRs obtained corresponding to three netlists(in same PDF file as above)
3. Analysis/Interpretation of results/QoR (in same PDF file as above)
4. 3 different constraint files, netlist, logfiles, commands that were run (in archived directory)

Step 4: Formal equivalence checking

1. Relevant snippet of the output of the equivalence checking for all three netlists generated by synthesis tool and the bad-netlist that was manually changed (in same PDF file as above)
2. Analysis/Interpretation of result for all the netlist, including the netlist that was manually changed (in same PDF file as above)
3. logfiles, commands that were run (in archived directory)

Step 5: Static Timing Analysis (STA)

1. Timing reports for all three STA (in same PDF file as above)
2. Analysis/Interpretation of result (in same PDF file as above)
3. logfiles, commands that were run (in archived directory)

Step 6: Test insertion.

1. Timing and area reports for scan-cell inserted vs. not-inserted netlist (in same PDF file as above)
2. Analysis/Interpretation of result (in same PDF file as above)
3. logfiles, commands that were run, original and scan inserted netlist (in archived directory)

Library to be used

90/45-nm libraries available in the installation of Cadence tools should be used.

If you want you can use open-source libraries also (Example: M. Martins, J. M. Matos, R. P. Ribas, A. Reis, G. Schlinker, L. Rech, and J. Michelsen, “Open cell library in 15nm FreePDK technology," in *Proceedings of the 2015 Symposium on International Symposium on Physical Design*, pp. 171-178, 2015.)

Deadlines

As communicated in Google Classroom.

Doubts

All doubts/confusion should be cleared well in advance. No clarification will be entertained close to the deadlines (clarification will stop two days before the deadline).

Late submission

20% penalty/day rule as communicated in the first lecture.