# FinFET Based Computation Intensive Sub-Circuits Designs for Low Power VLSI Application

Shyamosree Goswami, Nadegouni Sandeep, Dara Sai Kumar, and Anup Dandapat

***Abstract*—In order to improve operational efficiency for low-power VLSI (Very Large Scale Integration) designs, the current research presents a simulation-based comparison of the results of basic gates, computation-intensive circuits, and basic memory blocks designed using FinFET technology. This creative designs in multipliers effectively consolidates input data by stacking blocks, which significantly reduces time in next-stage operations. To assess the suggested compressor design in terms of average power, delay, and Power Delay Product (PDP), extensive simulations and analyses are carried out. When conducted within the same technological and environmental conditions as existing designs, proposed designes demonstrate clear advantages. The proposed compressor shows an amazing 64.91% decrease in delay, a significant 87.85% improvement in average power consumption, and a notable 95.74% improvement in energy efficiency for proposed 4-3 compressor and similar outcomes for 5-3 compressor. In memory design, it also demonstrates a notable enhancement, achieving an 87.95% reduction in average power, a 49.57% decrease in delay, and an 87.57% improvement in Energy Delay Product (EDP) compared to both conventional NAND-based and FinFET-based Content Addressable Memory (CAM) designs.**

***Index Terms*—Compressors, CAM, FinFET, High speed, Multiplier, Stacking, Time delay.**

*Original Research Paper*
*DOI: 10.53314/ELS2630021G*

## I. Introduction

MODERN semiconductor technology depends on VLSI (Very-Large-Scale Integration) architecture that consumes less power. This design is motivated by the growing demand for portable and battery-powered gadgets like smartphones, tablets, and wearable electronics. The main goal of low power VLSI design is to reduce power consumption while maintaining optimal performance and functionality. To achieve this goal, it is necessary to use a variety of methods at several levels of design, including architectural, circuit, and process levels. Memory elements and combinational circuits constitute fundamental components of VLSI systems, playing a pivotal role in the development of low-power computing devices. Memory elements, such as SRAM and DRAM, are indispensable for efficient data storage and retrieval, directly influencing power consumption and system performance. Combinational circuits, which execute logic operations without the need for data storage, are critical for performing arithmetic and logic functions with high efficiency and speed. The synergy between these elements ensures optimal data processing and minimizes energy expenditure, thereby enhancing the functionality and power efficiency of advanced VLSI systems. Their intricate design and operation are paramount in driving the evolution of energy-efficient computing technologies.

As semiconductor technology progresses towards lower nodes, the importance of low power VLSI design increases. This progression requires a thorough and all-encompassing design strategy that incorporates cutting-edge materials, inventive device structures, and superior power management methods. The goal is to achieve extended battery life, minimise heat generation, and improve the dependability of electronic systems.

At the manufacturing level, the use of sophisticated transistor technologies like FinFETs (Fin Field-Effect Transistors) has shown to be crucial in improving control over leakage currents [1]. These structures in three dimensions offer excellent control over the gate, which is crucial for maintaining low levels of leakage currents and allowing operation at lower supply voltages.

FinFET technology is employed in semiconductor devices for low-power applications because it offers several significant advantages compared to conventional planar transistors. The main factors that make FinFETs preferable for low-power applications are [1], [2]:

- **Improved Control Over Channel**
  1) FinFETs offer enhanced electrostatic control over the channel, resulting in a reduction of short-channel effects commonly observed in conventional planar transistors as device dimensions decrease.
  2) FinFETs have a three-dimensional structure that enables more precise control of the gate over the channel. This reduces leakage currents and enhances the switching performance of the transistor.
- **Lower Leakage Currents**
  1) Subthreshold Leakage: FinFETs exhibit considerably reduced subthreshold leakage currents in comparison to planar transistors. The reason for this is that the gate completely encloses the channel on three sides, resulting in enhanced control and less leakage.
  2) Gate-Induced Drain Leakage (GIDL) is reduced in FinFETs due to increased gate control, resulting in

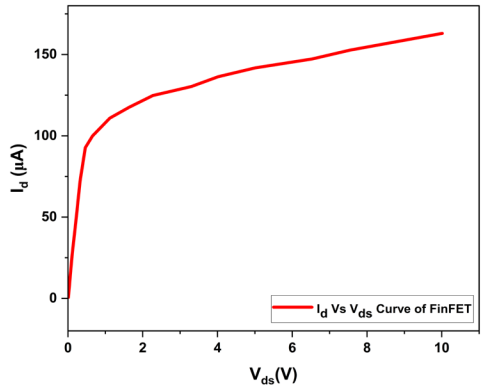
Manuscript received on January 31st, 2025. Received in revised form on September 22nd, 2025. Accepted for publication on October 30th, 2025.

Shyamosree Goswami (e-mail: shyamosreegoswami@gmail.com, ORCID: 0009-0002-5498-1740), Nadegouni Sandeep (e-mail: sandeepnadegouni7282@gmail.com, ORCID: 0009-0008-9167-0865), Dara Sai Kumar (e-mail: saikumardhara923@gmail.com, ORCID: 0009-0003-9113-9269) and Anup Dandapat (e-mail: anup.dandapat@gmail.com, ORCID: 0000-0002-0997-1220) are with the Department of Electronics and Communication Engineering, National Institute of Technology Meghalaya, Sohra 793108, India.

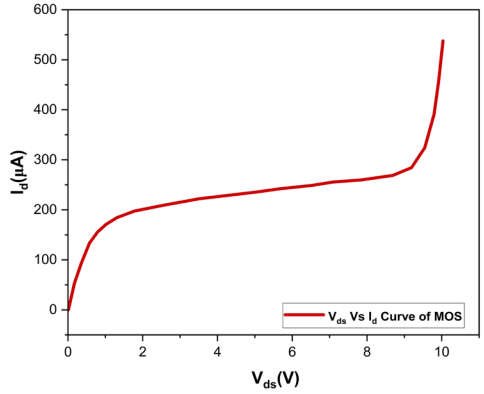
decreased power usage.

- **Lower Supply Voltage**
  1) FinFETs exhibit superior electrostatic control, enabling them to function efficiently at reduced supply voltages. This characteristic leads to a significant decrease in power consumption, as dynamic power consumption is directly proportional to the square of the supply voltage.
- **Scalability**
  1) FinFET technology allows for continued scaling of transistors beyond the limits of planar technology, enabling more transistors to be packed into the same area without significantly increasing power consumption.

This study involved doing extensive simulations of all basic logic gates and other combinational circuits using FinFET technology. The simulations were compared to equivalent designs using standard CMOS technology to assess performance parameters and power efficiency. In addition, we have thoroughly analyzed the performance of FinFETs at the 20 nm technology node, specifically studying their electrical properties and operational benefits compared to conventional planar transistors.




(a) $I_d$ vs $V_{ds}$ curve of FinFET




(b) $I_d$ vs $V_{ds}$ curve of MOSFET

Fig. 1. Characteristics Curve of CMOS and FinFET.

The inquiry entailed the analysis of crucial delay, power dissipation, and overall circuit reliability. Fig. 1 shows the characteristics curve of CMOS and FinFET. We conducted a thorough comparison of FinFET and CMOS technologies by modelling fundamental logic gates including AND, OR, NOT, NAND, and NOR, as well as complex combinational devices such as multiplexers and adders. This work emphasises the influence of FinFET's three-dimensional configuration on improving gate control and mitigating short-channel effects, which are significant obstacles in sub-20 nm CMOS technology. The paper is organised as the basic of FinFET is described in Section II, Section III presents the related works, Section IV presents the simulated results and Section V discusses the conclusion.

## II. Overview of FinFET Technology

A FinFET is a "fin" field-effect transistor (FET). It is a metal-oxide-semiconductor (MOS) device where the gate is placed on two sides of a thin vertical semiconductor body standing on a substrate as shown in Fig. 2. FinFETs, known as Multigate Field Effect Transistor (MGFET), are predicted to be one of the best promising devices to substitute bulk MOSFET due to its improved SS (Subthreshold Swing) slope, better stability, higher $(I_{on}/I_{off})$ ratio, better short channel performance, smaller intrinsic gate capacitance. Here is the main structure of MOSFET and FinFET gate [3].

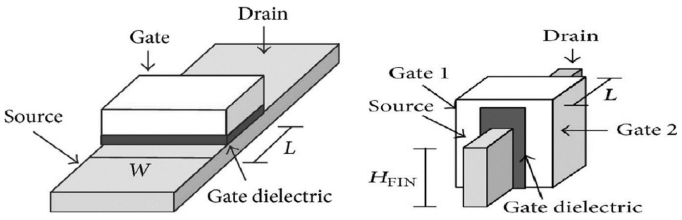



Fig. 2. Basic Structure of conventional CMOS and FinFET.

### A. Characteristics of a FinFET

The I-V (current-voltage) behavior of FinFETs closely resembles that of conventional MOSFETs but exhibits several unique traits.

- Threshold Voltage ($V_{th}$): In FinFETs, the threshold voltage refers to the gate voltage level at which the device begins to allow current flow. Owing to better gate control over the channel, this voltage is generally lower compared to that of traditional MOSFETs [3].
- Drain-Source Current ($I_{ds}$): This indicates the current flowing between the drain and source terminals. It is influenced by both the gate-to-source voltage ($V_{gs}$) and the drain-to-source voltage ($V_{ds}$). As $V_{gs}$ rises, $I_{ds}$ also increases, eventually entering a saturation phase where further increase in $V_{gs}$ results in minimal change in $I_{ds}$ [3].

After observing both the plots in Fig. 1 we can observe that in CMOS after reaching certain voltage with increase in $V_{DS}$ continuously a breakdown has occurred due to the SCE which can be minimized in FinFET based designs. The finFET parameters of 20nm model is shown in Table I [1].

TABLE I
PARAMETERS OF FINFET [4]

| FinFET parameters | Technology (nm) | $V_{DD}$(V) | Leff (nm) | hfin (nm) | tfin (nm) | toxide (nm) |
|---|---|---|---|---|---|---|
| Value | 20 | 0.9 | 24 | 28 | 15 | 0.9 |

## III. RELATED WORKS

There are several methods for achieving low power design in VLSI systems, each targeting different aspects of the design process to minimize energy consumption while maintaining performance. Power gating is a widely used technique that reduces leakage power by turning off the power supply to inactive modules, Dynamic voltage and frequency scaling (DVFS) adjusts the voltage and frequency according to the workload, Multi-threshold CMOS (MTCMOS) technology employs transistors with different threshold voltages within the same design, balancing speed and power efficiency by using low-threshold transistors for critical paths and high-threshold transistors for non-critical paths to reduce leakage.the use of advanced transistor technologies like FinFETs enhances control over leakage currents and allows for operation at lower voltages, further contributing to power savings. The basic gates can be implemented using the FinFET technology for better performance over MOS design. Again multiplication, an intricate and time-intensive arithmetic operation, plays a pivotal role in numerous signal processing algorithms [5]–[8]. Recognizing its significance, the multiplication process is systematically delineated into three key stages: the generation of partial products, the subsequent reduction of these partial products, and the ultimate computation of the resultant product [3], [6], [7], [9], [10]. The exploration of various methodologies for optimization at each stage has become a focal point of extensive research. This includes innovative compressor designs and the introduction of time-efficient structures [11]–[16].

A content addressable memory (CAM) utilizes specialized comparison logic to perform table lookup operations within a single clock cycle [17]–[19]. CAMs are widely used in applications that demand high-speed search capabilities. In this work, the study and analysis of various CAM architectures are centered around comparison circuits based on N-type, P-type, and Transmission Gate (TG) configurations [20]–[22]. The DLPCAM design [23] proposed a TCAM memory cell using FinFET technology and evaluated key parameters such as search delay, average power consumption, and energy usage in a 32-nm process. On the other hand, PVECAM investigated the impact of process variations on FinFET-based CAMs using a 20-nm HSPICE model [24].

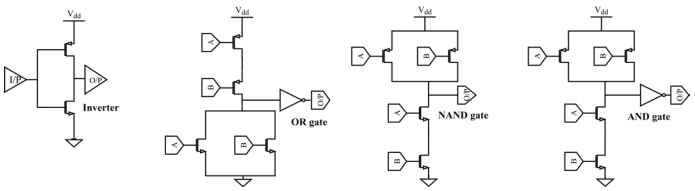



Fig. 3. Schematics of basic gates.

## IV. COMPUTATION INTENSIVE CIRCUIT DESIGN AND SIMULATION

### A. Logic gates Design using FinFET

In this work we have designed basic gates, inverter as shown in Fig. 3 and few combination circuits like compressor, Multiplier using FinFET and compare their average power consumption and Delay for all the designs.

The comparison Table II clearly shows that FinFET-based designs outperform CMOS-based designs. For low-power design, using FinFETs represents a significant advancement. FinFETs can be used in lower technology nodes without being affected by short channel effects, making them more area-efficient as well.

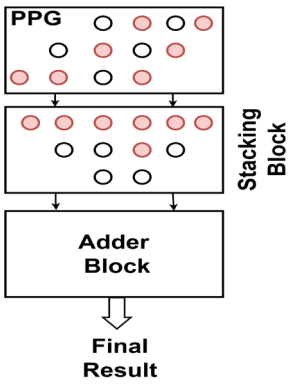



Fig. 4. Architecture of compressor circuit.

### B. Compressor Design using FinFET

Compressors are basically internal components of multipliers. In the context of digital circuit design, compressors are specialized circuits used within multipliers to efficiently reduce and process partial products during multiplication operations. These compressors help optimize the overall speed and efficiency of the multiplication process by reducing the critical path delay, which is essential for high-performance applications. Compressors play a vital role in handling binary data and are integral to the internal architecture of multipliers. Their effective implementation is crucial for achieving fast and accurate multiplication in various digital systems.

Over the last two decades, research has focused on the use of various compressors for partial product reduction. Examples include the 3-2, 4-2, 5-2, and 5-3 compressors. Nonetheless, it is important to note that these compressors perform best when the multiplier size is minimal. Larger multipliers, like 16×16 and 32×32 bit combinations, require huge compressor

TABLE II
COMPARISON RESULT OF LOGIC GATE USING FINFET AND TRADITIONAL CMOS

| | INVERTER | | NAND | | AND | | OR | |
|---|---|---|---|---|---|---|---|---|
| | FinFET | Traditional | FinFET | Traditional | FinFET | Traditional | FinFET | Traditional |
| DELAY(ps) | 1.122 | 3.270 | 1.549 | 7.490 | 4.804 | 19.18 | 4.828 | 20.020 |
| POWER (nW) | 9.992 | 69.83 | 22.88 | 58.14 | 29.55 | 134.82 | 39.53 | 102.06 |

sizes. In this setting, high-order compressors appear as a feasible option, providing greater power efficiency and speed. However, it is important to remember that using high-order compressors has trade-offs because their implementation takes up more silicon area than their low-order counterparts. The complex trade-offs between power, speed, and performance efficiency highlight the subtle difficulties in fine-tuning the partial product reduction phase of large-scale multipliers.

In this paper, we provide a novel stacked-based compressor in which a specific stacking step is used to sort partial product data. In this phase, ones (1s) in a column are systematically moved to the Most Significant Bits (MSBs), and zeros (0s) are simultaneously moved to the Least Significant Bits (LSBs). We have created two compressor designs in the proposed design: a 4:3 and a 5:3 compressor block diagram shown in Figs. 4 and 5. Multipliers make heavy use of both designs. In order to make the design process easier, we have purposefully divided the entire circuit into a well-organized set of separate functional components. Within this structure, two essential blocks are carefully defined:

- **Stacking Block:** Data compression takes on a new dimension when stacking blocks are incorporated into multiplier design. Here, a stacking block combines the input data by piling up the number of 1s without taking into account their particular place values. For instance, the stacking block output, given the input "1010," combines into "1100," which represents the count of 1s.
  In the proposed design, two different stacks a 4-bit stack for the 4:3 compressor and a 5-bit stack for the 5:3 compressor are integrated, each specifically designed for a different compressor. These stacking blocks are essential to the compression process since they maximise data representation for multiplier stages that come after. When we examine the details of this design, one important aspect that stands out as being able to improve performance and efficiency in multiplication operations is the use of two stacks that are specifically designed for separate compressors. We implemented the circuit based on the Boolean equation we derived for proposed design of the 4:3 and 5:3 compressor stack blocks.
  Let's consider the inputs to be $a_0$, $a_1$, $a_2$, $a_3$ and respective outputs be $b_1$, $b_2$, $b_3$, $b_4$.

$$b_1 = a_0 + a_1 + a_2 + a_3 \tag{1}$$
$$b_2 = a_0(a_2 + a_3) + a_1(a_0 + a_2) + a_3(a_1 + a_2) \tag{2}$$
$$b_3 = a_3(a_0a_2 + a_1a_2 + a_0a_1) + a_0a_1a_2 \tag{3}$$
$$b_4 = a_0a_1a_2a_3 \tag{4}$$

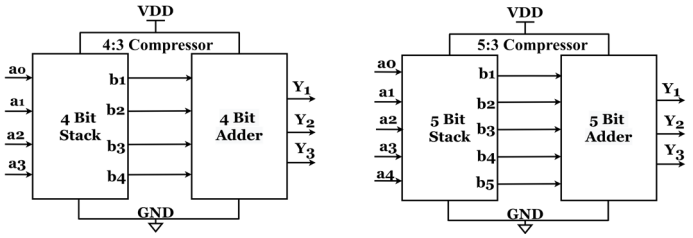



Fig. 5. Block diagram of Compressor.

Similar to the 4:3 compressor stack block, we have derived the boolean expression of the 5-bit stack for the proposed design.

$$b_1 = a_0 + a_1 + a_2 + a_3 + a_4 \tag{5}$$
$$b_2 = (a_0a_1) + (a_0a_2) + (a_0a_3) + (a_0a_4) + (a_1a_2) + (a_1a_3) + (a_1a_4) + (a_2a_3) + (a_2a_4) + (a_3a_4) \tag{6}$$
$$b_3 = P_1 + P_2 + P_3 + P_4 + P_5 + P_6 + P_7 + P_8 + P_9 + P_{10} \tag{7}$$

where

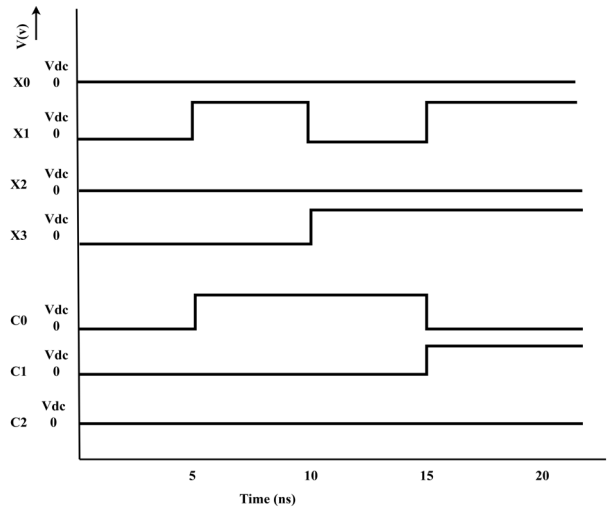
$$P_1 = a_0(a_1a_2), \quad P_2 = a_0(a_1a_3), \quad P_3 = a_0(a_1a_4),$$
$$P_4 = a_0(a_2a_3), \quad P_5 = a_0(a_2a_4), \quad P_6 = a_0(a_3a_4),$$
$$P_7 = a_1(a_2a_3), \quad P_8 = a_1(a_2a_4), \quad P_9 = a_1(a_3a_4),$$
$$P_{10} = a_2(a_3a_4)$$
$$b_4 = P_{11} + P_{12} + P_{13} + P_{14} + P_{15} \tag{8}$$

where

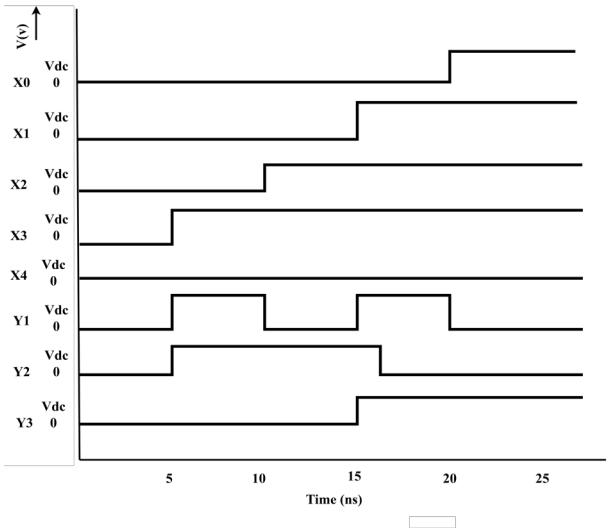
$$P_{11} = (a_0P_7), \quad P_{12} = (a_0P_8), \quad P_{13} = (a_0P_9),$$
$$P_{14} = (a_0P_{10}), \quad P_{15} = (a_1P_{10})$$
$$b_5 = a_0a_1a_2a_3a_4 \tag{9}$$

It is clear that several AND operations are being carried out from equations (1) to (9). The repetition of several AND blocks, which allows us to extract outputs from previously computed results, is a remarkable observation. This discovery offers a chance to reduce circuits because we can make the process more efficient by using previous findings. The stack block's output then plays a crucial role in calculating the summation for the next step, demonstrating a clever use of previously acquired data in the circuit.

- **Adder Block:**The adder block, which produces the compression process's end result, is the next part of the




(a) 4:3 compressor output graph




(b) 5:3 compressor output graph

Fig. 6. Output characteristics of the proposed compressors.

compressor. We assess the input conditions to this block in proposed design strategy rather than employing a half adder or full adder right away. Proposed method takes advantage of the fact that the inputs are already sorted after going through the stacking block. Because of its intrinsic orderliness, the computation of the result is made easier because there is less chance that the least significant bits (LSB) contain 1s. Taking use of this feature, the approach determines the output expression by applying Boolean expression analysis to the input conditions. These phrases are rigorously verified against every possible combination to guarantee that the result is accurate in every scenario. This thorough validation ensures that the design is resilient and reliable and shows that it is effective in generating accurate results for a range of input scenarios.

Let the final results for the 4 bit adder be $Y_1$, $Y_2$, $Y_3$ and

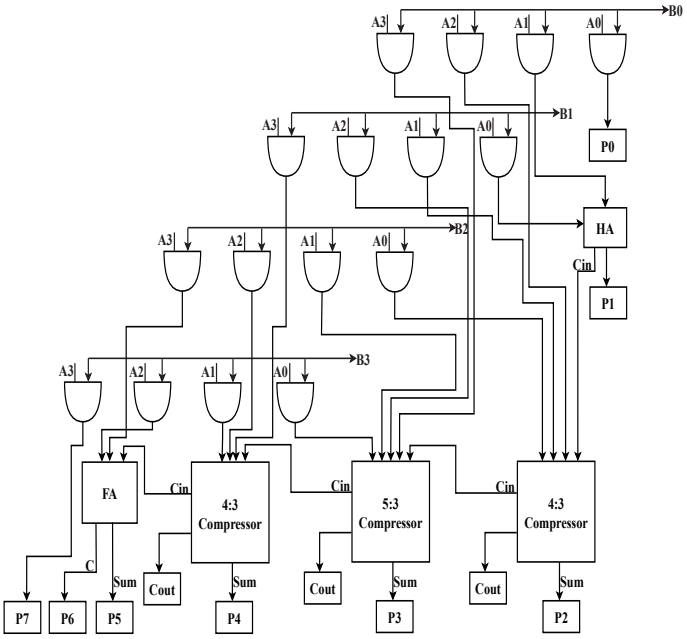



Fig. 7. Block diagram of Multiplier.

the inputs be $b_1$, $b_2$, $b_3$, $b_4$.

$$Y_1 = b_1 b_2' + b_3 b_4' \tag{10}$$
$$Y_2 = b_2 b_3' + b_3 b_4' \tag{11}$$
$$Y_3 = b_1 b_2 b_3 b_4 \tag{12}$$

Similar way, we derived the expression for the 5-bit adder:

$$Y_1 = b_1 b_2' + b_3 b_4' + b_5 \tag{13}$$
$$Y_2 = b_2 b_3' + b_3 b_4' \tag{14}$$
$$Y_3 = b_4 b_5' + b_5 \tag{15}$$

We can see that the number of computing blocks (AND, OR) required for this adding block is less than the conventional 4-bit adder and 5-bit adder. The output graphs of both compressors are shown in Fig. 6, and the performance comparison is tabulated in Table III.

TABLE III
COMPARISON OF POWER, DELAY, AND PDP FOR 4-3 AND 5-3 COMPRESSORS

| Design | Type | Power | Delay | PDP |
|---|---|---|---|---|
| 4-3 Compressor | Conventional | 4.28 $\mu$W | 92.2 ps | 394.62 aJ |
| 4-3 Compressor | Proposed | 0.52 $\mu$W | 32.35 ps | 16.82 aJ |
| 5-3 Compressor | Conventional | 6.58 $\mu$W | 303 ps | 1.9 fJ |
| 5-3 Compressor | Proposed | 1.12 $\mu$W | 72.3 ps | 80.98 aJ |

*1) Application of Compressor:* In the proposed multiplier design as shown in Fig. 7, we have implemented the proposed compressor instead of the conventional method, which

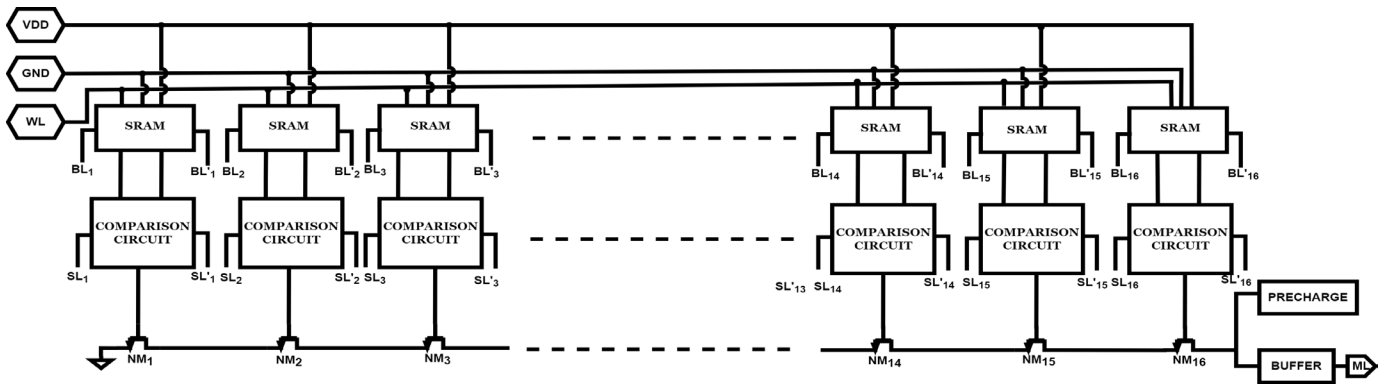
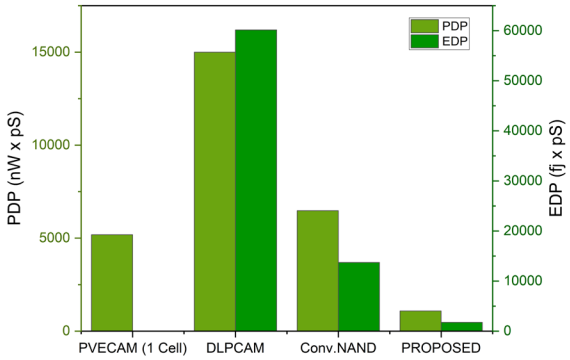
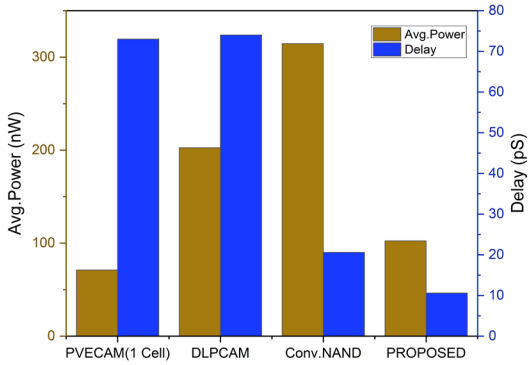



Fig. 8. 16 Bit CAM Array Block Design.




(a) PDP vs EDP comparison




(b) Power vs Delay comparison

Fig. 9. Performance analysis of proposed CAM design.

employs half adders (HA) and full adders (FA) for partial product addition across multiple stages. By integrating the proposed compressor, we significantly reduce the number of addition stages required. This reduction not only minimizes power consumption but also substantially decreases the overall delay. Consequently, the approach demonstrates a marked improvement in efficiency and performance over traditional

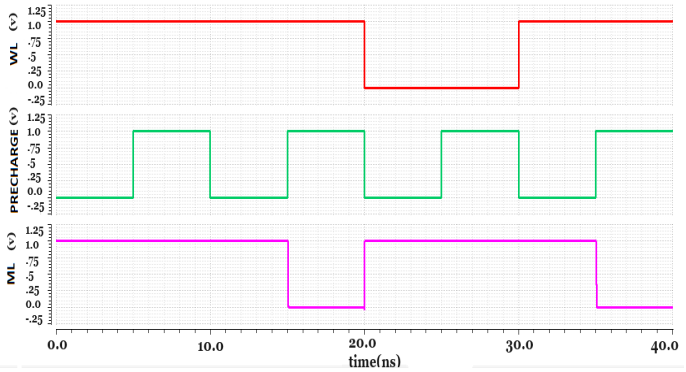



Fig. 10. Timing diagram of 16-bit CAM array using FinFET.

multiplier designs as tabulated in Table IV.

From the comparison table, it can be concluded that the

TABLE IV
COMPARISON TABLE OF MULTIPLIER

| Design | Technology | Voltage | Power ($\mu$W) | Delay | PDP |
|---|---|---|---|---|---|
| Proposed | FinFET | 0.9V | 389 | 337.83ps | 0.13pJ |
| Conv. | CMOS | 0.9V | 695.59 | 8.380ns | 5.82pJ |

TABLE V
PERFORMANCE OF $1 \times 16$ ARRAY

| | DLPCAM [23] | Conv.NAND (MOSFET) | Proposed (FinFET) |
|---|---|---|---|
| Avg.Power (nW) | 202.63 | 850 | 102.40 |
| Delay(ps) | 74.01 | 21 | 10.60 |
| PDP($nW \times ps$) | 14997.14 | 17850 | 1085.44 |
| Energy (fJ) | 813 | 666 | 164 |
| EDP($fJ \times ps$) | 60170.13 | 13986 | 1738 |

utilization of a FinFET-based compressor, as opposed to traditional FA and HA, results in a reduced PDP. This FinFET-based approach is more efficient in terms of delay, which is particularly significant given that multipliers are fundamental components of arithmetic and logical units where speed is a critical factor. Thus, the implementation of FinFET-based compressors enhances the overall performance and efficiency of these operations.

### C. Basic CAM Memory Design Using FinFET

CAM, also called associative memory, is a unique kind of memory that is used to search through the data contained inside. Traditional CAM designs use large and active parallel hardware, which leads to high dynamic power usage. With the move to smaller technology nodes and the shift from planar MOSFETs to multi-gate devices, new design trade-offs are expected in CAMs. FinFET technology helps reduce short channel effects at these smaller scales. In this work, we present a simple 16-bit NAND-type CAM cell shown in Fig. 8 built using FinFETs and evaluate its performance, focusing mainly on average power consumption and delay shown in Fig. 9. Fig. 10 shows the timing diagram of a 16-bit CAM array using FinFET. We have simulated the CAM design and compared it with existing designs available and tabulated the values as tabulated in Tables V and VI.

TABLE VI
PER BIT CAM CELLS PERFORMANCE

| | PVECAM (1CELL) [24] | Conv.NAND (MOSFET) | Proposed (FinFET) |
|---|---|---|---|
| Avg.Power (nW) | 70.89 | 53.13 | 15.69 |
| Delay(ps) | 72.56 | 10.10 | 3.50 |
| PDP($nW \times ps$) | 5143.77 | 536.61 | 54.91 |

## V. CONCLUSION

In conclusion, the analysis of the sub-circuits, which includes FinFET based designs like basic gates, shows improved performance as compared to MOS based designs in terms of power and delay.The stack-based digital compressors and unique adder blocks, showed encouraging results across multiple performance parameters. The simulations and analyses indicate that the proposed concept outperforms previous designs significantly. Specifically, the results show an average power usage improvement of 87.85% for the 4-3 configuration and 82.98% for the 5-3 configuration. Additionally, there is a delay reduction of 64.91% for the 4-3 configuration and 76.14% for the 5-3 configuration. Furthermore, the Power Delay Product (PDP) decreased by 95.74% for the 4-3 configuration and 94.97% for the 5-3 configuration. Proposed design successfully balances power efficiency, speed, and performance, making it a promising leap in multiplier architectures.

Detailed analysis revealed that incorporating FinFET technology into CAM design led to enhanced performance when compared to conventional MOSFET-based CAMs. The design achieved noticeable improvements, reducing average power consumption by 87.95%, delay by 49.57%, and Energy Delay Product (EDP) by 87.57% compared to both traditional NAND-based and earlier FinFET-based CAM implementations [24].

## REFERENCES


[1] M. Masahara, T. Matsukawa, K. Endo, Y.-X. Liu, W. Mizubayashi, S. Migita, S. O'uchi, H. Ota, and Y. Morita, "Advanced finfet process technology for 20 nm node and beyond," in *Proc. IEEE Int. NanoElectronics Conf.*, pp. 1–2, 2011. doi: 10.1109/INEC.2011.5991771.

[2] R. K. Maurya and B. Bhowmick, "Review of finfet devices and perspective on circuit design challenges," *Silicon*, vol. 14, no. 11, pp. 5783–5791, 2022. doi:10.1007/s12633-021-01366-z.

[3] S. Goswami, A. Kumar, and A. Dandapat, "Design of a stack based compressor for high speed computation," in *2024 IEEE 9th International Conference for Convergence in Technology (I2CT)*, pp. 1–5, IEEE, 2024. doi:10.1109/I2CT61223.2024.10544000.

[4] S. Goswami, K. H. Varma, R. Gudla, and A. Dandapat, "Cascam: Comparative analysis of search operation of finfet-and mosfet-based content-addressable memory," in *International Symposium on Sustainable Energy and Technological Advancements*, pp. 423–433, Springer, 2024. $doi : 10.1007/978 - 981 - 97 - 7018 - 2_30$.

[5] A. Fathi, B. Mashoufi, and S. Azizian, "Very fast, high-performance 5-2 and 7-2 compressors in cmos process for rapid parallel accumulations," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 28, no. 6, pp. 1403–1412, 2020. doi:10.1109/TVLSI.2020.2983458.

[6] S. Venkatachalam and S.-B. Ko, "Design of power and area efficient approximate multipliers," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 25, no. 5, pp. 1782–1786, 2017. doi:10.1109/TVLSI.2016.2643639.

[7] W. Guo and S. Li, "Fast binary counters and compressors generated by sorting network," *IEEE Transactions on very large scale integration (VLSI) systems*, vol. 29, no. 6, pp. 1220–1230, 2021. doi:10.1109/TVLSI.2021.3067010.

[8] M. Ha and S. Lee, "Multipliers with approximate 4–2 compressors and error recovery modules," *IEEE Embedded Systems Letters*, vol. 10, no. 1, pp. 6–9, 2017. doi:10.1109/LES.2017.2746084.

[9] R. Marimuthu, Y. E. Rezinold, and P. S. Mallick, "Design and analysis of multiplier using approximate 15-4 compressor," *IEEE Access*, vol. 5, pp. 1027–1036, 2016. doi:10.1109/ACCESS.2016.2636128.

[10] A. Dandapat, S. Ghosal, P. Sarkar, and D. Mukhopadhyay, "A 1.2-ns16× 16-bit binary multiplier using high speed compressors," *International Journal of Electrical and Electronics Engineering*, vol. 4, no. 3, pp. 234–239, 2010.

[11] A. Momeni, J. Han, P. Montuschi, and F. Lombardi, "Design and analysis of approximate compressors for multiplication," *IEEE Transactions on Computers*, vol. 64, no. 4, pp. 984–994, 2014. doi:10.1109/TC.2014.2308214.

[12] K. M. Reddy, M. H. Vasantha, Y. N. Kumar, and D. Dwivedi, "Design of approximate booth squarer for error-tolerant computing," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 28, no. 5, pp. 1230–1241, 2020. doi:10.1109/TVLSI.2020.2976131.

[13] C. Fritz and A. T. Fam, "Fast binary counters based on symmetric stacking," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 25, no. 10, pp. 2971–2975, 2017. doi:10.1109/TVLSI.2017.2723475.

[14] T. Satish and K. S. Pande, "Multiplier using nand based compressors," in *2019 3rd International Conference on Electronics, Materials Engineering & Nano-Technology (IEMENTech)*, pp. 1–6, IEEE, 2019. doi:10.1109/IEMENTech48150.2019.8981067.

[15] C. S. Wallace, "A suggestion for a fast multiplier," *IEEE Transactions on Electronic Computers*, vol. EC-13, no. 1, pp. 14–17, 1964. doi:10.1109/PGEC.1964.263830.

[16] X. Yi, H. Pei, Z. Zhang, H. Zhou, and Y. He, "Design of an energy-efficient approximate compressor for error-resilient multiplications," in *2019 IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 1–5, IEEE, 2019. doi:10.1109/ISCAS.2019.8702199.

[17] L. Chisvin and R. J. Duckworth, "Content-addressable and associative memory: Alternatives to the ubiquitous ram," *Computer*, vol. 22, no. 7, pp. 51–64, 1989. doi:10.1109/2.30732.

[18] K. Pagiamtzis and A. Sheikholeslami, "A low-power content-addressable memory (cam) using pipelined hierarchical search scheme," *IEEE Journal of Solid-State Circuits*, vol. 39, no. 9, pp. 1512–1519, 2004. doi:10.1109/JSSC.2004.831433.

[19] C. A. Zukowski and S.-Y. Wang, "Use of selective precharge for low-power content-addressable memories," in *1997 IEEE International Symposium on Circuits and Systems (ISCAS)*, vol. 3, pp. 1788–1791, IEEE, 1997. doi:10.1109/ISCAS.1997.621492.

[20] S. Kumar, A. Noor, B. K. Kaushik, and B. Kumar, "Design of ternary content addressable memory (tcam) with 180 nm," in *2011 International Conference on Devices and Communications (ICDeCom)*, pp. 1–5, IEEE, 2011. doi:10.1109/ICDECOM.2011.5738528.

[21] T. V. Mahendra, S. W. Hussain, S. Mishra, and A. Dandapat, "Performance analysis of n-cam, p-cam and tg-cam using 45-nm technology," in *2019 International Conference on Intelligent Computing and Control Systems (ICCS)*, pp. 621–625, IEEE, 2019. doi:10.1109/ICCS45141.2019.9065404.

[22] T. V. Mahendra, S. W. Hussain, S. Mishra, and A. Dandapat, "Low match-line voltage swing technique for content addressable memory," in *2019 7th International Conference on Smart Computing & Communications (ICSCC)*, pp. 1–5, IEEE, 2019. doi:10.1109/ICSCC.2019.8843590.

[23] M.-C. Chang and K.-L. He, "Design of low-power finfet-based tcams with unevenly-segmented matchlines for routing table applications," in *2015 IEEE 11th International Conference on ASIC (ASICON)*, pp. 1–4, IEEE, 2015. doi:10.1109/ASICON.2015.7517182.

[24] M. Arulvani, M. M. Ismail, and R. M. Issakki, "Process variation effects on finfet based content addressable memory," in *2017 International Conference on Communication and Signal Processing (ICCSP)*, pp. 2106–2110, IEEE, 2017. doi:10.1109/ICCSP.2017.8286776.