ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 6, June 2016

Design of APD Efficient CSLA

Mugdha Godbole

M.E Student (VLSI & Embedded System), Dept. of E & TC, Dhole Patil College of Engineering, SPPU, Pune, India

ABSTRACT: Many types of adders available to perform fast operation in digital signal processing. Carry Select Adder is high speed device used for the fast computation. In developing era the key contributing factors are faster arithmetic unit, low power and low area arithmetic units are needed. Binary to Excess-1 converter is used to the modified carry select adder (CSLA). In the proposed architecture scheme, some new technique used which is different from conventional approach. Using optimized logic unit efficient CSLA design is obtained. Recently proposed Binary to Excess I Converter based CSLA design involves significantly more delay and area than the recently proposed CSLA. The newly design proposed CSLA is a best platform for square-root (SQRT) CSLA.

KEYWORDS: Adders, Carry Select Adder, Xilinx 14.5, Low power design, ALU.

I. Introduction

Low power, high performance, and area efficient VLSI systems are increasingly used in mobile and portable devices, standard receivers, and some biomedical instrumentation. Adder is the main component of an ALU. A DSP system involves many adders. For performance improvement efficient adders are involve in DSP system. RCA uses a simple design, but main factor in the adder is delay. A conventional carry select adder is an double RCA configuration. This configuration generates a sum and output carry either 0 or 1 corresponding the (Cin = 0 and 1). In comparison of CSLA and RCA, CSLA has less delay. because of double use of RCA this design is not acceptable. Some solutions described to avoid double use of RCA in design.

Kim and Kim [4] developed the system using a Ripple Carry Adder with one add-one circuit. Using Multiplexer the add-one circuit is designed. Y. He, C. H. Chang, and J. Gu. [5] describe a large width adders with comparatively less delay using Square root CSLA, Where CS adders with increasing size are connected in a proper structure. The main important objective of design is to provide parallel path for forward carry which reduces adder delay . Ramkumar and Kittur [6] introduce the new technology that is Binary to Excess-1 Converter based CSLA. It consists of less logic resources as compared to conventional CSLA at the cost of higher delay. In [7] The Common Boolean logic based CSLA design includes significantly less logic resource is needed as compare to conventional CSLA at the cost of longer Carry propagation delay, which is almost equal to RCA. To address this issue, adder based on Common Boolean Logic based SQRT CSLA is proposed in [8].

However, Binary to Excess I Converter based square-root CSLA [6] requires less logic resource and delay than the Common Boolean Logic-based Square root CSLA [8]. In short some availability of few redundant operations in the logic optimization which is available in the formulation, whereas adder delay depends on data dependence.

II. PROPOSED SYSTEM

The APD efficient proposed Carry select adder is based on the logic given in the equation is expressed with the help of block diagram in Fig. 1.

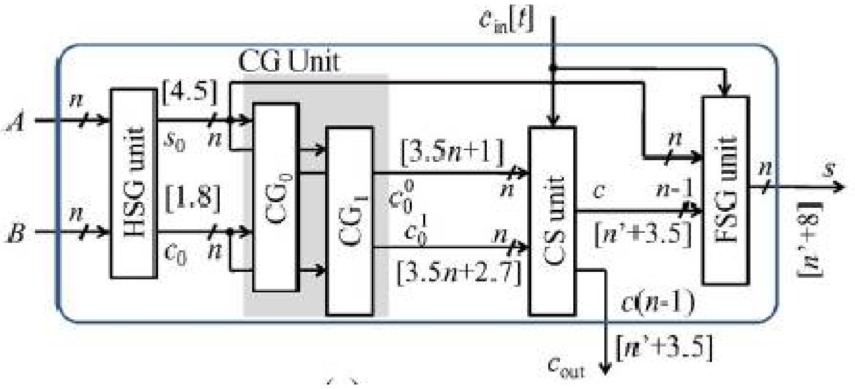


Fig.. 1. APD Efficient Proposed Carry Select Adder Design

It consists of one Half Sum Generator unit(HSG), one Full Sum Generator unit(FSG), one Carry Generate unit(CG), and one Carry Select unit(CS). The Carry Generator unit is made up of CG0 for input-carry '0' and CG1 for input carry '1'. The Half Sum Generator (HSG) unit takes two n bit inputs A and B both have same bit length and generate half-sum word and carry word. The width of the same is n bits. the of HSG unit CG0 and CG1 which generate two n-bit full-carry words Carry01 for input carry 0 and Carry11 for input carry 1 . The Fig 2 below shows logic circuit of the Half Sum Generator unit(HSG).

S0(i)= A(i) XOR b(i), C0(i)= A(i) AND b(i)..........................(1)

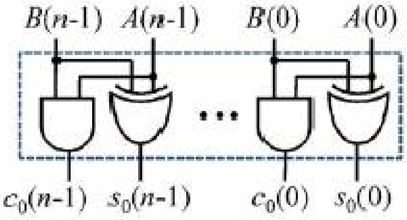


Fig. 2. Gate level design of the Half Sum Generator(HSG)

The logic circuits of CG0 and CG1 are optimized to take benefit of the fixed input-carry bits. The optimized designs of input carry 0 and 1 are shown in Fig. 3 and 4 respectively.

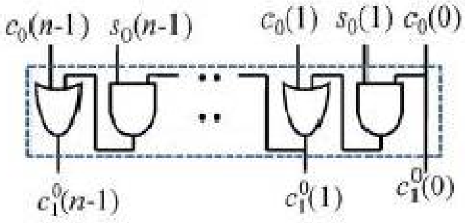


Fig. 3. Gate-level optimized design of (CG0) for 0 input carry

DOI: 10.15680/IJIRCCE.2016. 0406357

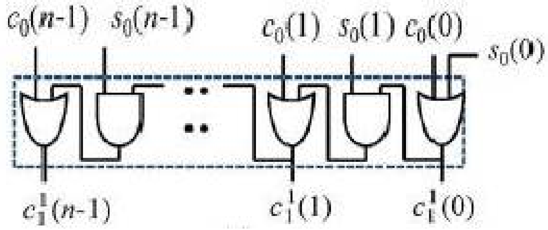


Fig. 4. Gate-level implemented optimized design of (CG1) for 1 input-carry

The Carry Select unit selects one final carry word from the two carry words available at its input line using the control signal. It selects Carry01 when Cin = 0; otherwise, it selects Carry11. The Carry Select unit can be implemented using an n bit 2:l MUX. If Carry01(i) = ‘1’, then Carry11(i) = 1, This feature is used for optimization of logic of the Carry Select unit. The gate level implemented optimized design of the Carry Select(CS) unit is shown in Fig. 5

Cout= C(n-1)..(2)

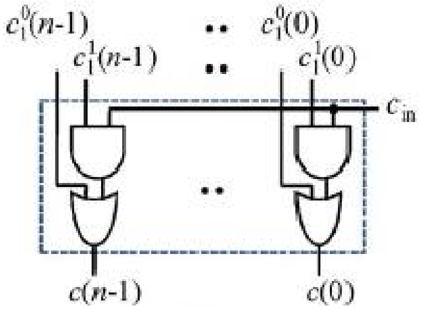


Fig. 5. CS Design

The final carry bit Carry is obtained from the Carry Select unit (CS). The Most Significant Bit (MSB) of Carry sent to output carry as Carryout, and (n − 1) Least Significant Bits(LSB) are XOR with (n − 1) MSBs of half-sum (Sum0) to obtain (n − 1) MSBs of Sum (s). as [shown in Fig. 6, FSG Design. In order too obtain LSB of sum, LSB of Sum0 is XOR with Carry$_{in}$

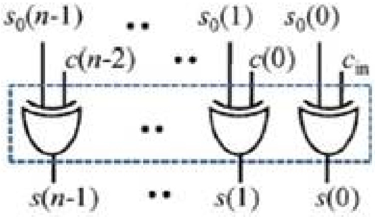


Fig. 6. FSG design

III. System Block Diagram

The proposed Carry Select Adder this adder is design using other sub block like CG0 (input carry 0),CG1(input carry 1),CS(carry select),FSG(final sum generator),HFG(half sum generator) .

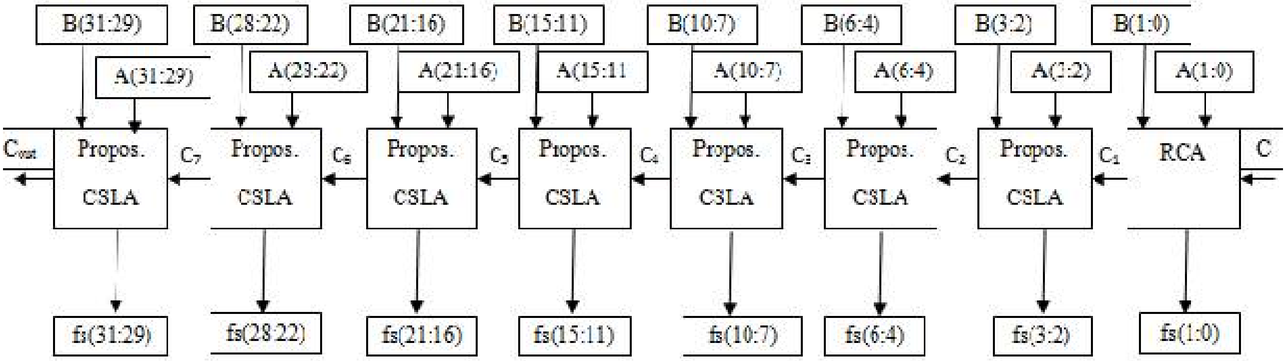


Fig. 7. Proposed SQRT-CSLA for n = 32. All intermediate and output signals are labeled with delay

first design all the sub block using logic gates, then connect all the sub block properly And design Proposed Carry Select Adder. Using Carry Select Adder design 2bit Carry Select Adder,3 bit Carry Select Adder, 4bit Carry Select Adder, 5bit Carry Select Adder and also design one 2 bit Ripple carry adder. Connect all Carry Select Adder and ripple carry adder according to the take two input of n-bits .

All output of Carry Select Adder connect according to block diagram and generate final output .Firstly two input of n bit is given to Half Sum Gate unit and output of HSG is given to CG0 ,CG1 and Full Sum gate unit. The multipath (many different path) carry propagation feature of the carry select adder is fully exploited in the square-root carry select adder, which is composed of CSLA. Carry select adders of increasing size are used in the square-root carry select adder (SQRT-CSLA) to extract the maximum concurrence in the carry propagation path. Using the square-root carry select adder (SQRT-CSLA) design, Large size adders are implemented Using the square-root carry select adder (SQRT-CSLA) design. Due to fast generation of Cout that is output-carry , the recently proposed CSLA design is more favorable in the APD design than the existing CSLA design. This is mainly important for APD efficient implementation of SQRT-CSLA. A 32-bit (SQRT-CSLA) design using the proposed CSLA where the 2-bit RCA, and CSLA such as 2-bit, 3-bit, 4-bit , and 5-bit , 6-bit , 7-bit , and again use of 3-bit CSLA are used. If we design 8 bit adder then remaining 5 bit are not considered in calculation which increase delay.

IV. RESULTS

The Comparison of the three logic techniques such as Binary to excess I converter, Common Boolean Logic, Proposed Square root Carry Select Adder. The table contains the delay, number of LUT's , Bonded IOB's.

TABLE 1: Comparison Of Delay

Design	Width (n)	Delay (ns)
SQRT-CSLA (CONV)[5]	16	5.61
	32	6.56
	64	8.37
SQRT-CSLA (BEC)[6]	16	5.96
	32	7.64
	64	10.18
SQRT-CSLA (CBL)[7]	16	10.45
	32	18.72
	64	35.10
Proposed SQRT-CSLA	**16**	**2.48**
	32	**5.98**
	64	**8.248**
	128	**14.349**

TABLE 2: Device Utilization Summary

Logic Utilization used	Bit			
	16	32	64	128
Number of Slice LUTs	28	66	138	298
Number of Fully Used LUT-FF Pairs	0	0	0	0
Number of Bonded IOBs	67	98	194	386

V. SIMULATION RESULTS

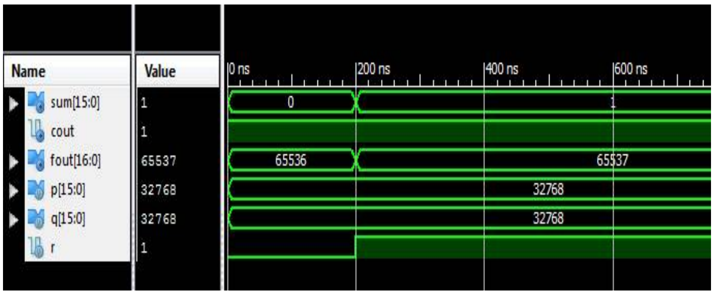


Fig. 8. Addition of 16 bit adder

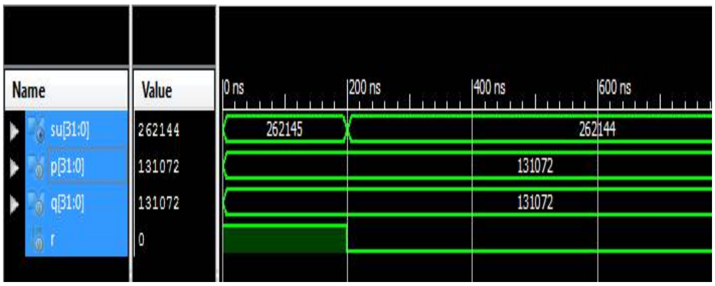


Fig. 9. Addition of 32 bit adder

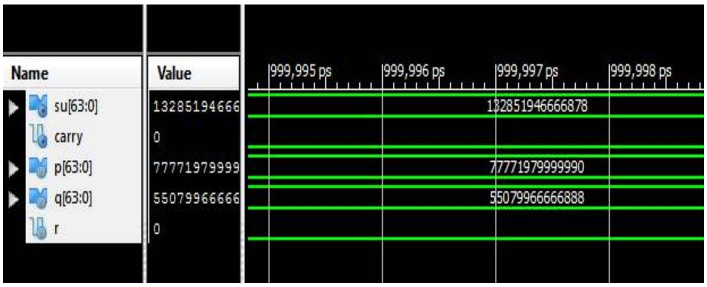
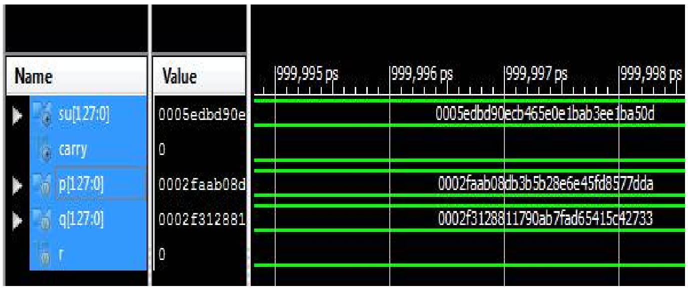


Fig. 10. Addition of 64 bit adder

DOI: 10.15680/IJIRCCE.2016. 0406357



Name	Value
su[127:0]	0005edbd90e
carry	0
p[127:0]	0002faab08d
q[127:0]	0002f312881
r	0

999,995 ps
999,996 ps
999,997 ps
999,998 ps
0005edbd90ecb465e0e1bab3ee1ba50d
0002faab08db3b5b28e6e45fd8577dda
0002f3128811790ab7fad65415c42733

Fig. 11. Addition of 128 bit adder

VI. CONCLUSION

A good method is describe in this to reduce efficiently APD for SQRT CSLA. The proposed the CSLA is good technique as compare to the other technique describe in the paper. Due to the small output delay of carry , the proposed Carry Select Adder design is a good for the Square Root adder.

REFERENCES

1. B. Ramkumar and H.M. Kittur, “Low-power and area-efficient carry-select adder,” IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 20, no. 2, pp. 371–375, Feb. 2012.
2. I.-C. Wey, C.-C. Ho, Y.-S. Lin, and C. C. Peng, “An area-efficient carry select adder design by sharing the common Boolean logic term,” in Proc. IMECS, pp. 1–4,2012.
3. S. Manju and V. Sornagopal, “An efficient SQRT architecture of carry select adder design by common Boolean logic ” in Proc. VLSI ICEVENT, pp. 1–5,2013.
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. A. P. Chandrakasan, N. Verma, and D. C. Daly, “Ultralow-power electronics for biomedical applications,” Annu. Rev. Biomed. Eng., vol. 10, pp. 247–274, Aug. 2008.
6. K. K. Parhi, VLSI Digital Signal Processing. New York, NY, USA:Wiley,1998.
7.O. J. Bedrij, “Carry-select adder,” IRE Trans. Electron. Comput., vol. EC-11, no. 3, pp. 340–344, Jun. 1962.
8. Y. He, C. H. Chang, and J. Gu, “An area-efficient 64-bit square root carry select adder for low power application,” in Proc. IEEE Int. Symp. Circuits Syst., vol. 4, pp. 4082–4085,2005.
9.B. Parhami, Computer Arithmetic: Algorithms and Hardware Designs,2nd ed. New York, NY, USA: Oxford Univ. Press,2010.