JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

IMPLEMENTATION OF HIGH-SPEED WALLACE TREE MULTIPLIER USING PARALLEL PREFIX ADDERS

[1]D.Sivakumar, [2]K.Dhanumjaya
[1]PG Scholar, [2]Professor,Dept. of E.C.E
[1,2]Audisankara College of Engineering and Technology(Autonomous), Gudur, SPSR Nellore(dt.) A.P

ABSTRACT:

The design of high-performance adders has experienced a renewed interest in the last few years, among high performance schemes, parallel prefix adders constitute an important class. They require a logarithmic number of stages and are typically realized using AND-Orl ogic, moreover with the emergence of new device technologies based on majority logic, new and improved adder designs are possible. However, the best existing majority gate-based prefix adder incurs adelayof2log2(n-1) (due to the nth carry); this is only marginally better than a design using only AND-OR gates (the latter design has a 2log2(n-1) gate delay). This initially shows that delay is caused by the output carry equation in majority gate-based adders that is still largely defined in terms of AND-OR gates. In this, two new majority gate-based recursive techniques are proposed. The first technique is based on a novel formulation of the majority gate-based equations in the employed group generate and group propagate hardware, which results in a new definition for the output carry, resulting in a reduction in latency. The second contribution of this publication reduces the circuit complexity of prefix adder designs by utilising the recursive features of majority gates (via a novel operator). Overall, the proposed methodologies result in the determination of an n-bit adder's output carry with just a log2 majority gate delay (n- 1).

I.INTRODUCTION

The design of high-performance multi-bit adders has been an active research topic for many years, schemes have been developed with the predominant goal of reducing the worst-case delay under a possible CMOS implementation. Existing multi-bit adders reduce the worst-case delay based on strategies such as unrolling the carry recurrence, and calculating the results prior to each possible carry input. Based on these strategies, several high-performance designs have been proposed in the literature. Among them, prefix adders constitute an important class, because they yield high-performance at a relatively small fan-out and hardware requirements. Prefix adders have been extensively used specially on critical paths due to a compact yet fast implementation. Traditional prefix addition (as well as other schemes, such as carry look ahead) are based on generate and propagate signals derived using AND-OR logic. An n-bit prefix adder requires only O(log2(n)) stages for the calculation of the delay. In terms of AND-OR gates, the Kogge-Stone and Ladner-Fischer adders incur a delay of (2log2(n-1) gates. Alternatives to AND-OR logic have

also been considered for arithmetic circuit designs. In particular, majority logic has been of, interest fromtheearly1960s. The realization of a one-bit adder using three majority gates and two inverters hasbeen proposed as well as techniques based on decomposition and rearrangement to majority element-based synthesis of networks with limited fan-in components. It has presented a geometric method that utilizes Veitch diagrams for synthesis using i-input majority gates fora variety of n-argument switching functions. An approach based on Logically Passive Self-Dual (LPSD) has been presented in an extension to this work has been presented in interest in majority logic has been revived recently in the context of digital design for several emerging nanotechnologies (such as domain wall nano magnets , resonant tunnelling diodes and quantum dot cellular automata (QCA) . A few multi-bit adder designs in QCA have been reported. However, the best existing majority gate-based n-bit adder design still incurs a delay of 2log2(n)-1 for the nth carry, so only slightly better than using merely AND-ORgates. A close examination of this design reveals the cause for this limitation, i.e., the AND-ORlogic has been primarily used to derive the majority gate-based designs. The goal of this project is to first redefine the output carry of an n-bit adder in terms of only majority gates for delay reduction; the proposed formulations are useful for parallel adders not for synthesizing general majority gate circuits. This project provides two contributions to adder design using majority logic by which new majority gate-based recursive techniques are proposed. The first contribution is based on a new definition for the majority gate equations of the group generate and group propagate signals, such that the output carry is generated at a reduced delay. The second contribution is based nonoperator that exploits novel recursive properties of majority logic to achieve saving in circuit complexity (as given by the number of majority gates required in a design) for prefix adders. Overall, the proposed formulation results in calculating the output carry of an n-bit adder with a reduced majority gate delay of log2(n -1). Moreover, the proposed approach leads to a reduction of 40 percent in normalized delay and 30 percent in circuit complexity (in terms of required majority gates) for multi-bit addition compared to the best existing design.

II.EXISTING MULTIPLIER:

A multiplier is an important building piece in the construction of digital systems. Several algorithms for implementing rapid multipliers have been documented in the literature. Another approach for creating an efficient multiplier is to use the VEDIC multiplication algorithm. The VEDIC multiplier is discussed in this paper. In VEDIC mathematics, there are three ways to implement multiplication. Only one way is generic and may be applied to all cases, while the other two are for specific cases. Urdhva Triyakbhyam is the main algorithm of Vedic multiplication. It is a generic multiplication formula that applies to all multiplication situations. The VEDIC multiplier is an excellent alternative to existing rapid multiplicative techniques. When compared to other techniques, VEDIC multipliers reduce both hardware and delay. Figure 1 depicts a 2-bit multiplier.

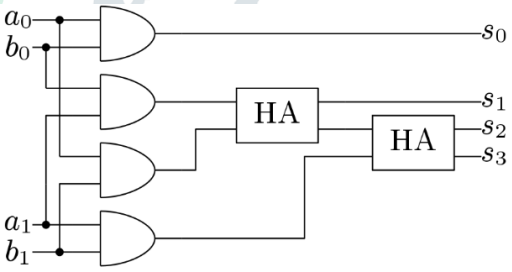

Figure 1: VEDIC multiplier for 2-bit data width

Drawbacks of Vedic multipliers are

1. Consuming more power.
2. Large delay.
3. Required more area to design.

III.TRADITIONAL WALLACE TREE MULTIPLIER

Most extensively used multiplier design in many processors and memory units is

Wallace tree structure [4]. The Wallace tree multiplication process majorly has two phases. In phase 1, the input numbers are applied to AND gate to produce partial products. These partial products are added in step-by-step process by using half and full adders in phase 2 to obtain final product output. In detailed multiplication process of Wallace tree multiplier is explained through Fig.1 for input size of 4-bits. The phase 1 comprises of generation of partial products through multiplying every bit of given input numbers with each other. Four rows of partial products are generated as the size of input is 4- bits. The phase 2 comprises of many sub phases of addition of the partial products obtained in phase. The addition operation is carried out using half and full adders. Initially in phase 2, the addition operation is performed on first three rows of partial products generated in phase 1 which generates result of two rows having sum terms in first row and carry terms in second row. Then, the last row of partial products of phase 1 result is added with the sum and carry row which again result two rows comprising of one sum row and one carry row. To acquire final product, the sum and carry row are added. The addition process is very important in the multiplication process. Carry must be propagated quickly to perform fast addition. [8] designed a Wallace tree multiplier with a Carry select adder. However, the carry propagation delay is greater in this existing methodology, which is the major disadvantage. To overcome this limitation, parallel prefix adders are used in place of half and full adders in the final stage of addition in phase 2 of this multiplier. The main motivation for this proposed design is to achieve Wallace tree multiplier architecture at a faster rate than existing designs.

IV.PROPOSED WALLACE TREE MULTIPLIER

A. Parallel Prefix adders:

In this paper, Wallace tree multiplier structure is modified by using parallel prefix adders to add partial products in final phase addition process to obtain final product. The reason behind to use PPA in place of full adders is to improve the speed of operation. In PPA, the carry input for the next bits is generated at a time with the help of parallel prefix carry tree which consists of black cells and grey cells. There are many types of PPAs are present whose basic design idea is originated from carry look ahead adder. The PPA consists of three main blocks as shown in fig 2.

In first block which is also called as pre-processing block, propagate(Pi) and generate(Gi)signals are generated by

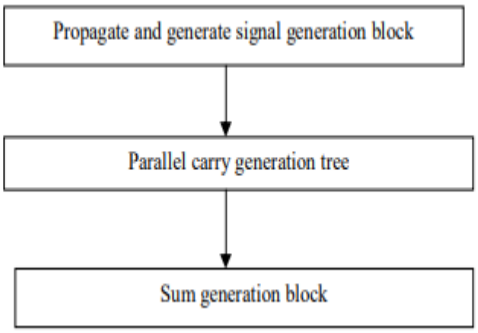

Fig. 2. Parallel prefix adder block diagram

V.COMPARISON TABLE:

S. No	ADDERS	Area (No. of LUT's)	Time delay(ns)	Power(W)	Speed
1.	Kogge Stone	23	11.059	0.203	4.68
2.	Sklansky	17	12.134	0.203	3.80
3.	Brent Kung	22	10.465	0.203	5.03
4.	Ladner Fischer	16	9.900	0.203	4.07
5.	Han Carlson	23	11.059	0.203	4.68

As shown in the above table total five multipliers were designed by using different adders i.e., kogge Stone Adder, Sklansky Adder, Brent Kung Adder, Ladner Fischer Adder and Han Carlson the proposed multipliers using PPAs are synthesized using Xilinx XST synthesizer. The synthesis report consists of area details in terms of number of LUTs occupied and delay in terms of nano seconds. From table it can be seen that Wallace tree multiplier using Ladner Fischer adder is having least delay and low number of LUTs occupied when compared to other structures.

VI.SIMULATION RESULTS

Kogge Stone Adder Simulation:

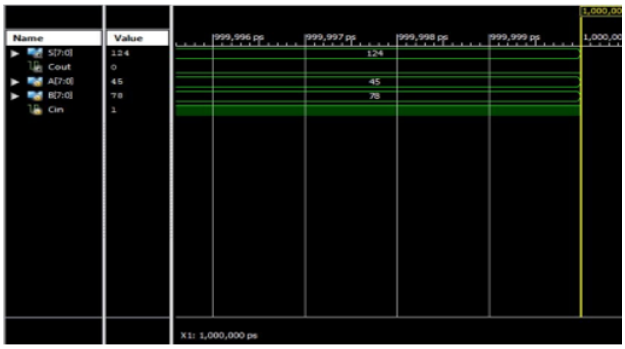

Fig 3: Simulated And Synthesized Result of kogge-stone adder

Sklansky adder Simulation:

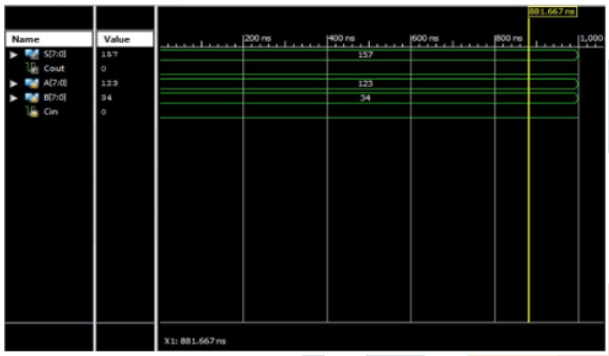

Fig 4: Simulated And Synthesized Result of Sklansky adder

Brent Kung Adder :

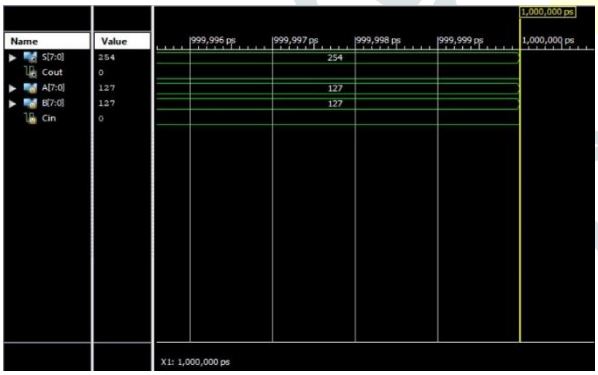

Fig 5 : Simulated And Synthesized Result of Brent Kung Adder

Ladner Fischer Simulation :

Fig 6 : Simulated And Synthesized Result of Ladner Fischer Adder

VII.CONCLUSION

A new majority gate-based approach for high performance adder design has been presented. The two contributions of this manuscript (the formulation of the carry output and the recursive prefix operator for majority logic) have resulted in a reduction in circuit complexity (as requiring a lower number of majority gates in an adder design) as well as lower propagation delay for the leading carry. The proposed strategy has been applied to various prefix adders including the Kogge-Stone, Ladner-Fischer and Brent-Kung adders. Itis observed that these new results achieve a reduction in delay of at least [log2n] over the bestexisting majority gate-based adders found in the technical literature. Specifically, reductions of 40 percent in delay and 30 percent in circuit complexity (in terms of the number of majority gates) has been accomplished for multibit adder schemes. As shown in Table , Is(n) remains constant and for very large values of n, Ic(n) and the normalized delay d(n) show considerable reductions, for example a nearly 50 percent for d(n) for the three prefix adders considered in this manuscript. Current research deals with the applications of these findings to emerging technologies and in particular, the implications on different applications requiring fast arithmetic processing.

REFERENCES

[1] N.Weste and D. Harris, CMOS VLSI Design. Reading, MA: Addison Wesley, 2004.

[2] N. Weste and K. Eshragian, Principles of CMOS VLSI Designs: A System Perspective, 2nd ed., Addison-Wesley, 1985-1993.

[3] Milos D. Ercegovac and Thomas Lang, "Digital arthimetic," Morgan Kaufmann, Elsevier INC, 2004

[4] J. M. Rabaey, Digtal Integrated Circuits—A Design Perspective. Upper Saddle River, NJ: Prentice-Hall, 2001. [5] Himanshu Bansal, K. G. Sharma, Tripti Sharma," Wallace Tree Multiplier Designs: A Performance Comparison Review",Innovative Systems Design and Engineering, Vol.5, No.5, 2014.

[6] Nehru, K., A. Shanmugam, and S. Vadivel. "Design of 64-bit low power parallel prefix VLSI adder for high speed arithmetic circuits." In 2012 International Conference on Computing, Communication and Applications, pp. 1-4. IEEE, 2012.

[7] Rakesh.S,K.S.Vijula Grace, "A comprehensive review on the VLSI design performance of different Parallel Prefix Adders" ScienceDirect, Materials Today: Proceedings 11 (2019) 1001–1009

[8] Kesava, R. Bala Sai, B. Lingeswara Rao, K. Bala Sindhuri, and N. Udaya Kumar. "Low power and area efficient Wallace tree multiplier using carry select adder with binary to excess-1 converter." In 2016 Conference on Advances in Signal Processing (CASP), pp. 248-253. IEEE, 2016.

[9] Sunil M, Ankith R D, Manjunatha G D and Premananda B S, "Design and Implementation of Faster Parallel Prefix Kogge-Stone adder",Int. Journal of Electrical and Electronic Engineering & Telecommunications, Vol.4, No.1, pp 116-121, 2015.

[10] Raghumanohar Adusumilli and Vinod Kumar K, "Design and Implementation of a high speed 64 bit Kogge-Stone adder using Verilog HDL", Int. Journal of Electrical and Electronic Engineering & Telecommunications, Vol.3, No.1, pp 13 – 18, 2014.

[11] Nurdiani Zamhari, Peter Voon, Kuryati Kipli, Kho Lee Chin and Maimun Huja Husin, "Comparison of Parallel Prefix Adder (PPA)", Proc.World Congress on Engineering 2012 (WCE 2012) Vol II, London, July 2012.

[12] David H. K. Hoe, Chris Martinez and Sri Jyothsna Vundavalli, "Design and Characterization of Parallel Prefix Adders using FPGAs",IEEE 43rd Southeastern Symposium on System Theory (SSST), USA, pp 168 – 172, March 2011.

[13] Jasmine Saini, Somya Agarwal and Aditi Kansal, "Performance, Analysis and Comparison of Digital Adders", Proc. IEEE Int. Conference on Advances in Computer Engineering and Applications (ICACEA), Ghaziabad, India, pp 80 – 84, 2015.

[14] Sudheer Kumar Yezerla and B Rajendra Naik, "Design and Estimation of delay, power and area for Parallel prefix adders", Proc. IEEE 2014 RAECS UIET Panjab University, Chandigarh, March 2014.

[15] R.Brent and H.Kung, "A regular layout for parallel adders", IEEE Transaction on Computers, vol. C-31, no.3, pp 260 – 264, March 1982.

[16] K.Babulu, Y.Gowthami," Implementation and Performance Evaluation of Prefix Adders using FPGAs", IOSR Journal of VLSI and Signal Processing, Vol. 1, Iss.1, pp 51 - 57, 2012.

[17] S. Sri Katyayani, Dr.M.Chandramohan Reddy and Murali.K, "Design of Efficient Han-Carlson-Adder", International Journal of Innovations in Engineering and Technology, Special issue on ETiCE, pp 69 – 75, 2016.