International Journal of Advance Research in Science and Engineering
Vol. No.5, Special Issue No. (02), March 2016
www.ijarse.com
IJARSE
ISSN 2319 - 8354

IMPLEMENTATION OF AREA, POWER, DELAY EFFICIENT CARRY-SELECT ADDER

Mugdha Godbole[1], Kanchan Pujari[2]

[1] *DPCOE Pune MEVLSI & Embedded System Pune University,(India)*

[2] *Department of E & TC, DPCOE Pune,(India)*

ABSTRACT

Carry Select Adder (CSLA) is one of the high speed adders used in many computational systems to perform fast arithmetic operations .Due to the rapidly growing mobile industry not only the faster arithmetic unit but also less area and low power arithmetic units are needed. The modified CSLA architecture has developed using Binary to Excess-1 converter (BEC). 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 C_{in} = 0 and 1) and fixed C_{in} bits are used for logic optimization of CS and generation units. An efficient CSLA design is obtained using optimized logic units. The proposed CSLA design involves significantly less area and delay than the recently proposed BEC-based CSLA. Due to the small carry-output delay, the proposed CSLA design is a good candidate for square-root (SQRT) CSLA.

Keywords: Carry Select Adder, arithmetic unit, low-power design

I. INTRODUCTION

Low-Power, area-efficient, and high-performance VLSI systems are increasingly used in portable and mobile devices, multistandard receivers, and biomedical instrumentation[1] . An adder is the main component of an arithmetic unit. A complex digital signal processing (DSP) system involvesseveral adders. An efficient adder design essentially improves the performance of a complex DSP system. A ripple carry adder (RCA) uses a simple design, but carry propagation delay (CPD) is the main concern in this adder. Carry look-ahead and carry select (CS) methods have been suggested to reduce the CPD of adders. A conventional carry select adder (CSLA) is an RCA–RCA configuration that generates a pair of sum words and output carry bits corresponding the anticipated input-carry (Cin = 0 and 1) and selects one out of each pair for final-sum and final-output-carry [3]. A conventional CSLA has less CPD than an RCA, but the design is not attractive since it uses a dual RCA. Few attempts have been made to avoid dual use of RCA in CSLA design. Kim and Kim [4] used one RCA and one add-one circuit instead of two RCAs, where the add-one circuit is implemented using amultiplexer (MUX). He et al. [5] proposed a square-root (SQRT)-CSLA to implement large bit-width adders with less delay. In a SQRT CSLA, CSLAs with increasing size are connected in a cascadingstructure. The main objective of SQRT-CSLA design is to provide a parallel path for carry propagation that helps to reduce the overall adder delay. Ramkumar and Kittur [6] suggested a binary to BEC-based CSLA. The BEC-based CSLA involves less logic resources than the conventional CSLA, but it has marginally higher delay. A CSLA based on common

Boolean logic (CBL) is also proposed in [7] and [8]. The CBL-based CSLA of [7] involves significantly less logic resource than the conventional CSLA but it has longer CPD, which is almost equal to that of the RCA. To overcome this problem, a SQRT-CSLA based on CBL was proposed in [8]. However, the CBL-based SQRTCSLA design of [8] requires more logic resource and delay than the BEC-based SQRT-CSLA of [6]. We observe that logic optimization largely depends on availability of redundant operations in the formulation, whereas adder delay mainly depends on data dependence. In the existing designs, logic is optimized without giving any consideration to the data dependence. we have proposed a logic formulation for the CSLA. The main contribution in this brief are logic formulation based on data dependence and optimized carry generator (CG) and CS design.

II. PROPOSED SYSTEM

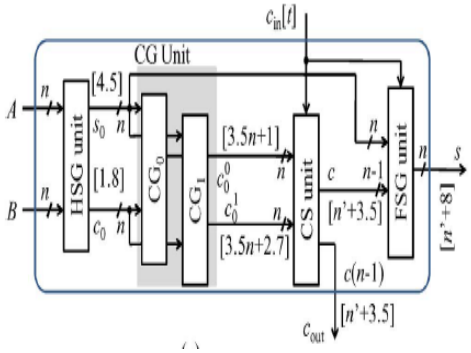

Fig 2.1 Proposed CS Adder Design

The proposed CSLA is based on the logic formulation given in (2a)–(2g), and itsstructure is shown in Fig. 2.1. 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 C_0 of width n bits each. Both CG_0 and CG_1 receive S_0 and C_0 from the HSG unit and generate two n-bit full-carry words C_{01} and C_{11} corresponding to input-carry '0' and '1', respectively. The logic diagram of the HSG unit is shown in Fig. 2.2

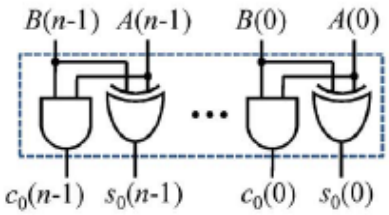

Fig 2.2 Gate level design of the HSG

The logic circuits of CG_0 and CG_1 are optimized to take advantage of the fixed input-carry bits. The optimized designs of CG_0 and CG_1 are shown in Fig. 2.3 and 2.4 respectively.

International Journal of Advance Research in Science and Engineering
Vol. No.5, Special Issue No. (02), March 2016
www.ijarse.com
IJARSE
ISSN 2319 - 8354

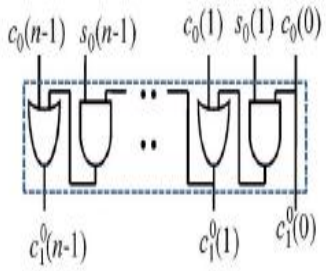

Fig 2.3 Gate-level optimized design of (CG0) for input-carry = 0

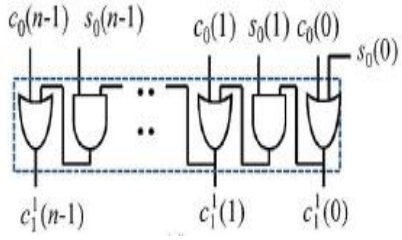

Fig 2.4 Gate-level optimized design of (CG1) for input-carry = 1

The CS unit selects one final carry word from the two carry words available at its input line using the control signal C_{in}. It selects C_{01} when C_{in} = 0; otherwise, it selects C_{11}. The CS unit can be implemented using an n-bit 2-to-l MUX. However, we find from the truth table of the CS unit that carry words C_{01} and C_{11} follow a specific bit pattern. If $C_{01}(i)$ = '1', then $C_{11}(i)$ = 1, irrespective of $S_0(i)$ and $C_0(i)$, for $0 \leq i \leq n-1$. This feature is used for logic optimization of the CS unit. The optimized design of the CS unit is shown in Fig. 2.5

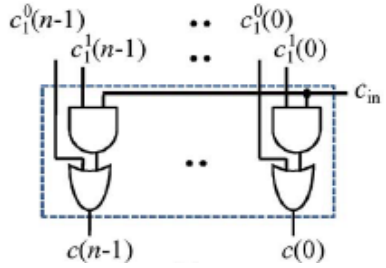

Fig 2.5 Gate level CS design

The final carry word c is obtained from the CS unit. The MSB of c is sent to output as Cout,and (n − 1) LSBs are XOR with (n − 1) MSBs of half-sum (S_0) in the FSG [shown in Fig. 2.6 to obtain(n − 1) MSBs of final-sum (s). The LSB of s0 is XOR with C_{in} to obtain the LSB of s.

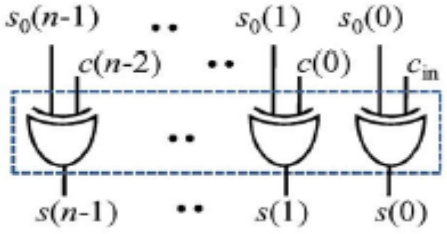

Fig 2.6 Gate Level Design of FSG

International Journal of Advance Research in Science and Engineering
Vol. No.5, Special Issue No. (02), March 2016
www.ijarse.com
IJARSE
ISSN 2319 - 8354

III. SYSTEM BLOCK DIAGRAM

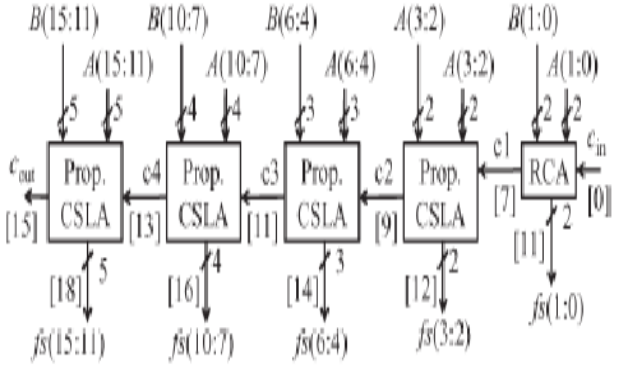

Fig. 3.1 Proposed SQRT-CSLA for *n* = 16. All intermediate and output signals are labeled with delay

The proposed CS Adder this adder is design using other sub block like CG0(input carry 0),CG1(input carry 1) ,CS(carry select),FSG(final sum gate),HFG(half sum gate) .first design all the sub block using logic gates, then connect all the sub block properly And design Proposed CS Adder. Using CS Adder design 2bit CS adder,3 bit CS adder, 4bit CS adder, 5bit CS adder and also design one RCA Adder of 2bit.connect all CS adder and RCA according to the take two input of n-bits ,this input given the fig according to circuit diagram. All output of CS Adder connect according to diagram and generate final output .Firstly two input A and B of n bit is given to HSG unit and output of HSG is given to CG0 ,CG0 and FSG .the all CS Adder connect each other but in first 2bit of input is given tothe RCA and remaining bit given to the CS Adders and generate the final output. 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 where the 2-bit RCA, 2-bit CSLA, 3-bit CSLA, 4-bit CSLA, and 5-bit CSLA are used.

IV. RESULTS

Sr No.	Bit	Delay(ns)
1	16	2.48
2	32	5.03

International Journal of Advance Research in Science and Engineering
Vol. No.5, Special Issue No. (02), March 2016
www.ijarse.com
IJARSE
ISSN 2319 - 8354

V. SIMULATION RESULTS

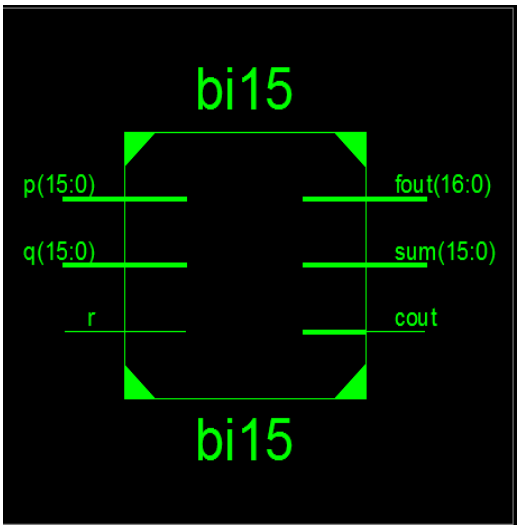

Fig. 5.1 RTL Schematic of 16 Bit SORT CSLA

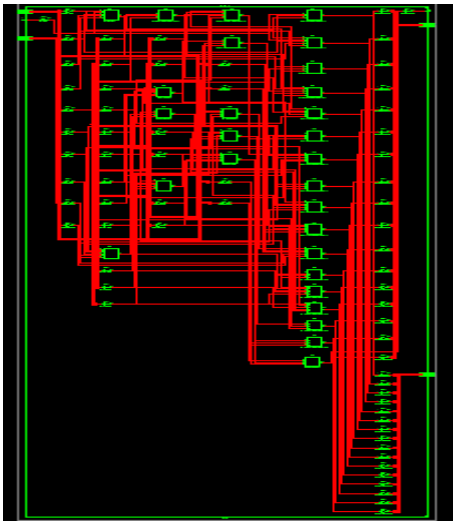

Fig. 5.2 Technology Schematic of 16 Bit SORT CSLA

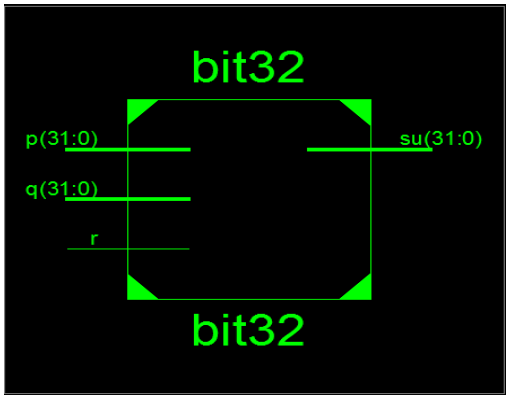

Fig. 5.3 RTL Schematic of 32 Bit SORT CSLA

International Journal of Advance Research in Science and Engineering
Vol. No.5, Special Issue No. (02), March 2016
www.ijarse.com
IJARSE
ISSN 2319 - 8354

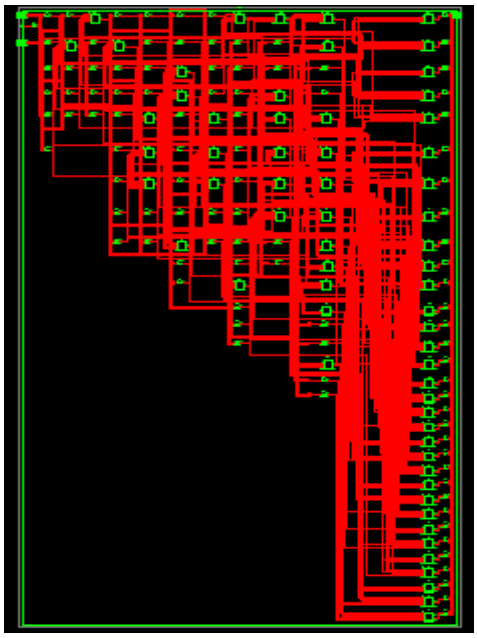

Fig. 5.4 Technology Schematic of 32 Bit SORT CSLA

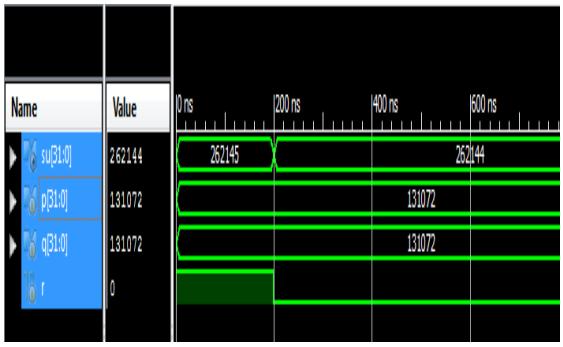

Fig. 5.5 Addition of 32 bit adder

VI. APPLICATION

- Arithmetic logic units
- High Speed multiplications
- Advanced microprocessor design
- Digital signal process

VII. ADVANTAGES

- Low Power Consumption
- Low area
- Low Complexity
- More Speed Compare Regular CSLA

VII. CONCLUSION

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 power. The modified CSLA architecture is therefore, low area, low power, simple and efficient for VLSI hardware implementation. The proposed CSLA design involves significantly less area and delay than the recently proposed BEC-based CSLA. Due to the small carry output delay, the proposed CSLA design is a good candidate for the SQRT adder.

REFERENCE

[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.