Area Efficient Carry Select Adder with Low Power

S.Balaprasad[#1], M.Jeyalakshmi[#2]

[1]*PG Student,M.E VLSI Design, Department of Electronics and Communication Engineering ,SBM College of Engineering and Technology,Dindigul,India.*

[2] *Assistant Professor, Department of Electronics and Communication Engineering ,SBM College of Engineering and Technology,Dindigul,India.*

ABSTRACT: ***Carry SeLect Adder (CSLA) is known to be the fastest adder among the Conventional adder structures. Carry select adders provides a good compromise between cost, area and performance among carry propagation adders. Due to its less complex structure, there is still scope for obtaining better design Carry SeLect Adder (CSLA) architecture designs for power optimization combined with better performance. The conventional CSLA architecture uses dual RCAs which has large area consuming.Proposed technique eliminates all the redundant logic operations present in the conventional CSLA and proposed a new logic formulation for CSLA. In the proposed scheme, the Carry Select (CS) operation is scheduled before the calculation of final-sum, which is different from the conventional approach. Bit patterns of two anticipating carry words (corresponding to cin = 0 and 1) and fixed cin bits are used for logic optimization of CS and generation units. Due to its small carry-output delay,the proposed CSLA design is a good candidate for SQuare-RooT (SQRT) CSLA.***

Keywords— ***Adder, Area efficient, Boolean logic.***

I. INTRODUCTION

Addition usually impacts widely the overall Performance of digital systems and an arithmetic function. In electronic applications adders are most widely used. Applications where these are used are multipliers, DSP to execute various algorithms like FFT, FIR and ITR. In microprocessors, millions of instructions per second are performed. So, speed of operation is the most important constraint. 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 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. However, the CSLA 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 existing modified SQRT CSLA is to use Binary to Excess-1 Converter (BEC) instead of RCA with Cin=1 in the regular CSLA to achieve lower area and power consumption with slightly increase in the delay. The basic idea of the proposed architecture is that which replaces the BEC by Modified Logic formulation method.

In this paper, an area-efficient carry select adder by sharing the modified logic term is proposed. After Boolean simplification, it can remove the duplicated adder cells in the conventional carry select adder. The multiplexer is used to select the correct output according to its previous carry-out signal.

This paper is organized as follows: In section II conventional CSLA is shown, section III deals with proposed CSLA design, section IV explains Regular SQRT CSLA.Results are analyzed in section V and section VI concludes.

II.CONVENTIONAL CSLA

The carry ripple adder is constructed by cascading each single-bit full-adder. In the carry ripple adder, each full-adder starts its computation till previous carry-out signal is ready. Therefore, the critical path delay in a carry ripple adder is determined by its carry-out propagation path. For an N-bit full-adder as illustrated in Fig. 1, the critical path is N-bit carry propagation path in the full-adders. As the bit number N increases, the delay time of carry ripple adder will increase accordingly in a linear way.

In order to improve the shortcoming of carry ripple adder to remove the linear dependency between computation delay time and input word length, carry select adder is presented .When the actual carry input is ready, either the result of carry "0" path or the result of carry "1" path is selected by the multiplexer according to its carry input value.

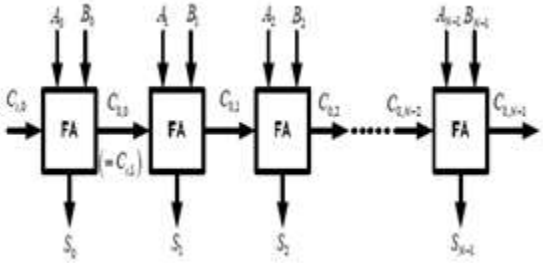

Fig. 1 N bit carry ripple adder by single N bit adder

The conventional CSLA is based on the logic formulation and its structure is shown in Fig. 2(a). It consists of one HSG unit, one FSG unit, one CG unit, and one CS unit. The CG unit is composed of two CGs (CG0 and CG1) corresponding to input-carry '0' and '1'. The HSG receives two n-bit operands (A and B) and generate half-sum word s0 and half-carry word c0 of width n bits each. Both CG0 and CG1 receive s0 and c0 from the HSG unit and generate two n-bit full-carry words c01 and c11 corresponding to input-carry '0' and '1', respectively.The logic diagram of the HSG unit is shown in Fig. 2(b). The logic circuits of CG0 and CG1 are optimized to take advantage of the fixed input-carry bits. The optimized designs of CG0 and CG1 are shown in Fig. 2(c) and (d), respectively.The CS unit selects one final carry word from the two carry words available at its input line using the control signal cin. It selects c01 when cin = 0; otherwise, it selects c11. Fig 2(e)and f shows the gate design of CS and Full sum generator unit.

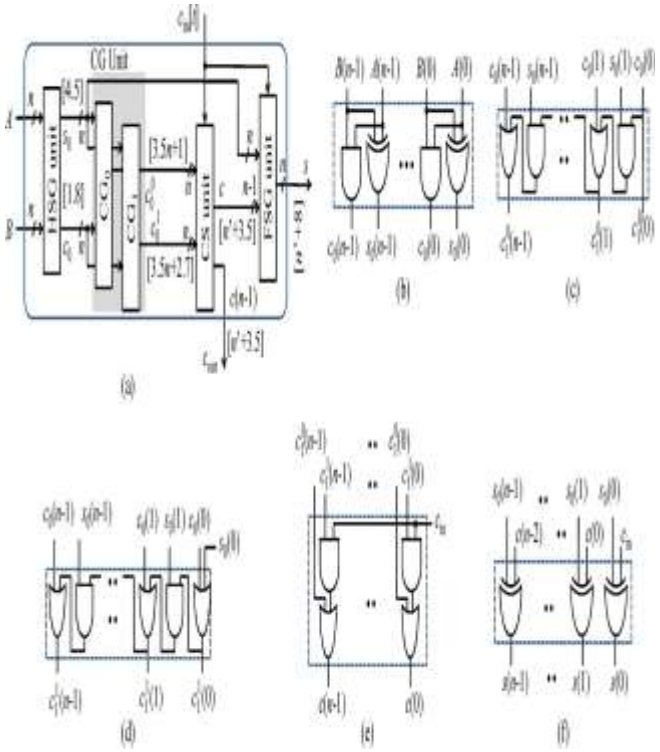

Figure 2 (a) Existing CS Adder Design (b) Gate-Level Design of the HSG (c) Gate-Level Optimized Design (d) Gate-Level Optimized Design (e) Gate-Level Design of the CS Unit. (f) Gate-Level Design of the Final-sum Generation (FSG) Unit.

III.PROPOSED CSLA DESIGN

Regular Carry select adder is designed by using RCA & multiplexer. It is cascading of RCAs (RCA is cascading of full adders). One RCA with carry input Cin=0, and other RCA with Cin=1. Design of 16-bit Regular CSLA is done by using this technique. Area count (gate count) is done by AND, OR, INVERTER (AOI) implementation.

Gate count = FA + HA + MUX

(Number of AND, OR, INVERTERs present in FA (full adder), HA (half adder) & multiplexer).

The proposed CSLA is based on the modified logic formulation and its structure is shown in Fig. 3(a). It consists of one HSG unit, one FSG unit, one CG unit, and one CS unit. The CG unit will generate

carry corresponding to input-carry '0' and '1'. The HSG receives two n-bit operands (a and b) and generate half-sum word s0 and half-carry word c0 of width n bits each.CG unit receives s0 and c0(0) from the HSG unit and generate two n-bit full-carry words corresponding to input-carry '0' and '1', respectively. The logic diagram of the proposed CSLA using AND-OR-INVERT is shown in Fig. 3(b). The logic circuits of Cin are optimized to take advantage of the fixed input-carry bits. The CS unit selects one carry word from the carry words available at its input line using the control signal cin and sends signal to CG unit according to Cin.

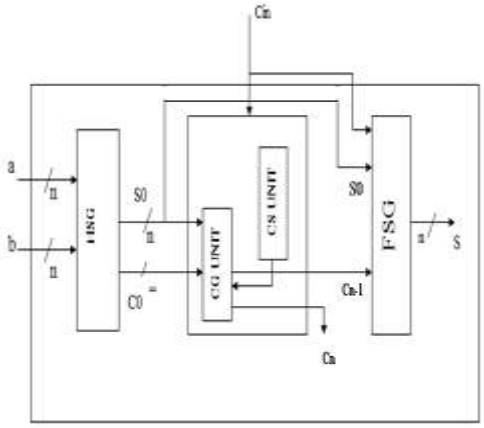

Fig.3(a). Proposed carry select adder design, where n is the input operand bit-width.

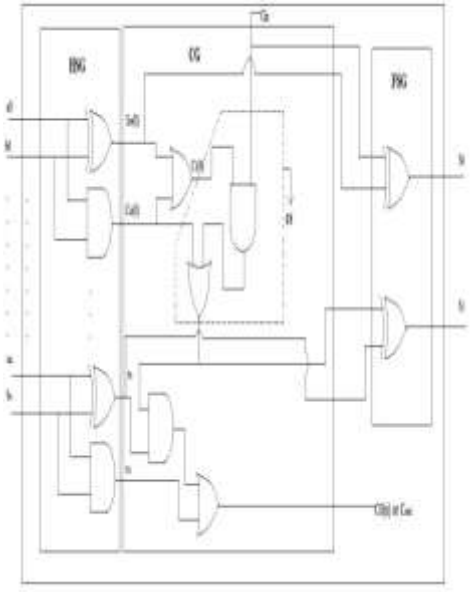

Fig. 3(b). Logic diagram of proposed CSLA using AND-OR-INVERT

As compared with the Modified Carry Select adder, the proposed CSLA is little bit faster, and the speed is faster than the Regular CSLA.Modified FIR filter using Multi Chip Module(MCM) is shown in Fig.4(a).The proposed CSLA design using AND-OR-INVERT with MUX is shown in Fig.4(b). The delay time in our proposed adder design is also proportional to the bit number N; however, the delay time of multiplexer is shorter than that of full adder. The Proposed CSLA is Area efficient & low power,and the speed is higher than the Regular CSLA.

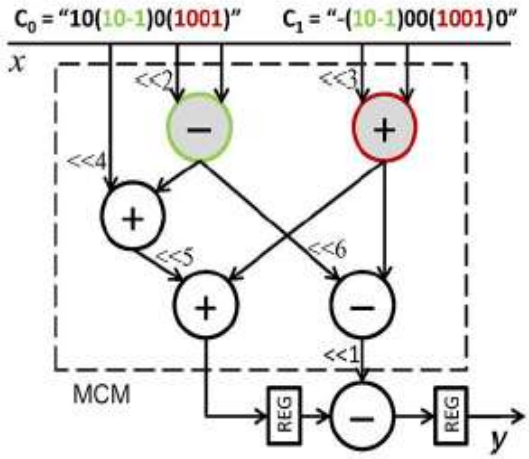

Fig. 4(a). Modified FIR Filter Using MCM.

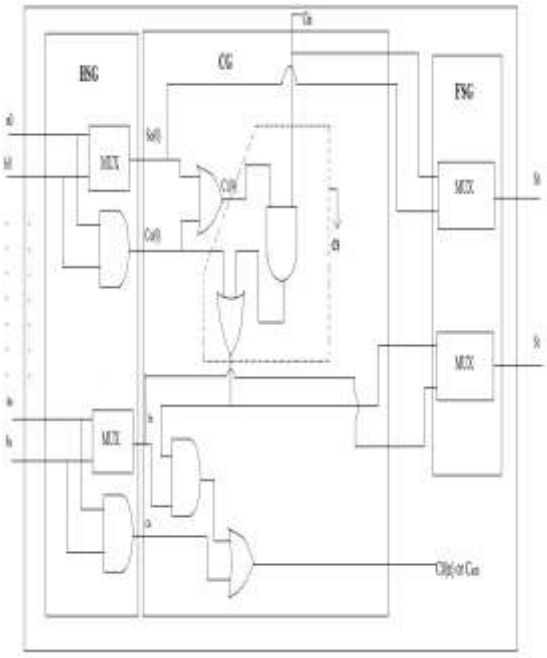

Fig. 4(b). Logic diagram of proposed CSLA using AND-OR-INVERT with MUX.

IV.REGULAR SQRT-CSLA

The multipath carry propagation feature of the CSLA is fully exploited in the SQRT-CSLA,which is composed of a chain of CSLAs. CSLAs of increasing size are used in the SQRT-CSLA to extract the maximum concurrence in the carry propagation path. Using the SQRT-CSLA design, large-size adders are implemented with significantly less delay than a single-stage CSLA of same size. However, carry propagation delay between the CSLA stages of SQRT-CSLA is critical for the overall adder delay.

Due to early generation of output-carry with multipath carry propagation feature, the proposed CSLA design is more favorable than the existing CSLA designs for area–delay efficient implementation of SQRT-CSLA. A 16-bit SQRT-CSLA design using the proposed CSLA is shown in Fig.5.To demonstrate the advantage of the proposed CSLA design in SQRT-CSLA, we have estimated the area and delay of SQRTCSLA using the proposed CSLA design for 16 bit. Compared with SQRT-CSLA designs of , the proposed SQRTCSLA design, respectively, involves less ADP, on average, for different bit-widths. Compared with the CBL-based CSLA of , the proposed CSLA design has marginally less ADP. However, in the CBL-based CSLA, delay increases at a much higher rate than the proposed CSLA design for higher bit widths.

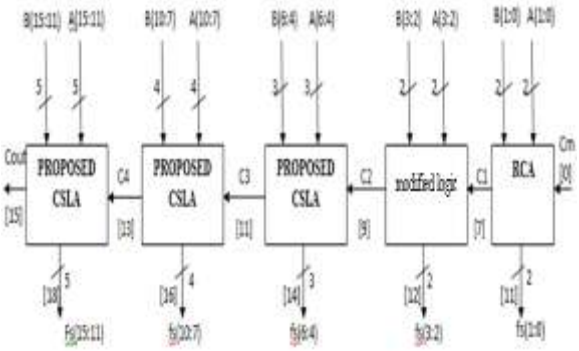

Fig.5 16 bit-SQRT CSLA

V.RESULTS

The area-efficient carry select adder achieves an outstanding performance in power consumption. Power consumption can be greatly saved in our proposed area-efficient carry select adder because we only need XOR gate and as well as AND gate and OR gate in each carry-out operation .Fig.6 shows the output of CSLA using AOI.Here we have simulated various Regular CSLA, modified CSLAs & CLA. The Xilinx ISE 8.1i software is used for synthesising the adders, & Modelsim6.4a is used to compile & simulate to verify the VHDL code.

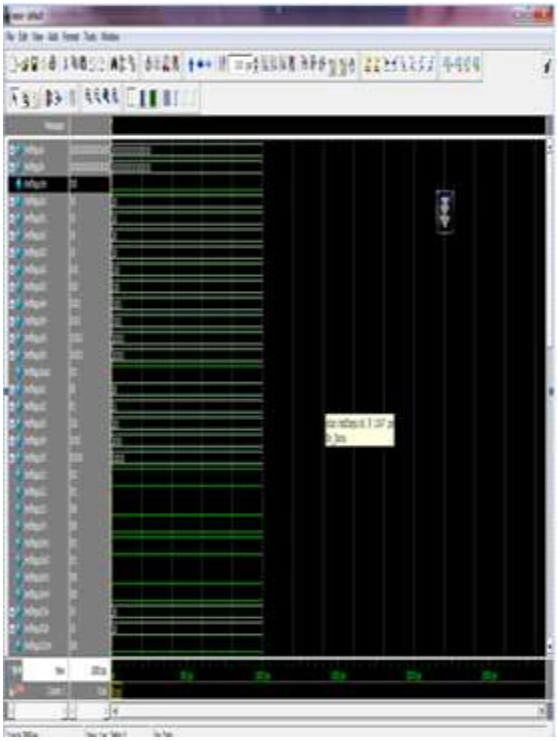

Fig.6 Output of CSLA.

As the input bit number increases, the slope of power consumption increase in the conventional carry select adder would be larger than that in our proposed design.Fig.7(a),7(b) shows the simulation result for gate count , delay. We can find out that the PDP of our proposed design is smaller as compare with the conventional carry select adder and carry ripple adder design. The difference of PDP between these three designs is small in the case of the smaller input bit number. However, as the input bits increases, the slope of power consumption increment in the conventional carry select adder would be larger than that of the proposed design. Our proposed design can compute the addition function more efficiently by means of logic circuit sharing and partial parallel computation architecture retaining; therefore, the power saving ratio in our design would be much higher than the ratio of speed sacrificeAs the input bit number of the conventional carry select adder increases to, the power consumption in the conventional carry select adder will be larger than that in our proposed area-efficient carry select adder.

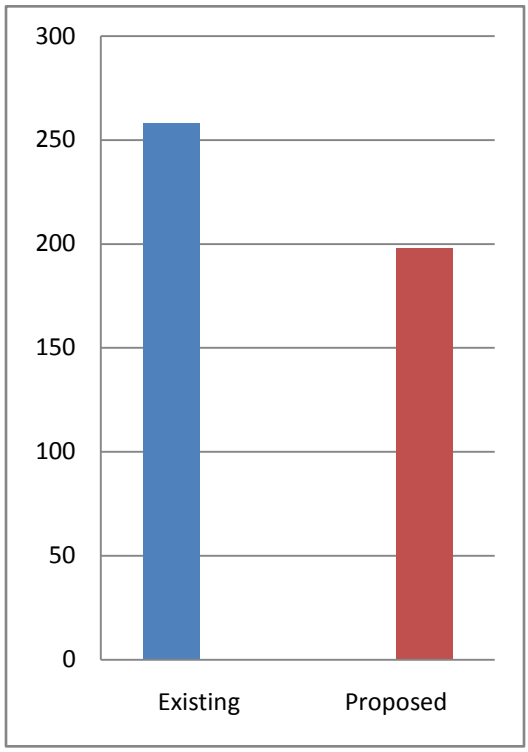

Fig.7(a) Simulation result for Gate count.

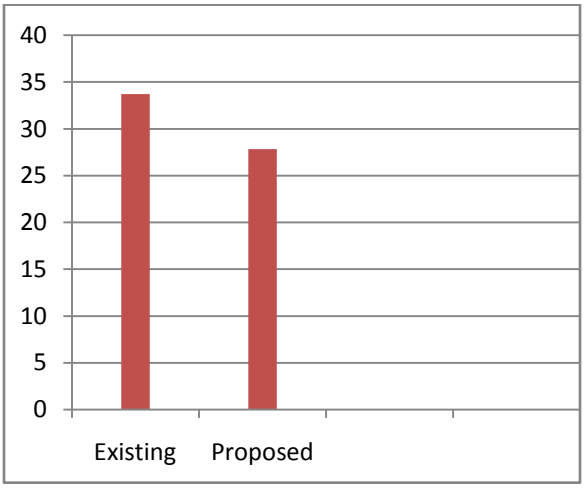

Fig.7(b) Simulation result of Delay.

VI.CONCLUSION

In this paper, simulation results of the Carry Select Adder, modified CSLA with MUX is estimated. The Xilinx ISE 8.1i software is used for synthesising the project, & Modelsim6.4a is used to compile & simulate to verify the Verilog code.

REFERENCES

Journal Papers:

[1]. Basant Kumar Mohanty, "Area-Delay Power Efficient Carry Select Adder",IEEE Transactions On Circuits And Systems—Ii: Express Briefs, Vol. 61, No. 6, June 2014.

[2]. 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.

[3] J. M. Rabaey, Digital Integrated Circuits," IEEE Trans. on VLSI Systems, 2003.

[4] Ramkumar, B. and Harish M Kittur, (2011) 'Low Power and Area Efficient Carry Select Adder', IEEE Transactions on Very Large Scale Integration (VLSI) Systems, pp.1-5.

Chapters in Books:

[5]. B. Ramkumar and Harish M. Kittur "Low-Power and Area-Efficient Carry Select Adder", IEEE transactions on very large Scale integration (VLSI) systems, vol. 20, pp.371-375, February 2012.

[6]. Sara deep Singh and Dilip Kumar "Design of Area and Power Efficient Modified Carry Select Adder", International Journal of Computer Applications, vol.33, pp. 14-18, November 2011.

[7]. O. Bedrij, "Carry Select Adder," IRE Trans. on Electronic Computers, vol. EC-11, pp. 340- 346, 1962.

[8]. Mr. C. S. Manikandababu " An Efficient Carry Select Adder architecture for VLSI hardware implementation", IJMIE, vol.2, pp.610-622, May 2012.

[9]. T. Y. Ceiang and M. J. Hsiao, "Carry-select adder using single ripple carry adder", vol. 34, No. 22, pp. 2101–2103, Oct. 1998.

[10] O. Bedrij, "Carry Select Adder," IRE Trans. on Electronic Computers,Vol. EC-11, pp. 340-346, 1962.

[11] Ramkumar, B. Kittur, H.M. and Kannan, P (2010) 'ASIC Implementation of Modified Faster Carry Save Adder', Eur. J. Sci. Res., Vol.42, No.1, pp.53–58.