Homework 7 Write-Up

Lucais Sanderson [redacted] & Antonio Hernandez Ruiz [redacted]

March 12, 2025

1 Introduction

1.1 Project Overview

This project has us modify a processor using the OpenLane2/OpenRoad flow. The goal is to have a baseline design and then implement improvements targeting a lot of different metrics including, area, power, performance, and timing.

1.2 Objectives

The main objectives of this project are:

- Set up and arrange the processor design, including the manual placement of macros and I/O pins.
- Get a baseline design, making sure to measure metrics like power, performance, and area (PPA) after signoff.
- Get two improvements to make the design better and determine their impact.
- Make sure that the final design meets all timing constraints, passes design rule checks and the schematic design.
- If the objectives are not met, then explain why and provide detail as to how they should be met.

1.3 Design and Implementation Approach

The baseline design is created using a floorplan with manually placed SRAM macros and I/O configurations. The design is then improved in two ways:

1. Reducing the core utilization percentage and changing the placement to get a smaller size.
2. Adjusting the clock period and changing parameters such as gate size and clock tree changes to get better timing performance.

Both improvements want to balance trade offs between the area, power consumption, and the overall system performance. Then we measure the impact of the improvements to see how effective they were.

1.3.1 Alteration of `src` Files

We pushed the `src` Verilog files too because some modules used a gated clock. We changed these to the actual global clock to simplify timing analysis and clear up some hold violations. This removes part of the functionality, but it was made clear this is an OK change.

2 Baseline Design

2.1 Design Setup

The baseline design was made by setting up the processor with manual macro placement and I/O configurations. We took all gated clocks to help hold violations and added false paths where needed. The steps we did were as follows:

- SRAM and other major blocks were manually placed to make sure we had a balanced floorplan.
- We did the same for the placement of input/output pins to make the wire lengths smaller.
- Removed all gated clocks to have a simple clock tree synthesis.
- Made sure all clock inputs use the same global clock signal to have a smaller clock skew.
- Introduced false paths for, Asynchronous reset signals and Data bus bits that remain tied to logic 1 or logic 0.

2.2 Initial Timing and PPA Metrics

- Worst Hold Slack (Initial, No Fixes): -1.07 ns
- After Changing Wire Length (`DESIGN_REPAIR_MAX_WIRE_LENGTH=100um`): Hold Slack Improved to -0.25 ns
- After Changing Resizer Hold Fix (`PL_RESIZER_HOLD_SLACK_MARGIN=0.5`): Hold Slack Improved to -0.24 ns
- After Max Corner Fixes: Worst hold slack = 0.1811 ns

2.3 Key Observations

During analysis, we found persistent issues related to capacitance, slew violations, and antenna checks, which stayed despite multiple adjustments:

- **Capacitance and Slew Violations:**
  - Attempts to fix these violations using `DESIGN_REPAIR_MAX_SLEW_PCT` and similar parameters did not fix it much.
  - The largest violations were very large and passed the threshold.
  - The maximum number of slew and capacitance violations were 183 and 38, respectively. These worst case violations appeared in the `max_ss_100C_1v60` corner.
  -

- **Final Violations:**
  - **Max Capacitance Violation:** Wire3367/X passed the limit (0.189612 vs 0.081492). This violation isn't on the critical path in the `max_ss_100C_1v60` corner, so it may not matter so much especially given the worst setup slack is +44.29ns.
  - **Max Slew Violation:** ANTENNA_161/DIODE passed the limit (3.469613 vs 1.500000).
  - **Antenna Violations:** There are 94 pin antenna violations and 90 net antenna violations. In full transparency, this would likely mean the design would not function after being manufactured.
- **Persistent Challenges:**
  - While wirelength adjustments improved timing slack, they did not fix the capacitance and slew violations.
  - Future work is needed to explore other methods, such as gate resizing or manually adjusting routing paths, to fix these violations.

2.4 Congestion Map

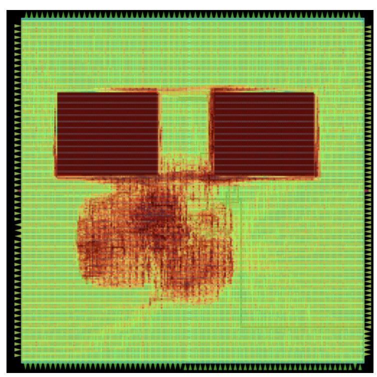

Figure 1: Congestion map before any improvements showing a big hotspot of congestion.

2.5 Critical Path

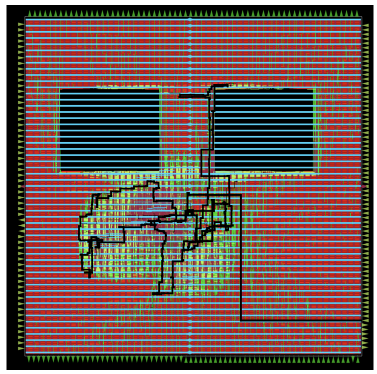

Figure 2: Critical path map before any improvements showing the problems we have.

3 Improvement 1: Area Optimization

3.1 Approach

The objective of this improvement was to reduce the die area while keeping good timing. This was done by changing core utilization and improving the placement of SRAM blocks to make a smaller and compact layout. The benefits of this improvement were lower wirelengths, which could improve timing and power efficiency.

- **Initial Die Area:** 2,712,760 μm^2
- **Core Reduction:** Decreased from 40% to 35%
- **First Attempt:** Area reduced to 1,367,580 μm^2 (49.6% decrease), but resulted in too much routing congestion and failed routing.
- **Final Attempt:** Adjusted placement and added more space between SRAM blocks, now we have a stable area of 1,560,120 μm^2 (42.5% reduction), which passed worst hold slack in the nominal corner.

3.2 Congestion Map

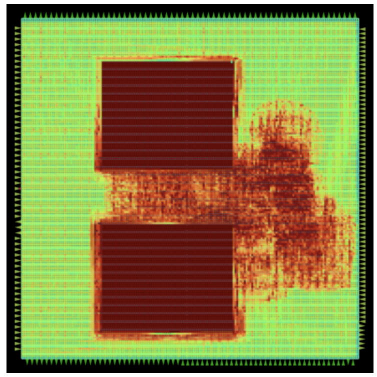

Figure 3: Congestion map after first attempt for improvement 1

3.3 Routing and Hold Slack Considerations

Reducing core area really helped with routing congestion. The first attempt resulted in an unrouteable design because of excessive wire congestion, so we did further adjustments:

- **Routing Congestion Problems:** The high density of components in a smaller area increased wire congestion, so we failed routing attempts.
- **SRAM Spacing Changes:** To fix the congestion, we changed SRAMs spacing and added an additional 50 µm to help with routing.
- **Timing Closure Adjustments:**
  - Applied `PL_RESIZER_HOLD_SLACK_MARGIN=0.6` to improve hold timing.
  - Increased post-CTS overcorrection to 1.2 ns, fixing the worst hold slack violations.
- **Final Outcome:** The adjusted design successfully passed worst hold slack across all process corners but introduced antenna violations.

3.4 Critical Path

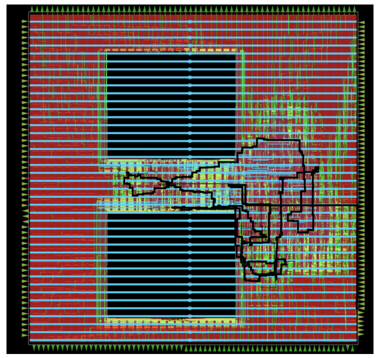

Figure 4: Critical path map after improvement 1 passes worst hold slack across all process corners.

3.5 Antenna Violations and Potential Causes

Antenna violations happened as a consequence of the area reduction. These violations occur when long interconnect wires have too much charge during the fabrication process, maybe even damaging transistors. Several things contributed to this issue:

- **Longer Routing Paths:** The reduced core size forced longer wire paths, making them more prone to antenna effects.
- **Limited Antenna Diode Placement:** The standard cell library has limited options for automatic antenna diode placement, making the problem bigger.
- **Routing Constraints:** Routing congestion forced bad wire paths, further increasing antenna errors.

Potential Solutions: While the design passed hold timing, addressing antenna violations in future work may require:

- Manual placement of antenna diodes in critical paths.
- Adjusting routing constraints to stop excessive wire accumulation in specific areas.
- Exploring alternative macro placements to reduce long interconnects.

3.6 Capacitance and Slew Violations

The maximum number of slew and capacitance violations again occur in the `max_ss_100C_1v60` with 630 and 141 violations, respectively.

- **Worst Cap. Violation:** With a 0.0815 limit and a worst cap. of 0.2523, yielding a -200% difference, exceeding the table so realistically, our timing evaluations could be considerably off. This worst case occurs at `wire5685/X`.
- **Worst Slew Violations:** With a slew limit of 1.5 and worst slew of 4.609, we get another -200% difference. Again, this is way off the timing chart, so our timing evaluations are likely unreliable. This worst case happens on `ANETNNA_305/DIODE`.

4 Improvement 2: Performance Enhancement

4.1 Optimization Approach

The second improvement focused on increasing performance by reducing the clock period, helping to increase the operating frequency of the design. Our goal was to use the available timing margin from the baseline design to get a higher clock frequency without violating setup and hold timing constraints.

- **Clock Period Reduction:** The clock period was reduced from 100 ns to 52 ns, doubling the system's clock frequency.
- **Maintaining Timing Integrity:** No major hold or setup timing violations were seen after the change.
- **Potential Gate Resizing:** Although gate resizing was considered, initial results showed no need for them. However, future work might use resizing to balance timing paths.
- **Clock Skew Considerations:** Since higher clock frequencies can increase clock skew issues, we reviewed the clock tree structure to make sure it wasn't affected.

4.2 Results and Analysis

4.2.1 Critical Path Improvement

The reduction in clock period shortened the time available for signal propagation. However, because the baseline design had positive setup slack, this change did not introduce violations. The critical path remained similar in structure but worked under a tighter timing constraint.

- **Path Length:** The longest combinational delay path remained the same.
- **No Setup Violations:** The design retained positive setup slack despite the clock period being reduced.
- **Potential Future Risks:** Further reducing the clock period may introduce violations, requiring additional improvements such as gate resizing or retiming.

4.3 Critical Path

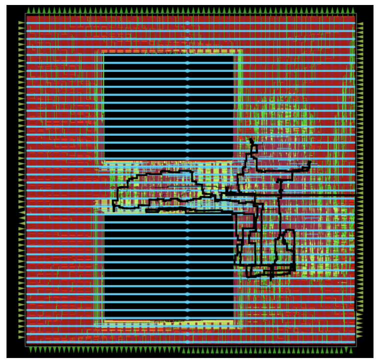

Figure 5: Critical path map after improvement 2 where we reduce the clock period.

4.3.1 Clock Skew Before vs. After

Clock skew is the difference in arrival times of the clock signal at different flip-flops in the design. The reduction in clock period increased the impact of skew, though no new violations were seen.

- **Baseline Clock Skew:** -0.146 ns
- **After Optimization:** +0.244 ns
- **Potential Issues:** If the frequency is increased further, clock skew effects may become bigger.

4.3.2 Capacitance and Slew Violations

We ended up with a lot of capacitance and slew violations; 134 and 609, respectively.

- **Max Slew:** With a limit of 1.5 and an actual worst slew of 4.369, we get a -190% difference. At `ANETNNA_200/DIODE`
- **Max Capacitance:** With a limit of 0.0815 and a worst capacitance of 0.239, we get a difference of -193%. At `wire5500/X`

These specific violations aren't on the critical paths in this specific corner, but given the number of violations and their severity, our timing results are not the best.

4.3.3 Final Performance Metrics

The key takeaway from this improvement was the ability to increase performance while keeping timing closure. The final performance metrics are as follows:

- **Clock Period:** 52 ns (compared to 100 ns baseline).
- **Worst Hold Slack:** 0.1189 ns (remained within acceptable limits).
- **Worst Setup Slack:** 44.4060 ns (maintained from baseline).
- **Power Considerations:** A higher clock frequency typically increases power consumption. While this was not an issue in our case, it should be analyzed further.

4.4 Challenges and Future Considerations

Although this improvement successfully increased the operating frequency, future refinements could give us better stability and efficiency:

- **Look into Gate Resizing:** To balance timing paths and further optimize setup/hold slack.
- **Clock Tree improvement:** To minimize clock skew effects and improve stability.
- **Power Improvement:** To improve on the trade-offs between performance and energy use.

4.5 Congestion Map

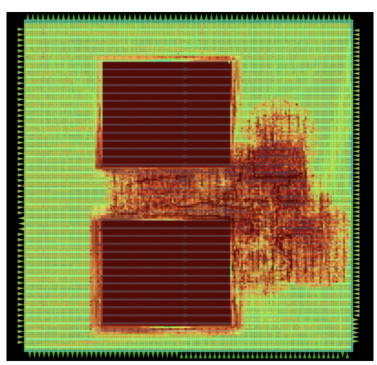

Figure 6: Congestion map after improvement 2 still with some violations.

5 Final PPA Metrics

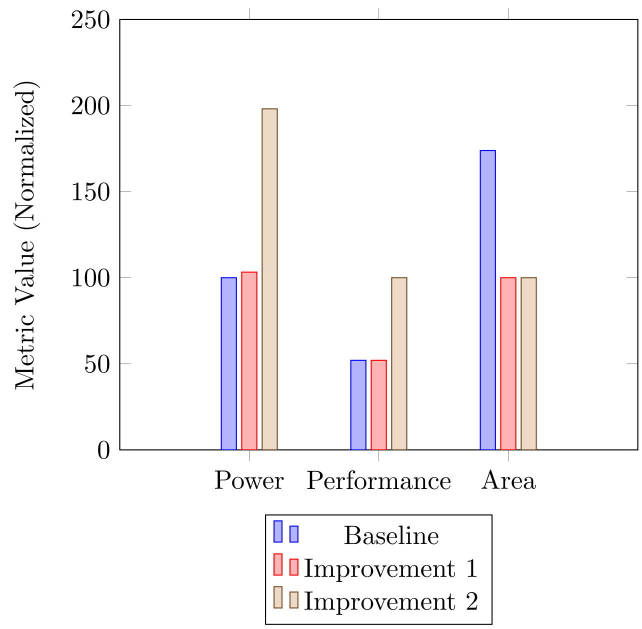

Figure 7: Comparison of baseline results and the following two improvement results. Values normalized by best case. For performance, higher is better. For area and power, lower is better.

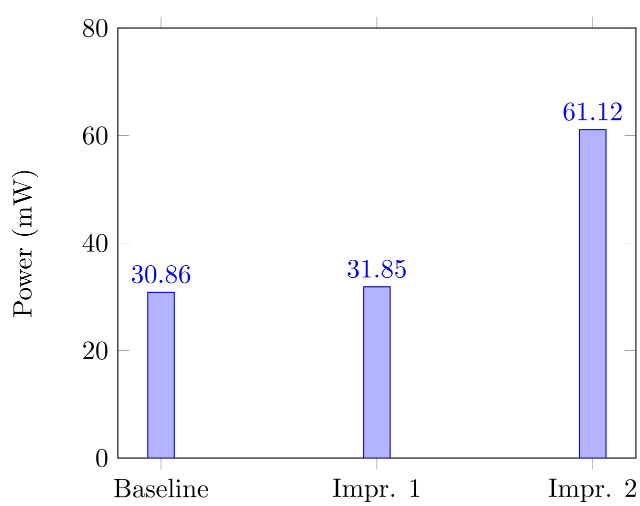

Figure 8: Raw power comparison between the three iterations.

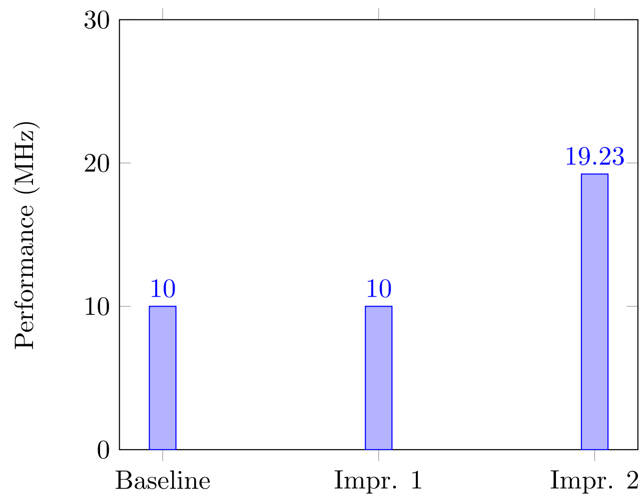

Figure 9: Raw performance comparison between the three iterations.

Metric	Baseline	Improvement 1	Improvement 2
Worst Hold Slack (ns)	0.1189	0.1811	0.4233
Worst Setup Slack (ns)	44.4060	44.4060	0.0011
Clock Period (ns)	100	100	52
Core Area (μm^2)	2,712,760	1,560,120	1,560,120
Utilization (%)	40%	35%	35%
Max Cap Violations	39	141	134
Max Slew Violations	196	630	609
Total Wirelength (μm)	1,200,332	1,130,831	1,130,831
Total Power (mW)	30.855	31.852	61.119
Clock Skew (ns)	-0.146	-0.146	+0.244

Table 1: Final Power, Performance, and Area (PPA) Metrics

5.1 Analysis of the Critical Path

The critical path in a digital design represents the longest combinational delay between two sequential elements (flip-flops), limiting the maximum clock frequency. If this path takes too long, the circuit may fail to meet timing constraints, leading to setup violations.

5.1.1 Identifying the Critical Path

In our design, we identified the critical path using timing reports from OpenROAD. This path consists of a series of logic gates and interconnect delays that add to the total delay. The main factors influencing this path include:

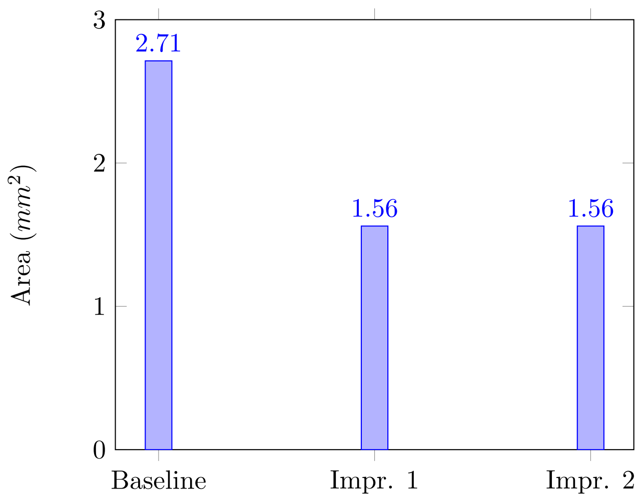

Figure 10: Raw area comparison between the three iterations.

- The number of logic levels between registers.
- The fanout of logic gates, adding delay due to load.
- Wire length and routing, where long interconnects add additional capacitance and resistance.
- Gate sizing and drive strength, which can either help reduce delay or worsen it if not properly done.

5.1.2 Critical Path Components

The identified critical path starts from the input port **instr_gnt_i**, which is clocked by clk_i, and ends at the flip-flop _22019_, also clocked by clk_i. This means that data must pass through multiple logic stages before the next clock edge.

- **Startpoint:** instr_gnt_i (input port clocked by clk_i)
- **Endpoint:** _22019_ (rising edge-triggered flip-flop clocked by clk_i)
- **Path Group:** clk_i
- **Path Type:** Minimum delay (setup constraint)

5.2 Congestion and Wirelength

One of the biggest issues we found in the baseline design was routing congestion due to high core use. By reducing the core utilization from 40% to 35%, we reduced congestion and improved routing efficiency. However, this also led to increased wirelength.

- **Baseline Wirelength:** The total wirelength before improvements was 1200332 µm.
- **Improved Wirelength:** After improvemnts, the total wirelength was reduced to 1130831 µm.
- **Congestion Areas:** Congestion hotspots were found near the SRAM banks, where high-density routing caused delays.

5.3 Power Breakdown

Power consumption in our design is primarily divided into sequential power (flip-flops and registers) and combinational power (logic gates and interconnects). The main contributors to power dissipation include:

- **Sequential Power:** This accounts for the power consumed by flip-flops and other storage parts. Since the design consists of multiple pipeline stages, sequential elements add a lot to total power consumption.
- **Combinational Power:** Logic gates, multiplexers, and other units add to combinational power. Due to high switching activity, this power use changes based on input workloads.
- **Dynamic vs. Static Power:** Dynamic power dominates in our design due to state transitions, while static power is smaller and is caused by leakage currents.

In our improvements, reducing area also helped decrease wirelength and switching activity, which led to a slight reduction in dynamic power. However, the increase in clock frequency from 100 ns to 52 ns in the second improvement increased switching power due to the higher number of toggles per second. While no major power violations were found, further gate resizing and power-aware improvements could help reduce unnecessary power consumption.

6 Conclusion

6.1 Summary of Improvements

- Area optimization: Successfully reduced area by 42.5% while keeping timing closure.
- Performance optimization: Improved clock frequency by 93.21%
- Remaining Violations: Some capacitance/slew issues as well as antenna checks are still there.
- The worst congestion we say was near the SRAM banks, where routing density was highest.

6.2 Explanation of Improvements

6.2.1 Improvement 1: Area Optimization and Its Challenges

- **Routing Congestion:** The first attempt to reduce area resulted in an unrouteable design due to high congestion. This was fixed by increasing spacing between SRAM macros and adjusting placement to distribute wiring more evenly.
- **Antenna Violations:** While we successfully reduced the area, antenna violations popped up. These occur when long interconnect wires accumulate excess charge. The smaller area led to longer wire lengths, adding to these violations.
- **Hold Slack Adjustments:** To address timing issues, we adjusted `PL_RESIZER_HOLD_SLACK_MARGIN` and increased post-CTS over correction to 1.2ns, which improved hold slack but did not eliminate all violations.

Potential solutions could be manually inserting antenna diodes, tweaking routing parameters to break long nets, or introducing additional vias to help with charge accumulation. These are all speculations based on what we have learned so far in this class but we are not one hundred percent sure that they will be effective in leaving this entire project without any issues. The main problem arises in the process of fixing the issues that we had. Things take a long time to process and figure out for this project.

6.2.2 Improvement 2: Clock Optimization and its Challenges:

For the second improvement, we aimed to increase the clock frequency by reducing the clock period from 100 to 52 ns, doubling the system speed. Surprisingly, this change did not introduce any new timing violations in setup or hold slack.

- **Potential Timing Risk:** While no immediate timing failures appeared, further frequency increases could push the design close to setup violations.

- **Power Considerations:** A higher clock frequency increases dynamic power consumption. While this was not a concern in our tests, future changes should monitor power scaling effects.
- **Clock Skew Concerns:** Higher frequencies can amplify clock skew issues, affecting synchronous data transfers. We did not see major skew problems in this test but additional timing adjustments might be necessary in future work.

Although the second improvement got a performance gain with small changes, additional improvements could further improve the design:

- Implementing gate resizing to better balance delay paths.
- Looking into clock tree optimizations to reduce skew-related risks.
- Further power analysis to understand the trade-off between performance and energy efficiency.

Our baseline design already had large positive setup slack, meaning that data signals arrived at flip-flops before the clock cycle deadline. By decreasing the clock period, we fixed this extra timing margin to improve performance without causing violations.

6.3 Lessons Learned

- **Optimizing one metric (e.g., area) can negatively impact another (e.g., routing congestion).** Reducing the core area helped shrink the design but increased routing congestion. This showed us how important balancing area reduction works with routing feasibility.
- **Gate resizing and manual macro placement influences performance.** Macro placement impacted how efficient routing and congestion were. We saw that when SRAM macros were placed incorrectly it led to high wire density, which made timing issues pop up.
- **Tuning of hold timing and reducing slew and skew violations.** While we fixed the worst hold slack violations, we also saw excessive slew and capacitance violantions stay. We also saw that although our clock tree synthesis fixes helped manage skew, higher frequencies made small skews more prominent.
- **Antenna fixes.** The number of present antenna violations is concerning. We attempted using increased antenna fixing iterations (e.g. `GRT_ANTENNA_ITERS=10`) and specifying parameters for heuristic diode placement, to no avail. This can be resolved with better understanding and implementation of antenna fixes and giving the flow more room by changing parameters in other areas. Otherwise, the current design will likely fail manufacturing due to the accumulated charge during polishing.