Design and Analysis of Carry Select Adder with RCA and BEC Circuits

M.N.V.anusha, K. Babulu
Project Associate, Professor of ECE Dept.

***Abstract*-- The major speed limitation in any adder is in the production of carries and addition problem. Carry Select Adder (CSA) is one of the fastest adders used in many data processing processors to perform fast arithmetic functions. The carry select adder 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 sum. However the carry select adder is not area 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 (MUX). The basic idea of this paper is to use binary to excess converter (BEC) instead of RCA to achieve high speed. The proposed work in this paper uses simple and gate level modification to enhance the features of propagation speed, power, area etc.**

***Index terms*-- CSA, RCA, BEC, MUX, Design, Simulation, Synthesis.**

I. INTRODUCTION

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. The CSA 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. The CSA 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_{in}=0 and C_{in}=1,then the final sum and carry selected by multiplexers(MUX).

The basic idea of this work is use of binary to excess-1 converter instead of RCA with C_{in}=1.The main advantage of binary to excess converter is having lesser number of logic gates than the n-bit full adder structure.

The Fig.1.1 shows the logic diagram of 4-bit BEC whereas the Fig.1.2 shows the truth table of 4-bit BEC.

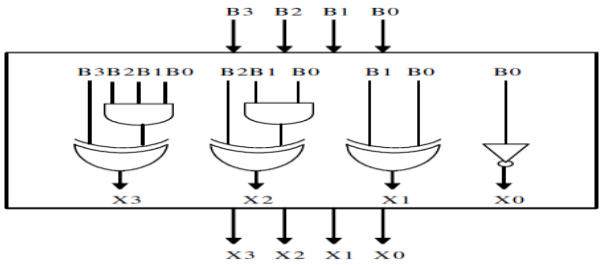


Fig.1.1.Binary to Excess Converter

From Fig.1.1, inputs are B3,B2,B1,B0 and outputs are X3,X2,X1,X0.The Boolean expressions of BEC is listed below,(Note: functional symbols ~ NOT ,& AND,^ XOR).

X0= ~ B0, X1= B0 ^ B1, X2= B2 ^ (B0 & B1), X3= B3 ^ (B0&B1&B2).

B[3:0]	X[3:0]
0000	0001
0001	0010
0010	0011
0011	0100
0100	0101
0101	0110
0110	0111
0111	1000
1000	1001

Fig.1.2 Binary to Excess Converter Truth table

ISSN: 2278 – 7798

II. CSA WITH RCA CIRCUIT

The CSA generally consist of two ripple carry adders and a multiplexer. Adding two n-bit with a carry select adder is done with two adders(therefore two ripple carry adders) in order to perform the calculation twice, one time with the assumption of the carry being zero and the other assuming one. After two results are calculated, the correct sum as well as correct carry is then selected with the multiplexer once the correct carry is identified. Ripple carry adder is combination of full adders. Each full adder inputs a Cin which is the Cout of previous full adder. This kind of RCA is used for which each carry bit ripples to the next full adder. Thus the proposed RCA is now constructed by adding full adder blocks in series, which is shown in Fig.(A).

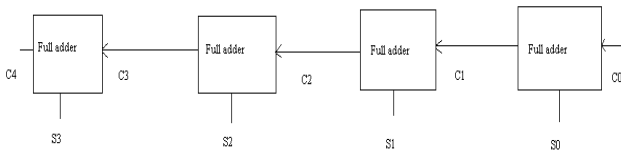


Fig (A) :Proposed RCA Chain

The modified RCA Chain structure in Fig.2.1 shows the how sum and carry generated. Let us consider the case, if we want to add two operands A and B where A=1 0 1 1 , B=1 1 0 1.
By adding two operands, A+B =1 1 0 0 0.
i.e, Cout S3 S2 S1 S0.

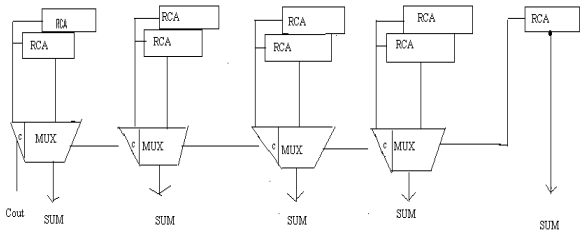
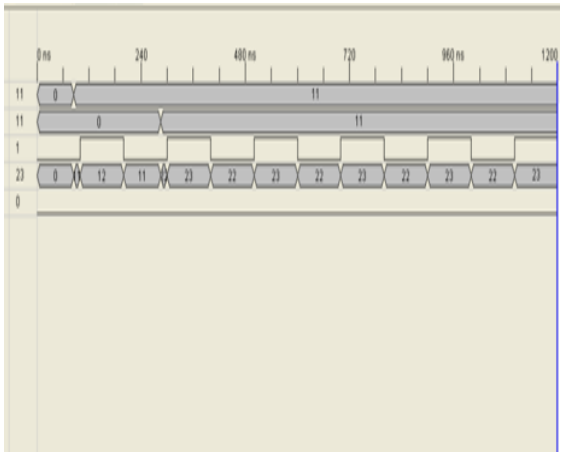


Fig (B) : CSA with Proposed RCA Circuit

The Fig.(B) depicts that the basic building blocks of CSA with proposed RCA circuit. Here, two RCA circuits connected with MUX. the upper RCA selected when carry is “0” and lower RCA selected when carry is “1”.the selected RCA connected to MUX. Depending on the carry value the MUX will select the correct RCA and generate Sum. In general it is to be noted that the operation of RCA is slowest in all adders. But it is very compact in size. If the ripple carry adder is implemented by N full adders, the delay is 2N. The delay of adder increases linearly with increase in number of bits.

C. Simulation Results of 16-bit CSA with RCA



The Fig.(C) shows the simulation results of !6 bit CSA with RCA,these values are calculated by Xilinx software. Whereas A,B and Cin are input values, these values generate the corresponding SUM and Cout values.

D. Synthesis Report of 16 bit CSLA with RCA

```
    LUT3:I2->O            2   0.479   0.915  g0/FA2/CO (c0)
    LUT3:I1->O            2   0.479   0.804  Ker41 (N4)
    LUT3:I2->O            2   0.479   0.915  g12/M<2>1 (M1<2>)
    LUT3:I1->O            2   0.479   0.804  Ker51 (N5)
    LUT3:I2->O            2   0.479   0.804  Ker11 (N1)
    LUT3:I2->O            3   0.479   1.066  g22/M<3>1 (M2<3>)
    LUT3:I0->O            2   0.479   0.804  Ker13_SW1 (N158)
    LUT3:I2->O            1   0.479   0.851  Ker845 (Ker8_map19)
    LUT3:I1->O            2   0.479   0.804  Ker873 (N126)
    LUT3:I2->O            3   0.479   0.794  g32/M<4>1 (M3<4>)
    LUT4:I3->O            1   0.479   0.851  Ker924 (Ker9_map33)
    LUT4:I1->O            2   0.479   0.804  Ker941 (N157)
    LUT3:I2->O            2   0.479   0.804  Ker21 (N21)
    LUT3:I2->O            1   0.479   0.681  g42/M<4>1 (M4<4>)
    OBUF:I->O                 4.909          S_15_OBUF (S<15>)
    ----------------------------------------
    Total                    26.319ns (12.809ns logic, 13.510ns route)
                                      (48.7% logic, 51.3% route)

=========================================================================
CPU : 6.81 / 7.13 s | Elapsed : 7.00 / 7.00 s

-->

Total memory usage is 106512 kilobytes
```


ISSN: 2278 – 7798

Synthesis report will generate after the simulation process. In this Delay values are generated. The problem in this design is the number of full adders increased then the circuit complexity also increases. The number of full adders increases the design of area and decrease the speed of the system.

III. CSA WITH BEC CIRCUIT

In this design the CSLA is obtained by BEC together with RCA. In this design we give the MUX inputs are RCA output and BEC output. Based on the carry input the MUX will select the carry input.
How the goal of fast addition is achieved using BEC is described by Fig (A).

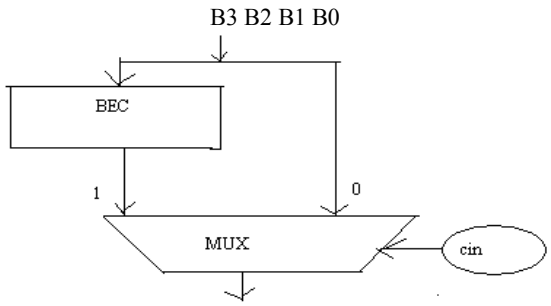


Fig (A) : BEC with MUX

One input of the MUX gets as it input (B3,B2,B1,B0) and another input of the MUX is BEC output. This produces two partial product results in parallel and MUXES are used to select either BEC output or direct inputs according to the control signal Cin.
Fig 3.2 is the modified version of CSLA with RCA. In this the RCA with Cin=1 is replaced by BEC in order to reduce the circuit complexity and delay.

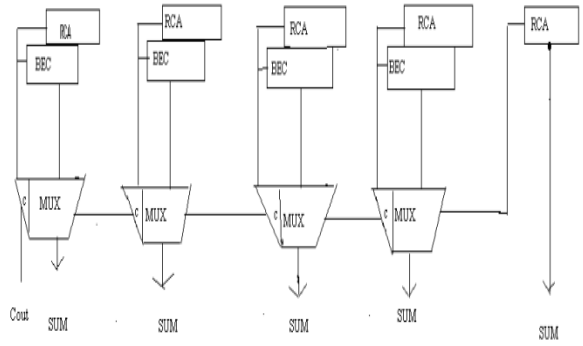
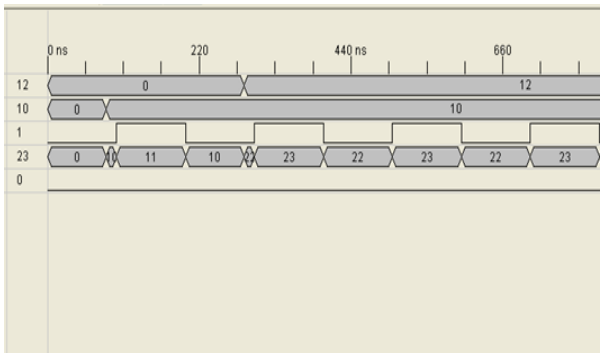


Fig (B): Basic building block of CSLA with BEC

The basic work of fig (B) is, the RCA and BEC circuits connected to MUX. It will select the carry value and generate the required sum.

C: CSLA with BEC 16 bit Simulation results



This is the screen shot of CSLA with BEC simulation results. Here inputs are A,B and Ci, outputs are SUM and Cout.

D: Synthesis Report of 16-Bit CSLA with BEC

```
Data Path: CI to CO
                                    Gate     Net
   Cell:in->out      fanout   Delay   Delay  Logical Name (Net Name)
   ----------------------------------------  ------------
    IBUF:I->O             3   0.715   1.066  CI_IBUF (CI_IBUF)
    LUT3:I0->O            1   0.479   0.740  g0/FA2/CO_SW0 (N74)
    LUT3:I2->O            3   0.479   1.066  g0/FA2/CO (c0)
    LUT3:I0->O            2   0.479   0.804  g12/M<2>1_SW0 (N78)
    LUT3:I2->O            5   0.479   1.078  g12/M<2>1 (M1<2>)
    LUT4:I0->O            1   0.479   0.704  g32/M<4>1_SW0 (N80)
    LUT4:I3->O            6   0.479   0.876  g32/M<4>1 (M3<4>)
    LUT4:I3->O            1   0.479   0.740  g42/M<5>1_SW0 (N76)
    LUT4:I2->O            1   0.479   0.681  g42/M<5>1 (M4<5>)
    OBUF:I->O                 4.909          CO_OBUF (CO)
   ----------------------------------------
   Total                     17.210ns (9.456ns logic, 7.754ns route)
                                      (54.9% logic, 45.1% route)

=========================================================================
CPU : 8.99 / 9.42 s | Elapsed : 9.00 / 9.00 s

-->

Total memory usage is 106512 kilobytes
```


ISSN: 2278 – 7798

This is the improved version of CSLA, it is clearly shows that BEC reduces the delay compare to CSLA with RCA. The RCA contains more full adder circuits which leads to more delay. That is the why BEC is replaced with RCA circuit. The main advantage of BEC is less logic gates and less delay.
After comparisons of both performances it is shows that CSLA with RCA Delay=26.319ns and CSLA with BEC Delay =17.210ns,So we can conclude that BEC circuit minimize the delay value.

IV. COMPARISON OF CSLA WITH RCA AND CSLA WITH BEC

CSLA with RCA	CSLA with BEC
Total time=26.319ns	Total time=17.210ns

From the above comparison shows that using BEC circuit in place of RCA improves the speed and performance. So CSLA with BEC is better than CSLA with RCA. Using this BEC circuit we can also improve the power consumption, and area efficient.

V. CONCLUSION

Now a day's carry select adder used in many data processing processors to perform fast arithmetic functions. That's why we have designed a configurable adder with minimum delay overhead, with the help of BEC circuit. The reduced number of gates of this work offers the great advantage in the performance. This paper proposes a scheme that increases the speed. The simulation and synthesis results are done by using Xilinx-ISE 8.1 version.

VI. ACKNOLEDGEMENT

I would like to express my deepest appreciation to my Professor and guide K.Babulu. He continually and convincingly render his knowledge and experience to develop this paper. His word of guidance and encouragement throughout the course has given me courage and confidence in my practice. Without his guidance this paper would not have been possible.
I would like to thank all the professors who are helped me in learning software and hardware tools.
An addition a thank you to friends and family members who give their valuable suggestions to enhance my knowledge. I also thank the University of JNTUK who introduce me.
Finally I take this opportunity to enhance my skills and knowledge.

REFERENCES

[1]. Yajuanhe,chip-hongchang, and jiangmin "An Area Efficient 64-bit Square root Carry-Select adder for low power applications".
[2]. 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.
[3]. Samir palniktar,"verilog Hdl:A guide to digital design and synthesis"2005,2nd Edition.
[4]. Pucknell Douglas A, Eshraghan, Kamran, "Basic VLSI Design, "Third edition 2003, PHI Publication,pp.242-243.
[5].O.J. Bedrij, "carry-select adder," IRE Trans. Electron. Comput, p.p340–344, 1962.
[6].M. Alioto,G.palumbo,"Analysis and comparision on full adder block in submicron technology,"IEEE Trans.VLSI systems,Vol.10,pp.806-823,Dec.2002.
[7].Kuldeep rawat,Tarak Darwish and Magdy Bayoumi, "A Low power and reduced area Carry select adder", 45th Midwest symposium on circuits and Systems, vol,1.p.p.467-470,march 2002.
[8].K.Rawwat,t.darwish and M.Bayoumi,"a low power carry select adder with reduced area",proc.of Midwest Symposium on Circuits and Systems,pp.218-221,2001.

M.N.V ANUSHA is currently pursuing her Masters Degree in Instrumentation and Control Systems specialization in Jawaharlal Nehru Technological University, Kakinada. She Pursued her B.Tech in Electrical and Electronics Engineer from G.I.E.T Engineering college ,Rajahmundry.

DR. K.BABULU is working as Professor of ECE Department and Controller Examination in Jawaharlal Nehru Technological University, Kakinada. He Pursued his M.Tech and Ph.D in VLSI & Embedded System Field of specialization.

ISSN: 2278 – 7798