RESEARCH ARTICLE OPEN ACCESS

Performance Enhancement of Multi-Output Carry Look-Ahead Cmos Csa

K. Ram Babu, N.V.P Naidu Babu, M. Tech, Aditya Putta
M. Tech Student
Assistant Professor
Professor & HOD (11H91D6806)

Abstract
Carry Select Adder (CSA) is solitary of the best ever adders worn in numerous data-processing processors to complete speedy arithmetic functions. From the construction of the CSA, it is lucid that there is scope for sinking the vicinity and clout burning up in the CSA. This adder is based on both a static and compact multi-output carry look-ahead (CSA) circuitof highly area-efficient CMOS carry-select adder (CSA) with a regular and iterative-shared transistor structure very suitable for implementation in VLSI and a very simple select circuit. Comparisons with other representative 32-bit CSAs show that the proposed adder reduces the area. This paper uses a easy and proficient gate-level amendment to drastically diminish the vicinity and power of the CSA. Based on this modification different square-root CSA (SQRT CSA) constructions have been developed and compared with the customary square-root architecture. The projected intend has reduced vicinity and power as compared with the regular Square Root CSA with only a slight increase in the delay. In this paper, conventional CSA is compared with Modified Carry select adder (MCSA), Regular Square Root CSLA (SQRT CSA), Modified SQRT CSA and Proposed SQRT CSA in terms of area, delay and power consumption. The result analysis shows that the proposed structure is better than the conventional CSA.

Keywords: delay, area , SQRT CSA, VLSI, CMOS,

I. INTRODUCTION

Major revolution in the field of electronics has occurred by the invention of transistor in 1947 by William.B.Shockley and his colleagues at Bell laboratories. This invention has subsequently created a powerful platform for the emergence of a new industry in electronics called ‘Microelectronics’. This new industry has helped in the emergence of Integrated Circuits (IC) in the beginning of 1960s. As the technology advanced day by day, the number of devices per IC has gone rapidly increasing over the decades which resulted in rapid transition from various levels of integrations to the present VLSI (Very Large Scale Integration) technology. VLSI IC’s are those circuits which contain more than 105 transistors and these circuits can be used as general purpose IC’s such as microprocessors, memories, DSPs and also as Application Specific IC’s (ASICs). In VLSI technology, the main design entity is area which measures the cost and power consumption of that IC. 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. In rapidly growing mobile industry, faster units are not the only concern but also smaller area and less power become major concerns for design of digital circuits. In mobile electronics, reducing area and power consumption are key factors in increasing portability and battery life. Even in servers and desktop computers power dissipation is an important design constraint. Addition is the heart of computer arithmetic, and the arithmetic unit is often the work horse of a computational circuit. They are the necessary component of a data path, e.g. in microprocessors or a signal processor. There are many ways to design an adder. The Ripple Carry Adder (RCA) provides the most compact design but takes longer computing time. If there is N-bit RCA, the delay is linearly proportional to N. Thus for large values of N the RCA gives highest delay of all adders. The Carry Look-Ahead Adder (CLA) gives fast results but consumes large area.

If there is N-bit adder, CLA is fast for N≤4, but for large values of N its delay increases more than other adders. So for higher number of bits, CLA gives higher delay than other adders due to presence of large number of fan-in and a large number of logic gates. In digital adders, the speed of addition is limited by the time required to propagate a carry through the adder. The sum for each bit position in an elementar y adder is generated sequentially only after the previous bit position has been summed and a carry propagated into the next position. The Carry Select Adder (CSA) provides a compromise between RCA and CLA. CSA is one of the fastest adders used in many data-processing processors to perform fast arithmetic functions. The CSA partitions the adder into several groups, each of which performs two additions in parallel. Therefore, two copies of ripple-carry adder act as carry evaluation block per select stage. One copy evaluates the carry chain assuming the block carry-in is zero, while the other assumes it to be one. Once the carry signals are finally computed, the correct sum and carry-out signals will be simply selected by a set of multiplexers. However, the CSA is not area efficient because it uses multiple pairs of Ripple Carry Adders (RCA) to generate partial sum and carry by considering carry input Cin=0 and Cin=1, then the final sum and carry are selected by the multiplexers (mux). The square root carry select adder (SQRT CSA) for 8, 16, 32 input using add one circuit is proposed to minimize the area and power. The SQRT CSLA and the SQRT CSA with Binary to Excess-1 Converter (BEC) has been chosen for comparison with the proposed design.

II. CARRY SELECT ADDER

Addition is the most commonly used arithmetic operation. It is the speed- limiting element as well. The ripple carry adder is composed of many cascaded single- bit full-adders. The circuit architecture is simple and area-efficient. However, the computation speed is slow because each full-adder can only start operation till the previous carry-out signal is ready. In the carry select adder, N bits adder is divided into M parts .The carry-select adder can compute faster because the current adder stage does not need to wait the previous stage's carry-out signal. The summation result is ready before the carry-in signal arrives; therefore, we can get the correct computation result by only waiting for one multiplexer delay in each single bit adder. In the carry select adder, the carry propagation delay can be reduced by M times as compared with the carry ripple adder. The carry select adder is faster and intermediate when compared with other adders.

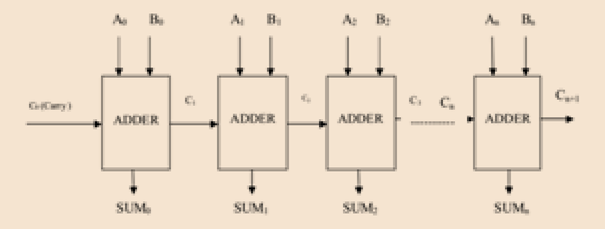
*A. **Adder topologies***

Many different adder architectures have been proposed for speeding up binar y addition over the literature survey . For cell-based design techniques they can be well characterized with respect to circuit area and speed as well as suitabilit y for logic optimization and synthesis. Few of them are

- Ripple Carry Adder
- Carry Save Adder
- Carry Look-Ahead Adder
- Carry Increment adder
- Carry By pass Adder (Carry Skip Adder)
- Carry Select Adder

*B. **Ripple Carry Adder***

The ripple carry adder (RCA) is constructed by cascading full adders)FA) blocks in series. One full adder is responsible for the addition of two binary digit s at any stage of the ripple carry . The carry out of one stage is fed directly to the carry in of the next stage. Even though this is a simple adder and can be used to add unrestricted bit length numbers, it is however not very efficient when large bit numbers are used. One of t he most serious drawbacks of this adder is that the delay increases linearly with the bit length.



giF1 N-bit Ripple carry adder

The worst-case delay of the RCA is when a carry signal transition ripples through all stages of adder chain from the least significant bit to the most significant bit, which is approximated by:

$$t = (n - 1)\, tc + ts \qquad (1)$$

To solve the carry propagation delay, Carry Select Adder (CSA) is developed which drastically reduces t he area and delay to a great extent. The CSA is used in many computational systems design to moderate the problem of carry propagation delay by independently generating multiple carries and then select a carry to generate the sum.

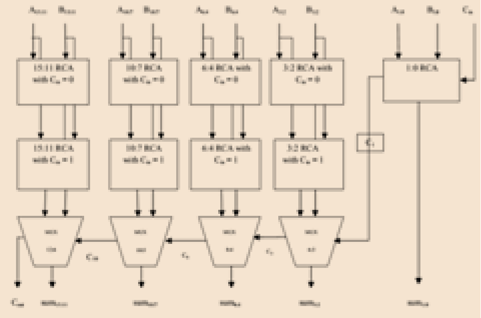


Fig 2 4- Bit carry select adder

III. SQUARE ROOT CARRY SELECT ADDER

To optimize a design, it is essential to locate the critical timing path first. Consider the case of a 16-bit linear carry-select adder. To simplify the discussion assume that the full-adder and multiplexer cells have identical propagation delays equal to a normalized value of 1. This analysis demonstrates that the critical path of the adder ripples through the multiplexer networks of the subsequent stages.

Consider the multiplexer gate in the last adder stage. The inputs to this multiplexer are the two carry chains of the block and the block-multiplexer signal from the previous stage. A major mismatch between the arrival times of the signals can be observed. The results of the carry chain are stable long before the multiplexer signal arrives. It makes sense to equalize the delay through both paths.

This can be achieved b y progressively adding more bits to the subsequent stages in the adder, requiring more time for the generation of the carry signals. For example, the first stage can add 2 bits, the second contains 3, the third has 4, and so forth. This type of adder is called square root carry select adder

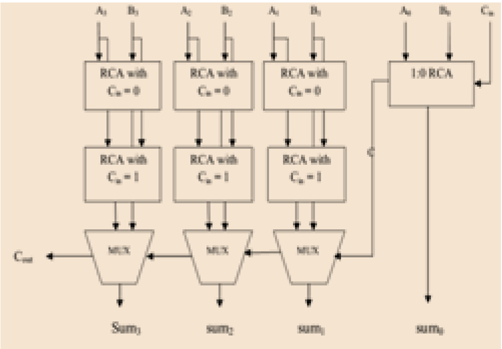


Fig 3 16-Bit square root carry select adder

IV. DELAY AND AREA EVALUATION OF 16-BIT SQUARE ROOT CSA

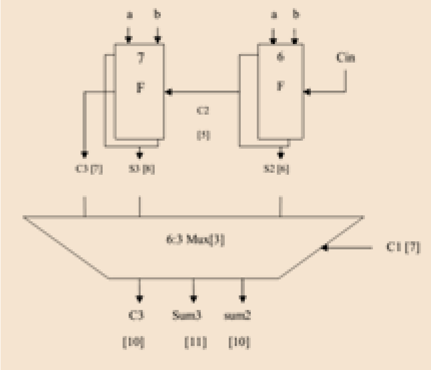
A. Area evaluation of 16-bit Square Root CSA

GROUP-1:

Group 1 consists of a single 2-bit RCA. The 2-bit RCA consists of two full adders in turn each full adder is composed of 13 gates i.e., 2 XOR gates (each XOR gate is composed of 5 gates) =10 gates, 2 AND gates and 1 OR gate. So the total gate count in group 1 is 26(2 *13).

GROUP-2:

Group 2 consists of two RCAs of 2-bit size. The RCA with Cin=0 consist s of one half adder and a full adder. The half adder consists of 6 gates i.e., one XOR gate which is in turn composed of 5 gates and one AND gate. The full adder consists of 13 gates. The RCA with Cin=1 consists of two full adders. The multiplexer used in group 2 is a 6:3 multiplexer which has three 2:1 multiplexers. Each 2:1 multiplexer consists of 4 gates (one inverter, one OR and two AND gates).



giF4 GROUP 2 block

So the total gate count of group two is Gate count =57(FA+HA+MUX)

FA=3*13=39

HA=1*6=6

Mux =3*4=12

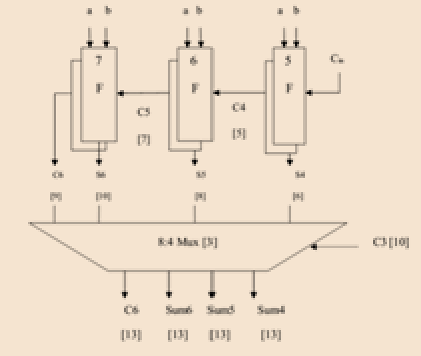
GROUP-3:

Gate count =87(FA+HA+MUX)

FA=5*13=65

HA=1*6=6

Mux =4*4=16



giF5 GROUP 3 block

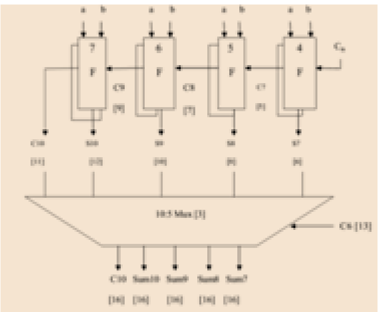
GROUP-4:

Gate count =117(FA+HA+MUX)

FA=7*13=91

HA=1*6=6

Mux =5*4=20



giF6 GROUP 4 block

GROUP-5:

Gate count =147(FA+HA+MUX)

FA=9*13=117

HA=1*6=6
Mux =6*4=24

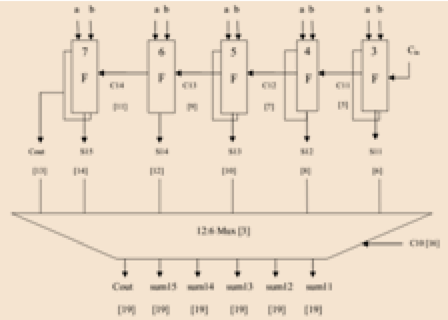


Fig 7 GROUP 5 block

V. DELAY EVALUATION OF 16-BIT SQUARE ROOT CSLA

The structure of the 16-bit regular SQRT CSLA is shown in Fig. 3. It has five groups of different size RCA. The delay evaluation of each group are shown in Fig. 4 to 7, in which the numerals within specify the delay values, e. g., sum2 requires 10 gate delays. The steps leading to the evaluation are as follows-1) The group2 [see Fig.2.13] has two sets of 2-b RCA. Based on the consideration of delay values of Table I, the arrival time of selection input c1[t=7] of 6: 3 mux is ear lier than S3[t=8] and later than mux [t=3]. Thus, sum2 [t=10] is summation of c1 and mux and sum3 [t=11] is summation of s3 and mux.2) Except for group2, the arrival time of mux selection input is always greater than the arrival time of data outputs from the RCA's. Thus for remaining groups, the delay s for sum are the summation of delay of mux and carr y from previous group.

Table 1. Area and delay count of 16-bit SQRT CSA

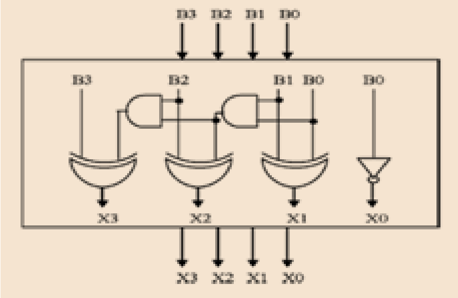
S.NO	Group	Delay	Area (number of gates)
1	Group 1	7	26
2	Group 2	11	57
3	Group 3	13	87
4	Group 4	16	117
5	Group 5	19	147

VI. SQUARE ROOT CARRY SELECT ADDER WITH BINARY TO EXCESS-1 CONVERTER

The area of the CSLA is increasing due to the use of dual RCAs. The area can be reduced by replacing RCA wit h Cin =1 with Binary to Excess-1 Converter (BEC). BEC adds one to the given input, that means if the output of RCA with Cin=0 is given as input to BEC than it adds one to the input. So, this circuit performs the same operation of RCA with Cin=1.

A. Binary TO Excess-1 Converter:

The main idea of this work is to use BEC instead of the RCA with cin=1 in order to reduce the area and power consumption of the regular CSLA. To replace the n-bit RCA, an n-bit BEC is required. A structure of a 4-b BEC is shown in Fig 8



giF8 Structure of 4-bit BEC

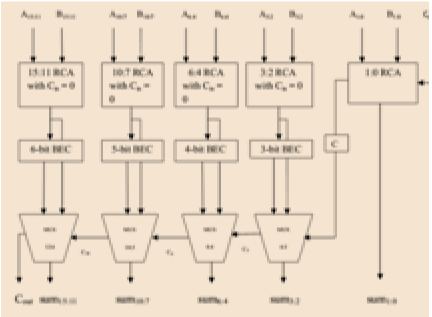


Fig 9 16-bit Square root carry select adder with BEC

As shown in fig 9, the architecture of modified carry select adder is obtained by replacing the RCA with Cin=1 with Binary-1 Converter (BEC) in the regular CSLA to achieve low area and power consumption. The modified 16-bit carr y select adder can be divided into 5 groups. Group 1 consists of only a 2-bit Ripple Carr y Adder (RCA) where as remaining groups i.e., group 2 to group 5 consists of a RCA with carry in 0, a BEC and a multiplexer. The multiplexer is used to select the sum and carry values from the RCA and BEC by using the control signal to it. The control signal to multiplexer is nothing but the carry out of the previous group. If the control signal is 1 then sum and carry out of BEC is selected b y the multiplexer and if control signal is 0 then sum and carry out of RCA with Cin=0 is selected by the multiplexer.

VII. DELAY AND AREA EVALUATION OF 16-BIT SQUARE ROOT CSA

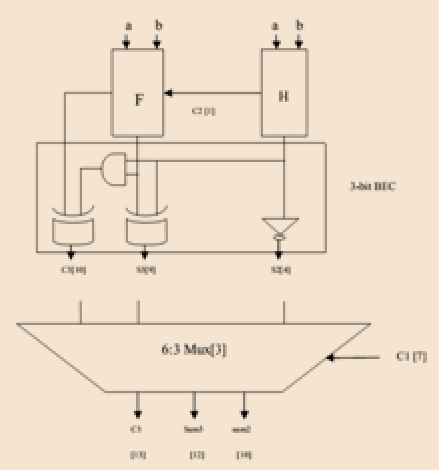
A. Area evaluation of 16-bit square root CSA

GROUP-1:

Group 1 consists of a single 2-bit RCA. The 2-bit RCA consists of two full adders in turn each full adder is composed of 13 gates i.e., 2 XOR gates (each XOR gate is composed of 5 gates) =10 gates, 2 AND gates and 1 OR gate. So the total gate count in group 1 is 26(2*13).

GROUP-2:

Gate count =43(FA+HA+MUX+BEC)
FA=1*13=13
HA=1*6=6
AND=1
NOT=1
XOR=10(2*5)
Mux =3*4=12



giF10 Group 2 block with 3-bit BEC

GROUP-3:

Gate count=66(FA+HA+MUX+BEC)
FA=2*13=26
HA=1*6=6
NOT=1
XOR=3*5=15
AND=2
MUX =4*4=16

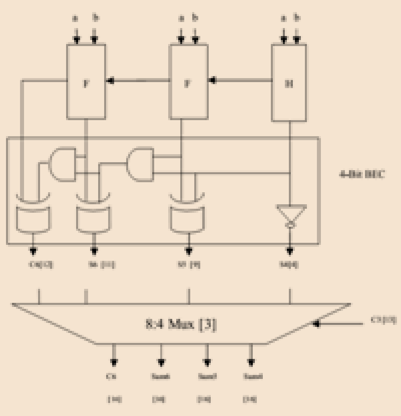


Fig 11 Group 3 block with 3-bit BEC

GROUP-4:

Gate count =89(FA+HA+MUX+BEC)
FA=3*13=39
HA=1*6=6
NOT=1
XOR=4*5=20
AND=3
MUX =5*4=2

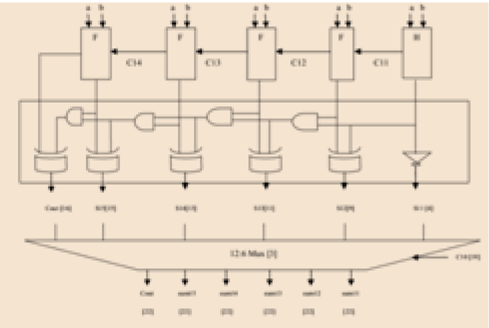
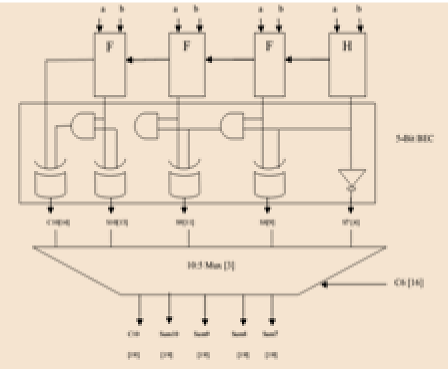


Fig 12 Group 4 block with 3-bit BEC

GROUP-5:

Gate Count =112(FA+HA+MUX)
FA=4*13=52
HA=1*6=6
NOT=1
XOR=5*5=25
AND=4
Mux =6*4=24



giF13 Group 5 block with 6-bit BEC

VIII. DELAY EVALUATION OF 16-BIT SQUARE ROOT CSA WITH BEC

The structure of the 16-b SQRT CSLA using BEC instead of RCA with Cin=1 to optimize the area and power is shown in Fig. 3.3. We again split the structure into five groups. The steps for the delay evaluation are

1) The group 2[see Fig. 3.4] has one 2-b RCA which has 1 FA and 1 HA for Ci n=0. Instead of another 2-b RCA with Cin=1 a 3-b BEC is used which adds one to the output from 2-b RCA. Based on the consideration of delay values of Table 2 the arrival time of selection input c 1 [t=7] of 6:3 mux is earlier than the S 3[t=9] and C 3[t=10] and later than theS 2[t=4]. Thus, the sum3 and final C 3(output from mux) are depending onS3 and mux and partial C 3(input to mux) and mux, respectivel y . The sum2 depends on C1 and mux.

2) For the remaining group's the arrival t ime of mux selection input is alway s greater than the arrival time of data inputs from the BEC's. Thus, the delay of the remaininggroups depends on the arrival time of mux selection input and the mux dela.

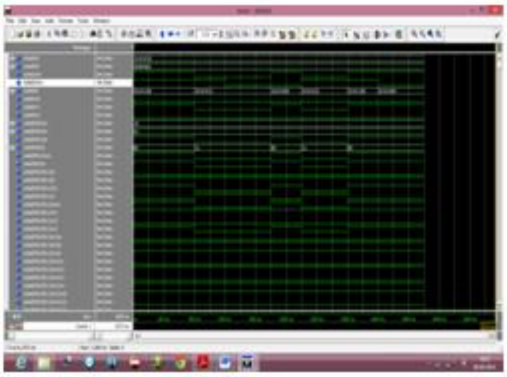
elbaT2 . Area and delay count of 16-bit SQRT CSA with BEC

S.NO	Group	Delay	Area (number of gates)
1	Group 1	7	26
2	Group 2	13	43
3	Group 3	16	66
4	Group 4	19	89
5	Group 5	22	112

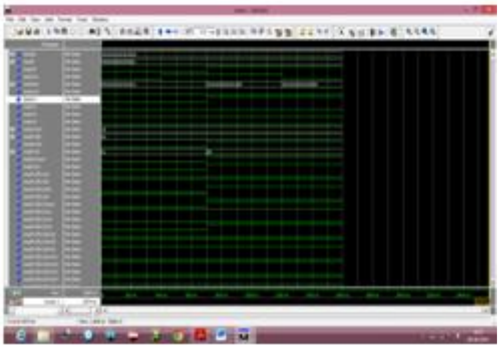
The estimated maximum delay and area of the other groups of the SQRT CSLA with BEC are evaluated and listed in Table 2. By comparing table 1 and 2, delay is increased in 16-bit SQRT CSA with BEC. To reduce this delay penalty , a new architecture of SQRT CSA using add one circuit is proposed.

IX. OUTPUTS

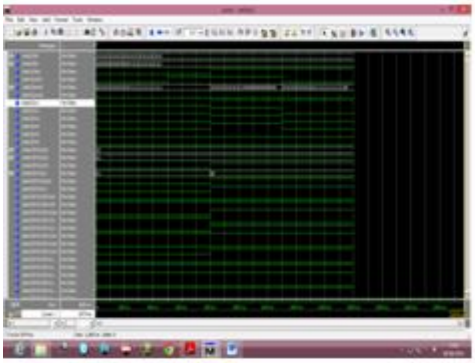
A. 8-ASC tiB



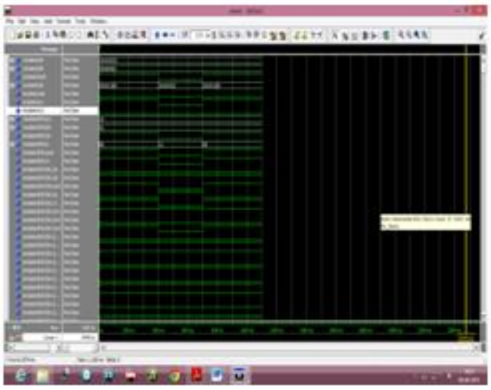
16-Bit CSA



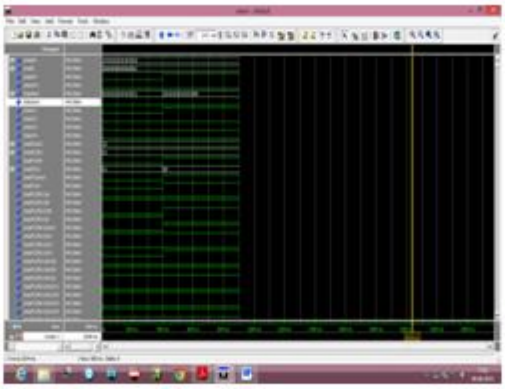
32-ASC tiB



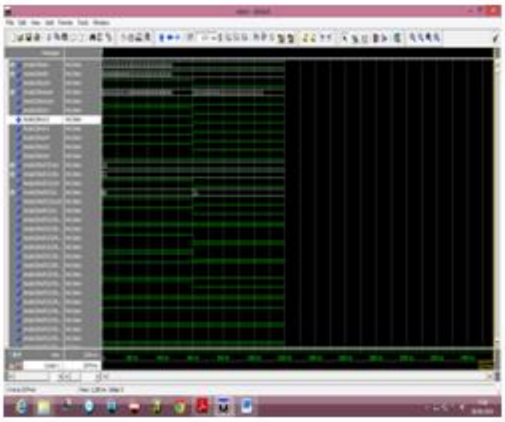
8-CEB ASC tiB



16-CEB ASC tiB



32-CEB ASC tiB



X. CONCLUSION

The area and delay of 8-bit, 16-bit, 32-bit, traditional SQRT CSA, SQRT CSA with BEC logic are evaluated and compared with the proposed SQRT CSA with BEC. It is clear from table- that, the proposed adder takes less delay and area when compared with SQRT CSA. It is also observed that in the proposed adder the reduction in area is very high with insignificant penalty in the delay when compared with traditional SQRT CSA.As the input length is progressed the area is decreased in the same proportion, but in the same proportion the delay penalty is not increased. Since the area in the proposed adder is very less, it is obvious that, the power consumption is also very less. Therefore this adder can be preferred for low power applications. In this work, a Square root Carry Select adder is designed using BEC. The design is synthesized by using Leonardo Spectrum. This synthesis tool gives area in terms of number of gates but not able to provide any information about the power. By using more advanced tools which can provide results of power, the designer can analyze that the design is efficient for low power application or not. And also validation can be done for this work. The adder can be designed for more number of input bits for observing the decrease in delay.

REFERENCES

[1] N. Weste and K. Eshragian, *Principles of CMOS VLSI Designs: A System Perspective*, 2nd ed., Addison-Wesley, 1985-1993.

[2] Morinaka, H., Makino, H., Nakase, Y. et. al, "*A 64 bit Carry Look-ahead CMOS adder using Modified Carry Select*". Cz/stoin Integrated Circuit Conference, 1995, pages 585-588

[3] Milos D. Ercegovac and Thomas Lang, "*Digital arthimetic,*" Morgan Kaufmann, Elsevier INC, 2004.

[4] W.Jeong and K.Roy, *"robust high-performance low power adder*",proc,of the Asia and South Pacific Design Automatin Conference,pp.503-506,2003

[5] D.C Chen, L. M. Guerra,E. H. Ng, M. Potkonjak, D.P. Schultz and J. M. Rabaey, "*An integrated system for rapid prototyping of high performance algorithm specific data paths*," in Proc. Application specific Array Processors, pp.134-148,Aug 1992.

[6] N.Weste and D. Harris, *CMOS VLSI Design. Reading*, MA: Addison Wesley, 2004.

[7] O. J. Bedrij, "*Carry-select adder*," IRE Trans. Electron. Comput.,pp.340–344, 1962. [2] 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.

[8] 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.

[9] 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.