International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878 (Online), Volume-8 Issue-4, November 2019

FPGA Implementation of Area Efficient CMOS Multiplier using Fast Kogge Stone Look Ahead Logarithmic Adder

L.Malathi, A.Bharathi, A.N.Jayanthi

Abstract*: Though Multipliers play a major role in all digital processing systems, still there is a research challenge related with area, delay, power, speed and accuracy parameters. Basically multipliers contains more number of adders (i.e.,) multiplication is done by repetitive additions. Highest care should be taken on adders. Partial Products (PP) part is middle process between multiplier, multiplicand and final addition. Next one about the methodology that Serial/parallel, Pipeline/parallel, Floating/decimal, Array, Binary/BCD, Fixed/Variable, Gate Level/Transistor Level. All the predecessors are having more controversy parameters. The forthcoming research concentrated on parallel, pipelining, decimal, binary and transistor level.

Keywords* : Kogge Stone Adder, Flagged Perfix Adder, Area, Delay, Power.

I. INTRODUCTION

T he time taken by serial multiplier is very high when Compared to parallel multiplier to perform multiplication process. It will be calculated in propagation delay analysis. Secondly, pipeline process gives the high speed compared to parallel. Array type will be save the process time. Thirdly, while shifting and rotating process binary may be preferred over Binary Coded Decimal (BCD). The usage of buffers and registers will be comparatively reduced. In transistor level design, transistor size can be reduced moderately.

Manuscript published on November 30, 2019.
* Correspondence Author
L.Malathi*, Assistant Professor, Department of Electronics and Communication Engineering Sri Ramakrishna Institute of Technology, Coimbatore, India, lmalathigraj@gmail.com
Dr.A.Bharathi, Professor & Head, Department of Information Technology Bannari Amman Institute of Technology, Erode, India, bharathia@bitsathy.ac.in
Dr.A.N.Jayanthi, Associate Professor, Department of Electronics and Communication Engineering Sri Ramakrishna Institute of Technology, Coimbatore, India, jayanthi.ece@srit.org

II. MULTIPLIER

A. Partial Product

A and B are the multiplicand and multiplier. If the inputs are 6-bits, in order to get final multiplied value, it consists almost six steps of partial products. Which means the number of partial product steps highly depends on the multiplicand and multiplier.

A = (an, a(n-1), a(n-2)……a0) ------ (1)

B = (bn, b(n-1), b(n-2)……b0) ------ (2)

A.B= (A.2n bn+A.2(n-1) b(n-1) + A.2(n-2). b(n-2) .. A.20b0) ----- (3)

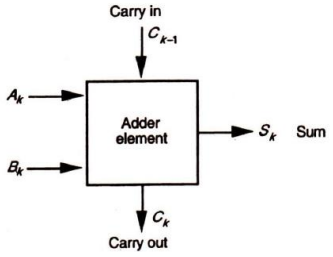

Fig.1. n-bit Adder

A_k and B_k are the inputs that is given to adder element block. Carry input is $C_{(k-1)}$ and the carry output is C_k. The sum of the input is mathemeticlly writtern as,

$$S_k = A_k + B_k \quad \text{------ (4)}$$

In the adder, for 4-bit input equivalent 4-bit output will produced with a bit of carry. According to the adder two cases are here to notice that, one is with carry and other without carry.

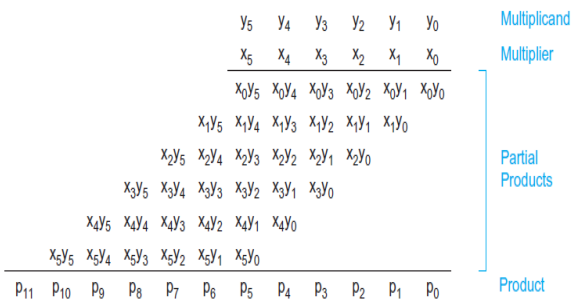

Fig.2. General Process of Multiplication

In Fig.3 one of the input is Y0…Y5 and another one of the input is X0…X5. While the product of Y{0 to 5} and X{0 to 5} is P0…P11. Totally six steps of partial products is produced for final output. All the partial products are added in columnwise manner. For this addition full adder and half adders ares used.

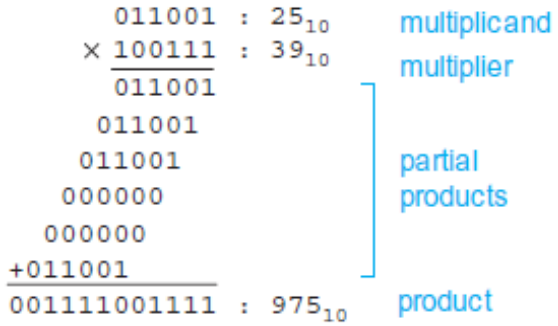

Fig.3. Simple Example of Multiplication

The addtion value is 975 when the inputs are 25 and 39. In this example carry is 0. In carry save adder that carry is save and propogate to next level. Carry select adder case among the n-bit inputs the particular carry is alone transferred to next stage. Carry propogate adder will thrugh the carry over to adjacent levels with the suitable analysis and it should satisfy the delay requirements.

$$PP = \left(\sum_{j=0}^{M-1} y_j 2^j\right)\left(\sum_{i=0}^{N-1} x_i 2^i\right) = \sum_{i=0}^{N-1}\sum_{j=0}^{M-1} x_i y_j 2^{i+j} \quad \text{------ (5)}$$

$$Si = Pi \text{ xor } G(i-1):0 \quad \text{------ (6)}$$

Eq.(5) depicts the partial product of x and y. In that propagate (eq.(6)) and generate results are done by xor operation.

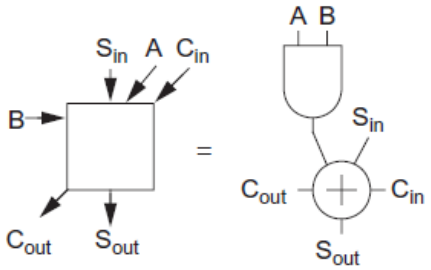

Fig.4. Array Multiplier

Apart from kogge stone adders, the various types of adders were already undergone by the researchers. In order to reduce delay, area and power of adders and its general parameters. Fig.4 depicts the schematic view of array multiplier. A, B, $Carry_{in}$, Sum_{in} are inputs. S_{out} is output. By using adder cin and sin will added. Actually user input is A and B. remaining all of the other parameters will generated by array multiplier block.

B. Kogge Stone Adder

The simple kogge stone adder having 4-bit input and 4-bit output. The carry is propogated. If carry is discarded, then separate block will used to collect those generated carry for further processing.

The main drawback in kogge stone adder is area. The area is obviously enhanced because of its structural fects. But it is having lot of advantages to its equal drawback. Hence the acceptable drawback is considerably reduced in the proposed technique.

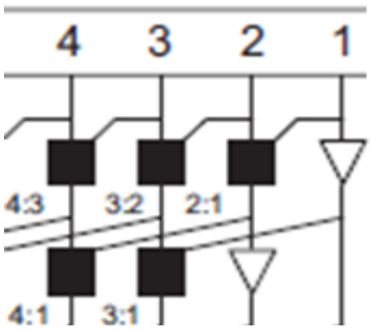

Fig.5. 4-bit Kogge Stone Adder

There are so many adders are there to perform addition. The main moto to select the Kogge-Stone adder as it is having lower fanout in each and every stage in partial product portion. Automatically it increases the performance.

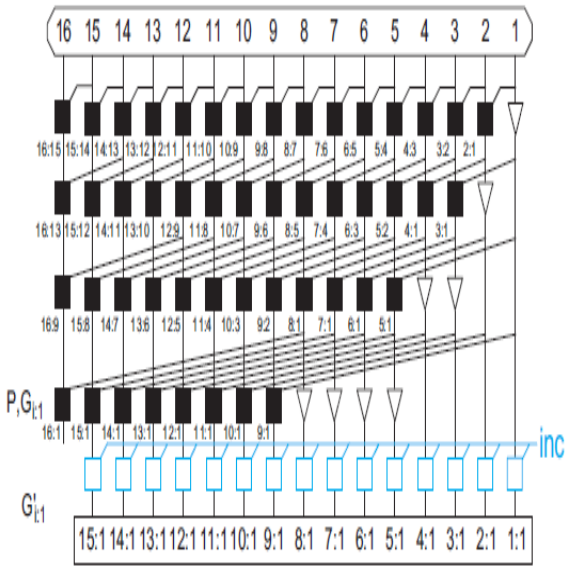

Fig.6. 16-bit Kogge Stone Adder

In 16-bit adder, four stages are generated before the output stage. From 9:1 to 16:1 outputs at stage 1 gives P, G(k-1) for processing. At stage 1 remaining eight outputs preceding of 9:1 four output will be directly propagated from second stage 5:1 to 8:1. From stage 3, the outputs 4:1 and 3:1 are propagated. Output 2:1 is directly from stage 4 and 1:1 is direct input in architecture.

Table - I: Propogation – Stagewise (16-bit)

Stage	*Propogated Outputs*
1	16:1 to 9:1
2	16:9 to 5:1
3	16:13 to 3:1
4	16:15 to 2:1

In the Table.1. input 1:1 is directly propagated but th impact of that input is play a role till 16:1.

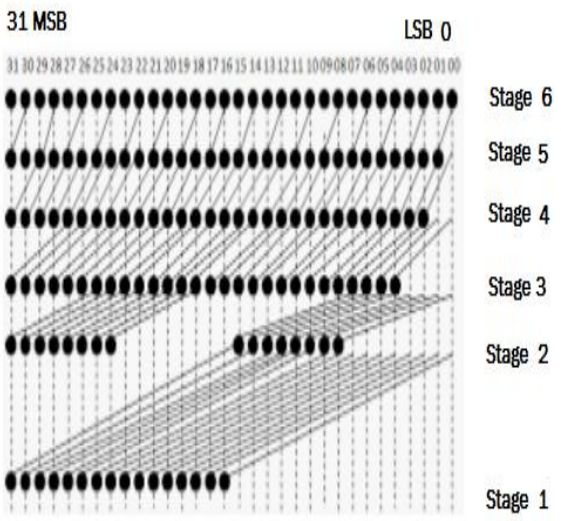

Fig.7. 32-bit Kogge Stone Adder

In 32-bit kogge stone adder six stages are having to produce an output.

Table - II: Propogation – Stagewise (32-bit)

Stage	*Propogated Outputs*
1	16 to 31
2	6 to 15 and 24 to 31
3	4 to 31
4	2 to 31
5	1 to 31
6	0 t0 31

While compare Table-I (16-bit) and Table-II (32-bit) two more stages for 32-bit. When the bit is increases from 16 to 31 two stages are increased. This is the main advantage of this adder. Direct propogation via buffers are [stage-1, 0 to 15 from stage-2], [stage-2, 16 to 23 from stage-, 4 to 7 from stage-3, 3 and 2 from stage-4, 1 from stage-5, 0 from stgae-6], [stage-3 4 to 31, 3 and 2 from stage-4, 1 from stage-5, 0 from stgae-6] , [stage-4 2 to 31, 1 from stage-5, 0 from stgae-6], [stage-5 1 to 31, 0 from stgae-6]

Table - III: Buffers – Stagewise

Stage	*No.of buffers used (32-bit)*	*No.of buffers used (16-bit)*
1	16	4,4
2	16	2,2
3	4	1,1
4	2	1
5	1	-
6	0	-

Number of propogated outputs equal to number of adders used.

III. PROPOSED DESIGN

In Fig.(5,6,7). 4-bit five adders and two buffers are required. 16-bit adders 49 adders and 8 buffers are required. 32-bit adders 151 adders are required. Basically if we reduce the delay of half and full adder automatically it will pull down the delay, power and area.

Kogge-Stone structure is very attractive for high-speed applications. The delay of the structure is given by 2 log n . The Kogge-Stone scheme addresses the problem of fanout by introducing a recursive doubling algorithm [6]. It uses idempotency property to limit the lateral fan-out, but at the cost of a dramatic increase in the number of lateral wires at each stage.

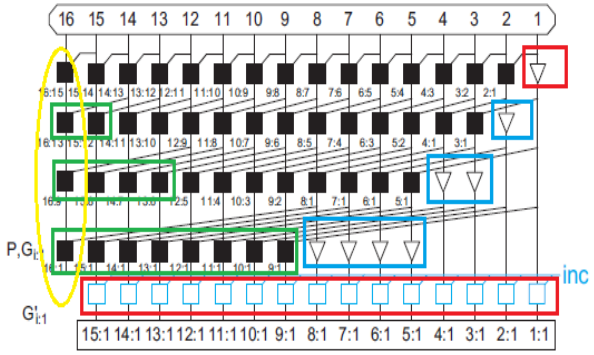

Fig.8. 16-bit Kogge Stone Adder – Directly Propogted to Adjacent Stage

In Fig.8. from stage-2 31:16 outputs are directly propogated to stage-1, from stage-3 19:8 outputs are directly propogated to stage-2, from stage-4 30:31 outputs are directly propogated to stage-5, from stage-6 31output is directly propogated to stage-5.

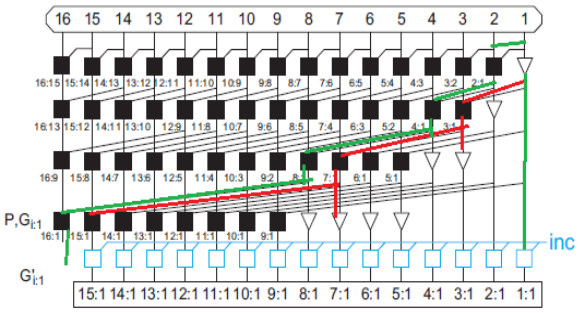

Fig.9. 16-bit Kogge Stone Adder – Path - Impact of 1:1

In Fig.9. input 1:1 is propogated till 16:1. LSB of output is LSB of input with buffer.

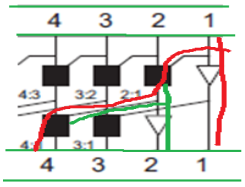

Fig.10. 4-bit Kogge Stone Adder-Impact of input bits

In Fig.(9,10) shows that each input is impact with throughout the PP for output. The proposed architectures have the least number of computation nodes when compared with existing one's. This reduction in hardware of the proposed architectures helps to reap a benefit in the form of reduced power and power-delay product. This is because there, is a massive overlap between the prefix sub-terms being pre-computed.

This reduction in delay time can be achieved by using a parallel prefix adder network (PPA) [10]. This parallel prefix adder is inserted in the middle stages of RCA chain. The Kogge-Stone adder is used for this parallel prefix network.

In addition to this, the power consumption of this structure is reduced by using AND-Or-Invert and Or-And-Invert gates for skip logics. An incrementation scheme increases the speed of the structure. These two modifications make this modified carry skip adder more efficient. Analyzing from the last stage gives us a much better understanding of the redundant cells. In the last stage there are no redundant cells as it contains only grey cells and hence none of them can be removed.

Kogge-Stone adders are popular choice in high speed ALU design due to its faster operation, regular structure and balanced loading in internal nodes compared to other sparse tree adders. In this section, first we briefly discuss the design, operation and general properties that are relevant from fault tolerance point of view. Next, we elaborate the impact of faults in the intermediate computations and their effect on the overall Sum generation.

IV. REVIEW CRITERIA

Area of 130nm, 90nm, 65nm and 40 nm technologies [7] are based on the simulation reviews. The Brent Kung Adder exhibit the best performance in terms of area and power consumption for both the 32 and 64 bit adder categories. Han-Carlson is a parallel prefix adder and is formed by the combination of Kogge-Stone having an advantage of providing higher speed and Brent-Kung adder having an advantage of providing low area [6].

Architecture Logic Levels Speed (ns): [11] Ripple Carry (N-1) ,31.744, Brent-Kung $2(\log_2(N-1))$, 19.059, Han-Carlson $\text{Log}_2(N+1)$, 16.943, Sklansky Log_2N. 15.604, Kogge-Stone Log_2N, 15.160, Kogge-Stone (Rerouting) Log_2N, 15.017 Kogge-Stone (Reducing Black Cells), Log_2N 13.667.

V. COMPARISION

Table - IV: Comparison of Adders [2] [6]

Adder	Area (µm2)	Total Power (µW)	Delay (ps)
Brent Kung	329.83	20.93	522
Han Carlson	366.05	22.48	444
Knowles	502.15	27.35	428.8
Lander Fischer	335.87	21.17	458.6
Sklansky	366.05	22.60	429.6
Kogge Stone	502.16	27.35	428

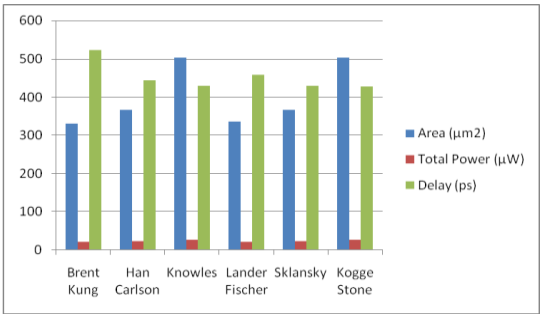

Fig.11. Performance of Adders

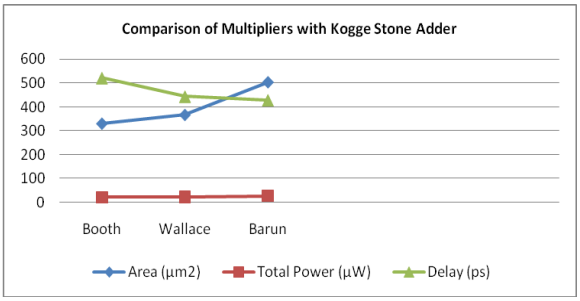

Fig.12. Parameter Analysis of Adders

VI. SIMULATION RESULTS

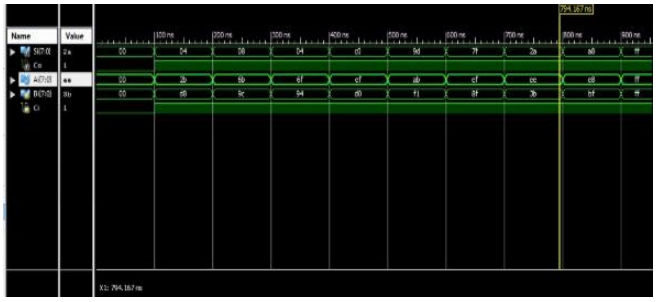

Fig.13. Simulation of Kogge Stone Adder

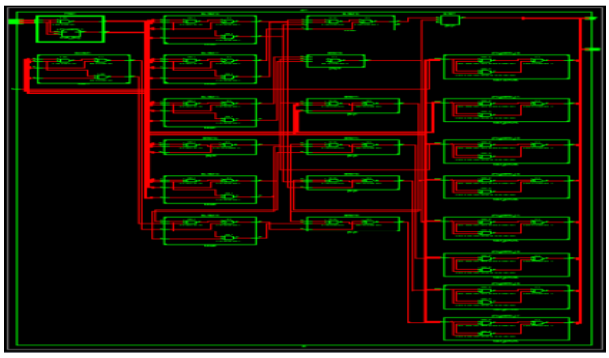

Fig.14. RTL - Kogge Stone Adder

VII. DISCUSSION

Recent VLSI design, the occurrence of delays is predictable. Xilinx ISE is used for simulation and synthesis Delay of 13.88 ns [6] for a 16 bit Ling adder and 64 bit Sparse 2 adder has a delay of 35.026 ns [8].

Area is also measured and comparison is made. Several techniques have been proposed in past to tolerate various kinds of defects in arithmetic and logic circuits. In [1], the authors isolate the critical paths of random logic circuits by proper synthesis and sizing.

VIII. CONCLUSION

Proposed modified fast architectures for 4-bit, 16-bit, 32-bit and 64-bits is proposed that offers minimal power dissipation and least power-delay product. The proposed structure achieves 3% to 7% power savings and 15% to 35% improvement in speed compared with Brent Kung adder [4] which has nearly same logic depth and number of computation nodes.

REFERENCES

1. Aribam Balarampyari Devi, Manoj Kumar, Romesh Laishram,"Design and Implementation of an improved carry increment adder, "International Journal of VLSI design and Communication Systems(VLSICS)vol.7, NO.1, February 2016.
2. Dubey, N., & Akashe,S.(2014)"Implementation of An Arithmetic Logic Using Area Efficienyt Carry-Look-Ahead Adder"'Int J VLSI Des Commun Syst(VLSICS),Vol.5, No.6, pp29.
3. Deepa Sinha, Tripti Sharma, K.G.Sharma, B.P. Singh, "Design and Analysis of low Powetr 1-bit Full Adder Cell",IEEE,2011.
4. Themozhi.G, And Themozhi V."Propagation Delay Based Comparison Of Parallel Adders,"Journal of Theoretical and Applied Information Technology,vol.67,pp.306-315,September 2014.
5. Maroju Saikumar, and Dr. P. Samundiswary,"Design and Performance Analysis of various Adders using Verilog," International Journal of Computer Science and Mobile Computing, Vol.2, pp. 128-138, September 2013.

6. S.Manju and V.Somagopal,"An efficient SQRT architecture of carry select adder design by common Boolean logic," in Proc. VLSI ICEVENT, 2013,PP.1-5.
7. Nabihah Ahmad, Rezaul Hasan,"A new Design of XOR-XNOR gates for Low Power application"International Conference on Electronic Devices, System and Applications(ICEDSA),2011.
8. Laxmi Kumre, Ajay Somkumar and Ganga Agnihotri, (2013) "Power Efficient Carry Propogate Adder" Int J VLAI Des Commun Syst(VLSICS), Vol.4, No.3.
9. K.K.Parhi,VLSI Digital Signal Processing Newyork, NY, USA:Wiley, 1998.
10. B.Ramkumar and H.M. Kittur, "Low-Power and area-efficient carryselect adder,"IEEE Trans. Very Large Scale Integr.(VLSI)Syst, Vol.20 No.2, pp.371-375, Feb 2012.
11. Rupashree Sahu, Asit Kumar Subudhi "An area optimized Carry Select Adder",2015 IEEE Power, Communication and Information Technology Conference(PCITC),2015.
12. Shubham Sarkar, Sujan Sarkari, Ayan Dutta, Jisha Mehedi. "Power and Area Aware Improved SQRT Carry Select Adder(CSIA)",2018 2nd International Conference in Electronics and Information (ICOEI), 2018.
13. Bhavani Koyada, N. Meghana, Md. Omair Jaleel, Praneet Raj Jeripotula."A comparative study on adders", 2017 International Conference on Computer Communication and Informatics (ICCCI), 2018.
14. S. Rakesh, K.S. Vijula Grace."A Comprehensive review on the VLSI design performance of different Parallel Prefix Adders", Materials Today: Proceedings, 2019.
15. Reshma Mary James, Ajith Ravindran, "Review of Full Adder Performance Analysis Using Kogge Stone Adder and Magnetic Tunnel Junction", 2018 4th International Conference on Devices, Circuits and Systems (ICDCS), Yea-: 2018.

ACKNOWLEDGMENT

The authors thank the Management and Principal of Sri Ramakrishna Institute of Technology, Coimbatore, Tamilnadu, India for providing excellent computing facility and encouragement.

AUTHORS PROFILE

Ms. L. Malathi, She received her B.E. degree in Electronics and Communication Engineering in the year 2005 and M.E. degree in Applied Electronics from Anna University, Chennai in the year 2008. Currently she is working as Assistant Professor (Sr. Gr.) in Department of Electronics and Communication Engineering at Sri Ramakrishna Institute of Technology, Coimbatore, India. She is having 11 years of teaching experience. She has presented papers in various National and International Conferences & Journals. Her field of interest is VLSI and Digital Signal Processing. She is the Life Time Member ISTE, IAENG and ACM. She is currently pursuing Ph.D. in the area of VLSI, Anna University, Chennai.

Dr. A. Bharathi, She received her Ph.D degree in 2014 in Faculty of Information and Communication Engineering, Anna University, Chennai in the year 2007. She did her Bachelor's degree at Kongu Engineering College, Perundurai in Computer Science and Engineering under Bharathiar University, Coimbatore in the year 1998. She then completed her Post Graduate Degree at Bannari Amman Institute of Technology in Computer Science and Engineering under Anna University, Chennai, in the year 2007. She finished her Doctoral degree in Information and Communication Engineering specializing in Data Mining. She is the Life Time Member of ISTE. She has over 21 years of teaching experience. She got Awards and Patent. She has delivered Guest Lectures in various institutions.

Dr. A.N. Jayanthi, She received her Ph.D degree in 2014 in Faculty of Information and Communication Engineering, Anna University, Chennai. She received her M.E degree in VLSI Design from Government College of Technology , Coimbatore, Anna university, Chennai in the year 2008 and her B.E degree in Electronics and Communication Engineering from Sri Ramakrishna Engineering College, Coimbatore under Bharathiar University ,Coimbatore in the year 1998. She is having 21 years of teaching experience .Currently she is working as Associate Professor in the Department of Electronics and Communication Engineering at Sri Ramakrishna Institute of Technology, Coimbatore. Her area of specialization in Ph.D. is VLSI Design. She has published many research papers in referred National and International Journals & Conferences. She is a Life Time Member of ISTE