Design and Simulation of Digital PWM Generator Using Verilog HDL

B. Karuna Sree[1], Madaboi Nagaraju[2], G. Praveen Patil [3], Nagula Naveen[4], Singuri Aravind Kumar [5]

[1]Assisstant Professor Dept. of ECE, CMR Technical Campus, Telangana, India
[2 3 4 5]Student, Dept. of ECE, CMR Technical Campus, Telangana, India
Email: karunasree.ece@cmrtc.ac.in, 247r5a0410@cmrtc.ac.in, 237r1a04b5@cmrtc.ac.in,
237r1a04a8@cmrtc.ac.in, 247r5a0414@cmrtc.ac.in

Abstract - In modern digital systems, achieving both high speed and low power consumption has become a major challenge. The Kogge-Stone Adder (KSA) is widely used due to its fast parallel prefix structure, but it suffers from increased power usage caused by glitch activity. These glitches mainly arise due to unequal delays along different signal paths. In this work, an improved version of the KSA is presented with a focus on reducing unnecessary switching activity. The design introduces delay balancing by inserting buffers in selected paths, along with transistor sizing adjustments based on path criticality. These modifications help in synchronizing signal transitions and minimizing glitches. The proposed design is implemented using Cadence Virtuoso with GPDK090 technology. Simulation results indicate a noticeable reduction in glitch activity and power consumption, while maintaining acceptable delay performance. This makes the design more suitable for low-power VLSI applications.

Key Words: Kogge-Stone Adder, Glitch Reduction, Parallel Prefix Adder, Low Power Design, Delay Balancing, Transistor Sizing, CMOS, PDP.

1. INTRODUCTION

Adders are fundamental components in digital electronics and are used in almost every arithmetic operation inside processors, ALUs, and signal processing units. As technology continues to scale, the need for faster and more energy-efficient circuits has increased significantly.

Among the various adder architectures, the Kogge-Stone Adder stands out due to its high-speed performance. It uses a parallel prefix approach to compute carry signals, which reduces propagation delay compared to traditional adders. Because of this, it is often used in high-performance computing systems.

However, this advantage comes with certain limitations. The complex structure of KSA leads to a large number of interconnections, which increases switching activity. One major issue that arises is the presence of glitches. These are unwanted temporary transitions caused by differences in signal arrival times across different paths.

Even though glitches last for a short duration, they contribute significantly to dynamic power consumption. Therefore, reducing glitch activity is essential for improving overall efficiency.

In this project, a modified Kogge-Stone Adder is developed by applying delay balancing and transistor optimization techniques. A buffer is introduced in selected paths to align signal timing, and transistor sizes are adjusted to achieve a better trade-off between speed and power.

2. LITERATURE SURVEY

This section reviews past research and existing work related to high-speed adders and low-power VLSI design, with a focus on Kogge-Stone Adders and glitch reduction techniques. Several studies have explored different adder architectures, including ripple carry, carry look-ahead, and parallel prefix adders, to improve speed and efficiency. Among these, the Kogge-Stone Adder is widely recognized for its fast carry computation due to its parallel structure.

[1] The Zhang et al. (2021), proposes a glitch-aware delay modeling technique for digital circuits, which focuses on analyzing transient switching behavior in high-speed VLSI systems. The study highlights that glitches significantly contribute to dynamic power consumption due to unequal propagation delays. By incorporating glitch-aware models, the authors achieved better estimation of delay and power, which helps in designing optimized circuits with reduced switching activity.

[2] The Kumar et al. (2022) presented a hybrid adder by combining different prefix structures. Their design showed better performance in terms of speed and power, although the complexity of the circuit increased.

[3] The Reddy et al. (2023) introduced a modified adder using a Binary Excess-1 Converter. This approach helped in reducing delay, but required additional hardware, which increased the area.

[4] The Sharma et al. (2023), proposes a reversible logic-based Kogge-Stone Adder for low-power applications. The use of reversible logic helps in reducing power dissipation by minimizing information loss during computation. Although the design achieves significant

power savings, it introduces complexity in circuit implementation and requires advanced design techniques.

[5] The Patel et al. (2024), proposes a fault-tolerant adder architecture aimed at improving reliability in digital systems. The design incorporates error detection and correction mechanisms, ensuring stable operation even under fault conditions. The results indicate improved reliability and reduced power consumption, but at the cost of increased hardware overhead.

[6] The Singh et al. (2024) combined a Kogge-Stone Adder with a multiplier structure to improve processing speed. However, this also increased the overall circuit complexity.

[7] The Gupta et al. (2024) proposed a hybrid carry-select adder to achieve better speed and energy efficiency. Their work showed good results but required careful optimization.

[8] The Mehta et al. (2025), proposes a high-speed hybrid adder using domino logic for improved performance. The design focuses on reducing delay and enhancing speed using dynamic logic techniques. While the approach improves performance, it may lead to increased power consumption if not properly optimized.

[9] The Verma et al. (2025), proposes an approximate Kogge-Stone Adder for low-power applications. The design reduces switching activity and power consumption by allowing controlled inaccuracies in computation. This approach is suitable for error-tolerant applications such as multimedia processing, but may not be suitable for precision-critical systems.

The Lee et al. (2026), proposes advanced adder designs using emerging technologies such as F in FET and nanoscale devices. These designs aim to improve power-delay performance and overcome the limitations of traditional CMOS technology. However, practical implementation of such technologies remains complex and costly.

3. PROPOSED SYSTEM

This project proposes a glitch-reduced design of the Kogge-Stone Adder by applying balanced delay techniques and transistor optimization. The main objective is to reduce unnecessary switching activity and improve power efficiency while maintaining acceptable speed performance. The design follows the basic structure of a conventional Kogge-Stone Adder, which consists of three stages: pre-processing, prefix computation, and post-processing. In the pre-processing stage, the input signals A and B are used to generate propagate and generate signals using the expressions.

$P = A \oplus B$ ..... (3.1)

and $G = A \cdot B$ .....(3.2)

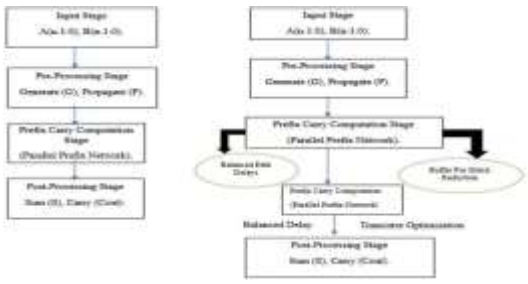


Fig 3.1: Block Diagram of the Proposed Glitch-Reduced KSA Diagram

These signals will be then passed to the prefix network for carry computation.

The prefix network is constructed using multiple stages of black cells and a final gray cell. The black cells compute both propagate and generate outputs, while the gray cell produces only the final carry output. In the conventional design, unequal delays in different paths lead to glitches, which increase power consumption. To overcome this issue, the proposed method introduces delay balancing in the prefix network. A buffer, implemented using two cascaded inverters, is inserted in the generate (G) path between intermediate stages of the black cells. This buffer helps in synchronizing signal transitions so that all inputs to a stage arrive at nearly the same time. As a result, mismatched delays are reduced and glitch activity is minimized.

Apart from this, the transistor sizes are modified based on the importance of the path., transistor sizing is optimized based on the criticality of the circuit paths. Transistors are in the critical path, such as those in black cells and the gray cell, are given larger widths to maintain very speed, while transistors in non-critical paths are assigned smaller widths to reduce power consumption. This selective sizing wiil be helps in achieving a balance between delay and energy efficiency.

The final stage of the design computes the sum output using the relation

Sum $= P \oplus$ Carry.....(3.3)

Where, the carry is obtained from the prefix network. The entire design is implemented in GPDK090 technology using Cadence Virtuoso, and its performance is evaluated through transient simulation. Overall, the proposed method effectively reduces glitch activity and improves power efficiency by combining delay balancing and transistor optimization, while maintaining correct functionality and acceptable delay performance.

4. Modelling and Analysis

Input Stage

This is the first block of the design where the input binary numbers are applied.

Inputs : A(n−1:0), B(n−1:0) .....(4.1)

These represent the bits of two numbers to be added. Each bit of A is paired with the corresponding bit of B.

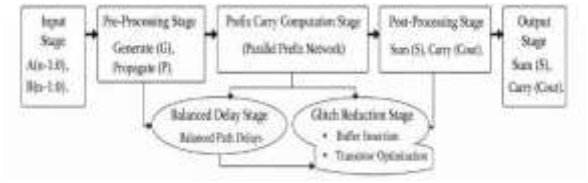


Fig. 4.1 : Block diagram first block of the design

To provide input signals for further processing in the adder.

Pre-Processing Stage

In this stage, two important signals are generated:

- Propagate (P) = A ⊕ B.....(4.2)
- Generate (G) = A · B ..... (4.3)

P (Propagate): decides whether carry will pass to next stage.

G (Generate): decides whether carry is generated at that bit.

To convert inputs into signals that help in fast carry calculation.

Prefix Carry Computation Stage

The signals enter the prefix carry computation stage, which is the core part of the Kogge-Stone Adder. This stage is implemented using a parallel prefix network consisting of multiple levels of black cells and a final gray cell. The black cells take propagate and generate inputs from previous stages and produce updated propagate and generate outputs, while the gray cell produces only the generate output, which corresponds to the final carry. The prefix structure allows carry signals to be computed simultaneously across all bit positions, significantly reducing the overall delay compared to conventional adders. This parallel processing is the key reason for the high-speed performance of the Kogge-Stone Adder.

In the Kogge-Stone Adder, the carry is generated using a parallel computation method instead of a sequential approach. Because of this, all carry values are calculated simultaneously across multiple stages, which helps in reducing the overall delay of the circuit. The structure is organized in multiple levels using prefix cells, where each stage combines the propagate (P) and generate (G) signals from the previous stage to produce faster carry outputs.

Balanced Path Delay

In the proposed design, an effort is made to balance the delay across different signal paths. In a normal Kogge-Stone Adder, signals travel through paths of different lengths and logic levels, so they do not reach the next stage at the same time. Because of this mismatch in arrival times, temporary unwanted transitions (glitches) are produced.

To reduce this effect, the delays of different paths are adjusted so that signals become more aligned with each other. When signals arrive almost at the same time, the chances of glitch formation decrease. This balancing of path delays improves the overall stability and reliability of the circuit.

Buffer for Glitch Reduction

To further minimize glitches, a small modification is introduced by placing a buffer in the generate (G) path. This buffer is implemented using two cascaded inverters and is inserted between stages of the prefix network.

The main idea behind this is simple: some signals travel faster than others, so the buffer slightly slows down the faster ones. By doing this, the timing difference between signals is reduced. As a result, all inputs to a stage tend to change more uniformly.

This helps in reducing unnecessary switching activity, which in turn lowers dynamic power consumption. In short, the buffer acts as a timing adjustment element that improves synchronization between signals and reduces glitch effects.

Transistor Optimization

After buffer delays, transistor optimization is applied to further improve the performance of the circuit. In this approach, the sizes of transistors are adjusted based on the importance of the signal path. Critical paths, which directly affect the overall delay of the circuit, are assigned larger transistor sizes to ensure faster switching. On the other hand, non-critical paths are assigned smaller transistor sizes to reduce power consumption. This selective sizing technique helps in achieving an effective trade-off between speed and energy efficiency without affecting the functional correctness of the design.

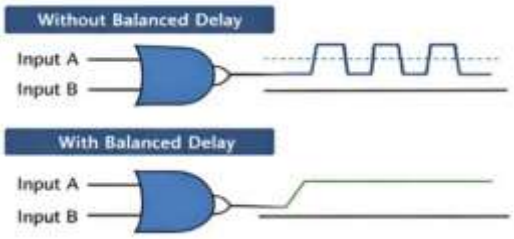


Fig. 4.2 : Optimized Block diagram

Post-Processing Stage

In the last stage, the circuit produces the final result. The sum is calculated by applying an XOR operation between the propagate signal and the carry coming from the prefix section. The carry output (Cout) is directly obtained from the final gray cell. These outputs represent the result of the binary addition operation. The post-processing stage ensures that the computed values are accurate and consistent with the expected logical behavior of an adder.

This is the final output stage.

- Sum (S) = P ⊕ Carry..... (4.4)
- Carry (Cout) from prefix network .....(4.5)

Combines propagate and carry signals. Produces final addition result.

Overall, the block diagram illustrates a well-structured flow of data from input to output, highlighting both the conventional Kogge-Stone Adder operation and the improvements introduced in the proposed design.

CMOS Gate Design

↓

Generate & Propagate (G & P)

↓

Prefix Cell Design

↓

Prefix Tree Construction
↓
Post-Processing (Sum Generation)
↓
Complete Kogge-Stone Adder
↓
Glitch Reduction (Balanced Delay)
↓
Simulation & Waveform Analysis

Fig. 4.3: Flow chart of Design

The addition of balanced delay, buffer insertion, and transistor optimization effectively reduces glitch activity and power consumption while maintaining acceptable speed performance.

5. RESULTS & DISCUSSIONS

The schematic window of the conventional Kogge-Stone Adder illustrates the transistor-level implementation of the pre-processing stage, where propagate and generate signals are derived from the input operands. The design is realized using CMOS logic in the Cadence Virtuoso environment, with proper connections to power (VDD) and ground (GND).



Fig. 5.1 : Screenshot of Conv-KSA Schematic_1 Screen Window

This stage forms the initial step in carry computation and serves as the input to the subsequent prefix network for fast parallel processing.

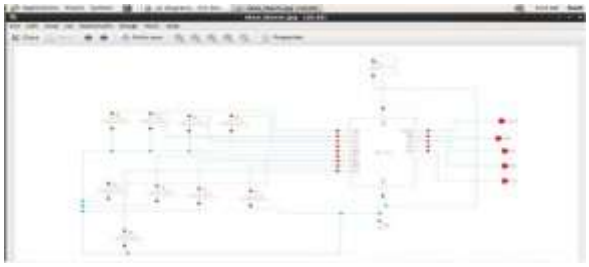


Fig. 5.2 : Screenshot of Conv-KSA tb_Schematic_2 Screen Window

This figure shows the testbench schematic of the conventional Kogge-Stone Adder implemented in the Cadence Virtuoso environment. The setup includes pulse input sources for applying dynamic signals to inputs A and B, along with the power supply (VDD) and ground connections. The outputs, namely Sum and Carry (Cout), are connected to observation nodes for waveform analysis. This testbench is used to verify the functional correctness and evaluate the performance of the conventional KSA.



Fig. 5.3 : Screenshot of Conv-KSA Simulated Wave Screen Window

The simulated waveform window of the conventional Kogge-Stone Adder shows the input and output signal transitions obtained from transient analysis in the Cadence Virtuoso environment. The input signals A and B are applied as pulse waveforms, and the corresponding outputs, Sum and Carry (Cout), are observed. The waveform exhibits noticeable glitches and unwanted transitions due to unequal propagation delays in the prefix network. These fluctuations indicate increased switching activity and highlight the limitations of the conventional design in terms of power efficiency and signal stability.

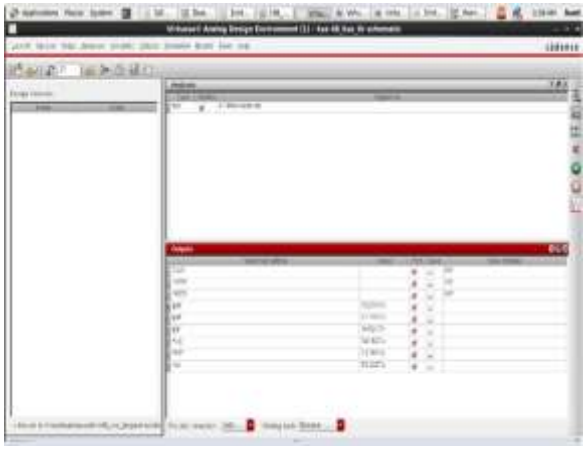


Fig.5.4 : Screenshot of Calculation Value's Screen Window

This figure shows the calculation window in the Cadence Virtuoso environment used to extract performance parameters of the circuit. The tool is utilized to measure key values such as rise delay, fall delay, propagation delay, and average power from the simulated waveforms. These calculated values are used for analyzing and comparing the performance of the conventional and glitch-reduced Kogge-Stone Adders.



Fig 5.5 : Screenshot of Schematic view of Reduced KSA Screen Window

This figure shows the schematic view of the proposed glitch-reduced Kogge-Stone Adder designed in the Cadence Virtuoso environment. The schematic includes the pre-processing stage, prefix network with black and gray cells, and the post-processing stage for sum generation. Additional modifications such as buffer insertion and delay balancing are incorporated in the generate path to reduce glitch activity. The design also reflects optimized transistor sizing for improved power efficiency and performance.

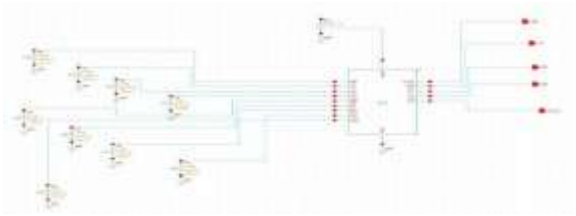


Fig.5.6 : Screenshot of Schematic Test_bench of Reduced KSA Screen Window

This figure shows the testbench schematic of the proposed glitch-reduced Kogge-Stone Adder implemented in the Cadence Virtuoso environment. The setup includes pulse input sources applied to inputs A and B, along with power supply (VDD) and ground connections. The outputs, namely Sum and Carry (Cout), are connected to observation nodes for waveform analysis. The testbench also reflects the inclusion of buffer elements and delay balancing techniques in the design, which are used to minimize glitch activity and improve overall performance.

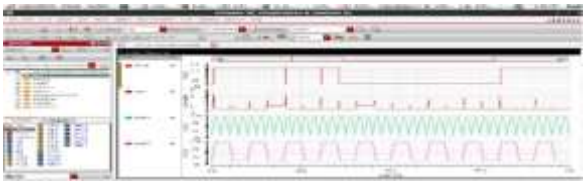


Fig 5.7 : Screenshot of Glitch Reduced-KSA Simulated Wave Screen Window

This figure shows the simulated waveform of the proposed glitch-reduced Kogge-Stone Adder obtained from transient analysis in the Cadence Virtuoso environment. The input signals are applied as pulse waveforms, and the corresponding outputs, Sum and Carry (Cout), are observed. The waveform exhibits smooth transitions with minimal glitch activity due to the implementation of delay balancing and buffer insertion techniques. This confirms improved signal stability and reduced switching activity compared to the conventional design.



Fig 4.8 : Screenshot of Reduced KSA Calculation Value's Screen Window

This figure shows the calculation window of the proposed glitch-reduced Kogge-Stone Adder in the Cadence Virtuoso environment. The simulation results are used to obtain important performance parameters such as rise delay, fall delay, propagation delay, and average power. These values are then studied carefully to understand how well the proposed design is working. They are also compared with the conventional Kogge-Stone Adder to see the improvements in performance, especially in terms of power consumption and glitch reduction.

6. CONCLUSIONS

In this project, a glitch-reduced Kogge-Stone Adder was successfully designed and analyzed. The main issue with the conventional design was excessive power consumption due to glitches caused by unequal delays. To address this, delay balancing and buffer insertion techniques were applied. Additionally, transistor sizing was optimized to improve overall efficiency.

The simulation results confirm that the proposed design reduces switching activity and improves power performance. Although there is a slight increase in delay, the overall improvement makes it suitable for modern low-power applications.

ACKNOWLEDGEMENT

We would like to express our sincere gratitude to **B. Karuna Sree** of the Department of Electronics and Communication Engineering, CMR Technical Campus, Hyderabad, for her invaluable guidance, encouragement, and continuous support

throughout the development of our project entitled **"**Design of Glitch-Reduced Kogge–Stone Adder Using Balanced Delay and Transistor Optimization in GPDK090 Technology**."** We are thankful to our faculty members and peers for their valuable suggestions and constructive feedback, which helped us improve the design and implementation of the proposed adder. We also express our gratitude to CMR Technical Campus for providing the necessary infrastructure, facilities, and resources required for the design, simulation, and performance analysis of the proposed circuit using Cadence Virtuoso.

Finally, we extend our sincere thanks to everyone who directly or indirectly contributed to the successful completion of this project.

REFERENCES

[1] S. Gupta and R. Mehra, "Design and Performance Analysis of High-Speed Parallel Prefix Adders," IEEE Access, 2021.

[2] A. Kumar and P. Singh, "Low-Power Kogge-Stone Adder Design Using CMOS Logic," Int. Journal of Electronics, 2021.

[3] M. Sharma et al., "Power Efficient Parallel Prefix Adders," Microelectronics Journal, 2022.

[4] H. Patel and S. Shah, "Design Optimization of Kogge-Stone Adder," IEEE Conference, 2022.

[5] R. Verma and S. Bansal, "Glitch Reduction Techniques in CMOS Circuits," IEEE TCAS-II, 2023.

[6] K. Reddy and P. Kumar, "FPGA Implementation of Parallel Prefix Adders," 2023.

[7] J. Lee et al., "Energy Efficient Arithmetic Unit Design," IEEE VLSI Systems, 2024.

[8] A. N. Rao and P. R. Kumar, "Balanced Delay Design for Glitch Reduction," 2024.