

Design of a 64-bit SQRT-CSLA with Reduced Area and High-Speed Applications in Low Power VLSI Circuits

CH. Pallavi¹, C. Padma², R. Kiran Kumar³, T. Suguna⁴, C. Nalini⁵

^{1,2}Associate Professor, Dept. of ECE, Sri Venkateswara College of Engineering (SVCE), Tirupati, A.P, India.

³Assistant Professor, Department of ECE, Madanapalle Institute of Technology & Sciences, Madanapalle, A.P, India.

⁴Assistant Professor, Dept. of ECE, Panimalar Engineering College, Chennai, India.

⁵Assistant Professor, Dept. of ECE, Mohan Babu University (MBU), Tirupati, A.P, India.

KEYWORDS:

CSLA (Carry Select Adder),
SQRT CSLA (Square Root Carry Select
Adder),
VLSI,
Xilinx.

ARTICLE HISTORY:

Received : 08.07.2024

Revised : 22.11.2024

Accepted : 28.02.2025

DOI:

<https://doi.org/10.31838/jvcs/07.01.06>

ABSTRACT

The main areas of research in VLSI system design include area, high speed, and power-efficient data route logic systems. The amount of time needed to send a carry through the adder limits the pace at which addition can occur in digital adders. One of the quickest adders, the Carry Select Adder (CSLA), is utilized by various data processing processors to carry out quick arithmetic operations. It is evident from the CSLA's structure that there is room to cut back on both the area and the delay. This work employs a straightforward and effective gate-level adjustment (in a regular structure) that significantly lowers the CSLA's area and delay. In light of this adjustment Square-Root Carry Select Adder (SQRT CSLA) designs with bit lengths of 8, 16, 32, and 64. When compared to the standard SQRT CSLA, the suggested design significantly reduces both area and latency. Xilinx ISE tool is used for Simulation and synthesis. The performance of the recommended designs in terms of delay is estimated in this study using the standard designs. The study of the findings indicates that the suggested CSLA structure outperforms the standard SQRT CSLA.

Author e-mail and ORCID ID: pallavi.ch@svcolleges.edu.in, ORCID ID: 0000-0002-3283-8460

How to cite this article: Pallavi CH, Padma C, Kumar KR, Suguna T, Nalini C. Design of a 64-bit SQRT-CSLA with Reduced Area and High-Speed Applications in Low Power VLSI Circuits, Journal of VLSI Circuits and System, Vol. 7, No. 1, 2025 (pp. 40-45).

INTRODUCTION

Power-efficient and fast data route logic system design is among the most important fields of research in VLSI. With regard to digital adders, the propagation time of a carry through the adder limits the adder's speed. Any digital system, whether it be for digital signal processing or control, needs to be able to add. A digital system's ability to operate quickly and precisely is largely dependent on how well its resident adders execute. Because adders are widely used in other fundamental digital operations like subtraction, multiplication, and division, they are also a crucial part of digital systems. Therefore, enhancing the digital adder's performance would significantly improve the way binary operations are carried out inside a circuit made up of these blocks.

Analysing a digital circuit block's power dissipation, layout area, and operation speed allows one to determine how

well it performs. The two primary areas of research on VLSI system design are reduced area and fast data route logic systems. High-performance processors and systems have always required addition and multiplication to operate at high speeds. The amount of duration required for a carry through the adder to propagate limits the pace at which addition can occur in digital adders. In a basic adder, the sum for each bit position is produced in a sequential manner only after the preceding bit position and a carry has propagated into the subsequent position. There are several computational systems that employ the CSLA.

In an elementary adder, each bit sum is formed by serially after adding the preceding bit, the carry created by this addition is transferred to the following bit.. The Carry Select Adder (CSLA) generates numerous carry bits and selects a carry for the required output in a number of computer systems to alleviate the problem of carry

propagation delay. However, because the CSLA uses multiple pairs of Ripple Carry Adders (RCA) to produce a partial sum and carry by taking carry data into account, and multiplexers select the final total and carry (mux), the CSLA is not area-efficient. High-performance processors and systems have always required addition and multiplication to operate at high speeds.

Design to separately produce many carries, one carry is selected to generate the aggregate in order to mitigate the issue of carry propagation latency. To get the final amount, it employs $C_{in}=0$ and $C_{in}=1$. However, because it employs numerous pairs of RCA to generate limited sum or carry by taking carry input into consideration, the regular CSLA is not area or speed efficient. The multiplexers determine the final sum and carry (mux). The area will grow as a result of using two separate RCAs, which will increase the delay.

The fundamental idea of the suggested Utilizing n-bit BEC is the task to increase addition speed in order to solve the aforementioned issue. To further increase speed and decrease the delay, the logic can be replaced by means of C_{in} is 1 in RCA. To speed up the addition process by reducing area and delay, the standard CSLA can be replaced with BEC. The primary benefit of the BEC logic is that it uses fewer logic gates than the Full Adder (FA) structure since fewer gates are needed.

Used CSLA is to deduce the complexity of delay in carry creation and then selecting the sum [1]. The CSLA, on the other hand, greater area, and it employs multiple blocks of RCA to calculate the carry and sum by first treating the carry inputs as 0 and 1, after which the multiplexer selects the amount of sum or carry [2-6]. The fundamental concept of this paper is that AND/OR gates generate carry and is carried to the subsequent step, whereas only sum is chosen by the multiplexer. Figure 1 shows a block diagram of regular CSLA.

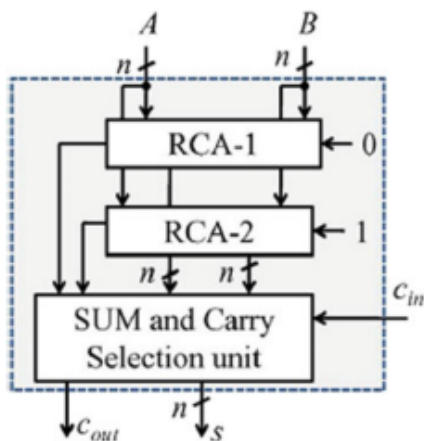


Fig. 1: Architecture of a R-CSLA

The carry input is first taken as 0 to generate the sum and carry, and then it is again taken as 1 to generate the sum and carry. The multiplexer will choose the sum and carry appropriately since it receives the actual input carry-in the pick line at the end. As a result, ripple carry adder blocks are utilized for computation in each step. In the diagram, A and B stand for the input values, while S and Cout stand for the output sum and carry.

The structure of this document is as follows. The previous carry select adder works are presented in Section 2. In Section 3, the suggested CSLA is described and the area and power reduction are assessed. Section 4 analyses the proposed CSLA's execution details and results, and wraps up the work in Section 5.

RELATED WORKS

The amount of time it takes to propagate carry signal through digital adders limits the adder's speed. To address the issue of carry propagation delay, the R-CSLA was developed. It selects the appropriate carry output and sum based on the preceding carry's value, after independently producing several carries. The following are the current carry select adder works:

M. Vinod Kumar Naik et al,^[7]proposed a CSA designed for rapid speed and low-powered VLSI applications. This work suggests a novel logic formulation for CSLA and offers a way to remove the all-logic operations seen in standard CSLA. This scheme differs from the traditional CSLA. Xilinx ISE was used to synthesize the suggested CSLA, and Xilinx Power Analyzer was used to analyse the power.

Nilkantha Rooj et al^[8], provided an analysis of the internal design of the traditional CSLA and other carry select adder designs in addition to developing a new design. In this work, carry selection operates ahead of the sum computation for each stage, but it does so in a different way than a traditional CSLA. The carry selectthe unit's logic has been minimized by using a certain bit pattern found in the carry out results produced by the distinct carry values (C_{in} is 0 & C_{in} is 1) in the suggested CSLA. A novel and efficient design for CSLA is suggested, based on this enhanced logic formulation.

Gagandeep Singh and Chakshu Goe,^[9] explained EX-OR gates are the fundamental building blocks and 8-bit adder is designed using a 3-T EX-OR gate. In comparison to standard CSLA and modified CSLA, the suggested CSLA has fewer transistors, consumes less power, and has a lower power-delay product (PDP). This research uses a straightforward method to improve the EX-OR gate's efficiency. The 3-T EX-OR gate used in the proposed

design results in a significant reduction in the no. of MOSFETs. This suggested 8-bit CSLA has a power decrease of 27.7% and 21.7%, respectively.

R. Sakthivel and G. Ragnath,^[10] Three blocks, HSCG, FCG, and FSG, were proposed for a low power, high speed, efficient CSLA. In comparison to the current CSLA, the HSCG is faster and requires less space and power thanks to its low-complexity Boolean expression design. In comparison to the current design, a 64-bit suggested CSLA delivers an average 43% power savings and an average 25% reduction in area consumption.

S. Allwin Devaraj et al,^[11] Proposed a Pass Transistor Logic (PTL) technology is being used in the design of CSLA to further reduce power and space. PTL reduces the no. of transistors needed for each logical circuit, which lowers the carry select adder's size and power consumption. Tanner EDA tool is used to run the simulation. Moreover, superfluous transistors and propagation delay can be decreased. In order to obtain less area and power, the PTL approach is employed to minimize the complexity at the transistor level.

Dr. D. B. Kadam et al,^[12] provided a straightforward method for using the BEC-1 architecture to minimise the SQRT-CSLA delay and area. The decrease of gates and LUT's is made possible by the BEC in the structure. This modified approach is a good substitute for adder implementation in many data processors as it reduces both area and delay. The goal of this work is to minimise the Area, delay and power of the CSLA architecture.

S. Balaprasad et al,^[13] developed a new logic formulation for CSLA and proposed a method that removes all of the unnecessary logic operations included in the traditional CSLA. Unlike the traditional method, the suggested technique schedules the final-sum calculation before the CS operation.

B. Ramkumar et al,^[14] presented with only a minor increase in delay, the suggested design outperforms the standard SQRT-CSLA. The study of the findings indicates that the suggested CSLA structure outperforms the standard SQRT CSLA.

Nagulapati Giri et al,^[15] Through comparison of metrics like as area, delay, and power consumption, the effectiveness of all design techniques has been examined. High-speed multiplication, arithmetic logic units, sophisticated microprocessor architecture, and other applications can benefit from the high efficacy design. The gpdk180 library was used, and Cadence Virtuoso Analog Design Environment was used to simulate each architecture.

Bagya Sree Auvla et al,^[16] High-performance, low-power, and area-efficient Multi-standard wireless receivers, biomedical instruments, and portable and mobile devices are among the growing number of applications for VLSI systems. By employing AOI logic to simplify the BEC and RCA units, the gate counts are decreased. The suggested CSLA is put into practice for various word sizes. Low power applications like digital signal processing, multipliers, filter ERROR accuracy, and design may benefit from the suggested architecture.

Kadaru Prasanna et al,^[17] The CSLA is frequently used to address this. A major breakthrough in system design of digital adders in VLSI is presented by the suggested improvements of including a BEC and a Kogge-Stone adder into the CSLA architecture.

Nelanti Harish et al,^[18] In contrast to the traditional method, Prior to the final-sum calculation, a new logic formulation for the CSLA operation is scheduled. Carry words that correspond to input-carry '0' and '1' produced by the CSLA using the suggested technique adhere to a particular bit pattern that is utilized for the CS unit's logic optimization. An ideal design for the CS and CG units is produced as a result. These optimized logic units are used to create an effective CSLA design.

S. Muminthaj et al,^[19] The suggested design uses less power than the standard adder circuits. This proposed study offers a straightforward method to lower the CSLA architecture's area, power consumption, and latency. The traditional Carry Select Adder's drawbacks include higher power consumption, increased chip area use, and a significant latency. The outcomes are compared in terms of area, delay, and power consumption. When compared to the previous model, the D-Flip-flop-based Improved CSLA turns out to be the High Speed and Low Power CSLA.

METHODOLOGY AND IMPLEMENTATION

The difference between this architecture and a standard 64-bit SQRT CSLA is that a BEC is used to substitute the RCA with $C_{in}=1$ out of the two available RCAs in a group. One of the features of this BEC is that it can carry out operations that are compared with BEC logic. Fig. 2 illustrates the 64-bit SQRT CSLA's modified diagram. A bit more bits are needed for BEC logic than for RCA logic. Along with being divided into several groups of bits of varying sizes, the updated block diagram also has appropriate muxes, BECs, and ripple carry adders for each group. Group 0 comprises a single RCA that receives a lower significant bit as input, carries it. Finally, the selected input arrives after the RCA and BEC for the remaining groups.

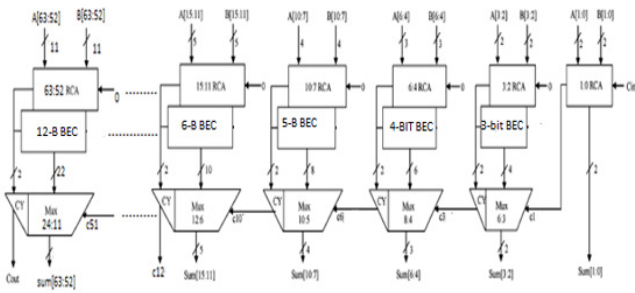


Fig. 2: Modified 64-bit SQRT CSLA block diagram

Consequently, the output from mux, sum1, and the results calculated by BEC and RCA, depend on mux. sum 2 is dependent upon mux and c1. The onset time of the selected muxinput for the remaining components is consistently higher than the arrival time of the BEC data inputs.

The fundamental operation of 6-bit addition, which consists of 12-bit mux, 6-bit data, and BEC logic (6-bit), is depicted in Fig. 3.2. The addition process is carried out for Cin = 0 and Cin = 1. Ripple carry adders are used for addition when Cin = 0, and 6-bit BECs are used for operations when Cin = 1 (replacing the RCA). Based on the previous group's carry in signal, the resultant is chosen. The previous group's Cin signal and mux delay determine the overall delay. Figure 3, depict the construction of a 6-bit BEC with 12:6 MUX.

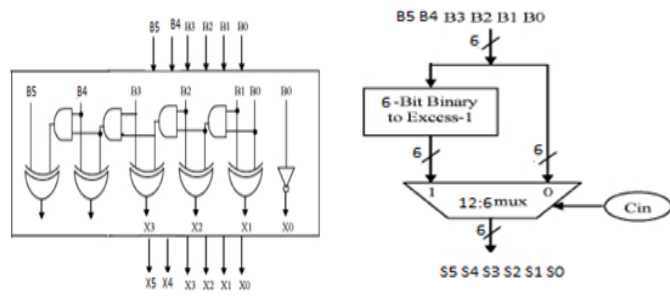


Fig. 3: Structure of a 6-bit BEC

Estimation of A 64-Bit modified SQRT CSLA

Figure 3 illustrates the architecture of the suggested 64-bit SQRT CSLA that uses BEC for Ripple Carry Adders (RCA) with CIN = 1 to maximize power and area. There are five groups within the framework. Fig. 3.3 displays the area estimation and delay for group 5.

These are the steps that lead to the evaluation:

1. For CIN = 0, the group 2 [see Fig. 3.3(a)] has one 2-b RCA with 1 FA and 1 HA. A 3-b BEC is utilized in place of an additional 2-b RCA with CIN = 1, adding one to the output from the 2-b

2. RCA. The arrival time of selection input c1 [time (t) = 7] of the 6:3 mux is later than the s2 [t = 4] and earlier than the s3 [t = 9] and c3 [t = 10], according to Table 1's delay values. As a result, mux and s3 and partial c3 (input to mux) and mux, respectively, are required for the sum 3 and final c3 (output from mux). The sum2 is dependent upon mux and c1.
3. The mux selection input arrival time for the remaining groups is consistently higher than the arrival time of the BEC data inputs. As a result, the arrival time of the mux selection input and the mux delay determine the remaining groups' delays.
4. The following formula is used to get group 2's area count: (FA + HA + Mux + BEC) = 43 gates XOR = 10(2 * 5) (BEC) Mux = 12(3 * 4) FA = 13(1 * 13) HA = 6(1 * 6) AND = 1, NOT = 1.

Table 1 : Modified SQRT CSLA groups' area count and delay

Delay (in ns)	Area (in gates)	Group
13	43	2
16	61	3
19	84	4
22	107	5

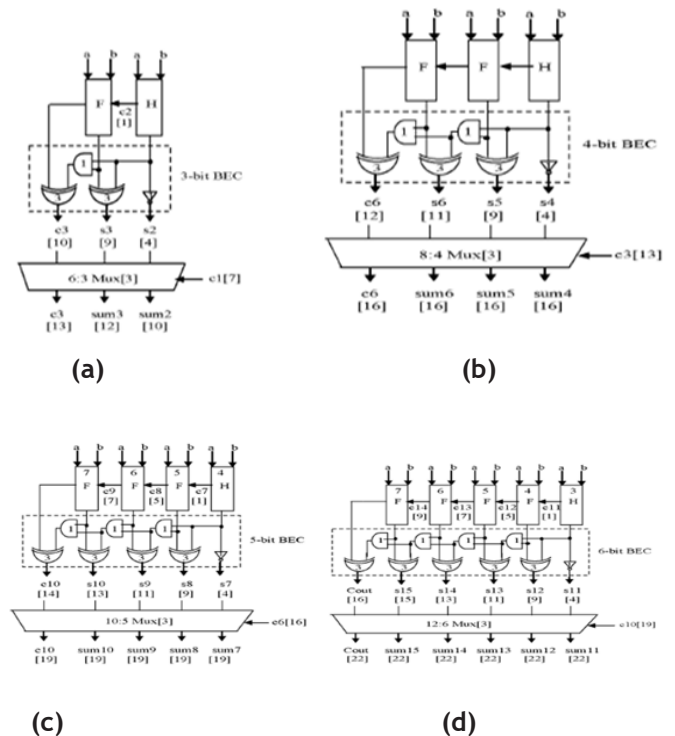


Fig 4: Delay and area analysis of the suggested SQRT CSLA Groups (a) and (b), groups (c) and (d) and (e)

5. Similarly, Table 2 lists and evaluates the anticipated maximum latency and area of the other groups of the updated Sqrt CSLA.

RESULTS AND DISCUSSION

This work mainly concentrates on the design of “Gate level modification” of a 64-bit Sqrt CSLA for reduced area application. The Model Sim and Xilinx ISE tools are used for simulation and synthesis process. The simulation and synthesis results are shown below. The simulation result for the 64-bit Sqrt CSLA for reduced area application in Model Sim (shown in Fig. 5) is as follows,

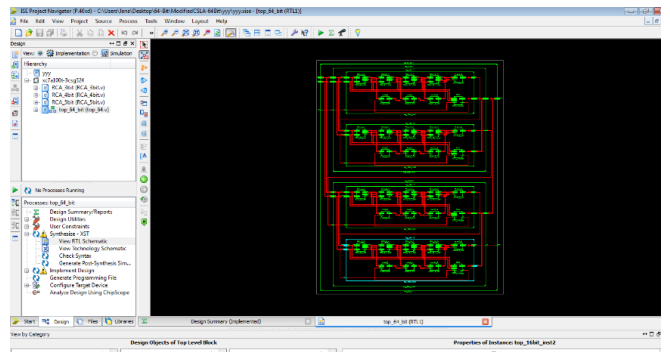


Fig. 7: RTL schematic

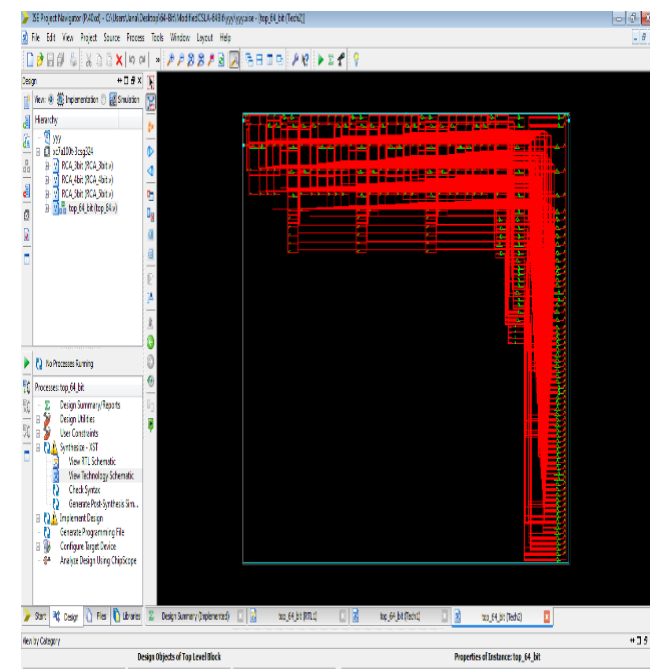


Fig. 8: Technology schematic

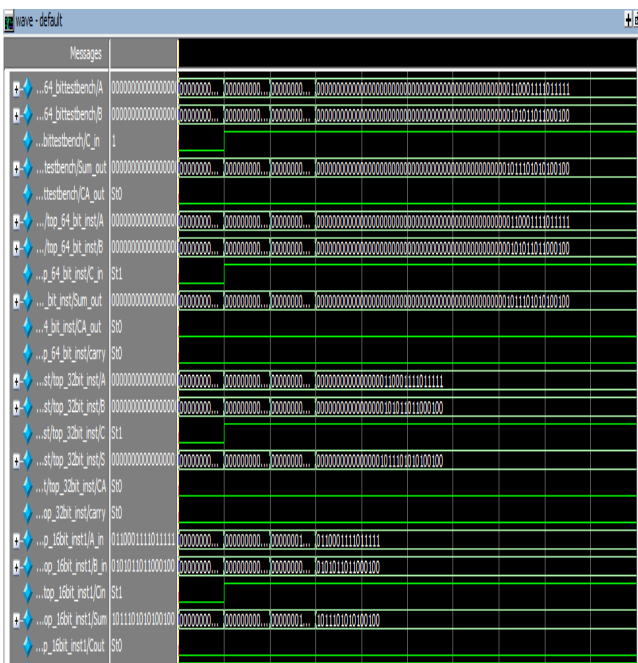


Fig. 5: Simulation result for the 64-bit Sqrt CSLA

Input Data:

A=64’d25567; B=64’d22212; Cin=1’b1;

Output Data:

Sum=64’d47780; Cout=1’b0;

The synthesis of 64-bit Sqrt CSLA for reduced area applications in Xilinx yields Top level schematic, RTL schematic and Technology View (shown in Fig. 5, 6 and 4.4) as follows,

The comparison table for regular & modified 64-bit Sqrt CSLA is shown below.

Table 2: Comparison Table for Regular & Modified 64-bit Sqrt CSLA

Parameter	Regular	Modified
No. of bits	64-bit	64-bit
Type of Algorithm	RCA	BEC
No. of LUT’s	162	135
No. of Adders	240	96
Area (in gates)	1352	1169
Delay (in ns)	20.461ns	17.596 ns
Speed	Low	High

It is evident from the foregoing that, in comparison to the ordinary approach, the 64-bit modified method has a lower delay. Thus, the new method greatly reduces both the area and the latency.

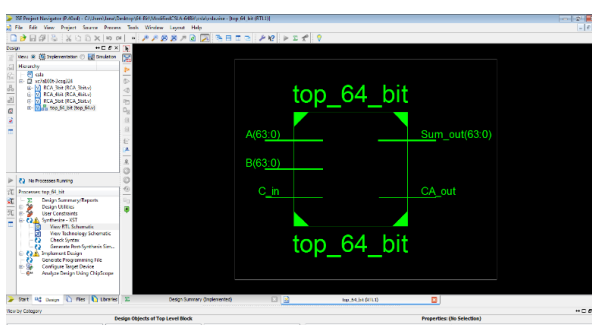


Fig.6: Top level schematic

CONCLUSION AND FUTURE WORK

This study presents an effective method for decreasing the size and delay of 64-bit SQRT CSLA architecture. All that needs are used to get the number of gates to be reduced in the structure and is to swap out the RCA for BEC. This work's fewer gates provide a substantial help in terms of decreased area, latency, and overall power. When compared to the standard 64-bit SQRT CSLA architecture, the modified architecture has a smaller area and less delay, as demonstrated by the comparison findings. The findings therefore indicate that the area and time would decrease when the improved method is used. In hardware implementation of VLSI, the 64-bit SQRT CSLA modified architecture is utilized since it is high speed, low area and delay, straightforward, and effective. This technology is utilized to execute several algorithms, such as FFT, FIR, and IIR, in a variety of applications, including multipliers and DSP.

The idea can be further developed for a larger bit count in the future. 512, 128, 256, and so forth.

REFERENCES

- [1] Chang, T. Y., & Hsiao, M. J. (1998). Carry-select adder using single ripple-carry adder. *Electronics letters*, 34(22), 2101-2103. <https://doi.org/10.1049/el:19981706>
- [2] Ramkumar, B., & Kittur, H. M. (2011). Low-power and area-efficient carry select adder. *IEEE transactions on very large scale integration (VLSI) systems*, 20(2), 371-375. <https://doi.org/10.1109/TVLSI.2010.2101621>
- [3] Manju, S., & Sornagopal, V. (2013, January). An efficient SQRT architecture of carry select adder design by common Boolean logic. In 2013 International Conference on Emerging Trends in VLSI, Embedded System, Nano Electronics and Telecommunication System (ICEVENT) (pp. 1-5). IEEE.
- [4] Diwakar, D., Shnain, A. H., Praveenraj, D. D. W., & Saini, R. (2024). Analysis, Assessment, And Management Of Environmental Air Pollution Using Environmental Engineering In Developing Countries. *Acta Innovations*, 12-21.
- [5] Usikalu, M. R., Alabi, D., & Ezeh, G. N. (2025). Exploring emerging memory technologies in modern electronics. *Progress in Electronics and Communication Engineering*, 2(2), 31-40. <https://doi.org/10.31838/ECE/02.02.04>
- [6] Mohanty, B. K., & Patel, S. K. (2014). Area-delay-power efficient carry-select adder. *IEEE transactions on circuits and systems II: express briefs*, 61(6), 418-422.
- [7] Naik, M. V. K. (2015, March). Design of carry select adder for low-power and high speed VLSI applications. In 2015 IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT) (pp. 1-4). IEEE.
- [8] Rooj, N., Majumder, S., & Kumar, V. (2018). A novel design of carry select adder (CSLA) for low power, low area, and high-speed VLSI applications. *Methodologies and application issues of contemporary computing framework*, 13-21.
- [9] Chakma, K. S. (Trans.). (2025). Flexible and wearable electronics: Innovations, challenges, and future prospects. *Progress in Electronics and Communication Engineering*, 2(2), 41-46. <https://doi.org/10.31838/ECE/02.02.05>
- [10] McCorkindale, W., & Ghahramani, R. (2025). Machine learning in chemical engineering for future trends and recent applications. *Innovative Reviews in Engineering and Science*, 3(2).
- [11] Devi, P., Girdher, A., & Singh, B. (2010). Improved carry select adder with reduced area and low power consumption. *International journal of computer applications*, 3(4), 14-18.
- [12] Lakshmi Jagan, B. O. (2024). Low-power design techniques for VLSI in IoT applications: Challenges and solutions. *Journal of Integrated VLSI, Embedded and Computing Technologies*, 1(1). <https://doi.org/10.31838/JIVCT/01.01.47>
- [13] Syrlybekkyzy, S., Serikbayeva, A., Suleimenova, B., Taizhanova, L., Dinmukhambet, B., Jumasheva, K., & Nurbayeva, F. (2024). Study Of The Relation Between The Capacity Of A Heliostationary Installation And Climatic Conditions In The Mangystau Region. *Acta Innovations*, 1-11. <https://doi.org/10.32933/ActaInnovations.44.1>
- [14] Ramkumar, B., & Kittur, H. M. (2011). Low-power and area-efficient carry select adder. *IEEE transactions on very large scale integration (VLSI) systems*, 20(2), 371-375. <https://doi.org/10.1109/TVLSI.2010.2101621>
- [15] Quinby, B., & Yannas, B. (2025). Future of tissue engineering in regenerative medicine: Challenges and opportunities. *Innovative Reviews in Engineering and Science*, 3(2).
- [16] Auvla, B. S., & Kalyan, R. (2015). Low Power, Area Efficient & High Performance Carry Select Adder on FPGA. *International Journal of Innovative Research in Computer and Communication Engineering*, 3(5).
- [17] Kadaru Prasanna, Dewkathé Divya, and Godise Badrinath, "Design of High Speed and Low Power Carry Select Adder", *International Journal of Research Publication and Reviews*, Vol 4, no 6, pp 1472-1478 June 2023.
- [18] Abdullah, D. (2024). Strategies for low-power design in reconfigurable computing for IoT devices. *SCCTS Transactions on Reconfigurable Computing*, 1(1). <https://doi.org/10.31838/RCC/01.01.39>
- [19] Mumintaj, S., Kayalvizhi, S., & Sangeetha, K. (2019). Low power and area efficient carry select adder using D-flip flop. *Int. J. Sci. Eng. Res*, 8(11), 964-967.
- [20] Baotic, A., & Silva, D. (2024). Techniques on controlling bandwidth and energy consumption for 5G and 6G wireless communication systems. *International Journal of Communication and Computer Technologies*, 12(2)