ISSN 2347 - 3983

Volume 8, No. 1 January 2020

International Journal of Emerging Trends in Engineering Research

Available Online at http://www.warse.org/IJETER/static/pdf/file/ijeter03812020.pdf

https://doi.org/10.30534/ijeter/2020/03812020

Low Power Carry Look-Ahead Adder using Transmission Gate Multiplexer

M Siva Kumar[1], Fazal Noorbasha[2], Syed Inthiyaz[3], M. Jameela[4], A. Sandhya[5], Md. Imran[6], Sanath Kumar Tulasi[7]

Associate Professor[1,2,3,7], Student[4,5,6]

[1,2,3,4,5,6,7] Department of ECE, Koneru Lakshmaiah Education Foundation, Vaddeswaram, Guntur.

syedinthiyaz@kluniversity.com

ABSTRACT

We know that among all the adders, carry look-ahead adder (CLA) is the quick adder. In this paper we have implemented the CLA circuit for 32 bits. The carry look-ahead adder expands execution by diminishing the time it takes to perform addition operation on bits. By using full adder construction with ex- or multiplexer gates, it can be contrasted with the easier. In carry look-ahead adder, the carry bit of present full adder (FA) is always depends on its previous one carry. Sometimes, however, the present carry bit is decided on board the addition bit, and each bit information must delay until the preceding carry is calculated to start its execution. It measures one or more carry bits before performing the addition that decreases the delay time for larger bits. The results indicated through simulation shows that our proposed design has more advantages than conventional circuits design in terms of power consumption, speed and area.

Key words: CLA, FA, Power.

1. INTRODUCTION

Adders are frequently used in electronics to add data to the processor in standard computers, and many algorithms like FIR, IIR are most commonly used in different electronic applications such as digital image and signal processing. As a VLSI programmer the main goal should be to optimize the chip area using efficient optimization techniques and then the next step is to improve processing speed so as to accomplish fast calculations. Addition is the widely used arithmetic process on CPUs, digital signal CPUs and in digital computers. The synthesis of all other arithmetic processes can be done using adders. The arithmetic logic unit is the central processing unit where all the mathematical operations and logical operations are performed. The binary adder structures thus become an exceptionally basic equipment unit with regards to the effective implementation of an arithmetic unit. In any computer arithmetic book, it looks like there are few different circuit structures with various execution qualities that are commonly used.

In VLSI circuits, there is a lot of research going on to reduce power usage. There are 3 performance parameters to optimize the development of VLSI styles. They are area, speed and Power consumption. In addition, various kinds of adders are discussed that includes carry skip adder (CSK), carry save adder (CSA), ripple carry adder (RCA), carry select adder and carry bypass adder.

Present, addition have several logic styles, like ripple adder, carry select adder etc. The Carry-sum of larger bits depends on the carry-out of smaller significant bits, resulting in a more propagation delay.

Although a lot of research has been done on binary adder structures, studies are based on their high-performance analysis. Among a greater number of adders, we implemented VHDL code for Carry-look ahead 4-bit and 32-bit in this full adder is designed by using EX-OR gate and multiplexers. The adder structures can be classified into two essential classes in terms of delay time and power consumption. The first class consists of the full adder design with EX-OR and multiplexer. In the second class this full adder is instantiated in the 4-bit carry look a-head adder. From this we can optimize the power utilization and delay time.

1.1 Final Stage

Full Adder is an adding machine that adds three (3) inputs and gives the sum and carry of two (2) outputs. The three inputs are A, B and C (IN). The execution output is referred as S and C (OUT), which is referred to as sum and output carry. The full adder logic is designed to take together eight inputs in order to develop a byte-wide (8) adder and then cascade the bit information from one adding machine to the other.

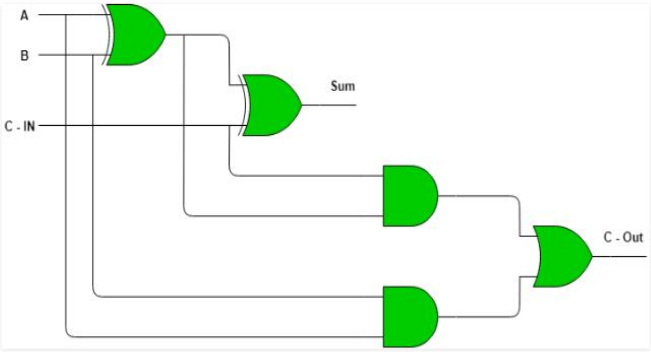


Figure 1: Full adder design

2. RELATED WORK

The output carry of each state full adder is given as input to the next state in parallel adders. Hence, these adders it is not possible to produce the sum and output carry of any state except a input carry is available for that particular state. So, for computation to occur, the circuit has to wait up to the carry bit is propagated to all the states. Forward, this reduces carry propagation delay of the circuit.

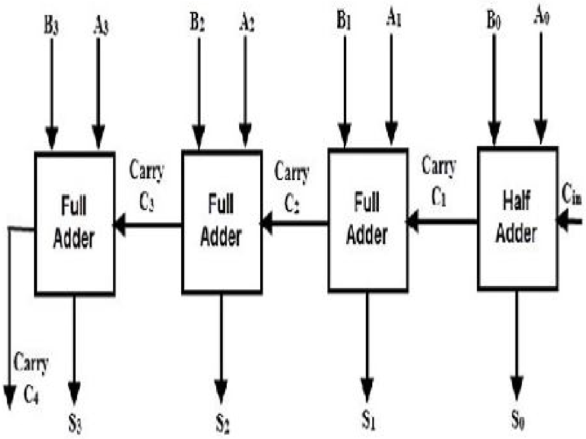


Figure 2: 4-bit Adder

Consider a four bit ripple carry adder circuit above. Here the sum S3 can be produced as soon as the inputs A3 and B3 are given. But carry C3 cannot be computed until the carry bit C2 is applied where as C2 depends on C1. Therefore, to produce final steady-state results, carry must propagate through all the states. This increases the carry propagation delay.

The adder's propagation delay is calculated as "the propagation delay of each gate times the number of stages in the circuit". In order to compute for more number of bits, more stages must be added, which makes the delay much worse. Hence, to solve this situation, Carry Look-ahead Adder was introduced.

The carry propagated Pi is related to the carry propagation from Ci to Ci+1. It can be calculated from Pi = Ai ⊕ Bi. This adder's truth table can be derived from the modification of a full adder's truth table.

Si and Ci+1 indicates sum and carry are given as below by using the terms Gi and Pi

$$Si = Pi \oplus Gi.$$
$$Ci + 1 = Ci.Pi + Gi.$$

Therefore, the equations of C1, C2, C3, and C4 carry bits can be written as

$$C1 = C0.P0 + G0.$$
$$C2 = C1.P1 + G1 = (C0.P0 + G0).P1 + G1.$$
$$C3 = C2.P2 + G2 = (C1.P1 + G1).P2 + G2.$$
$$C4 = C3.P3 + G3 = C0.P0.P1.P2.P3 + P3.P2.P1.G0 + P3.P2.G1 + G2.P3 + G3.$$

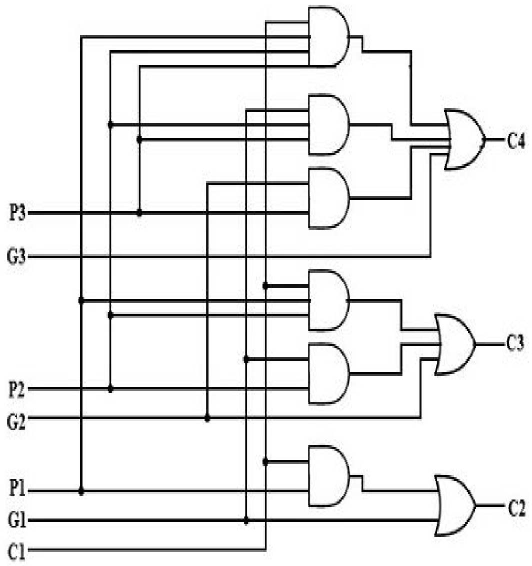


Figure 3: Carry Look-ahead Adder circuit

3. PROPOSED FULL ADDER

Two EX-OR cells and a transmission gate multiplexer are used to construct a full adder. The sum is provided by two EX-OR gates and 2 ×1 Multiplexer (MUX) is used to produce carry output.

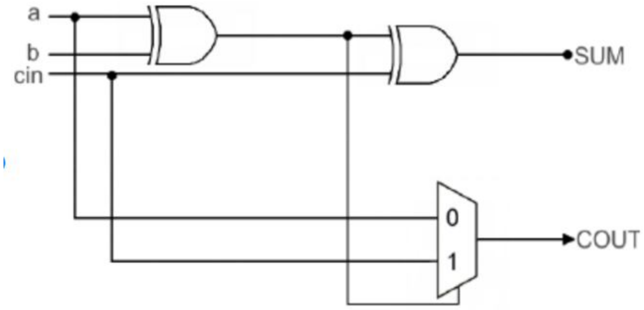


Figure 4: Proposed Full Adder

3.1 Two input Multiplexer Design

The two (2) input multiplexer is a hybrid circuit with two data inputs and produces the single output depending on the control or selection of input Logn (base2) selection lines for n input lines, otherwise we can also define as for 2n input lines, require n selection lines.

It is also called as "Parallel to serial converter, Data selector, universal logic circuit and multiple to one circuit" multiplexer. Multiplexers are primarily used to increment the amount of data that can be transmitted in a certain amount of time over the network.

3.2 EX-OR Gate

EX-OR gate is one of the logic gate which gives output as either logic HIGH (bit'0') or LOW (bit'0') this gate performs an exclusive OR. It shows the un similarity function, that means if the inputs are not alike the resultant output will be HIGH (or ON state) otherwise result will be LOW (or OFF state). The result of the XOR must be either ONE or ZERO but not both.

4 PROPOSED CARRY LOOK A-HEAD ADDERS (CLA)

To speed up addition required to determine the carry to the most significant bits soon. There are various schemes for finding out the carry; the worst-case situation is a depends on the number of bits log2 in the adder. The signals proposed must be quick because they are in order to pass through many gates, but they have more number of gates to predict the correct carry. The thing is to understand that the fast carry schemes is to memorize as changes of inputs effects the execution of hardware in parallel.

Using the first abstraction level: generate and propagate fast carry systems decreases the complication of equations for the complexity of the hardware. CLA depends on in its implementation and level of abstraction.

Equation for 1st step is:

$$c_{i+1} = (b_i.c_i) + (a_i.c_i) + (a_i.b_i) = (a_i.b_i) + (a_i+b_i).c_i$$

By using second stage of abstraction, first we consider that a single building block is a four bit adder with its CLA logic. So, if our connection fashion is like ripple carry for 16-bit adder, then that adder will give quicker response than its original which needs a lesser hardware. At higher level we must prefer this faster adder.

In ripple carry adder provides more delay, because it always depends on its previous state carry out value. So, in our circuit carry is not depend on its previous state carry value for the quicker response we provide generate and propagate values by using their operation we designed quick response adder.

5. SIMULATION AND RESULTS

Using the Xilinx tool, the various bits of carry look a-head adder are simulated. It is to ensure the design circuits operate with the correctness of required specifications. The proposed model has been simulated for various bits using Xilinx and simulation results are to prove how various bits are simulated. Consequently, power utilization and carry propagation delay in a circuit are varied.

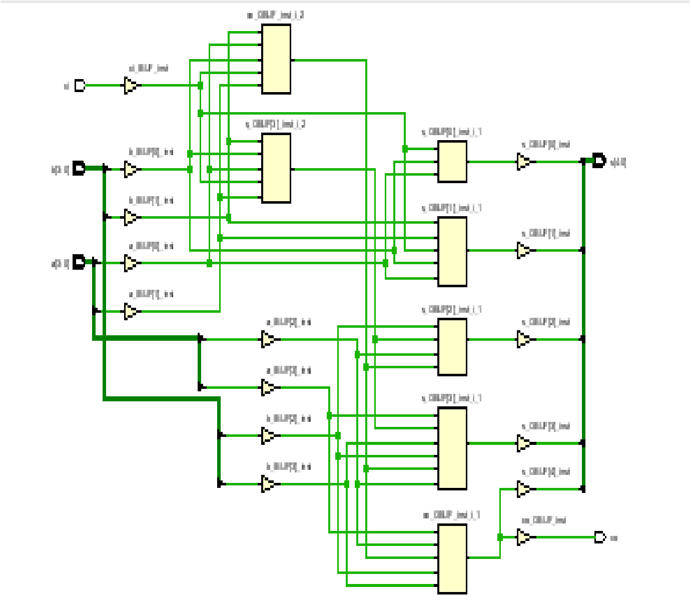


Figure 5: Existing Method of CLA for 4bit

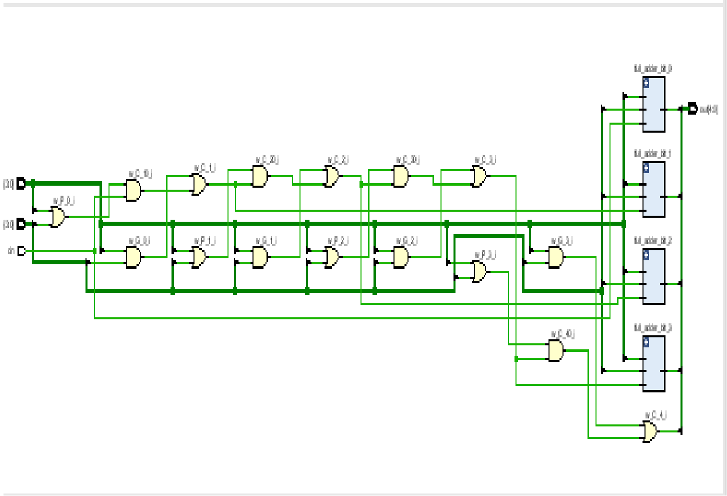
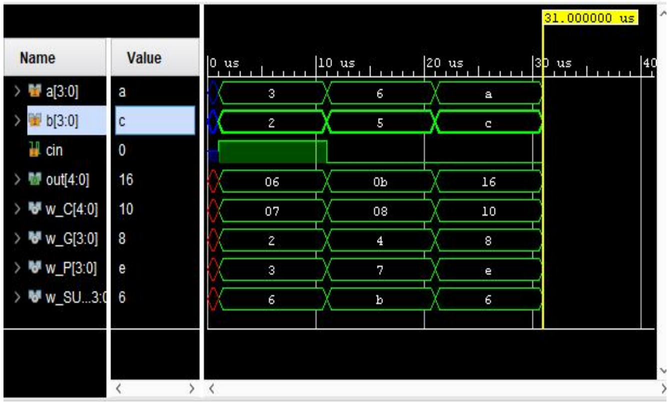


Figure 6: Proposed Method of CLA for 4bit



```
Name         Value    0 us        10 us       20 us       30 us   40
                                                    31.000000 us
a[3:0]       a           3           6           a
b[3:0]       c           2           5           c
cin          0
out[4:0]     16          06          0b          16
w_C[4:0]     10          07          08          10
w_G[3:0]     8           2           4           8
w_P[3:0]     e           3           7           e
w_SU...3:0   6           6           b           6
```

Figure 7: Simulation waveform of CLA for 4bit

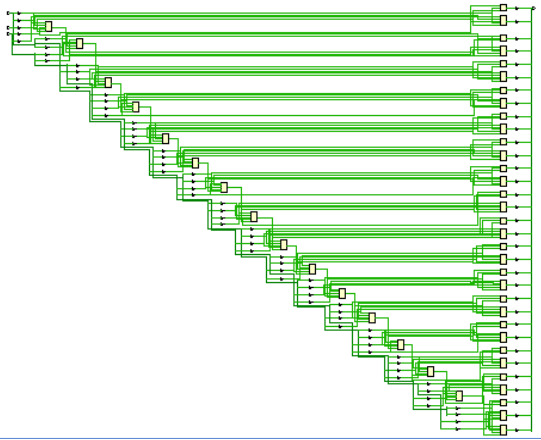
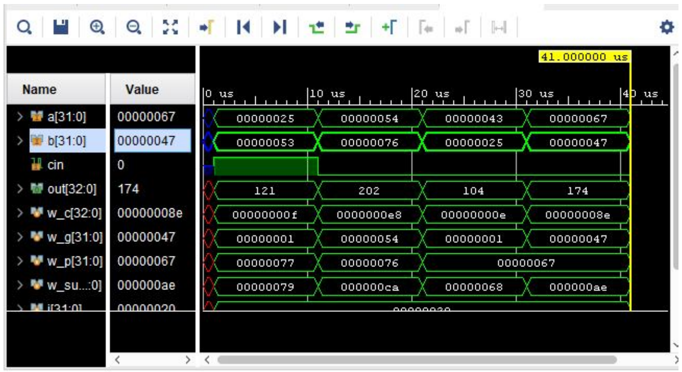


Figure 8: CLA schematic for 32bit



Name	Value
a[31:0]	00000067
b[31:0]	00000047
cin	0
out[32:0]	174
w_c[32:0]	0000008e
w_g[31:0]	00000047
w_p[31:0]	00000067
w_su...:0]	000000ae
i[31:0]	00000020

\

Figure 9: Simulation waveform of 32 -bit CLA

Table 1: Tabular Column

Method	Existing	Proposed
No. of LUTs	5	4
Power	3.14W	2.96W
Path Delay	9.718ns	9.65ns

4. CONCLUSION

In this paper we proposed that CLA adder with full adder design based and compare the path delay and power. By implementing and comparing them with different parameters such as Area, Delay and then Area Delay Product, studied various bits of adders. On contrasting different performance metrics of adders for various lengths with Xilinx and Verilog as a synthesis tool, the carry look a-head adder has the less power consumption and area-delay product.

ACKNOWLEDGEMENT

Authors like to express their gratitude to department of ECE of K L University and DST through ECR/2016/000569, SR/FST/ETI-316/2012 and EEQ/2016/000604.

REFERENCES

1. Siva Kumar, Syed Inthiyaz, Ch. Krishna Vamsi, Sk. Hasane Ahammad, K. Sai Lakshmi, P. Venu Gopal, A. Bala Raghavendra 2019. Power Optimization using Dual SRAM Circuit. 2019 International Journal of Innovative Technology and Exploring Engineering.
2. M.Siva Kumar, Syed Inthiyaz, P.Venkata Krishna ,Ch. Jyothsna Ravali, J.Veenamadhuri, Y.Hanuman Reddy, Sk. Hasane Ahammad 2019. Implementation of Most Appropriate Leakage Power Techniques In VLSI Circuits Using NAND and NOR Gates. 2019 International Journal of Innovative Technology and Exploring Engineering
3. Prasanna kumar.P,Siva kumar.M,"Implementation of digital beam former software on FPGA based system" 2018 journal of advanced research in Dyanamical and control systems
4. R. Bisht, P. Aggarwal, P. Karki and P. Pande. 2016. 4 SRAM array design using CMOS logic and differential sense amplifier with low power and noise resistance. 2016 International Computing, Communication and Automation Conference (ICCCA), Noida. Pages 1474-1478. https://doi.org/10.1109/CCAA.2016.7813954
5. D. Ho, K. Iniewski, A. Ivanov, S. Kasnavi, S. Natarajan. 2006. Ultra-low 90 nm 6 T SRAM cell for applications in the wireless sensor network. 2006 IEEE International Circuit and Systems Symposium, Kos Island. P. 4, doi: ISCAS 10.1109.2006.169353.
6. A. I. Shwartz and A. Morgenshtein. Fih. 2010. Gate Diffusion Input (GDI) logic in the nanoscale process of standard CMOS. 2010 IEEE 26th Electrical and Electronic Engineering Convention in Israel, Eliat. Pp. 000776-000780.doi: 2010.5662107.10.1109/EEEI.
7. N. Lindert, T. Sugii, S. Tang and Chenming Hu. 1999. In IEEE Journal of Solid-State Circuits. 34(1): 85-89, doi: 10.1109/4.736659. Lindert, T. Sugii, S. Tang and Chenming Hu. 1999. https://doi.org/10.1109/4.736659
8. K. C. Shah, D. N. Gandhi and B. H. Nagpara. 2013. 4-bit SRAM model and analysis in Deep Submicron CMOS Technologies. Journal of Electronics and Communication Technology Data, Knowledge and Study. 2: 812-817.
9. P.S. Bellerimath and R. M. Banakar. 2013. 4X4 SRAM Flash Array Architecture using 180 nm Technology. European Journal of Current Engineering and Technology. pp. 288-292.
10. Syed Inthiyaz, M V D Prasad, R. Usha Sri Lakshmi, N.T.B. Sri Sai, P. Pavan Kumar, Sk Hasane Ahammad, "Agriculture based plant leaf health assessment tool: A

Deep Learning perspective" International Journal of Emerging Trends in Engineering Research, 7(11), November 2019, 690-694. https://doi.org/10.30534/ijeter/2019/457112019

11. Ahammad, S.H., Rajesh, V., Venkatesh, K.N., (...), Rao, P.R., Inthiyaz, S." Liver segmentation using abdominal CT scanning to detect liver disease area" International Journal of Emerging Trends in Engineering Research, 7(11), November 2019, 664-669. https://doi.org/10.30534/ijeter/2019/417112019
12. Prasad, M.V.D., Inthiyaz, S., Teja Kiran Kumar, M., (...), Kumari, R., Ahammad, S.H.. "Human activity recognition using deep learning" International Journal of Emerging Trends in Engineering Research, 7(11), November 2019, 536-541. https://doi.org/10.30534/ijeter/2019/227112019
13. Syed Inthiyaz, Sanath Kumar Tulasi, R.S.L.Jayanthi, Ch.sahitya, Ch.jyothi, "Design Of Bi-Trigger Sram Using Schmitt Trigger For Low Power 13t Cmos Application. INTERNATIONAL JOURNAL OF SCIENTIFIC & TECHNOLOGY RESEARCH VOLUME 8, ISSUE 12, DECEMBER 2019,1466-1471.
14. Myla, S., Marella, S.T., Swarnendra Goud, A., (...), Kumar, G.N.S., Inthiyaz, S. "Design decision taking system for student career selection for accurate academic system" in International Journal of Recent Technology and Engineering, VOLUME 8, ISSUE 09, SEPTEMBER 2019, pp-2199-2206
15. Prasanna kumar.P,Siva kumar.MImplementation of digital beam former software on FPGA based system 2018 journal of advanced research in Dyanamical and control systems
16. Mohanachandrika, O., Siva Kumar, M, "Design of high speed single ended 6T and 8T SRAM cells" in International Journal of Advanced Trends in Computer Science and Engineering, volume 8 Issue 3, June 2019, pp 470-478 https://doi.org/10.30534/ijatcse/2019/21832019
17. M.Siva Kumar, Syed Inthiyaz, J. Dhamini, A.Sanjay,U.Chandu Srinivas, "Delay Estimation of Different Approximate Adders using Mentor Graphics" in International Journal of Advanced Trends in Computer Science and EngineeringVolume 8, No.6, November – December 2019,pp:3584-3587. https://doi.org/10.30534/ijatcse/2019/141862019
18. Siva Kumar M., Inthiyaz S., Narsimha Nayak V., Bhavani M., Charan Teja K., Rajesh S.J.S., Eswar Reddy K., Sruthi Keerthana G." Analysis of low power conditional sum adder" in Indian Journal of Science and Technology, Volume 9, Issue 17, article no:93033. https://doi.org/10.17485/ijst/2016/v9i17/93033
19. Murali Krishna B., Siva Kumar M., Rajesh J., Inthiyaz S., Mounica J., Bhavani M., Adidela C.N., "FPGA implementation by using XBee transceiver" in Indian Journal of Science and Technology, Volume 9, Issue 17, article no:93032.
20. Inthiyaz, S., Madhav, B.T.P., Madhav, P.V.V. "Flower segmentation with level sets evolution controlled by colour, texture and shape features" in Cogent Engineering 4(1) https://doi.org/10.1080/23311916.2017.1323572
21. Inthiyaz, S., Madhav, B.T.P., Kishore, P.V.V. "Flower image segmentation with PCA fused colored covariance and gabor texture features based level sets" in Ain Shams Engineering Journal 9(4), pp. 3277-3291 https://doi.org/10.1016/j.asej.2017.12.007
22. Hasane Ahammad, S., Rajesh, V., Hanumatsai, N., Venumadhav, A., Sasank, N.S.S., Bhargav Gupta, K.K., Inithiyaz, S." MRI image training and finding acute spine injury with the help of hemorrhagic and non hemorrhagic rope wounds method" in Indian Journal of Public Health Research and Development 10(7), pp. 404-408 https://doi.org/10.5958/0976-5506.2019.01603.6