ISSN:2348-2079

Volume-7 Issue-1

International Journal of Intellectual Advancements and Research in Engineering Computations

Design of energy efficient multiplier using high radix encoding techniques

D.Ramya[1], S.T.Sadish Kumar[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

Approximate computing forms a design alternative that exploits the intrinsic error resilience of various applications and produces energy-efficient circuits with small accuracy loss. In this proposed method, an approximate hybrid high radix encoding for generating the partial products in signed multiplications that encodes the most significant bits with the accurate radix-4 encoding and the least significant bits with an approximate higher radix encoding. Required method is used for approximations of the high radix values to their nearest power of two. To achieve the desired energy–accuracy tradeoff while comparing with the accurate radix-4 multipliers, the proposed multipliers delives upto 56% energy and 55% area savings, when operating at the same frequency, while comparing with state-of-the art inexact multipliers, outperforming them by upto 40% in energy consumption.

Index term: Low Power, Radix Encoding, Multipliers, Signed multipliers.

INTRODUCTION

Multipliers are widely used in arithmetic units of microprocessors, multimedia and digital signal processors. Moreover, high performance and low power multipliers are in high demand for embedded systems. It is becoming extremely difficult to further improve performance and reduce the power consumption of multipliers under the requirement of full accuracy; however, the requirements of high precision and exactness are not so strict for many applications related to human perception, such as multimedia signal processing and machine learning. High precision and exactness in the operations of digital logic circuits are related to the generally accepted requirement of correctness of information processing; numerous error-tolerant applications can be found in computing and by relaxing the requirement of strict accuracy, performance and power consumption can be substantially improved. This design principle is generally known as approximate or inexact computing. Digital Signal Processing (DSP) is the numerical manipulation of signals, usually with the intention to measure, filter, produce or compress continuous analog signals. It is characterized by the use of digital signals to represent these signals as discrete time, discrete frequency, or other discrete domain signals in the form of a sequence of numbers or symbols to permit the digital processing of these signals [1-4].Even if that whole sequence is more complex than analog processing and has a discrete value range, the application of computational power to signal processing allows for many advantages over analog processing in many applications, such as error detection and correction in transmission as well as compression. Digital

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

signal processing and analog signal processing are subfields of signal processing.DSP applications include audio and speech signal processing, sonar and radar signal processing, sensor array processing, spectral estimation, statistical signal processing, digital image processing, signal processing for communications, control of systems, biomedical signal processing, Seismic data processing, among others. DSP algorithms have long been run on standard computers, as well as on specialized processors called digital signal processors, and on purpose-built hardware such as application-specific integrated circuit. Digital signal processing can involve linear or nonlinear operations. Nonlinear signal processing is closely related to nonlinear system identification and can be implemented in the time, frequency, and spatio-temporal domains. In inexact multipliers, approximations can be applied on the partial product generation, as well as the partial product accumulation. Approximations on the partial product generation and approximations on their accumulation are synergistic, and can be applied in collaboration in order to achieve higher power reduction [5, 6].

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. 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

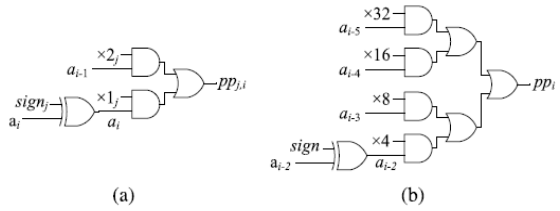
- Partial product generation
- Partial product reduction
- Final addition

PARTIAL PRODUCT GENERATORS (PPG)

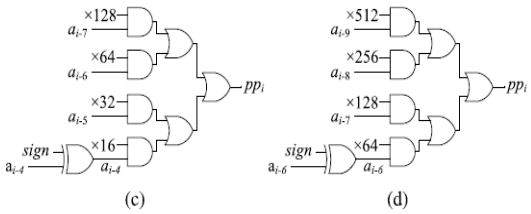
An under-designed 16×16 multiplier using inaccurate 2 × 2 Partial Product Generators (PPG) while guaranteeing the minimum and maximum accuracy fixed at design time. Each PPG has fewer transistors compared with the accurate 2×2 one, reducing both dynamic and leakage energy at the cost of some accuracy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

EXISTING METHOD

High radix encodings offer partial products reduction, and as a result, their accumulation requires smaller trees, leading to energy, area, and/or delay savings. However, high radix encodings require complex encoding and partial product generation circuits, negating thus the benefits of the partial products reduction. In this section, the proposed hybrid high radix encoding and the performed approximations for simplifying its circuit complexity are presented.



(a) Accurate Radix-4 encoding (b) Radix-64

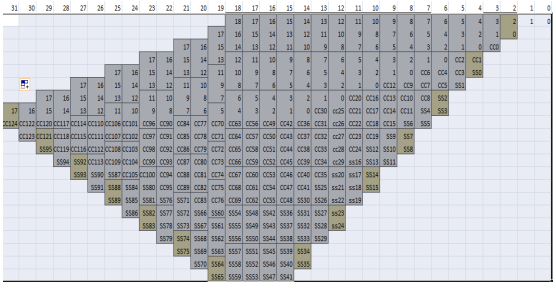


(c) Radix-256 (d) Radix-1024 encoding.

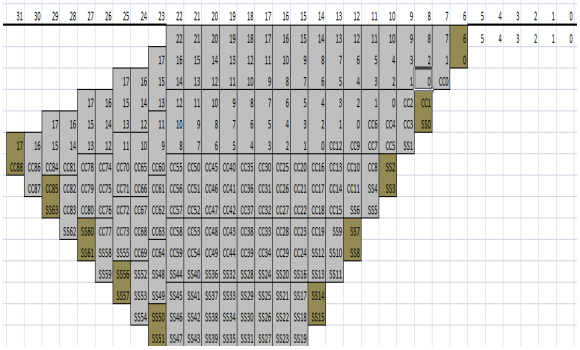
In the four partial product generators are presented, i.e., the circuit of the accurate radix-4 encoding and the ones of the three approximate high radix encodings [7, 8]. The partial products created from each encoding are shown in Table I. In addition, the three hybrid high radix encodings create the partial product trees. The trees also include the encoding's correction term (constant terms and sign factors).

Approximate hybrid high radix encoding for designing energy-error efficient inexact multipliers is proposed. The difference with the existing related work is that it concerns approximations on the generation of the partial products, and can be combined with any accumulation technique, approximate or not. Another significant is that the error imposed depends only on the configuration parameter k, and as a result, it can be calculated without the need for exhaustive simulations. Consequently, a precise estimation of the output quality can be extracted for the application's inputs, giving the flexibility to target the maximum energy reduction for a specific error bound.

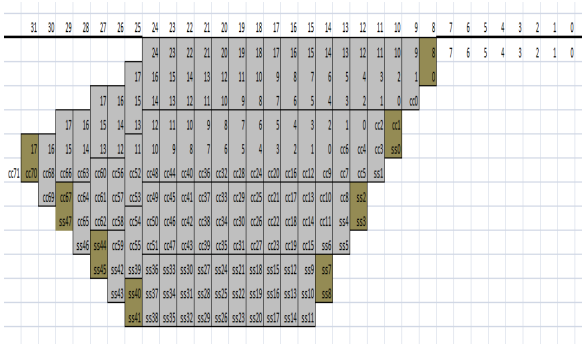
The implementation of the partial product accumulation can be chosen by the designer. In this paper, an accurate Wallace tree is used to implement the partial product's sum, whereas the two outputs produced by the Wallace tree are added using a prefix (fast) adder. Overall, the multiplication circuit consists of stages of operand hybrid radix encoding, partial product generation, partial product accumulation, and final addition. The proposed approximate multipliers are named RAD2*k*, showing the selected approximate high radix encoding, e.g., RAD64, RAD256, and RAD1024.



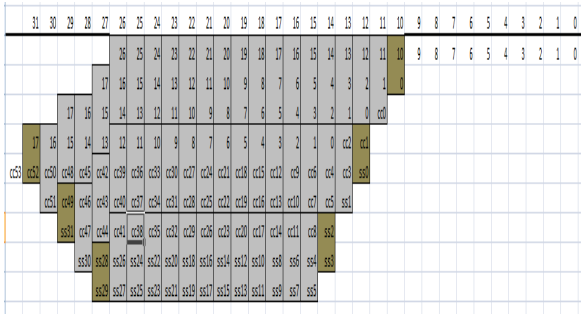
Radix 4



Radix 64



Radix 256



Radix 1024

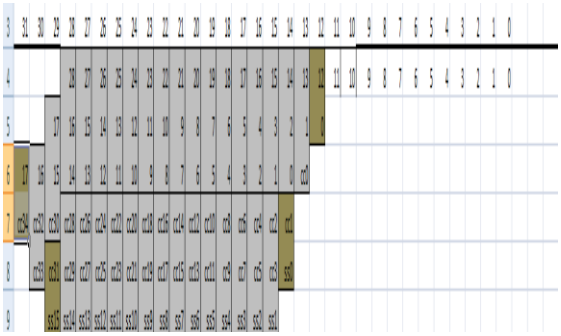
The modified Booth encoding is commonly used in signed multipliers Although these techniques perform fast multiplications, the number of the partial products is not reduced in most cases, in contrast with our design [9, 10]. The partial product perforation technique, where they omit the generation of some partial products based on the modified Booth encoding. an approximate radix-8 booth multiplier that uses an approximate adder for producing 3 A, and combine this idea with the truncation method. Recently approximate modified Booth encoders by modifying its KMap, and combined them with an approximate compressor.

TABLE I
PARTIAL PRODUCTS PER RADIX ENCODING

Radix Encoding	Partial products
Radix 4	0,±A,±2A
Radix 64	0,±4A,±8A,±16A,±32A
Radix 256	0,±16A,±32A,±64A,±128A
Radix 1024	0,±64A,±128A,±256A,±512A
Radix 4096	0,±256A,±512A,±1024A,2048A

PROPOSED METHOD

Approximate hybrid high radix encoding for designing energy-error efficient inexact multipliers is proposed. The difference with the existing related work is that it concerns approximations on the generation of the partial products, and can be combined with any accumulation technique, approximate or not. Another significant aspect of this paper is that the error imposed depends only on the configuration parameter k, and as a result, it can be calculated without the need for exhaustive simulations. Consequently, a precise estimation of the output quality can be extracted for the application's inputs, giving the flexibility to target the maximum energy reduction for a specific error bound.



Radix 4096

In order to give a theoretical evaluation of the proposed multipliers, an area gate model is included. The area evaluation is performed by using the unit gate model of a XOR-2 gate counts as 2 unit gates, an AND-2 or an OR-2 gate is equal to 1 unit gate, and a NOT gate is equal to 0.5 unit gate. According to this model, the number of unit gates of each circuit used in our multipliers [11-14].

Table II

	Gate count	Power
Radix 4	5438	58.56
Radix 64	4920	51.05
Radix 256	4236	51.04
Radix 1024	3552	45.63
Radix 4096	2771	44.79

The advantage of the approximate hybrid high radix multipliers is their simple logic, resulting in fast operation and low power performance. Although overhead is added because of the encoding circuits, it is insignificant because of the approximations made. Also, this offset is compensated with the partial product generators that deliver low area, and the reduction of the number of the partial products.

CONCLUSION

A critical issue in approximate computing designs is the error imposed due to the approximations and how it affects the final results. An error evaluation analysis of the proposed multipliers is presented. In an error evaluation metric is proposed, being called mean relative error distance (MRED). RED is defined as the arithmetic difference between the accurate product and the approximate product divided by the accurate product. we propose an approximate hybrid high radix encoding for generating the

partial products of a signed multiplier. The MSBs of the multiplicand are encoded with the accurate radix-4 encoding, while its k LSBs are encoded with an approximate high radix-2k encoding, with k being a configuration parameter that adjusts the tradeoff between accuracy and energy consumption. The error of the proposed technique follows a Gaussian distribution with near zero average. Compared with state-of-the-art inexact multipliers, the proposed ones constitute better approximate design alternative, outperforming them in both energy consumption and accuracy.

REFERENCE

[1]. A. Lingamneni, C. Enz, K. Palem, and C. Piguet, "Highly energy efficient and quality-tunable inexact FFT accelerators," in Proc. IEEE Custom Integr. Circuits Conf., 2014, 1–4.

[2]. V. Gupta, D. Mohapatra, A. Raghunathan, and K. Roy, "Low-power digital signal processing using approximate adders," IEEE Trans. Comput.- Aided Design Integr. Circuits Syst., 32(1), 124–137, 2013.

[3]. P. Kulkarni, P. Gupta, and M. Ercegovac, "Trading accuracy for power with an under designed multiplier architecture," in Proc. 24th Int. Conf. VLSI Design, 2011, 346–351.

[4]. A. Momeni, J. Han, P. Montuschi, and F. Lombardi, "Design and analysis of approximate compressors for multiplication," IEEE Trans. Comput., vol. 64(4), 2015, 984–994.

[5]. C.-H. Lin and I.-C. Lin, "High accuracy approximate multiplier with error correction," in Proc. IEEE 31st Int. Conf. Comput. Design, 2013, 33–38.

[6]. J. Park, E. Amaro, D. Mahajan, B. Thwaites, and H. Esmaeilzadeh, "AxGames: Towards crowdsourcing quality target determination in approximate computing," in Proc. ACM Int. Conf. Architectural Support Programm. Lang. Oper. Syst., 2016, 623–636.

[7]. I. Sobel and G. Feldman, "A 3×3 isotropic gradient operator for image processing," in Stanford Artificial Intelligence Project. 1968.

[8]. 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, vol. 57(4), 2010, 850–862.

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

[10]. S. T. Chakradhar and A. Raghunathan, "Best-effort omputing: Re-thinking parallel software and hardware," in Proc. ACM/IEEE Design Autom. Conf., 2010, 865–870.

[11]. N. Zhu, W. L. Goh, W. Zhang, K. S. Yeo, and Z. H. Kong, "Design of low-power high-speed truncation-error-tolerant adder and its application in digital signal processing," *IEEE Trans.* Very Large Scale Integr. (VLSI) Syst., 18(8), 2010, 1225–1229.

[12]. 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.

[13]. S. Hashemi, R. I. Bahar, and S. Reda, "DRUM: A dynamic range unbiased multiplier for approximate applications," in Proc. IEEE/ACM Int. Conf. Comput.-Aided Design, 2015, 418–425.

[14]. G. Zervakis, K. Tsoumanis, S. Xydis, D. Soudris, and K. Pekmestzi, "Design-efficient approximate multiplication circuits through partial product perforation," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., 24(10), 2016, 3105–3117.