

Multiplication and Accumulation (MAC) Datapath for Neural Networks - Final Report

Rahaan Gandhi - 653566752
Daood Shah - 668067960
Kajimusugura Hoshino - 660711348
Santan Settipalli - 678179543

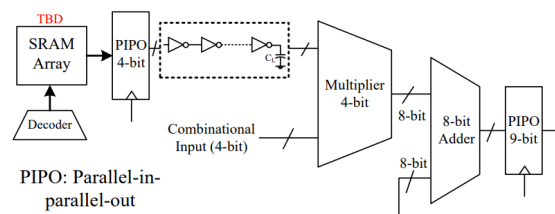
I. INTRODUCTION

For our project, we focused on Multiplication and Accumulation (MAC) Datapath for Neural Networks. Neural Network is a bunch of algorithms. The MAC is an important block in technology devices. The MAC performs the multiplication and accumulation process. It consists of a Multiplier, accumulator, and adder. Our design was designed to be fast and able to take in videos, images and large data. In our project, we focused on the business aspect of how fast and cheap we can design our system.

II. SYSTEM OVERVIEW

Figure 1, below, shows us the architectural system overview proposed for this project.

Figure: Architectural System Overview



The system is made of many complex and individual components like the Inverter, the **Multiplier**, **Full Adder** unit, 4-bit and 9-bit **PIPOs**, **Decoder**, and **SRAM** array. The system is built by connecting all of these components from one output to another's input.

For this project, our group of 4 has unanimously chosen to build and design for a low-power model instead of the high-performance model. We aimed to achieve the functionality as well as application techniques we have learned from this course to our MAC data path project. For the course of our project, all decisions made during the system design have been taken in the full presence of every member as well as their opinions.

III. COMPONENT-LEVEL DESIGN STYLES

For each system, component add description and figure to show component level design styles (this could be a transistor level or block level figure). Discuss if anything unique you try beyond classroom notes. Even if the ideas didn't work, it is useful to document them and it is learning as well.

This section of our report will give a slightly more in-depth description of our Main components that are listed below.

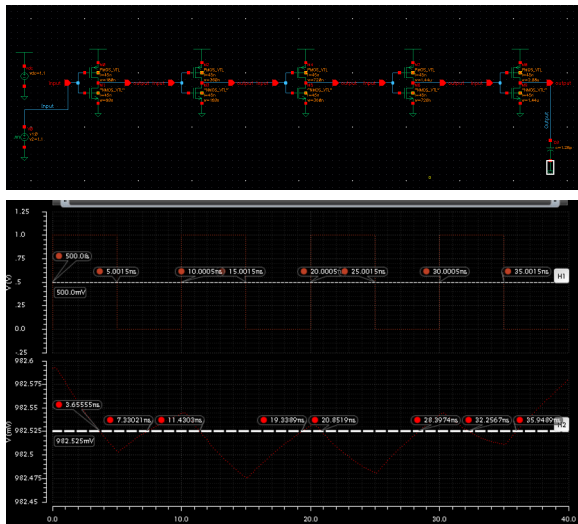
Inverter Chain

The Inverter chain designed here will contain 5 stages. Each stage will consist of a PMOS and an NMOS transistor. Our choice of using 5 stages was based on calculations performed and the results of calculations are below.

Table 1:

Number of stages (N)	Upsizing Factor (u)	Propagation delay (tp)
1	1	2.45ns
2	11.31	2.26ns
3	5.04	3.82ns
4	3.36	2.65ns
5	2.64	1.32ns

Figure: 5 stage Inverter chain



The inverter chain is crucial to the system since it can be used to reduce the delay

Adder

For this project, we chose a static CMOS adder instead of the Transmission Gate Adder since the former showed better performance as can be seen below. Please see Table 1 and Table 2 below.

Table2: Static CMOS Area VS Delay

Width	Delay for Sum scenario (111) [ns]	Delay for Carry scenario (110) [ns]
Nmos = 90nm Pmos = 180 nm	0.2944	0.227
Nmos = 180nm Pmos = 360 nm	0.1671	0.168

Table 3: Transmission Gate Adder Area VS Delay

Width	Delay for Sum Scenario	Delay for Carry Scenario
Nmos = 90nm Pmos = 180 nm	0.66 ns	0.33 ns
Nmos = 180nm Pmos = 360 nm	0.41	0.15 ns

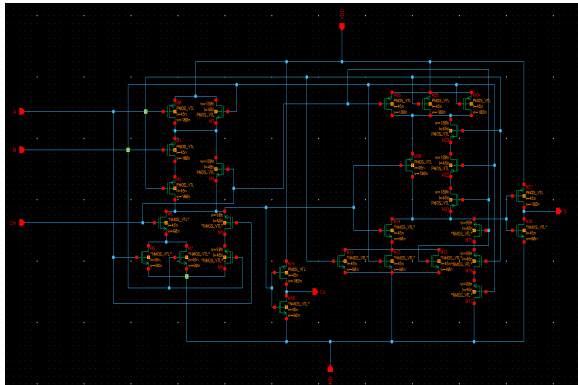


Figure 6: static CMOS 1 bit adder

Between Carry Select Adder and Ripple Carry Adder, we chose the former since the delay for the former was less than that of the latter.

Table 4: Supply volt VS WCSD for CSA

Supply Voltage	Worst Case Sum Delay (WCSD)
0.9	0.27925ns
1.0	0.2261ns
1.1	0.19304ns
1.2	0.15157ns

Table 5: Supply volt VS WCSD for RCA

660Supply Voltage	Worst Case Sum Delay
0.9	0.25235ns
1.0	0.22181ns
1.1	0.1821ns
1.2	0.15722ns

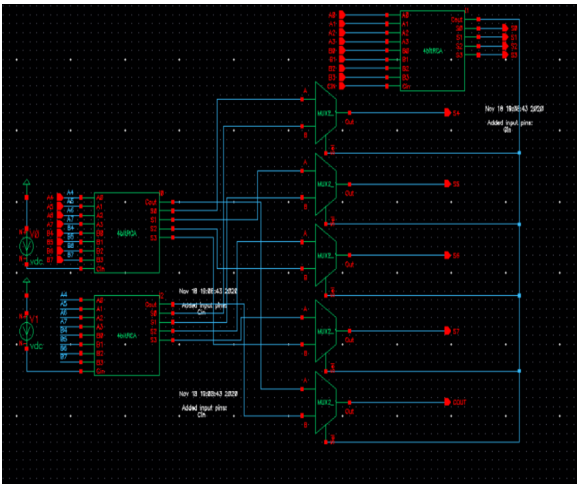
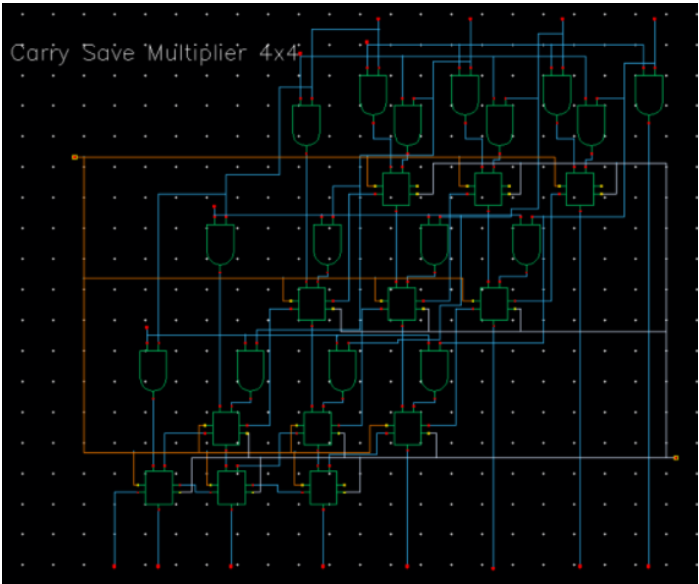


Figure 7: Carry Select adder

Multiplier

The multiplier is a combinational logic circuit used for the sole purpose of multiplication of binary digits and built by Adders.

Figure: Carry-Save multiplier 4x4



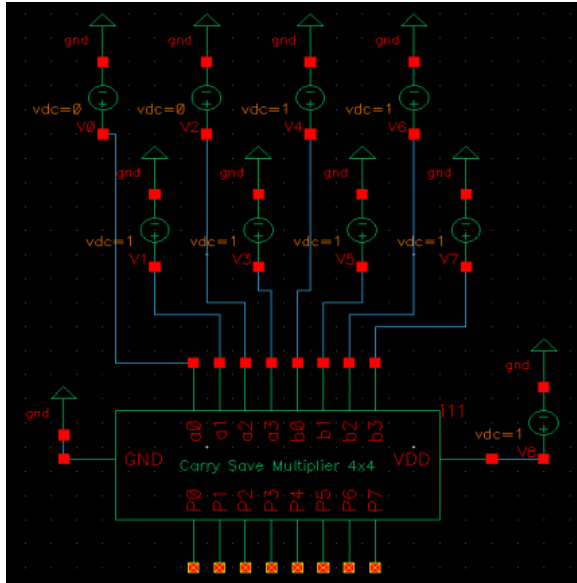


Figure: Carry save multiplier symbol

SRAM and SRAM array

The SRAM array is based on overlapping and connection of 32 by 32 6T SRAM cells.

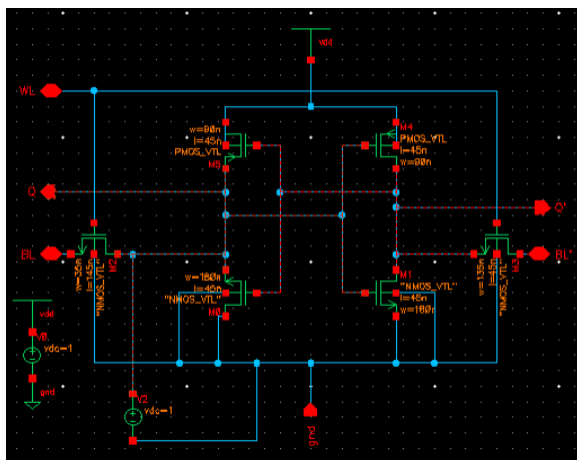
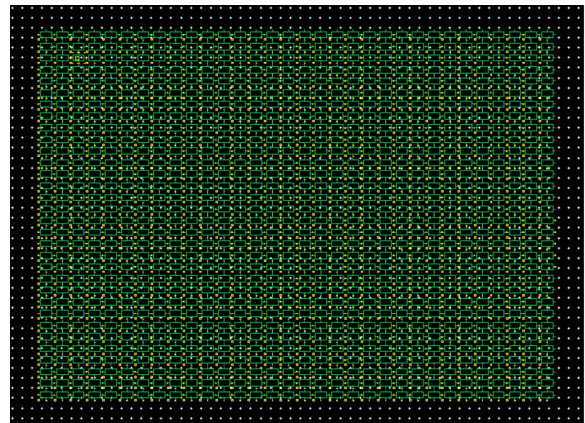


Figure: 6T SRAM cells

Figure: 32 by 32 SRAM unit



The SRAM is crucial for our system as it is the unit from where the data is fetched for our MAC data path operations. Our Designed 6T SRAM cell gave a perfect Read-to-Noise margin.

Figure: SNM butterfly curve

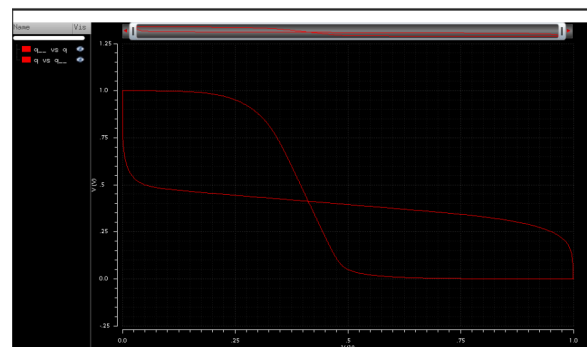
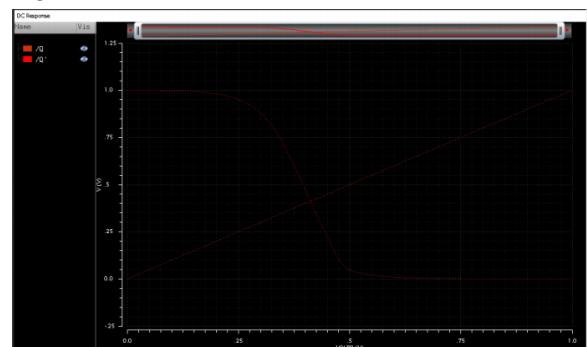


Figure: Q vs Q_bar results

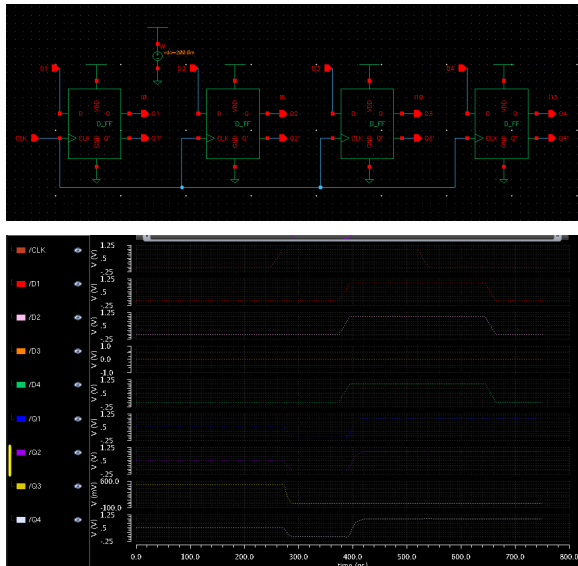


PIPO Register

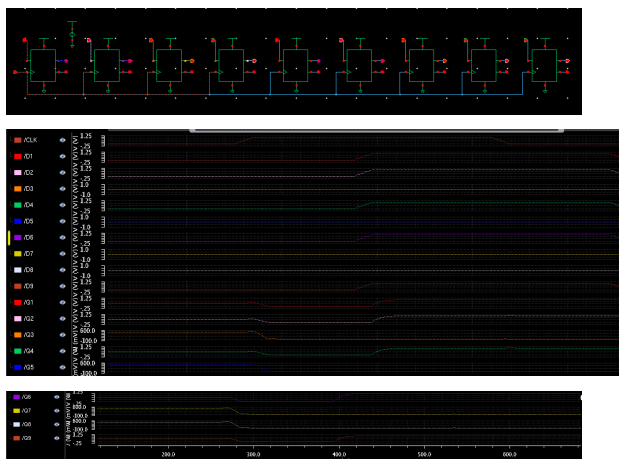
The Parallel in Parallel Out

Parallel In Parallel Out (PIPO) shift registers are the type of storage devices in which both data loading as well as data retrieval processes occur in parallel mode.

Figure: 4-bit PIPO



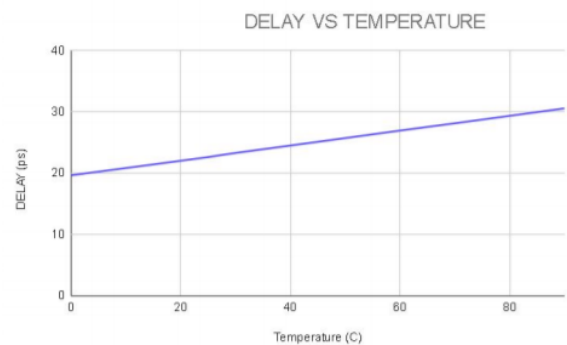
9-bit PIPO



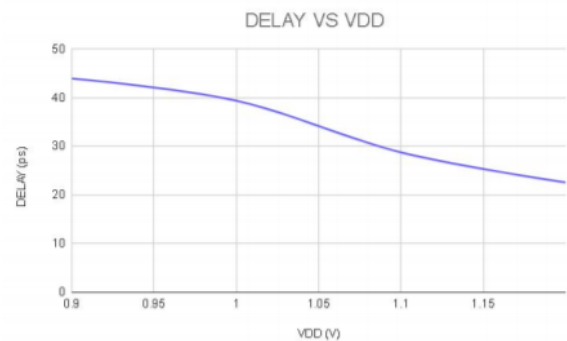
Decoder

We designed two decoders to work with our 32 by 32 SRAM array. They would output 4-bit results that would be directly inputted into the 4-bit PIPO. They were 5:32 row decoder and 3:8 column decoder.

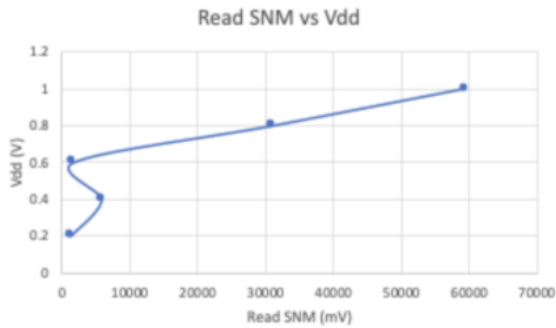
IV. SYSTEM CHARACTERIZATION



DelayVs Temperature Graph 4-Bit PIPO



Delay Vs VDD 4-bit PIPO



Read SNM Vs VDD



Write SNM Vs VDD

Row Decoder	Column Decoder	Register
21.63ps	800ps	131ps

Row Decoder Vs Column decoder vs Register

SRAM Cell	Multiplier	Adder
54ps	800ps	131ps

SRAM Cell Vs Multiplier Vs Adder Delays

In this section, discuss the essential results that you obtained. Start from a table to show system-level characteristics – power, clock frequency, design focus (low power or high performance), power breakdown among building blocks, which building block was performance or speed bottleneck. Then, for each building block show functional characteristics that show that the design is operating – design the most suited test

stimuli based on your understanding. Add a table that shows component level characteristics – power, performance, number of transistors, minimum power and VDD until when the design component works, maximum clock frequency, and VDD until when the design component works.

V. TASK ORGANIZATION

All tasks for every section or component of the project were equally divided among our 4 members. The Inverter Chain was led by Rahaan and was his responsibility. The Logic gate design was led by Daood and Kajimusugura who were responsible for all the logic gates. Everyone had an equal contribution in testing and debugging of Cadence designs. The CMOS adder was led by Rahaan and the transmission gate design and tests were done by Kajimusugura. Daood and Santan were responsible for designing and testing the RCA and CSA respectively. The 4 and 9-bit PIPO was completely built by Kajimusugura. Santan along with Daood was in charge of designing and testing the 6T SRAM cell and SRAM array. Rahaan was in charge of designing and testing the Multiplier. In the end, everyone was working together to put the system together and everyone has had equal contributions so far.

VI. CONCLUSIONS

After finishing this project we learned a lot about circuit design. We gained a lot of knowledge on how to use Cadence and then applied that to our project. We got to understand and work with low power and higher performance systems. We also got familiar with the debugging features of Cadence. We also learned about the different trade-offs in designing different circuits. What we believe will help us the most in the future is the experience and knowledge we gained from different designing and testing techniques as well as the knowledge of using cadence.