Review of Approximate Multiplier for High Speed Digital Signal Processing Application

[1]Rashmi Malve, [2]Prashant Chaturvedi

[1]Research Scholar, [2]Assistant Professor,
[1&2]Department of Electronics & Communication Engineering,
[1&2]Lakshmi Narain College of Technology, Bhopal, India.

Abstract : Digital multiplier plays key role to compute and give fast response of input data. Approximate multiplier is best suited for error resilient applications, such as signal processing and multimedia. Approximate computing reduces accuracy, but it still provides meaningful and faster results with usually lower power consumption; this is particularly attractive for arithmetic circuits. Research are continue going on various existing multipliers for enhancing in terms of performance improvement like high speed, low delay, low area, low power etc. In this paper, reviews of various approximate multiplier design, implementation and achieved results of previous works.

***IndexTerms* - Approximate, Multipliers, Delay, Power, Area, Speed.**

I. INTRODUCTION

Arithmetic units such as adders and multipliers are key components in a logic circuit. The speed and power consumption of arithmetic circuits significantly influence the performance of a processor. High-performance arithmetic circuits such as carry look ahead adders (CLAs) and Wallace tree multipliers have been widely utilized. However, traditional arithmetic circuits that perform exact operations are encountering difficulties in performance improvement. Approximate arithmetic that allows a loss of accuracy can reduce the critical path delay of a circuit. Since most approximate designs leverage simplified logic, they tend to have a reduced power consumption and area overhead. Thus, approximate arithmetic is advocated as an approach to improve the speed, area and power efficiency of a processor due to the error-resilience of some algorithms and applications [1]. As an important arithmetic module, the multiplier has been redesigned to many approximate versions. The often conflicting advantages and disadvantages of these designs make it difficult to select the most suitable approximate multiplier for a specific application. Thus, approximately redesigned multipliers are reviewed in this paper and a comparative evaluation is performed by considering both the error and circuit characteristics.

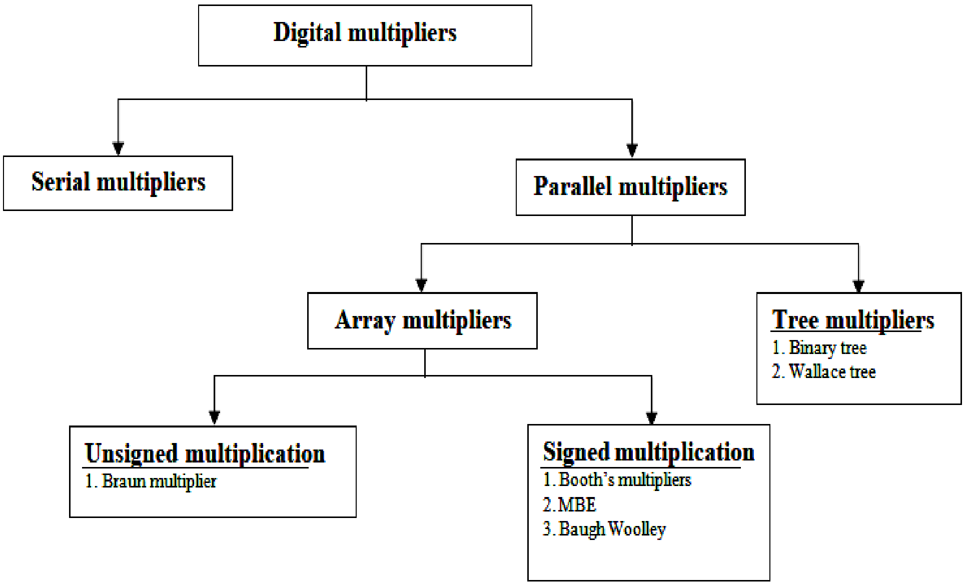

Figure 1: Types of Multiplier

Approximate computing has emerged as a potential solution for the design of energy-efficient digital systems [1]. Applications such as multimedia, recognition and data mining are inherently error-tolerant and do not require a perfect accuracy in computation. For these applications, approximate circuits may play an important role as a promising alternative for reducing area, power and delay in digital systems that can tolerate some loss of accuracy, thereby achieving better performance in energy efficiency. As one of the key components in arithmetic circuits, adders have been extensively studied for approximate implementation.

II. RELATED WORK

P. J. Edavoor, et al.,[2019] presents the designs which are implemented using 45 nm CMOS technology and efficiency of the proposed designs have been extensively verified and projected on scales of area, delay, power, Power Delay Product (PDP), Error Rate (ER), Error Distance (ED), and Accurate Output Count (AOC). The proposed approximate 4:2 compressor shows 56.80% reduction in area, 57.20% reduction in power, and 73.30% reduction in delay compared to an accurate 4:2 compressor. The proposed compressors are utilised to implement 8 × 8 and 16 × 16 Dadda multipliers. These multipliers have comparable accuracy when compared with state-of-the-art approximate multipliers. The analysis is further extended to project the application of the proposed design in error resilient applications like image smoothing and multiplication. [1]

F. Sabetzadeh, et al.,[2019] Approximate computing is an emerging approach for reducing the energy consumption and design complexity in many applications where accuracy is not a crucial necessity. In this study, ultra-efficient imprecise 4:2 compressor and multiplier circuits as the building blocks of the approximate computing systems are proposed. The proposed compressor uses only one majority gate which is different from the conventional design methods using AND - OR and XOR logics. Furthermore, the majority gate is the fundamental logic block in many of the emerging majority-friendly nanotechnologies such as quantum-dot cellular automata (QCA) and single-electron transistor (SET). The proposed circuits are designed using FinFET as a current industrial technology and are simulated with HSPICE at 7nm technology node. The results indicate that our imprecise compressor is superior to its previous counterparts in terms of delay, power consumption, power delay product (PDP) and area, and improves these parameters on average by 32%, 68%, 78%, and 66%, respectively. [2]

H. Saadat, et al.,[2018] Approximate multipliers enable the saving of area and power for implementation of many modern error-resilient compute-intensive applications. In this work, it is first propose a novel error-configurable minimally biased approximate integer multiplier MBM design. The proposed MBM design is devised by coupling a unique error-reduction mechanism with an approximate log based integer multiplier. Next, it is propose an optimization (by removing leading-one detection and barrel shifting logic) of the MBM and a class of state-of-the-art approximate integer multipliers DRUM and SSM, so that they can be efficiently used in approximate floating-point (FP) multipliers.[3]

W. Liu, et al.,[2017] Approximate computing is an attractive design methodology to achieve low power, high performance (low delay) and reduced circuit complexity by relaxing the requirement of accuracy. In this work, approximate Booth multipliers are designed based on approximate radix-4 modified Booth encoding (MBE) algorithms and a regular partial product array that employs an approximate Wallace tree. Two approximate Booth encoders are proposed and analyzed for error-tolerant computing. The error characteristics are analyzed with respect to the so-called approximation factor that is related to the inexact bit width of the Booth multipliers. Simulation results at 45 nm feature size in CMOS for delay, area and power consumption are also provided. The results show that the proposed 16-bit approximate radix-4 Booth multipliers with approximate factors of 12 and 14 are more accurate than existing approximate Booth multipliers with moderate power consumption. [4]

A. Mokhtari, et al.,[2016] This work considers decentralized consensus optimization problems where nodes of a network have access to different summands of a global objective function. Nodes cooperate to minimize the global objective by exchanging information with neighbors only. A decentralized version of the alternating directions method of multipliers (DADMM) is a common method for solving this category of problems. DADMM exhibits linear convergence rate to the optimal objective for strongly convex functions but its implementation requires solving a convex optimization problem at each iteration. This can be computationally costly and may result in large overall convergence times. The decentralized quadratic ally approximated ADMM algorithm (DQM), which minimizes a quadratic approximation of the objective function that DADMM minimizes at each iteration, is proposed here. [5]

B. Shao et al.,[2015] it is propose a general model for array-based approximate arithmetic computing (AAAC) to guide the minimization of processing error. As part of this model, the Error Compensation Unit (ECU) is identified as a key building block for a wide range of AAAC circuits. it is develop theoretical analysis geared towards addressing two critical design problems of the ECU, namely, determination of optimal error compensation values and identification of the optimal error compensation scheme. it is demonstrate how this general AAAC model can be leveraged to derive practical design insights that lead to optimal tradeoffs between accuracy, energy dissipation and area overhead.[6]

Z. Zhi-Qing Lü et al.,[2014]A non-conforming finite element tearing and interconnecting method is proposed for the numerical analysis of large-scale three-dimensional electromagnetic problems. The whole computational domain is partitioned into many smaller subdomains, and a Robin-type transmission condition is introduced at the shared interfaces to exchange data among subdomains. A set of orthogonal polynomials is introduced to approximate auxiliary unknowns between adjacent subdomains. Then, a one-Lagrange multiplier scheme is applied to deal with non-conforming meshes at the interfaces. With the help of the Schur complement approach, the method formulates a reduced-order interface problem, which can be solved using an iterative algorithm. Once the resulting interface problem is solved, the unknown electric field in each subdomain can be calculated in parallel. Numerical examples are given to demonstrate the efficiency of the method. [7]

D. De Caro, et al.,[2013] Fixed-width multipliers have two n-bits operands and produce an approximate n-bits results for their product. These multipliers discard part of the partial products matrix, to reduce hardware cost, and employ extra correction functions to reduce approximation error. While previous papers mainly focus on average error metrics (like mean-square error), it is present an in-depth analysis of the maximum absolute error (MAE) of these circuits. The MAE is the main parameter to be considered in important applications, like function evaluation. it is describe an efficient numerical method to compute the MAE in fixed-width multipliers and fixed-width multiplier-accumulator (MAC) circuits. Further it is present a technique to compute a compensation function that can be efficiently implemented in hardware, aimed to minimize the MAE. [8]

F. Auger et al.,[2011] Many signal processing algorithms require the computation of the ratio of two numbers, the square root of a number, or a logarithm. These operations are difficult when using fixed point hardware that lack dedicated multipliers, such as low-cost microcontrollers, application specific integrated circuits (ASICs), and field programmable gate arrays (FPGAs). This article presents straightforward, multiplier free algorithms that implement both division and square roots, based on a technique known as dichotomous coordinate descent (DCD). [9]

I. Park et al.,[2009] presents **s**quare and square-root are widely used in digital signal processing and digital communication algorithms, and their efficient realizations are commonly required to reduce the hardware complexity. In the implementation point of view, approximate realizations are often desired if they do not degrade performance significantly. In this paper, we propose new linear approximations for the square and square-root functions. The traditional linear approximations need multipliers to calculate slope offsets and tables to store initial offset values and slope values, whereas the proposed approximations exploit the inherent properties of square-related functions to linearly interpolate with only simple operations, such as shift, concatenation and addition, which are usually supported in modern VLSI systems. [10]

M. Rentzsch et al.,[2008] A closed analytical model of an asymmetrically switched class D converter with series-parallel-resonant (LCC) tank and three-stage Walton Cockroft multiplier featuring output voltage adjustable from zero to 20 kV and a maximum output power of 800 W is presented. The converter circuit is briefly described. A model for the dynamic behaviour of the Walton Cockroft multiplier is developed via state space modelling in the discrete time domain, which then allows it to be approximated as a low-pass filter with parameters that are a function of the converter operating point. The analytical model of the converter is based on the extended describing function and the generalised averaging technique. [11]

J. W. Hauser et al.,[2006] This work addresses the problem of efficiently approximating a function for systems-on-a-chip and other FPGA applications, hi these systems, high speed, minimal chip size, and efficient computation are necessary. Examples of common functions requiring computation include trigonometric functions and other nonlinear functions such as computing temperature using a thermistor. The specific runtime algorithm implementation used to evaluate the set of 3rd degree polynomials is directly dependent on the hardware available and the tradeoffs are discussed. Specifically, we present an efficient multiplier-less method of evaluating the 3rd degree polynomials based on logarithms targeted for FPGA applications. [12]

Table 1: Summary of literature survey

S.No	Author Name	Publish Detail	Proposed Work	Outcome
1	P. J. Edavoor	IEEE, 2020	Approximate Multiplier Design Using Novel Dual-Stage	The proposed 4: 2 compressor shows 56.80% reduction in area, 57.20% reduction in power, and 73.30% reduction in delay.
2	F. Sabetzadeh	IEEE, 2019	Ultra efficient imprecise 4:2 compressor approximate computing systems are proposed.	Delay, power consumption, PDP and area, and improve these parameters on average by 32%, 68%, 78%, and 66%, respectively.
3	H. Saadat	IEEE, 2018	It is first propose a novel error-configurable minimally biased Approximate multiplier	Significant power and area reduction with minimal degradation
4	W. Liu,	IEEE, 2017	Approximate Booth multipliers are designed based on approximate radix-4 MBE algorithms	More accurate with moderate power consumption.
5	A. Mokhtari	IEEE, 2016	This work considers decentralized consensus optimization problems.	Numerical results demonstrate advantages of DQM relative to DADMM and other alternatives in a logistic regression problem.
6	B. Shao	IEEE, 2015	A general model for array-based approximate arithmetic computing to guide the minimization of processing error.	The most accurate reported approximate Booth multiplier and outperforms the same design significantly by 19.10%
7	Z. Zhi-Qing Lü	IEEE, 2014	A set of orthogonal polynomials is introduced to approximate auxiliary unknowns between adjacent sub-domains.	Numerical examples are given to demonstrate the efficiency of the method.
8	D. De Caro	IEEE, 2013	A technique to compute a compensation function that can be efficiently implemented in hardware.	Reducing the MAE without worsening the electrical performances.

III. ADVANTAGES AND CHALLENGES

Advantages of Approximate Multipliers

- Incorrect reading is obtained when the noise signal is occurred.
- The filter is used to reduce the noise signal which also reduces the total speed of operation.
- The accuracy of the whole system is depends on accuracy of digital to analog converter and accuracy of internal reference supply.
- The speed of operation is restricted. The speed is depends on which type of switches are used.

- The conversion time required for digital to analog converter.
- This method is very systematic.
- This method takes lesser time in solving transportation problem.
- Less computation are involved in these methods.

Challenges

- Iterative channel decoders such as Turbo-Code and LDPC decoders show exceptional performance and therefore they are a part of many wireless communication receivers nowadays.
- The implementation cost of traditional soft-output de mapping methods is relatively large in high order modulation systems, and therefore low complexity de mapping algorithms are indispensable in low power receivers.
- In the presence of multiple wireless communication standards where each standard defines multiple modulation schemes, there is a need to have an efficient architecture covering all the flexibility requirements of these standards.

Existing System

Design a Wallace tree adder based approximate multiplier architecture. This architecture is to reduce the number of partial products to be added into 2 final intermediate results. Existing system is to modify the regular adder process and to optimize the partial product generator architecture circuit complexity level. Exiting system is used to optimize the critical path section and to reduce the overall tree based structure work.

A fast process for multiplication of two numbers was developed by Wallace. Using this method, a three step process is used to multiply two numbers; the bit products are formed, the bit product matrix is reduced to a two row matrix where sum of the row equals the sum of bit products, and the two resulting rows are summed with a fast adder to produce a final product.

Three bit signals are passed to a one bit full adder (“3W”) which is called a three input Wallace tree circuit and the output of sum signal is supplied to the next stage full adder of the same bit. The carry output signal is passed to the next stage full adder of the same no of bit, and the carry output signal thereof is supplied to the next stage of the full adder located at a one bit higher position. Wallace tree is a tree of carry-save adders (CSA) arranged as shown. A carry save adder consists of full adders like the more familiar ripple adders, but the carry output from each bit is brought out to form second result vector rather being than wired to the next most significant bit.

The carry vector is 'saved' to be combined with the sum later. In the Wallace tree method, the circuit layout is not easy although the speed of the operation is high since the circuit is quite irregular. Wallace tree is known for their optimal computation time, when adding multiple operands to two outputs using carry-save adders. The Wallace tree guarantees the lowest overall delay but requires the largest number of wiring tracks. The number of wiring tracks is a measure of wiring complexity. To improve speed, Wallace Tree algorithm can be used to reduce the number of sequential adding stages.

IV. CONCLUSION AND FUTURE WORK

This paper presents review of various approximate multiplier technique of previous research. Therefore it is clear that such multiplier is designed and implemented for high speed in various applications. 16-bit and 32-bit multipliers is designed and tested. In future we implement 64-bit approximate multiplier using verilog coding on Xilinx 14.7 software. Implementation will be helpful for advance digital signal applications with improved performance.

REFERENCES

[1]. P. J. Edavoor, S. Raveendran and A. D. Rahulkar, "Approximate Multiplier Design Using Novel Dual-Stage 4:2 Compressors," in IEEE Access, vol. 8, pp. 48337-48351, 2020, doi: 10.1109/ACCESS.2020.2978773.

[2]. F. Sabetzadeh, M. H. Moaiyeri and M. Ahmadinejad, "A Majority-Based Imprecise Multiplier for Ultra-Efficient Approximate Image Multiplication," in *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 66, no. 11, pp. 4200-4208, Nov. 2019.

[3]. H. Saadat, H. Bokhari and S. Parameswaran, "Minimally Biased Multipliers for Approximate Integer and Floating-Point Multiplication," in *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 37, no. 11, pp. 2623-2635, Nov. 2018.

[4]. W. Liu, L. Qian, C. Wang, H. Jiang, J. Han and F. Lombardi, "Design of Approximate Radix-4 Booth Multipliers for Error-Tolerant Computing," in *IEEE Transactions on Computers*, vol. 66, no. 8, pp. 1435-1441, 1 Aug. 2017.

[5]. A. Mokhtari, W. Shi, Q. Ling and A. Ribeiro, "DQM: Decentralized Quadratically Approximated Alternating Direction Method of Multipliers," in *IEEE Transactions on Signal Processing*, vol. 64, no. 19, pp. 5158-5173, 1 Oct.1, 2016.

[6]. B. Shao and P. Li, "Array-Based Approximate Arithmetic Computing: A General Model and Applications to Multiplier and Squarer Design," in *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 62, no. 4, pp. 1081-1090, April 2015.

[7]. Z. Zhi-Qing Lü and X. An, "Non-conforming finite element tearing and interconnecting method with one Lagrange multiplier for solving large-scale electromagnetic problems," in *IET Microwaves, Antennas & Propagation*, vol. 8, no. 10, pp. 730-735, 15 July 2014.

[8]. D. De Caro, N. Petra, A. G. M. Strollo, F. Tessitore and E. Napoli, "Fixed-Width Multipliers and Multipliers-Accumulators With Min-Max Approximation Error," in *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 60, no. 9, pp. 2375-2388, Sept. 2013.

[9]. F. Auger, Z. Lou, B. Feuvrie and F. Li, "Multiplier-Free Divide, Square Root, and Log Algorithms [DSP Tips and Tricks]," in *IEEE Signal Processing Magazine*, vol. 28, no. 4, pp. 122-126, July 2011.

[10]. I. Park and T. Kim, "Multiplier-less and table-less linear approximation for square and square-root," *2009 IEEE International Conference on Computer Design*, Lake Tahoe, CA, 2009, pp. 378-383.

[11]. M. Rentzsch, F. Gleisberg, H. Guldner, F. Benecke and C. Ditmanson, "Closed analytical model of a 20 kV output voltage, 800 W output power series-parallel-resonant converter with Walton Cockroft multiplier," *2008 IEEE Power Electronics Specialists Conference*, Rhodes, 2008, pp. 1923-1929.

[12]. J. W. Hauser, "A Multiplier-less Algorithm for Evaluating Intrinsic and Nonlinear Functions," *2006 49th IEEE International Midwest Symposium on Circuits and Systems*, San Juan, 2006, pp. 525-529.