ECE 4310-001

Analysis of Designing a 4-Bit Shift Register Using VLSI Techniques

Jason Bissias

Texas Tech University

Electrical and Computer Engineering Department

December 2023

Abstract

This report outlines the design process and considerations of a 4-bit shift register, with the aim of maximizing Figure of Merit.

Acknowledgements

My appreciations to Dr. Li and Mr. Carman for their time and effort in answering my many questions. I would also like to thank Dr. Kim for his bountiful insight into transistor design.

Table of Contents

List of Figures 4

1. Introduction 5

2. Schematic Design 5

2.1 Choosing Device Design Types 6

2.2 Implementing and Optimizing 10

2.3 Simulation Results 11

3. Layout Design 12

4. Conclusion 13

List of Figures

Figure 1: Generic 4-bit Shift Register 6

Figure 2: Static Flip Flop 7

Figure 3: Dynamic Flip Flop 8

Figure 4: The Final Design Choices 9

Figure 5: Comparison between Dynamic & Individual Gate Design (Old) and the Static Design & Compound Gate Design (New) 10

Figure 6: Final Simulation Design 12

Figure 7: Final Layout of the 4-bit Shift Register (each dot is a distance of 5 μm) 13

1. Introduction

The main objectives of this project were to design a 4-bit shift register in Cadence Virtuoso with the goal of achieving as best a Figure of Merit as possible. Doing so involved minimizing transistor amounts, optimizing transistor designs, and designing the layout of the device to be as tight as possible, all of which contribute to maximum clock frequency one way or another.

2. Schematic Design

Designing the schematic, the logic of the device in visually digestible form, took a few attempts to nicely realize. The changes from the original plan stem from the desire for optimization and simplification, both for the sake of the Shift Register's performance and of comprehensibility. To give some context, in Figure 1 is a schematic of a generic 4-bit Shift Register. For brevity's sake, I will refer to the AND and OR gates preceding the Flip-Flop stage as the Enable stage, since that is where it is determined whether each bit is to run with the inputted values, with fixed default values, or to write new data.

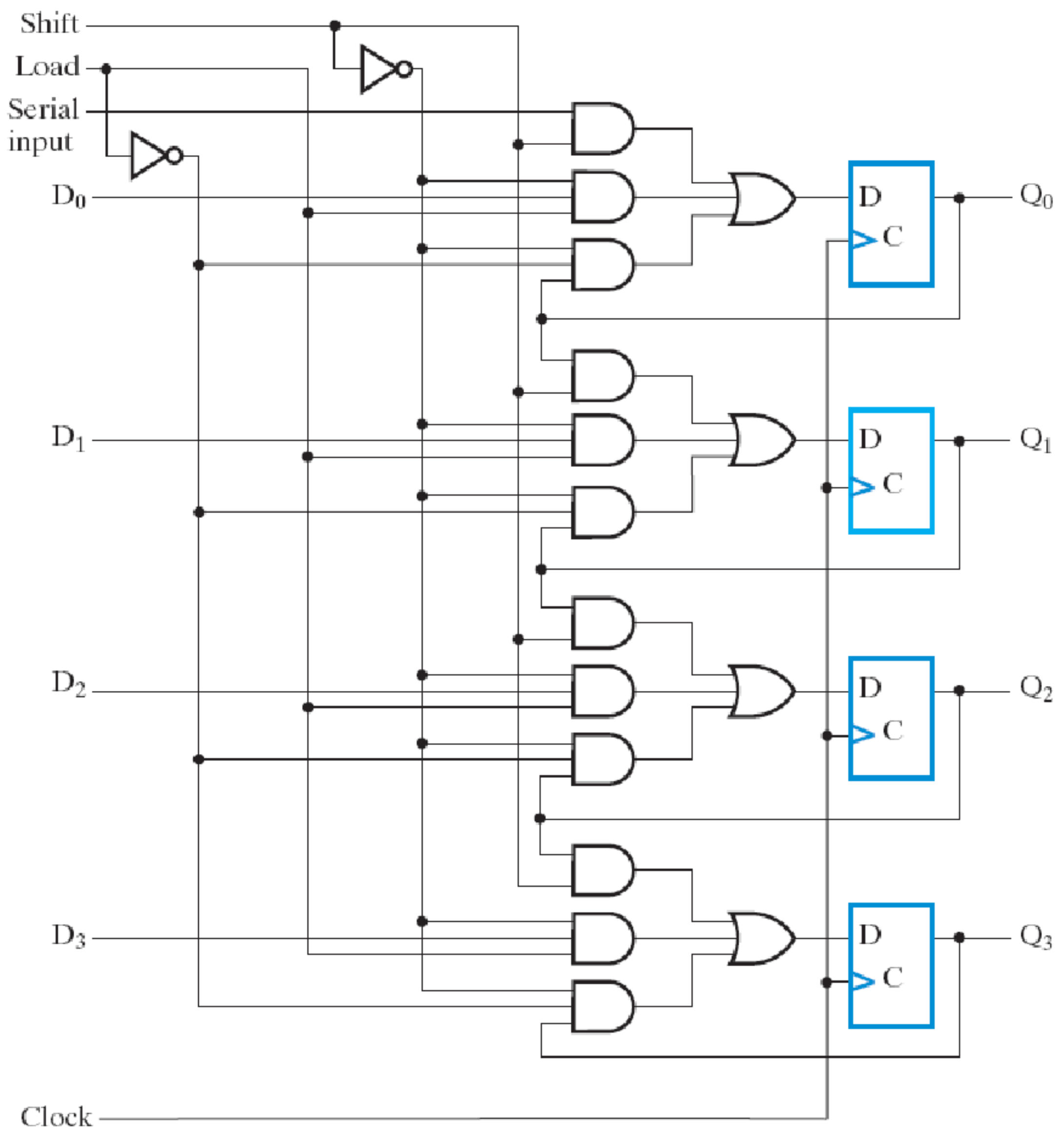

Figure 1: Generic 4-bit Shift Register

2.1 Choosing Device Design Types

The main places we have choice of device design is in the Flip-Flop and in the Enable stage.

For the Flip-Flop, we can select either a static or a dynamic design. The main benefit of the static design, shown in Figure 2, is that it is robust and much harder to cause error. However, this comes at the expense of more transistors, which means a larger footprint and a slower output. Initially, a static Flip-Flop was chosen as it is what will most likely be implemented in a real design, since this shift register is not a device that needs to be the fastest nor smallest. However, by the end of the project time, a dynamic Flip-Flop design, shown in Figure 3, was chosen since the objective of this project was to build a shift register with the best possible Figure of Merit. Using the dynamic Flip-Flop drives the design closer to the goal, allowing for 10 less transistors per iteration, or 40 less transistors in our total design.

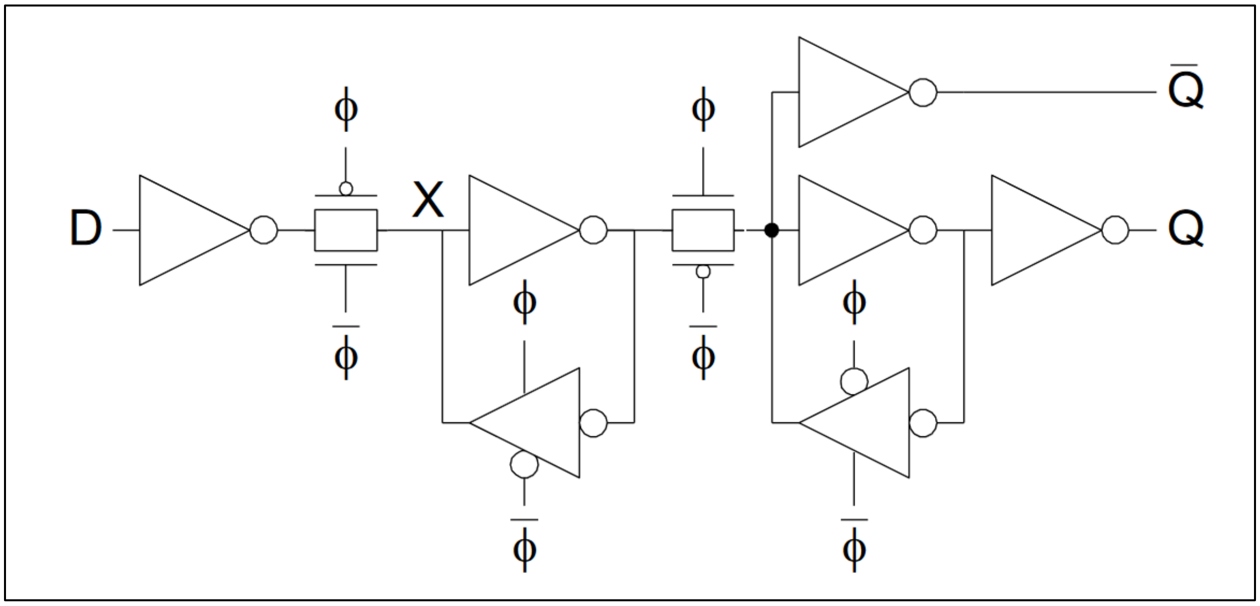

Figure 2: Static Flip Flop

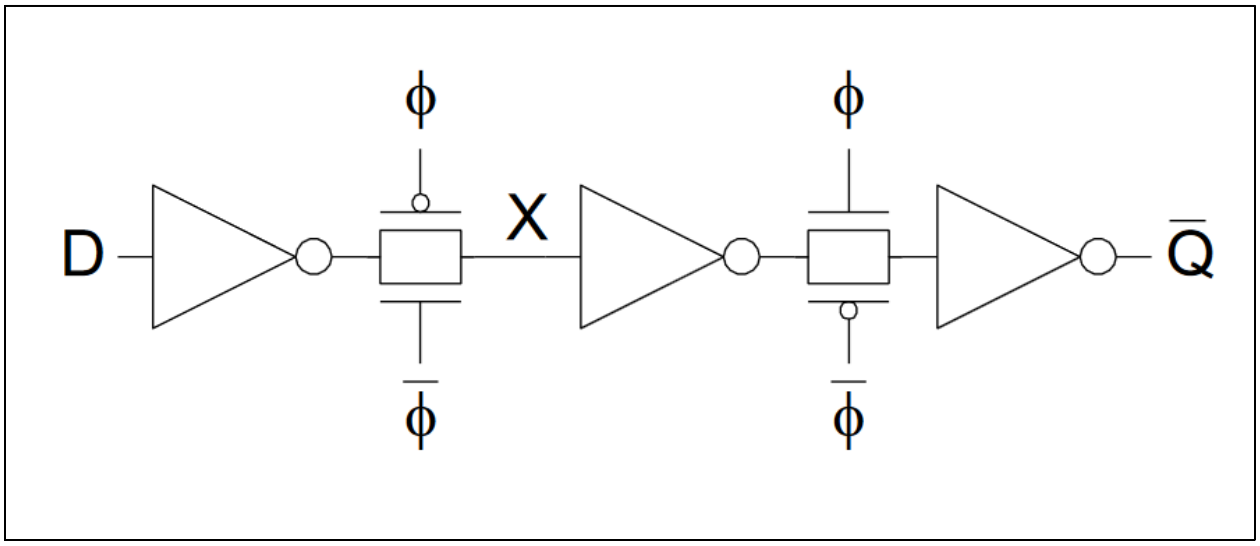

Figure 3: Dynamic Flip Flop

As per the Enable stage, the answer is a lot more clear-cut. The initial idea on designing this stage is to just follow the schematic as shown in Figure 1. Albeit, doing so requires designing NAND and NOR gates, each followed by an Inverter, since we use CMOS logic. This allows for a simple implementation and works reasonably well. However, the far better way to implement this stage is by using a single Compound gate for each bit. By replacing four gates into one, the transistor amount is reduced substantially, allowing for a simpler transistor-level design and faster results. Additionally, unlike with the Flip-Flop portion, using this method does not cause the robustness of the device to suffer. With this final design in mind, it is important to note the numbers: The Compound gate reduces the transistor amount from 30 to 18 in the Enable stage, compared to using individual NAND and NOR gates.

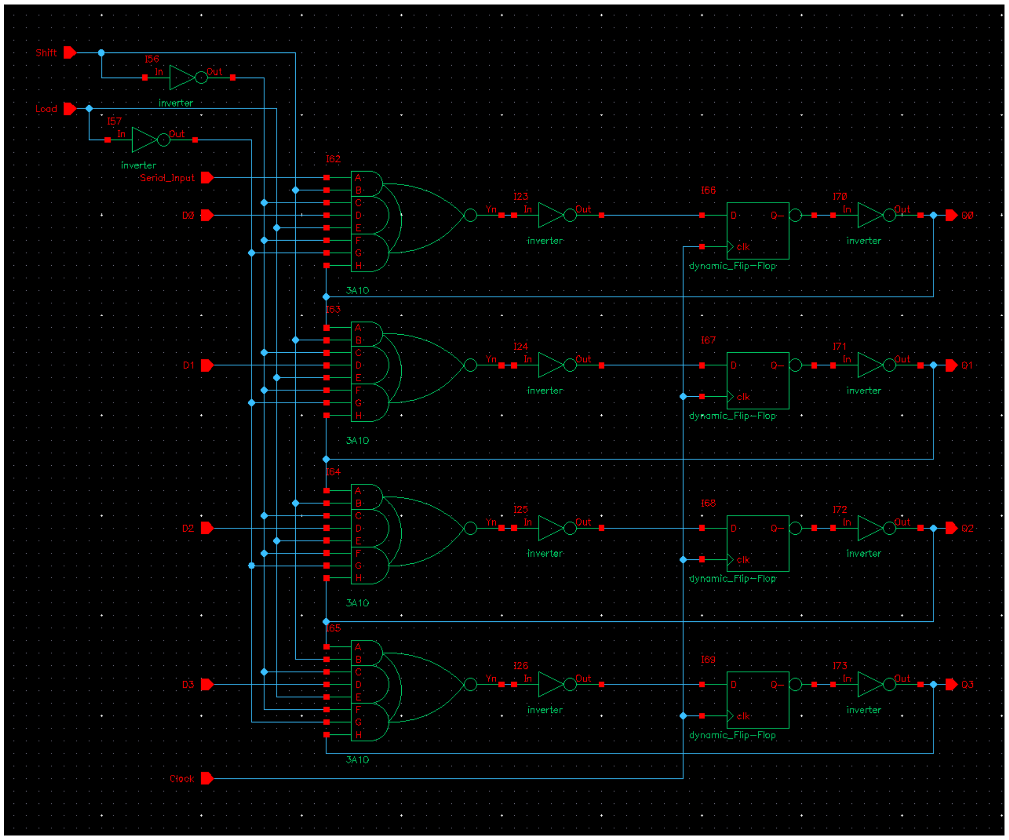

Figure 4: The Final Design Choices

Combining the two final designs produces a substantial improvement in Figure of Merit. On size of the design alone, a Static Flip-Flop design along with individual NAND and NOR gates yields 220 total transistors used. By utilizing the Dynamic Flip-Flop along with the Compound gate, the total design is reduced to just 132 transistors. That is ~60% the original amount, allowing our Figure of Merit to nearly double. For speed, the results are also substantial. As visible in Figure 5, Q0 of the Shift Register with a Static Flip-Flop is nearly 2 ns slower than the Shift Register with a Dynamic Flip-Flop. By the last bit, Q3, the Dynamic design is about 3.5 ns faster.

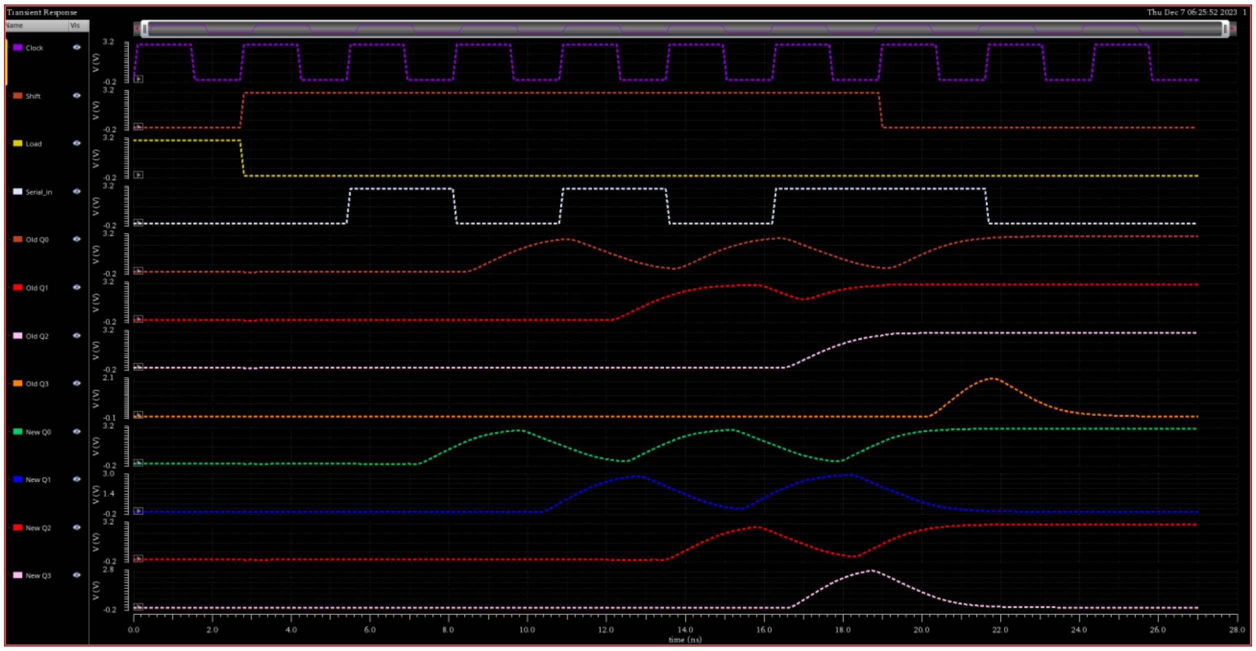

Figure 5: Comparison between Dynamic & Individual Gate Design (Old) and the Static Design & Compound Gate Design (New)

To note with Figure 5, Q1 old is outputting the wrong value perhaps because the old design is not made for such high clock speeds, albeit they were only increased by 1.3 ns.

2.2 Implementing and Optimizing

The main way for optimization in this circuit was by adjusting the widths of transistors in the Compound gate and the Flip-Flop. The interesting thing to note is that keeping the Inverters throughout the system unskewed allowed for the best balance between rising and falling propagation delay. The testing was conducted at a clock cycle of 3.75 ns before the optimizations were solidified, at which point the clock cycle was run lower and lower.

As per the Compound gate, the initial values set were 1.5 µm for the two NMOS in series, 3 µm for the two PMOS in parallel, 2.25 µm for the rest of the NMOS, and 4.5 µm for the rest of the PMOS. These initial values were chosen because the branches for the “three input" are

1.3 times larger than the “two input” portion, therefore the widths were set to match that. However, with experimentation, an interesting result emerged: The ideal balance of sizes for the transistors in the Compound gate are, in the order declared before, 1.5, 3.25, 1.5, and 2.6 μm respectively. The benefits of these optimizations are two-fold: Less power is lost along the chain, as the initial sizes caused a ~0.25 V drop in peak voltage at the last bit whereas the optimized values reduce the voltage by no more than 1 mV; and the propagation delay reduced by ~0.4 ns in the first bit and by ~1.3 ns in the last bit.

As per the Flip-Flop optimization, it came down to the Transmission gates since the change in inverters did not help. The original assumption was that keeping the ratio between the PMOS and NMOS 2:1 would be ideal. The experimentation, however, proved that keeping them at equal sizes was the ideal, though the improvement was in the thousandths of nanoseconds.

2.3 Simulation Results

With these optimizations, the absolute fastest the clock speed could be set at was 2.7 ns per cycle (370.4 MHz), however that was approaching the borderline, as Q3 was just barely able to reach above the 0.8 V_{DD} line (2.4 V in this case). A more comfortable setting is at 3 ns per clock cycle (333.3 MHz), as at that point the voltage loss was only just less than 0.2 V at Q3, allowing for better signal integrity further down the line.

However, for the purposes of this project, the 370.4 MHz clock frequency will be kept. As Figure 6 illustrates, it takes ~17.1 ns for 0000 to turn into stored 1010.

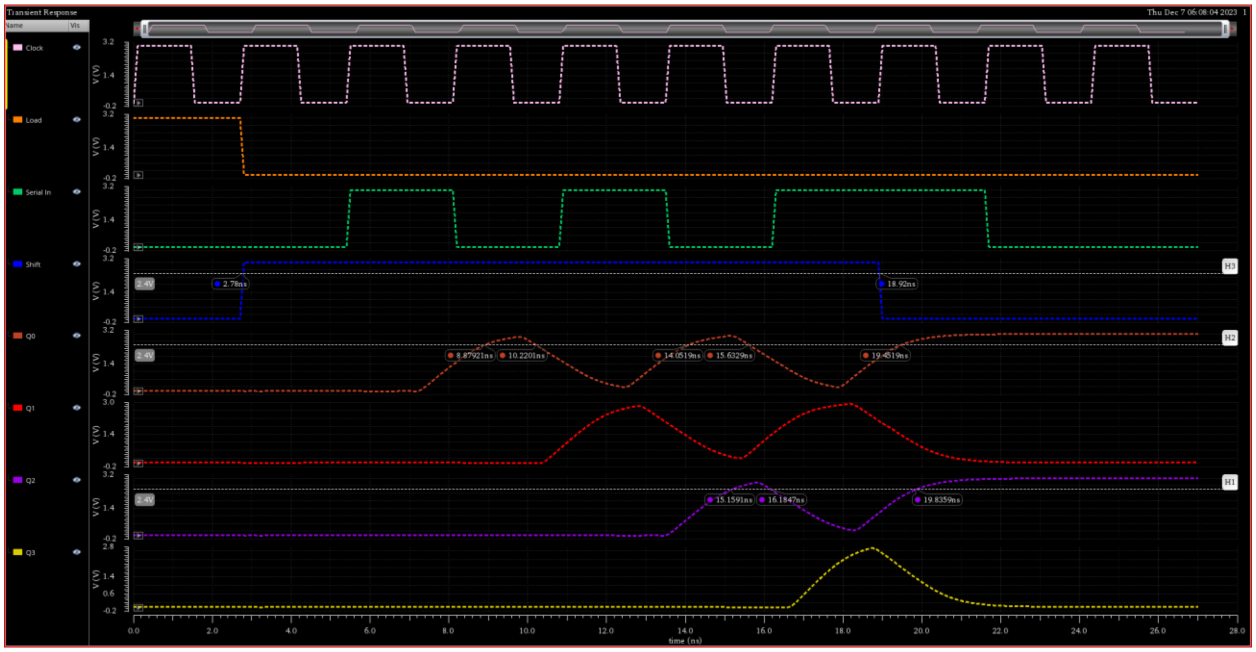

Figure 6: Final Simulation Design

3. Layout Design

The first step in designing a layout is planning it out. There is only so much overlap that can happen amongst various lines before one runs out of space, layers, or paths. Stick diagrams help massively with this, though there are some parts one can easily overlook.

The final design of the 4-bit Shift Register is 100 × 125 μm, as in 12,500 μm^2. This was achieved with a decent number of inefficiencies, as can be seen in Figure 7. Notably, in the input portion of I/O. Shaving the Shift and Load inputs down so that they are level with the top of the N-well can easily solve 1000 μm^2 off the total area, for example. Additionally, there may be some better methods to getting metal lines to stick tight and in an orderly fashion, however the limiting resources and capabilities of having only three metal lines to one's disposal makes it for a truly monumental puzzle to solve.

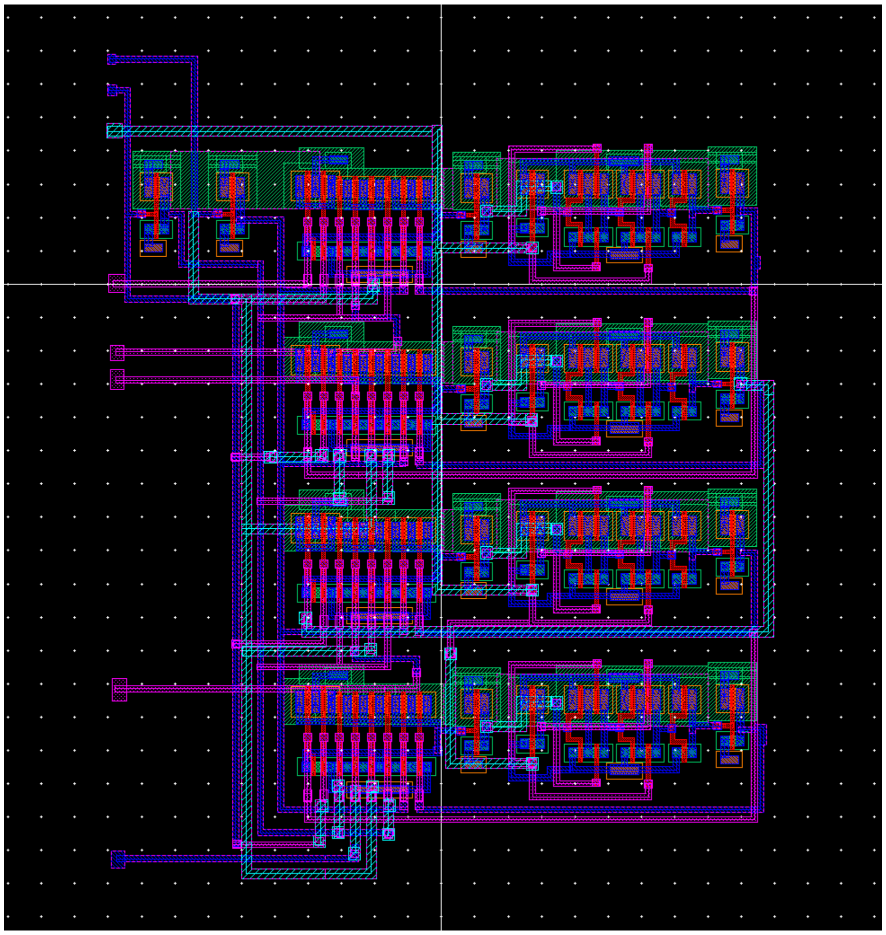

Figure 7: Final Layout of the 4-bit Shift Register (each dot is a distance of 5 µm)

4. Conclusion

The final Figure of Merit score:

$$FOM = \frac{1}{Ts \times A} = \frac{1}{17.1\text{ ns} \times 12{,}500\ \mu\text{m}^2} = 4.68 \times 10^9$$

This FOM is substantially better than if we retained the old system, which we can predict to be twice the size in either direction and with more than a clock cycle of difference by the time Q3 propagates.