International Journal of Modern Engineering Research (IJMER)
www.ijmer.com Vol. 3, Issue. 5, Sep - Oct. 2013 pp-2626-2630 ISSN: 2249-6645

An efficient model for design of 64-bit High Speed Parallel Prefix VLSI adder

M. Mahaboob Basha[1], Dr. K. Venkata Ramanaiah[2], Dr. P. Ramana Reddy[3], B. Lokeshwar Reddy[4]

[1]*Assoc.Professor, ECE, AVR & SVR College of Eng. Technology, Nandyal, JNTUA, A.P.State*
[2]*Assoc. Professor & Head, Dept. of ECE, YSR Engg.College,Yogi Vemana University, Kadapa, A.P.State, India*
[3]*Assoc.Professor, Department of ECE, JNTUA, Anantapur, A.P.State, India*
[4]*PG Student, AVR & SVR College of Eng. Technology, Nandyal, JNTUA, A.P.State*

Abstract: *To make addition operations more efficient parallel prefix addition is a better method. In this paper 64-bit parallel prefix addition has been implemented with the help of cells like black cell and grey cell operations for carry generation and propagation. This process gives high speed computations with high fan-out and makes carry operations easier. Xilinx 14.1 vivado tool has been used for the simulation of proposed 64-bit adder. The comparison can be made with the help various range of inputs and the proposed parallel prefix adder has produced high speed computation and also efficient in terms of number of transistors and their topology and number of nodes.*

Keywords: *parallel prefix adder; dot operators and logic cells; high speed adder; 64- bit addition.*

I. INTRODUCTION

Parallel addition is the most important computation in many processors like microprocessors, DSPs, mobile devices and other high speed applications. The main theme of the parallel prefix adder is to reduce logic complexity and delay. However, improving the performance of Design constraint with other factors like area and power. Basing on the implementation of adder like Carry Look Ahead adder, the parallel prefix computation has been developed [1] by targeting high performance computations. Parallel prefix adder's terminology is equivalent to Carry Look Ahead addition, but the transistor topology is different. Parallel prefix adders have unique style of carry generation and carry propagation. The main theme of the parallel prefix addition is to accelerate the n-bit addition process in VLSI technology. Parallel prefix adders are desired in high speed arithmetic circuits and they are popular since twenty years. Parallel prefix addition involves in three steps. First, computation of carry generation and carry propagation signals by using number of input bits. Second, calculating all the carry signals in parallel i.e. prefix computation. Third, evaluating total sum of the given inputs. The entire mechanism consists of XOR gates, AND gates and OR gates. Black and grey cells plays key role in the generation and propagation of carry to a particular stage. Black cells consists one OR gate and two AND gates, whereas grey cell consists one AND gate and one OR gate. In some parallel prefix adders there will be another module as white cell, which is a replica of general half adder circuit.

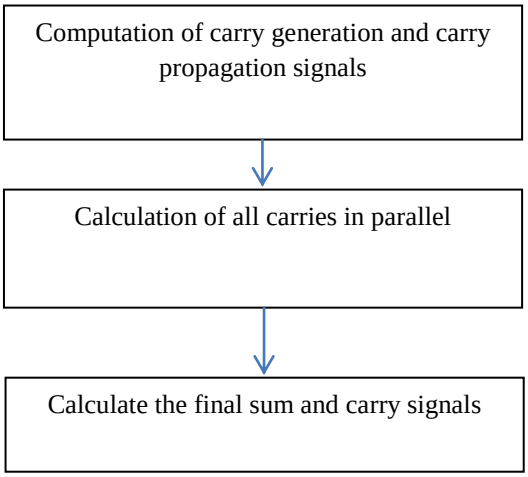


Figure 1: parallel prefix adder mechanism

II. EXISTED PARALLEL PREFIX ADDERS

Design of prefix addition network specifies the model of parallel prefix adder. The parallel prefix adder given by Kogg Stone gives low fan-out and high logic depth which leads to a complex network of prefix addition and also contains more number of interconnections. Brent Kung adder design gives minimal number of calculating nodes, but the design itself has maximum logic depth. The area will be reduced due to number of nodes but complexity will be high due to high logic

depth. Han-Carlson adder design is the combination of both Brent Kung and Kogg Stone adders, which gives balance between nodal count and logic depth. Knowles also presented an adder design, which gives low logic depth and improved fan-out. There is another adder which combines the benefits of prefix addition and Carry save adder i.e. sparse tree binary adder. Sklansky developed a topology which leads to low depth in the interconnecting nodes. Ladner and Fischer proposed a general design to construct a parallel prefix network with high depth when compared with Sklansky topology but achieved maximum fan-out for the critical path. Integer linear programming is also another method for parallel prefix computation. Mathew Ziegler and Mircea Stan proposed an adder for parallel prefix computation for minimizing delay and area known as best logarithmic adder for fan-out of two. There are some more adders designed by Haiku Zhu, Chung-Kuan and Ronald Graham, can produce minimal depth for n-bit adder. The proposing adder is about 64-bit adder with parallel prefix computation which is closely related to Ladner and Fischer adder.

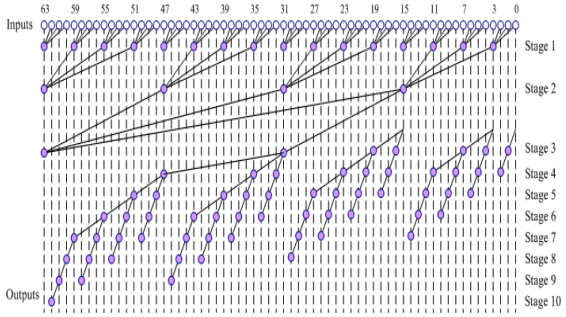


Fig2: existed 64-bit parallel prefix adder

III. PROPOSEDHIGH SPEED 64-BITPARALLEL PREFIX ADDER

The proposed high speed 64-bit parallel prefix adder is shown fig 2. Main theme of the design is to eliminate huge delays in overall for carry calculations. So the logic depth of the proposed design is optimal.The 64-bit parallel prefix adder has seven stages of computations. Design of parallel prefix adder can be implemented with the help of CMOS logic, but CMOS logic constructs only inverting functions so that cascading odd cells and even cells alternatively gives the result of eliminating inverters between those two cells. Two inputs of each stage will be given to XOR gate and AND gate such that it looks like a half adder circuit. In the first stage we calculate all 64-bit inputs with the help of half adders such that the results will be propagated towards next stages. Subsequent stages follows the same procedure so as the number of half adders will be reduced stage by stage. Each half adder consist one AND gate, which requires five transistors and XOR gate requires thirteen transistors so that each half adder requires 65 transistors. The topology of the design must be simple to reduce the logic depth, because each input again requires one buffer for impedance matching. So we are going to use cells like black, grey and white cells for the reduction of number of transistors. White cell is a half adder and black cell consists one OR gate and two AND gates and grey cell consists of one OR gate and one AND gate. By all means we reduced transistor count, which also leads to low power and low area specifications.In the proposed 64-bit parallel prefix adder logic cells has been preferred, depending upon the stage we use number of logic cells. The final stage of the design gives the sum signal as one output and carry signal as other output. The stages at both ends can be operated with high speed because of simple topology in the network. Amid, all of this the performance depends on the intermediate stages. We also compared Brent Kung and Ladner Fischer adders for n-bit at various values of n, like propagation delay andnetwork complexity. The proposed adder is as shown in Fig(3), which can overcome various problems in existed adders like shown in Fig(2).

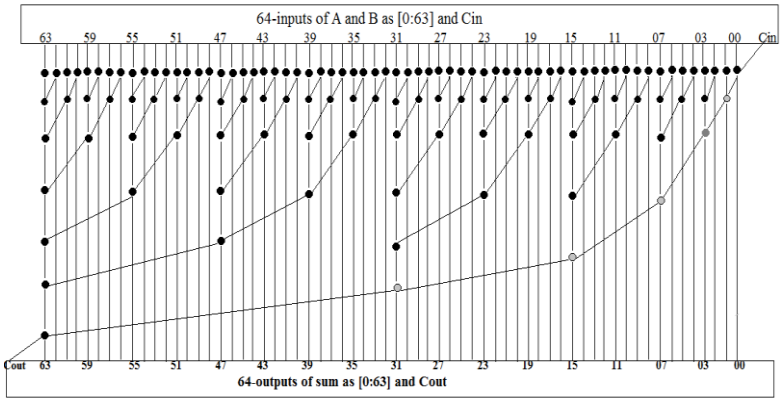


Fig3: proposed 64-bit parallel prefix adder

IV. CELLS LOGIC DIAGRAMS

A. BLACK CELL

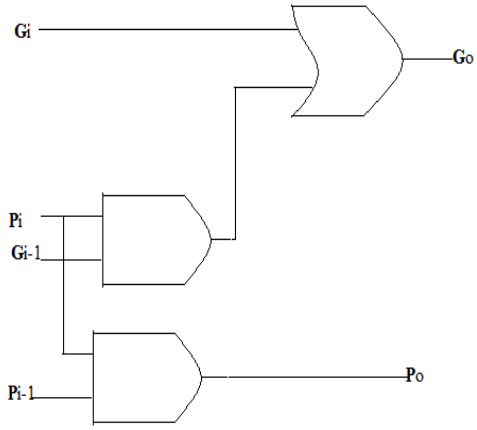


Fig4: Black Cell Logic Diagram

The Boolean equations (1) for the above logic diagram of black cell are

(1) Go = Gi +Pi.Pi-1; Po = Pi.Pi-1

B.GREY CELL

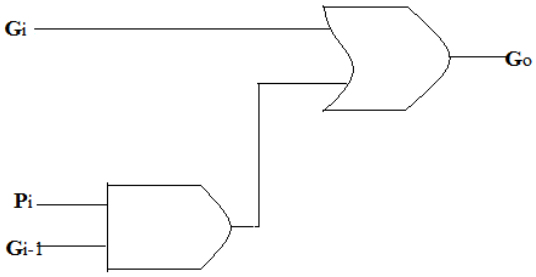


Fig5: Logic Diagram of Grey Cell

The Boolean equation (2) of the above logic diagram of grey cell is

(2) Go = Gi+Pi.Pi-1

The two cells shown in the figures (4) and (5) are for the purpose of reducing number of transistors such that the logic will become simple and the number of transistors be also reduced. Black cell gives two outputs one for normal sum and the other is for propagation of carry.

C. WHITE CELL

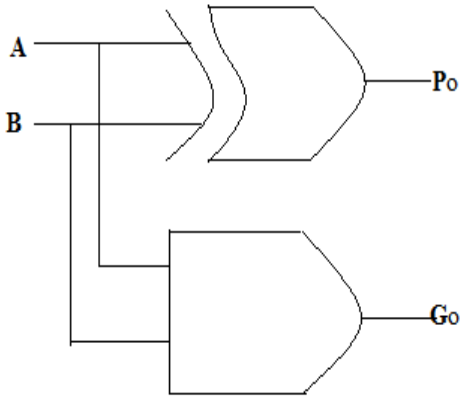


Fig6: Logic Diagram of White Cell

The Boolean equation (3) of the above logic diagram of white cell is

(3) Po = A ⊕ B = AB □+A □B Go = AB

The simulation results shown in Fig (7) gives complete idea about the design of proposed 64-bit parallel prefix adder and also gives the values like propagation delay and logic depth. The values shown in the table are obtained from the code and simulation results. We also compared those results with the existed adders like Brent Kung, Kogg Stone and etc. so that the comparison describes the efficiency of the proposed 64-bit parallel prefix adder. Chart of the simulated results is a pictorial representation of the design's comparison is also shown. Transistor count also reduced such that the logic depth and propagation delay reduced as followed in the charts.

V. SIMULATION RESULTS

A. Proposed 64-bit parallel prefix adder

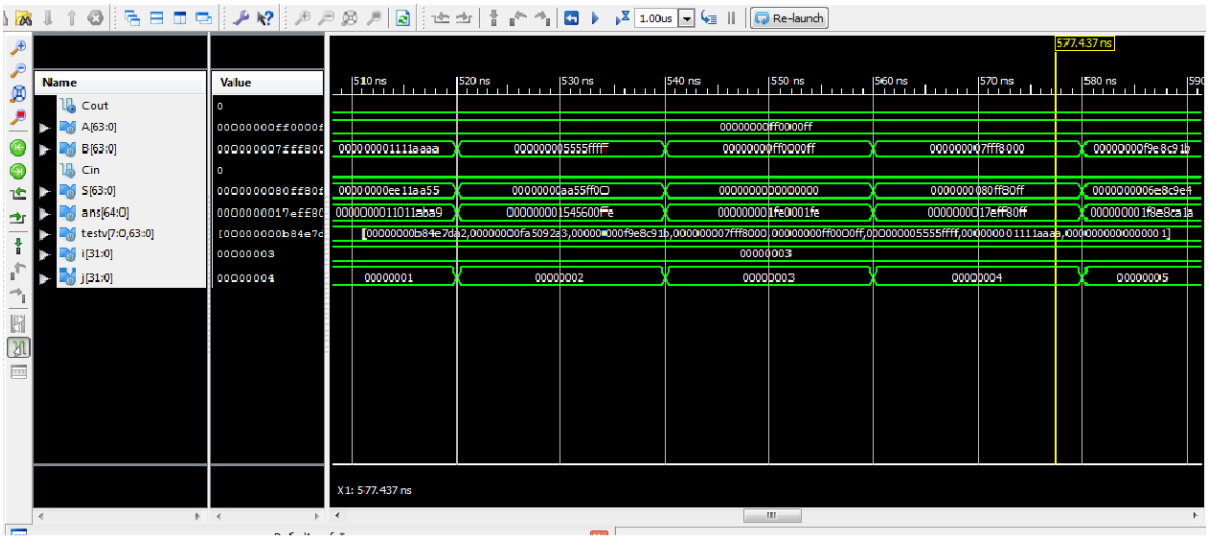
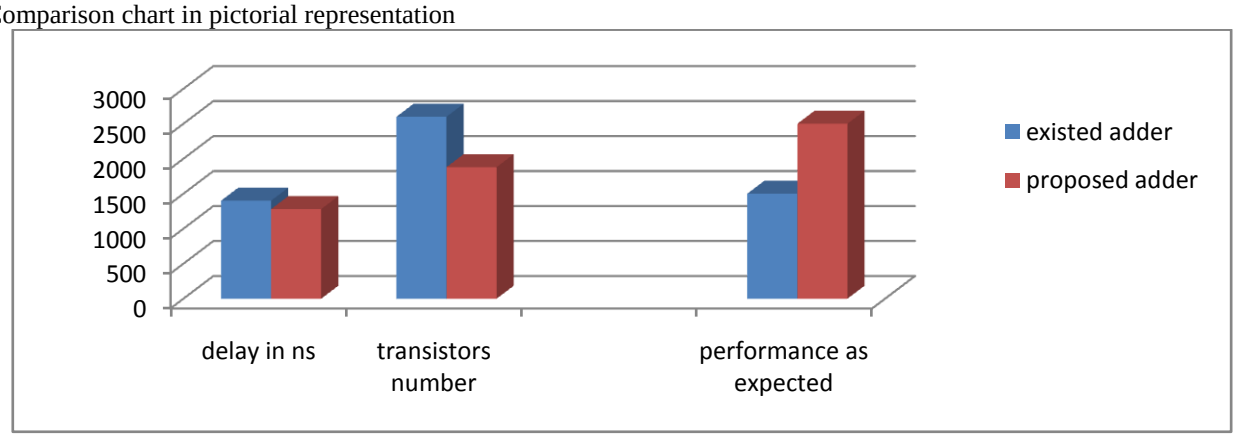


Fig7: simulation result of proposed 64-bit parallel prefix adder

B. Comparison table with output results

Parameter	Existed adders	Proposed adder	Comparison
Delay in ns	**>1280(B.K)**	**1280**	**Reduced**
Logic depth	**High(B.K)**	**low**	**Reduced**
Transistor count	**>2500**	**1880**	**Reduced**

C. Comparison chart in pictorial representation



VI. CONCLUSION

The parallel prefix adders design is an efficient method of all conventional adders. Proposed 64-bit high speed parallel prefix adder has achieved the low propagation delay by overcoming the problems in previous adders like 16-bit, 32-bit Brent Kung and Ladner and Fischer designs. The performance has been compared at various input ranges by keeping trade-off between propagation time and logic depth, and also achieved high fan-out. The proposed 64-bit high speed parallel prefix VLSI adder made the computation in seven numbers of stages and also generated sixty three outputs with the help of 845 nodes. We have used Xilinx 14.1 vivado version tool for the purpose of simulating the Verilog code. The simulation result of proposed 64-bit parallel prefix adder has proved the efficiency of the design. This design is a conventional one for the purpose of arithmetic and logic calculations at wide and complex range of inputs. The proposed design is also preferable for multipliers and for various data computations.

REFERENCES

[1] S.Knowles' "A family of adders"' Proceedings of the 15th IEEE Symposium on Computer Arithmetic. Vail' Colarado' pp.277-281' June 2001.
[2] David Harris' "A Taxonomy of parallel prefix networks"' Proceedings of the 37th Asilomar Conference on Signals' Systems and Computers Pacific Grove' California' pp.2213-2217' November 2003.
[3] Andrew Beaumont-Smith' and Cheng-Chew Lim' "Parallel Prefix Adder Design"' Department of Electrical and Electronic Engineering' the University of Adelaide' 2001.
[4] P.Ramanathan' P.T.Vanathi' "Novel Power Delay Optimized 32-bit Parallel Prefix Adder for High Speed Computing"' International Journal of Recent Trends in Engineering' Vol 2' No. 6' November 2009.
[5] R. Zimmermann' Binary Adder Architectures for Cell-Based VLSI and their Synthesis' ETH Dissertation 12480' Swiss Federal Institute of Technology' 1997.
[6] R.P Brent and H.T Kung 'A regular layout for parallel adders' IEEE Trans., C-31(3):260-264, March 82
[7] R. Ladner and M. Fischer' "Parallel prefix computation"' Journal of ACM. La Jolla' CA' vol.27' no.4' pp. 831-838' October 1980.
[8] J. Sklansky' "Conditional sum addition logic"' IRE Transactions on Electronic computers. New York' vol. EC-9' pp. 226-231' June 1960.
[9] P.Kogge and H.Stone' "A parallel algorithm for the efficient solution of a general class of recurrence relations"' IEEE Transactions on Computers' vol. C-22' no.8' pp.786-793' August 1973.
[10] P.Ramanathan' P.T.Vanathi' "A Novel Logarithmic Prefix Adder with Minimized Power Delay Product"' Journal of Scientific & Industrial Research' Vol. 69' January 2010' pp. 17-20.
[11] T. Han and D. Carlson' "Fast area efficient VLSI adders"' Proceedings of the Eighth Symposium on Computer Arithmetic. Como' Italy' pp.49-56' September 1987. (Pubitemid 17613979) .