Design & Implementation of Low Power Architecture of Carry Select Adder with Binary to Excess-1 Converter

Dr P G Kuppusamy, Bikkireddy Teja, Kondari Vineesha,
B Manikanta, Chekkiri Vinay
Department of Electronics and Communication Engineering
Siddharth Institute of Engineering & Technology (Autonomous)
Puttur, Andhra Pradesh, India – 517583

Abstract: One of the most substantial areas of research in VLSI is the design of power-efficient and high-speed data path logic systems. The speed of addition is constrained in digital adders by the time taken to propagate a carry through the adder. When the previous bit has been added and the carry generated from this addition is propagated onto the next bit, the sum for each bit in an elementary adder is generated sequentially in this manner. In several computer systems, the Carry Select Adder (CSLA) is used to mitigate the issue of carry propagation delay by generating multiple carry bits and then selecting a carry for the desired output. The CSLA, however, is not area-efficient since it utilizes several Ripple Carry Adders (RCA) pairs to produce partial sum and carry by considering carry data, and then multiplexers pick the final sum and carry (mux). The core concept of this study is to use the Binary to Excess-1 Converter (BEC) to achieve high speed and low-power consumption instead of the RCA in the regular CSLA. The results have been analyzed and compared for implementation of three adders (conventional CSA, CSA with RCA, CSA with BEC). The results from the performance evaluations of the adders are compared with each other. All the simulation is carried out in 45nmtechnology.ThedelayofCSAandCSAwithRCA are same, but the main difference is in reduction of area and power. Similarly when CSA (BEC) and CSA(RCA) are compared, the area has been reduced by approximately 18.67% and power has been reduced by 25.85%.

Keywords: CSLA, RCA, BEC, Area, Power, Delay,45nm
HDL Language used: Verilog HDL
Simulation tool: Model sim
Synthesize Tool: Xilinx/Altera Quartus Tool

1. Introduction

Power consumption is an important efficiency factor in designing Very Large Scale Integrated (VLSI) circuit. In many computers and other kinds of processors, adders are used in arithmetic logic units or ALU. They are also used in other parts of the processor, such as to calculate addresses, table indices, increment and decrement operators, and similar operations. Although adders can be constructed for various number representations, such as binary-coded decimal or excess-3, the most common adders operate on binary numbers. In cases where two‘s complement or one‘s complement is being used to represent negative numbers; it is trivial to modify an adder into an adder–subtractor. Other signed number representations require more logic around the basic adder. Hybrid adders combine several addition schemes to achieve implementation delay/area constraints. Hybrid adders use carry-look ahead and carry-select schemes. Adders are the most used of the more complex operators in a typical design. In certain cases, ASIC designers sometimes employ special versions using combinations of half-adders and full adders. This may work very efficiently in the case of a gate array device, for example, but it will typically result in a very bad FPGA implement tation. The simplest possible adder circuit for binary digits is called a half-adder, and it allows two bits to be added, with a main output and a carry output. The half-adder can be constructed by using a combination of an exclusive-OR gate and an AND gate. The carry bit is 0 except when both inputs bits are 1, which is as required by the rules of binary arithmetic. This half adder is such a useful circuit that it is made in IC form, and it can, in turn, be used to create other circuits. The name _half-adder‘ arises because it can be used only as a first stage in an adder circuit. If we need to add only two bits, the half-adder is sufficient, but if we need to add, say, eight pairs of bits, as when we add two bytes, then the other adders will have three inputs: the bits that are to be added plus the carry bit from the previous stage. For example, adding 1011 and 0011 can use a half-adder for the lowest order pair, giving a 0 output and a carry bit. The addition of the next bits is 1 + 1 + 1 and the carry bit. This gives a 1 output and a 1 carry. The next addition uses the carry from the previous stage, and the last addition uses no carry.

2. Literature Survey

[1] R. Balasai Kesava, B.lingeswara Rao, K. Balasindhuri and N Udaya Kumar. ‖ Low power and area efficient Wallace tree multiplier using carry select adder with binary to excess-1 converter ‖. Multipliers are major blocks in the most of the digital and high performance systems such as Microprocessors, Signal processing Circuits, FIR filters etc. In the present scenario, Fast multipliers with less power consumption are leading with their performance. Wallace tree multiplier with carry select adder (CSLA) is one of the fastest multiplier but utilizes more area. To improve the performance of this multiplier, CSLA is replaced by binary excess-1 counter (BEC) which not only reduces the area at gate level but also reduces power consumption. Area and power calculations for the Wallace tree multiplier using CSLA with BEC are giving good results compared to regular Wallace tree multiplier.

[2] B. Ramkumar and H M Kittur. ―Low power and Area efficient carry select adder‖. Carry Select Adder (CSLA) is one of the fastest adders used in many data-processing processors to perform fast arithmetic functions. From the structure of the CSLA, it is clear that there is scope for reducing the area and power consumption in the CSLA. This work uses a simple and efficient gate-level modification to significantly reduce the area and power of the CSLA. Based on this modification 8-, 16-, 32-, and 64-b square-root CSLA (SQRT CSLA) architecture have been developed and compared with the regular SQRT CSLA architecture. The proposed design has reduced area and power as compared with the regular SQRT CSLA with only a slight increase in the delay. This work evaluates the performance of the proposed designs in terms of delay, area, power, and their products by hand with logical effort and through custom design and layout in 0.18-μm CMOS process technology. The results analysis shows that the proposed CSLA structure is better than the regular SQRT CSLA.

[3] Shivani Parmar and Kirat pal Singh , ‖Design of high speed hybrid carry select adder‖ The paper describes the power and area efficient carry select adder (CSA). Firstly, CSA is one of the fastest adders used in many data-processing systems to perform fast arithmetic operations. Secondly, CSA is intermediate between small areas but longer delay Ripple Carry Adder (RCA) and a larger area with shorter delay carry look-ahead adder. Third, there is still scope to reduce area in CSA by introduction of some add-one scheme. In Modified Carry Select Adder (MCSA) design, single RCA and BEC are used instead of dual RCAs to reduce area and power consumption with small speed penalty. The reason for area reduction is that, the number of logic gates used to design a BEC is less than the number of logic gates used for a RCA design. Thus, importance of BEC logic comes from the large silicon area reduction when designing MCSA for large number of bits. MCSA architectures are designed for 8-bit, 16-bit, 32-bit and 64-bit respectively. The design has been synthesized at 90nm process technology targeting using Xilinx Spartan-3 device. Comparison results of modified CSA with conventional CSA show better results and improvements.

[4] T. Y. Ceiang and M. J. Hsiao, ―Carry select Adder using single ripple carry Adders,‖ A carry-select adder can be implemented by using single ripple carry adder and an add-one circuit instead of using dual ripple-carry adders. This paper proposes a new add-one circuit using the first zero finding circuit and multiplexers to reduce the area and power with no speed penalty.

[5] S. Manjuiand V. Sornagopae, ―An Efficient SQRT Architecture of Carry Select Adder Design by Common Boolean Logic Carry Select adder (CSLA) is known to be the fastest adder among the Conventional adder structures. This work uses an efficient Carry select adder by sharing the Common Boolean logic (CLB) term. After a logic simplification, we only need one OR gate and one inverter gate for carry and summation operation. Through the multiplexer, we can select the correct output according to the logic states of the carry in signal. Based on this modification Square root CSLA (SQRT CSLA) architecture have been developed and compared with the regular and Modified SQRT CSLA architecture. The Modified CSLA architecture has been developed using Binary to Excess -1 converter (BEC). This paper proposes an efficient method which replaces a BEC using common Boolean logic. The result analysis shows that the proposed architecture achieves the three folded advantages in terms of area, delay and power

3. Proposed System

In the proposed design, Carry Select Adder (CSA) is designed by using Ripple Carry Adder (RCA) with Binary to Excess-1 Converter (BEC). Because RCA would incur more delay and more power consumption, to reduce the propagation delay and power consumption, we use BEC-based Carry Select Adder. The goal of fast addition is achieved using BEC together with a multiplexer (mux). One input of the 2:1 mux, as shown in Fig. 2, has inputs (B3, B2, B1, and B0), The other input of the mux is the BEC output.

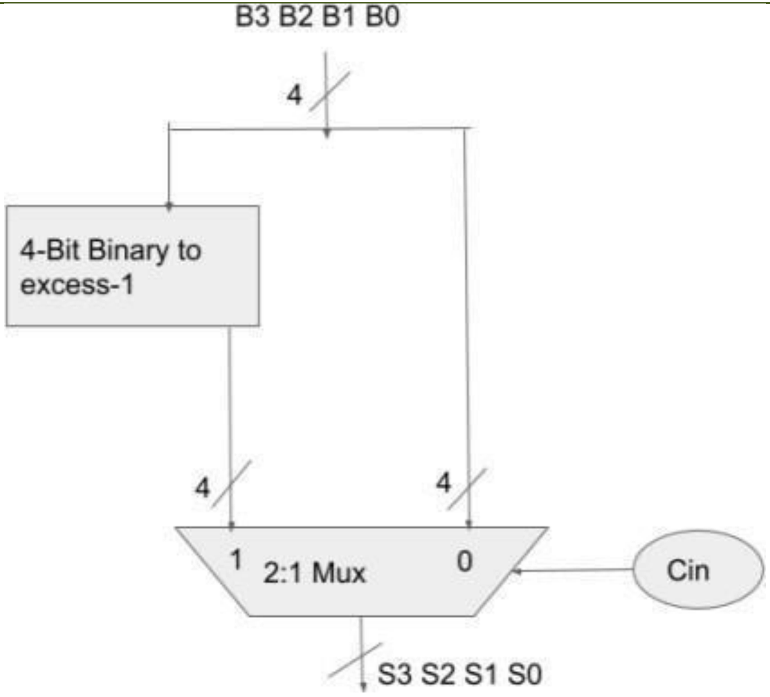


Figure3.1: 4-bit BEC with 2:1 mux

The CSA functions by using the 4-bit BEC along with the multiplexer. One input of the 2:1 multiplexer gets as is input (B3, B2, B1, and B0) and another input of the multiplexer is the BEC output. This produces the two possible sectional results in parallel, and according to the control signal Cin, the multiplexer is used to select either the BEC output or the direct inputs.

The importance of the BEC logic stems from the large silicon area reduction when the CSLA with large number of bits are designed. The Boolean expressions of the 4-bit BEC is listed as

$X0 = \neg B0$ (1)
$X1 = B0 \vee B1$ (2)
$X2 = B2 \vee (B0\ AB1)$ (3)
$X3 = B3 \vee (B0\ AB1\ AB2)$ (4)

The basic idea of this work is to use BEC instead of RCA with Cin=1. CSLA BEC uses fewer number of logic gates than N-bit full adder structure in order to optimize area and power. N-bit RCA is replaced by (N+1)-bit BEC.

Therefore, the modified CSLA has low power and less area than conventional CSLA. CSLA has been picked for correlation with modified design using BEC as it has more stabilized delay, less area and low power. To reduce the delay, area and power, the design is modified by using BEC instead of RCA with Cin=1.

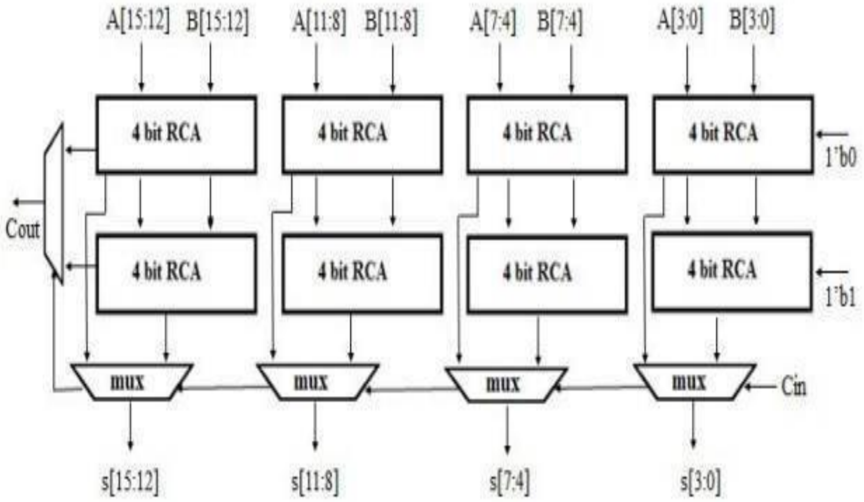


Figure3.2: Block diagram of CSA with RCA

3.1Proposed Architecture:

The entire work has been performed by using of Binary to Excess- Converter (BEC) instead of RCA with Cin = 1 in the regular CSLA to achieve lower power consumption The main advantage of this BEC logic comes from the fewer number of logic gates than the n-bit Ripple Carry Adder (RCA). The structure and the truth table of 4-bit BEC is shown in below Fig. 2 and Table 1, respectively. FIg.3 shows the block diagrram of CSA with RCA, where few of the full adders froma conventional CSA have been replaced by 4-bit RCA. This architecture is used to reduce area and lower the power when compared to converntional CSA

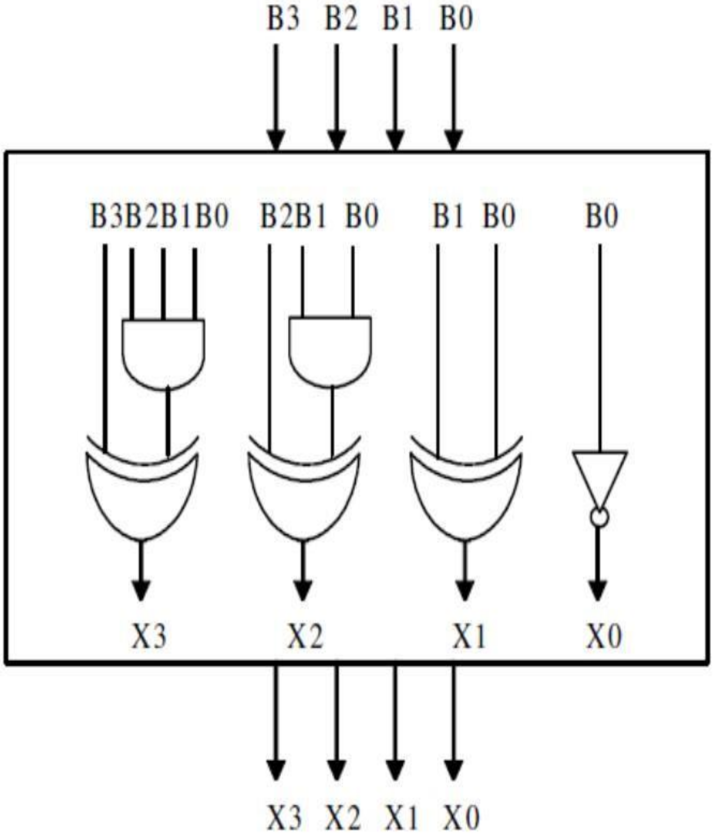


Figure3.3: Modified architecture of BEC-1

The carry-select adder generally consists of ripple-carry adders and a multiplexer. Adding two n-bit numbers with a carry-select adder is done with two adders (therefore two ripple-carry adders), in order to perform the calculation twice, one time with the assumption of the carry-in being zero and the other assuming it will be one. After the two results are calculated, the correct sum, as well as the correct carry-out, is then selected with the multiplexer once the correct carry-in is known.

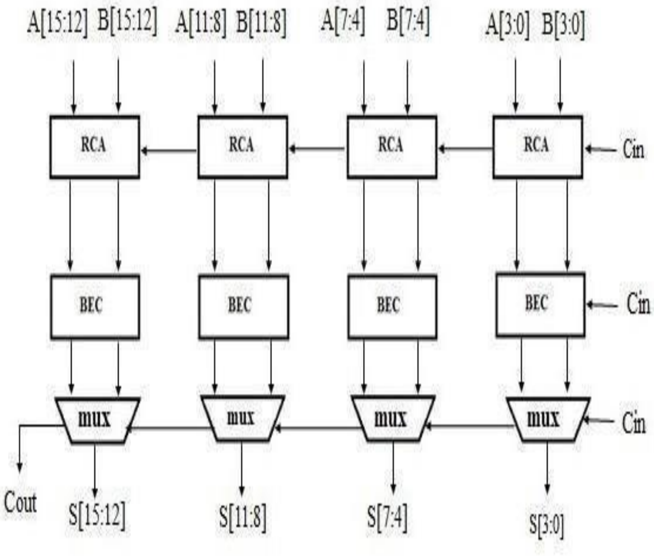


Figure3.4: Block diagram of proposed 16-bit CSA with BEC

Advantage Of Proposed Architecture:

a) Consumes less power since the switching activity is reduced

b) Reduce the critical.

c) Reduces the number of logic gates.

Table 1: Truth table of BEC

Binary Logic	***Excess-1 Logic***
0000	0001
0001	0010
0010	0011
0011	0100
0100	0101
0101	0110
0110	0111
0111	1000
1000	1001
1001	1010
1010	1011
1011	1100
1100	1101
1101	1110
1110	1111
1111	0000

3.2 Methodology and Flow Chart:

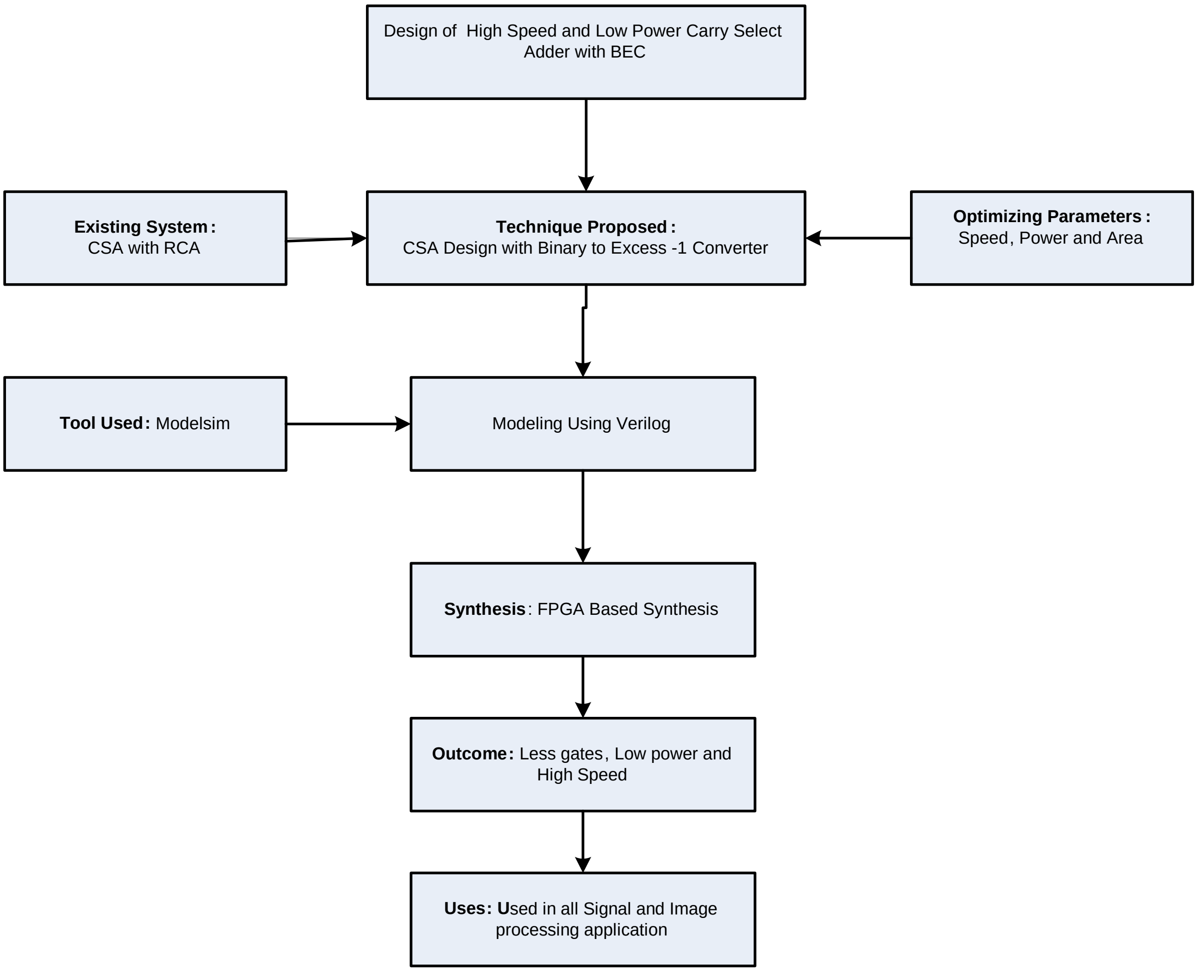

Design of High Speed and Low Power Carry Select Adder with BEC
Existing System: CSA with RCA
Technique Proposed: CSA Design with Binary to Excess -1 Converter
Optimizing Parameters: Speed, Power and Area
Tool Used: Modelsim
Modeling Using Verilog
Synthesis: FPGA Based Synthesis
Outcome: Less gates, Low power and High Speed
Uses: Used in all Signal and Image processing application

Figure3.5 Flowchart

4. Applications

4.1Signal Processing

Digital signal processing (DSP) has many advantages over analog signal processing. Digital signals are more robust than analog signals with respect to temperature and process variations. The accuracy in digital representations can be controlled better by changing the word length of the signal. Furthermore, DSP techniques can cancel the noise and interference while amplifying the signal. In contrast, both signal and noise are amplified in analog signal processing. Digital signals can be stored and recovered, transmitted and received, processed and manipulated, all virtually without error. While analog signal processing is indispensable for systems that require extremely high frequencies such as the radio frequency transceiver in wireless communications, or extremely low area and low power such as micro machine sensors used to detect cracks and other stress-related material defects, many complex systems are realized digitally with high precision, high signal to noise ratio (SNR), repeatability, and flexibility.

Fig 4.1 Signal processing

4.2Image Processing

The development and implementation of systems for the more complex real time image processing and scene understanding tasks, such as robot vision and remote surveillance, calls for faster computation than that possible using the traditional serial computer. The advent of VLSI has made feasible the consideration of more specialized processing architectures, designed to support these data rates, while keeping systems compact and relatively cheap. Two approaches are discussed: the use of a programmable processor array, and the customizing of image processing algorithms in silicon. This paper examines designs based upon each approach in the light of the techniques and constraints of VLSI. In particular we describe in some detail an example of a VLSI parallel array processor, the Grid (GEC rectangular image and data processor), and a number of special- purpose CMOS/SOS chips based on systolic design techniques.

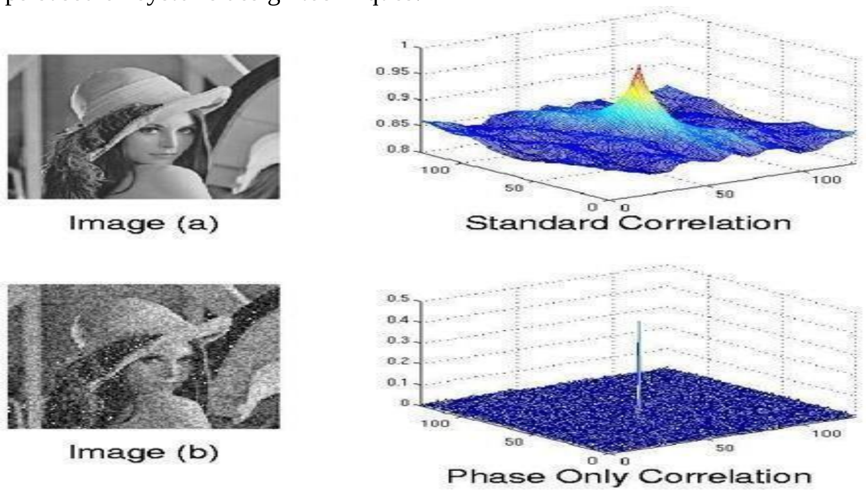


Fig 4.2 Image processing

4.3 Advanced Microprocessors Design

Advanced Microprocessor Design provides a comprehensive guide for Embedded System designers and Computer Engineers. It provides a broad and in-depth overview of important topics ranging from Computer Architecture and Operating System design to PCB Layout and manufacturing guidelines. It is a fast paced course intended for graduate and post graduate students in Electrical and Computer Engineering, as well as the Embedded System professional

Fig 4.3 Advanced Microprocessor Design

4.4 High Speed Multiplications

Multiplication involves two basic operations: the generation of partial products and their accumulation. Consequently, there are two ways to speed up multiplication: reduce the number of partial products or accelerate their accumulation. Clearly, a smaller number of partial products also reduces the complexity, and, as a result, reduces the time needed to accumulate the partial products. High-speed multipliers can be classified into three general types.

The first generates all partial products in parallel, and then uses a fast multi-operand adder for their accumulation. This is known as a parallel multiplier.

The second, known as a high-speed sequential multiplier, generates the partial products sequentially and adds each newly generated product to the previously accumulated partial product.

The third is made up of an array of identical cells that generate new partial products and accumulate them simultaneously. Thus, there are no separate circuits for partial product generation and for their accumulation. This is known as an array multiplier, and it tends to have a reduced execution time, at the expense of increased hardware complexity.

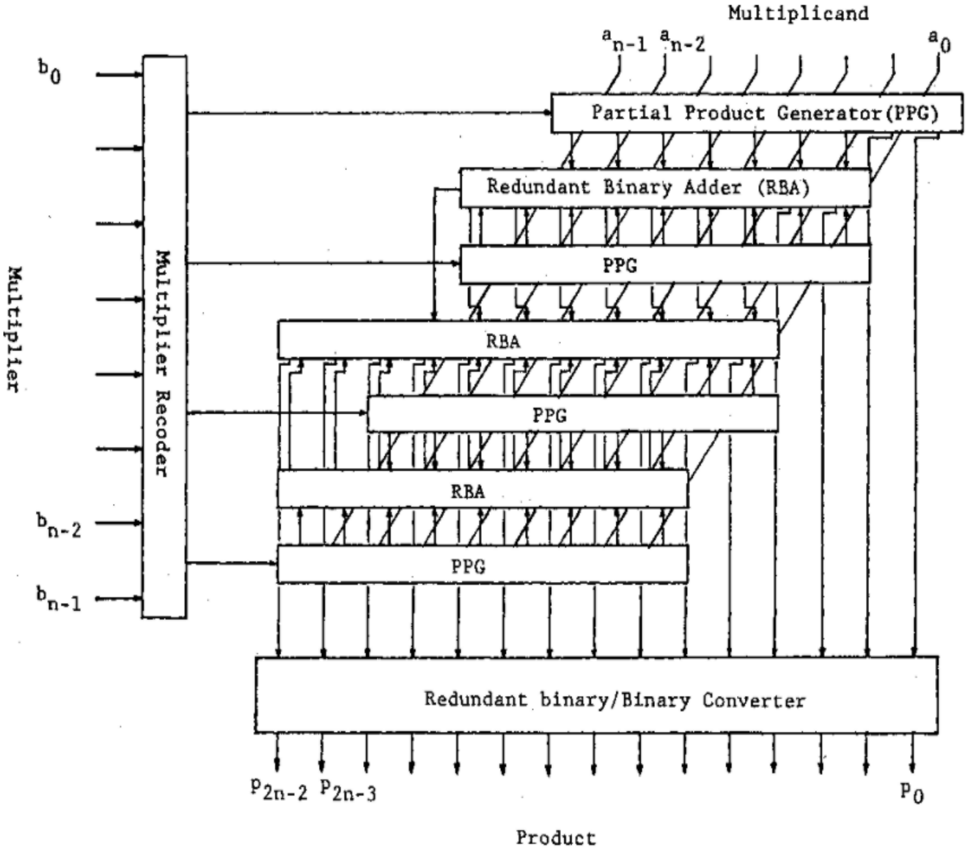
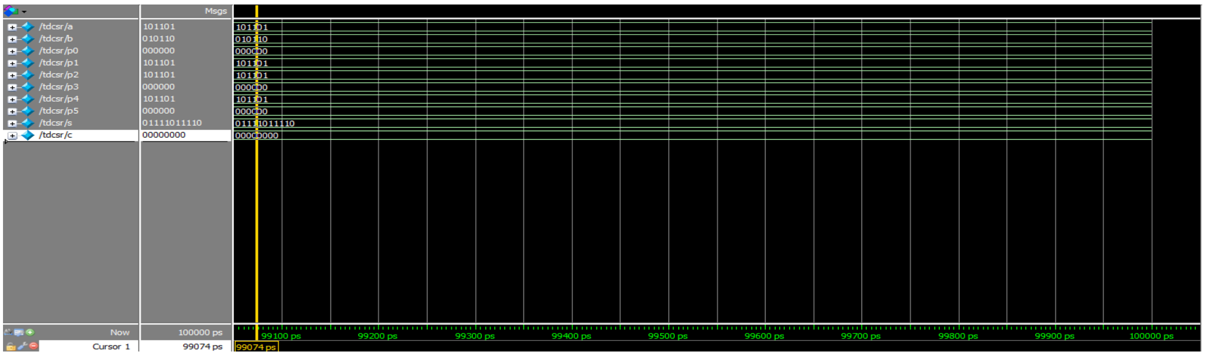

Multiplicand
a_{n-1} a_{n-2} a_0
b_0
Multiplier
Multiplier Recoder
b_{n-2}
b_{n-1}
Partial Product Generator(PPG)
Redundant Binary Adder (RBA)
PPG
RBA
PPG
RBA
PPG
Redundant binary/Binary Converter
P_{2n-2} P_{2n-3} P_0
Product

Fig 4.4 High Speed Multiplications

4.5 Arithmetic Logic Unit

In computing, an arithmetic logic unit (ALU) is a combinational digital circuit that performs arithmetic and bitwise operations on integer binary numbers. This is in contrast to a floating-point unit (FPU), which operates on floating point numbers. It is a fundamental building block of many types of computing circuits, including the central processing unit (CPU) of computers, FPUs, and graphics processing units (GPUs). The inputs to an ALU are the data to be operated on, called operands, and a code indicating the operation to be performed; the ALU's output is the result of the performed operation. In many designs, the ALU also has status inputs or outputs, or both, which convey information about a previous operation or the current operation, respectively, between the ALU and external status registers.

5. Results and Discussions



```
                 Msgs
/tdcsr/a         101101        101101
/tdcsr/b         010110        010110
/tdcsr/p0        000000        000000
/tdcsr/p1        101101        101101
/tdcsr/p2        101101        101101
/tdcsr/p3        000000        000000
/tdcsr/p4        101101        101101
/tdcsr/p5        000000        000000
/tdcsr/s         01111011110   01111011110
/tdcsr/c         00000000      00000000

Now              100000 ps
Cursor 1         99074 ps      99074 ps
```

Fig 6.1 Simulation result of CSA

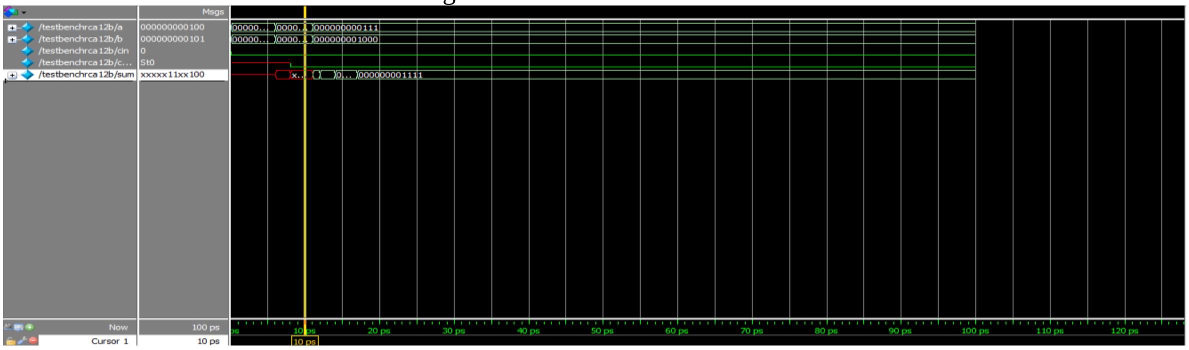


Fig 6.2 Simulation result of CSA with RCA

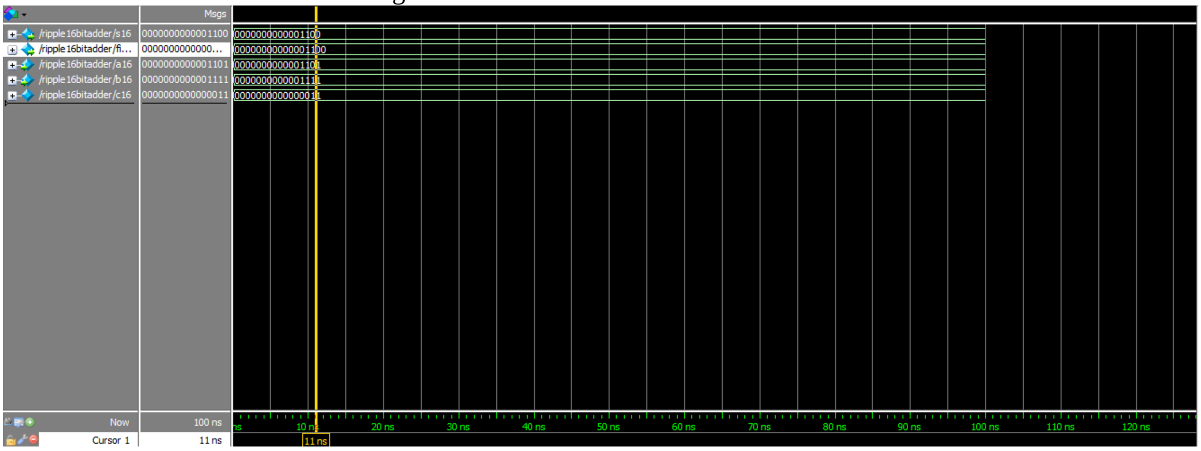


Fig 6.3 Simulation result of CSA with BEC

The simulation has been carried out using cadence tool. The required libraries are already inbuilt in cadence. We call the required libraries by using the attributes from cadence. Table 2 represents the comparision of three characteristics (area, power, delay) of the three adders discussed. Figures 5 and 6 depict the simulation results of the respective adders. This concept is mainly focused on reducing the power and area of the adders, but not the delay. When a conventional CSA is used we can observe that the area and power are very high because of using high number of gates. By using RCA in addition to CSA, we can reduce the number of gates used, which in turn reduces the area and power of the adder. Later on by using the BEC , the carry is not propagated. Therefore, CSA with BEC gives the optimum result in reducing the power and delay of the carry select adder.

Table 2: Comparison of different adders

Adder	**16-bit adder**		
	Area(um2)	***Delay(ps)***	***Power(nw)***
CSA	239	113	10268.941
CSA with RCA	231	204	59268.941
CSA with BEC	161	98	56361.481

6. Conclusion

The architectures of Carry Select Adder (CSA) with Ripple Carry Adder (RCA) and carry select adder with Binary to Excess-1 Converter (BEC) is designed. The conventional carry select adder has the disadvantage of more power consumption and occupying more chip area. With respect to delay time and power consumption, we can conclude that the implementation of CSA with BEC is more efficient as compared to existing methods. The main advantage of this BEC logic comes from the fewer number of logic gates than the 4-bit Full Adder (FA). With BEC, there is no carry propagation, thereby lower area and less dealy which in turn leads to low power consumption. Furthermore, the area and delay can be reduced by using 4-bit parallel prefix adders. Using this concept we can reduce the area, minimise the power consumption and increases the speed of the devices. In future these circuits can be implemented in low nano meter technologies and get the high performance of the systems. If the nanometres technology is reduced the area and power also reduced the life time circuit increases. In future carry select adder can be implemented in higher bit levels. In fututre it can be used in advanced microprocessor design.

7. References

[1] 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.

[2] D. Radhakrishnan, ―Low-voltage low power CMOS full adder,‖ in Proc. IEEE Circuits Devices Syst., vol. 148, Feb. 2001.

[3] E. Abu-Shama and M. Bayoumi, ―A new cell for low power adders,‖ in Proc. Int. Midwest Symp. Circuits and Systems, pp. 1014–1017,1995.

[4] T. Y. Ceiang and M. J. Hsiao, ―Carry select Adder using single ripple carry Adders,‖ Electron. Lett., vol. 34, no. 22, pp. 2101–2103, Oct. 1998.

[5] B. Ramkumar and H M Kittur ―Low power and Area efficient carry select adder,‖ IEEE Trans,Vol.20,Feb 2012.

[6] ShivaniParmar and Kirat pal Singh , ‖Design of high speed hybrid carry select adder,‖ IEEE 2012.

[7] S.Manjui and V. Sornagopae, ―An Efficient SQRT Architecture of Carry Select Adder Design by Common Boolean Logic,‖ IEEE, 2013.

[8] U Sreenivasulu and T V Sridhar, ―Implementation of an 4-bit ALU using low power and area efficient carry select adder,‖ International Conference on Electronics and Communication Engineering,20 May2012.

[9] A Tyagi, ―A Reduced Area Scheme for Carry-Select Adders,‖ IEEE International Conference on Computer Design, pp.255-258, Sept 1990

[10] J. M. Rabaey, ―Digital Integrated Circuits- A Design Perspective,‖ New Jersey, Prentice-Hall, 2001.. 836 — P a g e

[11] K Rawat, TDarwish. and M Bayoumi, ―A low power and reduced area Carry Select Adder, ‖ 45th Midwest Symposium on Circuits and Systems, vol.1, pp. 467470,March 2002.

[12] M. Moris Mano, ―Digital Design,‖ Pearson Education, 3rd edition, 2002.

[13] O. J. Bedrij, ―Carry-Select Adder,‖ IRE Transactions on Electronics Computers, vol.EC- 11, pp. 340-346, June1962. [15] Y Kim and LKim, ―A low power carry select adder with reduced area, ‖ IEEE International Symposium on Circuits and Systems, vol.4, pp.218221, May 2001

[14] B Amelifard, F Fallah, and Massoud Pedram, ― Closing the gap between Carry Select Adder and Ripple Carry Adder: a new class of low-power high-performance adders,‖ Sixth International Symposium on Quality of Electronic Design, pp.148-152. April 2005.