Chat-GPT aware for Full-Custom Design 8-bit CAM using 9T SRAM for Digital Integrated Design Department of Electrical and Computer Engineering, Birzeit University

Ahmad Eiss[1], *Saja Shareef*[2], *ShereenIbdah*[3]

1202384@*student.birzeit.edu*[1], 1200901@*student.birzeit.edu*[2], 1200373@*student.birzeit.edu*[3]

***Abstract*—The primary objective of this project is to develop an 8-bit Content Addressable Memory (CAM) using a 9T SRAM configuration in the Electric Generic 14nm CMOS Library. The design implementation focuses on achieving maximum power savings and minimizing the area required for the CAM cell, specifically during the write operation within the 14nm process technology. This design approach aims to strike a balance between low power consumption and low delay, optimizing the performance of the CAM. The key aspects of this project include the utilization of 9T SRAM, CAM cell design, low power considerations, XOR operation, and efficient read and write operations.**

I. Introduction

Random Access Memory (RAM) is a critical hardware component within a computer's main memory, providing direct access to the Central Processing Unit (CPU). Its primary function is to enable data read and write operations at specific memory addresses. RAM also serves as a temporary storage space for actively running programs, facilitating efficient processing by the CPU. It is important to note that RAM is a volatile memory type, meaning that data stored within it is lost when power is turned off. To retrieve data from RAM, the memory address is used as input, and the corresponding content word is obtained through multiple cycles until the desired data is accessed.

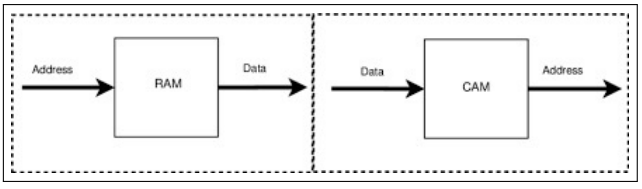

Fig. 1: CAM vs RAM [7].

In contrast, Content Addressable Memory (CAM), also known as Associative Memory, shares similarities with RAM in terms of read and write operations targeting specific addresses. However, CAM incorporates additional searching functionalities. It conducts parallel comparisons between the input data word and the entirety of its memory, seeking matches and generating a list of addresses where the data word is located [2]. This process involves the activation of a matching line when the stored data matches the incoming data. An encoder is then utilized to output an encoded representation of the match location, typically represented by log2 w bits. While CAM offers faster search times compared to RAM, its implementation necessitates extra circuitry, resulting in increased silicon area and power consumption.

The implementation of CAM can be achieved using SRAM schematics, as both CAM and SRAM perform similar operations. Various SRAM configurations, including 6T, 8T, 9T, and 10T, can be employed based on the specific requirements and project specifications. For this project, the design and implementation will focus on an 8-bit CAM utilizing a 9T SRAM configuration. The 9T SRAM design presents advantages in terms of power consumption, stability, and area efficiency. The schematic representation of the 9T SRAM will serve as the foundation for the CAM implementation, providing the necessary framework for subsequent stages of development and optimization.

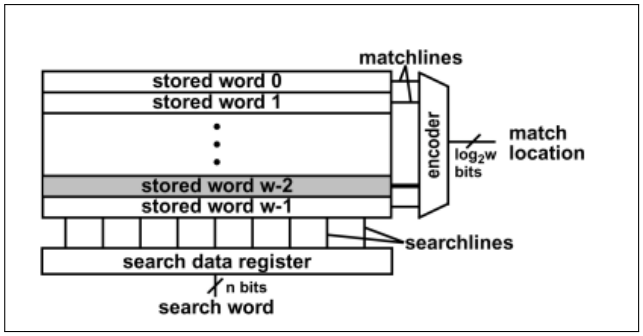

Fig. 2: Conceptual view of a content-addressable memory [8].

II. DESIGN AND IMPLEMENTATION

In this project, the design was implemented using a 14nm CMOS library to ensure accurate timing and mitigate signal

skew. Our primary focus was on achieving balanced rise and fall times for both NMOS and PMOS devices. This required careful consideration of factors such as the intrinsic delay and output resistance of each transistor. Through extensive simulation and testing, we determined that a ratio of approximately 2:1 between PMOS and NMOS devices yielded optimal results in terms of achieving equal rise and fall times within the 14nm process.

As an initial step in the design process, we integrated a 9T SRAM cell. This type of memory cell is renowned for its simplicity, reliability, and rapid operation, making it widely adopted in digital systems. The incorporation of the 9T SRAM cell aimed to enhance the read performance of the overall CAM design. Figure 1 provides a schematic representation of the 9T SRAM, showcasing its structural configuration and interconnections.

By incorporating these design considerations and leveraging the benefits of the 9T SRAM cell, our objective was to optimize the performance and functionality of the CAM design while adhering to the constraints imposed by the 14nm CMOS process technology.

The components are as the following:

A. 9T SRAM

The 9T SRAM circuit, as depicted in the accompanying figure, encompasses four input lines: the word line (WL), the read line (Rd), and two additional bit lines (BL and BLB). This configuration enables the circuit to receive input signals and process them accordingly. Furthermore, the 9T SRAM circuit provides two output lines: QB and Q, through which the processed data is conveyed. These output lines facilitate the retrieval and transmission of the stored information within the circuit [1].

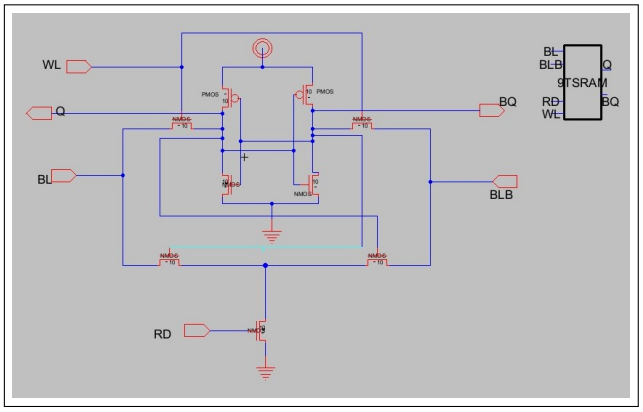

Fig. 3: The schematic of 9T SRAM.

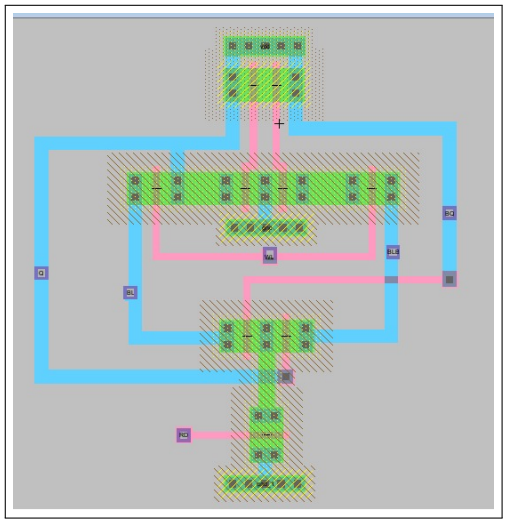

Fig. 4: The layout of 9T SRAM

B. 1-Bit CAM

Then,The 1-bit Content-Addressable Memory (CAM) circuit was implemented by incorporating additional inverters, pass gates, and SRAM components for the purpose of comparison, which ultimately resulted in the generation of a match signal [9]. Figure 4 depicts the schematic representation of the 1-bit CAM circuit.

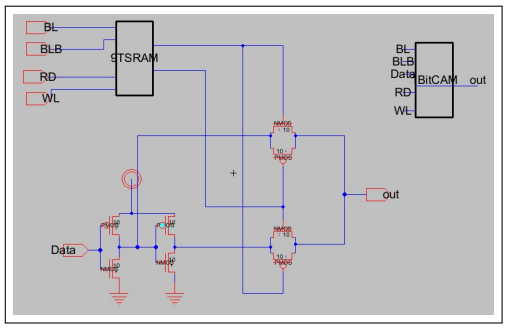

Fig. 5: The schematic of 1-Bit CAM circuit

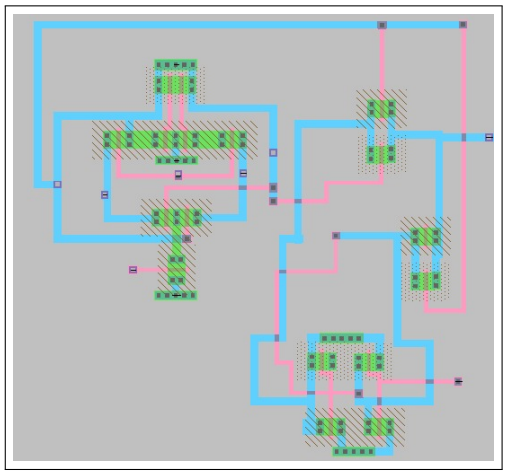

Fig. 6: The layout of the 1-bit CAM.

C. 3-input NOR

The 3-input NOR gate schematic was an essential component used in this project. This logic gate accepts three input signals and performs the logical NOR operation. It generates an output signal based on the input combination, producing a low output only when all three input signals are high, and a high output for any other input combination [10].

Input			Output
A	B	C	Y
0	0	0	1
0	0	1	0
0	1	0	0
1	0	1	0
1	0	0	0
1	1	1	0
1	1	0	0
1	1	1	0

TABLE I: 3-input NOR Truth Table [4]

D. The 3x8 decoder

In the 8-bit CAM implementation, the 3x8 decoder selects the appropriate 1-bit CAM cell from eight options during the search. It decodes the input address and activates the corresponding output line for the desired cell. The decoder's schematic and layout show its configuration in the CAM design. By using the 3x8 decoder, the CAM efficiently accesses the desired data in the 8-bit CAM array [3].

Inputs			Outputs							
x	y	z	D0	D1	D2	D3	D4	D5	D6	D7
0	0	0	1	0	0	0	0	0	0	0
0	0	1	0	1	0	0	0	0	0	0
0	1	0	0	0	1	0	0	0	0	0
0	1	1	0	0	0	1	0	0	0	0
1	0	0	0	0	0	0	1	0	0	0
1	0	1	0	0	0	0	0	1	0	0
1	1	0	0	0	0	0	0	0	1	0
1	1	1	0	0	0	0	0	0	0	1

TABLE II: 3x8 decoder Truth Table [5]

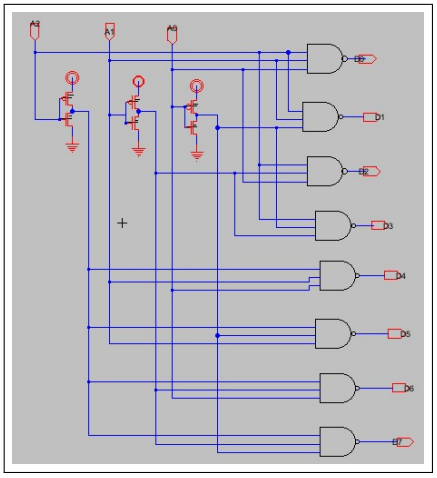

Fig. 7: The schematic of The 3x8 decoder

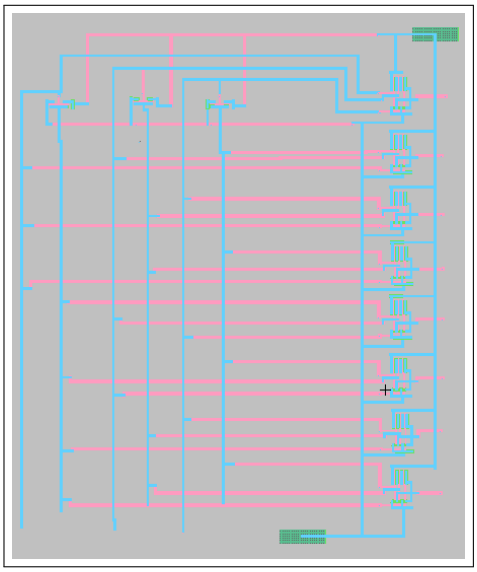

Fig. 8: The layout of The 3x8 decoder

E. The 8-input NAND

The 8-input NAND gate will be utilized in the implementation of the 8-bit CAM when connecting the 8 blocks of the 1-bit CAM to the final output. By incorporating the 8-input NAND gate, the CAM system can efficiently combine the outputs of the individual 1-bit CAM blocks to produce the final output [6].

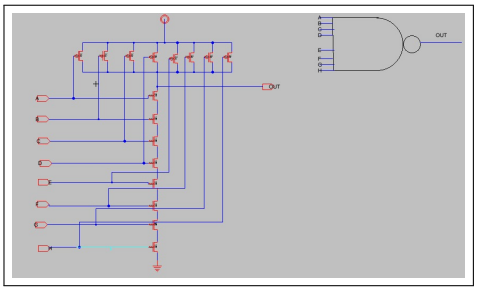

Fig. 9: The schematic of 8-input NAND

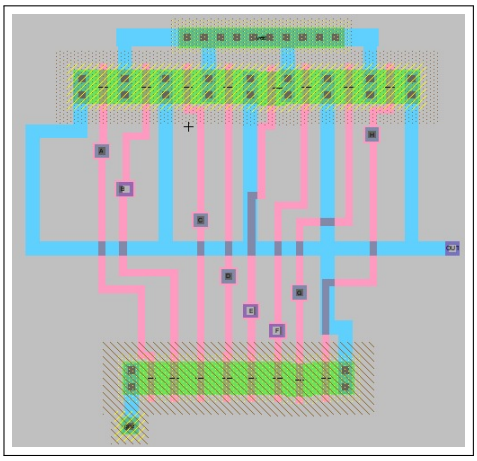

Fig. 10: The layout of the 8-input NAND

F. Invertor

Finally, the inverter circuit serves as the last component required for the 8-bit CAM circuit. The schematic and layout representations of the inverter component are depicted in the accompanying figures.

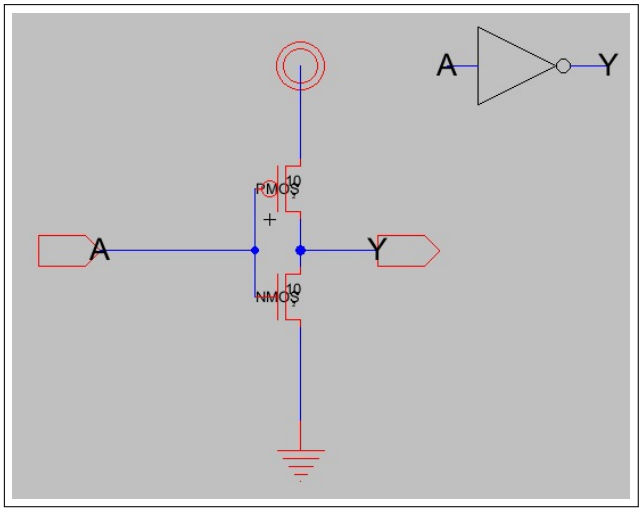

Fig. 11: The schematic of Inverter

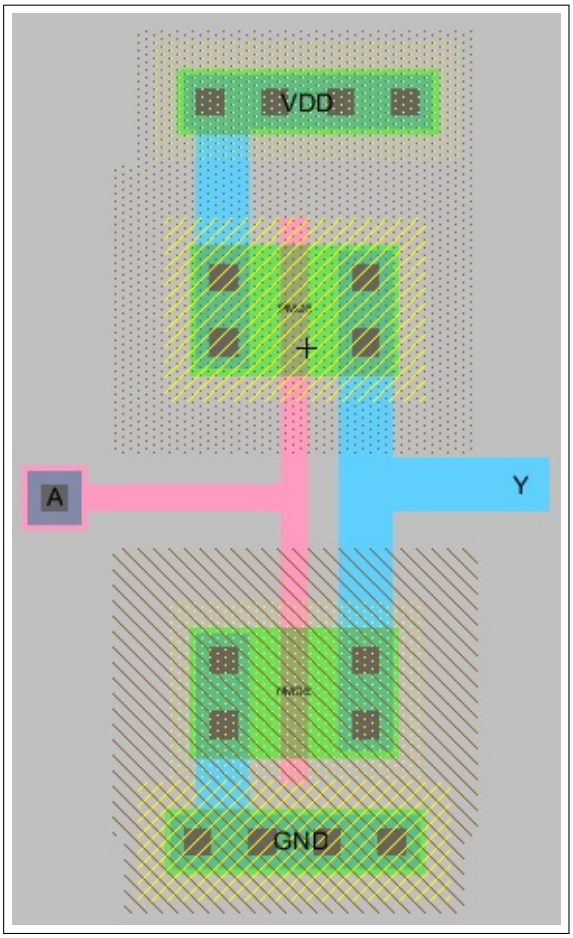

Fig. 12: The layout of Inverter

G. 8-Bit CAM

The circuit of the 8-Bit CAM using 9T SRAM was implemented as depicted in the accompanying figure. To achieve this, a decoder, 1-bit CAM, NAND gates, and an inverter were utilized in the design and implementation process to determine the output of the 8-Bit CAM.

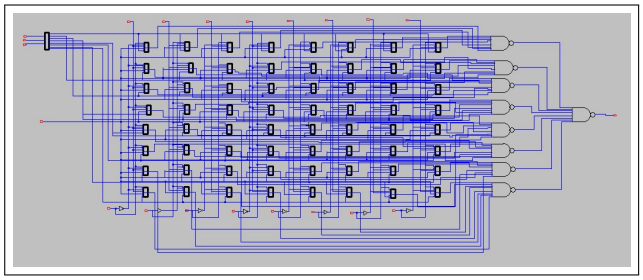

Fig. 13: The schematic of the 8-Bit CAM using 9T SRAM

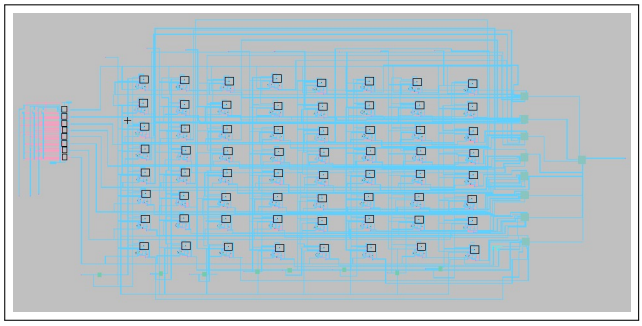

Fig. 14: The layout of the 8-Bit CAM using 9T SRAM

III. RESULTS

In SRAM, the write operation is controlled by the write line (WL line). When the write line is high, indicating a write-enable state, the memory cells can receive and store new data, allowing for data updates. Conversely, when the write line is low, the memory cells are in a read-only mode, preventing any modifications to the stored information. This mechanism ensures the stability and integrity of the data, protecting against unintended changes. By controlling the write line, SRAM provides a selective and controlled approach to managing and updating data, maintaining the reliability of stored information.

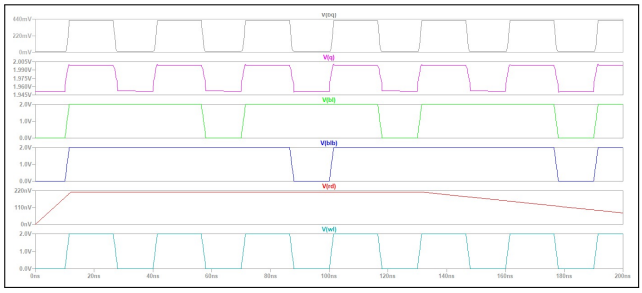

Fig. 15: The simulation of 9T SRAM

The simulation of the 1-bit CAM yielded positive results, with the output indicating a successful match between the

input data and the stored data. The obtained outcome verifies the efficient functioning of the CAM, as the expected result aligns with the desired outcome. This outcome affirms the reliability and effectiveness in accurately retrieving and identifying the stored data, further reinforcing its potential for practical applications.

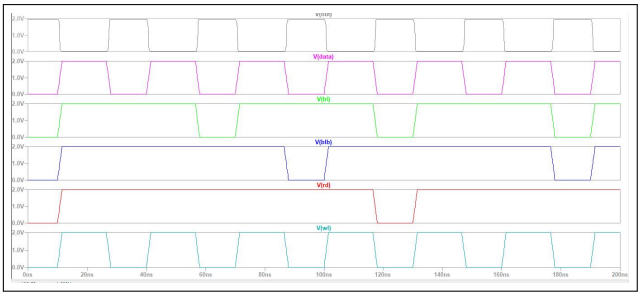

Fig. 16: The simulation of 1bit cam

In 3-input NOR simulation. The output of the gate was observed to be true when all three input signals were false, while the output was false when at least one of the inputs was true. This result is consistent with the truth table, validating its ability to perform logical negation on multiple input signals. The successful simulation further solidifies its usability in various logic circuit designs and reinforces its reliability in logical operations.

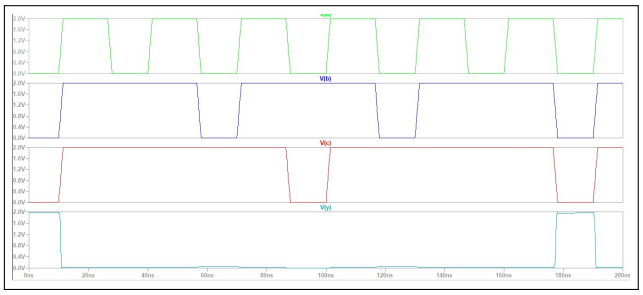

Fig. 17: The simulation of 3-input NOR

In the output of the decoder, as we know the output of it must be one active in each case, so the figure shows that there is just one active in the output in every case. So, the results are true.

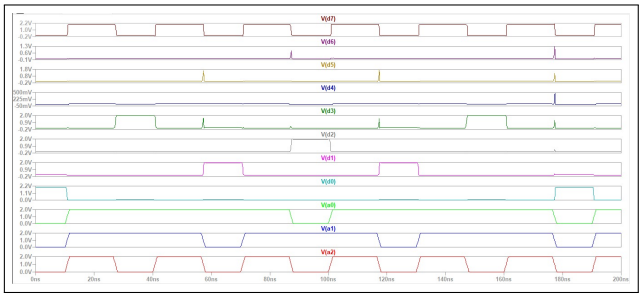

Fig. 18: The simulation of 3x8 decoder

The simulation of the inverter circuit, using a 22nm technology node, demonstrated the expected behavior and functionality. The output of the inverter was observed to be the logical complement of the input signal, adhering to the principles of logic inversion.

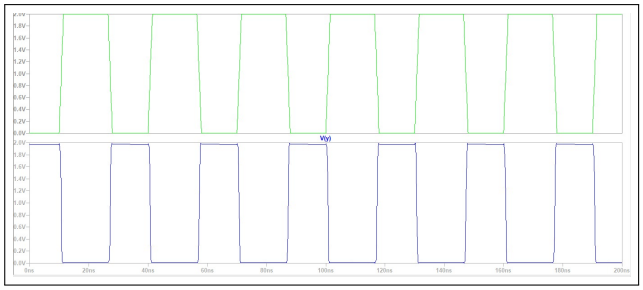

Fig. 19: The simulation of Invertor (22nms)

The output of the NAND gate was observed to be true (logic 1), which corresponds to a voltage level of 2 volts only when all eight input signals were false (logic 0). In all other input combinations, the output was false (logic 0). This outcome aligns with the truth table, confirming its ability to perform logical conjunction and produce the logical negation of the combined inputs.

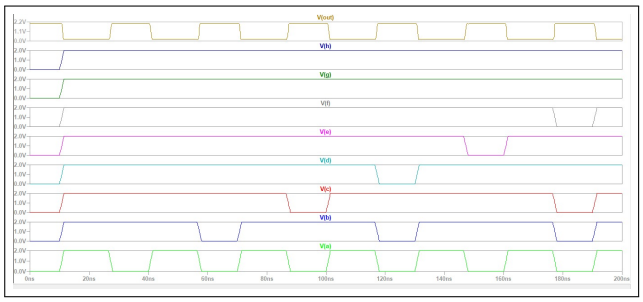

Fig. 20: The simulation of 8-input Nand

The 8-Bit Content-Addressable Memory (CAM) using a 9T SRAM is a comprehensive computational analysis that models the functionality and performance of the CAM circuit implemented with a 9T static random-access memory (SRAM). This simulation aims to validate the behavior and efficiency of the 8-bit CAM design in terms of its ability to quickly and accurately retrieve data based on a given search key.

The 9T SRAM implementation in the simulation offers certain advantages, such as reduced leakage power and improved stability compared to conventional 6T SRAM cells. The simulation provides valuable insights into the performance characteristics of the 8-bit CAM, enabling designers to optimize its architecture and configuration for specific application requirements.

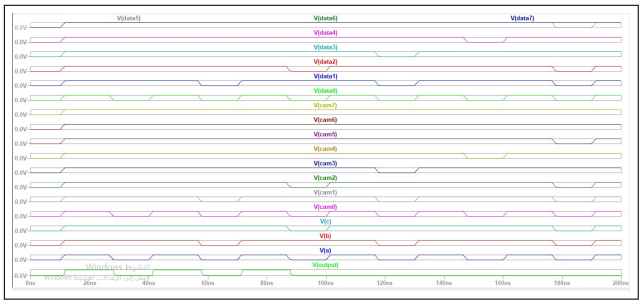

Fig. 21: The simulation of the 8-Bit CAM using 9T SRAM

IV. AREA, POWER, AND DELAY OPTIMIZATION

While, in terms of area, and cost optimization, we have taken steps to make the circuit more efficient and cost-effective without affecting its output quality or performance. For instance, let's consider the 8-Input NAND gate shown in the figure. Initially, it required seven 2-NAND gates, totaling 28 gates. However, we have made changes to reduce the number of gates needed while maintaining the circuit's functionality and performance.

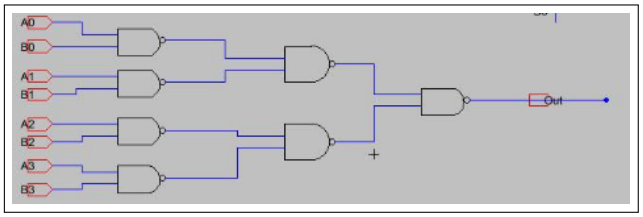

Fig. 22: 8-NAND gate

So, in hence to reduce both areas and cost we designed the 8-NAND gate using 8 parallel PMOS And 8-Series NMOS total of 16 Transistors.

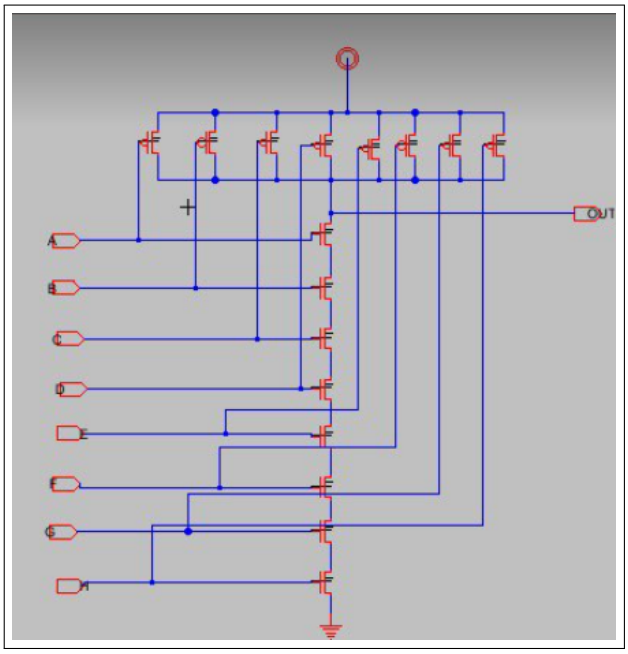

Fig. 23: 8-NAND gate using PMOS And NMOS

	Delay	Power
9t-SRAM	25.62u	5ps
1bit-CAM	21ps	43.789µW
8bit-CAM	38ps	54.168µW
Transistor Weight 10	0.052ns	1.26µW
Transistor Weight 2	0.521	0.13µW
Transistor Weight 12	0.27ns	1.22µW

TABLE III: Output of changing the size of the CMOS.

V. CONCLUSION

In conclusion, The design and implementation steps of the 8-Bit CAM circuit were thoroughly examined and presented in a comprehensive manner. The layout and schematic circuit for each component utilized in the final circuit were clearly illustrated. Moreover, specific techniques were employed to optimize power consumption, area utilization, and delay in the 8-Bit CAM, ensuring that these optimizations did not compromise the performance or searching capabilities of the Content Address Memory (CAM). Notably, simulation results of various circuits were carefully obtained, meticulously analyzed, and effectively communicated.

REFERENCES

[1] Design of High Speed 9T SRAM Cell at 16 nm Technology with Simultaneous Read-Write Feature — https. https://ieeexplore.ieee.org/document/9929643. [Accessed 7-Jul-2023].

[2] Difference between Random Access Memory (RAM) and Content Addressable Memory (CAM) - GeeksforGeeks — geeksforgeeks.org. https://www.geeksforgeeks.org/difference-between-random-access-memory-ram-and-content-addressable-memory-cam. [Accessed 6-Jul-2023].

[3] Logic diagram of a 3-to-8-line decoder using NOR, NOT gates, enable input — Intro. to Logic Design — youtube.com. https://www.youtube.com/watch?v=Nz3$_ya - iDiE$. [$Accessed6 - Jul - 2023$].

[4] NOR Gate in Digital Electronics - Javatpoint — javatpoint.com. https://www.javatpoint.com/nor-gate-in-digital-electronics. [Accessed 8-Jul-2023].

[5] FAHAD, E. Decoder, 3 to 8 Decoder Block Diagram, Truth Table, and Logic Diagram — electroniclinic.com. https://www.electroniclinic.com/decoder-3-to-8-decoder-block-diagram-truth-table-and-logic-diagram. [Accessed 7-Jul-2023].

[6] KUMAR, M., HUSSAIN, M., AND PAUL, S. Performance of a two input nand gate using subthreshold leakage control techniques.

[7] MUJAHID, O., ULLAH, D. Z., HAFEEZ, A., AND FOUZDER, T. *Design Exploration of LH-CAM with Updating Mechanism*. 05 2020, pp. 477–486.

[8] PAGIAMTZIS, K., AND SHEIKHOLESLAMI, A. Content-addressable memory (cam) circuits and architectures: A tutorial and survey. *IEEE journal of solid-state circuits 41*, 3 (2006), 712–727.

[9] S.A, S., SWEDHA, A., AND NAVEEN, D. Survey of content addressable memory. 1516.

[10] TAN, M., LENTARIS, G., AND AMARATUNGA, G. Device and circuit-level performance of carbon nanotube field-effect transistor with benchmarking against a nano-mosfet. *Nanoscale research letters 7* (08 2012), 467.