A CIM-Based Implementation of 7-Segment Decoder Using Low-Leakage 8T SRAM Cells

P. Deepika
Department of ECE, RV College of Engineering, Bangalore, India | Visvesvaraya Technological University, Belagavi, Karnataka, India
deepikaprabhakar@rvce.edu.in

N. Shylashree
Department of ECE, RV College of Engineering, Bangalore, India | Visvesvaraya Technological University, Belagavi, Karnataka, India
shylashreen@rvce.edu.in (corresponding author)

Received: 25 September 2025 | Revised: 25 October 2025, 9 December 2025, 30 January 2026, and 5 March 2026 | Accepted: 17 March 2026

 DOI: https://doi.org/10.48084/etasr.15105

ABSTRACT

Computation-in-Memory (CIM) is an emerging paradigm that aims to overcome the performance and energy bottlenecks of traditional von Neumann architectures. In conventional systems, data are constantly shuttled back and forth between the memory unit and the processing unit, resulting in significant latency, high energy consumption, and bandwidth limitation, often referred to as the memory wall problem. CIM commonly employs 8T Static Random-Access Memory (SRAM) cells, as their decoupled read and write ports allow reliable in-memory operations. Compared to conventional 6T SRAM, the 8T design provides improved stability and better support for parallel computation. This makes 8T SRAM a strong candidate for energy-efficient and high-performance CIM architectures. However, the sleep-based pass transistor for read operation faces limitations, as the absence of a dedicated data retention mechanism results in potential data loss. The proposed design uniquely integrates CIM with a low-leakage 8T SRAM cell to enable direct logic implementation within memory, significantly reducing power and latency. This paper implements a CIM-based seven-segment decoder using a low-power data retention 8T SRAM cell, which demonstrates a 51% reduction in processing delay and a 56% decrease in power consumption during read/write operations. The complete decoder circuit has a power consumption of 36.89 µW and occupies an area of 1.509 mm², as confirmed through Cadence Virtuoso environment simulation utilizing gpdk45 technology.

Keywords-8T SRAM; Computation-in-Memory (CIM); decoder design; low-leakage SRAM; memory design

I. INTRODUCTION

With the increasing demand for low-power, high-performance systems, Computation-in-Memory (CIM) has emerged as a viable alternative to conventional von Neumann architectures. CIM alleviates the memory access bottleneck and enables efficient execution of arithmetic and logic operations by integrating processing capabilities directly within the memory array, as shown in Figure 1. Among the various memory technologies, Static Random-Access Memory (SRAM) has been extensively studied because of its compatibility with CMOS, scalability, and high speed. However, the traditional 6T SRAM cell faces challenges such as read disturbance and inadequate stability during in-memory operations, which limit its usefulness. To address these challenges, 8T SRAM cells that include data retention capability enhance read stability [1] and reduce leakage, making them more appropriate for power-sensitive CIM circuits.

In this study, a low-leakage 8T SRAM cell combined with transmission gates and latch-based sensing is employed to implement a Binary-Coded Decimal (BCD) to 7-segment decoder. The latch-based sense amplifiers effectively detect minor variations in the bit-line, enhancing the speed and reliability of in-memory computations. The proposed architecture utilizes the SRAM array for storing BCD inputs, a mapping scheme for decoder functionality, and latch-based sensing [2] for effective signal amplification. This design serves as a concrete example of SRAM-based CIM for efficient arithmetic and display-related logic operations.

CIM is a promising approach to overcome memory-computation limitations in von Neumann architectures, with applications in neural network accelerators and low-power edge devices. The proposed design adds to this developing field by showcasing effective BCD to 7-segment conversion using a novel SRAM design based on CIM, validated through Cadence Virtuoso simulations.

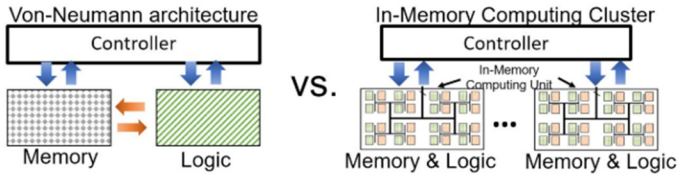

Fig. 1. Von Neumann vs CIM architecture.

II. PREVIOUS WORK

Authors in [1] present an 8T SRAM cell with sleep functionality and a pass transistor-based reading scheme. Although sleep mode reduces power consumption, it compromises the volatile nature of SRAM, leading to potential data loss. In [3], SRAM is utilized for half-adder computation and storage. Similarly, logic gates with sleep transistors and data retention mechanisms are proposed in [4] to improve power efficiency. An 8T SRAM-based half-adder with a latch-based sense amplifier is introduced in [2], improving Arithmetic Logic Unit (ALU)–memory interaction and overall computational efficiency. Authors in [5] present a half-subtractor and XOR/XNOR-based full adder design, achieving reduced power consumption and latency. In [6], SRAM-based architectures are further extended to in-memory Boolean operations. Authors in [7] present energy-efficient Resistive Random-Access Memory (RRAM) crossbars for big data processing, whereas authors in [8] present an on-chip Read-Only Memory (ROM) design using logic transistors. Authors in [9] introduce high-precision sensing and low-power amplifiers, and authors in [10] use 6T SRAM for a Binary Content-Addressable Memory (BCAM)/Ternary Content-Addressable Memory (TCAM) design, saving energy in search operations.

The 6T SRAM has long been considered a standard design due to its stable operation, compact structure, and strong noise margins, making it widely adopted in embedded systems [10-12]. However, it suffers from limitations such as read disturb susceptibility, leakage current, and reduced stability under near-threshold voltage operation, motivating continued research into improved SRAM designs [13-19]. In-memory computation techniques for XNOR and other basic gates have also been reported in [20]. Furthermore, comparative studies of 6T, 7T, 8T, and 9T SRAM cells highlight the trade-offs between stability, area overhead, and power consumption in scaled technologies [21].

To address these limitations, the 8T SRAM architecture has been proposed, featuring decoupled read and write paths that improve read stability and reliability under near-threshold operation. Beyond enhancing robustness against process variations [16], the 8T SRAM cell supports in-memory computation by enabling both Boolean and arithmetic operations. Its potential as a multi-bit dot-product engine demonstrates its suitability for beyond von Neumann architectures. Extensions using optimized sense amplifier designs [22-26] and non-volatile memory-based CIM schemes further enhance its applicability. Despite its increased area overhead and sensitivity to defects, the 8T SRAM cell remains a promising candidate for future CIM solutions, where reduced data movement and enhanced energy efficiency are critical.

III. PROPOSED DESIGN

The proposed design significantly improves processing time and power efficiency compared to existing designs [10]. Figure 2 depicts the design of the common-anode style BCD to 7-segment display decoder. The SRAM cells A, B, C, and D store input bits. The read circuitry, comprised of logic gates using these cells, generates the outputs. A latch-based sense amplifier circuit is used for sensing. The decoder's outputs (a, b, c, d, e, f, g) light the corresponding segments as shown in Figure 2.

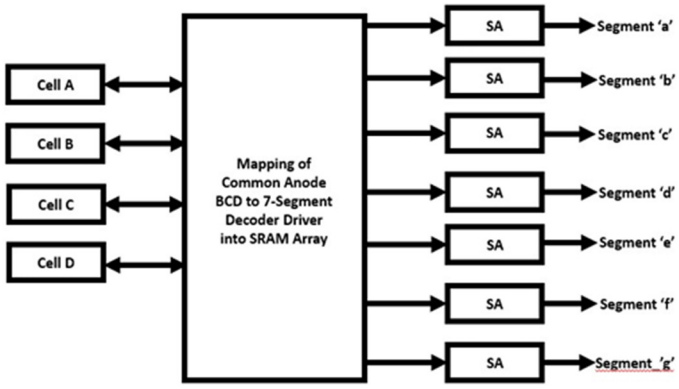

Fig. 2. Realization of SRAM operation using common-anode BCD to 7-segment decoder.

An 8T SRAM cell with sleep functionality and a pass transistor-based reading scheme [1] is implemented. Although the higher transistor count increases power consumption, the 8T SRAM mitigates the limitations of the 6T SRAM cell. Notably, sleep transistors must be ON during read and write operations; when OFF, data loss occurs. The existing SRAM cell in [1] consumes 2.833 mW power and exhibits a 116.574 ps delay after simulation in Cadence Virtuoso 45 nm technology.

A. Proposed 8T SRAM Cell

Figure 3 illustrates the novel low-power data retention 8T SRAM with distinct read and write pathways. The transmission gate-based architecture primarily focuses on read operations, which are typically handled by peripheral circuits in conventional designs.

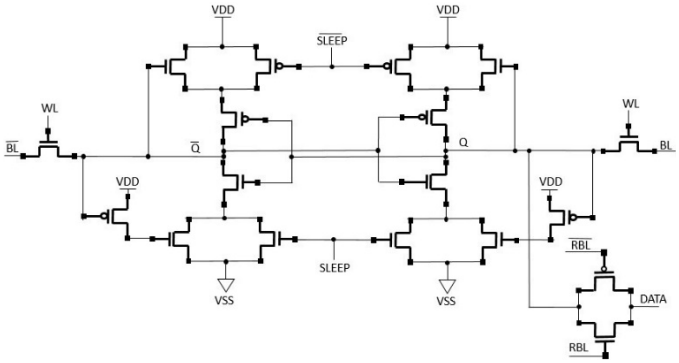

Fig. 3. Proposed low-power data retention 8T SRAM cell circuit.

Conventional SRAM cells operate similarly during the write operation. First, the sleep transistors are activated, and the word line (WL) controls access. The input data are applied to the bit-lines BL and BLb. Once WL is asserted, the access transistors turn ON, allowing the bit-line data to be written into the cell. The two storage nodes Q and Qb thus store the input data.

After the write operation, the sleep control is deactivated to power-gate the SRAM cell. The data retention circuit maintains Q and Qb at VDD and VSS through NMOS pull-up and pull-down paths, respectively, controlled via a PMOS gate as shown in Figure 3. Before read operation, the complementary read lines RL and RLb are enabled. A transmission gate is used to implement the read port. The primary goal in this case is to read the data from storage node Q. The stored data at node Q are transferred through the transmission gate to the output node DATA when the read path is enabled.

This approach eliminates the need for complex peripheral read circuitry and allows direct access to the stored data without keeping the entire SRAM cell continuously active. As a result, the SRAM operates only during write and read cycles, while remaining OFF during idle periods, thereby reducing power consumption significantly. Compared to conventional 8T SRAM designs that rely on pass-transistor logic and Qb sensing, the proposed transmission gate-based read circuitry directly accesses node Q, improving read accuracy and speed. Importantly, this design retains data in a manner similar to non-volatile memory even when the SRAM cell is power-gated. Figures 4 and 5 present the schematic implementation and output waveform of the proposed SRAM cell, respectively.

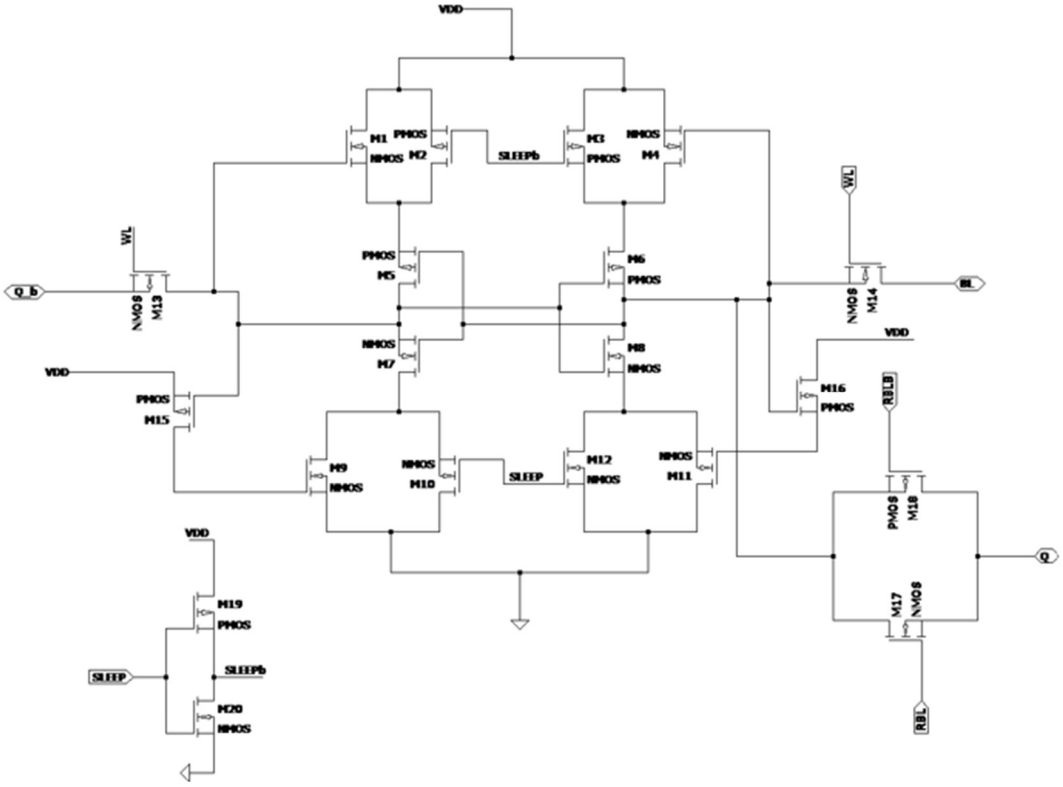

Fig. 4. Schematic of the low-power data retention 8T SRAM cell circuit.

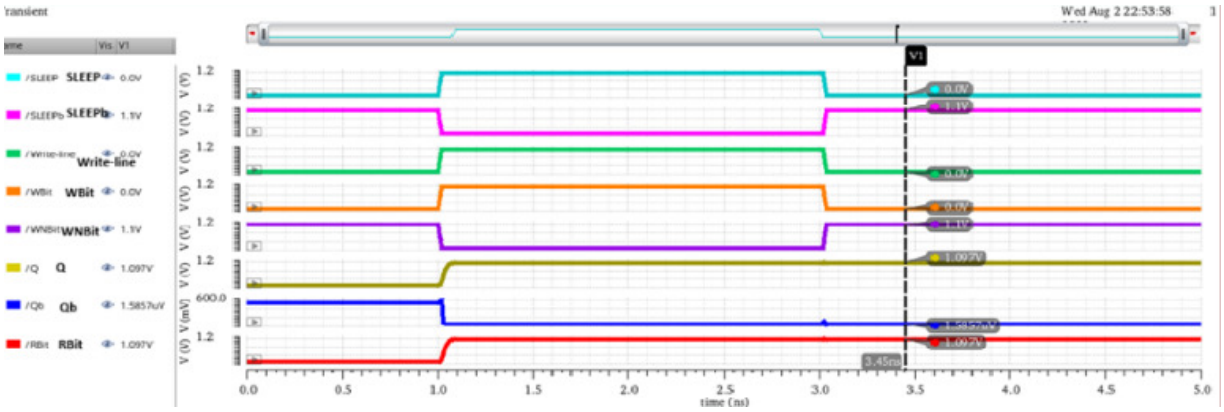

Fig. 5. Output waveform of the proposed SRAM.

In Figure 5, the SRAM cell initially operates in active mode with SLEEP = 0 V and SLEEPb = 1.1 V, enabling the header and footer sleep transistors and connecting the cell to VDD and VSS. The word line (WL / write_line) is asserted, activating the access transistors. Complementary write data are applied: WBit = 0 V and WNBit = 1.1 V. With WL enabled, the bit-line data are written into the cell, overwriting the previous state. The storage nodes settle correctly at Q ≈ 1.097 V and Qb ≈ 0 V.

After completion of the write operation, SLEEP transitions from 0 V to 1.1 V and SLEEPb transitions from 1.1 V to 0 V. The header and footer sleep transistors are turned OFF, disconnecting the cell from the power rails. The word line is deasserted, isolating the cell from bit-line disturbances. At this stage, the data-retention circuit maintains the internal nodes, keeping Q near VDD and Qb near VSS. Despite power gating, Q remains ≈ 1.097 V and Qb remains near 0 V, confirming successful retention in sleep mode.

During the sleep interval, the bit-lines remain inactive. At t = 3.2 ns, the cell is reactivated: SLEEP returns to 0 V and SLEEPb returns to 1.1 V, reconnecting the power rails. The stored data are recovered without degradation, with Q ≈ 1.097 V and Qb ≈ 0 V. RBit correctly reflects the stored value during read operation, confirming read stability after wake-up. Minimal voltage degradation is observed at the storage nodes.

Power is consumed only during write and initial read operations, whereas subsequent cycles keep the cell in sleep mode, while the data retention circuit (based on [4]'s BSIM design technique) maintains data integrity. The proposed SRAM cell achieves a Static Noise Margin (SNM) of 284.77 mV, as obtained from the butterfly curve analysis.

IV. DESIGN METHODOLOGY

A 7-segment display is a common electronic display showing decimal digits (0–9) and extra characters (A–F for hex). It comprises seven LEDs arranged in a standard pattern, with an optional decimal point LED. Displaying desired digits or characters involves activating or deactivating specific LED combinations. The display has segments a–g. BCD to 7-segment decoders (Figure 6) exist in common-anode and common-cathode configurations. The distinction lies in the LED connections.

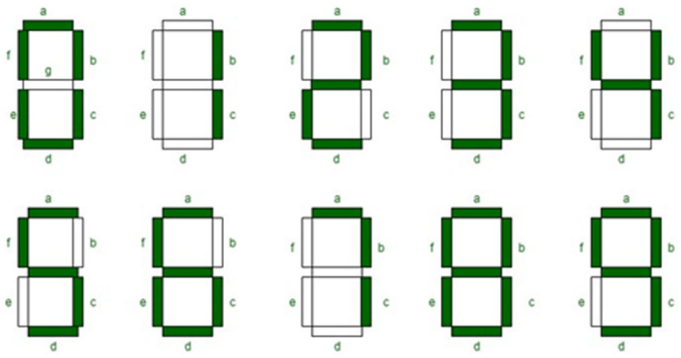

Fig. 6. 7-segment display.

This paper implements a common-anode type BCD to 7-segment decoder using the proposed low-power data-retention 8T SRAM design. A similar approach can be used to realize its counterpart, i.e., common-cathode design, using the proposed SRAM design.

Using Table I, individual common-anode segments can be realized using the K-map technique as:

- Segment $'a' = A + C + BD + \overline{B}\,\overline{D}$
- Segment $'b' = \overline{B} + \overline{C}\,\overline{D} + CD$
- Segment $'c' = B + \overline{C} + D$
- Segment $'d' = A + \overline{B}\,\overline{D} + \overline{B}C + C\overline{D} + B\overline{C}D$
- Segment $'e' = \overline{B}\,\overline{D} + C\overline{D}$
- Segment $'f' = A + B\overline{C} + B\overline{D} + \overline{C}\,\overline{D}$
- Segment $'g' = A + B\overline{C} + \overline{B}C + C\overline{D}$

After realizing all the segments using the derived logical equations, the common-anode BCD to 7-segment decoder gate-level schematic is designed as shown in Figure 7.

TABLE I. COMMON-ANODE TRUTH TABLE

Number	BCD input				7-segment output						
	A	B	C	D	a	b	c	d	e	f	g
0	0	0	0	0	1	1	1	1	1	1	0
1	0	0	0	1	0	1	1	0	0	0	0
2	0	0	1	0	1	1	0	1	1	0	1
3	0	0	1	1	1	1	1	1	0	0	1
4	0	1	0	0	0	1	1	0	0	1	1
5	0	1	0	1	1	0	1	1	0	1	1
6	0	1	1	0	1	0	1	1	1	1	1
7	0	1	1	1	1	1	1	0	0	0	0
8	1	0	0	0	1	1	1	1	1	1	1
9	1	0	0	1	1	1	1	0	1	1	1

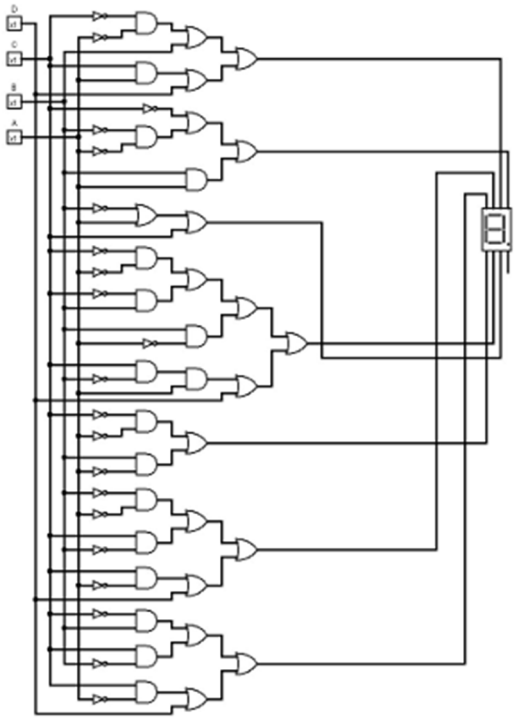

Fig. 7. Common-anode gate-level schematic.

A. Mapping the Decoder Netlist into SRAM Array

Figure 8 depicts one such cell that implements a logic function. The SRAM shown in Figure 3 is mapped to the structure in Figure 8. The SRAM cells, represented as A, B, C, and D, form an SRAM array as shown in Figure 2. The outputs of the corresponding logic gates are realized as illustrated.

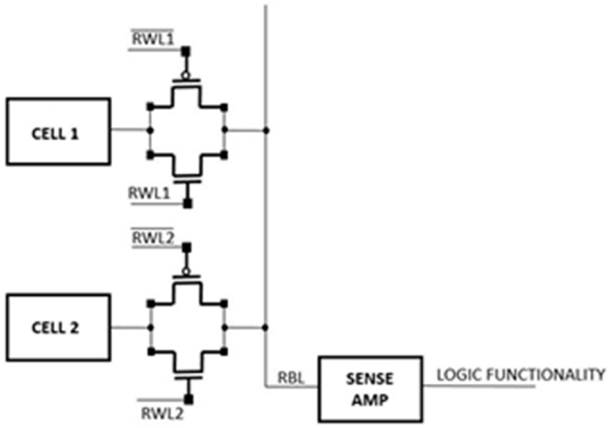

Fig. 8. Circuit realization of a logic function.

SRAM is a volatile memory that holds data as long as power is present. This work intends to modify the volatile nature of SRAM into a quasi-non-volatile behavior to reduce static and dynamic power dissipation in memory units. Sense amplifiers detect voltage differences at bit-lines, amplifying signals for accurate reading [22]. This reduces read time, as the memory interprets signals instantly without waiting for a full voltage swing. Sense amplifiers include voltage-mode (differential) and current-mode (non-differential) types. Voltage sensing includes static (latch-based) and dynamic designs connected to bit-lines [15]. Capacitances affect efficiency; higher capacitance increases Resistance–Capacitance (RC) delay and output latency. Latch-type amplifiers are fast due to strong positive feedback, but noise can introduce errors [23].

V. RESULTS AND DISCUSSION

The schematic of the common-anode BCD to 7-segment decoder driver CIM architecture is shown in Figure 9. The RET CELL receives input data for computation, where cells (A, B, C, and D in Figure 2) are referred to as RET. COMPUTE_CA is the schematic of the common-anode BCD to 7-segment decoder driver computation circuit. The computation circuit is implemented as a COMPUTE_CA block and integrated with the RET CELL. These RET CELLS feed BCD inputs to the decoder driver computational block COMPUTE_CA, which performs computation and reads data from the RET CELLS in the form of a BCD-to-7-segment decoder driver operation.

The functionality of the design is verified by storing the sample data 0111 in the four RET CELLS. The inputs A = 0, B = 1, C = 1, and D = 1 from these cells are fed to the COMPUTE_CA block (Figure 10). Notably, even when SLEEP is OFF and the RET CELLS are unpowered, the outputs A , B , C , and D continue to retain data. This demonstrates the data-retention capability of the proposed low-power data-retention 8T SRAM cell, distinguishing it from conventional SRAM cells, which cannot retain data without power.

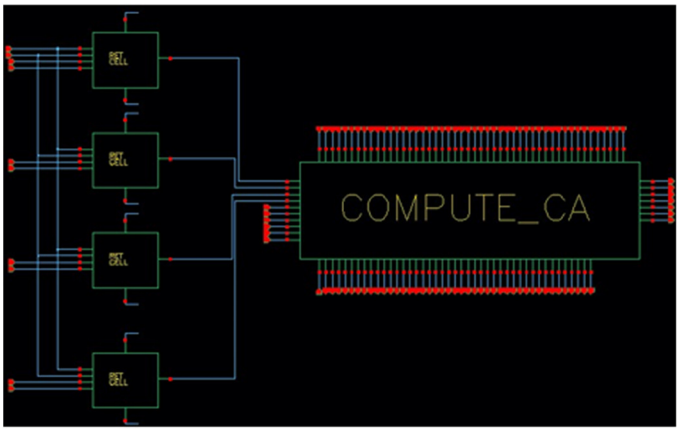

Fig. 9. Schematic of common-anode BCD to 7-segment decoder driver CIM architecture.

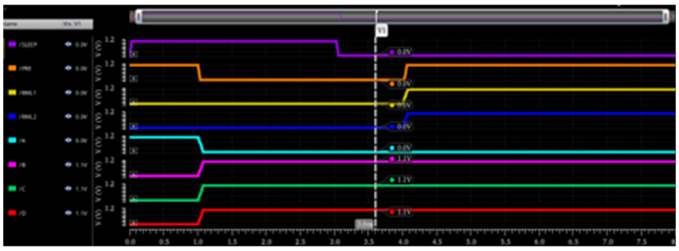

Fig. 10. Input 0111 to the common-anode block.

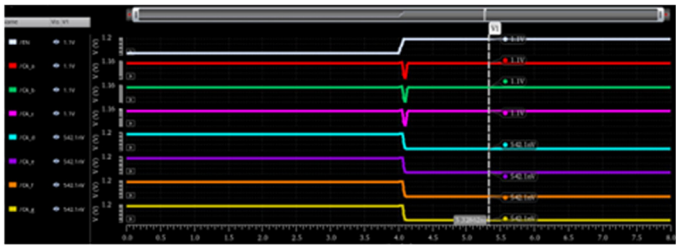

Fig. 11. Output 1110000 from the common-anode block.

As soon as the enable (EN) pin of the COMPUTE_CA block is triggered with the corresponding reference voltages set, the output 1110000 is obtained as shown in Figure 11, where segment a = 1, b = 1, c = 1, d = 0, e = 0, f = 0, and g = 0, indicating the display of digit '7' on the display module (if connected to the circuit).

The corresponding layout of the full BCD to 7-segment decoder driver using CIM architecture, occupying an area of 1.509 mm², is shown in Figure 12.

From Table II, it is observed that the proposed 8T SRAM cell achieves approximately 56% reduction in power consumption and 51% reduction in delay compared to the existing design. This SRAM was used to design the common-anode BCD to 7-segment decoder driver circuit using the CIM technique.

TABLE II. SRAM BIT-CELL DESIGN COMPARISON

Sl. no	Parameter	8T SRAM cell (existing)	8T SRAM cell (proposed)
1	Power (nW)	171.7	97.52
2	Delay (ns)	5.92	2.969
3	Transistor count	10	20

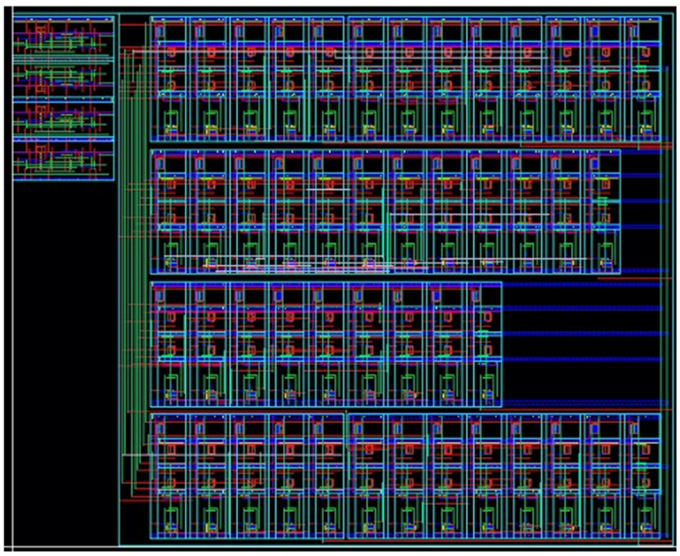

Fig. 12. Layout of common-anode BCD to 7-segment decoder driver CIM architecture.

Lastly, the full BCD to 7-segment decoder driver with SRAM inputs dissipates a power of 36.89 μW, which is lower than the single SRAM bit cell of [1], and occupies an area of 1.509 mm².

The scalability of the proposed design is an important consideration for larger CIM arrays and emerging Artificial Intelligence (AI) accelerator circuits. As the memory array grows, performing computation directly within SRAM cells significantly reduces data movement and peripheral switching activity, improving energy efficiency. While integrating decoder functionality inside the memory introduces a modest area overhead at the cell/periphery boundary, it eliminates external logic blocks and reduces interconnect complexity. These trade-offs make the proposed design a promising candidate for scalable CIM architectures.

VI. CONCLUSION

This work demonstrates the implementation of a Binary-Coded Decimal (BCD) to 7-segment decoder using a low-leakage, data-retention 8T Static Random-Access Memory (SRAM) cell integrated with transmission gates and a latch-based sensing scheme for Computation-in-Memory (CIM) applications. Although it is common practice to use 6T SRAM arrays, they suffer from read disturbance and stability issues when they are used for CIM operation, which limits their reliability. The proposed 8T SRAM design exhibits improved read stability, lower leakage, and enhanced data retention, making it suitable for arithmetic and logic-intensive CIM applications. Furthermore, the design employs latch-based sense amplifiers to enable fast and energy-efficient sensing of small bit-line differentials, using strong positive feedback for reliable read operations.

In this work, the header/footer sleep transistors are sized at 45 nm to ensure reliable low-leakage operation and state retention in the proposed 8T SRAM cell. A comprehensive parametric analysis of sleep transistor sizing and its impact on voltage drop, read/write delay, and overall power–delay trade-offs can be carried out in future work to further optimize the design for performance.

DECLARATION OF COMPETING INTERESTS

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

ACKNOWLEDGMENT

We gratefully acknowledge the Center of Excellence in Integrated Circuits and Systems (COE-ICAS) and the Department of Electronics and Communication Engineering at RV College of Engineering, Bengaluru, for their invaluable support and resources in the completion of this research.

DATA AVAILABILITY

The data that support the findings of this study are available from the authors upon reasonable request.

FUNDING

No funding was received for conducting this study.

REFERENCES

[1] M. A. Turi and J. G. Delgado-Frias, "Effective Low Leakage 6T and 8T FinFET SRAMs: Using Cells With Reverse-Biased FinFETs, Near-Threshold Operation, and Power Gating," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 67, no. 4, pp. 765–769, Apr. 2020, https://doi.org/10.1109/TCSII.2019.2922921.

[2] A. K. Rajput and M. Pattanaik, "Implementation of Boolean and Arithmetic Functions with 8T SRAM Cell for In-Memory Computation," in *2020 International Conference for Emerging Technology*, Belgaum, India, 2020, pp. 1–5, https://doi.org/10.1109/INCET49848.2020.9154137.

[3] J. Han and Y. Kim, "A Fast Half Adder using 8T SRAM for Computation-in-Memory," in *2021 IEEE International Conference on Consumer Electronics-Asia*, Gangwon, Republic of Korea, 2021, pp. 1–3, https://doi.org/10.1109/ICCE-Asia53811.2021.9641964.

[4] D. J. Frank, R. H. Dennard, E. Nowak, P. M. Solomon, Y. Taur, and H.-S. P. Wong, "Device scaling limits of Si MOSFETs and their application dependencies," *Proceedings of the IEEE*, vol. 89, no. 3, pp. 259–288, Mar. 2001, https://doi.org/10.1109/5.915374.

[5] D. Prabhakar, N. Shylashree, S. Y. Narasimhaiah, Y. B. Mariswamappa, and S. S. Hemaraj, "A fast half-subtractor using 8T static random access memory for in-memory computation," *International Journal of Reconfigurable and Embedded Systems*, vol. 14, no. 1, pp. 273–281, Mar. 2025, https://doi.org/10.11591/ijres.v14.i1.pp273-281.

[6] B. Wicht, T. Nirschl, and D. Schmitt-Landsiedel, "Yield and speed optimization of a latch-type voltage sense amplifier," *IEEE Journal of Solid-State Circuits*, vol. 39, no. 7, pp. 1148–1158, July 2004, https://doi.org/10.1109/JSSC.2004.829399.

[7] A. Jaiswal, I. Chakraborty, A. Agrawal, and K. Roy, "8T SRAM Cell as a Multibit Dot-Product Engine for Beyond Von Neumann Computing," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 27, no. 11, pp. 2556–2567, Nov. 2019, https://doi.org/10.1109/TVLSI.2019.2929245.

[8] S. Jain, A. Ranjan, K. Roy, and A. Raghunathan, "Computing in Memory With Spin-Transfer Torque Magnetic RAM," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 26, no. 3, pp. 470–483, Mar. 2018, https://doi.org/10.1109/TVLSI.2017.2776954.

[9] A. Agrawal, A. Jaiswal, C. Lee, and K. Roy, "X-SRAM: Enabling In-Memory Boolean Computations in CMOS Static Random Access Memories," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 65, no. 12, pp. 4219–4232, Dec. 2018, https://doi.org/10.1109/TCSI.2018.2848999.

[10] N. Verma *et al.*, "In-Memory Computing: Advances and Prospects," *IEEE Solid-State Circuits Magazine*, vol. 11, no. 3, pp. 43–55, 2019, https://doi.org/10.1109/MSSC.2019.2922889.

[11] J. Mu and B. Kim, "A 65nm Logic-Compatible Embedded and Flash Memory for In-Memory Computation of Artificial Neural Networks," in *2020 IEEE International Symposium on Circuits and Systems*, Seville, Spain, 2020, pp. 1–4, https://doi.org/10.1109/ISCAS45731.2020.9181104.

[12] P. P. Ravichandiran and P. D. Franzon, "A Review of 3D-Dynamic Random-Access Memory based Near-Memory Computation," in *2021 IEEE International 3D Systems Integration Conference*, Raleigh, NC, USA, 2021, pp. 1–6, https://doi.org/10.1109/3DIC52383.2021.9687615.

[13] G. Manikannan, K. Mahendran, and P. Prabakaran, "Low Power High Speed Full Adder Cell with XOR/XNOR Logic Gates in 90nm Technology," in *2017 International Conference on Technical Advancements in Computers and Communications*, Melmaurvathur, India, 2017, pp. 61–65, https://doi.org/10.1109/ICTACC.2017.25.

[14] A. Bhaskar, "Design and analysis of low power SRAM cells," in *2017 Innovations in Power and Advanced Computing Technologies*, Vellore, India, 2017, pp. 1–5, https://doi.org/10.1109/IPACT.2017.8244888.

[15] A. Manna and V. S. K. Bhaaskaran, "Improved read noise margin characteristics for single bit line SRAM cell using adiabatically operated word line," in *2017 International Conference on Nextgen Electronic Technologies: Silicon to Software*, Chennai, India, 2017, pp. 385–393, https://doi.org/10.1109/ICNETS2.2017.8067965.

[16] L. Ammoura, M. L. Flottes, P. Girard, and A. Virazel, "Preliminary Defect Analysis of 8T SRAM Cells for In-Memory Computing Architectures," in *2021 16th International Conference on Design & Technology of Integrated Systems in Nanoscale Era*, Montpellier, France, 2021, pp. 1–4, https://doi.org/10.1109/DTIS53253.2021.9505101.

[17] W. Choi, J. Park, and G. Kang, "Dynamic stability estimation for latch-type voltage sense amplifier," in *2014 International SoC Design Conference*, Jeju, South Korea, 2014, pp. 218–219, https://doi.org/10.1109/ISOCC.2014.7087614.

[18] S. Fairooz, P. Thanapal, P. Ganesan, M. S. Prakash Balaji, and V. Elamaran, "Revisiting the Utility of Transmission Gate and Passtransistor Logic Styles in CMOS VLSI Design," in *2021 3rd International Conference on Signal Processing and Communication*, Coimbatore, India, 2021, pp. 276–280, https://doi.org/10.1109/ICSPC51351.2021.9451645.

[19] M. Kutila, A. Paasio, and T. Lehtonen, "Comparison of 130 nm technology 6T and 8T SRAM cell designs for Near-Threshold operation," in *2014 IEEE 57th International Midwest Symposium on Circuits and Systems*, College Station, TX, USA, 2014, pp. 925–928, https://doi.org/10.1109/MWSCAS.2014.6908567.

[20] D. B. L. Quoc, P. L. Vo, P. N. P. Thien, and L. Tran, "High-Performance In-Memory XNOR Computing: A 65 nm 12T SRAM Architecture for Neural Network Acceleration," *Engineering, Technology & Applied Science Research*, vol. 15, no. 4, pp. 24930–24939, Aug. 2025, https://doi.org/10.48084/etasr.11646.

[21] P. Athe and S. Dasgupta, "A comparative study of 6T, 8T and 9T decanano SRAM cell," in *2009 IEEE Symposium on Industrial Electronics & Applications*, Kuala Lumpur, Malaysia, 2009, pp. 889–894, https://doi.org/10.1109/ISIEA.2009.5356318.

[22] D. Mittal and V. K. Tomar, "Performance Evaluation of 6T, 7T, 8T, and 9T SRAM cell Topologies at 90 nm Technology Node," in *2020 11th International Conference on Computing, Communication and Networking Technologies*, Kharagpur, India, 2020, pp. 1–4, https://doi.org/10.1109/ICCCNT49239.2020.9225554.

[23] Y. Liu *et al.*, "A Compact Model for Relaxation Effect in Analog RRAM for Computation-in-Memory System Design and Benchmark," in *2021 5th IEEE Electron Devices Technology & Manufacturing Conference*, Chengdu, China, 2021, pp. 1–3, https://doi.org/10.1109/EDTM50988.2021.9421000.

[24] A. Chandras and V. S. K. Bhaaskaran, "Sensing schemes of sense amplifier for single-ended SRAM," in *2017 International Conference on Nextgen Electronic Technologies: Silicon to Software*, Chennai, India, 2017, pp. 379–384, https://doi.org/10.1109/ICNETS2.2017.8067964.

[25] H. Jeong *et al.*, "Switching pMOS Sense Amplifier for High-Density Low-Voltage Single-Ended SRAM," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 62, no. 6, pp. 1555–1563, 2015, https://doi.org/10.1109/TCSI.2015.2415171.

[26] V. N. R and S. Hiremath, "Development of Verification Infrastructure to Validate RAM Low Power Features at Unit Level," *International Journal of Engineering Research & Technology*, vol. 11, no. 7, pp. 310–314, July 2022, https://doi.org/10.5281/zenodo.18412837.