

## High speed modified carry save adder using a structure of multiplexers

Ahmed Salah Hameed, Marwa Jawad Kathem

Department of Computer Engineering, College of Engineering, University of Diyala, Iraq

### Article Info

#### Article history:

Received Jun 4, 2020

Revised Aug 20, 2020

Accepted Nov 4, 2020

#### Keywords:

Binary adder

Carry save adder

High speed

Low power

Multi operand addition

### ABSTRACT

Adders are the heart of data path circuits for any processor in digital computer and signal processing systems. Growth in technology keeps supporting efficient design of binary adders for high speed applications. In this paper, a fast and area-efficient modified carry save adder (CSA) is presented. A multiplexer based design of full adder is proposed to implement the structure of the CSA. The proposed design of full adder is employed in designing all stages of traditional CSA. By modifying the design of full adder in CSA, the complexity and area of the design can be reduced, resulting in reduced delay time. The VHDL implementations of CSA adders including (the proposed version, traditional CSA, and modified CSAs presented in literature) are simulated using Quartus II synthesis software tool with the altera FPGA EP2C5T144C6 device (Cyclone II). Simulation results of 64-bit adder designs demonstrate the average improvement of 17.75%, 1.60%, and 8.81% respectively for the worst case time, thermal power dissipation and number of FPGA logic elements.

*This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.*



### Corresponding Author:

Ahmed Salah Hameed

Department of Computer Engineering

University of Diyala

College of Engineering Branch-P.O.BOX-1, Baquba, Diyala, Iraq

Email: ahmedhameed\_eng@uodiyala.edu.iq

## 1. INTRODUCTION

Binary addition is a very basic and important operation in digital computer system. Adder circuits are used in implementing most of the other arithmetic operations. It is becoming increasingly important to enhance the data path units' performance to meet grows of high performance processor. The use of the adder unit in implementing most arithmetic operations increases the need to have low power, high speed and small area design of adder [1-7]. Adders are primarily used as basic units in implementing digital signal processing (DSP) systems for applications like design of analog to digital converters (ADCs) and design of digital filters [8-13]. Different techniques are used in the design of a multi bit binary adder, for example ripple carry adder (RCA), carry increment adder (CIA), carry skip adder (CSkA), carry look ahead adder (CLA), carry select adder (CSIA), carry save adder (CSA), and carry bypass adder (CBA) [14-18].

The very basic algorithm for multi-bit binary addition is RCA. RCA is the simplest but not the best way to build the multi-bit binary addition, through the use of N cascading full adders to add two N-bit numbers. Accumulated delay is produced in RCA adder as bits number N is large through the rippling of the carry in the N cascading full adders. On the other hand, CSA is one of the best architecture used in solving the delay problem associated with RCA. CSA is a powerful architecture for fast multi-operand addition adder. CSAs are primarily used with array multipliers to build the process of accumulating the partial products [19-23].

The major goal of this work is to provide a fast, and area efficient modified implementation of the CSA. The rest of the paper is organized as follows: section 2 deals with the related work. Section 3 describes the implementation of the proposed CSA. Finally, in section 4 and 5 simulation results and conclusion are discussed.

**2. RELATED WORK**

**2.1. Traditional CSA**

A traditional CSA uses a ladder of full adder units to build the structure of addition. Unlike basic structure of RCA, Saving the carry to a next level of addition is used to reduce the rippling of carry in CSA [24]. CSA is fast but not area and power efficient because it uses a large number of full adders including RCA uses in the final stage. The weakness in CSA is in its final stage which is basically designed with a RCA. Figure 1 shows the general structure of four operands N-bit traditional CSA. The basic implementation of 1-bit full adder used in the structure of traditional CSA is founded in accordance to the following equations:

$$SUM = A \oplus B \oplus Cin \tag{1}$$

$$Cout = (A \oplus B)Cin + AB \tag{2}$$

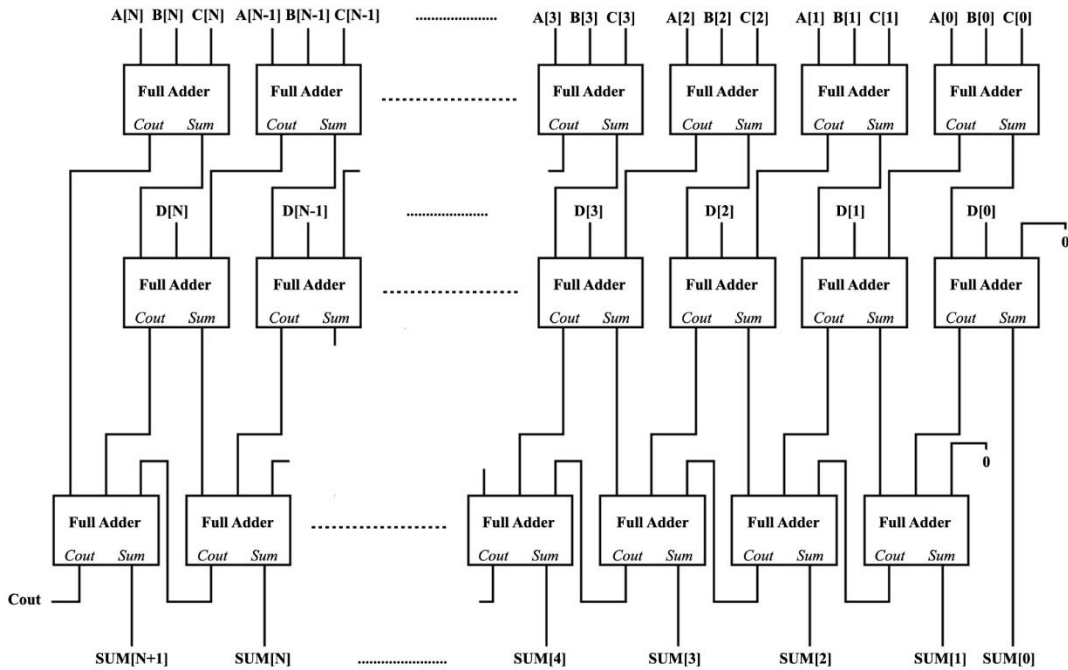


Figure 1. The general structure of four operands N-bit traditional CSA

**2.2. CSA using CLA**

To solve the delay caused by RCA in the final stage of traditional CSA, a modified version of CSA was presented in [24]. The modified version was done through the use of CLA in implementing the final stage of traditional CSA. All the full adders keep using the same implementation of full adder used with traditional CSA as mentioned previously in equations 1, and 2. The full adders in the last stage are modified to fit the implementation of CLA according to the following equations:

$$Pi = Ai + Bi \tag{3}$$

$$Gi = Ai * Bi \tag{4}$$

$$SUM = Pi \oplus Ci \tag{5}$$

$$C_{out} = G_i + P_i * C_i \tag{6}$$

The general structure of the modified CSA using CLA is shown in Figure 2. The use of CLA adder aims to speed up the process of adding. The modified CSA in [10] is faster than the traditional CSA, but still not area efficient implementation.

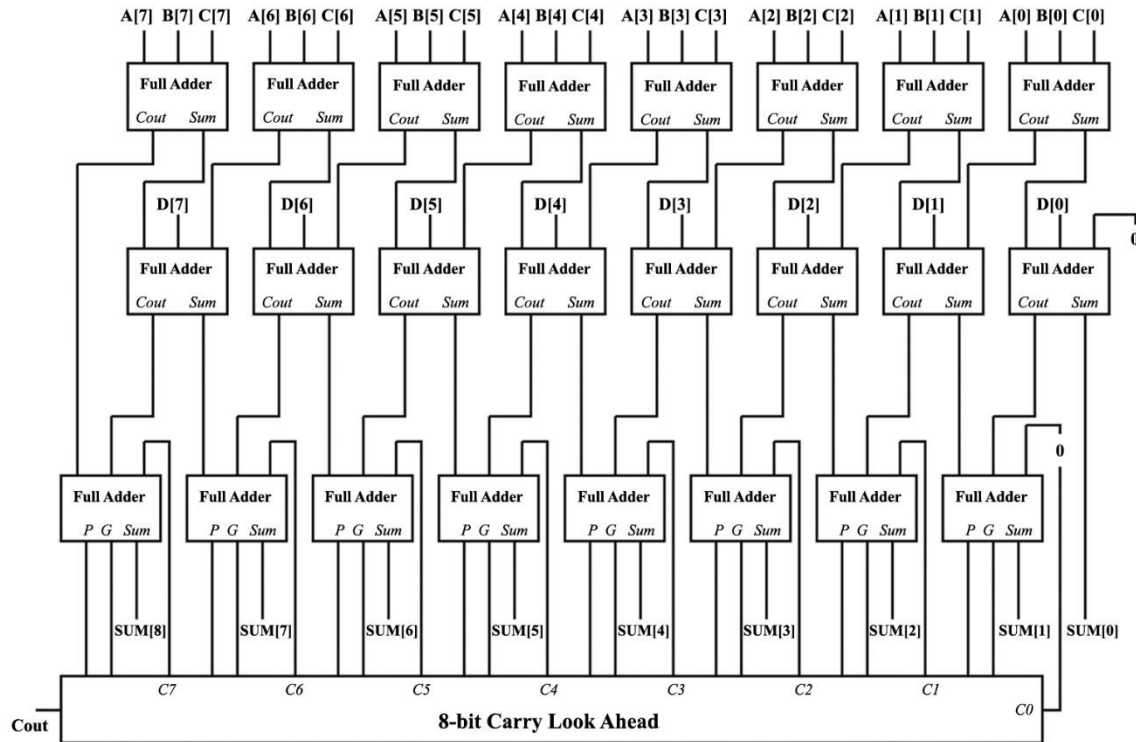


Figure 2. The general structure of four operands 8-bit CSA with CLA

### 2.3. CSA using modified full adder

The authors in [25] use the same structure of a traditional CSA with a modified design of full adder. The suggested design of full adder is based on using two (4 to 1) multiplexers. The design aims to reduce the delay time of traditional CSA and present area efficient implementation. The design of full adder based on multiplexer used in [25] is shown in Figure 3. Using the structure of MUXs to build the full adders in CSA participates in speeding up the process of adding. The generated CSA is faster than the traditional CSA, but also still not area and power efficient.

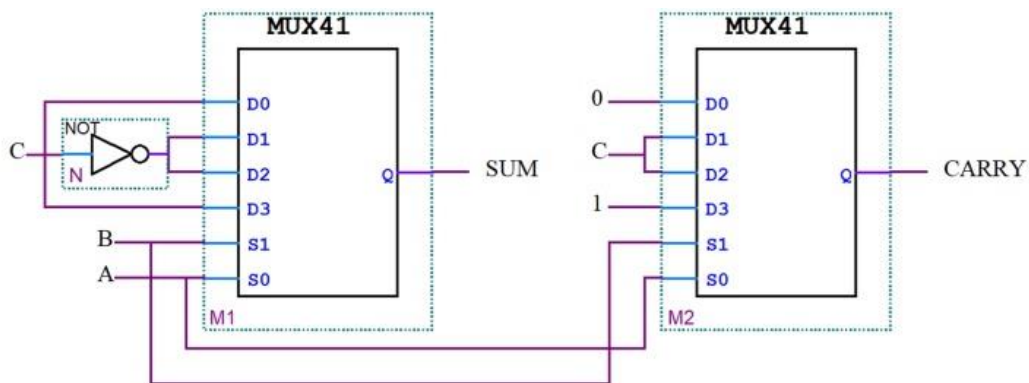


Figure 3. One-bit full adder used in the structure of modified CSA [25]

**3. PROPOSED WORK**

It is clearly seen that full adder represent the basic unit design in the structure of implementing the traditional CSA. Accordingly, the proposed CSA design relies on using a modified structure of full adder. The suggested structure of full adder aims to reduce the power, the area, and the delay time for the CSA design. A new sum and carry generating architecture for full adder is used with a structure of three (2 to 1) multiplexers. Based on the basic truth table of 1-bit full adder shown in Table 1, an implementation of Sum and Carry outputs could be done as following:

The  $(X \oplus Y)$  could be simply implemented with one 2 to 1 multiplexer as follow:

$$X \oplus Y = Y \quad \text{when } X = 0 \tag{7}$$

$$X \oplus Y = \overline{Y} \quad \text{when } X = 1 \tag{8}$$

Sum could be generated as output of 2 to 1 multiplexer using  $(X \oplus B)$  as a selector:

$$Sum = Cin \quad \text{when } X \oplus Y = 0 \tag{9}$$

$$Sum = \overline{Cin} \quad \text{when } X \oplus Y = 1 \tag{10}$$

Carry could be generated as output of 2 to 1 multiplexer using  $(X \oplus B)$  as a selector:

$$Cout = X \quad \text{when } X \oplus Y = 0 \tag{11}$$

$$Cout = Cin \quad \text{when } X \oplus Y = 1 \tag{12}$$

Table 1. The basic truth table of 1-bit full adder

X	Y	Cin	$X \oplus Y$	Sum	Cout
0	0	0	0	0	0
0	0	1	0	1	0
0	1	0	1	1	0
0	1	1	1	0	1
1	0	0	1	1	0
1	0	1	1	0	1
1	1	0	0	0	1
1	1	1	0	1	1

The general block diagram of 1-bit multiplexer based design of full adder using three (2 to 1) multiplexers is shown in Figure 4. Based on the suggested architecture of full adder, the proposed implementation of the CSA is replacing all the units of full adders in the traditional structure of CSA with the suggested one shown in Figure 4.

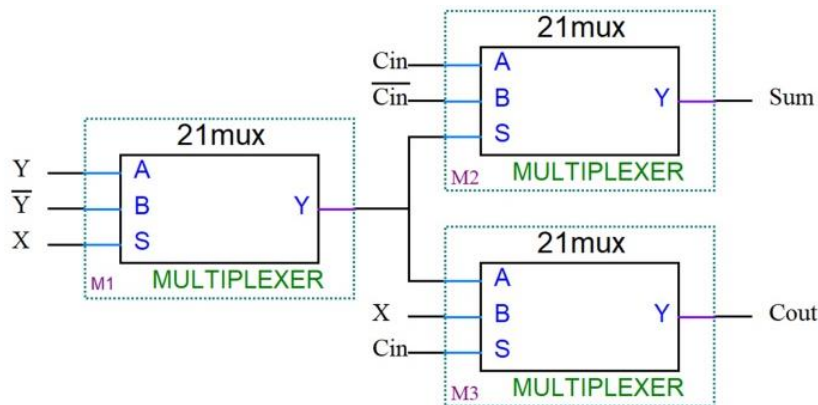


Figure 4. The general block diagram of 1-bit full adder (proposed)

#### 4. SIMULATION RESULTS

This study demonstrates the addition of different design structures for CSA. Two different sizes of adders, i.e., 8-bit and 64-bit four binary operands for the implemented designs including (traditional CSA, Modified CSA [24], Modified CSA [25], and proposed version of CSA) are simulated to do the performance analysis in terms of time delay, power dissipation, and number of logic elements.

To increase the reliability of measures, all the designs were carried out using Quartus II synthesis software tool with VHDL to obtain simulation results. Altera FPGA EP2C5T144C6 device (Cyclone II) was selected and worst case time, thermal power dissipation reports, and number of logic elements were used to demonstrate the performance of all the designs. The simulation results for 8-bit and 16-bit CSA design structures discussed in this paper are shown in Table 2.

Table 2. Results for the proposed and related CSA design structures

Size of adder	Type of adder	Worst case time (ns)	Total thermal power dissipation (mW)	No. of FPGA blocks total logic elements
8-bit	Traditional CSA	19.872	34	48
	CSA [24]	19.184	33.97	70
	CSA [25]	19.5	33.98	48
	Proposed CSA	18	33.95	46
64-bit	Traditional CSA	47.807	148.44	352
	CSA [24]	74.764	150.11	511
	CSA [25]	45.054	148.39	370
	Proposed CSA	37.059	146.01	321

Comparisons between the discussed designs in terms of worst case time, thermal power dissipation reports, and number of logic elements were presented as follows:

##### 4.1. Worst case time

Figure 5 shows the comparison of worst case time results, and this reveals that the proposed CSA design offers better performance than others. When compared to existing designs, the 8-bit proposed version of the CSA has around 6.17% improvement in worst case time and the 64-bit proposed version has around 17.75% improvement in worst case time.

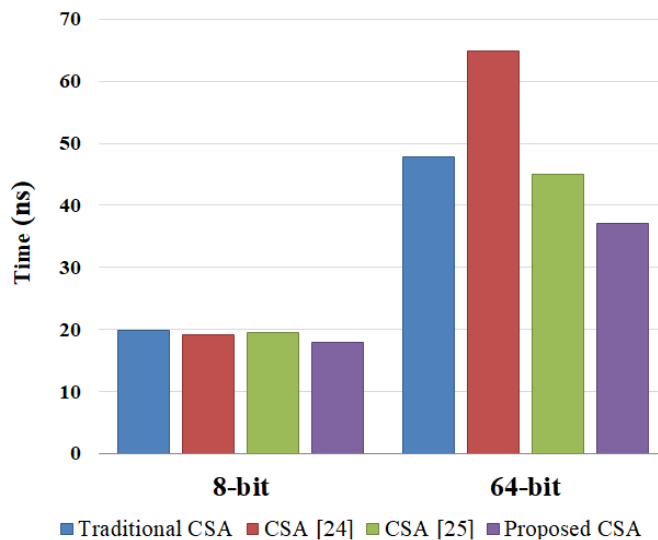


Figure 5. Worst case time results

##### 4.2. Total thermal power dissipation

The power dissipation results of all the designs are shown in Figure 6. It is evident that the proposed CSA requires slightly lower power dissipation than other available designs. The 8-bit proposed version of the CSA has around 0.06% improvement in the total thermal power dissipation and the 64-bit proposed version has around 1.60% improvement in the total thermal power dissipation.

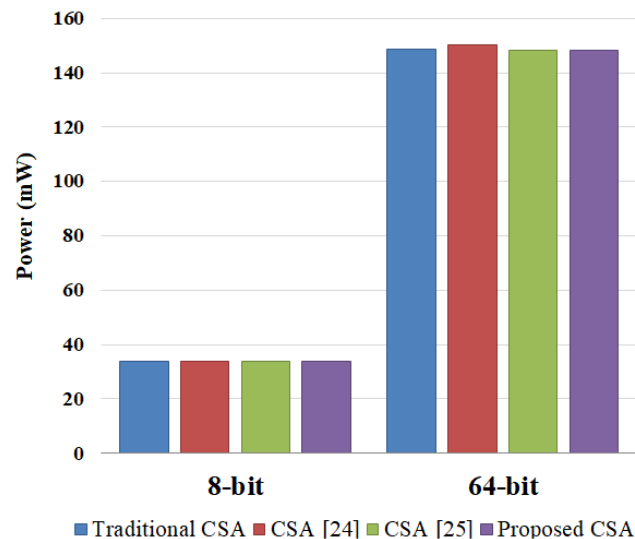


Figure 6. Total thermal power dissipation results

#### 4.3. Number of FPGA logic elements

Figure 7 shows the total number of logic element resources used in the selected FPGA device and the results reveal that the second version of proposed CSA offers a lower number of logic elements than other available designs. The 8-bit proposed version of the CSA has around 4.17% improvement in the number of total logic elements and the 64-bit proposed version has around 8.81% improvement in the number of total logic elements.

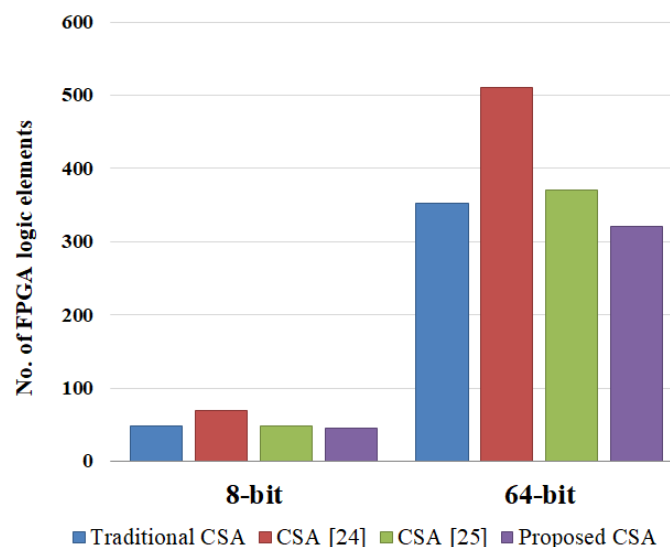


Figure 7. Number of FPGA logic elements

## 5. CONCLUSION

In this paper we have presented a new structure to implement high speed, and area efficient CSA. A design of multiplexer based full adder with three (2 to 1) multiplexers is used to implement the proposed structure of CSA. The aim of the proposed design is reducing the complexity and area of implementation to reduce the delay time. A detailed comparison of worst case time, thermal power dissipation, and number of logic elements between the proposed implementation and the related designs is done. Simulations in Quartus II synthesis software tool with VHDL demonstrate that the proposed version of CSA is faster than the existing designs. Simulation results of 64-bit adder designs demonstrate the average improvement of 17.75%,

1.60%, and 8.81% respectively for the worst case time, thermal power dissipation and number of FPGA logic elements. We accomplished the best results by a design of multiplexer based full adder in which a fast and reduced area implementation is achieved.

## REFERENCES

- [1] F. Jafarzadehpour, et al., "New energy-efficient hybrid wide-operand adder architecture," in *IET Circuits, Devices & Systems*, vol. 13, no. 8, pp. 1221-1231, 2019.
- [2] C. Vudadha and M. B. Srinivas, "Design of High-Speed and Power-Efficient Ternary Prefix Adders Using CNFETs," in *IEEE Transactions on Nanotechnology*, vol. 17, no. 4, pp. 772-782, 2018.
- [3] R. Katreepalli and T. Haniotakis, "High Speed Power Efficient Carry Select Adder Design," *2017 IEEE Computer Society Annual Symposium on VLSI (ISVLSI)*, Bochum, 2017, pp. 32-37.
- [4] J. Kandpal, A. Tomar, M. Agarwal and K. K. Sharma, "High-Speed Hybrid-Logic Full Adder Using High-Performance 10-T XOR–XNOR Cell," in *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 28, no. 6, pp. 1413-1422, 2020.
- [5] R. H. V. and S. Hiremath, "Low Power Design and Implementation of Multi-Output Carry Look-Ahead Adder," *2018 3rd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT)*, Bangalore, India, 2018, pp. 1912-1916.
- [6] P. Bhattacharyya, B. Kundu, S. Ghosh, V. Kumar and A. Dandapat, "Performance Analysis of a Low-Power High-Speed Hybrid 1-bit Full Adder Circuit," in *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 23, no. 10, pp. 2001-2008, 2015.
- [7] A. Raghunandan and H. V. R. Aradhya, "Area and Timing Analysis of Advanced Adders under changing Technologies," *2019 4th International Conference on Recent Trends on Electronics, Information, Communication & Technology (RTEICT)*, Bangalore, India, 2019, pp. 33-38.
- [8] P. J. Edavoor and A. D. Rahulkar, "Design and implementation of a novel low complexity symmetric orthogonal wavelet filter-bank," in *IET Image Processing*, vol. 13, no. 5, pp. 785-793, 2019.
- [9] P. Patali and S. T. Kassim, "High throughput FIR filter architectures using retiming and modified CSLA based adders," in *IET Circuits, Devices & Systems*, vol. 13, no. 7, pp. 1007-1017, 2019.
- [10] A. Liacha, et al., "Design of high-speed, low-power, and area-efficient FIR filters," in *IET Circuits, Devices & Systems*, vol. 12, no. 1, pp. 1-11, 2018.
- [11] R. Mahalakshmi and T. Sasilatha, "An Improved Digital FIR Filter Design Using Fast FIR Algorithm and Modified Carry Save Addition," *National Academy Science Letters*, vol. 41, no. 3, pp. 147-150, 2018.
- [12] Z. Huang, et al., "A 16-bit Hybrid ADC with Circular-Adder-Based Counting for 15 $\mu$ m Pitch 640 $\times$ 512 LWIR FPAs," in *Chinese Journal of Electronics*, vol. 29, no. 2, pp. 291-296, 2020.
- [13] M. Jeon, et al., "A Stochastic Flash Analog-to-Digital Converter Linearized by Reference Swapping," in *IEEE Access*, vol. 5, pp. 23046-23051, 2017.
- [14] G. Liu, et al., "A Carry Lookahead Adder Based on Hybrid CMOS-Memristor Logic Circuit," in *IEEE Access*, vol. 7, pp. 43691-43696, 2019.
- [15] N. Kito and N. Takagi, "Concurrent Error Detectable Carry Select Adder with Easy Testability," in *IEEE Transactions on Computers*, vol. 68, no. 7, pp. 1105-1110, 2019.
- [16] A. A. Ananthakrishnan, et al., "FPGA Based Performance Comparison of Different Basic Adder Topologies with Parallel Processing Adder," *2019 3rd International conference on Electronics, Communication and Aerospace Technology (ICECA)*, Coimbatore, India, 2019, pp. 87-92.
- [17] P. Balasubramanian, "Asynchronous carry select adders," *Engineering Science and Technology, an International Journal*, vol. 20, no. 3, pp. 1066-1074, 2017.
- [18] B. Koyada, et al., "A comparative study on adders," *2017 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET)*, Chennai, 2017, pp. 2226-2230.
- [19] P. Patali and S. T. Kassim, "An Efficient Architecture for Signed Carry Save Multiplication," in *IEEE Letters of the Computer Society*, vol. 3, no. 1, pp. 9-12, 2020.
- [20] S. Nagaraj, B. Vamsi Krishna, Ganekanti Naresh and P. K. Anand Prem, "Design and Analysis of 8-bit Array, Carry Save Array, Braun, Wallace Tree and Vedic Multipliers," *International Journal of Psychosocial Rehabilitation*, vol. 24, no. 3, pp. 2687-2697, 2020.
- [21] D. Esposito, et al., "On the use of approximate adders in carry-save multiplier-accumulators," *2017 IEEE International Symposium on Circuits and Systems (ISCAS)*, Baltimore, MD, 2017, pp. 1-4.
- [22] A. A. D. Barrio, et al., "A Partial Carry-Save On-the-Fly Correction Multispeculative Multiplier," in *IEEE Transactions on Computers*, vol. 65, no. 11, pp. 3251-3264, 2016.
- [23] S. Jia, et al., "Simplified carry save adder-based array multiplier scheme and circuits design," *International Journal of Circuit Theory and Applications*, vol. 43, no. 9, pp. 1226-1234, 2014.
- [24] R. A. Javali, et al., "Design of high speed carry save adder using carry lookahead adder," *International Conference on Circuits, Communication, Control and Computing*, Bangalore, 2014, pp. 33-36.
- [25] A. K. Vamsi, N. U. Kumar, K. B. Sindhuri and G. S. C. Teja, "A Systematic Delay and Power Dominant Carry Save Adder Design," *2018 International Conference on Smart Systems and Inventive Technology (ICSSIT)*, Tirunelveli, India, 2018, pp. 359-362.

**BIOGRAPHIES OF AUTHORS**

**Ahmed Salah Hameed** received the B.Sc. degree in Computer and Software Engineering from the University of Diyala/College of Engineering [Iraq, Diyala] in 2008 and the M.Sc. degree in Computer Engineering from the University of Bridgeport/[USA, Bridgeport, CT] in 2013. He is currently lecturer at the Department of Computer Engineering/College of Engineering, University of Diyala. He has 12 years experience in practice of Computer Engineering. He is a member in Upsilon Pi Epsilon Honor Society (UPI), USA since May 2013. His research interests include VLSI Design, VHDL Design, Digital Systems Design, Computer Architecture, Parallel Processing, Computer Networks, and Signal Processing.



**Marwa Jawad Kathem** received the B.Sc. degree in Computer and Software Engineering from the University of Diyala/College of Engineering [Iraq, Diyala] in 2008. Her research interests include Computer Networks, Databases Programming, Computer Architecture, Digital Systems Design, Parallel Processing, and Signal Processing.