INDIAN JOURNAL OF SCIENCE AND TECHNOLOGY

ORIGINAL ARTICLE

Performance Analysis of High Speed Low Power BCD Adder using CMOS and Dynamic logic

K Srilakshmi[1]*, A V S Karthikeya Chowdary[2], D Lakshmi Soumya[3], Ch. Hemasri[4], G Pavan Kumar[5]

1 Associate Professor, Department of Electronics and Communication Engineering, Seshadri Rao Gudlavalleru Engineering College, Andhra Pradesh, India
2 UG Student, Department of Electronics and Communication Engineering, Seshadri Rao Gudlavalleru Engineering College, Andhra Pradesh, India
3 UG Student, Department of Electronics and Communication Engineering, Seshadri Rao Gudlavalleru Engineering College, Andhra Pradesh, India
4 UG Student, Department of Electronics and Communication Engineering, Seshadri Rao Gudlavalleru Engineering College, Andhra Pradesh, India
5 UG Student, Department of Electronics and Communication Engineering, Seshadri Rao Gudlavalleru Engineering College, Andhra Pradesh, India

OPEN ACCESS

Received: 13/04/2025

Accepted: 12/05/2025

Published: 10/06/2025

Citation: Srilakshmi K, Karthikeya Chowdary AVS, Lakshmi Soumya D, Hemasri C, Pavan Kumar G (2025) Performance Analysis of High Speed Low Power BCD Adder using CMOS and Dynamic logic. Indian Journal of Science and Technology 18(21): 1703-1715. https://doi.org/10.17485/IJST/v18i21.700

* **Corresponding author**.

slkaza06@gmail.com

Funding: None

Competing Interests: None

Published By Indian Society for Education and Environment (iSee)

ISSN
Print: 0974-6846
Electronic: 0974-5645

Abstract

Background: In the field of high-speed digital circuits, the efficiency of Binary Coded Decimal (BCD) adders consists of significant importance in optimizing the speed of arithmetic operations in computing systems. BCD arithmetic is crucial in scientific and financial computing systems that require decimal accuracy. **Objectives:** This study examines the performance of BCD adders as designed using two different logic families, Complementary Metal Oxide Semiconductor (CMOS) and dynamic logic. CMOS logic which is meant to have low static power dissipation and dynamic logic which is meant to have high-speed switching capabilities and reduced transistor count. **Methods:** A new dynamic logic realization of a high-speed BCD adder improved from the previously introduced CMOS-based BCD adder is presented in this work. Dynamic logic design keeps the same pull-down network as the equivalent CMOS but with a precharge PMOS and an evaluate NMOS transistor. This design is implemented under 45nm technology and is optimized for a single digit BCD adder. Schematics are made using DSCH, and Verilog files were developed which are simulated using MICROWIND 3.9. The delay and power dissipation values are obtained with an operating voltage of 1V, and a parametric analysis was conducted comparing delay, and power dissipation for varied temperatures in the range (-40°C to 120°C). **Findings:** The findings depict enhanced performance of dynamic logic implementation compared to static CMOS implementations with reduced power dissipation and increased switching speed due to the inherent benefits of dynamic logic. **Novelty:** For comparison purpose this work also implements two different adders along with the static CMOS logic implemented adder which are already present

in literature. The final results depict that dynamic logic implemented adder outperforms all the three adders with a power delay product (PDP) of 14.75 fJ when compared to 72.07 fJ, 80.48 fJ, and 50.92 fJ PDP of other works.
Keywords: BCD adder; CMOS logic; Dynamic logic; Pull down network; Precharge; Evaluate; Parametric analysis; Delay; Power dissipation; Power delay product (PDP)

1 Introduction

Binary Coded Decimal (BCD) adders are specialized circuits that perform decimal addition directly in hardware. Unlike traditional binary adders, which work solely with binary representations, BCD adders use a 4-bit format for each decimal digit (from 0000 to 1001 for values 0-9). These methods[(1)] help to avoid inaccuracies related to conversions, keeping calculations firmly within the decimal system. BCD arithmetic is essential in such situations where precision, financial accuracy, and human readability are crucial. Since decimal numbers are a core element in finance, commerce, and scientific calculations, BCD adders are vital components in contemporary digital systems that require accurate decimal arithmetic.

While regular binary adders can handle the decimal numbers by converting them to their binary equivalents, this method[(1)] can lead to notable computational inefficiencies and inaccuracies. The primary drawback arises from binary arithmetic, which may cause rounding errors due to the restricted precision of floating-point representation. In critical areas such as banking, accounting, tax processing, and scientific research, even the smallest loss of precision can lead to huge financial discrepancies or inaccurate scientific findings.

1.1 Conventional BCD adder

To mitigate these risks, BCD adders work directly in decimal form, producing accurate results without necessitating binary-to-decimal conversions. A conventional BCD adder is shown in the [Figure 1] which illustrates the working of the adder[(1)] in the early days of literature.

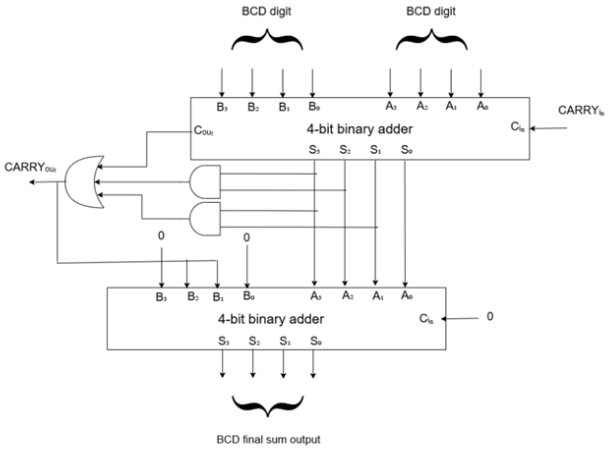

Fig 1. Block diagram of a conventional BCD adder

A standard BCD adder functions in two main phases in which the first phase consists of Initial binary addition and the second phase consists of correction mechanism. Two decimal digits, each represented in 4-bit BCD format, are added using a typical 4-bit binary adder and a carry input from the previous stage in the initial binary addition phase. The correction mechanism involves in addition of result in the first phase with 0110, because valid BCD digits range only from 0000 (0) to 1001 (9), any sum that exceeds 1001 must be corrected. If the total is greater than 9 or if a carry occurs, a correction factor (0110 in binary, which equals +6 in decimal) is added to convert the sum into a valid BCD value.

1.2 Limitations of conventional BCD adder

But this design has many limitations which makes the design inefficient to use in modern digital circuits. some of the limitations are increased delay, higher power consumption and overall area. The extra correction step results in additional computation time, thereby increasing overall propagation delay of the BCD adder. The need for correction logic demands additional transistors which results in greater switching activity and higher power usage. The additional circuitry for correction increases the transistor count, making the overall layout larger and less efficient in terms of area.

1.3 Major advancements in the area

To tackle these limitations there are many BCD adders which had been designed to overcome the disadvantages of the conventional BCD adder. The paper presents a review of various BCD adder designs developed by different authors, highlighting their efficiency in terms of area, power, and delay. It emphasizes the significance of BCD adders in computational systems like ALUs and DSP processors for effective arithmetic processing. The works[(1)],[(2)],[(3)] involves design of a carry look ahead adder and focussing on its performance improvement. The work[(4)] involves design of BCD adder using mirror adder in order to improve speed and reduce area. The work[(5)] involves design of a BCD adder using three-input XOR and majority gates. The work[(6)] involves design of a BCD adder using five-input majority gate using quantum cellular automata (QCA) realization. The works[(7)],[(8)] involves in designing a BCD adder using power gating techniques in order to reduce the power dissipation for future high-speed and low-power electronic applications. The work[(9)] involves design and implementation of high-speed carry look ahead decimal adder (CLDA) Using CMOS technology. The work involves in design of a new dynamic logic circuit model using a NMOS based keeper.

The works[(10)],[(11)],[(12)] are designs of three BCD adders that we are going to explore in this work. The Compact Adder values minimizing transistor area and count, a perfect application for low-power settings. The Novel Adder maximizes carry propagation speed, which allows for faster addition cycles. The Proposed High-Speed BCD Adder, implemented with static CMOS logic, eliminates redundant correction delays, which improves its efficiency for large-scale decimal computation.

1.4 Aim of the work

Our main aim is to realize dynamic logic implementation of the high-speed BCD adder which has been implemented in static CMOS logic in the work[(12)]. Our work also aims to further reduction of the power consumption, improving speed, and minimizing transistor count by employing precharge and evaluate techniques, the proposed design enhances performance while maintaining the accuracy required for decimal arithmetic. The structures of the other adders[(10)],[(11)],[(12)] which are being used as part of comparison purpose are discussed in the above section.

1.5 An overview of existing works

This work also consists of implementation of the other works[(10)],[(11)],[(12)] along with the proposed BCD adder for comparison purpose. The overview of the three BCD adders along with the schematic diagrams are discussed below

1.5.1. Compact adder

The Compact Adder in the work[(10)] is a carry-select adder (CSA) that features a com-pact carry look-ahead unit (CLA) to boost speed and efficiency in area usage. Conventional carry-select adders achieve quicker addition by precomputing two potential sums (one with an assumed carry of 0 and the other with 1), then selecting the appropriate output once the actual input carry is determined. This CLA generates the Pi signal in the same way as the Pi signal is generated in the normal CLA. But the Gi signal is not required here, and the carry-generation is based on the Pi signal only. The schematic diagram of the compact adder is shown in the below [Figure 2].

The key specifications of compact adder are:

1. A compact and static CLA unit that reduces propagation delays.
2. An NMOS pass-transistor-based selection circuit, streamlining the design and cutting overall power usage.
3. A multi-output CLA methodology, which allows propagating carry efficiently throughout several adder stages.

1.5.2. Novel adder

The Novel Adder that has been proposed in the work[(11)] takes the concepts of carry look-ahead (CLA) and parallel prefix adders to further improved critical path delay and power dissipation. The design features an advanced propagate (Pi)-generate (Gi) mechanism that speeds up the compute of carry while minimizing dependence on traditional ripple carry structures. The first change in the Novel CLA is substituting the XOR-based propagate function with a NOR-based function, which makes the circuit simpler and lowers switching power. The second change is using an inverted version of the generate (Gi) signal to further enhance the logic structure. The schematic of the novel BCD adder consisting of 4 blocks which will be generating the carry at each stage and carry should be made to integrate with the inputs in order to produce the final sum output. The [Figure 3] shows the schematic capture of the novel adder.

The key features of the Novel Adder include:

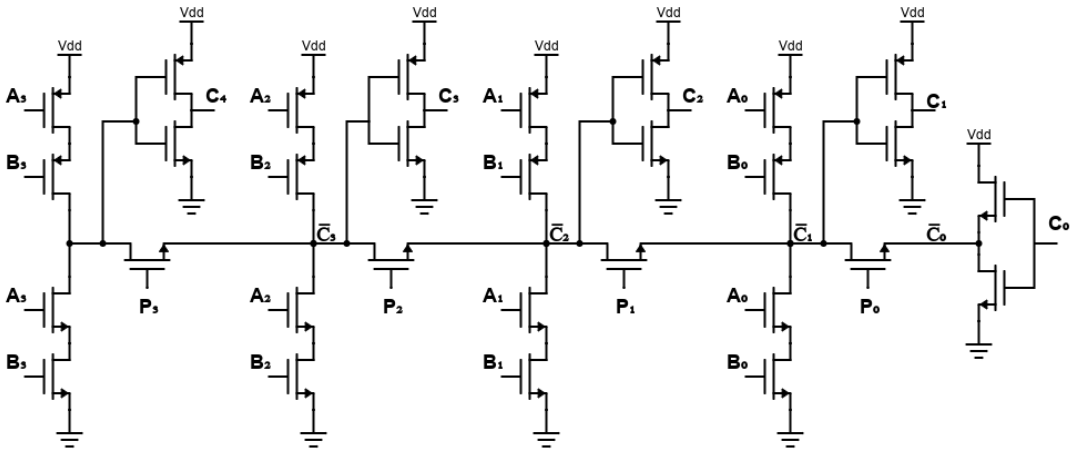

Fig 2. Compact adder

1. Uses a modified CLA method to improve carry computation and generation of propagate and generate signals.
2. The propagate (Pi) logic is implemented using NOR gates instead of XOR gates.

1.5.3. Static Implementation of High-Speed BCD Adder

The Proposed BCD adder high-speed BCD adder in the work[12] is implemented using static CMOS technology with an objective of optimizing speed, power dissipation and circuit complexity. This design is implemented in 45nm, 65nm and 180nm CMOS technologies and has better power delay product (PDP) and speed than the other BCD adders invented before.

The key specifications for the adder in this work are:

1. unlike conventional BCD adders which utilizes a correction phase after binary addition, but this new adder uses a two-level netlist structure with no correction phase and minimized computation delay correction-free design removes the correction circuitry by generating valid BCD arithmetic computation with minimized delay and enhanced speed better than conventional correction-based BCD adders.
2. Two-level netlist structure makes use of two optimized logic blocks called netlist1 for middle computation and netlist2 for obtaining final sum and carry enhancing speed.
3. The first netlist calculates an intermediate value 2 × K by converting the 4-bit inputs A and B into J and I, where $A = (2 \times J) + A_0$ and $B = (2 \times I) + B_0$, where J and I are 3 bit and K = J + I and are done in NAND-NAND optimized CMOS implementation. The least significant bit of 2 × K is always 0, making carry detection easy. The intermediate carry (K_3) and even decimal digits ($K_2K_1K_0$) are passed to Netlist2, improving the addition process by positioning the result correctly for final calculation.
4. The second netlist computes the intermediate result from Netlist1 and input bits A_0, B_0, and Cin to produce the final BCD sum ($S_3S_2S_1S_0$) and carry (Cout). The addition is organized so that 2 × K is properly aligned with the input values to facilitate correct decimal computation. Employing an optimized NAND-NAND CMOS implementation, Netlist2 accurately calculates all carry and sum combinations with minimal logic depth and enhanced speed.

2 Methodology

The proposed work consists of implementation of the high-speed BCD adder which was implemented in static CMOS logic in the work[12] into dynamic logic in order to enhance the performance of the BCD adder. The methodology and design are as follows:

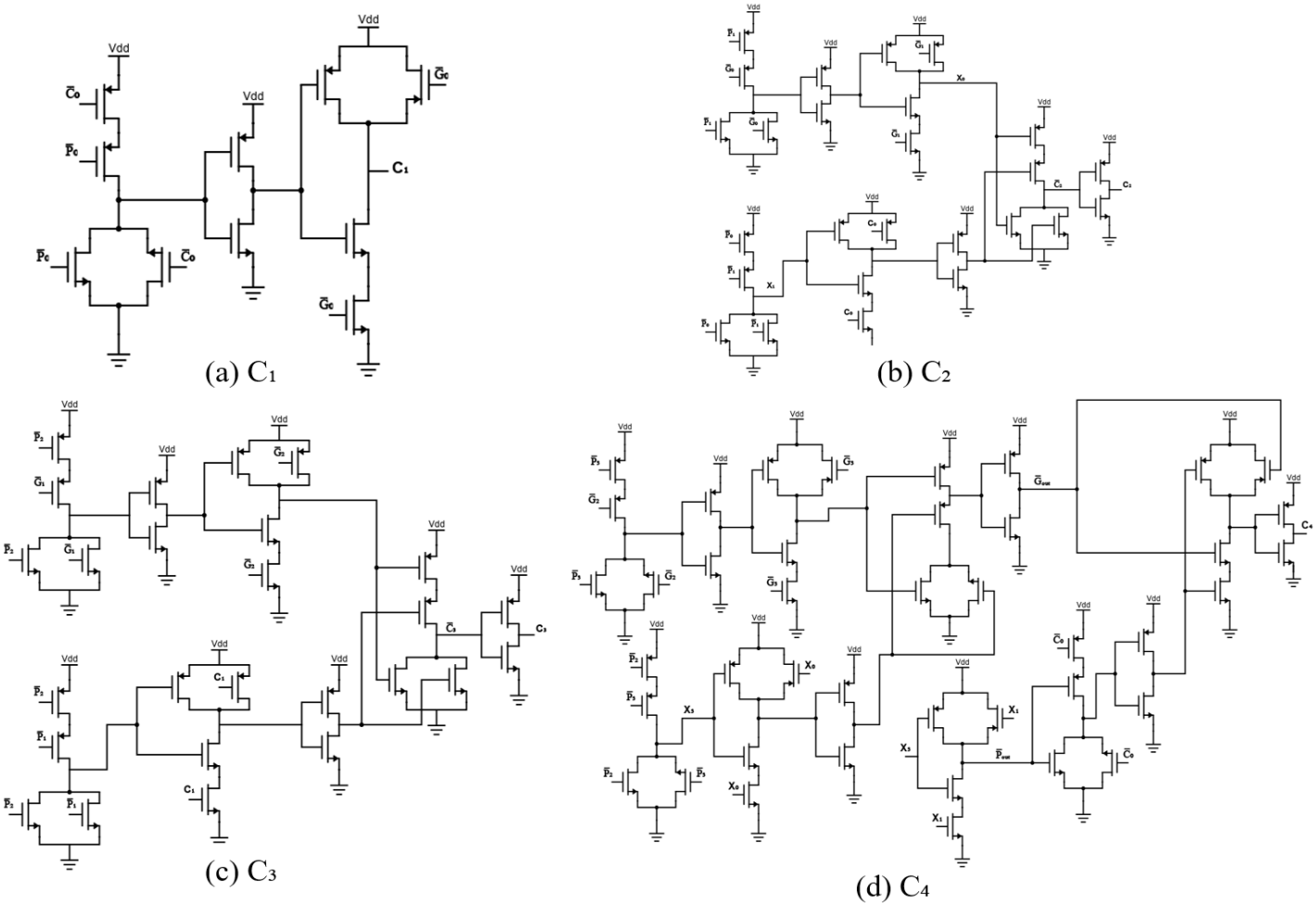

Fig 3. Novel adder (a)C_1 (b)C_2 (c)C_3 (d)C_4

2.1 Introduction to Dynamic Logic

Dynamic logic is a low-power, high-speed alternative to static CMOS logic, which is widely used in high-performance signal processing and arithmetic circuits. Unlike static logic, which contains both pull-up (PMOS) and pull-down (NMOS) networks in every gate dynamic logic involves two distinct phases of operation. They are:

1. ***Precharge Phase (Clock = 0):***
   - The PMOS precharge transistor charges the output node to logic high.
   - The pull-down network is turned off to stop leakage currents.
2. ***Evaluation Phase (Clock = 1):***
   - The NMOS evaluate transistor, pull-down network is enabled
   - Output goes low to 0. In case the value of pull-down path is high, it stays at high to 1.
   - Output is used in the subsequent phase of BCD addition.

The key features of dynamic logic implementation over static logic implementation [9] are

1. ***Speed Enhancement:***

   Dynamic logic prevents excess PMOS transistors from being present in the pull up network, preventing capacitive loading and increased switching speeds. Direct PMOS transitions being avoided also prevent parasitic delays, enabling the evaluation of logic functions to be done at higher speeds.
2. ***Lower Transistor Count:***

   CMOS logic is having high transistor count than dynamic logic due to the existence of dual networks (pull-down and pull-up). The dynamic equivalent provides the same functionality with less transistor count, since there exists only needed a single NMOS pull-down network with a single precharge PMOS transistor per gate. This results in less area consumption, allowing minimized layout design and reduced power dissipation, as fewer switching elements reduce short-circuit power dissipation.
3. ***Reduced Power Dissipation:***

   Although dynamic logic dissipates slightly more power per cycle for clocking, the total power dissipation is lower than in static CMOS when considering the lower per gate capacitance, resulting in lower dynamic power, less redundant switching activity, optimizing energy utilization under high-speed operation and precharge/evaluation cycle control, lowering the leakage power.

2.2 Structure of Proposed Dynamic BCD Adder

The [Figure 4] describes the basic topology of dynamic logic which consists of the following elements which regulates the operation of dynamic logic.

1. ***Pull-down network (PDN*)**: It is a network consisting of NMOS transistors which are preserved from the static CMOS structure but developed further with dynamic logic concepts.
2. ***Precharge PMOS transistor:*** This transistor puts the output into logic high state initially at the start of each clock cycle which results in charging of the load capacitor.
3. ***Evaluation NMOS transistor:*** This transistor will turn on during the clock high state (logic 1) that Regulates the output transition in the evaluate phase provides a path for the capacitor to discharge through the pull-down network if the inputs have been set a path that connects the capacitor to the evaluate transistor.
4. ***Parasitic capacitance utilization:*** The intrinsic capacitance in the layouts of the output node of about 20 fF helps in maintaining logic levels, minimizing the use of external storage devices.

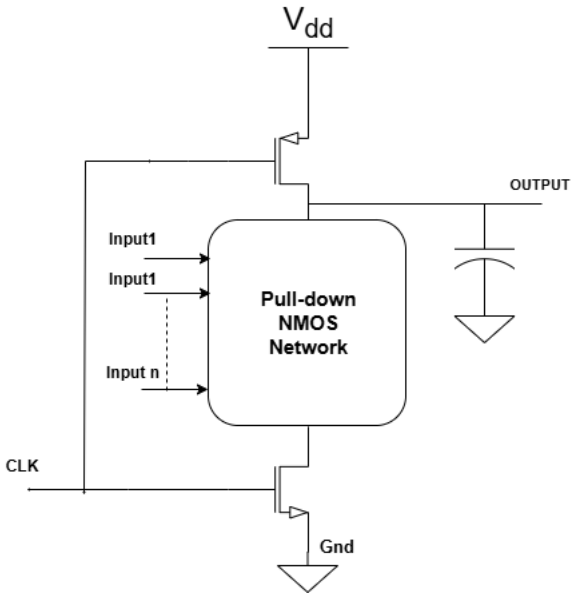

Fig 4. Basic dynamic logic topology

2.2.1. Boolean expressions

The boolean expressions that are shown below for the implementation of this adder are taken from another work[7] and block diagram for the dynamic adder is given in the [Figure 5]

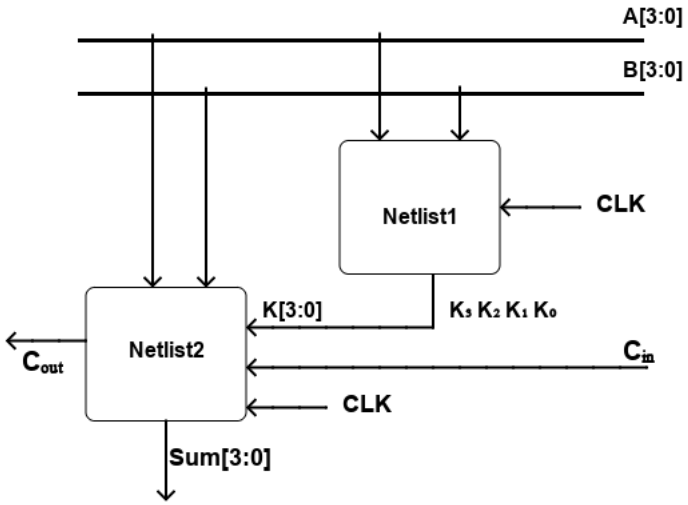

Fig 5. Block diagram of proposed dynamic BCD adder

From the [Figure 5] we have implemented the BCD adder in two level netlist structure where the first netlist takes the inputs A_3 A_2 A_1 from the input A and B_3 B_2 B_1 from input B and produces the outputs K_3 K_2 K_1 K_0. Similarly, the second netlist takes the LSB's of input A and B, input carry C_{in}, and the outputs of the first netlist K_3 K_2 K_1 K_0 as inputs which produces the final 4-bit sum and output carry C_{out}

The Boolean expressions for netlist 1 outputs are given by

$\overline{K}_0 = \overline{X}_0.\overline{X}_1.\overline{X}_2.\overline{X}_3.\overline{X}_4.\overline{X}_5.\overline{X}_6.\overline{X}_7$

$\overline{K}_1 = \overline{X}_0.\overline{X}_8.\overline{X}_9.\overline{X}_{10}.\overline{X}_{11}.\overline{X}_{12}.\overline{X}_{13}.\overline{X}_7$

$\overline{K}_2 = \overline{X}_{14}.\overline{X}_{15}.\overline{X}_{16}.\overline{X}_{17}.\overline{X}_{18}$

$\overline{K}_3 = \overline{X}_{19}.\overline{X}_{20}.\overline{X}_{21}.\overline{X}_{22}.\overline{X}_{23}.\overline{X}_{24}.\overline{X}_7$ and the boolean expressions for the intermediate X terms are given by

$X_0 = \overline{A}_3\overline{A}_2\overline{A}_1B_2B_1$ $X_1 = A_1\overline{B}_3\overline{B}_2\overline{B}_1$

$X_2 = \overline{A}_3\overline{A}_1\overline{B}_2B_1$ $X_3 = \overline{A}_2A_1B_2\overline{B}_1$

$X_4 = A_2A_1B_2B_1$ $X_5 = A_2\overline{A}_1B_3$

$X_6 = A_3B_2\overline{B}_1$ $X_7 = A_3\overline{B}_3$

$X_8 = A_2\overline{B}_3\overline{B}_2\overline{B}_1$ $X_9 = \overline{A}_3\overline{A}_2B_2\overline{B}_1$

$X_{10} = A_2\overline{A}_1\overline{B}_2B_1$ $X_{11} = \overline{A}_2A_1\overline{B}_2B_1$

$X_{12} = A_2A_1\overline{B}_3$ $X_{13} = A_3B_2B_1$

$X_{14} = \overline{A}_3\overline{A}_2\overline{A}_1B_3$ $X_{15} = A_3\overline{B}_3\overline{B}_2\overline{B}_1$

$X_{16} = A_2\overline{A}_1B_2\overline{B}_1$ $X_{17} = A_2A_1\overline{B}_2B_1$

$X_{18} = \overline{A}_2A_1B_2B_1$ $X_{19} = A_2A_1B_2$

$X_{20} = A_2\overline{A}_1B_3$ $X_{21} = A_2B_2B_1$

$X_{22} = A_3B_2\overline{B}_1$ $X_{23} = A_1B_3$

$X_{24} = A_3B_1$

Similarly, the Boolean expressions for netlist 2 outputs are given by

$\overline{S}_0 = \overline{Y}_0.\overline{Y}_1.\overline{Y}_2.\overline{Y}_3$,

$\overline{S}_1 = \overline{Y}_4.\overline{Y}_5.\overline{Y}_6.\overline{Y}_7.\overline{Y}_8.\overline{Y}_9$,

$\overline{S}_2 = \overline{Y}_{10}.\overline{Y}_{11}.\overline{Y}_{12}.\overline{Y}_{13}.\overline{Y}_{14}.\overline{Y}_{15}.\overline{Y}_{16}$,

$\overline{S}_3 = \overline{Y}_{17}.\overline{Y}_{18}.\overline{Y}_{19}.\overline{Y}_{20}.\overline{Y}_{21}.\overline{Y}_{22}$,

$C_{out} = \overline{Y}_{23}.\overline{Y}_{24}.\overline{Y}_{25}.\overline{Y}_{26}$, and the Boolean expressions for the intermediate Y terms are given by

$Y_0 = C_{in}\overline{A}_0\overline{B}_0 \quad Y_1 = C_{in}A_0\overline{B}_0$
$Y_2 = C_{in}\overline{A}_0B_0 \quad Y_3 = C_{in}A_0B_0$
$Y_4 = \overline{K}_2\overline{K}_0C_{in}B_0 \quad Y_{5=}\overline{K}_2\overline{K}_0C_{in}A_0$
$Y_6 = \overline{K}_2\overline{K}_0A_0B_0 \quad Y_7 = K_0\overline{A}_0\overline{B}_0$
$Y_8 = K_0C_{in}\overline{B}_0 \quad Y_9 = K_0C_{in}\overline{A}_0$
$Y_{10} = \overline{K}_1K_0C_{in}B_0 \quad Y_{11} = \overline{K}_1K_0C_{in}A_0$
$Y_{12} = \overline{K}_1K_0A_0B_0 \quad Y_{13} = K_1C_{in}\overline{B}_0$
$Y_{14} = K_1C_{in}\overline{A}_0 \quad Y_{15} = K_1\overline{A}_0\overline{B}_0$
$Y_{16} = K_1\overline{K}_0 \quad Y_{17} = K_1K_0C_{in}B_0$
$Y_{18} = K_1K_0C_{in}A_0 \quad Y_{19} = K_1K_0A_0B_0$
$Y_{20} = K_2C_{in}\overline{B}_0 \quad Y_{21} = K_2C_{in}\overline{A}_0$
$Y_{22} = K_2\overline{A}_0\overline{B}_0 \quad Y_{23} = K_2C_{in}B_0$
$Y_{24} = K_2C_{in}A_0 \quad Y_{25} = K_2A_0B_0$
$Y_{26} = K_3$

2.2.2. Schematic diagrams

The schematics for the above boolean expressions are drawn in [Figure 6, 7] and the simulation of the schematics will be explained in the implementation section. These schematics below are implemented using dynamic logic which is having a precharge transistor at top and an evaluation transistor at the bottom. The first netlist takes the inputs A_3 A_2 A_1 from the input A and B_3 B_2 B_1 from input B and produces the outputs K_3 K_2 K_1 K_0. The schematics of K_3 K_2 K_1 K_0 are shown in the [Figure 6].

Similarly, the second netlist takes the LSB's of input A and B, input carry C_{in}, and the outputs of the first netlist K_3 K_2 K_1 K_0 as inputs which produces the final 4-bit sum S_3 S_2 S_1 S_0 and output carry C_{out}. The schematics of S_3 S_2 S_1 S_0 are shown in the below [Figure 7].

2.3 Implementation

The proposed dynamic logic-based BCD adder is designed and simulated with DSCH for schematics creation and Verilog files are also generated which are compiled using microwind 3.9 for layout design and performance simulation. The circuit is implemented using 45nm CMOS technology, which makes it compatible with today's low-power, high-speed applications. Then the layouts of the adders are compiled and obtained the transient analysis and checked for various combinations of the inputs and verified the outputs

The layout goes through DRC (Design Rule Check) verification to ensure that it adheres to the 45nm fabrication rules. The circuit is subsequently simulated under various conditions in order to study its electrical parameters, such as propagation delay, power dissipation, and transient response. The transistor count of the adder is obtained through extracting the layout which gives the PMOS and NMOS transistor count. The height, width, and the surface area of the layout also obtained through the extraction.

For the purpose of analysing the performance parameters like propagation delay and power dissipation are observed under different conditions. The operating voltage is set to 1V, and the results are obtained through the performance analysis in the analysis section. Similarly, the thermal stability of the circuit is assessed by varying temperature from -40°C to 120°C and the reading of the power dissipation and delay are obtained. The simulated results obtained are helpful in understanding the efficiency of the dynamic logic design in high-speed arithmetic circuits which are very much required in many operations in digital systems.

3 Results and Discussion

The parameter readings are taken through the transient analysis of the BCD adder by compiling its layout using microwind 3.9 software. The parameters like propagation delay, power dissipation were taken at the time of simulation and the surface area and transistor count were taken in the general properties section.

For a clear understanding of how our adder outperforms all the three remaining adders[10],[11],[12]. We will discuss each parameter using data visualization and will discuss about the percentage of the performance. We will rename the adders with some symbols for better understanding in the data visualizations shown in the [Figures 8, 9, 10]. The data visualizations are made using clustered columns for delay, power, power delay product (PDP), transistor count, and area.

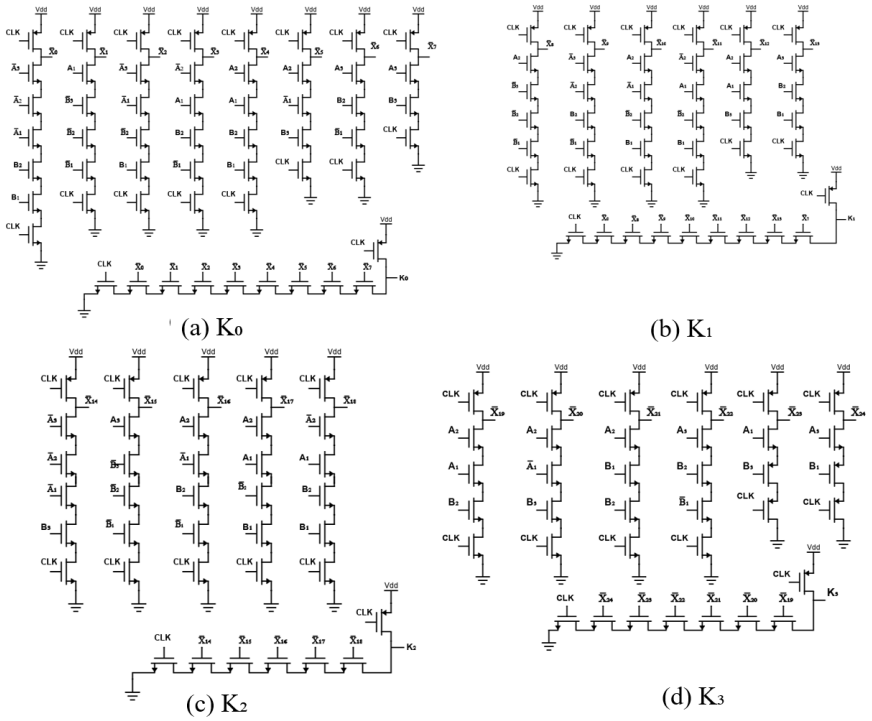

Fig 6. Netlist 1 (a)K_0 (b)K_1 (c)K_2 (d)K_3

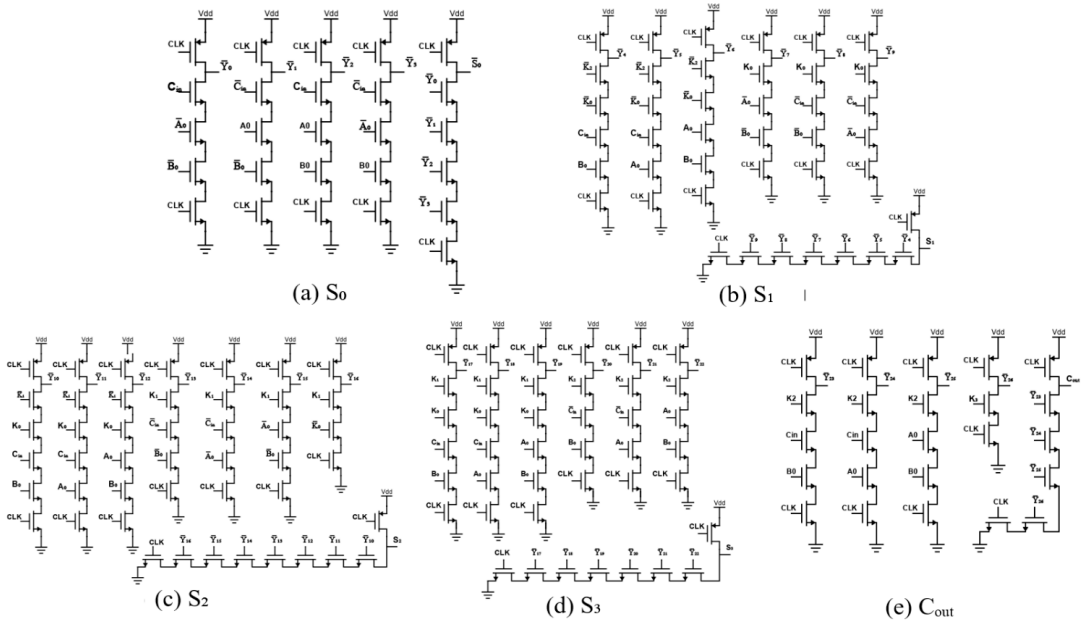

Fig 7. Netlist 2 (a)S_0 (b)S_1 (c)S_2 (d)S_3 (e)C_{out}

The following [Table 1] shows the symbolic representation of the four BCD adders which makes to understand the comparison between other adders and proposed adder using the data visualizations shown in the [Figure 8, 9, 10].

Table 1. Symbols used to refer the four BCD adders for comparison purpose

BCD Adder	Symbol
Compact Adder [10]	ADD_1
Novel Adder [11]	ADD_2
High-speed BCD Adder (Static) [12]	ADD_3
Proposed Dynamic BCD Adder	Proposed

The clustered column representation in the [Figures 8, 9] shows the power dissipation which is measured in μW and visualized in multiple of 10 μW, propagation delay which is measured in ps which has been taken from LSB of the input to the MSB of the final sum output of the BCD adder, and the product of the power dissipation and the delay called power delay product which is the most crucial parameter of any digital circuit which is measured in fJ. The less the power delay product more efficient the circuit will be. The final readings depicts that our dynamic logic implemented adder outperforms all the three adders with a power delay product (PDP) of 14.75 fJ when compared to 72.07 fJ, 80.48 fJ, and 50.92 fJ PDP for other works.

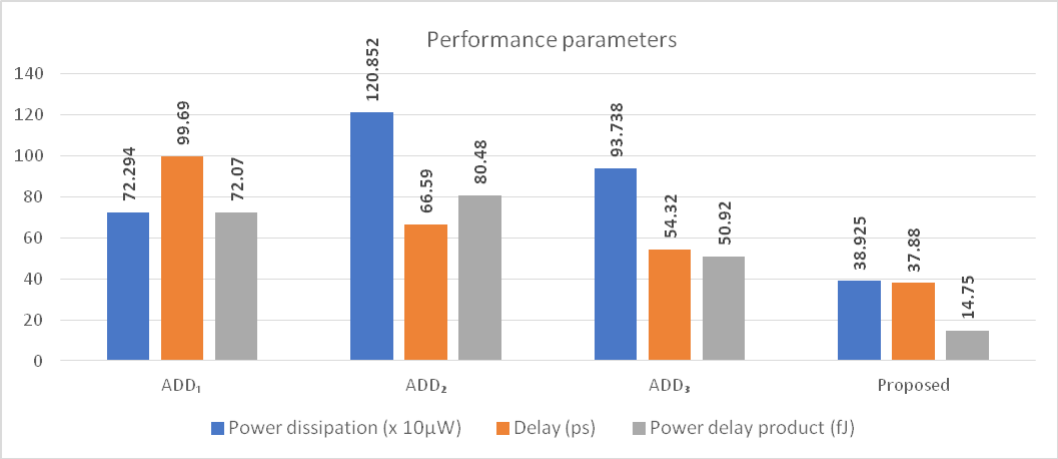

Fig 8. Power dissipation, Delay, PDP of BCD adders

The performance parameters are shown in the below [Figures 9(a), 9(b), 9(c), 9(d)]

Similarly, from the [Figure 9(b) and 9(c)] the power dissipation of high-speed dynamic BCD adder dissipates 43% less power and 62% more speeder when compared to compact adder, dissipates 67% less power and 43% more speeder when compared to novel adder, dissipates 59% less power and 30% more speeder when compared to static implementation of high-speed adder. The only limitation of the proposed adder is it consumes 119% more area, and 239 more transistors required when compared to compact adder, also consumes 23% more area and 48 more transistors required when compared to novel adder, but it completely outperforms the static implementation of the adder in all aspects of comparison with 25% less area and 124 less transistors. Although area can be minimized using various techniques in the technological era of digital systems, the proposed adder is superior among all the adders we have discussed above.

In order to check the stability and robustness of the BCD adder due to temperature variation, we also performed the temperature analysis where the values of power dissipation and delay were measured with varying temperature. The temperature variation range is from -40 °C to 120 °C. The results shown in the [Figure 10(a), 10(b)] are taken from the parametric analysis for checking the stability of the BCD adder circuit.

The data visualization of the power dissipation with varying temperature which is measured in μW is shown in [Figure 10(a)], Similarly, the data visualization of the delay with varying temperature which is measured in ps is shown in [Figure 10(b)].

The [Figure 10] depicts that the dynamic logic implementation of the BCD adder that we have built is stable and robust to temperature variation. From the [Figure 10(a), 10(b)] the proposed adder shows a better performance in power dissipation and delay than other works [10], [11], [12] even with the variation of temperature showcasing its temperature stability and robustness.

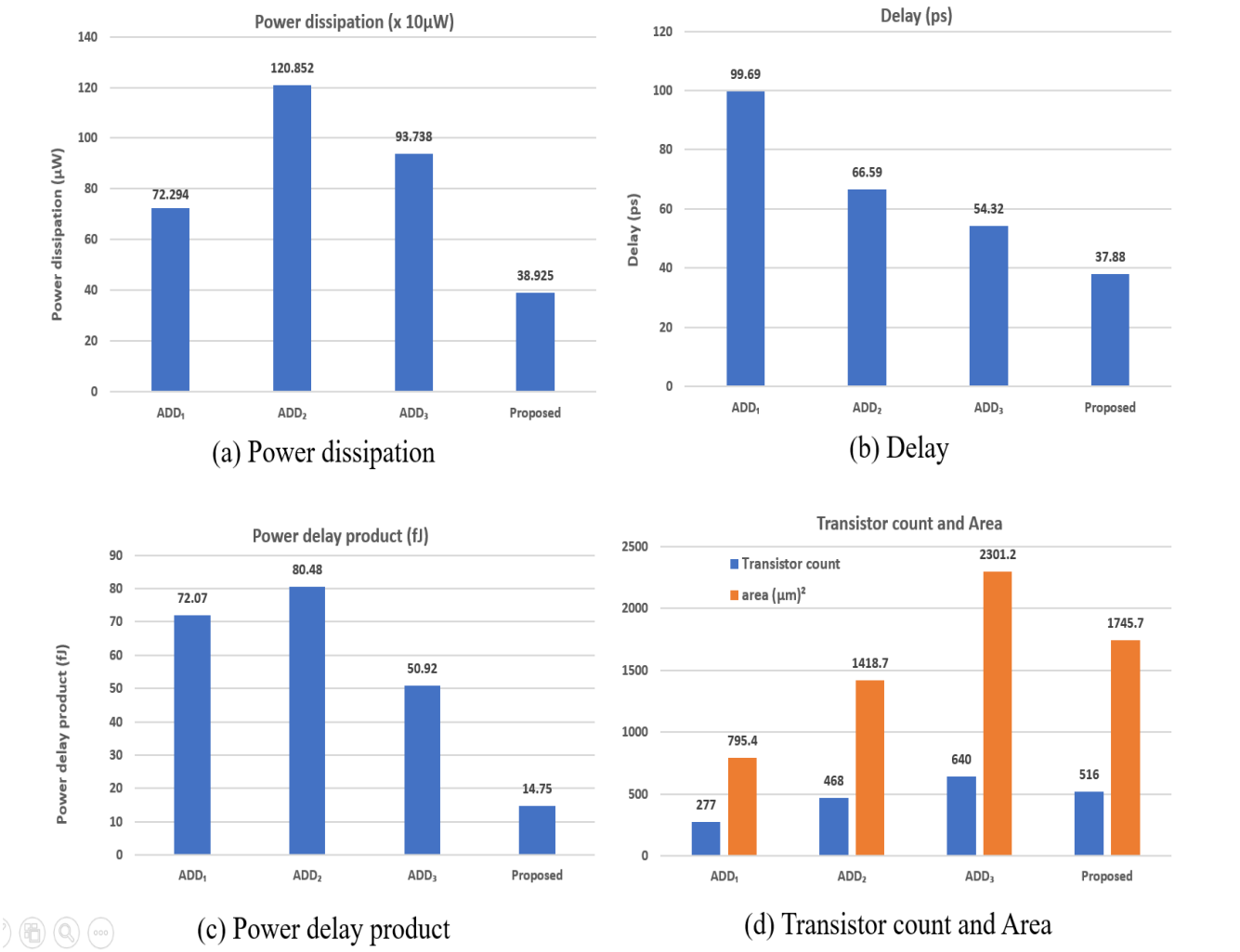

Fig 9. Performance Parameters (a) Power dissipation (b) Delay (c) Power delay product (d) Transistor count

4 Conclusion

This paper proposed a dynamic logic-based design of a high-speed BCD adder which makes it a better choice for performing decimal arithmetic operations. The novelty of this work lies in realizing an optimized static BCD architecture into dynamic logic using a two-level netlist structure with optimized speed by 30%, power minimization by 59%, and occupies 25% less area and 124 less transistors required compared to static CMOS BCD adder, maintaining its logic efficiency, while reducing complexity.

Compared to existing designs such as the Compact Adder[10], Novel Adder[11], and Static CMOS BCD Adder[12], our proposed BCD adder completely outperforms in terms of power dissipation, delay, power-delay product (PDP), and transistor efficiency. Lack of correction circuit and carry propagation optimization further improve its ability for high-speed arithmetic computations. The dynamic adder also exhibits thermal robustness, maintaining stability across a temperature range of -40°C to 120°C.

In addition to that, improvements can be made further by extending the architecture to multi-digit BCD adders by cascading the single digit BCD adders making it reliable for more advanced and complex operations. We can also implement various pipelining strategies to improve speed and allowing faster carry propagation. We can also map the design onto FPGA or ASIC implementations for developing various applications where the decimal arithmetic is required. In addition, the application of modern techniques like adiabatic logic, FinFET technology to this dynamic adder would further minimize power dissipation,

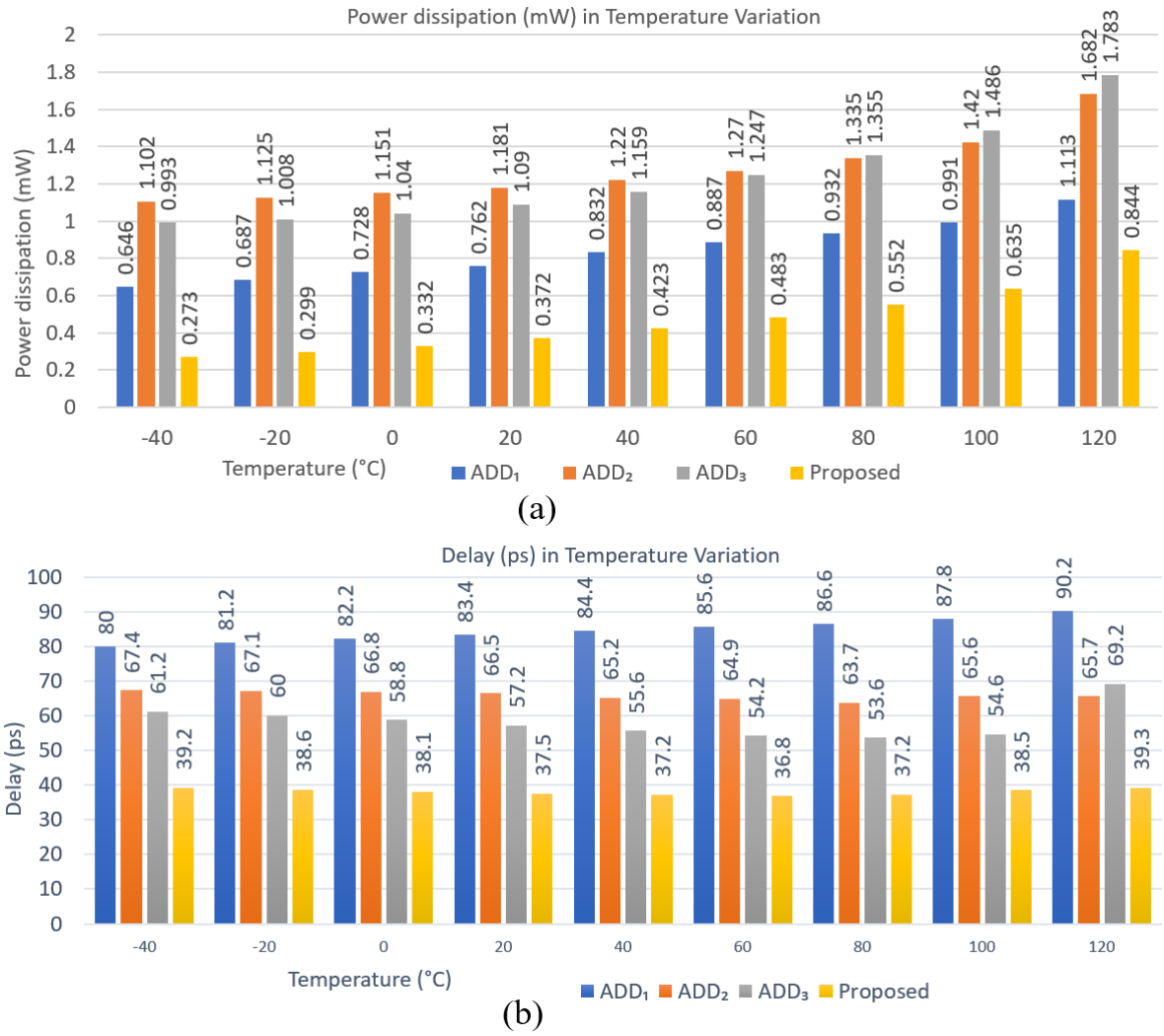

Fig 10. Temperature analysis (a)Power dissipation in Temperature Variation (b) Delay in Temperature Variation

maximize speed and improve performance showcasing the design more appropriate for low-power, high-speed arithmetic applications which are required in financial and scientific computing systems.

References

1) Chauhan P, Bandil M. Review: Different Designs of BCD Adder. *Journal of Emerging Technologies and Innovative Research (JETIR)*. 2019 06;6(6):981–986. Available from: https://www.jetir.org/papers/JETIR1907444.pdf.
2) Hasan M, Islam MS, Ahmed MR. Performance Improvement of 4-Bit Static CMOS Carry Look-Ahead Adder Using Modified Circuits for Carry Propagate and Generate Terms. *Science Journal of Circuits, Systems and Signal Processing*. 2019 12;8(2):76–81. https://doi.org/10.11648/j.cssp.20190802.16.
3) Hasan M, Hossain MS, Siddique AH, Hossain M, Zaman HU, Islam S. A High-Speed 4-Bit Carry Look-Ahead Architecture as a Building Block for Wide Word-Length Carry-Select Adder. *Microelectronics Journal*. 2021 03;109. https://doi.org/10.1016/j.mejo.2021.104992.
4) Ykuntam YD, Prasad SH. A Modified High Speed and Less Area BCD Adder Architecture Using Mirror Adder. In: Proceedings of the 2nd International Conference on Smart Electronics and Communication (ICOSEC), Trichy, India. 2021;p. 624–627. https://doi.org/10.1109/ICOSEC51865.2021.9591842.
5) Chu Z, Li Z, Xia Y, Wang L, Liu W. BCD Adder Designs Based on Three-Input XOR and Majority Gates. *IEEE Transactions on Circuits and Systems II: Express Briefs*. 2021 06;68(6):1942–1946. https://doi.org/10.1109/TCSII.2020.3047393.
6) Arulkarthick VJ, Rathinaswamy A, Srihari K. Design of BCD Adder with Five Input Majority Gate for QCA. *Microprocessors and Microsystems*. 2020;74. https://doi.org/10.1016/j.micpro.2020.103040.
7) Srinivas M, Sagar KVD. Analysis on Power Gating Circuits Based Low Power VLSI Circuits (BCD Adder). *Journal of Physics: Conference Series*. 2021 09;2089(1):012080. https://doi.org/10.1088/1742-6596/2089/1/012080.

8) Sri KU, Badhshah M, Ajayan J. Design and Analysis of Compact BCD Adder for Future High-Speed and Low-Power Electronic Applications. In: Proceedings of the 2024 International Conference on Emerging Frontiers in Electrical and Electronic Technologies (ICEFEET). 2024. https://doi.org/10.1109/ICEFEET64463.2024.10866669.
9) Share AA, Al-Khaleel O, Zghoul FN, Al-Khaleel M, Papachristou C. Design and Implementation of High-Speed Carry Look-Ahead Decimal Adder (CLDA) Using CMOS Technology. *IEEE Access*. 2025 02;13:18672–18681. https://doi.org/10.1109/ACCESS.2025.3540836.
10) Ruiz GA, Granda M. An Area-Efficient Static CMOS Carry-Select Adder Based on a Compact Carry Look-Ahead Unit. *Microelectronics Journal*. 2004 12;35(12):939–944. https://doi.org/10.1016/j.mejo.2004.09.002.
11) Miao J, Li S. A Novel Implementation of 4-Bit Carry Look-Ahead Adder. In: Proceedings of the 2017 International Conference on Electron Devices and Solid-State Circuits (EDSSC), Hsinchu, Taiwan. 2017;p. 1–2. https://doi.org/10.1109/EDSSC.2017.8126457.
12) Share AA, Zghoul FN, Al-Khaleel O, Al-Khaleel M, Papachristou C. Design of High Speed BCD Adder Using CMOS Technology. *IEEE Access*. 2023 12;11:141628–141639. https://doi.org/10.1109/ACCESS.2023.3342328.