INDIAN JOURNAL OF SCIENCE AND TECHNOLOGY

RESEARCH ARTICLE

FPGA Implementation of a Novel Two-internal-State Memristor and its Two Component Chaotic Circuit

Njimboh Henry Alombah[1]*, Achille Ecladore Tchahou Tchendjeu[2], Kengne Romanic[3], François Calvin Talla[3], Hillaire Bertrand Fotsin[3]

1 Department of Electrical and Electronics Engineering, College of Technology, University of Bamenda, P.O. Box 39, Bambili, Cameroon
2 Department of Computer Engineering, National Higher Polytechnic Institute, University of Bamenda, P.O. Box 39, Bambili, Cameroon
3 Unit of Condensed Matter Research, Electronics and Signal Processing, LAMACET, Faculty of Science, Department of Physics, University of Dschang, P.O. Box 67, Dschang, Cameroon

OPEN ACCESS

Received: 01.04.2021
Accepted: 14.07.2021
Published: 04.08.2021

Citation: Alombah NH, Tchendjeu AET, Romanic K, Talla FC, Fotsin HB (2021) FPGA Implementation of a Novel Two-internal-State Memristor and its Two Component Chaotic Circuit. Indian Journal of Science and Technology 14(27): 2257-2271. https://doi.org/10.17485/IJST/v14i27.532

* **Corresponding author**.

alonjims@yahoo.com

Funding: None

Competing Interests: None

Published By Indian Society for Education and Environment (iSee)

ISSN
Print: 0974-6846
Electronic: 0974-5645

Abstract

Objectives: To propose a second-order locally- active memristor, a two-component chaotic circuit resulting from this current-controlled generic memristor and an application in steganography. **Methods/statistical analysis**: Using a one-state-variable first-order memristor, a model is proposed which is obtained by modifying a locally-active memristor based on a current-controlled generic memristor. The model has two internal state variables: a voltage stored up in a capacitor and a current stored up in an inductor. With an external inductor, 3D-two-component chaotic circuit is developed. Numerical studies are made using MATLAB and confirmed by a field programmable gate array (FPGA) based hardware implementation. **Findings**: A two-state-variable based second-order memristor model is presented. The novel memristor configuration leads to the design of a simple two-component chaotic circuit. By investigating the characteristics of the memristor, it is shown that the memristor can be switched from a predominantly passive region to an active region with a wide locally-active region. An application in steganography helps to hide a secrete message inside an image. **Application/improvements**: The results obtained in this investigation will enrich the literature of memristive circuits, enhance the simplification of chaotic circuits and can be used to improve the memristive circuit based applications in many research domains such as secure information in telecommunications, Random Number Generation (RNG) and image encryption.

Keywords: Memristor; Two component circuit; Chaos; Local activity; FPGA; Steganography

1 Introduction

A number of important applications would benefit from the design and implementation of a locally-active memristor, which is defined to be any memristor that exhibits negative

differential memristance or memductance for at least a current or a voltage applied to the memristor[1]. This is one of the drives of this research work, to realize a locally active memristor that has a wide locally active region characteristics and for some parameter values or frequencies of excitation is entirely active. Recently, the investigation of memristor chaotic circuits and systems is a hot issue and many researchers are delving into such studies and many applications are being derived from memristor based chaotic circuits[2,3].

Even though memristors have a promising future, solid-state samples are still unavailable and this constitutes a problem to researchers. Thus, there have been many initiatives to develop memristor emulators and to study their behavior and the possible applications[4,5]. The emulation circuits are built to mimic the behavior of the memristor based on the modeling equations.

The analog memristor emulators are usually built using analog commercial off-the-shelter components[6]. This is inefficient for large scale applications that require a huge number of memristors such as neural networks. On the other hand, FPGA implementations of the memristors are much more advantageous because they are easily programmable, reconfigurable, controllable, precise and exhibit better performance[6].

Due to the unique electrical performances of the memristor, 3-element or N-element memristive circuits, especially the chaotic circuits, have been widely explored and reported in recent years. Muthuswamy and Chua in 2010[7] observed chaos with an autonomous chaotic circuit that uses only three elements: a linear passive inductor, a linear passive capacitor and a nonlinear active memristor. The circuit was modelled by a system of three differential equations. As of 2010, it was the simplest chaotic circuit in literature. In 2012, Tchitnga et al.[8] showed evidence of chaos in an autonomous Hartley Oscillator made simply of a JFET and a tapped coil. The authors modeled the system by four differential equations. In 2017, in a paper titled: "A simple meminductor-based chaotic system with complicated dynamics",[9] the authors proposed a three component chaotic circuit made up of a resistor, a capacitor and a meminductor connected in parallel. The circuit is modeled in a three differential equation system and shows complicated dynamics. Many more strives have been made of recent in an effort to simplify chaotic systems in terms the number of elements used in the circuit[10–12]. Recently, researchers are using not just the properties of memristors, but memory devices (memdevices) to reduce the number of elements in a circuit that are required for chaotic oscillations[13]. This push has been ramped up with the advent of these memory devices.

A number of memristor models have been developed and their circuit emulators, however, most of these models are first-order and utilized only a one-state-variable. This is a simplification which is not adequate for accurate modeling.

In this work, we exploit the "actual simplest chaotic circuit" proposed by Muthuswamy and Chua in 2010 to develop a much simpler chaotic circuit made up of two components (the memristor and an inductor) and modelled by three differential equation having two nonlinear terms. Our memristor model has two internal state variable, its memristance can be positive, zero and negative, has a wide range for local activity and will act like the memristor model in[7] for some parameter values. Since our model has two state variables, we require just a single variable to satisfy the condition for the system to be chaotic. This third variable is made available by a passive linear inductor which is added to our memristor model. The hardware implementation of this generic charge controlled memristor and the two component chaotic oscillator is designed utilizing analogue circuitry and FPGA.

2 The Two State Locally-Active Generic Memristor Model

2.1 Circuitry and mathematical model

In[14,15], all memristors are classified into three classes including the ideal memristor, the generic memristor and the extended memristor and presented the mathematical definitions of memristors. The ideal memristor is the simplest and most practical model. The ideal memristor models developed by the HP Lab is widely used but does not fit the anticipated nonlinear behaviors of a real memristor. The extended memristor is an extended form of ideal memristor and Chua generalized the memristor concept to a much broader class of nonlinear dynamical systems and named it as memristive systems[16]. The mathematical definition of generic memristor model is given below.

$$\left.\begin{aligned} v &= R(\mathbf{y})i \\ \frac{d\mathbf{y}}{dt} &= \mathbf{f}(\mathbf{y}, i) \end{aligned}\right\} \tag{1}$$

This is for a generic current – controlled memristor, representing the state-dependent Ohm's Law equation and the state equation. $\mathbf{y}$ is an n-dimensional vector of the internal state variables and $d\mathbf{y}/dt$ is the time derivative of the state vector. In[7] an active generic memristor is defined where n = 1; a voltage stored up in a capacitor inside the memristor.

The circuit design and implemented memristor is given by the equation below:

$$\left.\begin{aligned} v_M &= \beta I_S \left(y^2 - 1\right) i_M \\ \frac{dy}{dt} &= -\frac{I_S i_{L1}}{R_b C_M} - \frac{y}{R_P C_M} + \frac{y I_S i_{L1}}{R_a C_M} \end{aligned}\right\} \tag{2}$$

Where: $i_{L1} = -i_M$. i_{L1} is the current through the external inductor used to measure the current entering memristor as seen in [7].

This represents the Muthuswamy-Chua memristor model, which is a single state (first order) locally active generic memristor.

To obtain a two state (second order) generic memristor, we introduced another state variable by connecting an inductor in parallel with the capacitor C_M in Figure 1, (we produce a tank circuit represented by RSTU). This second state variable is a current which is stored up in the inductor, L_M.

In the modified memristor, the first part of Eq. (2) is maintained, and the equations for the dynamics of the states of our model is obtained by applying KCL at the junction A and KVL round loop RSTUR to obtain Eq. (3):

$$\left.\begin{aligned} \frac{dy}{dt} &= \frac{I_s i_M}{R_b C_M} - \frac{y}{R_p C_M} - \frac{y I_S i_M}{R_a C_M} - \frac{i_{L_M}}{C_M} \\ \frac{di_{L_M}}{dt} &= \frac{y}{L_M} - \frac{r i_{L_M}}{L_M} \end{aligned}\right\} \tag{3}$$

So our model of the memristor has two internal variables: **(i)** the voltage across the capacitor C_M and **(ii)** the current through the inductor L_M. This makes it a second order memristor.

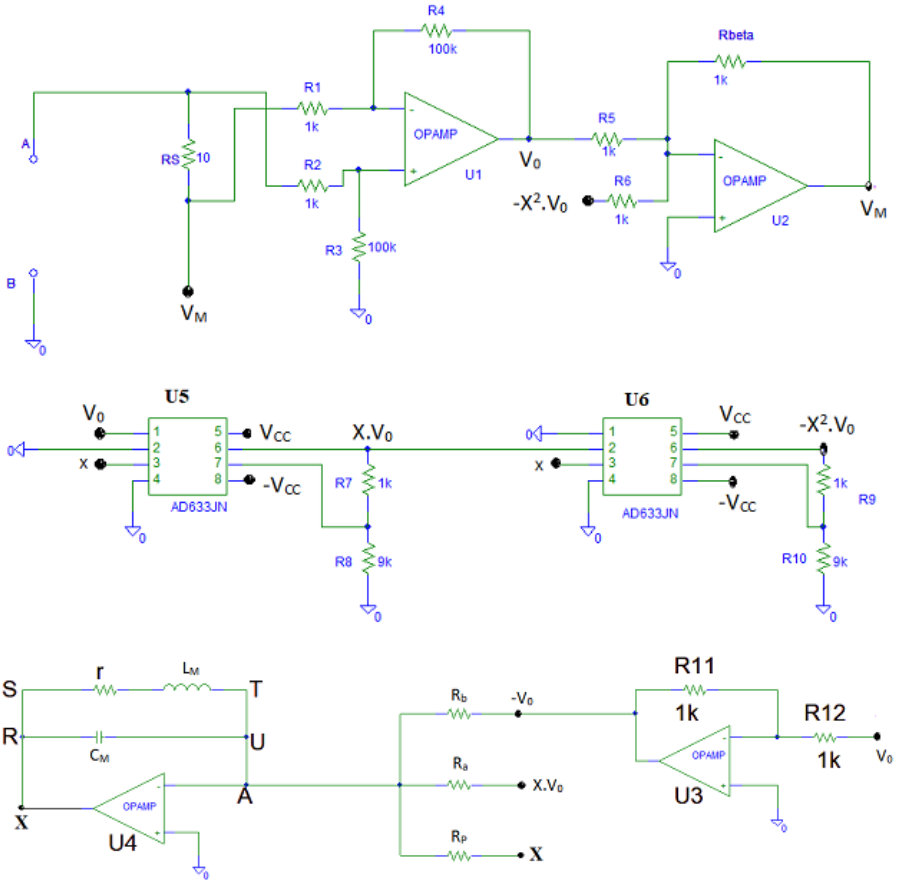

Fig 1. Detailed schematic of the modified memristor model.

2.2 Transforming the Circuit Equations to System Equation

Let us choose the non-dimension time: $\tau = \frac{t}{R_a C_m}; v_M(t) = v_M(\tau); y(t) = y(\tau); z(\tau) = I_s i_M(t)$.

The first part of Eq. (2) becomes: $v_M(\tau) = \beta\left(y^2 - 1\right) i_M(\tau)$.

To excite our memristor circuit, a sinusoidal current given by (4) is used.

$$i_M(\tau) = I_0 \sin\left(2\pi f_d \tau\right) \tag{4}$$

where the real frequency in Hz, $f = \frac{f_d}{R_a C_m}$ and the amplitude of the current in amperesis: $I_{\max} = \frac{I_S}{I_0}$.

Eq. (3) becomes Eq. (5) as follow:

$$\left.\begin{aligned} \dot{y} &= P i_M - \alpha y - y i_M - \gamma z \\ \dot{z} &= K_1 y - K_2 z \end{aligned}\right\} \tag{5}$$

where: $P = \frac{R_a}{R_b}, \alpha = \frac{R_a}{R_P}, \gamma = \frac{R_a}{I_s}, K_1 = \frac{R_a C_m I_S}{L_M}, K_2 = \frac{R_a C_M r}{L_M}$ with the parameter values: $P = 5_2 \alpha = 1, \gamma = 1, \beta = 1, K_1 = 1, K_2 = 0.1$ and the initial conditions: $y(0) = 0.1, z(0) = 0.1$, we investigate the fingerprints of this memristor model.

2.3 Memristor Characteristics

The next step is to prove that our model satisfies the definition of a memristor or memristive system and also verify the local activity of our memristive system.

2.3.1 Pinched Hysteresis Loop of the Memristor

Any two-terminal device exhibiting a pinched hysteresis loop which always passes through the origin in the voltage-current plane, when driven by any periodic input current source, or voltage source with zero DC component, is called a memristor[14]. Thus, the pinched hysteresis loop is always used as the characteristic fingerprint to identify the memristor[17]. Moreover, the pinched hysteresis loop should shrink to a single-valued function when the frequency tends to infinity[18].

Figure 2 shows the pinched hysteresis loops of the above generic memristor at different frequencies when it is driven by a sinusoidal current with an amplitude of 1mA. Figure 2(a) shows the curve at frequencies: f = 0.25 kHz, 0.3 kHz, 0.4 kHz , 0.8 kHz; while Figure 2(b) shows the curve at frequency f = 0.85 kHz, 1 kHz, 2 kHz, and 10 kHz. By combining Figure 2(a) and (b), we observe that the pinched hysteresis loop of memristor shrinks to a single-valued function as we increase the frequency of input current. Notice that Figure 2(b) shows that the memristor has negative memristance region and also degenerates into a linear negative memristance for frequencies greater than or equal to 10 kHz. The figures have been plotted separately for the purpose of clarity; they can actually be plotted on the same graph. However, the larger amplitudes at lower frequencies will dwarf the curves at higher frequencies.

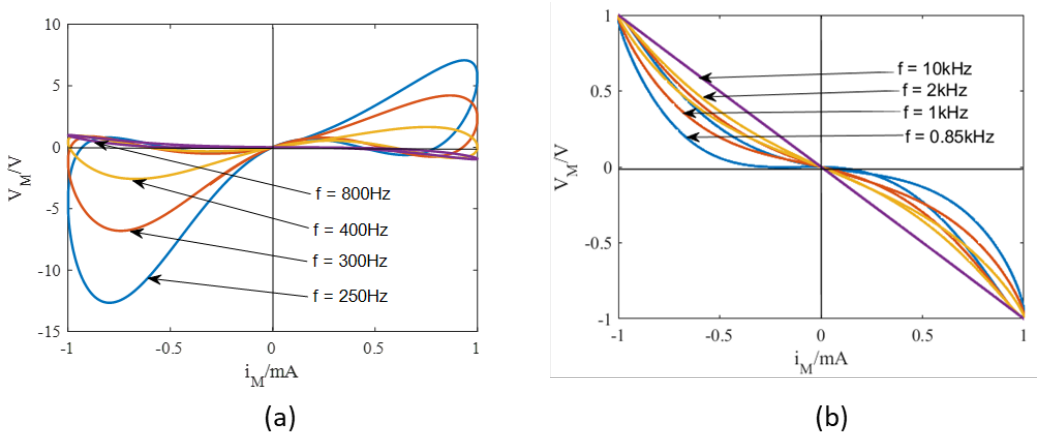

Fig 2. The relationship between current and voltage for the locally active memristor- pinched hysteresis loops varying with the frequency of excitation. The loops switching from the predominantly passive region (a) to the active region (b), second and fourth quadrant where the memristance is negative.

It also worth noting that the nature of the hysteresis loops is also dependent on the parameters of the memrsitor. In Figure 3, with changing values of the parameter P, at fixed amplitude (1mA) of excitation and fixed frequency (0.5 kHz), the area of the hysteresis loops decreases with decrease in the value of P and also switches from predominantly passive to the active region. With values of P less than or equal to 0.1, the loop degenerates into a linear negative memristance.

As we know, the pinched i–v hysteresis loops called fingerprints provide necessary and sufficient conditions for identifying whether a device is a memristor or not. But the most general memristor identification scheme is the Coincident Zero- Crossing Signatures, i.e. the waveform of the current v(t) associated with the voltage i(t) measured from a current-controlled memristor must cross the time axis whenever i(t) = 0. However, observe fromFigure 3(b) that the zero-crossing points of v(t) include not only all zero-crossing points of i(t), but also several additional zero-crossing points when i(t) $\neq$0, which are shown with the little black dots.

2.3.2 DC I-V Loci of the Memristor

The DC Voltage-Current characteristic (DC I-V loci) is a smooth curve consisting of a set of points and a sufficient test to claim that the memristor is locally-active[13]. These points of DC I-V loci can be derived in steps[19] depending on the Eq. 5 and the Ohm's laws dependent equation.

To derive the dc characteristics, we set: $\dot{y} = 0, \quad \dot{z} = 0$ in (5) since by definition at DC all derivatives are zero. We solve for the internal state (y) of the memristor in terms of the current (worth noting that y is a function of z).

$$y = \frac{Pi_M}{\alpha + i_M + \Omega} \tag{6}$$

where: $\Omega = \frac{\gamma K_1}{K_2}$.

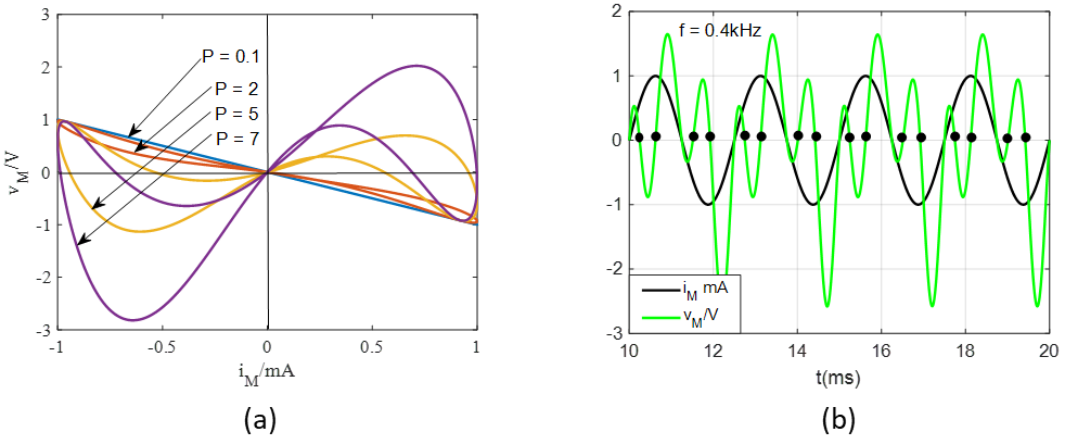

Fig 3. The relationship between current and voltage for the locally active memristor: (a) i-v pinched hysteresis loops varying with the parameter P with f = 0.5 kHz. (b) time-domain waveforms of the current and voltage with f = 0.4 kHz, used to indicate that voltage and current do not cross zero at the same time all the times (b)

This parameter brings in the contribution of the second state variable z. Substituting for y in the Ohm's law dependent equation, we obtain:

$$v_M = -\beta i_M \left[1 - \frac{P^2 i_M^2}{(\alpha + i_M + \Omega)^2}\right] \tag{7}$$

Using the parameters: $\alpha = 1, \beta = 1, P = 1, \Omega = 1$, a plot of the (I_m-V_M) is shown in Figure 4 with the locally active region highlighted. In Figure 4, we observe that the memristor system given by (5) is locally active. The same analysis method about locally-active memristor is used in[20] and[21]. We observe from the graphs that the parameters introduced by the second state variable greatly influence the characteristics. For, $\Omega = 0.0$ we obtain the Muthuswamy-Chua system and I-V characteristic as shown in[7]. To obtain the exact graph, we set $\alpha = 0.2$ and $\beta = 1.7$ to obtain Figure 4(a).

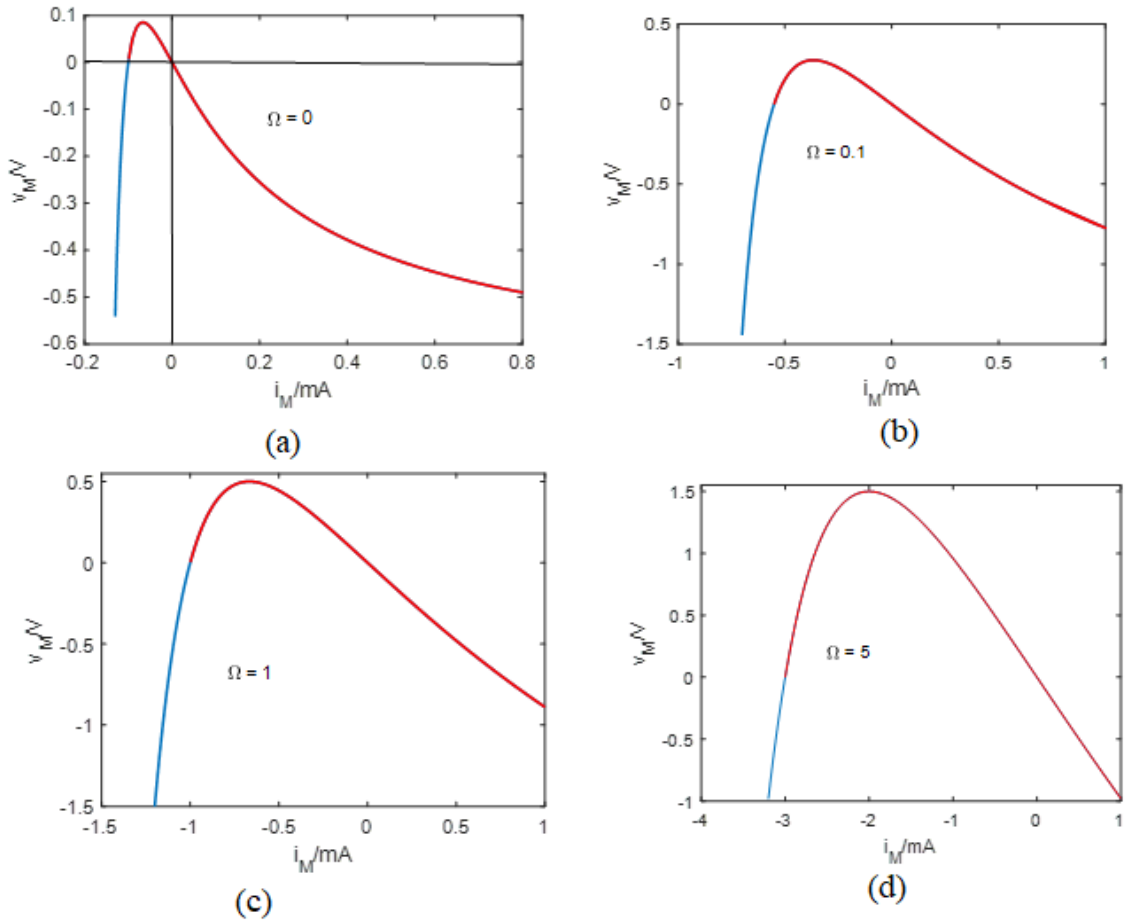

Fig 4. Plots of I-V DC curves for various values of W. Notice that most points lie in the locally-active region, marked in red

3 Novel Two Component Memristive Circuit

Based on the memristor model defined by the equations above and the poof of the characteristic signatures, a new chaotic oscillator is designed as shown in Figure 5. Its peculiarity is the fact that it has only two components: a memristor and an inductor. The inductor provides one state variable whereas the memristor provides two state variables. Hence we get a total of three state variables, the minimum required for chaotic behavior in a continuous time smooth dynamical system

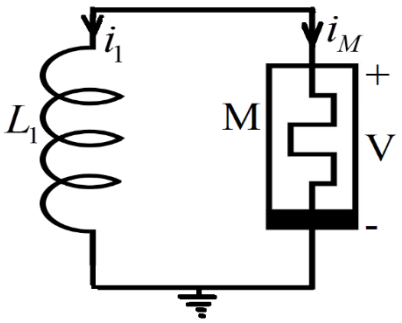

Fig 5. A two component memristor chaotic oscillator

3.1 Circuit equations and mathematical model

From Kirchhoff's circuit laws, the following differential equations are obtained which govern the dynamics of the circuit.

$$\left.\begin{aligned}\frac{di_{L1}}{dt} &= -\frac{1}{L_1}\left[\beta I_S\left(y^2-1\right)i_{L1}+R_S i_{L1}\right]\\ \frac{dy}{dt} &= -\frac{I_S i_{L1}}{R_b C_M}-\frac{y}{R_P C_M}+\frac{y I_S i_{L1}}{R_a C_M}-\frac{i_{L_M}}{C_M}\\ \frac{di_{L_M}}{dt} &= \frac{y}{L_M}-\frac{r i_{L_M}}{L_M}\end{aligned}\right\} \tag{8}$$

Using the same transformation and time scale as above, we obtain equation (9), where:

$$B=\frac{R_a I_S C_M}{L1}, R=\frac{R_S}{I_S}, K_1=\frac{R_a I_S C_M}{L2}, K_2=\frac{R_a I_S C_M}{L2} P=\frac{R_a}{R_b}, \alpha=\frac{R_a}{R_{po\,t}}, \beta=\frac{R_D}{R_6}, T=\frac{R_a}{I_S}.$$

We choose our parameters such that: $B=5;\beta=1;R=0.01;P=0.5;\alpha=1;\gamma=1$; $K_1=1;K_2=0.1$.

With these parameters and considering that R is small and does not affect the dynamics of the system, the system (8) becomes:

$$\left.\begin{aligned}\frac{dx}{d\tau} &= -B\beta\left(y^2-1\right)x\\ \frac{dy}{d\tau} &= -Px-\alpha y+xy-\gamma z\\ \frac{dz}{d\tau} &= K_1 y-K_2 z\end{aligned}\right\} \tag{9}$$

The system given by Eq. (9) has one cubic nonlinearity and one quadratic nonlinearity, making it a simple system in terms of mathematical representation. This complements the fact that our system is indeed the simplest chaotic system; just two circuit components and two nonlinear terms.

Using the initial conditions as: x(0) = y(0) = 0.1, z(0) = 0.0 and the parameters above, the Lyapunov exponents can be calculated as: $LE_1=0.165, LE_2\approx 0.00, LE_3=-0.258$, using the algorithm proposed by Wolf *et al.*[22] . This means that the system is chaotic. The phase portraits of system (9) are investigated by numerical simulation as shown in Figure 6. The projections of phase portrait on z-y, y-x planes are shown in Figure 6(a) and 6(b) respectively. From the numerical simulation results we know system (9) can generate a chaotic attractor.

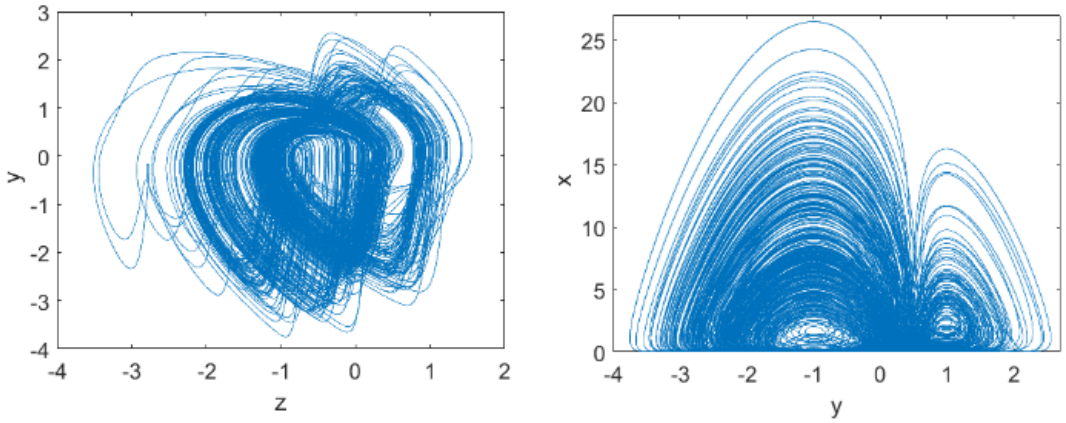

Fig 6. Chaotic phase portraits of system (9) (a) Projection on z–y plane, (b) Projection on y–x plane

3.2 Equilibrium Point and Stability

To analyze the system, a good start is to find its equilibrium(s), and then to characterize the local dynamical behaviours of the system orbits near these point(s). Distribution and local dynamical characteristics of the equilibrium(s) greatly influence the nonlinear dynamics of the system.

By solving the equilibrium system obtained from (9), we obtain 3 equilibrium points given by:

$$E_0 = (0,0,0)_2^T E_1 = (22,1,10)^T, S_2 = (22/3,-1,-10)^T$$

The Jacobian matrix of (9) with the parameters defined above is given by:

$$J = \begin{pmatrix} -5\left(y^2-1\right) & -10xy & 0 \\ y-0.5 & x-1 & -1 \\ 0 & 1 & -0.1 \end{pmatrix} \tag{10}$$

The stability of equilibrium point can be judged by the eigenvalues of the characteristic equation det (λI − J) = 0.

The eigenvalues of the first equilibrium point are given by: $\lambda_{1,2} - 0.55 \pm 0.89\text{i}, \lambda_3 = 0.00 + 0.00\text{i}$ .This shows that the equilibrium point E_0 is a stable point. For the second equilibrium point E_1 , the eigenvalues are given by: $\lambda_{1,2} = 10.50 \pm 0.86\text{i}, \quad \lambda_3 = -0.10$. The eigenvalues of the third equilibrium point E_2 are given by: $\lambda_{1,2} = 3.17 \pm 10.05\text{i}, \quad \lambda_3 = -0.10$. From the solutions of E_1 and E_2, there is one real eigenvalue and a pair of complex conjugate eigenvalues (a so- called index-2 saddle-focus), which are the criteria to generate a chaotic attractor [23,24].

4 FPGA Implementation of the Memristor Model and the Memristor-Based Chaotic Circuit

In this section, we are going to implement practically the proposed memristor model and the memristor-based chaotic circuit using an FPGA development board instead of an analogue discrete circuits as usually seen. The FPGA we are using is robust, portable and re-configurable. It overcomes weaknesses of analogue devices (sensitivity to temperature, sensitivity to initial conditions, etc.). With FPGA implementation, we have some flexibilities such as setting control parameters, frequency and initial conditions accurately, reducing the system to a portable source code, realizing complicated mathematical operations or algorithms. These benefits increase the number of realizable memristor chaotic circuits [25], making the implementation process simple and accurate. In this work, the forward Euler method is used because of its execution speed and its low cost in term of FPGA resources needed [26,27]. The proposed memristor model (5) and the memristor-based chaotic circuit (9) are discretised as follow:

$$\left.\begin{array}{l} y_{n+1} = y_n + h\left(P\left(I\sin\left(2\pi f_n\right)\right) - \alpha y_n\right. \\ \left. -y_n\left(I\sin\left(2\pi f_n\right)\right) - \gamma z_n\right) \\ z_{n+1} = z_n + h\left(K_1 y_n - K_2 z_n\right) \\ i_{n+1} = I\sin\left(2\pi f_n\right) \\ v_{n+1} = \beta\left(y_n^2 - 1\right)\left(I\sin\left(2\pi f_n\right)\right) \end{array}\right\} \tag{11}$$

$$\left.\begin{array}{l} x_{n+1} = x_n + h\left(-B\left(\beta y_n^2 - 1\right)x_n\right. \\ y_{n+1} = y_n + h\left(-Px_n - \alpha y_n + y_n z_n - \gamma z_n\right) \\ z_{n+1} = z_n + h\left(K_1 y_n - K_2 z_n\right) \end{array}\right\} \tag{12}$$

The step size of h = 0.001 is chosen to achieve a better accuracy. The calculations in the hardware implementation is done using 32 bits floating point of the IEEE754 standards then converted into signed-fixed. The outputs are adjusted to extract only the decimal part of the signed-fixed value then converted into standard logic vector. The dynamic state of each sample is sent out into GPIO ports of the FPGA board. The discrete state of the equations (5) and (9) are directly describe using VHDL into Quartus II 10.1 design software and the behavioral simulation is performed with ModelSim-Altera 6.5e. The Altera chip of type Cyclone IV E EP4CE115F29C7N is chosen as target device. The Register Transfer Level (RTL) schematic presented in Figure 7(a) is the top-model of the implemented the proposed memristor model and the RTL schematic presented in Figure 7(b) is the top-model of the implemented memristor-based chaotic circuit. Pin X and Y correspond to the current (i) and the voltage (V) respectively for Figure 7(a) and Pin X, Y and Z in Figure 7(b) correspond to current through the inductor L_1, the voltage across the capacitor C_M and the current through the inductor L_M respectively.

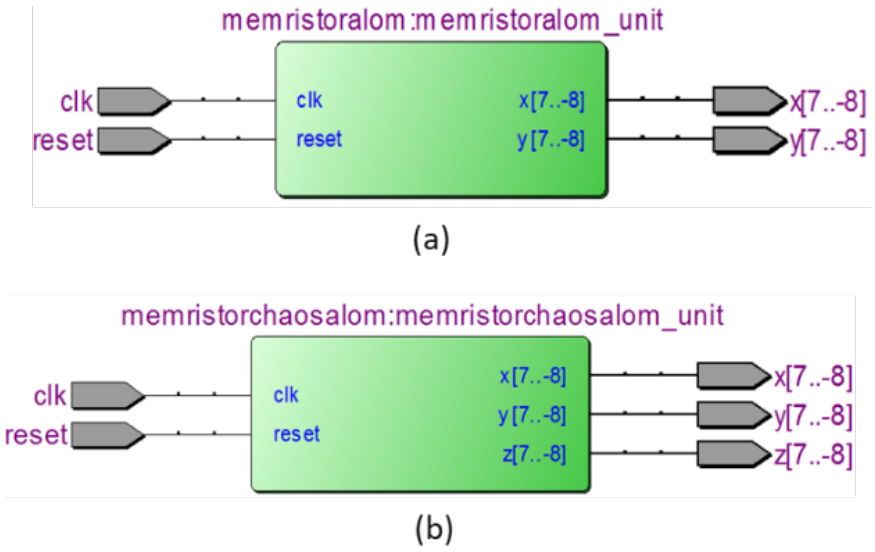

Fig 7. RTL view of: (a) the memristor model (b) the memristor-based chaotic circuit.

The practical demonstration of the feasibility of the proposed memristor model and the memristor-based chaotic circuit are done through the FPGA implementation shown by the set-up in Figure 9. Two R/2R resistors ladders are used as digital to analog converter (DAC) and connected to the GPIO ports of the DE2 115 FPGA board. The resulting analog signals are then filtered using capacitors. The filtered signals are then connected to the probe of oscilloscope to visualize the portraits.

Figure 8 shows the experimental relationship between current and voltage for the locally active memristor with varying frequency of excitation. We observe that the loops switching from the predominantly passive region (a, b, c) to the active region (d, e, f) for frequencies: f = 0.2 KHz, f = 0.3 KHz, f = 0.4 KHz, f = 0.85 KHz, f =1.0 KHz and f =10.0 KHz. We also notice that for increasing frequencies, the pinched hysteresis of the memristor shrinks to a single valued function for f = 10 kHz.

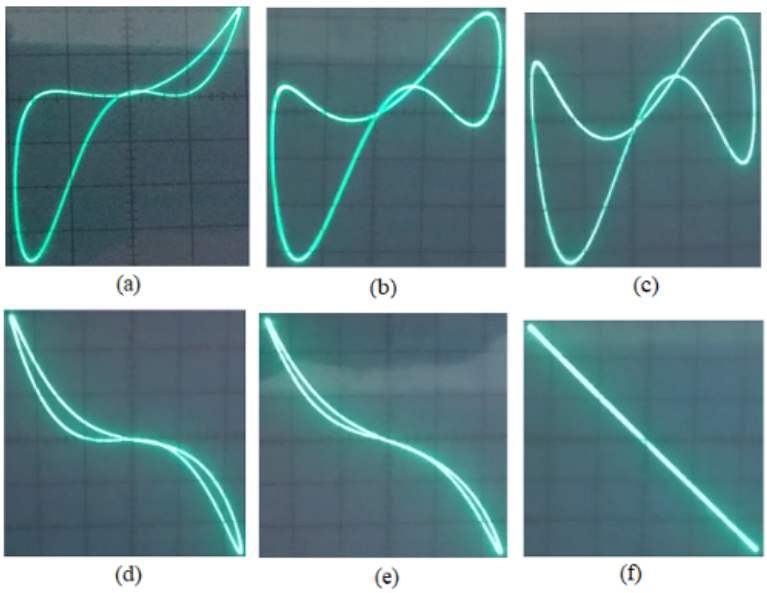

Fig 8. Experimental relationship between current and voltage for the locally active memristor. The loops switching from the predominantly passive region (a, b, c) to the active region (d, e, f) for frequencies: f = 0.2 KHz, f = 0.3 KHz, f = 0.4 KHz, f = 0.85 KHz, f =1.0 KHz and f =10.0 KHz.

With the initial conditions, and the parameters used in the numerical simulation, we obtain the experimental phase portraits of the two component chaotic system given by Eq. (9) as shown in Figure 9(a) and (b) representing the phase portraits in the z-y and y-x planes respectively. As one can see the results obtained from the practical implementation are in accordance with the numerical simulations.

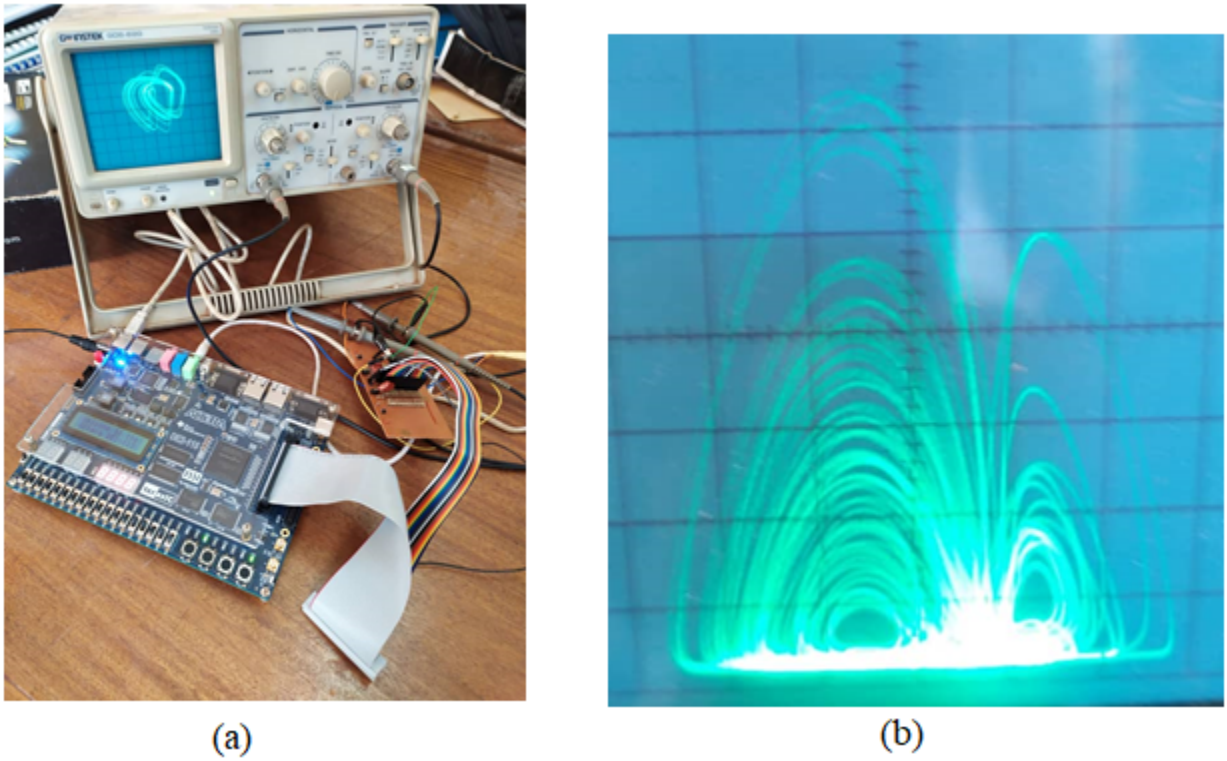

Fig 9. Experimental phase portraits of memristor chaos system derived from the FPGA board.

5 An Application in Steganography

In this section, we proposed a method to hide a secret message inside an image using steganography and chaos encryption. Steganography deals with the art of hiding information with an interesting property of hiding the mere existence of the secret information [28]. Steganography is different from cryptography. Cryptography is the practice of scrambling a message to an obscured form to prevent others from understanding it while steganography is the study of obscuring the message so that it cannot be seen [28]. To make a steganography we require generally two files: the cover/carrier file and the secret file. However, many multi-media carriers like, video, audio, image, text, etc. can act as a cover media to carry the secret message. Here, we used an image and test file as cover file and secret message respectively.

Many algorithms and procedures, such as Least Significant Bit (LSB), have been written to hide text in an image. [29,30] LSB method allows to start by passing both secret message and cover image into the encoder. The LSB embeds the secret message encrypted in the least significant bits of pixel values of the cover image. The secret message is encrypted using an affine cipher based on the two component memristor chaotic system. This with the support of the date of birth (DOB) keys. DOB key enables to construct a key using birth day, month, and year of the sender and receiver. The Figure 10 shows us the scheme of our proposed method.

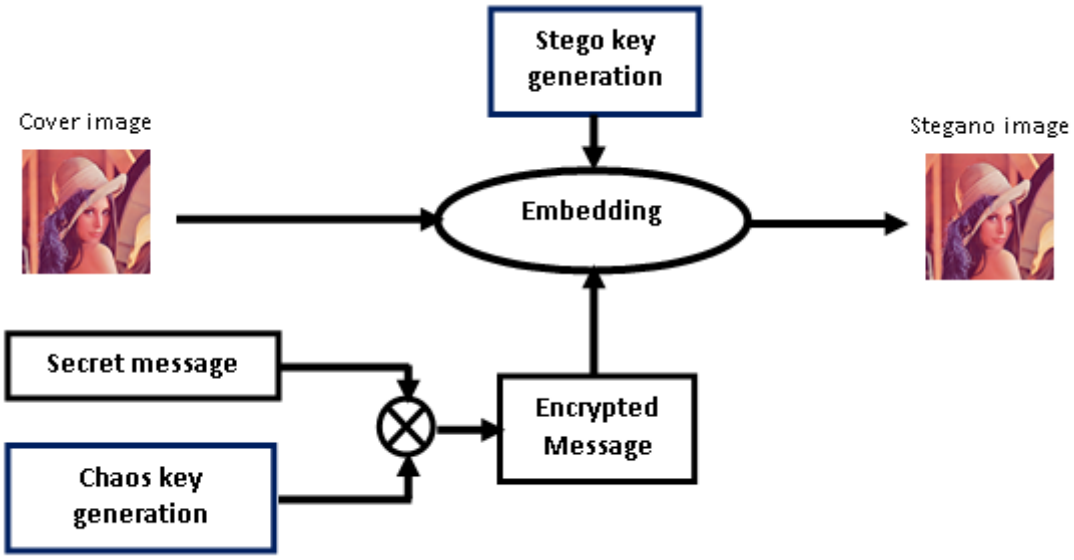

Fig 10. Cryptography and steganography system.

We performed an encryption of secret message before performing the steganography. This enables us to have a good integrity of our message between the sender and receiver.

5.1 Affine cipher based on the two component memristor chaotic system and DOB

Ciphers convert the message by a rule, known only to the sender and the recipient, which change each individual letter[31]. An affine cipher is a combination of shift and multiplication cipher. E(.) and D(.) respectively the encryption and decryption processes of affine cipher. For given message M, E(.) and D(.) are defined in[31] by:

$$\left.\begin{aligned} E(M) &= (l^{*}M) + m(\bmod p) \\ D(M) &= l' * (E(M) - m)(\bmod p) \end{aligned}\right\} \tag{14}$$

Where: l, m are parameters of affine cypher key $k(k = (1,m))$, l' is inverse of 1 modulo p and p is a positive integer.

To apply this method, we considered S and R in our cryptosystem as sender and receiver respectively. To exchange a secret message, S and R have to generate their own secret key pair using DOB and fractional order hyper chaotic system. Consider $D = (D_1, D_2)$ be a pair of DOB of S and R who only shared between them. D_1 is DOB of S and D_2 is the DOB of R. They can be represented as: $D_1 = DD - MM - YY$ and $D_2 = dd - mm - yy$ in the day, month, and year format.

a) S and R key generation

- S solves the two-component system (10) at time t and generates the equations (15) and (16)

$$\left.\begin{aligned} K_1 &= DD^{*}x(t) + MM^{*}y(t) \\ &+ YY^{*}z(t)(\bmod p) \end{aligned}\right\} \tag{15}$$

$$\left.\begin{aligned} K_2 &= dd^{*}x(t) + mm^{*}y(t) \\ &+ yy^{*}z(t)(\bmod p) \end{aligned}\right\} \tag{16}$$

- R solves the two-component system (10) at time t and generates the Eq. (17) and (18)

$$\left.\begin{aligned} L_1 &= DD^{*}x(t) + MM^{*}y(t) \\ &+ YY^{*}z(t)(\bmod p) \end{aligned}\right\} \tag{17}$$

$$\left.\begin{array}{l} L_2 = dd^* x(t) + mm^* y(t) \\ +yy^* z(t) (\bmod p) \end{array}\right\} \tag{18}$$

b) Encryption and decryption message

- S wants to send a message M to R secretly, he encrypts M using E(.) function given by (19)

$$E(M) = (M^* K_1) + K_2 (\bmod p) \tag{19}$$

- When R receives a message from S and want to recover an original message M, he uses the D(.) function given by

$$D(M) = (E(M) - L_2)^* L_1^{-1} (\bmod p) \tag{20}$$

5.2 Steganograph

In this part, we have to hide the secret message into cover image using LSB method. Consider a color image that we decomposed into 3 sub images component (Red, Green and Blue). Each pixel of components assumes a value between [0, 255] and represented with 8 bit. We replaced the least significant bit of some pixels of components by each bits of the secret message. This technic performed with good efficient because does not affect human perception.

5.3 Example

The Figure 11 shows us our 2 covers image with size [512 x 512] that we used to hide the secret message encrypted. Let choice the secret message as $M = MY$ NAME IS STEGANO . When we apply the encryption E(.) to the message M with the following parameters: $DD = 10, MM = 06$, $YY = 1990; dd = 22, MM = 10, YY = 1984; p = 128; x = 10.093, y = 0.63, z = 5.63$, we obtain the encrypted message M'=□□□□"□□□□. The stegano images are represented in Figure 12.

Fig 11. Cover Images

In the Figures 11 and 12, we observe that the cover image have the same visual aspect with stegano image. To see the difference between these images, we have to calculate the Peak Signal to Noise ratio (PSNR). The PSNR is given in [27] by Eq. (21).

$$PSNR = 10 \log_{10} \left(\frac{255^2}{MSE} \right) \tag{21}$$

The PSNR gives an objective measure of the distortion introduced by the steganography. It is expressed in decibels (dB). We found the PSNR of Lena and Flower equal to 59.28 dB and 62.01 dB respectively. Table 1 and Figure 13 show the evolution of PSNR between cover and stegano images according to length of secret message.

Fig 12. Stegano images

Table 1 and Figure 13 show how the message length influence the quality of stegano image. We see that the proposed steganography algorithm achieves high stegano image quality.

In fact, in Table 1 and Figure 13, we observe the PSNR is maximum when the message length equal to 150. And when M increase, PSNR decrease slowly. This proof that our chaotic cipher adapted very well the LSB method for steganography.

The experimental results reveal the practicability and superiority of the presented technique.

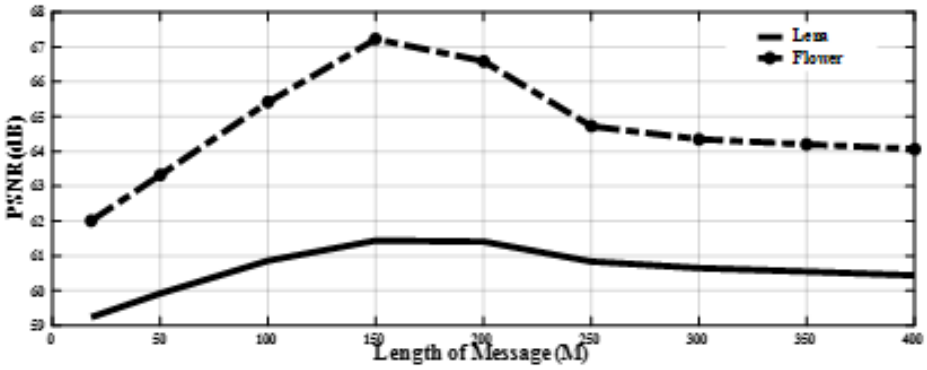

Fig 13. PSNR of Lena and Flower according to length of M

Table 1. PSNR between cover and stegano images according to length of M

Length of M	18	50	100	150	200	250	300	350	400
Parameter		PSNR (dB)							
Lena	59.28	59.92	60.79	61.36	61.31	60.72	60.53	60.43	60.29
Flower	62.01	63.32	65.41	67.22	66.59	64.72	64.35	64.20	64.07

6 Conclusion

In this work, we proposed a locally active memristor based on a current-controlled generic memristor that has two internal state variables- a second order memristor. The characteristics of the memristor were investigated and the memristor shown to have a wide locally active region and for properly chosen parameter values, the memristor can be switched from a predominantly passive region to an active region. A memristor-based chaotic circuit is presented, which only consists of a memristor and an inductor. The memristor model and the two component circuit were numerically simulated using MATLAB and the hardware implementation was designed using FPGA. The experimental results are consistent with the software simulations. For exploring

the potential application of the circuit, an application in steganography is included where we proposed a method to hide a secrete message inside an image using steganography and chaos encryption.

Acknowledgement

The authors received no financial support for the research, authorship, and/or publication of this article.

References

1) Ying J, Wang G, Dong Y, Yu S. Switching Characteristics of a Locally-Active Memristor with Binary Memories. *International Journal of Bifurcation and Chaos*. 2019;29(11). Available from: 10.1142/S0218127419300301.
2) Wang S, Wang X, Zhou Y, Han B. A memristor-based hyperchaotic complex Lü system and its adaptive complex generalized synchronization. *Entropy*. 2016;18(2):58. Available from: 10.3390/e18020058.
3) Luo J, Xu X, Ding Y, Yuan Y, Yang B, Sun K, et al. Application of a memristor-based oscillator to weak signal detection. *The European Physical Journal Plus*. 2018;133(6). Available from: 10.1140/epjp/i2018-12041-y.
4) Alharbi AG, Fouda ME, Khalifa ZJ, Chowdhury MH. Electrical Nonlinearity Emulation Technique for Current-Controlled Memristive Devices. *IEEE Access*. 2017;5:5399–5409. Available from: https://dx.doi.org/10.1109/access.2017.2695402.
5) Zhao Q, Wang C, Zhang X. A universal emulator for memristor, memcapacitor, and meminductor and its chaotic circuit. *Chaos: An Interdisciplinary Journal of Nonlinear Science*. 2019;29(013141). Available from: 10.1063/1.5081076.
6) Tolba MF, Fouda ME, Hezayyin HG, Madian AH, Radwan AG. Memristor FPGA IP Core Implementation for Analog and Digital Applications. *IEEE Transactions on Circuits and Systems II: Express Briefs*. 2019;66(8):1381–1385. Available from: https://dx.doi.org/10.1109/tcsii.2018.2882496.
7) Muthuswamy B, Chua LO. Simplest Chaotic Circuit. *International Journal of Bifurcation and Chaos*. 2010;20(05):1567–1580. Available from: https://dx.doi.org/10.1142/s0218127410027076.
8) Tchitnga R, Fotsin HB, Nana B, Fotso PHL, Woafo P. Hartley's oscillator: The simplest chaotic two-component circuit. *Chaos, Solitons & Fractals*. 2012;45(3):306–313. Available from: https://dx.doi.org/10.1016/j.chaos.2011.12.017.
9) Xu B, Wang G, Shen Y. A simple meminductor-based chaotic system with complicated dynamics. *Nonlinear Dynamics*. 2017;88(3):2071–2089. Available from: 10.1007/s11071-017-3363-y.
10) Talla FC, Tchitnga R, Kengne R, Nana B, Fomethe A. Didactic model of a simple driven microwave resonant T-L circuit for chaos, multistability and antimonotonicity. *Heliyon*. 2019;5(10):e02715. Available from: https://dx.doi.org/10.1016/j.heliyon.2019.e02715.
11) Deng Y, Li Y. A memristive conservative chaotic circuit consisting of a memristor and a capacitor. *Chaos: An Interdisciplinary Journal of Nonlinear Science*. 2020;30(1):013120. Available from: 10.1063/1.5128384.
12) Negou AN, Kengne J. Dynamic analysis of a unique jerk system with a smoothly adjustable symmetry and nonlinearity: Reversals of period doubling, offset boosting and coexisting bifurcations. *AEU - International Journal of Electronics and Communications*. 2018;90:1–19. Available from: https://dx.doi.org/10.1016/j.aeue.2018.04.003.
13) Yuan F, Li Y. A chaotic circuit constructed by a memristor, a memcapacitor and a meminductor. *Chaos: An Interdisciplinary Journal of Nonlinear Science*. 2019;29(10):101101. Available from: 10.1063/1.5125673.
14) Chua L. Everything You Wish to Know About Memristors but Are Afraid to Ask. In: In Handbook of Memristor Networks. Springer. 2019;p. 89–157. Available from: 10.1007/978-3-319-76375-0_3.
15) Khalid M. Review on various memristor models, characteristics, potential applications, and future works. *Transactions on Electrical and Electronic Materials*. 2019;20(4):289–298. Available from: 10.1007/s42341-019-00116-8.
16) Chua LO, Kang SM. Memristive devices and systems. *Proceedings of the IEEE*. 1976;64(2):209–223. Available from: https://dx.doi.org/10.1109/proc.1976.10092.
17) Sah MP, Yang C, Kim H, Muthuswamy B, Jevtic J, Chua L. A Generic Model of Memristors With Parasitic Components. *IEEE Transactions on Circuits and Systems I: Regular Papers*. 2015;62(3):891–898. Available from: https://dx.doi.org/10.1109/tcsi.2014.2373674.
18) Adhikari SP, Sah MP, Kim H, Chua LO. Three Fingerprints of Memristor. *IEEE Transactions on Circuits and Systems I: Regular Papers*. 2013;60(11):3008–3021. Available from: https://dx.doi.org/10.1109/tcsi.2013.2256171.
19) Chua L. If it's pinched it'sa memristor. *Semiconductor Science and Technology*. 2014;29(10):104001. Available from: 10.1088/0268-1242/29/10/104001.
20) Ascoli A, Tetzlaff R, Chua LO. The first ever real bistable memristor. *IEEE International Symposium on Circuits and Systems (ISCAS)*. 2016;p. 2896–2896. Available from: 10.1109/ISCAS.2016.7539199.
21) Ascoli A, Slesazeck S, Mahne H, Tetzlaff R, Mikolajick T. Nonlinear Dynamics of a Locally-Active Memristor. *IEEE Transactions on Circuits and Systems I: Regular Papers*. 2015;62(4):1165–1174. Available from: https://dx.doi.org/10.1109/tcsi.2015.2413152.
22) Wolf A, Swift JB, Swinney HL, Vastano JA. Determining Lyapunov exponents from a time series. *Physica D: Nonlinear Phenomena*. 1985;16(3):285–317. Available from: https://dx.doi.org/10.1016/0167-2789(85)90011-9.
23) Negou AN, kengne J, Tchiotsop D. Periodicity, chaos and multiple coexisting attractors in a generalized Moore–Spiegel system. *Chaos, Solitons & Fractals*. 2018;107:275–289. Available from: https://dx.doi.org/10.1016/j.chaos.2018.01.011.
24) Talla FC, Tchitnga R, Fotso PHL, Kengne R, Nana B, Fomethe A. Unexpected Behaviors in a Single Mesh Josephson Junction Based Self-Reproducing Autonomous System. *International Journal of Bifurcation and Chaos*. 2020;30(07):2050097. Available from: https://dx.doi.org/10.1142/s0218127420500972.
25) Ding Q, Pang J, Fang J, Peng X. Designing of chaotic system output sequence circuit based on FPGA and its applications in network encryption card. *International Journal of Innovative Computing, Information and Control*. 2007;3(2):449–456.
26) Sadoudi S, Azzaz MS, Djeddou M, Benssalah M. An FPGA real-time implementation of the Chen's chaotic system for securing chaotic communications. *International Journal of Nonlinear Science*. 2009;7(4):467–474.
27) Rajagopal K, Guessas L, Karthikeyan A, Srinivasan A, Adam G. Fractional Order Memristor No Equilibrium Chaotic System with Its Adaptive Sliding Mode Synchronization and Genetically Optimized Fractional Order PID Synchronization. *Complexity*. 2017;2017:1–19. Available from: https://dx.doi.org/10.1155/2017/1892618.

28) Wu MY, Ho YK, Lee JH. An iterative method of palette-based image steganography. *Pattern Recognition Letters*. 2004;25(3):301–310. Available from: 10.1016/j.patrec.2003.10.013.
29) Shehab G, Jawar ZM, Khami M. Matlab Coding For Text Steganography System By Using LSB Insertion Method With Key. *Basrah journal of science*. 2015;33(2):37–51. Available from: https://iraqjournals.com/article_105863_4adf62c29da0c6082e9e167ada785ad5.pdf.
30) Cheddad A, Condell J, Curran K, Kevitt PM. Digital image steganography: Survey and analysis of current methods. *Signal Processing*. 2010;90:727–752. Available from: https://dx.doi.org/10.1016/j.sigpro.2009.08.010.
31) Muthukumar P, Balasubramaniam P, Ratnavelu K. Fast projective synchronization of fractional order chaotic and reverse chaotic systems with its application to an affine cipher using date of birth (DOB). *Nonlinear Dynamics*. 2015;80(4):1883–1897. Available from: https://dx.doi.org/10.1007/s11071-014-1583-y.