AOI: 10.100.234513.0204
13th international conference on Microelectronics Circuits and Systems, Micro2026

Performance-Enhanced Carry Look-Ahead Decimal Adder for Next-Generation Computing Applications

J. Priyanka, M. Pavan Kumar, T. Ushasri, P. Pavan Kumar, T. Guna Sekhar, T. Vamsi

Electronics & Communication Engineering
Lendi Institute of Engineering and Technology, Vizianagaram, India
Corresponding Email: priyanka.j@lendi.edu.in

ABSTRACT

Decimal computation/calculation playing a major role and gaining attention by scientists and analysts because of their significance in various human-centric applications. Executing these operations in hardware offers better speed than software methods. This is highly beneficial for systems that need quick responses. Decimal addition is most important and basic decimal operation among all remaining operations and also one of the essential one. The Complementary Metal Oxide Semiconductor (CMOS) technology has the benefit of decreasing the propagation delay in virtual systems. Decimal adders can be developed as Ripple Carry Addition (RCA) structure or as a Carry Look-Ahead Addition (CLA) structure with more rapid and extra hardware cost. This paper discusses the design and performance analysis of a Carry Look-Ahead Decimal Adder (CLDA) implemented using CMOS technology. This work developed CLDA design to mitigate the latency limitations of conventional Ripple Carry Decimal Adder (RCDA) by bettering carry propagation efficiency. The design is executed and simulated using 45*nm* and 90*nm* technologies, and key performance metrics such as delay, power consumption, PDP, EDP are evaluated. The results shows that the proposed CLDA achieves notable performance improvement over the conventional RCDA. In 45*nm* technology, the CLDA provides approximately 19.9% reduction in delay and 19.8% improvement in Power Delay Product (PDP), indicating enhanced speed and energy efficiency. In 90*nm* technology, a delay reduction of about 13.3% is observed, while PDP remains nearly comparable. The novelty of this work lies in the effective CMOS-based execution of CLDA and its comparative analysis across different technology nodes. These results confirm that the proposed CLDA is more suitable for high-speed arithmetic applications.

Keywords: CMOS technology, CLDA, RCDA, BCD adder, Decimal adder, PDP, EDP, Cadence virtuoso.

I. INTRODUCTION

Binary arithmetic plays an important role in various human-centric applications, financial computations, digital signal processing, accounting, tax calculations, conversion, banking and internet-based applications, where exact decimal representation is required. In such applications, Binary-Coded Decimal (BCD) adders are widely used to avoid errors related with binary-to-decimal conversion and rounding mistakes.

The Ripple Carry Decimal Adder (RCDA) is one of the simplest executions for BCD addition. However, its major restriction is the sequential propagation of carry, which leads to increased delay as the number of digits increases. Due this RCDA less suitable for high-speed arithmetic systems [1].

To overcome this limitations, Carry Look-Ahead Decimal Adders (CLDA) have been proposed, in this carry signals are generated in parallel using propagate and generate logic. Due to this carry propagation delay reduces and improves overall performance of the adder. However, this improvement leads to increase in complexity of circuit, transistor count and power consumption [2], [3].

Several existing works have investigated high-speed decimal adder architectures using different logic approaches and optimization techniques. On the other hand, various designs either mainly concentrate on delay reduction without analyzing power and energy efficiency, or lack comparative analysis across different technology nodes.

In this work, a CMOS-based CLDA is designed and analyzed using 45*nm* and 90*nm* technologies. The proposed design is compared with RCDA in terms of delay, power, Power Delay Product (PDP), and Energy Delay Product (EDP). The main contribution of this work lies in providing a comprehensive performance comparison across technology nodes while highlighting the trade-off between speed, power and area [3].

The results shows that the proposed CLDA attains significant delay reduction enhanced energy efficiency, making it suitable for high-speed arithmetic applications.

The rest of this document is structured as follows: the drawing figures and background about that figures are covered in Section II. The experimental results, the comparisons and discussions of carry look-ahead decimal adder and ripple carry decimal adder are

ISBN: 978-81-985770-0-9

presented in Section III. Finally, conclusions are provided in Section IV.

II. DRAWING FIGURES

Binary Coded Decimal (BCD) is a 4bit binary representation for decimal digits in number system. Every decimal digit is represented by 4-bit using it binary equivalents. For example, $(5)_{10} = (101)_2$ is represented in BCD as $(0101)_{\text{BCD}}$, another example $(3)_{10} = (11)_2$ is $(0011)_{\text{BCD}}$. The traditional BCD addition of two decimal digits needs correction whenever the results was exceeded the digit $(9)_{10} = (1001)_{\text{BCD}}$. This was done by adding $(6)_{10} = (0110)_{\text{BCD}}$ to the results. [1], [3], [4].

BCD adder uses the two levels of binary addition at the top; two BCD digit are added with carry in using a 4-bit binary adder. At bottom $(6)_{10} = (0110)_{\text{BCD}}$ is added with second 4-bit binary adder which acts as correction logic.

A traditional 1 -digit BCD adder that adds the two BCD digits $A = (A_3A_2A_1A_0)_{\text{BCD}}$ and $B = (B_3B_2B_1B_0)_{\text{BCD}}$ with carry in C_{in} and produces the BCD results $S = (S_3S_2S_1S_0)_{\text{BCD}}$ along with the carry out C_{out} is illustrated in Fig. 1.

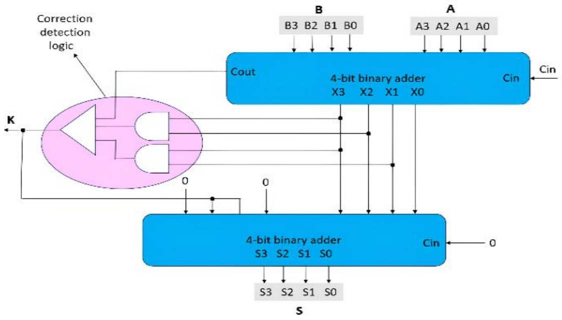

Fig. 1. 1-digit BCD adder [1]

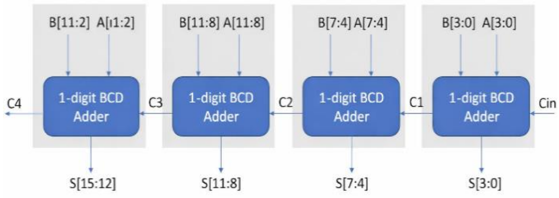

Fig. 2. 4-digit Ripple Carry Decimal Adder (RCDA)

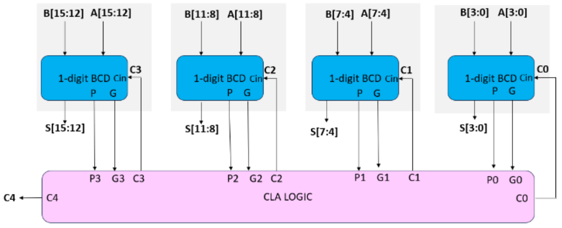

Fig. 3. A block diagram of the proposed 4-digit CLDA[2],[3]

An N -digit BCD adder can be designed by cascading N 1 -digit BCD adders in a ripple carry style. This work refers to ripple carry decimal adder (RCDA) in Fig. 2. In the binary adders, the ripple carry addition is known as the one of the slowest addition schemes.

The propagation delay of carry out of the 1 -digit BCD adder is higher than that of a 4-bit binary adder because it passes the correction detection logic as Fig. 1. So, RCDA suffer the speed problem.

The ripple carry decimal adder (RCDA) follows the sequential propagation and carry of one digit get input to other and goes in sequential due to that the delay was increases and slow calculation happen. In the binary adders, carry look-ahead addition (CLA) is one method that is used to avoid the carry propagation problem faced in ripple carry addition (RCA). CLA calculate carry signals in advance using inputs. So, it doesn't depend on the sequential carry propagation and the delay decreases than the ripple carry chain process and gives enhanced performance [2], [3].

Equations:

For Fig. 1. the equations of Binary Addition (First stage)

Add two BCD digits and carry input:

$$X_3X_2X_1X_0 = A+B+C_{in} \qquad (1)$$

Correction Detection logic

Correction is needed when result>9:

$$K = X_3X_2 + X_3X_1 + C_{out} \qquad (2)$$

Correction Addition if K=1, add 6 (0110):

$$S = X + (K \,.\, 0110)$$

Corrected Output:

$$S_3S_2S_1S_0 = X + (K \,.\, 0110) \qquad (3)$$

Final Carry:

$$C_{out}\,(\text{final}) = K \qquad (4)$$

The Fig. 1, illustrates the 1-digit BCD adder, where the intermediate Sum (X) is generated using a 4-bit binary adder. The correction logic computes signal (K) based on the condition given in Eq. (2). When (K =1), a correction value of 6 is added to obtain the final BCD output (S), as described in Eq. (3) and Eq. (4) is final carry equation.

For Fig. 2. the equations are

Initial Carry:

$$C_0 = C_{in} \qquad (5)$$

Ripple Carry Operation:

$$S_i,\, C_{i+1} = A+B+C_{in}\,(i = 0,1,2,3) \qquad (6)$$

i.e., Digit-wise BCD Addition (Ripple form):

ISBN: 978-81-985770-0-9

$S\,[3{:}0],\ C_1 = A\,[3{:}0] + B\,[3{:}0] + C_0$

$S\,[7{:}4],\ C_2 = A\,[7{:}4] + B\,[7{:}04] + C_1$

$S\,[11{:}8],\ C_3 = A\,[11{:}8] + B\,[11{:}8] + C_2$

$S\,[15{:}12],\ C_4 = A\,[15{:}12] + B\,[15{:}12] + C_3$

The Fig. 2 shows the RCDA architecture, where carry propagation sequentially from one stage to the next as defined by Eq. (5)-(6), resulting in increased delay.

For Fig. 3. the equations are:
CLDA Section
Bitwise Generate (G):

$$G_i = A_i\,.\,B_i\,(i = 0,1,2,3) \qquad (7)$$

Bitwise Propagate (P):

$$P_i = A_i \oplus B_i\,(i = 0,1,2,3) \qquad (8)$$

Sum Equations:

$$S_i = P_i \oplus C_i\,(i = 0,1,2,3) \qquad (9)$$

Carry Look-Ahead Equations:

$$C_1 = G_0 + P_0C_{in} \qquad (10)$$

$$C_2 = G_1 + P_1G_0 + P_1P_0C_{in} \qquad (11)$$

$$C_3 = G_2 + P_2G_1 + P_2P_1G_0 + P_2P_1P_0C_{in} \qquad (12)$$

$$C_4 = G_3 + P_3G_2 + P_3P_2G_1 + P_3P_2P_1G_0 + P_3P_2P_1P_0C_{in} \qquad (13)$$

Group Propagate and Generate:

$$P = P_3P_2P_1P_0 \qquad (14)$$

$$G = G_3 + P_3G_3 + P_3P_2G_1 + P_3P_2P_1G_0 \qquad (15)$$

The Fig. 3, presents the proposed CLDA, Eq. (7)-(9) define the generate and propagate signals used to compute carry efficiently in the CLDA architecture. Using Eq. (10)-(13), all carry signals are computed in parallel, eliminating sequential delay and improving speed. Eq. (14)-(15) shows the group propagate and generate signals further optimize carry computation for multi-bit addition.

III. RESULTS and DISCUSSIONS

The performance of the designed Carry Look-Ahead Decimal Adder (CLDA) is compared with the Ripple Carry Decimal Adder (RCDA) in terms of delay, power, PDP, EDP for both 45*nm* and 90*nm* technology nodes simulations are functionally tested and verified using Cadence virtuoso simulation tool. A block diagram of Fig. 2, Fig. 3 are designed in cadence virtuoso.

The simulation results indicate that the CLDA noticeably improves speed compared to RCDA. In 90*nm* technology, the CLDA achieves a delay reduction of approx. 13.3%, showing faster carry computation. In 45*nm* technology, the delay reduction is about 19.9%, demonstrating even greater performance improvement due to efficient parallel carry generation. In terms of energy efficiency, the PDP of CLDA in 45*nm* technology is reduced by approx. 19.8% compared to RCDA, indicating better overall performance. However, in 90*nm* technology, PDP remains nearly comparable between CLDA and RCDA.

It is also observable that the power consumption of CLDA is higher than RCDA, particularly in 45*nm* technology was due to higher transistor count and increased switching activity associated with parallel carry computation. Additionally, leakage power becomes more dominant in deep submicrometric technology. Despite this the proposed CLDA offers better speed and energy efficiency. This highlights the CLDA is suitable for high-speed applications, while RCDA preferred in lower-power designs [5], [6].

TABLE I: Critical paths for 1,2,3,4 -digit decimal adder for Carry Look-Ahead Adder

Digit for 90nm	Max Path	Max Delay(ps)	Min Path	Min Delay(ps)
1	Cin-S3	554.8	A0-Cout	181.6
2	Cin-S7	554.8	A0-Cout	181.6
3	Cin-S11	554.8	A0-Cout	181.6
4	Cin-S15	554.8	A0-Cout	181.6
Digit for 45nm	Max Path	Max Delay(ps)	Min Path	Min Delay(ps)
1	Cin-S3	524.5	A0-Cout	172.0
2	Cin-S7	524.5	A0-Cout	172.0
3	Cin-S11	524.5	A0-Cout	172.0
4	Cin-S15	524.5	A0-Cout	172.0

TABLE 2: Critical paths for 1,2,3,4 -digit decimal adder for Ripple Carry Adder

Digit for 90nm	Max Path	Max Delay(ps)	Min Path	Min Delay(ps)
1	Cin-S3	625.0	A0-Cout	230.0
2	Cin-S7	630.0	A0-Cout	235.0
3	Cin-S11	635.0	A0-Cout	240.0
4	Cin-S15	640.0	A0-Cout	245.0
Digit for 45nm	Max Path	Max Delay (ps)	Min Path	Min Delay (ps)
1	Cin-S3	595.0	A0-Cout	210.0
2	Cin-S7	615.0	A0-Cout	220.8
3	Cin-S11	635.0	A0-Cout	230.3
4	Cin-S15	655.0	A0-Cout	240.5

TABLE 3: Transistor Count

ISBN: 978-81-985770-0-9

Ripple Carry Adder	Carry Look-Ahead Adder
392 Transistors	470 Transistors

TABLE 4: Comparison against 4 -digit decimal adders

Tech. Node	Adder	Power (µW)	Delay (ps)	Tran count	PDP (fJ)	EDP (J.s)
90nm	RCA	10.54	640.0	1568	6.75	4.32 E-24
90nm	CLA	12.21	554.8	1948	6.77	3.76 E-24
45nm	RCA	192.3	655.0	1568	125.96	8.25 E-23
45nm	CLA	192.7	524.5	1948	101.02	5.30 E-23

Table-4 presents the performance comparison of RCDA and CLDA in terms of power, delay, PDP and EDP. It is observed that the proposed CLDA attains significant delay reduction in both 45*nm* and 90*nm* technologies. Although power consumption is higher in 45*nm* due to increased leakage [9], the PDP and EDP are improved, showing better efficiency of proposed design [7].

Results can be expressed as Fig-4 & Fig-5:

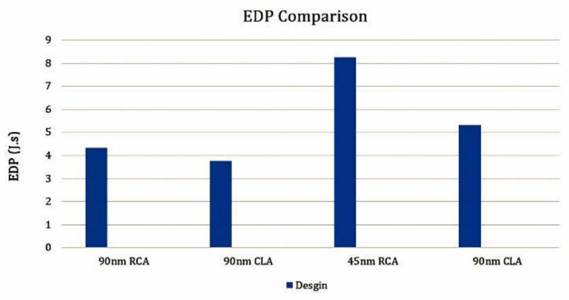

Fig. 4. Power Delay Product (PDP) comparison

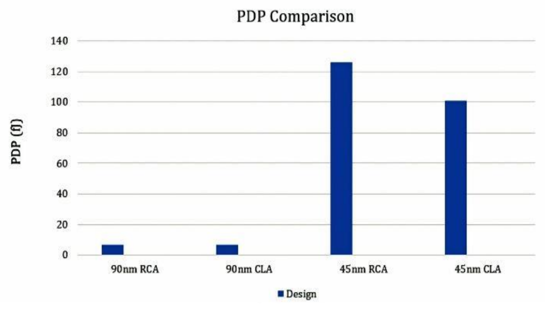

Fig. 5. Energy Delay Product (EDP) Comparison

Fig. 4, shows the comparison of PDP for RCDA and CLDA is 45*nm* and 90*nm* technologies. The CLDA shows lower PDP in 45*nm*, indicating better energy efficiency.

Fig. 5, presents the EDP comparison. The proposed CLDA achieves improved EDP compared to RCDA, showing better performance in terms of both speed and energy.

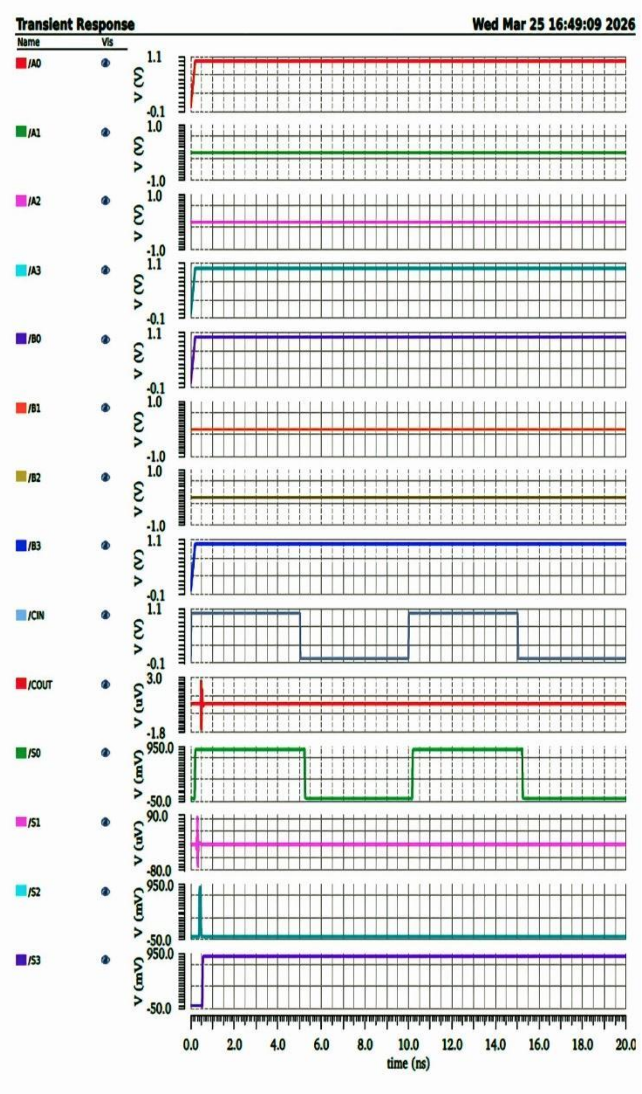

Fig. 6. 1 – digit Ripple Carry Decimal Adder waveforms

The designs are simulated, For N =1, N = 2, N =3 and N = 4. Using the 45*nm* and 90*nm* technology libraries respectively with a supply voltage of VDD = 0.9V, typical process conditions and room temperature (27°C). Transient analyses are performed to evaluate the power consumption, propagation delay, and power-delay product (PDP) across the RCDA and CLDA of different digits [3], [8]. A clock frequency of 50MHz is applied

ISBN: 978-81-985770-0-9

for transient analysis to evaluate the speed and performance measures under realistic operating conditions.

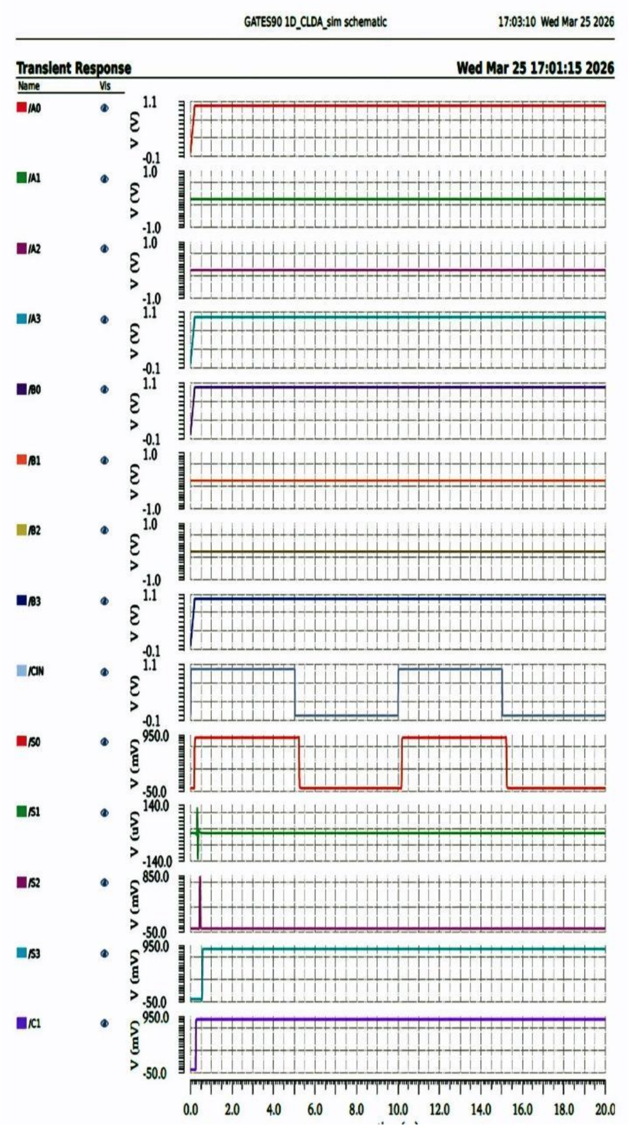

Fig. 7. 1 -digit Carry Look-ahead Decimal Adder waveforms

These produces the waveforms at the output Fig. 6, Fig.7 that shows the delay at the critical and min path. They shows the transient response of the proposed CLDA design. The input signals $A[3:0]$, $B[3:0]$ and C_{in} are applied and corresponding outputs $S[3:0]$ and C_{out} are observed. The waveform shows correct BCD addition and proper carry generation. The output transition occurs with minimal delay, confirming the high-speed operation of proposed CLDA.

From the waveform, it is observed that changes in input signals are immediately showed in the output with reducing propagation delay. Unlike RCDA which generates propagation carry sequentially, the CLDA generates signals in parallel, resulting in faster response.

Based on the traditional BCD 1 -digit adder, which is shown in Fig. 1, the CLDA and RCDA are compared in different performance measurement. By observing the values at table-1,2 critical path and minimum path delays of different 1-4 digits in 45*nm* ,90*nm* technology libraries respectively the CLDA was faster than RCDA in both of them. In CLDA critical and min path delay with 554.8ps and 181.6ps for 90*nm*, 524.5ps and 172.0ps for 45*nm* are constant but in RCDA critical and min path varies and increases with increasing no. of digits, critical path was C_{in} *to* S_{15} and min path was A_0 *to* C_{out} with 640.0ps and 245.0ps, 645.0ps and 240.0ps respectively with 90*nm* and 45*nm* technologies. In Table 3 the 1 -digit transistor counts of RCA and CLA are counted. Therefore, finally they show that CLDA was faster in performance than RCDA and promises the enhanced performance compare to ripple carry decimal adder [1], [5]. So, the simulations and graphs show promising results in terms of speed, etc., in CLDA compare to RCDA.

IV. CONCLUSION

Decimal arithmetic is getting enhanced investigation because of its function in human-centric applications. For real-time use, hardware applications are mostly preferred over software because enhanced performance. Decimal addition plays key role and fundamental, as other operations can be derived from it. This paper presented the design and analysis of a Carry Look-Ahead Decimal Adder (CLDA) executed using CMOS technology and compared with the Ripple Carry Decimal Adder (RCDA). The proposed design concentrates on enhancing speed by reducing carry propagation delay through parallel carry calculation. The simulation results shows that the CLDA attains significant delay reduction compared to RCDA. In 90*nm* technology, a delay improvement of approx. 13.3% is observed. While in 45*nm* technology, the delay reduction 19.9%. Moreover, the CLDA shows enhanced energy efficiency in 45*nm* technology with a PDP reduction of approx. 19.8%. While CLDA offers superior speed, its higher transistor count results to increased power consumption and area, mainly in 45*nm* technology where leakage power comes becomes significant. Finally, while CLDA is suited for high-speed applications, RCDA can be still preferable for low-power designs. Although the concept of Carry Look-Ahead Adder has been studied widely, this study provides a comparative analysis of decimal adder architectures across multiple technology nodes (45*nm* and 90*nm*) using CMOS execution. By evaluating delay, power, PDP, and EDP together it outpaces simple delay enchantment to offer a more understanding performance trade-off. This makes the work relevant for practical VLSI design considerations. The current proposed work is restricted to schematic-level simulation and up to 4-digit BCD addition, does not included detailed layout or process variation analysis. Further efforts should aim to expand the design to higher-digit architectures and perform layout-level executions and evaluating the design under process, voltage, and temperature (PVT) variations. Finally, executing the design on FPGA or ASIC platforms will be necessary to validating its performance in real-world applications.

ACKNOWLEDGEMENT

I would like to be thankful to my guide for their valuable guidance, continuous encouragement and insightful suggestions and my sincere gratitude to those who guided me, and supported me throughout this work. I also acknowledge the use of Cadence Virtuoso simulation tools which made this work possible.

REFERENCES

[1] A. A. Share, F. N. Zghoul, O. Al-Khaleel, M. Al-Khaleel, and C. Papachristou, ‘‘Design of high-speed BCD adder using CMOS technology,’’ *IEEE Access*, vol. 11, pp. 141628–141639, 2023.

[2] R. Zlatanovici, S. Kao, and B. Nikolic, ‘‘Energy–delay optimization of 64-bit carry-lookahead adders with a 240 ps 90 nm CMOS design example,’’ *IEEE J. Solid-State Circuits*, vol. 44, no. 2, pp. 569–583, Feb. 2009.

[3] A. Al Share, O. Al-Khaleel, F. N. Zghoul, M. Al-Khaleel and C. Papachristou, "Design and Implementation of High-Speed Carry Look-Ahead Decimal Adder (CLDA) Using CMOS Technology," in IEEE Access, vol. 13, pp. 29361-29374, 2025, doi: 10.1109/ACCESS.2025.3540836.

[4] Z. Chu, Z. Li, Y. Xia, L. Wang, and W. Liu, ‘‘BCD adder designs based on three-input XOR and majority gates,’’ *IEEE Trans. Circuits Syst. II, Exp.Briefs*, vol. 68, no. 6, pp. 1942–1946, Jun. 2021.

[5] S. Gorgin, H. Z. Nejad, and Jeong-A. Lee, ‘‘A practical energy/power reduction approach for parallel decimal multiplier,’’ *IEEE Access*, vol. 10, pp. 11372–11381, 2022.

[6] 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,doi:10.17485/IJST/v18i21.700

[7] Santhoshini.P, Varsha.M, Varshini.S and Prathiyaa.P, "Exploring CMOS Technology for High-Speed and Efficient BCD Adder Designs," International Journal of Innovative Research in Technology (IJIRT), vol. 11, no. 5, pp. 1282–1286, 2024.

[8] M. Hasan, M. S. Hossain, A. H. Siddique, M. Hossain, H. U. Zaman, and S. Islam, ‘‘A high-speed 4-bit carry look-ahead architecture as a building block for wide word-length carry-select adder,’’ *Microelectron. J.*, vol. 109, Mar. 2021, Art. no. 104992.

[9] L.Wang, X.Chen, and Y.Zhang, “High-Performance Arithmetic Unit Design for Next-Generation VLSI Systems,” *IEEE Access*, vol. 13, 2025.

[10] J.Singh and M.Saxena, “Impact of Leakage Power on Nanoscale CMOS Circuits,” *IEEE Transactions on Nanotechnology*, vol. 23, 2024.

Cite this article as:

J. Priyanka, M. Pavan and et. al. "Performance-Enhanced Carry Look-Ahead Decimal Adder for Next-Generation Computing Applications", Proceedings of 13th international conference on Microelectronics, Circuits and Systems, Micro2026.
Displayed as online on 5th June 2026.

Link: http://actsoft.org/science/micro2026-pro/240-micro2026.pdf

ISBN: 978-81-985770-0-9