International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878, Volume-8 Issue-3, September 2019

Magnitude Comparator Realization using Threshold Logic

Soumya R, V. Sureh Babu, Varun P. Gopi

Abstract*: Threshold logic design can be regarded as an alternative method of traditional logic gate design. Threshold logic can be considered as the primary logic of brain. Neurons can be considered as threshold logic gates. Each neuron fires when its activation threshold is crossed. The basic idea behind threshold logic is threshold decision principle. Threshold decision principle checks whether the weighted sum of inputs crosses a threshold and accordingly output is defined. Several implementations for threshold logic were proposed. Resistive threshold logic is a new method for threshold logic implementation. In this paper, using the concept of resistive threshold logic, a magnitude comparator is realized. LTSpiceIV and Electric VLSI Design System are the simulation tools used in this work.

Keywords *: Comparator, digital circuits, memristors, threshold logic.*

I. INTRODUCTION

The idea of threshold logic was introduced many years ago in about 1960's. Threshold logic primarily focuses on setting threshold values and checks whether the weighted sum of inputs crosses the threshold or not and accordingly output is defined. Muroga in [1] describes the details of threshold functions which form the basis of threshold logic. [1] Also compares majority element and a threshold element. A threshold element is similar to majority element except that threshold element has two asymmetrical values 0 and 1, whereas majority element has two symmetrical values 1 and -1. Shuzo [2] extensively studied the properties of completely monotonic functions in order to clarify the basic properties of threshold functions. Shuzo in [3] used the idea of completely monotonic functions to realize any arbitrary logic functions. The method introduced by Shuzo worked well for completely monotonic functions and hence for threshold functions since both were closely related. This idea was of great importance since any arbitrary function could be realized using the concept of threshold logic.

Manuscript published on 30 September 2019
* Correspondence Author
Soumya.R, V. Sureh Babu*, Dept. of Electronics and Communication Engineering College of Engineering Trivandrum, Email: soumyarenjit1991@gmail.com, vsb@ece.ac.in
Varun P. Gopi, Dept. of Electronics and Communication Engineering National Institute of Technology Tiruchirappalli, India. Email: varun@nitt.edu

Threshold logic gates form the heart of threshold logic. The difference between threshold gates and conventional logic gates is that a single threshold gate can be made to perform different functions but it is not possible for conventional logic gates. The main highlight of threshold logic is the ability to implement complex functions in an efficient way. Conventional realizations become difficult when the number of input variables increases and in those situations threshold logic has great significance.

The concept of threshold logic has been known for a long time, but they were not a part of mainstream digital design due to lack of efficient implementations for threshold logic. The earlier concepts of threshold logic concentrated on analyzing the theoretical aspects and little attention was paid to synthesis and optimization of threshold networks. The lack of efficient implementations for threshold logic when compared to static fully complementary MOS techniques led to loss of interest in threshold logic. Several implementations were developed for threshold logic and [4] gives the details of these implementations. These implementations are differing on the basis of how inputs are weighted and how the thresholding operation is done. Low power threshold gate [5], Current mode threshold gate [6], Charge recycling threshold logic [7], are some of the implementations. [8] Introduces a new method for threshold logic implementation called Resistive Threshold Logic.

Threshold logic can be used to implement logic circuits. Quintana [9] proposed threshold implementation of a multiplier which improved implementation costs significantly. Threshold logic based design of compressors [10] was proposed in 2002 and such compressors showed better performance in delay and power-delay product when compared to conventional implementations.

In this paper, a magnitude comparator is realized using the idea of resistive threshold logic and its performance is evaluated. The simulations are carried out using LTSpiceIV and Electric VLSI Design System

II. BACKGROUND

A. Resistive Threshold Logic

A new method for threshold logic implementation is resistive threshold logic [8]. Basically a threshold logic unit must contain an input weighting part and a thresholding part. The weighting part weighs the input and calculates the weighted sum of inputs. The thresholding part fixes a value for threshold and sets output to value one, when the weighted sum exceeds threshold otherwise output value is set to zero. Different implementations for threshold gates depend on how the inputs are weighted and how thresholding is done.

The resistive threshold logic uses a resistive network for weighing the inputs. This logic has the ability to implement large variable functions efficiently, which is considered as an advantage.

Fig. 1 shows the basic cell of resistive threshold logic. It consists of a resistive divider network and a variable threshold inverter. The inputs given are weighted by the resistive network. The inputs to the cell are binary values and the resistive network calculates weighted sum of inputs. The threshold voltage of the variable threshold inverter can be varied to make the cell perform different functions. Here, V_1 and V_2 are the inputs V_o is the output of resistive divider and V_{OUT} is the output of the threshold cell. The output of resistive divider is given to a variable threshold inverter.

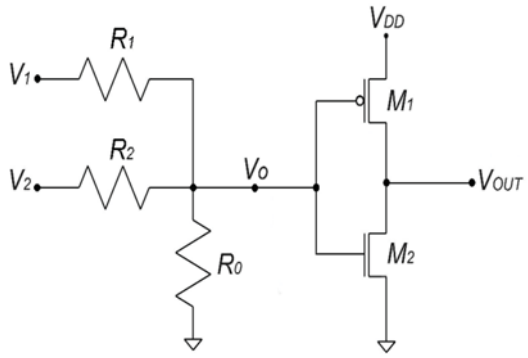

Fig. 1. Basic cell of Resistive Threshold Logic

The inverter threshold can be varied to make the cell perform different functions. The cell can work either as a NAND gate or as a NOR gate by setting different values to the inverter threshold. This forms one of the main ideas of threshold logic, i.e., the same circuit can be made to perform the functions of NAND and NOR gates by simply varying the threshold of the inverter.

The resistive divider has N resistors R_i and a reference resistor R_o. The inputs to the resistive divider V_i are binary inputs which takes values either 1 or 0. The output of resistive divider network V_o is given as,

$$V_o = \frac{\sum_{i=1}^{N}(V_i/R_i)}{(1/R_o)+\sum_{i=1}^{N}(1/R_i)} \quad (1)$$

If equal values are kept for all resistors R_i and if R_o=mR_i then the equation becomes,

$$V_o = \frac{\sum_{i=1}^{N} V_i}{(1/m)+N} \quad (2)$$

Resistive part does the weighing of inputs. Semiconductor resistors have leakage current issue and hence memristors are used for implementing the resistive part, which has negligible leakage current. Memristor can be considered as a fourth basic circuit element. Memristor is a contraction for memory resistor. Memristor behavior could not be obtained by any combination of other three existing fundamental elements (resistor, capacitor, and inductor), even of non-linear types and hence memristor is considered as fundamental element. Memristor gives the missing relationship between flux and charge. Chua [11] formulated the memristor theory. Hewlett-Packard (HP) laboratories gave the first physical model of a memristor. In this paper, memristor model as described in [12] is employed.

B. Realization of NAND and NOR using Resistive Threshold Logic

NAND and NOR gates are considered as universal gates in digital circuits since other gates like AND, OR, NOT, XOR etc can be implemented using NAND and NOR only. Resistive threshold basic cell can be made to perform NAND and NOR operations by varying threshold of inverter. Table I shows the truth table for two input NAND and NOR gates. The output voltage is calculated using (1). By setting inverter switching threshold between (1/3) and (2/3) the basic cell in Fig.1 works as a NAND gate and by setting inverter switching between 0 and (1/3) the basic cell functions as a NOR gate. Thus two functions can be accomplished in same structure by just changing the value of inverter threshold. Thus separate structures are not needed for implementing NAND and NOR gates and the same structure can perform both functions.

Table-I: Truth table of two input resistive threshold logic.

V_1	V_2	Output voltage V_o	NAND	NOR
0	0	0	1	1
0	1	(1/3)	1	0
1	0	(1/3)	1	0
1	1	(2/3)	0	0

III. MAGNITUDE COMPARATOR REALISATION

Data comparison is needed in digital systems while performing arithmetic or logical operations. This comparison determines whether one number is greater than, equal to or less than the other number. A digital comparator is widely used in combinational system and is specially designed to compare the relative magnitudes of binary numbers.

A digital magnitude comparator is a combinational circuit that compares two digital or binary numbers (A and B) and determines their relative magnitudes in order to find out whether one number is equal to, less than or greater than the other digital number. Magnitude comparator forms the basis of decision making in logic circuits. Comparators are used in Central Processing Units (CPUs) and microcontrollers (MCUs). Comparators have many applications. They are used in the address decoding circuitry in computers and microprocessor based devices to select a particular input or output device. Comparators are also used in control applications where they are used for comparing binary numbers representing physical variables with reference values Fig.2 shows a magnitude comparator. It has two binary inputs A and B and three binary outputs A>B, A=B, A<B. It compares the two numbers and establishes the relation between them. Comparators can be 1 bit, 2 bit etc depending on the no. of bits present in the inputs.

The three outputs A>B, A=B, A<B indicates the relation between two inputs A and B.

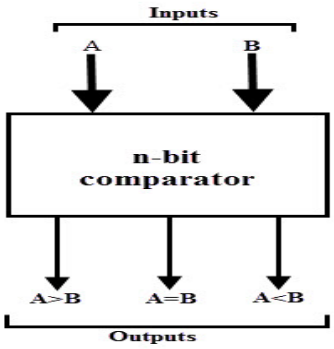

Fig. 2. Basic Comparator

IV. DESIGN AND SIMULATION OF MAGNITUDE COMPARATOR USING RESISTIVE THRESHOLD LOGIC

A. Design of One bit Magnitude Comparator

A one-bit comparator takes binary inputs that are one bit wide and produces three binary outputs to check whether the one input is equal to, less than or greater than the other input. Let A and B be the two one bit inputs. Fig.3 shows a one-bit comparator.

Fig. 3. One bit magnitude comparator

Table II shows the truth table of comparator.

Table-II: Truth table of one bit magnitude comparator

A	B	A>B	A=B	A<B
0	0	0	1	0
0	1	0	0	1
1	0	1	0	0
1	1	0	1	0

From truth table, output equations are obtained.
For A<B the equation is A'B
For A=B the equation is AB+A'B'=AʘB
For A>B the equations are AB'.
These equations are implemented with resistive threshold logic to realize the one-bit magnitude comparator.
Spice simulation circuit for the proposed resistive threshold logic realization of one-bit magnitude comparator is shown in Fig.4. Resistive threshold logic with memristors forming the input weighing part is used for realization of comparator. Memristor is modeled as in [10].

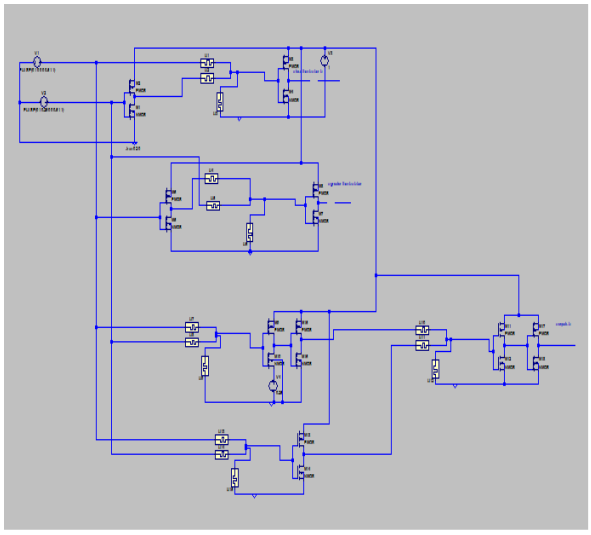

Fig. 4. Proposed resistive threshold logic realization of one bit magnitude comparator

The inputs applied to one bit magnitude comparator and the corresponding outputs are shown in Fig.5

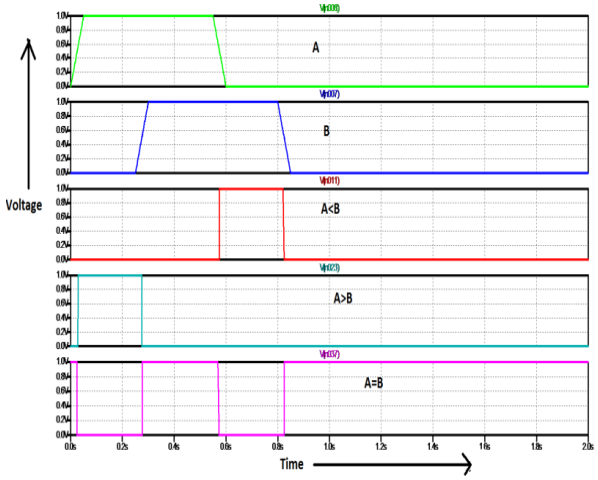

Fig. 5. Inputs and corresponding outputs of one bit magnitude comparator

B. Design of two bit Magnitude Comparator

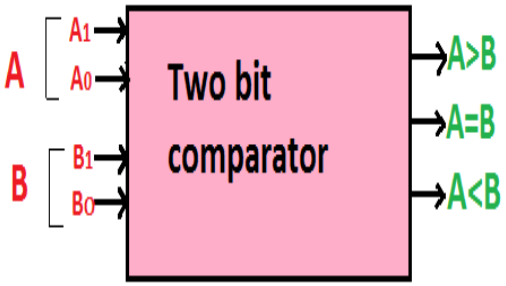

Fig. 6. Two bit magnitude comparator

Two-bit comparator is similar to that of one-bit comparator except that each input is composed of two bits. Input A is composed of bits A_1A_0 and input B is composed of bits B_1B_0. Fig.6 shows a two-bit magnitude comparator. The outputs are A>B, A=B, A<B.
The output equations are:
For A>B → $A_0B_1'B_0' + A_1B_1' + A_1A_0B_0'$
For A<B→$A_1'A_0'B_0 + A_1'B_1 + A_0'B_1B_0$
For A=B→ $(A_0ʘB_0)\ (A_1ʘB_1)$.

Fig.7 shows the circuit diagram of resistive threshold implementation of two-bit magnitude comparator. The circuit complexity of resistive threshold implementation is less compared to conventional CMOS implementation. Two-bit magnitude comparator realization requires fewer transistors compared to that of conventional CMOS logic and this leads to reduction in implementation area.

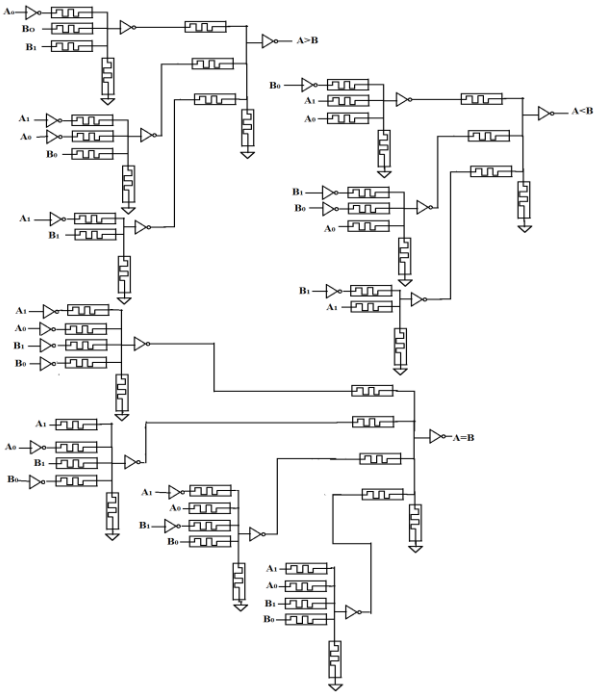

Fig. 7. Circuit diagram of two bit magnitude comparator implementation using resistive threshold logic.

V. RESULT ANALYSIS

Resistive threshold realization of magnitude comparator can be compared with conventional CMOS implementation. Comparison can be made for transistor count, circuit complexity, implementation area and power dissipation.

A. Transistor Count Comparison

Transistor count refers to the number of transistors used for circuit implementation. A comparison can be made about the transistor count required for resistive threshold logic and conventional CMOS logic. Table III shows the comparison of transistor count. There is a significant reduction in transistor count for resistive threshold logic and this reduction becomes prominent as the number of input bits increases. In case of resistive threshold logic, the transistor count does not increase significantly in comparison to the conventional CMOS logic. The complexity of circuits will also be reduced to a great extent for resistive threshold logic based implementation.

Fig.8 shows the comparison between transistor count required for resistive threshold implementation and conventional CMOS implementation of magnitude comparators. It can be seen from the graph that the transistor count required for threshold logic is significantly smaller compared to conventional CMOS implementation. This reduction will tend to reduce the implementation area and hence the implementation costs. As the input bits of comparator increases, the transistor count required for threshold based implementation does not show a progressive increase. This signifies the fact that the threshold logic can be used to implement complex circuits in a much efficient and effective way.

Table -III.: Comparison of Transistor Counts

No. of bits	Resistive Threshold Logic	Conventional CMOS	Percentage Reduction (%)
1	18	26	30
2	56	120	53.33
3	118	380	68.94

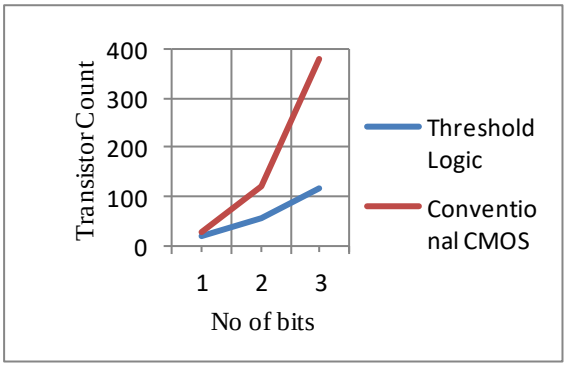

Fig. 8. Comparison of transistor counts

B. Comparison of area and power dissipation

Implementation area and power dissipation are critical factors in chip design. Implementation area of circuits must be less in order to save chip area. The main highlight of using threshold logic is the reduction in circuit complexity and implementation area. Resistive threshold realization of magnitude comparator leads to significant reduction in implementation area of circuits when compared to conventional CMOS implementations.

A comparison can be made on area and power consumption of resistive threshold based magnitude comparator implementation and conventional CMOS based implementation. The area of the circuits can be found out from the layout. Electric VLSI Design System version 9.06 is used for drawing the layout of circuits. Fig.9 shows the layout of one-bit magnitude comparator using resistive threshold logic and Fig.10 shows the layout of CMOS implementation of one-bit magnitude comparator. From the layout, implementation area is calculated. The implementation area required for resistive threshold implementation of one-bit comparator is about 464.4μm^2, whereas for conventional CMOS based implementation of one-bit comparator requires an area of about 640μm^2. Hence there is a reduction in implementation area. This reduction in area becomes more convincing as the no of bits of comparator increases, which leads to increase in circuit complexity for conventional CMOS, based implementations.

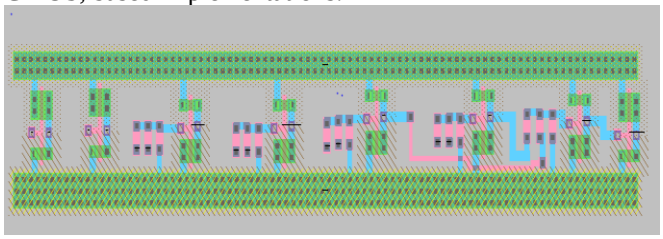

Fig. 9. Layout of resistive threshold based one bit magnitude comparator

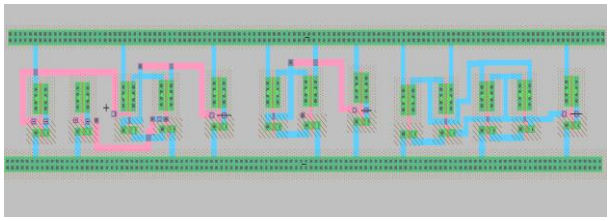

Fig. 10. Layout of one bit magnitude comparator implementation using conventional CMOS logic.

A comparison of implementation area and power dissipation of one-bit magnitude comparator implementation using resistive threshold and conventional CMOS implementation is given in Table IV.

Table-IV: Comparison Of Area And Power Dissipation For One Bit Magnitude Comparator

Logic	Area (μm^2)	Power dissipation
Conventional CMOS	640	171.66 µW
Resistive Threshold Logic	464.64	167.69 mW

It can be seen from Table IV that resistive threshold based implementation requires less implementation area. But power dissipation is somewhat higher compared to that of conventional CMOS based implementation. But the implementation area and transistor count required for resistive threshold based implementation is less which can be considered as a powerful feature of threshold logic.

VI. CONCLUSION

Threshold logic can be regarded as an alternative design technique for digital circuits. The concept of threshold logic can be made use of in implementing circuits and using the idea of resistive threshold logic, magnitude comparator was realized in this paper. The resistive threshold based implementation of magnitude comparator gives the same results as that of conventional CMOS implementation, but with some added advantages in transistor count and implementation area. The complexity of circuits developed using the idea of threshold logic is less compared to the conventional implementations. The significance of threshold logic becomes evident in the implementation of complex circuits, since using the concept of threshold logic complex circuits can be implemented in a better and economical way. The idea of threshold logic can be extended for implementing complex circuits in a better way. This concept is suitable for realizing complex circuits in a well-organized manner and can be made use of in the design of circuits to implement them in an efficient way. Thus the concept of threshold logic can be used as an alternative method for the design and implementation of logic circuits.

REFERENCES

1. Saburo Muroga, "Generation and Asymmetry of Self-Dual Threshold Functions" *IEEE Trans. on Electronic Computers*, vol.EC-14, issue-2, pp.125-136,April 1965.
2. Shuzo Yajima, Toshihide Ibaraki, "A theory of completely monotonic functions and its application to threshold logic," *IEEE Trans. on Computers* , vol. C-17, no.3, March 1968.
3. Shuzo Yajima, Toshihide Ibaraki, "Realization of Arbitrary Logic Functions by completely Monotonic Functions and its applications to Threshold Logic," *IEEE Trans. Comput.*, vol. C-17, no.4, April 1968.
4. Valeiru Beiu, Jose M Quintana, Maria.J. Avedillo "VLSI Implementations of Threshold logic-A comprehensive Survey, " *IEEE Trans. Neural Netw.*, vol. 14, no.5, September 2003.
5. M.J. Avedillo, J. M. Quintana, A.Rueda, and E. Jimenez "Low power CMOS threshold logic gate ,"*Electron. Lett.*, vol.31, no.25, 7th December 1995.
6. S. Bobba and I.N. Hajji, "Current Mode Threshold Logic Gates ," *Proc. International Conference on Computer Design*, pp. 235-240, 2000.
7. P.Celinski, J. F. Lopez, S.H. Sarawi, and D.Abbott, "Low power, high speed,charge recycling CMOS threshold logic gates," *Electron. Lett.*, vol. 37, issue.17, 16 August 2001.
8. Alex Pappachen James, Linu Rose, V.J. Francis, and Dinesh.S.Kumar, "ResistiveThreshold Logic," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol.22, no.1, pp. 190-195, January 2014.
9. Jose M Quintana, Maria.J.Avedillo, A.Rueda, "Low cost BSA technique for threshold logic gate based multiplier implementations," *Electron. Lett.*, vol.33, no.12, 5th June 1997.
10. Jose M Quintana, Maria.J. Avedillo, Esther Rodriguez Villegas and Adoracion Rueda "Threshold logic based Design of Compressors, " 9th *International Conference on Electronics, Circuits and Systems*, vol. 2, pp. 661-664, 2002.
11. Leon O Chua, "Memristor-The Missing Circuit Element," *IEEE Trans. on Circuit Theory*, vol.CT-18, no.5, pp. 507-519, September 1971.
12. Z.Biolek, D.Biolek, V.Biolkova, "Spice Model of Memristor with Nonlinear Dopant Drift," Radio Engineering, vol.18, no.2, June 2009.

AUTHORS PROFILE

Soumya R received the B. Tech. degree in electronics and communication engineering from from LBS Institute of Technology for Women, Kerala University, Kerala, India in 2013 and M. Tech. degree in Applied Electronics and Instrumentation from College of Engineering Trivandrum, Kerala University, India, in 2016. Her research interests include image & signal processing.

Dr. V. Suresh Babu, Professor and Head of Department of Electronics and Communication Engineering, Government Engineering College, Barton Hill, Thiruvananthapuram, Kerala, India, has graduated in Electronics and Communication Engineering from TKM College of Engineering Kollam in 1988 and took his post graduate degree on Integrated Electronic Devices and Circuits from IIT, Madras. His Ph. D degree is on Design and analysis of CMOS based Neurons from university of Kerala. His current research interests are in the area of solar photovoltaics and nanoelelectronics.

Varun P. Gopi received the B. Tech. degree in electronics and communication engineering from Amal Jyothi College of Engineering, Kanjirappally affiliated to Mahatma Gandhi University, Kottayam, India in 2007, and M. Tech. degree in signal processing from College of Engineering Trivandrum affiliated to Kerala University, Trivandrum, India in 2009. He pursued the Ph.D. degree in medical image processing with the Department of Electronics and Communication Engineering, National Institute of Technology, Tiruchirappalli, India in 2014. He is currently an Assistant Professor with the Department of Electronics and Communication Engineering, National Institute of Technology, Tiruchirappalli, India. His research interests include biomedical signal & image processing, deep learning, artificial intelligence and IoT. He was a recipient of the Canadian Commonwealth Scholarship Award 2011–2012 under the Graduate Student Exchange Program in the Department of Electrical and Computer Engineering, University of Saskatchewan, Saskatoon, SK, Canada.