Design of an 8-bit CAM using 9T SRAM cell Department of Electrical and Computer Engineering, Birzeit University

Aya Dahbour[1], Yasmeen Kamal[2], Meral Issa[3]
1201738[1], 1201146[2], 1200527[3]

***Abstract*—We introduce a revamped 8-bit CAM that harnesses 9T SRAM cells, known for their efficiency. Unlike conventional CAMs which are quick but energy-intensive, our model stands out for its energy-saving operation while maintaining high reliability. The innovative architecture of our 8-bit CAM uses the 9T SRAM's structure to enhance reading accuracy and writing capability, key for swift search tasks. Our custom design technique further refines the layout and boosts overall performance. Tests validate the enhancements in speed, power economy, and stability, marking a significant step toward energy-efficient, high-velocity CAMs ideal for data management and retrieval tasks.**

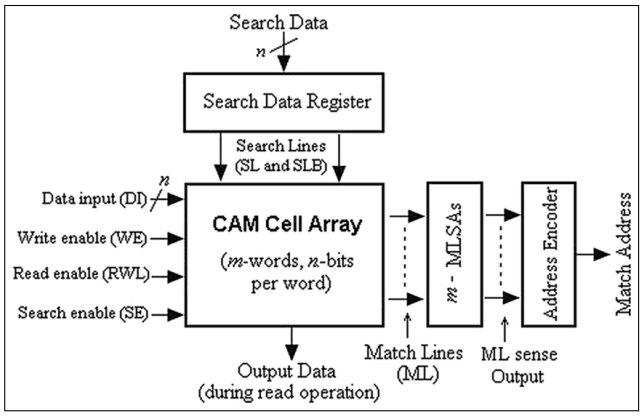

Fig. 1: Conceptual view of a CAM

I. Introduction

RAM stands as the cornerstone of a computer's quick data access capabilities, functioning without concern for the data's location. It's a fleeting form of memory, wiping its slate clean when power is lost. Among its types, DRAM and SRAM stand out; DRAM is the more common, necessitating continuous data refresh, while SRAM operates at greater speeds, suiting it for the swift demands of cache memory despite its higher cost.

Boosting the RAM in devices paves the way for smoother multitasking and faster data retrieval, underpinning its essential role in digital devices. Content Addressable Memory (CAM), akin to RAM in read-write capabilities, brings its own twist: it can search for data directly, comparing input data with its stored contents and pinpointing the matching addresses. This process, illustrated in Figure 1, triggers a match line when data aligns, and the match's location is encoded. In essence, CAM's read operations are the inverse of RAM's, as demonstrated in Figure 2.

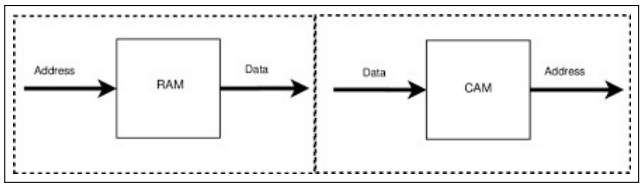

Fig. 2: CAM vs RAM [?].

RAM's data retrieval is sequential and location-based, which can become less efficient with larger datasets, potentially slowing access times. In contrast, Content Addressable Memory (CAM) revolutionizes this approach by allowing an instantaneous scan of all data in a single operation, greatly benefiting tasks requiring quick access such as complex database searches and network traffic optimization. Despite its speed, CAM's sophisticated circuitry leads to higher power consumption and a larger physical size, posing a significant design challenge. The need to balance CAM's rapid data access against these drawbacks is essential, as it requires careful planning to ensure system performance is optimized without undue compromise on energy efficiency and hardware scalability. This balancing act is critical in the development of high-performance computing systems, where speed cannot be achieved at the expense of sustainability and practicality.

II. DESIGN AND IMPLEMENTATION

In our 8-bit CAM project, we embraced a modular approach, assembling various components into a cohesive block for streamlined design and debugging, reminiscent of the architectural principle of simplifying design through abstraction. We utilized a 14nm CMOS library to refine timing and minimize signal skew, focusing on balanced transistor performance. Incorporating a 9T SRAM cell was key, chosen for its straightforwardness and efficiency, to boost the CAM's read capabilities. This melding of careful component selection and strategic design principles aimed to enhance the CAM's functionality within the 14nm process constraints.
The components are as follows:

A. 9T SRAM

The 9T SRAM design features a configuration with four input lines including the word line (WL), read line (RL), and two-bit lines (BL and BLB), allowing for efficient signal processing. It also boasts two output lines, QB and Q, which facilitate the smooth retrieval and transmission of data, showcasing the circuit's ability to effectively manage and convey information. [1].

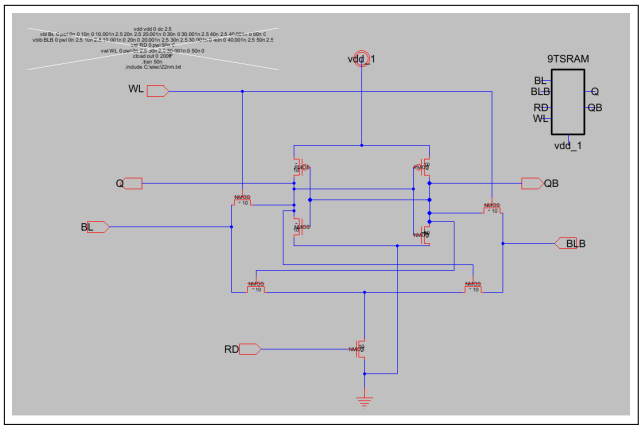

Fig. 3: The schematic of 9T SRAM circuit

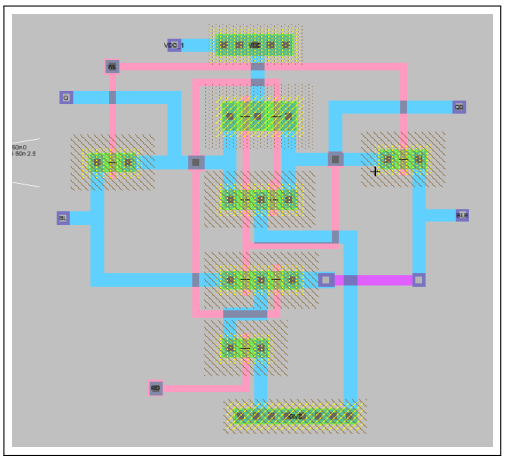

Fig. 4: The layout of 9T SRAM

B. 1-Bit CAM

The 1-bit CAM circuit was designed to include additional elements like inverters and pass gates alongside SRAM components to enable comparison operations, culminating in the generation of a match signal. This design is visualized in the schematic representation, illustrating the circuit's functionality and output process. As shown in figure 5 represents the schematic and Figure 6 represents the layout.

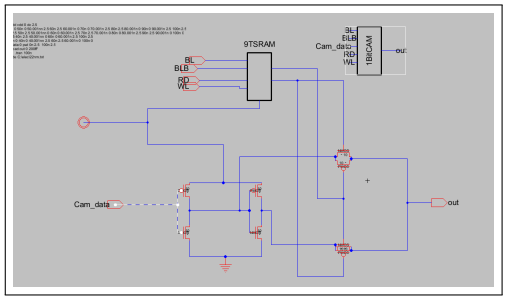

Fig. 5: The schematic of 1-Bit CAM circuit

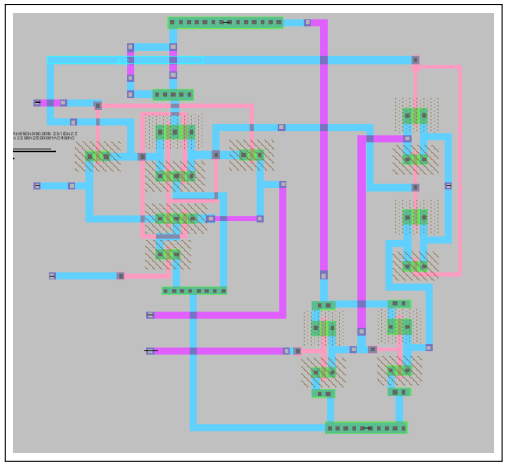

Fig. 6: The layout of the 1-bit CAM.

C. The 3x8 decoder

In our 8-bit CAM setup, the 3x8 decoder is key for pinpointing the exact 1-bit CAM cell we need out of eight, based on the search criteria. It cleverly interprets the input address and activates just the right output to connect us to the cell we're after. You can see how it fits into the bigger picture of the CAM design through the detailed schematic and layout we've put together, ensuring swift and precise data retrieval within the array. [2].

Inputs			Outputs							
A	B	C	Y_0	Y_1	Y_2	Y_3	Y_4	Y_5	Y_6	Y_7
0	0	0	0	1	1	1	1	1	1	1
0	0	1	1	0	1	1	1	1	1	1
0	1	0	1	1	0	1	1	1	1	1
0	1	1	1	1	1	0	1	1	1	1
1	0	0	1	1	1	1	0	1	1	1
1	0	1	1	1	1	1	1	0	1	1
1	1	0	1	1	1	1	1	1	0	1
1	1	1	1	1	1	1	1	1	1	0

Fig. 7: Truth Table for 3x8 Decoder

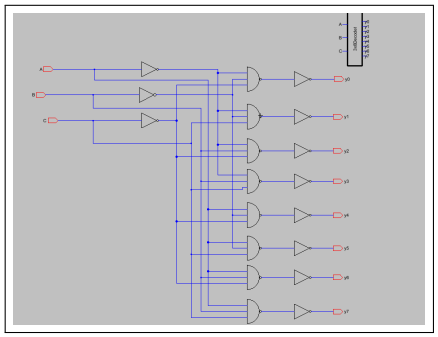

Fig. 8: The schematic of The 3x8 decoder

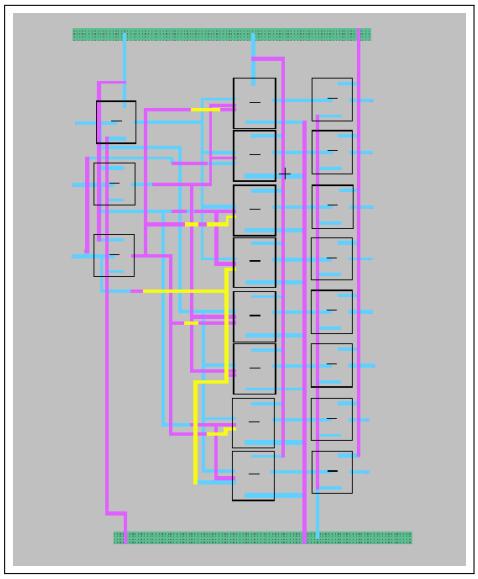

Fig. 9: The layout of The 3x8 decoder

D. The 8-input NAND

For our 8-bit CAM, we're leveraging an 8-input NAND gate to link the eight 1-bit CAM cells to the final output, streamlining the process. This setup ensures the individual outputs from the CAM cells are effectively combined, culminating in the final output signal, a crucial step illustrated in our schematic and layout diagrams. [3].

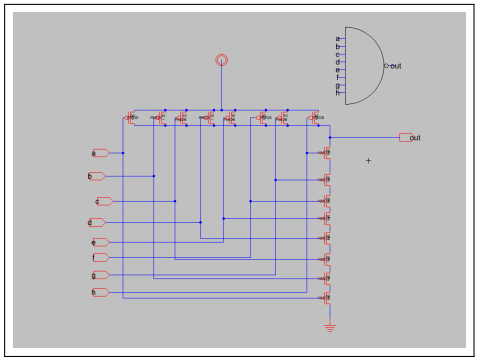

Fig. 10: The schematic of 8-input NAND

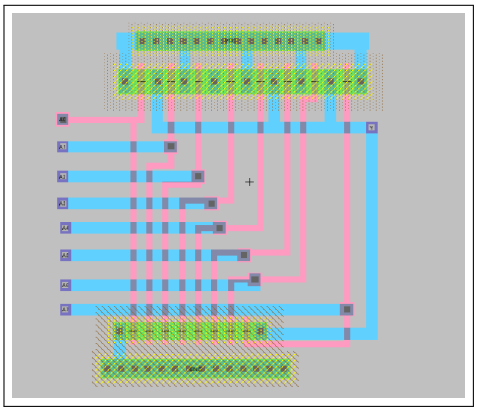

Fig. 11: The layout of the 8-input NAND

E. 3-inputs NAND gate

The 3-input NAND gate was a fundamental building block in crafting a more extensive circuit destined to be integrated into the 8-bit Content Addressable Memory (CAM). Its inclusion in the design proved invaluable, enabling a higher degree of abstraction during both the debugging and simulation stages. The schematic and layout are provided below for reference: [4].

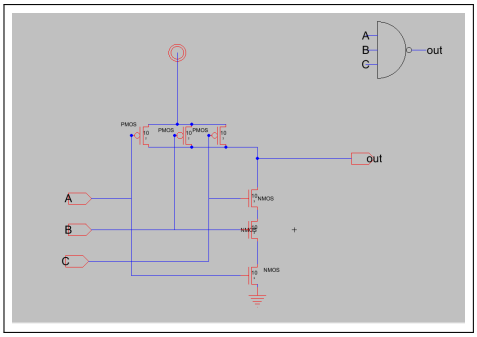

Fig. 12: The schematic of 3-input NAND

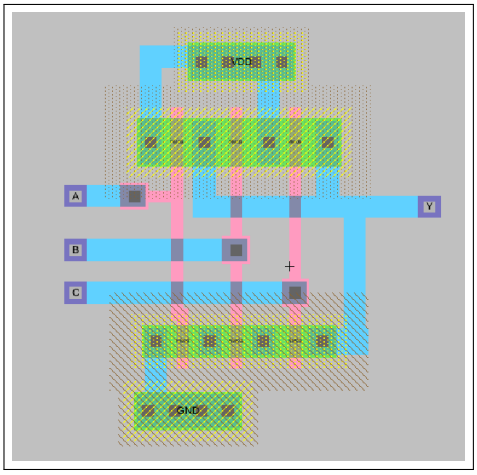

Fig. 13: The layout of the 3-input NAND

F. Invertor

As the final touch to our setup, we added an inverter, a simple yet pivotal component. Its design details, including the schematic and layout, are laid out in the accompanying visuals, highlighting its key role in rounding off our circuit design.

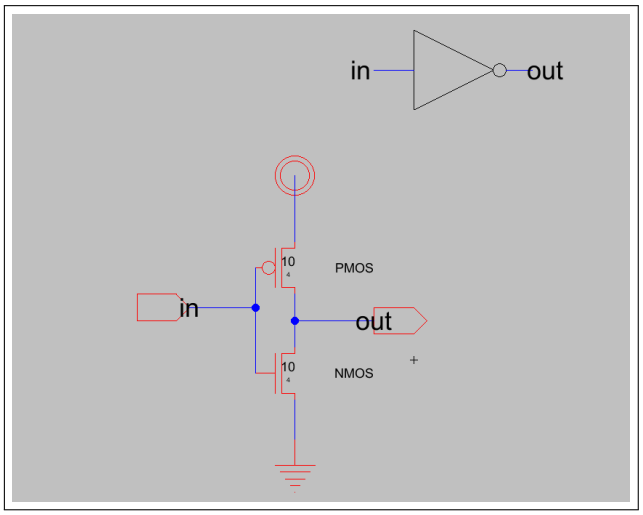

Fig. 14: The schematic of Inverter

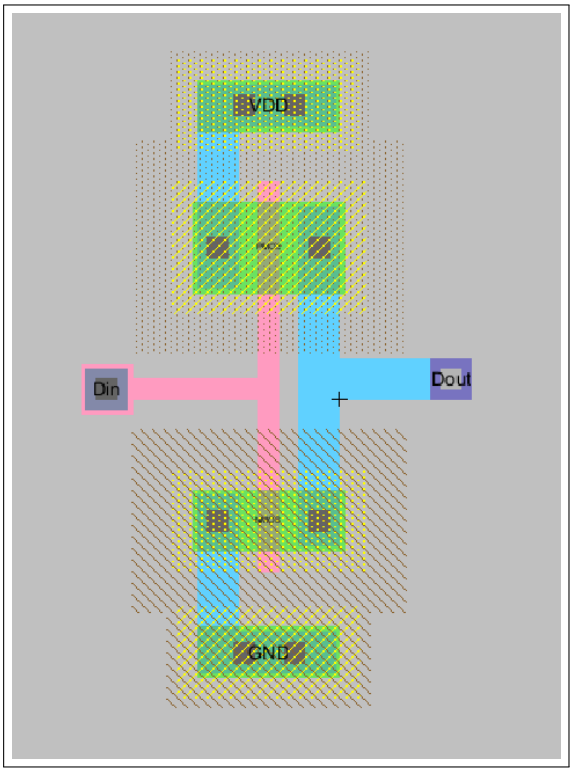

Fig. 15: The layout of Inverter

G. 8-Bit CAM

Putting together our 8-bit CAM, we drew on the robustness of 9T SRAM technology. The design is a tapestry of components – a decoder, 1-bit CAM cells, NAND gates, and an inverter, all working in concert as shown in the detailed figure. This orchestration of parts is what enables our 8-bit CAM to execute its functions with precision, facilitating smooth data retrieval and matching tasks. [5].

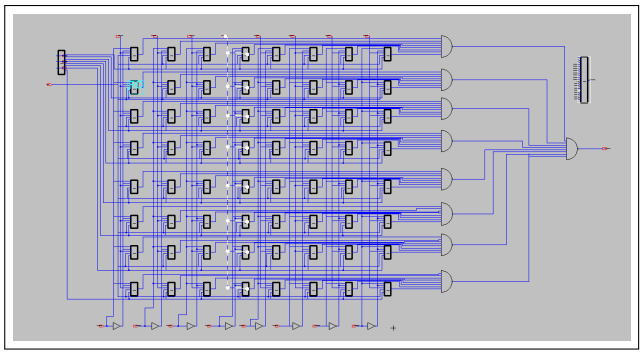

Fig. 16: The schematic of the 8-Bit CAM using 9T SRAM

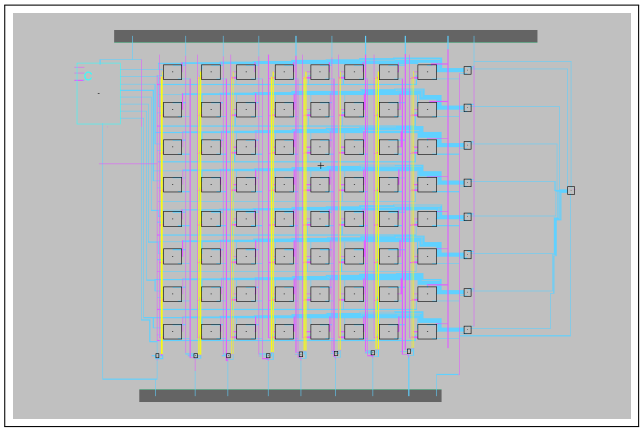

Fig. 17: The layout of the 8-Bit CAM using 9T SRAM

III. RESULTS

In the world of SRAM, the green light for writing new data comes down to what's happening with the write line, which folks often call the WE line. Think of this line as the boss of the operation, deciding when the memory cells are ready to take in new info (that's when the line is up high, signaling "go ahead, write") or when they're just there to hold onto what they've already got (that's when the line is low, saying "nope, we're just reading today"). This setup is pretty smart because it keeps the data safe and sound, making sure nothing changes unless it's supposed to. At the same time, it's not all locked down; you can still update things when you need to, which is a neat trick for keeping everything running smoothly and making sure the info stored is both safe and up-to-date.

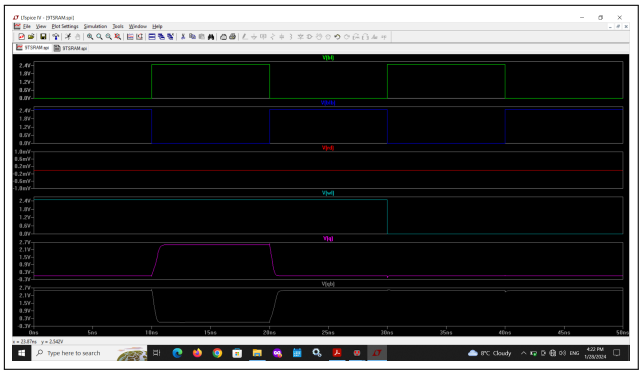

Fig. 18: The simulation of 9T SRAM

Running the 1-bit CAM through its paces turned out pretty well. The screen lit up with the right signals, showing that the puzzle pieces of input and stored data fit together perfectly. This thumbs-up result is a solid nod to the CAM's knack for doing its job, hitting the bullseye on what we were aiming for. It's a bit like a pat on the back, confirming that this piece of tech is not just dependable but also sharp at sifting through its memory to find exactly what we're looking for. It kind of sets the stage for how useful this can be when we roll it out into the real deal. [6].

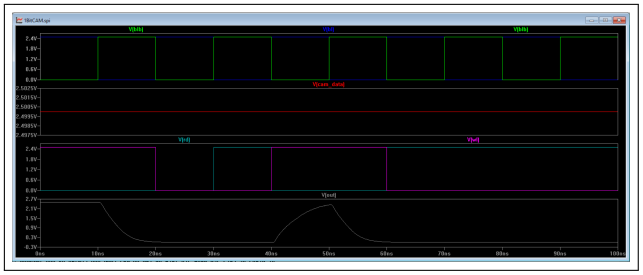

Fig. 19: The simulation of 1bit cam

Looking at the decoder's output, there's a clear rule that it should only light up one location each time it accomplishes something. And, as expected, the diagram supports this, demonstrating that there is always one bright spot in the mix. It's like a continuous yes all around, indicating that everything is running smoothly.

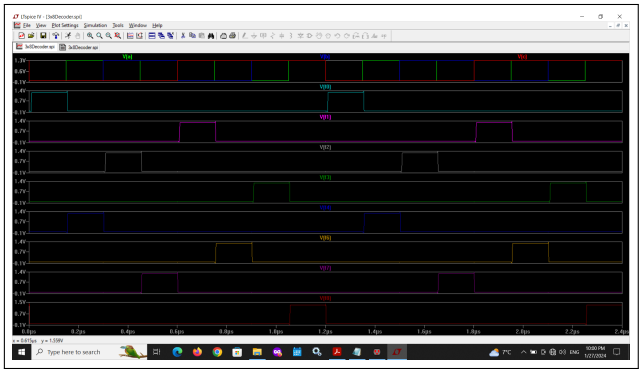

Fig. 20: The simulation of 3x8 decoder

The inverter circuit simulation, using 22nm technology, confirmed the expected behavior, producing outputs that logically complemented the inputs. This simulation is crucial for understanding digital circuit behavior, aiding in design optimization, and ensuring reliable signal processing, by examining input-output relationships and propagation delays to enhance circuit efficiency and reliability.

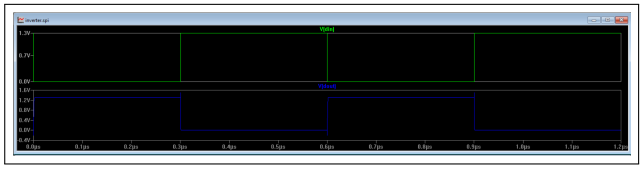

Fig. 21: The simulation of Invertor (22nms)

Upon analysis, it was clear that the NAND gate's output maintained true, indicating logic 1, and only remained at a voltage level of 2 volts when all eight input signals were false, indicating logic 0. In all other cases, the output consistently displayed logic 0, according to the truth table demonstrating its capacity to perform logical conjunction while outputting the logical negative of the combined inputs. Simulating an 8-bit input NAND gate is a valuable tool for engineers, providing insights into its operational features and usefulness. They may fine-tune circuit designs to provide dependable logic operations inside digital systems, hence enhancing overall performance.

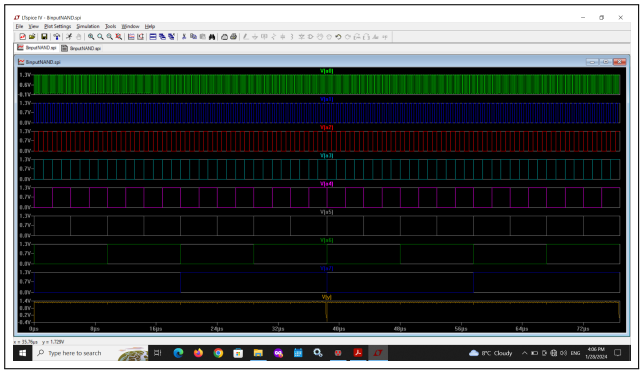

Fig. 22: The simulation of 8-input Nand

Examining the operation of a 3-input NAND gate through simulation offers valuable insights into its functionality and logical behavior. Engineers can enhance circuit designs and guarantee dependable performance in digital systems by studying input combinations and their corresponding outputs.

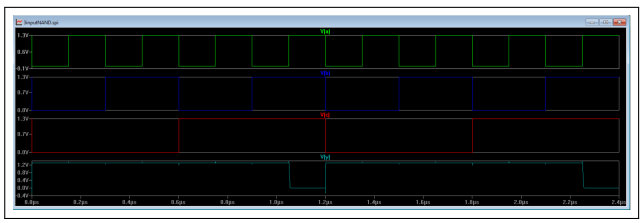

Fig. 23: The simulation of 3-input Nand

The 8-Bit Content-Addressable Memory (CAM) simulation, which includes a 9T SRAM, represents the CAM circuit's behavior and efficiency, with a focus on rapid and accurate data retrieval via search keys. The benefits of using 9T SRAM include lower leakage power and increased stability. This simulation gives information for improving the 8-bit CAM's design.

In the field of CAM simulation, the 8-bit CAM inside an integrated circuit architecture is investigated. This simulation seeks to reproduce CAM's key properties using digital logic components, allowing for parallel comparisons and rapid data retrieval. It shows potential for applications that need efficient content matching and address-based data access inside bigger systems.

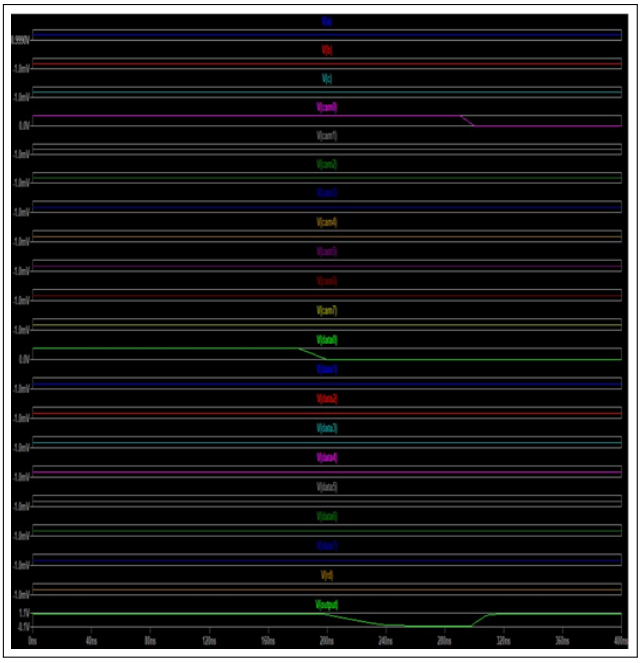

Fig. 24: The simulation of the 8-Bit CAM using 9T SRAM

IV. AREA, POWER, AND DELAY OPTIMIZATION

We adopted the 22nm technology node and harnessed electric Binary to craft the layout specifically for this node. The 22nm node corresponds to the distance between the source and drain of the transistor, referred to as the channel length. This choice brings several advantages, including a smaller form factor, quicker switching speeds, and reduced power consumption. Following extensive research and experimentation, it was determined that for the optimal design of 9T SRAM, an inverter size of 3 delivers the best results. In this particular setup, the access transistors should be sized at four times the inverter's dimensions (12), while the two NMOS transistors need to be twice the inverter's size (2). Furthermore, the NMOS transistor connected to the Read line should be sized proportionately between the access transistor and the other two NMOS transistors.

	Delay	Power	Area
9T-SRAM	41ps	2.98uW	9.3mm^2
1bCAM	59ps	24.54uW	28.5mm^2
8bCAM	3.48 e4 ps	4.56uW	0.0096*U^2

TABLE I: Design Comparison: Power,Delay and Area

V. CONCLUSION

In conclusion, our journey led us to the successful design and implementation of the 8-bit Content Addressable Memory (CAM) circuit. Throughout this process, we remained steadfast in our commitment to optimizing power efficiency, compact area utilization, and minimal delay, all while upholding peak performance standards. Our presentation included detailed layouts and clear schematic representations for each circuit component, all rigorously validated through comprehensive simulation results. The resulting circuit not only fulfills our specifications but also excels in terms of enhanced power efficiency, efficient use of space, and rapid response times, firmly establishing itself as a robust solution for efficient data retrieval and matching operations.

REFERENCES

[1] KUMAR, G., AND YADAV, S. K. Design and performance analysis of sram cells. *IUP Journal of Electrical & Electronics Engineering 14*, 1 (2021).

[2] MISHRA, J. K., SRIVASTAVA, H., MISRA, P. K., AND GOSWAMI, M. Analytical modelling and design of 9t sram cell with leakage control technique. *Analog Integrated Circuits and Signal Processing 101* (2019), 31–43.

[3] PAL, S., GUPTA, V., KI, W. H., AND ISLAM, A. Transmission gate-based 9t sram cell for variation resilient low power and reliable internet of things applications. *IET Circuits, Devices & Systems 13*, 5 (2019), 584–595.

[4] ROY, C., AND ISLAM, A. Characterization of single-ended 9t sram cell. *Microsystem Technologies 26* (2020), 1591–1604.

[5] WANG, B., NGUYEN, T. Q., DO, A. T., ZHOU, J., JE, M., AND KIM, T. T. A 0.2 v 16kb 9t sram with bitline leakage equalization and cam-assisted write performance boosting for improving energy efficiency. In *2012 IEEE Asian Solid State Circuits Conference (A-SSCC)* (2012), IEEE, pp. 73–76.

[6] WANG, B., NGUYEN, T. Q., DO, A. T., ZHOU, J., JE, M., AND KIM, T. T.-H. Design of an ultra-low voltage 9t sram with equalized bitline leakage and cam-assisted energy efficiency improvement. *IEEE Transactions on Circuits and Systems I: Regular Papers 62*, 2 (2014), 441–448.