IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 5, Issue 1, January 2016

Review Paper on VLSI Architecture for Carry Select Adder

K. Pitambar Patra[1], Sambit Patnaik[2], Swapna Subudhiray[3], Janmejaya Samal[4]

Asst. Prof. Einstein Academy of Technology & Management, Bhubaneswar[1, 2, 3, 4]

Abstract: A multiplier is one of the key hardware blocks in most digital and high performance systems such as FIR filters, digital signal processors and microprocessors etc. With advances in technology, many researchers have tried and are trying to design multipliers which offer either of the following- high speed, low power consumption, regularity of layout and hence less area or even combination of them in multiplier. However area and speed are two conflicting constraints. So improving speed results always in larger areas. So here we try to find out the best trade off solution among the both of them. While comparing the adders we found out that Ripple Carry Adder had a smaller area while having lesser speed, in contrast to which Carry Select Adders are high speed but possess a larger area. And a Carry Look Ahead Adder is in between the spectrum having a proper tradeoff between time and area complexities.

Keywords: Ripple Carry Adder, Carry Select Adder (CSLA), Booth Encoder (BEC).

I. INTRODUCTION

Addition usually impacts widely the overall performance of digital systems and a crucial 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 IIR. Wherever concept of multiplication comes adders come in to the picture. As we know millions of instructions per second are performed in microprocessors. So, speed of operation is the most important constraint to be considered while designing multipliers.

Improved adders generate carries simultaneously [1]. These adders employ the principle that the carry from each bit position may be generated independently as an explicit function of all the less significant addend and augend bits. However, because of the inherent limitations in available components, the constructors of simultaneous carry-generation adders are not always practical.

Multiplication is a fundamental operation in most signal processing algorithms. Multipliers have large area, long latency and consume considerable power. Therefore low-power multiplier design has been an important part in low-power VLSI system design. There has been extensive work on low-power multipliers at technology, physical, circuit and logic levels. A system's performance is generally determined by the performance of the multiplier because the multiplier is generally the slowest element in the system. Furthermore, it is generally the most area consuming. Hence, optimizing the speed and area of the multiplier is a major design issue. However, area and speed are usually conflicting constraints so that improving speed results mostly in larger areas. As a result, a whole spectrum of multipliers with different area- speed constraints has been designed with fully parallel.

The rest of the paper is organized as follows. In section 2, a brief about ripple carry adder, carry select adder and carry skip is given. In section 3 carry look head adder is introduced along with partitioning methodology. Also a new architecture with clock sharing is introduced. Section 4 provides the simulation results obtained. Section 5 provides the conclusion and future scope obtained.

II. ADDITION ALGORITHM

○ Ripple Carry Adder

This procedure is an asynchronous addition used in the first electronic computers. Addition of binary numbers of several digits is accomplished in the same manner as that of decimal numbers.

Concatenating the N full adders forms N bit Ripple carry adder. In this carry out of previous full adder becomes the input carry for the next full adder. It calculates sum and carry according to the following equations. As carry ripples from one full adder to the other, it traverses longest critical path and exhibits worst case delay.

$$S_i = A_i \oplus B_i \oplus C_i \qquad (1)$$

$$C_{i+1} = A_i \bullet B_i + B_i \bullet C_i + C_i \bullet A_i \qquad (2)$$

Where i = 0, 1, 2, 3 …n-1

RCA is the slowest in all adders but it is very compact in size. If the ripple carry adder is implemented by concatenating N full adders, the delay of such an adder is 2N gate delays from C_{in} to C_{out}. The delay of adder increases linearly with increase in number of bits. Block diagram of RCA is shown in below Figure

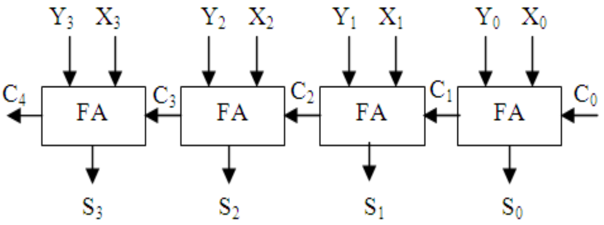

Figure 1: Block Diagram of RCA

DOI 10.17148/IJARCCE.2016.51116

○ Carry Select Adder

In Carry select adder scheme, blocks of bits are added in two ways: one assuming a carry-in of 0 and the other with a carry-in of 1.This results in two precomputed sum and carry-out signal pairs ($s^0_{i-1:k}, c^0_i$; $s^1_{i-1:k}, c^1_i$) , later as the block's true carry-in (c_k) becomes known , the correct signal pairs are selected. Generally multiplexers are used to propagate carries.

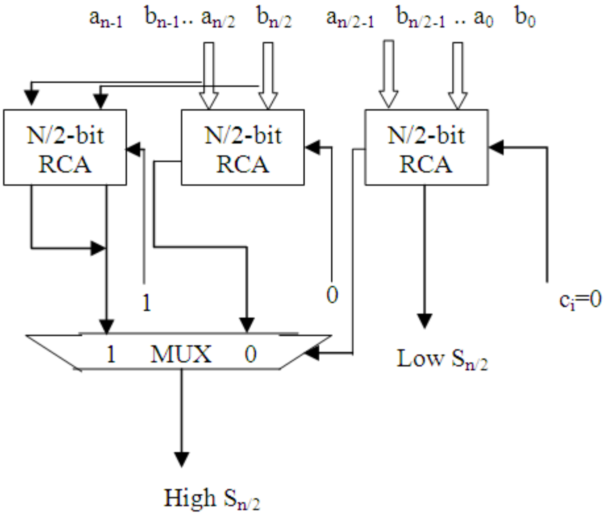
a_{n-1} b_{n-1} .. $a_{n/2}$ $b_{n/2}$ $a_{n/2-1}$ $b_{n/2-1}$ .. a_0 b_0
N/2-bit RCA
N/2-bit RCA
N/2-bit RCA
1
0
c_i=0
1 MUX 0
Low $S_{n/2}$
High $S_{n/2}$

Figure 2: A Carry Select Adder with 1 level using n/2- bit RCA

Because of multiplexers larger area is required. Have a lesser delay than Ripple Carry Adders (half delay of RCA). Hence we always go for Carry Select Adder while working with smaller no of bits.

Design of area- and power-efficient high-speed data path logic systems are one of the most substantial areas of research in VLSI system design. 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

III. PROPOSED ARCHITECTURE

The structure of the proposed 16-b modified CSLA using BEC for RCA with $c_i = 1$ is shown in Fig.4 In the proposed architecture we have replaced RCA with a BEC for $C_i = 1$ .The main advantage of this BEC logic comes from the lesser number of logic gates than the n-bit Full Adder (FA) structure. 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 $C_i = 0$ and $c_i = 1$, then the final sum and carry are selected by the multiplexers (mux). When $c_i = 1$, RCA will have one extra bit and this extra bit increases complexity in further stages and results more no. of slices. When we replace this RCA for $C_i = 1$ with a BCE no extra occurs and it results less no. of slices then the conventional RCA architecture. To replace the n-bit RCA, an n+1-bit BEC is required. 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

IV. RESULT AND SIMULATION

We have implemented Ripple Carry Adder (RCA), Carry Select Adder (CSA), and Carry Look-head Adder (CLA architecture. A comparison between RCA, CSA and CLA design based architecture for 4-bit, 16-bit is given in Table I and Table II respectively.

TABLE I Comparison between CSA and MCSA design based architecture for 16-bit sparten-2

	CSA	MCSA
Number of Slices	30 out of 192	25 out of 192
Number of 4 input LUTs	53 out of 384	45out of 384
Maximum combinational path delay	28.079ns	24.9408ns

TABLE II Comparison between CSA and MCSA design based architecture for 16-bit Vertex E

	CSA	MCSA
Number of Slices	30 out of 768	25 out of 768
Number of 4 input LUTs	53 out of 1536	45 out of 1536
Maximum combinational path delay	26.635ns	22.001ns

TABLE III Comparison between CSA and MCSA design based architecture for 16-bit Vertex-2

	CSA	MCSA
Number of Slices	30 out of 256	25 out of 256
Number of 4 input LUTs	53 out of 512	45 out of 512
Maximum combinational path delay	14.791ns	12.330ns

DOI 10.17148/IJARCCE.2016.51116

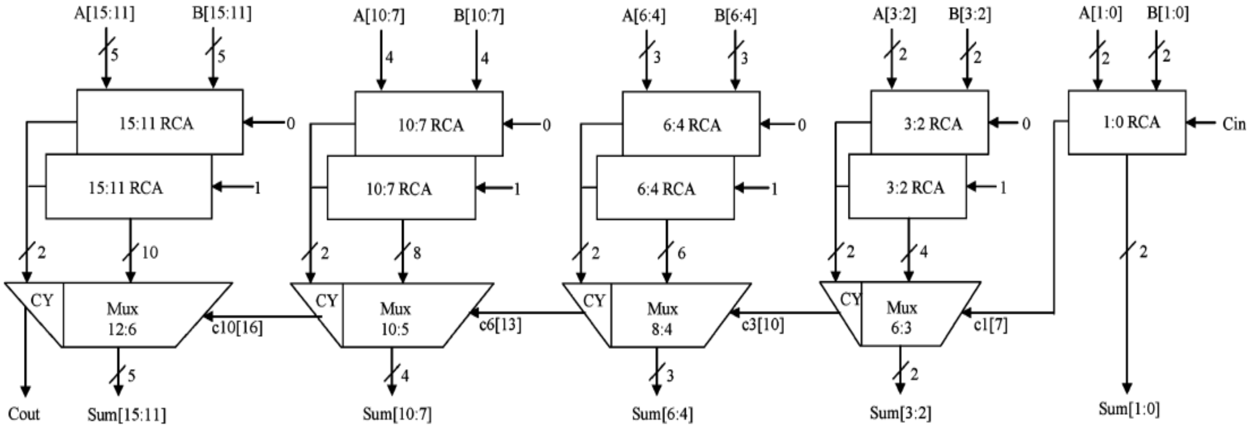

Figure 3: Design Structure of 16-bit Carry Select Adder

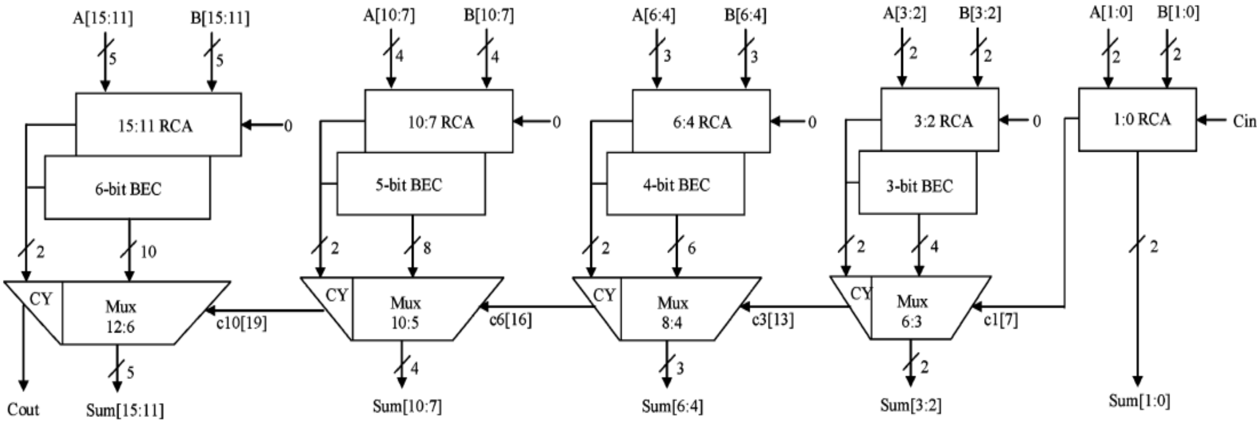

Figure 4: Design Structure of 16-bit Modified Carry Select Adder. The parallel RCA $C_i = 1$ is replaced with BEC

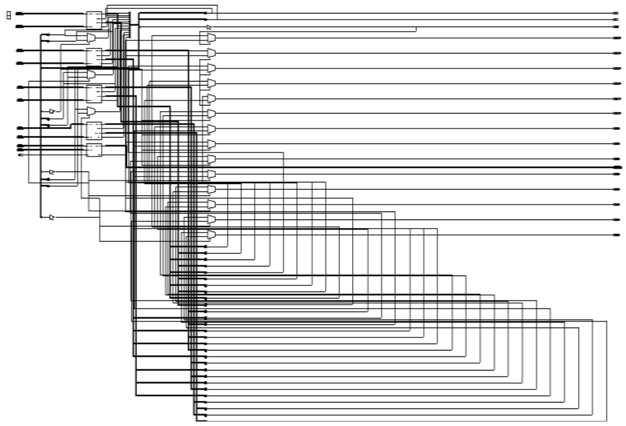

Figure 5: RTL view of 16-bit Modified Carry Select Adder.

Implementing the proposed CLA based architecture has been captured by VHDL and the functionality is verified by RTL and gate level simulation. To estimate the number of slices, flip flop, required time and minimum period information for ASIC design, we have used Xilinx Design Compiler to synthesize the design into gate level. Comparison of practical result up to third decomposition level architecture for 2-D DWT is given in Table I and Table II respectively.

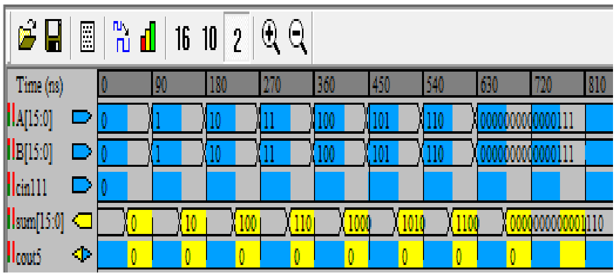

Figure 6: output waveform of 16-bit Modified Carry Select Adder.

V. CONCLUSION

We studied about different adders among compared them by different criteria like number of slice and Time so that we can judge to know which adder was best suited for situation. After comparing all we came to a conclusion that Carry Look-head Adders are best suited for situations where Speed is the only criteria. Similarly Ripple Carry Adders are best suited for Low Power Applications. But Among all the Carry Look Ahead Adder had the least Area-Delay product that tells us that, it is suitable for situations where both low power and fastness are a criteria such that we need a proper balance between both as is the case with our Paper.

DOI 10.17148/IJARCCE.2016.51116

REFERENCES

[1] K. Rawwat, T. Darwish, and M. Bayoumi, “.A low power carry select adder with reduces area”, Proc. Of Midwest Symposium on Circuits and Systems, pp. 218- 221, 2001.

[2] W. Jeong and K. Roy, “Robust high-performance low power adder”, Proc. of the Asia and South Pacific Design Automation Conference, pp. 503-506, 2003.

[3] Y. Kim and L-S Kim, “64-bit carry-select adder with reduced area”, Electronics Letters, vol. 37, pp. 614-615, May 2001.

[4] O. Kwon, E. Swartzlander, and K. Nowka, “A fast hybrid carry-look ahead/carry-select adder design”, Proc. of the 11th Great Lakes symposium on VLSI, pp.149-152, March 2001.

[5] B. Parhami, Computer Arithmetic, Algorithm and Hardware Design, Oxford University Press, New York, pp.91-119, 2000.

[6] Wang, Y. Pai, C.Song, X., “The design of hybrid carry look ahead/ carry-select adders”, Circuits and Systems II: Analog and Digital Signal Processing, IEEE Transactions on Volume 49, pp.16-24, 2002.

[7] Z. Abid, H. El-Razouk and D.A. El-Dib, “Low power multipliers based on new hybrid full adders”, Microelectronics Journal, Volume 39, Issue 12, Pages 1509-1515, 2008.

[8] Hasan Krad and Aws Yousif Al-Taie, “Performance Analysis of a 32-Bit Multiplier with a Carry-Look-Ahead Adder and a 32-bit Multiplier with a Ripple Adder using VHDL”, Journal of Computer Science 4 (4): 305-308, 2008.

[9] Behnam Amelifard, Farzan Fallah, and Massoud Pedram, "Closing the gap between carry select adder and ripple carry adder: a new class of low-power high-performance adders", in Proc. of IEEE International Symposium on Quality Electronic Design (ISQED), 2005.

DOI 10.17148/IJARCCE.2016.51116