IOSR Journal of VLSI and Signal Processing (IOSR-JVSP)
Volume 4, Issue 2, Ver. IV (Mar-Apr. 2014), PP 69-74
e-ISSN: 2319 – 4200, p-ISSN No. : 2319 – 4197
www.iosrjournals.org

Performance Analysis of Different Multipliers using Square Root Carry Select Adders

[1]Damarla Paradhasaradhi, [2]K. Anusudha
[1]*M. Tech, Department of Electronics Engineering Pondicherry University Pondicherry, India*
[2]*Assistant Professor, Department of Electronics Engineering Pondicherry University Pondicherry, India*

***Abstract*:** A high-speed and energy-efficient multiplier is always required in electronics industry particularly in digital signal processing, arithmetic units in microprocessor and image processing. Multiplier is a significant element which contributes to the total power utilization of the system. By comparing different types of adders it is found that the ripple carry adder has a smaller area with lesser speed performance, in contrast to which carry select adders have high speed but posses a larger area. In the existing models of the multipliers, the regular square root carry select adder and modified square root carry select adder using Binary to Excess-1 logic are designed and implemented on both Array and Wallace tree multipliers respectively. In the proposed work a proficient square root carry select adder is designed using common Boolean logic and is implemented on both Array and Wallace tree multipliers respectively. A well-organized Verilog code has been written and successfully synthesized and simulated using Xilinx ISE 14.2. The simulation results gives the performance of Array and Wallace tree multipliers using proposed square root carry select adder is excellent compared with other structures of square root carry select adders.

Keywords: Array Multiplier, Wallace Tree Multiplier, Square Root Carry Select Adder, Binary to Excess-1 Converter (BEC), Common Boolean Logic (CBL).

I. INTRODUCTION

In recent power utilization as well as area and speed are the most important issues in VLSI design. As the VLSI field is moving towards more compact design with higher performance and the need to develop the components which are attuned with the advancement becomes mandatory. Multiplication is one of the most significant operations in every computational system. Multiplier is the logic device of great concern in terms of performance of a processor also multiplier is one of the key hardware blocks in most digital and high performance systems such as digital signal processor, FIR filters, microprocessors etc. In any structure, the task of processor is very critical. The task that takes most of the processors time is multiplication thus enhancing the performance of multiplier leads to better performance of processor especially in field of digital signal processing and data processing ASIC [8]. The multiplier is an essential element of the digital signal processing such as filtering and convolution. The multiplier is also an important element in microprocessor. The demand of fast processors is increasing for high-speed data processing. Since the multiplier requires the longest delay among the basic operational blocks in digital system. Any multiplier can be divided into three stages: Partial products generation stage these are generated by AND operation, partial products addition stage can be carried by different adders, and the final addition stage. Many high-performance algorithms and architectures have been proposed to speed up multiplication. The speed of multiplication can be increased by reducing the number of partial products. Various multiplication algorithms like Booth, Modified Booth, Braun, and Baugh-Wooley have been proposed [11]. Generally Ripple Carry Adder (RCAs) has the most compact design among all types of adders. The Carry select adder is used in many computational systems to improve the problem of carry propagation delay by independent generation multiple carries and then select a carry to generate the sum. whereas, the carry select adder is not area efficient because it uses multiple pairs of Ripple Carry Adders(RCA) to generate partial sum and carry by considering carry input as C_{in}=0 and C_{in}=1 [1]. This paper work presents two different structures of multipliers using different adder circuits namely Regular square root carry select adder (RCSLA), Modified square root carry select adder (MCSLA) and proposed square root carry select adders (PCSLA) respectively.

The basic idea of the proposed architecture is that the RCA is replaced by Common Boolean Logic. The proposed Square root carry select adder uses Common Boolean logic term instead of RCA with C_{in}=1 in the regular carry select adder to achieve lower area and lower delay. The proposed architecture generates a duplicate sum and carry-out signal by using NOT and OR gate and select value with the help of multiplexer. By using the multiplexer, select the correct output according to its previously carry out signal. Thus it can be interpreted from this fact that addition is a sub-process in multiplication criterion that has to be fulfilled [1]. After developing these two different forms of Wallace tree multiplier a comparative study is being carried out on the basis of area and delay consumption by the two designs.

This paper is organized as follows: Section II describes the Multipliers structures using regular square root carry select adder, section III explain the Multipliers structures using modified square root carry select adder respectively. A section IV deals with Multipliers structures using proposed square root carry select adder. The simulation results are analyzed in the section V. and finally Section VI with the conclusion.

II. MULTIPLIERS STRCUTERS USING REGULAR SQUARE ROOT CARRY SELECT ADDDER

a. Array Multiplier using RCSLA:

This is one of the simplest techniques for implementing multiplication. This idea is to add all the N partial products sequentially using N-1 adders. In general, the basic square root carry select adder has a dual ripple carry adder with 2:1 multiplexer, the main disadvantage of regular square root carry select adder (RCSLA) is the large area due to the multiple pairs of ripple carry adder. The regular 16-bit square root carry select adder is shown in Figure 1. This structure is divided into five groups with different bit sizes of ripple carry adders [1]. From the structure of RCSLA, there is scope for reducing delay and area consumption. The carry out is calculated from the last stage; in this the selection is done by using a multiplexer.

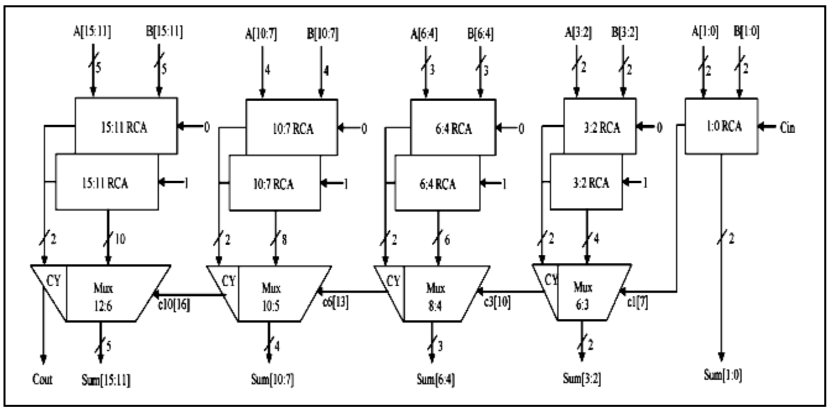
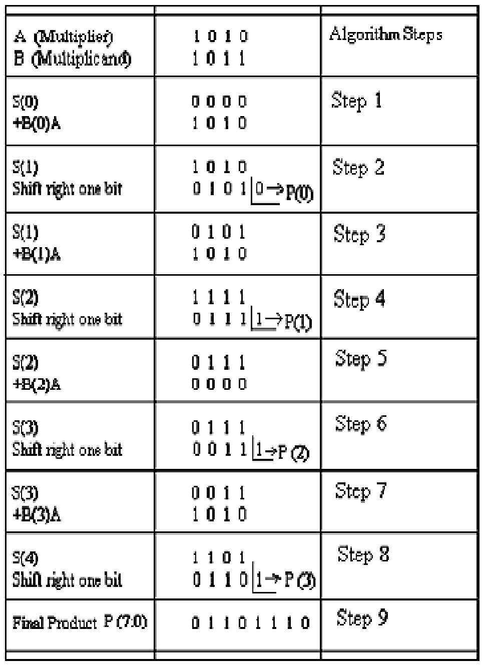


Figure 1. Regular 16-bit square root carry select adder

Figure 2. gives the algorithm steps and shows the complete multiplication process with an example. In the array algorithm, S (i) represents sum of each product term, (B (i), A) represent each product term and P (i) represents individual bit- term of final product. In case of an array multiplier with RCSLA, all partial product additions as well as final addition are carried out by using regular square root carry select adder [4]. The array multiplier of 4-bits using Regular square root carry select adder is shown in Figure 3.



A (Multiplier) B (Multiplicand)	1 0 1 0 1 0 1 1	Algorithm Steps
S(0) +B(0)A	0 0 0 0 1 0 1 0	Step 1
S(1) Shift right one bit	1 0 1 0 0 1 0 1 | 0 → P(0)	Step 2
S(1) +B(1)A	0 1 0 1 1 0 1 0	Step 3
S(2) Shift right one bit	1 1 1 1 0 1 1 1 | 1 → P(1)	Step 4
S(2) +B(2)A	0 1 1 1 0 0 0 0	Step 5
S(3) Shift right one bit	0 1 1 1 0 0 1 1 | 1 → P (2)	Step 6
S(3) +B(3)A	0 0 1 1 1 0 1 0	Step 7
S(4) Shift right one bit	1 1 0 1 0 1 1 0 | 1 → P (3)	Step 8
Final Product P (7:0)	0 1 1 0 1 1 1 0	Step 9

Figure 2. Algorithm for Array Multiplier

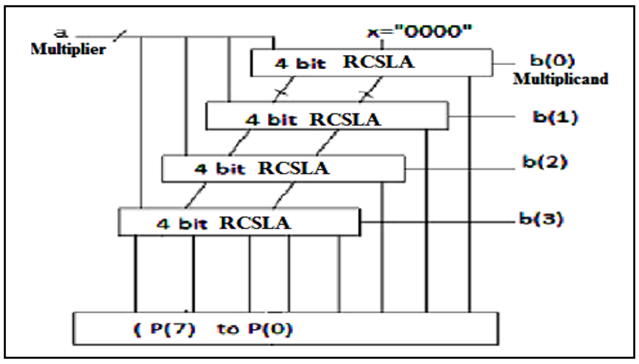


Figure 3. Array Multiplier using RCSLA

b. Wallace Tree Multiplier using RCSLA:

The Wallace tree multiplier is considerably faster than a simple array multiplier, because array multiplier height is logarithmic in word size. The summing of the partial product bits in parallel using a tree of carry-save adders is generally known as the "Wallace Tree Multiplier" [3]. To multiply any two numbers these are the main steps:

- Formation of partial products.
- Reduction of the partial products matrix into a two row matrix by means of a carry save adder.
- Addition of remaining two rows using a faster Carry Look Ahead Adder (CLA).

The Wallace tree multiplier of 4-bits using regular square root carry select adder is shown in Figure 4.

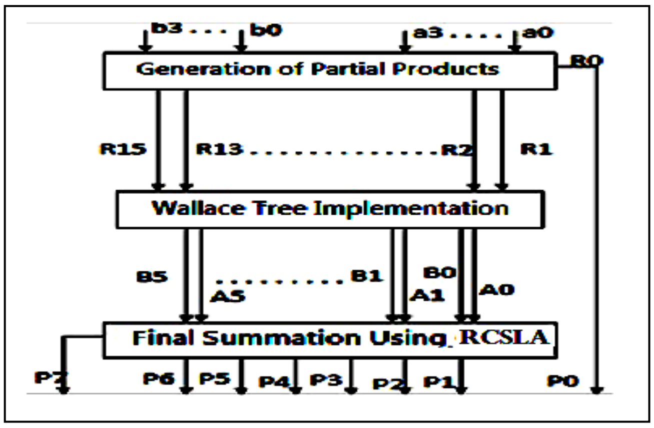

b3 . . . b0
a3 . . . . a0
Generation of Partial Products
R0
R15
R13 . . . . R2
R1
Wallace Tree Implementation
B5
. . . . B1
A5
B0
A1
A0
Final Summation Using RCSLA
P7
P6
P5
P4
P3
P2
P1
P0

Figure 4. Wallace Tree Multiplier using RCSLA

III. MULTIPLIERS STRUCUTERS USING MODIFIED SQUARE ROOT CARRY SELECT ADDER

a. Array Multiplier using MCSLA:

The main idea of this work is to use Binary to Excess one Converter (BEC) instead of the RCA with $C_{in}=1$ in order to reduce the delay and area utilization of the regular square root carry select adder. To replace the n-bit RCA, an n+1 bit BEC is required. The modified 16-bit square root carry select adder using BEC structure is shown in Figure 5. In this, the parallel ripple carry adder with $C_{in}=1$ is replaced with BEC logic. One input to the multiplexer goes from the RCA with $C_{in}=0$ and other input from BEC [1]. By comparing the individual groups of both regular and modified Square root carry select adder, it is inferred that the BEC structure reduces delay.

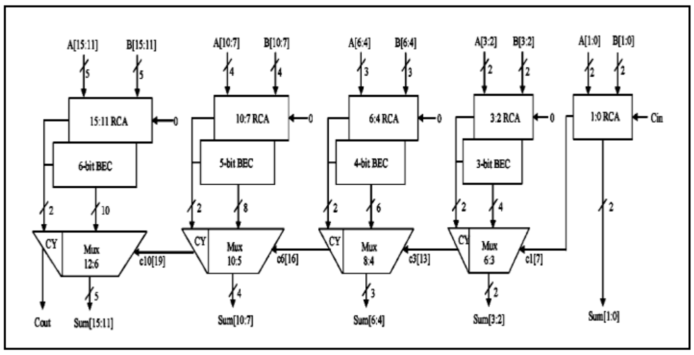


Figure 5. Modified 16-bit square root carry select adder

Similarly in case of multiplier with MCSLA, all partial product additions as well as final addition is carried out by using Modified square root carry select adder [5]. The array multiplier of 4-bits using Modified square root carry select adder is shown in Figure 6.

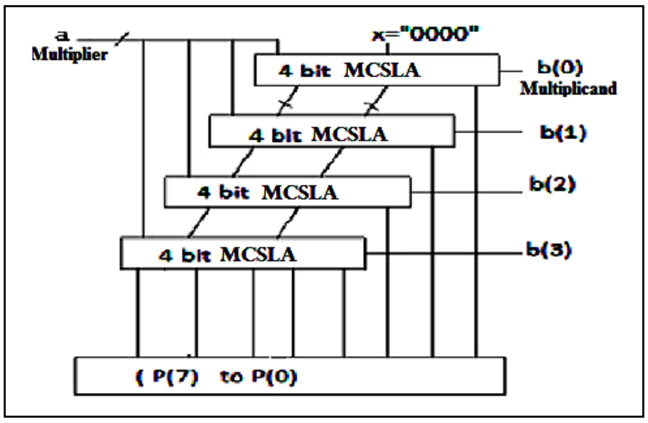


Figure 6. Array Multiplier using MCSLA

b. Wallace Tree Multiplier using MCSLA:

Similarly the Wallace tree multiplier of 4-bits using Modified square root carry select adder (MCSLA) is shown in Figure 7.

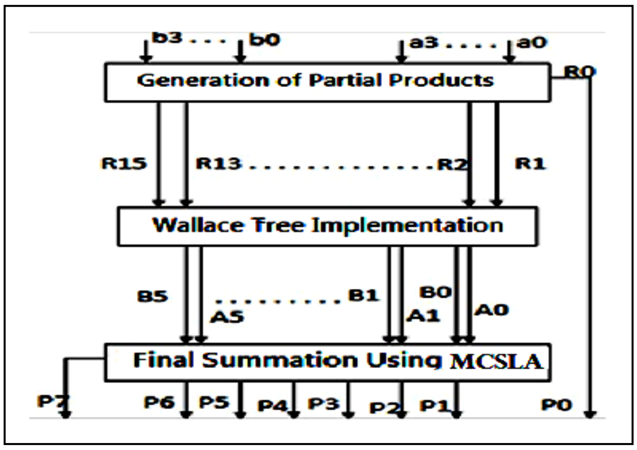

b3 . . . b0
a3 . . . . a0
Generation of Partial Products
R0
R15
R13 . . . . R2
R1
Wallace Tree Implementation
B5
. . . . B1
A5
B0
A1
A0
Final Summation Using MCSLA
P7
P6
P5
P4
P3
P2
P1
P0

Figure 7. Wallace tree multiplier using MCSLA

IV. Multipliers Strucuters using Proposed Square Root Carry Select Adder

a. Wallace Tree Multiplier using PCSLA:

This method replaces the BEC add one circuit by Common Boolean Logic. The proposed 16-bit Square root carry select architecture is shown in Figure 8. The summation and carry signal for full adder which has C_{in}=1, generate by INV and OR gate. Through the multiplexer, the final correct output is selected according to the logic state of carry-in signal. One input to the mux goes from ripple carry adder block with C_{in}=0 and other input from the Common Boolean logic (CBL). The CBL block has a 4:2 multiplexer to select the appropriate carryout and summation signal for carry-in signal "1". Through 2:1 multiplexer the carry signal is propagate to the next adder cell.

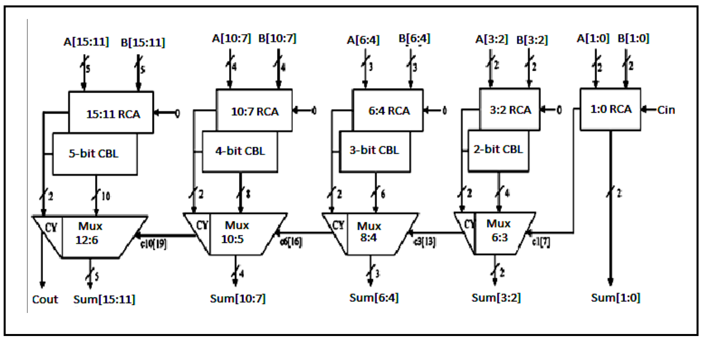


Figure 8. Proposed 16-bit square root carry select adder

Similarly in case of multiplier with PCSLA, all partial product additions as well as final addition is carried out by using proposed square root carry select adder. The 4-bits array multiplier using proposed square root carry select adder is shown in Figure 9.

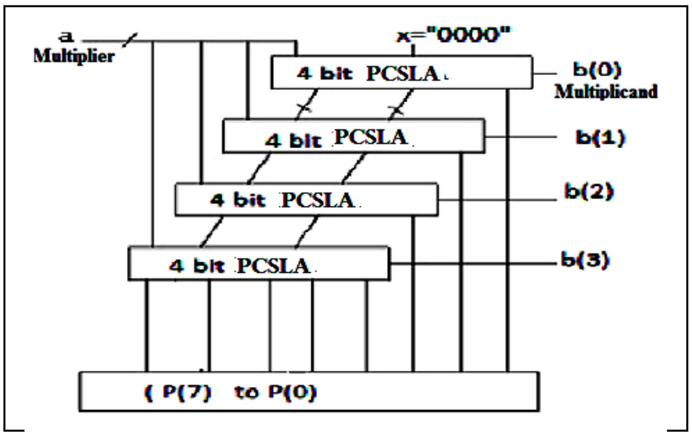


Figure 9. Array Multiplier using PCSLA

b. Wallace Tree Multiplier using PCSLA:

Similarly the Wallace tree multiplier of 4-bits using proposed square root carry select adder (PCSLA) is shown in Figure 10.

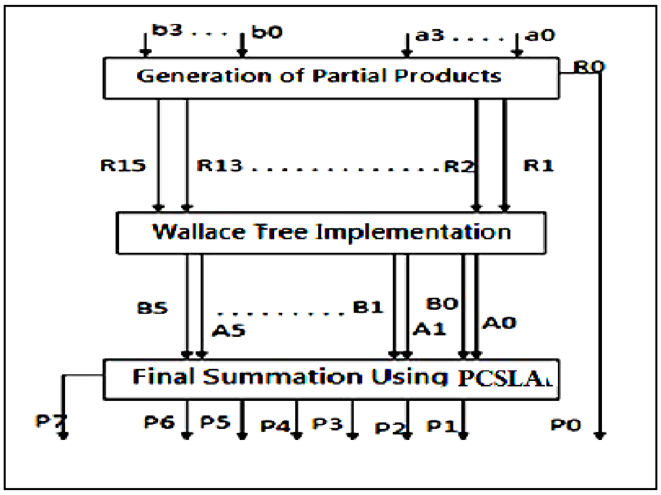

b3 . . . b0
a3 . . . . a0
Generation of Partial Products
R0
R15
R13 . . . . R2
R1
Wallace Tree Implementation
B5
. . . . B1
A5
B0
A1
A0
Final Summation Using PCSLA
P7
P6
P5
P4
P3
P2
P1
P0

Figure 10. Wallace Tree Multipier using PCSLA

V. RESULT SUMMARY

The comparison table for the 8-bit Array Multiplier using Regular square root carry select adder, Modified square root carry select adder and proposed square root carry select adders are shown in Table 1. Likewise for Wallace tree multiplier using different square root carry select adder's topologies comparison is shown in Table 2. Similarly for the 16-bit Array and Wallace tree multipliers using different square root carry select adder's topologies comparison is shown in Table 3 and Table 4 respectively. The total simulation done in Xilinx ISE 14.2 targeted Spartan 3E device. From the tables it is observed that the numbers of slices are reduced for proposed square root carry select adder based multiplier structure. So the proposed array and Wallace tree multipliers have an advantage in reducing area.

The performance analysis of Array and Wallace tree multipliers using different architectures of square root carry select adder are distinguished in following tables.

TABLE I. 8-BIT ARRAY MULTIPLIER WITH DIFFERENT ADDER TOPOLIGES

Array Multiplier	Delay (ns)	No. of Slices	Logic Levels
Using RCSLA	18.16	74	22
Using MCSLA	25.02	135	34
Using PCSLA	17.76	72	20

TABLE II. 8-BIT WALLACE TREE MULTIPLIER WITH DIFFERENT ADDER TOPOLIGES

Wallace Multiplier	**Delay (ns)**	**No. of Slices**	**Logic Levels**
Using RCSLA	36.73	97	31
Using MCSLA	23.64	102	32
Using PCSLA	22.42	91	29

TABLE III. 16-BIT ARRAY MULTIPLIER WITH DIFFERENT ADDER TOPOLIGES

Array Multiplier	**Delay (ns)**	**No. of Slices**	**Logic Levels**
Using RCSLA	25.36	94	40
Using MCSLA	38.05	155	52
Using PCSLA	20.13	82	34

TABLE IV. 16-BIT WALLACE TREE MULTIPLIER WITH DIFFERENT ADDER TOPOLIGES

Wallace Multiplier	**Delay (ns)**	**No. of Slices**	**Logic Levels**
Using RCSLA	47.62	127	43
Using MCSLA	38.47	138	50
Using PCSLA	34.73	115	37

VI. CONCLUSION

The performance analysis of the Array Multiplier and Wallace Tree Multiplier architectures designed in this paper work on the basis of the area and delay reveals the fact that the Array and Wallace Tree Multipliers using proposed square root carry select adder is better than the regular and modified square root carry select adder structures in terms of both area and delay shown in comparison tables. It would be fascinating to test the design of the 32-bit and 64-bits of Array and Wallace tree Multipliers.

REFERENCES

[1] B. Ramkumar and Harish M Kittur, "Low power and area efficient carry select adder", IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 20, no. 2, pp. 371-375, Feb 2012.

[2] Damarla Paradhasaradhi and K. Anusudha, "An Area Efficient Enhanced SQRT Carry Select Adder", International Journal of Engineering Research and Applications, vol. 3, Issue 6, Nov-Dec 2013.

[3] Jasbir Kaur and Kavita, " Structural VHDL Implementation of Wallace Multiplier", International Journal of Scientific & Engineering Research, vol. 4, Issue. 4, pp. 1829-1833, April 2013.

[4] N. Sureka, R. Porselvi and K. Kumuthapriya, "An efficient high speed Wallace tree Multiplier", Proceedings of IEEE International Conference on Information Communication and Embedded Systems, pp. 1023-1026, Feb 2013.

[5] Naveen K Gahlan, Prabhat, Jasbir Kaur " Implementation of Wallace Tree Multiplier Using Compressor" International Journal of Computer & Technology.

[6] W.J. Townsend, E.E. Swartzlander Jr., and J.A. Abraham, "A Comparison of Dadda and Wallace Multiplier Delays," Proc. SPIE, Advanced Signal Processing Algorithms, Architectures, and Implementations XIII, pp. 552-560, 2003.

[7] B. Ramkumar , Harish M Kittur and P. M. Kannan, "ASIC implementation of modified faster carry save adder", Eur. J. Sci. Res. , vol. 42, no. 1, pp. 53-58, Jun 2010.

[8] P. Sreenivasulu, K. Srinivasa rao, Malla Reddy and A. Vinay Babu, "Energy and area efficient carry select adder on a reconfigurable hardwware", International Journal of Engineering Research and Applicaions, vol. 2, Issue. 2, pp. 436-440, Mar 2012.

[9] R. Priya and J. Senthilkumar, "Implementation and comparision of effective area efficient architecuture for CSLA", Proceedings of IEEE International Conference on Emerging trends in Computing, Communicaton and Nano Technology, pp. 287-292, 2013.

[10] I-Chyn Wey, Cheng-Chen Ho, Yi-Sheng Lin and Chien-Chang Peng, "An area efficient carry select adder design by sharing the common boolean logic term", Proceedings on the International Multiconference of Engineering and computer scientist, IMECS 2012.

[11] T. Y. Ceiang and M. J. Hsiao, "Carry select adder using single ripple carry adder", Electron. Lett, vol. 34, no. 22, pp. 2101-2103, Oct 1998.

[12] J. M. Rabaey, *Digital Integrated Circuits-A Design Perspective*, Upper Saddle River, NJ: Prentice-Hall, 2001.

[13] Y. Kim and L. S. Kim, "64-bit carry select adder with reduced area", Electron. Lett. Vol. 37, no. 10, pp. 614-615, May 2001.

[14] Edison A. J and C. S. Manikanda babu, "An efficient CSLA architecture for VLSI hardware implementation", Interanational Journal for Mechanical and Industrial Engineering, vol. 2, Issue 5, 2012.