

# An Efficient Area and Delay Reduction for Architecture of 256 bit Carry Select Adder

**PRASANTHI BOYINA**

Assistant Professor in E.C.E  
Vishnu Institute of Engg&Tech, Bhimavaram,A.P.India.  
nissi5711@gmail.com

## Abstract

A novel approach of an efficient area and delay reduction of 256 -Bit Carry Select Adder Architecture for Low Power Application is designed. Technology is shrinking from nanometer to micrometer technology in order to meet these design issues. Adder is one of the designs which are used in many applications like DSP's, ALU's, subtractions and high speed multiplications. In this approach one of the fastest adders is Carry Select Adder (CSLA) is used . CSLA is one of the adder designs which improve the speed of addition operation to a great extent when compared to traditional adder designs. This increased speed and less area application design can be implemented by modifying the Regular adder design by replacing some of the gates with the implemented complex BEC logic which leads to Modified Carry Select Adder design. Based on this design idea 16, 32, 64, 128 and 256-bit Modified CSLA design have been developed and compared with Regular adder design. This Comparison estimates the performance of the proposed designs with the regular designs in terms of delay, area and synthesis are implemented in Xilinx 12.1. The results analysis shows that the proposed SQRT CSLA structure is efficient one compared with the regular SQRT CSLA.

**Index Terms** - BEC, SQRT CSLA,RCA ,low delay, Efficient Area.

## INTRODUCTION

In electronics, an adder or summer is a digital circuit that performs addition of numbers. In many computers and other kinds of processor applications, adders are used not only in the arithmetic logic unit, but also in other parts of the processor, where they are used to calculate addresses, table indices, etc[2] . Although adders can be constructed for many numerical representations, such as binary-coded decimal or excess-3, the most common adders operate on binary numbers. In cases where two's complement or ones' complement is being used to represent negative numbers, it is trivial to modify an adder into a subtraction[1] .

In electronics, a carry-select adder is a particular way to implement an adder, which is a logic element that computes the  $(n+1)$ -bit sum of two  $n$ -bit numbers [3]-[5].Reduced area and high speed data path logic systems are the main areas of research in VLSI system design. High-speed addition and multiplication has always been a fundamental requirement of high-performance processors and systems. The addition speed is majorly depends upon carry propagation time which is propagated from the lower group bits to the next higher group [6]-[8]. The sum for each bit position in an elementary adder is generated sequentially only after the previous bit position has been summed and a carry propagated into the next position. The adder designs are of many types like (Ripple Carry Adder, Carry Look Ahead Adder, Carry Save Adder, Carry Skip Adder) which have its own advantages and disadvantages. In any adder the major speed limitation is due to the production of carries and many authors considered the addition problem. In order to reduce the carry propagation delay CSLA is developed which reduces the area and delay to some extent. To moderate the problem of carry propagation delay in many computational systems CSLA is used which generating multiple carries independently and then select a carry to generate the sum[1]. To generate the resultant sum. it uses independent ripple carry adders in ( $C_{in}=0$  and  $C_{in}=1$ ) Conditions. However, the

existing method that is Regular CSLA is not efficient in area and speed because of multiple pairs utilization of Ripple Carry Adders (RCA) to generate partial sum and carry by considering carry input. Multiplexers (MUX) is used to select the final sum and carry. Due to the use of two independent RCA the area will increase which leads an increase in delay. To reduce the area and delay the basic idea of the proposed work is to use n-bit binary to excess-1 code converters (BEC) to improve the speed of addition. When in RCA for  $C_{in}=1$  replaced then the logic of BEC is performed Which improves the speed and thus reduces the delay as well as area. The main reason to use BEC is that it is designed using the lesser number of logic gates .thus the number of gates used are decreased. So, area and delay decreases This work structured briefly is as follows. Section II deals with the delay and area evaluation methodology of the basic adder blocks and its corresponding delay and area values. Section III deals with the structure and function of BEC logic and its corresponding function table and logic equations. Section IV presents the architecture of the Regular CSLA of 128-bits. This SQR CSLA has been developed using ripple carry adders and multiplexers. The architecture of the Modified SQR CSLA is presented in Sections V. In section VI implementation methodologies and corresponding design tools are explained and finally the paper is concluded in section VIII.

## II THE EVALUATION OF AREA, DELAY OF THE BASIC ADDER BLOCKS

The adder block using a Ripple carry adder, BEC and MUX is explained in this section. In this we calculate and explain the delay & area using the theoretical approach and show how the delay and area effect the total implementation. The AND, OR, and Inverter (AOI) implementation of an XOR gate is shown in Fig 1. The delay and area evaluation methodology considers all gates to be made up of AND, OR, and Inverter, each having delay equal to 1 unit and area equal to 1 unit. The number of gates in the longest path of a logic block that contributes to the maximum delay are added up. The area evaluation is done by counting the total number of AOI gates required for each logic block. Based on this approach, the blocks of 2:1 MUX, Half Adder (HA), and FA are evaluated and listed in Table 1.

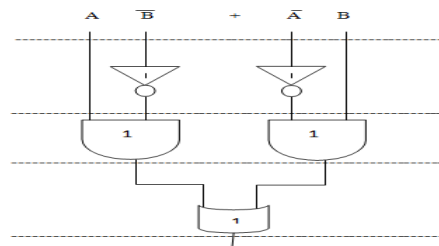


Fig 1. Delay and Area evaluation of Xor

Adder Blocks	Delay	Area
Xor	3	5
2:1 Mux	3	4
Half adder	3	6
Full adder	6	12

Table 1. Delay and Area Evaluation of basic blocks

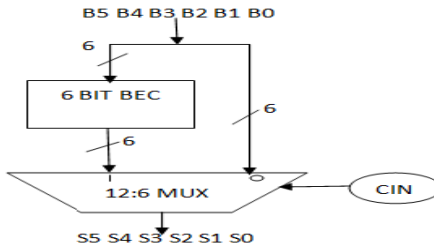


Fig2 : 6-bit BEC with 12:6 MUX

Fig2 shows the basic 6-bit addition operation which includes 6-bit data, a 6-bit BEC logic and 12:6 MUX. The addition operation is performed for  $Cin=0$  and for  $Cin=1$ . For  $Cin=0$  the addition is performed using ripple carry adder and for  $Cin=1$  the operation is performed using 6-bit BEC (replacing the RCA for  $Cin=1$ ). The resultant is selected based on Carry in signal from the previous group. The total delay depends on MUX delay and  $Cin$  signal from previous group.

### III 5-BIT BEC

Binary to Excess-1 Converter (BEC) in the regular CSLA to achieve lower area and increased speed of operation. This logic is replaced in RCA with  $Cin=1$ . This logic can be implemented for different bits which are used in the modified design. The main advantage of this BEC logic comes from the fact that it uses lesser number of logic gates than the n-bit Full Adder (FA) structure. As stated above the main idea of this work is to use BEC instead of the RCA with  $Cin=1$  in order to reduce the area and increase the speed of operation in the regular CSLA to obtain modified CSLA. To replace the n-bit RC, an (n+1) bit BEC logic is required. The structure and the function table of 5-bit BEC are shown in Fig.3 and Table2.

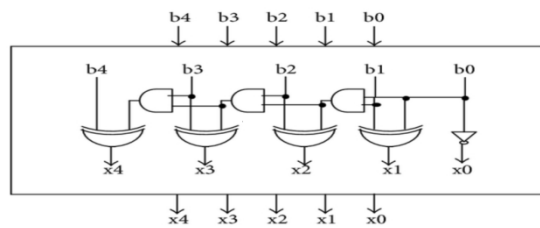


Fig 3. 5 bit binary to excess-1 converter

B[4:0]	X[4:0]
00000	00001
00001	00010
00010	00011
⋮	⋮
11111	00000

Table 2. Function table of 5 bit BEC

The Boolean expressions for the 5-bit BEC logic are expressed below

$$X0 = \sim B0$$

$$X1 = B0 \wedge B1$$

$$X2 = B2 \wedge (B0 \& B1)$$

$$X3 = B3 \wedge (B0 \& B1 \& B2)$$

$$X4 = B4 \wedge (B0 \& B1 \& B2 \& B3)$$

#### IV. ARCHITECTURE OF REGULAR 256-BIT Sqrt CSLA

A traditional carry select adder can be implemented in two different modes, namely uniform block mode and variable block mode. In uniform block mode the data bits are divided into groups which are of equal size for entire design, similarly in variable block mode the data bits are divided into different groups of variable sizes i.e., not equal in size. Among the two modes of design variable mode is highly recommended design, because the propagation time of carry in this mode takes less when compared with uniform design. The implementation architecture of Regular 256-bit Sqrt Carry Select Adder is shown in below figure. The design is first divided into different groups of variable sizes. Each group has some bit length of different sizes. These bit lengths are developed using two Ripple Carry Adder (RCA) and a multiplexer to each group except the Group 0, because it is having only one RCA in that group, so there is no need of multiplexer to select the data. Starting from the Group 0, it contains only one 2-bit RCA, which adds the data input bits supplied and the input carry which results a sum [1:0] and also the carry out. The carry out which is generated by Group 0 will act as a selection line for the next higher group in the series that is Group 1. Based on the selection line from Group 0, the multiplexer selects the corresponding upper RCA ( $C_{in}=0$ ) or lower RCA ( $C_{in}=1$ ). Similarly the pattern continues for the remaining groups depending on the  $C_{out}$ .

#### V. ARCHITECTURE OF MODIFIED 256-BIT Sqrt CSLA

The Modified 256-bit Sqrt CSLA Architecture design is obtained by replacing RCA of  $C_{in}=1$  in Regular architecture with complex BEC logic. The Major advantage of this BEC logic is that, it can perform the similar addition operation as that of the RCA with  $C_{in}=1$  in Regular architecture and the area and delay is reduced compared to regular design. Figure below shows the Modified block diagram of 256-bit Sqrt CSLA. Similar to the regular design the Modified method can be implemented using the variable size bit pattern. The number of bits required for BEC logic is 1 bit more than the RCA bits. The same architecture division is performed in the Modified method as that of Regular method in order to differentiate both the designs. Here the main difference in the design comes from the BEC logic design which is instantiated in the Regular design and the use of second multiplexer for carry bits in each group in order to get the carry bit output with less delay. Based on the design considerations Group 0 contain only one RCA, having input of lower significant bit and a carry in bit and produces sum [1:0] and carry out as an output bits, and multiplexer of higher group has selection input from the lower group carry output bit. The delay calculation of this Modified design is calculated as follows  $C1$  is the carry output from Group 0 has the delay of five gate delays which is acted as input to multiplexer. Based on this value it is clear that  $C1$

arrival time is earlier than the results computed by the RCA and BEC, so it has to wait until the results are computed, after the results are computed, C1 will select the corresponding results which leads to some delay zat the Group 0. For the remaining groups the arrival time of mux selection input time is later than the RCA and BEC results computation time and also a second set multiplexer selects the carry bit computation which leads to speed of operation. So, these steps progresses for higher order bits and computes results in less time.

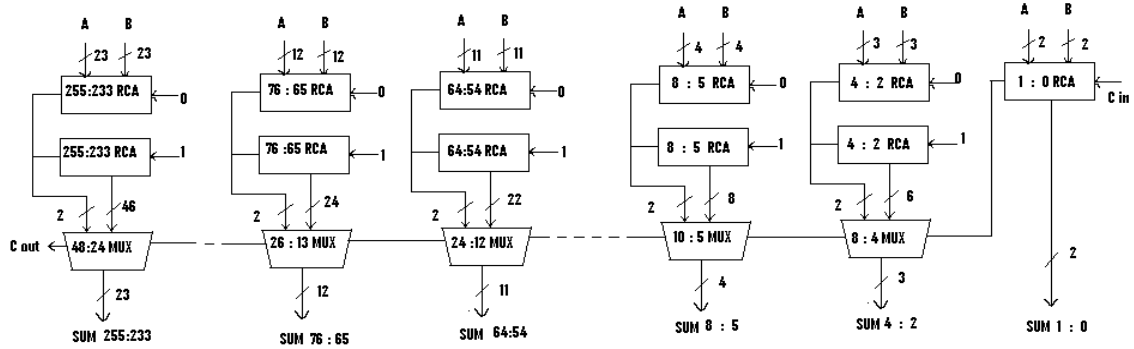


Fig 4:Architecture of Regular 256-bit Sqrt CSLA

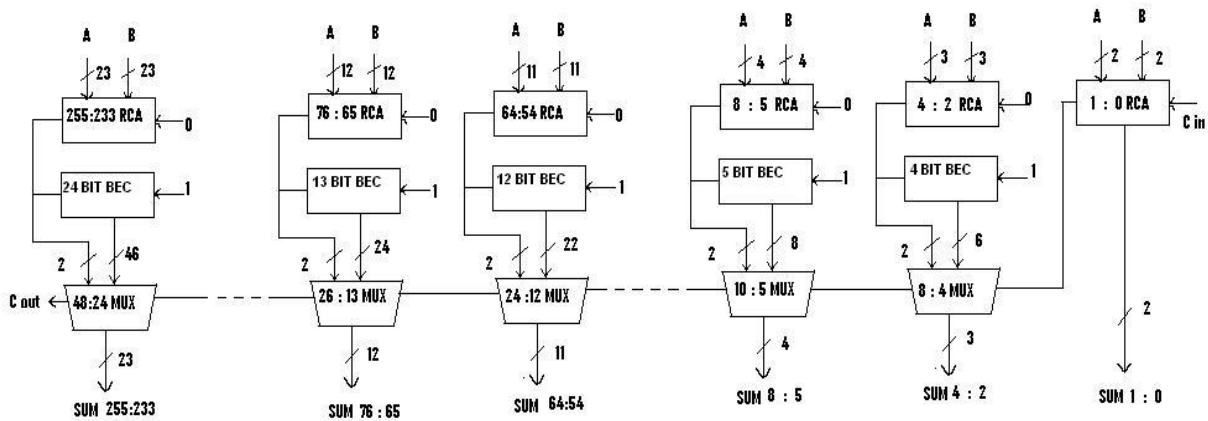


Fig 5: Architecture of Modified 256-bit Sqrt CSLA

Adder	Delay(ns)	Area(LUTs)
256-bit Regular Sqrt CSLA	112.889	896
256-bit Modified Sqrt CSLA	81.900	874

The bedside table is comparison of area and delay of regular and modified architecture in terms of LUT's .The Area and delay of the modified architecture is less compared to the regular architecture.

**RESULT:**

**GROUP AREA CALCULATIONS**

Group	Area	
	256 BIT	
	REGULAR	MODIFIED
GROUP2	57	41
GROUP 3	87	66
GROUP 4	117	89
GROUP 5	147	112
GROUP 6	169	135
GROUP 7	207	158
GROUP 8	237	181
GROUP 9	267	204
GROUP 10	297	227
GROUP 11	327	250
GROUP 12	357	273
GROUP 13	387	296
GROUP 14	417	319
GROUP 15	447	342
GROUP 16	477	365
GROUP 17	507	388
GROUP 18	537	411
GROUP 19	597	457
GROUP 20	627	480
GROUP 21	657	503
GROUP 22	687	526

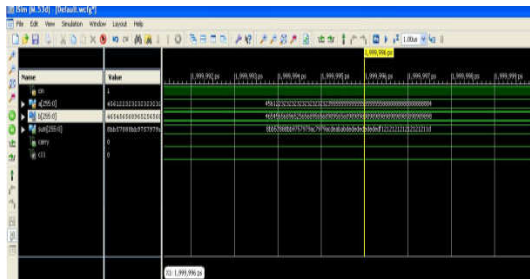


Fig 6: Regular 256-bit SQRT CSLA

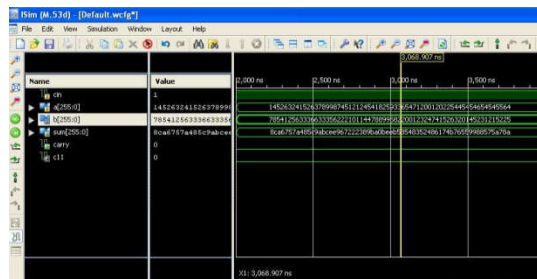


Fig 7: Modified 256-bit Sqrt CSLA

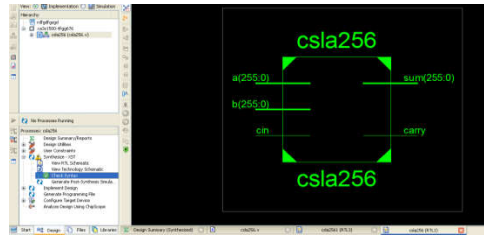


Fig 8: RTL Schematic of regular Sqrt CSLA

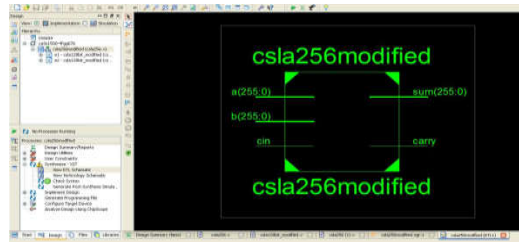


Fig 9: RTL Schematic of modified Sqrt CSLA

RTL schematic of regular 512 bit Sqrt CSLA Architecture of top module is synthesized by using Xilinx ISE 12.1i as shown in Fig 9, 10. In this a, b and Cin are the inputs of the architecture and sum, carry are the outputs of the architecture.

```
Device utilization summary:
-----
Selected Device : xa3s1500fgg676-4

Number of Slices:          480 out of 13312   3%
Number of 4 input LUTs:    896 out of 26624   3%
Number of IOs:             770
Number of bonded IOBs:     770 out of 487  158% (*)

WARNING: Xst:1336 - (*) More than 100% of Device resources are used
```

Fig 9: Area of the regular architecture.

```
Device utilization summary:
-----
Selected Device : xa3s1500fgg676-4

Number of Slices:          497 out of 13312   3%
Number of 4 input LUTs:    874 out of 26624   3%
Number of IOs:             770
Number of bonded IOBs:     770 out of 487  158% (*)

WARNING: Xst:1336 - (*) More than 100% of Device resources are used
```

Fig 10: Area of modified architecture

```
Timing Detail:
-----
All values displayed in nanoseconds (ns)

=====
Timing constraint: Default path analysis
Total number of paths / destination ports: 437039 / 257
-----
Delay:          98.930ns (Levels of Logic = 56)
Source:         b<0> (PAD)
Destination:    carry (PAD)
```

Fig 11: Delay of regular Architecture.

```

Timing Detail:
-----
All values displayed in nanoseconds (ns)

-----
Timing constraint: Default path analysis
Total number of paths / destination ports: 212541 / 257
-----
Delay:          112.889ns (Levels of Logic = 70)
Source:         h<2> (PAD)
Destination:    carry (PAD)

```

Fig 12: Delay of modified Architecture

## VII. CONCLUSION

This paper has really given a effective description of an higher bit high speed and area efficient carry select adder. This has been achieved by altering the logic blocks of the regular module, which intern helped us to have a new advantageous adder of higher bit than previous one. Many electronic applications are required of faster adders of higher number of bits which helps their process very faster. Replacing RCA with BEC and using a additional multiplexer for carry out which is decreasing the propagation delay. This paper can provide us for future scope like we can increase the number of bits of this adder and we can think of new ideas to decrease still more area of the device as it increases in its number of bits.

## REFERENCES

1. Bedrij, O. J., (1962), "Carry-select adder," IRE Trans. Electron. Comput. Pp.340–344.
2. Kim ,Y. and Kim ,L.-S.,(May2001), "64-bit carry-select adder with reduced area," Electron Lett., vol. 37, no. 10, pp. 614–615 Ceiang, T. Y. and Hsiao. J., (Oct 1998),"Carry-select adder using single ripple carry adder," Electron. Lett., vol. 34, no. 22, pp. 2101– 2103
3. He, Y., Chang, C. H. and Gu, J., (2005), "An Area efficient 64-bit square root carry-select adder for low power application," in Proc. IEEE Int. Symp.Circuits Syst. vol. 4, pp. 4082–4085
4. He, Y., Chang, C. H. and Gu, J., (2005), "An Area efficient 64-bit square root carry-select adder for low power application," in Proc. IEEE Int. Symp.Circuits Syst. vol. 4, pp. 4082–4085
5. E. Abu-Shama and M. Bayoumi, "A New cell for low power adders,"in P. roc.Int.Midwest Symp. Circuits and Systems, 1995, pp. 1014–1017
6. Verilog HDL-Premier, by j.Bhaskar