Design 8-Bit Content Addressable Memory (CAM) Using 9T SRAM

Nour Saleh Rabee'
Bachelor of Computer Engineering
Birzeit, Ramallah, Palestine
rabee.nour123@gmail.com

Yousef Moahammad Shamasneh
Bachelor of Computer Engineering
Birzeit, Ramallah, Palestine
shamasnehyousef173@gmail.com

Saif Al-deen Shakour Battah
Bachelor of Computer Engineering
Birzeit, Ramallah, Palestine
Saifbattah25@gmail.com

***Abstract*—Content-Addressable Memories (CAMs) serve as high-speed hardware search engines, outperforming algorithmic approaches for search-intensive applications. CAMs combine conventional semiconductor memory, typically SRAM, with specialized comparison circuitry to enable single-clock-cycle search operations. CAMs find wide application in tasks such as packet forwarding and packet classification in Internet routers. [1] Efficiently reducing power consumption without compromising speed or memory density presents a significant design challenge for CAMs due to the parallel active circuitry involved. [2] In this paper, we present the design of an 8-bit CAM using a 300 nm process file, employing nine-transistor SRAM cells.**

By going to the circuit level and providing detailed schematics and layouts for each component, we aim to demonstrate the feasibility and practicality of implementing the proposed 8-bit CAM design.

Keywords—CAM memory, content addressable memory, 9T SRAM, Associative Memory, Associative Storage

I. Introduction

Content Addressable Memory (CAM) It is also known as associative memory or associative storage, is a type of memory that compares the input data with the preloaded contents of the CAM block and generates a given output depending on the kind of CAM. This kind of memory provides a distinct speed advantage over RAM in systems requiring quick address comparison or retrieval. Typical RAM applications utilize a counter to load the addresses into the RAM at a rate of one address per clock cycle. The data from the RAM block would then be compared with the expected data using XOR logic. When a match is found, the address becomes the valid output. The number of clock cycles required for this whole process is the number of addresses. For most applications, this would take more than one clock cycle. Because CAM can take the input (data) and compare it with all of the preloaded contents in the CAM array simultaneously, it can execute the entire lookup in a single clock cycle. This speed is ideally suited for network applications such as address lookups and filtering, packet encryption, and firewalls.[4]

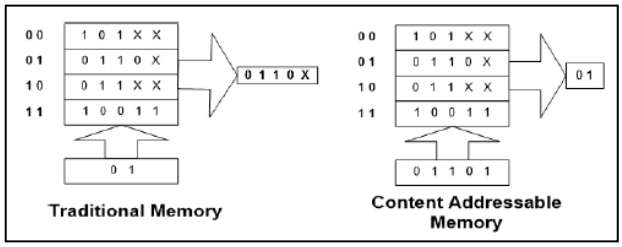

Fig. 1: RAM Vs CAM in read operation

However, the speed of a CAM comes at the cost of increased silicon area and power consumption, two design parameters that designers strive to reduce. As CAM applications grow, demanding larger CAM sizes, the power problem is further exacerbated. Reducing power consumption, without sacrificing speed or area, is the main thread of recent research in large-capacity CAMs.[5]

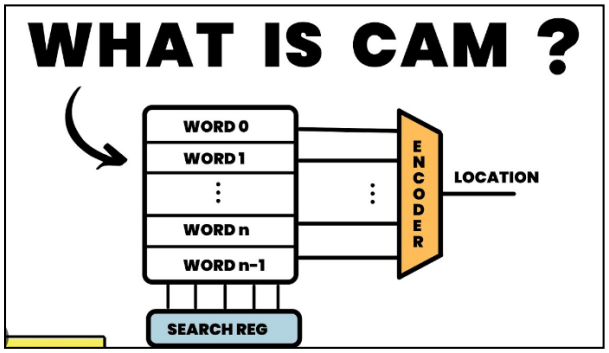

Fig. 2: Conceptual view of a CAM[6]

Figure 2 is a conceptual view of a CAM. As we can see whatever the data we are writing in the searching tab, it will be compared with all the words from 0 to n-1 in a single cycle and the match flag will be high at a single time.

Content-Addressable Memories (CAMs) rely on Static Random-Access Memory (SRAM) instead of Dynamic Random-Access Memory (DRAM) due to several key reasons. SRAM offers faster access times, making it well-suited for the high-speed search operations required in CAMs. Unlike DRAM, SRAM provides non-volatile storage, ensuring the content remains intact during power interruptions or resets. SRAM's parallel access capability allows for efficient comparison of multiple data patterns simultaneously, a crucial requirement for CAMs. Additionally, SRAM consumes lower power and is easier to integrate into CAM designs compared to DRAM. These advantages make SRAM the preferred choice for CAMs, enabling quick and efficient content comparison in a single clock cycle while ensuring data integrity and minimizing power consumption. In this paper we are going to adopt the 9T SRAM. The 9T SRAM cell incorporates three additional transistors compared to the 6T SRAM cell, resulting in improved performance and stability. These extra transistors enhance the read, hold, and write stability, which are crucial factors in CAM operations.

II. DESIGN, IMPLEMENTATION, AND DISCUSSION

To construct an 8-bit CAM using 9T SRAM, a set of individual components needs to be independently designed and fabricated. These components will be later integrated into the 8-bit CAM schematic circuit as a unified block.

A. 9T SRAM

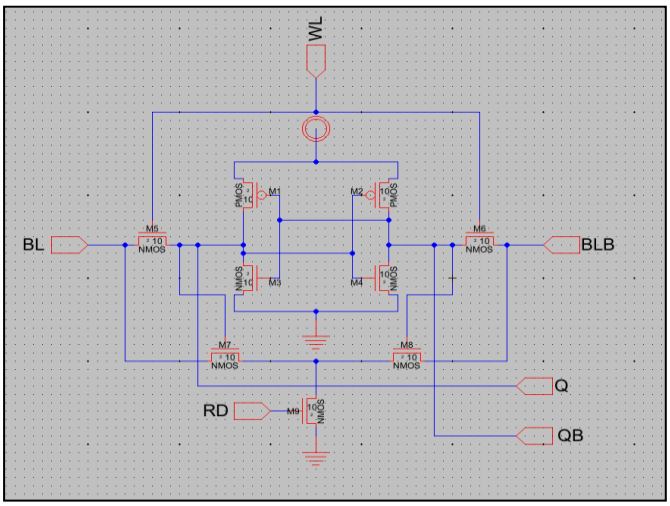

Fig. 3: 9T SRAM Schematic

It's obvious that 9 transistors (2 PMOS, and 7 NMOS) were used to construct the SRAM. In a 9T the inputs used are the Word Line (WL), Bit Line (BL), Read (RD), and Bit Line Bar (BLB). The Word Line selects the memory cell or word for read or write operations. The Bit Line is responsible for reading and writing data, with the BLB being its complement. The Read signal initiates a read operation, and the voltage difference between the BL and BLB signals determines the logic level of the stored data.

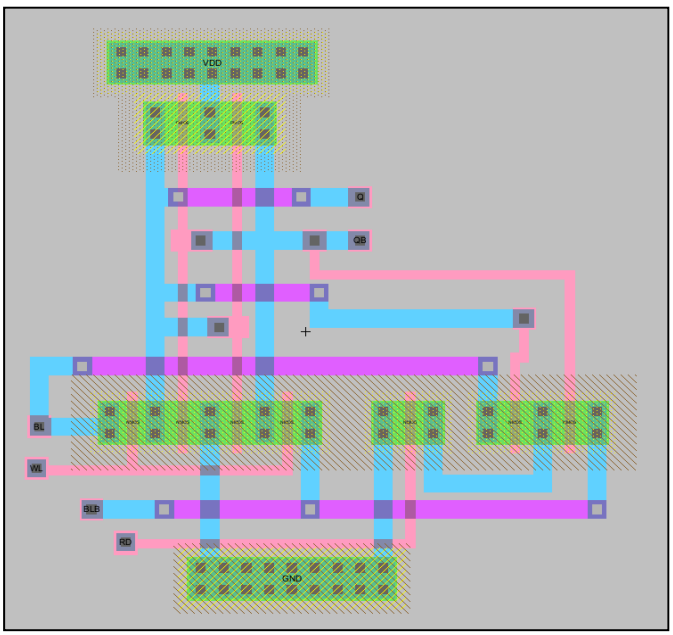

Fig. 4: 9T SRAM Layout

While the outputs include Q and QB. The Q output represents the stored data in the selected memory cell, reflecting the logic level of the data that was read from or written to the cell during the corresponding operation. QB, on the other hand, is the complement of the Q output, representing the inverted logic level of the stored data.

The Static Random Access memory device can perform the operation which is as follows: hold, read and write.

A 9T SRAM cell is presented in Fig.3. The upper sub circuit of the new memory cell is essentially a 6T SRAM cell with minimum sized devices (composed of M1, M2, M3, M4, M5, and M6). The two write access transistors (M5 and M6) are controlled by a write signal (WL). The data is stored within this upper memory sub-circuit. The lower sub-circuit of the new cell is composed of the Bit Line access transistors (M7 and M8) and the read access transistor (M9). The operations of M7 and M8 are controlled by the data stored in the cell. M9 is controlled by a separate read signal (RD).
This structure completely isolates the bit lines from the data storage nodes during read operation hence improving the static noise margin (SNM). But during write operation it utilizes both bit line(BL) and bit line bar (BLB) capacitances for charging and discharging, resulting in increased dynamic power consumption. [7]

When WL signal is high, the data on BL and BLB are written into the storage nodes Q and QB respectively. And when WL signal is low the Q and QB nodes store the recent written data before WL going to low. [8]

B. 1-bit CAM

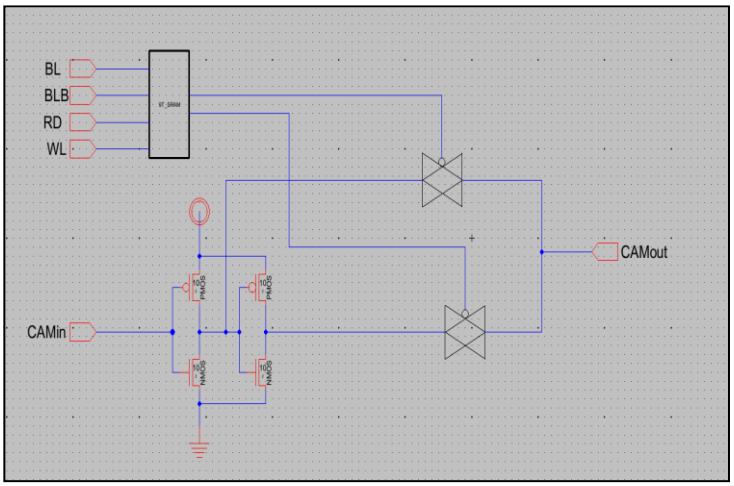

Fig. 5: 1-bit CAM Schematic

A 1-bit Content-Addressable Memory (CAM) can be implemented using the following components: 9T SRAM Cell, which is used as the storage element in the CAM, Inverters, and the Pass Gate also known as a transmission gate or pass transistor, is used as a switch to control the flow of data between the SRAM cell and the output.

When a search operation is initiated, the searched data (the data to be matched) is provided as input. The stored data is fed into two inverters, which generate the original and complemented versions of the stored data. These inverters ensure that both the stored data and its complement are available for comparison.

The CAM utilizes two transmission gates. One transmission gate connects the original stored data with the searched data, while the other transmission gate connects the stored data with the complemented searched data. The transmission gates are selectively activated based on the desired comparison operation. The comparison process results in either a match or a mismatch. If the stored data matches the searched data (or its complement), a match signal is generated. When a match occurs, the address of the matched cell is typically retrieved. This address indicates the location in the CAM where the match was found.

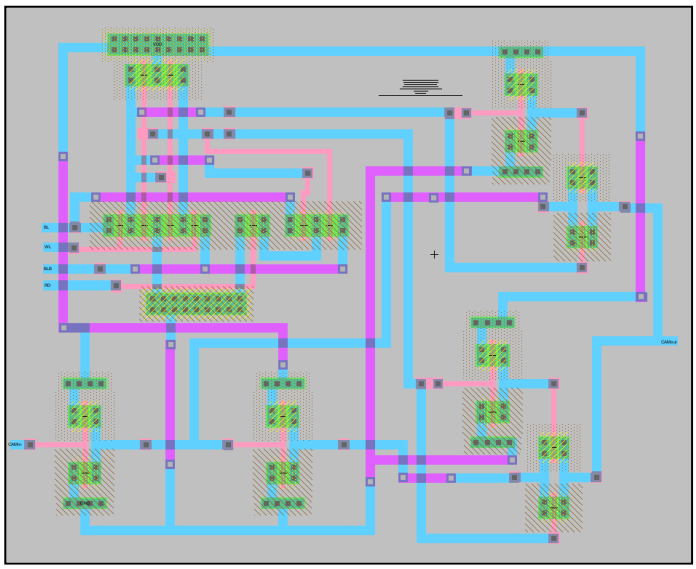

Fig. 6: 1-bit CAM Layout

C. *Inverter*

The following two figures presented illustrate the schematic and layout of an inverter.

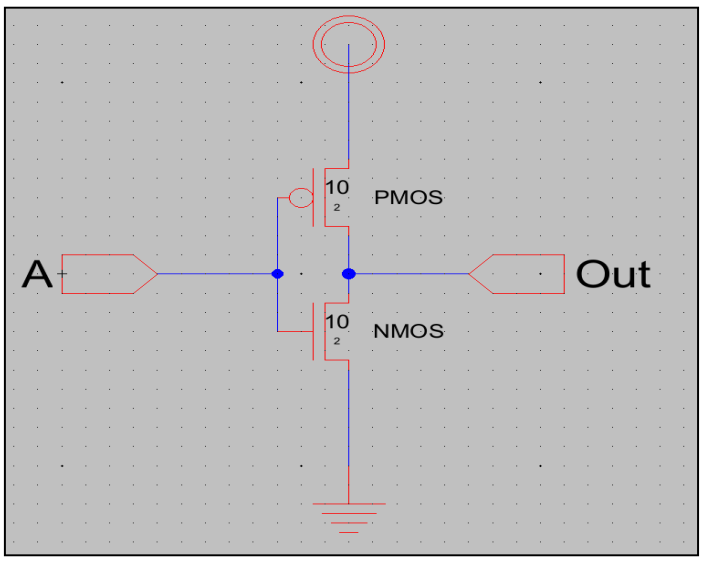

Fig. 7: Inverter Schematic

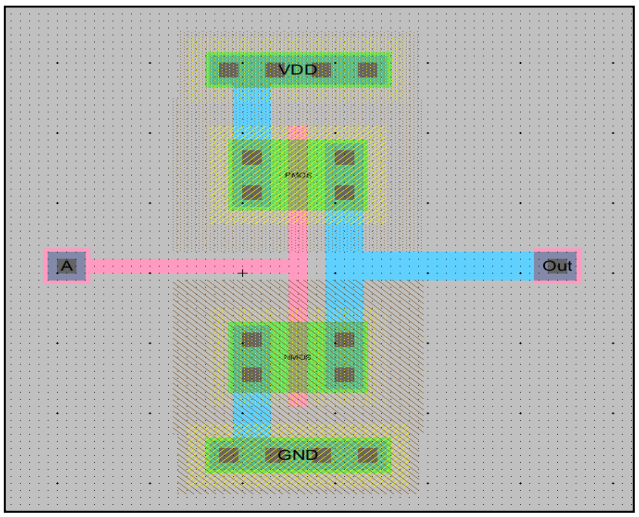

Fig. 8: Inverter Layout

D. *3x8 Decoder*

The following two figures presented illustrate the schematic and layout of 3 to 8 Decoder.

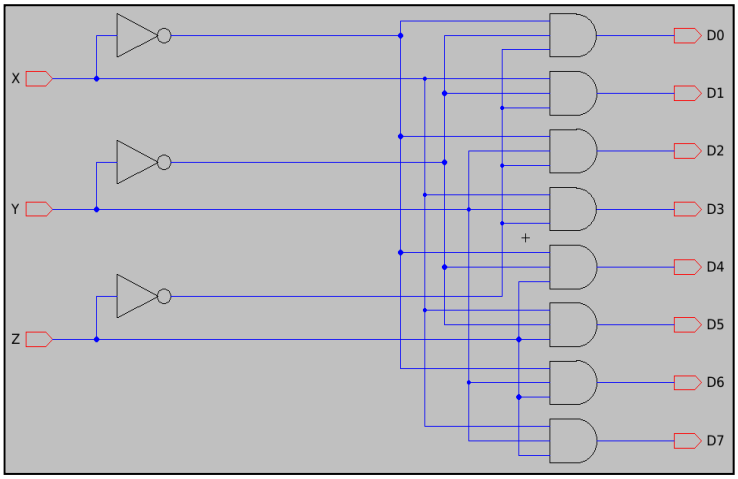

Fig. 9: 3x8 Decoder Schematic

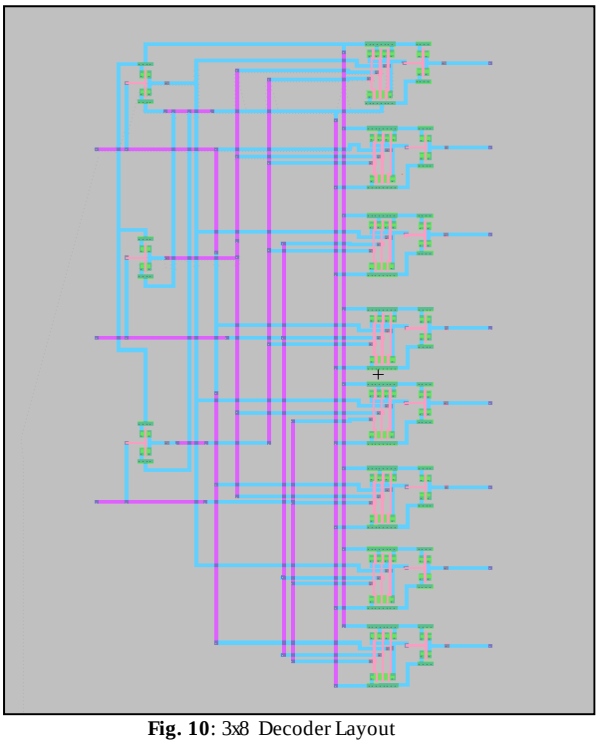

Fig. 10: 3x8 Decoder Layout

E. 4 to 1 NAND Gate

The following two figures presented illustrate the schematic and layout of a four input NAND gate.

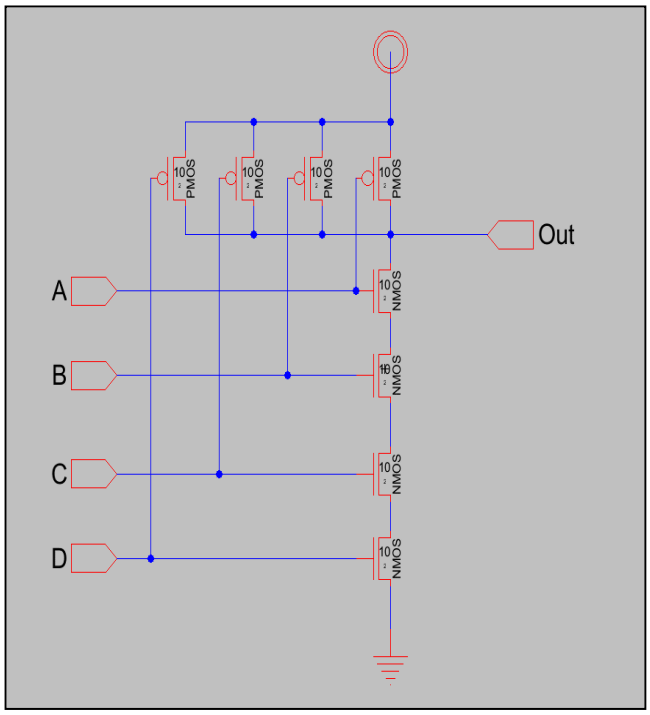

Fig. 11: 4to1 NAND Schematic

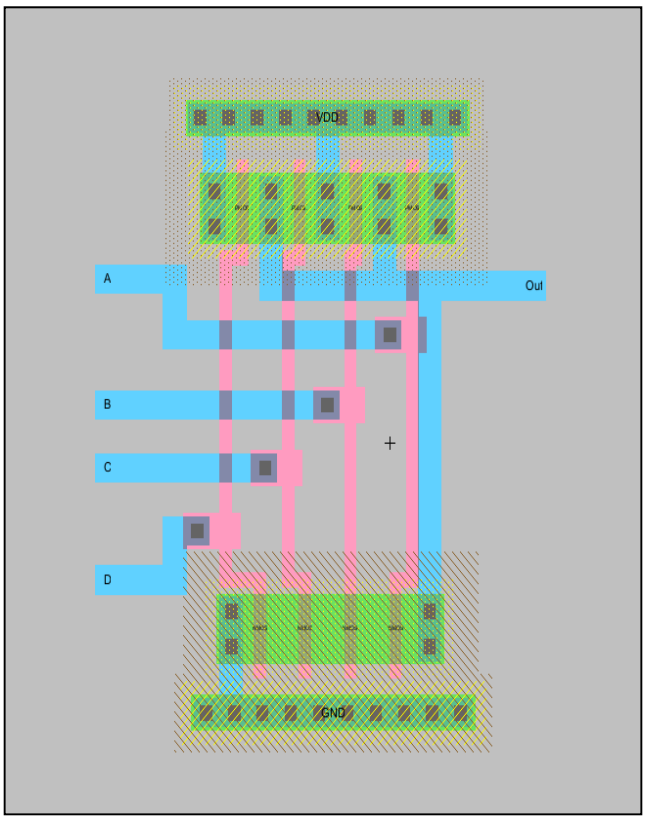

Fig. 12: 4to1 NAND Layout

F. 3 to 1 AND Gate

The following two figures presented illustrate the schematic and layout of a three input AND gate.

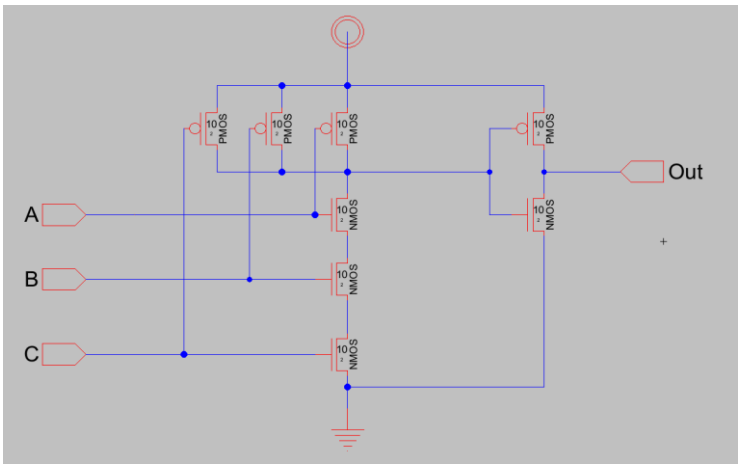

Fig. 13: 3to1 AND Schematic

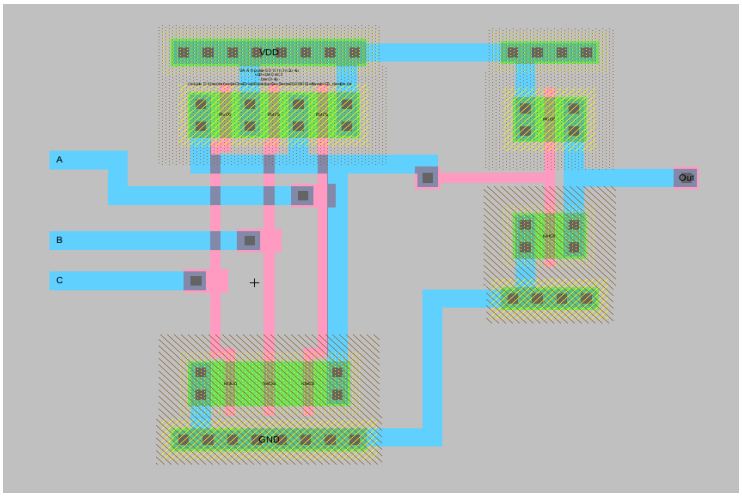

Fig. 14: 3to1 AND Layout

G. 8-bit CAM

By assembling the components discussed earlier, we have developed an 8-bit Content-Addressable Memory (CAM) using the 9T SRAM cell. The schematic and layout diagrams provided below depict the arrangement and organization of these components in the 8-bit CAM design.

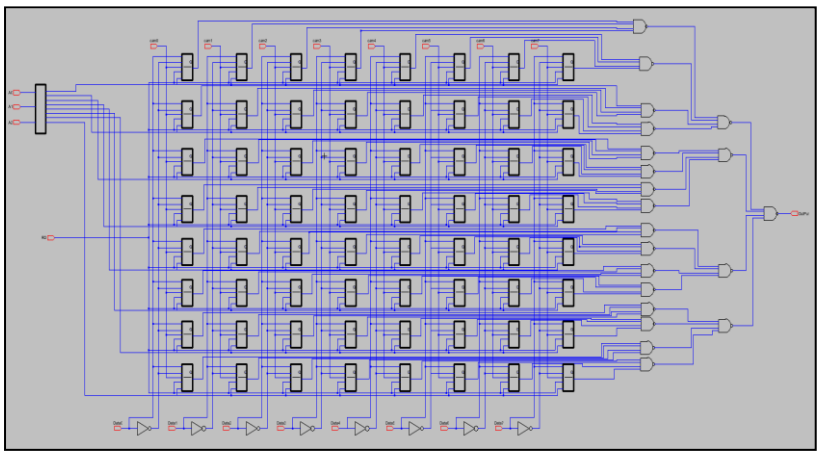

Fig. 13: 8-bit CAM Schematic

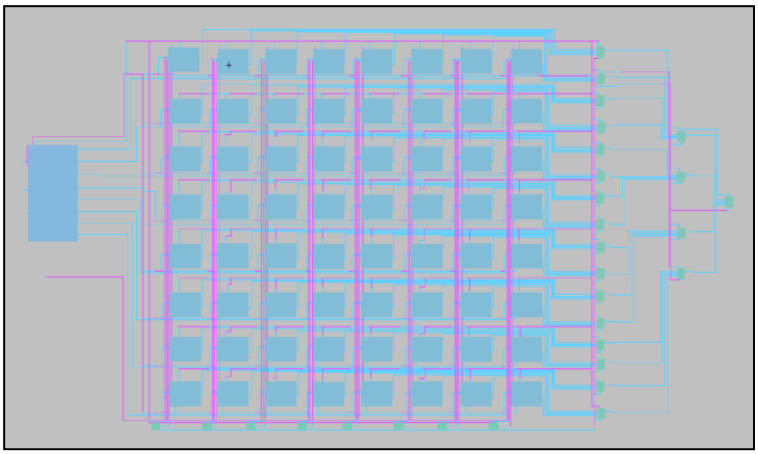

Fig. 14: 8-bit CAM Layout

III. Area, Power, and Delay Optimization

A. Power Optimization

In the proposed design, an optimized 9T-SRAM was developed to address power and stability concerns. The design introduced an NMOS transistor between the supply voltage (VDD) and the latches, operating in a special mode to reduce VDD and achieve a significant 98% reduction in power consumption. To maintain stability, an additional PMOS transistor was added between specific components to separate the storage and writing nodes of data. This transistor also prevented undesired changes in the cell's contents at low voltages. The proposed SRAM design successfully achieved a 98% reduction in power consumption, improved stability parameters by up to 93%, and reduced static power by 55% due to a stacking effect.

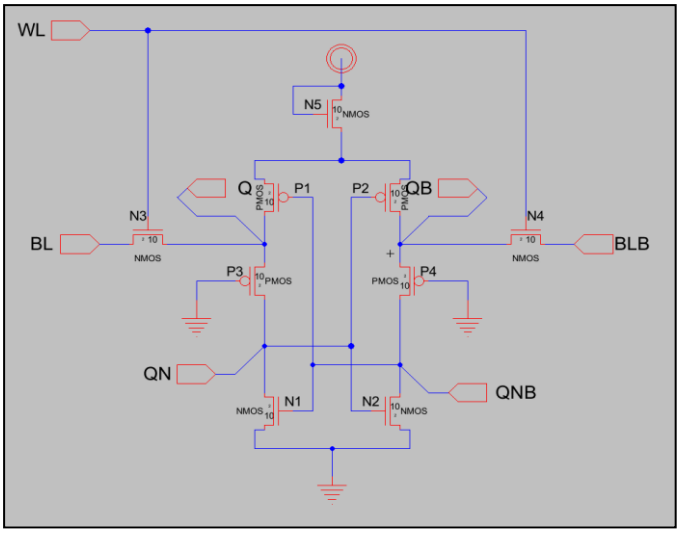

Fig. 15: Proposed 9T SRAM cell Schematic

In the proposed 9T SRAM cell, we have included a special type of transistor called NMOS (N5) that is connected like a diode. This transistor helps reduce the supply voltage (VDD), which significantly decreases the dynamic power consumption. Dynamic power is the power consumed when the cell is switching its state. By reducing VDD, we can reduce the power consumed during these switching operations.

However, scaling down the supply voltage can negatively affect the stability of the cell. There is a risk of the contents of the storage nodes flipping, especially when the cell is operating at very low supply voltages. To address this stability issue, we have added extra PMOS transistors (P3 and P4) between the driver and access transistors. These transistors are always turned ON and help improve the stability during read operations.

During a read operation, when we want to read a '1' from the cell, the voltage on the QNB node needs to be suppressed to prevent it from changing and flipping the contents. The added PMOS transistor (P4) and other transistors in the path help suppress this voltage, ensuring the stability of the cell's contents.

During a write operation, when we want to write a '0' to the cell, the voltage on the QN node needs to drop. The added transistors ensure that the QN node does not drop below a certain threshold voltage, preventing unintended changes in the cell's state.

Furthermore, the proposed design also reduces static power dissipation by using stacked MOSFETs (PMOS, PMOS, and NMOS). Stacking these transistors helps minimize power loss when the cell is in a static state, without switching.

Overall, the proposed 9T SRAM cell aims to optimize power consumption, improve stability, and reduce static power dissipation compared to conventional 6T SRAM cells. [9]

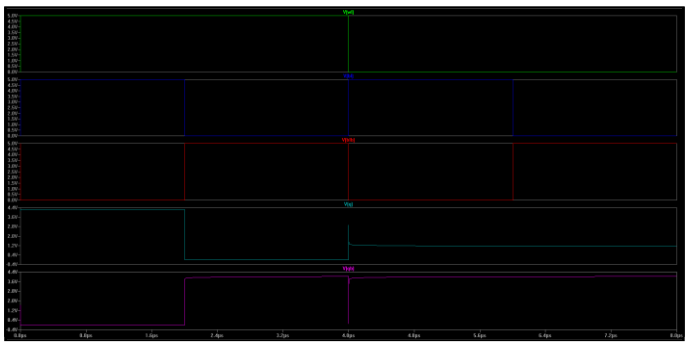

Fig. 15: Proposed 9T SRAM cell Simulations

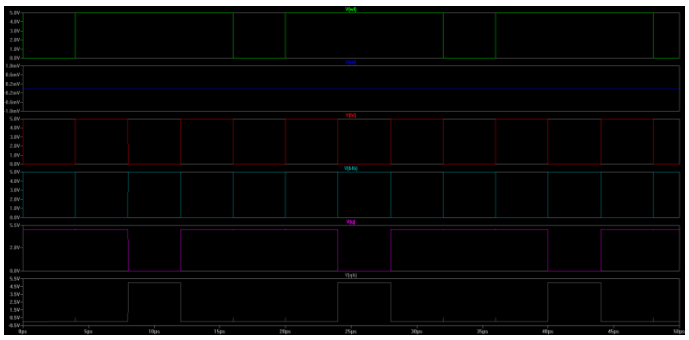

Fig. 16: Conventional 9T SRAM cell Simulation

By implementing the proposed 9T SRAM design, we were able to achieve significant power savings by reducing the supply voltage (VDD). According to the mathematical relationship between power (P) and supply voltage (VDD), P is directly proportional to VDD^2. By lowering the VDD in our design, we observed a substantial reduction in power consumption. In the simulations, we compared the Conventional 9T SRAM (figure 16) and the proposed 9T SRAM (figure 15), and found that the maximum output

voltage in the Conventional 9T SRAM reached 5 volts, while the proposed 9T SRAM had a maximum voltage of only 4.4 volts. This outcome demonstrates the effectiveness of reducing VDD in minimizing power consumption, as per the mathematical relation between power and VDD.

$$P = \alpha F C V_{DD}^{2}$$

Fig. 17: Relationship between power and VDD

B. Area Optimization

To optimize the area of our design, we made modifications to the 1-bit content-addressable memory (CAM) structure. Initially, each 1-bit CAM included two pass gates, with one of them comprising an inverter. However, to reduce the area footprint, we eliminated the need for this inverter. Since the 9T SRAM in the 1-bit CAM already outputs the address and its complement, we utilized this complement output as an input to the modified pass gate, thereby eliminating the requirement for an additional inverter.

After the optimization, each 1-bit CAM now consists of the two modified pass gates. Typically, an inverter consists of two transistors, so by eliminating the inverter in each pass gate, we effectively reduce the number of gates required by four transistors per 1-bit CAM. Considering that an 8-bit CAM comprises 64 1-bit CAMs, this optimization results in a significant reduction in the total number of gates by 256 gates (64 x 4).

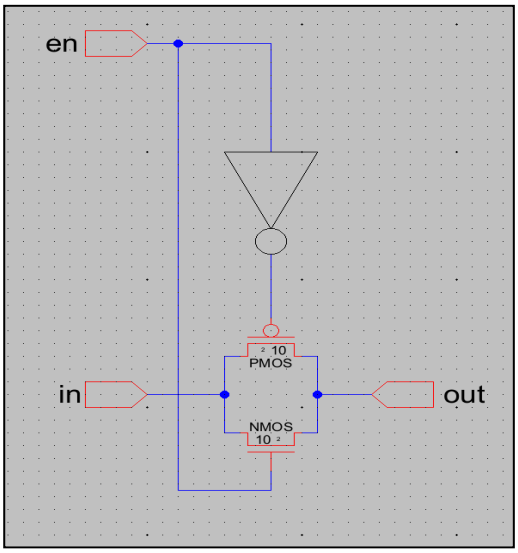

Fig. 18: Pass gate schematic before modification

C. Delay Optimization

When the inverter in the pass gate of the 1-bit CAM is eliminated, it reduces the number of logic gates in the circuit. This reduction in the number of gates leads to a simplification of the signal path and a reduction in the overall circuit complexity. As a result, the propagation delay through the 1-bit CAM is reduced.

In an inverter, there are metal wires, transistors, and other components that contribute to the capacitance. When the inverter is removed, these components are also eliminated, resulting in a decrease in the total capacitance of the circuit.

The RC time constant is a product of the resistance (R) and the capacitance (C) in the circuit. A lower capacitance value means that it takes less time for the voltage on a node to charge or discharge through the resistance. As a result, the RC time constant is reduced.

By reducing the RC time constant, the signal transitions in the circuit occur more quickly, leading to improved delay performance. The reduction in capacitance enables faster charging and discharging of the signal, resulting in improved overall circuit speed and reduced delay.

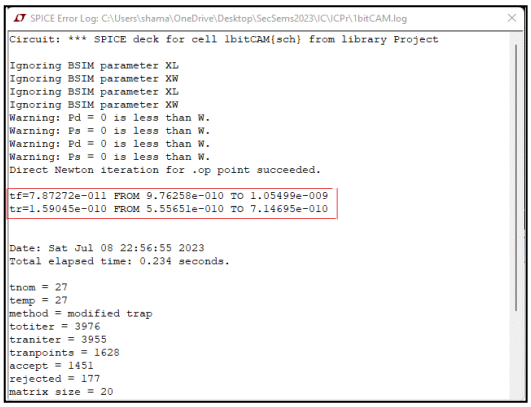

Fig. 19: TF and TR for 1-bit CAM before optimization

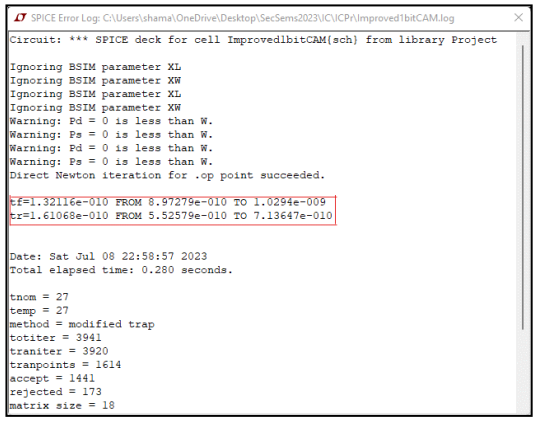

Fig. 20: TF and TR for 1-bit CAM after optimization

IV. RESULTS AND SIMULATIONS

A. 9T SRAM Simulation

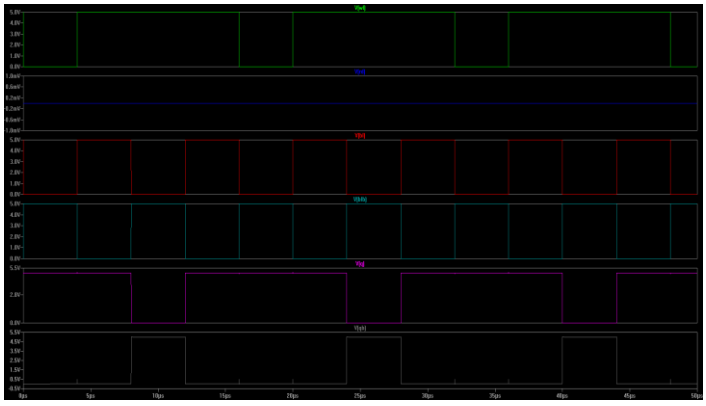

Fig. 21: Conventional 9T SRAM cell Simulation

B. 1-Bit CAM

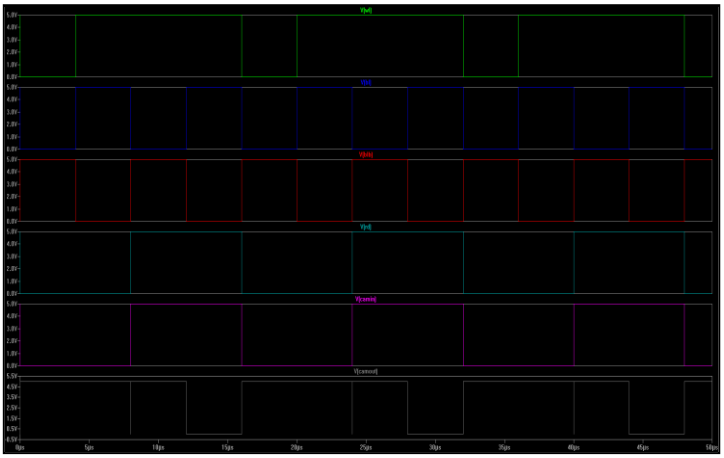

Fig. 22: 1-Bit CAM Simulation

C. 8-Bit CAM

- First test case --> DECODER IN ZEROS --> D0 = 1; [DataIn = 0, CamIn = 0, for the first row WL = 1, BL = 0, Q = 0, CamOut = CamIn, Other rows Q = 0, CamOut = CamIn = 0 --> Output = 1]
- Second test case --> DECODER IN ZEROS --> D0 = 1; [DataIn = 0, CamIn = 1, for the first row WL = 1, BL = 0, Q = 0, CamOut = CamIn, Other rows Q = 0, CamOut = CamIn = 0 --> Output = 0]
- Third test case --> DECODER IN ZEROS --> D0 = 1; [DataIn = 1, CamIn = 0, for the first row WL = 1, BL = 1, Q = 1, CamOut = not CamIn, Other rows WL = 0, Q randomly set to 0, CamOut = CamIn = 0 --> Output = 1]

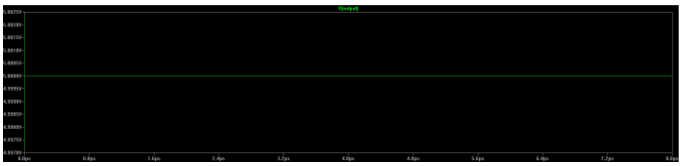

Fig. 23: **8-bit CAM Simulation test case 1& 3**

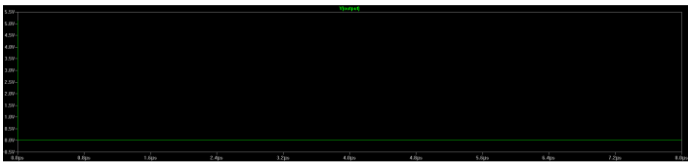

Fig. 24: 8-bit CAM Simulation test case 2

V. IMPROVMENTS

One possible improvement for Content-Addressable Memory (CAM) is to implement a pipelined matchlines architecture. In conventional CAM designs, only one word of search-line registers is compared to all entries in the CAM structure at a time. However, with a pipelined approach, multiple comparison operations can be performed in parallel. This means that different words of search data registers can be compared concurrently, increasing the overall speed and efficiency of the CAM.

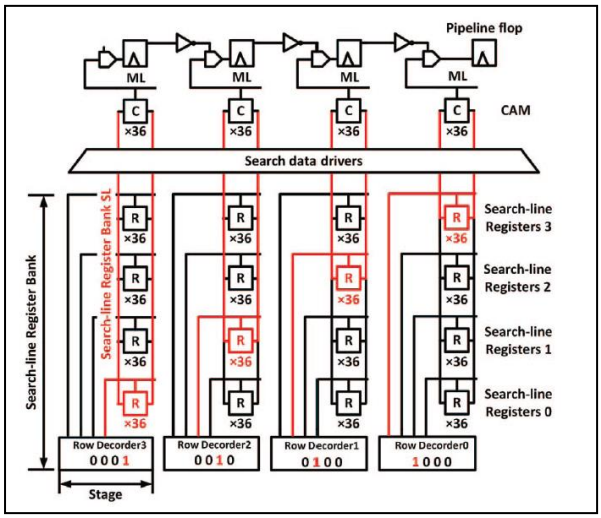

Fig. 25: Pipelined Search Data Registers Architecture.

In this pipelined scheme, the CAM is divided into segments, each containing a portion of the search-line registers. The data from these segments is then compared simultaneously during different clock cycles. By distributing the search-line registers across segments, the data arrival and comparison can be coordinated, allowing for efficient parallel processing. This approach reduces the latency associated with sequential comparisons and improves the overall performance of the CAM.

By implementing a pipelined matchlines architecture, the CAM can achieve higher throughput and reduced latency compared to conventional designs. This improvement in performance can have significant implications for applications that rely on CAM for fast and efficient data retrieval.[10]

VI. CONCLUSION

To sum up, in this project, the main objective was to design an 8-bit CAM (Content-Addressable Memory) using the 9T SRAM cell. The project involved a comprehensive understanding of the working principles of both CAM and SRAM. Each component of the 1-bit CAM was carefully examined, and the necessary components for building the 8-bit CAM were identified and explained. Additionally, the project explored various ideas and suggestions to enhance and optimize the power efficiency, delay, and area utilization of the 8-bit CAM. By implementing these proposed improvements, the project aimed to create a more efficient and effective CAM design using the 9T SRAM technology.

VII. REFERENCES

[1] https://www.pagiamtzis.com/cam/camintro/

[2] https://ieeexplore.ieee.org/document/1599540

[4]http://mdgomez.webs.uvigo.es/SED/practicas/Practica6_CAM.pdf

[5] https://www.pagiamtzis.com/pubs/pagiamtzis-jssc2006.pdf

[6]https://www.youtube.com/supported_browsers?next_url=https%3A%2F%2Fwww.youtube.com%2Fwatch%3Fv%3DGaMR6vYeHoY

[7]http://ijettjournal.org/2018/volume-55/number-2/IJETT-V55P219.pdf

[8] https://ieeexplore.ieee.org/document/7807806

[9] https://ieeexplore.ieee.org/abstract/document/7807806

[10] https://ieeexplore.ieee.org/document/7406979

[11]https://www.sciencedirect.com/science/article/abs/pii/S0026269216307650

[12] https://ieeexplore.ieee.org/document/6508061

[13] https://ieeexplore.ieee.org/document/5356318