ESE5730 Chip Design

Computing: Digital 8-bit 6x6 Matrix Multiplier Engine

Team 8

Shun Yao

Yue Zhang

TA: Claire Kendell

DEPARTMENT OF ELECTRICAL AND SYSTEM ENGINEERING

UNIVERSITY OF PENNSYLVANIA

I. Layout Design Flow

To finalize the layout of our design, we began by implementing and verifying basic building blocks such as AND2 and AND3 gates. Each of these fundamental cells was laid out following a consistent standard layout style, including uniform height, pin alignment, and metal routing conventions. This standardization made it easier to integrate the basic cells into larger blocks later in the design process. We used them as confirmed fundamentals after making sure each cell had DRC-clean layouts and a successful LVS match.

Once the basic cells were completed, we constructed higher-level sub-blocks by instantiating and interconnecting these primitives and then blocks. After completing the layout of each block, we ran post-layout simulations to verify their functional correctness. Only after a block passed post-layout verification did we proceed to the next. Finally, all the blocks were integrated into the top-cell layout. During this phase, we resolved inter-block routing, verified global connections, and ran final DRC, ANTENNA DRC, LVS, and top-level post-layout simulations to ensure the entire design met both functional and physical requirements.

II. System Block Diagram, Schematics and Layout

Top-Cell:

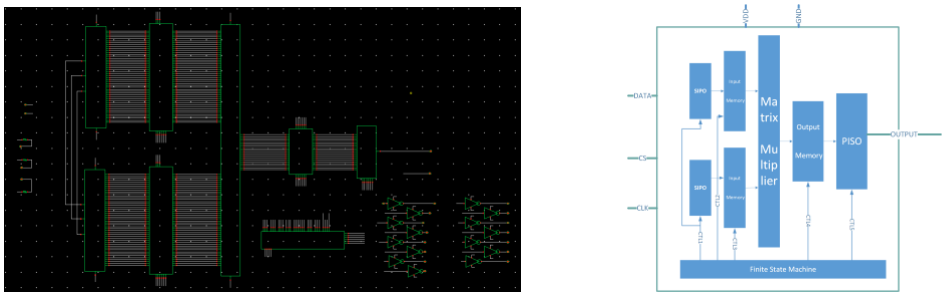


Figure 1. Top – Cell

SIPO:

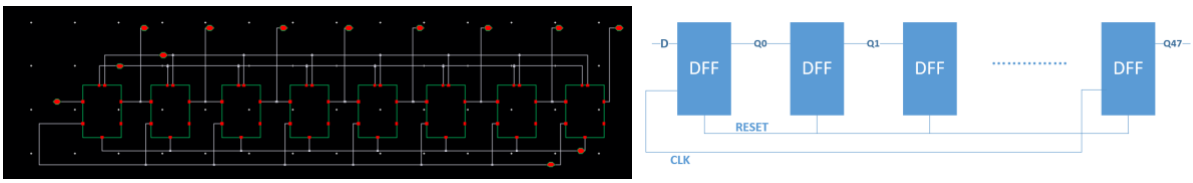


Figure 2. SIPO Shift Register (left: Detailed Schematic, right: Full Schematic)

Input Memory Array:

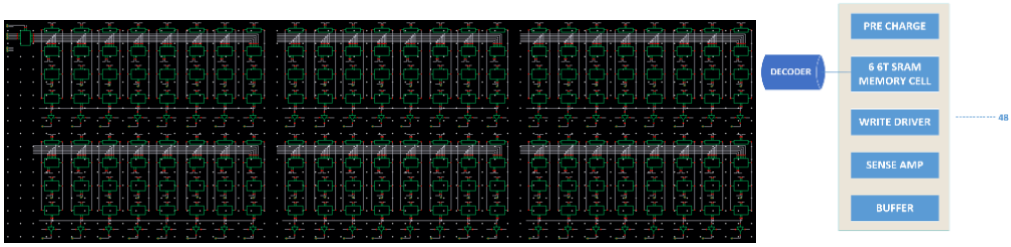


Figure 3. Input 8 bit 6x6 Memory Array

Matrix Multiplier:

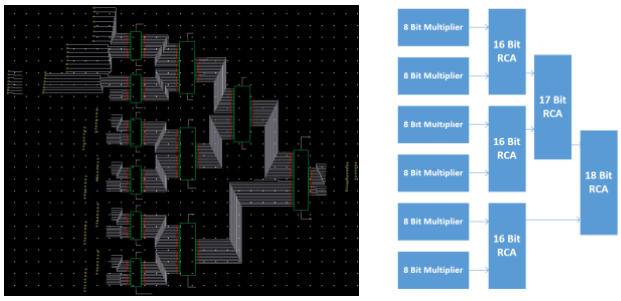


Figure 4. Matrix Multiplier

Output Memory Array:

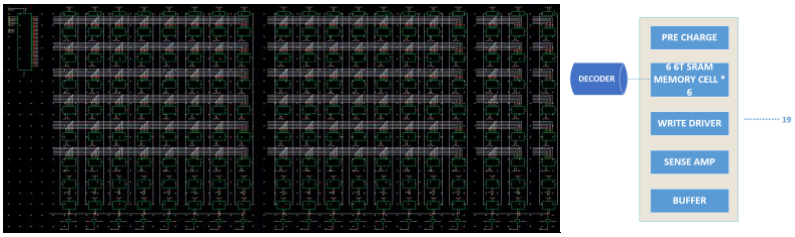


Figure 5. Output 19bit 6x6 Memory Array

PISO:

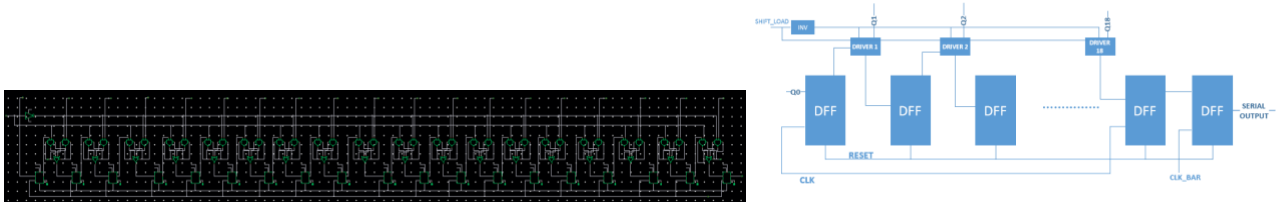


Figure 6. PISO Shift Register

Finite State Machine:

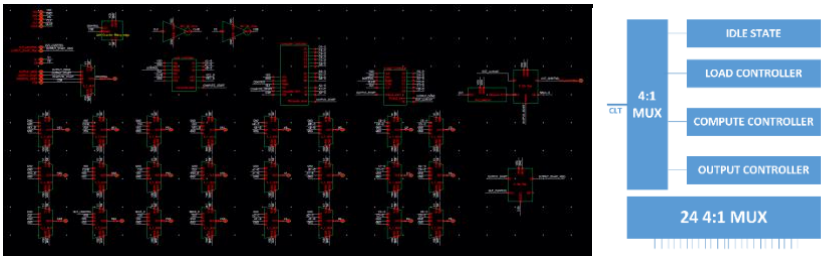
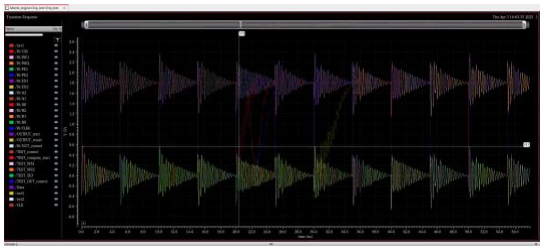


Figure 7. FSM Schematic

Optimization from Design Review 1

After Design Review 1, we continued the design by implementing the FSM block. In Design Review 2, the FSM was fully completed and integrated into the system. No other blocks were modified during this phase. After parasitic extraction, simulation results are correct with significant oscillations. To improve the performance, we decided to add small resistors to reduce this underdamped oscillation as show in the Figure below. Additionally, we plan to further optimize the existing layout in the next phase to improve overall chip performance, focusing on reducing parasitic delay.



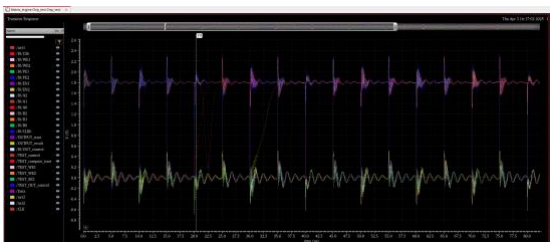


Figure 8. Oscillation Before (up) and Oscillation After (down)

III. Circuit Block Layouts

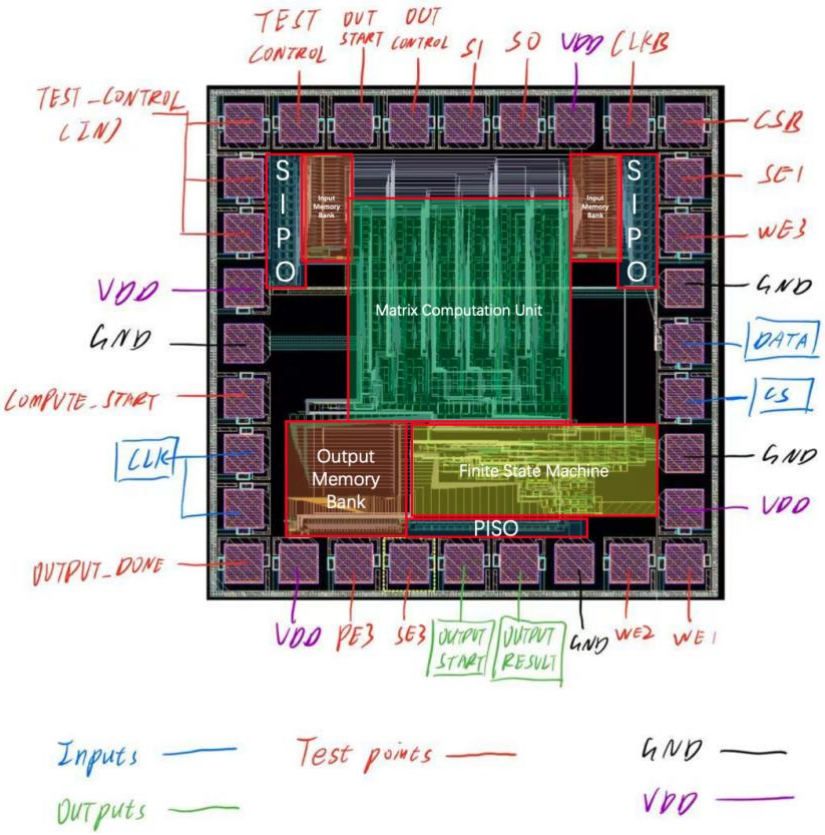


Figure 9. Top - Cell Pad Pinout

IV. Pre-Layout and Post-Layout Simulation

Top Cell: Transient simulations have run for top cell schematic. Take one calculation circle as an example, six 0011 0101 times six 1111 1111 correspondingly, and adding the six multiplication results together, the final result is supposed to be 19bit 0010 0111 1001 1000 010 as show in the Figure10 below (First Computation Result).

The system is firstly in an IDLE state waiting for the start signal ‘CS’; when signal ‘CS’ pulls down from 1 to 0, the system goes to LOAD state loading data to MatrixA and MatrixB; when the load is done, the finite state machine would push the system to COMPUTE state, the system will read data from MatrixA and B, compute it, and write it to MatrixC in this state; when computation is done, FSM will drive the system to OUTPUT state, which will read data from MatrixC and load it to PISO register

and the result will output in serial as shown in figuree10: (Post_layout Result will be present in critical design review)

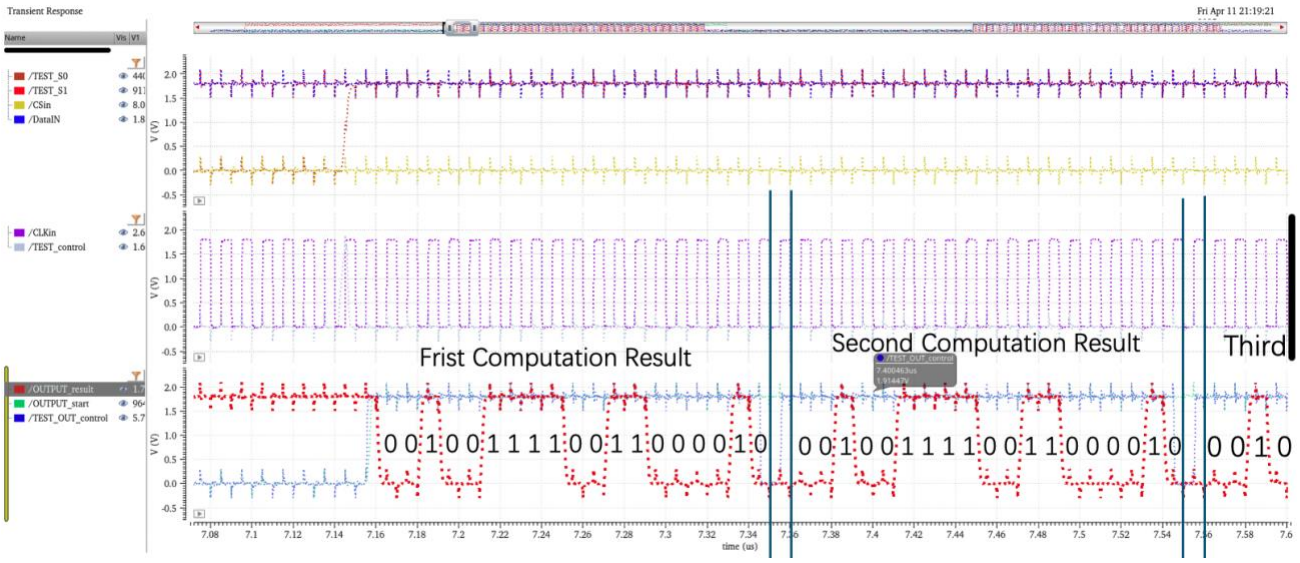


Figure 10. Top – Cell Simulation Result (with 2nH inductors)

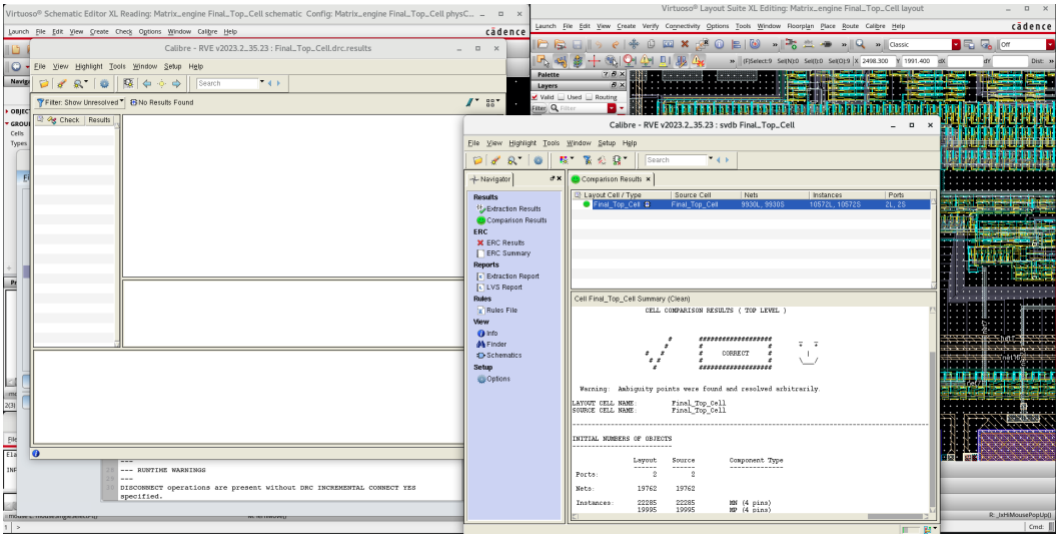


Figure 10. Top – Cell DRC/LVS Result (Included ant.drc test)

SIPO: Input data (D) (10101100) (iterate) with 10ns period and 30ns delay time; RESET is set to have 30ns low and long period high to simulate the SPI is enabled; CLK is 10ns period. The expected output is 1010110010101100…10101100. Figure 11 below shows the first five simulation results, and Figure 12 shows the comparison between Pre-layout and Post-layout results.

Figure 11. SIPO Serial Out Verification (First Five Outputs)

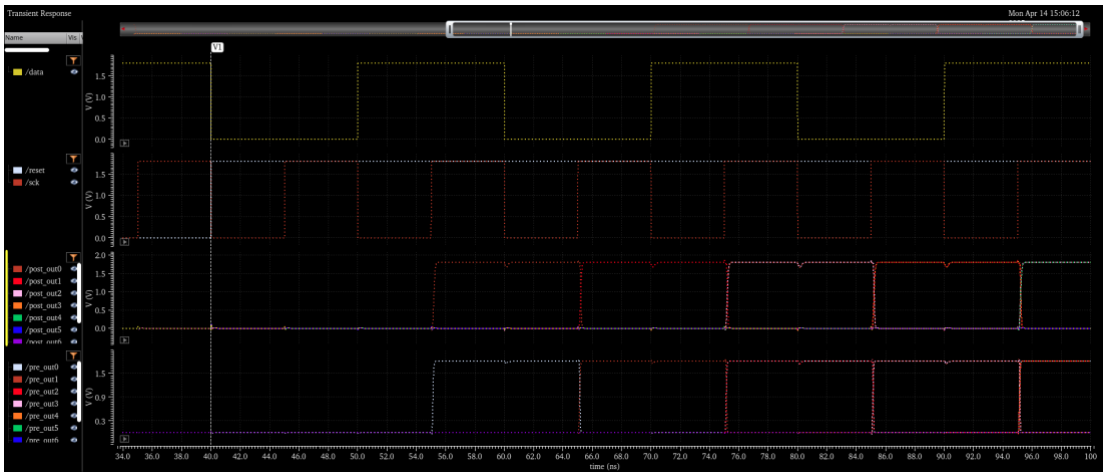


Figure 12. SIPO Simulation Result (Post_layout matches Pre_layout)

Input Memory Banks: take 8 data points for demonstration (address:111). Figure 13 below have shown a comparation between Pre/Post layout simulation results, which is basically same with each other. Therefore, the post layout result is correct.

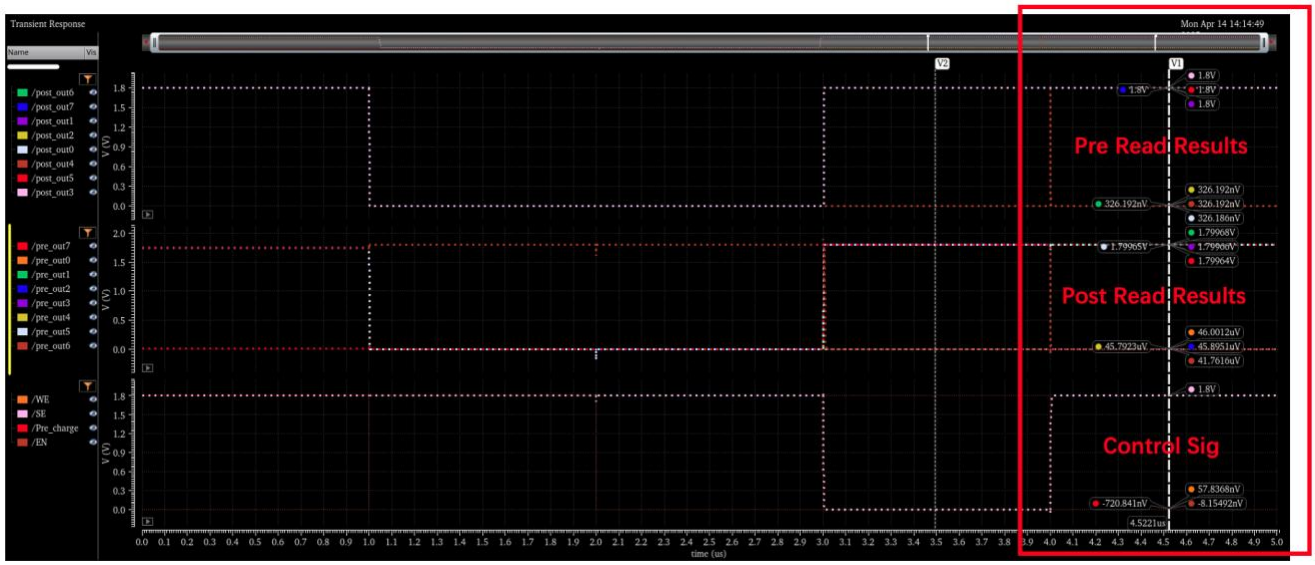
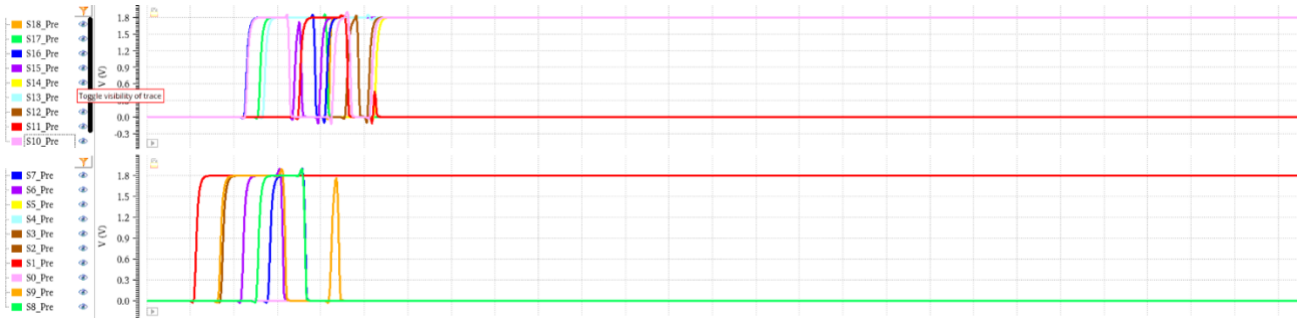


Figure 13. Input Memory Array Simulation Result

Matrix Multiplier: Transient simulation is run. Set the column (6 8bit) and row (6 8bit) to 1, the expected result is 19bit 101 1111 0100 0000 0110. The results below meet the expectations. Pre-layout correctness has been proven in Design Review 1; therefore, here, we present the consistency between the pre-layout and post-layout to verify the functionality.



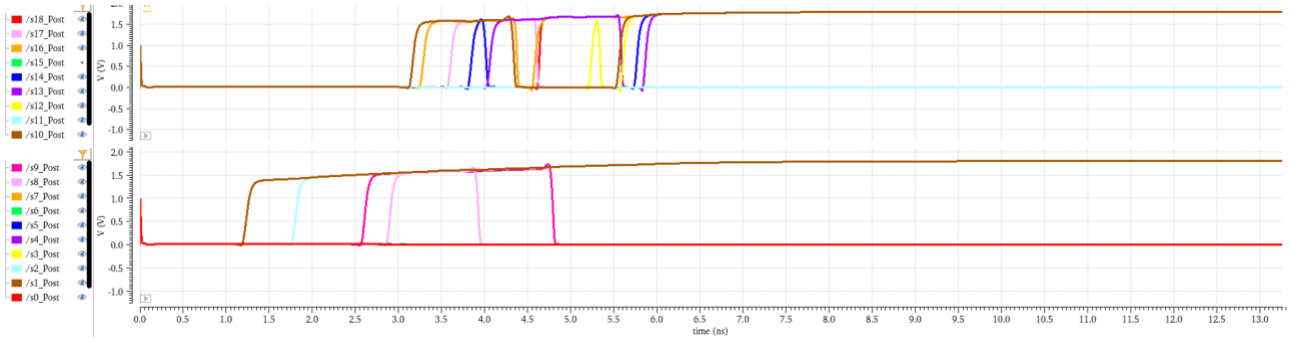


Figure 14. Matrix Multiplier Pre-Layout(up) and Post-Layout(down) Simulation Results

Output Memory Array: take 8 data points for demonstration (1010 1111) (address:111111). Figure 13 below have shown a comparation between Pre/Post layout simulation results, which is basically same with each other. Therefore, the post layout result is correct.

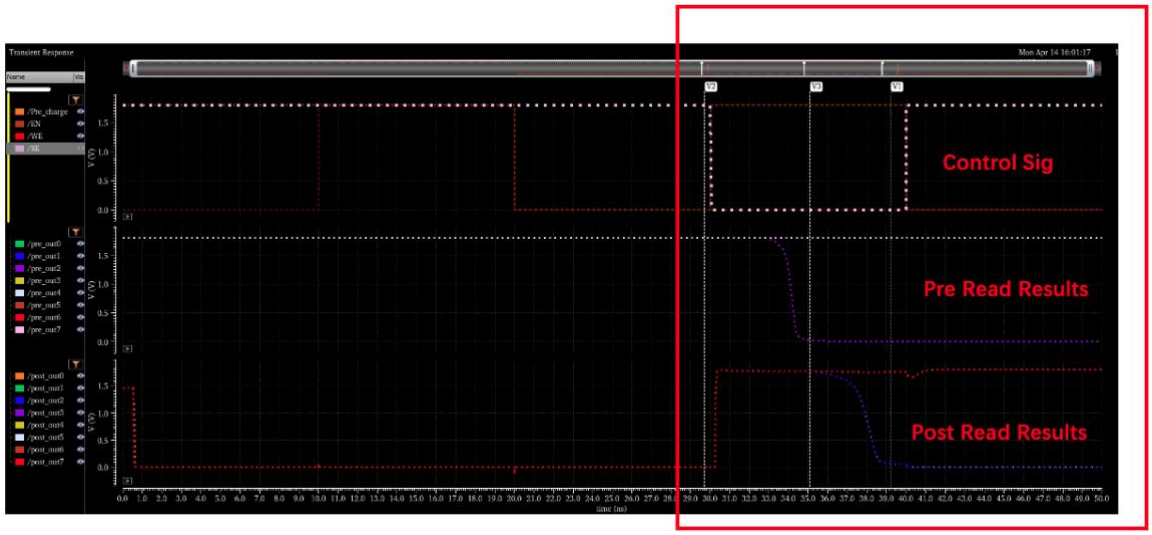


Figure 15. Output Memory Array Simulation Result

PISO: Transient simulation is run with 200ns stop time. Control input is set with 5ns low and long period high, CLK is 10ns period, and parallel input from Q18 to Q0 is 0111111111000000000. The serial output is shown in the following figure. Pre/Post layout simulation both correct.

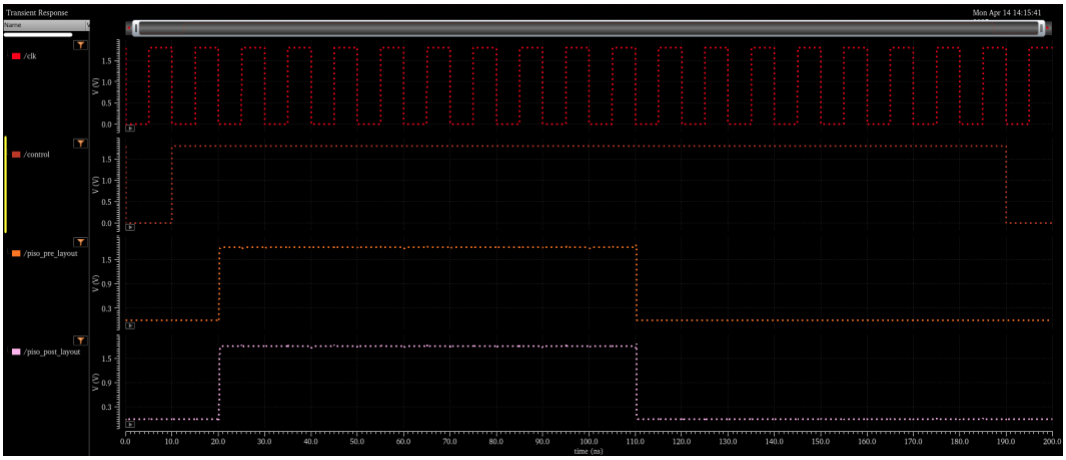


Figure 16. PISO Simulation Result

FSM: Transient simulation has run for 14us. The pre_layout and post_layout control signals generated by FSM were both correct and matches each other. As presented in the figure below.

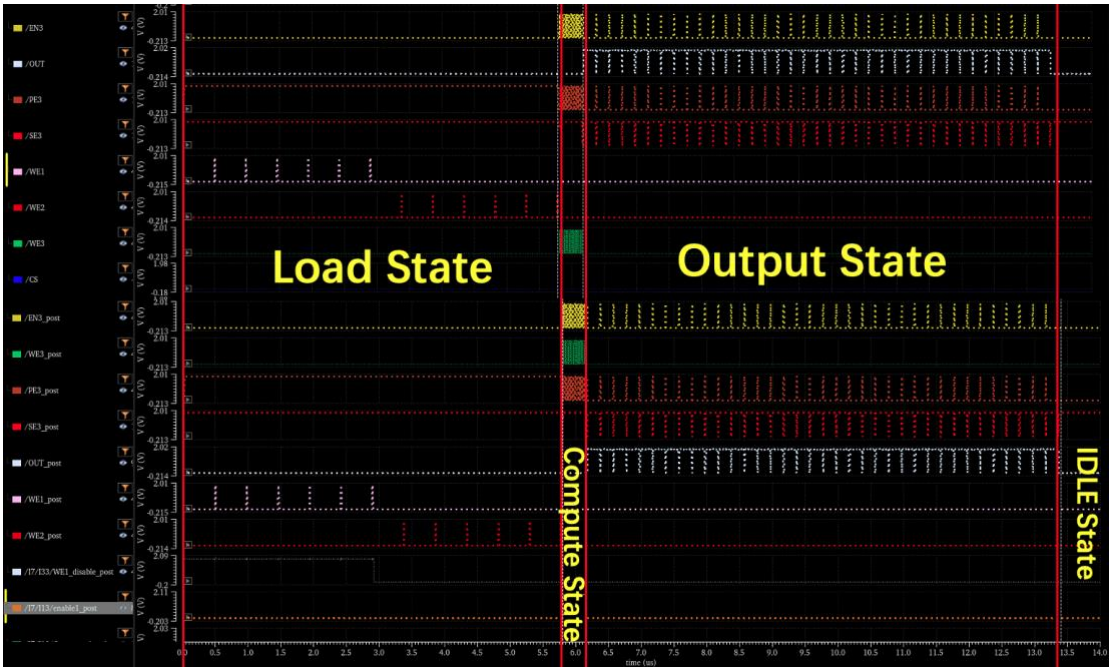


Figure 17. PISO Simulation Result

Schematic and Layout File Path (Top Cell):

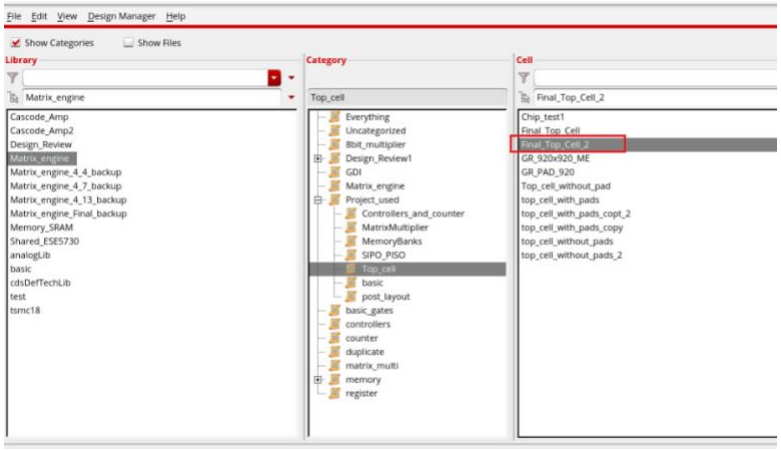


Figure 18. File Path

V. Table of Performance

Table 1. Performance and Compariso

Block	VCC	Review2 Speed	Review1 Speed	Review2 Power	Review1 Power
Matrix Multiplier	1.8	10ns	6ns	120.37n*36*1.8	93.632n*36*1.8
SIPO		~1ns	<1ns	5.1n*1.8	3.2n*1.8
PISO		~1ns	<1ns	4.3n*.18	2.8n*1.8
Input Memory		<8ns	4ns	30.6n*1.8	23n*1.8
Output Memory		<11ns	6ns	34n*1.8	26.3n*1.8

VI. Project Timeline and Task Assignment

Table 2. Project Timeline with Current Status, Task Explaination and Task Assignment

Task	Start Date	End Date	Task Description
Design Confirmation	2.26	3.5	Determine overall architecture, design specifications, and performance goals before starting (Both);
Block Schematic Design (Pre-layout Simulation)	3.6	3.26	Shun: Input/Output Memory Array (with decoder), FSM; Yue: SIPO, Matrix Multiplier, PISO;
Top-Cell Schematic Design (Pre-layout Simulation)	3.25	3.26	Schematic design and pre layout simulation to check the correctness(Both);
Block and Top-Cell Layout (DRC, LVS, RC extraction, Post-Layout Simulation)	3.27	4.14	Shun: FSM Simulation, Input/Output Memory Cell, FSM, Top Cell layout Yue: SIPO, PISO, Matrix Multiplier, FSM, Top Cell Layout
Layout Improvement	4.15	4.23	Check top cell post layout simulation and further optimize the existing layout to improve overall chip performance, focusing on reducing parasitic delay and enhancing timing closure.
Check and Ready for Tape-Out	4.24	4.30	Check each block (schematic, layout, performace), DRC/LVS clean, performance and ready for tape-out (Both).

Reflection:

During this phase of the project, we realized that our initial timeline underestimated the time required for both top-cell pre-layout simulation and individual block post-layout verification. These simulation steps turned out to be more time-consuming than expected, which significantly compressed the available time for layout implementation. Given that our chip integrates a relatively large and complex module, the layout workload remains highly demanding.

VII. Updated Measurement Plan

Serial Communication:

ESP32 (80MHz) (MCU) communicate with our chip through SPI with MOSI, SCLK and CS, corresponding to DATA, CLK and CS in our chip. When CS is switched from high to low, the SPI communication channel is opened. To load the data from MCU to SIPO (in chip), an enable signal (CSB) triggers the FSM change state from idle to load data. The calculation output from SIPO can be captured by logic analyzer/Oscilloscope for following verification.

Functional verification:

1. *Basic Arithmetic Correctness Test:* Send matrix data to the input of the chip through the MCU and compare the calculation results with theoretical calculation values to verify whether the matrix multiplication is correct using logic analyzer.

Performance Evaluation:

1. *Maximum Operating Frequency Test:* Gradually increase the clock frequency to test the stability of the chip at different frequencies and determine its maximum reliable operating frequency.
2. *Power Consumption Measurement:* Use a power measurement tool to measure operating power, compared to simulated power consumption.

Typical Application Scenarios

1. *Artificial Intelligence & Machine Learning:* Acceleration of small matrix operations in Convolutional Neural Networks (CNNs) to improve AI computational efficiency.
2. *Digital Signal Processing (DSP):* For image processing, edge detection, filtering, and other scenarios to increase processing speed.
3. *Embedded Computing & Low Power Applications:* Fast matrix computation in IoT devices to improve energy efficiency and reduce computing resources.

No Changes Requires.