IOSR Journal of Electronics and Communication Engineering (IOSR-JECE)
e-ISSN: 2278-2834,p- ISSN: 2278-8735.Volume 10, Issue 1, Ver. 1 (Jan - Feb. 2015), PP 36-44
www.iosrjournals.org

Multiplier and Accumulator Using Csla

Asst Prof, Anoop M M
Electronics and Communication Engineering MG University College Of Engineering And Technology,Thodupuzha

Abstract: *With the recent rapid advances in multimedia and communication systems, real-time signal processing like audio signal processing, video/image processing, or large-capacity data processing are increasingly being demanded. The multiplier and multiplier-and-accumulator (MAC) are the essential elements of the digital signal processing such as filtering, convolution, and Inner products.*

Index Terms: *Booth multiplier, carry save adder,Booth multiplier,Digital signal processing(DSP), multiplier and-accumulator(MAC).*

I. Introduction

A. Ripple Carry Adder

Multiple full adder circuits can be cascaded in parallel to add an N-bit number. For an N- bit parallel adder, there must be N number of full adder circuits. A ripple carry adder is a logic circuit in which the carry-out of each full adder is the carry in of the succeeding next most significant full adder. It is called a ripple carry adder because each carry bit gets rippled into the next stage. In a ripple carry adder the sum and carry out bits of any half adder stage is not valid until the carry in of that stage occurs. A propagation delay inside the logic circuitry is the reason behind this. Propagation delay is time elapsed between the application of an input and occurrence of the corresponding output. Consider a NOT gate, When the input is “0″ the output will be “1″ and vice versa. The time taken for the NOT gate’s output to become “0″ after the application of logic “1″ to the NOT gate’s input is the propagation delay here. Similarly the carry propagation delay is the time elapsed between the application of the carry in signal and the occurrence of the carry out (Cout) signal. Circuit diagram of a 4-bit ripple carry adder is shown below.

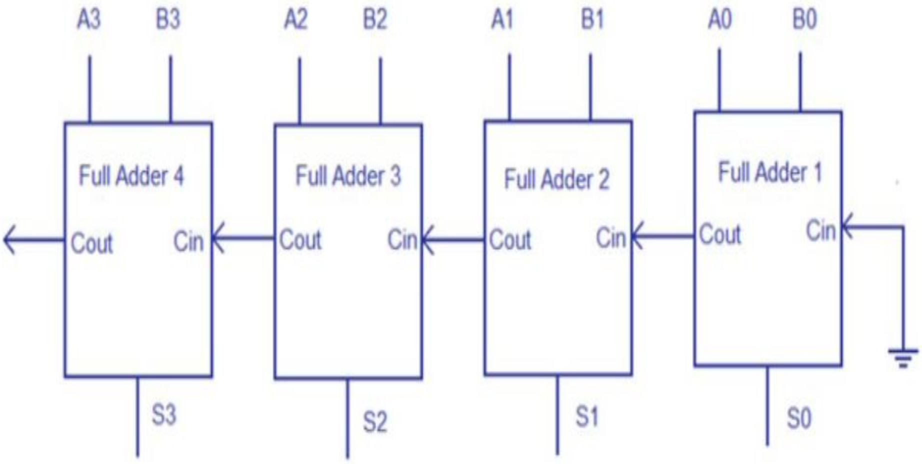


Fig. 3.1. Ripple carry adder

Sum out S0 and carry out Cout of the Full Adder 1 is valid only after the propagation delay of Full Adder 1. In the same way, Sum out S3 of the Full Adder is valid only after the joint propagation delays of Full Adder 1 to Full Adder 4. In simple words, the final result of the ripple carry adder is valid only after the joint propagation delays of all full adder circuits inside it.

B. Regular Csla Architecture

The structure of the 16-b regular SQRT CSLA is shown in Fig.3.2. It has five groups of different size RCA. From the structure of CSLA, it is evident that there is scope for reducing area and power consumption.

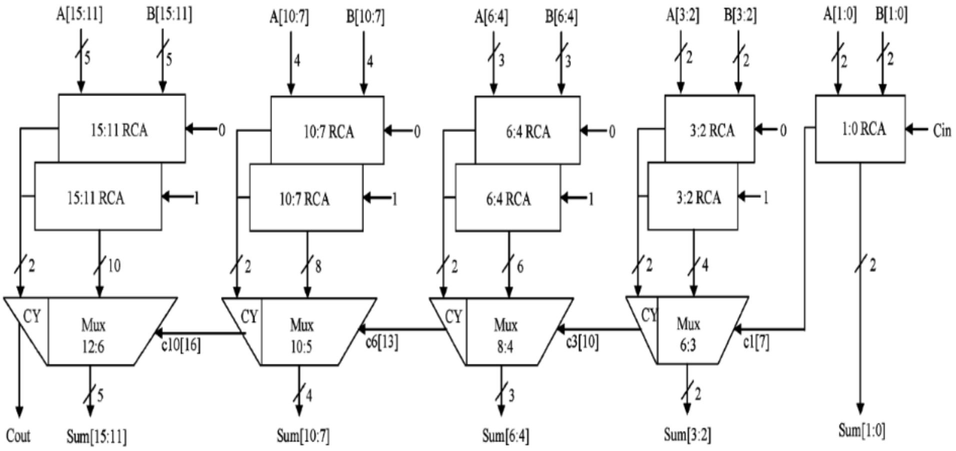


Fig.3.2.16B-Regular CSLA Architecture

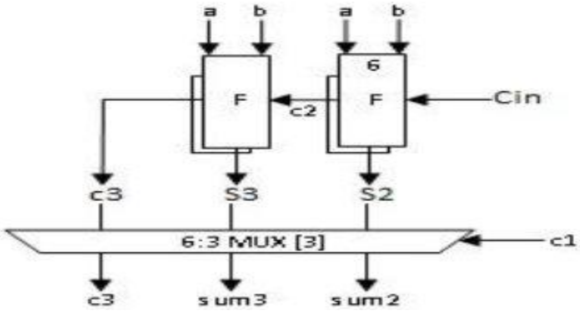


Fig.3.2(a).Group2 of 16B CSLA

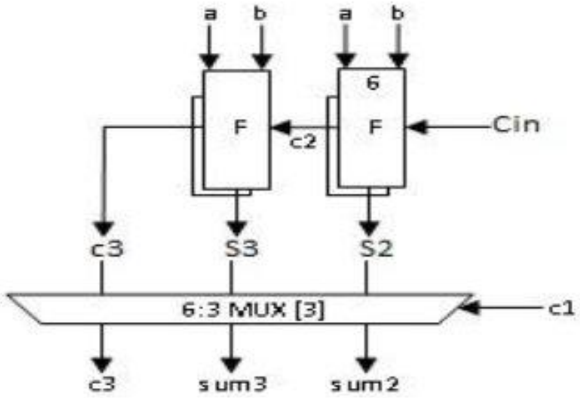


Fig.3.2(b). Group3 of 16B CSLA

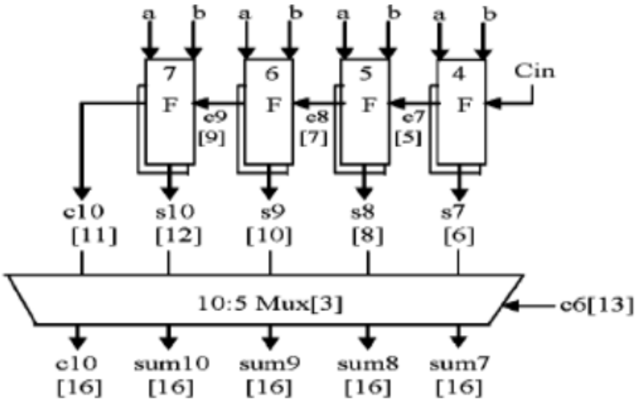


Fig.3.2(c). Group4 of 16B CSLA

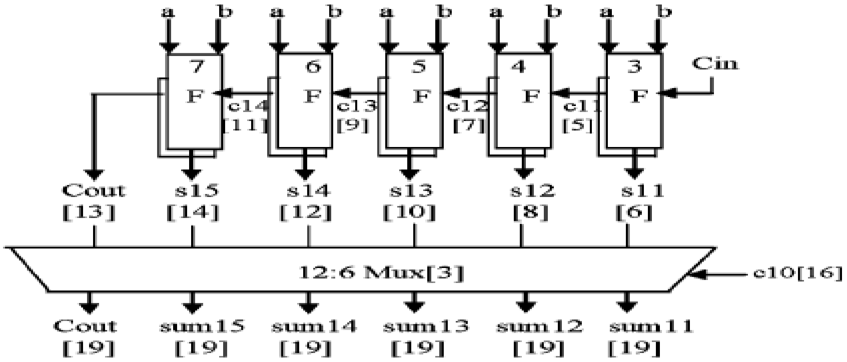


Fig.3.2 (d).Group5 of 16B CSLA

The steps leading to the evaluation are as follows.

1) The group2 [see Fig.3.2(a)] has two sets of 2-b RCA. Based on the consideration of delay values of Table I, the arrival time of selection input c1[time(t)=7] of 6:3 mux is earlier than s3[t=8] and later than s2[t=6]. Thus, sum3[t=11] is summation ofs3 and mux[t=3] and sum2[t=10] is summation of c1 and mux.
2) Except for group2, the arrival time of mux selection input is always greater than the arrival time of data outputs from the RCA's. Thus, the delay of group3 to group5 is determined, respectively as follows:

$$\{c6, sum[6:4]\} = c3[t = 10] + mux \quad (3.1)$$

$$\{c10, sum[10:7]\} = c6[t = 13] + mux \quad (3.2)$$

$$\{cout, sum[15:11]\} = c10[t = 16] + mux \quad (3.3)$$

3) The one set of 2-b RCA in group2 has 2 FA for Cin=1 and the other set has 1 FA and 1 HA for Cin=0. Based on the area count the total number of gate counts in group2 is determined as follows:

$$Gate\ count = 57(FA + HA + Mux) \quad \ldots \quad (3.4)$$

$$FA = 39(3 * 13) \quad \ldots \quad (3.5)$$

$$HA = 6(6 * 1) \quad \ldots \quad (3.6)$$

$$Mux = 12(3 * 4) \quad \ldots \quad (3.7)$$

GROUP	DELAY	AREA
GROUP2	11	57
GROUP3	13	87
GROUP4	16	117
GROUP5	19	147

Table 3.1: Delay and Area of Regular CSLA Groups

4) Similarly, the estimated maximum delay and area of the other groups in the regular SQRT CSLA are evaluated and listed in Table 3.1.

The carry out calculated from the last stage i.e. least significant bit stage is used to select the actual calculated values of the output carry and sum. The selection is done by using a multiplexer. Internal structure of the group 2 of regular 16-bit

CSLA is shown Fig.3.2(a). By manually counting the number of gates used for group 2 is 57 (full adder, half adder, and multiplexer). One input to the mux goes from the RCA with Cin=0 and other input from the RCA with Cin=1.

3.3. Csla Using Bec

1.1. 1. Binary to excess convertor 1

Excess-1 binary coded decimal (XS-1) or Stibitz code, also called biased representation or Excess-N, is a complementary BCD code and numeral system. Excess-1 was used on some older computers as well as in cash registers and hand held portable electronic calculators of the 1970's, among other uses. It is a way to represent values with a balanced number of positive and negative numbers using a pre-specified number N as a biasing value. It is a non weighted code. In XS-1, numbers are represented as decimal digits, and each digit is represented by four bits as the digit value plus 1 (the "excess" amount)

To reduce the area and power consumption of regular CSLA, RCA with Cin=1 is replaced with BEC. An n+1 bit BEC replaces the n bit RCA. The function table of a 3-b BEC is shown in Fig.3.3. and Table3.3

respectively. By the use of BEC logic, we can reduce the significant amount of silicon area reduction in the VLSI design. The Boolean expressions of the 3-bit BEC are given below.

S0 = ~ B0 (3.8)
S1 = B0 ^ B1 (3.9)
S2 = B2 ^ (B0 & B1) (3.10)

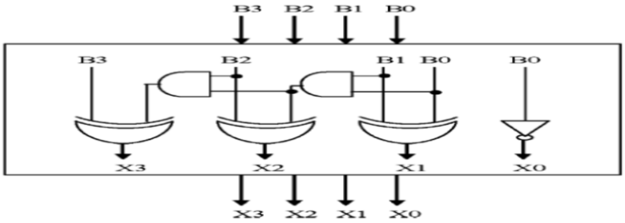


Fig.3.3. Binary to Excess Convertor-I

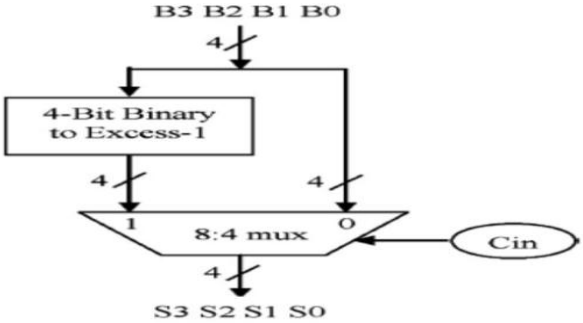


Fig.3.4. Binary to Excess Convertor-I

ADDER BLOCKS	DELAY	AREA
XOR	3	5
2:1 MUX	3	4
HALF ADDER	3	6
FULL ADDER	6	13

Table 3.2. Delay and Area Count of the Basic Blocks of CSLA

BINARY VALUE	BEC-1
0000	0001
0001	0010
0010	0011
0011	0100
0100	0101
0101	0110
0110	0111
0111	1000
1000	1001
1001	1010
1010	1011
1011	1100
1100	1101
1101	1110
1110	1111
1111	0000

Table 3.3. Binary to BEC I Truth Table(4-Bit)

3.3.2. Architecture of CSLA with BEC

The Binary to excess one Converter (BEC) replaces the ripple carry adder with Cin=1, in order to reduce the area and power consumption of the regular CSLA. The modified16-bit CSLA using BEC is shown in Fig.3.4. The structure is again divided into five groups with different bit size RCA and BEC. The group 2 of the modified 16-bit CSLA is shown Fig. 6. By manually counting the number of gates used for group 2 is 43 (full adder, half adder, multiplexer, BEC).

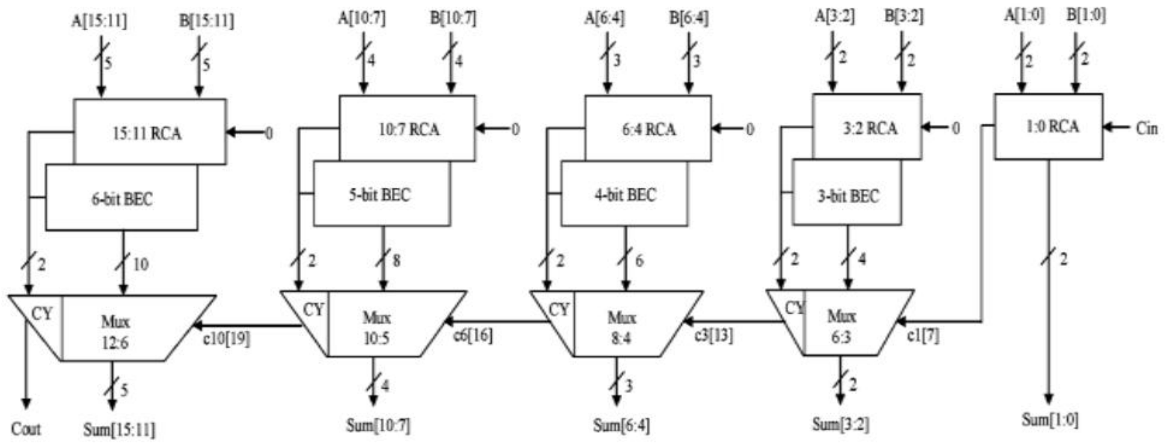
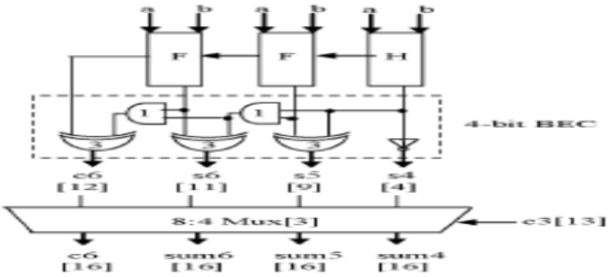


Fig. 3.4.16B-Modified CSLA architecture

Fig.3.4 (a). Group 2 of Modified CSLA Architecture



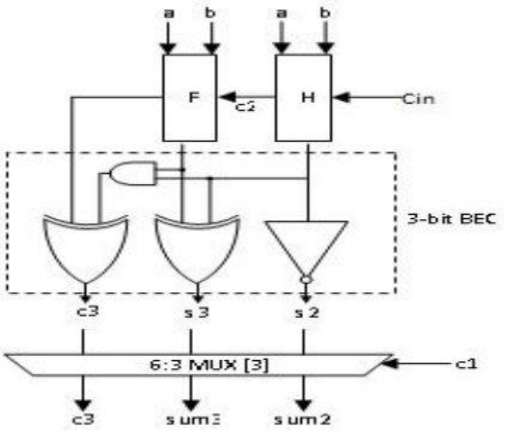


Fig. 3.4 (b). Group 3 of Modified CSLA Architecture

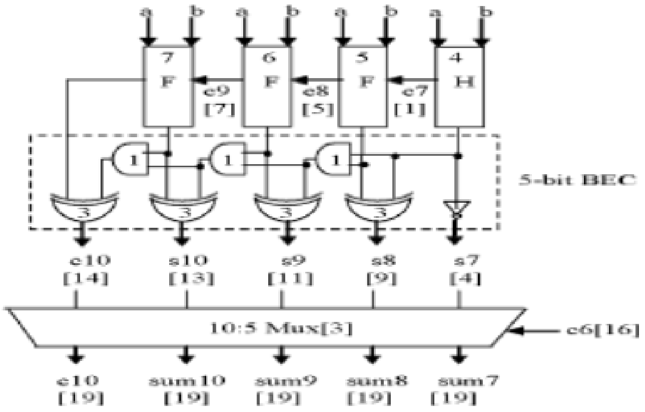


Fig. 3.4 (c). Group 4 of Modified CSLA Architecture

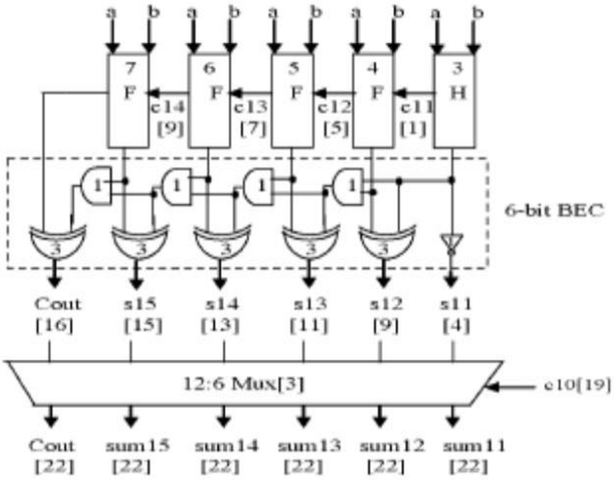


Fig. 3.4 (d). Group 5 of Modified CSLA Architecture

One input to the mux goes from the RCA with Cin=0 and other input from the BEC. Comparing the group 2 of both regular and modified CSLA, it is clear that BEC structure reduces the area and power. But the disadvantage of BEC method is that the delay is increasing than the regular CSLA.

The steps leading to the evaluation are given here.

1) The group2 [see Fig.3.4(a)] has one 2-b RCA which has 1 FA and 1 HA for Cin=0. Instead of another 2-b RCA with Cin=1 a 3-b BEC is used which adds one to the output from 2-b RCA. Based on the consideration of delay values, the arrival time of selection input c1[time(t)=7] of 6:3 mux is earlier than the and s3[t=9] and c3[t=10] and later than the s2[t=4]. Thus, the sum3 and final c3 (output from mux) are depending on s3 and mux and partial c3 (input to mux) and mux, respectively. The sum2 depends on c1 and mux

2) For the remaining group's the arrival time of mux selection input is always greater than the arrival time of data inputs from the BEC's. Thus, the delay of the remaining groups depends on the arrival time of mux selection input and the mux delay.

3) The area count of group2 is determined as follows:

$$Gate\ count = 43(FA + HA + Mux + BEC) \qquad (3.11)$$

$$FA = 13(1 * 13) \qquad (3.12)$$

$$HA = 6(6 * 1) \qquad (3.13)$$

$$AND = 1 \qquad (3.14)$$

$$NOT = 1 \qquad (3.15)$$

$$XOR = 10(2 * 5) \qquad (3.16)$$

$$Mux = 12(3 * 4) \qquad (3.17)$$

$$Mux = 12(3 * 4) \qquad (3.18)$$

GROUP	DELAY	AREA
GROUP2	13	43
GROUP3	16	61
GROUP4	19	84
GROUP5	22	107

Table 3.4. Delay and Area count of Modified CSLA

4) Similarly, the estimated maximum delay and area of the other groups of the modified SQRT CSLA are evaluated and listed in Table.3.4.

1.2. Csla Using D-Latch

3.4.1. D-latch

One very useful variation on the RS latch circuit is the Data latch, or D latch as it is generally called. As shown in the logic diagram below, the D latch is constructed by using the inverted S input as the R input signal. The single remaining input is designated "D" to distinguish its operation from other types of latches. It makes no difference that the R input signal is effectively clocked twice, since the CLK signal will either allow the signals to pass both gates or it will not.

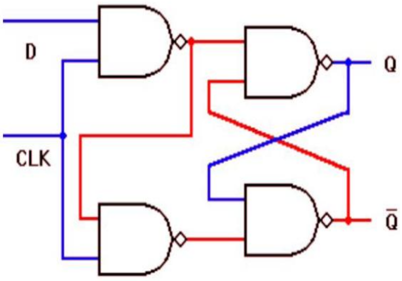


Fig.3.5. D-Latch Structure

In the D latch, when the CLK input is logic 1, the Q output will always reflect the logic level present at the D input, no matter how that changes. When the CLK input falls to logic 0, the last state of the D input is trapped and held in the latch, for use by whatever other circuits may need this signal. Because the single D input is also inverted to provide the signal to reset the latch, this latch circuit cannot experience a "race" condition caused by all inputs being at logic 1 simultaneously. Therefore the D latch circuit can be safely used in any circuit. Although the D latch does not have to be made edge triggered for safe operation, there are some applications where an edge-triggered D flip-flop is desirable. This can be accomplished by using a D latch circuit as the master section of an RS flip-flop. Both types are useful, so both are made commercially available. Except for the change in input circuitry, a D flip-flop works just like the RS flip-flop.

3.4.2. Architecture of CSLA with D latch

In this method replace any one of the RCA structure (i.e. cin =1 or cin =0) by parallel structure of D-latches. For n bit RCA structure it required n D-latches with enable pin as a clk. Latches are used to store one bit information. The RCA structure cin is replace by enable pin , where enable signal is clk signal. When enable pin en =1 then the RCA structure is calculate for cin=1 that result is stored in D-latch. When en =0 then it will calculate for cin =0 and the D-latch output and full adder output is given to the mux. By using selection line it will gives the proper output. Where the enable time period for ‘1’ is very less when compared to the enable pin ‘0’. Initially RCA structure will calculate for en=1 and then en =0. The architecture of proposed 16-b CSLA is shown in Fig. 10. It has different five groups of different bit size RCA and D-Latch. Instead of using two separate adders in the regular CSLA, in this method only one adder is used to reduce the area, power consumption and delay. Each of the two additions is performed in one clock cycle. This is 16-bit adder in which least significant bit (LSB) adder is ripple carry adder, which is 2 bit wide. The upper half of the adder i.e., most significant part is 14-bit wide which works according to the clock. Whenever clock goes high addition for carry input one is performed. When clock goes low then carry input is assumed as zero and sum is stored in adder itself. Latch is used to store the sum and carry for Cin=1.

Carry out from the previous stage i.e., least significant bit adder is used as control signal for multiplexer to select final output carry and sum of the 16-bit adder. If the actual carry input is one, then computed sum and carry latch is accessed and for carry input zero MSB adder is accessed. Cout is the output carry. The Fig.9 shows the internal structure of group 2 to 5 of the proposed 16-bit CSLA.

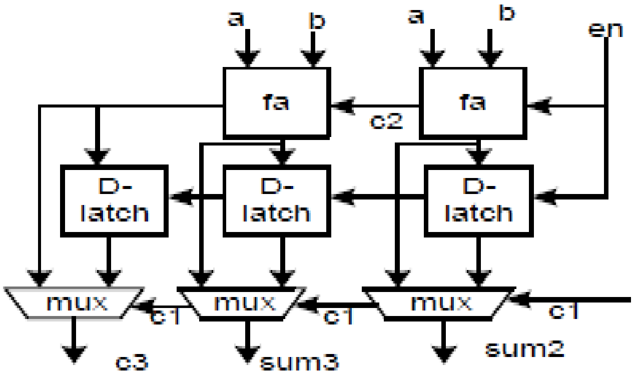


Fig.3.6 (a). Internal structure of 3 D-latches in parallel

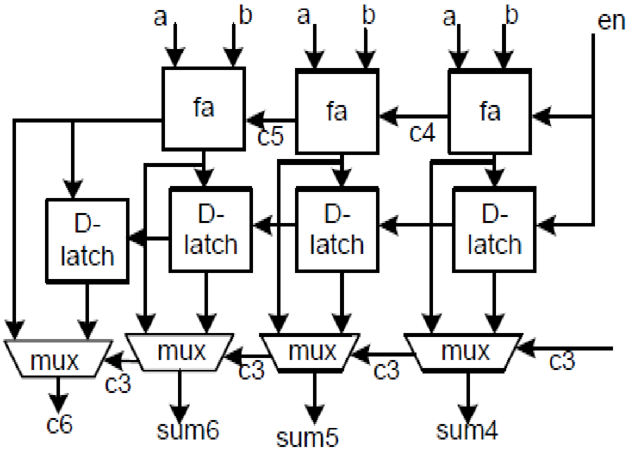


Fig.3.6(b). Internal structure of 4 D-latches in parallel

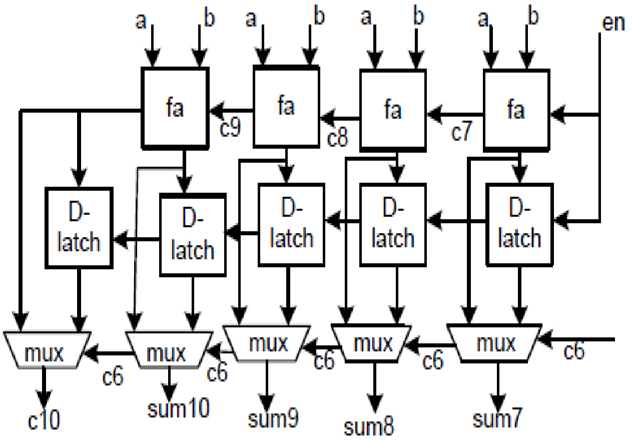


Fig. 3.6(c). Internal structure of 5 D-latches in parallel

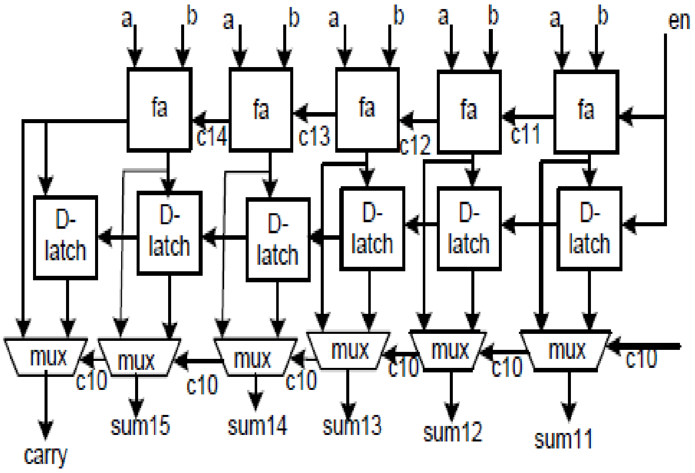


Fig.3.6 (d). Internal structure of 6 D-latches in parallel block of Fig.10

Fig.3.6. internal structures of Proposed CSLA by using D-Latch

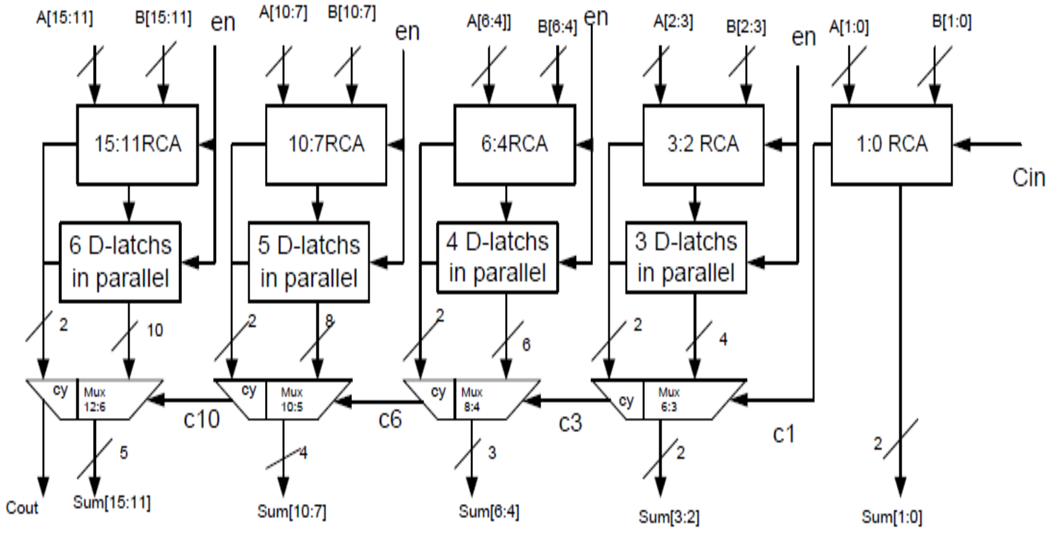


Fig.3.7. CSLA using D-Latch

1.3. Research methodologies and approach

This project is aimed for implementing high performance optimized FPGA architecture. This work has been developed using Verilog-HDL. Isim 13.3 is used for simulating the CSLA and synthesized using Xilinx PlanAhead13.3. Xilinx is commonly available tool. Availability is the main reason to select this tool for doing the project.

In this project I tried to reduce the delay a little bit by using Xilinx synthesis technique. These synthesis techniques promote fast and efficient FPGA design development. There are many synthesis options that can help you to obtain your performance and area objectives. Here I use the timing constraints to drive the optimization.

References

[1]. Fayez Elguibaly, " A fast parallel multiplier –accumulator using the modified booth algorithm," IEEE Trans on Circuits and systems,vol.47,September 2000

[2]. Young-Ho Seo,Dong –Wook Kim," A New VLSI Architecture of Parallel Multiplier and Accumulator Based on Radix-2 Modified Booth Algorithm,"IEEE Trans.on VLSI,vol.18,February 2012

[3]. B Rajani kumari,K.V Ramana Rao,"Dynamic Power Suppression Technique in Booth Multipliers",IJITEE 2278-3075, Volume-1, Issue-4, September 2012

[4]. G.Sasi,"Design of Low Power /High Speed Multiplier Using Spurious Power Suppression Technique" IJCSMC**,** Vol. 3, Issue. 1, January 2014, pg.37 – 41

[5]. R.Sheshadri M.E," Spurious Power Suppression Technique for VLSI Architecure",IJCSE Vol. 3 No.6 Dec 2012-Jan 2013