Area-Efficient 128-bit Carry Select Adder Architecture

B.Srinivasareddy , D.Manjularani

Abstract-- 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. 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 design in terms of area, power. The results analysis shows that the proposed CSLA structure is better than the regular SQRT CSLA.

***Index Terms*— Application specific integrated circuit (ASIC), area-efficient,CSLA, low power.**

I.INTRODUCTION

Design of area-eficient hight data path logic systems are one of the most important areas of research in VLSI system design. Design of high speed data processing processors ,adders are playing a major role. 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 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. Fastest adders are used to perform the fast arithmetic fuctions for high speed data processing processors.

Carry Select Adder (CSLA) is one of the fastest adders used in many data-processing processors to perform fast arithmetic functions. The CSLA is used in many computational systems to alleviate the problem of carry propagation delay by independently generating multiple carries and then select a carry to generate the sum [1]. However, the CSLA is not area efficient because it uses multiple pairs of RippleCarry 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).

In CSLA ,Use Binary to Excess-1 Converter (BEC) instead of RCA with Cin=0 in the regular CSLA to achieve lower area and power consumption [2]–[4]. Because BEC logic comes from the lesser number of logic gates than the n-bit Full Adder (FA) structure. The details of the BEC logic are discussed in Section III.

Delay and area evaluation methodology of the basic adder blocks disussed in Section II. the detailed structure and the function of the BEC logic are discussed in Section III. Section IV presents the architecture of the Regular CSLA of 128-bits. This SQRT CSLA has been developed using ripple carry adders and multiplexers. The architecture of the Modified SQRT 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 VII

II. BASIC ADDER BLOCKS

In this section we explained how to calculate delay and area theoretically. The AND, OR, and Inverter (AOI) implementation of an XOR gate is shown in Fig. 1. The gates between the dotted lines are performing the

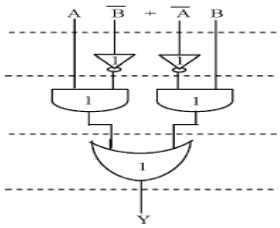

Fig .1 Delay and Area of XOR gate

operations in parallel and the numeric representation of each gate indicates the delay contributed by that gate. Basic adder block 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.We then add up the number of gates in the longest path of a logic block that contributes to the maximum delay.The area evaluation is done by counting the total number of AOI gates required for each logic block. Based on this approach, the CSLA adder blocks of 2:1 mux, Half Adder (HA), and FA are evaluated and listed in Table I.

Table 1:Delay and area evolution of CSLA

Adder blocks	Delay	Area
XOR	3	5
2:1 Mux	3	4
Half adder	3	6
Full adder	6	13

III. INTRODUCTION TO BEC

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 power consumption of the regular CSLA. To replace the n-bit RCA, an n + 1-bit BEC is required. A structure and the function table of a 5-bit BEC are shown in Fig. 2 and Table II, respectively. Fig.3 illustrates how the basic function of the CSLA is obtained by using the 5-bit BEC together with the mux.

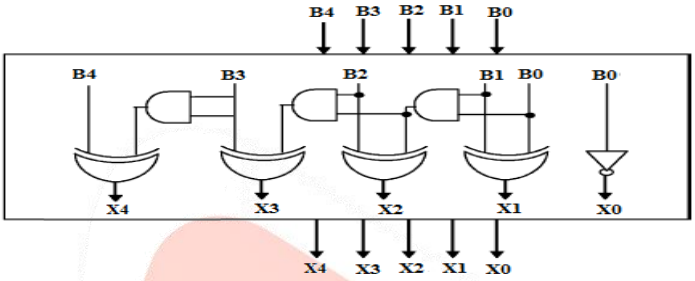

Fig. 2. 5-bit BEC.

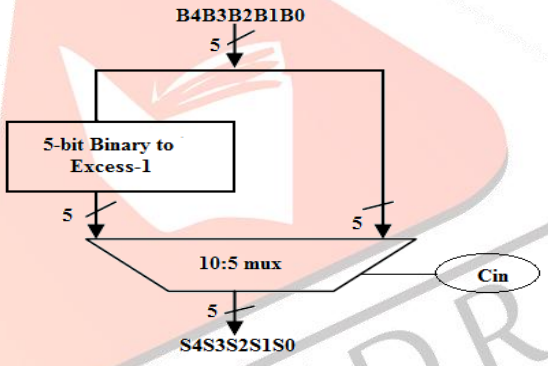

Fig .3 8 bit BEC with 10:5 mux

One input of the 16:8 mux gets as it input and another input of 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 Boolean expression of 5-bit BEC is (note the functional symbols ~ NOT, & AND,XOR).

X0 = ~B0
X1 = B0^B1
X2 = B2^ (B0 & B1)
X3 = B3^ (B0 & B1 & B2)
X4 = B4^ (B0 & B1 & B2 & B3)
X5 = B5^ (B0 & B1 & B2 & B3 & B4).

Table 2: BEC functional table

B[4:0]	X[4:0]
00000	00001
00001	00010
00010	00011
.	.
.	.
11110	11111
11111	00000

IV. ARCHITECTURE OF 128 BIT CSLA

A 16-bit carry select adder can be developed in two different sizes namely uniform block size and variable block size. Similarly 128-bit can also be developed in two modes of different block sizes. Ripple-carry adders are the simplest and most compact full adders, but their performance is limited by a carry that must propagate from the leastsignificant bit to the most-significant bit. The 128-bit CSLA can also be developed by using ripple carry adders. 128-bit CSLA has fourteen groups of different size RCA.Parlally each group exicution will start.Depending upon the carry propagation delay size of the group has declared.The speed of the CSLA is depening upon the carry propagation form one group to onether group. The structure of the 128-bit regular SQRT CSLA is shown in Fig. 4.

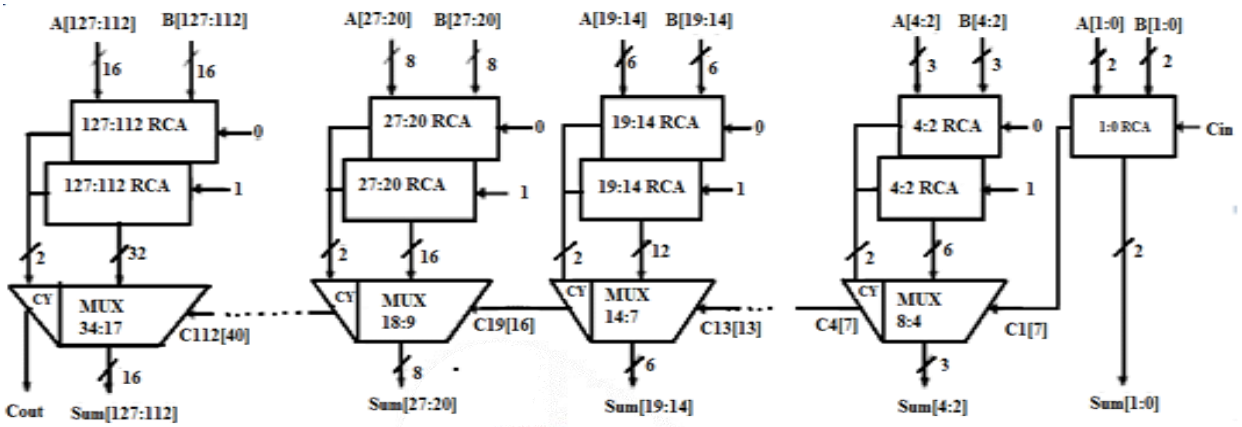

Fig .4 Architecture of Regular 128-b CSLA

Number of gates used in Regular 128-bit CSLA as shown in Table 3.Total 3764 gates used in Regular 128-bit CSLA.

Table3.
AREA COUNT OF REGULAR SQARE ROOT CSLA

GROUP	AREA
GROUP1	19
GROUP2	91
GROUP3	117
GROUP4	147
GROUP5	177
GROUP6	237
GROUP7	267
GROUP8	297
GROUP9	327
GROUP10	357
GROUP11	387
GROUP12	417
GROUP13	447
GROUP14	477

V. MODIFIED ARCHITECTURE OF 128 BIT CSLA

This architecture is similar to regular 128-bit SQRT CSLA, the only change is that, we replace RCA with Cin=1 among the two available RCAs in a group with a BEC. This BEC has a feature that it can perform the similar operation as that of the replaced RCA with Cin=1. Fig 5 shows the Modified block diagram of 128-bit SQRT CSLA. The number of bits required for BEC logic is 1 bit more than the RCA bits. The modified block diagram is also divided into various groups of variable sizes of bits with each group having the ripple carry adders, BEC and corresponding mux. As shown in the Fig.5, Group 0 contain one RCA only which is having input of lower significant bit and carry in bit and produces result of sum[1:0] and carry out which is acting as mux selection line for the next group, similarly the procedure continues for higher groups but they includes BEC logic instead of RCA with Cin=1.

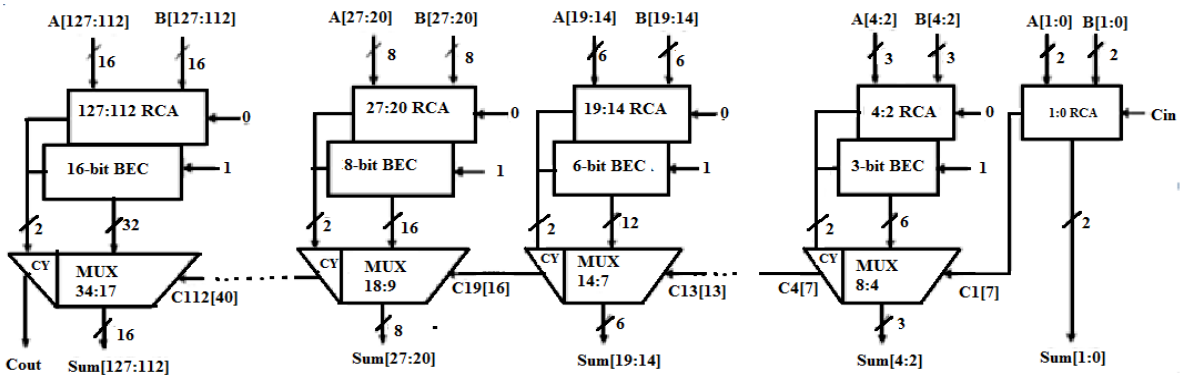

Fig .5 Modified128 bit SQRT CSLA

The number of gates used in each group is shown in Table 4. Total 2895 gates used in Regular 128-bit CSLA.Total 869 gates reduced in Modified SQRT CSLA comapared with Regular SQRT CSLA.

Based on the consideration of delay values, the arrival it can be seen that upto 16% area and upto 22% power reduced, Delay has incresed 2.43%

TableIV
AREA COUNT OF MODIFIED SQARE ROOT CSLA

GROUPS	AREA
GROUP1	19
GROUP2	70
GROUP3	89
GROUP4	112
GROUP5	148
GROUP6	181
GROUP7	204
GROUP8	227
GROUP9	250
GROUP10	273
GROUP11	296
GROUP12	319
GROUP13	342
GROUP14	365

VI.RESULT

The implemented design in this work has been implimented in Verilog-HDL langvage simulated using Modelsim. Regular and Modified are also simulated in Modelsim and corresponding results are compared. After simulation the different codes are synthesized using Cadence. The simulated files are imported into the synthesized tool and corresponding values of area and power are noted. for both the regular and modified CSLA.

Table V exhibits the simulation results of both the CSLA structures in terms of area and power. The area indicates the total cell area of the design and the total power is sum of the leakage power, internal power and switching power.

Table V
COMPARISON OF THE REGULAR AND MODIFIED SQRT CSLA

Word Size	TOTAL AREA(um^2)	TOTAL POWER(mW)
128-bit Regular SQRT CSLA	20471	17.45
128-bit Modified SQRT CSLA	17051	13.54

VII.CONCLUSION

A simple approach is proposed in this paper to reduce the area and power of CSLA architecture. The reduced number of gates of this work offers the great advantage in the reduction of area and also the total power. The compared results show that the modified CSLA has a slightly larger delay (only 3.76%), but the area and power of the 128-bit 16% ,22%. The power-delay product and also the area-delay product of the proposed design show a decrease for 128-bit sizes which indicates the success of the method and not a mere tradeoff of delay for power and area. The modified CSLA architecture is therefore, low area, low power, simple and efficient for VLSI hardware implementation. It would be interesting to test the design of the modified 256-bit SQRT CSLA.

ACKNOWLEDGMENT

B.Srinivasareddy would like to thank Mrs. D.Manjularani, Associate professor ECE Department who had been guiding throughout the project and supporting me in giving technical ideas about the paper and motivating me to complete the work effectively and successfully.

REFERENCES

[1] O. J. Bedrij, "Carry-select adder," *IRE Trans. Electron. Computes.*, 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] J. M. Rabaey, *Digtal Integrated Circuits—A Design Perspective*. Upper Saddle River, NJ: Prentice-Hall, 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] B. Ramkumar and Harish M Kittur," Low-Power and Area-Efficient Carry Select Adder" IEEE Trasactions on Very Large Scale Integration(VLSI) Systems, VOL. 20, NO. 2, Feb 2012.