International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-8 Issue-5, March 2019

A Survey on Various VLSI Architectures of Carry Select Adder

Nagulapati Giri, Muralidharan D

Abstract*: Adders are the basic logical elements of arithmetic circuits in any microprocessor or digital signal processor. These act as basic blocks and are widely used components in digital integrated circuits. Optimizing such blocks increase the performance of integrated circuits. A small amount of area or delay reduction leads to great improvement in the performance. Carry chain plays a major role in adders on which the speed of an adder depends. Several adders have been proposed earlier to overcome the problems associated with area, power consumption and speed. Carry select adder is one among the adders with better performance. Carry select adder is favored broadly because it limits the issue of carry propagation delay. However, it occupies more area and power because of the repetitive blocks in the design. In this article, various available design methodologies of carry select adder, such as carry select adder using carry lookahead adder, square-root carry select adder using common Boolean logic, altered XOR gate and binary-to-excess-1 converter, have been discussed. The efficacy of all the design methodologies have been investigated by comparing the parameters like area, delay and power consumption. The design with high efficacy can be used in high speed multiplication, arithmetic logic units, advanced microprocessor design and so on. All the architectures are simulated in Cadence Virtuoso Analog Design Environment and gpdk180 library was utilized.

***Index Terms*: Cadence, Carry select adder, Ripple carry adder, Propagation delay**

I. INTRODUCTION

In the VLSI system, an adder is the basic logical element. The adder circuit predominantly determines the performance of any processor. The performance criteria include speed of the circuit, power consumption, area, wiring complexity, etc. The performance varies with the logic styles used to design the adder [1]. The logic style has a great influence on the area, speed, and power consumption. Area depends on the number of transistors utilized in the design and the size of each transistor. Delay majorly depends on the carry propagation path. The increase in demand for high performance and low power devices made the research growing towards the development of different design methodologies for different applications [2]. The performance of binary adders is limited by carry propagation delay [3]. Ripple carry adder (RCA) is constructed by cascading multiple single bit full adders [4]. It is simple to design but with an increase in the number of bits to be computed, the carry propagation delay increases drastically. The limitations of RCA can be overcome using carry lookahead adder (CLAA). CLAA has the disadvantage of increase in area with the number of input bits. Then comes the carry select adder CSLA which fits in the frame between RCA and CLAA and is utilized generally in high-speed applications [5]. The CSLA structure consists of two units: (i) generation unit and (ii) selection unit [6]. It has a disadvantage of more area consumption. The carry select algorithm of 8-bit CSLA is as follows:

1) Perform three ripple carry additions simultaneously
   a) One addition with lower half bits
   b) Two additions with upper half bits
      i. One with Cin set to 0
      ii. One with Cin set to 1
2) Pick the correct upper half.

II. VARIOUS METHODOLOGIES

A. Conventional CSLA

The CSLA provides the high performance of the process in terms of speed by depreciating the carry propagation delay. This improvement in performance can be achieved by independently producing several carries and using the carries to determine the produced sums. Fig. 1 shows the 16-bit conventional CSLA [7]. It consists of dual RCAs and multiplexers at each stage. The addend and augend are subdivided and added twice to generate two subsums. One addition is done by fixing the input Cin = 0 and the other is done fixing Cin = 1. The carry from the former block is utilized to choose the right sum and carry for the following stage using multiplexers. The main downside of regular CSLA comes from more area utilization because of usage of repetitive RCA blocks at each stage [8]. Fig. 2 displays the snapshot of the 16-bit conventional CSLA designed in Cadence. Fig. 3 and 4 show the square-root CSLA (SQRT-CSLA) and snapshot of it designed in Cadence. It varies from conventional CSLA in the sense that the block size is variable which is uniform in the former. Because of the variable input bit size, the delay is a little bit reduced [9, 10]. In Fig. 3, the adder with non-uniform block sizes of 2-2-3-4-5 is shown [9]. The idea of variable size was preferred in all the following designs.

Manuscript published on 30 March 2019.
***Correspondence Author(s)**
Nagulapati Giri, VLSI Design, School of Computing, SASTRA Deemed To Be University, Thanjavur, India.
Muralidharan D, Department of Information Technology, School of Computing, SASTRA Deemed To Be University, Thanjavur, India.

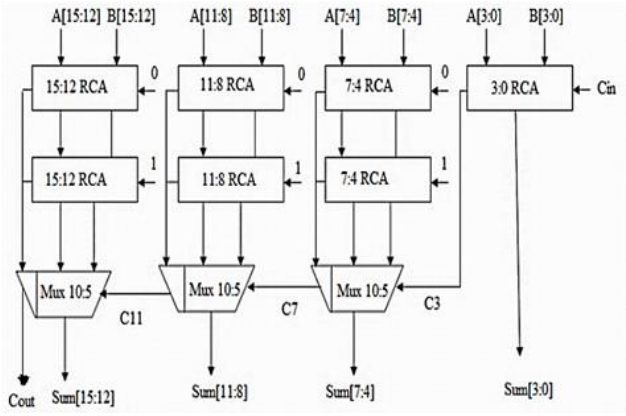

Fig. 1 Uniform 16-bit CSLA

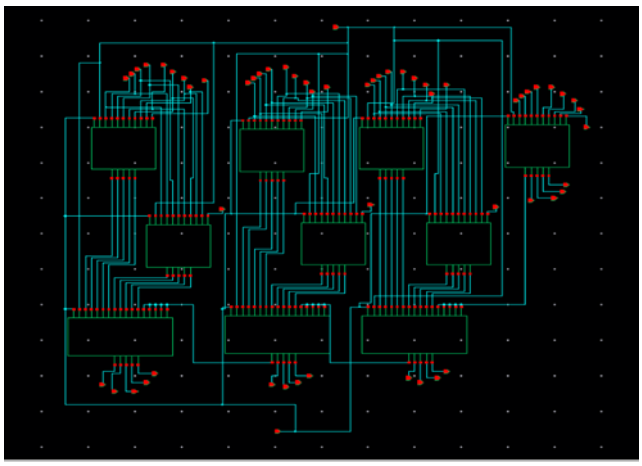

Fig. 2 Uniform 16-bit CSLA designed in Cadence

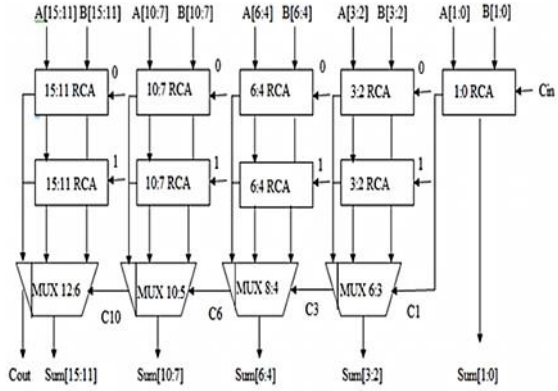

Fig. 3 Non-uniform 16-bit SQRT-CSLA

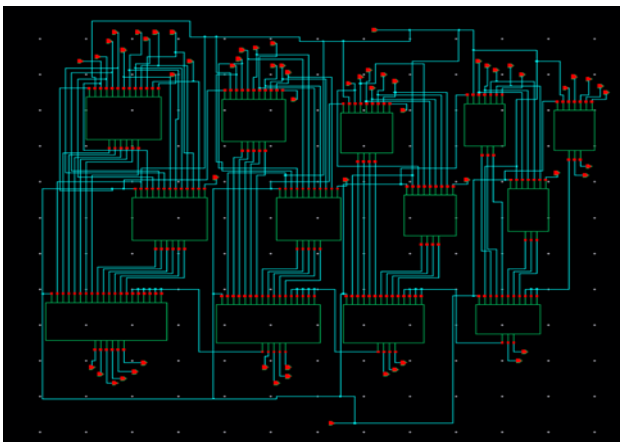

Fig. 4 Non-uniform 16-bit SQRT-CSLA designed in Cadence

B. CSLA using CLAA

RCAs have the compact design (O(n) area) along with large delay (O(n) time) [11]. Since in RCA one full adder awaits for the previous full adder to complete its operation and propagate the carry, the delay is more. To avoid this problem, CLAA is used instead of the RCA. CLAA produces the output carries before calculating the sum which reduces the waiting time of adders. The equation (1) that generate and propagate the carry are given as follows:

$$G_i = A_i \bullet B_i \qquad (1)$$
$$P_i = A_i \oplus B_i$$

Hence, sum and carry are autonomous of the previous bits. However, as the bit size increases, CLAA consumes more area. It is faster for less number of input bits and gradually becomes slower as the bit size increases. It can be concluded that though a reduction in delay may be achieved for less number of input bits, area and power consumption are always high [12]. Fig. 5 and 6 show the CSLA using CLAA and the same designed in Cadence.

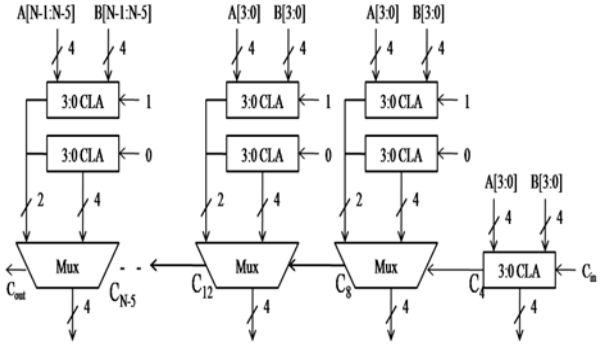

Fig. 5 Regular CSLA using CLAA

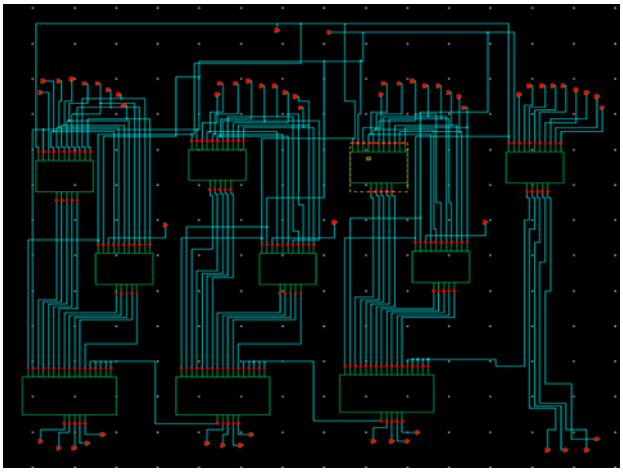

Fig. 6 Regular CSLA using CLAA designed in Cadence

C. SQRT-CSLA using common Boolean logic

Fig. 7 represents the truth table of the single-bit full adder, from which it tends to be seen that the summation signal when Cin = 0 is the inverse of itself when Cin = 1. One XOR gate with INV gate is only needed to generate the summation signal pair [13].

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

The correct summation output is decided by the ready carry-in signal. It is also possible that using the same carry-in signal, the correct carry-out output can be determined. In this way, both the summation and carry-out generation circuits can be in parallel. Fig. 8 and 9 display CSLA developed by sharing the regular Boolean logic term [14].

Cin	A	B	Sum	Carry
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Fig. 7 The truth table of the single-bit full adder

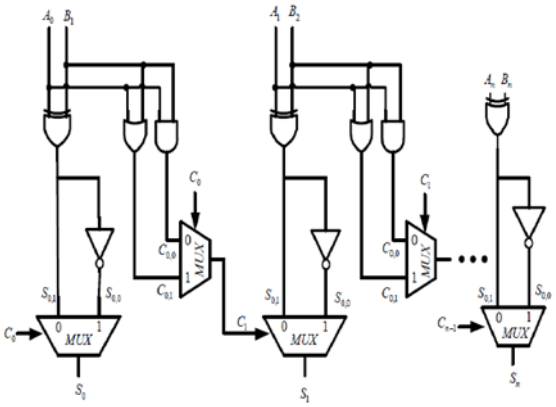

Fig. 8 High speed CSLA by sharing the regular Boolean logic

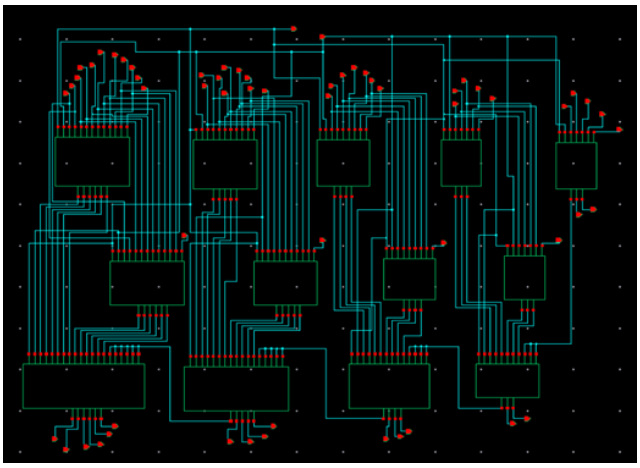

Fig. 9 High speed CSLA by sharing the regular Boolean logic designed in Cadence

D. SQRT-CSLA using altered XOR gate

Ragunath and Sakthivel [15] proposed a new strategy to design SQRT-CSLA, in which an altered XOR gate was utilized. Half adder and full adder are the principal blocks in any adder circuit, which comprises of XOR gates. The basic idea is to modify the XOR gate with the expectation of reduction in number of transistors required and power consumption. The traditional and modified XOR gates are shown in Fig. 10 and 11 [15]. There is a reduction from five gates in traditional XOR gate to four gates in modified XOR gate. Fig. 12 demonstrates the RCA utilizing modified XOR gate [16]. The logical expression (2) is given as follows:

$$\begin{aligned} Output &= (A+B)\bullet\overline{(A\bullet B)} \\ &= (A+B)\bullet(\overline{A}+\overline{B}) \\ &= (A\oplus B) \end{aligned} \quad (2)$$

Modified XOR gate is put in the design of the RCA and CSLA which lessens the number of gates needed leading to area minimization. Fig. 13 represents SQRT-CSLA using modified XOR gate designed in Cadence.

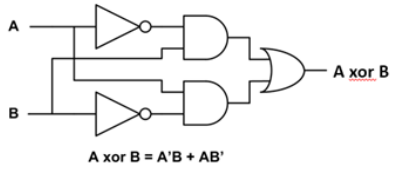

Fig. 10 Conventional XOR gate design

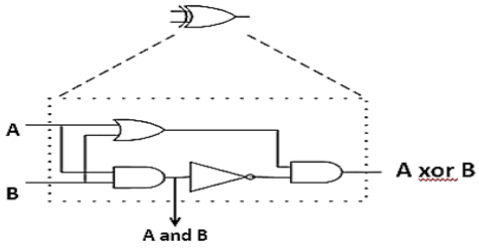

Fig. 11 Modified design of XOR gate

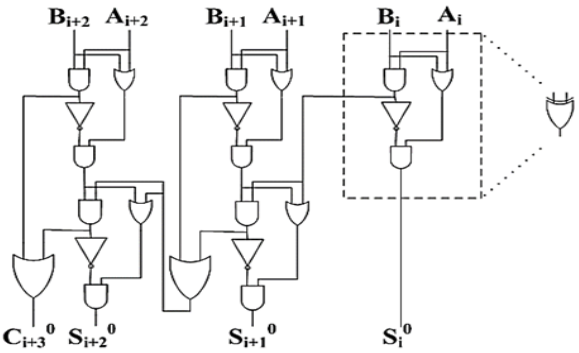

Fig. 12 3-Bit RCA implementation using modified XOR gate

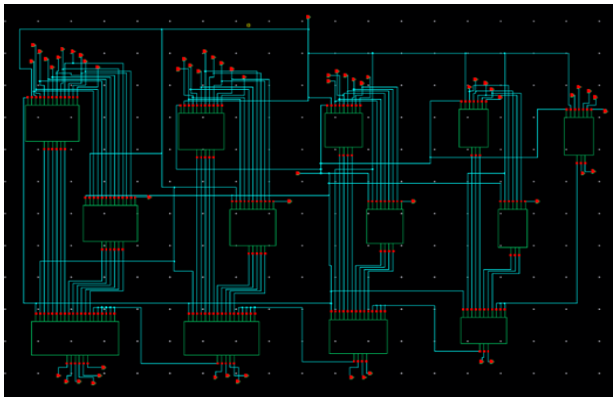

Fig. 13 SQRT-CSLA implementation using altered XOR gate designed in Cadence

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

E. SQRT-CSLA using binary-to-excess-1 converter

In this methodology, the RCA with Cin = 1 was supplanted by BEC-1 unit in this methodology. At each stage, an n-bit RCA is supplanted by an n + 1-bit BEC-1 unit. The output of RCA is given as input to the BEC-1. The outputs of BEC-1 and RCA are given as inputs to the multiplexer which selects the correct summation signal and carry based on actual carry input [17, 18]. A 4-bit BEC-1 unit is shown in Fig. 14. The Boolean expressions for BEC-1 (3) are expressed as follows:

$$X_0 = \sim B_0$$
$$X_1 = B_0 \wedge B_1 \quad (3)$$
$$X_2 = B_2 \wedge (B_0 \,\&\, B_1)$$
$$X_3 = B_3 \wedge (B_0 \,\&\, B_1 \,\&\, B_2)$$

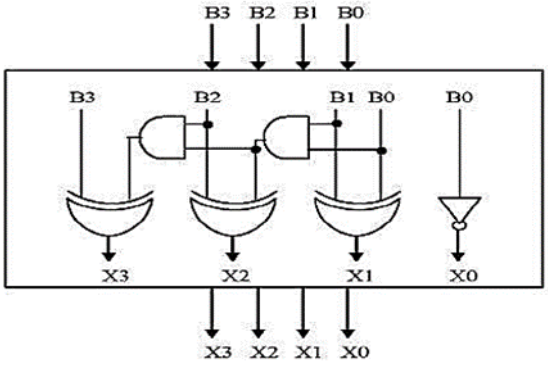

Fig. 14 The logic diagram of 4-bit BEC-1

The modified SQRT-CSLA using BEC-1 consumes fewer logic gates as it eliminates RCAs. It is observed that both the conventional CSLA and SQRT-CSLA using BEC-1 has almost the same power consumption. The drawback of this methodology is that there is a slight increment in delay [19, 20]. Fig. 15 and 16 display a 16-bit SQRT-CSLA constructed by replacing RCAs with BEC-1 units and snapshot of it designed in Cadence.

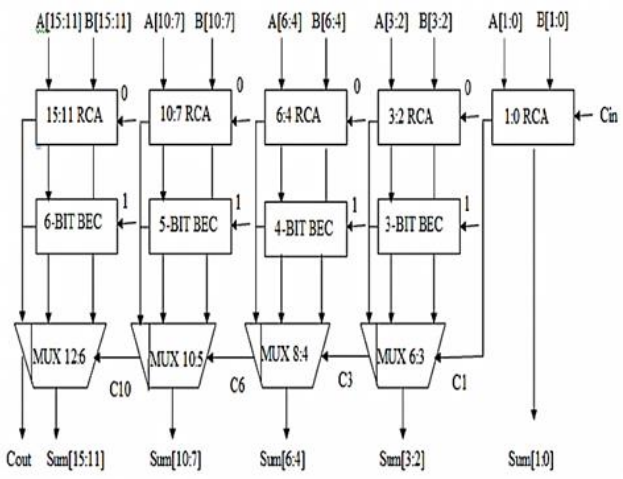

Fig. 15 16-Bit area efficient SQRT-CSLA using BEC-1 unit

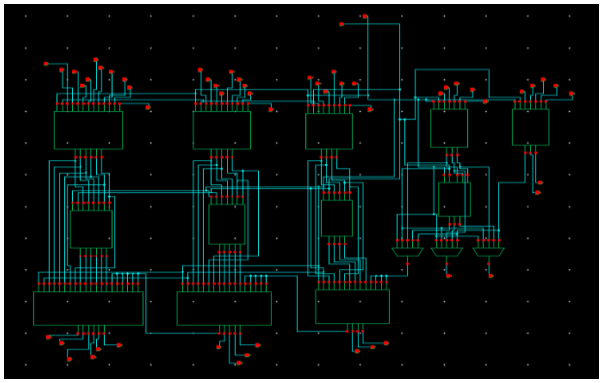

Fig. 16 16-Bit area efficient SQRT-CSLA using BEC-1 unit designed in Cadence

III. RESULTS AND DISCUSSION

All the architectures are simulated in Cadence Virtuoso Analog Design Environment and gpdk180 library was utilized. Table 1 shows the simulated results viz. number of transistors, delay and power consumption of various 16-bit architectures. Results show that SQRT-CSLA using BEC-1 has less power consumption and require less number of transistors when compared with other architectures. However, SQRT-CSLA using BEC-1 has an increased delay when compared with delay of other architectures. The results show that CSLA using CLAA is not convenient for high input bit size. Fig. 17–19 depict comparison of various adders regarding number of transistors required, delay and power consumption.

Table 1. Comparison of various 16-bit adder architectures

Adder architecture	Number of transistors	Delay (ps)	Power consumption (µW)
Conventional CSLA	2036	314.8	221
SQRT-CSLA	2220	190.7	302.6
CSLA using CLAA	3060	597.6	429.6
CSLA using Boolean logic	2416	130.4	280.4
SQRT-CSLA using altered XOR	1740	116.4	282.9
SQRT-CSLA using BEC-1	1728	241.6	223.1

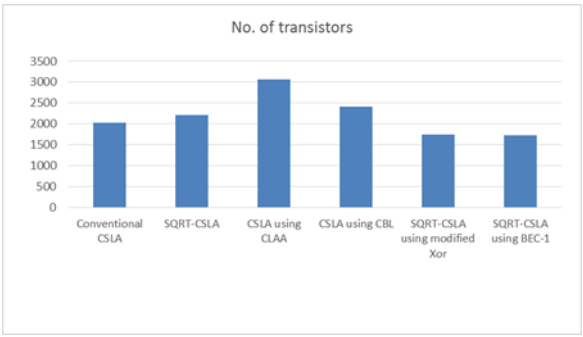

Fig. 17 Comparison of adders for number of transistors

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

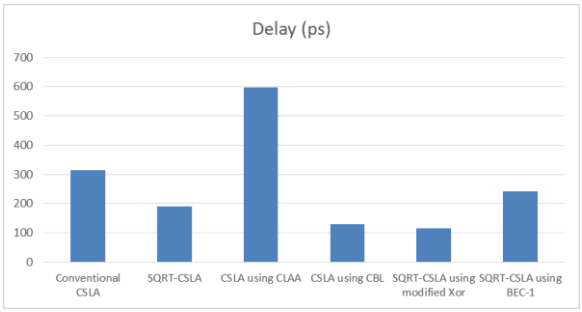

Fig. 18 Comparison of adders for delay

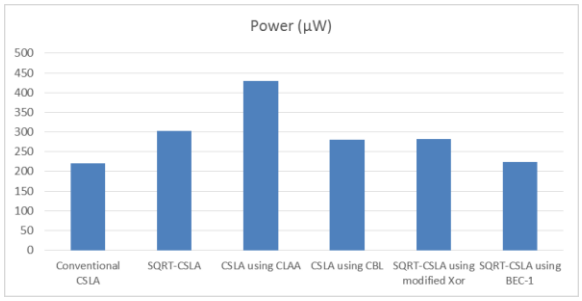

Fig. 19 Comparison of adders for power consumption

IV. CONCLUSION

In this article, various architectures of CSLA, such as SQRT-CSLA, CSLA using CLAA, modified XOR, common Boolean logic and BEC-1, so far available have been addressed. The results revealed that SQRT-CSLA using BEC-1 can be used in applications where the power consumption and transistor count need to be minimum. The calculated results also revealed that SQRT-CSLA using altered XOR gate is preferable in the designs where performance should be high. Every one of these models are designed for 16-bit input size only. This work can be additionally extended for 32, 64 and 128-bit input sizes. Since there is a scope for further reduction in area, delay, and power consumption, the future work involves proposing a new design methodology focusing on further minimization of the above constraints.

REFERENCES

1. R. Srilatha, *Simulation and Analysis of High Speed Conditional Carry Select Adder*. Kingsville, TX: Texas A&M University-Kingsville, 2003.
2. T. W. Lynch, *Binary Adders*. Austin, TX: The University of Texas at Austin, 1996.
3. O. Macsorley, "High-speed arithmetic in binary computers," *Proceedings of the IRE*, vol. 49, no. 1, pp. 67–91, 1961, DOI: 10.1109/jrproc.1961.287779.
4. J. M. Rabaey, *Digital Integrated Circuits*. London: Pearson Education, 2003.
5. B. S. Premananda, M. K. Chandana, K. P. Shree Lakshmi, and A. M. Keerthi, "Design of low power 8-bit carry select adder using adiabatic logic," in *2017 International Conference on Communication and Signal Processing (ICCSP)*, 2017, DOI:10.1109/iccsp.2017.8286696.
6. B. Parhami, *Computer Arithmetic: Algorithms and Hardware Designs*. New York: Oxford University Press, 2010.
7. O. J. Bedrij, "Carry-select adder," *IRE Transactions on Electronic Computers*, vol. EC-11, no. 3, pp. 340–346, 1962, DOI: 10.1109/iretelc.1962.5407919.
8. A. Tyagi, "A reduced-area scheme for carry-select adders," *IEEE Transactions on Computers*, vol. *42, no.* 10, pp. 1163–1170, 1993, DOI: 10.1109/12.257703.
9. M. Priya and S. Mamta, "Design of modified area efficient square root carry select adder (SQRT CSLA)," *International Journal of Industrial Electronics and Electrical Engineering*, special issue 4, pp. 216–219, 2015.
10. E. Abu-Shama and M. Bayoumi, "A new cell for low power adders," in *1996 IEEE International Symposium on Circuits and Systems. Circuits and Systems Connecting the World (ISCAS'96)*, 1996, DOI: 10.1109/iscas.1996.541898.
11. B. Amelifard, F. Fallah, and M. Pedram, "Closing the gap between carry select adder and ripple carry adder: A new class closing the gap between carry select adder and ripple carry adder: A new class of low-power high-performance adders," in *Sixth International Symposium on Quality of Electronic Design (ISQED'05)*, 2005, DOI: 10.1109/isqed.2005.131.
12. J. Lakshmesha and K. U. Rani, "A novel ripple/carry lookahead hybrid carry select adder architecture," *International Journal of Computer Applications*, vol. 70, no. 27, pp. 5–9, 2013.
13. I. C. Wey, C. C. Ho, Y. S. Lin, and C. C. Peng, "An area-efficient carry select adder design by sharing the common Boolean logic term," in *Proceedings of the International Multiconference of Engineers and Computer Scientists*, Hong Kong, vol. 2, 2012.
14. S. Sanooja and B. Aswathi, "A modified carry select adder using common Boolean logic," *International Journal of Engineering and Technical Research*, vol. 3, no. 7, pp. 229–232, 2015.
15. G. Ragunath and R. Sakthivel, "Low-power and area-efficient square-root carry select adders using modified XOR gate," *Indian Journal of Science and Technology*, vol. 9, no. 5, pp. 1–8, 2016, DOI: 10.17485/ijst/2016/v9i5/87181.
16. J. R. Padmasri, P. A. Christina, P. Kavitha, and T. Mullai, "Modified area efficient carry select adder (MA-CSLA)," *International Journal of Advanced Electrical and Electronics Engineering*, vol. 2, no. 2, pp. 75–79, 2013.
17. M. Anto Bennet, S. Sankaranarayanan, V. Banu Priya, P. Jaya Pretheena, and S. Yamini, "Performance and analysis of low power, area-efficient and high speed carry fast adder," *International Journal on Smart Sensing and Intelligent Systems*, special issue, pp. 522–538, 2017.
18. S. Prajwal, "Enhanced low power, fast and area efficient carry select adder," *International Journal of Research in Engineering and Technology*, vol. 3, no. 5, pp. 441–443, 2014.
19. B. Ram Kumar and M. K. Harish, "Low-power and area-efficient carry select adder," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 20, no. 2, pp. 371–375, 2011.
20. T. Abhiram, T. Ashwin, B. Sivaprasad, S. Aakash, and J. P. Anita, "Modified carry select adder for power and area reduction," in *2017 International Conference on Circuits Power and Computing Technologies*, 2017, DOI: 10.1109/ICCPCT.2017.8074371.

AUTHORS PROFILE

Nagulapati Giri is currently pursuing M. Tech. in VLSI Design at SASTRA Deemed To Be University, Thanjavur, Tamil Nadu, India. He received his B.E. degree in Electronics and Communication Engineering from Saveetha Engineering College, Chennai, Tamil Nadu, India.

Muralidharan D has been working at SASTRA Deemed to be University as Assistant Professor since 2006. He completed Ph.D. in Network-on-Chip at SASTRA Deemed to be University. He completed his M.E. degree in VLSI Design at Government College of Technology, Coimbatore, Tamil Nadu, India. He obtained his Bachelor's degree in Electronics and Communication Engineering from Sri Ramakrishna Engineering College, Coimbatore, Tamil Nadu, India. He has published more than 20 papers in various journals.

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)