# Design of High Speed Power Efficient Wallace Tree Adders

Sakshi Sharma[1], Pallavi Thakur [2]
*M.Tech. Student[1] Assistant Professor[2]*
*[1]University College of Engineering, Punjabi University, Patiala, Punjab, India*
*[2]Surya World Institutions of Academic Excellence, Punjab, India*
*Email: Sakshi.sharma.1591@gmail.com, pallavi.thakur55@yahoo.com*

***Abstract*— In this paper FIFB, FIEB and FISB Carry Save Adders and Wallace Tree Adders are designed, encoded in Verilog and simulated using Cadence Software. The 180 nm CMOS technology is used for implementation of adders.The simulation results are compared for power consumption, delay, silicon area and dynamic power dissipation. As the length of inputs increase, power dissipated, silicon area and delay increase in both Carry Save Adder and Wallace Tree Adder. Compared to traditional CSA, the proposed Wallace Tree Adder is found to have shorter delay, lesser power dissipation and lesser silicon area and hence more cost efficient and a better option for real-time applications.**

*Keywords:* **Carry save adders, Power consumption, Verilog, Wallace tree based adders, delay, Silicon area, Dynamic power dissipation**

## I. INTRODUCTION

The necessity and popularity of portable electronics is driving designers to endeavor for smaller area, higher speeds, longer battery life and more reliability. Power and delay are the premium resources a designer tries to save when designing a system. The most fundamental units in various circuits such as compressors, comparators and parity checkers are full adders. Other potential applications of these adders are ALU, digital signal processing, counters, graphic processors, calculate addresses and code compressor. Enhancing the performance of the full adders can significantly affect the overall system performance. The data path consumes roughly 30% of the total power of the system. Adders are an extensively used component in data path and therefore careful design and analysis of adders is required.

So far several logic styles have been used to design full adders. Each design has its own pros and cons. One example of such design is the standard static CMOS full adder. The main drawback of static CMOS circuits is the existence of the PMOS block, because of its low mobility compared to the NMOS devices. Therefore, PMOS devices need to be seized up to attain the desired performance. However, the dynamic CMOS logic provides a higher speed of operation. In this paper an analysis of the area, power consumption and delay time of carry save adder and Wallace tree based adders implemented for 180nm dynamic CMOS technology is done and compared using Cadence Software for simulation of the Verilog code. Wallace tree adders are found to have significant complexity and timing advantages over traditional carry-save adders. The main disadvantage to Wallace tree adder is its irregular structure, making layout difficult. Furthermore, all adder blocks are active regardless of multiplicand size, making this architecture wasteful. The Wallace Tree Adder is more cost effective and has a shorter delay across the critical path because the number of adders needed on the critical path is reduced compared to Carry Save Adder. Since the design uses less adders than conventional carry save adder design, it is not only faster, but takes up less area and therefore cost less to manufacture. Another benefit of using less adders is that less transistors are needed, and therefore less power is consumed by the device. There is one disadvantage to a Wallace Tree multiplier however, and that is that it is difficult to manufacture efficiently due to its irregular layout. Yet, the benefits associated with using a Wallace Tree Multiplier far outweigh the small increase in production difficulty.

Section II discusses the research methodology used for the work presented in the paper and investigates the problems with conventional techniques used for implementing full adders. Section III outlines the implementation and working of four input four bit (FIFB), Four Input Eight Bit (FIEB) and Four Input Sixteen Bit (FISB) carry save adders and proposed Wallace tree adders. It also analyses the power, area and performance issues (delay) of the proposed adder compared to CSA adder. Finally, section IV outlines the conclusions obtained from the simulation results.

## II. RESEARCH METHODOLOGY

Adder is the core element of complex arithmetic circuits like addition, multiplication, division, exponentiation, etc. It's a speed limiting element. There are standard implementations with various logic styles that have been used in the past to design full-adder cells. Although, they all have similar function, the way of producing the intermediate nodes and the transistor count is varied. Optimization is required and it can come in picture either at logic level or at circuit level. Logic level optimization can be done by rearranging the Boolean equations. After logic level optimization we can get faster and smaller circuit. The architecture used for implementation also influences the speed, size, power dissipation and the wiring

 

complexity of a circuit. The circuit delay is determined by the number of inversion levels, the number of transistors in series, transistor sizes (i.e. channel widths) and the intra-cell wiring capacitances. Circuit size depends upon the number of transistors, their sizes and on the wiring complexity. Power dissipation depends upon the switching activity, node capacitances (made up of gate, diffusion, and wire capacitances) and control circuit size. There are some addition techniques which are commonly used in many addition operations such as Ripple carry, Carry look ahead and Carry save adders in the increasing order of speed performance.

The idea of Carry save adder is to take 3 numbers that we want to add together, x + y + z, and convert it into 2 numbers c + s such that x + y + z = c + s. In carry save addition, we refrain from directly passing on the carry information until the very last step. Computation of these three numbers has divided into two steps, In first steps CSA is used to compute S and C, then CPA is used to compute the total sum. The delay can thus be reduced by using carry save adder.

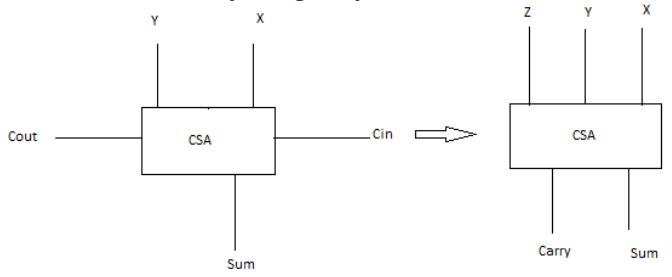


Figure 1: Carry Save Adder

## III. PROPOSED WORK

In this section, we design and implement four input four bit, four input eight bit and four input sixteen bit carry save adders and proposed Wallace tree adders architectures using Verilog language for 180nm CMOS technology. The Verilog codes are simulated using Cadence Software and intended for the ASIC design. Simulation of adders is generally used for verification and optimization.

### *A. Carry Save Adder:*

#### *1) Architecture of FIFB Adder*

In FIFB Adder four inputs A, B, C and D each of four bits are added. The process of addition is divided into three steps. In first step A and B are added as shown in figure 2. Next C and D are added as shown in figure 3. Finally, the results of first and second steps are added to get the final output which is shown in figure 4.

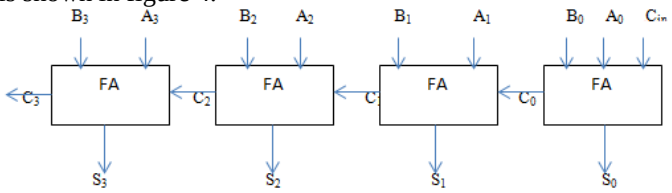


Figure 2: A and B addition Block

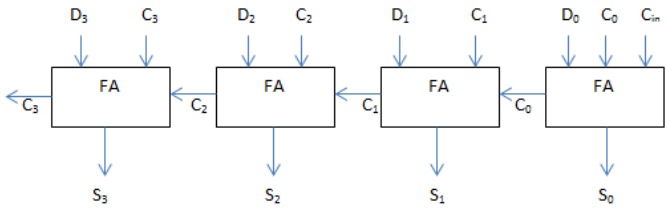


Figure 3: C and D addition Block

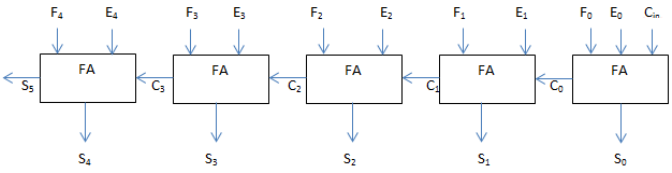


Figure 4: Addition block to add results of addition of A, B and C, D

#### *2) Architecture of FIEB Adder*

In FIEB Adder four inputs A, B, C and D each of eight bits are added in figure 5. The addition process is divided into three steps. In first step A and B are added, next C and D are added, and the results of first and second are added to get the output.

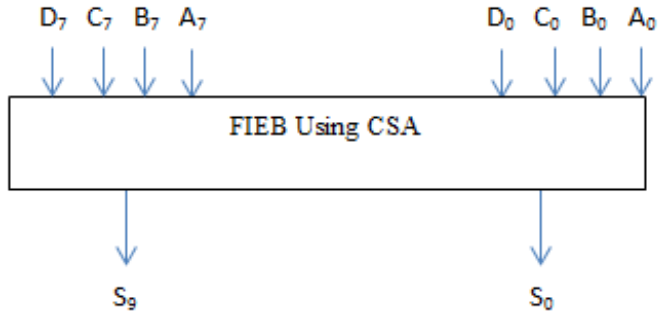


Figure 5: FIEB Adder Using CSA

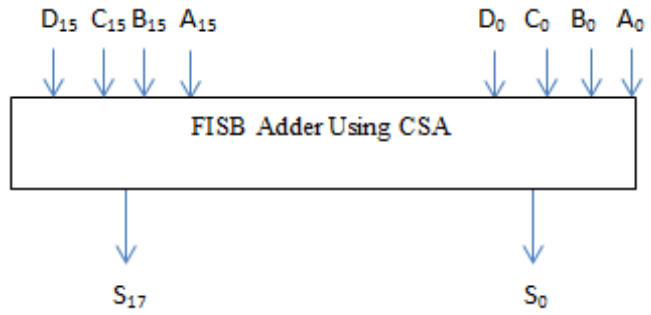


Figure 6: FISB Adder Using CSA

#### *3) Architecture of FISB Adder*

In FISB Adder four sixteen bit inputs A, B, C and D in figure 6. In first steps A and B are added next C and D are added, and finally the results of first and second step are added using CPA (carry propagation adder) to get the final outcome.

 

### *B. Wallace Tree Adders*

The designed and proposed four input four bit, four input eight bit and four input sixteen bit Wallace Tree Adder architectures are depicted in figures 8, figure 9, and figure 10 respectively. In this architecture, parallel processing is done for addition of four inputs. CSA and CPA are also used in the implementation of Wallace tree based adder. A series of half adder and full adders connections are used to design the CPA and CSA. The design is coded in Verilog and simulated using simulated cadence software and are designed considering for ASIC implementation.

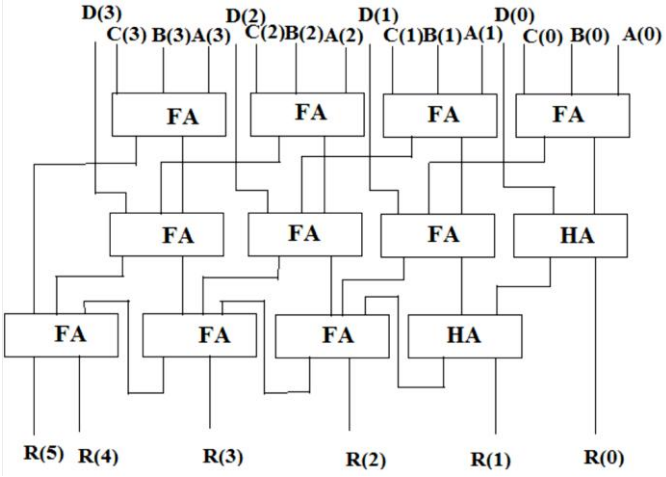


Figure 8: Four input four bit Wallace Tree Adder

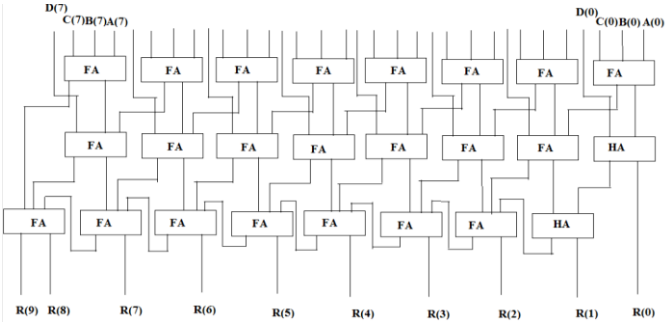


Figure 10: Four input sixteen bit Wallace Tree Adder

## IV. SIMULATION RESULTS AND DISCUSSION

The ASIC implemented Verilog codes for the designed architecture are simulated using Cadence Software and the simulation results obtained are analysed in this section.

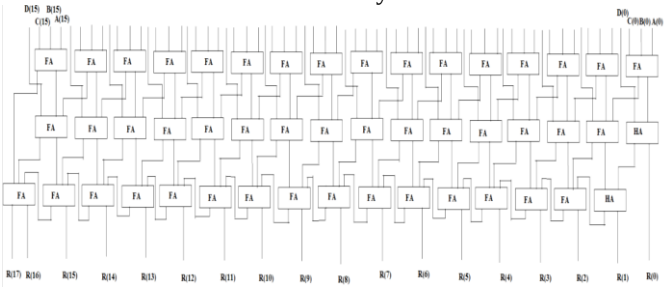

Figure 9: Four input eight bit Wallace Tree Adder

### *A. Carry Save Adders*

*Analysis:* The power dissipation in the three Carry Save Adders is depicted in Table 1. As length of inputs increases, the power dissipated also increases. For the case of delay in the three Carry Save Adders, it's also directly proportional to the length of the inputs. The silicon area and dynamic power also shows the same tendency, with the increase in length of three Carry Save Adders they also get increased. The result also suggests that the silicon area, delay, power dissipation and dynamic power are getting nearing to double as the number of bits increasing from 4 bits to 8bits, and 8 bits to 16 bits.

Table 1: Result Analysis Carry Save Adder

| **Results Analysis CSA** | | | | |
|---|---|---|---|---|
| **Adders** | **Area(μm2)** | **Delay(ps)** | **Power(μw)** | **Dynamic Power(μw)** |
| **FIFB** | **866** | **1025** | **171.54** | **170.61** |
| **FIEB** | **1645** | **2066** | **323.15** | **322.46** |
| **FISB** | **3311** | **2987** | **691.11** | **692.33** |

### *B. Wallace Tree Adders*

*Analysis:* The power dissipation in the three Wallace Tree based Adders is exhibited in Table 2. As length of inputs increases, the power dissipated also increases. For the case of delay in the three Save Adders is depicted, it's also directly proportional to the length of the inputs. The silicon area and dynamic power also shows the same tendency, with the increase in length of these three Wallace Tree based Adders they also get increased.

Table 2: Results Analysis Wallace Tree Based Adder

| **Results Analysis Wallace tree based adder** | | | | |
|---|---|---|---|---|
| **Adders** | **Area(μm2)** | **Delay(ps)** | **Power(μw)** | **Dynamic Power(μw)** |
| **FIFB** | 772 | 951 | 133.68 | 132.98 |
| **FIEB** | 1620 | 1053 | 315.23 | 315.10 |
| **FISB** | 3213 | 2290 | 689.30 | 689.20 |

## V. COMPARISON OF CARRY SAVE ADDERS AND WALLACE TREE ADDER

The relative power dissipation, silicon area , delay induced and dynamic power dissipation for the FIFB, FIEB and FISB Carry Save and Four Input, Eight Input and Sixteen Input Wallace Tree Adders is furnished in figures 8, figure 9, figure 10 and figure 11 respectively. The comparative results suggest that the power dissipation and dynamic power get reduced in higher order for lesser length of inputs Wallace Tree Adder. There is considerable cut down in power dissipation and dynamic power always in the case of Wallace Tree Adder even when number of inputs gets increased. The silicon area consumption relative results for the CSA and WTA also indicate significant bring down of area utilized for the WTA. The relative delay induced outcomes for CSA and WTA emphasizes for the noticeable high-speed nature of the Wallace Tree Adders. [3]

 

The comparison results shows that there will be the trade off between the area and power of carry save adder and Wallace tree adder. The comparison results with the other literature indicate that that the current proposed adder shows better result than other. The delays indicate shows that the time taken by the process execution. CSA shows better result than WTA at the same frequency. Different types of adder architecture have different delays depending upon their processing time.

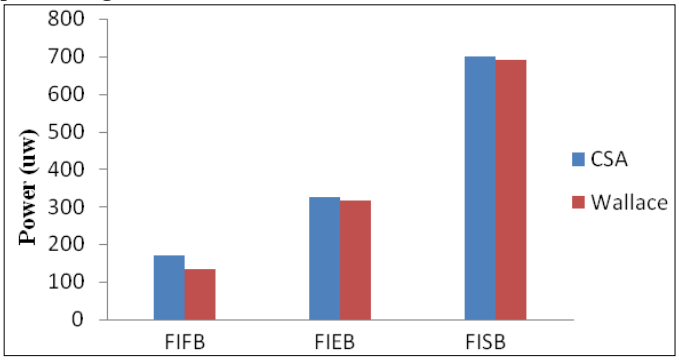


Figure 11: Power Dissipation in CSA and WTA

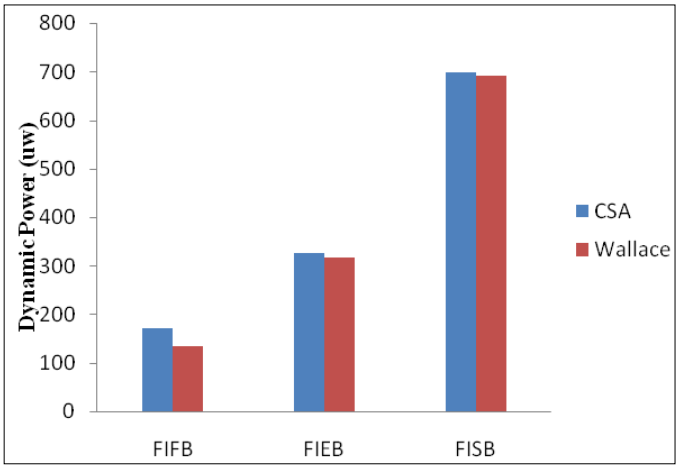


Figure 12: Dynamic Power Dissipation in CSA and WTA

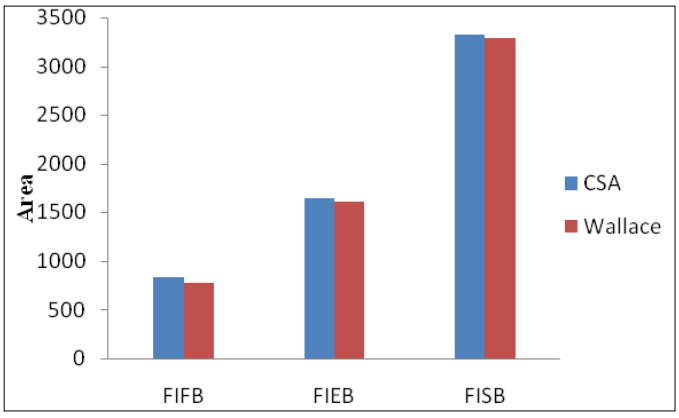


Figure 13: Silicon Area in CSA and WTA

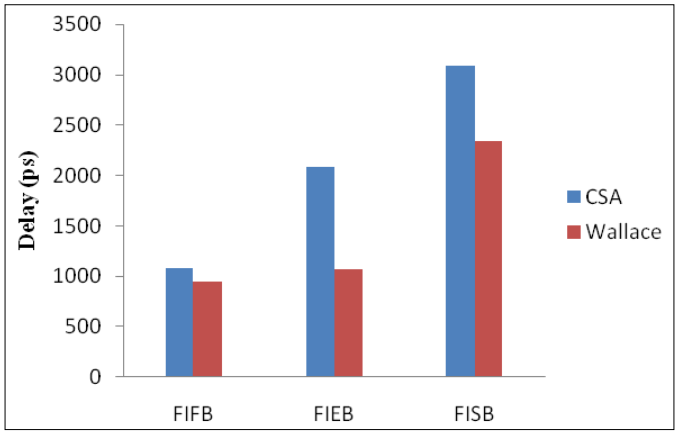


Figure 14: Delay in CSA and WTA

## CONCLUSION

Wallace tree adders are found to have significant complexity and timing advantages over traditional carry-save adders. The main disadvantage to Wallace tree adder is its irregular structure, making layout difficult. The Wallace Tree Adder is more cost effective and has a shorter delay across the critical path because the number of adders needed on the critical path is reduced compared to Carry Save Adder. Since the design uses fewer adders than conventional carry save adder design, it is not only faster, but takes up less area and therefore cost less to manufacture. Another benefit of using fewer adders is that fewer transistors are needed, and therefore less power is consumed by the device. There is one disadvantage to a Wallace Tree Adders however, and that is that it is difficult to manufacture efficiently due to its irregular layout. Yet, the benefits associated with using a Wallace Tree Adders far outweigh the small increase in production difficulty. The Wallace trees adders not only are high speed but also consume lesser power with fewer area utilization.

## REFERENCES

[1] Manuel Qrtiz, franciscoquiles,Javier Hormigo, "Efficient Implementation of Carry-Save Adders in FPGAs", International Conference on ASAP, IEEE, pp-207-210, 2009.

[2] R.P.P Singh, Praveen Kumar, Balwindar Singh, "Performance Analysis Of Fast Adders Using VHDL", International Conference on ARTCC, IEEE, pp-189-193, 2009.

[3] Shivani Parmar, Kirat Pal Singh, "Design of high speed hybrid carry select adder", Advance Computing Conference (IACC) 2013 IEEE 3rd international, pp-1653-1663, 2013.

[4] Gopal Paul, Rohit Reddy, C. R. Mandal Bhargab, B. Bhattacharya, "A BDD-based Design of an Area-Power Efficient Asynchronous Adder", International Symposium on VLSI, IEEE, pp-29-34, 2010.

[5] Mark S. K. Lau, Keck Voon Ling, Yun Chung Chu, and Arun Bhanu, "Modeling of Probabilistic Ripple-Carry Adders", International Symposium on Electronic Design, Test & Applications, IEEE, pp-201-206, 2010.

[6] Alvaro Vazquez, Elisardo Antelo, "Multi-Operand Decimal Addition by Efficient Reuse of a Binary Carry-Save Adder Tree", International Conference, IEEE, pp-1685-1689, 2010.

[7] Vladimir V. Shubin, "Analysis and Comparison of Ripple Carry Full Adders by Speed", International Conference and Seminar on EDM, IEEE, pp-132-135, 2010.

[8] Pallavi Saxena, Urvashi Purohit, Priyanka Joshi, "Analysis of Low Power, Area- Efficient and High Speed Fast Adder", International

 

Journal of Advanced Research in Computer and Communication Engineering, pp. 3705-3710, 2013.

[9] Arun Kumar P., "Reusable High Speed Architecture Design for Video Transmission in Satellite Applications", Recent Researches in Electrical Engineering, pp. 316-324, 2014.

[10] Maroju SaiKumar,Dr. P. Samundiswary, "Design and Performance Analysis of Various Adders using Verilog", International Journal of Computer Science and Mobile Computing, pp. 128-138, 2013.

[11] Neeraj Jain,Puran Gour,Brahmi Shrman, "A High Speed Low Power Adder in Multi Output Domino Logic", International Journal of Computer Applications, pp.30-33, 2014.

[12] Kirat Pal Singh, Shivani Parmar, Dilip Kumar, "Design of High Performance MIPS Cryptography Processor", Proc. 9th International Conference Heterogeneous Networking for Quality, Reliability, Security and Robustness, Springer LNICST, pp. 778-793, January 2013.

[13] Shiwani Dod, "Modified Booth Dadda Multiplier using Carry Look ahead adder Design and Implementation", International Journal of Computer Science & Engineering Technology, pp. 86-93, 2016.

[14] Shruti Murgai, Ashutosh Gupta, Gayathri Muthukrishnan, "Energy Efficient and High Performance 64-bit Arithmetic Logic Unit using 28nm Technology", International Conference on Advances in computing, Communications and Informatics,IEEE, pp. 453-456, 2015.

[15] Atef Ibrahim, Fayez Gebali, "Optimized structures of hybrid ripple carry and hierarchical carry lookahead adders", Microelectronics Journal, Elsevier, pp. 783-794, 2015.

[16] Kirat Pal Singh, Shivani Parmar, "VHDL Implementation of a MIPS-32 bit Pipeline Processor", International Journal of Applied Engineering Research, pp. 1952-1956, 2012.

[17] Priya Meshram, Mithilesh Mahendra, Parag Jawarkar, "Designed Implementation of Modified Area Efficient Enhanced Square Root Carry Select Adder", International Journal for Research in Emerging Science and Technology, pp. 96-99, 2015.

[18] Pallavi Saxena, "Design of Low Power and High Speed Carry Select Adder Using Brent Kung Adder", International Conference on VLSI Systems, Architecture, Technology and Applications,IEEE, pp. 1-6, 2015.

 