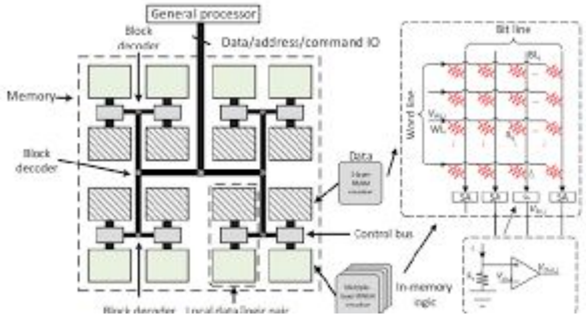
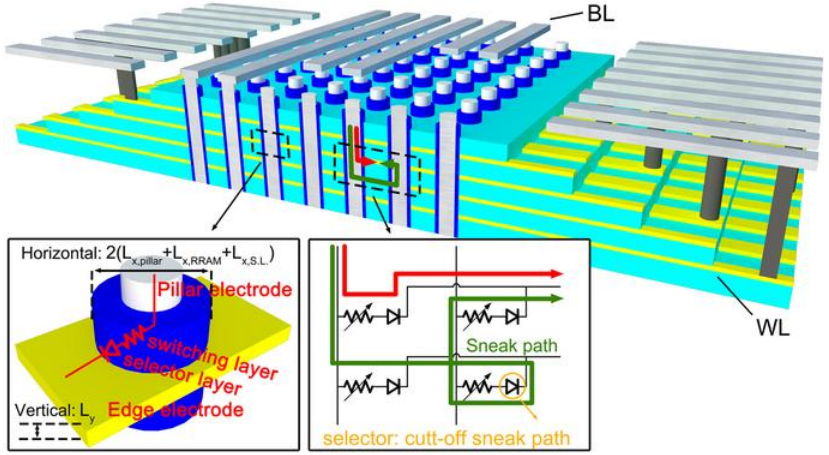
Contextualization/Design Check-in

By: Noah Mack, Olivia Price, Travis Jakl, Sam Burns
Advisor: Professor Henry Duwe III
Client: Professor Cheng Wang
SDMay25-19

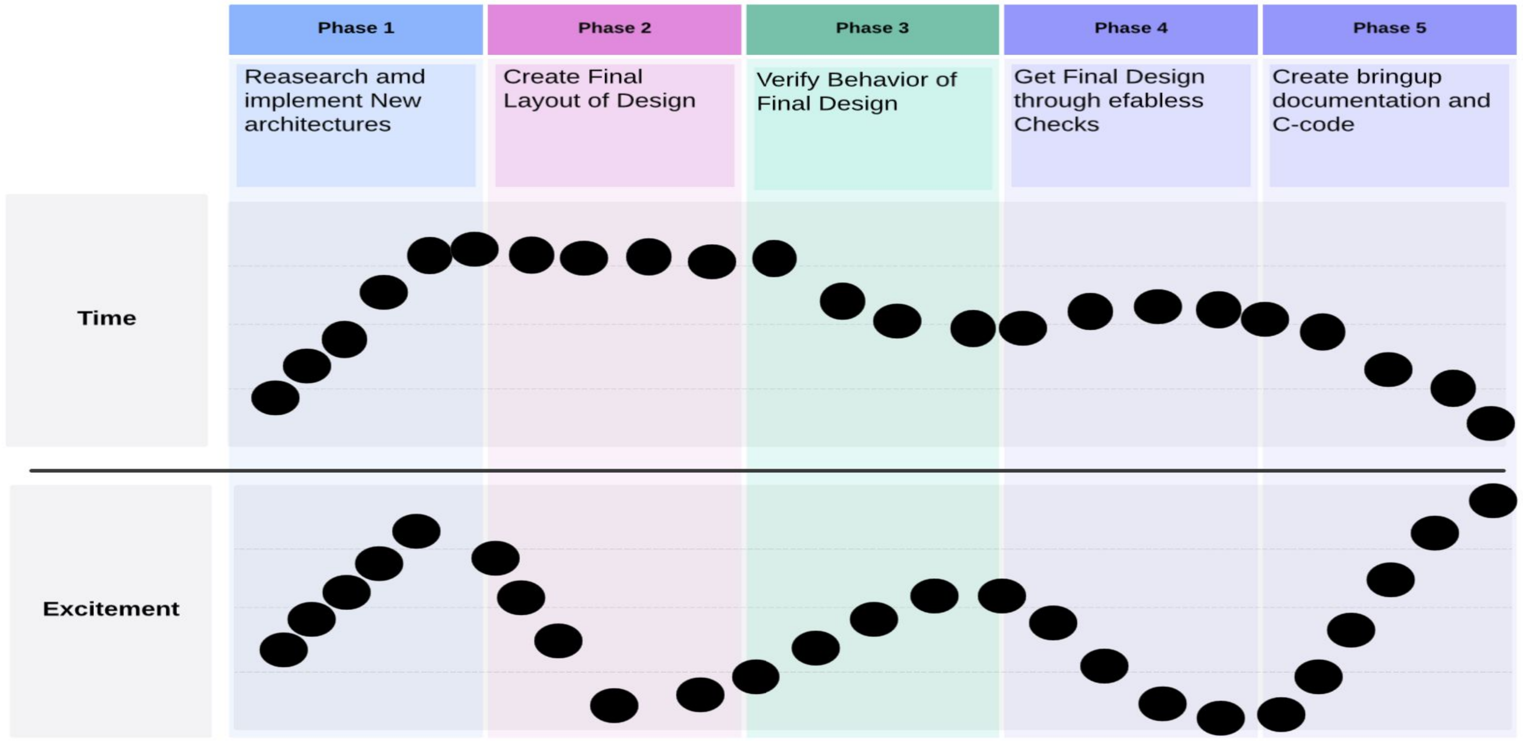
Project Overview

Our project focuses on creating a ReRAM-based compute-in-memory (CIM) test chip to improve matrix-vector multiplication efficiency in machine learning. Traditional CPUs struggle with the data bottleneck and energy cost of constant data movement, so our design incorporates multiple ReRAM architectures to enable parallel computation directly in memory. We'll fabricate the chip using the Skywater 130nm process, allowing ISU researchers and ChipForge club members to test and analyze different ReRAM designs. Alongside the chip, we'll provide documentation and C code for interfacing, helping research teams evaluate CIM's potential in low-power computing.

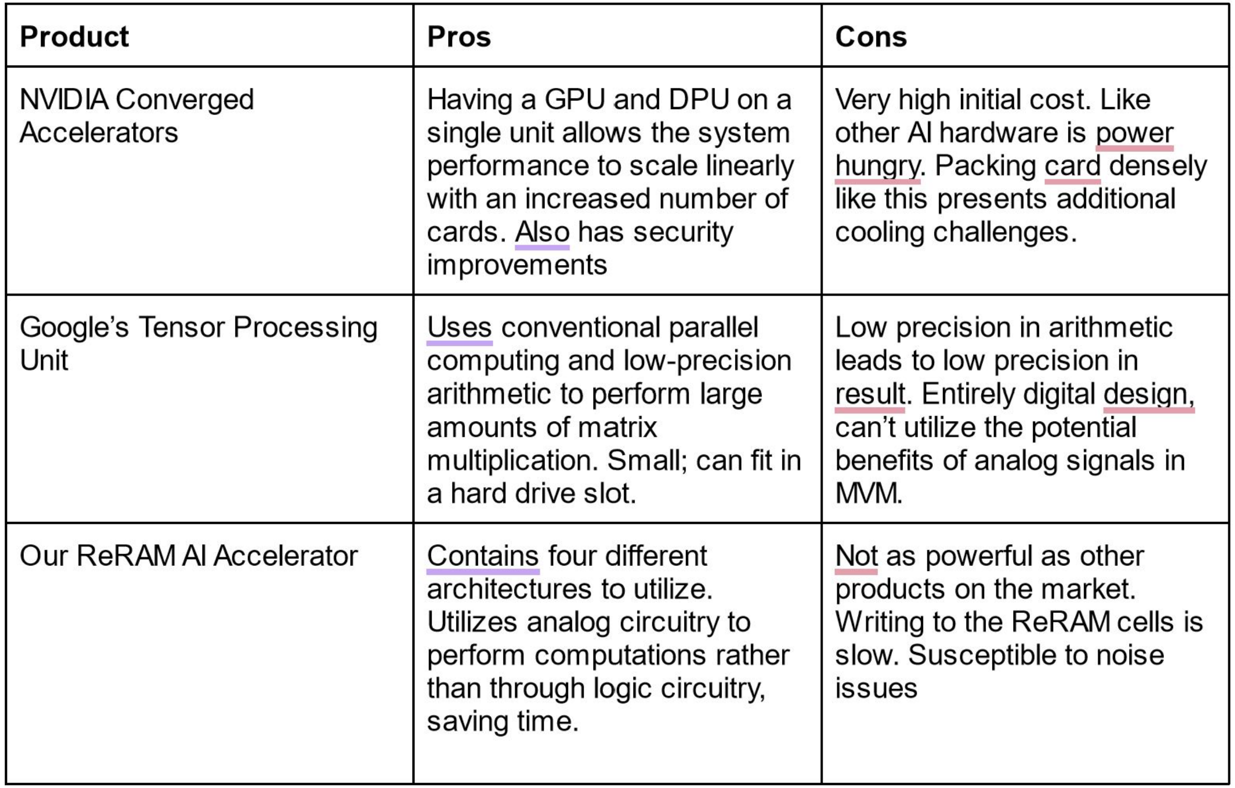




Artifacts (Journey Map)



Artifact (Pros/Cons Table)



Product	Pros	Cons
NVIDIA Converged Accelerators	Having a GPU and DPU on a single unit allows the system performance to scale linearly with an increased number of cards. Also has security improvements	Very high initial cost. Like other AI hardware is power hungry. Packing card densely like this presents additional cooling challenges.
Google's Tensor Processing Unit	Uses conventional parallel computing and low-precision arithmetic to perform large amounts of matrix multiplication. Small; can fit in a hard drive slot.	Low precision in arithmetic leads to low precision in result. Entirely digital design, can't utilize the potential benefits of analog signals in MVM.
Our ReRAM AI Accelerator	Contains four different architectures to utilize. Utilizes analog circuitry to perform computations rather than through logic circuitry, saving time.	Not as powerful as other products on the market. Writing to the ReRAM cells is slow. Susceptible to noise issues

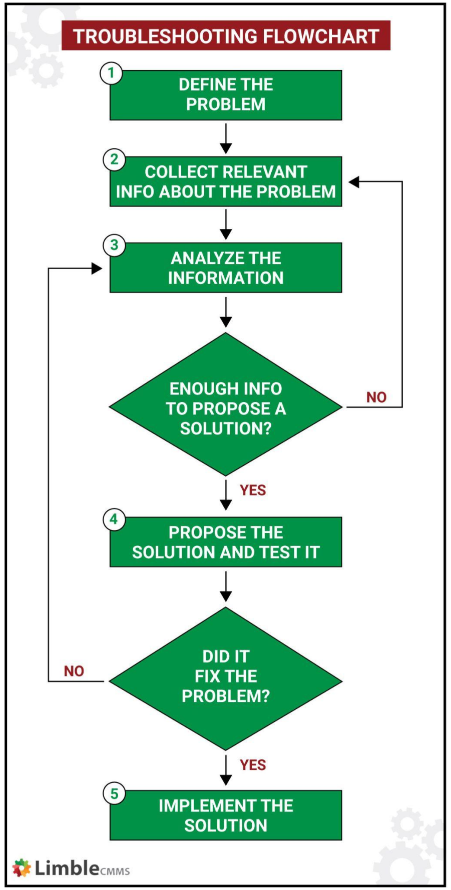
Artifact (Technical Complexity Analysis)

Our design presents a variety of technical challenges:

1. Since we are using open source software, none of our team members are familiar with the tools we are using. This provides a barrier of entry to getting started with our actual design.
2. ReRAM is an emerging technology, and there are very few opportunities available for fabrication of ReRAM chips. Also, there is little information about ReRAM usage for compute-in-memory applications, so we are exploring a new frontier.
3. We are including four different ReRAM architectures in our design, which of course increases the complexity of our design. However, this will likely help us down the line: if we find that one of our architectures doesn’t work as intended, there are still three others to test.
4. We are also integrating a number of components acting as the peripheral circuitry of the design. This includes S&H circuits, TIAs, DACs, and ADCs. While our team doesnt have direct experience with each of these components, they are very well studied in academia and in industry, so finding documentation about them shouldn’t be a struggle.

Addressing Client’s Needs

The current design solution effectively addresses user needs by providing a test vehicle that integrates multiple ReRAM architectures. Comprehensive documentation ensures that users can easily navigate the design process, troubleshoot issues, and perform testing.



TROUBLESHOOTING FLOWCHART
1 DEFINE THE PROBLEM
2 COLLECT RELEVANT INFO ABOUT THE PROBLEM
3 ANALYZE THE INFORMATION
ENOUGH INFO TO PROPOSE A SOLUTION?
NO
YES
4 PROPOSE THE SOLUTION AND TEST IT
DID IT FIX THE PROBLEM?
NO
YES
5 IMPLEMENT THE SOLUTION
LimbleCMMS

Economic

Improvement Over Existing Solutions: This solution improves upon traditional CPU-based matrix vector multiplication by minimizing energy consumption and speeding up computation through the use of computation in memory with ReRAM. By integrating different architectures into a single test vehicle, it allows for comparisons to be made that can lead to more efficient designs in future endeavors.

Drawbacks: Since no one has ever made a testing vehicle that can make matrix vector multiplication there are no hints or really any documentation that can guide us. Additionally, the complexity of managing noise within the ReRAM architectures could pose challenges during testing

Mitigation: The way to help some of these are to create robust testing procedures that can tell us if the architectures are working properly. To help the noise, we need to make sure components are far enough apart, look at each component individually and see if their precision needs to be changed.

Technical

Complexity: The internal complexity of this design comes from putting various architectures put together and integrating them in order to communicate with the microcontroller. The testing and calibration needs to be done externally after everything is made, so we need to create a user friendly interface so people can test the different architectures.

Expertise We Have Developed: With learning how to use the open source tools and creating testbenches, while also talking to other team members, we have developed skills to create two new architectures, testbenches, and hardware designs. With these expertise we have developed, we hope to overcome these complexities.

Conclusion

Design Updates:

- Journey Map
- Pros/Cons table
- Technical Complexity Analysis
- Addressing Client's Needs
- Economics
- Technical

With these complexities and challenges, we will overcome them with the plans portrayed in these slides and try to mitigate the risks.