ISSN:2348-2079

Volume-7 Issue-1

International Journal of Intellectual Advancements and Research in Engineering Computations

Implementation of Efficient Approximate Unsigned Multipliers for DSP Applications

S.Arunodhaya[1], Dr.S.Kavitha[2]

[1]PG Scholar Department of Electronics and Communication Engineering, Nandha Engineering College (Autonomous), Erode.

[2]Professor Department of Electronics and Communication Engineering, Nandha Engineering College (Autonomous), Erode.

ABSTRACT

Multipliers are key arithmetic circuits in many of these applications including digital signal processing (DSP). A novel approximate multiplier with a low power consumption and a short critical path is proposed for high-performance DSP applications. This multiplier leverages a newly designed approximate adder that limits its carry propagation to the nearest neighbors for fast partial product accumulation. Different levels of accuracy can be achieved by using either OR gates or the proposed approximate adder in a configurable error recovery circuit. The approximate multipliers using these two error reduction strategies are referred to as AM1 and AM2, respectively. Both AM1 and AM2 have a low mean error distance, i.e., most of the errors are not significant in magnitude. Image processing applications, including image sharpening and smoothing, are considered to show the quality of the approximate multipliers in error-tolerant applications. By utilizing an appropriate error recovery scheme, the proposed approximate multipliers achieve similar processing accuracy as exact multipliers, but with significant improvements in power.

Index Terms: Multiplier, Adder, Error Recovery, Low-Power, Image Processing.

INTRODUCTION

Multiplication is a fundamental operation in most signal processing algorithms. Multipliers have large area, long latency and consume considerable power. Therefore low power multiplier design has an important part in low-power VLSI system design. A system is generally determined by the performance of the multiplier because the multiplier is generally the slowest element and more area consuming in the system. Hence optimizing the speed and area of the multiplier is one of the major design issues. However, area and speed are usually conflicting constraints so that improvements in speed results in larger areas [1]. Multiplication is a mathematical operation that include process of adding an integer to itself a specified number of times. A number (multiplicand) is added itself a number of times as specified by another number (multiplier) to form a result (product). Multipliers play an important role in today's digital signal processing and various other applications. Multiplier design should offer high speed, low power consumption.

Multiplication involves mainly 3 steps

- Partial product generation
- Partial product reduction
- Final addition

Author for correspondence:
Department of Electronics and Communication Engineering, Nandha Engineering College (Autonomous), Erode.

PARTIAL PRODUCT GENERATORS (PPG)

An under-designed 16×16 multiplier using inaccurate 2 × 2 Partial Product Generators (PPG) while guaranteeing the minimum and maximum accuracy fixed at design time. Each PPG has fewer transistors compared with the accurate 2×2 one, reducing both dynamic and leakage energy at the cost of some accuracy loss novel iterative log approximate multiplier using Leading One Detectors (LODs) to support variable accuracy [2-5].

A binary multiplier is an electronic circuit used in digital electronics, such as a computer, to multiply two binary numbers. It is built using adders.

A variety of computer arithmetic techniques can be used to implement a digital multiplier. Most techniques involve computing a set of partial products, and then summing the partial products together. This process is similar to the method taught to primary schoolchildren for conducting long multiplication on base-10 integers, but has been modified here for application to a base-2 (binary) numeral system.

A binary multiplier is an electronic circuit used in digital electronics, such as a computer, to multiply two binary numbers. It is built using adders.

MULTIPLICATION ALGORITHM

- If the LSB of Multiplier is 1, then add the multiplicand into an accumulator.
- Shift the multiplier one bit to the right and multiplicand one bit to the left.
- Stop when all bits of the multiplier are zero.

For designing a multiplier circuit we should have circuitry to provide or do the following three things:

- It should be capable of identifying whether a bit 0 or 1 is.
- It should be capable of shifting left partial products.

A multiplier usually consists of three stages: partial product generation, partial product accumulation and a carry propagation adder (CPA) at the final stage. In the designed multiplier (UDM), approximate partial products are computed using inaccurate 2 × 2 multiplier blocks, while accurate adders are used in an adder tree to accumulate the approximate partial products.

A novel approximate multiplier is designed using a simple, yet fast approximate adder. This newly designed adder can process data in parallel by cutting the carry propagation chain. It has a critical path delay that is shorter than a conventional one-bit full adder. Albeit with a high error rate, this adder simultaneously computes the sum and generates an error signal; this feature is employed to reduce the error in the final result of the multiplier. In this approximate multiplier, a simple tree of the approximate adders is used for partial product accumulation and the error signals are used to compensate errors for obtaining a better accuracy. This multiplier can be configured into two designs by using OR gates and the proposed approximate adders for error reduction, referred to as approximate multiplier 1 (AM1) and approximate multiplier 2 (AM2), respectively. Different levels of error recovery can also be achieved by using a different number of MSBs for error recovery in both AM1andAM2. As per the analysis, the proposed multipliers have significantly shorter critical paths and lower power dissipation than the traditional Wallace multiplier. Functional and circuit simulations are performed to evaluate the performance of the multipliers. Image sharpening and smoothing are considered as approximate multiplication-based DSP applications. Experimental results indicate that the proposed approximate multipliers perform well in these error-tolerant applications. The proposed designs can be used as effective library cells for the synthesis of approximate circuits [6-10].

PROPOSED APPROXIMATE MULTIPLIER

The Approximate Adder

In this section, the design of a new approximate adder is presented. This adder operates on a set of pre-processed inputs. The input pre-processing (IPP) is based on the commutativity of bits with the same weights in different addends. The proposed adder can process data in parallel by

cutting the carry propagation chain. Let A and B denote the two input binary operands of an adder, S be the sum result, and E represent the error vector. Ai, Bi, Si and Ei are the ith least significant bits of A, B, S and E, respectively. A carry propagation chain starts at the ith bit when ˙ Bi = 1, ˙ Ai+1 = 1, ˙ Bi+1 = 0. In an accurate adder, Si+1 is 0 and the carry propagates to the higher bit. However, in the proposed approximate adder, Si+1 is set to 1 and an error signal is generated as Ei+1 =1. This prevents the carry signal from propagating to the higher bits [20]. Hence, a carry signal is produced only by the generate signal, i.e., Ci = 1 only when ˙ Bi = 1,and it only propagates to the next higher bit, i.e., the (i+1)th position.

Proposed Approximate Multiplier

In the proposed design, the simplification of the partial product accumulation stage is accomplished by using an adder tree, in which the number of partial products is reduced by a factor of 2 at each stage of the tree [11-15]. This adder tree is usually not implemented using accurate multi-bit adders due to the long latency. However, the proposed approximate adder is suitable for implementing an adder tree, because it is less complex than a conventional adder and has a much shorter critical path delay.

Exact fast multipliers often include a Wallace or Dadda tree using full adders (FAs) and half adders (HAs); compressors are also utilized in the Wallace or Dadda tree to further reduce the critical path with an increase in circuit area. These designs require a proper selection of different circuit modules; for example, 4:2 compressors, FAs and HAs are commonly used in a Wallace tree and a judicious connection of these modules must be considered in a tree design. This increases the design complexity, especially when multipliers of different sizes are considered; the proposed design is simple for various multiplier sizes. This increases the design complexity, especially when multipliers of different sizes are considered; the proposed design is simple for various multiplier sizes.

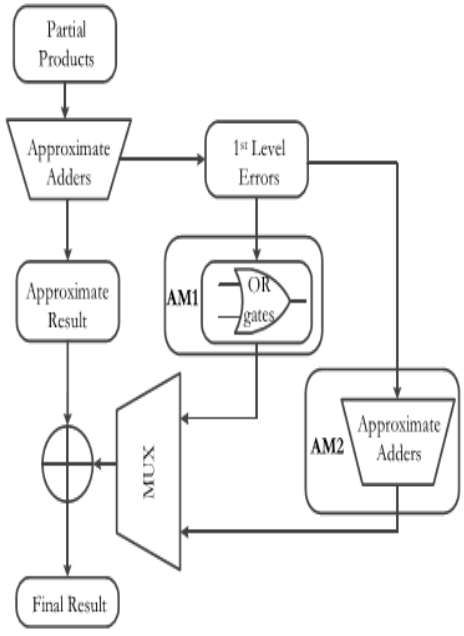

Fig.1. Block diagram of approximate multiplier

The approximate adder generates two signals: the approximate sum S and the error E; the use of the error signal is considered next to reduce the inaccuracy of the multiplier. As applicable to the sum of every single approximate adder in the tree, an error reduction circuit is applied to the final multiplication result rather than to the output of each adder. Two steps are required to reduce errors: i) error accumulation and ii) error recovery by the addition of the accumulated errors to the

adder tree output using a CPA. In the error accumulation step, error signals are accumulated to a single error vector, which is then added to the output vector of the partial product accumulation tree [16, 18]. Two approximate error accumulation methods are proposed, yielding the approximate multiplier 1 (AM1) and approximate multiplier 2 (AM2).

The n×n approximate multiplier with k -MSB OR-gate based error reduction is referred to as an n/k AM1, while an n×n approximate multiplier with k-MSB approximate adder based error reduction is referred to as an n/k AM2.

IMAGE PROCESSING APPLICATIONS

Image processing is the technique to convert an image into digital format and perform operations on it to get an enhanced image or extract some useful information from it. Changes that take place in images are usually performed automatically and rely on carefully designed algorithms. Image processing is a multidisciplinary field, with contributions from different branches of science including mathematics, physics, optical and electrical engineering. Moreover, it overlaps with other areas such as pattern recognition, machine learning, artificial intelligence and human vision research. Different steps involved in image processing include importing the image with an optical scanner or from a digital camera, analyzing and manipulating the image (data compression, image enhancement and filtering), and generating the desired output image [17].

Approximate circuits can be used in error-tolerant applications such as image processing; image sharpening and smoothing applications are studied next. Since multiplication is the arithmetic operation under investigation, accurate multipliers are replaced by the proposed approximate multipliers (i.e., AM1 and AM2). All other processing steps (such as addition) are kept accurate.

The peak signal-to-noise ratio (PSNR) is used for comparison of the difference between the images obtained by the accurate and approximate multiplications.

Fig. 2. Images multiplied by approximate multipliers

COMPARISON WITH EXISTING APPROXIMATE MULTIPLIERS

To evaluate the performance of each approximate multiplier, image multiplication is selected because it directly employs multiplication without any other operations [18, 19]. The existing Multiplier provides PSNR and SSIM values which defines the image quality. Though the PSNR value of the proposed approximate multiplier gets decreased, the SSIM value of the proposed multiplier gets increased as comparing it with the existing multiplier. It is shown in the comparison able given below.

Table I
Comaparison With Existing Multiplier

Multiplier	Existing	Proposed
PSNR	15.2271	5.7818
SSIM	0.1998	0.8923

CONCLUSION

This paper proposes a high-performance and low-power approximate partial product accumulation tree for a multiplier using a newly designed approximate adder. The proposed approximate adder ignores the carry propagation by generating both an approximate sum and an error signal. OR gate and approximate adder based error reduction schemes are utilized, yielding two different approximate 8×8 multiplier designs: AM1 and AM2. The application of the proposed multipliers to image sharpening and smoothing has shown that the proposed designs are very competitive in performance with their accurate counterpart.

REFERENCES

[1]. Honglan Jiang, Cong Liu, Fabrizio Lombardi and Jie Han, "Low-Power Approximate Unsigned Multipliers With Configurable Error Recovery," IEEE Transactions On Circuits And Systems–I: Regular Papers, 66(1), 2019.

[2]. H. Jiang, J. Han, F. Qiao, and F. Lombardi, "Approximate radix-8 booth multipliers for low-power and high performance operation,"IEEE Trans. Comput., 65(8), 2016, 2638–2644.

[3]. S. Narayanamoorthy, H. A. Moghaddam, Z. Liu, T. Park, and N. S. Kim, "Energy-efficient approximate multiplication for digital signal processing and classification applications," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., 23(6), 2015, 1180–1184.

[4]. B. Shao and P. Li, "Array-based approximate arithmetic computing: A general model and applications to multiplier and squarer design," IEEE Trans. Circuits Syst. I, Reg. Papers, 62(4), 2015, 1081–1090.

[5]. K. Bhardwaj, P. S. Mane, and J. Henkel, "Power- and area-efficient approximate wallace tree multiplier for error-resilient systems," inProc. 15th Int. Symp. Qual. Electron. Design, 2014, 263–269.

[6]. J. Miao, K. He, A. Gerstlauer, and M. Orshansky, "Modeling and synthesis of quality-energy optimal approximate adders," inProc. IEEE/ACM Int. Conf. Comput.-Aided Design,No, 2012, 728–735.

[7]. J. Han and M. Orshansky, "Approximate computing: An emerging paradigm for energy-efficient design," in Proc. 18th IEEE Eur. Test Symp., 2013, 1–6.

[8]. A. B. Kahng and S. Kang, "Accuracy-configurable adder for approximate arithmetic designs," inProc. Design Automat. Conf., 2012, 820–825.

[9]. J. Liang, J. Han, and F. Lombardi, "New metrics for the reliability of approximate and probabilistic adders," IEEE Trans. Comput., 62(9), 2012, 1760–1771.

[10]. R. Venkatesan, A. Agarwal, K. Roy, and A. Raghunathan, "MACACO: Modeling and analysis of circuits for approximate computing," in Proc. IEEE/ACM Int. Conf. Comput.-Aided Design, 2011, 667–673.

[11]. N. Zhu, W. L. Goh, and K. S. Yeo, "An enhanced low-power high-speed adder for error-tolerant application," inProc. 12th Int. Symp. Integr. Circuits, 2009, 69–72.

[12]. S.-L. Lu, "Speeding up processing with approximation circuits," Computer, 37(3), 2014, 67–73,

[13]. A. K. Verma, P. Brisk, and P. Ienne, "Variable latency speculative addition: A new paradigm for arithmetic circuit design," in Proc. Design, Automat. Test Eur., 2008, 1250–1255.

[14]. H. R. Mahdiani, A. Ahmadi, S. M. Fakhraie, and C. Lucas, "Bio-inspired imprecise computational blocks for efficient VLSI implementation of soft-computing applications," IEEE Trans. Circuits Syst. I, Reg. Papers, 57(4), 2010, 850–862.

[15]. V. Gupta, D. Mohapatra, S. P. Park, A. Raghunathan, and K. Roy, "IMPACT: IMPrecise adders for low-power approximate computing," in Proc. IEEE/ACM Int. Symp. Low Power Electron. Design, 2011, 409–414.

[16]. J. Huang, J. Lach, and G. Robins, "A methodology for energy-quality tradeoff using imprecise hardware," in Proc. Design Automat. Conf., 2012, 504–509.

[17]. Y.-H. Chen and T.-Y. Chang, "A high-accuracy adaptive conditional probability estimator for fixed-width booth multipliers," IEEE Trans. Circuits Syst. I, Reg. Papers, 59(3), 2012, 594–603.
[18]. M. A. Breuer, "Intelligible test techniques to support error-tolerance," in Proc. 13th Asian Test Symp., 2004, 386–393.
[19]. N. Weste and H. David, CMOS VLSI Design: A Circuits and Systems Perspective, 3rd ed. London, U.K.: Pearson, 2005.
[20]. M. S. K. Lau, K.-V.Ling, and Y.-C. Chu, "Energy-aware probabilistic multiplier: Design and analysis," in Proc. Int. Conf. Compil., Archit., Synthesis Embedded Syst., 2009, 281–290.