International Journal of Electrical and Electronics Engineers
Vol. No.8 Issue 01, January-June 2016
www.arresearchpublication.com
IJEEE
ISSN 2321 - 2055

AN ENHANCED AREA AND POWER OPTIMIZED SQRT CSA

E.Srividya[1], V.Balaji[2]

[1]PG Scholar, VLSI &ES, [2]Asistant Professor, ECE, Vizag Institute of Technology, Dakamarri (India)

ABSTRACT

Carry Select Adder (CSLA) is one of the fastest adders use 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.

Keywords: SQRT CSA, Carry Propagation, Gate Level, Optimization, Latency, Worest Case Delay, Combination Path Delay.

I. INTRODUCTION

The challenge of the verifying a large design is growing exponentially. There is a need to define new methods that makes functional verification easy. Several strategies in the recent years have been proposed to achieve good functional verification with less effort. Recent advancement towards this goal is methodologies. The methodology defines a skeleton over which one can add flesh and skin to their requirements to achieve functional verification. DESIGN of area- and power-efficient high-speed data path logic systems are one of the most substantial areas of research in VLSI system design. 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 elementary adder is generated sequentially only after the previous bit position has been summed and a carry propagated into the next position. The report is organized as two major portions; first part is brief introduction and history of the functional verification of regular Carry select adder which tells about different advantages of Carry select adder and RCA architecture and in this Regular model, there is a drawback and in order to overcome that complexity, the modified architecture of CSLA has been designed. The importance of a fast, low-cost binary adder in a digital system is difficult to overestimate. Not only are adders used in every arithmetic operation, they are also needed for computing the physical address in virtually every memory fetch operation in most modern CPUs. Adders are also used in many other digital systems including telecommunications systems in places where a full-fledged CPU would be superfluous. Many styles of adders exist. Ripple adders are the smallest but also the slowest. More recently, carry-skip adders [1, 2, 3] are gaining popularity due to their high speed and relatively small size. Normally, in an N-bit carry-skip adder divided into a proper number of M-bit blocks [1, 4], a long-

range carry signal starts at a generic block Bi, rippling through some bits in that block, then skips some blocks, and ends in a block Bj. If the carry does not end at the LSB of Bj then rippling occurs in that block and an additional delay is needed to compute the valid sum bits. Carry-look-ahead and carry-select adders [1] are very fast but far larger and consume much more power than ripple or carry-skip adders. Two of the fastest known addition circuits are the Lynch-Swartzlander's [5] and Kantabutra's [6]. hybrid carry-look-ahead adders. They are based on the usage of a carry tree that produces carries intoappropriate bit positions without back propagation. In order to obtain the valid sum bits as soon as possible, in both Lynch-Swartzlander's and Kantabutra's adders the sum bits are computed by means of carry-select blocks, which are able to perform their operations in parallel with the carry-tree.

BASIC IDEA

The CSLA is used in many computational systems to alleviate theproblem of carry propagation delay by independently generating multiplecarries and then select a carry to generate the sum. However,the CSLA is not area efficient because it uses multiple pairs of RippleCarry Adders (RCA) to generate partial sum and carry by consideringcarry input Cin=0 and Cin=1, then the final sum and carry areselected by the multiplexers (mux).

The basic idea of this work is to use Binary to Excess-1 Converter(BEC) instead of RCA with Cin=1 in the regular CSLA to achievelower area and power consumption. The main advantage of thisBEC logic comes from the lesser number of logic gates than the n-bitFull Adder (FA) structure.

II. NEED FOR LOW POWER AND AREA EFFICIENT DESIGN

Design of area and powerefficient highspeed data path logicsystems are one of the most substantial areas of research in VLSIsystem design. In digital adders, the speed of addition is limited by thetime required to propagate a carry through the adder. The sum for eachbit position in an elementary adder is generated sequentially only afterthe previous bit position has been summed and a carry propagated intothe next position.

Thisadder plays a vital role in many data processing processors to perform fast arithmetic functions.Hence to resolve this issue, this adder has been developed to reduce the propagation delay for the carry to propagate to the next position.

Now another important point here is the evaluation of the carry select adder is compared with the proposed design as it has a more balanced delay and requires lower power andarea.

III. REGULAR CARRY SELECT ADDER

The Regular Carry Select Adder is represented in Figure 1.1. Basically this project is mainly targeted for data processing processors to perform fast arithmetic functions. The carry select adder comes in the category of conditional sum adder. Conditional sum adder works on some condition. Sum and carry are calculated by assuming input carry as 1 and 0 prior the input carry comes. When actual carry input arrives, the actual calculated values of sum and carry are selected using a multiplexer. The conventional carry select adder consists of k/2 bit adder for the lower half of the bits i.e. least significant bits and for the upper half i.e. most significant bits (MSB's) two k/2 bit adders. In MSB adders, one adder assumes carry input as one for performing addition and another assumes carry input as zero. The carry out calculated from the last stage i.e. least significant bit stage is used to select the actual calculated values of output carry and sum.

International Journal of Electrical and Electronics Engineers
Vol. No.8 Issue 01, January-June 2016
www.arresearchpublication.com
IJEEE
ISSN 2321 - 2055

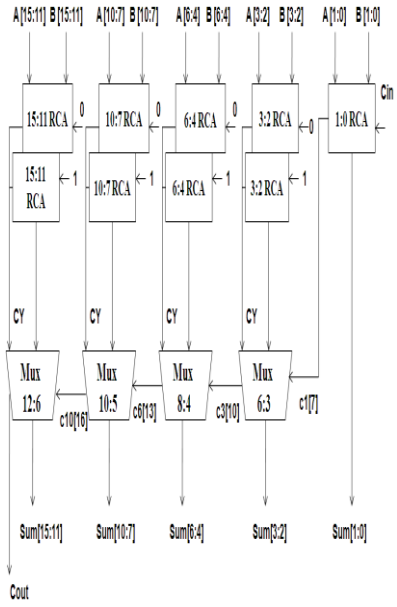

Fig1.1: Regular Carry Select Adder

Here in this design, the carry select adder is designed using Ripple carry adders and multiplexers. The design can be viewed as groups where the groups are internally designed using n-bit RCA and multiplexers.

Since this design uses multiple pairs of Ripple carry adders to generate partial sum and carry, it is not area efficient. Thus Binary to Excess-1 Converter is used (BEC) instead of RCA with Cin=1 in the regular CSLA to achievelower area and power consumption. The main advantage of thisBEC logic comes from the lesser number of logic gates than the n-bitFull Adder (FA) structure.

IV. MODIFIED CARRY SELECT ADDER

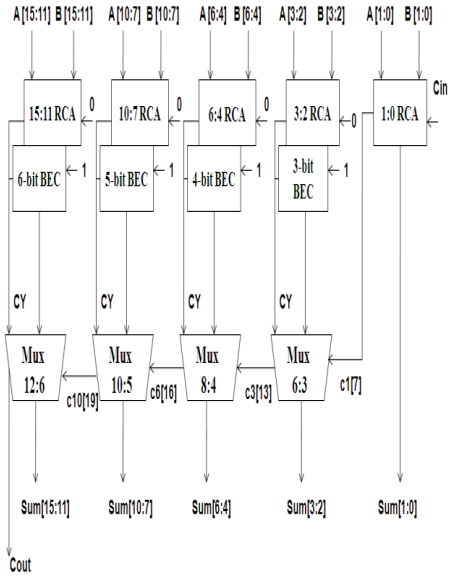

Fig 1.2: Modified Carry Select Adder

As stated above, the main idea of this work is to use BEC instead ofthe RCA with Cin=1 in order to reduce the area and power consumptionof the regular CSLA. To replace the n-bit RCA, an (n+1)-bit BECis required.

Finally, the performance of the two designs is evaluated in terms of area, power, delay and their products- area-delay product and power-delay product.

4 Delay and Area Evaluation of CSLA groups:

The structure of the 16-b regular SQRT CSLA is shown in the figure 3.5. Ithas five groups of different size RCA. The delay and area evaluation ofeach group are shown in Fig. 5, in which the numerals within [ ] specifythe delay values, e.g., sum2 requires 10 gate delays. The steps leadingto the evaluation are as follows.

1) The group2 has two sets of 2-b RCA. Based onthe consideration of delay values of Table 3.1, the arrival time ofselection input c1[time(t)= 7] of 6:3 mux is earlier than s3[t=8] and later than s2[t=6]. Thus, sum3[t=11]is summation of s3 and mux[t=3] and sum2[t=10] is summation of c1 and mux.

2) Except for group2, the arrival time of mux selection input is alwaysgreater than the arrival time of data outputs from the RCA's.Thus, the delay of group3 to group5 is determined, respectively asfollows:

{c6, sum [6:4]}=c3 [t=10] +mux

{c10, sum [10:7]}=c6 [t=13] +mux

{Cout, sum [15:11]}=c10 [t=16] +mux

3) The one set of 2-b RCA in group2 has 2 FA forCin=1 and theother set has 1 FA and 1 HA for Cin=0.

Based on the area countof Table 3.1, the total number of gate counts in group2 is determinedas follows:

Gate count =57 (HA+FA+Mux)

FA=39(3*13)

HA=6(1*6)

Mux=12(3*4)

4) Similarly, the estimated maximum delay and area of the othergroups in the regular SQRT CSLA are evaluated and listed.

The area and delay values of all the groups of square root CSLA are shown in the Table 3.2.

Group	Delay	Area
Group2	11	57
Group3	13	87
Group4	16	117
Group5	19	147

Table 3.2: Delay and area count of regular SQRT CSLA groups

BEC

BEC stands for Binary to Excess-1 Converter.

Design of CSLA with BEC: As stated above, the main idea of this work is to use BEC instead ofthe RCA with Cin=1 in order to reduce the area and

power consumptionof the regular CSLA. To replace the n-bit RCA, an (n+1)-bit BECis required. The structure and the function table of a 4-b BEC are shownin the figure 3.7 and Table 3.3 respectively.

International Journal of Electrical and Electronics Engineers
Vol. No.8 Issue 01, January-June 2016
www.arresearchpublication.com
IJEEE
ISSN 2321 - 2055

Structure of BEC:

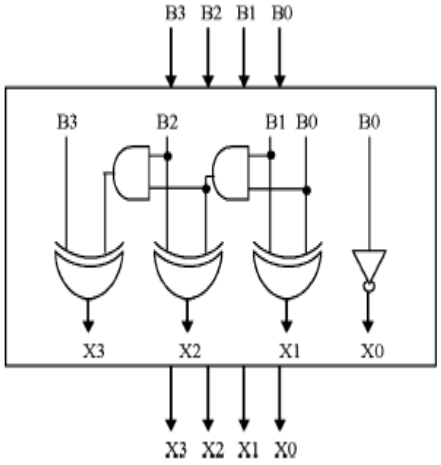

Fig : Structure of 4-bit BEC

B [3:0] (Binary inputs)	X[3:0] (Excess-1 outputs)
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

Table : Function table of BEC

The Boolean expressions of the 4-bit BEC are listed as below:

X0 = ~B0

X1 = B0 ^ B1

X2 = B2 ^ (B0 & B1)

X3 = B3 ^ (B0 & B1 & B2)

The figure 3.8 illustrates how the basic function of the CSLA is obtained byusing the 4-bit BEC together with the mux. One input of the 8:4 muxgets its input from B3 B2 B1 B0 and another input to the mux is the BEC

output. This produces the two possible partial results in parallel and the mux is used to select either the BEC output or the direct inputs according to the control signal Cin.

The importance of the BEC logics tems from the large silicon area reduction when the CSLA with large number of bits are designed.

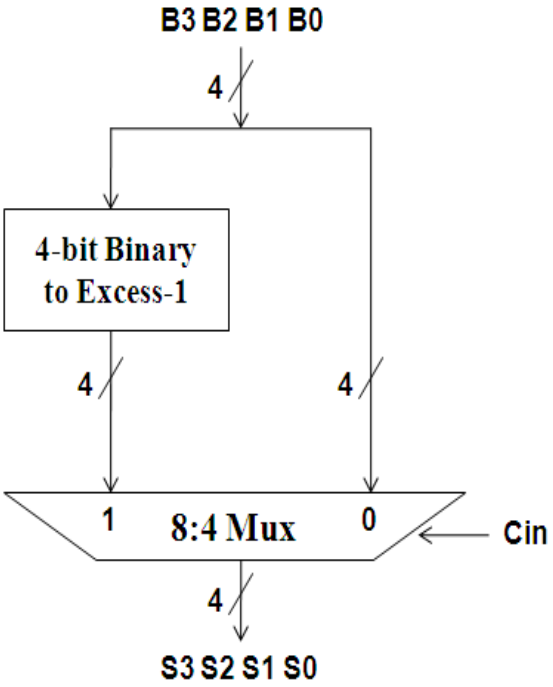

Fig : 4-b BEC with 8:4 Mux

Block diagram of CSLA with BEC:

The structure of the proposed 16-b SQRT CSLA using BEC for RCAwithCin=1 to optimize the area and power is shown in the figure 3.9.

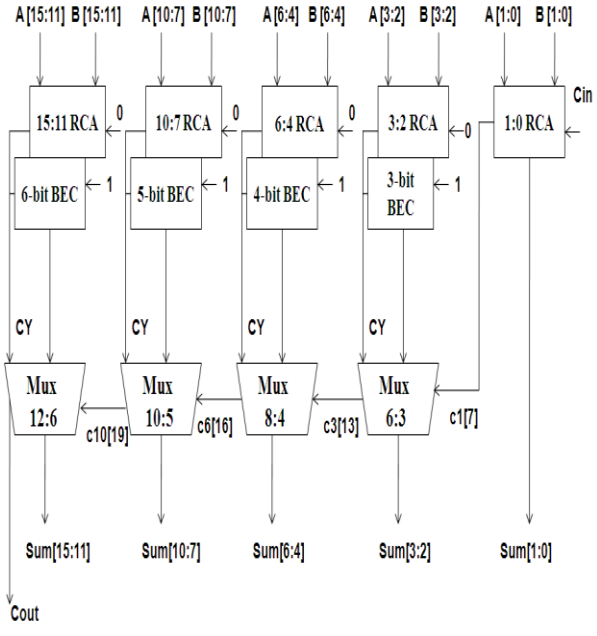

Fig Modified system (Modified 16-b SQRT CSLA)

Comparing the block diagram of the regular square root CSLA with the modified square root CSLA, it can be seen that the RCA with Cin=1 is replaced by BEC (binary to excess-1 converter). This is done to reduce the area consumption.

RESULT:

V. CONCLUSION

A simple approach is proposed in this paper to reduce the area andpower of SQRT CSLA architecture. The reduced number of gates ofthis work offers the great advantage in the reduction of area and also thetotal power. The compared results show that the modified SQRT CSLAhas a slightly larger delay (only 3.76%), but the area and power of the64-b modified SQRT CSLA are significantly reduced by 17.4% and15.4% respectively. The power delay product and also the area delayproduct of the proposed design show a decrease for 16-, 32-, and 64-bsizes which indicates the success of the method and not a mere tradeoffof delay for power and area. The modified CSLA architecture is thereforelow area, low power, simple and efficient for VLSI hardware implementation.

REFERENCES

[1]. O. J. Bedrij, “Carry-select adder,” IRE Trans. Electron. Comput., pp.340–344, 1962.
[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]. 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.
[4]. Y. He, C. H. Chang, and J. Gu, “An area efficient 64-bit square root carry-select adder for low power applications,” in Proc. IEEE Int. Symp. Circuits Syst., 2005, vol. 4, pp. 4082–4085.
[5]. Cadence, “Encounter user guide,” Version 6.2.4, March 2008.
[6]. www.uvmworld.org
[7]. www.testbench.in
[8]. www.ovmworld.org
[9]. www.systemVHDL.org