Low Power Multiplier using Approximate Compressor for Error Tolerant Applications

Balaji M, Vishal Gundavarapu, Sasipriya P, Kanchana Bhaaskaran V S

Abstract: ***Approximate or inexact computing has gained a significant amount of attention for error tolerant systems such as signal processing and image processing applications. In this paper, a comprehensive analysis and evaluation of multipliers realized using the existing approximate 4-2 compressors towards achieving low power has been presented. 8-bit Dadda multiplier has been chosen and the power consumption comparison has been performed. The exact multiplier has also been realized to enable the calculation of power savings for the approximate multipliers. An image compression algorithm using approximate multipliers has been implemented to analyze the operability of the approximate multipliers. Accuracy of the approximate multipliers has also been computed by means of Normalized Error Distance (NED) and PSNR. All the circuits are designed using 45nm CMOS process technology and simulations are carried out using Cadence® Virtuoso design tools.***

Keywords: 4-2 compressor, Approximate compressor, Dadda multiplier, Low power multiplier.

I. INTRODUCTION

Accuracy is one of the most important facets of arithmetic circuits. However, in order to maintain a high level of accuracy, a lot of power is consumed. With current advances in technology, power consumption and speed have been gaining significant attention. Therefore, in places where one might not require perfect accuracy, inaccurate arithmetic circuits are used. Approximate circuits function at a high speed and consume considerably lesser power than their accurate counterparts.

In this paper, the focus is on approximate multipliers. The approximation in these multipliers is done after the partial product generation stage, i.e., the approximation is performed while adding the partial products generated. Instead of using a conventional full adder or 4:2 compressor, approximate 4:2 compressors have been employed in the multiplier. A 7-bit Brent-Kung adder has been used in the final stage of the multiplier [5]. There are variety of 4:2 compressors proposed in [2]-[4]. The exact compressor [1] employs XOR-XNOR modules (referred to as XOR* modules) as shown in Fig. 1 to reduce power consumption. In this paper, XOR* module shown in Fig.1 has been employed for the accumulation of partial products to produce the final products.

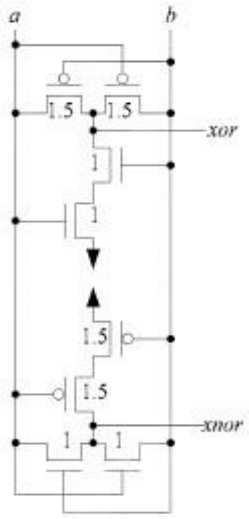

Fig. 1. XOR* module

The approximate multipliers are evaluated by designing an image compression method which employs Discrete Cosine Transform (DCT). The accuracy of these circuits are estimated by measuring the parameters such as, Peak Signal to Noise Ratio (PSNR), Error Rate (ER) and Normalized Error Distance (NED) [6] and comparison has also been made against each other.

The remainder of the paper is organized as follows. The approximate 4:2 compressors used for the evaluation of multipliers are presented in section 2. The multiplication and multiplier architecture are presented in section 3. Simulation results for the approximate multipliers and the comparison against exact multiplier are discussed in section 4. The image processing algorithm and simulation results of image processing algorithm using approximate multipliers are presented in section 5. Section 6 concludes.

II. APPROXIMATE 4-2 COMPRESSORS

Throughout the literatures referenced in this paper, the approximate (or inexact) compressors are designed by introducing some amount of error in the truth table of the exact 4-2 compressor. The error is introduced to reduce the logic complexity of the compressor and also to realize reduced delay and reduced power consumption. Design of 4:2 compressors [2]-[4] which have been used for comparative evaluation of multiplier has been presented in this section.

Revised Manuscript Received on December 16, 2019.
***** Correspondence Author
Balaji M , School of Electrical Engineering, VIT, Chennai – 600 127. Email: balajimurugan5598@gmail.com
Vishal Gundavarapu, School of Electrical Engineering, VIT, Chennai – 600 127. Email: g.vishal1998@gmail.com
Sasipriya P*, School of Electrical Engineering, VIT, Chennai – 600 127. Email: sasipriyap@yahoo.com
Kanchana Bhaaskaran V S, School of Electronics Engineering, VIT, Chennai – 600 127. Email: vskanchana@gmail.com

A. *Exact Compressor*

A generic 4:2 compressor, shown in Fig.2, consists of five inputs, i.e., four inputs *x1*, *x2*, *x3* and *x4* and one input

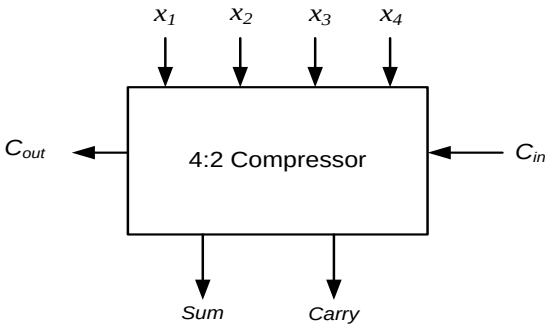

Fig.2. Exact compressor

C_{in}, which the compressor receives as a carry bit from another module that is one bit lower, and three outputs, S*um*, *Carry* and C_{out}. The *Carry* and C_{out} is passed to the next compressor modules of higher significance. 4:2 compressors have to be designed to satisfy the logic equation given by

$$x1 + x2 + x3 + x4 + C_{in} = sum + 2(\mathrm{C}arry + C_{out}) \quad (1)$$

Conventionally, a 4-2 compressor consists of two serially connected full adders. However, in this paper, the architecture with XOR modules [1] have been realized, where the structure of XOR* module is represented by the circuit shown in Fig.1.

B. *Design 1*

This design takes into consideration the fact that the C_{out} of an exact 4:2 compressor has the same value as C_{in} in 24 out of 32 states. Therefore, the design modifies the truth table values of the C_{out} to always be equal to C_{in}. However, since the C*arry* bit has a higher order, the error introduced would be too high. Therefore, this error is compensated by reducing the values of S*um* and C_{out}. In this design, the value of *Sum* in the second half of the truth table is reduced to 0, this introduction of error for the *Sum* value not only reduces the delay of the overall circuit but also reduces the total effective error produced by the first approximation. The new *Sum* is given by the following boolean equation:

$$Sum = \overline{C_{in}}(\overline{x1 \oplus x2}) + (\overline{x3 \oplus x4}) \quad (2)$$

Finally, the value of C_{out} is changed further to reduce the overall error of the circuit and also to simplify the design. The delay of the critical path of this design is 3Δ.

$$C_{out} = (\overline{\overline{x1x2} + \overline{x3x4}}) \quad (3)$$

Table I shows the truth table of this design and Fig.3 shows the gate level implementation of design 1. As observed from the truth table, this design has an error rate of 37.5%

C. *Design 2*

A second design proposed in [2] focus on further improvement in performance. Since the C_{out} and C*arry* have the same weight and therefore can be used interchangeably, C_{out} is always equal to C_{in}. As C_{in} is always zero in the first stage, therefore, C_{out} and C_{in} will be zero in all the stages. The boolean equations for S*um* and *Carry* are given as follows:

C_{in}	*x1*	*x2*	*x3*	*x4*	C_{out}	*Carry*	*Sum*
0	0	0	0	0	0	0	1
0	0	0	0	1	0	0	1
0	0	0	1	0	0	0	1
0	0	0	1	1	0	0	1
0	0	1	0	0	0	0	1
0	0	1	0	1	1	0	0
0	0	1	1	0	1	0	0
0	0	1	1	1	1	0	1
0	1	0	0	0	0	0	1
0	1	0	0	1	1	0	0
0	1	0	1	0	1	0	0
0	1	0	1	1	1	0	1
0	1	1	0	0	0	0	1
0	1	1	0	1	1	0	1
0	1	1	1	0	1	0	1
0	1	1	1	1	1	0	1
1	0	0	0	0	0	1	0
1	0	0	0	1	0	1	0
1	0	0	1	0	0	1	0
1	0	0	1	1	0	1	0
1	0	1	0	0	0	1	0
1	0	1	0	1	1	1	0
1	0	1	1	0	1	1	0
1	0	1	1	1	1	1	0
1	1	0	0	0	0	1	0
1	1	0	0	1	1	1	0
1	1	0	1	0	1	1	0
1	1	0	1	1	1	1	0
1	1	1	0	0	0	1	0
1	1	1	0	1	1	1	0
1	1	1	1	0	1	1	0
1	1	1	1	1	1	1	0

$$Sum = (\overline{x1 \oplus x2}) + (\overline{\mathrm{x}3 \oplus \mathrm{x}4}) \quad (4)$$

$$Carry = (\overline{\overline{x1x2} + \overline{x3x4}}) \quad (5)$$

Table- I: Truth table of approximate 4:2 compressor design 1

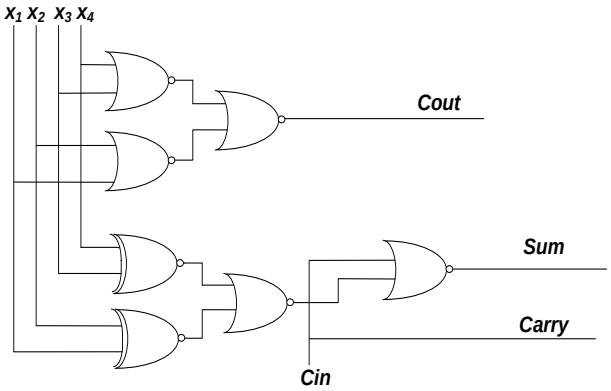

Fig. 3. Gate level implementation of Design 1

Fig.4 show the gate level implementation of this design and table II shows the truth table. This design has a critical path delay of 2Δ, which is lesser than that of design 1. It also has a reduction in the number of gates used.

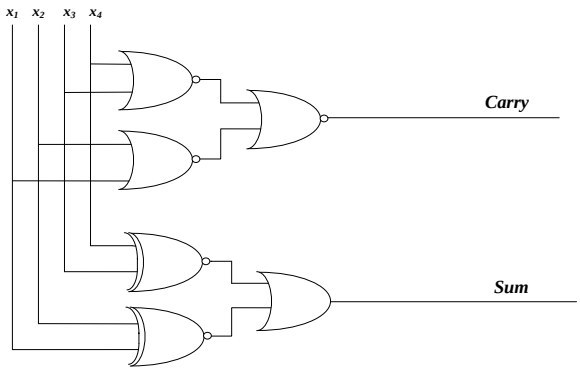

Fig. 4. Gate level implementation of Design 2

Table-II: Truth table of approximate 4:2 compressor design 2

x1	*x2*	*x3*	*x4*	*Carry*	*Sum*
0	0	0	0	0	1
0	0	0	1	0	1
0	0	1	0	0	1
0	0	1	1	0	1
0	1	0	0	0	1
0	1	0	1	1	0
0	1	1	0	1	0
0	1	1	1	1	1
1	0	0	0	0	1
1	0	0	1	1	0
1	0	1	0	1	0
1	0	1	1	1	1
1	1	0	0	0	1
1	1	0	1	1	1
1	1	1	0	1	1
1	1	1	1	1	1

D. Design 3

The design of approximate counter proposed in [3] employs MUX by replacing an XOR gate and therefore achieves the advantages of reduced delay and lesser power consumption.

Similar to design 2, S*um* and *Carry* are the two outputs produced in design3. In this case, the error is produced when all the four inputs have a value of 1, and the output here is reduced to 10_2 from 111_2. This error is intentionally introduced to generate a simplified algorithm, similar to the previous architectures. The gate level implementation of design 3 is shown in Fig. 5.

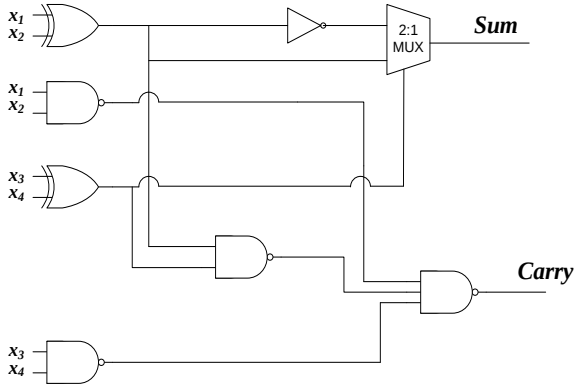

Fig. 5. Gate level implementation of Design 3

E. Designs 4, 5 and 6

The compressors proposed in [4] were designed under the constraint of achieving a low error rate. It has been assumed that the inputs to the multiplier are evenly distributed, therefore resulting in probability of the partial product being '0' is equal to 0.75. Similar to the previous cases, C_{in} and C_{out} are ignored here. When the inputs are all equal to 1, the output is equal to 100_2 in an accurate compressor, however, by approximating the truth table, the output in this case is modified to 11_2. Since *Carry* has a higher precedence, it is set to 1 when all inputs have a value of 1. The equation for *Sum* is given

$$Sum = (x1 \oplus x2) \oplus (x3 \oplus x4) = x1x2x3x4x \quad (6)$$

Here, the last term is due to the inputs '1111'. In order to reduce the complexity, S*um* is changed to '1' where the input is '0011', therefore, the Boolean expression for *Sum* in design 5 changes to:

$$Sum = (x1 \oplus x2) \oplus (x3 \oplus x4) + x3x4 \quad (7)$$

Table-III: Truth table of approximate 4:2 compressor design 4

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

Table- IV: Truth table of approximate 4:2 compressor design5

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

Table- V: Truth table of approximate 4:2 compressor design 6

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

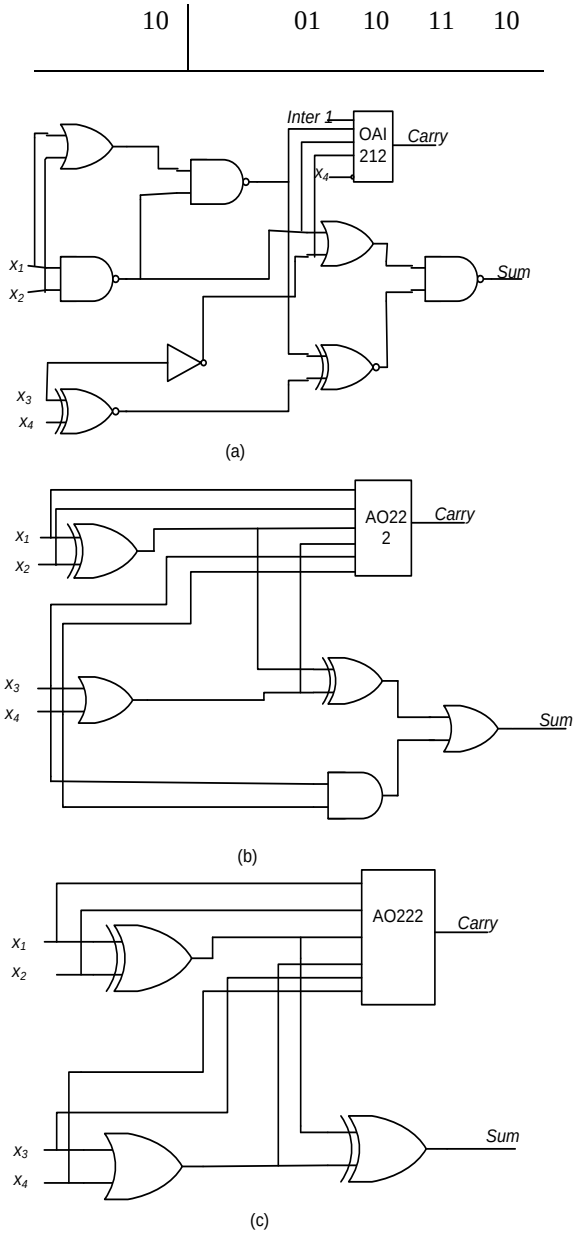

Fig. 6. Gate level implementation of (a) Design 4, (b) Design 5 and (c) Design 6

Further modification on design 5 is made to reduce the complexity by removing the last term in Sum equation of design 5. The truth table for designs 4, 5 and 6 is given in table IV, V and VI respectively. The gate level implementations are given in Fig.6 (a), (b) and (c) respectively.

III. DADDA MULTIPLIER

The architecture of the proposed multiplier consists of three stages, namely, i) the partial product generation by using AND gates, ii) accumulation of partial products by using compressors and iii) tree adder at the final stage for generating final product bits. The architecture of Dadda multiplier using compressor is shown in Fig. 7. The compressor shown in the Fig. 7 has been implemented with necessary modification based on the design of approximate compressor [2]. Thus, two Dadda multiplier circuits have been designed in this paper. The exact 4-2 compressors and design 1 of the approximate 4-2 compressor were employed in the first design, while the other compressors are used in the second design of Dadda multiplier. The implementation shows that second design occupies lesser area due to the reduced number of compressors.

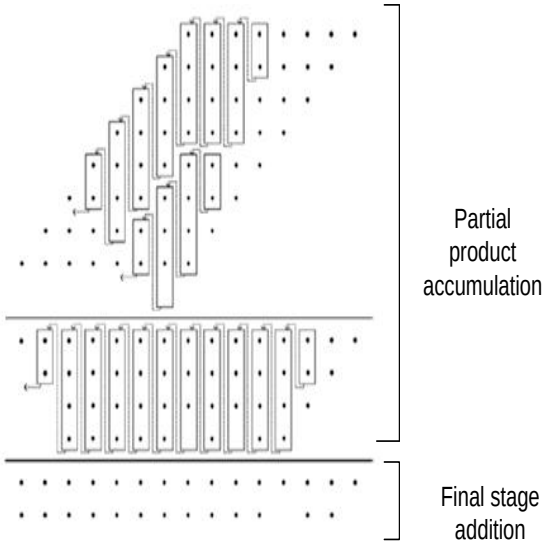

Fig. 7. Architecture of Dadda Multiplier

IV. SIMULATION RESULTS

The 8-bit Dadda multiplier has been designed using all the approximate compressors discussed in section II. All the multiplier circuits are designed using 45nm CMOS process technology and the simulations are carried out using Cadence Cadence® Virtuoso. The power consumed by each of the multiplier was measured and it was found that the multipliers designed using approximate compressors certainly did consume a lot less power than the exact multiplier. Table VI shows the power consumed by these multipliers operated at the 100MHz switching frequency.

Table- VI: Power consumption comparison of multipliers

Compressor used	Power consumed (µW)	Power Savings (%)
Exact	14.29	
Design 1 [2]	6.755	52.72
Design 2 [2]	5.808	59.35
Design 3 [3]	8.252	42.25
Design 4 [4]	7.593	46.86
Design 5 [4]	8.788	38.50
Design 6 [4]	9.022	36.86

V. APPLICATION IN IMAGE PROCESSING

The application of proposed approximate multipliers to image processing has been presented in this section. Image compression has been performed using all the multipliers. There are two types of image compression, namely, i) lossy and ii) loss-less compression. DCT is used in lossy image compression where the output image is reduced in quality compared to the input, contrary to loss-less compression in which the same input quality is got as output.

Lossy compression techniques have been used in this work. The multiplier employed in DCT which is usually done using accurate multipliers is substituted here with approximate multipliers. Thus, image compression is done by using these approximate multipliers to compute the direct cosine transform coefficients and then applying the inverse transform to get back the original image data. Fig. 8 shows the output images after inverse DCT. DCT has been realized in Matlab environment using the approximate multipliers.

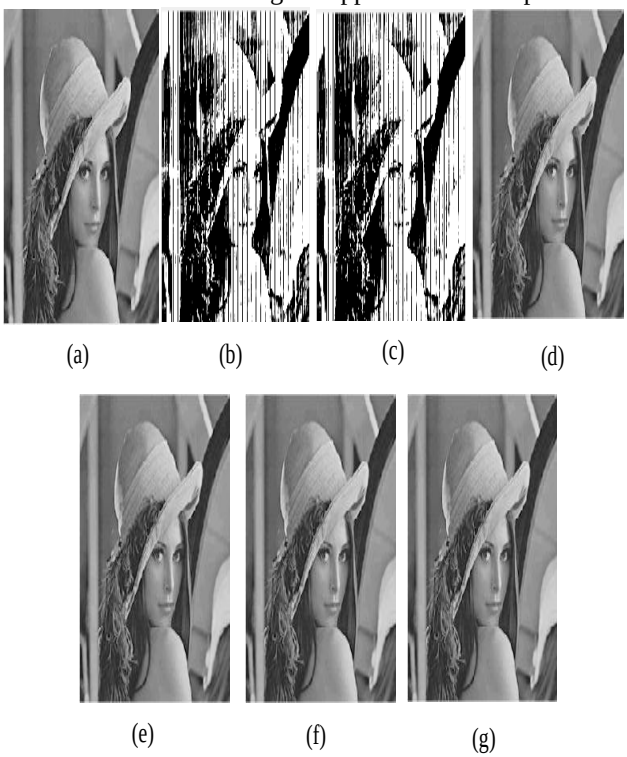

Fig. 8. Image compression results for (a) Exact multiplier, (b) Design 1, (c) Design 2, (d) Design 3, (e) Design 4, (f) Design 5 and (g) Design 6

Table- VII: PSNR and average NED for all the multipliers

Multiplier used	ER	NED	PSNR
Design 1	0.9978	0.0540	8.5609
Design 2	0.9913	0.0503	8.5576
Design 3	0.0888	0.0032	41.1808
Design 4	0.3294	0.0053	41.1602
Design 5	0.4038	0.0070	41.1723
Design 6	0.0888	0.0016	41.0289

The average NED and the PSNR [2] has been computed to evaluate the quality of the output image. The results for the image processing using all the approximate multipliers are shown in Table VII. The result shows that the design 1 and 2 has reduced power consumption when compared to the exact multiplier architecture. However, they have produced inaccurate outputs and have higher error rate.

VI. CONCLUSION

In this paper, six different architectures of 4-2 compressors are used in implementing a Dadda multiplier. The power consumption of these multipliers was computed and they were further used in image processing applications. Various factors such as PSNR, ER and NED were measured for the images that went through the image processing. It was found that the multiplier realized using design 2 had the most power reduction (59.35%). The multiplier made with design 3 had the highest PSNR value (41.1808) and the lowest error rate (0.0888). Design 6 had the lowest NED value (0.0016). However, in terms of overall performance, design 4 has an acceptable level of error while also maintaining a PSNR value similar to design 3 while consuming lesser power than design 3. Therefore, out of all the designs studied, multipliers made using design 4 have the most favorable results.

REFERENCES

1. C. Chang, J. Gu, M. Zhang, “Ultra Low-Voltage Low- Power CMOS 4-2 and 5-2 Compressors for Fast Arithmetic Circuits,” IEEE Transactions on Circuits & Systems, Vol. 51, No. 10, pp. 1985-1997, Oct. 2004.
2. A. Momeni, J. Han, P.Montuschi, and F. Lombardi, “Design and Analysis of Approximate Compressors for Multiplication”, IEEE Transactions on Computers, Volume: 64 , Issue: 4 , April 2015.
3. Chia-Hao Lin and Ing-Chao Lin, “High Accuracy Approximate Multiplier With Error Correction”, 2013 IEEE 31st International Conference on Computer Design, 6-9 October 2013.
4. Zhixi Yang, Jie Han, Fabrizio Lombardi, “Approximate Compressors for Error-Resilient Multiplier Design”, 2015 IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFTS), 12-14 October 2015.
5. Perneti Balasreekanth Reddy and V. S. Kanchana Bhaaskaran, “Design of Adiabatic Tree Adder Structures for Low Power”.
6. Jinghang Liang, Jie Han and Fabrizio Lombardi, “New Metrics for the Reliability of Approximate and Probabilistic Adders”, IEEE Transactions On Computers, 62(9), 1760-1771.
7. Transactions on Computers, Volume: 62 , Issue: 9 , Sept. 2013.

AUTHORS PROFILE

Balaji M is an Electrical and Electronics student from VIT Chennai. He has experience in Circuit design using Verilog.

Vishal Gundavarapu is a final year Electrical and Electronics student studying in VIT Chennai. He has experience in using Cadence Virtuoso for simulations for circuit design.

Sasipriya . P has 16 years of teaching and research experience. Her specializations include Low Power VLSI Circuit Design and System design using Microprocessors and controllers.

Kanchana Bhaaskaran.V.S is a Professor and Dean (Academics) at VIT University, Chennai. She obtained her undergraduation degree in Electronics and Communication Engineering from Institution of Engineers (India), Calcutta and her M.S. degree in Systems and Information from Birla Institute of Technology and Sciences, Pilani and Ph D from V I T University. She has more than 35 years of industry, research and teaching experience by serving the Department of Employment and Training, Government of Tamil Nadu, IIT Madras, Salem Cooperative Sugar Mills’ Polytechnic College, SSN College of Engineering and VIT University Chennai. Her specializations include Low Power VLSI Circuit Design, Microprocessor architectures and Linear Integrated Circuits.

She has published around 70 papers in International Journals and conferences, and has one patent published and one filed. She is a reviewer for peer reviewed international journals and conferences. She is the Fellow of the Institution of Engineers (India), Fellow of the Institution of Electronics and Telecommunication Engineers, Life Member of the Indian Society for Technical Education and Senior Member of the Institute of Electrical and Electronics Engineers Inc., USA.