© 2018 IJSRST | Volume 4 | Issue 5 | Print ISSN: 2395-6011 | Online ISSN: 2395-602X
Themed Section: Science and Technology

Comparative Analysis of Adders Parallel-Prefix Adder for Their Area, Delay and Power Consumption

Dr. V. Sidharthan*[1], M. Prasannakumar[2]

*[1]Assistant Professor, Department of Electronics, Sri Ramakrishna College of Arts and Science (Autonomous), Nava India, Coimbatore, Tamil Nadu, India

[2]Assistant Professor, Department of Electronics, Sri Ramakrishna College of Arts and Science (Autonomous), Nava India, Coimbatore, Tamil Nadu, India

ABSTRACT

Parallel Prefix adders have been one of the most notable among more than a few designs proposed in the past. Parallel Prefix adders (PPA) are family of adders derived from the generally known carry look ahead adders. The need for a PPA is that it is mostly fast when compared with Ripple Carry Adders (RCA). The classical parallel prefix adder structures presented in the literature over the years optimize for logic depth, area, and fan-out and interconnect count of logic circuits. In this paper, a comparison of four 8-bit parallel-Prefix adders (Ladner-Fischer Adder (LFA), Kogge-Stone Adder (KSA), Bent-Kang Adder (BKA) and Han-Carlson Adder (HCA)) in their area, delay, power is proposed. In this proposed system Ladner-Fischer adder, Kogge-Stone adder, Bent-Kang adder and Han-Carlson adder, the Parallel Prefix adder are used for comparison. The results reveal that proposed Han-Carlson adder Parallel-Prefix Adder is more competent than other three types of Parallel-Prefix adder in terms of area, delay & power. Simulation results are compared and verified using Xilinx 8.1i software.

Keywords: Parallel-Prefix Adder, Area, Power, Delay.

I. INTRODUCTION

The arithmetic operations of binary numbers are one of the most stimulating problems in modern digital VLSI systems consuming a major design effort of digital signal processors and general purpose microprocessors. The design of high-speed, low-power and area efficient binary adders always receives a great deal of attention. Among the hundreds adder architectures known in the literature, when high performances are mandatory, parallel prefix trees are generally preferable [1].

Parallel Prefix Adders have been recognized as the most efficient circuits for binary addition in digital systems. Their regular structure and fast performance makes them mostly attractive for VLSI implementation. The delay of a parallel prefix adder is directly relative to the number of levels in the carry propagation stage. The need for a Parallel Prefix adder is that it is primarily fast when compared with ripple carry adders. Parallel Prefix adders have been recognized as the most efficient circuits for binary addition [2]. These adders are best suited for adders with wider word lengths.

This paper investigates the performance of two different parallel prefix adders. The parallel prefix adders investigated in this paper are: Ladner-Fischer adder (LFA), Kogge-Stone Adder (KSA), Bent-Kang Adder (BKA) and Han-Carlson adder (HCA). The performance metrics considered for the analysis of the adders are: area, delay and power. Using simulation studies area, delay and power performance of the various adder modules were obtained. It was observed that Han-Carlson adder has better circuit characteristics in terms of delay compared to adders realized using other algorithms [3].

The rest of the paper is organized as follows: In Section 2, 3, 4, 5 and 6 a brief description of the four different parallel prefix adders are given, in Section 7, the methodology used for the research is explained. Section 8 gives results analysis and Section 9 gives conclusions.

II. PARALLEL- PREFIX ADDER

Parallel prefix adders are constructed out of fundamental carry operators denoted by ¢ as follows:

$$(G'', P'') ¢ (G', P') = (G''+G'·P'', P'·P'')$$

where P” and P’ indicate the propagations, G” and G’ indicate the generations. The fundamental carry operator is represented as Figure 1.

where P” and P’ indicate the propagations, G’’ and G’ indicate the generations. The fundamental carry operator is represented as Figure 1.

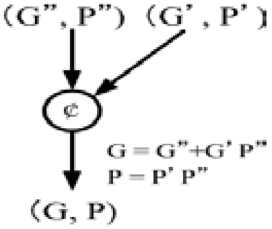

Figure 1. Carry operator

A parallel prefix adder can be represented as a parallel prefix graph consisting of carry operator nodes.

III. LADNER-FISCHER ADDER

In 1980, R. Ladner and M. Fischer described this clever modification, Ladner-Fischer adder is a parallel prefix form carry look ahead adder. A parallel prefix adder can be represented as a parallel prefix graph consisting of carry operator nodes. The time required to generate carry signals in this prefix adder is o (log n). The Ladner-Fischer adder tree diagram is shown in figure 2.

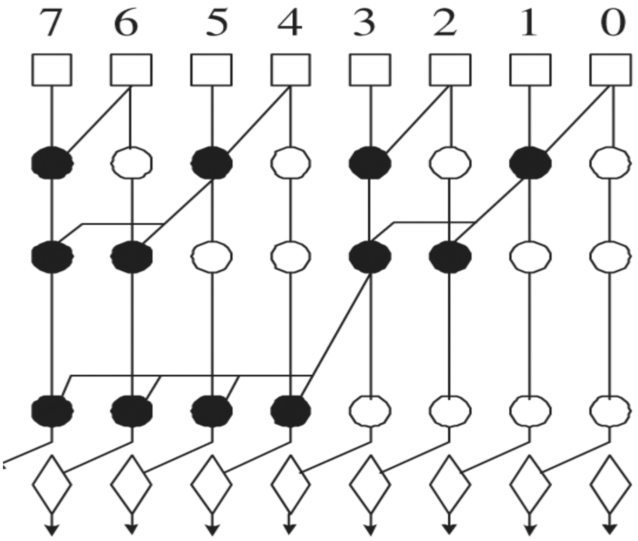

Figure 2. LFA adder tree diagram

It is a fastest adder design and mutual design for high performance adders. The large number of levels in Ladner-Fischer Adder (LFA) however reduces its operational speed. The better performances of Ladner-Fischer adder are minimum logic depth and bounded fan-out. But it has large area. LFA is also power efficient because of its lowest area delay with large number of input bits [4]. The parallel prefix adder of a LFA structure has minimum logic depth, but has large fan-out requirement up to n/2.

IV. KOGGE-STONE ADDER

In 1973, probably while listening to a Yes or King Crimson album, Kogge-Stone came up with the idea of parallel prefix computation. The Kogge Stone Adder (KSA) has regular layout which makes them favoured adder in the electronic technology. The Kogge Stone Adder tree diagram is shown in figure 3. Another reason the KSA is the favored adder is because of its minimum fan-out or minimum logic depth. As a result of that, the KSA becomes a fast adder but has a large area [5]. The delay of KSA is equal to log2n which is the number of stages for the “o” operator. The KSA has the area (number of “o” operators)

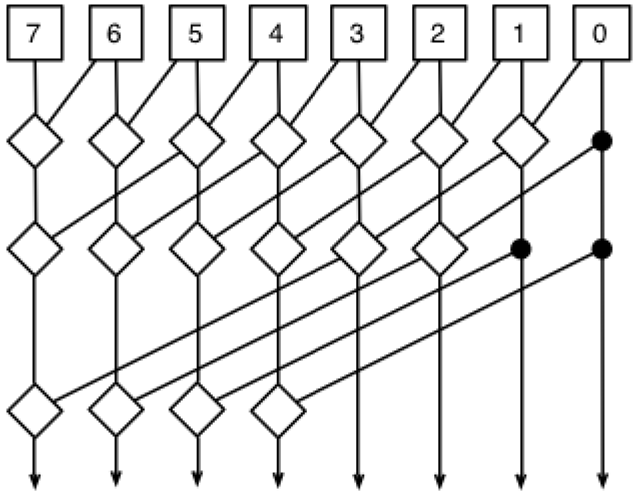

Figure 3. KSA adder Tree Diagram

of (n*log2n)-n+1 where n is the number of input bits [6].

V. Brent Kung Adder

In 1982, Brent & Kung described this clever modification, which just computes the left-most column in a binary tree, and then fills in the intermediate columns in a reverse tree. The Brent Kung Adder tree diagram is shown in figure 2.

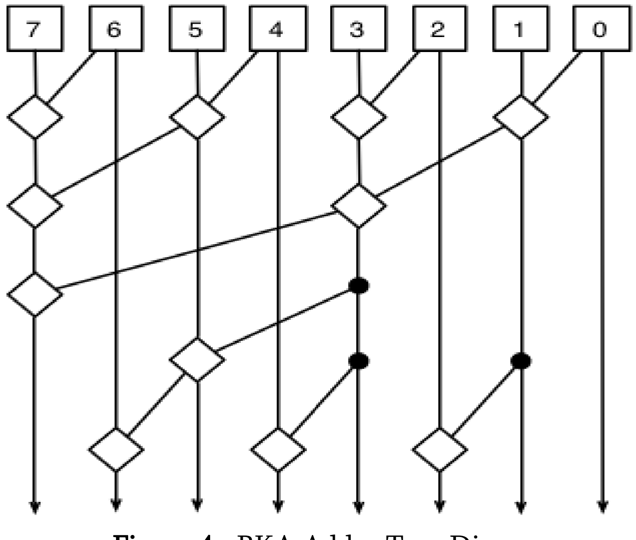

Figure 4 . BKA Adder Tree Diagram

The large number of levels in Brent Kung Adder (BKA) however reduces its operational speed. BKA is also power efficient because of its lowest area delay with large number of input bits. The delay of BKA is equal to (2*log2n)-2 which is also the number of stages for the "o" operator. The BKA has the area (number of "o" operators) of (2*n)-2-log2n where n is the number of input bits. The BKA is known for its high logic depth with minimum area characteristics [7]. High logic depth here means high fan-out characteristics.

VI. HAN-CARLSON ADDER

Han-Carlson adder contains a good trade-off between fan out, number of logic levels and number of black cells. Because of this, Han-Carlson adder can achieve equal to speed execution esteem to Kogge-Stone adder, at lower power utilization and area. In this manner it is fascinating to execute a speculative Han-Carlson adder. Moved by these reasons have generated a Han-Carlson speculative prefix-processing stage by removing the final rows of the Kogge-Stone part of the adder.

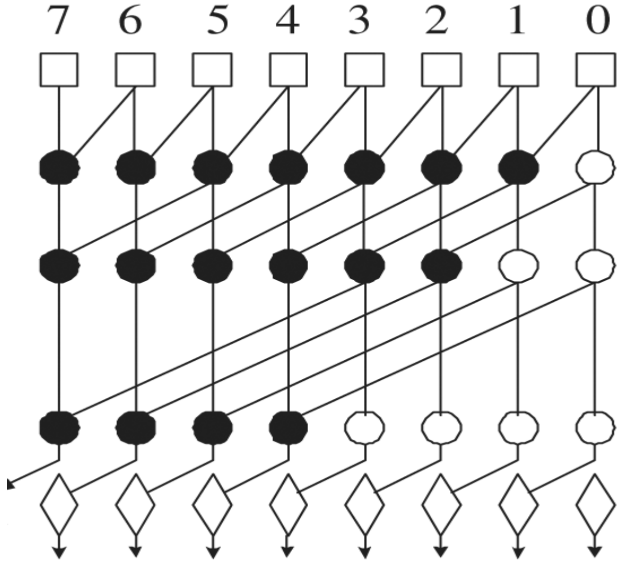

Figure 5. HCA adder tree diagram

Han-Carlson adder tree diagram is shown in figure 3 in which the two Brent-Kung rows at the initial and toward the end of the graph are unaltered, while the last Kogge-Stone row is pruned [8]. This yields a speculative stage with K=8=n/2. In general, one has K ==n/2p. Where p is the quantity of pruned levels; the number of levels of the speculative Han-Carlson stage lessens from 1+ log(n) to1+ log(k).

VII. METHODOLOGY

The designs for the adders were produced by writing Very High Speed Integrated Circuit (VHSIC) Hardware Description Language (VHDL) source file. Figure 4 shows the system design flow chart. The VHDL source code writing is the most important part in this project. There are a total of four VHDL source codes for 8 bits Ladner-Fischer Adder, Kogge-Stone Adder, Brent-Kung Adder and Han-Carlson Adder. For this paper, the VHDL source codes contain elements such as entity, library, architecture, function and array. The design file has to be examined, synthesis and compile before it can be simulated. Simulation results in this project come in the form of Register Transfer Level (RTL) diagram and functional waveform. The synthesis report obtained in the form of gate count for the design (area), delay report and power consumption report. The RTL design can be obtained by using the RTL viewer based on the Net list viewer.

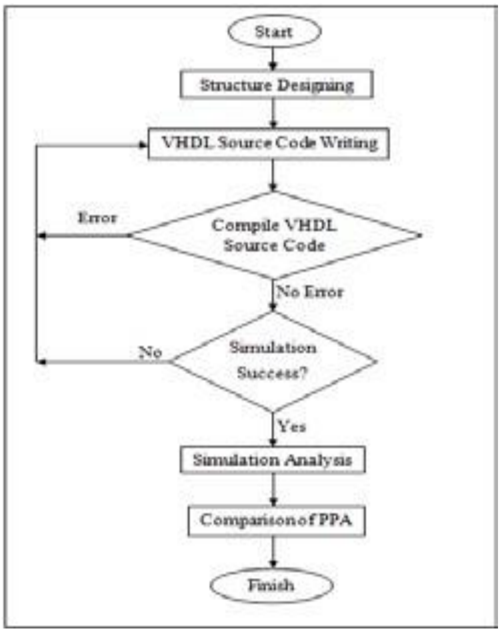

Figure 6. System design flow chart

Finally, the PPA comparison will be made once the four simulation results are analysed. LFA, KSA, BKA and HCA will be compared at this stage. The comparisons will be based on the power consumption, propagation delay and area.

VIII. RESULTS ANALYSIS

The LFA, KSA, BKA and HCA are compared in three main aspects, area, delay and power. The comparison for area, delay and power consumed by the adder circuits are based on tree diagram of each adder. The results are shown in tabulation.

Table 1. Comparison of LFA, KSA, BKA and HCA

S.No	Specifications	LFA	KSA	BFA	HCA
1.	Gate count	951	924	671	656
2.	Delay	3.579ns	3.409n	3.229n	3.129ns
3.	Power Consumption	42mW	40mW	39mW	38mW

IX. CONCLUSION

In terms of area between the four PPAs, the HCA proves to be a better. Even though the HCA's area rises as the bit size increase, it does not rise as radically as LFA. The higher the number of bits supported by the PPAs, the bigger is the adder in terms of area. In terms of computational delay also HCA is better in time propagation delay (tpd) for the bit size of 8-bits, In terms of total power estimated also the HCA consumes less amount of power. Therefore the results reveal that the proposed Han-Carlson Parallel-Prefix adder is more competent than other three adders Parallel-Prefix adder in terms of area, delay and power for the bit size of 8bits.

X. REFERENCES

[1]. Pawan Kumar, Jasbir Kaur, "Design of Modified Parallel Prefix Knowles Adder", International Journal of Science and Research (IJSR), Volume No. 3, Issue No. 7, Page No. 199-202, July 2014.

[2]. P.Chaitanya kumari, R.Nagendra, "Design of 32 bit Parallel Prefix Adders", IOSR Journal of Electronics and Communication Engineering (IOSR-JECE), Volume No. 6, Issue No. 1, Page No. 01-06, May - June 2013.

[3]. V.N.Sreeramulu, "Design of High Speed and Low Power Adder by using Prefix Tree Structure", International Journal of Science, Engineering and Technology Research (IJSETR), Volume No. 4, Issue No. 9, September 2015.

[4]. Padmajarani, S.V. M.Muralidhar, "Comparison of Parallel Prefix Adders Performance in an FPGA", International Journal of Engineering Research and Development, Volume No. 3, Issue No. 6, Page No. 62-67, September 2012.

[5]. Babulu, K., Y.Gowthami, "Implementation and Performance Evaluation of Prefix Adders using FPGAs", IOSR Journal of VLSI and Signal Processing, Volume No. 1, Issue No.1 , Page No. 51-57, September-October 2012.

[6]. Sivannarayana Gandikota, Raveendra babu Maddasani and Padmasree CH. "Design and Implementation of Carry Tree Adders using Low Power FPGAs", International Journal of Advanced Research in Computer Engineering & Technology, Page No. 295-299, September 2012.

[7]. R.P Brent & H. T. Kung, "A Regular Layout for Parallel Adders," IEEE Trans. Computers, Volume No. 31, Page No. 260-264, 1982.

[8]. M. M. Ziegler & M. R. Stan, "A Unified Design Space for Regular Parallel Prefix Adders", IEEE Journal of Design, Automation and Test in Europe Conference and Exhibition, Volume No. 2, Page No. 1386 - 1387, 2004.