Multiplier using Low Power Compressors

JALLI SUNDAR SINGH , MTech (VLSI AND EMBEDDED SYSTEMS)
Department of Electronics and Communication Engineering, Anil Neerukonda Institute of Technology & Sciences, Visakhapatnam, Andhra Pradesh, India - 531162.
Email: sundarjalli@gov.in

Ravi Kumar Sariki
Assistant Professor, Department of Electronics and Communication Engineering, Anil Neerukonda Institute of Technology & Sciences, Visakhapatnam, Andhra Pradesh, India - 531162.
Email: ravi.cool434@gmail.com

***Abstract*— The current VLSI systems face two important design consideration issues: power dissipation and computation speed, with the increasing demand for the performance of portable computing and multimedia devices. One of the most important arithmetic components, multipliers, are a considerable contributor to the overall power consumption and the processing time. In this work, a low power and high performance 8x8 wallace tree multiplier with changed compressor topologies is shown. The proposed method uses the optimised 4:2 and 5:2 compressors designed with the XOR-XNOR logic and multiplexer assisted structures for better efficiency of partial product reduction. The architecture is designed to reduce switching activity, shorten the critical path and enhance the computational performance while keeping the hardware complexity down. The modified Wallace reduction stage reduces the partials before summing them up to get the final sum. The entire design has been written in Verilog HDL and simulated using Xilinx Vivado 2024.2. The proposed architecture offers an efficient hardware design solution for high speed and low power applications with multipliers in VLSI systems.**

"Index Terms — Wallace Tree Multiplier, Compressor Architecture, XOR–XNOR Logic, Low-Power VLSI, Verilog HDL, High-Speed Multiplication."

I. INTRODUCTION

The fast growth of portable electronics, multimedia systems and high performance embedded platforms has led to a tremendous increase in the demand for efficient arithmetic hardware in modern VLSI systems . One of the most crucial arithmetic blocks are the multipliers, as they directly affect the speed, power usage and complexity of the hardware used in a system. The multiplication operation is extensively employed in digital signal processing, image processing, communication systems, cryptographic applications and processor architectures. Hence, reduced latency and low power consumption multipliers design is an important research objective in the design of VLSI circuits [10].

Multiplier topologies usually consist of three major stages: partial product production, partial product reduction and final addition. These phases are effective on the multiplication process, with their efficiency. The conventional implementations of the multipliers can lead to a greater delay and power consumption due to the large number of arithmetic operations required to accumulate partial products. In order to address these limitations, tree-based reduction techniques have been proposed to speed up multiplication processes and increase the computing throughput [11].

The existing multiplier structures that are used are often the Wallace Tree multiplier as it can be used to reduce the partial products in parallel and reduce the critical path latency. Wallace reduction technique can get the intermediate results reduced in several steps which makes it faster than the regular array based multipliers [9]. But, although these Wallace tree structures increase the speed, most of the benefits are largely dependent on the performance of the internal reduction devices like compressors and counters.

Complete adder is the basis of high speed arithmetic elements, called compressor circuits, which are often implemented in 3:2 or 4:2 or 5:2 configurations. These compressors process multiple input bits and compress them to less output bits while still being mathematically correct. Previously, it has been demonstrated that the compressor topologies optimized can significantly enhance the speed of the multipliers while consuming less power and hardware overhead [1]. Progress has been made in applying low-voltage versions of compressors and optimisation at the transistor level [2].

In recent years, with the development of XOR-XNOR logic design, many benefits have been shown for decreasing the switching activity and optimal number of transistors in arithmetic circuits [4]. Fast architectures for XOR-XNOR gates are possible which can achieve higher speed and lower power consumption compared to the standard CMOS approach [8]. Encouraged by the above findings, in this study, an optimised 4:2 and 5:2 compressor structure based Wallace Tree multiplier is proposed. The proposed design realizes XOR-XNOR logic to improve the efficiency of partial product reduction, and reduce the critical path time of the multiplexer.

All the architecture is designed using Verilog HDL and functionally simulated using Xilinx Vivado 2024.2. The approach suggested is to design a small and efficient multiplier architecture for low power and high-speed VLSI applications by redesigning different types of compressors in the Wallace reduction step.

II. RELATED WORK

The increasing need of high speed and energy efficient digital systems has rendered the arithmetic circuits optimisation as an important study topic of VLSI design. Multipliers have garnered much attention among arithmetic units since they add a great deal to the total system delay, switching activity and power consumption. In order to achieve an acceptable level of hardware complexity, while still gaining better computational performance, different architectures and reduction methods of the multipliers have been suggested [13].

The tree structure of multiplication has been suggested as a very efficient way of accelerating the arithmetic operations through simultaneous reduction of partial products. Wallace suggested a fast multiplication algorithm that used a partial product technique that was reduced in a hierarchical manner. This technique greatly reduces the computational time as compared to standard multiplication techniques [11]. Later

improvements developed these concepts by providing optimised reduction structures and different compressor arrangements to further increase hardware efficiency [12]. The importance of optimising Wallace tree multipliers for high performance implementation in terms of operating frequency and critical path delay has been found [9]. The generic Wallace reduction principle is shown in Fig. 1 and the multiplier design is based on this principle.

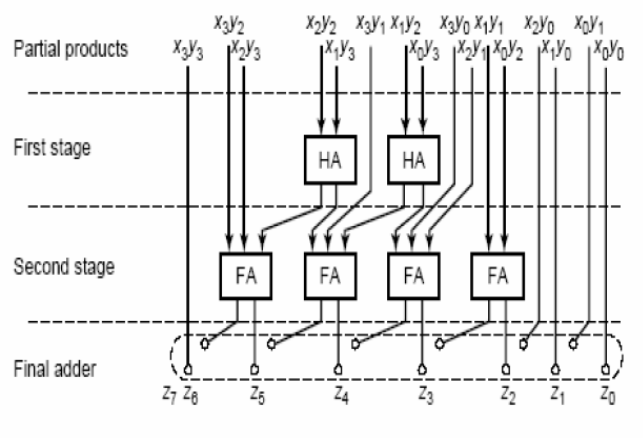

Fig.1 Wallace Tree Multiplier

Since the need for reduction efficiency is a very important study field, the compressor based arithmetic structures have been an important one. Compressor circuits compress a large number of input bits into a smaller number of output bits and preserve the arithmetic correctness, so reducing number of reduction stages in the implementation of the multipliers. Some studies have been done on the optimised compressor architectures (3:2, 4:2, 5:2), for faster and more power efficient arithmetic circuits [1]. Transistor level optimisation and switching activity minimisation within the low-voltage compressor was shown to significantly improve the overall performance characteristics [2]. Likewise, efficient architectures for both counter and compressor are shown to improve the throughput of the multiplication with minimal propagation delay [3].

Besides topology of the compressor, the logic implementation techniques have also contributed to performance enhancements in arithmetic operations. To minimize energy consumption in digital systems, while keeping the computing capacity, low power logic designs and architectures of transistor efficient were examined [5]. The XOR and the XNOR circuits are especially interesting because they are widely used to perform arithmetic operations and to construct a compressor. Several transistor level XOR–XNOR designs were designed to minimise the number of transistors and increase the speed characteristics [8]. Subsequent work suggested optimum designs of XOR–XNOR gates for low power processing and efficient signal transfer [6]. The feasibility of getting enhanced arithmetic performance with lower hardware overhead was further established by compact transistor level implementations [7]. Moreover, the low power full adder architectures using the XOR-XNOR concepts have inspired the use of the concept for the compressor based multiplier architectures [4].

Previous work has achieved great progress on compressor design and multiplier optimisation, but, most of the current techniques mainly focus on either speed or power optimisation individually. There are some architectures that have added complexity in the hardware, or that make use of many transistors, to get speed increases. In addition, some of the classical compressor designs produce excessive switching activity and delay in the partial product reduction, limiting their efficiency in the compact multiplier architectures. A comparative evaluation of the existing approaches and their limitations is given in Table 1.

Table.1 Summary of Existing Approaches and Identified Research Gap

Existing Technique	Primary Objective	Limitation Identified	Proposed Improvement
Conventional Multiplier Architectures	Perform multiplication using standard arithmetic implementation	Higher delay and increased hardware utilization during partial product accumulation	Introduce compressor-assisted reduction to improve multiplication efficiency
Wallace Tree Multiplier	Reduce multiplication delay through parallel partial product reduction	Performance strongly depends on efficiency of internal reduction circuits	Employ optimized compressor structures inside Wallace reduction stages
Compressor-Based Reduction (3:2, 4:2, 5:2)	Reduce intermediate rows and improve processing speed	Existing compressor designs may increase switching activity and logic dependency	Use modified compressor architecture for efficient reduction
Low-Voltage Compressor Designs	Lower power consumption and improve energy efficiency	Often introduce implementation complexity or limited speed improvement	Balance power and speed through architecture-level optimization
XOR–XNOR Logic-Based Arithmetic Design	Reduce transistor count and improve signal efficiency	Conventional integration may not fully utilize generated intermediate outputs	Reuse XOR–XNOR outputs through multiplexer-assisted implementation
Existing Low-	Improve either speed	Difficulty achieving	Integrate XOR–

Power Multiplier Optimization Techniques	or power independently	balanced performance with compact hardware	XNOR and compressor optimization within Wallace Tree architecture
Proposed Modified Wallace Tree Multiplier	Improve partial product reduction efficiency while maintaining compact implementation	Requires further quantitative evaluation using synthesis and power analysis for large-scale hardware validation	Modified 4:2 and 5:2 compressors integrated with XOR–XNOR and multiplexer-assisted Wallace reduction

Given this, this study proposes a variation of the Wallace tree design for the modified compressor with XOR-XNOR logic and multiplexer assisted implementation for 4:2 and 5:2 compressors. The proposed approach will focus on the improvement of the reduction efficiency by reducing the critical path latency, logic transitions, and retaining a compact architecture. The design approach is to develop a balanced solution to implement the multipliers for the current VLSI applications at low power and high speed.

III. Methodology

The appropriate architecture and design approach for the implementation of the multiplier is described in this section to enhance the efficiency of the VLSI multiplication process. The proposed study is to reduce the power consumption and propagation delay of the partial product accumulation by restructured compressor architectures in an 8x8 Wallace WT multiplier. The overall design is coded in Verilog HDL and simulated using the Xilinx Vivado 2024.2.

A) Wallace Tree Multiplier Architecture

The ability of the partial product reduction is the key to the effectiveness of the multipliers. In the traditional multiplication, partial products are stacked sequentially resulting in more delay and hardware overhead. The performance is improved due to parallel reduction of the partial products using the reduction stages as done in the Wallace Tree multiplier.

There are three steps to multiplication:

- Partial product generation
- Partial product reduction
- Final addition

First, the partial products are calculated by performing AND on bits of the multiplicand with bits of the multiplier. These intermediate products are then compressed, one by one, in the reduction steps, to reduce the number of rows before the last step of summation.

The basic design assumes the commonly used 8×8 Wallace Tree topology shown in Fig. 2.

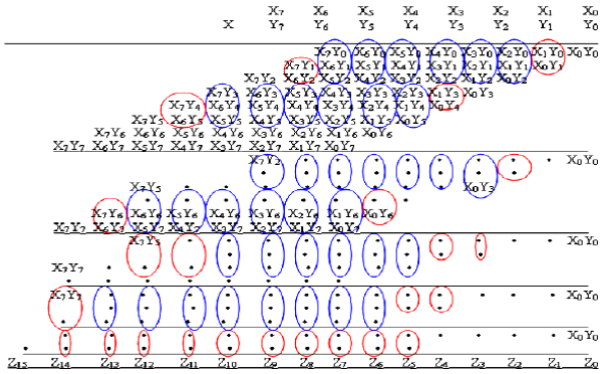

Fig.2 Conventional 8×8 Wallace Tree Multiplier

The Wallace Tree structure is able to reduce the critical route latency and is faster than the traditional multiplication methods, and can be integrated with the proposed compressor based optimisation.

B) Proposed 4:2 Compressor Design

Partial product accumulation will need the use of compressor circuits in multiplier architectures to reduce the many input bits. A modified 4:2 compressor is demonstrated to provide better compression ratios in Wallace Tree structure in the present study.

The proposed concept is based on XOR-XNOR logic and implemented using a multiplexer instead of a cascaded XOR operation. The output of the XOR and XNOR is reused in order to eliminate unnecessary logic transitions and to ease the internal signal propagation. Multiplexers are also used for reduction, to optimise the switching operations.

The improved 4:2 compressor architecture that is proposed is seen in Fig. 3.

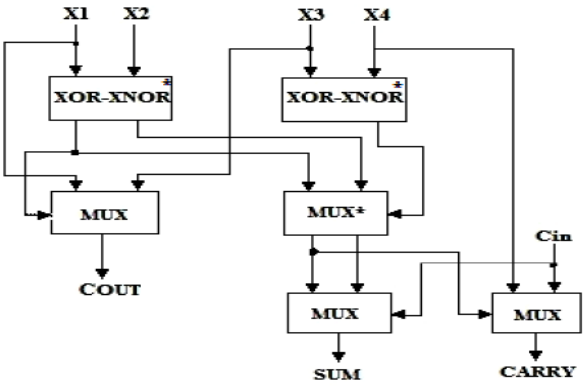

Fig.3 Proposed modified 4:2 Compressor using XOR-XNOR

The suggested compressor decreases the internal transition activity and the propagation path, hence the compression efficiency with maintaining the mathematical accuracy.

C) Proposed 5:2 Compressor Design

A modified 5:2 compressor is employed in the proposed multiplier architecture in order to further enhance the partial product reduction. The compressor receives five primary

inputs and carry inputs and compresses all these inputs to fewer outputs and at reduced computational cost.

The design uses the XOR-XNOR logic along with multiplexer-aided implementation to optimise the internal signal propagation as seen in the proposed 4:2 compressor. The architecture reduces the dependency on the logic by eliminating unnecessary inverter stages and makes good use of intermediate outputs.

The 5:2 compressor proposed is shown in Fig. 4.

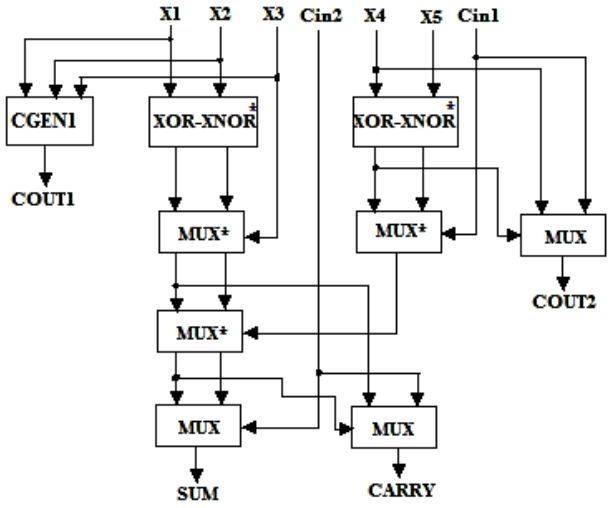

Fig.4 5-2 Compressor block with XOR-XNOR

The Wallace Tree multiplier provides an optimized reduction scheme for reducing switching activity and efficiency of the partial product compression.

D) Proposed 8×8 Multiplier Architecture

The final architecture of the multiplier is a 8×8 Wallace Tree with the enhanced compressor designs. The partial products are the outputs of the parallel ANDs of the inputs to the multiplier and multiplicand.

The resulting partial products are subjected to a number of Wallace reduction stages with modified 4/2 and 5/2 compressors. These stages combine intermediate rows before adding it up at the end and output the multiplication result.

The entire proposed 8×8 Wallace Tree multiplier design is depicted in Fig. 5.

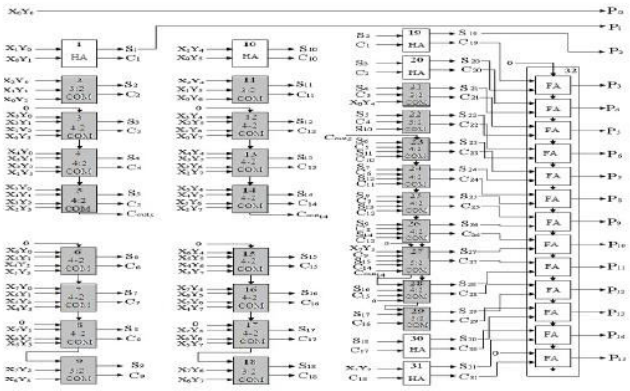

Fig.5 Proposed 8x8 Wallace Tre Multiplier using compressor

The proposed architecture is based on the idea of designing an efficient architecture by optimizing the compressor and parallelizing the reduction process to minimize the critical path latency while simultaneously improving the hardware efficiency without adding much to the complexity of the design.

E) Verilog Implementation Flow

The proposed multiplier architecture is designed using Verilog HDL using the modular design technique. The individual modules for the improved 4:2 compressor, 5:2 compressor and Wallace reduction stages were created individually and merged into a top-level multiplier architecture.

The implementation process was broken down into four phases: Design Development, functional simulation, synthesis and output verification. Functional simulation of the multiplication operation is checked for different inputs. So, the design is synthesised and analysed in Xilinx Vivado 2024.2. The implementation cycle provides a step-by-step process for the realisation of hardware and the extension of higher order multiplier systems in the future.

IV. Experimental Results & Discussion

This section covers the implementation environment, the functional verification process and the functional discussion of the proposed multiplier architecture. The aim of the experimental evaluation is to ensure that the enhanced compressor-based Wallace Tree multiplier works and to investigate the potential effect on the computational efficiency. The design of the proposed architecture has been done with Verilog HDL and validated using Xilinx Vivado 2024.2.

A) Experimental Setup

The suggested multiplier architecture was designed and evaluated utilising the hardware description and simulation based approach. All of the design is based on the modified 4:2 and 5:2 compressor modules in an 8×8 Wallace Tree multiplier architecture. Functional verification was carried out to prove that partial products generation, reduction and accumulation are correct.

A few key stages in implementing the process were:

- Verilog module development
- Functional simulation
- Design synthesis
- Output verification

Connectivity of compressor blocks and the compressor stages was verified by simulation and synthesis.

B) Functional Verification

The logical behaviour of the proposed multiplier architecture is verified during multiplication operation to conduct functional verification. The simulation test cases verified the correct creation and reduction of partial products of the upgraded compressor stages.

The simulation results confirm the successful incorporation of the XOR-XNOR based compressor blocks in the Wallace reduction network. The sequence of outputs

produced by the multiplier was the same as expected, starting from the partial products up to the final accumulator result, indicating correct operation.

The functional simulation wave form for the implementation is shown in Fig.6.

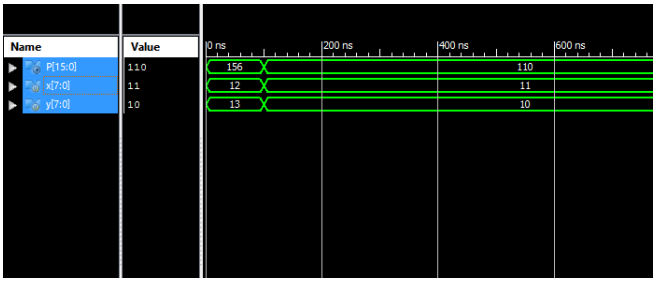

Fig.6 Functional Simulation Output Waveform

C) Architecture Discussion

With the proposed multiplier architecture, the optimisation is performed at the compressor level, but the Wallace Tree multiplication structure is still maintained. The reduction stages include the use of modified 4:2 and 5:2 compressors to increase the partial product compression.

To simplify the internal signal flow, and to reduce the dependency of the logic in the reduction operations, the architecture is designed using XOR-XNOR logic and multiplexer-assisted implementation. Intermediate outputs are re-used and therefore, the reduction structure is more compact.

The proposed Multiplier design is synthesized into Hardware and shown in Figure 7.

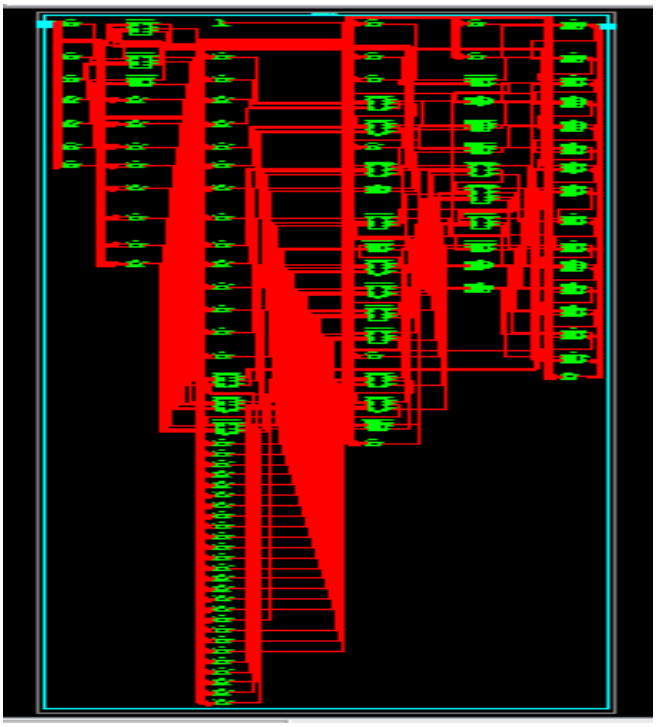

Fig.7 RTL Schematic

The RTL implementation confirms correct association of the re-designed compressor blocks of Wallace Tree architecture and shows the structural realisation of the proposed design.

D) Power–Delay Analysis

Existing implementation lacks quantitative power and timing data in detail. So the performance analysis is done on the basis of architectural analysis of proposed compressor based multiplier design.

The anticipated rise in the efficiency of the multipliers is due to:

- To minimize switching activity, XOR-XNOR logic is reused.
- A smaller critical path due to multiplexer based implementation.
- Partial product reduction and reduced logic dependencies
- New compressor stage geometry for better compression efficiency
- In realisation, the hardware components are kept under control in complexity.

These optimizations of the architecture demonstrate the applicability of the proposed architecture for low power and high speed multiplier applications.

E) Summary of Observations

The following observations are made during implementation and functional verification:

- Used modified 4:2 and 5:2 compressor architectures in Verilog HDL to achieve success.
- Proper functionality of the proposed 8×8 Wallace Tree multiplier
- A logic using XOR and XNOR can be implemented effectively using multiplexer based logic.
- Improved reduction scheme with reduced processing of intermediates
- The usefulness of the architecture in designing efficient VLSI multipliers is discussed.

The experimental discussion shows that the efficiency of the implementation of the multiplier can be improved by optimising the level of the compressor structure with a small size of the hardware embedded.

V. Conclusion

Design and Implementation of 8 * 8 Wallace Tree Multiplier using Modified 4:2 & 5:2 Compressor Architectures with Low Power and High Speed. Regarding the Wallace reduction stage, the proposed approach enhanced the partial product reduction by the integration of the XOR-XNOR logic and multiplexer assisted implementation. To reduce reliance on logic, optimize for reduction and performance, an optimisation was carried out at the compressor level rather than changing the multiplication structure.

The entire design has been coded in Verilog HDL and functionally simulated using Xilinx Vivado 2024.2. The functional study showed a successful integration of the re-designed compressor structures in the multiplier architecture while maintaining the correct functional behaviour. The suggested method shows that the optimisation at compressor level can lead to a compact and efficient design methodology for the implementation of multipliers. So, the proposed

architecture is suitable for the current VLSI system, which is used in arithmetic applications with low power and high speed applications.

The proposed architecture is extendable to higher order multiplier designs like 16x16, 32x32 etc. which can be used in complicated arithmetic applications. Synthesis reports, timing analysis, power estimation and hardware resource utilisation can be used in the future to evaluate the quantitative. Some other optimisation techniques such as pipelining, the use of different compressor architectures, and FPGA or ASIC implementation could also be explored as tools to show the viability and applicability of the proposed concept in high-performance VLSI systems.

REFERENCES

[1] S. Veeramachaneni, K. Krishna, L. Avinash, S. R. Puppala, and M. B. Srinivas, "Novel architectures for high-speed and low-power 3-2, 4-2 and 5-2 compressors," in Proc. IEEE Int. Conf. VLSI Design (VLSID), 2007, pp. 324–329.

[2] C. H. Chang, J. Gu, and M. Zhang, "Ultra low-voltage low-power CMOS 4-2 and 5-2 compressors for fast arithmetic circuits," IEEE Trans. Circuits Syst. I, vol. 51, no. 10, pp. 1985–1997, Oct. 2004.

[3] S. F. Hsiao, M. R. Jiang, and J. S. Yeh, "Design of high speed low-power 3-2 counter and 4-2 compressor for fast multipliers," Electron. Lett., vol. 34, no. 4, pp. 341–343, 1998.

[4] H. T. Bui, Y. Wang, and Y. Jiang, "Design and analysis of low-power 10-transistor full adders using novel XOR–XNOR gates," IEEE Trans. Circuits Syst. II, vol. 49, no. 1, pp. 25–30, Jan. 2002.

[5] R. Zimmermann and W. Fichtner, "Low-power logic styles: CMOS versus pass-transistor logic," IEEE J. Solid-State Circuits, vol. 32, no. 7, pp. 1079–1090, Jul. 1997.

[6] N. Ahmad and R. Hasan, "A new design of XOR–XNOR gates for low power application," in Proc. IEEE Int. Conf. Electronic Design, Systems and Applications (ICEDSA), 2011.

[7] H. T. Bui, A. K. Al-Sheraidah, and Y. Wang, "New 4-transistor XOR and XNOR designs," in Proc. IEEE Asia Pacific Conf. ASICs, 2000, pp. 25–28.

[8] J. M. Wang, S. C. Fang, and W. S. Feng, "New efficient designs for XOR and XNOR functions on the transistor level," IEEE J. Solid-State Circuits, vol. 29, no. 7, pp. 780–786, Jul. 1994.

[9] N. Itoh, Y. Naemura, H. Makino, Y. Nakase, T. Yoshihara, and Y. Horiba, "A 600-MHz 54 × 54-bit multiplier with rectangular-styled Wallace tree," IEEE J. Solid-State Circuits, vol. 36, no. 2, pp. 249–257, Feb. 2001.

[10] K. Roy and S. C. Prasad, Low Power CMOS VLSI Circuit Design. New York, NY, USA: Wiley, 2000.

[11] C. S. Wallace, "A suggestion for a fast multiplier," IEEE Trans. Electron. Comput., vol. EC-13, no. 1, pp. 14–17, Feb. 1964.

[12] L. Dadda, "Some schemes for parallel multipliers," Alta Frequenza, vol. 34, no. 5, pp. 349–356, 1965.

[13] M. D. Ercegovac and T. Lang, Digital Arithmetic. San Francisco, CA, USA: Morgan Kaufmann, 2004.

[14] N. H. E. Weste and D. Harris, CMOS VLSI Design: A Circuits and Systems Perspective, 4th ed. Boston, MA, USA: Addison-Wesley, 2011.

[15] J. M. Rabaey, A. Chandrakasan, and B. Nikolic, Digital Integrated Circuits: A Design Perspective, 2nd ed. Upper Saddle River, NJ, USA: Prentice Hall, 2003.

[16] B. Parhami, Computer Arithmetic: Algorithms and Hardware Designs, 2nd ed. New York, NY, USA: Oxford University Press, 2010.

[17] A. Bellaouar and M. I. Elmasry, Low-Power Digital VLSI Design: Circuits and Systems. Boston, MA, USA: Kluwer Academic Publishers, 1995.

[18] S. Knowles, "A family of adders," in Proc. IEEE Symp. Computer Arithmetic, 2001, pp. 277–281.