Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 4, Issue. 1, January 2015, pg.284 – 290

RESEARCH ARTICLE

An Efficient Performance Analysis of Different Adder Topologies

T. Manoj Kumar[1], S.Navaneethan[2], A.C.Murugesapandian[3]

[1]Assistant Professor, Department of ECE, Karpagam College of Engineering
[2]Assistant Professor, Department of ECE, M.A.M College of Engineering
[3]Assistant Professor, Department of ECE, Annai Mookambigai College of Engineering
srimanojkumar@gmail.com [1], jssnavi.37@gmail.com [2], acpandi86@gmail.com [3]

Abstract -This survey deals with the study of the comparative analysis of the performance of different adder logics. Here for our analysis we have considered the performance of four different adders such as carry look-ahead adder (CLAA), carry select adder (CSLA), Modified Carry select adder and carry save adder. The above design of multiplier multiplies two 16-bit unsigned integer values and gives a product term of 32-bit values. These multipliers have been designed in Verilog, simulated on Modelsim and synthesized on Xilinx FPGA Spartan 3E xc3s500E and their area; power and delay are calculated and compared.

Keywords: carry look ahead adder, carry select adder, modified carry select adder, propagation delay, Synthesis, Simulation

I. INTRODUCTION

Improving the processing performance and enhancing the reduction of power dissipation of the systems are the most predominant design challenges for all multimedia and signal processing (DSP) applications, in which multipliers mostly decides the system's performance and power dissipation. Therefore several modifications are added to the multiplier design to improve the speed and reduce its power dissipation and delay which significantly improves the system performance.

Multiplication results due to the repeated form of the addition and shifting operations. Since addition is a fundamental operation for any digital multiplication, a fast, area efficient and accurate operation of a digital system is greatly decided by the performance of the resident adders. Therefore modification in the adder results in the performance variations in the multiplier.

Performance of an adder is decided by its speed and often speed of an adder can be directly related by its carry propagation from LSB to MSB. In this project we have selected three different adders which has its own variation in the speed, area and power consumption to determine the performance of multipliers designed with the above three adders.

In this project we are going to evaluate the performance of multiplier designed with different adders, since multiplication can be done by repeated addition and shifting operations. Using different adders we can realize the different architecture of multipliers.

We have selected Carry look ahead adder, Carry select adder, Carry save adder and multipliers are designed using these adders and performance was evaluated by considering input as 16 bit.

II. CARRY LOOK AHEAD ADDER

Carry look ahead addition was introduced by Weinberger & Smith in 1956. Carry look ahead adder is an adder circuit which detects the carry's well in advance with the help of some additional logical circuit. By using carry look ahead adders the need for carry propagation in the adder is avoided and the latency of one addition is equal to the gate delay of a full adder, independently of the data word length. Output can be expressed only with the help of input. The carry look ahead adder finds the carry of corresponding bits by using two terms. Those two terms decides the carry of corresponding bits. They are Generate term and Propagate term.

Generate and propagate term of a corresponding bit can be found from the below expressions.

$G_i = A_i \cdot B_i$ called the generate function

If G^i=1 then Cout=1 Independent of previous carry

$P_i = A_i \oplus B_i$ called the propagate function

If Pi=1 then Cout=C (ie previous carry)

The two outputs of carry look ahead adder can be calculated from the above two terms.

$S_i = P_i \oplus C_i$ & $C_{i+1} = G_i + G_{i-1}.P_i$

For example if we need to add the third bit of two four bit numbers we can perform addition directly without waiting for the carry generated from the result of second digit addition. That carry can be generated priory with the help of generate and propagate terms.

$C_2 = G_2 + G_1.P_2 + G_0.P_1.P_0$

Where $G_2 = A_2.B_2$, $P_2 = A_2 \oplus B_2$, $G_1 = A_1.B_1$, $P_1 = A_1 \oplus B_1$, $P_0 = A_0 \oplus B_0$

In the previous expression of C_2 there is no necessary C1.Therefore we can calculate the C2 as function independent of C1 and C0. Thus generally carry at a particular digit can be calculated independently of previous carries in carry look ahead adders. Thus delay encountered in ripple carry adder is eliminated in the carry look ahead adder with the cost of additional circuitry. Thus Carry look ahead adder occupies more area than that of ripple carry adder but its performance is higher.

III. CARRY SELECT ADDER

Carry select adder eliminates the delay of waiting the computation of i^{th} bit until the outputs of previous stages are obtained. Carry select adder is a adder which has two sets of adder circuit which precomputes the output for the two possible carry input either zero or one. Sum and carry output is calculated by assuming one carry input as 1 and other as 0. Once the carry input is available, with the help of 2 to 1 multiplexer one of the output is selected based on the value of Cin. Since the output can be obtained even before the third input arrival, this adder circuit is efficient than that of carry look ahead adder. It minimizes the propagation delay but the circuit is complex and occupies more area because of the two carry look ahead adder and a mux circuit. If we're willing to throw hardware at the problem, we can speed things up by duplicating adders .It is not efficient in area.

IV. MODIFIED CARRY SELECT ADDER

Carry select adder which was discussed in previous section can be modified. In carry select adder we are having two adder structures which performs the addition operation by assuming the carry of each stages as either one or zero. Modification is done in the adder structure which adds the input values with carry input as zero. Since the carry input is zero the full adder at the first stage can be replaced with half adder since both yields the same output in this case. By the usage of this modification in each stage of adder a significant amount area can be saved and propagation delay also can be reduced. This adder is more efficient in area and delay than that of conventional carry select adder.

V. CARRY SKIP ADDER

To increase the speed of addition operation the carry is skipped from one position to other with the inclusion of additional logic circuit, thus eliminating the delay occurred in calculating that skipped carry. While considering group of adders, CSA eliminates the propagation delay by skipping over consecutive stages of adder. Carry skip logic is added in each stage to determine when carry input can be directly passed to next block.

Syntax of the carry skip logic for a four bit adder is given as

If(X1 or C3 =1) ; Where X1=C0.($P_1 = A_1 \oplus B_1$)

Cout=1

Else

Cout=0

Thus either Carry input and propagation term of first bit or carry output of fourth bit decides the final carry output of an adder stage.Delay occurred in carry look ahead adder is minimized in carry skip adder but it occupies more area than that of carry look ahead adder because of the additional carry skip logic in each adder stage.

VI. SIMULATION RESULTS

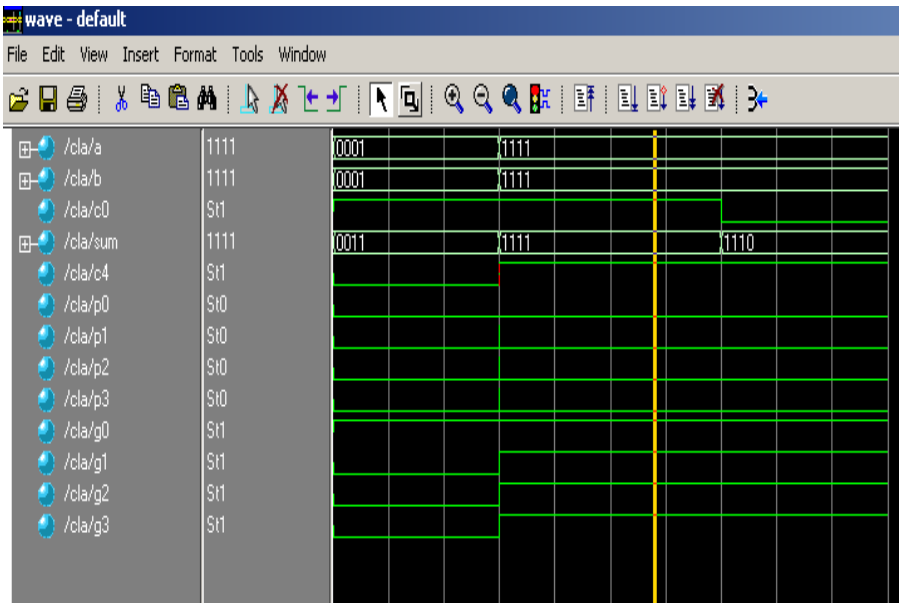

Fig.1 Carry look ahead adder

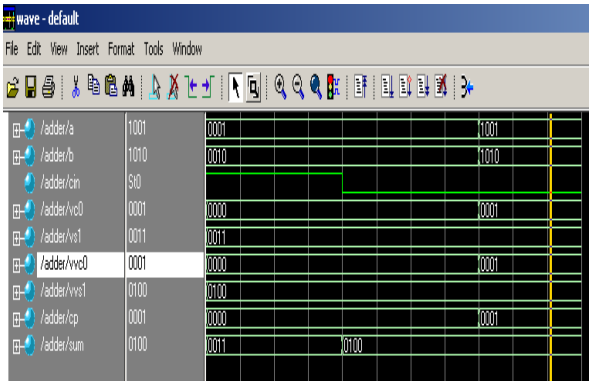

Fig.2 Carry select adder

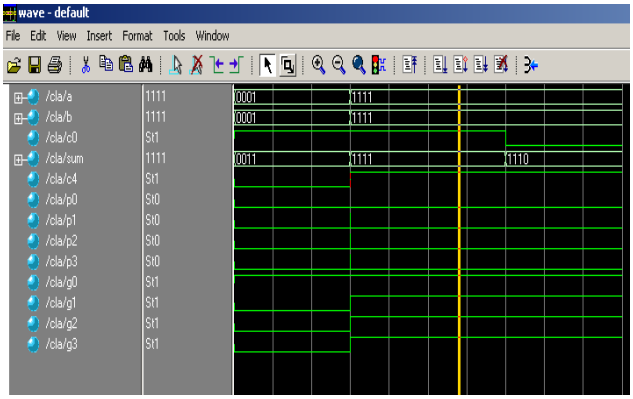

Fig.3 Modified carry select adder

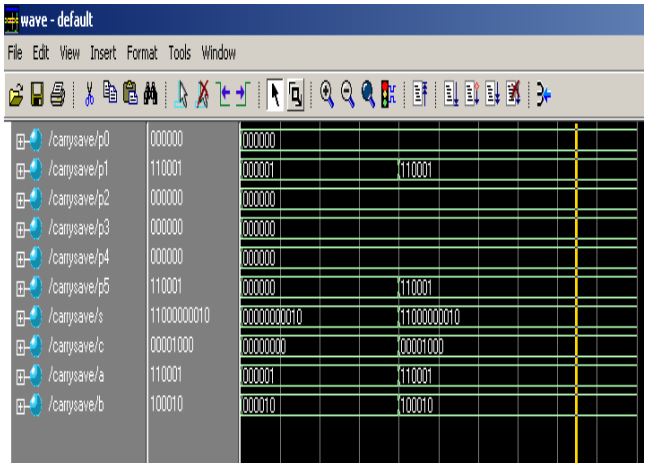

Fig.4 Carry skip adder

VII. SYNTHESIS RESULTS

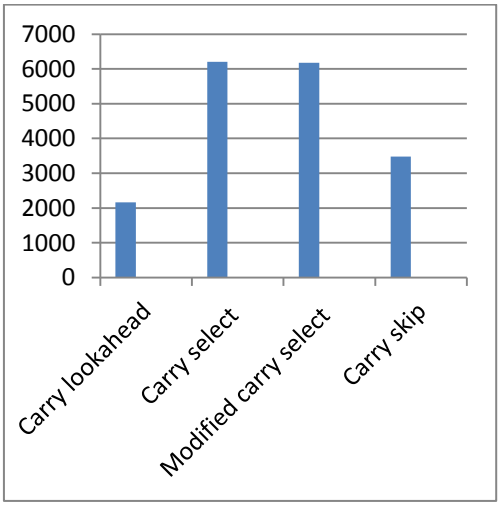

Fig.5 Area occupied

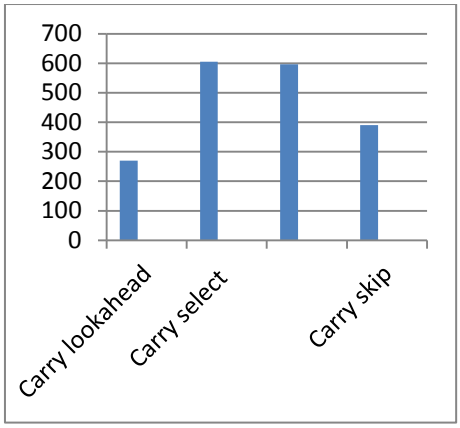

Fig.6 Gate used

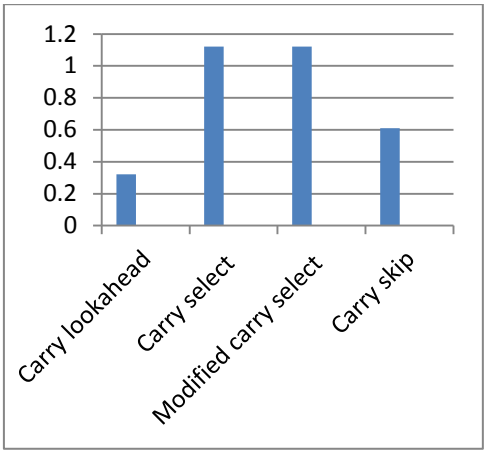

Fig.7 Propagation delay

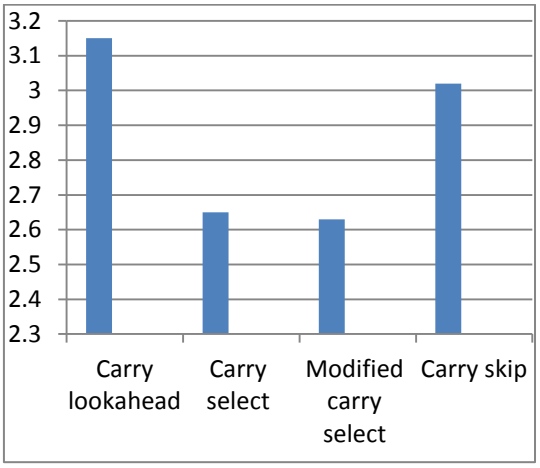

Fig.8 Power consumed

TABLE I
COMPARISON OF SYNTHESIS RESULTS OF DIFFERENT ADDERS.

S.No	Adder	Area (um^2)	Gates used	Delay (ns)	Power (mW)
1	Carry Look Ahead	2160	270	3.15	0.321
2	Carry Select	6205	605	2.65	1.12
3	Modified Carry Select	6180	596	2.63	1.12
4	Carry Skip	3480	390	3.02	0.61

CONCLUSION:

In this work we have studied and demonstrated the performance of different adder structures. From the synthesis results it is clear that carry look ahead adder is an area efficient adder which has lesser number of gates in its structure but the propagation delay of carry look ahead adder is very high. Therefore it is not suitable for the applications which requires high speed operation. When the speed is an important criteria we can go for modified carry select adder which is area efficient and has lesser propagation delay when compared to that of carry select adder. This work can be extended by analyzing the performance of these adders for a particular operation like multiplication in image processing, signal processing etc..

REFERENCES

1. Shreehari veeramachaneni, M B Srinivas, " New improved 1 Bit full adder cells" IEEE 2008

2. Romana Yousuf and Najeeb-ud-din, " Synthesis of carry select adder in 65nm FPGA",IEEE

3. N. Weste and K. Eshraghian, Principles of CMOS VLSI Design - A Systems Perspective, Addition-Wesley Publishers, 1985.

4. LSI Logic Inc., G10-p Cell-Based ASIC Products Databook, 1996.

5. B.Ramkumar, Harish M Kittur, P.Mahesh Kannan, "ASIC Implementation of Modified Faster Carry Save Adder", European Journal of Scientific Research ISSN 1450-216X Vol.42 No.1, pp.53-58,2010.

6. David J. Willingham and izzet Kale, "A Ternary Adiabatic Logic (TAL) Implementation of a Four- Trit Full-Adder,IEEE, 2011.

7. G.Shyam Kishore, "A Novel Full Adder with High Speed Low Area", 2nd National Conference on Information and Communication Technology (NCICT) 2011 Proceedings published in International Journal of Computer Applications® (IJCA).

8. Pudi. V, Sridhara., K, "Low Complexity Design of Ripple Carry and Brent Kung Addersin QCA",Nanotechnology,IEEE transactions on,Vol.11,Issue.1,pp.105-119,2012.

