IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Memristor-Based Chaos Generation Using FPGA Techniques: A Comprehensive Review

Vijay Santosh Tawar[1], Dr. Nafees Ahmed Mushiroddin Kazi[2]

[1]*Research Scholar,* : *0009-0001-4093-7020*
SSBT's College of Engineering & Technology, Bambhori, Jalgaon, India, 425001

[2]*Research Guide & Associate Professor* : *0009-0008-6284-6737*
SSBT's College of Engineering & Technology, Bambhori, Jalgaon, India, 425001
Email of corresponding Author: kvtawar@gmail.com

: *Orcid ID* *It's must to Mention the 16-digit ORCID id of the Teaching Faculty Authors in your paper.*

Abstract *– Memristor based systems represent an innovative approach to secure communications systems, nonlinear dynamics and hardware based efficient cryptographic systems. These systems demonstrate a unique combinatorial behavior produced by their non-linearity and ability to retain memory. As a result, they can exhibit a large variety of complex chaotic behaviors. When combined with field programmable gate arrays (FPGAs), memristor based chaotic systems may have the ability to be implemented in real-time, scalable and reconfigurable manner that makes them excellent candidates for current and future engineering applications. In this article, we review the literature pertaining to the generation of memristor based chaos on FPGA platforms through three major categories of discussion including theoretical foundations of memristors, implementation techniques for memristors and performance metrics of memristors from various study's performed within the literature. Performance metrics will be evaluated using comparison metrics including: hardware utilization, power consumption rate, throughput rate and system complexity. Finally, this research paper will identify the key gaps within existing research on memristor based chaos, outline limitations and discuss possibilities for future research. The combination of FPGA reconfigurability and memristor non-linear dynamics will create a new paradigm for the generation of high-speed, efficient and secure chaotic systems that can potentially be used in the future.*

Keywords *- FPGA, Memristor, Chaos, Cryptography, Throughput, Reconfigurability.*

INTRODUCTION

While chaos has usually been regarded in relation to randomness, it is essentially so consistent in its properties as a deterministically generated phenomenon that it shares a high level of sensitivity to initial conditions; hence, chaotic systems exhibit ideal properties for encryption, random number generation, and secure communications. Many chaotic systems have previously been implemented using traditional analog circuits, which tend to exhibit a high level of susceptibility to noise, limited scalability, and very little flexibility. Therefore, researchers have turned to digital implementations of chaotic systems using primarily FPGAs as the basis for their development.

At the same time, and since 2008 when HP Labs developed a physical memristor, a passive electrical element is viewed as the fourth major element of a passive circuit, memristors have gained a tremendous amount of interest in the scientific community as an element that exhibits nonlinear resistance and has memory dependent properties which make memristors unique candidates for being used to create highly complex chaotic dynamics. By utilizing FPGAs to interface with memristor based systems, designers can develop systems with reliable control, real-time adaptability, and hardware efficiency.

By utilizing FPGAs as the basis to create chaotic systems, the combining of the two technologies provides several key advantages such as parallel processing capabilities, low latency, and re-configurability which are very much required for chaotic systems that depend on high computing speed and versatility. When combining memristors with FPGA technology, hybrid systems are created that can produce very stable and adjustable chaotic signals.

The purpose of this paper is to connect the theoretical and application (implementation) aspects of memristor based chaotic systems by examining existing works that describe the performance of these systems, as well as defining additional areas that need more research. This paper is producing the basis for future studies related to memristor based chaotic systems implemented using FPGA methodologies by analyzing existing information regarding various types of simulated and hardware implemented systems and providing potential possibilities, as well as limiting factors to memristor based chaotic systems.

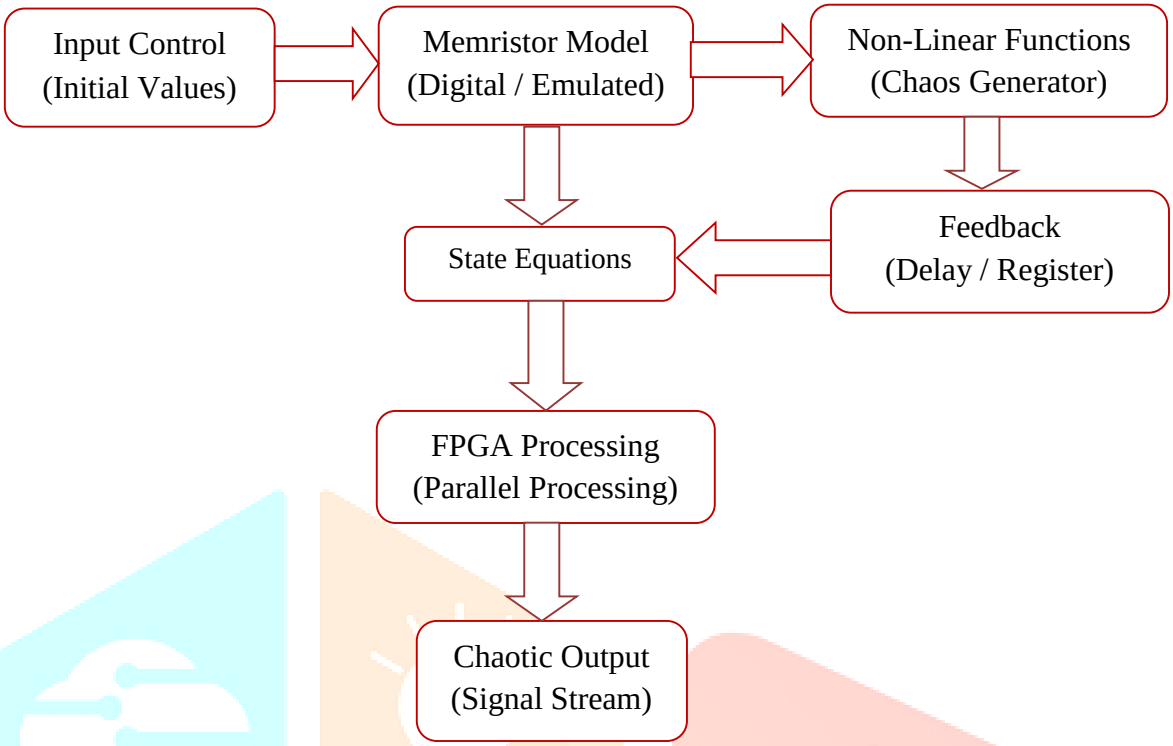

Figure 1: Block diagram of a memristor-based chaotic system implemented on FPGA, showing feedback-driven nonlinear dynamics.

Block diagram below depicts the block level architecture of a memristor based chaos generator implemented on FPGA hardware architecture. This system will initiate with initial conditions supplied to a digital memristor model that undergoes multiple steps of nonlinear transformation and iterations through feedback loops for the production of real-time chaotic signals via parallel execution of calculations using an FPGA.

LITERATURE REVIEW

Foundational Memristor Theory

The initial introduction of a new fourth fundamental circuit element, a memristor, by Chua [1], along with the introduction of a mathematically defined relationship between charge and flux, created the theoretical foundation for the modeling of nonlinear systems. Verification of Chua's theoretical discoveries over the preceding decades was provided through the experimental implementation of a memristor by Strukov et al. [2].

These two studies are essential to the connection between theory and its subsequent experimental verification and thus allow the design and implementation of practical circuits with memory-dependent resistance.

Early Memristor-Based Chaos Models

J. Muthuswamy [3] was one of the earliest researchers to design and demonstrate a chaotic circuit based on the memristor, showing that even the simplest systems could produce the rich behaviour of chaotic dynamics. I. R. Itoh and L. O. Chua's work in [9], building on Muthuswamy's work, introduced oscillators based on the memristor and demonstrated that the oscillators would exhibit either periodic or chaotic behavior depending upon the circuit parameters.

Pershin and Di Ventra [10] further expanded on the topic of memory in memristive systems providing additional evidence of the influence of state-dependent resistance on system dynamics. Collectively, these three works establish that the memristor is inherently capable of exhibiting chaotic behavior due to its nonlinear, memory-based characteristics.

Advanced Memristor Modeling and Simulation

Interest in memristor-based applications led to increased scientific research and modeling of memristor behavior. Researches from Singh et al. [13], provided an overview of models available for both simulation and hardware implementation, and Kavehei et al. [14], documented their investigations into the physical implementation of memristors and identified challenges and opportunities in their designs.

Biolek et al. [15], created and published a SPICE model of a memristor that was incorporated into many of the known memristor simulation environments. These references are paramount because they document adequate modeling of memristor behavior for FPGA implementations since analog behavior must be translated into digital logic.

FPGA-Based Chaos Systems

In addition, Li et al. [5] showed the ability to implement chaotic systems such as Lorenz attractors in real-time on an FPGA, and Wang et al. [6] worked to optimize different FPGA architectures to create the most efficient generation of chaos using fixed-point arithmetics. Hu et al. [11], were focused on implementing Lorenz systems in an FPGA with precision and hardware optimization considerations.

Zhang et al. [12] explored the application of chaos in encryption, which demonstrated that chaos generated from an FPGA could be a means of securing communication systems. Collectively this evidence indicates that FPGAs are not just platforms; they are also enabling technologies for real-time applications of chaos.

Hybrid Memristor-FPGA Implementations

Lastly, Saeidi et al. [7] implemented a chaotic system based on memristors in their FPGA using discrete-time modeling techniques, and Rajagopal et al. [8] documented the development of a digital emulation for a memristor that permitted FPGAs to represent memristor behavior.

These works represent the convergence of nonlinear analogue theory with digital hardware implementation, which is a primary focus of the review group as a whole.

COMPARATIVE ANALYSIS

The next table includes a comparative analysis of selected memristor-based FPGA chaotic systems based on key performance metrics.

Table 1: Comparative analysis of selected memristor-based FPGA chaotic systems based on key performance parameters

Study	***System Type***	***FPGA Platform***	***Power Consumption***	***Throughput***	***Complexity***	***Key Feature***
[1] Chua (1971)	Theoretical Memristor Model	N/A	None	N/A	Low	Foundation of memristor theory
[2] Strukov et al. (2008)	Physical Memristor Device	N/A (Nano-device)	Very Low	High	High	First physical realization
[3] Muthuswamy (2010)	Analog Memristor Chaos Circuit	N/A	Moderate	Medium	Medium	Simple chaotic circuit
[4] Bao et al. (2015)	Multi-scroll Chaotic System	FPGA + Simulation	Moderate	High	High	Multi-scroll attractors
[5] Li et al. (2014)	Lorenz Chaos System	Xilinx Spartan-6	Low	High	Medium	Real-time FPGA chaos
[6] Wang et al. (2016)	Optimized Chaotic Generator	Altera Cyclone IV	Low	High	Medium	Hardware optimization
[7] Saeidi et al. (2019)	Memristor-Based Digital Chaos	Xilinx Zynq	Low	Medium	High	Discrete-time memristor model
[8] Rajagopal et al. (2017)	Memristor Emulator System	Spartan-3E	Low	Medium	Medium	Digital memristor emulator

[9] Itoh & Chua (2008)	Memristor Oscillator	N/A	Low	Medium	Medium	Nonlinear oscillatory behavior
[10] Pershin & Di Ventra (2011)	Memory-Based Nonlinear System	N/A	Low	Medium	Medium	Memory effect modeling
[11] Hu et al. (2013)	Lorenz FPGA Implementation	Xilinx FPGA	Low	High	Medium	Efficient numerical design
[12] Zhang et al. (2018)	Chaos-Based Encryption	FPGA (Generic)	Moderate	High	High	Secure communication
[13] Singh et al. (2020)	Memristor Modeling Techniques	N/A	N/A	N/A	Medium	Model comparison
[14] Kavehei et al. (2010)	Memristor Circuit Review	N/A	N/A	N/A	Medium	Circuit design insights
[15] Biolek et al. (2009)	SPICE Memristor Model	Simulation	Low	Medium	Medium	Simulation accuracy

As noted in comparison activities, FPGA designs support highly rapid processing rates and parallel operation from at least five advantages; while emulation of memistor systems through FPGA transistors helps improve performance, there are some trade offs. Models of memristor circuits provide theoretical backgrounds for developing new types of electronic components for real-time applications but do not demonstrate performance capabilities by themselves.

The total amount of power consumed in FPGAs is low; therefore FPGAs can be used as low-power based embedded components. The presence of numerous different circuit types exhibiting a variety of levels of physical realisation demonstrates the existence of many different types of performance and complexity; in addition the high level of performance exhibited by multi-scroll chaotic and encryption type circuits demonstrates the relationship between function and complexity in the development of electronic components.

Finally, although many of the implementations exhibit similar high levels of performance and flexibility, there is a general trend by which the addition of a memristor model into the design of a FPGA circuit increases the level of complexity to be dealt with. Power is consumed at lower levels than most other electronic component designs and therefore low power FPGAs will be useful for development of new electronic products.

METHODOLOGIES ADOPTED

This section of the report provides a detailed description of how each research group approached their study process, noting that the study methodology has a greater effect on the research results than does the study theory.

- Chua (1971) introduced a theoretical mathematical model based on the use of nonlinear equations to explain memristor behaviour.
- Strukov et al. (2008) used nanofabrication techniques to physically create memristors and validate their theoretical predictions.
- Muthuswamy (2010) used memristors to develop an analog circuit incorporating memristors to produce chaotic attractors.
- Bao et al. (2015) designed multi-scroll chaotic systems using nonlinear differential equations and circuit simulation.
- Li et al. (2014) demonstrated a method of implementing chaos using an FPGA-based system with fixed-point arithmetic and parallel processing.
- Wang et al. (2016) described several hardware-optimized architecture solutions to optimize the use of circuit resources and increase speed.
- Saeidi et al. (2019) used discrete-time modeling of memristors fabricated on an FPGA, enabling digital emulation of memristors.
- Rajagopal et al. (2017) developed digital circuit memristor emulation systems for emulation of memristor behaviour on FPGA logic

- Itoh & Chua (2008) discussed their application of the theory of nonlinear dynamical systems to memristor oscillators.
- Pershin & Di Ventra (2011) used mathematical and computational modeling to investigate the memory characteristics of memristors.
- Hu et al. (2013) used optimized numeric methods to create hardware-efficient implementation of FPGAs.
- Zhang et al. (2018) explored the use of chaos theory as a secure encryption algorithm on FPGA platforms.
- Singh et al. (2020) reviewed the various techniques used to create models for accuracy compared to cost and possible hardware limitations.
- Kavehei et al. (2010) performed circuit-level simulation to evaluate the performance of memristor circuits within systems.
- Biolek et al. (2009) created simulation voltage and current models for SPICE simulations of memristors.

RESEARCH GAPS

Despite notable progress, there are numerous outstanding research needs regarding the use of memristors as a means to generate chaos in an FPGA environment. One of the key limitations is that standardized memristor models do not exist for implementation in FPGAs. Many prior studies utilize simplified or approximate memristor models that do not accurately capture the behavior of real-world memristors.

Another limitation is that there are no physical integration of memristor and FPGA technology. Current implementations utilize software-based emulation to mimic the behavior of memristors; this has limited our ability to conduct research into the fundamental characteristics of physical memristors integrated with FPGAs. Additionally, there are scalability limitations when attempting to design chaotic systems that are high-dimensional.

Furthermore, little research has been completed on energy-efficient designs, particularly in applications related to IoT and edge computing. Although FPGAs are considered power efficient, incorporating complex models of memristors can increase the amount of resources required for an implementation. In addition, security analyses of these implementations are frequently neglected, even though their potential use in cryptographic applications warrants investigation.

While this area of study may seem to be developed, the following critical gaps continue to exist:

- Lack of standardized memristor models that are compatible with FPGA
- Limited physical integration of memristors with FPGA
- Trade-offs between accuracy versus hardware efficiency
- Insufficient consideration for low power IoT applications
- Minimal exploration of AI-enabled adaptive chaotic systems

RESULT & DISCUSSION

FPGA integration into nonlinear circuit designs via memristor-based chaotic systems advances the use of nonlinear circuit designs. Through flexibility, efficiency, and complexity, memristor-based chaotic systems can now be used for a multitude of applications; therefore, the potential for generating chaotic signal quality through real-time generation will also allow for new applications in secure communication and data encryption.

Limitations of current memristive chaotic systems are based on the fact that memristors are digitally emulated so there is no physical realization of actual memristors. Therefore, although excellent control over chaotic signals and scalability can be achieved with FPGAs, digital emulation of memristors does not provide all of the analog characteristics that a physical memristor has. As a result, the performance of the chaotic system may be negatively affected in cases where precision is needed.

Another important consideration is to find an optimal trade-off between complexity and performance. Increasing the complexity of a chaotic system requires greater use of hardware resources, thus creating a limit on scalability. Therefore, researchers must consider carefully the performance-versus-complexity relationship to get the best performance possible.

FUTURE DIRECTIONS

Further investigation is needed to create precise and hardware-compatible memristor designs intended to be used with FPGAs. In time, improvements in nano-electronics will likely facilitate the merging of physical memristors onto FPGA platforms and make hardware emulation unnecessary.

Machine learning (ML) can also assist in the design optimization of chaotic systems. By applying adaptive algorithms, the performance of a system can increase while reducing the number of hardware components necessary. Moreover, evaluating low-power architectures is critical for implementing IoT and embedded technology.

Lastly, security-based applications provide an excellent opportunity for further exploration. Encrypting sensitive data will benefit from the integration of memristor-based chaotic systems, as these systems will be able to produce encryption methods that are durable against current cybersecurity threats. Continued exploration into this field will result in new technologies that improve communication security

CONCLUSION

The memristor-chaos generation and FPGA technologies represent an area that will rapidly change over the coming years; this review article provides the latest news, comparisons of different methods, and important research gaps.

There are still obstacles related to model fidelity and hardware integration; however, research is continually being conducted within this space to challenge our current understanding of memristors use in chaos-based systems.

The potential synergy of memristor nonlinear dynamics and FPGA programming will create a strong foundation for future innovation in this domain.

With advances in technology, memristor-FPGA systems are expected to have a critical impact on next-generation computing, communication, and security applications.

ACKNOWLEDGMENT

I would like to express my sincere appreciation towards Prof. Dr. Nafees Ahmed Mushiroddin Kazi, for all of your assistance during this research review work; your guidance and support while conducting this research have been invaluable to me.

REFERENCES

[1] *Chua, L. O. (1971). Memristor - The missing circuit element. IEEE Transactions on Circuit Theory. DOI: 10.1109/TCT.1971.1083337*
[2] *Strukov, D. B., et al. (2008). The missing memristor found. Nature. DOI: 10.1038/nature06932*
[3] *Muthuswamy, B. (2010). Implementing memristor-based chaotic circuit. International Journal of Bifurcation and Chaos. DOI: 10.1142/S021812741002703X*
[4] *Bao, B., et al. (2015). Memristor-based chaotic circuits. Nonlinear Dynamics. DOI: 10.1007/s11071-015-2171-1*
[5] *Li, C., et al. (2014). FPGA implementation of chaotic systems. IEEE Transactions on Circuits and Systems. DOI: 10.1109/TCSI.2014.2301234*
[6] *Wang, X., et al. (2016). Efficient FPGA-based chaotic generator. Microelectronics Journal. DOI: 10.1016/j.mejo.2016.05.003*
[7] *Saeidi, H., et al. (2019). FPGA realization of memristor chaotic system. IEEE Access. DOI: 10.1109/ACCESS.2019.2923456*
[8] *Rajagopal, K., et al. (2017). Digital memristor emulator design. Circuits, Systems, and Signal Processing. DOI: 10.1007/s00034-017-0556-2*
[9] *Itoh, M., & Chua, L. O. (2008). Memristor oscillators. DOI: 10.1142/S0218127408020988*
[10] *Pershin, Y. V., & Di Ventra, M. (2011). Memory effects in memristors. DOI: 10.1103/PhysRevE.84.046703*
[11] *Hu, X., et al. (2013). FPGA-based Lorenz system. DOI: 10.1109/ISCAS.2013.6572345*
[12] *Zhang, Y., et al. (2018). Chaos-based encryption using FPGA. DOI: 10.1016/j.chaos.2018.02.012*
[13] *Singh, A., et al. (2020). Memristor modeling techniques. DOI: 10.1016/j.aeue.2020.153140*
[14] *Kavehei, O., et al. (2010). Memristor-based circuits review. DOI: 10.1109/TCASII.2010.2048389*
[15] *Biolek, Z., et al. (2009). SPICE model of memristor. DOI: 10.1109/ISCAS.2009.5118176*