Approximate Compressors for Error-Resilient Multiplier Design

Zhixi Yang
Sch. of Mechatronics and Automation
National University of Defense Technology
Changsha, Hunan, China
email:nudtyzx@163.com

Jie Han
Dept. of Electrical and Computer Engineering
University of Alberta
Edmonton, AB, Canada
email:jhan8@ualberta.ca

Fabrizio Lombardi
Dept. of Electrical and Computer Engineering
Northeastern University
Boston, MA, USA
email: lombardi@ece.neu.edu

***Abstract*— Approximate circuit design is an innovative paradigm for error-resilient image and signal processing applications. Multiplication is often a fundamental function for many of these applications. In this paper, three approximate compressors are proposed with an accuracy constraint for the partial product reduction (PPR) in a multiplier. Both approximation and truncation are considered in the approximate multiplier design. An image sharpening algorithm is then investigated as an application of the proposed multiplier designs. Extensive simulation results show that the proposed designs achieve significant reductions in area and power while achieving a high signal-to-noise ratio (SNR > 35 dB), compared to their exact counterparts as well as other approximate multipliers.**

***Index Terms*—Compressor, Multiplier, Approximate circuit design**

I. INTRODUCTION

Functional accuracy is a major requirement in conventional arithmetic circuits. However for some applications, arithmetic processing can be performed on an "inexact" or "approximate" basis. Approximate arithmetic circuits have been extensively considered for error resilient applications especially involving human hearing or vision [1, 2]. Multiplication is a fundamental arithmetic operation for many digital signal processing (DSP) algorithms and the approximation of a multiplier offers an effective approach to obtain low hardware utilization.

Two types of approximate compressors are proposed in [3] for use in approximate multipliers. These compressors have shown shorter delay and lower power consumption compared with an accurate compressor; an image processing application has shown a relatively high accuracy with considerable power reduction. An imprecise counter based 4 by 4 bit multiplier (ICM) is proposed in [4] to build a multiplier of a larger size. Four different modes of an approximate Wallace tree multiplier (AWTM) using a carry prediction method are proposed in [5], resulting in hardware reduction and hence, reduced power, area and delay compared to the accurate Wallace tree multiplier. An approximate multiplier is proposed in [6] by using consecutive m bits in an operand as segmented inputs; m is usually no smaller than half of the operand bit width n. The static segment method (SSM) in [6] is to fix the start point of a segment to make the method with scalable accuracy.

In this paper, three approximate compressors are proposed by modifying logic structure of an accurate compressor. Different from [3], in which no constraint is applied during the design phase, the compressor designs proposed in this paper are restricted to keep a low probability of error occurrence. Due to the high accuracy of compressors, both approximation and truncation are employed for partial product reduction (PPR), whereas in [3] only approximate compressors are used. Finally an image sharpening algorithm is implemented for evaluating various multiplier designs.

II. PROPOSED APPROXIMATE COMPRESSORS

In this section, approximate compressors are designed under the constraint of a low error rate. Assume the inputs to a multiplier are uniformly distributed, so the probability that a partial product (PP), generated by an AND gate, equals to '1' ('0') is 1/4 (3/4). Clearly, the probability of PP being'0' is much higher than being '1'. Table I shows the truth table of an accurate compressor without $C_{in}$and C_{out}. When the inputs are all 1s, the actual output requires three bits. However, the bit (in bold in Table I) is ignored and instead two bits are used as $Carry$ and Sum (denoted as CS). Note for X_1~X_4probability of logic '1'is 1/4.

Table I
Truth table for an accurate compressor without C_{in} and C_{out}[3]

CS	X_1X_2 00	01	11	10
X_3X_4 00	00	01	10	01
01	01	10	11	10
11	10	11	**1**00	11
10	01	10	11	10

In order to obtain high accuracy, a low error rate (ER) is employed as a constraint. ER is defined as

$$ER = \frac{\sum_{t=1}^{T} D_t}{T}, \quad (1)$$

where T is the number of input values. D_t is defined as:

$$D_t = \begin{cases} 0 & if\ a_t = b_t \\ 1 & if\ a_t \neq b_t \end{cases}, \quad (2)$$

where a_t and b_t are the correct and incorrect outputs respectively for a given input t.

Obviously, modification on the entries in the row (shaded in Table 1) when X_3X_4 is '11' can lead to a lower ER compared to changing the entries in other rows. Equivalently, the column for $X_1X_2 = '11'$ can instead be modified since Table I is symmetric along the diagonal. By approximating the truth table, three compressors are designed, referred to as the Approximate Compressors with C_{in} and C_{out} Ignored (ACCIs).

Table II is the truth table of the first ACCI, ACCI1. Note that in an accurate compressor, when inputs are all "1", the output is "100", whereas this output is modified to "11" in ACCI1 with an ER of 1/256. Since $Carry$ is more important to accuracy than Sum, $Carry$ is fixed at '1' when inputs are all 1's in all of the proposed designs.

Table II
Truth table of ACCI1: the entry '100' in Table I is modified to '11.'

CS	X_1X_2 00	01	11	10
X_3X_4 00	00	01	10	01
01	01	10	11	10
11	10	11	**11**	11
10	01	10	11	10

However, the modified truth table results in a rather complex sum, $Sum = (X_1 \oplus X_2) \oplus (X_3 \oplus X_4) + X_1X_2X_3X_4$, where the last term is due to the inputs '1111.' In order to reduce the complexity, Sum is modified to '1' for input '0011.' As a result, the first term becomes $(X_1 \oplus X_2) \oplus (X_3 + X_4)$ and the last term becomes X_3X_4 for Sum, i.e., $Sum = (X_1 \oplus X_2) \oplus (X_3 + X_4) + X_3X_4)$.

Table III
Truth table for ACCI2

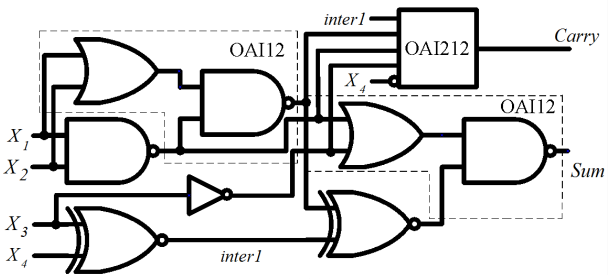
CS	X_1X_2 00	01	11	10
X_3X_4 00	00	01	10	01
01	01	10	11	10
11	1**1**	11	**11**	11
10	01	10	11	10

The third compressor design is based on ACCI2 to further reduce the logic complexity for generating Sum by removing the last term. Table IV is the truth table for ACCI3.

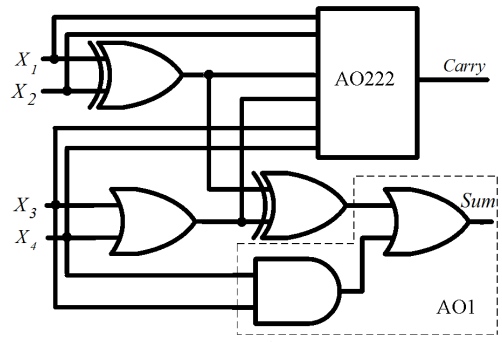
Table IV
Truth table for ACCI3

CS	X_1X_2 00	01	11	10
X_3X_4 00	00	01	10	01
01	01	10	11	10
11	1**1**	1**0**	**11**	1**0**
10	01	10	11	10

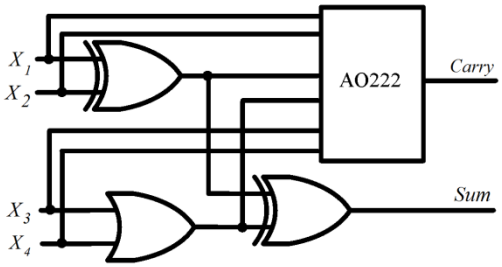
ER is chosen as the constraint metric for the approximate designs. The ERs of ACCI 1~3 and the design in [3] are 1/256, 10/256, 1/16 and 25/64, so the proposed ACCIs have better accuracy. Since the ACCIs have better accuracy, the truncation of lowest PP columns is employed in the multiplier design, while the approximate compressors are used for the PPR at more significant bits.



(a)



(b)



(c)

Fig. 1. Schematic for (a) ACCI1, (b) ACCI2 and (c) ACCI3. OAI212 (OR-AND-INV) , AO222 (AND-OR) are complex compound gates

III. Proposed Approximate Multipliers

A binary multiplier usually consists of three stages:

- Partial product generation using an AND gate.
- Partial product reduction using an adder tree.
- Carry propagation adder (CPA) for the addition of the final results.

In the design of a multiplier, the partial product reduction plays a pivotal role in determining the delay, power consumption and circuit complexity of the multiplier [3]. Compressors are often used to achieve reductions in power and delay (compression is executed in parallel). By replacing exact compressors, an approximate multiplier is obtained at a reduced circuit complexity and possibly with reduced power dissipation.

In the proposed approximate multipliers, both approximation and truncation are used for the partial product reduction. An 8 by 8 bit unsigned Dadda multiplier is considered, as shown in Fig. 2.

In Fig. 2, the first stage of PP generation is not shown; only the second and third stages are illustrated (a dot represents a PP). The proposed ACCIs are implemented with no C_{in} or C_{out} signals . C_{out} from the half or full adder is grouped as inputs to the next reduction stage. Fig. 2 also shows that the least significant 4 bits are truncated and the next 4 bits are used for approximation. For the remaining more significant partial products, accurate compressors are applied for PPR. Hence nine accurate compressors, eight approximate compressors, three full adders and two half adders are necessary for the approximate multiplier.

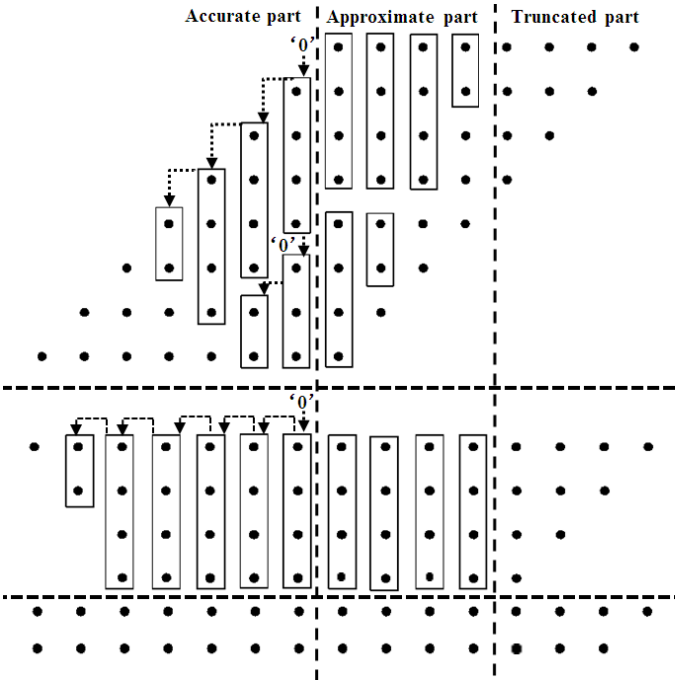


Fig. 2. Partial product reduction using truncation and the proposed approximate compressors for an 8x8 bit Dadda multiplier.

The use of truncation and approximate compressors in the less significant bits decreases power consumption and circuit area, while the accurate compressors used in the more significant bits reduce the loss of accuracy.

IV. Image Processing

In this section, an image sharpening algorithm is considered as an application of the proposed multiplier. Other designs from the literature are also included for comparison. The features of the multiplier are summarized in Table V. Image quality and circuit related metrics are considered for assessing the designs. A joint analysis of accuracy and power-delay-area product is also performed.

Table V
Features of approximate 8x8 bit multipliers

Multiplier Design	Features
M_1~ M_3	Multiplier with ACCI1~3 implemented as in Fig. 2 with both truncation and approximation
ICM	Multiplier with imprecise counters [4]
APC	Lower 6 columns with compressors in [3] for a Dadda tree multiplier;
AWTM	Approximate Wallace tree multiplier with mode 4 as in [5]
SSM	SSM in [6] with segment length m=6

Assume I is the original image and S is the processed image, the sharpening algorithm in [7] performs:

$$S(x,y) = 2I(x,y) - \frac{1}{273}\sum_{i=-2}^{2}\sum_{j=-2}^{2} G(i+3,j+3)I(x-i,y-j) \quad (3)$$

where G is a matrix given as:

$$G = \begin{bmatrix} 1 & 4 & 7 & 4 & 1 \\ 4 & 16 & 26 & 16 & 4 \\ 7 & 26 & 41 & 26 & 7 \\ 4 & 16 & 26 & 16 & 4 \\ 1 & 4 & 7 & 4 & 1 \end{bmatrix}$$

This algorithm is performed on blocks of 5x5 pixels in an image. Only the multiplications are approximate, while the other operations including addition, subtraction and division are accurate.

Table VI shows the processed images by different multipliers. The simulation results for signal noise ratio (SNR) and Structure SIMilarity (SSIM) [8] are presented in Table VII in comparison with the accurately processed images. SSIM evaluates the similarity of two images, while SNR is defined as

$$SNR = 10log_{10}(A/MSE), \quad (4)$$

where A is the amplitude of a signal and MSE is given by:

TableVI
Processed image by different multipliers

Accurate	APC	ICM	SSM	AWTM	M1
					
					

$$MSE = \frac{1}{N}\sum_{t=0}^{N-1} error^2(t), \quad (5)$$

with *N* the number of inputs and *error* defined as:

$$error(t) = a_t - b_t. \quad (6)$$

TableVII
Processed image quality comparison

	Lenna.jpg		Ela.jpg	
Comparison	SNR	SSIM	SNR	SSIM
M1	34.817	0.9991	39.276	0.9996
M2	36.074	0.9986	40.531	0.9993
M3	35.916	0.9985	40.166	0.9993
ICM	19.654	0.9549	24.316	0.9725
APC	25.898	0.9978	30.149	0.9989
AWTM	3.182	0.8605	7.794	0.9112
SSM	30.585	0.9985	34.381	0.9992

As seen in Table VI, images processed by AWTM and ICM show visible degradations of image quality while the images processed by APC, M1 and SSM are difficult to distinguish visibly from the accurately processed images. Note that M2 and M3 result in visually non-distinguishable images as the accurately processed ones, so they are not shown.

As shown in Table VII, M1~M3 result in better image qualities in terms of both SNR and SSIM compared to the other designs; in particular, M2 has the highest SNR value. Moreover, the SSIM values are larger than 0.99 for M1~M3, which indicates a high similarity between the approximately and accurately processed images.

The approximate multipliers are further analyzed for power, area and delay. The multipliers are implemented in VHDL and synthesized to gate-level netlist using Cadence RTL Complier (RC) with a standard STMicroelectronics (STM) 65nm CMOS cell library at a typical corner process with 1.0V as supply voltage (at 25℃). Table VIII shows the power, area and delay for different designs generated by the synthesis report. An accurate Dadda tree multiplier is used for assessment of power, area and delay improvements.

TableVIII
Power, delay and area comparison for different multipliers

	Power (uW)	Delay (ns)	Area (um^2)
M1	43.9	2654	655
M2	38.7	2610	646
M3	37.3	2309	601
ICM	45.5	2692	832
APC	46.4	2973	706
AWTM	26.6	3000	508
SSM	29.9	2720	496
Accurate	52.7	3166	766

- M1 consumes a larger power than M2 and M3. AWTM and SSM are good at reducing power, achieving 49% and 43% reduction respectively.
- In terms of delay, M3 has a shorter delay than M1 and M2; while SSM has the smallest delay among all.
- As for area, SSM and ESSM outperform the others; M3 requires the smallest area among the proposed multipliers.

To consider both the accuracy and circuit characteristics, a joint analysis is performed to evaluate a figure of merit (FOM). Power and delay are chosen as major metrics to be compared since area shows the same trend as power. So FOM of QUPD as similarly did in [9], is defined in this paper as

$$QUPD = QUality \times Power\ saving \times Delay\ saving, \quad (7)$$

where quality is measured by the average of $(SNR)^2$ for the two processed images. Note that M1 consumes more power and has a larger delay than M2 and M3 with a lower image quality, so it is not considered in the FOM comparison. Table VIII gives the QUPD of each design. AWTM has the lowest QUPD while M2 and M3 have larger values than SSM, indicating that the proposed designs of M2 and M3 achieve better trade-off between accuracy and power-delay product.

Table VIII
Analysis of QUPD

	M2	M3	AWTM	SSM	APC	ICM
QUPD	68.67	114.82	0.92	64.53	5.75	9.99

V. Conclusion

In this paper, three designs of approximate 4-2 compressors are proposed and these designs are used in the partial product reduction circuit of a multiplier. As constrained by a low error rate, all three designs have a very high accuracy, so both truncation and approximation are used in the multiplier design to further reduce power, area and delay. For an image sharpening application, the compressors and multipliers show significant improvements in power consumption, delay and area compared with an accurate and other approximate designs.

References

[1] J. Han and M. Orshansky."Approximate Computing: An Emerging Paradigm For Energy-Efficient Design." *In proceedings of the 18th IEEE European Test Symposium*, Avignon, France, May 2013, pp.1-6.

[2] J. Liang, J. Han and F. Lombardi. "New metrics for the reliability of approximate and probabilistic adders." *IEEE Trans. Computers.* vol.62, no.9, pp.1760-1771, Sept. 2013.

[3] A. Momeni, J. Han, P. Montuschi and F. Lombardi. "Design and Analysis of Approximate Compressors for Multiplication." *IEEE Trans. Computers,* vol.64, no.4, pp. 984-994, Apr. 2015.

[4] C.H. Lin and I.C. Lin. "High accuracy approximate multiplier with error correction." *In Proc. ICCD'13: In the 2013 IEEE 31st International Conference on Computer Design (ICCD)*. Asheville, NC, USA, Oct. 2013, pp. 33-38.

[5] K. Bhardwaj, P.S. Maneand J. Henkel. "Power-and area-efficient Approximate Wallace Tree Multiplier for error-resilient systems." *In Sym. ISQED'14: In 15th International Symposium on Quality Electronic Design (ISQED)*, Santa Clara, CA, USA. Mar. 2014, pp. 263-269.

[6] S. Narayanamoorthy, H.A. Moghaddam, Z. Liu, T. Park, N.S. Kim, "Energy-Efficient Approximate Multiplication for Digital Signal Porcessing and Classificaiton Applications." *IEEE Trans. Very Large Scale Integration Systems (VLSI)*, vol.23, no.6, pp.1180-1184, Jun. 2015.

[7] M.S. Lau, K.V. Ling and Y.C. Chu. "Energy-aware probabilistic multiplier: design and analysis." *In Proceedings of the 2009 international conference on Compilers, architecture, and synthesis for embedded systems*, Grenoble, France, Oct. 2009, pp.281-290.

[8] Z. Wang, A.C. Bovik, H.R. Sheikh and E.P. Simoncell. "Image quality assessment: From error visibility to structural similarity". *IEEE Trans. Image Processing.* vol.13, no.4. pp. 600-612, Apr. 2004.

[9] V. Gupta, D. Mohapatra, S.P. Park, A. Raghunathan and K. Roy. "IMPACT: imprecise adders for low-power approximate computing." *In ISLPED'11: Proceedings of the 17th IEEE/ACM international symposium on Low-power electronics and design*, 2011, pp. 409-414.