Discover Internet of Things

Discover

Research

Reliable and secure memristor-based chaotic communication against eavesdroppers and untrusted foundries

Rahul Vishwakarma[1] · Ravi Monani[2] · Ava Hedayatipour[2] · Amin Rezaei[1]

Received: 1 December 2022 / Accepted: 6 March 2023
Published online: 20 March 2023

Abstract
Chaos is a deterministic phenomenon that occurs in a non-linear dynamic system under specific condition when the trajectories of the state vector become periodic and extremely sensitive to the initial conditions. While traditional resistor-based chaotic communications are primarily concerned with the safe transfer of information across networks, the transceivers themselves can be compromised due to outsource manufacturing. With the growth of wireless sensors in resource-constrained implantable and wearable devices, chaotic communication may be a good fit if the information transmitted is reliable and the transmitter devices are secure. We believe that memristor, as the fourth fundamental two-terminal circuit element, can close the gap between reliable communication and secure manufacturing since its resistance can be programmed and saved by the designer and not the foundry. Thus, in this paper, we propose a memristor-based Chua's chaotic transceiver that is both reliable in the presence of eavesdroppers and secure against untrusted foundries. Specifically, we consider the pair of transmitter and receiver under the same memristor value to show the possibility of uninterrupted communication as well as cases where different values of memristors are used to find out the possible range in which the message can still be meaningfully decoded. Experimental results confirm that both reliable communication and secure design can be achieved via our proposed memristor-based chaos transceivers.

Keywords Chaos · Memristor · Security · Reliability · Wearable devices · Chua's circuit

1 Introduction

The number of wearable devices have increased exponentially in the recent times and is expected to exceed more than $67 Billion by 2024 [1]. Much like embedded devices, they are loaded with sensors for collecting data from the surrounding ecosystem and transmitting the data to the destination for desired analysis and actions. With the advancement in healthcare wearable devices [2] such as implantable pacemakers, biofluidic-based wearables, and skin-based wearables, there has been growing concern over their manufacturing and their implications for security and privacy [3]. Consider a pacemaker as an example. If an eavesdropper gets access to the data, it creates a privacy issue for the user [4], but if the receiver's circuit is duplicated, the attacker can decode the data and pose a life-threatening threat by sending malicious

Rahul Vishwakarma, Ravi Monani, Ava Hedayatipourand Amin Rezaei contributed equally to this work

✉ Amin Rezaei, amin.rezaei@csulb.edu; Rahul Vishwakarma, rahuldeo.vishwakarma01@student.csulb.edu; Ravi Monani, ravi.monani01@student.csulb.edu; Ava Hedayatipour, ava.hedayatipour@csulb.edu | [1]Department of Computer Engineering & Computer Science, California State University Long Beach, Long Beach, CA, USA. [2]Department of Electrical Engineering, California State University Long Beach, Long Beach, CA, USA.

actions to the pacemaker. Therefore, not only is a solution needed for reliable communication resistant to eavesdropping, but the risk-sensitive issues of device duplication must be addressed.

Because chaos-based communication [5, 6] is highly sensitive to initial conditions and synchronize over time, it is a suitable choice for reliable communication [7] against eavesdroppers. Moreover, use of chaos is suggested as a potential alternative in post-quantum cryptography [8]. Although there are improvements in chaotic communication, for example, the implementation of hyperchaotic circuits [9], not much work has been done to secure the design and manufacturing of chaotic transceivers at an untrusted foundry [10]. Due to the growing trend in outsource manufacturing, the electronic industry must deal with various hardware threats that an untrusted foundry can pose. Since the third-party foundry has access to the circuit design, it can do a lot of unintended activities, which not only have a negative impact on the designers' revenues but, more importantly, can be risk-sensitive to end users, especially in the case of healthcare wearable and implantable devices. One approach to addressing untrusted foundries is to implement logic locking (a.k.a. logic encryption) on the circuits. Logic locking [11] is a mechanism to enable the desired behavior of the circuit only when the correct key is applied to the circuit. Recently, Register-Transfer Level (RTL) locking is proposed to protect sensitive Intellectual Property (IP) semantics against untrusted entities [12]. However, this approach is not suitable for mixed-signal circuits like transceivers. Moreover, different chaos-based implementations of basic logic gates to obfuscate power profiles and mitigate power analysis-based side channel attacks are proposed [13, 14], and the concept of asymmetry in chaotic Boolean gates is exploited to lock the circuit [15]; but the main goal of these initiatives is to use the unique features of chaos to reach hardware security goals in digital circuits rather than proposing secure and reliable mixed-signal chaotic transceivers. In addition, an approach to securing mixed-signal circuits via logic locking is proposed [16], which relies on logic locking of the digital portion of the mixed-signal IC such that unless the correct digital key is provided, the mixed-signal performance will be pushed outside of the acceptable specification range. However, directly locking the analog portion of the mixed-signal ICs via an analog key has not been investigated.

Thus, in this paper, we use a low-overhead yet effective approach to enable logic locking using memristors [17] in chaotic circuits. We believe that memristor, as the fourth fundamental two-terminal circuit element, can close the gap between reliable communication and secure manufacturing since its resistance can be programmed by the designer and not the foundry.

Figure 1 shows our proposed solution to address the issue of untrusted foundries and eavesdroppers together. On a logic-locked Chua's chaotic circuit, an input signal is encrypted and sent over a public channel. The encrypted signal is decoded at the receiver, which is also logic-locked, and only with the correct key values for the memristor can the message be successfully decoded. Because chaotic maps are highly sensitive and the key space is exponentially large, logic locking based on memristors makes it safe against an attacker attempting to duplicate the receiver without knowing the correct key value. The key contributions of this work are threefold:

- Establishing reliable communication against eavesdroppers using memristor-based Chua's chaotic system;
- Proposing a novel logic locking mechanism based on the use of memristors as the keys for securing the circuit design against untrusted foundries;
- Practical realization of the proposed system on LTSpice and NI Multisim; and validating the system of equations in MATLAB Simulink.

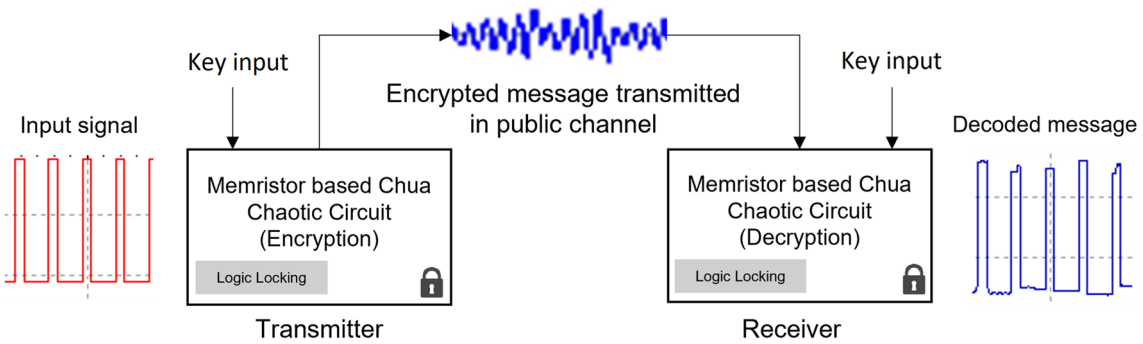


Fig. 1 Logic-locked memristor-based Chua's chaotic system

The remainder of this paper is organized as follows. First, the attack models are discussed, and the memristor model as well as Chua's chaotic circuit are designed in the LTSpice Simulator in Sect. 2. The implementation of memristor-based Chua's chaotic circuit and mathematical simulation of the circuit using MATLAB Simulink is shown in Sect. 3. Further, in Sect. 4, a secure transceiver is designed using a logic-locked memristor-based chaotic circuit, and its security is analyzed. Section 5 shows the experimental results, and the paper concludes with Sect. 6.

2 Preliminaries

2.1 Attack model

In this paper, we consider two attack models:

- Eavesdropper: The attacker passively listens to transceivers' communications to gain access to transmitted information. It is a privacy threat, and in our case, the goal of the eavesdropper is to find out private information about the user who uses implementable and wearable devices.
- Untrusted foundry: The attacker resides in the foundry and can duplicate the transceivers. It is a security threat, and in our case, the goal of the untrusted foundry is either to make a financial profit by selling unauthorized devices to the gray market or to synchronize with authorized implementable and wearable devices to send malicious actions.

Please note that we consider that testing will be done in-house, and thus the test/verification engineer is trusted.

2.2 Memristor model

A memristor is a two-terminal electrical component relating electric charge and magnetic flux linkage. It can be viewed as a form of non-volatile memory that is based on resistance switching, which increases the flow of current in one direction and decreases the flow of current in the opposite direction. There are primarily three types of memristor model: linear, non-linear, and threshold adaptive. To add non-linearity at the boundaries, window functions are used [18, 19]. We use HP Memristor Model with non-linear dopant drift for transient analysis as it offers results which are in good agreement with a part of the hitherto published experiments. A detailed implementation of the model can be found in [20]. The total resistance of the memristor R_{MEM} is given by the below equations.

$$R_{MEM}(x) = R_{ON}(x) + R_{OFF}(1 - x), \tag{1}$$

$$x = \frac{w}{D} \in (0, 1) \tag{2}$$

Figure 2 shows the component model of the memristor on LTSpice XVII(x64) (17.0.34.0).

2.3 Chua's chaotic circuit

Chaos can be defined as the unpredictability of a deterministic system that is highly dependent on its initial conditions. In [6] different modes of chaotic equations such as Lorenz [21], Rössler [22], Chua [23] and Lü [24] have been compared for various performance metrics.

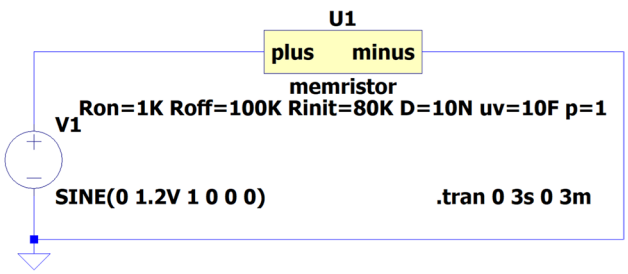


Fig. 2 Memristor component model

Chua's circuit generates a chaotic behavior which is identified by a double scroll attractor or a spiral attractor. There are a few criteria for a circuit to exhibit chaotic behavior which include one or more non-linear elements, one or more resistors operating locally at the same time, and three or more energy storage elements.

By varying the value of one of the parameters from 1.0 to 1.1, the trajectories of x(t) are extremely different as depicted by Fig. 3. Thus, the Chua's circuit is the most sensitive of conventional chaotic circuits. Most importantly, the Chua's circuit is one of the most basic durable experimental proofs of chaos that can be readily implemented in a variety of ways [25], and provides the best performance trade-off among the others.

As presented in Fig. 4, two different design for Chua's chaotic equations are implemented to study the behavior. Figure 4a demonstrate the design with inductor (L) valued 28mH and Fig. 4b represents the design with the reduced inductance using the additional non-inverting amplifiers. The inductance of the circuit on the left (i.e., C2) in Fig. 4b can be calculated as below.

$$L = \frac{R7 \times R9 \times R10 \times C}{R8}$$

The simulation of these LTSpice designs is shown in Fig. 4c and d for designs in (a) and (b) respectively, generating Chua's attractor shape. However, the fine distinction can be observed even though they both develop Chua's attractor. The basic building block of these designs is non-inverting Operational Amplifier (Op-Amp).

In communication when the signal is discussed, it is always important to pay attention to the noise component involved within the signal. Signal to Noise Ration (SNR) is the ratio of signal power compared to all other electrical signals, which can be identified as noise. SNR can be calculated by the mean of the signal divided by the standard deviation. SNR provides very critical information about the usability of the signal. Lower SNR tends to introduce the more Gaussian noise, as signal becomes unusable. This is also called "noise floor". During the analysis of the chaotic equations, it is observed that the chaotic signal of Chua's circuit has the SNR near to 10 dB that makes signal more prone to noise and jitter attacks. Figure 5 illustrates the SNR plot of signal X from Chua's equation. The signal with SNR near to 10 dB can be identified as noise and the signal becomes unusable. The characteristic of these chaotic equations is that they generate a signal with similar characteristics as noise signal, even with a message decoded in it. Thus, when transmitted through public channels these signals are just noise. Chua's circuit attains more chaos with parameter sensitivity, that symbolizes the robustness of the chaotic equations to the static noise and eliminates possibilities of data loss.

3 Memristor-based Chua's chaotic circuit

The implementation of memristor-based Chua's chaotic circuit is described in [26] and the same chaotic circuit is implemented using two memristors in [27]. In this work, to increase the key space, we have used more than two memristors in Chua's chaotic circuit.

3.1 Circuit design in LTSpice

Usually, for the non-linear element, a Chua's diode is utilized; however, we have used a memristor instead of a Chua's diode and also replaced a few other resistors with memristors as shown in Fig. 6.

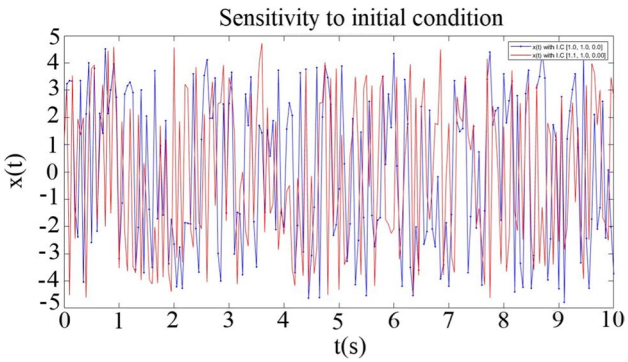


Fig. 3 The plot of sensitivity to initial conditions of Chaos for Chua's circuit

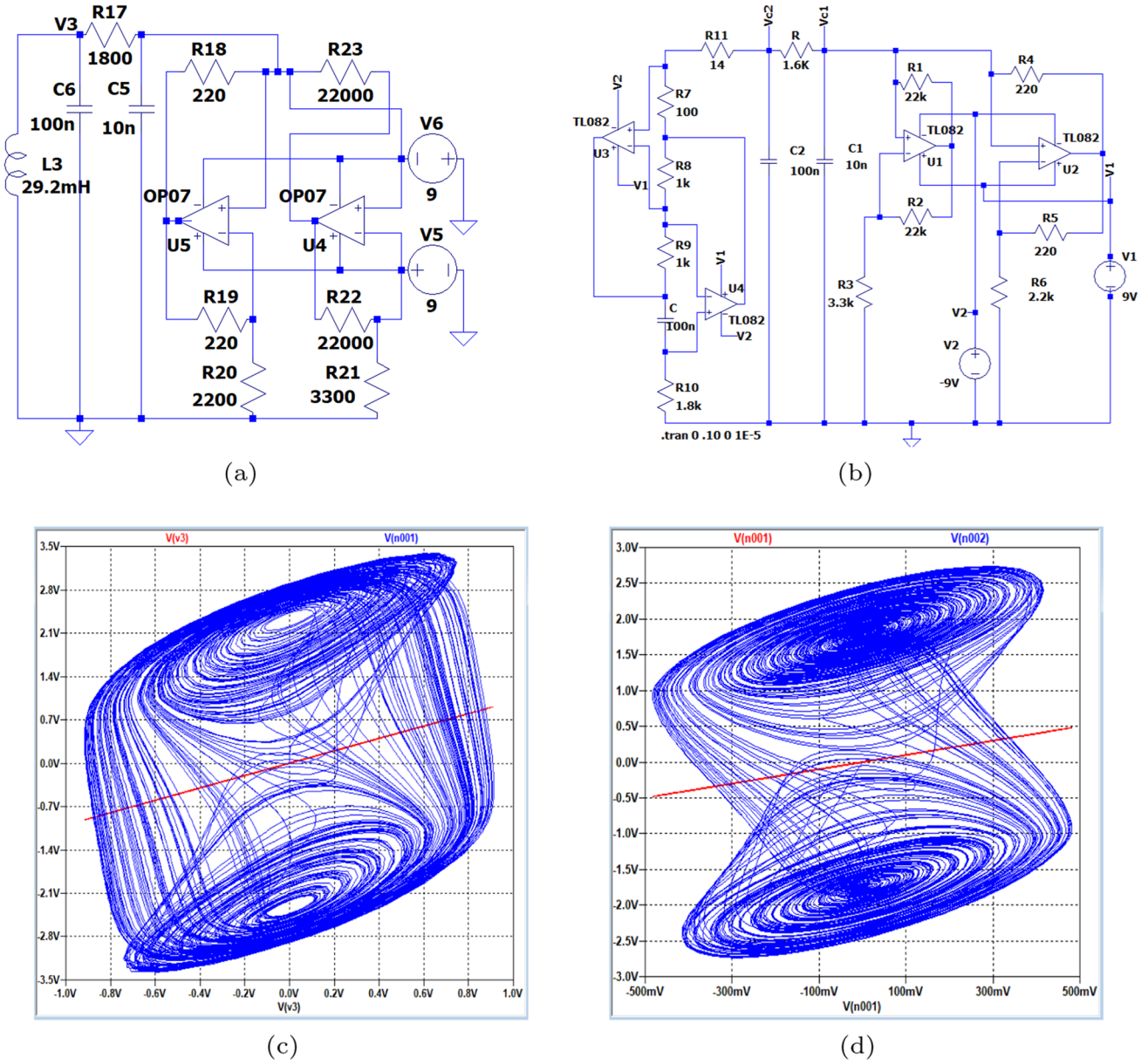


Fig. 4 LTSpice design of Chau's chaos **a** With Inductor (L), **b** Without Inductor (L). The attractor **c** Corresponds to design in **a**, and **d** Corresponds to design in **b**

The chaotic behavior of our designed circuit is shown in Fig. 7 as a double scroll attractor by plotting I-V characteristics for memristor U3 in the circuit. The values of the other memristors are tuned in a range such that they satisfy the chaotic equations to exhibit chaotic behavior.

3.2 Mathematical model in MATLAB Simulink

Most of the researchers have focused on designing a system to reliably transmit the information from the sender to receiver [28] by making improvement in the way chaotic circuit is designed. For example [29] studies the synchronization of chaotic circuit for discontinuous chaotic systems, and [30] uses time-scaling chaotic shift keying encryption for wireless systems. Based on the introduction of a non-linear element, i.e., memristor in the Chua's chaotic circuit, we derived

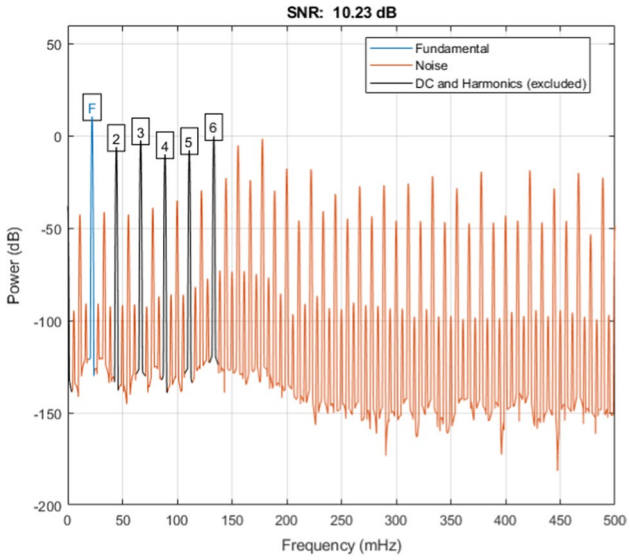


Fig. 5 Signal to Noise Ratio (SNR) for Chua's signal X

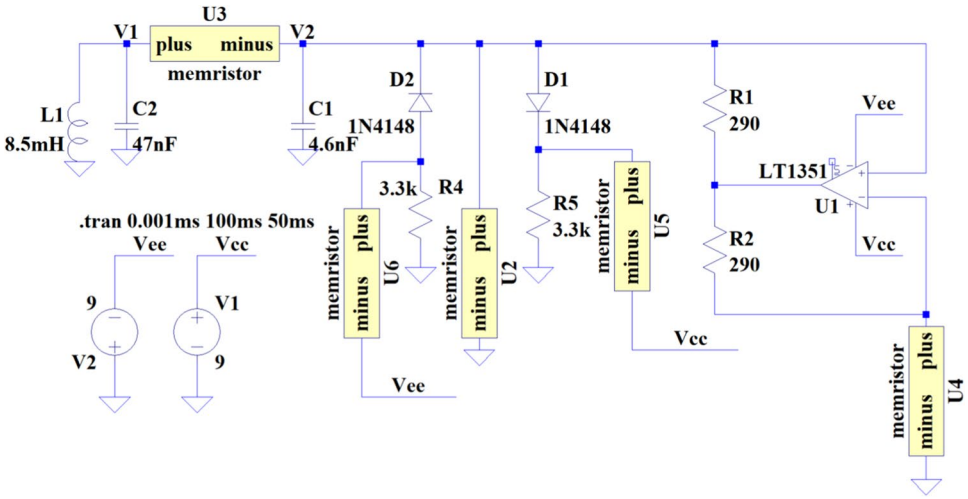


Fig. 6 Implementation of memristor-based Chua's chaotic circuit

the following system of equation by applying Kirchhoff Law around the loop. Figure 8 shows the simulation of chaotic communication using memristor-based Chua's circuit in MATLAB Simulink.

$$\frac{d\Phi}{dt} = v_1 \tag{3}$$

$$\frac{dv_1(t)}{dt} = \frac{1}{C_1}\left(\frac{v_2(t) - v_2(t)}{R} - i_L(t)\right) \tag{4}$$

$$\frac{dv_2(t)}{dt} = \frac{1}{C_2}\left(\frac{v_2(t) - v_2(t)}{R} - i_L(t)\right) \tag{5}$$

$$\frac{di_L(t)}{dt} = \frac{v_2(t)}{L} \tag{6}$$

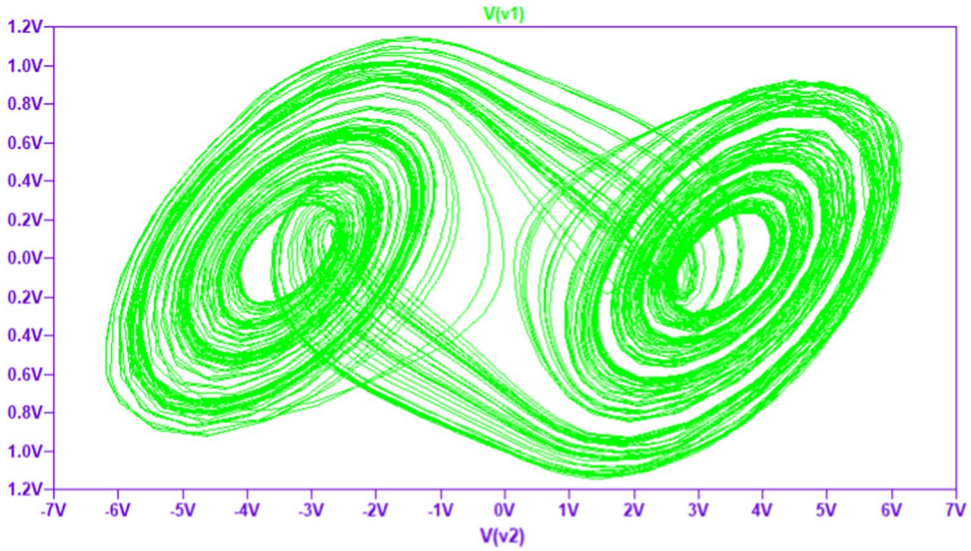


Fig. 7 Double scroll attractor for memristor U3 in Chua's chaotic circuit with values R_{on} = 0.8K R_{off} = 1.65K R_{init} = 1.65K D = 70N uv = 10F p = 1

The Runge–Kutta method was used, using the ode45 function, with a tolerance of 1×10^{-9}, with a sample step of 0.01 and simulating 500 s. Here are the initial values used in Simulink: r = 2000.0; c1 = 6.8e−9; c2 = 68.0e−9; bind = 18.0e−3; alpha = – 0.667D-03; beta = 0.029D-03; tau = 3.499e−05; cn1 = tau/c1r; cn2 = tau/c2*r; cn3 = tau/bind; tspan = 0:0.01:500; x1 = 0.1; x2 = −0.2; x3 = −0.003; x4 = 0.05;

Figure 9 depicts the comparison of the input signal in red and the decoded message in blue. This can be further calibrated to obtain the accurate decoded message.

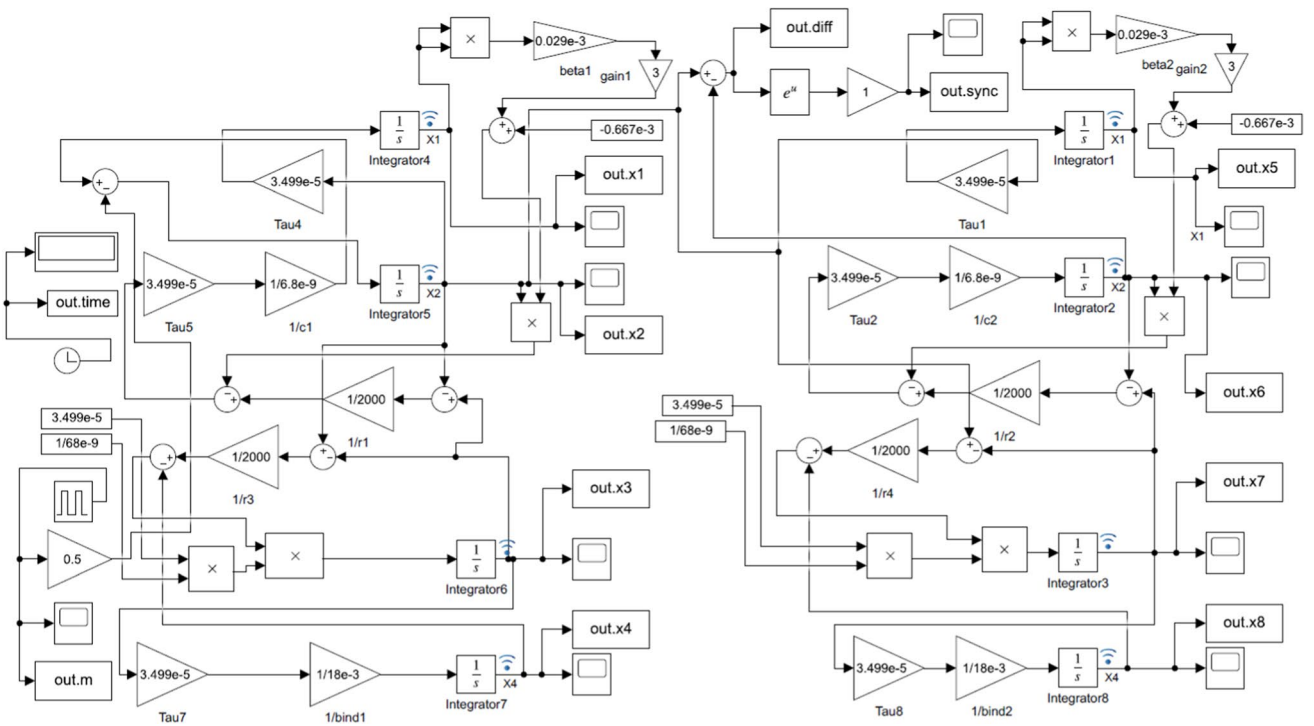


Fig. 8 Simulation of memristor-based Chua's chaotic circuit in MATLAB Simulink

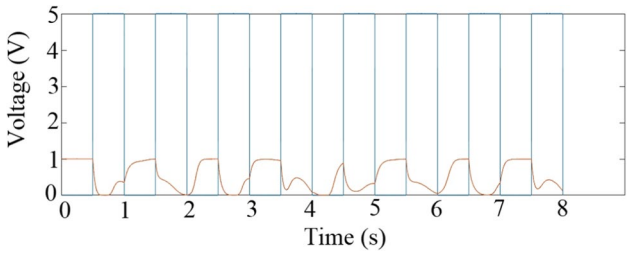


Fig. 9 Comparison of input message and decoded message

4 Logic-locked secure chaotic transceivers

With the existing research, much of the work has been done to design and improve the reliability of data transfer against eavesdroppers using chaotic communication; however, we see a gap in the security of the transceiver design itself. There can be a risk-sensitive scenario where an attacker from an untrusted foundry recreates the transceivers. To overcome this problem, we have the design goal of securing the transceivers by taking inspiration from the properties of logic locking in digital circuits. The circuit can be locked, and the locked version can be sent to the foundry for manufacturing; once the prototyped locked circuit has been returned to the design house, only the authorized users can unlock the original functionality by plugging in the secret key (i.e, programming the memristors based on their predefined continues values.)

While state-of-the-art logic locking methods [11, 12, 31] have focused on designing key-controlled logic gates targeted to secure purely digital circuits or designing digital keys for mixed-signal circuits [16], the main idea in our work is to lock the mixed-signal circuit using the analog memristors, and as we can tune the resistance value of the memristors, these continuous values can act like an analog secret key. Please note that, structurally, all the memristors used are indistinguishable, and the inside foundry attacker cannot find the value of their resistance by analyzing the circuit layout. While the key value in digital logic locking needs to be inserted into a so-called tamper-proof memory that itself may be the point of attack, here the key value is embedded into the circuit and programmed after the manufacturing.

Another part of secure and reliable communication is the design of chaotic circuits. The "key" values (i.e., memristors' values) will be in a certain range, which will always satisfy the chaotic equations. Although this gives a hint to the attacker to at least make an estimate of the range of values and perform the attack, even then, it will be difficult to start with brute-force as there are multiple memristors in the circuit with an exponentially large key space. LTSpice model of our proposed logic-locked transmitter and receiver is shown in Fig. 10. The input message is a pulse signal, which is mixed with the chaotic message, and finally it will be decoded at the receiver end. A β-modulator is also used in the circuit, which is a variable gain circuit designed with an inverting summing amplifier and two separate inverting amplifiers, and is controlled by a switch.

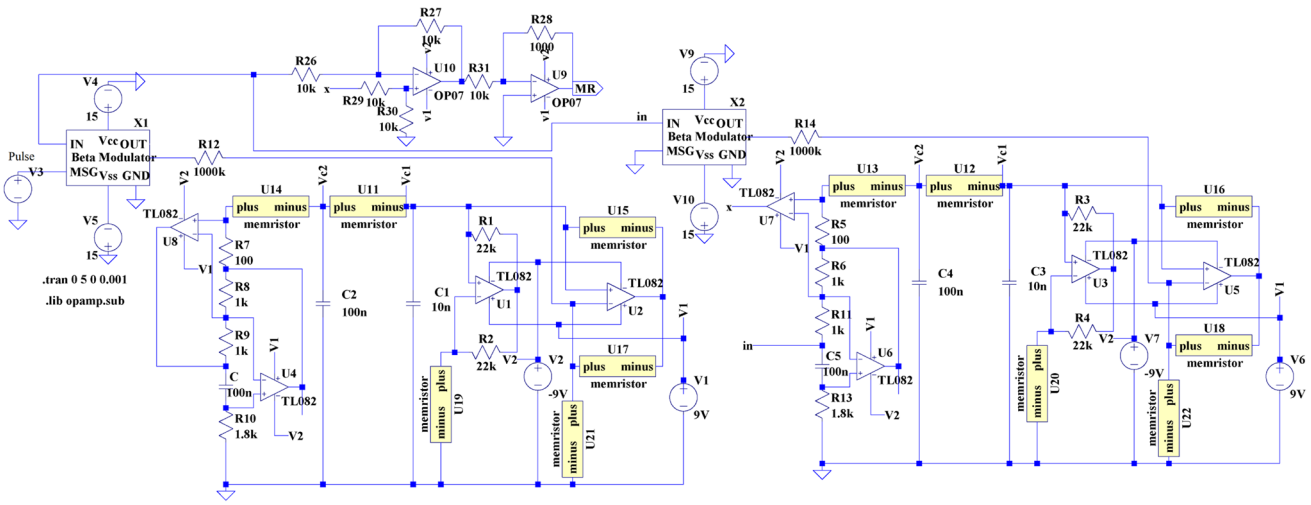


Fig. 10 LTSpice design of transmitter and receiver for memristor-based Chua's chaotic circuit

4.1 Attack analysis

The logic locking methods designed for digital circuits with an *n*-bit key size have 2^n cases with discrete values, such that $R_i \in \{0, 1\} \rightarrow 2^n$ which cannot be broken by brute-force attack but may be challenged by the SAT-based attack [32] that adopts a stronger attacker model by using both a locked version of the circuit and an activated transceiver bought off the market. As the SAT-based attack can handle only Boolean variables, it cannot break analog logic locking. But an advanced attack based on Satisfiability Modulo Theories (SMT) [33] can be applicable here.

However, our research focuses on the use of continuous values for the keys $R_i \in [0, R_{max}]$ which is equivalent to the discrete form with many possibilities. Assuming we have $R_{max} = m\Delta$, the above relationship can be written as $R_i \in [0, \Delta, 2\Delta, 3\Delta, ..., m\Delta] \rightarrow m^n$. For each memristor value, there are $m\Delta$ possibilities with m>>n. In comparison with the traditional method, we have $m^n >> n^n >> 2^n$. In this case, even for the SMT attack [33], the key size will be exponentially large, and not possible to be deciphered in polynomial time. We complement our attack analysis via experiments in Sect. 5.

5 Experimental results

We experimentally implemented different transceivers simulated in [6] using FPGA board Nexys A7 for the system generator design of Lorenz, SprottD, and Chua as shown in Fig. 11. The parameter values, generated in XSG is taken to the MATLAB to plot them and is shown in Fig. 12. When comparing these experimental plots with the simulation plot, it can be seen that the hardware experimental data matches the attractor shape for each equation and generates the chaos. The details of these hardware implementation is published in our recent work [34].

The post-synthesis and the post-implementation data are represented in Table 1 for the hardware utilization. These hardware resources can be considered as Look-Up Table (LUT), Flip-Flops (FF), and Digital Signal Processors (DSP). The chart of these comparisons can also be seen in Fig. 13b.

The circuit design is shared with the untrusted foundry, and once the manufactured circuits are received, the memristor values can be tuned in-house to create a pair of transmitter and receiver that can only work together. The use of memristors secures the mechanism in that even if someone gains access to the entire circuit, the circuit will not function without the specific key inserted. On the other hand, in a situation where an eavesdropper has access to the message transmitted, he or she cannot decode the message without knowing the actual values of the memristors used in the pair of transceivers.

Thus, for experimental validation, we will discuss two scenarios for performing chaos-based communication:

- Scenario 1: We only vary the circuit's component parameters of the transmitter, keeping the receiver's parameters constant to achieve a chaotic circuit and perform secure communication.
- Scenario 2: We can have different values of the memristor component in the transmitter as well as receiver and still achieve secure communication.

In the first scenario, the focus is on the chaotic circuit that can be regulated by the similar system of equations but having slightly different parameters which are in the limit of component tolerances. The second scenario focuses on the chaotic circuit with different dynamical behaviors because of the component's different parameters (parametric mismatches). Finally we also show the results of the SMT attack on our proposed transceiver in Fig. 10.

5.1 Transmitter and receiver with identical memristors

The two circuits, transmitter and receiver are identical, but their component's tolerance range obey the set of equations mentioned in [17] to generate Chua's chaotic behavior. In our experiment we consider two cases and show that if the memristor's values are within the tolerance range, a secure communication is established and the encoded message is decoded successfully.

Case 1: Memristor parameter R_{OFF} = 1.6K for T_x (U11) and R_x (U12)

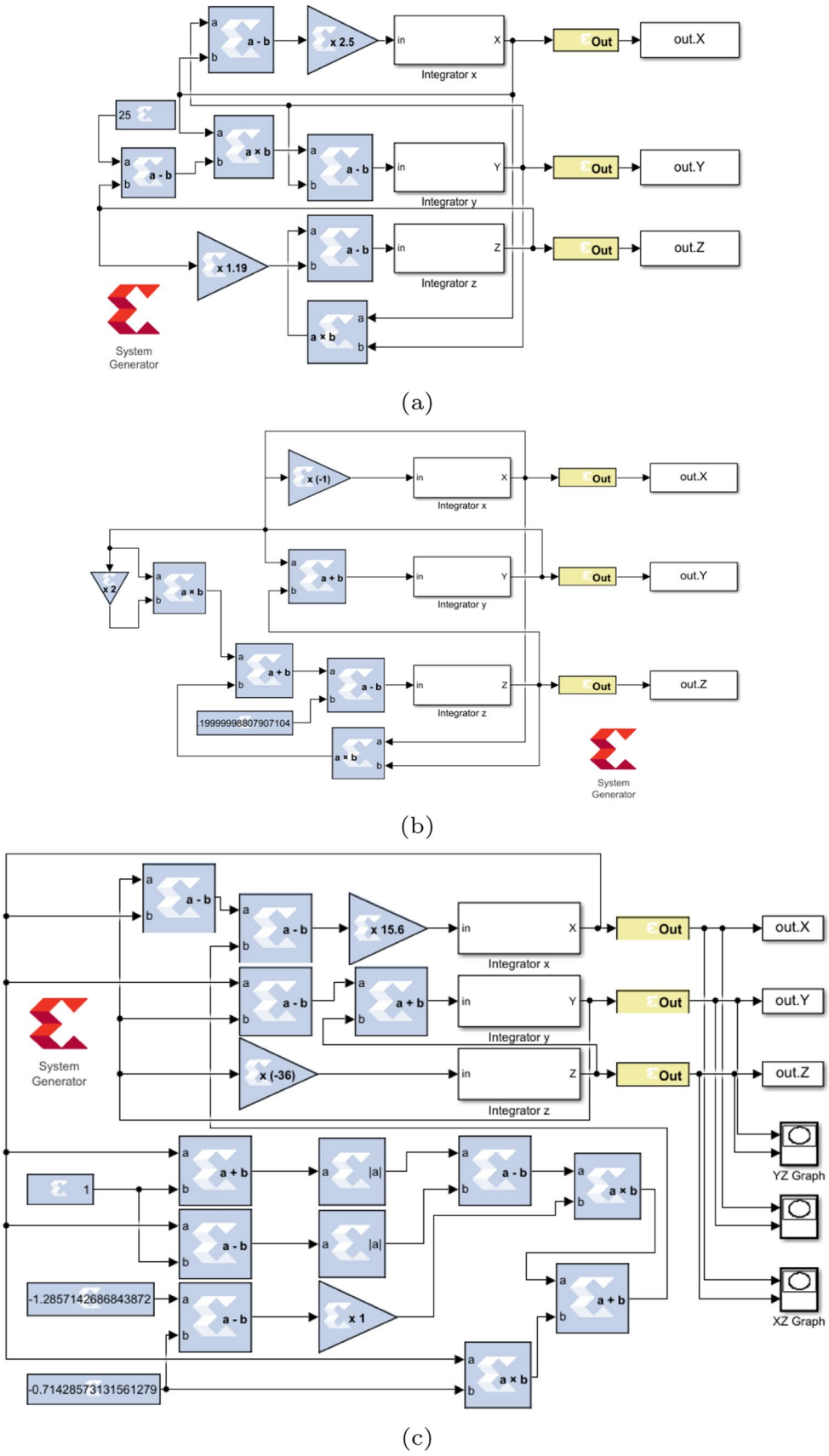


Fig. 11 XSG design for transmitter of **a** Lorenz, **b** SprottD, and **c** Chua

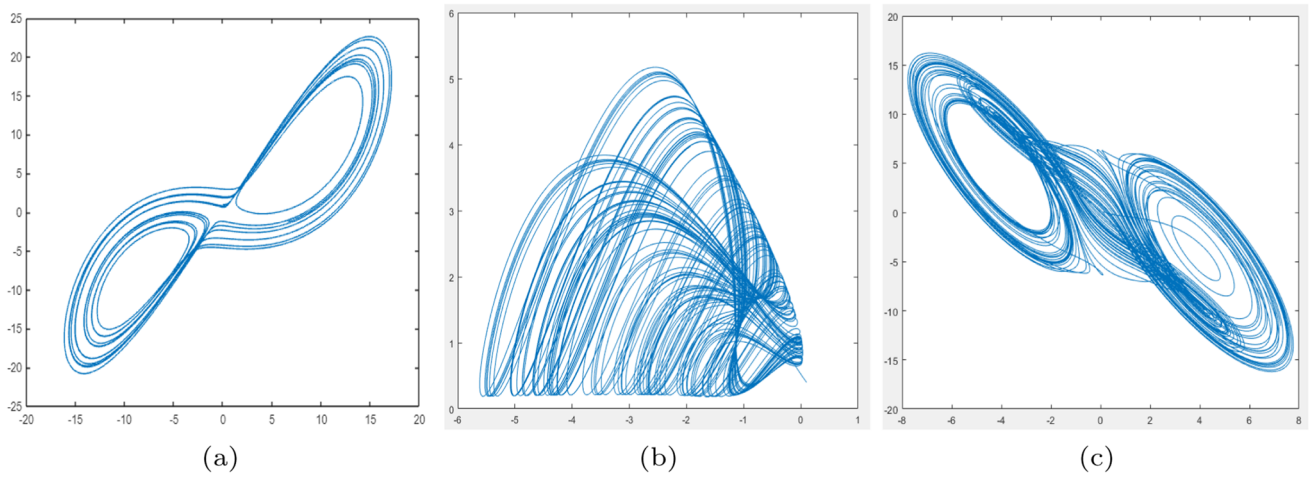


Fig. 12 XSG hardware co-simulation plot for **a** Lorenz, **b** (SprottD), and **c** Chua

Table 1 Comparison of resource utilization

Design	Resource	Available	Utilization	Utilization in %
Lorenz	LUT	20800	850	4.09
	FF	41600	96	0.23
	DSP	90	15	16.67
SprottD	LUT	20800	578	2.78
	FF	41600	96	0.23
	DSP	90	14	15.56
Chua	LUT	20800	933	4.49
	FF	41600	96	0.23
	DSP	90	20	22.22

The corresponding memristor values of T_x and R_x are set identical, but different from the suggested values mentioned in [17]. The motive is to demonstrate large key space using memristors and establish secure and reliable chaotic communication.

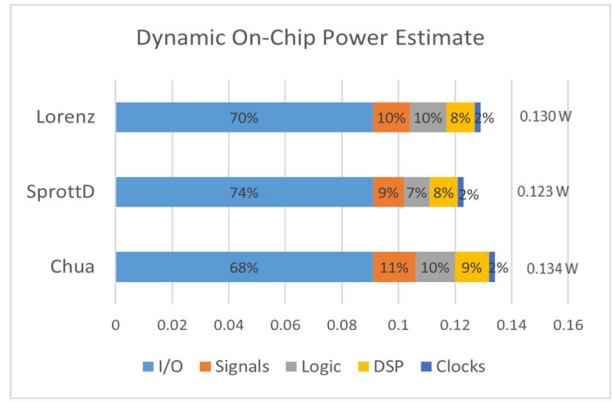
Tx	Rx	Memristor parameters
U11	U12	Ron = 0.8K Roff = 1.6K D = 70N uv = 10F p = 1
U14	U13	Ron = 7 Roff = 14 D = 70N uv = 10F p = 1
U19	U20	Ron = 1.65K Roff = 3.3K D = 70N uv=10F p = 1
U21	U22	Ron = 1.1K Roff = 2.2K D = 70N uv = 10F p = 1

The (T_x, R_x) memristor pairs (U11, U12), (U14, U13), (U19, U20), (U21, U22) also have a slightly different values (within the tolerance range) as mentioned in [20] and the chaotic circuit produces a double scroll attractor as shown in Fig. 14.

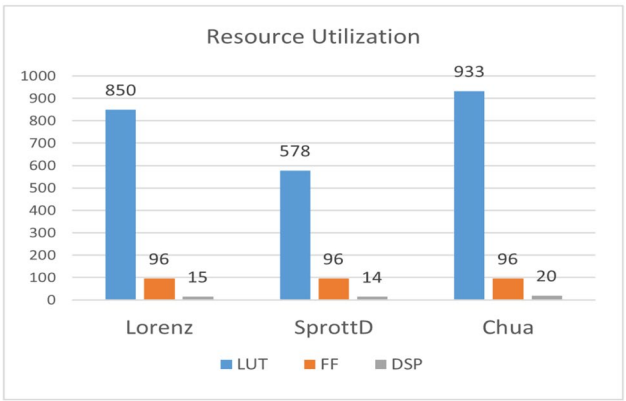
Figure 15 shows the decrypted signal V(mr) and is compared with the originally received signal V(n013) at the receiver end. The signal difference between them is amplified to retrieve the message signal data in order to complete the transaction.

Case 2: Memristor parameter R_{OFF} = 1.8K for T_x (U11) and R_x (U12)

To show that the memristor-based chaotic circuit still holds the chaotic behavior with different parameters as compared to the values used in case 1, a circuit was simulated using the below parameters of the memristors. Figure 16 shows the chaotic behavior of the circuit with memristor value R_{OFF} = 1.8K for T_x (U11) and R_x (U12). The comparison of input message and the decoded message is shown in Fig. 17.



(a)



(b)

Fig. 13 Comparison of chaotic equation for **a** On-chip power estimate, **b** Resource utilization

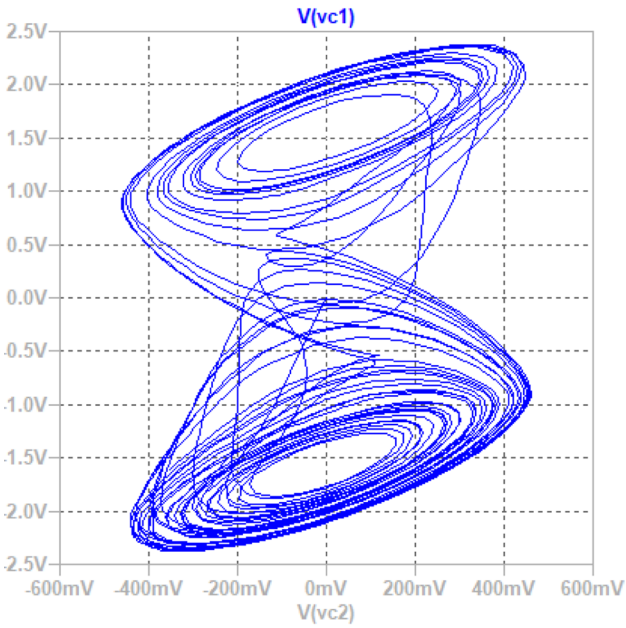


Fig. 14 Double scroll attractor of Chua's chaotic circuit with memristor value R_{OFF} = 1.6K for T_x (U11) and R_x (U12)

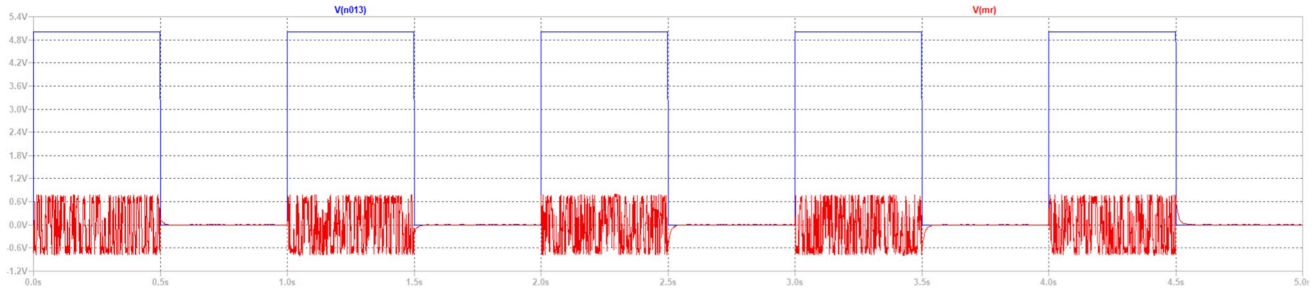


Fig. 15 Case 1: Original message V(n013) in blue color and decoded message V(mr) in red color

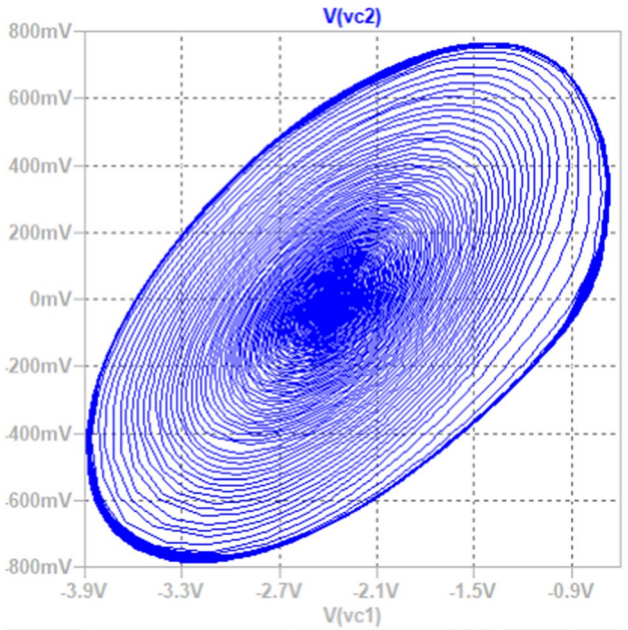


Fig. 16 Spiral attractor of Chua's chaotic circuit with memristor value R_{OFF} = 1.8K for T_x (U11) and R_x (U12)

Fig. 17 Case 2: Original message (V(n013) in blue color and decoded message V(mr) in red color

Tx	Rx	Memristor parameters
U11	U12	Ron = 0.9K Roff = 1.8K D = 70N uv = 10F p = 1
U14	U13	Ron = 7 Roff = 13 D=70N uv = 10F p = 1
U19	U20	Ron = 1.65K Roff = 3.2K D = 70N uv = 10F p = 1
U21	U22	Ron = 1.1K Roff = 2.19K D = 70N uv = 10F p = 1

Fig. 18 Message is not decoded if the memristors values are changed beyond the tolerance

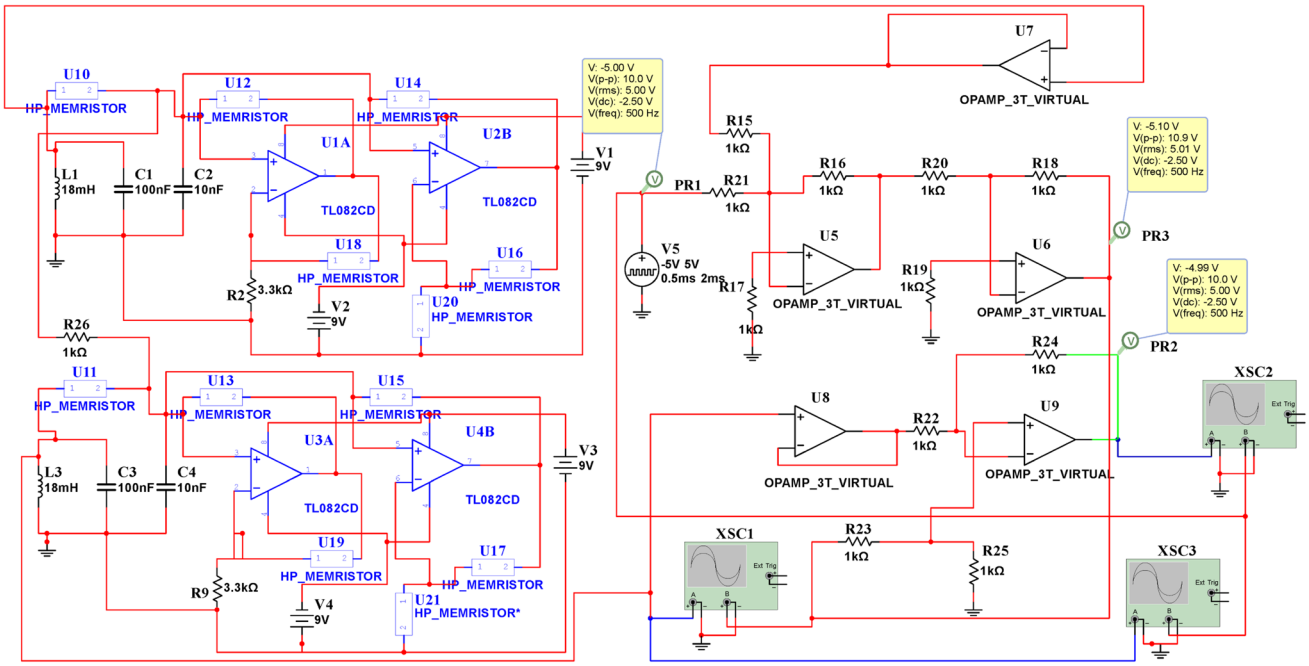


Fig. 19 Design of memristor-based Chua's chaotic circuit—non-identical transmitter and receiver

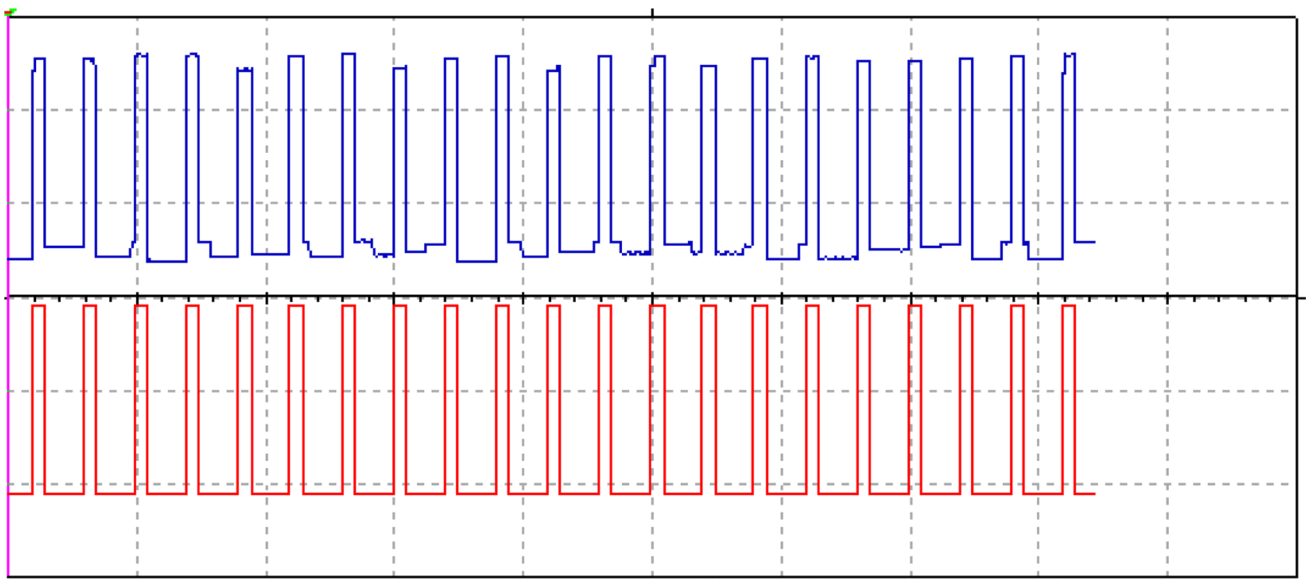


Fig. 20 Comparison of original message in red and decoded message in blue

Further, if the values of the memristor is changed beyond the tolerance range for which it stops exhibiting chaotic behavior, it is observed in Fig. 18 that the transceiver fails to decode the message. If we change the value of R_x (U12) from Ron = 0.9K Roff = 1.8K D = 70N uv = 10F p = 1 to Ron = 0.9K Roff = 1.79K D = 70N uv = 10F p = 1, the message will not be decoded. So, the attacker has to be aware of all the values of the memristors present in the transmitter circuit.

5.2 Transmitter and receiver with non-identical memristors

Second scenario is to validate if the original message is decoded at the receiver end when the Chua's chaotic circuit are non-identical, i.e., parameter mismatch. For simplicity, only one memristor value is different in transmitter and receiver, and other values of the memristor are identical.

The table below shows the values of the memristors used in the circuit of Fig. 19, which remains same as Tx and Rx. Figure 20 shows the original message is decoded successfully even when the circuit are non-identical.

		Memristor parameters
Tx	U10	Ron = 0.9K Roff = 1.85K D = 10N uv = 10F p = 1
Rx	U11	Ron = 1K Roff = 1.75K D = 10N uv = 10F p = 1
Tx	Rx	Memristor parameters
U12	U13	Ron = 11K Roff = 22K D = 10N uv = 10F p = 1
U14	U15	Ron = 110 Roff = 220 D = 10N uv = 10F p = 1
U16	U17	Ron = 110 Roff = 220 D = 10N uv = 10F p = 1
U18	U19	Ron = 11K Roff = 22K D = 10N uv = 10F p = 1
U20	U21	Ron = 1.1K Roff = 2.2K D = 10N uv = 10F p = 1

According to the above experiments, if the attacker discovers the correct key (i.e., the resistance of the memristors) within the range of $\Delta = 0.1$, he or she may still decode the message; however, beyond this range, he/she has no luck. As a result, we take this value into account and employ the SMT attack [33] on our proposed transceiver in Fig. 10. We assume that the attacker has access to the prototyped receiver bought off the market and the transmitter design. The goal is to find the resistance of the memristors in such a way that the transmitter will be able to sync with the receiver. Each transmitter has six memristors; with R_{max} set to 10K, each memristor can take 100,000 different values resulting in $100,000^6$ cases which can be modeled with a key size of 100 bits. We used 8GB of RAM and a 4-core processor at 2.20Ghz to run the attack. The attack, however, reports no key after 24 h of running. This is due to the fact that the modeling of our analog logic locking circuit requires a large key space.

6 Conclusion

In this paper, we proposed a reliable and secure memristor-based chaotic transceiver that is resistant both to eavesdroppers and untrusted foundries. The approach is effective due to the large key space created by implementing memristors in the circuit, along with the logic locking. A simulation of the proposed system was performed in MATLAB Simulink to validate the memristor-based chaotic system of equations. Moreover, the research complements the communication

security of risk-sensitive implantable and wearable devices at the application level. A possible future direction could be designing a chaotic system with memcapacitor and meminductor along with the memristors to achieve an even larger key space.

Acknowledgements This material is based upon work supported by the National Science Foundation under Grant No. 2131156.

Author contributions RV was involved in performing the literature review and designing the system; RV was also involved in the simulation and analysis of the system and the development of the manuscript. RM performed the literature review, designed, simulated, and analyzed the system. RM has also been involved the development of the manuscript. AH was involved in designing the system, performing simulation and analysis of the system, and developing the manuscript. AR was involved in performing the literature review and designing the system; AR was also involved in the simulation and analysis of the system and the development of the manuscript. All authors reviewed the manuscript. All authors read and approved the final manuscript.

Data availibility Data sharing is not applicable to this article as this paper generates and analyze simulations that are done through Matlab Simulink and LTSPice. The Simulink and LTspice files generated during and/or analyzed during the current study are not publicly available but are available from the corresponding author upon reasonable request.

Declarations

Competing interests The authors declare no competing interests.

References

1. Hedayatipour A, Mcfarlane N. Wearables for the next pandemic. IEEE Access. 2020;8:184457–74. https://doi.org/10.1109/ACCESS.2020.3029130.
2. Wang Z, Gong L, Yang J, Zhang X. Cloud-assisted elliptic curve password authenticated key exchange protocol for wearable healthcare monitoring system. Concurr Comput Pract Exp. 2022;34(9):5734.
3. Wen H, Zhang C, Chen P, Chen R, Xu J, Liao Y, Liang Z, Shen D, Zhou L, Ke J. A quantum chaotic image cryptosystem and its application in iot secure communication. IEEE Access. 2021;9:20481–92.
4. Al Momin MA. Medical device security. In: Security, Data Analytics, and Energy-Aware Solutions in the IoT, pp. 173–191. IGI Global; 2022
5. Zhong G-Q, Ayrom F. Experimental confirmation of chaos from Chua's circuit. Int J Circuit Theory Appl. 1985;13(1):93–8.
6. Hedayatipour A, Monani R, Rezaei A, Aliasgari M, Sayadi H. A comprehensive analysis of chaos-based secure systems. In: Silicon Valley Cybersecurity Conference, pp. 90–105 (2021). Springer
7. Duan Z, Wang H, He S, Li S, Yan S, Zhao X, Yu X, Yang G, Tan H. A fully integrated chaos generator based on voltage controlled oscillator. Microelectron J. 2022;126: 105514.
8. Onuki K, Cho K, Horio Y, Miyano T. Secret-key exchange through synchronization of randomized chaotic oscillators aided by logistic hash function. IEEE Trans Circuits Systems I: Regular Papers. 2022;69(4):1655–67.
9. Zhang J, Guo Y, Xu L, Zhu X, Yang J. Hyperchaotic circuit design based on memristor and its application in image encryption. Microelectron Eng. 2022;265: 111872.
10. Rezaei A, Gu J, Zhou, H. Hybrid memristor-cmos obfuscation against untrusted foundries. In: 2019 IEEE Computer Society Annual Symposium on VLSI (ISVLSI), pp. 535–540 (2019)
11. Roy JA, Koushanfar F, Markov IL. Epic: Ending piracy of integrated circuits. In: 2008 Design, Automation and Test in Europe, pp. 1069–1074 (2008). https://doi.org/10.1109/DATE.2008.4484823
12. Limaye N, Chowdhury AB, Pilato C, Nabeel MT, Sinanoglu O, Garg S, Karri R. Fortifying rtl locking against oracle-less (untrusted foundry) and oracle-guided attacks. In: 2021 58th ACM/IEEE Design Automation Conference (DAC), pp. 91–96 (2021). IEEE
13. Majumder B, Hasan S, Uddin M, Rose GS. Chaos computing for mitigating side channel attack. In: 2018 IEEE International Symposium on Hardware Oriented Security and Trust (HOST), pp. 143–146 (2018). https://doi.org/10.1109/HST.2018.8383903
14. Hasan MS, Majumder MB, Shanta AS, Uddin M, Rose GS. A chaos-based complex micro-instruction set for mitigating instruction reverse engineering. J Hardw Syst Secur. 2020;4:69–85.
15. Kamali HM, Azar KZ, Homayoun H, Sasan A. Chaolock: Yet another sat-hard logic locking using chaos computing. In: 2021 22nd International Symposium on Quality Electronic Design (ISQED), pp. 387–394 (2021). https://doi.org/10.1109/ISQED51717.2021.9424321
16. Leonhard J, Yasin M, Turk S, Nabeel MT, Louërat M-M, Chotin-Avot R, Aboushady H, Sinanoglu O, Stratigopoulos H-G. Mixlock: Securing mixed-signal circuits via logic locking. In: 2019 Design, Automation and Test in Europe Conference & Exhibition (DATE), pp. 84–89 (2019). https://doi.org/10.23919/DATE.2019.8715043

17. Chua L. Memristor-the missing circuit element. IEEE Trans Circuit Theory. 1971;18(5):507–19.
18. Joglekar YN, Wolf SJ. The elusive memristor: properties of basic electrical circuits. Eur J Phys. 2009;30(4):661.
19. Prodromakis T, Peh BP, Papavassiliou C, Toumazou C. A versatile memristor model with nonlinear dopant kinetics. IEEE Trans Electron Devices. 2011;58(9):3099–105. https://doi.org/10.1109/TED.2011.2158004.
20. Biolek Z, Biolek D, Biolkova V. Spice model of memristor with nonlinear dopant drift. Radioengineering. 2009;18(2).
21. Cuomo KM, Oppenheim AV, Strogatz SH. Synchronization of lorenz-based chaotic circuits with applications to communications. IEEE Trans Circuits Systems II: Analog Digital Signal Process. 1993;40(10):626–33. https://doi.org/10.1109/82.246163.
22. Butusov DN, Karimov TI, Lizunova IA, Soldatkina AA, Popova EN. Synchronization of analog and discrete rössler chaotic systems. In: 2017 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (ElConRus), pp. 265–270 (2017). https://doi.org/10.1109/ElConRus.2017.7910544
23. Chua LO. Chua's circuit 10 years later. Int J Circuit Theory Appl. 1995;22(4):279–305. https://doi.org/10.1002/cta.4490220404.
24. Liao T-L, Chen H-C, Peng C-Y, Hou Y-Y. Chaos-based secure communications in biomedical information application. Electronics. 2021;10(3):359. https://doi.org/10.3390/electronics10030359.
25. Fortuna L. Chua's circuit implementations: yesterday, today and tomorrow. World Sci Ser Nonlinear Sci Ser A. 2009;65
26. Muthuswamy B. Implementing memristor based chaotic circuits. Int J Bifurcation Chaos. 2010;20(05):1335–50.
27. Bao B, Jiang T, Wang G, Jin P, Bao H, Chen M. Two-memristor-based chua's hyperchaotic circuit with plane equilibrium and its extreme multistability. Nonlinear Dynam. 2017;89(2):1157–71.
28. Guo Q, Wang N, Zhang G. A novel four-element rclm hyperchaotic circuit based on current-controlled extended memristor. AEU-Int J Electron Commun. 2022;156: 154391.
29. Yang X, Yang Z, Nie X. Exponential synchronization of discontinuous chaotic systems via delayed impulsive control and its application to secure communication. Commun Nonlinear Sci Numer Simul. 2014;19(5):1529–43.
30. Anderson K, Hedayatipour A, McFarlane N. A wireless time-scaling chaotic shift keying encryption system for biosensing systems. In: 2021 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC), pp. 7594–7597 (2021). https://doi.org/10.1109/EMBC46164.2021.9630142
31. Pilato C, Chowdhury AB, Sciuto D, Garg S, Karri R. Assure: Rtl locking against an untrusted foundry. IEEE Transactions on Very Large Scale Integration (VLSI) Systems 29(7), 1306–1318 (2021). https://doi.org/10.1109/TVLSI.2021.3074004
32. Subramanyan P, Ray S, Malik S. Evaluating the security of logic encryption algorithms. In: 2015 IEEE International Symposium on Hardware Oriented Security and Trust (HOST), pp. 137–143 (2015). https://doi.org/10.1109/HST.2015.7140252
33. Jayasankaran NG, Borbon AS, Abuellil A, Sánchez-Sinencio E, Hu J, Rajendran J. Breaking analog locking techniques via satisfiability modulo theories. In: 2019 IEEE International Test Conference (ITC), pp. 1–10 (2019). https://doi.org/10.1109/ITC44170.2019.9000113
34. Monani R, Rogers B, Rezaei A, Hedayatipour A. Implementation of chaotic encryption architecture on fpga for on-chip secure communication. In: 2022 IEEE Green Energy and Smart System Systems (IGESSC), pp. 1–6 (2022). IEEE