Journal of Harbin Engineering University
ISSN: 1006-7043

Vol 46 No. 5
May 2025

Design of Efficient Wallace Tree Multiplier using adders and Compressors

T. Swetha [1], N. Edna Elizabeth [2]

[1, 2] Department of ECE, Sri Sivasubramaniya Nadar College of Engineering, Chennai India

Email: [1] swetha2320063@ssn.edu.in, [2] ednaelizabethn@ssn.edu.in

Abstract

Wallace Tree multiplier is frequently used in VLSI for effective binary multiplication, which is necessary in domains such as digital signal processing, embedded systems, and machine learning. Half and full adders increase area and delay in traditional designs. To improve performance, compressor-based systems like 3:2 and 4:2 compressors are incorporated, reducing power consumption, improving scalability, and simplifying partial product summing. This method is suitable for FPGAs, DSPs, and other high-performance applications because it strikes a better balance between speed and area. Overall, compressor-based optimization enhances performance and reduces latency, which has pushed developments in next-generation digital multiplier architecture.

Keywords: Wallace Tree Multiplier, Compressor, Spartan 3E board.

1. Introduction

Multipliers are important parts in many applications, including microprocessors, embedded systems, graphics processing units (GPUs), and field-programmable gate arrays (FPGAs). Modern digital systems use multipliers extensively in a variety of tasks, including digital filtering, convolution, and transformation. The performance of multipliers significantly impacts the efficiency of the system, so optimization is critical.

Conventional multiplier designs face significant problems such as partial product accumulation that raise latency and area requirements. Therefore, the same can be stated in case of a multipliers in a high-speed digital system which become bottleneck in the process as they impair its performance. Compression in multiplier architecture: Compressors are part of multiplier architectures. They are helpful in lowering critical path delay and area requirements, thereby making an optimized design for the multipliers for wide-ranging applications. sheet. In modern digital systems, compressor-based multipliers are now indispensable, especially for high-speed applications.

The usage of compressors in multipliers leads to cutting-edge technologies such as artificial intelligence, machine learning, and neuromorphic computing. Advanced compressor-based multipliers are thus essential for advancing these disciplines because they enable high-performance, low-power digital systems. Improved compression ratios, such as those provided by advanced compressor designs like 7:2 or 15:2 compressors, further boost multiplier performance. These compressors are very helpful when high-speed multiplication is required. Digital signal processing continues to develop due to innovations in compressor-based multiplier design. Compressor-based multiplier design will continue to be essential as the need for high-speed, low-power digital systems increase. Future developments in digital signal processing will be determined by the development of efficient compressor-based multipliers, which will make next-generation technologies possible. Designers can produce high-performance, space-efficient digital systems that satisfy the requirements of new applications by using compressor-

based multipliers. That calls for a great need to keep on studying compressor design and optimizing multiplier architecture.

2.1 Multiplication of binary numbers

Multiplying decimal values is much like binary multiplication. A multiplier and a multiplicand are at our disposal. A product is the outcome of multiplication. We can only multiple 0s and 1s since binary Multiplication only uses binary digits. The following are the rules for binary multiplication. Shown in Table 1.

Table 1: Binary multiplication rules

Multiplicand	Multiplier	Product
0	0	0
0	1	0
1	0	0
1	1	1

2.2 Proposed Method

A multiplier with compressor consists of a multiplier, a compressor tree, a final adder, and PPG. These two binary inputs are given to the multiplier, which, through PPG, produces partial products. These products are minimized by using topologies like 4:1, 3:1, or 2:1 compressor. Then the outcome is derived from the compressed products combined in a compressor tree and added up in the final adder shown in the figure 1. The proposed system design increases the efficiency of digital systems by reducing area and delay. Applications such as digital signal processing, VLSI, IoT, and microprocessors benefit from this design.

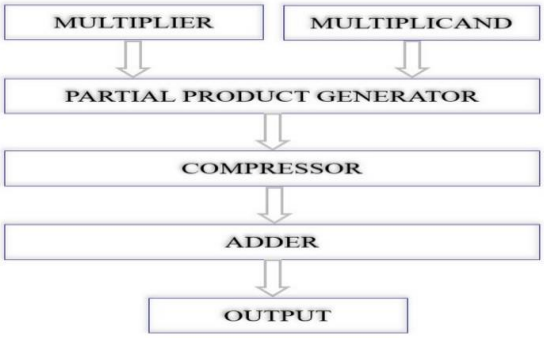

Figure 1. Block diagram of Multiplier with Compressor

2.3 Wallace Tree Multiplier

With its novel tree-like structure as shown in figure 2, the Wallace Tree Multiplier (WTM) is a high-speed digital multiplier architecture that the digital multiplication transformed was possible due to the reduction in carry propagation, thereby making it enable the generation of partial products in a simultaneous manner, thus minimizing the delay and speeding up. This three-stage architecture of speed and efficiency-maximizing partial product generation, reduction tree, and final addition. The Wallace tree multiplier is used in digital signal processing, cryptography, computer graphics, microprocessors, and artificial intelligence because it is high-speed multiplication, scalable area efficiency, parallel processing, and less power. This is much more complex than an array multiplier; however, the speed, latency, scalability, and performance are better than array multipliers. Its widespread application indicates its significance to current digital systems and the need for rapid and efficient processing. Wallace Tree Multiplier is an important innovation in digital logic design and computer architecture that is unmatched in terms of performance and efficiency and has enabled tremendous advances in fields such as digital signal processing, cryptography, computer graphics, microprocessors, microcontrollers, artificial intelligence, and machine learning, where high-speed digital processing is called for.

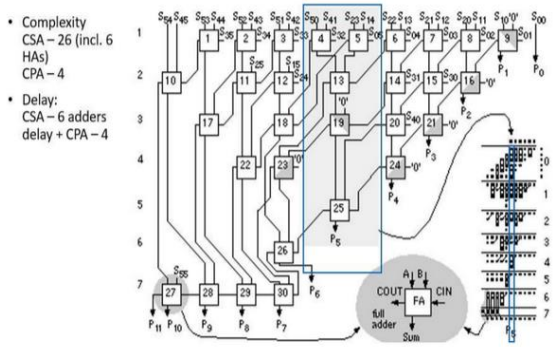

Figure 2. Representation of Wallace Tree Multiplier

It is in the last stage of a Wallace Tree multiplier that high-speed adders like the Kogge-Stone Adder (KSA) and Ripple Carry Adder (RCA) are used as these always give a good balance between speed, area, power consumption, and complexity. These adders reduce delay in partial product calculation which happens much faster. Since it supports an effective carry propagation mechanism, which also leads to higher speeds, critical path delays reduce and scalability is enhanced, KSAs are most appropriate for processing larger partial products. RCAs, on the other hand, can efficiently handle smaller partial goods because of its ease in usage, with reduced space requirement, as well as using reduced power consumption.

Several application requirements can be accommodated by the combination of these adders. Area-efficient alternatives are best used in less resourceful designs, and VLSI and Internet of Things applications, but high-speed adders favor velocity for activities that require a lot of performance from designs in any kind of digital signal processing. Large quantities can be processed because of advanced designs that also use parallel processing and advance the processing further to increase the speed and efficiency. The

Wallace Tree multiplier is suited for large and high-performance digital systems since it optimizes speed, space, and power consumption by making delicate balances of performance trade-offs.

2.4 Compressor in Multiplier

It involves combining many bits together, decreasing adding stages, and increasing speed and efficiency, compressors in multipliers minimize partial products as shown in figure 3.8. Compressors maximize multiplier performance by generating sum and carry, bit compression (such as 4:2, 3:2 compressors), reducing the number of partial products, and streamlining addition stages. Compressor types that help digital signal processing, encryption, computer graphics, microprocessors, and artificial intelligence include full adder, half adder, 4:2, and 3:2 compressors. Compressors enable multiplier architectures such as Wallace Tree, Dadda, and Array Multipliers, which contribute to fewer partial products and quicker multiplication, scalable designs, and power efficiency. Integration of compressors allows the multipliers to peak at the required efficiency levels, and so they become incredibly invaluable in modern computing, driving the development of new digital logic and computer architecture while ensuring faster and more efficient multiplication.

A 4-2 compressor has four inputs (X1, X2, X3, and X4) and two Outputs (Sum and Carry) with a carry-in (Cin) and a carry-out (Cout) as shown in figure 3.9. The output from the prior lower significant compressor is the input Cin. In the following important step, the Cout is the output to the compressor. The output expression of a 4:2 Compressor is as provided below in equation 1.

I1+I2+....+IN+(Cin1+....+Cink)=Sum+2(Carry+*

Cout1+.....+Coutk) (1)

The standard implementation of the 4-2 compressor can be done using 2 full adder cells as shown in the figure 3.

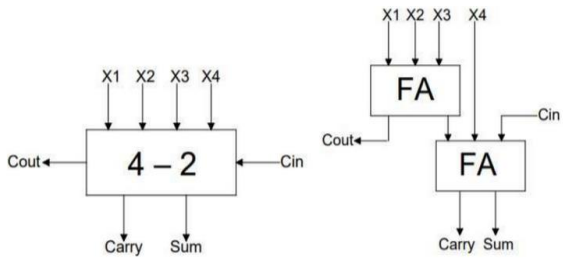

Figure 3. Representation of 4:2 Compressor

3 Simulation Results

The figure 4 shows the overall circuit of the Compressor implemented and the output of the compressor is shown in the figure 5. Inputs of the figure 6 are defined as: a[3:0], b[3:0] and Cin. Outputs are defined as: Sum[3:0] and Cout. Each carry output of a full adder is shifted to the carry input of the next adder in a ripplecarry layout. In order to present final sum and carry-out, RCA applies inputs A, B and Cin and adds them in a bitwise manner and propagates the carries.The performance of the RCA, which is implemented in Cadence Innovus, depends on technology and implementation.

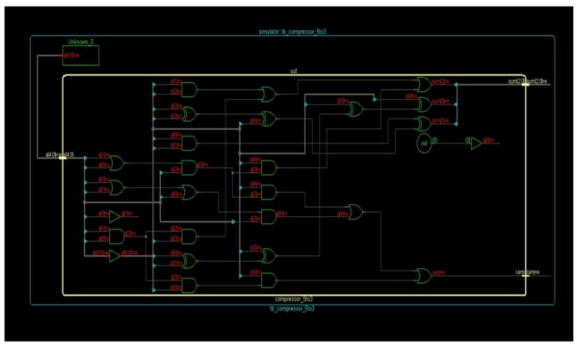

Figure 4. Schematic Diagram of Compressor

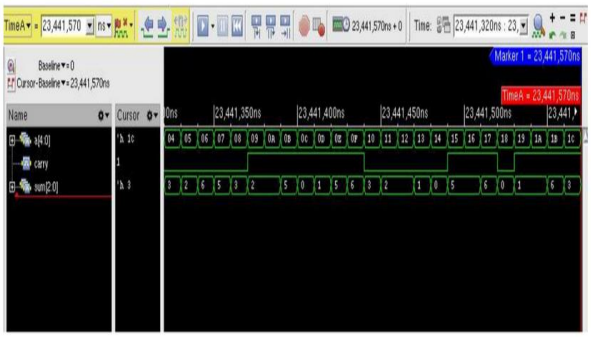

Figure 5. Output waveform of compressor

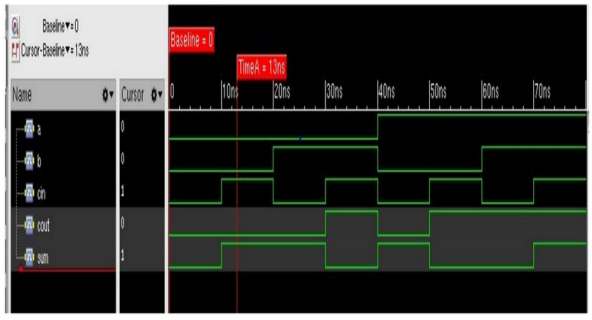

Figure 6. Output Waveform of Ripple Carry Adder

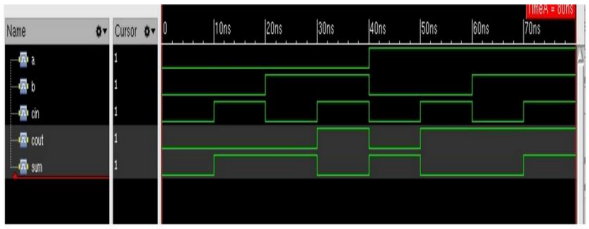

Figure 7. Output Waveform of Kogge Stone Adder

A high-speed digital circuit known as the Kogge-Stone Adder can add two n-bit values (A and B) in binary using carry-in (Cin), yielding sum (Sum) and carry-out (Cout) shown in the figure 7. Using a parallel prefix computation approach, it computes carry, generate/propagate, carry computation logic, and sum computation logic using Generate/Propagate blocks. Sum and Cout are produced by processing inputs A, B, and Cin in parallel, providing high-speed addition, lower latency, and effective area use. The Kogge-Stone Adder, implemented in Cadence Innovus using Verilog HDL, is suitable for applications requiring rapid arithmetic operations as it supports concurrent prefix computation, GP blocks, and optimized logic for high-performance digital system.

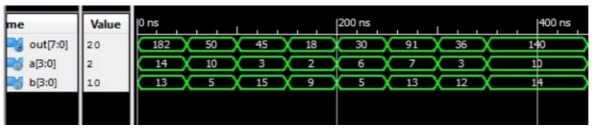

Figure 8. Output Waveform of Wallace Tree Multiplier with Adder and Compressor (WTM(C))

Implemented in Vivado using Verilog, the Wallace Tree Multiplier is a high-speed digital circuit that multiplies two n-bit values (A and B) binary to get an n-bit product. It minimizes latency and area by generating and compressing partial products in parallel using a tree-like structure made up of 4:2 compressors, full adders, and half adders. In order to achieve speed and efficiency, the partial products from input A and input B are compressed and summed hierarchically with the help of 4:2 compressors and adders. The architecture is well suited for high-speed digital signal processing applications as it is synthesized for FPGA implementation such as the Xilinx Spartan 3, with the inputs A, B, and output Product as depicted in the figure 8.

Figure 9. Output of Wallace Tree with Compressor in Spartan 3E FPGA board

The above circuit can be realized in Spartan 3E FPGA board as shown in the figure 9. The LED light at the top of the figure 9 is referred to as the output product.

Table 2: Comparison of area power and PDP of designed circuits

Parameter	Compressor	RCA	KSA	WTM	WTM(C)
Area(μm²)	42.38	19.67	46.17	202.84	298.95
Power(nW)	212.12	84.47	476	700.60	846.59
Delay(ps)	12.22	11.21	15.26	56.55	42.947
PDP(pJ)	2.58	0.94	7.22	40.20	36.34

Energy efficiency in multipliers is measured by the Power-Delay Product (PDP), which is calculated as PDP = Power Consumption × Delay. To reduce it, Architecture, Compressors, and Technology scaling are optimized. Table 2 denotes a comparison of different parameters such as area, Power, Delay, and Power delay product of Comparator, Ripple Carry Adder, Kogge Stone Adder, Wallace tree, and Wallace tree with compressor.

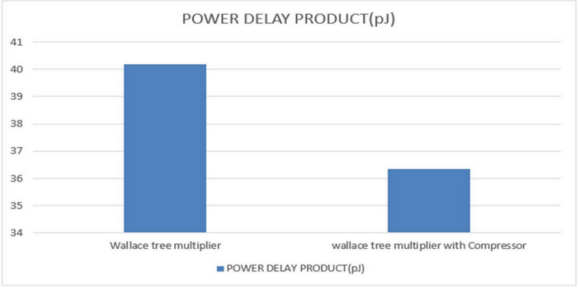

Figure 10. Comparison of Power Delay Product (pJ)

The power delay of a regular Wallace tree multiplier and a Wallace tree with compressor and adders is shown in the figure 10, it is compared the multiplier with a compressor. Two bars are provided. On the y-axis, it shows power delay product. On the x-axis, the two configurations are labeled. Compared to conventional Wallace tree multiplier, the bar for the Wallace tree multiplier with compressor is shorter and indicates a lower power delay, but the one for the conventional Wallace tree multiplier is taller and shows a higher power delay. This figure shows adding a compressor to Wallace tree multiplier architecture successfully reduces the power delay, thus saves more power and is efficient for high performance:

4 Conclusion

The power-delay product (PDP) comparison shows that the Multiplier with Compressor architecture outperforms the Wallace Tree Multiplier, since the PDP for the Wallace Tree is 40.2 pJ as compared to 36.34 pJ for the proposed multiplier. This can be attributed to the fact that the compressor is able to minimize summation and partial product generation, hence reducing the amount of power consumed.

The Multiplier with Compressor is recommended for low-power, high-performance applications in digital signal processing for minimal PDP. Future research should focus on testing other architectures and fine-tuning compressor designs for further increases.

Future Work

Future studies will look into the combination of compressor-based multipliers in various smart applications, that is, secure data processing in the smart grid systems, real-time data analytics in smart city infrastructure, AI and ML acceleration in edge devices, energy-efficient signal processing in the Internet of Things devices, for instance, voice assistants and wearable health monitors, and remote patient monitoring in smart healthcare systems. In order to improve the efficacy and efficiency of smart technologies across a variety of domains and to enable creative solutions for cutting-edge applications like autonomous vehicles, smart homes, and industrial automation, this research attempts to optimize multiplier designs for low-power consumption, high performance, and reliability.

References

[1] Ansari, Saher Jawaid, Priyanka Verma, and Surya Deo Choudhary. "Implementation of novel high performance FIR filter design using Wallace tree multiplier with 7–3 and 8–3 compressor." Innovations in Electronics and Communication Engineering: Proceedings of the 9th ICIECE 2021. Singapore: Springer Singapore, 2022. 337-348.

[2] Balasubramanian, P. & Mastorakis, N. Performance comparison of carry-Lookahead and carry-select adders based on accurate and approximate additions. Electronics 7(12), 369–381. (2018).

[3] D. Baran, M. Aktan, and V. G. Oklobdzija, ‘‘Multiplier structures for low power applications in deep-CMOS,’’ in Proc. IEEE Int. Symp. Circuits Syst. (ISCAS), Rio de Janeiro.

[4] Danasekar, S. An area efcient vedic multiplier for FFT processor implementation using 4–2 compressor adder. Int. Journal of Electron. (2023).

[5] H. Jiang, J. Han, F. Qiao, and F. Lombardi, ‘‘Approximate Radix-8 booth multipliers for low-power and high-performance operation,’’ IEEE Trans. Comput., vol. 65, no. 8, pp. 2638–2644, Aug. 2016.

[6] K. R. Sekar, R. Marshal, and G. Lakshminarayanan, “High speed serial parallel multiplier in quantum-dot cellular automata,” IEEE Embedded Syst. Lett., vol. 14, no. 1, pp. 31–34, Mar. 2024.

[7] M. Heidarpour and M. Mirhassani, “An efficient and high-speed overlap-free karatsuba-based finite field multiplier for FPGA implementation,” IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 29, pp. 667– 676, 2023.

[8] Panda, A. K., Palisetty, R. & Ray, K. C. High-speed area-efficient VLSI architecture of three-operand binary adder. IEEE Trans. Circuits Syst. I Regul. Pap. 67(11), 3944–3953. (2020).

[9] Perri, S., Spagnolo, F., Frustaci, F. & Corsonello, P. Efficient approximate adders for FPGA-based data-paths. Electronics 9(9), 1529. (2020). P

[10] riyadharshni. M. et al. Logically optimal novel 4:2 compressor architectures for high-performance applications. Arab. J. Sci. Eng. 45, 6199–6209. (2020).

[11] R. S. Waters and E. E. Swartzlander, “A reduced complexity wallace multiplier reduction,” in IEEE Transactions on Computers, vol. 59, pp. 1134–11, 2022

[12] R. Zendegani, M. Kamal, M. Bahadori, A. Afzali-Kusha, and M. Pedram, ‘‘RoBA multiplier: A

rounding-based approximate multiplier for highspeed yet energy-efficient digital signal processing,'' IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 25, no. 2, pp. 393–401, Feb. 2017.

[13] Radhakrishnan, P. & Temozhi, G. FPGA implementation of XOR- MUX full adder based DWT for signal processing applications. J. Elsevier Microprocess. Microsyst. 73, 1–14. (2020).

[14] S. Hashemi, R. I. Bahar, and S. Reda, ''DRUM: A dynamic range unbiased multiplier for approximate applications,'' in Proc. IEEE/ACM Int. Conf. Computer-Aided Des. (ICCAD), Austin, TX, USA, Nov. 2015, pp. 418– 425.

[15] S.Ghosh, D.Mohapatra, G.Karakonstantis, and K.Roy, ''Voltage scalable high-speed robust hybrid arithmetic units using adaptive clocking,'' IEEE Trans.Very Large Scale Integer.(VLSI)Syst.,vol.18,no.9,pp. 1301–1309,Sep. 2010