# DESIGN OF LOW POWER AND HIGH SPEED BEC 2248 EFFICIENT NOVEL CARRY SELECT ADDER

Sakshi Rajput[1], Gitanjali[2], Priya Sharma[2] and Garima[2]
[1]Assistant Professor, Department of Electronics and Communication Engineering,
Maharaja Surajmal Institute of Technology (GGSIPU), New Delhi, India
[2]Department of Electronics and Communication Engineering
Maharaja Surajmal Institute of Technology (Affiliated to GGSIPU), New Delhi, India

***ABSTRACT***

*Adders are one of the widely used digital components in digital integrated circuit design .Addition is the basic operation used in almost all computational systems. Therefore, the efficient implementation and design of arithmetic units requires the binary adder structures to be implemented in an equally efficient manner .A ripple carry adder has smaller area but less speed. A carry look ahead adder is faster though its area requirements are high. Carry select adders lie in between the spectrum. BEC 2248 Efficient Novel carry select adder proposed here provides good compromise between cost and performance thereby establishing a proper trade-off between time and area complexities. In this research Xilinx ISE Design Suite 12.1 is used for the comparison of all adders – Ripple carry adder, Bitwise carry select adder, Square root carry select adder, BEC 2248 Efficient Novel carry select adder.*

***KEYWORDS:*** *Carry select adder (CSA), Binary to excess-1 converter (BEC), Ripple Carry adder (RCA), Very large scale Integration (VLSI)*

## I. INTRODUCTION

In recent years, the increasing demand for high-speed arithmetic units in micro-processors, image processing units and DSP chips has paved the path for development of high-speed adders as addition is an indispensable operation in almost every arithmetic unit, also it acts as the basic building block for synthesis of all other arithmetic computations .To increase portability of systems and battery life, area and power are the critical factors of concern. Even in servers and personal computers (PC), power dissipation is an important design parameter. In today's scenario, Design of area-efficient and power-efficient high-speed logic systems are the one of the crucial areas of research in VLSI design.
In digital adders, the speed of addition is limited by the time required by the carry to propagate through the adder. In present scenario, where Computations need to be performed using low-power and an area-efficient circuit that must operate at greater speed which is achievable with lesser delay, efficient adder implementation becomes a necessity. Depending on the area, delay and power consumption requirements, several adder implementations have been proposed. Ripple Carry Adders with the most compact design (O (n) area) among all types of adders, are the slowest in speed (O (n) time). Carry Select Adders (O (n) time) and (O (2n) area) are in between RCAs and CLAs (O( n) time) and (O(n log n) area) thus providing an optimum solution between the area-efficient RCAs and the shortest-delay CLAs.

## II. RIPPLE CARRY ADDER

The most straight forward implementation of final stage adder is Ripple Carry Adder in which cascaded full adders are used. Carry generated in previous full adder works as input carry for next stage full adder. N bit Ripple Carry Adder requires N full adders as shown in Figure 1. It shows a n-bit ripple carry adder .It consists of n no. of 1-bit full adders. $y_i$ ,$x_i$ are the inputs to each full adder block that generates a sum , $s_i$ and a carry out, $c_i$ of stage where i=1 to n.

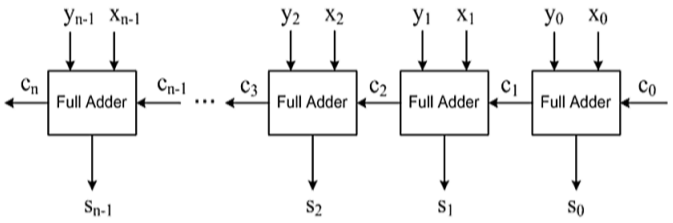


**Figure 1:** Block Diagram of n-bit Ripple Carry Adder

**Logic Equations**

$C_i = x_i \& y_i$ (1)
$C_i$=carry out of $i^{th}$ stage
$P_i = x_i \wedge y_i$ (2)
$P_i$= Partial sum
$S_i = P_i \wedge C_i$ (3)
$S_i$= Final Sum

**Drawbacks**

**1** It is not efficient when large numbers of bits are used.
**2** Carry propagation delay increases linearly with bit length as next stage output is dependent on previous stage output.

## III. BITWISE CARRY SELECT ADDER

The delay that the next stage encounters while waiting for previous carry and then performing the addition operation is reduced in bitwise carry select adder architecture. Addition of the two bits is performed taking both input carry possibility (Cin=0 and Cin=1) and selection is made on the basis of previous carry. Figure 2 shows the block diagram for bitwise CSA.
The n bit bitwise carry select adder consists of one n/2-bit adder for the lower half of the bits and two n/2-bit adders for the upper half of the bits. Out of the two adders one performs the addition assuming that Cin=0, whereas the other does this assuming that Cin=1.Using a multiplexer the value of carry out that is propagated from the adder for the n/2 LSB's , the correct value of the most significant part of the addition can be selected.

**Logic Equations**

**1** $Sa_i = a_i \wedge b_i$ output sum of ith stage block with input carry as '0'

**2** $Sb_i = \sim(a_i \wedge b_i)$ output sum of ith stage block with input carry as '0'

**3** $C_i = (a_i \& c_i)|(a_i \& b_i)|(b_i \& c_i)$ output carry for ith stage

**4** $S_i = c_i ? Sb_i : Sa_i$ final sum for ith stage

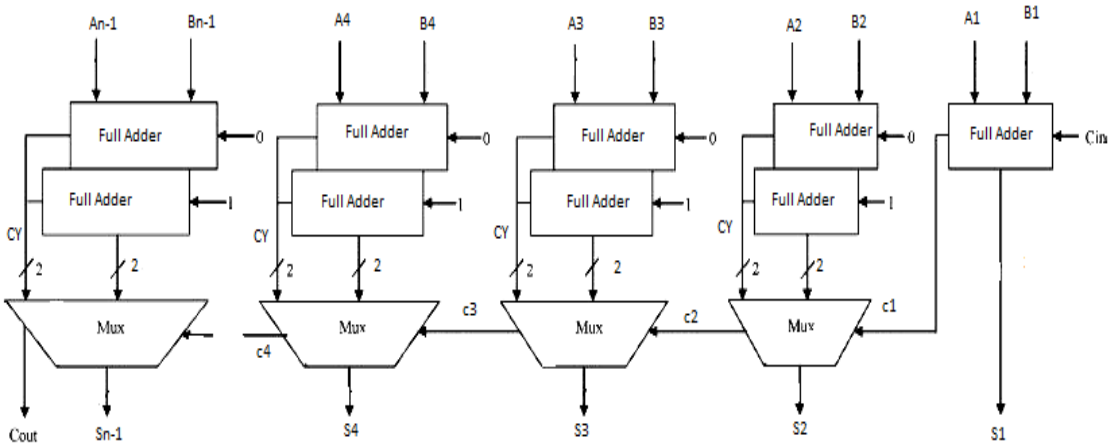


**Figure 2:** Block Diagram of Bitwise CSA

**Drawbacks:-**

**1** Larger area is required because of the multiplexers.
**2** Lesser delay than Ripple Carry Adders as carry propagation delay is reduced.
**3** Carry Select Adder is always preferred while working with smaller no of bits.

## IV. SQUARE ROOT CARRY SELECT ADDER

The strategy used for Square Root Carry Select adder is same as that of Bitwise Carry Select Adder(CSA) it also comprises of two blocks one with input carry as 1 and another block with input carry as 0. But the difference lies in the size of blocks used in it. In this, Ripple Carry Adders of more than 1 bit size are used. Figure 3 shows an example of a square root carry select adder (for n=16) i.e.16-bit square root CSA.

In a 16 –bit Square root carry select adder, ripple carry adders of more than 1-bit size are used. 2 bit, 2bit, 4 bit, 8 bit blocks of RCA have been used in this (right to left). At each stage we are getting different bits of sum, s [1:0], (1st stage), s [3:2] (2nd stage), s [7:4] (3rd stage), s [15:8] (final stage), Cout =final carry out.

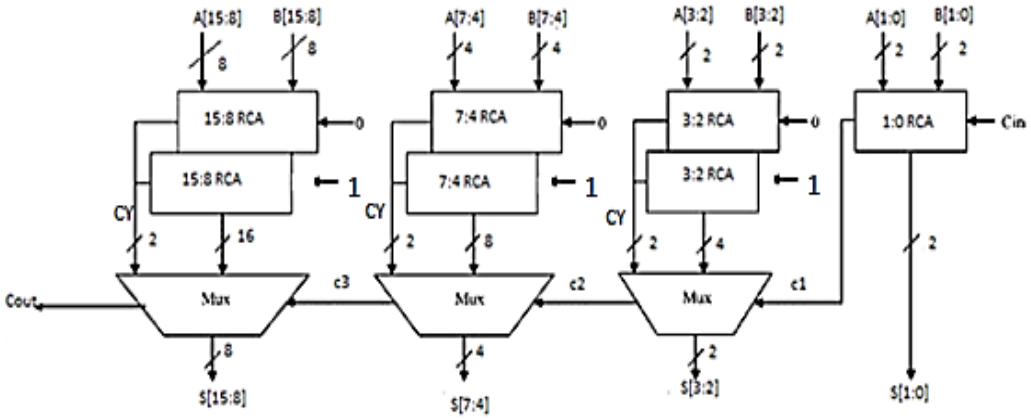


**Figure 3:** Block Diagram of Square root Carry Select Adder

## V. BEC 2248 EFFICIENT NOVEL CARRY SELECT ADDER

BEC 2248 Efficient Novel carry select adder works according to the following strategy. In this type of Adder, the block of Ripple Carry Adder with input carry as 1 has been replaced with a block of Binary to Excess-1 converter (BEC) as shown in Figure 4. For n-bit ripple carry adder, n+1 bit of Binary to Excess-1 converter is used. This is done in order to reduce the area and power requirement of the previous Carry Select Adder.

Figure 4 shows a Binary to excess-1 converter that replaces the RCA block with Cin=1 in conventional Carry select adder. It is a combination of NOT, AND, OR gates.CY= final Carry out. B2, B1, B0 are the inputs to BEC; X2, X1, X0 are the outputs.

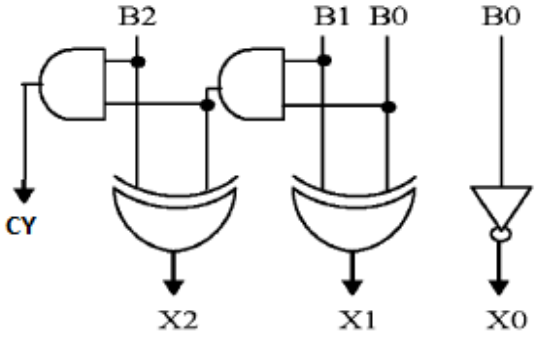


**Figure 4:** Block Diagram of BEC Converter

**Logic equations for Binary to Excess-1 Converter**:

**1** X0= ~B0 (4)

**2** X1=B1^B0 (5)

**3** X2=B2^(B1*B0) (6)

**4** CY=B0&B1&B2 (7)

Binary to excess-1 converter adds 1 to the binary input provided to it. It has less area requirement as compared to Ripple carry adder (RCA).

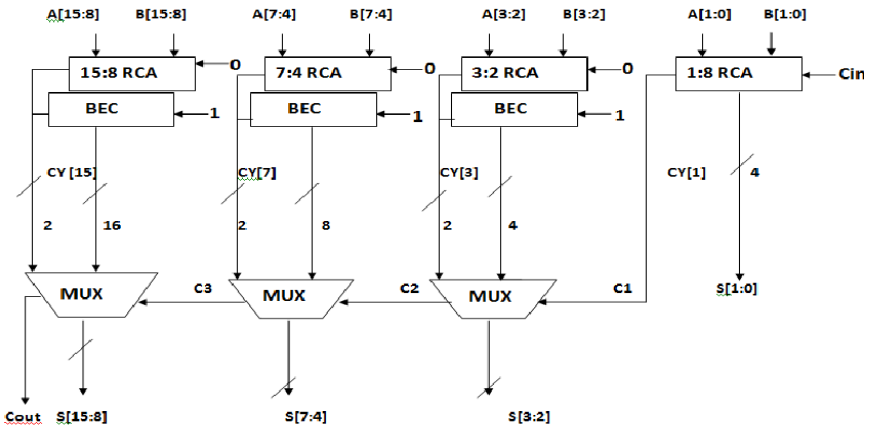


**Figure 5:** Block diagram of BEC 2248 efficient CSA

Figure 5 shows the architecture of proposed BEC 2248 Efficient carry select adder. It shows a (BEC) Efficient Novel CSA that replaces the RCA with input carry as '1' block in conventional square root CSA. This produces the two possible partial results in parallel and the multiplexer is used to select either the BEC output or the direct inputs according to the control signal 'C'$_i$. The LSB's are added using conventional RC. Once all the interim sums and carries are calculated, the final sums are computed using multiplexers having minimal delay. The multiplexer block receives the two sets of input and selects the final sum based on the select input from the previous stage.

The importance of the BEC logic emerges from the large silicon area reduction when CSA with large number of input bits are implemented in hardware. Use of BEC with multiplexer thus achieves faster incrementing action with reduced device count. Thus, the BEC 2248 Efficient Novel CSA excels the conventional CSA circuit in terms of area-efficient and power-efficient circuit.

## VI. SIMULATION AND IMPLEMENTATION RESULT

The simulation waveform (Figure 6) for BEC 2248 Efficient Novel 16-bit carry select adder (n=16) depicts the two 16 bit inputs **a[16:0],b[16:0]** the carry out-**cout** and the 16 bit sum **s[16:0].** The act of simulating something first requires that model to be developed; this model represents the key characteristics of the selected physical or abstract system or process.

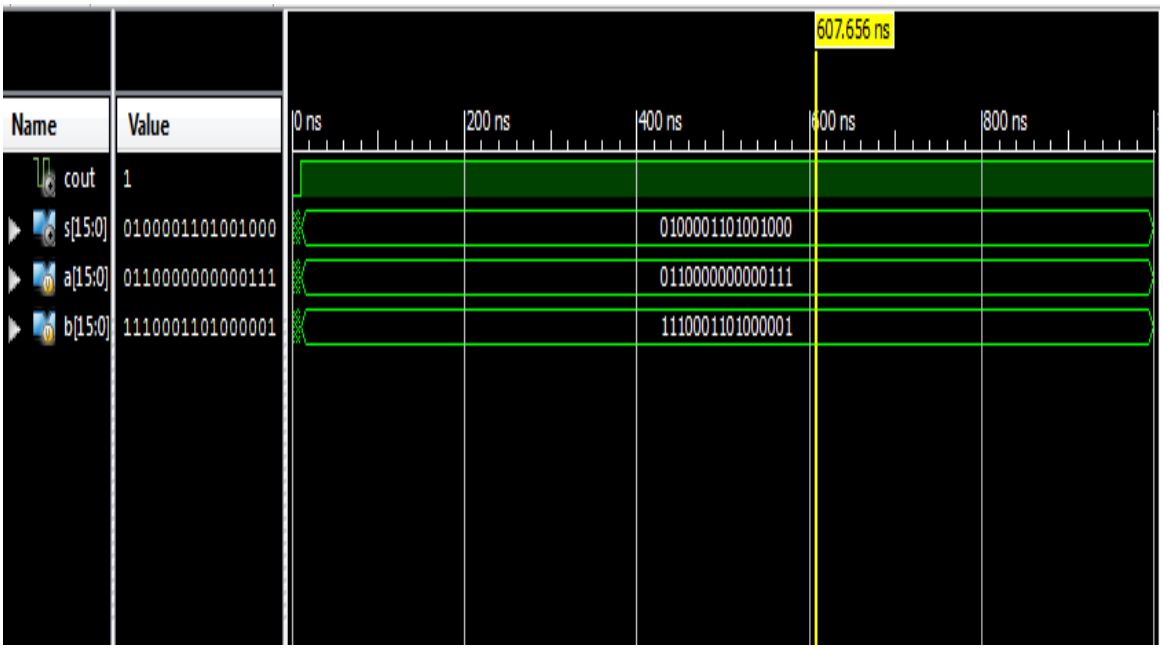


**Figure 6:** Simulation result of BEC based Adder

All the adders have been designed in Verilog HDL and synthesized in Xilinx ISE Design Suite 12.1. They have been compared in Figure 7, 8, 9 and 10 by considering the following factors: Power consumption, Combinational delay, Memory usage for synthesis, Peak Memory usage.
The power consumed is a sum of leakage and dynamic power. It has been proved from the graphs that BEC 2248 Efficient Novel carry select adder is more power efficient for VLSI Implementations.
Figure 7 depicts that least combinational delay (**8.303 ns**) is achieved in BEC 2248 Efficient carry select adder comparison to other adders.

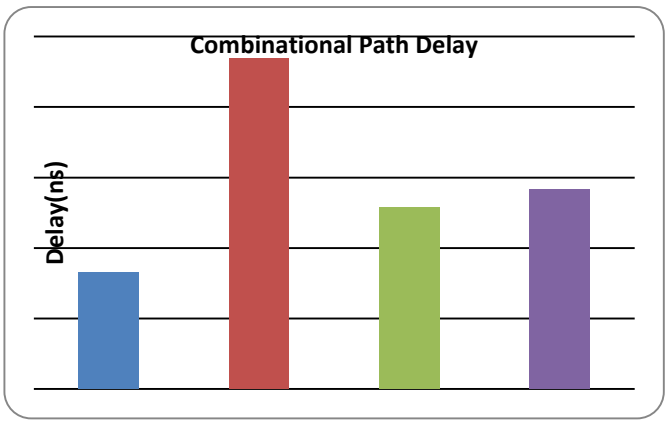


**Figure 7:** Comparison on the basis of Combinational Path Delay

Figure 8 depicts that BEC 2248 Novel Efficient carry select adder utilizes the maximum available memory for synthesis (**135508 kilobytes**).

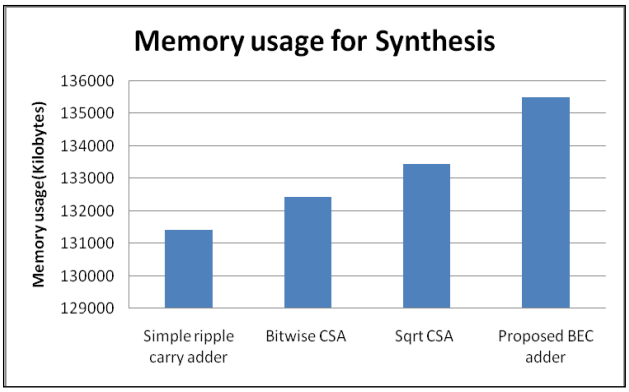


**Figure 8:** Comparison on the basis of Memory usage for Synthesis

Figure 9 depicts that BEC 2248 Novel Efficient carry select adder has the least peak memory usage (**92 MB**).

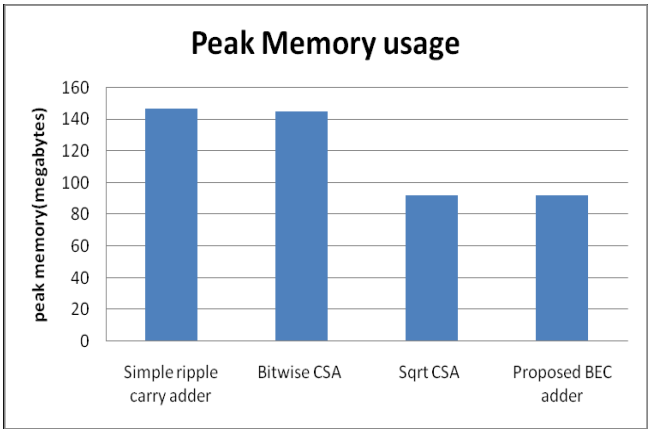


**Figure 9:** Comparison on the basis of Peak Memory Usage

Figure 10 depicts that better on chip power utilization has been achieved (**80.98 mW**) in case of BEC 2248 Novel Efficient carry select adder as compared to all other adders.

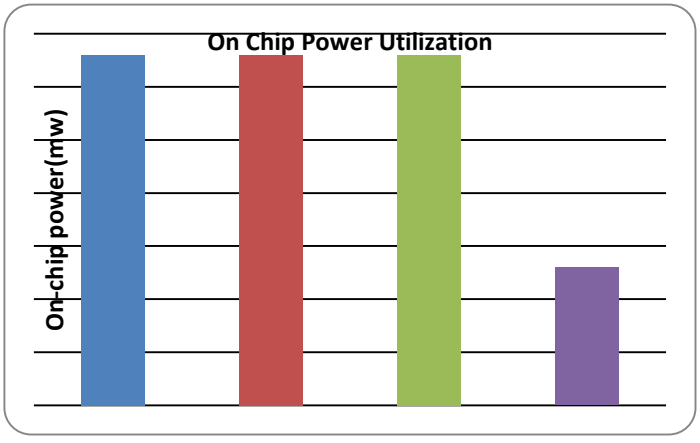


**Figure 10:** Comparison on the basis of On Chip Power Utilization

## VII. CONCLUSIONS

After comparing all the adders-Ripple carry adder, Bitwise CSA, Square root CSA, BEC 2248 Efficient Novel CSA, it is evident from the above graphs that 'on chip power utilization', 'combinational path

delay', 'memory usage for synthesis' and 'Peak memory usage' have been reduced to the desired extent. Hence it can be concluded that BEC 2248 Novel CSA provides a good trade off between area and power consumption and it can be used in implementations where speed is the primary design constraint and power and area are secondary constraints also it is more power efficient than the conventional bitwise CSA.

## ACKNOWLEDGEMENT

It takes a great privilege to thank Mr. S.P. Singh, Chairman, Maharaja Surajmal Institute of Technology (MSIT); Dr. Navin Kapur, Director, MSIT; Dr. V. Kumar (HOD, ECE Dept., and MSIT) for giving us a great opportunity to carry out this research and our heartfelt thanks to our mentor Ms. Sakshi Rajput (Astt. Professor, ECE Dept, MSIT) whose generous support at all stages of this research helped us achieve the desired objectives.

## REFERENCES

**[1]** Parth Sarthy Mohanty, "Design and implementation of low power fast multipliers", thesis report NIT Rourkela, 2009.

**[2]** B. Ramkumar, H.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, 2010.*

**[3]** K.S.Rao, M.Reddy, A.V.Babu and P.Srinivasulu, "Energy and area efficient carry select adder," *IJERA, vol. 34, issue 2, pp. 436-440, Mar-Apr 2012.*

**[4]** 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.*

## AUTHORS

**Sakshi Rajput** supervises a successful research group consisting of graduate students conducting experiments in the field of VlSI design and implementations in at Maharaja Surajmal Institute of Technology affiliated to Guru Gobind Singh Indrapastha University. She has a strong background of VLSI field. Currently she is the Assistant Professor in Electronics and Communications Dept., Maharaja Surajmal Institute of Technology, Guru Gobind Singh Indraprastha University, New Delhi, India.

**Priya Sharma** is currently pursuing B.tech (4th year) in Electronics and Communications from Maharaja Surajmal Institute of Technology affiliated to Guru Gobind Singh Indraprastha University, New Delhi, India.

**Gitanjal**i is currently pursuing B.tech (4th year) in Electronics and Communications from Maharaja Surajmal Institute of Technology affiliated to Guru Gobind Singh Indraprastha University, New Delhi, India.

**Garima** is currently pursuing B.tech (4th year) in Electronics and Communications from Maharaja Surajmal Institute of Technology affiliated to Guru Gobind Singh Indraprastha University, New Delhi, India.