Low Power High Speed Accuracy Controllable Approximate Multiplier Design

P Roja[1], SK Jafar Ameen[2]
[1]*PG Student, DSCE, Quba College of Engineering & Technology*
[2]*Assistant professor, DSCE, Quba College of Engineering & Technology*

***Abstract*— Many increasingly popular applications, such as image processing and recognition, are inherently tolerant of small inaccuracies. These applications are computationally demanding and multiplication is their fundamental arithmetic function, which creates an opportunity to trade off computational accuracy for reduced power consumption. Approximate computing is an efficient approach for error tolerant applications because it can trade off accuracy for power, and it currently plays an important role in such application domains. Different error-tolerant applications have different accuracy requirements, as do different program phases in an application. If multiplication accuracy is fixed, power will be wasted when high accuracy is not required. This means that approximate multipliers should be dynamically reconfigurable to match the different accuracy requirements of different program phases and applications. Approximate multiplication is considered to be an efficient technique for trading off energy against performance and accuracy. This work proposes an accuracy-controllable multiplier whose final product is generated by a carry-mask able adder. The proposed scheme can dynamically select the length of the carry propagation to satisfy the accuracy requirements flexibly. The partial product tree of the multiplier is approximated by the proposed tree compressor. An 8x8 multiplier design is implemented by employing the carry maskable adder and the compressor. Compared with a conventional Wallace tree multiplier, the proposed multiplier reduced power and critical path delay, depending on the required accuracy.**

***Index Terms*— Image Processing, Pattern Recognition, Machine Learning, Artificial Intelligence, Real-Time Embedded Systems**

I. INTRODUCTION

The electronic device utilization has increased in day-to-day life. The idea of digital data computation has made a dramatic effect in our society. In recent computing platforms, the computations are performed in precise way depending on application requirement. In recent years, the design performance and efficiency of computing platforms have an exponential increase in computation. From an application perspective, all computations are not equally important. Figure 1.1 shows two tasks that deal with division between 521 and 32. In the first task, the output is compared to 16.1, and in the second task, it is compared to 10. For human brain, it is simpler and easier to understand the second case than first, compared to computers. In present scenario, computers work harder, and produce efficient results for many applications with trade-off in area, power and delay.

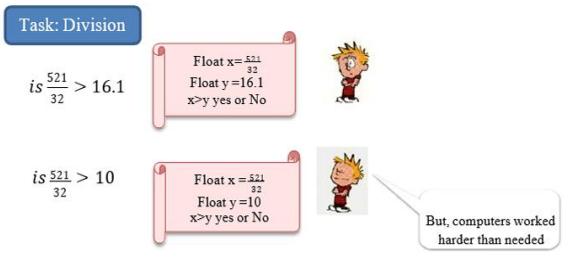

Figure 1.1 Computational Inefficiency in Computers

In recent research, inherent approximation is well-defined as the property of an application produces an acceptable output in spite of its fundamental precise computations.

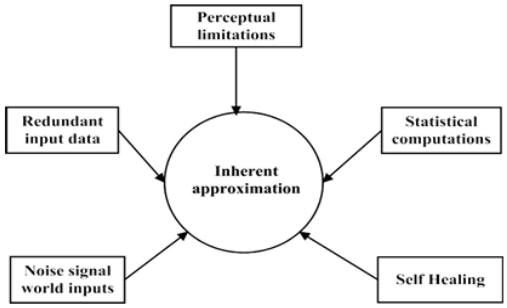

Figure 1.2 Sources of Inherent Approximation

1.1 APPROXIMATE COMPUTING

Humans have less perceptual abilities in identifying imprecisions during the process of image or video. Algorithms and precise models are not efficient to use in these applications. This allows inaccurate computation to the existing digital logic circuits by decreasing the logic complexity and increase in performance with trade-off in accuracy. This area of research is generally known as error tolerant computing and is referred to as approximate or inaccurate computing. Approximate computing is a suitable approach to improve the performance with certain loss in accuracy. Approximate computing allows optimizing logic complexity and performance for the sake of accuracy. Thus, an error in computation has been tolerated as long as it is small enough to maintain a feasible operation of the system [4].

The emerging multimedia applications are shown in Figure 1.3. According to the International Technology Roadmap for Semiconductors (ITRS) prediction, approximate computing-based designs are well-matched for error tolerant multimedia applications.

In many images processing, video processing, Digital Signal Processing (DSP) and multi media systems, inputs from the real world are noisy, so there is no strict correctness in these systems. The clustering and recognition algorithms used in processing data are based on statistical/probabilistic computation. Due to the statistical/probabilistic nature, small errors in the computation will not impose

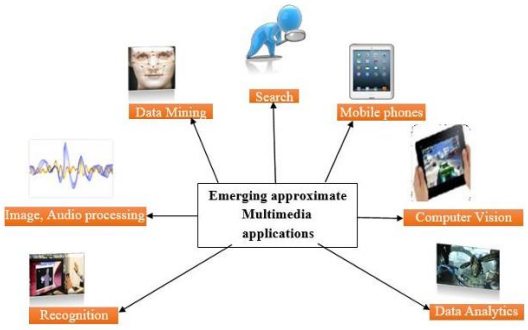

Figure 1.3 Emerging Approximate Multimedia Applications

1.2 DEVELOPMENTS IN APPROXIMATE COMPUTING

The growth in demands on computing platforms is expected to continue unabated. The workloads on computing platforms experience heavy processing of data to obtain a golden output. The best-known algorithms or processing elements fall short of perfection. Hence, functional correctness is redefined for obtaining precise output that is good enough to the end user. Figure 1.4 illustrates the growing challenges on computing platforms where the functional correctness is redefined for acceptable output and defines an approximation on computing platforms. Approximate computing is a promising technique for achieving energy efficiency.

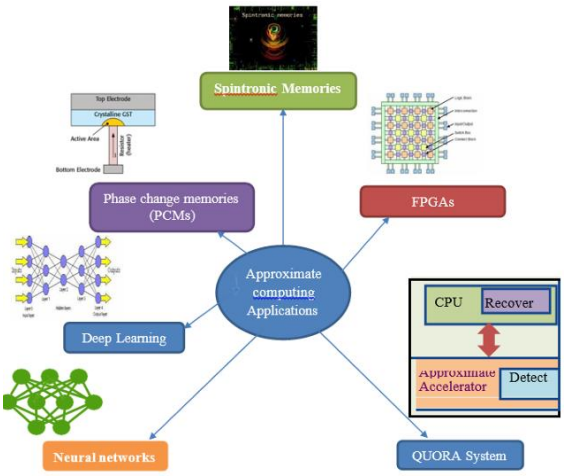

Figure 1.4 Approximate Computing Applications

II .LITERATURE SURVEY

According to reference, multipliers are essential arithmetic operations that play a critical role in various fault algorithms. The utilization of approximate multipliers is commonly acknowledged as a superior approach for integrating power, speed, and accuracy. The authors present two methodologies for approximating multipliers with reduced power consumption and latency compared to conventional multiplication techniques, employing approximate tree compressors. The compressor that is recommended has the capability to reduce the size of the partial product trees by 50%, and also provides vectors that enable the restoration of precision. In comparison to traditional Wallace tree multiplication, an 8-bit approximate multiplier that is suggested exhibits a reduction in power consumption and critical path latency by approximately 59.9% and 36.3%, respectively. Furthermore, given a standardized mean error of 0.28%, the total chip size required for the design of the multiplier is decreased by 50.1%. The proposed multiplier topologies exhibit superior performance compared to their predecessors in terms of energy efficiency, critical path latency, and overall

layout area. The utilisation of approximate circuitry, as cited in reference, has been implemented in error-tolerant circuits that can accommodate imprecision to enhance their functionality. Multipliers are critical analytical circuits utilized in various applications. The article analyzed a novel approximation multiplier designed for DSP applications, which exhibits reduced power consumption and a shorter critical path compared to conventional multipliers.

The multiplier utilises a redesigned approximation adder that restricts carry propagation to nearest neighbors, resulting in efficient partial product generation. Tailored error correction techniques can achieve varying degrees of precision by mitigating errors in distinct numbers of the most significant bits. The majority of errors are small in size, as the approximation multiplier has a low MED. The utilisation of a 28nm CMOS process in the design of a 16-bit approximation multiplication results in a reduction of 20% in time and up to 69% in power consumption when compared to the Wallace multiplication technique. The study shows that the proposed approximate multiplier yields comparable processing efficiency to traditional precise multipliers, but with notable enhancements in power and performance.

III. EXISTING MULTIPLIERS

The operation of multiplication holds significant importance in modern electronic circuits. Multiplication-based algorithms are commonly utilized in various implementations of Digital Signal Processing (DSP). Digital multipliers are essential elements in high-performance devices such as microcontrollers, digital signal processors, and FIR filters.

The multiplier component, which is known to be the most area-intensive and slowest, is often utilized as a metric for evaluating the efficacy of the system. Hence, augmenting the multiplier's efficacy and size would pose a significant limitation in the design process. The utilization of the multiplier has served as a fundamental structure in the development of a processor that is optimized for energy efficiency.

3.1 BASIC MULTIPLIER CONCEPTS

Multiplication is a fundamental activity in many signal-processing algorithms. Multipliers have a large size, a significant latency, and a high-power consumption. As a result, the implementation of low-power VLSI systems necessitates the design of appropriate low-power multipliers. A basic multiplier is made up of three components, as shown in Figure 3.1:

1. Production of a partial product
2. Partial product addition
3. Final Product Addition

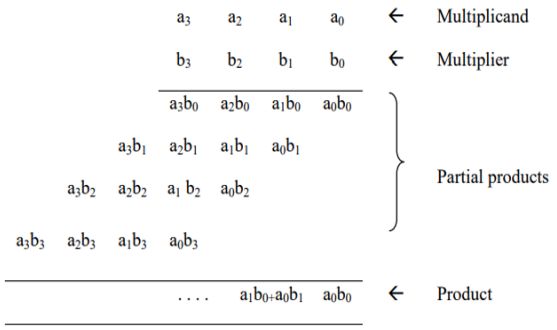

Figure 3.1 Architecture of conventional Multiplier

3.2 CONVENTIONAL MULTIPLIERS IN VLSI

In every digital circuit, multiplier uses the majority of the power and it will generate delays if the properly optimized multiplier is not employed. There are various multipliers, and every multiplier has a unique algorithm and architecture. All multiplier's performance characteristics are unique, but each one may be further adjusted to get superior performance characteristics. Various researchers have invented and refined different sorts of multipliers. Various traditional multipliers are addressed in this section.

IV. PROPOSED ACCURACY-CONTROLLABLE APPROXIMATE MULTIPLIER DESIGN

This work focuses on an approximate multiplier design that can control accuracy dynamically. A carry-maskable adder (CMA) is proposed that can be dynamically configured to function as a conventional carry propagation adder (CPA), a set of bit-parallel OR gates, or a combination of the two. This configurability is realized by masking carry propagation: the CPA in the last stage of the multiplier is replaced by the proposed CMA.

4.1 Approximate Tree Compressor Figure 4.1(a) shows an accurate half adder, for which the following equation can be obtained:

$$\{c, s\} = a + b = 2c + s = (c + s) + c,$$

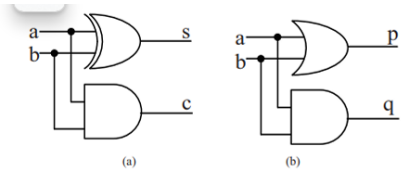

Figure 4.1. (a) Accurate half adder and (b) incomplete adder cell.

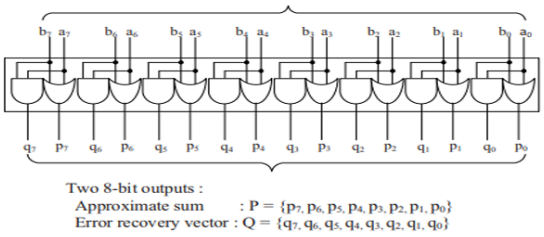

Figure 4.2: A row of incomplete adder cells with two 8-bit inputs.

V. RESULTS AND DISCUSSIONS

The various 8 X 8 multipliers outlined above have been synthesized and simulated using an FPGA board utilizing the Xilinx Spartan 7 XC7S15-1FTGB196C architecture. The inputs of the multiplier are connected to the input switches of the FPGA and outputs of the multiplier are connected to the LEDs present on the FPGA. For the implementation of the existing and proposed multipliers on to FPGA board, Xilinx ISE is used. By varying the switches on the FPGA, the corresponding LEDs are verified. However, this technique raises the delay time greater than the existing methods. In this case, simulation is performed for multipliers with an 8-bit width.

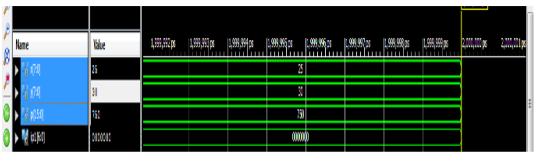

Figure 5.1 Simulation results of 8X8 Array Multiplier

A multiplication using the array multiplier with inputs X and Y as 8 bits each is depicted in Figure 5.1. The output result of the multiplication process is represented by P.

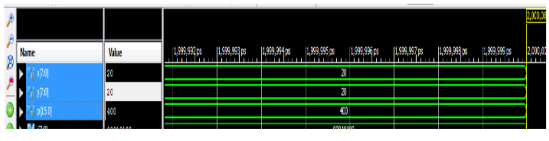

Figure 5.2 Simulation results of 8X8 Booth Multiplier

A multiplication using the Booth multiplier with inputs X and Y as 8 bits each is depicted in Figure 5.3. The output result of the multiplication process is represented by P.

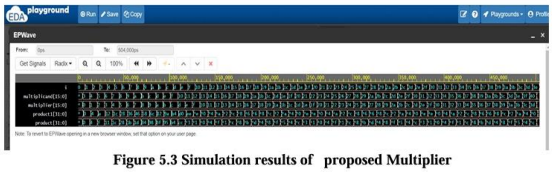

Figure 5.3 Simulation results of proposed Multiplier

A multiplication using the proposed multiplier with inputs X and Y as 16bits each is depicted in Figure 5.3. The output result of the multiplication process is represented by P. The multiplier implementations are synthesized using Xilinx vivado/ISE, and also the resulting delay & area parameters are shown in Table 5.1. Table 5.1 clearly show that the proposed Multiplier occupies less area than the other designs.

Table 5.1 Performance analysis of various multipliers

Multiplier Name	Area	Power(mw)
Existing Wallace tree Booth Multiplier	877	0.693
Existing Booth Approximate Multiplier	350	0.311
Proposed Approximate Multiplier	209	0.042

VI. CONCLUSION AND FUTURE SCOPE

An accuracy-controllable approximate multiplier has been proposed in this project that consumes less power and has a shorter critical path delay than the conventional design. Its dynamic controllability is realized by the proposed CMA. The multiplier was evaluated at both the circuit and application levels. The experimental results demonstrate that the proposed multiplier was able to deliver significant power savings and speedups while maintaining a significantly smaller circuit area than that of the conventional Wallace tree multiplier. Furthermore, for the same accuracy, the proposed multiplier delivered greater improvements in both power consumption and critical path delay than other previously studied approximate multipliers. Finally, the ability of our proposed multiplier to control accuracy was confirmed by an application-level evaluation. Furthermore, when the speed of technological advancement quickens, new inventive solutions and design elements may be created to fulfil technological scalability standards. Furthermore, as CMOS technology progresses, researchers may concentrate on CMOS IC design methodologies with a high degree of assurance and

power efficiency for wireless healthcare and medical application sectors. There is space to create building techniques at both the circuit block and system levels throughout the whole foundation of technical breakthroughs. Approaches for designing ultra-low-power CMOS circuits for the full network impact of wireless health and medical care that are strategically positioned at low-technological nodes.

REFERENCE

[1] J. M Rabaey, A. Chandrakasan, B, Nikolic, and Digital Integrated Circuits: A Design Perspective, Upper saddles rive, NJ: 2e Prentice-Hall, 2002.
[2] S. M. Kang, Y. Leblebici, CMOS Digital Integrated Circuits: Analysis & Design, TATA McGraw- Hill Publication, 3e, 2003.
[3] N. Weste, K. Eshraghian, Principles of CMOS VLSI Design, A systems perspective, Addison Wesley MA, 1988.
[4] J. P. Uyemura, CMOS logic circuit design, Kluwer Academic, 2002.
[5] K.S. Yeo, K. Roy, Low- Voltage, Low-Power VLSI Subsystems, McGraw Hill Professional, 2005.
[6] K. Roy, S. Prasad, Low-Power CMOS VLSI Circuit design, Willey Inter science Publication, 2000.
[7] K. K. Kim, Y. -B. Kim, M. Choi, N. Park, "Leakage minimization technique for nanoscale CMOS VLSI based on macro-cell modelling," in IEEE Design &Test of Computers, Dec. 13, 2006.
[8] I. -C. Wey, Y.-G. Chen, A.-Y. Wu, "Design and Analysis of Isolated Noise Tolerant (INT) Technique in Dynamic CMOS Circuits," IEEE Transactions on Very Large-Scale Integration (VLSI) Systems, vol. 16, no. 12, pp. pp.1708-1712, Dec. 2008.
[9] P. Meher, K. K. Mahapatra, "Ultra-low-power and noise tolerant CMOS dynamic circuit technique," in IEEE Region 10 Conference TENCON 2011, pp.1175-1179, Bali, 21-24 Nov. 2011.
[10] R. G. Dreslinski, M. Wieckowski, D. Blaauw, D. Sylvester, and T. Mudge, "Nearthreshold Computing: Reclaiming Moore's Law through Energy Efficient Integrated Circuits," Proceedings of the IEEE, vol. 98, no. 2, pp. 253–266, 2010.
[11] J. Wu, Y.-L. Shen, K. Reinhardt, H. Szu, and B. Dong, "A Nanotechnology Enhancement to Moore's Law," Applied Computational Intelligence and Soft Computing, vol. 2, pp. 2 15, 2013.
[12] J. Sorber, A. Kostadinov, M. Garber, M. Brennan, M. Corner, and E. Berger, "Eon: A language and runtime system for perpetual systems," in Proc. 5th Int. Conf. Embedded Networked Sensor Syst. (SenSys), 2007, pp. 161–174.