Design and Analysis of Approximate Sum-of-Products

Latha yerrabhumi[1], K.Mamatha[2]
[1]PG Research Scholar, Dept. of Electronics and Communication Engineering, JNTUCEA, AP, INDIA
[2]Assistant Professor (Adhoc), Dept. of Electronics and Communication Engineering, JNTUCEA, AP, INDIA

Abstract— **This paper deals with design and analysis of three approximate sum-of-products(ASOP) units for application of digital signal processing. The three ASOPs are based on a principle called Distributed Arithmetic(DA). Distributed Arithmetic is technique where conventional multipliers are replaced with look-up tables, adders, shifters and accumulators. The designed units employ different mechanisms for the reduction of area, power considering some error metrics. The error metrics includes mean relative error and normalised error distance . The implemented units are evaluated for noisy image smoothing application. It is shown that implemented models achieve higher peak signal to noise ratio than the existing exact models.**

***Keywords*— sum-of-products, distributed arithmetic, digital signal processing, power, area**

I. Introduction

Now–a-days companies dealing with huge data are interested in approximate computing even with some loss of accuracy. In applications like machine learning, data mining, financial modelling, statistics, gaming, digital signal processing, multimedia processing etc., approximate analysis preferred.

In arithmetic circuits SOP units are key elements but having less attention in approximate analysis [1]. The SOP units are evaluated based on the principle of distributed arithmetic. DA is a very systematic way to calculate inner product of vectors [2]. DA is bit serial in nature and computes the inner products in parallel[3] due to which area and power trade off can be adjusted. It replaces the multipliers with number of look-up tables and adders in such a way that no conventional multipliers are used. For analysing SOP units in arithmetic circuits three approximate models are proposed based on truncation [4]. First model provides low power and low area with considerable values of Mean relative error (MRE) and Normalized error distance (NED). The NED is defined as the difference between absolute value to approximate value and MRE is defined as ratio of error distance to the exact value. The error parameters are used as proficiency testing results for ASOPs. Generally digital signal processing applications are based on distributed arithmetic explained in [5]. Efficient design of approximate circuits and new metrics for error calculation are explained in [6-7]. This paper is organised as sections. Section II presents brief introduction about the related work of this paper. Section III presents the implemented work, Section IV presents results and Section V presents conclusion.

II. RELATED WORK

A. Prior Works

The SOP units are implemented with the well known technique called Distributed Arithmetic in which no multipliers are used. The multipliers are replaced by adders and lookup tables. The SOP units are generally used in filters and digital signal processing applications[5]. Consider unsigned K elements of N-bits $B_1, B_2, .., B_k$ and $A_1, A_2, .., A_k$. To perform SOP $A_1B_1+A_2B_2+ ..+A_kB_k$ the result Y_{out} can be written as

$$Y_{\text{out}} = \sum_{n=0}^{N-1} \mathrm{X_n} 2^n \quad (1)$$

Where

$$\mathrm{X_n} = \sum_{k=1}^{K} \mathrm{A_k B_{kn}} \quad (2)$$

A and B i.e., k=1,2,3 for bit position n. The sum of A inputs based on B are shown in Table I, where A_{ij} indicates $A_i + A_j$. To implement sum of A inputs a lookup table is used.

TABLE I

CONTENTS OF LOOKUP TABLE FOR K=3

B_{1n}	B_{2n}	B_{3n}	Contents
1	1	1	A_{123}
1	1	0	A_{12}
1	0	0	A_{13}
1	0	0	A_1
0	1	1	A_{23}
0	1	0	A_2
0	0	1	A_3
0	0	0	0

B. Exact sop for N=16

The exact SOP is considered for 16-bit input vectors. The possible combinations of vector A are stored in lookup table based on vector B_{kn} for n=0,1,2,..,N-1. The lookup table is same as multiplexer where the B_{1n},B_{2n} and B_{3n} acts as selection lines or address lines and A as input vector. For N-bit, N lookup tables are required. For 16-bit, 16 lookup tables are required which increases the area which in turn increases the power consumption.

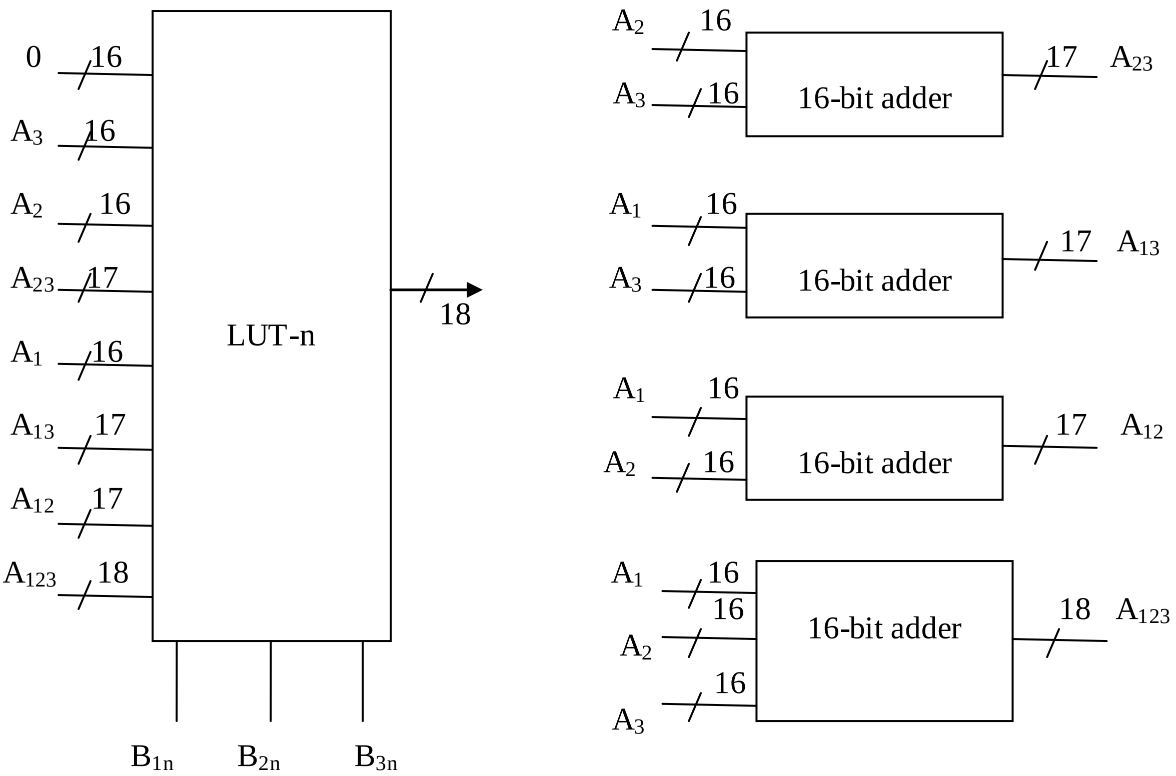

Fig.1. Lookup table and 16-bit adder of input vectors

The lookup table contents like A_{ij} are fed to the 16-bit adder for calculating A_i+A_j. In implementing exact SOP conventional multiplication is not performed instead of that shifting is employed. In fig.2. the exact SOP model is shown for K=3 and N=16 which requires 16 lookup tables and four adders and a accumulator. In exact SOP area occupied is more because all lookup tables are used and power consumption is also increases with zero mean relative error and normalised error distance.

number of lookup tables to be truncated where equation (1) becomes equation (3) as shown below

$$Y_{\mathrm{OUT}} = \sum_{n=m}^{N-1} X_n 2^n \qquad (3)$$

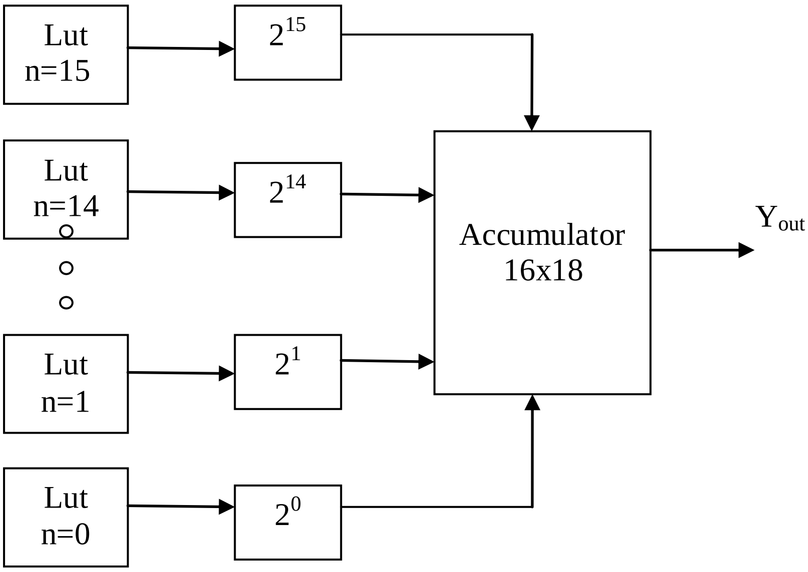

Fig.2. Exact SOP model for K=3 and N=16

III.IMPLEMENTED WORK

In implemented work three approximate SOPs are proposed for noisy image smoothing application. In proposed models hardware is considerably reduced compared to exact SOP model. The approximation models employs truncation mechanisms.

A. ASOP model 1

In this model truncation is employed at the least significant part of the input vectors A_k and B_k. The truncation of bits depends on m value i.e., m=4,6,8. For example if m=4 then four bits are truncated at the least significant part of input vectors. Similarly for m=6 and m=8 the least significant part of input vectors is truncated and implemented. If the bits are truncated then number of lookup tables are also truncated which in turn reduces the area and power compared to the exact SOP model. This model is implemented by changing the limits of equation (1) from n=m to N-1. For m-bit truncation number of lookup tables required starts from n=m to 15 for 16-bit. This model suffers from negative effects of truncation when information present least significant part of input vectors.

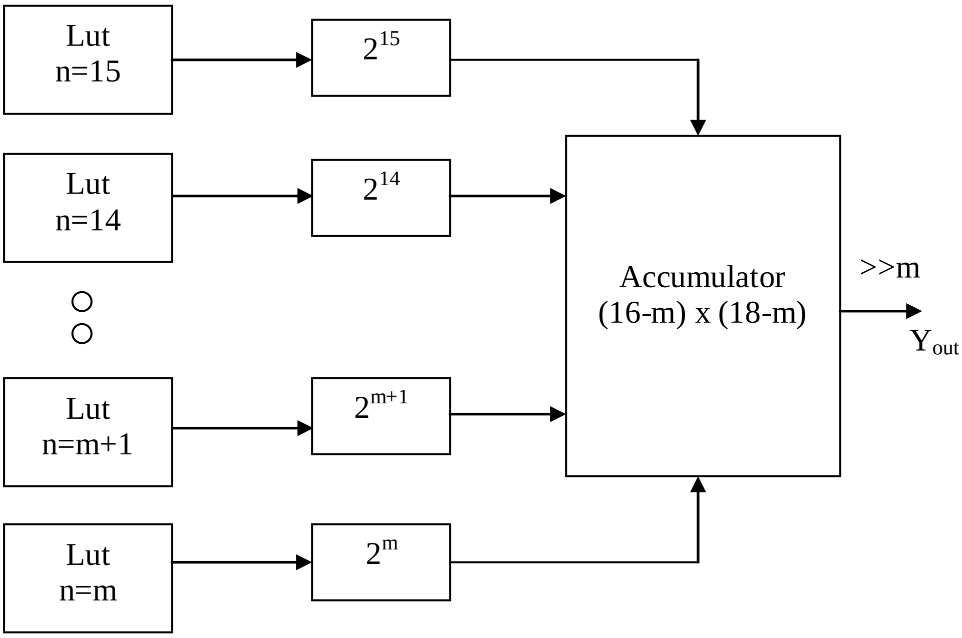

fig.3. ASOP1 for k=3 and N=16

This model is implemented to avoid the negative effects of truncation in model 1. This model employs leading one predictor. The leading one prediction is a method where OR operation is performed for the most significant part of input vectors for m-bit. If m=4 then OR operation for four most significant parts of A_K and B_K for K=1,2,3 is performed. The OR gates function can be given as $A_{mOR} = A_{1m}|A_{2m}|A_{3m}$ and $B_{mOR} = B_{1m}|B_{2m}|B_{3m}$. If leading one predictor predicts zeros in the most significant part then that zeros are truncated at the most significant part.ASOP1 is preferred after leading one prediction. This model reduces the negative effects of truncation. The fig 4 shows the implementation of ASOP2

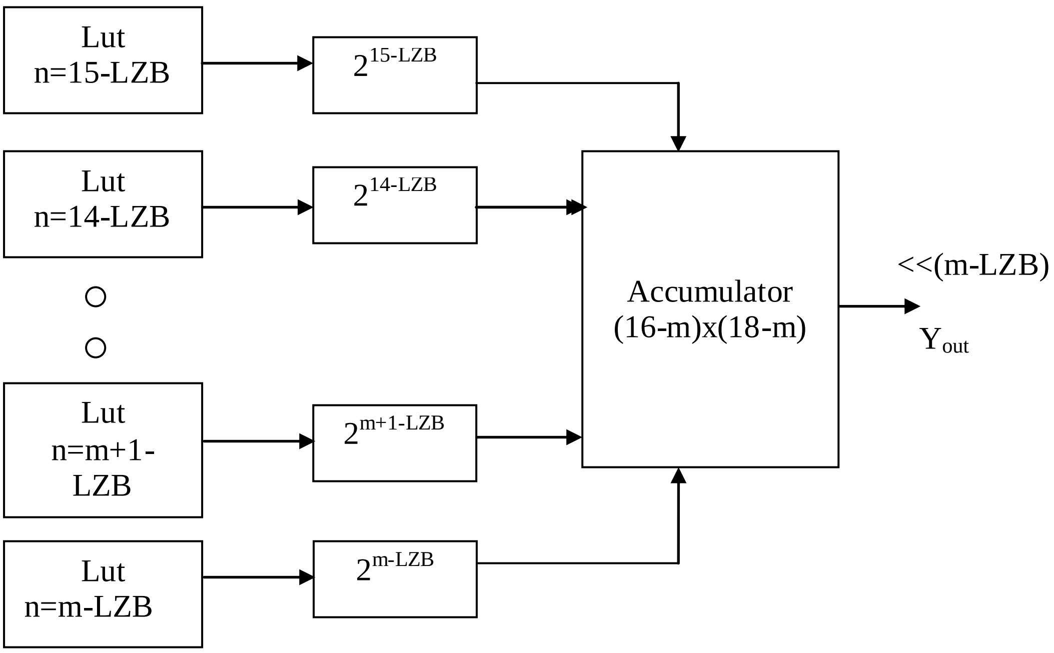

Fig.4. ASOP2 for k=3 and N=16

C. ASOP model 3

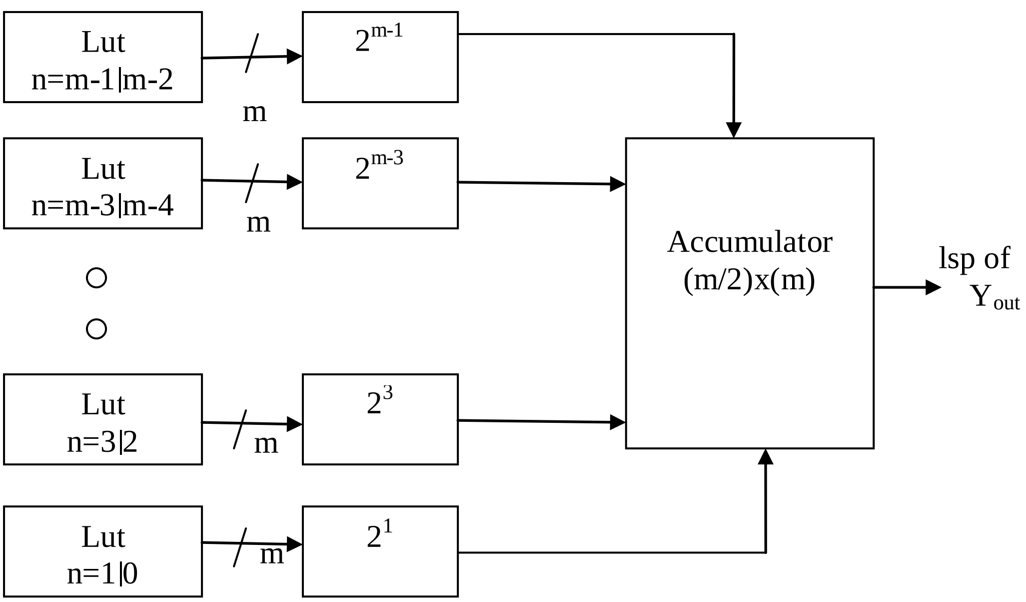

fig.5. lsp of approximate model 3

employed. The B inputs are divided into two groups, N-m group and m group, for N-m group model 1 is employed and for m group address lines are grouped in pair. If N=16 then for16-m group model1 is employed and for m group asop3 is used. In this model for m group the number of lookup tables are reduced to m/2.

The implemented models plays a vital role in many real time applications like digital signal processing such as image smoothing. The models implemented are used for performing convolution over every pixel of input noisy image with considered 3x3 Gaussian kernel[8]. The peak signal to noise ratio is calculated between the input image and the output image by using the below formula.

$$psnr = 10\log_{10}\frac{(2^N-1)^2}{MSE}$$

Where MSE is mean square error calculated between input and output image.

After convolution the image size gets reduced due to approximation using in this paper. The image smoothing using Gaussian kernel is obtained with the help of Matlab. For 16-bit nearly 1 million random combinations are obtained. The peak signal to noise ratio evaluates the quality of the compressed, image higher the ratio lower will be the noise and better the compressed image.

The image smoothing is performed using noisy image as input. The image taken is converted to text that is pixels. The properties of Gaussian includes; it is most common natural model, smooth model having infinite number of derivatives, it is symmetric, its fourier transform of Gaussian is Gaussian and there are cells in eye that performs Gaussian filtering. The Gaussian filter usually used to blur the image or to reduce noise and also used for edge detection and to reduce contrast.

Each pixel multiplied with the 3x3 Gaussian kernel and added using approximate sop model for particular truncation and then converted text to image. The peak signal to noise ratio is calculated between the noisy image and the output image i.e., reference image. The mean square error is also calculated between the noisy image and the compressed image. If the peak signal to noise ratio is higher then better will be the quality of compressed image.

IV. RESULTS

Simulation of exact and approximate models are described in verilog hardware description language . For 16-bit input vectors, nearly 1 million uniform random values are applied to get results. Synthesis results of exact and implemented sops are performed using design compiler by synopsis 28nm technology. Table II represents the results of area and power for exact and approximate models with different values of truncation.

TABLE 11
DESIGN PARAMETERS OF EXACT AND APPROXIMATE SOP MODELS

Sum of products type	Area (um^2)	Power (mW)	APP
Exact sop	12791	6.40	81862.4
ASOP1(m=4)	6826	3.78	25802.28
ASOP1(m=6)	5458	3.14	17138.12
ASOP1(m=8)	2353	1.45	3411.85
ASOP2(m=4)	9985	5.82	58112.7
ASOP2(m=6)	9948	4.89	48645.72
ASOP2(m=8)	9446	4.48	42318.08
ASOP3(m=4)	5078	4.19	21276.82
ASOP3(m=6)	6697	3.32	22234.04
ASOP3(m=8)	5078	2.40	12187.2

TABLE III

ERROR PARAMETERS OF IMPLEMENTED APPROXIMATE MODELS

SOP type	MRE	NED	PSNR(db)	MSE
Exact SOP	-	-	28.4919	6.0779e+06
ASOP1(m=4)	5.770152e-04	1.43850e-04	28.3567	6.2701e+06
ASOP2(m=4)	4.894470e-04	1.062371e-04	28.4819	6.0920e+06
ASOP3(m=4)	4.700975e-04	9.058400e-05	28.3584	6.2677e+06
ASOP1(m=6)	2.423657e-03	4.810485e-04	27.8676	7.0176e+06
ASOP2(m=6)	2.070835e-03	4.482602e-04	28.4865	6.0854e+06
ASOP3(m=6)	1.973640e-03	3.807249e-04	27.8848	6.9898e+06
ASOP1(m=8)	9.766575e-03	1.939316e-03	18.4602	6.1225e+06
ASOP2(m=8)	8.370204e-03	1.809833e-03	28.4209	6.1781e+06
ASOP3(m=8)	3.807249e-04	7.967006e-03	18.5301	6.0247e+06

The simulation waveforms of exact sum-of-products model is shown in figure 6 and the simulation waveforms of approximate sum-of-products model 1 is shown in figure7.

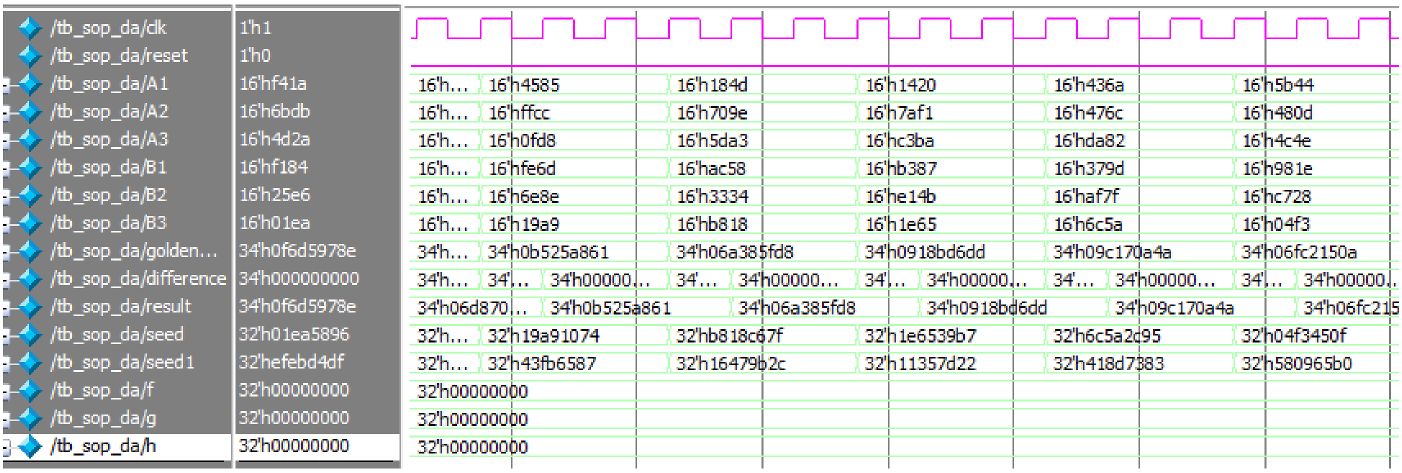

Fig .6. simulation waveforms of exact SOP model

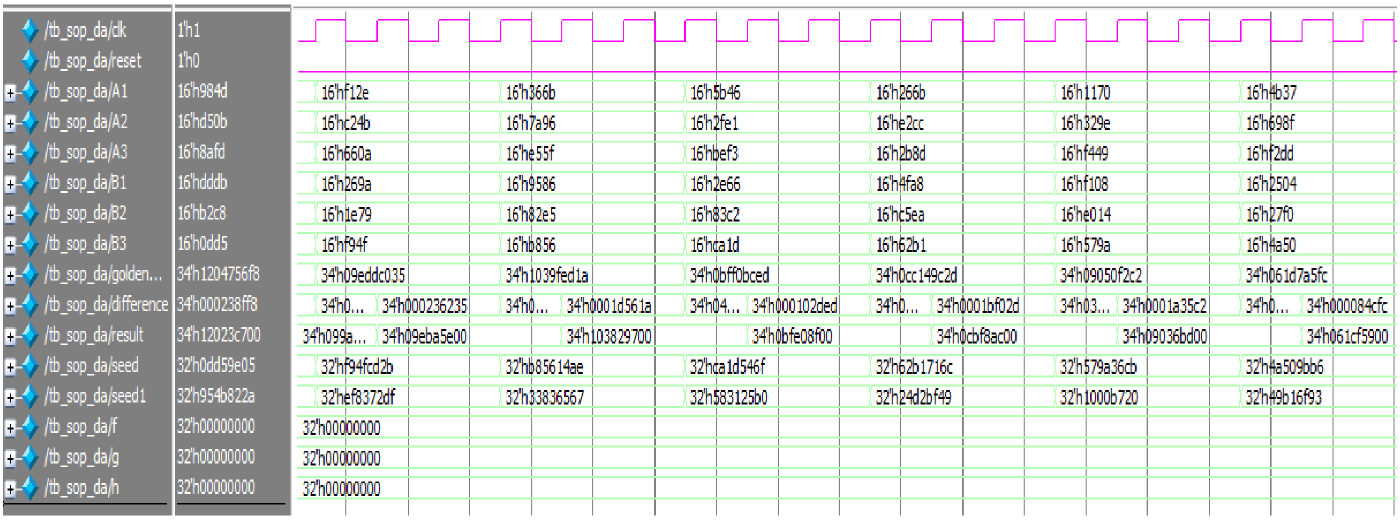

Fig .7. simulation waveforms of approximate model 1

The functional behaviour of approximate sum-of-products model 1 is shown in figure 8

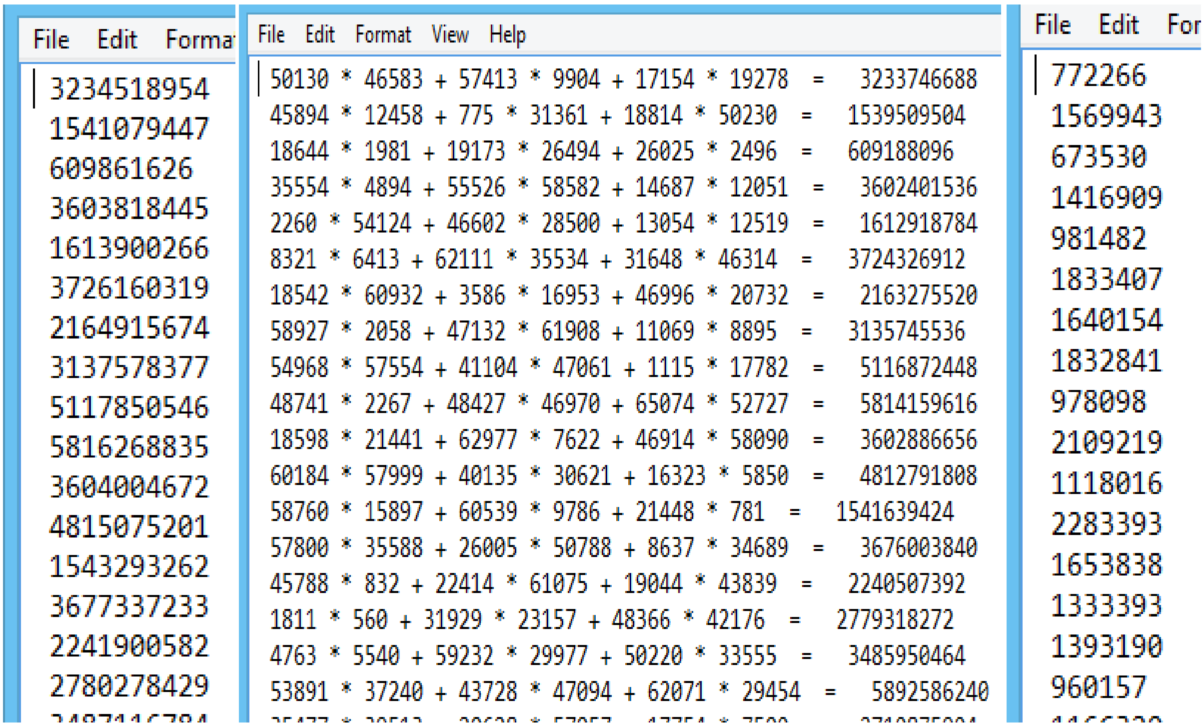

Fig.8.Functional behaviour of approximate SOP model

The input noisy image and images after smoothing for different approximate model for different truncation mechanisms is shown figure 9.

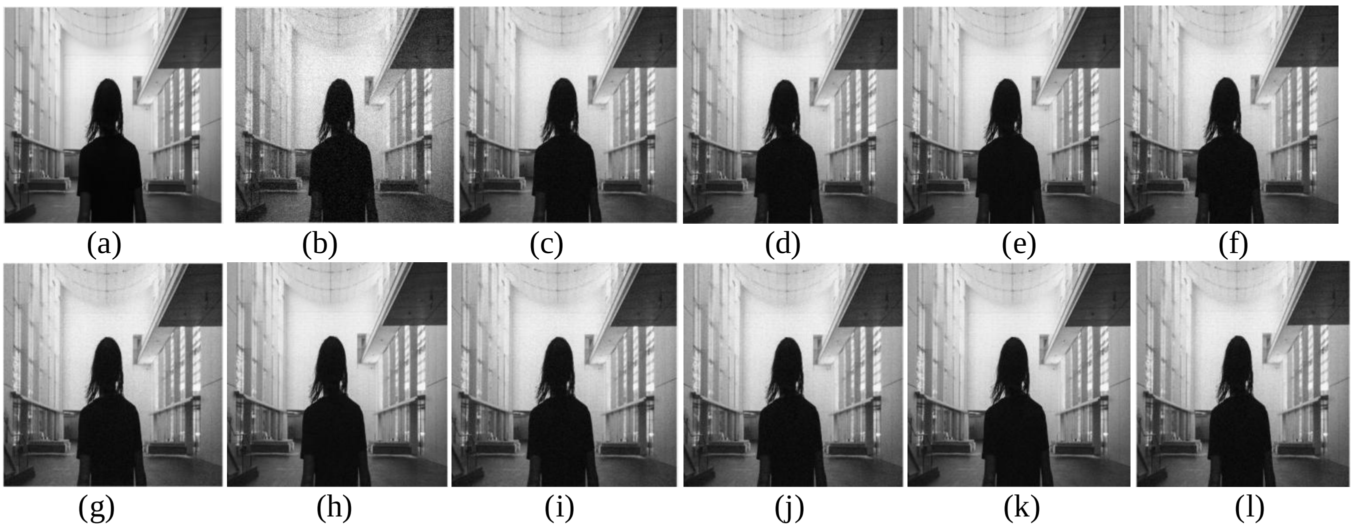

Fig.9. (a) Input image (b) Input image after application of noise. Images after smoothing using (c)ASOP1(m=4), (d)AS0P2(m=4), (e) AS0P3(m=4), (f) AS0P1(m=6), (g) ASOP2(6), (h)ASOP3(6), (i)ASOP1(m=8), (j)ASOP2(m=8), (k) ASOP3(m=8), (l)exact SOP

V CONCLUSION

Three approximate sum-of-product models are implemented for different levels of truncation. When these models operated in approximate mode they consume less area than the exact mode for different levels of truncation at the cost of less precision. The effectiveness of approximate sum-of-products models for bit length of 16 were compared with the exact sum-of-products model. The first model achieves an improvement of 62% on area and 43% on power. The second model achieves an improvement of 45% on area and 79% on power. The third model achieves an improvement of 43% on area and 51% on power compared with the exact sum-of-products model. The second model achieves lower mean relative error and normalised error distance compared to first and third model. The approximate sum-of-product models are designed for lower area and power used for image smoothing application and its functionality is verified.

REFERENCES

[1] J. Han and M. Orshansky, “Approximate computing: An emerging paradigm for energy-efficient design,” in *Proc. IEEE ETS*, May 2013, pp. 1–6.

[2] S. A. White, “Applications of distributed arithmetic to digital signal processing: A tutorial review,” *IEEE ASSP Mag.*, vol. 6, no. 3, pp. 4–19, Jul. 1989.

[3] L. Yuan, S. Sana, H. J. Pottinger, and V. S. Rao, “Distributed arithmetic implementation of multivariable controllers for smart structural systems,” *Smart Mater. Struct.*, vol. 9, no. 4, p. 402, Jan. 2000.

[4] W. Li, J. B. Burr, and A. M. Peterson, “A fully parallel VLSI implementation of distributed arithmetic,” in *Proc. IEEE Int. Symp. Circuits Syst.*, vol. 2. Jun. 1988, pp. 1511–1515.

[5] R. Amirtharajah and A. P. Chandrakasan, “A micropower program- mable DSP using approximate signal processing based on distributed arithmetic,” *IEEE J. Solid-State Circuits*, vol. 39, no. 2, pp. 337–347, Feb. 2010.

[6] 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.*, vol. 24, no. 10, pp. 3105–3117, Oct. 2016.

[7] J. Liang, J. Han, and F. Lombardi, “New metrics for the reliability of approximate and probabilistic adders,” *IEEE Trans. Comput.*, vol. 63, no. 9, pp. 1760–1771, Sep. 2013.

[8] J. Babaud, A. P. Witkin, M. Baudin, and R. O. Duda, “Uniqueness of the Gaussian kernel for scale-space filtering,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-8, no. 1, pp. 26–33, Jan. 1986.