6-1-2016

Modeling and design of memristor-based fuzzy systems

Sherif Hassanein Hamed Amer

The American University in Cairo

School of Sciences and Engineering

MODELING AND DESIGN OF MEMRISTOR-BASED FUZZY SYSTEMS

A Thesis Submitted to

Electronics and Communication Engineering Department

in partial fulfillment of the requirements for
the degree of Master of Science

by Sherif Hassanein Hamed Amer

under the supervision of Prof. Sherif Abdelazeem and Dr. Ahmed Madian
May/2016

Approval Sheet Goes Here

To my Mum and Dad

ACKNOWLEDGMENTS

I would like to acknowledge both my supervisors: Prof. Sherif Abdelazeem and Dr. Ahmed Madian for their great support throughout my thesis.

I would also like to acknowledge Dr. Hany ElSayed and Dr. Mohamed Bakr for their assistance throughout my research.

I would like to acknowledge my examiners Prof. Ahmed Soliman and Prof. Mohab Anis for their valuable and constructive comments.

Finally, I would like to acknowledge the graduate program directors: Dr. Ayman El Ezaby and Dr. Karim Seddik.

ABSTRACT **1**

I. INTRODUCTION **2**

II. LITERATURE REVIEW **5**

II.1. THEORY OF MEMRISTORS 6
II.2. CHARACTERIZATION OF THE MEMRISTOR ELEMENT 9
II.3. MATHEMATICAL MODELING MF THE MEMRISTOR 10
II.4. MEMRISTOR-BASED APPLICATIONS 14
II.4.1. review on memristor applications *14*
II.4.2. fuzzy systems *15*

III. MODELING AND DESIGN OF MEMRISTOR-BASED MIN-MAX CIRCUITS **21**

III.1. 2-INPUT MEMRISTOR-BASED MIN-MAX CIRCUITS 22
III.2. COMPARISON WITH PREVIOUS TRANSISTOR-BASED STRUCTURES 23
III.3. 3-INPUT MEMRISTOR-BASED MIN-MAX CIRCUIT 24
III.3.1. Proof of 3-input memristor-based min-max circuit *24*
III.3.2. Effect of the memristor threshold on 3-input memristor-based min-max circuit *26*
III.4. N-ARY MEMRISTOR-BASED MIN-MAX CIRCUIT 29
III.4.1. Proof of N-input memristor-based min-max circuit *30*
III.4.2. Effect of the memristor threshold on N-ary memristor-based min-max circuit *33*
III.5. SUMMARY 42

IV. MEMRISTOR-BASED CENTER-OF-GRAVITY (COG) DEFUZZIFIER **43**

IV.1. CENTER OF GRAVITY DEFUZZIFICATION 44
IV.2. PROPOSED DEFUZZIFIER CIRCUIT 45
IV.3. SIMULATION RESULTS AND DISCUSSIONS 47
IV.4. SUMMARY 49

V. CONCLUSIONS **50**

REFERENCES **52**

LIST OF FIGURES

FIGURE 1: THE FOUR FUNDAMENTAL CIRCUIT ELEMENTS 6

FIGURE 2: HYSTERESIS CURVE 7

FIGURE 3: TIME DOMAIN VOLTAGE/CURRENT SIGNALS ACROSS THE MEMRISTOR ELEMENT. 8

FIGURE 4: THE MEMRISTOR ELEMENT CHARACTERIZED BY HP 9

FIGURE 5: THE JOGLEKAR WINDOW FUNCTION [16] 11

FIGURE 6: THE BIOLEK WINDOW FUNCTION [15] 12

FIGURE7: THE PROPOSED STRUCTURE FOR THE MEMRISTOR-BASED FUZZIFIER; (A) DEPICTS THE CONTINUOUS FORM OF THE MEMBERSHIP FUNCTIONS (FUZZY SETS); (B) THE DISCRETIZED FORM OF THE MEMBERSHIP FUNCTIONS; (C) MEMRISTOR CROSS-BAR IMPLEMENTATION FOR THE MEMBERSHIP FUNCTION (PROPOSED DEFUZZIFER); (D) A -1VPULSE APPLIED TO ACTIVATE THE DESIGNATED COLUMN REPRESENTING A SPECIFIC INPUT; (E) THE EQUIVALENT CIRCUIT WHEN INPUT X=3 IS ACTIVATED. 16

FIGURE 8: VOLTAGE DIVIDER 17

FIGURE 9: MAX AND MIN CIRCUITS 18

FIGURE 10: FOLLOWER-AGGREGATOR BASED COG DEFUZZIFIER 19

FIGURE11: CURRENT MODE DEFUZZIFIER 20

FIGURE 12: 2-INPUT MINIMUM CIRCUIT 22

FIGURE 13: 3-INPUT MINIMUM CIRCUIT 26

FIGURE 14: THRESHOLD CURRENT VS MEMDUCTANCE AT $\mathbf{Roff = 200K\Omega}$ AND N = 3 39

FIGURE 15: THRESHOLD CURRENT VS. VOLTAGE STEP AT $\mathbf{Roff = 200K\Omega}$ AND DIFFERENT 'N'. I2 IS THE CURVE ASSOCIATED WITH N=2 39

FIGURE 16: BLOCK DIAGRAM FOR FUZZY CONTROLLERS 43

FIGURE 17: PROPOSED COG DEFUZZIFIER CIRCUIT 47

LIST OF TABLES

TABLE 1: COMPARISON BETWEEN TRANSISTOR/MEMRISTOR-BASED MIN-MAX CIRCUITS 24

TABLE 2: MEMRISTOR CURRENTS IN EVERY STATE 27

TABLE 3: SIMULATION RESULTS FOR 3-INPUT MINIMUM CIRCUIT 29

TABLE 4: VALIDATION FOR ε 41

TABLE 5: 2-INPUT COG DEFUZZIFIER 48

TABLE 6: COMPARISON BETWEEN THE FOLLOWER-AGGREGATOR AND THE PROPOSED CIRCUIT 48

List of Abbreviations

IC	Integrated Circuit
CMOS	Complementary Metal Oxide Semiconductor
VLSI	Very Large Scale Integration
HP	Hewlett Packard
TEAM	Threshold Adaptive Memristor Model
COG	Center Of Gravity

ABSTRACT

The American University in Cairo, Egypt

Modeling and Design of Memristor-based Fuzzy systems

Name: Sherif Hassanein Hamed Amer

Supervisors: Prof. Sherif Abdelazeem and Dr. Ahmed Madian

The incessant down scaling of CMOS technology has been the main driving force for the semiconductor industry over the past decades. Yet, as process variations and leakage current continue to exhibit more pronounced effect with every technology node, this down scaling paradigm is expected to saturate in the few coming years. This prospect has led the research community to seek new technologies to surpass those challenges. Amongst the promising candidates is the memristor technology recently characterized by HP Labs. The miniaturized features and the peculiar behavior exhibited by the memsitor make it very well suited in some applications. For instance, memristors are used as memory cells in state-of-the-art memories known as Resistive RAMs in which the non-volatility of the memristor is exploited. The programmable nature of the memristor has made it a powerful candidate in neuromorphic and fuzzy systems that, in essence, go beyond the classical Von Neumann computing paradigm. In such systems, ideas from Artificial Intelligence, that for so long have been implemented on the software level, are implemented as electronic circuitry which renders benefits such as compact area and reduced power consumption. This work focuses on memristor-based Fuzzy applications. First, memristor-based Min-Max circuit used in the Fuzzy Inference engine is analyzed. It is proven that memristor-based Min-Max circuits can be extended to an arbitrary number of inputs ‘N’ under the proper design constraints. In addition, the effect of the memristor threshold is analyzed and a closed form expression is derived. It is shown that, for a given memristor with a specific OFF resistance and threshold current, there is a trade-off between the size and the resolution of the circuit. Then, a memristor-based Defuzzifier circuit is proposed. A major challenge in Defuzzifiers is their area occupancy due to the use of Multiplier and Divider circuits. In this design, the memristor analog programmability is leveraged to reduce the multiplication operation into simple Ohm’s Law which alleviates the need for dedicated hardware for multiplier circuit and, accordingly, reduces the area occupancy.

I. INTRODUCTION

Complementary Metal Oxide Semiconductor (CMOS) technology has for so long stood as the cornerstone of all Very Large Scale Integration (VLSI) systems. However, other technologies such as the memristor [1] have been recently proposed that promise to push the semiconductor industry into new paradigms. The perpetual down scaling of CMOS technology has provided ever enhanced performance of electronic circuits over the past decades. With each technology node (a technology node is defined as the channel length of the transistor), average power consumption has decreased, device speed has been boosted and more integration has become achievable. This enhancement was predicted in 1965 by Gordon Moore and has ever since been known as “Moore’s Law”.

The sustainability of Moore’s Law, however, cannot last much longer due to two major challenges which are (1) the CMOS science has reached the fundamental physical limits [2] which prohibits further miniaturization and (2) process variations have skyrocketed in such Nano-scale regime.

These issues have instigated significant research trying to provide novel and innovative solutions to assuage the aforementioned challenges. Several endeavors have been proposed on both circuit and device levels. On the circuit level, new structures and circuit architectures have been proposed such as multilayered Integrated Circuits where an extra spatial dimension is exploited to provide a higher functionality per chip area ratio (i.e., 3D ICs) [2]. Also, on the device level, researchers have sought out new devices such as carbon nanotubes, spintronics and FinFets. Amongst the new devices tackled by the research community, a novel device known as “memristor” stands as a powerful candidate that has the potential to push the microelectronics industry into new paradigms.

Memristors are newly characterized devices that were first theoretically predicted by Leon Chua in 1971 but had not been physically realized until 2008 when HP announced the first manufactured memristor based on the Titanium dioxide Ti02 process [3]. Ever since that date, a significant research has been undergone in the area of memristors and mermsitor-based systems.

Leon Chua postulated that the memristor constitutes the missing link between the electric charge and the magnetic flux. He published a paper in 1971 [1] in which he

provided a merely theoretical treatment for the memristor element. Later in 1976 [4], he published another paper generalizing the concept of the memristor from an electrical element to a whole system theory. Recently, in 2015, Chua published an article summarizing his work on memristors. Section II.1 provides a detailed discussion about the theory of operation of the memristor element.

In order for the memristor element to be integrated into commercial CAD tools, several mathematical models were developed. The published models vary from a very simple model such as the one published by HP [3] to a very complex one such as the Simmon's barrier tunneling model [5]. Section II.2 will discuss the different models developed for the memristor as well as highlight the fundamental trade-offs in those models.

Despite the immaturity of the memristor science that calls for the need for more theoretical work regarding the memristor element and, accordingly, the continuous refinement of the models, the peculiar behavior of mermsitors and their miniaturized size have instigated a surge in memristor-based applications in which those features are leveraged to deliver new functions such as resistive RAMs (ReRAMs) and Neuromorphic and Fuzzy circuits or improve the performance of transistor-based architectures such as in the case of digital and analog circuits. Several memristor-based applications are discussed in section II.3 with emphasis on Fuzzy applications.

As CMOS technology approaches its fundamental limits, researchers have sought new design paradigms. Amongst the very promising emerging computer architectures is the concept of Neuromorphic computing and Fuzzy systems. While such architectures can be implemented using transistors, memristors exhibit peculiar characteristics that are well suited to such systems which enable the implementation of high density and power efficient systems [6].

To this end, chapter III will discuss the modeling and design of memristor-based Min-Max circuits. Min-Max circuits are essential and fundamental building blocks in many Neuromorphic and Fuzzy systems [7]. The use of memristors in such systems reduces the area occupancy, especially, that they are repeated in different parts of the Integrated Circuit (IC) [7]. Although memristor-based min-max circuits have been discussed by several researchers, their treatment was only confined to explaining the basic working principle and was limited to only 2-input circuits. In this work, the theory of memristor-based min-max circuits is generalized to N-input circuits. Also,

design constraints are derived. An important feature of the memristor is the existence of a Voltage/Current threshold below which resistive switching does not occur. Several researchers have pointed out that the memristor threshold might cause the circuit to malfunction if not accounted for in the design phase. Yet, no analytical treatment of the effect of the threshold was found in the literature. This work develops a closed form expression for the effect of the memristor threshold on memristor-based min-max circuits.

Chapter IV proposes the design of memristor-based Center-Of-Gravity (COG) Defuzzifier circuit. COG Defuzzifiers are essential building blocks in Fuzzy systems. A major challenge in the design of such system is the Multiplier/Divider circuits that are very area consuming. Hence, in this endeavor, memristors are leveraged to mitigate the prime challenge in COG Defuzzifiers which is area occupancy. The miniaturized size of the memristor and its programmable resistance are exploited in order to reduce the size of the multiplier circuit and, eventually, yield a more compact design.

Chapter V will present the conclusions.

II. LITERATURE REVIEW

The recent characterization of the memristor element by HP using the Titanium dioxide process has resulted in a surge in memristor-based applications. The peculiar features exhibited by this device open the door for a wide range of applications that have not been possible before. For example, researchers have exploited the memory feature rendered by the memristor element to build state-of-the-art Resistive RAMs. Also, apart from the classical Von Neumann computing paradigm, memristors are sought to be used in building new computing platforms such as Fuzzy and Neuromorphic systems that generally go under biologically-inspired circuits [8]. Meanwhile, memristors are also used in improving some current analog circuits, in which the analog programmability of memristors is exploited, and digital circuits where its miniaturized features and binary mode of operation are leveraged [9, 10].

Although this thesis is primarily devoted to memristor-based applications, an understanding of the theory of memristors and the associated mathematical models are essential. Recently, in [11], Chua published a tutorial in which he provided a thorough explanation of the working theory of the device based on his seminal work back in 1971 and 1976 [1,4]. Also, several mathematical models have been developed in [3, 12, 13, 14, 15] to bridge between the memristor theory and SPICE compatible models that enable circuit designers to simulate such devices in various applications.

To this end, this chapter will start by reviewing the key principles and fundamentals about memristors and their development based on Chua's treatment in [1, 4, 11]. This will be presented in section II.1. Section II.2 will discuss the characterization of the memristor using the Titanium dioxide (TiO2) process. Section II.3 will address the different models posed by researchers for the memristor unraveling some of the trade-offs existing in those models. Section II.4 will draw upon some of the interesting applications proposed in the literature. A particular emphasis will be placed on Fuzzy applications which are the major focus of this thesis.

II.1. THEORY OF MEMRISTORS

In 1971, Chua laid out a figure that establishes the relation amongst the four fundamental electrical quantities namely: Charge 'q', Flux 'φ', Voltage 'V' and Current 'I' as shown in figure 1.

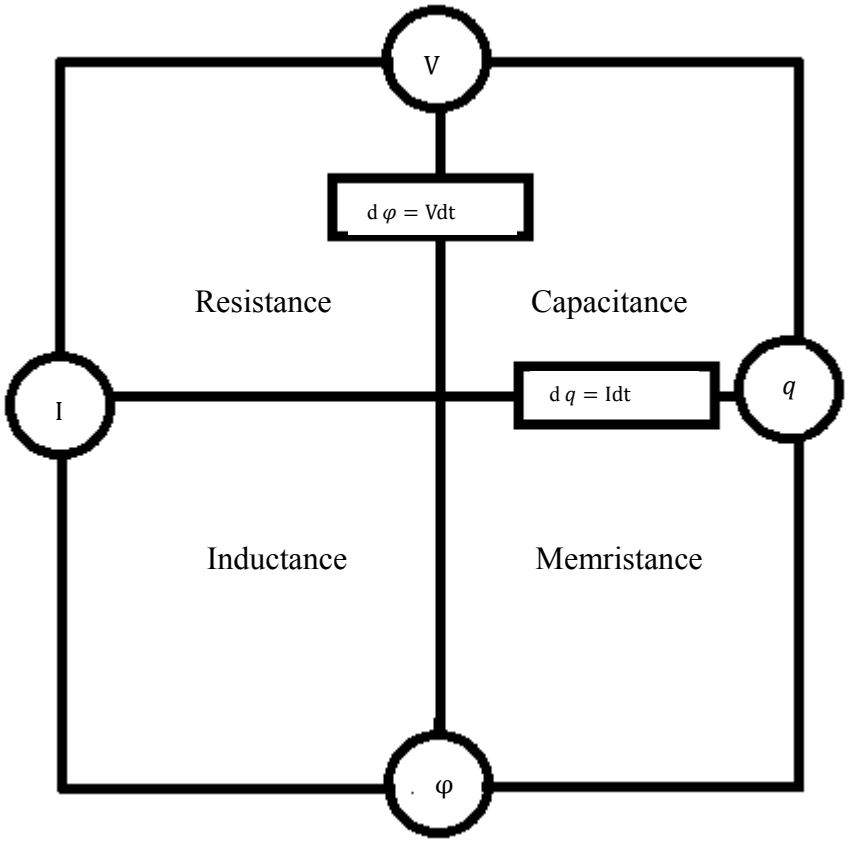

Figure 1: The four fundamental circuit elements

Chua noticed that there exists a missing relation between 'q' and 'φ' depicted in the bottom right corner of figure 1 which he postulated as being the memristor element [3, 1]. Later, in 1976, Chua published a paper generalizing the concept of a memristor from a single electrical element to a whole system theory [4]. In fact, memristive behavior was found in numerous living beings and plants [11]. Hence, Chua sought to characterize the memristive behavior based on an axiomatic definition that describes the fingerprints of any memristive system.

Experimental definition of memristors:

Any two terminal device exhibiting pinched hysteresis loop which always passes through the origin in the Voltage-Current plane when driven by a periodic Voltage/Current source with zero DC component is a memristor.

Also, in general, there are two types of memristors: voltage controlled memristors and current controlled memristors which are defined as follows based on Chua's definitions.

Voltage Controlled memristor:

A 2-terminal device is a voltage controlled memristor if, and only, if for all periodic input voltages which gives periodic current response with the same frequency, V(t)and I(t) plotted in the I vs. V plane always pass through the origin whenever V(t) = 0.

Current Controlled memristor:

A 2-terminal device is a current controlled memristor if, and only, if for all periodic input currents which gives periodic voltage response with the same frequency, V(t)and I(t) plotted in the V vs. I plane always pass through the origin whenever I(t) = 0.

Figure 2 depicts the hysteresis curve for a memristor driven by a sinusoidal signal.

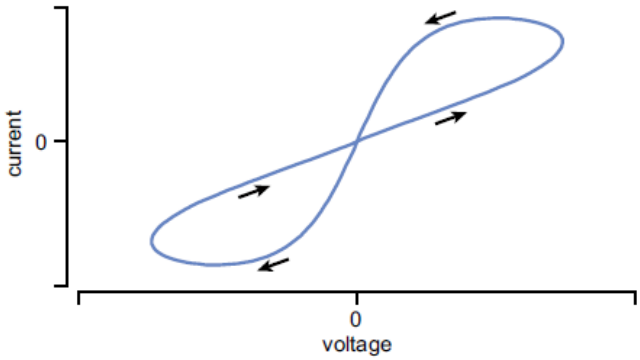

Figure 2: Hysteresis curve

Figure 3 shows the associated time domain Voltage-Current curves.

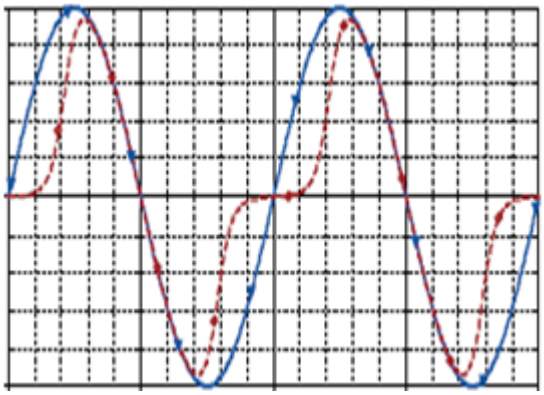

Figure 3: Time domain Voltage/Current signals across the memristor element.

Memristors can be also classified according to their mathematical complexity [11]. However, this section will be only concerned with the simplest of all memristors, ideal memristors, which coincides with the original definition in [1]. The ideal memristor model can be recovered from the constitutive relation between 'q' and 'φ' as follows:

$$\varphi(q) = \varphi(0) + \int_0^q R(q).\,dq \tag{2.1}$$

Differentiating both sides of (2.1) yields:

$$\frac{d\varphi}{dt} = R(q)\frac{dq}{dt} \tag{2.2}$$

Given that $\frac{d\varphi}{dt} = V$ and $\frac{dq}{dt} = I$, (2.2) can be written as:

$$V = R(q)I \tag{2.3}$$

Equations (2.1) through (2.3) pertain to the charge controlled memristor that are derived based on the constitutive relation $\varphi(q)$. The ideal current controlled memristor can be described based on the set of two equations that follows directly from the previous equations.

$$V = R(q)I \tag{2.4}$$

$$\frac{dq}{dt} = I$$

By the same token, the flux controlled memristor can be derived based on the constitutive relation $q(\varphi)$ and the voltage controlled memristor is described as follows:

$$I = G(\varphi)V \tag{2.5}$$

$$\frac{d\varphi}{dt} = V$$

II.2. CHARACTERIZATION OF THE MEMRISTOR ELEMENT

As mentioned before, memristive behavior is exhibited by various materials. Hence, several materials and processes were used to realize memristors. However, in this work, the focus will be on the Titanium dioxide (TiO2) based memristor that was characterized by HP. Figure 4 depicts the structure designed by HP [3].

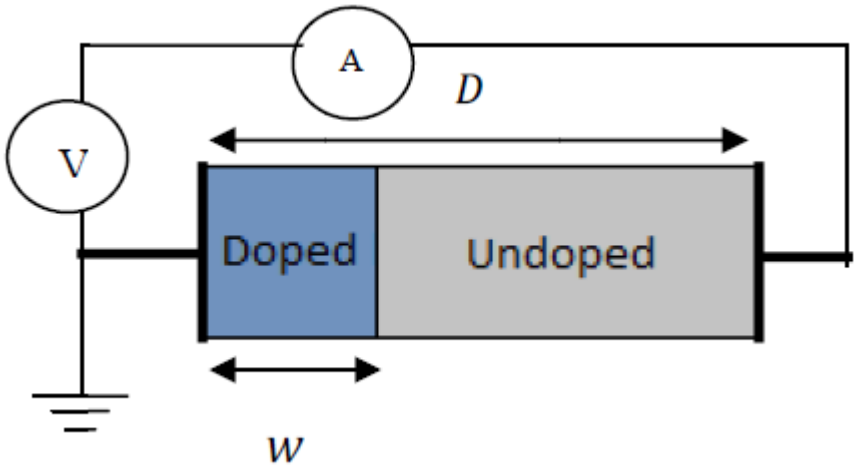

Figure 4: The memristor element characterized by HP

It consists of two Platinum electrodes represented by the two black vertical thick lines at the two extremes of the device and sandwiched between them two layers of TiO2 one of which is oxygen deficient. The Oxygen deficient layer contains oxygen vacancies that act like dopants and, hence, represented by the doped region. The other layer is free of carriers and represented by the undoped region. When voltage is

applied across the device, the length of the doped region '*w*' changes from zero (totally undoped) to D (fully doped).

II.3. MATHEMATICAL MODELING MF THE MEMRISTOR

This section will discuss various memristor models in the literature while highlighting the main pros and cons for each model. In general, there are two classes of models: linear models and nonlinear models. Linear models include the linear ion drift model which was proposed by HP Labs in [3], as well as, its variations which, essentially, add a window function to the linear ion drift model such as the Joglekar [16], the Biolek [15], the Sturkov [17] and the Prodromakis [13] models. On the other hand, several nonlinear models were proposed such as the Pickett [5] model, the TEAM model [12] and the Yakopcic model [14].

Linear ion drift model:

This model was developed by HP and presented as follows:

$$V = (R_{on}\frac{w(t)}{D} + R_{off}(1 - \frac{w(t)}{D}))I \tag{2.6}$$

$$\frac{dw(t)}{dt} = \mu_D \frac{R_{on}}{D^2} I(t)$$

In this model, the memristor is thought of as two series resistors and the total resistance is modulated via the state variable $w(t)$. R_{on} represents the resistance of the doped region and R_{off} represents the resistance of the undoped region. Although the model is intuitive and fairly simple, it fails to capture the deceleration of the dopants as they approach either extremes of the device. Physical arguments show that the speed of the dopants, represented by the time derivative of w(t), should gradually decrease before their speed drops to zero [15]. In order to address such issue, several researchers have proposed various window functions to capture such dynamics via multiplying the state equation in (2.6) by a window function f(x) as in (2.7):

$$\frac{dw(t)}{dt} = \mu_D \frac{R_{on}}{D^2} I(t) f(x) \tag{2.7}$$

.

Joglekar window fucntion

In [16], the window function in (2.8) was proposed where $x = \frac{w(t)}{D}$ is the normalized state variable and 'p' is a fitting parameter.

$$f(x) = 1 - (2x - 1)^{2p} \tag{2.8}$$

Figure 5 depicts the Joglekar window function for various 'p' values [16].

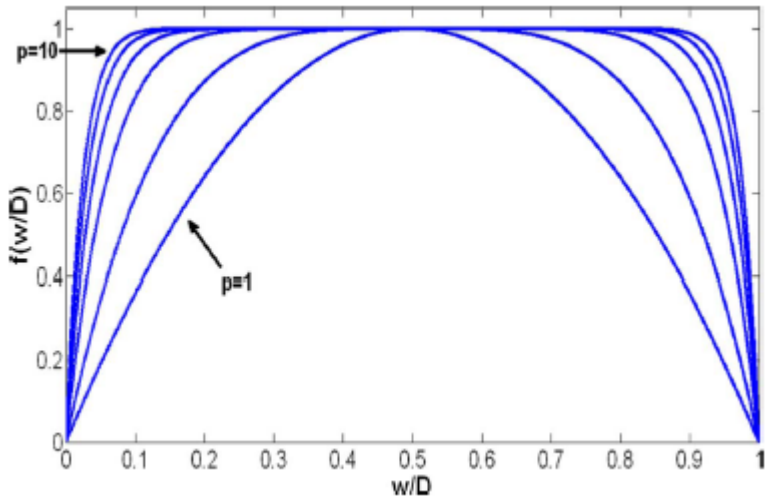

Figure 5: The Joglekar window function [16]

Although the Joglekar window addressed the problem in the HP model, a major numerical error was encountered in the simulation of the model [15] which was later solved by Biolek in his proposed window function.

Biolek window function

In [15], the window function in (2.9) was proposed.

$$f(w) = 1 - (2\frac{w(t)}{D} - stp(-I))^{2p} \tag{2.9}$$

$$stp(-i) = \begin{cases} 1, & I \geq 0 \\ 0, & I < 0 \end{cases}$$

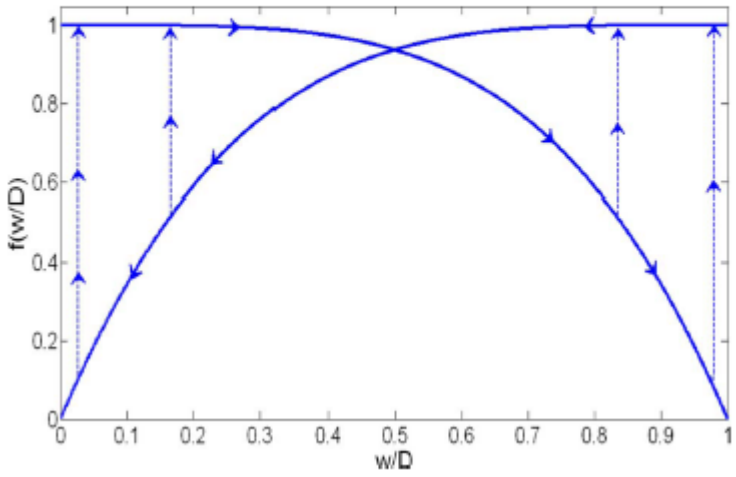

Figure 6: The Biolek window function [15]

Other window functions were also proposed in [13, 17]. Despite the simplicity of the linear models and their ease of integration in circuit simulators such as SPICE, experimental results deviate significantly from the linear models [12]. Hence, the need for nonlinear models to capture the memristor behavior more accurately was indispensable.

Nonlinear models

Several nonlinear models were proposed in the literature. Despite the differences they might have in terms of the mathematical model itself, they are all centered on two major characteristics that the TiO2 memristor exhibits that were not captured by the linear models which are: (1) the existence of a threshold (2) an exponential drift behavior of the state variable.

Yakopcic model

This model assumes an implicit relation between the current and voltage and described as follows:

$$I(t) = \begin{cases} a_1 x(t) \sinh(bV(t)), & V(t) \geq 0 \\ a_2 x(t) \sinh(bV(t)), & V(t) < 0 \end{cases} \quad (2.10)$$

Where a_1 and a_2 are fitting parameters.

Pickett model (Simmon's barrier tunneling model)

In [5], a model was proposed based on the simmon's barrier tunneling phenomenon described as follows:

$$\frac{dw(t)}{dt} = \begin{cases} f_{off} \sinh\left(\frac{I}{I_{off}}\right) \exp\left[-\exp\left(\frac{w - a_{off}}{w_c} - \frac{|I|}{b}\right) - \frac{w}{w_c}\right], & I(t) > 0 \\ f_{on} \sinh\left(\frac{I}{I_{on}}\right) \exp\left[-\exp\left(\frac{w - a_{on}}{w_c} - \frac{|I|}{b}\right) - \frac{w}{w_c}\right], & I(t) < 0 \end{cases} \quad (2.10)$$

Where i_{off} , i_{on} f_{off}, f_{on}, w_c, b, a_{off} and a_{on} are fitting parameters.

TEAM model

The TEAM model is a simplified version of the Simmon's barrier tunneling model. The simplifications made by the TEAM model are: (1) explicitly introducing threshold currents i_{off} and i_{on} as the switching threshold current from OFF to ON and ON to OFF, respectively and (2) assuming a polynomial dependence instead of an exponential dependence. These assumptions make the model more computationally efficient while preserving the same accuracy [12]. The TEAM model is described as follows:

$$\frac{dx}{dt} = \begin{cases} k_{off} \cdot \left(\frac{I(t)}{i_{off}} - 1\right)^{\alpha_{off}} \cdot f_{off}(x), & 0 < I_{off} < I(t) \\ 0, & I_{on} < I(t) < I_{off} \\ k_{on} \cdot \left(\frac{I(t)}{I_{on}} - 1\right)^{\alpha_{on}} \cdot f_{on}(x), & I(t) < I_{on} < 0 \end{cases} \quad (2.11)$$

In general, linear models are fairly simple both analytically and computationally. However, they are less accurate than their nonlinear counterparts. On the other hand, nonlinear models are complex both analytically and computationally. Therefore, continuous refinement for the models is necessary.

II.4. MEMRISTOR-BASED APPLICATIONS

Memristors have been primarily used in four applications: analog circuits, digital circuits, memories and neuromorphic and fuzzy systems (sometimes referred to as biologically-inspired systems or beyond Von Neumann architectures). Since this thesis is mainly devoted to memristor-based fuzzy systems, only a brief discussion will be presented about other applications for memristors and the rest of the section will discuss fuzzy applications for memristors.

II.4.1. REVIEW ON MEMRISTOR APPLICATIONS

The analog programmability of the memristor has inspired many researchers to utilize it in several applications other than memories and digital design. One potential application is programmable analog circuits. In [18], the authors reinvented a number of currently existing and extensively used analog blocks by employing memristors in the design. Their idea is hinged upon the threshold behavior of the memristor. The authors utilized this fact by building a memristor-based analog circuit that operates in two phases. In phase '1', the programming phase, high voltages (voltages higher than the threshold of the memristors), are used to program the memristor to the desired resistive value. In phase '2', the analog operation, low voltages (lower than the threshold of the memristor) are applied to perform the analog functionality of the circuit. For further details, the reader is referred to [18].

The non-volatility of memristors and their miniaturized features have instigated their use in state of the art memories known as Resistive RAMs. Unlike in regular CMOS designs where the logic is stored as a voltage, logic in Resistive RAMs is stored as resistive value whereby the information stored is not lost when the power supply is switched off owing to the peculiar nature of the memristive behavior (hence, non-volatile). Also, their miniaturized size has enabled building denser memory arrays. Three major challenges have been encountered by researchers in designing resistive RAMs which are: (1) non-destructive reading operation in which the reading circuitry and the applied read voltages should be designed in such a way that does not corrupt the data stored in the memory cell as in [19] (2) process variations and their effect on read/write operations as in [20] (3) sneak paths testing in memory arrays as in [21].

Another interesting application in which the compact size of the memristor is leveraged, is digital applications. In essence, there are two types of digital applications: logic in memory and conventional logic. In [9], material implication logic family was presented in which memristors are used as memory elements as well as perform logic operations. On the other hand, other logic families such as Memristor Ratioed Logic (MRL) described in [10] use memristor as computational elements such as in the case of standard CMOS architectures.

Memristors have been used in Neuromorphic systems to build memristor-based Artificial Neural Networks. The unique behavior of the memristor makes it very well suited to be used as synapses [22]. For further information, the reader is referred to [8].

II.4.2. FUZZY SYSTEMS

Fuzzy systems are comprised of three fundamental building blocks which are: the Fuzzifier, the Inference Engine and the Defuzzifier. In [23, 24], a memristor-based fuzzifier circuit was proposed. Inspired by the work in [23, 24], this thesis will address memristor-based Inference engine (which is comprised of a min-max circuit) and Defuzzifier. To this end, this section will start by discussing the design of memristor-based Fuzzifier circuit proposed in [23, 24]. Then, the relevant literature in min-max circuits and Defuzzifiers will be reviewed.

II.4.2.1. MEMRISTOR-BASED FUZZIFIER CIRCUIT

Figure 7 depicts the memristor-based Fuzzifier. Figure 7(a) represents the mathematical representation of the fuzzy sets (also known as the membership fucntion) where the y-axis represents the degree of membership of the independent variable (input) in the different fuzzy sets. For example, assume fuzzy set A represents ‘Cold’ and fuzzy set B represents ‘Hot’. At x = 0, the temperature is ‘Hot’ with degree ‘1’ and ‘Cold’ with degree ‘0’ while at x = 3, both sets A and B apply with the same degree which can be interpreted, for instance, as a moderate temperature.

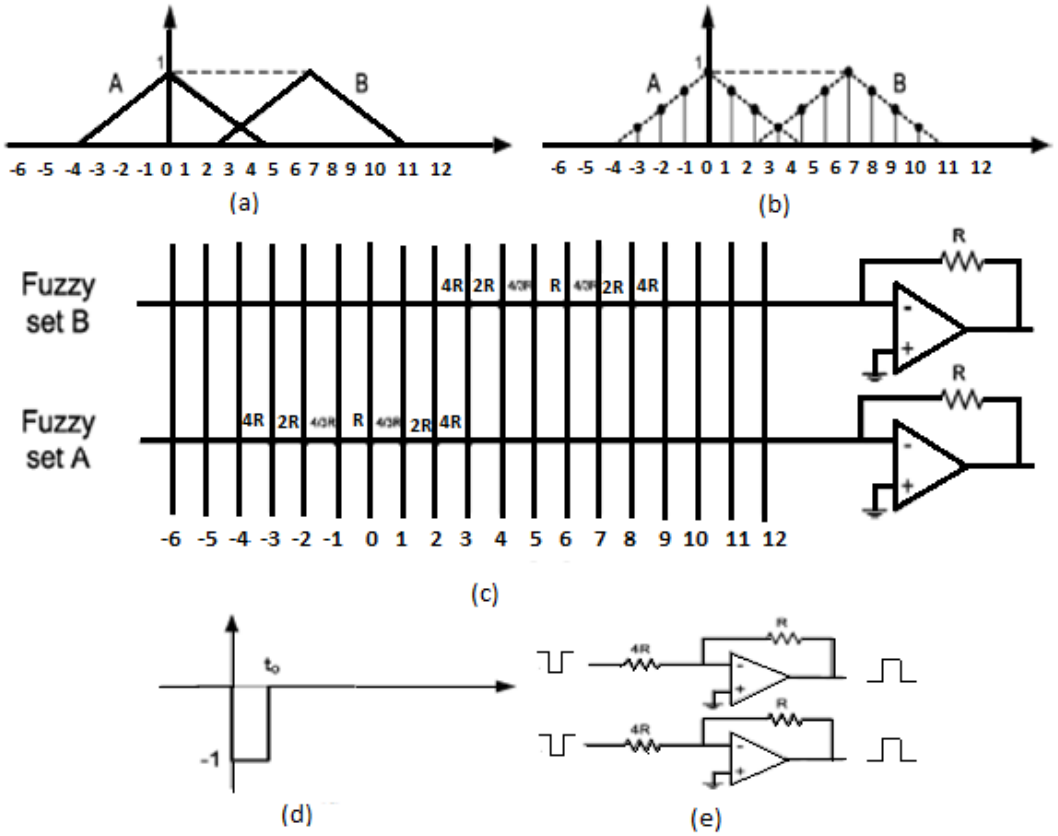

Figure7: The proposed structure for the memristor-based fuzzifier; (a) depicts the continuous form of the membership functions (fuzzy sets); (b) the discretized form of the membership functions; (c) memristor cross-bar implementation for the membership function (proposed defuzzifer); (d) a -1Vpulse applied to activate the designated column representing a specific input; (e) the equivalent circuit when input x=3 is activated.

The fuzzy sets are discretized in figure 7(b) in such a way that every discrete point represents a memristor that connects between the corresponding vertical and horizontal lines in figure 7(c) where the value of that point is programmed to the memristor. If for instance, the input is '3' (x = 3), the vertical line depicted by '3' is activated via the application of a -1V pulse to it as in figure 7(d) which yields the structure in figure 7(e). From circuit analysis, it can be shown that the voltage at A and B is $\frac{R}{4R}$ which means that the input x = 3 is a member in both fuzzy sets A and B with a degree of 0.25.

II.4.2.2. INFERENCE ENGINE (MIN-MAX CIRCUIT)

Conventionally, transistor-based are architectures were adopted in the design of min-max circuits where some of which were voltage mode [25, 26] and others were current mode [27, 28]. In [29], the first memristor based min-max circuit was proposed.

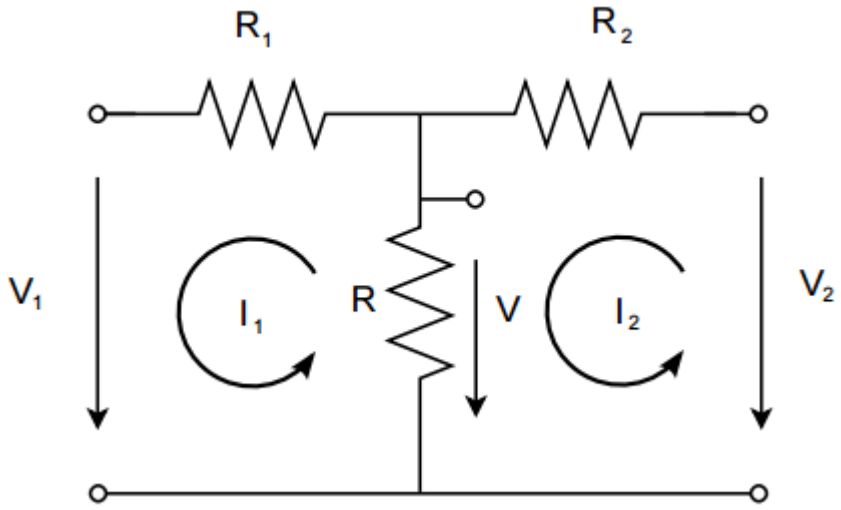

Figure 8: Voltage divider

I_1 and I_2 can be determined from Kirchhoff's Voltage Law (KVL) and using Cramer's rule:

$$I_1 = \frac{-V_1(R + R_2) + V_2 R}{\Delta} \qquad (2.12)$$

$$I_2 = \frac{V_1(R + R_1) - V_1 R}{\Delta}$$

Where $\Delta = (R + R_1)(R + R_2) - R^2 > 0$.

Hence, V can be interpreted as:

$$V = \frac{V_1 R_2 + V_2 R_1}{R_1 + R_2 + R_1 R_2 / R} \qquad (2.13)$$

If $R \gg \max(R_1, R_2)$, it can be shown that $V_1 < V < V_2$ or $V_2 < V < V_1$ for $V_1 < V_2$ or $V_2 < V_1$, respectively.

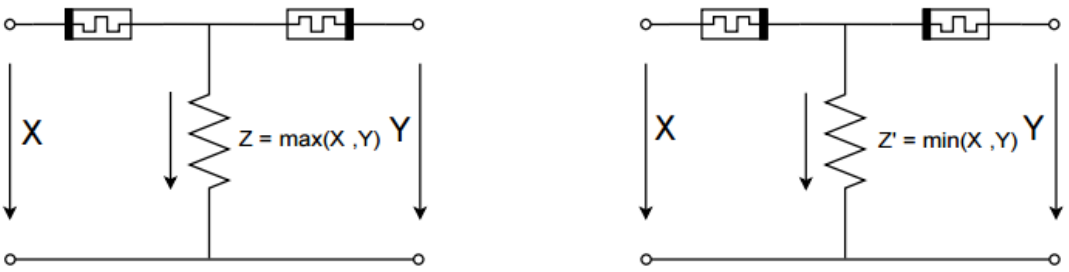

Figure 9: Max and Min circuits

Assume that R_1 and R_2 are replaced by memristors, $V_1 = X$ and $V_1 = Y$, and assume, according to the definition of the memristor, that the memristor switches ON if the current flows into the thick black line and switches OFF if the current flows outside the thick black line. For $Y < X$: $R_1 = R_{on}$ and $R_2 = R_{off}$ and the output voltage Z can be written as

$$Z = \frac{XR_{on} + YR_{off}}{R_{on} + R_{off} + R_{on}R_{off}/R} \tag{2.14}$$

For $R \gg R_{off} \gg R_{on}$, it can be readily shown that $Z = \max(X, Y) = X$. The minimum case can be derived using the same reasoning. It will be shown later in Chapter III that $R \gg R_{off} \gg R_{on}$ is satisfied via the inherent property of the memristor $R_{off} \gg R_{on}$ and leaving the output node floating which is equivalent to $R = \infty$.

II.4.2.3. DEFUZZIFIER

Unlike the fuzzifier and the inference engine, no attempts have been made to use memristors in the design of defuzzifiers. Hence, this section will review the most common transistor-based architectures for the design of defuzzifiers.

Center-of-gravity defuzzifiers are the most common type of defuzzifiers and are described as follows:

$$COG = \frac{\sum \mu_i . S_i}{\sum \mu_i} \tag{2.15}$$

where μ_i is the activation degree of every rule generated by the inference engine. S_i represents the singleton values of each output fuzzy sets and is usually defined by the system designer. COG is the output crisp value of the Defuzzifier. In essence, COG is the weighted average of the output fuzzy sets where the weights are the activation degrees of every rule.

There are two common techniques reported in the literature for the design of COG Defuzzifiers: the Follower-aggregator technique and current mode techniques.

Follower-Aggregator method

This structure is based on the use of Transconductance Amplifiers (TCA). The Follower-Aggregator structure is depicted in figure 10.

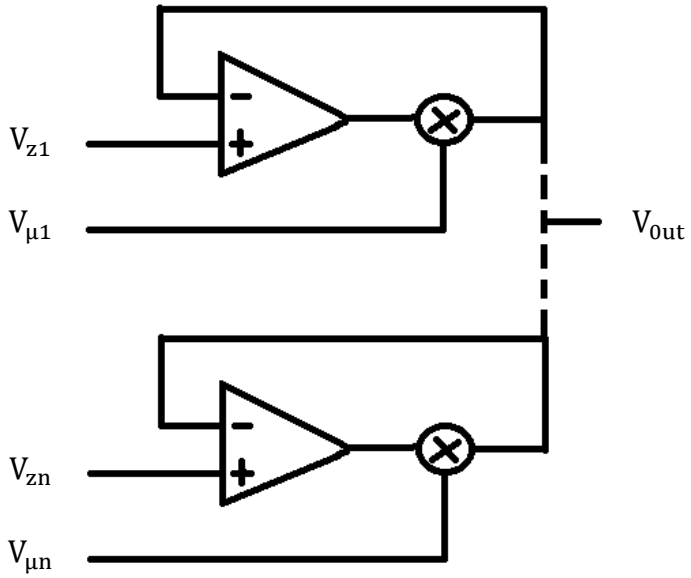

Figure 10: Follower-Aggregator based COG defuzzifier

In this circuit, the difference between V_{out} and V_zs are multiplied by the V_μs where V_z is the singleton value and V_μ is the activation degree. From KCL at the output node:

$$\sum_n I_i = 0 \tag{2.16}$$

$$\sum_n (V_{zi} - V_{out}).V_{\mu i} = 0 \tag{2.17}$$

Rearranging the term in (2.17), it can be shown that:

$$V_{out} = \frac{\sum V_{zi}.V_{\mu i}}{\sum V_{\mu i}} \tag{2.18}$$

Note that (2.18) is equivalent to (2.15) where COG is V_{out} , V_{zi} is S_i and $V_{\mu i}$ is μ_i.

Current mode method

The idea in the current mode approach is to represent the COG, the activation degrees and the singleton values as current signals. The advantage of the current mode domain is primarily attributed to the ease of the addition of the current signals. Figure 11 depicts the current mode circuit proposed in [30]

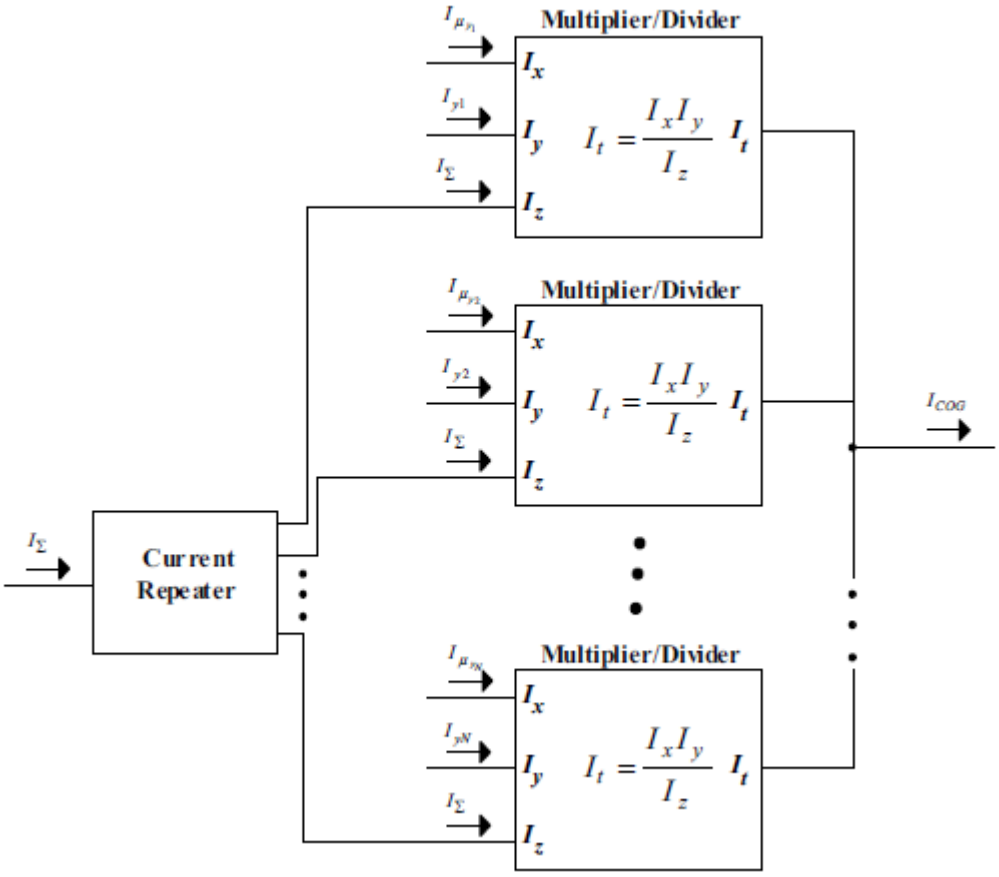

Figure11: Current mode defuzzifier

$I_{\mu y_i}$ is the activation degree of every rule, I_{y_i} is the singleton value and $I_\Sigma = \sum I_{\mu y_i}$.

As the branches are wired together, all branch current are summed such that:

$$\mathrm{I_{COG}} = \frac{I_{\mu y_1} . I_{y_1}}{\sum I_{\mu y_i}} + \cdots + \frac{I_{\mu y_n} . I_{y_n}}{\sum I_{\mu y_i}} = \frac{\sum I_{\mu y_i} . I_{y_i}}{\sum I_{\mu y_i}} \qquad (2.19)$$

Which, again, conforms with (2.15).

III. MODELING AND DESIGN OF MEMRISTOR-BASED MIN-MAX CIRCUITS

Neuromorphic systems have recently emerged as promising candidates for future computing paradigms [7]. Min-Max circuits are indispensable building blocks in Artificial Neural Networks and Fuzzy systems. For instance, the inference engine in fuzzy controllers that constitutes the decision making unit in such systems, is a min-max circuit. Conventionally, transistor-based architectures were adopted in the design of min-max circuits [7]. Several designs have been reported that primarily focus on reducing the area consumption where some of which were voltage mode [25, 26] and others were current mode [27, 28]. However, the miniaturized features of the memristor and the peculiar characteristics it exhibits have driven researchers to use it in state-of-the-art min-max circuits. In [29], a 2-input memristor-based min-max circuit was proposed. It was analytically proven that connecting two memristors in an antipodal fashion does implement min-max operation. In [31], the same structure was used as an experimental setup to investigate some of the characteristics of memristors. It was shown that a high 'ON' to 'OFF' ratio of memristance is required in order for the structure to function properly, i.e., compute min-max functions. In [32], a memristor-based min-max circuit was proposed as a potential application for memristor-based analog signal processing. In addition, the presence of a switching threshold for the memristor was highlighted without detailed analysis [31].

This chapter will discuss the theory of memristor-based min-max circuits. The basic concept and theory of operation will be discussed for the case of 2-input circuits in section III.1. Section III.2 will address 3-input circuits. The purpose of section III.2 is to (1) Establish a concrete analytical base for memristor-based min-max circuits from which a more general model can be derived, i.e., N-ary min-max circuits. (2) Derive analytical formulae to highlight the conditions and design constraints for the implementation of 3-input min-max circuits. (3) Analyze the effect of the memristor threshold on the design and constraints placed on the input voltages. Section III.3 will generalize the theory of min-max circuits to N-ary circuits and a closed form

expression for the effect of the memristor threshold, is derived. Section III.4 provides the summary of this chapter.

III.1. 2-INPUT MEMRISTOR-BASED MIN-MAX CIRCUITS

In general, the governing equation for the min-max operation is presented as follows:

$$X_{min} = \text{Min}\,(X_1, \ldots, X_n) \tag{3.1}$$

$$X_{max} = \text{Max}\,(X_1, \ldots, X_n) \tag{3.2}$$

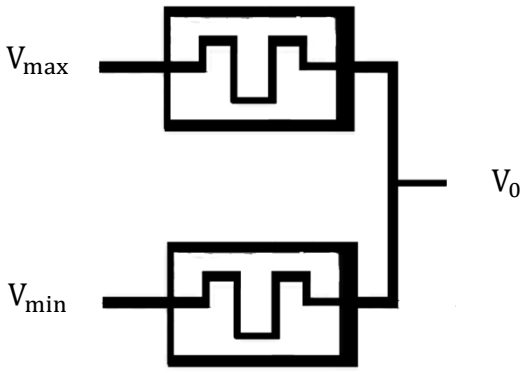

Figure 12: 2-input minimum circuit

If both v_1 and v_2 are equal, no current flows through the circuit and $v_0 = v_1 = v_2$. If $v_1 > v_2$, where $v_1 = v_{max}$ and $v_2 = v_{min}$, a current flows from the upper memristor to the lower one whereby it flows outside the thick line in the upper memristor and inside the thick line in the lower one. According to the definition of the memristor, the upper memristor will switch to 'OFF' acquiring the maximum resistance R_{off} while the lower one will switch to 'ON' acquiring the minimum resistance R_{on}. Since, by definition, $R_{on} \ll R_{off}$ or, equivalently, $G_{off} \ll G_{on}$ ($'G'$ is the memductance and defined as the reciprocal of the memristance such that $G = \frac{1}{R}$) and from Kirchhoff's law, it can be shown that the output voltage is computed as follows:

$$\mathbf{V_0} = \frac{\mathbf{V_{max}G_{off} + V_{min}G_{on}}}{\mathbf{G_{off} + G_{on}}} \approx \mathbf{V_{min}} \tag{3.3}$$

It can be readily shown that reversing the polarity of the memristors in figure 12 implements a maximum operation. Therefore, the forthcoming analysis will be only concerned with the minimum circuit.

An important characteristic of the memristor is the existence of a threshold below which no change in the memristance occurs. Hence, it is important to model the effect of the threshold on the operation of min-max circuits. Assuming a current controlled memristor [3, 12], there are two thresholds I_{on} and I_{off} that correspond to the memristor switching from 'ON' to 'OFF' or from 'OFF' to 'ON', respectively. I_{on} has a negative value corresponding to current flowing to the left while I_{off} is positive corresponding to current flowing to the right in figure. 12. Also, in general, I_{on} and I_{off} do not have to be equal. However, for the purpose of this work, in order to simplify the analysis, I_{on} and I_{off} are assumed equal which is known as symmetric switching [12]. This can be formalized as follows:

$$|I_{on}| = |I_{off}| = |I_t| \tag{3.4}$$

From now on, the absolute sign will be dropped and $|I_t|$ will be expressed as I_t.

In order to ensure the proper operation for the circuit, both memristors in figure 12 must be able to switch under all states. By inspection, since the circuit is a simple one, it can be shown that the lowest current occurs when both memristors are 'OFF' and the voltage difference between both inputs is minimal.

$$\frac{\partial V}{2R_{off}} > I_t \tag{3.5}$$

∂V is the minimum allowable difference between both inputs which reflects the resolution of the circuit, R_{off} is the maximum resistance of the memristor and I_t is the threshold current of the memristor.

III.2. COMPARISON WITH PREVIOUS TRANSISTOR-BASED STRUCTURES

In order to demonstrate the advantages rendered by the memristor-based implementation of Min-Max circuits over their transistor based counterparts, the proposed circuit is compared against the most commonly and widely used transistor based architecture known as WTA-LTA [26] structure where WTA stands for Winner-Takes-All and LTA stand for Looser-Takes-All. The base of comparison is

the area occupancy since, according to [25-28], it is the primary metric for hardware-based Fuzzy systems. Table 1 provides the comparison in terms of transistor/memristor count per input (note that memristors are smaller than transistors). Also, note that the memristor/transistor count rises linearly with the size of the circuit (number of inputs). For example, in the case of 2 inputs, the transistor count is 30/20 for Min/Max functions, respectively, while the memristor count is 2. This demonstrates that area savings become even more pronounced for multi-input circuits (as the number of inputs increase).

Table 1: Comparison between transistor/memristor-based min-max circuits

	WTA-LTA [26]		Memristor-based	
Transistor/Memristor count	Min circuit	Max circuit	Min circuit	Max circuit
	15 Transistors	10 Transistors	1 memristor	1 memristor

III.3. 3-INPUT MEMRISTOR-BASED MIN-MAX CIRCUIT

This section will discuss 3-input memristor-based min-max circuits. It will be proven that the structure in figure. 13 does implement min-max operation. Then, a case study will be presented on the effect of the memristor threshold on this circuit.

III.3.1. PROOF OF 3-INPUT MEMRISTOR-BASED MIN-MAX CIRCUIT

In [32], a 3-input min-max circuit was implemented and simulated with random inputs as a means of highlighting that min-max circuits are a potential application for memristors. In this section, it will be analytically proven that this structure does implement the min-max function for arbitrary input voltages, assuming that all currents are above the threshold of the memristor (assumption 'd' below). This is a crucial assumption that is the subject of Section III.2.2. The importance of the proof is

that (1) it discloses the conditions for the proper functioning of the min-max circuit (2) it proves that the structure is applicable for arbitrary inputs (3) it naturally lends itself to the analysis of the effect of the memristor threshold provided in section III.2.2. For convenience, the following analysis will use the reciprocal of the memristor which is the memductance (assumption 'a' below). Assumptions 'b' and 'c' below state the value of the memductance as a function of the direction of the current. Assumption 'e' states that the structure undergoes binary operation which is a core feature in min-max circuits.

Definitions and Assumptions:

a) $G = \frac{1}{R}$
b) If I_i is negative (left), then $G_i = G_{on}$
c) If I_i is positive (right), then $G_i = G_{off}$
d) $(|v_i - v_0|).G_i > |I_t|$.
e) G_{on} and G_{off} are the only allowed memductance values.

Figure 13 depicts a 3-input minimum circuit. From the definition of the memristor, $G_{on} \gg G_{off}$ where G_{on}/G_{off} are the maximum/minimum memductance (reciprocal of memristance). From Kirchhoff's Current Law (KCL) at v_0:

$$v_0.(G_1 + G_2 + G_s) = v_1.G_1 + v_2.G_2 + v_3.G_3 \quad (3.6)$$

Assume, without loss of generality, that $v_1 < v_2 < v_3$. Then, from (3.6), $v_1 \leq v_0 \leq v_3$.

Case (1): if $v_0 = v_1$, then $v_0 < v_2$ and $v_0 < v_3$ from the previous assumption. From (c), $G_2 = G_3 = G_{off}$ and I_2 and I_3 are positive. From KCL at v_0, $I_1 = I_2 + I_3$ and, therefore, I_1 is negative. From (b), $G_1 = G_{on}$. Substituting back in (3.6):

$$v_0.(G_{on} + 2G_{off}) = v_1.G_{on} + (v_2 + v_3).G_{0ff} \quad (3.7)$$

and therefore, $v_0 = v_1 = v_{min}$

Case (2): if $v_0 = v_2$, then $v_1 < v_0 < v_3$. From (b) and (c), $G_1 = G_{on}$ and $G_3 = G_{off}$, respectively. Since $v_0 = v_2$, then $I_2 = 0$ and $G_2 = G_x$, where G_x is undetermined memcductance. From (e), G_x is either G_{on} or G_{0ff}, Substituting in (3.6):

$$v_0.(G_{on} + G_x + G_{off}) = v_1.G_{on} + v_2.G_x + v_3.G_{off} \quad (3.8)$$

Since, $v_1 < v_2$ from the initial assumption, then from (3.8), $v_1 \leq v_0 < v_2$ and, accordingly, $G_1 = G_{on}$ and $G_2 = G_{off}$. Substituting in (3.6), equation (3.7) is obtained. Therefore, $v_0 = v_1 = v_{min}$.

Case (3): if $v_0 = v_3$, then $v_0 > v_1$ and $v_0 > v_2$. From (b), $G_1 = G_2 = G_{on}$, and I_1 and I_2 are negative. From KCL at node v_0, $I_3 = I_1 + I_2$ and, therefore, I_3 is positive. From (c), $G_3 = G_{off}$. Substituting back in (3.6):

$$v_0.(2G_{on} + G_{off}) = (v_1 + v_2).G_{on} + v_3.G_{0ff} \quad (3.9)$$

Since, $v_1 < v_2$ from the initial assumption, then from (3.9), $v_1 < v_0 < v_2$ and, accordingly, $G_1 = G_{on}$ and $G_2 = G_{off}$. Substituting in (3.6), equation (3.7) is obtained. Therefore, $v_0 = v_1 = v_{min}$.

Case (4): if $v_1 < v_0 < v_2$, knowing that $v_2 < v_3$, from (b) and (c), $G_1 = G_{on}$ and $G_2 = G_3 = G_{off}$, respectively. Substituting in (3.6), equation (3.7) is obtained. Therefore, $v_0 = v_1 = v_{min}$.

Case (5): if $v_2 < v_0 < v_3$, knowing that $v_1 < v_2$, from (b) and (c), $G_1 = G_2 = G_{on}$ and $G_3 = G_{off}$, respectively. Substituting in (3.6), equation (3.9) is obtained. Since, $v_1 < v_2$ from the initial assumption, then from (3.9), $v_1 < v_0 < v_2$ and, accordingly, $G_1 = G_{on}$ and $G_2 = G_{off}$. Substituting in (3.6), equation (3.7) is obtained. Therefore, $v_0 = v_1 = v_{min}$.

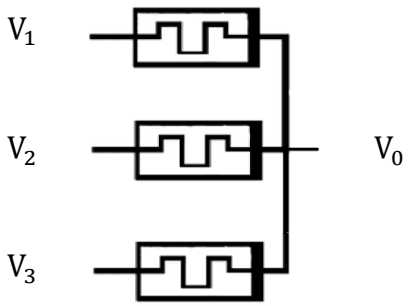

Figure 13: 3-input minimum circuit

III.3.2. EFFECT OF THE MEMRISTOR THRESHOLD ON 3-INPUT MEMRISTOR-BASED MIN-MAX CIRCUIT

In this section, the effect of the memristor threshold is analyzed for the specific case of 3-input circuits. The derived expressions establish a generic relationship between the memristor threshold current $'I_t'$ and the allowable applied voltage levels.

Assuming a memristor can only be ON 'N' or OFF 'F' (inherent assumption in min-max circuits), there are 8 possible transient states for the circuit at any point in time as the applied voltages change from the old values to the new ones. For example, NNF indicates that memristors 1 and 2 are ON while 3 is OFF. The goal is to ensure that the memristors are able to switch in order to, eventually, transmit the minimum voltage at the output node. Table 2 has the expressions for the currents in each memristor for each of the 8 states (using Kirchhoff's and Ohm's laws).

Table 2: Memristor currents in every state

State	I_1	I_2	I_3
NNN	$\frac{[2v_1-(v_2+v_3)]}{3R_{on}}$	$\frac{[2v_2-(v_1+v_3)]}{3R_{on}}$	$\frac{[2v_3-(v_1+v_3)]}{3R_{on}}$
NNF	$\frac{[v_1-v_2]}{2R_{on}}$	$\frac{[v_2-v_1]}{2R_{on}}$	$\frac{[2v_3-(v_1+v_2)]}{2R_{off}}$
NFN	$\frac{[v_1-v_3]}{2R_{on}}$	$\frac{[2v_2-(v_1+v_3)]}{2R_{off}}$	$\frac{[v_3-v_1]}{2R_{on}}$
NFF	$\frac{[2v_1-(v_2+v_3)]}{R_{off}}$	$\frac{[v_2-v_1]}{R_{off}}$	$\frac{[v_3-v_1]}{R_{off}}$
FNN	$\frac{[2v_1-(v_2+v_3)]}{2R_{off}}$	$\frac{[v_2-v_3]}{2R_{on}}$	$\frac{[v_3-v_2]}{2R_{on}}$
FNF	$\frac{[v_1-v_2]}{R_{off}}$	$\frac{[2v_2-(v_1+v_3)]}{R_{off}}$	$\frac{[v_3-v_2]}{R_{off}}$
FFN	$\frac{[v_1-v_3]}{R_{off}}$	$\frac{[v_2-v_3]}{R_{off}}$	$\frac{[2v_3-(v_1+v_2)]}{R_{off}}$
FFF	$\frac{[2v_1-(v_2+v_3)]}{3R_{off}}$	$\frac{[2v_2-(v_1+v_3)]}{3R_{off}}$	$\frac{[2v_3-(v_1+v_2)]}{3R_{off}}$

The next step is to find the worst case currents among the 24 expressions in Table 2. First, notice that the corresponding currents in states NNN and FFF have the same numerators but different denominators (larger denominator in FFF); if the conditions on the input voltages satisfy the currents in FFF are higher than I_t, they will automatically satisfy the currents in NNN. Similarly, states NFF, FNF and FFN eliminate NNF, NFN and FNN. By eliminating expressions with identical numerators but smaller denominators and assuming no two voltages are equal, the following will be the worst case currents and their corresponding states: I_1 in (FNF, FFN, FFF), I_2 in (NFF, FFN, FFF) and I_3 in (NFF, FNF, FFF). Solving for $I_1, I_2, I_3 > I_t$, it is concluded that:

$$|\mathbf{v_1} - \mathbf{v_2}| > \mathbf{R_{off}}.\mathbf{I_t} \quad (3.10)$$

$$|\mathbf{v_1} - \mathbf{v_3}| > \mathbf{R_{off}}.\mathbf{I_t} \quad (3.11)$$

$$|\mathbf{v_2} - \mathbf{v_3}| > \mathbf{R_{off}}.\mathbf{I_t} \quad (3.12)$$

In order to provide more insight into the analysis, an example is in order. Assume that at state FFF, arbitrary input voltages, $v_1 < v_2 < v_3$, are applied. From 'FFF' in the table provided that (3.10), (3.11) and (3.12) are satisfied, $|I_1| > |I_{on}|$ and the first memristor switches to R_{on}. $|I_3| > |I_{off}|$ and, in turn, the third memristor switches to R_{off}. The second memristor is contingent on the values of the applied voltages. However, in this particular state with the assumed applied voltages, whether $|I_2| > |I_{off}|$ or $|I_2| < |I_{off}|$, the second memristor remains R_{off} and the operation is correctly completed. On the other hand, assume the same voltages are applied but the structure was at state NNN; the first and third memristors will switch to $R_{on}$and R_{off}, respectively. The second memristor is, similar to 'FFF', contingent on the applied voltages. Yet, even if $|I_2| < |I_{off}|$ and it is not able to switch, the structure transfers to state NNF. From (3.10), $|I_2| > |I_{off}|$ in state NNF.

Algorithm 1. Modeling the effect of the threshold

1. Using KCL & Ohm's laws, Construct $I_i(v_1, v_2 v_3)$ for all states
2. Reduce the system to the states with the lowest currents (worst case currents "$I_{worstcase}$")
3. Solve the system of worst case currents (lowest) to get (3.10),(3.11)&(3.12)
4. (3.10),(3.11)&(3.12) ensure that, for arbitrary applied voltages:
   - Memristor with v_{min} switches to 'N'
   - Memristor with v_{max}switches to 'F'
   - **If** (memristor with the middle voltage is 'F')
     It remains 'F'
     Else if (it is 'N' & $I_{middle} > I_{off}(I_t)$)
     It switches to 'F'
     Else (it is 'N' & $I_{middle} < I_{off}(I_t)$)
     The system transfers its state to NNF, NFN or FNN

5. I_{middle} in (NNF, NFN, FNN) $>$ $I_{worstcase}$(worst case current) & the memristor will switch to 'F'

Algorithm 1 formalizes the modeling procedure. Simulations were conducted on a wide range of input vectors to validate the model with $I_t = 1\mu A$, $R_{off} = 200k\Omega$ and $R_{on} = 100\Omega$ as in [12]. Table 3 presents the results of the simulation.

Table 3: Simulation results for 3-input minimum circuit

v_1(v)	v_2 (v)	v_3 (v)	FP
0.1	0.4	0.7	-
0.3	0.1	0.5	-
0.35	0.7	0.5	FFN
0.1	0.2	0.4	FNN,FNF,FFF
0.2	0.45	0.5	FFF
0.5	0.7	0.35	NFF
0.6	0.35	0.2	FNF

where FP is the failing pattern. The failing pattern is the state at which the memristors are not able to switch and, accordingly, it malfunctions. Notice that the non-failing patterns have the difference between the input voltages equal to 0.2V which is $R_{off}I_t$.

III.4. N-ARY MEMRISTOR-BASED MIN-MAX CIRCUIT

This section will generalize the theory of memristor-based min-max circuits to N-input circuits. Section III.3.1 will provide a proof that, similar to 2-input and 3-input structures presented in figure 12 and figure 13, respectively, an N-input structure does implement min-max operation. Section III.3.2 will derive a closed form expression for the effect of the threshold on the circuit. It will be also shown that for the case of n=3 and under specific design conditions, the generalized model falls back to (3.10), (3.11) and (3.12) for 3-input memristor-based min-max circuits.

III.4.1. PROOF OF N-INPUT MEMRISTOR-BASED MIN-MAX CIRCUIT

Definitions and Assumptions:

(a) $G = 1/R$

(b) If I_i is negative(left), then $G_i = G_{on}$

(c) If I_i is positive(right), then $G_i = G_{off}$

(d) Switching time of the memristors is ignored (i.e., only steady state conditions are considered)

(e) G_{on} /G_{off} are the only allowed memductance values

(f) $G_{on} \gg G_{off}$irrespective of the number of memristors 'n'

(g) $(|v_i - v_0|).G_i > I_t$.

(a), (b), and (c) are inherent properties in the memristor device which were explained in previous sections. In essence, memristors possess a certain switching time (i.e., time taken to switch from G_{on} to G_{off} or vice versa) depending on the material characteristics. However, only steady state conditions are considered since the aim is to prove the viability of the structure not its switching dynamics. (e) is also an inherent characteristic in min/max circuits as mentioned before. (f) is considered valid throughout the development of the proof. However, it will be clear later in the section that (f) imposes design restrictions on N-ary min/max circuits. Finally, (g) is of a prime importance since it places restrictions on the applied voltages as a function of the memristor Current threshold. The effect of the threshold will be ignored in the proof. Yet, its effect will be studied in detail in the next section.

Suppose $v_{min} \leq v_i \leq v_{max}\ \forall v_i$ and $G_i \in \{G_{on}, G_{off}\}$, then from Kirchhoff's Current Law (KCL) and Ohm's Law applied at v_0:

$$v_0 = \frac{\sum_{i=1}^{n} v_i.G_i}{\sum_{i=1}^{n} G_i} \tag{3.13}$$

Suppose, without loss of generality, that $v_1 \leq v_2 \leq \cdots \leq v_n$, where $v_1 = v_{min}$ and $v_n = v_{max}$, then:

$$v_1 \leq v_0 \leq v_n \tag{3.14}$$

Suppose initially at the start of the operation that v_0 assumes an arbitrary value between v_1 and v_n such that:

$$v_1 \dots v_k < v_o \tag{3.15}$$

$$v_{k+1} \dots v_n > v_o \tag{3.16}$$

Then:

$$G_1 = \cdots = G_k = G_{on} \tag{3.17}$$

$$G_{k+1} = \cdots = G_n = G_{off} \tag{3.18}$$

Substituting back in (3.13):

$$v_{0N} = \frac{G_{on} \sum_{i=1}^{k} v_i + G_{off} \sum_{i=k+1}^{n} v_i}{kG_{on} + (n-k)G_{off}} \tag{3.19}$$

Where v_{0N} is the new output voltage.From (e), $G_{on} \gg G_{off}$ and (3.19) is reduced to:

$$v_{0N} = \frac{\sum_{i=1}^{k} v_i}{k} \tag{3.20}$$

Hence, $v_1 < v_{oN} < v_k$ and from (3.15), $v_k < v_o$. Therefore:

$$v_{0N} < v_o \tag{3.21}$$

Therefore, it is concluded that the process is recursive since for any arbitrary output voltage v_o, the new output voltage is v_{0N} and is less than v_o.

Also, note that when the output voltage changes from a value v_o to a new value v_{0N}, since $v_{oN} < v_k$, some memristors (Δ) switch from ON ($G_i = G_{on}$) to OFF ($G_i = G_{off}$). Therefore the change in the output voltage can be modeled as:

$$\Delta v = \frac{G_{on} \sum_{i=1}^{k} v_i + G_{off} \sum_{i=k+1}^{n} v_i}{kG_{on} + (n-k)G_{off}} - \frac{G_{on} \sum_{i=1}^{k-\Delta} v_i + G_{off} \sum_{i=k-\Delta+1}^{n} v_i}{(k-\Delta)G_{on} + (n-k+\Delta)G_{off}} \tag{3.22}$$

Such that $\Delta v = v_o - v_{oN}$. After some mathematical manipulation, (3.22) reduces to:

$$\Delta v = \frac{(k-\Delta) \sum_{i=k-\Delta+1}^{k} v_i - \Delta \sum_{i=1}^{k-\Delta} v_i}{k(k-\Delta)} \tag{3.23}$$

It was shown in (3.21) that v_{0N} is always less than v_0. However, it must be shown that the decrease in the output voltage Δv has a minimal finite value throughout the operation to ensure that the output voltage will eventually gravitate to v_{min} in a finite time. Since finding the minimal Δv might be mathematically tedious, especially, that the variables k and Δ in (3.23) can only assume integers, it is enough to show that Δv is always finite for all k and Δ and is always positive since we define $\Delta v = v_o - v_{oN}$ where $v_{oN} < v_o$. Since, by definition, $v_1 \le v_2 \le \cdots \le v_n$, each v_i in $\sum_{i=k-\Delta+1}^{n} v_i$ is, individually, larger than each v_i in $\sum_{i=1}^{k-\Delta} v_i$. Hence, assuming $v_{MIN} = \text{minimum}(v_{k-\Delta+1} \dots . v_n) \; \forall v_i$ and $v_{MAX} = \text{maximum}(v_1 \dots . v_{k-\Delta}) \forall v_i$, then from (3.23):

$$\Delta v \geq \frac{\Delta(v_{MIN} - v_{MAX})}{k} \tag{3.24}$$

Where $v_{MIN} > v_{MAX}$. Since, by definition, k and Δ are finite integers between '1' and 'n' and k>Δ, then Δv is a finite positive number. Note that v_{MIN}/v_{MAX}should not be confused with v_{min}/v_{max} which are the global minimum/maximum voltages in the system. Therefore, it can be shown that since:

- The N-ary minimum circuit has a minimum voltage (boundary) "v_{min}"
- The new output voltage is always smaller than the old output voltage "$v_{0N} < v_o$" and the process is recursive
- Δv is a finite positive number

It can be concluded that $v_0 = v_{min}$.

(f) is crucially important for the proper functioning of the circuit. In essence, this assumption is what allows the cancellation of the G_{off} term throughout the proof. However, this cancellation is not always valid but constrained by the values of applied voltages $'v_i'$ and the number of memristors 'n'. These constraints can be derived from (3.19) yielding:

$$\frac{n-k}{k} \ll \frac{G_{on}}{G_{off}} \tag{3.25}$$

$$\frac{\sum_{i=k+1}^{n} v_i}{\sum_{j=1}^{k} v_j} \ll \frac{G_{on}}{G_{off}} \tag{3.26}$$

To ensure the proper functioning of the system, (3.25) and (3.26) have to hold true under worst case state which is when $k = 1$, $v_j = v_{min}$ and $v_i = v_{max}$ (largest possible left hand side in (3.25) and (3.26)). Substituting back in (3.25) and (3.26):

$$n - 1 \ll \frac{G_{on}}{G_{off}} \tag{3.27}$$

$$(n-1)\left(\frac{v_{max}}{v_{min}}\right) \ll \frac{G_{on}}{G_{off}} \tag{3.28}$$

III.4.2. EFFECT OF THE MEMRISTOR THRESHOLD ON N-ARY MEMRISTOR-BASED MIN-MAX CIRCUIT

As mentioned earlier in the introduction and in section III.2.2, the threshold behavior of the memristor poses a crucial challenge in the design of memristor-based min-max circuits. Given a specific Current controlled memristor with threshold current $'I_t'$, ON resistance $'R_{on}'$, Off resistance $'R_{off}'$ and arbitrary size 'n', design constraints on the values of the input voltages are derived. This problem will be approached, first, analytically. Then, a MATLAB code is developed to validate the results computationally.

III.4.2.1. ANALYSIS OF THE EFFECT OF THE MEMRISTOR THRESHOLD

In order to ensure the proper functioning for the circuit, two conditions have to be met: (i) at least one memristor $G_i = G_{on}$ where $v_i = v_{min}$ and (ii) $G_j = G_{off} \forall\, j$ where $v_j \neq v_{min}$. Intuitively, if a group of memristors have the minimum voltage applied to them, from Kirchhoff's law, they become parallel and if only one memristor is ON, the effective resistance of the whole group is G_{on}. However, if only one memristor that has a voltage higher than the minimum applied to it is ON, it will contribute to the output voltage by pulling up the output node and, subsequently, the output voltage deviates from the minimum and the circuit malfunctions.

In general, based on Ohm's law and writing v_0 as a weighted average of all inputs, any current, for instance I_1, can be presented as:

$$I_1 = G_1.\left(v_1 - \frac{\sum_{j=1}^{n} v_j.G_j}{\sum_{j=1}^{n} G_j}\right) \tag{3.29}$$

Rearranging the terms in (3.29):

$$I_1 = \frac{G_1}{G_1 + \cdots + G_n}[G_2(v_1 - v_2) + \cdots + G_n(v_1 - v_n)] \tag{3.30}$$

Let $v_i = v_{min} + m_i.\partial V = v_{max} - x_i.\partial V$ where m_i and x_i are integers and $0 \leq m_i, x_i \leq \frac{v_{max}-v_{min}}{\partial V}$ such that for a given memristor G_i, $m_i = 0$ corresponds to $v_i = v_{min}$ and $m_i = \frac{v_{max}-v_{min}}{\partial V}$ corresponds to $v_i = v_{max}$ while $x_i = 0$ corresponds to $v_i = v_{max}$ and $x_i = \frac{v_{max}-v_{min}}{\partial V}$ corresponds to $v_i = v_{min}$. For example, assume a three input circuit with input vector **V**= (0.1V, 0.2V, 0.3V). Then, $v_{min} = 0.1$,

$\partial V = 0.1$, $m_2 = 1$ and $m_3 = 2$. By the same token, v_{max} and x_i can be deduced accordingly. It is important to note that v_{min}/v_{max} , in this particular analysis, are not necessarily the global minimum/maximum for the circuit. However, v_{min}/v_{max} represent the minimum/maximum values of the input vector applied to the circuit at a particular instant in time. For instance, while the global minimum/maximum voltages for the circuit might be 0V/1V, in this example, v_{min}/v_{max} = 0.1V/0.3V.

Assume, without loss of generality, that $v_1 = v_{min}$ and $v_n = v_{max}$

$$I_1 = \frac{-G_1 \partial V}{G_1 + \cdots + G_n}[G_2 m_2 + \cdots + G_n m_n] \tag{3.30}$$

$$I_n = \frac{G_n \partial V}{G_1 + \cdots + G_n}[G_1 x_1 + \cdots + G_{n-1} x_{n-1}] \tag{3.31}$$

Note that the negative sign in (3.30) indicates, for example, that the current is flowing to the left in the upper most memristor in figure 13, since, by definition, $v_1 = v_{min}$. Therefore, henceforth, the negative sign will be dropped since it indicates no more than the direction of the current. Also, note that $m_1 = 0$ since $v_1 = v_{min}$. (3.31) can be inferred accordingly for the case of $v_n = v_{max}$ based on the earlier discussion.

Condition (i):

Let an arbitrary number of memristor 'k' be 'ON' Such that (3.30) can be rewritten as:

$$I_1 = \frac{G_{off} \partial V}{kG_{on} + (n-k)G_{off}}[G_{on} \sum_{k} m_i + G_{off} \sum_{n-k-1} m_j] \tag{3.32}$$

Notice that the goal is to minimize (3.32) in order to find the worst case current (lowest current that results in circuit malfunctioning) and ensure that it is higher than the 'I_t'. This will ensure that condition (i) is satisfied. There are two ways to achieve this which are (1) vary 'k' and 'n' such that the number of 'ON' and 'OFF' memristors change (2) vary the values of m_i and m_j which, essentially, change the values of the input voltages. Also, notice that both ways are independent and can be treated separately since the combination of 'ON' and 'OFF' memristors during the transition state is independent from the steady state voltage values at the inputs (for more detail, please refer to section III.2.2).

Case (1): $1 \leq k \leq n-1$:

$$I_1 \approx \frac{G_{off}\,\partial V}{kG_{on}}[G_{on}\sum_{k} m_i + G_{off}\sum_{n-k-1} m_j] \tag{3.33}$$

Notice that, from (i), only one memristor is required to be ON. Hence, if for any of the 'k' memristors that are 'ON', $m_i = 0$, (i) will be automatically satisfied and there would be no need to worry about switching of the memristors. Conversely, the analysis is concerned with the case were none of the memristors for which $v_i = v_{min}$ is 'ON' which means that minimum $(\sum_k m_i) = k$ (i.e, $m_i \geq 1\ \forall\ i \in k$). Hence, (3.33) can be written as:

$$I_1 \approx \frac{G_{off}\,\partial V}{kG_{on}}[kG_{on}] \approx G_{off}\,\partial V \tag{3.34}$$

Case (2): $k = 0$:

Equation (3.32) is reduced to:

$$I_1 = \frac{\partial V}{n}[G_{off}\sum_{n-1} m_j] \tag{3.35}$$

If $\text{minimum}\left(\sum_{n-1} m_j\right) = 0$, this would mean that all voltages are equal to each other and equal to v_{min}, in which case no current flows and the voltage is transmitted normally to the output. Hence, $\text{minimum}\left(\sum_{n-1} m_j\right) = 1$ and (3.33) can be written as

$$I_1 = \frac{G_{off}\,\partial V}{n} \tag{3.36}$$

Condition (ii):

Using the same argument in (3.32), (3.31) can be described as:

$$I_n = \frac{G_{on}\,\partial V}{kG_{on} + (n-k)G_{off}}[G_{on}\sum_{k-1} x_i + G_{off}\sum_{n-k} x_j] \tag{3.37}$$

Case (1): $k = 1$:

$$I_n \approx \partial V[G_{off}\sum_{n-1} x_j] \tag{3.38}$$

Following the same reasoning as before, minimum$(\sum_{n-1} x_j) = 1$ and, accordingly,

$$I_n \approx G_{off}\,\partial V \tag{3.39}$$

Case (2): $1 < k \leq n-1$**:**

$$I_n = \frac{\partial V}{k}\left[G_{on}\sum_{k-1} x_i + G_{off}\sum_{n-k} x_j\right] \tag{3.40}$$

In order to minimize (3.40), as mentioned earlier, only one of the two summations can be zero, not both. Hence, since $G_{on} > G_{off}$, minimum($\sum_{k-1} x_i$) = 0 and minimum($\sum_{n-k} x_j$) = 1.

$$I_n \approx \frac{G_{off}\,\partial V}{k} \tag{3.41}$$

As mentioned earlier in condition (ii), $G_j = G_{off} \forall\, j$ where $v_j \neq v_{min}$. While (3.41) was only concerned with the memristor with the maximum voltage applied to it, once this memristor switches to OFF, the memristor with lower voltage than the maximum becomes the new maximum voltage (i.e., $k \rightarrow k-1$). Since the minimum value for the current 'I_n' is what is sought in this analysis, (3.41) can be further minimized to (largest possible denominator). Formally, $\frac{G_{off}\,\partial V}{n-1} \leq I_{v_j \neq v_{min}} \leq \frac{G_{off}\,\partial V}{2}$. Hence, minimum ($I_{v_j \neq v_{min}}$) = $\frac{G_{off}\,\partial V}{n-1}$.

$$I_n \approx \frac{G_{off}\,\partial V}{n-1} \tag{3.42}$$

Case (3): $\mathbf{k = n}$**:**

$$I_n \approx \frac{G_{on}\,\partial V}{n} \tag{3.43}$$

Hence, the worst case current can be computed as minimum ($G_{off}\,\partial V, \frac{G_{off}\,\partial V}{n}, \frac{\partial V G_{off}}{n-1}, \frac{\partial V G_{on}}{n}$) which is obviously $\frac{G_{off}\,\partial V}{n}$ and therefore:

$$\frac{\partial V}{n R_{off}} > I_t \tag{3.44}$$

Therefore, given a memristor with OFF resistance 'R_{off}' and threshold current 'I_t', there exists a trade-off between the minimum allowed voltage difference '∂V' and the size of the circuit 'n'. Also, note that substituting n=2, we arrive at (3.5) which is the case for two input circuits. Equations (3.10), (3.11) and (3.12) are special cases from

(3.35). Note that in the analysis of the 3-input case, it was assumed that no two voltages are equal $v_1 < v_2 < v_3$. For the case of n=3, (3.35) can be written as:

$$I_1 = \frac{\partial V}{3} G_{off}[m_2 + m_3] \tag{3.45}$$

Invoking that assumption, if $v_1 = v_{min}$, $m_2 = 1$ and $m_3 = 2$. Substituting in (3.45), we get (3.10), (3.11) and (3.12).

III.4.2.2. A SIMULATIVE ANALYSIS OF THE EFFECT OF THE MEMRISTOR THRESHOLD

In order to validate the model in (3.44), an algorithm is developed that emulates the circuit operation. The algorithm initializes a current threshold 'I_t' and runs an exhaustive simulation over all circuit pictures where a picture is defined as a combination of memristor states and applied input voltages. For every picture, the memristors are allowed to switch until the output voltage stabilizes at its final value (i.e., no more switching of the memristors is taking place). If the final output voltage is the minimum voltage, this particular picture is said to have succeeded. A picture succeeds when the output voltage gravitates to v_{min}. In contrast, a picture fails when the system is stuck at an output voltage that is not v_{min}. For example, for input voltages V1= 0.1, V2= 0.2 and V3= 0.3, the memristances should switch to M1= R_{on}, M2= R_{off} and M3= R_{off}. If for instance the system stabilizes at M1= M2= M3= R_{off} and the system cannot switch any further, the output voltage is not the minimum voltage and the picture is said to have failed. This failure occurs because a memristor or more are not able to switch because their currents are below the threshold current 'I_t'. Algorithm 2 presents a pseudo code for this procedure.

Algorithm 2. Modeling the effect of the threshold

1. Define G_{on} and G_{off}
2. Define threshold current: i_t
3. **Loop1:** $(v_1, v_2, v_3, G_1, G_2, G_3)$
4. Calculate v_{min}= minimum(v_i)
5. Compute $v_0 = \frac{\sum v_i G_i}{\sum G_i}$

6. Flag **UP**

7. Loop2: While Flag **UP**

8. Compute $I_i = (v_i - v_0) \times G_i$

9. Update G_i

10. Compute $v_{0n} = \frac{\sum v_i G_i}{\sum G_i}$ (updated G_i)

11. **If** $v_{0n} == v_0$ (Check if the output stabilized)

12. Flag **DOWN**

13. **If** $v_{0n} - v_{min} > \varepsilon$ (Check for failing states)

14. Output Failing state

15. **End If**

16. **Else**

17. $v_{0n} = v_0$

18. **End If**

19. **End Loop2**

20. **End Loop1**

Algorithm 2 generates the output failing picture(s) for every choice of I_t. Hence, I_t is decreased until no failing pictures occur as mentioned above. The choice of 'ε' is critical since it is the failing criterion (i.e., the criterion that decides whether failing pictures exist or not). Moreover, its value is a function in the applied voltages and the size of the circuit. Therefore, the next section is devoted to developing a mathematical model for ε. Note that although, theoretically speaking, $v_0 = v_{min}$, this is never precisely true since the 'ON' resistance of the memristor R_{on} has some finite value. Yet, the difference is extremely small such that $v_0 \approx v_{min}$.

In order to develop a model for the effect of the threshold on the system, two curves are plotted. Figure 14 plots I_t against $1/R_{off}$ for some specific size of the min-max circuit (n=3) and a constant voltage step ($\partial V = 0.1V$), where the voltage step is the minimum voltage difference between any two allowed voltage levels. Figure 15 plots I_t against ∂V at constant $R_{off} = 200K\Omega$ [33] and different sizes for the min-max circuit namely: n=2, 3, 4, 5, 6, 10.

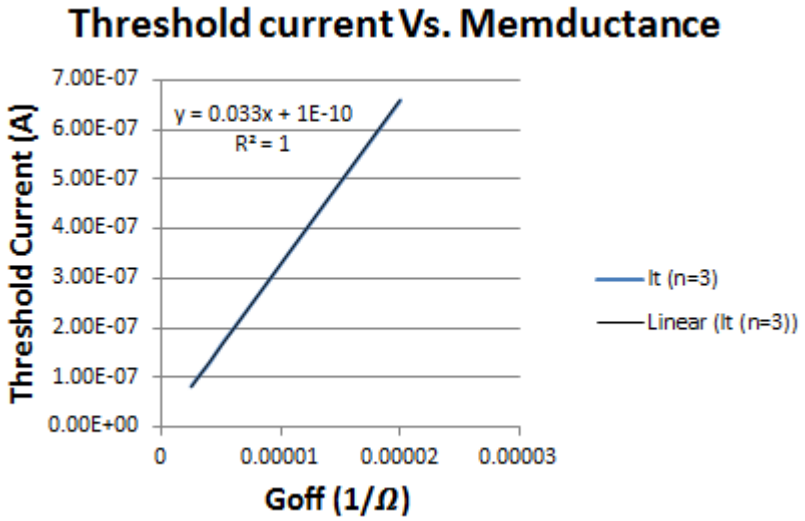

Figure 14: Threshold Current vs Memductance at $R_{off} = 200K\Omega$ and n = 3

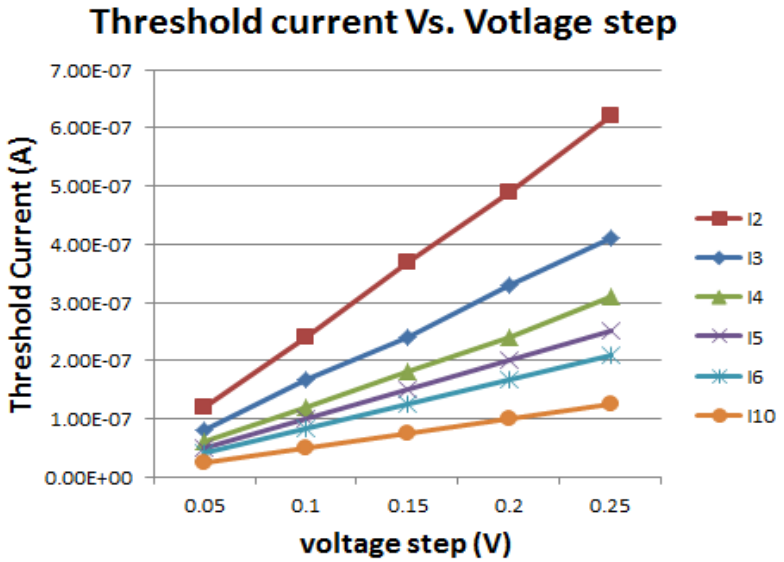

Figure 15: Threshold current Vs. voltage step at $R_{off} = 200K\Omega$ and different 'n'. I2 is the curve associated with n=2.

Note that figures 14 and 15 depict a strictly linear relation with almost perfect correlation (R^2 =1 in figure14 and R^2 >0.99 in figure 15 due to minor numerical errors) which is in agreement with (3.44). It is no surprise, however, that the relation is perfectly linear. In essence, there is a specific combination of memristor states and applied voltages that results in the worst case state for all sizes of the circuit. This combination is when v_{min} is applied to 'n − 1 'memristors, $v_{min} + \partial V$ is applied to the n^{th}memristor and all memristors have the maximum resistance R_{off}. Intuitively, the worst case picture occurs when the circuit has the maximum resistive state, all

memristors are 'OFF', minimum potential difference applied to the circuit, only one memristor has higher voltage than the rest of the memristors and the difference is minimal (∂V).

As previously stated, the choice of ε is crucial in the proposed algorithm. Hence, it is important to properly model it. In general, for arbitrary minimum input voltage v_{min} and k replicas of minimum voltage applied (note that you can have the minimum voltage applied to more than one memristor in the min-max circuit), the general equation for the output voltage v_0 is modeled as follows:

$$v_0 = \frac{kG_{on}v_{min} + G_{off}(\sum_{n-k} v_i)}{kG_{on} + (n-k)G_{off}} \tag{3.46}$$

Rearranging the terms in (3.46):

$$kG_{on}(v_0 - v_{min}) = G_{off}[(\sum_{n-k} v_i) - (n-k)v_0] \tag{3.47}$$

Notice that the third term in the left hand side in (3.47) represents the output voltage deviation from the minimum voltage such that $\text{deviation} = v_0 - v_{min}$ and, therefore, (3.47) can be written as follows:

$$\text{deviation} = \frac{((\sum_{n-k} v_i) - ((n-k)v_0))G_{off}}{kG_{on}} \tag{3.48}$$

Since in algorithm.2, the circuit fails when the deviation is more than ε, ε has to be the maximum possible deviation, $\varepsilon = \max(v_0 - v_{min})$, for the output voltage from the minimum so that further deviation would mean that the circuit failed. In order to maximize (3.48), substitute $v_i = v_{max}$, $k = 1$ and $v_0 = v_{min}$:

$$\varepsilon = (n-1)[v_{max} - v_{min}](\frac{G_{off}}{G_{on}}) \tag{3.49}$$

Equation (3.49) shows that the maximum deviation under correct functionality occurs when only one input has the minimum input voltage v_{min} (no replicas) applied to a specific memristor and all the other $'n-1'$ memristors have the maximum voltage in the circuit 'v_{max}' applied to them. In order to validate the model in (3.49), a numerical algorithm (algorithm 3) was developed to model ε. The model was compared to Spice simulations and algorithm 3 and showed excellent match. The results are tabulated in Table 4 where $v_{min} = 0$ and $v_{max} = 1$.

Table 4: Validation for ε

# of memristors	Model	Algorithm.3.	Spice
2	0.0005	0.0005	0.00049
3	0.001	0.0099	0.00099
4	0.0015	0.0015	0.00149
5	0.002	0.002	0.00199
6	0.0025	0.0025	0.00249
10	0.0045	0.0045	0.00447

Algorithm. 3. Presents a numerical method to generate ε. It simulates the circuit behavior under different voltage vectors and memristor states (exhaustive simulation as in algorithm. 2.). However, it does not account for the effect of the threshold so that it ensures the correct functionality for the circuit and, thus, ε can be calculated.

Algorithm.3. Epsilon ' ε ' generation

1. Define Array **E**
2. **Loop1**: (v_i, G_i)
3. Calculate v_{min}= minimum(v_i)
4. Compute $v_0 = \frac{\sum v_i G_i}{\sum G_i}$
5. Flag **UP**
6. **Loop2**: While Flag **UP**
7. Compute $I_i = (v_i - v_0) \times G_i$
8. Update G_i
9. Compute $v_{0n} = \frac{\sum v_i G_i}{\sum G_i}$ (updated G_i)
10. **If** $v_{0n} == v_0$
11. Flag **DOWN**
12. deviation $= v_{0n} - v_{min}$
13. Store deviation in Array **E**: **E**= [**E** deviation]
14. **End If**
15. $v_{0n} = v_0$
16. **End Loop2**
17. **End Loop1**
18. Output ε = maximum (**E**)

III.5. SUMMARY

This chapter discussed modeling, analysis and design of memristor-based min-max circuits. First, the basic theory and working principle were analyzed for the simple case of 2-input circuit. The effect of the memristor threshold was also modeled. Second, the theory of memristor-based min-max circuit was extended to 3-input circuits as an intermediate step towards generalizing the theory for N-ary min-max circuits. Once more than two memristors are connected, different currents are established in the circuit which makes the analysis much more complicated and, hence, the need for an intermediate stage was essential. Then, the theory was generalized to N-ary min-max circuits. It was shown that the same structure used for 2-input circuits can be extended to an arbitrary number of inputs 'n' under the proper design constraints. In addition, the effect of the memristor threshold was modeled and a closed form expression was derived. It was shown that for a given memristor with a specified OFF resistance and threshold current, there exists a trade-off between the number of inputs and the resolution of the circuit. The derived model was validated against a numerical simulation and an excellent match was observed.

IV. MEMRISTOR-BASED CENTER-OF-GRAVITY (COG) DEFUZZIFIER

Fuzzy controllers have for so long played a key role in the design of modern control systems. Software-based fuzzy systems have been traditionally implemented using standard digital processors [34]. However, when high speed, low area occupancy and/or low power consumption are required, hardware-based fuzzy controllers are often preferable [30]. Fuzzy controllers consist of three fundamental building blocks which are the fuzzifier unit, the inference engine unit and the defuzzifier unit [35] as depicted in figure 16.

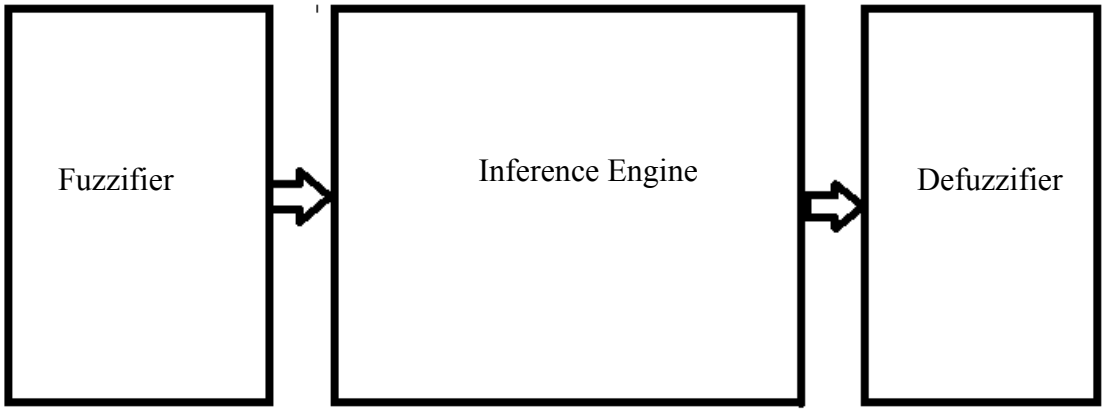

Figure 16: Block diagram for Fuzzy controllers

Center-Of-Gravity (COG) defuzzifiers have always been a bottleneck in fuzzy chips due to the requirement of multiplier/divider circuits that are both area and power consuming [36, 37, 38].

Four approaches were historically adopted for the hardware implementation of COG defuzzifier circuits. First, the fully digital technique was proposed in [38] in which multiplication/division were performed via iterative addition/subtraction. However, this brings about significant speed limitations [36] and occupies a relatively large chip area [39]. Second, the voltage follower aggregator structure was utilized [37]. This structure does not contain divider circuits which makes it advantageous. However, it uses several operational amplifiers which makes it area consuming. Third, current mode approach was proposed. It has the advantage of simple addition/subtraction of signals [30]. Yet, voltage mode approach is often preferred

since most sensors and auxiliary devices communicate with fuzzy systems in voltage mode [37].

Other voltage mode structures were proposed such as in [40]. However, they still occupy a relatively large area. In [35], a new voltage mode structure was proposed based on the voltage controlled resistor in which transistors are forced to work in the triode region. This structure is very innovative and provides a significant area reduction but might suffer some robustness issues due to transistor mismatch resulting form process variations.

A COG defuzzifier unit is proposed in this chapter using memristors. The proposed circuit uses two OPAMPS, one PMOS and memristors. The main idea of the design is to store the output fuzzy sets (which are singletons in this case) in the memductance (conductance of the memristor). Hence, the multiplication operation is reduced to Ohm's Law which saves significant area as it avoids complicated multiplier structures present in previous work. Also, the circuit avoids analog processing which makes it robust.

Section IV.1 will present a brief background about the theory of center-of-gravity (COG) defuzzifiers. The proposed design along with the design equations are presented in Section IV.2. Simulations and discussions are presented in Section IV.3. Section IV.4 has the summary of the chapter.

IV.1. CENTER OF GRAVITY DEFUZZIFICATION

The role of defuzzifiers in fuzzy logic controllers is to translate the fuzzy values inferred by the inference engine into output crisp values. The general form of COG defuzzification in case of singleton output fuzzy sets is as follows [30]:

$$\mathrm{COG} = \frac{\sum \mu_i . S_i}{\sum \mu_i} \tag{4.1}$$

where μ_i is the activation degree of every rule, S_i represents the singleton values of each output fuzzy set and COG is the output crisp value of the defuzzifier.

IV.2. PROPOSED DEFUZZIFIER CIRCUIT

The operation of the proposed circuit is divided into two phases: programming phase and operation phase.

In the programming phase, external signals are applied to the memristors in OPAMP 'B' in figure 17 in order to program them with the required singleton values, S_i, which define the output fuzzy sets. More precisely, the singletons are programmed to the memconductance which is the reciprocal of the memristance. Owing to the analog programmability of the memristor, the memristors can be programmed to a gradient of values as in [41] as shown in (4.1):

$$1/R_{off} < S_i < 1/R_{on} \tag{4.1}$$

Since the memristor exhibits a threshold behavior, the external voltage signals applied to program the memristors should exceed the threshold.

In the operation phase, the structure is static, i.e., memristors behave as regular resistors with no change in memristance. Therefore, the inference voltage, μ_i, coming from the inference engine should be below the threshold. The advantage of this setting is twofold: on one hand, it ensures that the singleton values are not distorted by the applied inference voltages, otherwise the circuit would malfunction; on the other hand, this enables the use of memristors instead of resistors in OPAMP 'A', except for the feedback resistor. This will provide significant area reduction, especially, that R_i (the summing memristors fed to the negative terminal of OPAMP'A') scale with the number of input weights, μ_i, while R_f does not. This translates into more area savings as the output fuzzy sets increase.

Figure 17 depicts the proposed design. OPAMP 'A' is used to sum the activation degrees (inference voltage) of the rules scaled by a factor "α" corresponding to the ratio between the feedback resistor R_f and the summing memristors R_1-R_N as shown in (4.3).

$$V_x = -R_f\left[\frac{\mu_1}{R_1} + .. + \frac{\mu_n}{R_n}\right] = -\alpha\sum \mu_i \tag{4.2}$$

for $$R_1 = \cdots = R_n = R$$

where

$$\alpha = \frac{R_f}{R} \tag{4.3}$$

R_f is then fed to the gate of the PMOS transistor. In order to ensure the correct functionality of the circuit in OPAMP 'B', the PMOS transistor has to operate in the deep triode region [42] and thereby can be approximated as a voltage controlled resistor. Therefore, the current I_{SD} is modeled as follows:

$$I_{SD} = k'_p \frac{w}{L}\left(\alpha \sum \mu_i\right).(-V_o) \tag{4.4}$$

where

$$\alpha \sum \mu_i - v_{th} \approx \alpha \sum \mu_i \tag{4.5}$$

For $$\alpha \sum \mu_i \gg v_{th}$$

Also, by applying Kirchhoff's Current Law (KCL) at the negative terminal of OPAMP'B':

$$I_x = \sum \mu_i . S_i \tag{4.6}$$

Since the input impedance of an OPAMP is very high:

$$I_x = I_{SD} \tag{4.7}$$

Substituting (4.4) and (4.6) into (4.7) and rearranging the terms:

$$V_o = -\frac{\sum \mu_i . S_i}{\beta . \sum \mu_i} \tag{4.8}$$

where

$$\beta = k'_p \frac{w}{L} \alpha \tag{4.9}$$

Therefore, it is obvious from (4.8) that V_o is the center-of-gravity (COG) scaled by a factor $1/\beta$ which is adjusted subject to the design requirements and constraints.

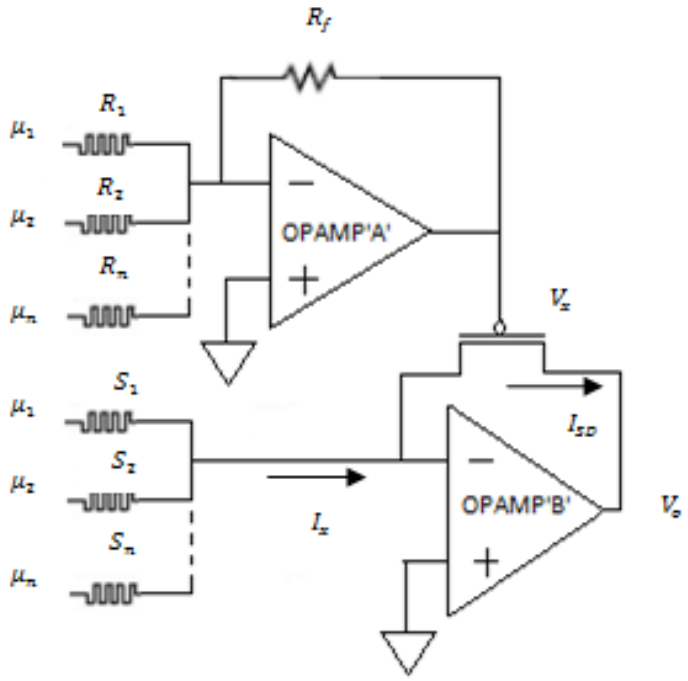

Figure 17: Proposed COG Defuzzifier circuit

IV.3. SIMULATION RESULTS AND DISCUSSIONS

In this section, several simulations are presented to demonstrate the functionality of the proposed design. Then, design constraints will be discussed. Finally, comparisons with existing designs in the literature are conducted.

Table 5 shows a number of case studies that were conducted on a two-input COG defuzzifier to demonstrate the functionality of the proposed design. Simulations were done using the Eldo simulator from Mentor Graphics. The simulations show a maximum error of less than 4% for the conducted case studies.

An important design parameter that should be carefully chosen is α, which represents the feedback gain of OPAMP'A'. α should be chosen such that the voltage fed to the gate of the PMOS is high enough to ensure its operation in the deep triode mode. Also, when multiplied by $\sum \mu_i$, the result should be much bigger than v_{th} of the PMOS to allow for the approximation in (4.5). Also, it is work mentioning that the proposed design is inspired by [43].

Another important parameter is the range of allowable values for the resistance of the memristors in OPAMP'B'. Equation (4.1) represents the total range of the memristance (resistance of the memristor). However, the effective range chosen for

the operation of the circuit should also ensure that the current flowing in the PMOS is low so that it operates in the deep triode mode and does not saturate.

It was found in the literature that a major drawback in the COG defuzzifier is its relatively large area which led to its implementation on separate processors [36]. Research ever since has evolved in the direction of designing smaller COG defuzzifiers to enable their integration with the other modules in the fuzzy controller. In order to mitigate the area shortcomings, several designs were proposed such as in [35] that rely on extensive analog techniques that suffer the disadvantage of reduced precision and sensitivity to mismatch and temperature [44].

The proposed design is compact, primarily, due to the use of memristors that scaled down beyond 10nm [19] which is much smaller than state-of-the-art transistor technology node. Also, the proposed design avoids extensive analog processing present in previous works. The proposed defuzzifier is compared against the most widely used architecture, the Follower aggregator architecture in [37], based on transistor/memristor count.

Table 5: 2-input COG defuzzifier

Simulations for the proposed COG defuzzifer						
μ_1 (V)	μ_2 (V)	$1/S_1$ kΩ	$1/S_2$ kΩ	COG (Theoretical) (mV)	COG (Observed) (mV)	Error (%)
0.8	0	80	30	-18.5	-18	2.7
0	0.8	80	30	-49.4	-48.4	2
0.1	0.7	80	20	-67.13	-66.5	0.9
0.2	0.6	100	100	-14.8	-14.3	3.3
0.2	0.6	40	60	-27.7	-27	2.5

Table 6: Comparison between the Follower-Aggregator and the proposed circuit

	Follower-Aggregator [37]		Proposed design	
Transistor/Memristor count	Transistors per input	Overhead Circuitry	Memristors per input	Overhead circuitry
	17 transistors	8 transistors	2 memristors	17 transistors

IV.4. SUMMARY

Fuzzy controllers find a wide range of applications in different systems. Hardware-based Fuzzy controllers are often preferred when high speed, low area occupancy and/or low power is required. A major building block in Fuzzy controllers is the defuzzifier. COG defuzzifiers occupy large area due to the Multiplication/Division circuits which dictated its implementation off chip. Several designs have been proposed to address this issue. However, the proposed circuits either still occupy relatively large area or compromise the robustness of the design due to the reliance on aggressive analog processing.

In this work, a memristor-based COG defuzzifier is proposed. The design occupies small area due to the use of memristors that scale down beyond 10nm.The main idea is to store the singletons (output fuzzy sets) in the memductance. Accordingly, the multiplication operation is reduced to Ohm's Law which is an inherent characteristic in electrical circuits. This avoids using complicated hardware to perform the multiplication process which is either area consuming or requires significant analog processing. The proposed design avoids the use of analog processing to provide descent robustness for the circuit.

A two-input COG defuzzfier was implemented as a case study. Simulations were conducted using the Eldo simulator from Mentor Graphics. The results deviate from the theoretical prediction by less than 4%.

V. CONCLUSIONS

The recent characterization of the memristor element via the TiO2 process has instigated significant research trying to leverage some of the characteristics of memristors in advancing currently existing technology, as well as, introducing new computing paradigms that have not been possible before.

This thesis focused primarily on hardware-based Fuzzy systems that generally go under beyond Von Neumann computing architectures. Although Fuzzy systems have for so long been implemented on the software level, their hardware implementation is generally more efficient in terms of power consumption and area occupancy. This notion has encouraged researchers over the past decades to propose several transistor-based architectures that, in essence, rely on significant analog processing trying to emulate the actual physical dynamics of the designated systems based on the system equations.

In this work, the author sought to explore the use of memristors in the design of two fundamental building blocks in Fuzzy systems which are: the Inference Engine and the COG Defuzzifier.

Fuzzy inference engines are mainly based on Min-Max circuits. Several transistor-based designs have been proposed in the literature. Recently, memristor-based Min-Max circuits were proposed and were proven to outperform their transistor based counterparts. Yet, their treatment was relatively basic. Only 2-input circuits were addressed. Also, the effect of the memristor threshold on Min-Max circuits was only highlighted without thorough analysis. This work extrapolated the work in memristor-based Min-Max circuits for an arbitrary number ‘N’ of memristors. It was proven that the same structure proposed for 2-input structures can be extended for N-input structures under the proper design constraints. Also, the effect of the memristor threshold is modeled and a closed form expression is derived. It is shown that for a specific memristor with a given OFF resistance and threshold current, there exists a trade-off between the number of inputs and the resolution of the circuit.

Unlike Fuzzy inference engines, only transistor-based architectures were reported in the literature for COG defuzzifiers. The major challenge in the design of COG defuzzifiers is their area occupancy due to the Multiplier/Divider circuits that usually consume significant hardware. In this work, the analog programmability of

memristors is leveraged in the design of multiplier circuits in which the multiplication operation is reduced to simple Ohm's law which is inherent in any circuit and, subsequently, spared the need for multiplier circuit and, ultimately, yields a more compact design.

REFERENCES

[1] L. O. Chua, "Memristor-the missing circuit element," *IEEE Transactions on Circuit Theory*, Vol. Ct-18, no. 5, pp. 507-519, September 1971.

[2] S. E. Thompson and S. Parathasarathy, "Moore's law: the future of Si Microelectronics," *Materials Today*, Vol. 9, No. 6, pp. 20-25, June 2006.

[3] D.B. Sturkov, G.S. Snider, D.R. Stewart and R.S. Williams, "The missing memristor found," *Nature*, Vol.453, no. 7191, pp. 80-83, May 2008.

[4] L. O. Chua and S. M. Kang, "Memristive devices and systems," *Proceedings of the IEEE*, Vol. 64, no. 2, pp. 209-223, February 1976.

[5] M. D. Pickett, D. B. Sturkov, J. L. Borghetti, J. J. Yang, G. S. Snider, D. R. Stewart and R. S. Williams, "Switching dynamics in titanium dioxide memristive devices," *Journal of Applied Physics*, Vol. 106, no. 7, pp. 1-6, October 2009.

[6] G. S. Rose, J. Rajendran, H. Manem, R. Karri and R. E. Pino, "Leveraging Memristive Systems in the Construction of Digital Logic Circuits," in *Proceedings of the IEEE*, vol. 100, no. 6, pp. 2033-2049, June 2012.

[7] R. Shayanfar, A. Khoei and K. Hadidi, " Multi-input Voltage-mode min-max circuit," Iranian Conference on Electrical Engineering (ICEE), Isfahan, Iran, pp. 413-416, May 2010.

[8] R. Kozma, R. Pino and G. Pazienza, "Memristor Minds," in *Advances in Neuromorphic Memristor Science and Applications*, Springer, 2012

[9] S. Kvatinsky, G. Satat, N. Wald, E. G. Friedman, A. Kolodny, U. C. Weiser, "Memristor-based Implication (IMPLY) Logic: Design principles and Methodologies," *IEEE Transaction on Very Large Scale Integration (VLSI) systems,* Vol. 22, no. 10, pp. 2065-2066, October 2014.

[10] S. Kvatinsky, N. Wald, G. Satat, A. Kolodny,U. C. Weiser and E. G. Friedman, "MRL- Memristor Ratioed Logic," International Workshop on Cellular Nanoscale Networks and Their Applications (CNNA), Turin, Italy, pp. 1-6, August 2012.

[11] L. Chua, "Everything You Wish to Know About Memristors But are Afraid to Ask," *Radioengineering*, Vol. 24, no. 2, June 2015.

[12] S. Kvatinsky, E.G. Friedman, A. Kolodny and U.C. Weiser, "TEAM:ThrEshold Adaptive Memristor Model," *IEEE Transaction on circuits and Systems I: Regular Papers*, Vol. 60, no. 1, pp. 211-221, January 2013.

[13] T. Prodromakis, B. P. Peh, C. Papavassiliou and C. Toumazou, "A Versatile Memristor Model With Nonlinear Dopant Kinetics," *IEEE Transactions on Electron Devices,* vol. 58, no. 9, pp. 3099-3105, September 2011.

[14] C. Yakopcic, T. M. Taha, G. Subramanyam, R. E. Pino and S. Rogers, "A Memristor Device Model," in *IEEE Electron Device Letters*, vol. 32, no. 10, pp. 1436-1438, Oct. 2011.

[15] Z. Biolek, D. Biolek and V. Biolkova, “Spice Model of Memristor with Nonlinear Dopant Drift,” *Radioengineering,* Vol. 18, no. 2, pp. 210-214, June 2009.

[16] Y. N. Joglekar and S. J. Wolf, “The elusive memristor Properties of basic electrical circuits,” European Journal of Physics, Vol. 30, no. 4, pp. 661-675, July 2009.

[17] S. Benderli and T. A. Wey, “On SPICE macromodelling of TiO2 memristors,” *Electronic Letters*, vol. 45, no. 7, pp. 377–379, March 2009.

[18] Y. V. Pershin, and M. Di Ventra, “Practical approach to programmable analog circuits with memristors,” *IEEE Transactions on Circuits and Systems I: Regular Papers*, Vol. 57, no. 8, pp. 1857-1864, August 2010.

[19] M. Elshamy, H. Mostafa and M. S. Said, “A novel nondestructive readout circuit for Memristor-based memory arrays,” Canadian Conference on Electrical and Computer Engineering (CCECE), Toronto, Canada, pp. 1-5, May 2014.

[20] D. Niu, Y. Chen, C. Xu and Y. Xie, “Impact of process variations on emerging memristor,” ACM/IEEE Design Automation Conference (DAC), pp. 877-882, June 2010.

[21] S. Kannan, N. Karimi, R. Karri and O. Sinanoglu, "Modeling, Detection, and Diagnosis of Faults in Multilevel Memristor Memories," in *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 34, no. 5, pp. 822-834, May 2015.

[22] A. Thomas, “Memristor-based neural networks,” *Journal of Physics D: Applied Physics*, Vol. 46, no. 9, February 2013.

[23] F. Merrikh-Bayat, S. Shouraki and F. Merrikh-Bayat, “Memristor crossbar-based hardware implementation of fuzzy membership functions,” International Conference on Fuzzy systems and Knowledge Discovery (FSKD), Shanghai, China, pp. 645-649, July 2011.

[24] F. Merrikh-Bayat and S. Shouraki “Memristive Neuro-Fuzzy System,” *IEEE Transactions on Cybernetics*, Vol. 43, no. 1, pp. 269-285, February 2013.

[25] G. Yosefi, S. Mirzakouchaki and S. Neda, “Design of new CMOS current mode min and max circuits for FLC chip applications,” European conference on Circuit Theory and Design (ECCTD), Anatalya, Turkey, pp. 89-92, August 2009.

[26] M. Padash, A. Khoei, K. Hadidi and H. Ghazemiyan, “A high precision high frequency VLSI multi-input min-max circuit based on WTA-LTA cells,” International Conference on Electronic Devices, Systems and Applications (ICEDSA), Kuala Lampur, Malaysia, pp. 152-156, April 2011.

[27] R. Khayatzadeh, G. Yazadani, A. Khoie and K. Hadidi, “A new structure for current mode min-max circuit using CMOS technology for fuzzy applications,” Iranian Conference on Electrical Engineering (ICEE), Tehran, Iran, pp. 346-350, May 2014.

[28] R. Khayatzadeh, S. Pourhasan and M. Mosazadeh, “A new current mode min-max circuit using CMOS technology for fuzzy applications,” Iranian Conference on Electrical Engineering (ICEE), Tehran, Iran, pp. 443-447, May 2014.

[29] M. Klimo and O. Such, “Memristors can implement Fuzzy Logic”, arXiv:1110.2074v1, [cs.ET], 2011.

[30] E. Farshidi, “A low-power current-mode defuzzifier for fuzzy logic controllers,” International Conference on Signals, Circuits and Systems, Hammamet, Tunisia, pp. 1-4, November 2008.

[31] M. Fratrik, M. Klimo and P. Jancovic. “Determinig the switching properties of the minimum function from a single VI characteristic,” Elektro, Rajecke Teplice, pp. 645-649, May 2014.

[32] D. Biolek, V. Biolkova and Z. Kolka, “Memristive systems for analog signal processing,” International Symposium on Circuits and Systems (ISCAS), Melbourne, Australia, pp. 2588-2591, June 2014.

[33] Available at: http://webee.technion.ac.il/people/skva/Memristor%20Models/Manual.pdf.

[34] M. Togai and H. Watanabe, “An inference engine for real-time approximate reasoning: Toward an expert on a chip, ” *IEEE Expert*, vol. 1, no. 3, pp. 55-62, August 1986.

[35] M. Ghasemzadeh, A. Akbari, A. Khoei and K. Hadidi, "A novel mixed-signal digital voltage mode CMOS fuzzy logic controller in 0.18µm technology," International Conference on Mixed Design of Integrated Circuits & Systems (MIXDES), pp.148-153, June 2014.

[36] Ch.-Y. Chen, Ch.-Y. Huang and B.-D. Liu, “A current-mode defuzzifier circuit to realize the centroid strategy,” IEEE International Symposium on Circuits and Systems, pp. 609-612, June 1997.

[37] P. Hoseini, A. Khoei, K. Hadidi, "Circuit design of voltage mode center of gravity defuzzifier in CMOS process," International Conference on Electronic Devices, Systems and Applications (ICEDSA), pp.169-173, April 2010.

[38] H. Watanabe, W. D. Dettloff and K. E. Yount, “A VLSI fuzzy logic controller with reconfigurable, cascadable architecture,” *Journal of Solid state Circuits*, vol. 25, no. 2, pp. 376-382, April 1990.

[39] A. Niknam, A. Khoei, K. Hadidi, M. Padash, “A nano-watt current-mode defuzzifier for fuzzy logic controllers,” International Conference on Electronic Devices, Systems and Applications (ICEDSA), pp.157-160, April 2011.

[40] A. Taji, A. Khoei, K. Hadidi, M. Padash, "Low-voltage and low-power consumption 0.35um CMOS voltage-mode defuzzifier," International Conference on Electronic Devices, Systems and Applications (ICEDSA), pp.161-164, April 2011.

[41] G. Ligang, F. Alibart and D.B. Strukov, “Programmable CMOS/Memristor Threshold Logic,” *IEEE Transactions on Nanotechnology*, vol.12, no.2, pp.115-119, March 2013.

[42] B. Razavi, “*Design of Analog CMOS Integrated Circuits,*” McGraw_Hill, 2001.

[43] M. Ismail, “*Analog VLSI: Signal and Information Processing*,” McGraw_Hill, 1994.

[44] J. Madrenas, D. Fernandez and J. Cosp, “A low-voltage current sorting circuit based on 4-T min-max CMOS switch,” International Conference on Electronics, Circuits, and Systems (ICECS), pp. 351-354, December 2010.