AN EFFICIENT 64-BIT SQRT CARRY SELECT ADDER WITH D-LATCH AND RCA

1. A.Soundarya,M.Tech (VLSI&ES),2. Shaila Shree , M.Tech (VLSI&ES), Assistant Professor ,
1,2. ECE Department,ST.MARY'S College of Engineering & Technology(Formely ASEC)

Abstract :The CSLA is used in many computational systems to relieve 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 time efficient because it uses multiple pairs of Ripple Carry Adders (RCA) to generate partial sum and carry by considering carry input , then the final sum and carry are selected by the multiplexers. The basic idea of this work is to use Binary to Excess-1 Converter (BEC) instead of RCA in the regular CSLA to achieve high speed and low power consumption. At the same time to further reduce the power consumption, a new approach of CSLA with D LATCH is proposed in this project.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 Carry selection. An efficient CSLA design is obtained using optimized logic units. The proposed Carry Select Adder design involves significantly less area and power than the recently proposed BEC-based CSLA.

KEYWORDS: CSLA, RCA, BEC, D-LATCH.

I.INTRODUCTION

Now days the portability of the electronic component have rapid growth, the low power arithmetic circuit has become very important in VLSI industry. In The digital signal processor(DSP) main building block is the Multiplier-Accumulator (MAC) unit. Full Adder used as a part of the MAC unit uses fulladder as part which can significantly influences the efficiency of total system. Full Adder circuit is necessary for low power application due to the reduction in power consumption.The basic operation Carry Select Adder (CSLA) is parallel computation. CSLA generates many carriers and partial sum. Multiplexers selects The final sum and carry. In the CSLA architecture, Addition operation usually trembles widely the overall performance of digital systems and a crucial arithmetic function. The adders are most widely used in the electronic applications. In the year 2002, a new concept of adders are come into existence those are called as the hybrid adders and those are used for increeses the speed of addition process by Wang et al. theadders gives hybrid carry look-ahead/carry select adders design. In 2008, the new hybrid full adders are used for designing low power multipliers. In digital adders, the speed of addition is mainly based on the propagation delay ,it is having the limitation by propagating delay through adder. In VLSI Design one of the most important research is the area and power optimized data path logic systems. The adders are most widely used In electronic system and applications. If we want design multipliers the concept of adders comes in to the picture because the adders are part of the multipliers designs . As we know millions of instructions per second are performed in microprocessors. In the microprocessor device area and power consumption are the most important factors in the designing multipliers andadders.The power consumption and area should low in the microprocessors. Devices like computers Mobile phones, Laptops etc... those achievesmore battery backup. So, a VLSI designer has to make perfect these parameters in a design. These are main constraints,so these are very difficult to achieve .so the constraints has to be made depending on demand or application of the circuit in the industry. the N full adders used in this architecture link together with N bit Ripple carry adder. In the ripple carry adder operation the carry out of previous full adder becomes the input carry for the next full adder. Like that sum and carry calculated at the end of the last full. As carry ripples from one full adder to the other, if the size of the full adder is high then it has a more delay.

II. REGULAR CSLA

The CSLA has two units: 1) the sum and carry generator unit (SCG) and 2) the sum and carry selection unit. The Sum and Carry Generation(SCG) unit consumes most of the logic resources of Carry Select Adder and significantly contributes to the critical path. Different logic designs have been suggested for efficient implementation of the Sum and Carry Generation unit. We made a study of the logic designs suggested for the SCG unit of conventional and binary excess-1 converters(BEC)-based Carrty Select Adders of by suitable logic expressions. The main objective of this study is to identify redundant logic operations and data dependence. Accordingly, we remove all redundant logic operations and sequence logic operations based on their data dependence.

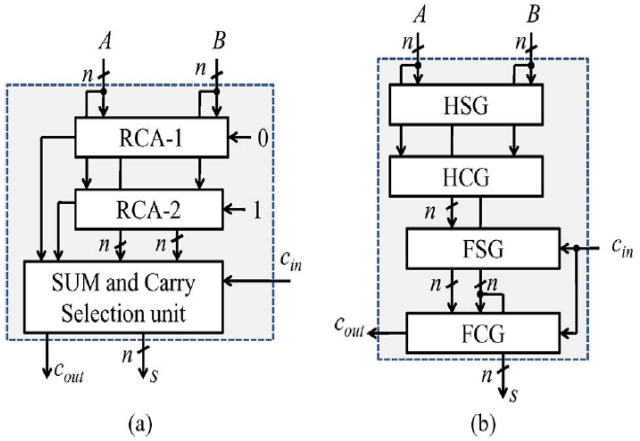


Fig. 1. (a) Conventional CSLA; n is the input operand bit-width. (b) The logic operations of the RCA is shown in

split form, where HSG, HCG, FSG, and FCG represent half-sum generation, half-carry generation, full-sum generation, and full-carry generation, respectively.

As shown in Fig. 1(a), the SCG unit of the conventional CSLA [3] is composed of two n-bit RCAs, where n is the adder bit-width. The logic operation of the n-bit RCA is performed in four stages: 1) half-sum generation (HSG); 2) half-carry generation (HCG); 3) full-sum generation (FSG); and 4) full carry generation (FCG). Suppose two n-bit operands are added in the conventional CSLA, then RCA-1 and RCA-2 generate n-bit sum (s0 and s1) and output-carry (c_{out}^0 and c_{out}^1) corresponding to input-carry (c_{in}= 0 and c_{in} = 1), respectively. Logic expressions of RCA-1 and RCA-2 of the SCG unit of the n-bit CSLA are given as

$$s_0^0(i) = A(i) \oplus B(i)\ c_0^0(i) = A(i) \cdot B(i) \quad (1a)$$

$$s_1^0(i) = s_0^0(i) \oplus c_1^0(i-1) \quad (1b)$$

$$c_1^0(i) = c_0^0(i) + s_0^0(i) \cdot c_1^0(i-1)\ c_{out}^0 = c1^0(n-1) \quad (1c)$$

$$s_0^1(i) = A(i) \oplus B(i)\ c_0^1(i) = A(i) \cdot B(i) \quad (2a)$$

$$s_1^1(i) = s_0^1(i) \oplus c_1^1(i-1) \quad (2b)$$

$$c_1^1(i) = c_0^1(i) + s_0^1(i) \cdot c_1^1(i-1)\ c_{out}^1 = c_1^1(n-1) \quad (2c)$$

Where $c_1^0(-1) = 0$, $c_1^1(-1) = 1$, and $0 \le i \le n-1$. As shown in (1a)–(1c) and (2a)–(2c), the logic expression of $\{s_0^0(i), c_0^0(i)\}$ is identical to that of $\{s_0^1(i), c0^1(i)\}$. These redundant logic operations can be removed to have an optimized design for RCA-2, in which the Half Sum Generation and Half Carry Generation of RCA-1 is shared to construct RCA-2. Based on this, [4] and [5] have used an add-one circuit instead of RCA-2 in the CSLA, in which a BEC circuit is used in [6] for the same purpose. Since the BEC-based CSLA offers the best area–delay–power efficiency among the existing CSLAs, we discuss here the logic expressions of the SCG unit of the BEC-based CSLA as well.

As shown in Fig. 2, the RCA calculates n-bit sum s_1^0 and c_{out}^0 corresponding to c_{in} = 0. The BEC unit receives s_1^0 and c_{out}^0 from the RCA and generates (n + 1)-bit excess-1 code. The most significant bit (MSB) of BEC represents c_{out}^1, in which n least significant bits (LSBs) represent s_1^1. The logic expressions

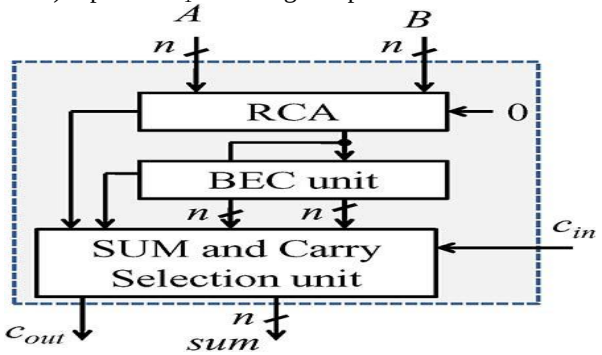


Fig. 2. Structure of the BEC-based CSLA; n is the input operand bit-width. of the RCA are the same as those given in (1a)–(1c). The logic expressions of the BEC unit of the n-bit BEC-based CSLA are given as

$$s_1^1(0) = s_1^0(0)\ c_1^1(0) = s_1^0(0) \quad (3a)$$

$$s_1^1(i) = s_1^0(i) \oplus c_1^1(i-1) \quad (3b)$$

$$c_1^1(i) = s_1^0(i) \cdot c_1^1(i-1) \quad (3c)$$

$$c_{out}^1 = c_1^0(n-1) \oplus c_1^1(n-1) \quad (3d)$$

for $1 \le i \le n-1$. We can find from (1a)–(1c) and (3a)–(3d) that, in the case of the BEC-based CSLA, c_1^1 depends on s_1^0, which otherwise has no dependence on $s1^0$ in the case of the conventional CSLA. The BEC method therefore increases data dependence in the CSLA. We have considered logic expressions of the conventional CSLA and made a further study on the data dependence to find an optimized logic expression for the CSLA.

III. CSLA USING D-LATCH LOGIC

This method replaces the BEC circuit by D-latch. Latches are used to store 1-bit binary information. The latch is one of the sequential circuits so Their outputs are depends on the present inputs and previous inputs. In other words, the latch is level sensitive, so whenlatch are enabled, the operation of latch is changes according with input signal of the latch. The architecture of proposed 16-bit Carry Select Adder is shown in Fig. 3. In this we are using five different ripple carry adders with different bit size and D-Latch. In this proposed method uses only one adder Instead of using two separate adders in the regular carry select adder(CSLA) to reduce the area, and power consumption. In CSLA Each of the two additions is performed in one clock cycle. In the 16-bit adder Ripple carry adder used in the least significant bit (LSB) which is 2 bit wide. The upper half of the adder is most significant part is 14-bit wide the upper of the adder is works depends on the clock, the input carry performed addition when carry goes high.

The carry input is assumed as zero While clock goes low and the sum of adder is stored in adder itself. From the Fig. it can understand that latch is used to store the sum and carry for Cin=1 and cin=0.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 is the output carry.

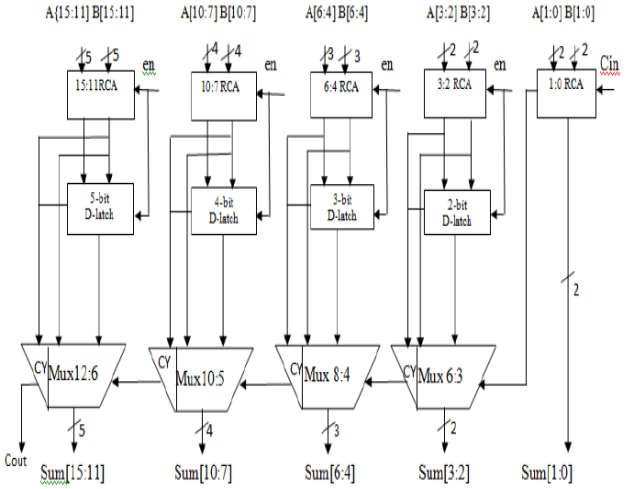


Fig.3. 16-Bit CSLA With D-Latch Architecture

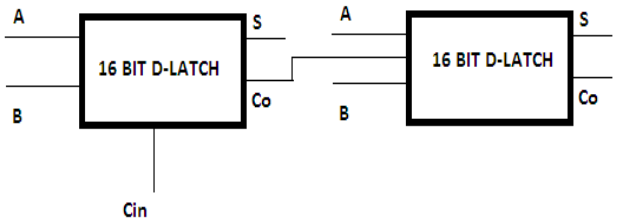


Fig.4.32- Bit CSLA With D-Latch Architecture

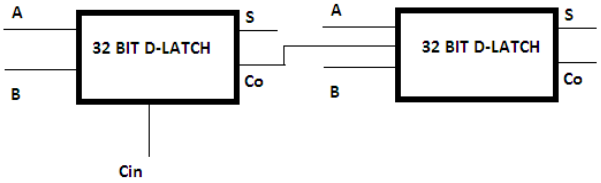


Fig.5. 64- Bit CSLA With D-Latch Architecture

The architectures of the 16-Bit, 32-Bit and 64-Bit CSLA with D-LATCH are shown in figure 3, figure 4 and 5. The 64-Bit CSLA with D-LATCH Architecture is designed based on the cascading of two 32-Bit Architectures. In the D-LATCH architecture first stage is designed based on Ripple Carry adders and the second stage is designed based on the D-LATCH logic.

IV. WORKING PRINCIPLE OF CSLA ADDER USING D LATCH

The bits from a and b (i,e a[1:0] and b[1:0]) as inputs to (1:0)RCA along with cin as input.The bits from a and b (i,e a[3:2] and b[3:2]) as inputs to (3:2)RCA along with en. When en=1, the output of the(3:2)RCA is fed as input to the (2 bit) D-Latch and the output ofthe (2 bit) D-latch follows the input and given as an input tothe (6:3)multiplexer.

When en=0, the last state of the (2 bit) Dinput is trapped and held in the latch and thereforethe output from the (3:2)RCA is directly given as an inputto the (6:3) multiplexer without any delay. Now the (6:3) multiplexer selectsthe sum bit according to the carry generated from (1:0) RCA which takes cin as input carry and it is theselection bit and the inputs of the (6:3) multiplexer are the outputsobtained when en=1 and 0.

The bits from a and b (i,e. a[6:4] and b[6:4]) are given as inputs to (6:4)RCA along with en as carry. When en=1, the output of the(6:4)RCA is fed as input to the (3 bit) D-Latch and the output ofthe (3 bit) D-latch follows the input and given as an input tothe (8:4)multiplexer. When en=0, the last state of the (3 bit) Dinput is trapped and held in the latch and thereforethe output from the (6:4)RCA is directly given as an inputto the (8:4) multiplexer without any delay. Now the (8:4) multiplexer selectsthe sum bit according to the carry generated from (6:3) multiplexer which is theselection bit and the inputs of the (8:4) multiplexer are the outputsobtained when en=1 and 0.

The bits from a and b (i,e a[10:7] and b[10:7]) are given as inputs to (10:7)RCA along with en as carry. When en=1, the output of the(10:7)RCA is fed as input to the (4 bit) D-Latch and the output ofthe (4 bit) D-latch follows the input and given as an input tothe (10:5)multiplexer.When en=0, the last state of the (4 bit) Dinput is trapped and held in the latch and thereforethe output from the (10:7)RCA is directly given as an inputto the (10:5) multiplexer without any delay.

Now the (10:5) multiplexer selectsthe sum bit according to the carry generated from (8:4) multiplexer which is theselection bit and the inputs of the (10:5) multiplexer are the outputsobtained when en=1 and 0.

The bits from a and b (i,e a[15:11] and b[15:11]) are given as inputs to (15:11)RCA along with en as carry. When en=1, the output of the(15:11)RCA is fed as input to the (5 bit) D-Latch and the output ofthe (5 bit) D-latch follows the input and given as an input tothe (12:6)multiplexer. When en=0, the last state of the (5 bit) Dinput is trapped and held in the latch and thereforethe output from the (15:11)RCA is directly given as an inputto the (12:6) multiplexer without any delay. Now the (12:6) multiplexer selectsthe sum bit according to the carry generated from (10:5) multiplexer which is theselection bit and the inputs of the (12:6) multiplexer are the outputsobtained when en=1 and 0.

V. RESULTS

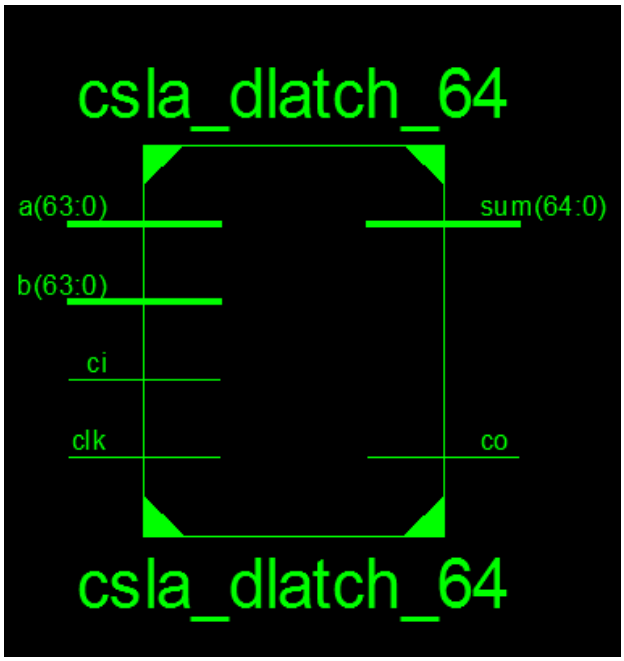


Fig 6: RTL Schematic view of 64 bit CSLA with D-Latch

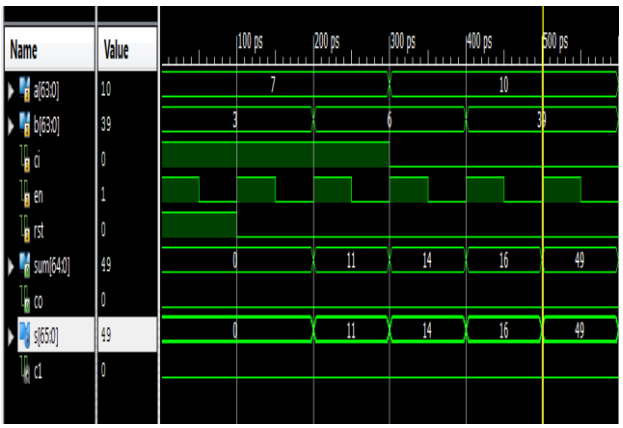
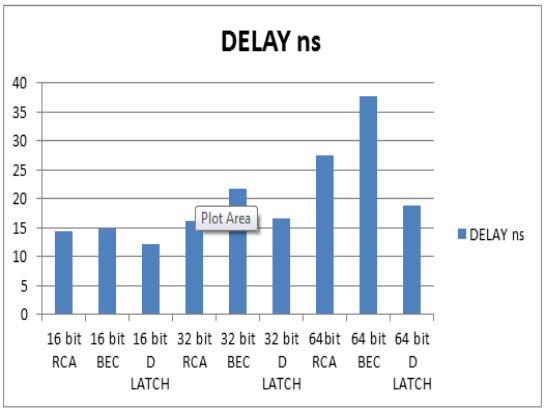


Fig 7: Waveform of 64 bit CSLA with D-Latch

	16 bit RCA	16 bit BEC	16 bit D LATCH	32 bit RCA	32 bit BEC	32 bit D LATCH	64bit RCA	64 bit BEC	64 bit D LATCH
Number of 4 input LUTs used	43	48	32	102	101	134	204	201	285
Available LUTs	9312	9312	9312	9312	9312	9312	9312	9312	9312
DELAY ns	14.362	14.73	12.211	16.117	21.756	16.506	27.539	37.772	18.873
Total supply power mw	76	76	76	76	76	76	76	76	76
Power Utilized mw	0.351	0.392	0.039	0.832	0.823	1.092	1.663	1.638	2.3235

Table 1: RESULTS FOR THE SELECTED DEVICE XC3S1600E-5FG320



Graphical representation of Delay in 64-bit CSLA using D-Latch

VI. CONCLUSION

Addition is the most common and often used arithmetic operation on microprocessor, digital signal processor, especially digital computers. Also, it serves as a building block for synthesis all other arithmetic operations. Therefore, regarding the efficient implementation of an arithmetic logic unit, the adder structures become a very critical hardware unit.A D-LATCH based CSLA architecture is proposed in this project to reduce the delay of CSLA architecture than the recently proposed BEC based CSLA architecture. The functionality verification of the design is carried out by using ISE Simulator and the synthesis is also carried out by the XILINX ISE 12.3i.The HDL used for obtaining an RTL schematic and for designing the modules is VERILOG. From the graphs and the tables it is concluded that, the proposed D-LATCH based design is having less delay when compare to the BEC based and RCA based architectures.With the increase in silicon densities, it is becoming feasible for compression systems to be implemented in a single chip. Here we have implemented CSLA using D-latch approach with less Delay. In future, we further reduce Area and Power parameters without the penalty of resource

REFERENCES

[1] B. Ramkumar and Harish M Kittur, "Low Power and AreaEfficient Carry Select Adder" IEEE TRANSACTIONS ON VERYLARGE SCALE INTEGRATION (VLSI) SYSTEMS-2011.

[2] Bedrij, O. J., (1962), "Carry-select adder," IRE Trans.Electron. Comput., pp.340–344 .

[3] Ceiang ,T. Y. and Hsiao,M. J. ,(Oct. 1998),"Carry-selectadder using single ripple carry adder," Electron. Lett., vol. 34, no.22, pp. 2101– 2103.

[4] Ramkumar,B. , Kittur, H.M. and Kannan ,P. M. ,(2010),"ASICimplementation of modified faster carry save adder," Eur. J. Sci.Res., vol. 42, no. 1,pp.53–58,2010.

[5] J. M. Rabaey, Digtal Integrated Circuits—A DesignPerspective. Upper Saddle River, NJ: Prentice-Hall, 2001.

[6] E. Abu-Shama and M. Bayoumi, "A New cell for low poweradders," in Proc.Int. Midwest Symp. Circuits and Systems, 1995,pp. 1014–1017.

[7] Y. Kim and L.-S. Kim, "64-bit carry-select adder with reducedarea,"Electron. Lett., vol. 37, no. 10, pp. 614–615, May 2001.

[8] Y. He, C. H. Chang, and J. Gu, "An area efficient 64-bit squareroot carry-select adder for low power applications," in Proc. IEEEInt. Symp. Circuits Syst., 2005, vol 4, pp.4082-4085.es