

Comparison of Adder Architectures and Design of a Modified Low-Power Energy-Efficient Parallel Prefix Adder

¹ Dr. Shaik Mastan Vali (profvali@mvgree.edu.in), ²A. Yogendra Sai (yogendraanakapalli@gmail.com),

³CH. Divya Sai (chdivyasai123@gmail.com), ⁴B. Sudheer (bsr2003comment@gmail.com),

⁵ Aga Naveed Abbas (aganaveedabbas@gmail.com)

¹Professor, ^{2,3,4,5}Student,

^{1,2,3,4,5}Electronics and Communication Engineering,

^{1,2,3,4,5}Maharaj Vijayaram Gajapathi Raj College of Engineering(Autonomous),
Vizianagaram,India.

Abstract - With the growing need for efficient digital circuits, particularly in processors and communication networks, there arises a need to design arithmetic units that are low in power and energy usage. These adders play a very significant role in determining not only performance but also the power consumed and the area. This work focuses on comparing some conventional adder designs with those of the parallel prefix adders with regard to delay, power, number of Transistors and Energy. From this comparative analysis, an innovative design of a low power and energy efficient parallel prefix adder is developed. The innovation involves making certain design changes in the prefix operation and carry propagation process, as well as implementing low-power measures through the Low power techniques like logic Reconstruction. All adders are modeled and simulated using standard CMOS technology library in Cadence tools (65nm technology) based on delay, power, number of Transistors and Energy. The simulation shows that the proposed design of the low power, energy efficient PPA provides significant reduction in power while retaining good speed.

Keywords: Low Power Design, Conventional Adder, Parallel Prefix Adder (PPA), Energy Efficiency, Propagation Delay, Logic Reconstruction, CMOS Technology, Cadence Tools.

1. INTRODUCTION

The growing need for energy-efficient and high-performance digital circuits has made it necessary to design adders that optimize the performance of digital circuits in terms of speed, power, and area. These are very important factors that should be considered in adder designs. Thus, there is a need for efficient adder designs that provide a balanced consideration of the discussed three important factors.

Traditional adder circuits like RCA (Ripple Carry Adder), CLA (Carry Lookahead Adder), CSKA (Carry Skip Adder), and CSLA (Carry Select Adder) are common because of their simplicity. These adders have less hardware requirements and consume less power than parallel prefix adders. But these adders have more propagation delay compared to other adders, limiting their effectiveness at high speeds.

The Parallel Prefix Adders (PPAs) provide faster carry generation, which is ideal for high-speed processing. However, there is an additional power cost associated with this feature. Compared to the regular adder circuits, the Parallel Prefix Adders consume more power and occupy larger areas. The Weinberger adder among the PPAs provides better compromise regarding the delay and power, which makes it a good choice for optimizing further.

Several designs for the various types of adders have been realized and analyzed using the Cadence CMOS 65nm technology. First a parallel prefix adder has been proposed based on the Weinberger adder in order to improve its delay characteristic. Then, the logic reconstruction method is applied in order to lower the power consumption. The proposed design achieves energy-efficient and low-power by reducing both delay and power consumption.

2. LITERATURE SURVEY

There are several adder designs that have been invented to enhance the digital systems' performance in respect of speed, power, and area. RCA, CLA, CSKA, and CSLA are some of the popularly used adder architectures since they are easy to design and consume less power and complexity than parallel prefix adders. Nevertheless, their main drawback is the higher propagation delay. Therefore, a

comparative study of various adder architectures will help understand their strengths and weaknesses.

Parallel Prefix Adders (PPAs) have been designed to counterbalance the delays associated with traditional adders through the use of parallel processing to compute the carries faster. Some of the common PPAs include Kogge–Stone and Weinberger adders. While the Kogge–Stone adder has minimal delay, it consumes higher power and occupies more space than other PPAs. The other PPAs, such as Weinberger, present a compromise between the delay and power and therefore, can be effectively used in designing optimized arithmetic logic. Most of the existing solutions are directed at addressing only delay or power consumption, making it difficult to achieve optimality in both properties simultaneously.

3. EXISTING ADDERS

The various architectures of the existing adders such as Conventional and Parallel prefix adders of 4bit,8-bit are designed and evaluated in terms of their performance. All the adders have been simulated to validate their correctness. Each adder architecture's performance is evaluated on the basis of power dissipation, propagation delay, Number of Transistors and energy consumption so that they can be compared with the new adder design proposed.

3.1 Conventional Adders

Traditional adders are simple combinational logic circuits that can be employed to carry out binary addition. They form the essential blocks for performing calculations within a digital system. They are quite straightforward in their design and construction, although they could suffer from increased delay and higher power dissipation in larger bit additions.

A. Ripple Carry adder:

The Ripple Carry Adder (RCA) is an n-bit adder circuit designed by cascading n full adders such that the carry-out from one adder is coupled to the carry-in input of the subsequent adder. In this configuration, the carry signal ripples from the LSB towards the MSB end. An RCA circuit comprises n full adders connected in sequence, and the total sum and carry

results appear at the output of the final stage of the adder circuit. The ripple effect of carry propagation makes the RCA relatively slow because its speed decreases with an increase in the bit width.

Equation for ith bit:

Sum equation is $S_i = A_i \oplus B_i \oplus C_i$

Carry equation is $C_{i+1} = A_i B_i + B_i C_i + A_i C_i$

B. Carry Look Ahead adder:

Carry-Look-Ahead Adder is an adder circuit that is intended to overcome the delay that is associated with the ripple effect in a ripple-carry adder. The idea of this adder is not to let the carry ripple from one place to another but rather calculate the carry ahead of time using parallel processing circuits.

Generate and propagate signals are used in each place for determining whether there is generation or propagation of carry. Using the value of these signals, the carry output can be directly obtained using carry look-ahead. The sum is obtained using the propagate and the carry input to the particular bit place. This leads to faster computation than the RCA. But at the same time, it leads to more complex circuit design.

C. Carry Skip adder:

The Carry Skip Adder (CSKA) is the enhanced form of Ripple Carry Adder and is designed to minimize carry propagation delay. The carry skip adder divides the adder circuit into blocks and uses carry skipping when particular criteria are met. Each bit produces both a sum and a carry bit from the input bits and the carry input, just like a full adder does. Moreover, a propagate signal is used to check if the carry should be propagated through the stage. When the propagate signals of all stages of the block become true, then the block propagate criterion is achieved. When the condition is satisfied, the input carry is skipped throughout the block and comes out of the other side of the block directly. Otherwise, the carry is computed using the ripple mechanism. This results in lower delay than the RCA but with increased hardware complexity.

D. Carry Select adder:

The Carry Select Adder (CSLA) is designed to reduce delay by computing sum and carry outputs for two possible carry inputs in parallel. Instead of waiting for the actual carry input, it assumes both conditions (carry-in as 0 and 1) and performs addition simultaneously. The structure consists of two Ripple Carry Adders for each block, where one operates with carry input as 0 and the other with carry input as 1. Once the actual carry input is known, a multiplexer selects the correct sum and carry output. This parallel computation significantly reduces propagation delay compared to RCA. However, the use of multiple adders and multiplexers increases hardware complexity, resulting in higher area and power consumption. CSLA provides a good trade-off between speed and complexity and is suitable for high-speed arithmetic applications.

E. Conditional Sum adder:

The Conditional Sum Adder (COSA) is a fast adder that minimizes delays by computing two sets of results for two potential carries. It is designed with the use of conditional cells in many levels with each level utilizing the results obtained from previous level. In order to compute two sets of sums and carries for the block, the adder performs calculations based on two sets of carries, either 0 or 1, that are used as input to the next stage of cells in the circuit. Every conditional cell uses the results calculated in previous cells and passes the correct sum and carry based on their respective carries to the next stage in the adder. Despite its fast operations, the COSA needs more hardware as compared to other adders due to its many stages and parallel computations.

3.2 Paralle prefix adders:

Parallel Prefix Adders (PPAs) are fast adders that find widespread application in VLSI implementations in order to minimize carry propagation time. As opposed to ordinary adders, PPAs are characterized by computing carry values in a parallel manner, thus improving the speed of calculations.

The principle of operation of PPAs consists of three key steps, namely pre-processing, prefix calculation, and post-processing. Pre-processing involves generation and propagation signal formation. In the process of prefix calculation, carry values are

calculated via a network of prefix cells. Finally, sum signals are calculated in the last step. Being highly parallel, PPAs ensure a higher speed than ordinary adders but consume more area and power.

A. Kogge Stone adder:

The Kogge-Stone Adder (KSA) is a very fast parallel prefix adder which helps eliminate carry propagation delay by calculating carries in parallel. Generation blocks are used in KSA to generate propagate and generate signals. Black cells help calculate propagate and generate signals, whereas gray cells generate carry signals. The EXOR gates of KSA help generate Sum signals. KSA ensure high speed but consume more power and area.

B. Weinberger Paralle prefix adder:

The Weinberger parallel prefix adder is a high-performance adder utilizing 2-bit conditional sum blocks to reduce carry logic complexity and improve computing speed. It is designed to minimize delay and power consumption for wide operands by optimizing carry generation and propagation. It skips the even carries in carry computation. This improves the overall performance of the adder and ensures optimized power and area than other Paralle prefix adders.

4. PROPOSED PARALLEL PREFIX ADDER

The suggested parallel prefix adder (PPA) is devised and implemented utilizing the optimized Weinberger recurrence formula. The structure is designed to enhance the calculation process of carry signals by omitting even carry locations and producing odd carry signals, thus decreasing computational cost and increasing processing speed. The proposed adder circuit is devised using the logic-designed carry generator unit and sum blocks of 2-bit sum to produce all sum signals (S0 to S7) and to reduce Power.

4.1 Novel Paralle prefix adder

As the Weinberger Parallel Prefix Adder offers moderate power consumption and high-speed performance, it is utilized in the proposed novel adder to achieve an efficient balance between speed and

power. The proposed adder uses the Weinberger recurrence algorithm. But, it skips odd carries and groups even carries using bitwise generate x_i , bitwise propagate y_i , group generate $X(i:j)$ and group propagate $Y(i:j)$ terms. It has yielded a significant improvement in performance by optimizing the number of stages, node counts, gates and transistor count in critical carry path which in turn optimizes delay and energy.

The carry block of the proposed novel PPA is designed to skip even carry positions while generating only odd carries, thereby reducing computation complexity and improving speed. The design schematic of the suggested PPA comprises two sections, namely the carry generation block and the sum generation blocks.

4-bit Modified PPA design:

The 4-bit Novel PPA is designed using 4-bit carry block and 2-bit sum block for all sum bits of novel adder.

Carry Equations:

$$C_1 = x_0 = a_0 b_0$$

$$C_2 = X(1:0) = x_1 + y_1 c_1$$

$$C_4 = X(3:0) = X(3:2) + Y(3:2) c_2$$

$$C_4 = (x_3 + y_3 x_2) + y_3 y_2 c_2$$

Twobit Sum block:

If $C_i = 0$

$$S_i = a_i \oplus b_i$$

Else

$$S_{(i+1)} = (a_i b_i) \oplus (a_{(i+1)} \oplus b_{(i+1)})$$

If $C_{i+1} = 1$

$$S_i = a_i \odot b_i$$

Else

$$S_{(i+1)} = \{(a_i \oplus b_i) + (a_i b_i)\} \oplus (a_{(i+1)} \oplus b_{(i+1)})$$

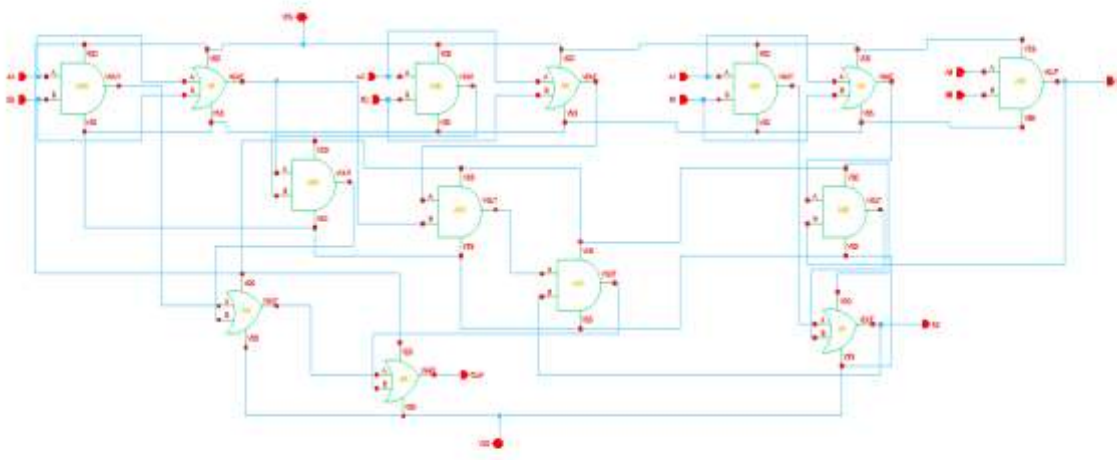


Fig -1: Carry Block of 4-bit Modified PPA

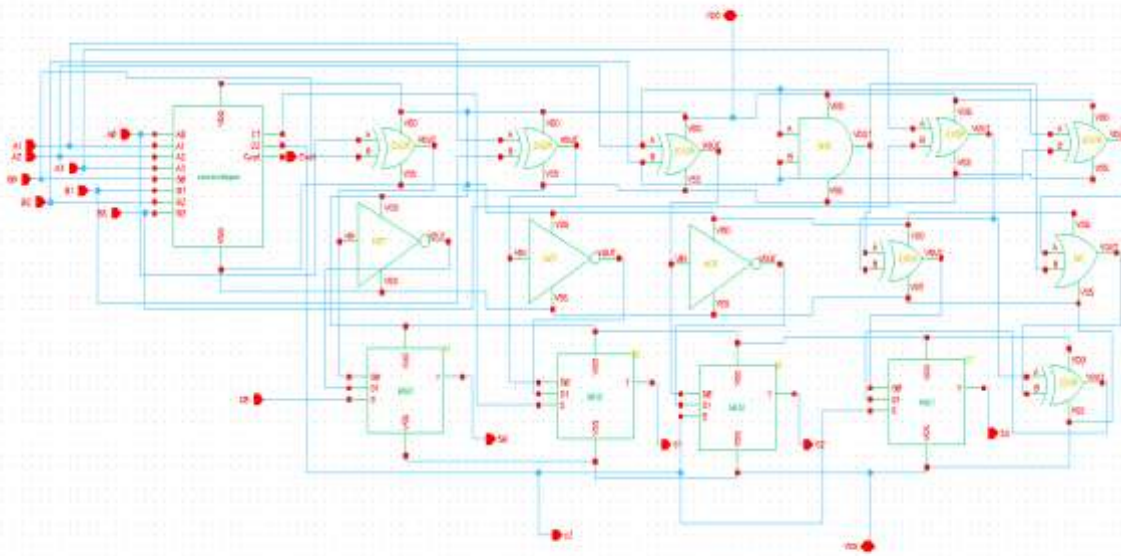


Fig -2: Schematic of 4-bit Modified PPA

4.2 Low Power Techniques:

Power dissipation reduction is one of the most crucial aspects of digital design, particularly in systems that work at higher speeds and have portability requirements. Various low power techniques exist to reduce the amount of power consumed by combinational circuits.

Logic Reconstruction:

The logic reconstruction is a method of saving power where Boolean operations are modified to decrease switching activity and glitches, which in turn results in saving dynamic power and at times even decreasing delay.

The use of logic reconstruction for the comparison is appropriate since it preserves the function, area, and delay of the circuit while making internal modifications to save power. While there can be an increase in the number of gates, this technique leads to reduced switching activity and glitches, resulting in less power consumption. As the two circuits perform in a similar manner in terms of performance

metrics such as area and delay, any difference in power can easily be observed.

4.3 Modified PPA after applying low power techniques:

While the suggested novel PPA has fast processing speed but medium power dissipation, its power requirement is greater compared to Conventional adders. Consequently, low-power techniques like Logic reconstruction have been adopted for effective reduction of power consumption. Carry block uses low power technique logic reconstruction to reduce switching activity and skips odd carries like novel PPA to reduce delay.

The schematic of the 4-bit low-power novel Parallel Prefix Adder (PPA) consists of an optimized carry generation block and 2-bit sum blocks. The carry block is designed to generate only odd carries while skipping even carries, reducing switching activity and power consumption. Low-power techniques such as NAND-based reconstruction are used to minimize hardware complexity. The sum is generated using two 2-bit blocks to produce outputs S0-S3. This structure

achieves high speed with reduced power consumption compared to conventional adders.

4-bit low power Modified PPA:

The 4-bit Novel PPA is designed using 4-bit logic reconstructed carry block and 2-bit sum block of novel adder.

Logic reconstructed Carry equations:

Carry c1:

$$c1 = x0 = a0 b0$$

Carry c2:

$$c2 = x1 + y1 c1$$

NAND reconstruction:

$$c2 = [(x1)' (y1 c1)]'$$

Carry c4:

$$c4 = (x3 + y3 x2) + y3 y2 c2$$

NAND reconstruction:

$$c4 = [(x3 + y3 x2)' (y3 y2 c2)]'$$

Twobit Sum Block:

If ci = 0

$$Si = ai \oplus bi$$

$$S(i+1) = (ai bi) \oplus (a(i+1) \oplus b(i+1))$$

If ci = 1

$$Si = ai \odot bi$$

$$S(i+1) = \{(ai \oplus bi) + (ai bi)\} \oplus (a(i+1) \oplus b(i+1))$$

8-bit logic reconstructed Modified PPA design:

The proposed novel parallel prefix adder (PPA), which is an 8-bit low-power adder, is achieved through the connection of two novel 4-bit PPA blocks. The adder is built using a logic reconstructed carry generation circuit implemented using the Weinberger recurrence algorithm where no carry bit is generated at even positions but only odd positions, thus reducing the switching activities and power dissipation. The novel prefix circuit minimizes the stages, nodes, and gates used in the critical carry chain to ensure efficient delay and power utilization. The final sum is calculated using 2-bit sum circuits to give a total of 8 bits output (S0-S7).

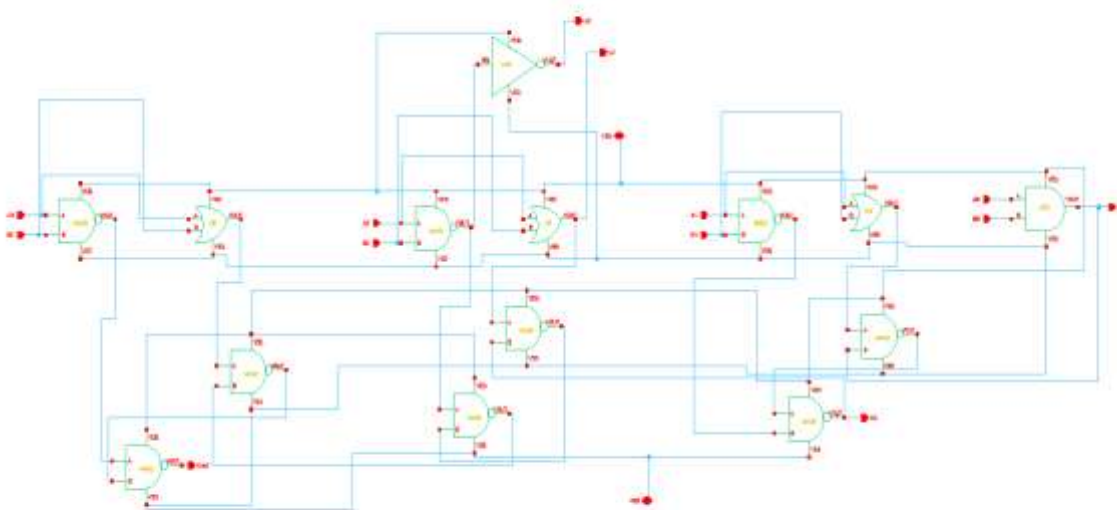


Fig -3:

Carry block of Logic reconstructed modified PPA

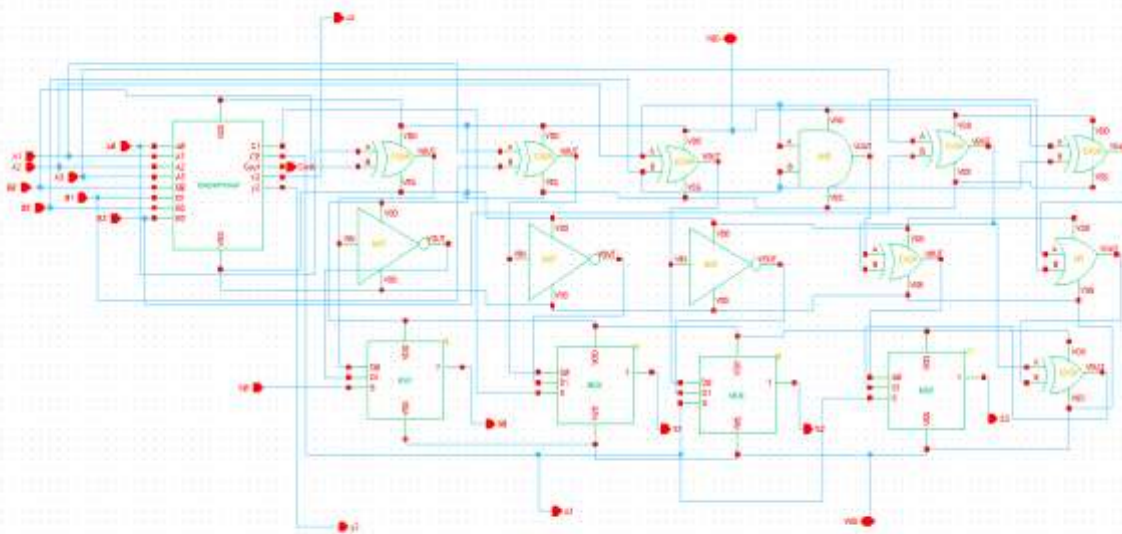


Fig -4: Schematic of Logic reconstructed modified PPA

5. PERFORMANCE ANALYSIS

Performance analysis is carried out for conventional adders, parallel prefix adders and the proposed novel PPA based on power, delay, transistor count and energy. The results are presented in tabular form. Conventional adders exhibit low power consumption but suffer from high delay. Parallel prefix adders improve delay using carry parallelism but have higher power.

The proposed novel PPA provides superior speed and lower energy consumption compared to conventional adders and PPAs, although it has slightly higher power than conventional adders. The logic-reconstructed novel PPA achieves better overall performance in terms of speed, power, and energy efficiency compared to all other adders.

Table -1: Power comparison of 4-bit adders

Adder	Power(μ w)
Ripple Carry adder	116.5
Carry Look Ahead adder	123.8

Carry Skip adder	117.2
Carry Select adder	125.4
Conditional Sum adder	141.4
Kogge stone adder	160.6
Weinberger PPA	145.01
Novel PPA	144.7
Logic reconstructed novel PPA	116.4

Table -2: Delay comparison of 4-bit adders

Adder	Delay(ps)
Ripple Carry adder	78.44
Carry Look Ahead adder	68.31
Carry Skip adder	73.06
Carry Select adder	70.54
Conditional Sum adder	54.86
Kogge stone adder	46.16
Weinberger PPA	42.35
Novel PPA	37.72
Logic reconstructed novel PPA	37.6

Table -3: Number of transistors comparison of 4-bit adders

Adder	nMOS	pMOS	Total
Ripple Carry adder	84	84	168
Carry Look Ahead adder	138	138	276
Carry Skip adder	103	103	206
Carry Select adder	218	218	436
Conditional Sum adder	132	132	264
Kogge stone adder	152	152	304
Weinberger PPA	140	140	280
Novel PPA	133	133	266
Logic reconstructed novel PPA	133	133	266

Table -4: Energy comparison of 4-bit adders

Adder	Energy(pJ)
Ripple Carry adder	0.00914
Carry Look Ahead adder	0.00846
Carry Skip adder	0.00857
Carry Select adder	0.00885
Conditional Sum adder	0.00776
Kogge stone adder	0.00741
Weinberger PPA	0.00614
Novel PPA	0.00546
Logic reconstructed novel PPA	0.00436

Table-5: Power comparison of 8-bit adders

Adder	Power((μ w))
Ripple Carry adder	321.3
Carry Look Ahead adder	351.8
Carry Skip adder	328.1
Carry Select adder	354.5
Conditional Sum adder	427.2
Kogge stone adder	461.2
Weinberger PPA	440.14
Novel PPA	441.3
Logic reconstructed novel PPA	321.1

Table -6: Delay comparison of 8-bit adders

Adder	Delay(ps)
Ripple Carry adder	158.2
Carry Look Ahead adder	140.6
Carry Skip adder	149.2
Carry Select adder	144.1
Conditional Sum adder	103.7
Kogge stone adder	92.35
Weinberger PPA	85.8
Novel PPA	80.44
Logic reconstructed novel PPA	80.5

Table -7: Number of Transistors comparison of 8-bit adders

Adder	nMOS	pMOS	Total
Ripple Carry adder	168	168	336
Carry Look Ahead adder	276	276	552
Carry Skip adder	206	206	412
Carry Select adder	302	302	604
Conditional Sum adder	264	264	528
Kogge stone adder	304	304	608
Weinberger PPA	280	280	560
Novel PPA	266	266	532
Logic reconstructed novel PPA	266	266	532

Table-8: Energy comparison of 8-bit adders

Adder	Energy(pJ)
Ripple Carry adder	0.0512
Carry Look Ahead adder	0.0493
Carry Skip adder	0.0484
Carry Select adder	0.0496
Conditional Sum adder	0.0443
Kogge stone adder	0.0426
Weinberger PPA	0.0378
Novel PPA	0.0355
Logic reconstructed novel PPA	0.0259

6. CONCLUSION

This paper offers a comparison between various adder structures like Ripple Carry Adder (RCA), Carry Look Ahead Adder (CLA), Carry Skip Adder (CSA), Carry Select Adder (CSLA), Conditional Sum Adder (CSA), Kogge Stone Adder (KSA), Weinberger Adder (WA), and the newly proposed modified Parallel Prefix Adder (PPA). Evaluation is done based on metrics such as power, delay, number of transistors used and energy consumption of the proposed adder designs in both 4-bit and 8-bit operations. In the case of 4-bit adders, the modified parallel prefix adder provides better performance with a delay of 37.6 ps, power consumption of 116.4 μ W, and energy of 0.00436 pJ, surpassing conventional architectures like RCA and CLA. Similarly, in the case of 8-bit adders, the newly proposed adder structure offers better performance than RCA and Kogge Stone Adder with a delay of 80.5 ps, power of 321.1 μ W, and energy of 0.0259 pJ. Overall, the proposed architecture achieves a better trade-off between speed and power consumption, demonstrating its effectiveness for low-power and energy-efficient arithmetic applications.

ACKNOWLEDGEMENT

I would like to express my sincere gratitude to everyone who supported and guided me throughout the completion of this project.

First and foremost, I sincerely thank my mentors for their valuable guidance, continuous support, and encouragement throughout this work. Their insightful suggestions and constructive feedback helped me to understand the concepts more clearly and significantly improved the quality of this project.

REFERENCES

1. Bavusaheb Kunchanur, Swapna Srinivasan: Design and analysis of full adder using different low power techniques (2015) 1–4.
2. B. Koyada, N. Meghana, Md. O. Jaleel, P. R. Jeripotula: A Comparative Study on Adders. Proc. IEEE Conf. Wirel. Commun. Signal Process. Netw. (2017).

3. Low Power Design Strategies and Power Management Techniques in VLSI. In: Proc. IEEE Conf. Emerg. Devices Smart Syst. (ICEDSS), Mahendra Eng. Coll., Tamil Nadu, India (2017). ISBN: 978-1-5090-5555-5.
4. Manjunath K. M., Abdul Lateef Haroon P. S., Amarappa Pagi, Ulaganathan J.: Analysis of various full-adder circuits in Cadence. Proc. Int. Conf. Emerg. Res. Electron. Comput. Sci. Technol. (2015) 90–97.
5. N. Jagadeeshkumar: A Systematic Design of Novel Energy Efficient 64-bit Parallel Prefix Adder. M.E. thesis, Anna University (2021). Available:Shodhganga(INFLIBNET Centre).
6. Sameen Gauhar, Adeeba Sharif, Naushad Alam: Comparison of Parallel Prefix Adders Based on FPGA & ASIC Implementations. Proc. IEEE Students Conf. Eng. Syst. (SCES) (2020).
7. Shweta Dubey, Gaurav Verma: Analysis of Basic Adder with Parallel Prefix Adder. Proc. IEEE Int. Conf. Meas. Instrum. Control Autom. (ICMICA) (2020).
8. U. Penchalaiah and S. K. VG: Design of High-Speed and Energy-Efficient Parallel Prefix Kogge Stone Adder. In: Proc. 2018 IEEE Int. Conf. Syst. Comput. Autom. Netw. (ICSCA), Pondicherry, India (2018).