International Journal of Scientific Research & Engineering Trends
Volume 9, Issue 4, Jul-Aug-2023, ISSN (Online): 2395-566X

Design and Implementation of 32bit Kogge Stone Adder

Banashankari, Eleena Mohapatra
Department of ECE
R.V. College of Engineering

Abstract- The Kogge Stone Adder is a high-performance parallel prefix adder designed to enhance the speed of binary addition operations in digital systems. By mitigating the propagation delay associated with carry signals, it significantly accelerates the computation of binary sums. The adder employs a hierarchical structure, breaking down input numbers into groups and calculating carry signals concurrently{1]. This parallelism enables faster addition, making the Kogge Stone Adder particularly advantageous for wide bit-width arithmetic. Although its gate count is higher compared to traditional ripple carry adders, the drastic reduction in propagation delay compensates for this increase. Consequently, the Kogge Stone Adder proves to be a pivotal advancement in achieving rapid and efficient binary addition, vital for various applications demanding computational speed[2,3].

Index Terms- NCLaunch, RCA,CSA,KSA,CLaA.

I. INTRODUCTION

In the realm of digital circuit design, efficient addition of binary numbers lies at the core of many computational tasks. The Kogge Stone Adder, named after its creators Peter Kogge and Harold Stone, stands as a significant innovation that addresses[4,5,6]. the limitations of conventional adder architectures. This introduction provides an overview of the Kogge Stone Adder, its motivation, and its essential characteristics. Motivation: Binary addition is fundamental in various digital applications, such as arithmetic calculations, data processing, and cryptography.[8] Traditional ripple carry adders perform additions sequentially, leading to significant carry propagation delay. As a result, their efficiency diminishes as bit widths increase.

This delay can severely limit the overall speed of calculations in high-performance systems[7]. Parallel Prefix Approach: The Kogge Stone Adder introduces a parallel prefix approach to address the carry propagation delay problem. It operates by breaking down the binary addition process into stages, where carry signals are computed simultaneously for different bits. This parallelism reduces the critical path delay and results in significantly faster addition[8,9], especially for large bit widths.

Hierarchical Structure: At the heart of the Kogge Stone Adder lies its hierarchical structure. The input bits are organized into groups, and carry signals are calculated hierarchically across these groups. The process starts with individual bits and moves upwards, culminating in the final carry signal for the entire adder. This approach minimizes the need for sequential carry propagation, drastically improving the efficiency of binary addition[10]. Performance Benefits: The primary advantage of the Kogge Stone Adder is its remarkable reduction in carry propagation delay. While its gate count is higher compared to simpler adder architectures, the significant speedup achieved justifies this trade-off, especially in applications where computational speed is critical.[11] The Kogge Stone Adder finds its application in various domains, including high-performance computing, digital signal processing, and real-time systems. Its ability to handle large bit widths and deliver faster results makes it an essential component for tasks that demand efficient binary addition. the Kogge Stone Adder represents a pivotal advancement in digital circuit design. [12]By employing a parallel prefix approach and hierarchical structure, it overcomes the limitations of traditional adders, enabling faster and more efficient binary addition. As technology continues to evolve, this innovative architecture remains integral to achieving high-speed computations in diverse digital applications[13].

II.METHODOLOGY

The methodology of the 32-bit Kogge Stone Adder centers around the parallel prefix approach. This approach leverages the hierarchical structure to calculate carry signals simultaneously for multiple bits, drastically reducing the propagation delay inherent in sequential carry computations. The input numbers are divided into groups, and the carry signals are propagated in a tree-like structure[14]. The hierarchical arrangement enables efficient utilization of logic gates and minimizes critical path delay, resulting in faster binary addition. **Implementation-** The implementation of the 32-bit Kogge Stone Adder involves designing the carry look-ahead (CLA) graph and the associated logic gates. The XOR, AND, and OR gates are interconnected according to the hierarchical structure, allowing the carry and sum outputs to be computed with parallelism. The CLA graph is organized in a manner that carries are propagated

upwards through the levels, culminating in the final carry output. The sum outputs are obtained through XOR gates at each stage[15,16,17].

Simulation Results - The simulation results of Verilog code for Kogge-Stone using Cadence NCLaunch and the Simulation output is as shown in Fig. 1. At 7ns clock edge arrives, we can see a=00000000 and b= 0000000F, cin=0, after the arrival of clock sum = 0000000F and cout=0. Similary at 2nd clock edge a=F0000000 and b=0000000F,cin=0, sum=F000000F and cout=0. At 3 rd clock edge a=80000000 and b=8000000F,cin=0, sum=0000000F and cout=1. At 4th

Table 1 Comparison of Kogge-Stone with different Adders.

Design	No. of cells	Area (um2)	Power (uw)	Delay (ns)	PD (x10^-17)
RCA	65	1154	34.818	9.079	316.112
CLaA	129	1204	37.880	7.169	271.561
CSA	119	1758	47.950	4.206	201.677
KSA	648	2927	76.716	2.872	220.328

clock edge a=55555555, b=AAAAAAAA and cin=1, sum=00000000 and cout=1. The design works fine as the functionality is verified

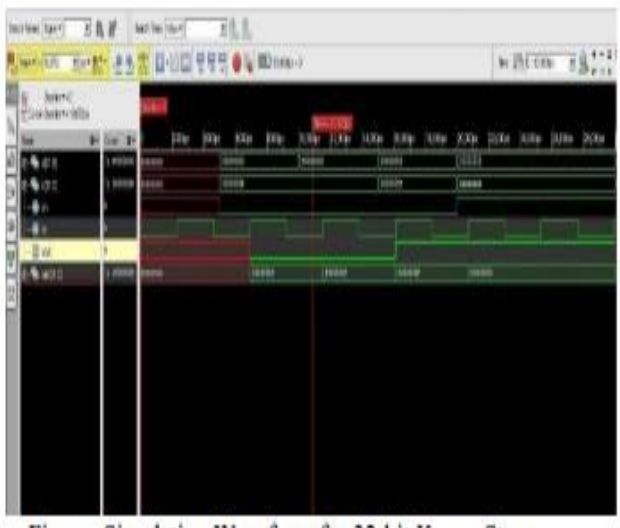

Figure: Simulation Waveform for 32-bit Kogge-Stone

Comparison of Kogge-Stone with different Adders. Here we compared Kogge-Stone with 5 different adders in terms of Power, Delay and PDP(Power Delay Product). The comparison isshown in Table 4.2. The Kogge-Stone has less delay than the other 3 adders. RCA is area efficient and consumes less power but has large delay. CSA has better PDP.

III.RESULTS

To evaluate the effectiveness of the 32-bit Kogge Stone Adder, it was simulated and compared against conventional ripple carry adders. The simulations were performed using digital circuit design software, measuring critical metrics such as propagation delay and power consumption. The results revealed a substantial reduction in propagation delay compared to the ripple carry adders, showcasing the efficiency and speed of the Kogge Stone architecture. The power consumption, though higher due to increased gate count, was justified by the significant speedup achieved.

IV.CONCLUSION

The 32-bit Kogge Stone Adder proves to be a breakthrough solution in addressing the limitations of traditional adders. Its parallel prefix approach, hierarchical structure, and efficient carry computation result in remarkable performance improvements, particularly for larger bit widths. While its gate count may be higher, the reduced propagation delay and faster addition make it a favorable choice for applications demanding rapid binary addition. As technology advances, the 32-bit Kogge Stone Adder remains a key component in achieving high-speed computations in diverse digital domains.

REFERENCES

1. S. Knowles, “A Family of Adders,” in Proc. of the IEEE Symp. On Computer Arithmetic. 2021
2. G. V. Resta, A. Leonhardt, Y. Balaji, S. De Gendt, P. Gaillardon, and G. De Micheli, “Devices and Circuits Using Novel 2-D Materials: A Perspective for Future VLSI Systems,” IEEE Trans. on VLSI Systems, vol. 27, no. 7, pp. 1486–1503, 2019.
3. G. Gore, P. Cadareanu, E. Giacomin, and P. Gaillardon, “A Predictive Process Design Kit for Three-Independent-Gate Field-Effect Transistors,” in 2019 IFIP/IEEE 27th International Conference on Very Large Scale Integration (VLSI-SoC), 2019, pp. 172–177
4. A. Antidormi, S. Frache, M. Graziano, P. Gaillardon, G. Piccinini, and G. De Micheli, “Computationally Efficient Multiple-Independent-Gate Device Model,” IEEE Transactions on Nanotechnology, vol. 15, no. 1, pp. 2–14, 2016. [5] R. P. J. Boyar, P. Matthews, “Logic Minimization Techniques with Applications to Cryptology,” J. Cryptology, Springer, pp. 280–312, 2013.
5. E. Testa, M. Soeken, L. Amaru, and G. De Micheli, “Reducing the ` Multiplicative Complexity in Logic Networks for Cryptography and Security Applications,” in Proceedings of the 56th Annual Design Automation Conference 2019. Association for Computing Machinery, 2019.

6. E. Testa, M. Soeken, H. Riener, L. Amaru, and G. D. Micheli, “A Logic Synthesis Toolbox for Reducing the Multiplicative Complexity in Logic Networks,” in 2020 Design, Automation Test in Europe Conference Exhibition (DATE), 2020, pp. 568–573.
7. Giorgos Dimitrakopoulos, Kleanthis Papachatzopoulos, and Vassilis Paliouras, "Kogge-Stone Adders", IEEE Transactions on Emerging Topics in Computing.
8. Arunkumar P Chavan .et.al “Design and implementation of 16 tap FIR filter for DSP Applications” IEEE Second International Conference on Advances in Electronics, Computer and Communications (ICAECC). [
9. P. M. Kogge and H. S. Stone, “A Parallel Algorithm for the Efficient Solution of a General Class of Recurrence Equations,” IEEE Transaction Computers, vol. C-22, pp. 786– 792.
10. R. Zimmermann, “Binary Adder Architectures for Cell-Based VLSI and their Synthesis,” Ph.D. dissertation, ETHZ, 1998.
11. S. Roy, M. Choudhury, R. Puri, and D. Z. Pan, “Towards Optimal Performance-Area Trade-Off in Adders by Synthesis of Parallel Prefix Structures,” in Design Automation Conference (DAC), 2013, pp. 1– 8. Design No. of cells Area (um2) Power (uw) Delay (ns) PD (x10^- 17) RCA 65 1154 34.818 9.079 316.112 CLaA 129 1204 37.880 7.169 271.561 CSA 119 1758 47.950 4.206 201.677 KSA 648 2927 76.716 2.872 220.328
12. T. Lynch and E. E. Swartzlander, “A Spanning Tree Carry Lookahead Adder,” IEEE Trans. On Computers, vol. 41, no. 8, pp. 931–939, 1992. [14]S. K. Mathew, M. A. Anders, B. Bloechel, Trang Nguyen, R. K. Krishnamurthy, and S. Borkar, “A 4- GHz 300- mW 64-bit Integer Execution ALU with Dual Supply Voltages in 90-nm CMOS,” IEEE Journal of Solid-State Circuits, vol. 40, no. 1, pp. 44–51, 2005
13. H. Ling, “High-Speed Binary Adder,” IBM Journal of Research and Development, vol. 25, pp. 156– 166, May 1981. N. Burgess, “Fast Ripple-Carry Adders in Standard-Cell CMOS VLSI,” in Proc. of the IEEE Symp. on Computer Arithmetic (ARITH 2011), jul 2011, pp. 103–111.
14. J. Grad and J. E. Stine, “New Algorithms for Carry Propagation,” in Proc. of the ACM Great Lakes Symp. on VLSI, 2005, p. 396–399.
15. G. Dimitrakopoulos, P. Kolovos, P. Kalogerakis, and D. Nikolos, “Design of High-Speed Low-Power ParallelPrefix VLSI Adders,” in Power and Timing Modelling, Optimization and Simulation (PAT-MOS), 2004, pp. 248–257.
16. T. Han and D. Carlson, “Fast Area-Efficient VLSI Adders,” in Proc. of the IEEE Symp. on Computer Arithmetic, May 1987, pp. 49–56.
17. A. K. Verma and P. Ienne, “Improving XORDominated Circuits by Exploiting Dependencies between Operands,” in 2007 Asia and South Pacific Design Automation Conference, 2007, pp. 601–608.
18. S. Knowles, “A Family of Adders,” in Proc. of the IEEE Symp. on Computer Arithmetic, April 1999,pp. 30–34