Modified High Speed CSLA for Low Power Applications

Nageswar Rao Paleti[1] R. Veera Bhadraiah[2]
[1]P.G. Scholar [2]Assistant Professor
[1,2]Nimra Institute of Science and Technology Vijayawada, A.P. India

Abstract— *Here a system is designed based on the strategy of the functions of the arithmetic operation based phenomena in a well oriented fashion and also the and the performance of the processor based strategy in a well effective manner through which there is an reduced complexity followed by the increase in the speed of the system in a well respective strategy with respect to the adders of the carry select based strategy including CSLA based phenomena in a well oriented fashion respectively. Here the implementation of the CSLA based architecture in a well oriented aspect by which where the main aim of the system is a key factor oriented analysis with respect to the reduced power oriented strategy followed by the reduced power consumption based phenomena in a well efficient fashion respectively. Where there is a quite significant modification in the system based strategy in an analogous form by the takes place in the gate level based scenario in a well efficient fashion respectively. Here the modification takes place by the help of the 64 bit, 32 bit and 16 bit in a well oriented fashion and in an effective manner respectively. Experiments have been conducted on the present method and a lot of analysis is made and an accurate analysis is made where there is a measurement of the performance based strategy followed by the outcome of the system in a well oriented fashion towards the entire system respectively. There is a lot of advancement takes place in the system in a well efficient manner respectively. Here in the present method oriented strategy in a well efficient manner there is an evaluation of the performance of the system takes place in the system in a well effective manner by the terms of the strategy oriented parameters in a well oriented fashion respectively and some of them includes the area, power and reduced complexity based parameters in a well oriented fashion respectively.*

Keywords— CSLA, ASIC, Area Efficiency, Low Power.

I. INTRODUCTION

In the systems based on the computation oriented strategy in a well effective manner by the implementation of the architecture oriented with CSLA based strategy in a well respective fashion takes place in the system based implementation respectively. Where there is a delay due to the strategy of the propagation based phenomena in a well respective fashion and also the problem due to the computational complexity oriented with the system based aspect in a well oriented fashion respectively. Where there is a generation of the carries of the multiple based phenomena followed by the sum of the generation of the carry select based strategy in a well effective manner respectively [1][2]. Here the strategy related to the aspect of the phenomena oriented with CSLA based analysis there is a system in the design oriented strategy by which generation of the adders based on the carry oriented ripple based phenomena followed by the carry followed by the sum based partial generation in a well oriented fashion respectively[3]. Here there is a huge research with respect to the strategy of the present design oriented method in which there is an implementation of the converter based on the binary to excess 1 from the BEC based phenomena in a well oriented fashion respectively [4].

II. BLOCK DIAGRAM

There is a huge challenge for the present method in which the present designed method is implemented in such a way that it must be accurate in terms of the analysis of the previous methods drawbacks in a well efficient manner followed by the control of the degraded performance of the previous method in a well oriented aspect respectively. In this paper a method is designed with a well efficient framework oriented strategy in which it is efficiently implemented by the help of the improvement in the performance based strategy followed by the entire system based outcome in a well oriented aspect respectively. Here the implementation of the present method is shown in the figure in the form of the block diagram and is explained in an elaborative fashion respectively. Here the implementation of the present method is shown to be effective and efficient in terms of the performance improvement followed by the accurate system based outcome in a well oriented fashion respectively [5][6].

III. METHODOLOGY

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 operations in parallel and the numeric representation of each gate indicates the delay contributed by that gate. The delay and area evaluation methodology 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.

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 4-b 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 4-bit BEC together with the mux. One input of the 8:4 mux gets as it input (B3, B2, B1, and B0) 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 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 (note the functional symbols ~ NOT, & AND, ^XOR)

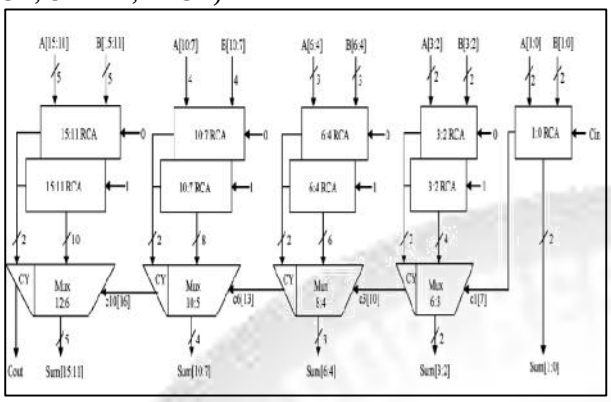


Fig. 1: Regular 16-b SQRT CSLA

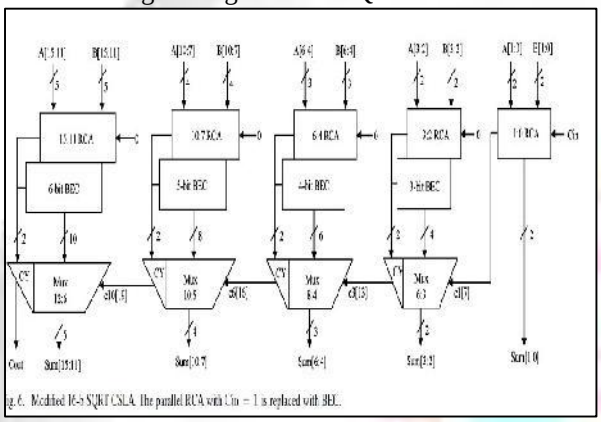


Fig. 2: Modified 16-b SQRT CSLA

TABLE I
DELAY AND AREA COUNT OF THE BASIC BLOCKS OF CSLA

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

TABLE II
FUNCTION TABLE OF THE 4-b BEC

B[3:0]	X[3:0]
0000	0001
0001	0010
⋮	⋮
1110	1111
1111	0000

IV. SOFTWARE SIMULATION AND RESULTS

The design proposed in this paper has been developed using Verilog-HDL and synthesized in Cadence RTL compiler using typical libraries of TSMC 0.18 um technology. The synthesized Verilog net list and their respective design constraints file (SDC) are imported to Cadence SoC Encounter and are used to generate automated layout from standard cells and placement and routing [7]. Parasitic extraction is performed using Encounter's Native *RC* extraction tool and the extracted parasitic *RC* (SPEF format) is back annotated to Common Timing Engine in Encounter platform for static timing analysis. For each word size of the adder, the same value changed dump (VCD) file is generated for all possible input conditions and imported the same to Cadence Encounter Power Analysis to perform the power simulations. The similar design flow is followed for both the regular and modified SQRT CSLA.

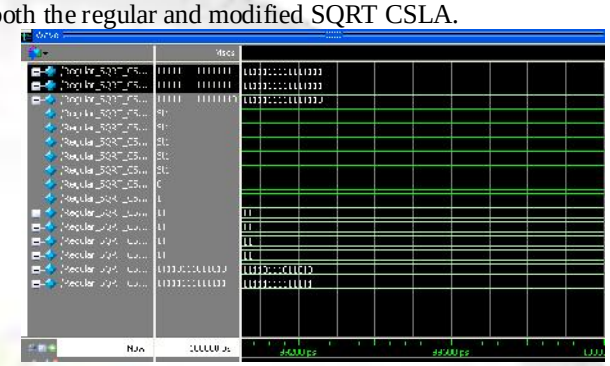


Fig. 4: CSA EXISTING 16 BIT

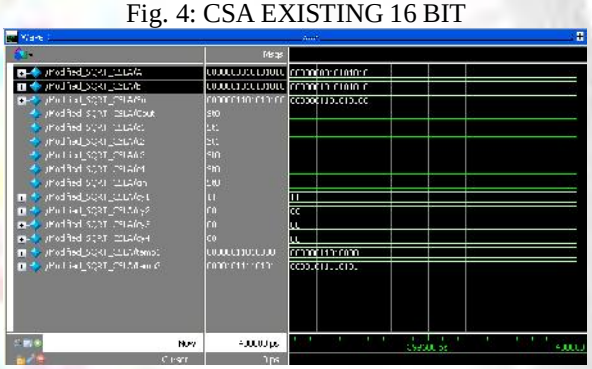
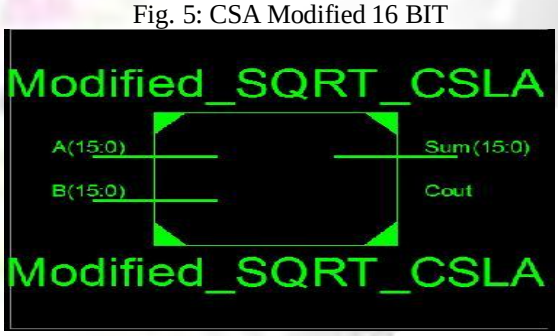


Fig. 5: CSA Modified 16 BIT



V. CONCLUSION

In the paper a method is designed with a powerful technique where there is a lot of analysis takes place in the system in which accurate analysis in terms of the improvement in the performance followed by the outcome of the entire system in a well oriented fashion respectively. A simple approach is proposed in this paper to reduce the area and power of SQRT 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 SQRT CSLA has a slightly larger delay (only 3.76%), but the area and power of the 64-b modified SQRT CSLA are

significantly reduced by 17.4% and 15.4% respectively. The power-delay product and also the area-delay product of the proposed design show a decrease for 16-, 32-, and 64-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 128-b SQRT CSLA

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] 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.
[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.
[5] J. M. Rabaey, Digtal Integrated Circuits—A Design Perspective. Upper Saddle River, NJ: Prentice-Hall, 2001.
[6] Y. He, C. H. Chang, and J. Gu, "An area efficient 64-bit square root carry-select adder for lowpower applications," in Proc. IEEE Int. Symp. Circuits Syst., 2005, vol. 4, pp. 4082–4085.
[7] Cadence, "Encounter user guide," Version 6.2.4, March 2008.
[8] Ramkumar, B. and Harish M Kittur, "Low Power and Area Efficient Carry Select Adder", IEEE Transactions on Very Large Scale Integration (VLSI) Systems, pp.1-5, 2012.
[9] He, Y. Chang, C. H. and Gu, J. "An Area Efficient 64-Bit Square Root Carry-Select Adder for Low Power Applications," in Proc. IEEE Int. Symp. Circuits Syst., Vol.4, pp. 4082–4085, 2005.
[10] Padma Devi, Ashima Girdher and Balwinder Singh "Improved Carry Select Adder with Reduced Area and Low Power Consumption," International Journal of Computer Applications, Vol.3, No.4, pp. 14-18, 1998.
[11] Akhilesh Tyagi, "A Reduced-Area Scheme for Carry-Select Adders," IEEE Transactions on Computers, Vol.42, No.10, pp.1163-1170, 1993.