Comparison of Parallel Prefix Adder (PPA)

Nurdiani Zamhari, Peter Voon, Kuryati Kipli, Kho Lee Chin, Maimun Huja Husin

***Abstract*—** The Parallel Prefix Adder (PPA) is one of the fastest types of adder that had been created and developed. Two common types of parallel prefix adder are Brent Kung and Kogge Stone adders. This research involves an investigation of the performances of these two adders in terms of computational delay and design area. The investigation and comparison for both adders was conducted for 8, 16 and 32 bits size. By using the Quartus II design software, the designs for both Brent Kung and Kogge Stone adders were developed. The simulation result produced the vector waveform which then shows the computational delay for the adders. Hence, this project is significant in showing which of the two adders being tested perform better in terms of computational delay and design area based on different sizes of bits.

***Keywords*—** parallel prefix adder, Brent Kung adder, Kogge Stone adder, computational delay comparison, area comparison

I. Introduction

Parallel Prefix Adder (PPA) is very useful in today's world of technology because of its implementation in Very Large Scale Integration (VLSI) chips. The VLSI chips rely heavily on fast and reliable arithmetic computation. These contributions can be provided by PPA. There are many types of PPA such as Brent Kung [1], Kogge Stone [2], Ladner Fisher [3], Hans Carlson [4] and Knowles [5]. For the purpose of this research, only Brent Kung and Kogge Stone adders will be investigated.

Fig. 1 shows the structured diagram of a PPA. PPA can be divided into three main parts, namely the pre-processing, carry graph and post-processing. The pre-processing part will generate the propagate (p) and generate (g) bits. The acquirement of the PPA carry bit is differentiates PPA from other type of adders. It is a parallel form of obtaining the carry bit that makes it performs addition arithmetic faster.

Nurdiani Zamhari is with the Electronic Engineering Department, Universiti Malaysia Sarawak, Sarawak, Malaysia (phone: 06-082-583323; fax:06-082-583410; e-mail: znrdiani@feng.unimas.my).

Peter Voon, was final year research student of Electronic Engineering Department, Universiti Malaysia Sarawak, Sarawak, Malaysia (e-mail:petervoon@hotmail.com).

Kuryati Kipli, Kho Lee Chin, Maimun Huja Husin are with the Electronic Engineering Department, Universiti Malaysia Sarawak, Sarawak, Malaysia (e-mail: kkuryati@feng.unimas.my, lckho@feng.unimas.my hhmaimun@feng.unimas.my).

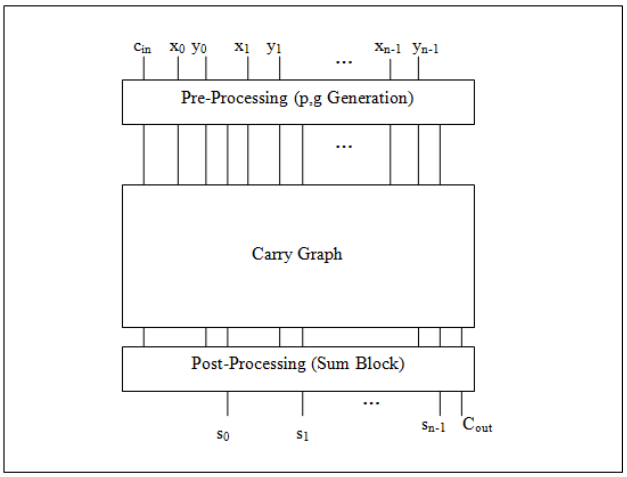

Fig. 1: PPA Structured Diagram [6]

Fig. 2 and equation (1) below show the acquirement of the carry bit at an instantaneous time, c_{in}. The symbol "o" is the associative operator and also known as the prefix operator.

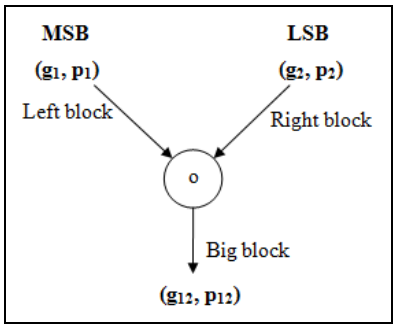

Fig. 2: Carry Operator, "o" Functions in Prefix Adder

$$(g_1, p_1)\ o\ (g_2, p_2) = (g_1 + g_2 \cdot p_1, p_1 \cdot p_2) \quad (1)$$

where g_1 and g_2 indicate generation bits, p_1 and p_2 indicate propagation bits.

Based on equation (1), there are two conditions for resulting generate-bit (g) of the big block to be created, which are;

i. Left block (Most Significant Bit, MSB) has a carry generated, **OR**
ii. Right block (Less Significant Bit, LSB) generates a carry **AND** left block propagates it.

The big block will propagates a carry if both right and left blocks are propagating the carry. The different types of PPA are uniquely diverse from each other based on their carry graph and number of levels.

ISBN: 978-988-19252-1-3
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

A. Fundamental Carry Operator (FCO)

The FCO blocks contain the associative operator (*o*) which produces the generate and propagate bits in the PPA carry graph. The FCO blocks process the generate and propagate bits from the pre-processing structure of the PPA. The FCO blocks contain two AND gates and one OR gate. The arrangement of these blocks is distinguish the different types of PPA.

B. Brent Kung Adder

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 [7]. The delay of BKA is equal to $(2*\log_2 n)-2$ which is also the number of stages for the *"o"* operator. The BKA has the area (number of *"o"* operators) of $(2*n)-2-\log_2 n$ where *n* is the number of input bits [1]. The BKA is known for its high logic depth with minimum area characteristics [8]. High logic depth here means high fan-out characteristics.

C. Kogge Stone Adder

The Kogge Stone Adder (KSA) has regular layout which makes them favored adder in the electronic technology. 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 [9]. The delay of KSA is equal to $\log_2 n$ which is the number of stages for the "*o*" operator. The KSA has the area (number of "*o*" operators) of $(n*\log_2 n)-n+1$ where *n* is the number of input bits [2].

II. Methodology

The number of associative operator (*o*) or FCO blocks and its stages for both BKA and KSA were calculated. The calculation was done for 8, 16 and 32 bits. The results of the calculation were compared between the two adders. Design software Quartus II sp2 Web Edition was used to produce design simulation of BKA and KSA. The designs for the adders were produced by writing Very High Speed Integrated Circuit (VHSIC) Hardware Description Language (VHDL) source file. Fig. 3 shows the system design flow chart.

The VHDL source code writing is the most important part in this project. There are a total of six VHDL source codes for 8 bits, 16 bits and 32 bits for Brent Kung and Kogge Stone adders. For this research, the VHDL source codes contain elements such as entity, library, architecture, function and array.

The design file has to be analyzed, synthesis and compile before it can be simulated. Simulation results in this project come in the form of Register Transfer Level (RTL) diagram, functional vector waveform outcome and classic timing analysis. The RTL design can be obtained by using the RTL viewer based on the Netlist viewer. Functional vector waveform outcome are produced by selecting random bit values and add up to produce the sum and carry bits. Timing analysis can be obtained by viewing the summary of the classic timing analysis after compiling the whole project. The simulations are done by using the functions that is included in the Quartus II design software.

Simulation analysis is prepared by viewing the results from the simulated VHDL source code. Analysis of the simulation is performed once the desired simulation outcome is obtained. Simulation results show the classic timing analysis, RTL schematic diagram and also vector waveform outcome of the simulated designs. The analysis of the PPA is conducted by viewing the time delay produced by both BKA and KSA adders in performing bits addition.

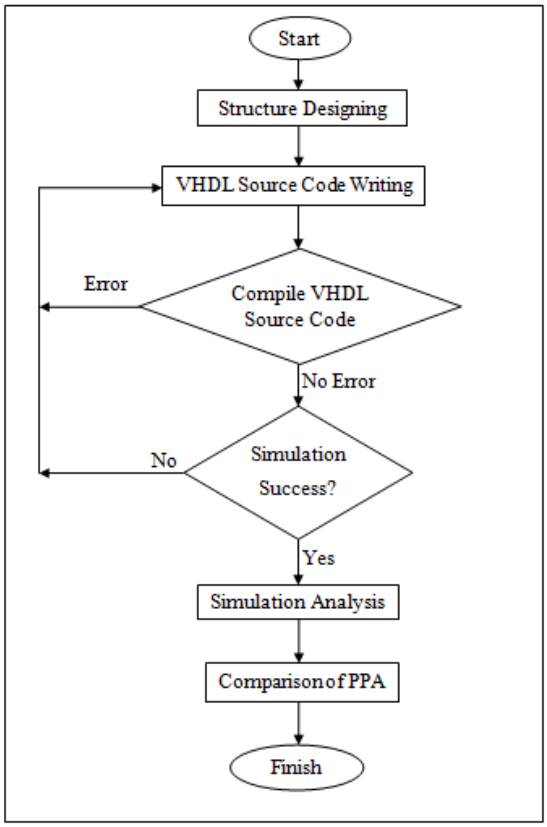

Fig. 3: System Design Flow Chart

Finally, the PPA comparison will be made once all six simulation results are analyzed. BKA and KSA will be compared at this stage and will be conducted in its bit category. The comparisons will be based on the computational speed or also known as time propagation delay and area (cost).

III. Result and Analysis

The BKA and KSA are compared in two main aspects, area and computational delay. The comparison for area is based on number of FCO blocks on each adder. On the other hand, the comparison on computational delay is based on the timing analysis of both adders.

A. Area Comparison

Fig. 4 shows the bar chart analysis of the number of FCO blocks versus bit size for both BKA and KSA. The number of FCO blocks for both BKA and KSA type adders increase proportionally with the bit size. However, the increment for

ISBN: 978-988-19252-1-3
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

number of FCO blocks in BKA is not significant compared to KSA. The number of FCO blocks in KSA increases drastically in proportion with the bit size compared to BKA. At 32 bits size, the KSA produces more than half FCO blocks than BKA produces.

The numbers of FCO blocks greatly affect the area of the adder. The FCO blocks consists of two AND gates and an OR gate. In the practical application, the increase in the number of FCO blocks also requires more wiring for connection in the circuit design. Hence the higher the numbers of FCO blocks exist, the higher cost involved. Cost is concerned because it is directly proportional to the area of the circuit design. As a result, the KSA is bigger in terms of area as the number of bits increase compared to BKA.

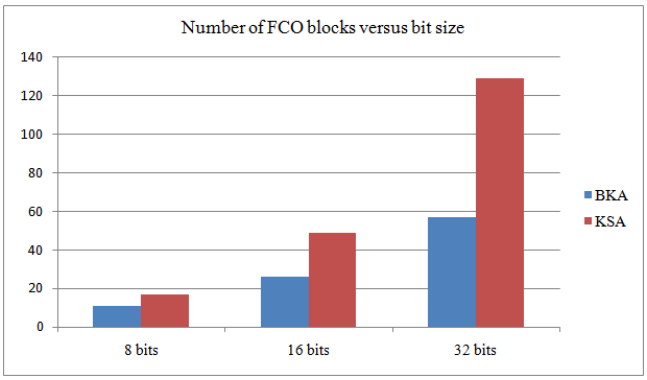

Fig. 4: Number of FCO Blocks versus Bit Size for BKA and KSA

B. Computational Delay Comparison

Fig. 5 shows the bar chart of time propagation delay (t_{pd}) in nanosecond versus bit size for both BKA and KSA. The t_{pd} for BKA is slightly lower than the t_{pd} for KSA at bit size of 8 bits. At 16 bits, both BKA and KSA appeared to have almost the same t_{pd}. However, at bit size of 32 bits, the time of KSA shows shorter t_{pd} than BKA. This is because the BKA's number of "*o*" stages starts to grow faster compared to KSA. From 8 bits to 16 bits, the number of "*o*" stages for BKA increased by 2. For KSA on the other hand, the number of "*o*" stages only increased by 1.

Adder with higher number of "*o*" stages will have longer propagation time. This is because each stage's logical computation depends on its previous stage(s) logical computation. Thus, longer propagation time is needed for high number of "*o*" stages. The BKA is expected to have even larger t_{pd} compared to KSA as the number of bit size increases. The t_{pd} for KSA does not increase extremely like BKA as the bit size grows because its "*o*" stages increase one by one.

Another factor that affects the t_{pd} of the PPAs besides the number of "*o*" stages is the number of FCO blocks. The KSA has higher t_{pd} than BKA at 8 bit and almost similar t_{pd} at 16 bit even though the number of "*o*" stages suggests the opposite result. This is because of the KSA has more FCO blocks than BKA. Therefore, propagation time is increases due to the increases in logical computational. However, as the result suggests, at 16 bits the number of FCO blocks effect on the t_{pd} is not significant compared to the number of "*o*" operator stages.

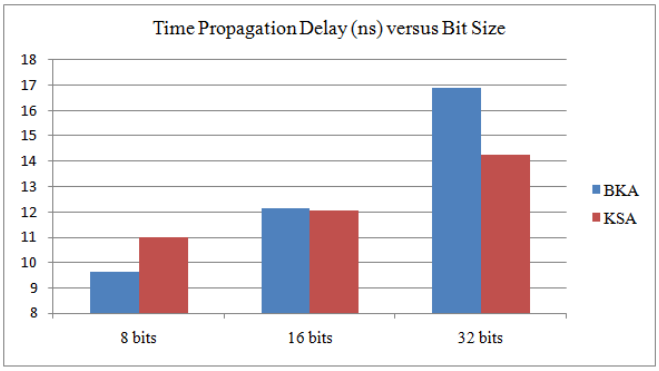

Fig. 5: Time Propagation Delay (ns) versus Bit Size for BKA and KSA

IV. Conclusion

In terms of area or cost between the two PPAs, the BKA proves to be a better choice. Even though the BKA's area rises as the bit size increase, it does not rise as drastically as KSA. The higher the number of bits supported by the PPAs, the bigger is the adder in terms of area. In terms of computational delay or time propagation delay (t_{pd}), KSA is a better choice. Although BKA has lower t_{pd} for bit size of 8 bits, the KSA has very low t_{pd} compared to BKA when the bit size is more than 16 bits. Therefore, only at bit size less than 16 bits the KSA has longer t_{pd}. The KSA is widely-known as a PPA that performs fast logical addition.

Acknowledgment

The authors would like to thank Universiti Malaysia Sarawak (UNIMAS) for providing financial support to attend the conference and present the paper.

References

[1] R. P Brent & H. T. Kung, "A Regular Layout for Parallel Adders," *IEEE Trans. Computers*, Vol C-31, pp 260-264, 1982.

[2] P. M. Kogge & H. S. Stone, "A Parallel Algorithm for the Efficient Solution of a General Class of Recurrence Equations," *IEEE Trans. Computers*, Vol. C-22, pp 786-793, 1973.

[3] R. E. Ladner & M. J. Fischer, "Parallel Prefix Computation," *JACM*, Vol. 27-4, pp 831-838, 1980.

[4] T. Hans & D. A Carlson, "Fast Area-Efficient VLSI Adders," *Proc. 8th IEEE Symposium on Computer Arithmetic*, pp 49-56, 1987.

[5] S. Knowles, "A Family of Adders," *Proc. 15th IEEE Symposium on Computer Arithmetic*, pp 277-281, 2001.

[6] Y. Choi, "Parallel Prefix Adder Design," *Proc. 17th IEEE Symposium on Computer Arithmetic*, pp 90-98, 27th June 2005.

[7] V. Ionescu, I. Bostan, & L. Ionescu, "Systematic Design for Integrated Digital Circuit Structures" *IEEE Journal of Semiconductor Conference, 2004*, Volume 2, pp 467 – 470, 2004.

[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 2, pp 1386 - 1387, 2004.

[9] I. Koren, *Computer Arithmetic Algorithms Ak Peters Series*, Second Edition, Massachusetts: A K Peters Ltd., 2002.

ISBN: 978-988-19252-1-3
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)