

Discrete Memristor and Discrete Memristive Systems

Shaobo He ¹, Donglin Zhan ¹, Huihai Wang ^{1,*}, Kehui Sun ¹ and Yuexi Peng ²

¹ School of Physics and Electronics, Central South University, Changsha 410083, China; heshabo@csu.edu.cn (S.H.); zdl_csu540@163.com (D.Z.); kehui@csu.edu.cn (K.S.)

² School of Cyberspace Science, Xiangtan University, Xiangtan 411105, China; pyx244896301@163.com

* Correspondence: wanghuihai_csu@csu.edu.cn

Abstract: In this paper, we investigate the mathematical models of discrete memristors based on Caputo fractional difference and G–L fractional difference. Specifically, the integer-order discrete memristor is a special model of those two cases. The “∞”-type hysteresis loop curves are observed when input is the bipolar periodic signal. Meanwhile, numerical analysis results show that the area of hysteresis decreases with the increase of frequency of input signal and the decrease of derivative order. Moreover, the memory effect, characteristics and physical realization of the discrete memristors are discussed, and a discrete memristor with short memory effects is designed. Furthermore, discrete memristive systems are designed by introducing the fractional-order discrete memristor and integer-order discrete memristor to the Sine map. Chaos is found in the systems, and complexity of the systems is controlled by the parameter of the memristor. Finally, FPGA digital circuit implementation is carried out for the integer-order and fractional-order discrete memristor and discrete memristive systems, which shows the potential application value of the discrete memristor in the engineering application field.

Keywords: memristor; discrete modeling of memristor; fractional-order difference; digital circuits; memristive system



Citation: He, S.; Zhan, D.; Wang, H.; Sun, K.; Peng, Y. Discrete Memristor and Discrete Memristive Systems. *Entropy* **2022**, *24*, 786. <https://doi.org/10.3390/e24060786>

Academic Editor: Reik Donner

Received: 25 March 2022

Accepted: 29 May 2022

Published: 3 June 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

In 1971, Chua [1] suggested there should be a fourth electron component in addition to resistance, capacitance and inductance. This new component was proposed to be called a memristor. In 2008, researchers at Hewlett Packard Labs reported that the memristor could be realized using nanoscale material [2]. Since this, there has been explosive growth in the research into memristors. It should be noted that the concept of the generalized memristor and the three fingerprint characteristics [3] of memristors have been reported, which provides a guide for the design of memristors. Owing to their properties, including memory and intrinsic nonlinearity, memristors have many potential applications in fields such as flash memory [4], reconfigurable computing [5], logic circuits [6], synapses [7], neural networks [8,9], solving mazes [10] and oscillation circuits [11]. Among those, memristor-based nonlinear systems or circuits have rich dynamics such as multistability and chaos [12,13].

The physical implementation of the memristor is important for real applications of memristors in the future. However, to date, there have been almost no memristor devices on sale, in comparison with resistance, capacitance and inductance ones. At present, researchers of material science are trying to realize memristors based on different kinds of nanometer materials, such as ZnO [14], TaOx [15], ZnO-rGO [16], VO₂ [17] and CuSO₄ [18]. A nanoscale memristor device can be applied in the neuromorphic systems as the synapse [19]. It should be noted that those nanoscale memristor devices satisfy the definition of a memristor, but do not have good mathematical models for theoretical analysis. In fact, researchers have used alternative techniques to design memristor devices based on mathematical models. Analog circuit implementation of memristors is

important, and there are many different implementations [20–23]. Meanwhile, FPGA implementation has aroused much interest among researchers due to its easily programmable, reconfigurable, controllable, precise and better performance [24,25]. To improve the theoretical framework and applications of memristors, it is necessary to investigate functional memristor emulators and their intrinsic features.

There are two main approaches to building mathematical models for real systems. One is to build continuous models, and the other is to establish discrete models. Differential calculus is used in continuous models, and difference calculus is used in discrete models. As a result, some practical methods are proposed based on the difference [26–29]. For instance, in digital image processing [26], image enhancement operators can be obtained using difference and fractional-order difference. Meanwhile, since we have continuous memristors, discrete memristors should also exist. In fact, the concept of discrete memristors was first mentioned by Wang et al. [30], but the proposed model is not satisfying. In 2020, He et al. [31] proposed a mathematical model for a discrete memristor based on difference. Later, they introduced a discrete memristor to a Hénon map [32] and a higher-dimensional chaotic map [33]. Meanwhile, Bao et al. [34–37] designed discrete memristor chaotic maps and investigated their multistability. Moreover, Simulink simulation of a discrete memristor has been carried out [38]. To date, most of the work has been carried out based on integer-order discrete memristors. To build mathematical models and explain the physical significance of discrete memristors, some issues should be noted:

- The mathematical model for integer-order discrete memristors is clear, but the physical significance of discrete memristors still needs to be explored.
- Since there are different kinds of fractional-order difference, it is necessary to build fractional-order discrete memristors using different differences. At present, the two widely used fractional-order differences are Caputo-like difference [39] and G–L difference [40].
- The characteristics of discrete memristors, such as the memory effect and frequency domain, should be discussed.
- Physical implementation of the designed discrete memristors should be carried out and discussed.

Figure 1 shows a short history of the study of memristors and the position of this paper. It was started by Chua in 1971, when the model for the generalized memristor and the three fingerprints for memristors were proposed. This indicated the rules for designing memristors. To date, the design of memristors has been investigated by scientists from different research fields, such as mathematical modeling, circuit implementation, nanoscale materials, and applications. Therefore, what is the role of discrete memristors in the memristor family? In our opinion, the research of discrete memristors and discrete memristive systems in this paper follows the dashed lines in Figure 1. We use Caputo difference and G–L difference to build the mathematical models of fractional-order discrete memristors and realize them in digital circuits. This shows that a discrete memristor can also be a device for real applications.

The outline of this paper is given as follows. In Section 2, definitions of fractional-order differences and generalized memristors are presented. In Section 3, models of fractional-order discrete memristors are proposed, numerical simulations are carried out, and the short-term memory effect is discussed. In Section 4, physical significance of the discrete memristor is discussed. In Section 5, two kinds discrete memristive systems are designed, and their dynamics are analyzed. In Section 6, FPGA implementations of discrete memristive systems are carried out. Finally, we summarize the whole analysis.

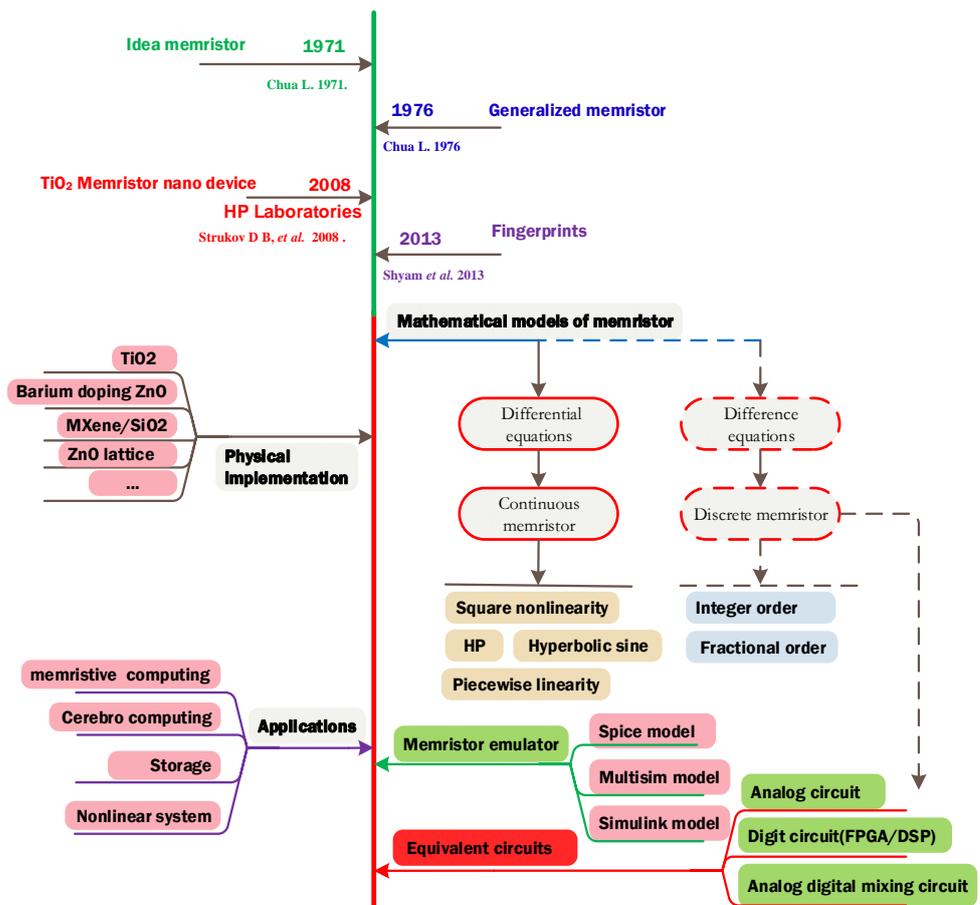


Figure 1. A short history of research about memristor and the position of this paper [1–3,41].

2. Preliminaries

2.1. Fractional-Order Differences

In this section, two different fractional-order differences are presented.

Definition 1 ([39,42]). For a given fractional-order $\alpha > 0, \alpha \notin \mathbb{N}$, when $u(t)$ is defined in \mathbb{N}_{t_0} , its Caputo difference is defined by

$$\begin{aligned} {}^C\Delta_{t_0}^\alpha u(t) &:= \Delta_{t_0}^{-(m-\alpha)} \Delta^m u(t) \\ &= \frac{1}{\Gamma(m-\alpha)} \sum_{s=t_0}^{t-(m-\alpha)} (t-\sigma(s))^{(m-\alpha-1)} \Delta^m u(s) \end{aligned} \tag{1}$$

where $\Gamma(\cdot)$ is the gamma function, $t \in \mathbb{N}_{t_0+m-\alpha}$, $\mathbb{N}_{t_0} := \{t_0, t_0 + 1, t_0 + 2, \dots\}$, $m = \lceil \alpha \rceil$, and $\sigma(s)$ denotes the next point in the time scale after s , namely $\sigma(s) = s + 1$ for $s \in \mathbb{N}_{t_0}$.

Obviously, when $\alpha = 1$, the fractional difference becomes $\Delta u(t_n) = u(t_{n+1}) - u(t_n)$.

Theorem 1 ([39,42]). For the Caputo-like fractional-order system

$${}^C\Delta_{t_0}^\alpha u(t) = g(t + \alpha - 1, u(t + \alpha - 1)), \tag{2}$$

its solution is given by

$$\begin{aligned} u(t) &= \sum_{k=0}^{m-1} \frac{(t-t_0)^{(k)}}{k!} \Delta^k u(t_0) \\ &+ \frac{1}{\Gamma(\alpha)} \sum_{s=t_0+m-\alpha}^{t-\alpha} \frac{g(s+\alpha-1, u(s+\alpha-1))}{(t-\sigma(s))^{1-\alpha}} \end{aligned} \tag{3}$$

where $u(t)$ is the system variable, $g(\cdot)$ is the system equation and $\Delta^k u(t_0) = u_k, k = 0, 1, \dots, m-1, m = \lceil \alpha \rceil$.

Definition 2 ([40]). The G–L fractional-order difference is defined by

$${}^G\Delta_{t_0}^\alpha g(t_n) = \sum_{j=0}^n (-1)^j \binom{\alpha}{j} g(t_{n-j}), \tag{4}$$

where α is the fractional difference order, and $\binom{\alpha}{j}$ is denoted as

$$\binom{\alpha}{j} = \frac{\Gamma(\alpha + 1)}{\Gamma(j + 1)\Gamma(\alpha - j + 1)}, \tag{5}$$

and $\binom{\alpha}{0} = 1$.

When $\alpha = 1$, we have

$${}^G\Delta_{t_0}^\alpha x(t_n) = x(t_n) - x(t_{n-1}) . \tag{6}$$

It can be seen that is a forward-difference operator.

Definition 3 ([43]). For a given G–L fractional-order discrete system with initial condition $x(t_0)$ defined by

$${}^G\Delta_{t_0}^\alpha x(t_n) = g(x(t_{n-1}), t_{n-1}), \tag{7}$$

since

$$\begin{aligned} {}^G\Delta_{t_0}^\alpha x(t_n) &= \sum_{j=0}^n (-1)^j \binom{\alpha}{j} x(t_{n-j}) \\ &= x(t_n) + (-1)^1 \binom{\alpha}{1} x(t_{n-1}) \\ &\quad + (-1)^2 \binom{\alpha}{2} x(t_{n-2}) + \dots + (-1)^n \binom{\alpha}{n} x(t_0) , \\ &= x(t_n) + \sum_{j=1}^n G_j^{(\alpha)} x(t_{n-j}) \end{aligned} \tag{8}$$

the solution of this system is denoted as

$$x(t_n) = g(x(t_{n-1}), t_{n-1}) - \sum_{j=1}^n G_j^{(\alpha)} x(t_{n-j-1}), \tag{9}$$

By formula (5)

$$G_j^\alpha = (-1)^j \frac{\Gamma(\alpha + 1)}{\Gamma(j + 1)\Gamma(\alpha - j + 1)} \tag{10}$$

and $G_0^\alpha = 1$.

In Definition 3, $G_j^{(\alpha)}$ is the coefficient of the history data, with $G_j^{(\alpha)} = 0 (j = -2, -1)$ and $G_j^{(\alpha)} = (-1)^j \binom{\alpha}{j}, (j = 0, 1, 2, \dots)$.

2.2. The Generalized Memristor

The definition of the generalized memristor is presented in Definition 3. It shows the relationship between the input $i(t)$ and output $y(t)$. For example, if $i(t)$ and $y(t)$ correspond

to current and voltage of two circuit variables, respectively, it defines a current-controlled memristive system or a charge-controlled memristor.

Definition 4 ([41]). *The generalized continuous memristor is defined by*

$$\begin{cases} y(t) = g(x(t), i(t), t)i(t) \\ \dot{x}(t) = f(x(t), i(t), t) \end{cases} \quad (11)$$

where $x(t)$ is the internal state variable of the memristor, $g(\cdot)$ is a function that represents the value of the memristor, and $f(\cdot)$ is a continuous function.

Definition 5. *describes the case of a continuous memristor. Here, we give a definition for the generalized discrete memristor.*

Definition 6. *The generalized fractional discrete memristor is defined as*

$$\begin{cases} y(t_n) = g(x_n, i(t_n), t_n)i(t_n) \\ \Delta_{t_0}^\alpha x(t_n) = f(x(t_{n-1}), i(t_{n-1}), t_{n-1}) \end{cases} \quad (12)$$

where $g(\cdot)$ is a function that represents the value of the discrete memristor, and $f(\cdot)$ is a continuous function. As with the continuous memristor, the generalized discrete memristor also describes the relationship between the input signal and the output signal.

The input and output signal are not limited by the voltage and current, but can be any discrete signals.

According to Definition 4, a universal fractional-order discrete memristor is defined by

$$\begin{cases} y(t_n) = g(x(t_n))i(t_n) \\ \Delta_{t_0}^\alpha x(t_n) = f(x(t_{n-1}), i(t_{n-1}), t_{n-1}) \end{cases} \quad (13)$$

An ideal fractional-order discrete universal memristor is defined by

$$\begin{cases} y(t_n) = g(x(t_n))i(t_n) \\ \Delta_{t_0}^\alpha x(t_n) = f(x(t_{n-1}))i(t_{n-1}) \end{cases} \quad (14)$$

In this paper, the ideal fractional-order discrete memristor is defined as

$$\begin{cases} y(t_n) = g(x(t_n))i(t_n) \\ \Delta_{t_0}^\alpha x(t_n) = f(i(t_{n-1})) \end{cases} \quad (15)$$

Here, $g(\cdot)$ is defined in two cases:

Case 1: $g(x) = a_1 + b_1x$.

Case 2: $g(x) = a_2 + b_2|x|$.

In this paper, we fix $a_1 = 0.8$, $b_1 = -0.0001$, $a_2 = 0.667$, $b_2 = 1.4828 \times 10^{-4}$, and the periodic input signals are defined as

$$i(t_n) = A \sin(\omega t_n), \quad (16)$$

where $t_n = 0, 1, 2, \dots$.

Theorem 2 ([3]). *The three fingerprints determine whether a device is a memristor or not. The fingerprints are:*

- (1) *When it is driven by a bipolar periodic signal, the device must exhibit a “pinched hysteresis loop” in the voltage–current plane, assuming the response is periodic.*
- (2) *Starting from some critical frequency, the hysteresis loop area should decrease monotonically as excitation frequency increases.*

- (3) The pinched hysteresis loop should shrink to a single-valued function when the frequency tends to infinity.

For the discrete memristor, it should have the three fingerprints when the input of the discrete memristor contains bipolar periodic signals. For the function $f(\cdot)$, there are many different choices, including piecewise nonlinearity, linear nonlinearity, absolute value, linear function, sine function, and square function [44]. In this manuscript, we use the idea of memristor models including HP memristors (linear function) and absolute function, to show the effectiveness of the proposed methods. However, for function $g(\cdot)$, this does not offer much choice, and they are $\dot{x}(t) = ki(t)$ and $\Delta x(t_n) = ki(t_n)$ for the continuous case and discrete case, respectively, where k is strength. In this paper, we also use three fingerprints to verify the rationality of the designed discrete memristors.

3. Models of Discrete Memristor

3.1. Fractional-Order G–L Difference-Based Model

Based on the G–L definition, the fractional-order discrete memristor is defined by

$$\begin{cases} \psi(t_n) = F(\varphi, i(t_n), t_n)i(t_n) \\ {}^G\Delta_{t_0}^\alpha \varphi(t_n) = f(\varphi(t_{n-1}), i(t_{n-1}), t_{n-1}) \end{cases} \quad (17)$$

According to Definition 4, the internal valuable in the G–L fractional-order discrete memristor is defined by

$$\varphi(t_n) = f(\varphi(t_{n-1}), i(t_{n-1}), t_{n-1}) - \sum_{j=1}^n G_j^{(\alpha)} \varphi(t_{n-j}). \quad (18)$$

As a result, this discrete memristor can be written as

$$\begin{cases} \psi(t_n) = F(\varphi(t_n), i(t_n), t_n)i(t_n) \\ \varphi(t_n) = f(\varphi(t_{n-1}), i(t_{n-1}), t_{n-1}) \\ \quad - \sum_{j=1}^n G_j^{(\alpha)} \varphi(t_{n-1}) \end{cases} \quad (19)$$

To simulate the system, we need to calculate coefficient $G_j^{(\alpha)}$ in a more effective way. First, we have the following equation

$$\Gamma(\alpha + 1) = \alpha\Gamma(\alpha). \quad (20)$$

According to the definition of the coefficient, $j = (1, 2, 3, \dots)$, we have

$$G_{j-1}^{(\alpha)} = (-1)^{j-1} \frac{\Gamma(\alpha + 1)}{\Gamma(j)\Gamma(\alpha - j + 2)}. \quad (21)$$

Then,

$$\begin{aligned} G_j^{(\alpha)} &= (-1)^j \frac{\Gamma(\alpha+1)}{\Gamma(j+1)\Gamma(\alpha-j+1)} \\ &= -(-1)^{j-1} \frac{\Gamma(\alpha+1)}{\Gamma(j)\Gamma(\alpha-j+2)} \cdot \frac{\Gamma(j)\Gamma(\alpha-j+2)}{\Gamma(j+1)\Gamma(\alpha-j+1)} \\ &= -G_{j-1}^{(\alpha)} \frac{\Gamma(j)\Gamma(\alpha-j+2)}{\Gamma(j+1)\Gamma(\alpha-j+1)} \\ &= -G_{j-1}^{(\alpha)} \frac{\Gamma(j)(\alpha-j+1)\Gamma(\alpha-j+1)}{j\Gamma(j)\Gamma(\alpha-j+1)} \\ &= -\frac{\alpha-j+1}{j} G_{j-1}^{(\alpha)} = \left(1 - \frac{\alpha+1}{j}\right) G_{j-1}^{(\alpha)} \end{aligned} \quad (22)$$

When $j = 0$, $G_0^{(\alpha)} = 1$. Thus, we can obtain the values of $G_j^{(\alpha)}$ using the relationship as given in Equation (22).

An example of the G–L fractional-order discrete memristor is given by

$$\begin{cases} y(t_n) = g(x(t_n))i(t_n) \\ {}^G \Delta_{t_0}^\alpha x(t_n) = ki(t_{n-1}) \end{cases} \quad (23)$$

Thus, the model is denoted as

$$\begin{cases} y(t_n) = g(x(t_n))i(t_n) \\ x(t_n) = i(t_{n-1}) - \sum_{j=1}^n G_j^{(\alpha)} x(t_{n-j}) \end{cases} \quad (24)$$

The hysteresis loop of the G–L difference-based discrete memristor with different functions $g(\cdot)$, order α and frequency ω is illustrated in Figure 2, where Figure 2a,b are plotted using $A = 4, \omega = 0.001$ and different order α and Figure 2c,d are plotted using $A = 4, \alpha = 0.95$ and different frequency ω . Obviously, the “∞” pinched hysteresis loops are observed with the given bipolar periodic inputs. As with the Caputo-like difference-based memristor, the area in the hysteresis loop shrinks with the decrease in derivative-order α and increase in frequency ω . According to the numerical simulation analysis results, the designed fractional-order memristor satisfies the three fingerprints of the memristor.

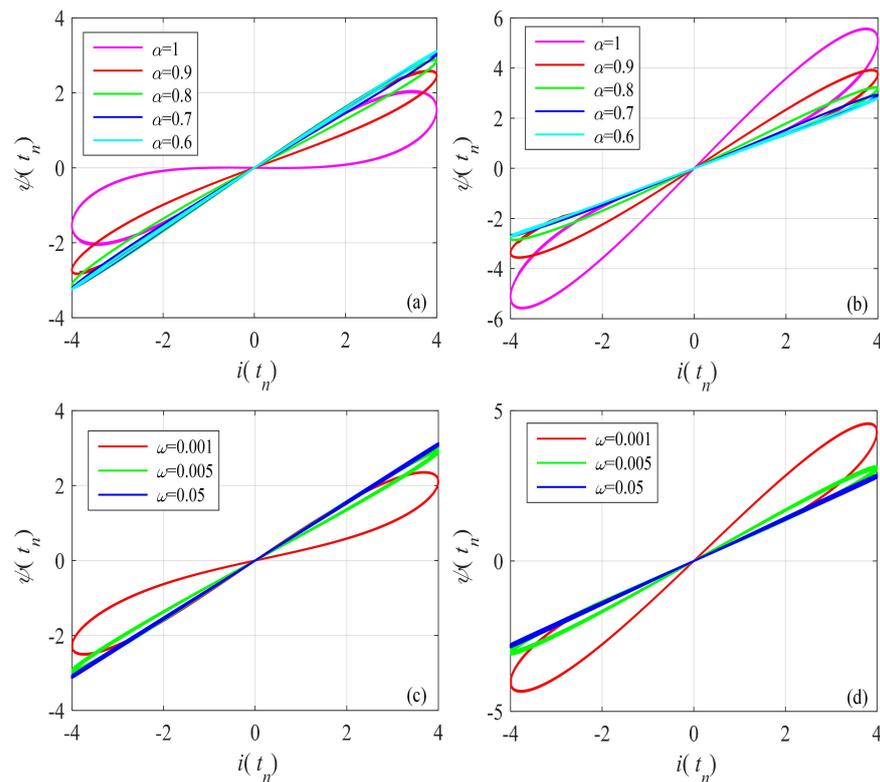


Figure 2. Pinched hysteresis loops of the G–L difference-based discrete memristor with $x(t_0) = 0.1$. (a) $g(x) = a_1 + b_1x$, amplitude $A = 4$ and frequency $\omega = 0.001$, and different derivative order; (b) $g(x) = a_2 + b_2|x|$, amplitude $A = 4$ and frequency $\omega = 0.001$, and different derivative order; (c) $g(x) = a_1 + b_1x$, $A = 4, \alpha = 0.95$ and different frequency ω ; (d) $g(x) = a_2 + b_2|x|$, $A = 4, \alpha = 0.95$ and different frequency ω .

3.2. Fractional-Order Caputo Difference-Based Model

Based on Caputo-like difference, the mathematical model of the fractional-order discrete memristor is denoted as [31]

$$\begin{cases} \psi(t_n) = g(\varphi, u, t_n)i(t_n) \\ {}^C\Delta_{t_0}^\alpha \varphi(t) = f(\varphi, i, t + \alpha - 1) \end{cases} \quad (25)$$

where $g(\cdot)$ and $f(\cdot)$ are the nonlinear functions and φ is the internal state variable of the memristor (equivalent to the “charge” or “magnetic flux” in the continuous memristors).

According to Definition 2, the solution of the second equation in Equation (25) is given by

$$\varphi(t_n) = \varphi(t_0) + \frac{1}{\Gamma(\alpha)} \sum_{j=1}^n C_j^{(\alpha)} f(\varphi(t_{j-1}), u, t_{j-1}), \quad (26)$$

where the coefficient is defined by

$$C_j^{(\alpha)} = \frac{\Gamma(n - j + \alpha)}{\Gamma(n - j + 1)}, \quad (27)$$

and it can be calculated in a more effective way, which is denoted as

$$C_j^{(\alpha)} = \begin{cases} \Gamma(\alpha), j = 0 \\ C_{j-1}^{(\alpha)} \frac{\alpha + j - 1}{j}, j = 1, 2, \dots \end{cases} \quad (28)$$

Finally, the fractional-order discrete memristor is rewritten as

$$\begin{cases} \psi(t_n) = g(\varphi, u, t_n)u(t_n) \\ \varphi(t_n) = \varphi(t_0) + \sum_{j=1}^n C_j^{(\alpha)} f(\varphi(t_{j-1}), u, t_{j-1}) \end{cases} \quad (29)$$

The nonlinear functions including $g(\cdot)$ and $f(\cdot)$ in the memristor have different kinds of considerations.

An example of the Caputo fractional-order discrete memristor is given by

$$\begin{cases} y(t_n) = g(x(t_n))i(t_n) \\ {}^C\Delta_{t_0}^\alpha x(t) = ki(t + \alpha - 1) \end{cases} \quad (30)$$

Thus, the model with coefficient is

$$\begin{cases} y(t_n) = g(x(t_n))i(t_n) \\ x(t_n) = x(t_0) + \frac{k}{\Gamma(\alpha)} \sum_{j=1}^n C_j^{(\alpha)} i(t_{j-1}) \end{cases} \quad (31)$$

The voltage–current curves of the discrete memristor with different functions $g(\cdot)$, derivative-order α , and frequency ω are shown in Figure 3. Figure 3a,b are plotted using $A = 4$, $\omega = 0.001$ and different derivative-order α . Figure 3c,d are plotted using $A = 4$, $\alpha = 0.9$ and different frequency ω . This shows that there exist pinched hysteresis loops when the input signal is bipolarly periodic. The area of the hysteresis loop decreases with the decrease of derivative-order α and the increase of frequency ω . First, this shows that the fractional difference makes the discrete memristor have a lower frequency domain, since the area of the hysteresis loops decreases with order α . Second, this shows that the pinched hysteresis loop shrinks with frequency ω and becomes a line when $\omega = 0.5$. It can be seen that the designed fractional-order memristor satisfies the three fingerprints of the memristor.

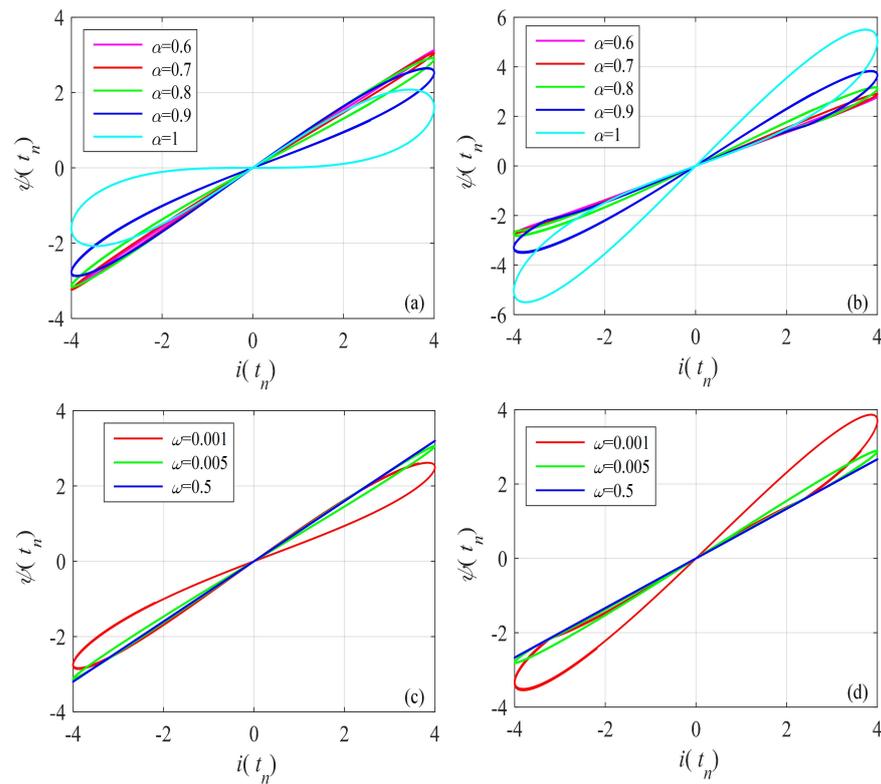


Figure 3. Pinched hysteresis loops of the Caputo-like difference-based discrete memristor with $x(t_0) = 0.1$. **(a)** $g(x) = a_1 + b_1x$, amplitude $A = 4$ and frequency $\omega = 0.001$, and different derivative order; **(b)** $g(x) = a_2 + b_2|x|$, amplitude $A = 4$ and frequency $\omega = 0.001$, and different derivative order; **(c)** $g(x) = a_1 + b_1x$, $A = 4$, $\alpha = 0.9$ and different frequency ω ; **(d)** $g(x) = a_2 + b_2|x|$, $A = 4$, $\alpha = 0.9$ and different frequency ω .

3.3. Integer-Order Discrete Memristor

For the G–L fractional-order difference-based discrete memristor, when $\alpha = 1$, the coefficients are given by

$$G_0^{(1)} = 1, G_1^{(1)} = -1, G_j^{(1)} = 0(j = 2, 3, \dots). \tag{32}$$

Thus, the “charge” or “magnetic flux” in the integer-order discrete memristor is denoted as

$$\varphi(t_n) = \varphi(t_{n-1}) + f(\varphi(t_{n-1}), i(t_{n-1}), t_{n-1}). \tag{33}$$

For the Caputo difference-based fractional-order discrete memristor, the coefficients are $G_j^{(1)} = 1(j = 0, 1, 2, \dots)$ for $\alpha = 1$. The solution of the “charge” or “magnetic flux” is denoted as

$$\begin{aligned} \varphi(t_n) &= \varphi(t_0) + \sum_{j=1}^n f(\varphi(t_{j-1}), i(t_{j-1}), t_{j-1}) \\ &= \varphi(t_0) + \sum_{j=1}^{n-1} f(\varphi(t_{j-1}), i(t_{j-1}), t_{j-1}) + f(\varphi(t_{n-1}), i(t_{n-1}), t_{n-1}) \\ &= \varphi(t_{n-1}) + f(\varphi(t_{n-1}), i(t_{n-1}), t_{n-1}) \end{aligned} \tag{34}$$

Thus, for both the Caputo difference-based memristor and the G–L difference-based memristor, when $\alpha = 1$, the integer-order counterpart is defined by

$$\begin{cases} \psi(t_n) = g(\varphi_n, i(t_n), t_n)i(t_n) \\ \varphi(t_n) = \varphi(t_{n-1}) + f(\varphi(t_{n-1}), i(t_{n-1}), t_{n-1}) \end{cases} \tag{35}$$

It should be noted that the integer-order discrete memristor is a special case of fractional-order discrete memristors. At present, integer-order discrete memristors are being investigated. For example, there is a chaotic map using a second-order discrete memristor [34], which is defined by

$$\begin{cases} x_{n+1} = x_n - kx_n \cos \varphi_n \\ \varphi_{n+1} = \varphi_n + \varepsilon x_n \end{cases}, \tag{36}$$

where the discrete memristor is given by

$$\begin{cases} x_{n+1} = kx_n \cos \varphi_n \\ \varphi_{n+1} = \varphi_n + \varepsilon x_n \end{cases}. \tag{37}$$

It is not difficult to verify that there is an “∞” hysteresis loop, which means that there exists a memristor in the given discrete memristor.

3.4. Short-Term Memory Effects and Frequency Domain

3.4.1. The Imperfect Memory Effect

As shown in Figure 4a, the integer-order discrete memristor has the ideal memory effect, since $\varphi(t_n)$ is the summation of all the history data. The reason for this simplified calculation is that all the coefficients for the history data are same, which is one.

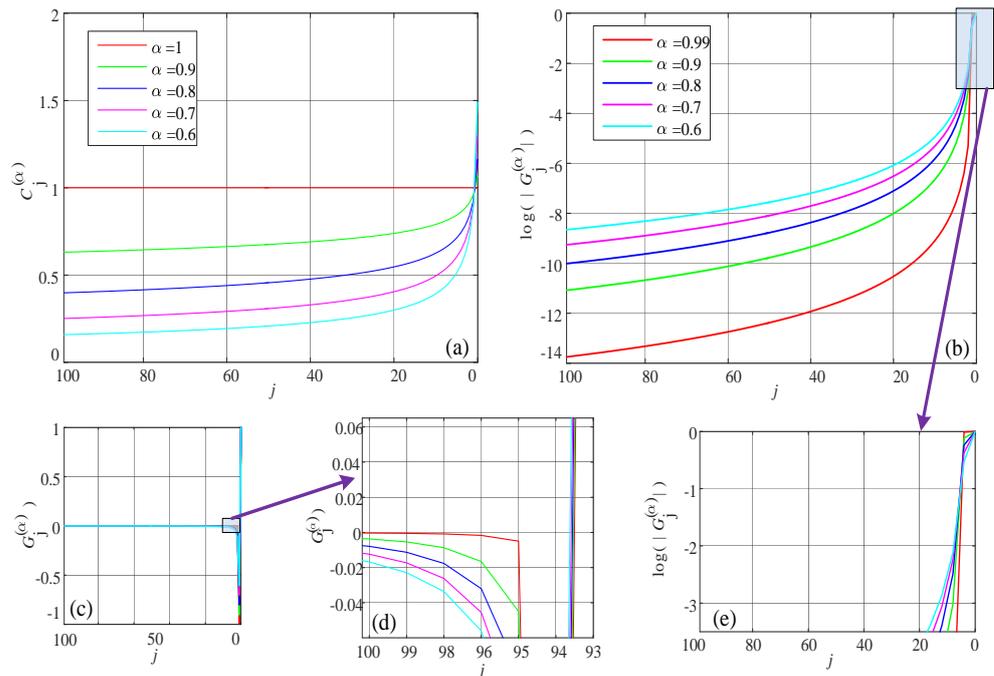


Figure 4. Memory effect of the fractional-order differences. (a) Caputo difference; (b–e) G–L difference, where (e) is the partial enlarged drawing of (b), and vertical coordinates of (b,e) are logarithmic, (d) is the partial enlarged drawing of (c).

However, when $0 < \alpha < 1$, then there is a different coefficient for each history data point, and the values of the coefficients based on the two fractional calculi are different. The kernel functions $C_j^{(\alpha)}$ and $G_j^{(\alpha)}$ are plotted in Figure 4, where $n = 100$. As shown in Figure 4, for the Caputo difference, $C_j^{(\alpha)}$ tends to zero with a speed that is not as fast as the $G_j^{(\alpha)}$ from G–L difference. However, when $\alpha = 1, j = 0, 1, 2, \dots$, and we have a horizontal line for $C_j^{(\alpha)}$. In addition, $\alpha = 1, C_j^{(\alpha)} = 1$ and for $j = 2, 3, 4, \dots$, there is no line for $G_j^{(1)}$, because when $\alpha < 1$, the coefficients for different positions are different.

First, for fractional-order cases, the calculation cost increases with iterations and, in real applications, we need to balance calculation cost and accuracy. Second, when $0 < \alpha < 1$, the memory effect is not “perfect”. Thus, from the point view of this paper, there exists an “imperfect memory effect” in the fractional-order discrete memristors.

3.4.2. Short-Term Memory Memristor

The short-term memory effect of the fractional-order difference has been discussed by different researchers [45,46]. It sacrifices the accuracy of the model to balance calculation cost. In this paper, we use the sliding-window memory scheme. It means that current data are related to the previously limited number of data. For the Caputo-like fractional-order discrete memristor with short-term memory effect, it is defined as

$$\varphi(t_n) = \varphi(t_0) + \frac{1}{\Gamma(\alpha)} \sum_{j=1}^L C_j^{(\alpha)} f(\varphi(t_{n-j}), i(t_{n-j}), t_{n-j}) \quad (38)$$

where $L = \min\{n, L_C\}$, and L_C is the memory length. For the Caputo fractional-order discrete memristor given in Equation (30), the short-term memory model is given by

$$x(t_n) = x(t_0) + \frac{k}{\Gamma(\alpha)} \sum_{j=1}^L C_j^{(\alpha)} i(t_{n-j}). \quad (39)$$

The bipolarly periodic input signals and parameters of the memristor function $g(\cdot)$ are given in Equation (16). Let $\alpha = 0.95$, $x(t_0) = 0.01$. The pinched hysteresis loops of the Caputo difference-based memristor with different memory length and frequency are shown in Figure 5. This shows that the area of the pinched hysteresis loops decreases with an increase of frequency and decrease of memory length. Thus, the memristor with short-term memory effect also satisfies the definition of the memristor, and the memory length should not be too short. For the given example, memory length should be larger than 1000.

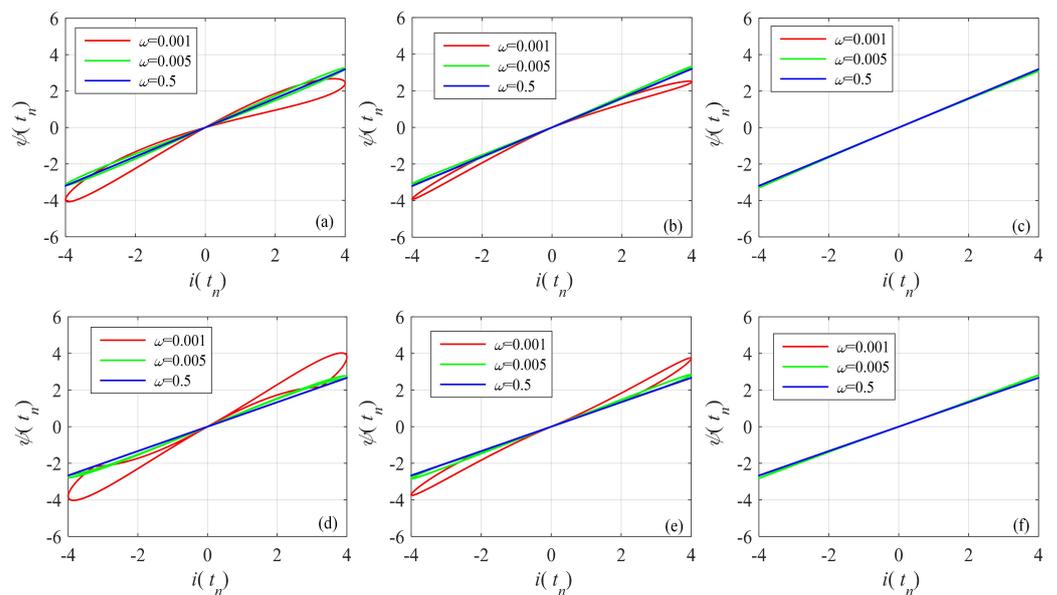


Figure 5. Pinched hysteresis loops of the Caputo difference-based discrete memristor with short-term memory effect. (a) $g(x) = a_1 + b_1x$, and $L_C = 2000$; (b) $g(x) = a_1 + b_1x$, and $L_C = 1000$; (c) $g(x) = a_1 + b_1x$, and $L_C = 100$; (d) $g(x) = a_2 + b_2|x|$, and $L_C = 2000$; (e) $g(x) = a_2 + b_2|x|$, and $L_C = 1000$; (f) $g(x) = a_2 + b_2|x|$, and $L_C = 100$.

For the G–L definition-based model, the memory part is in the second item. Here, the G–L definition-based fractional-order discrete memristor with memory length L is defined by

$$\varphi(t_n) = f(\varphi(t_{n-1}), u, t_{n-1}) - \sum_{j=1}^L G_j^{(\alpha)} \varphi(t_{n-j}), \tag{40}$$

where $L = \min\{n, L_{GL}\}$. Although $G_j^{(\alpha)}$ decreases with j significantly as shown in Figure 4, it is still necessary to investigate how memory length affects the nonlinearity of the memristor. The discrete memristor given in Section 3.2 with short-term memory effect is defined by

$$x(t_n) = i(t_{n-1}) - \sum_{j=1}^L G_j^{(\alpha)} x(t_{n-j}). \tag{41}$$

The bipolar periodic input signals and parameters of the memristor function $g(\cdot)$ are given in Equation (16). Let $\alpha = 0.95$, $x(t_0) = 0.01$. The pinched hysteresis loops of the G–L difference-based memristor with different memory lengths and frequencies are shown in Figure 6. This also shows that the area in the pinched hysteresis loops decreases with an increase in frequency and decrease in memory length. As shown in Figure 6, when the memory length is larger than 10, the nonlinearity of the memristor is acceptable. However, the larger memory means a larger area in the pinched hysteresis loops. Compared with the Caputo difference memristor, the G–L difference-based memristor has much less memory for the satisfying nonlinearity.

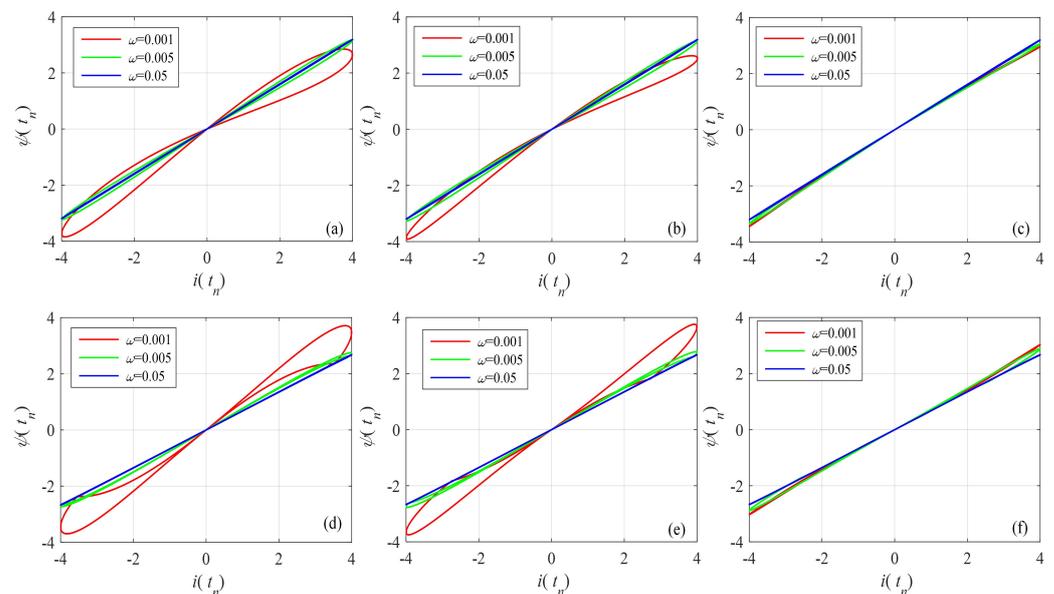


Figure 6. Pinched hysteresis loops of the G–L difference-based discrete memristor with short-term memory effect. (a) $g(x) = a_1 + b_1x$, and $L_{GL} = 100$; (b) $g(x) = a_1 + b_1x$, and $L_{GL} = 50$; (c) $g(x) = a_1 + b_1x$, and $L_{GL} = 10$; (d) $g(x) = a_2 + b_2|x|$, and $L_{GL} = 100$; (e) $g(x) = a_2 + b_2|x|$, and $L_{GL} = 50$; (f) $g(x) = a_2 + b_2|x|$, and $L_{GL} = 10$.

4. Physical Significance of Discrete Memristor

At present, scientists use different symbols to represent different memristors. Generally, a solid rectangle represents an integer-order memristor, while a solid triangle represents a fractional-order memristor. However, there are no recognized symbols for discrete memristors. Here, Table 1 is the summary of the symbols used for different kinds of memristors. For the discrete memristor, it can be realized in a digital circuit where parameters are set by the control signal. In fact, the symbol represents a black box, where the interior structure of different memristors is different with different implementation techniques.

Table 1. Types, formulas and symbols of generalized memristors.

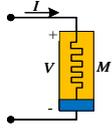
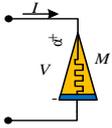
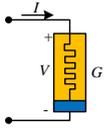
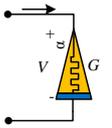
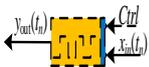
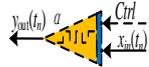
Memristor	Integer-Order		Fractional-Order	
	Formula	Symbol	Formula	Symbol
Charge-controlled memristor	$V = M(q, i)i$ $\frac{dq}{dt} = ki$		$V = M(q, i)i$ $D_{t_0}^\alpha q = ki$	
Magnetron controlled memristor	$i = G(\varphi, V)V$ $\frac{d\varphi}{dt} = kV$		$i = G(\varphi, V)V$ $D_{t_0}^\alpha \varphi = kV$	
Discrete memristor	$y_{out} = F(\varphi, x_{in})x_{in}$ $\Delta\varphi = f(\varphi, x_{in})$		$y_{out} = F(\varphi, x_{in})x_{in}$ $\Delta^\alpha \varphi = f(\varphi, x_{in})$	

Table 2 shows that characteristics of different kinds of memristors are different.

- Both continuous and discrete memristors have memory effects. In particular, there exists a perfect memory effect in integer-order memristors, according to their mathematical models.
- The theoretical work of continuous memristors has been investigated systematically. However, there is little work on discrete cases.
- There is some research regarding nano-device implementation of both continuous and discrete memristors. However, an issue should be resolved. The implemented nano-devices have “∞” hysteresis loop but usually do not relate to a mathematical model, and the “∞” hysteresis loops are not elegant. We believe that discrete memristor models can prove a useful tool for memristor nano-devices.
- Although there are reports of the FPGA implementation of continuous memristors, analog circuit implementation of continuous memristors is the main technical means. However, discrete memristors are naturally supposed to be realized in digital circuits including DSP and FPGA.
- Continuous memristors can be used in continuous systems such as nonlinear chaotic systems and neural networks.

In real applications, there are many discrete systems, such as chaotic maps and discrete neural networks. Thus, how to introduce memristors to those discrete processes is a challenging topic. However, the proposed discrete memristors prove good models for these issues.

Table 2. The characters of different kinds of memristors.

Type	Characteristics	Applications	Modeling
Continuous memristor	<ul style="list-style-type: none"> ★ Memory effect; ★ Mathematical models are mature; ★ Nano-device implementation; ★ The parameters of memristor are fixed. 	<ul style="list-style-type: none"> ★ Continuous system; ★ Differential system 	<ul style="list-style-type: none"> ★ Differential operator; ★ Analog circuit
Discrete memristor	<ul style="list-style-type: none"> ★ Memory effect ★ Mathematical models is not mature ★ Nano-device implementation ★ Digital circuit realization of controllability ★ Input and output can be digital sequences 	<ul style="list-style-type: none"> ★ Discrete system; ★ Difference system; ★ Iterative system ; ★ Signal processing 	<ul style="list-style-type: none"> ★ Difference operator; ★ Digital circuit; ★ Mixed circuit

At present, there is no uniform definition for fractional calculus and difference. Therefore, the proposed models for fractional-order discrete memristors may not be final results. We believe there could be better models and implementations for discrete memristors. Here,

we propose two different fractional-order discrete memristors using Caputo difference and G–L difference.

Moreover, it is necessary to discuss the scheme for the realization of discrete memristors in analog circuits. Here, a feasible way is presented in Figure 7, which is proposed by Wu et al. [47]. Since the value of the memristor is decided by charge or magnetic flux, we sample the current or voltage of the circuit, and then the charge or magnetic flux is estimated in the microprocessor based on sample data. Finally, the output signal of the microprocessor decides the value of the variable resistance, which can be a digital potentiometer. Figure 7 provides a scheme based on the analog–digital hybrid circuit, but it still needs to be further investigated by engineers.

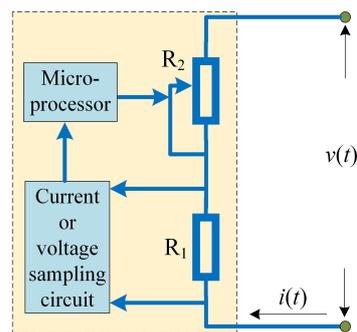


Figure 7. A physical realization scheme of the discrete memristor for analog and digital circuits (Reproduced with permission from [47] [Guangyi Wang], [J. Hangzhou Dianzi Univ. Nat. Sci]; published by [J. Hangzhou Dianzi Univ. (Natural Science edition)], [2018]).

5. Discrete Memristive Systems

5.1. Design of Discrete Memristive Map

The Sine map is defined by

$$y_n = \varepsilon \sin(\pi y_{n-1}), \tag{42}$$

where ε is the system parameter. Fixed $\varepsilon = 3$, phase diagram and time series of the Sine map are shown in Figure 8. In this section, the integer-order and fractional-order discrete memristors are introduced to this system.

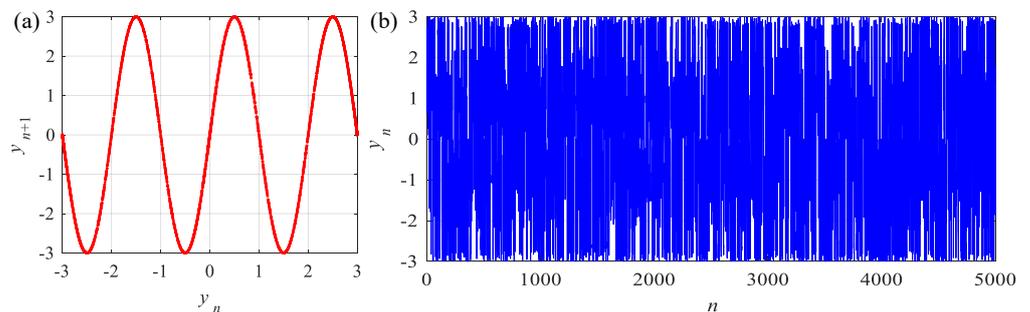


Figure 8. Phase diagram and time series of the Sine map. (a) Phase diagram; (b) Time series.

The design of discrete memristive chaotic maps is a hot topic. To date, as for the applications of discrete memristors in nonlinear systems, we have already introduced discrete memristors into chaotic maps, including the Sine map [31] and Hénon map [32] for better performance. However, designing a fractional-order chaotic memristive map is still a challenge. The block diagrams, as shown in Figure 9, are the Sine map and the discrete memristor Sine map. For a discrete memristive map, the signal y_{n-1} is the input of the discrete memristor and the output of the memristor V_{n-1} is then sent to the Sine map.

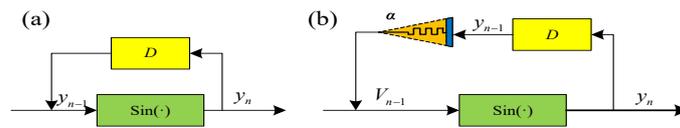


Figure 9. The block diagrams for the (a) chaotic map and (b) chaotic map with discrete memristor.

5.2. Caputo Fractional-Order Sine Map

According to Figure 9b, the fractional-order discrete memristive Sine map with Caputo discrete memristor is defined by [31]

$$\begin{cases} y(t_n) = \varepsilon \sin(\pi V_{n-1}) \\ V_{n-1} = y(t_{n-1})g(q(t_{n-1})) \\ {}^C \Delta_{t_0}^\alpha q(t) = ki(t + \alpha - 1) \end{cases} \quad (43)$$

where $g(q) = 0.667 + b|q|$. When we considered the short-term memory effect, the third equation in this model is solved by

$$q(t_n) = q(t_0) + \frac{k}{\Gamma(\alpha)} \sum_{j=1}^L C_j^{(\alpha)} i(t_{n-j}), \quad (44)$$

where $L = \min\{n, L_C\}$, and L_C is the memory length. In this study, $L_C = 1000$. The initial conditions of the system are $q_0 = 0.2$ and $y_0 = 0.63$.

Fixing $\varepsilon = 3$ and letting b take different values including $b = 1.4828 \times 10^{-1}$, $b = 1.4828 \times 10^{-2}$, $b = 1.4828 \times 10^{-3}$ and $b = 1.4828 \times 10^{-4}$, phase diagrams of the Caputo fractional-order memristive Sine map are shown in Figure 10. This shows that the distribution of the points in the phase plane $y_n - y_{n+1}$ becomes concentrated with the decrease of parameter b . Thus, the complexity of the system can be affected by memristor parameter b .

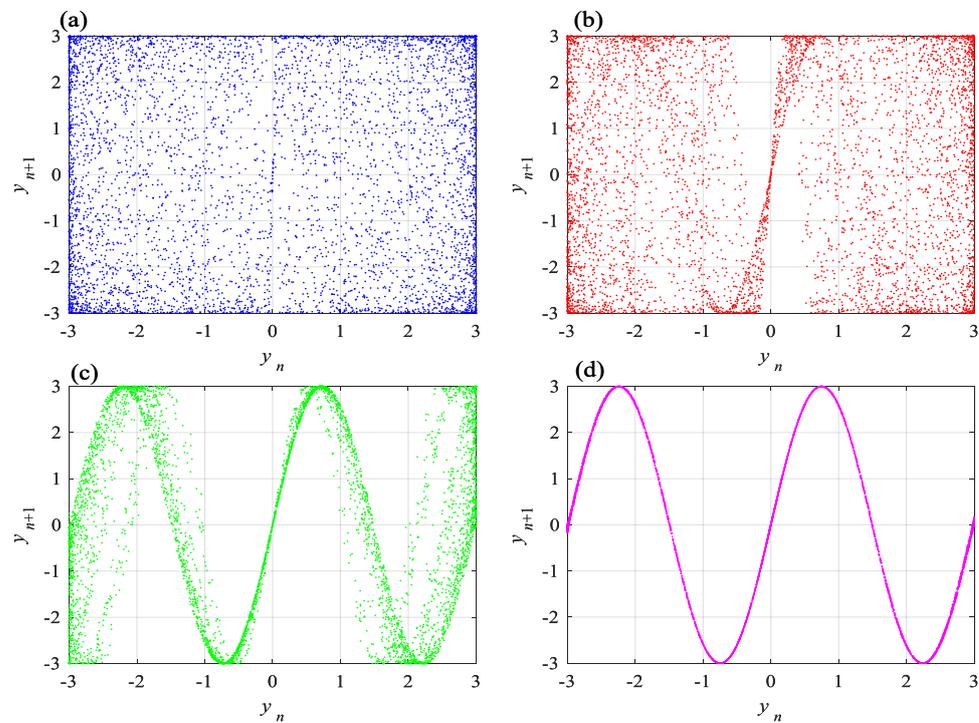


Figure 10. Phase diagrams of the Caputo fractional-order chaotic memristive Sine map with different parameter b . (a) $b = 1.4828 \times 10^{-1}$; (b) $b = 1.4828 \times 10^{-2}$; (c) $b = 1.4828 \times 10^{-3}$; (d) $b = 1.4828 \times 10^{-4}$.

5.3. G–L Fractional-Order Sine Map

Here, based on the G–L difference, the G–L memristor Sine map is defined as

$$\begin{cases} y(t_n) = \varepsilon \sin(\pi V_{n-1}) \\ V_{n-1} = y(t_{n-1})g(q(t_{n-1})) \\ {}^G\Delta_{t_0}^\alpha q(t_n) = ky(t_{n-1}) \end{cases}, \quad (45)$$

where $g(q) = 0.667 + b|q|$. The solution of the third equation in Equation (45) is given by

$$q(t_n) = ky(t_{n-1}) - \sum_{j=1}^L G_j^{(\alpha)} q(t_{n-j}), \quad (46)$$

where $L = \min\{n, L_C\}$, and L_C is the memory length. Here, $L_C = 50$. By putting $q(t_n)$ as the first item of Equation (45), the model of the system G–L difference-based Sine map is obtained.

Fix $\varepsilon = 3$ and let b take different values. Then, phase diagrams of the system are shown in Figure 11. These show that the phase diagram becomes to a Sine-like curve with the decrease of memristor parameter b . As shown in Figure 11, when $b = 1.4828$, the points of data fill the whole space. This assumes the highest complexity compared to other cases, and the complexity of the system decreases with the decrease of b .

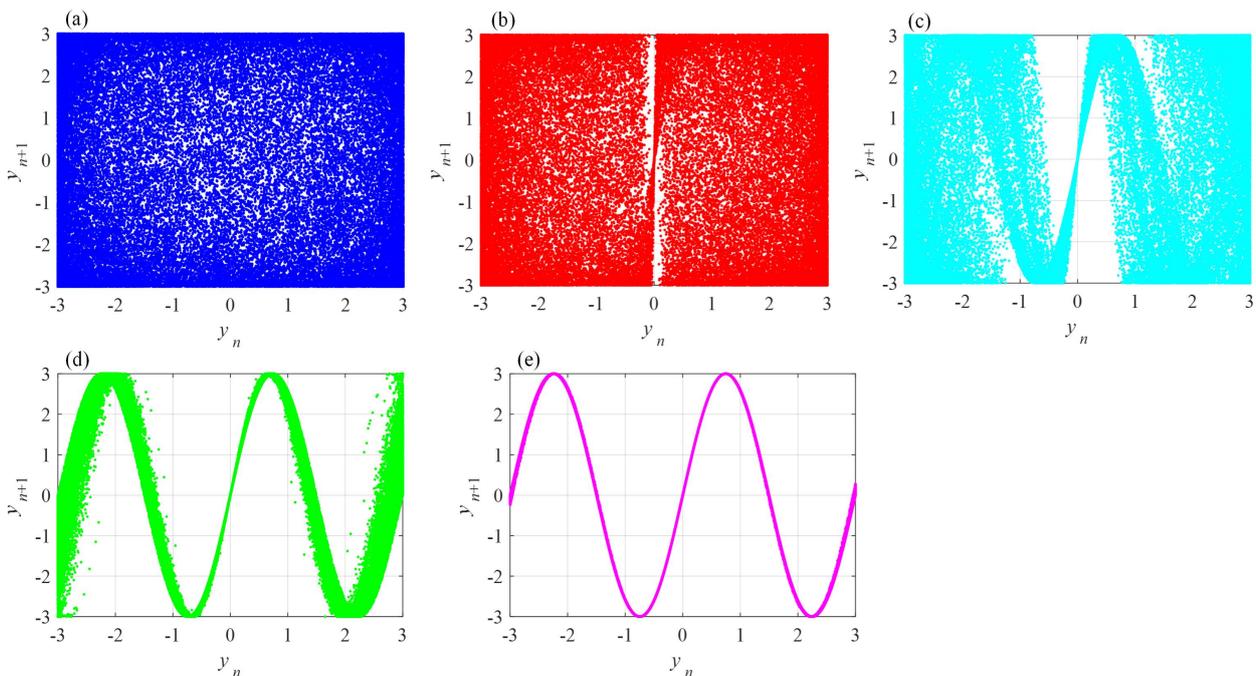


Figure 11. Phase diagrams of fractional-order chaotic memristive Sine map with different parameter b . (a) $b = 1.4828$; (b) $b = 1.4828 \times 10^{-1}$; (c) $b = 1.4828 \times 10^{-2}$; (d) $b = 1.4828 \times 10^{-3}$; (e) $b = 1.4828 \times 10^{-4}$;

5.4. Integer-Order Discrete Sine Map

For both fractional-order cases, when the difference order $\alpha = 1$, the discrete memristive Sine map is defined as

$$\begin{cases} y_n = \varepsilon \sin(\pi y_{n-1}g(q_{n-1})) \\ q_n = q_{n-1} + ky_{n-1} \end{cases}, \quad (47)$$

which is an integer-order discrete memristive system. Here, $g(q) = 0.667 + b|q|$. ε and b are the two parameters of this system.

Let $\varepsilon = 3$ and b take different values. Numerical analysis results of the system are shown in Figure 12. Similar results are obtained from the system. When b takes larger values, the points in the phase plane are more evenly distributed. Moreover, q_n and y_n are

plotted. q_n is the accumulation of time series y_n , which fluctuates on both sides of zero. In conclusion, complexity of the integer-order memristive Sine map can be modulated by the memristor parameter.

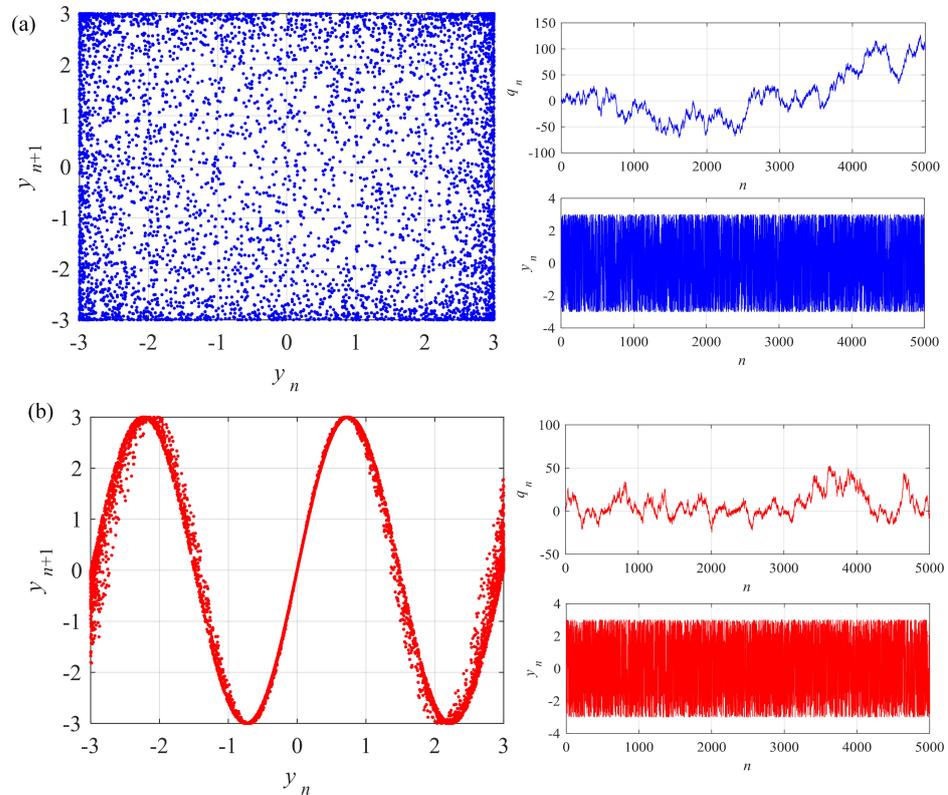


Figure 12. Phase diagrams and time series of the integer-order memristive Sine map with different parameter b . (a) $b = 1.4828$; (b) $b = 1.4828 \times 10^{-3}$.

5.5. Complexity Analysis

5.5.1. Maximum LEs of the Integer-Order System

Vary ϵ from 0 to 10 with step size of 0.01 and vary the parameter b from 0 to 1.4828 with step size of 0.0148. Maximum Lyapunov exponents (LEs) analysis results are shown in Figure 13. This shows that the maximum LEs increase with system parameter ϵ and memristor parameter b , and there are many periodic windows found. Meanwhile, the value of the maximum LEs can reach more than 8, which is larger than most of the chaotic maps. For instance, when $\epsilon = 3$, the Lyapunov exponent of the Sine map is 1.6721, which is much smaller than those values. Thus, the complexity of the Sine map is improved with the introduction of a discrete memristor.

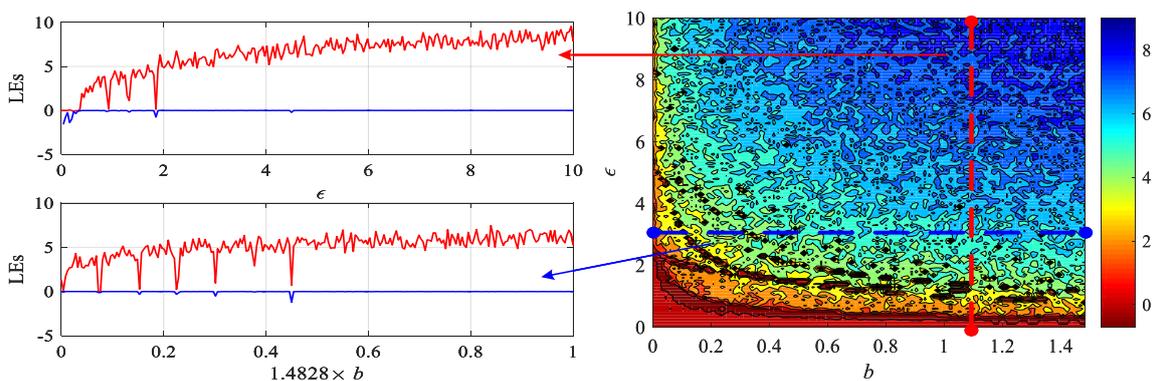


Figure 13. Maximum LEs of the integer-order discrete memristive Sine map.

5.5.2. SampEn and NetPE Complexity Analysis

Generally, the complexity of time series means the time series are closer to a random sequence. However, there are many different approaches to measure complexity. In this paper, we use SampEn [48] and NetPE (network permutation entropy) [49] to measure the complexity of the time series generated by different chaotic maps. NetPE measures complexity by building a network using the Bandt–Pompe patterns and their weights. SampEn measures complexity by estimating the probability of generating new patterns in the time series. The greater the probability of generating new patterns, the higher the complexity of the series.

The NetPE algorithm [49] is defined as

$$NetPE(x^N, d, error) = -\frac{1}{\log(N - d + 1)} \sum_{i=1}^{N-d+1} P(i) \log(P(i)), \tag{48}$$

where x^N represents the time series x with length N , d is the Bandt–Pompe dimension, and $error$ is the tolerance of difference between Bandt–Pompe vectors. More details about this algorithm can be found in [49]. In this study, $d = 3$, $error = 0.02$ and the length of time series $N = 50,000$.

For the given sequence $\{x^N : x_1, x_2, \dots, x_N\}$, the definition of SampEn [48] is given by

$$\begin{cases} SamEn(m, r, N) = \ln \Phi^m(r) - \ln \Phi^{m+1}(r) \\ \Phi^m(r) = (N - m + 1)^{-1} \sum_{i=1}^{N-m+1} C_i^m(r) \\ C_i^m(r) = (N - m)^{-1} \sum_{j=1, j \neq i}^{N-m+1} d_{i,j}^m \\ d_{i,j}^m = \max_{p=1,2,\dots,m} (|u(i + p + 1) - u_0(i)|) \end{cases}, \tag{49}$$

where m is the phase space dimension, r is the similarity tolerance, N is the sequence length, and $m(m \leq N - 2)$ is the non-negative integer. In this paper, $r = 0.15$ and $m = 3$. In addition, the length of time series for SampEn is 2500.

Let $\epsilon = 3$ and $\alpha = 0.95$. Complexity analysis results with b varying are shown in Figure 14. Since Sine does not have parameter b , there is a transverse line presented in this figure for comparison. Let $b = 1.4828 \times 0.8$ and $\alpha = 0.95$. The complexity analysis results with variation of ϵ are shown in Figure 15. This shows that memristive Sine maps have higher complexity than the Sine map, which means that the complexity of the original chaotic map can be improved with the introduction of discrete memristors.

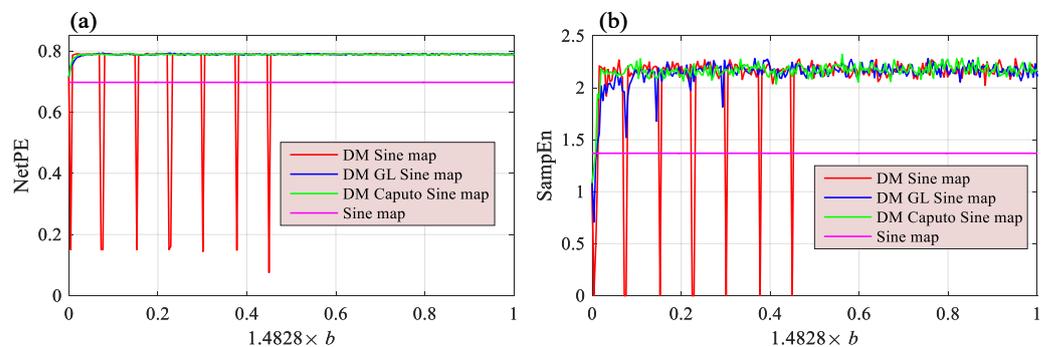


Figure 14. Complexity analysis results of the Sine maps with b varying measured by different methods (a) NetPE; (b) SampEn.

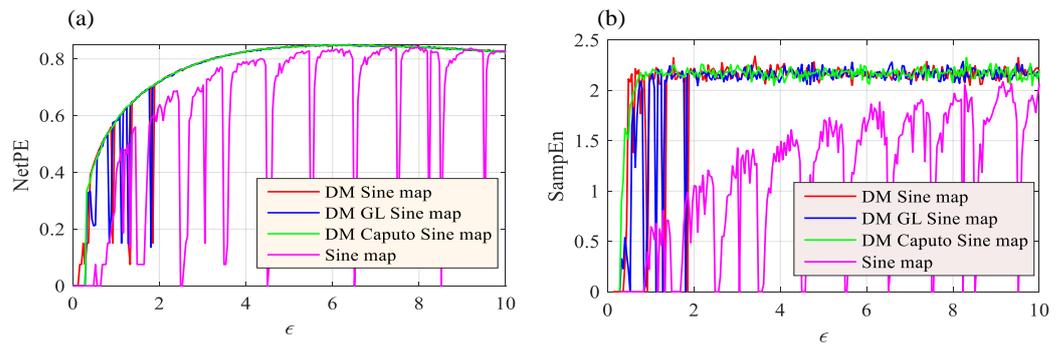


Figure 15. Complexity analysis results of the Sine maps with ϵ varying measured by different methods (a) NetPE; (b) SampEn.

Since $g(q) = 0.667 + b|q|$, when b takes larger values, the contribution of q to the system is obvious. For instance, when $b = 1.4828 \times 10^{-4}$, $g(q) \approx 0.667$. When $b = 1.4828$, $g(q)$ will reach 2.1498. This makes the system more complex. The main reason for this is that the internal nonlinearity of the discrete memristor can significantly improve the complexity of the system when b takes large values. In conclusion, the discrete memristor can improve complex behaviors of the chaotic map.

6. Implementations of the Discrete Memristive Systems

6.1. FPGA Digital Circuit Design

According to the Equation (39), the digital circuit of a Caputo fractional-order discrete memristor is shown in Figure 16. It shows *Reg_It* and *Reg_Ct* are the core components to implement the Caputo fractional-order discrete memristor, where the former is the queue to store input current I_{t_n} and the latter is the array to store the Caputo differential operator. Additionally, n and len represent the discrete time and the memory length, respectively. The value of len is n only when $len > n$, otherwise it is the set memory length. When the number of stored values reaches *Reg_It*'s maximum memory length len , the values in it move to their next address, and the new input current is stored in the lowest address [0]. The Caputo differential operator is generated by C_n^q operator module. The calculation process of the module is defined in Equation (28), and its digital implementation is presented in Figure 17a, where $n = 0, 1, 2, 3, \dots$, $Reg_Ct[0] = \Gamma(q)$ and q is the derivative order. In the Caputo differential calculation part of Figure 16, the values in *Reg_It* and *Reg_Ct* are multiplied in accordance with the rule of the sum of address being len . The multiplied values are accumulated and then combined with constants $k, \Gamma(q)$ and m_0 to form $x(t_n)$, where $m_0 = x(t_0)$. $x(t_n)$ is the input into the compute unit, $g(x(t_n))$ is the output, and then it multiplies by I_{t_n} to obtain the voltage output V_{t_n} , which also is $y(t_n)$.

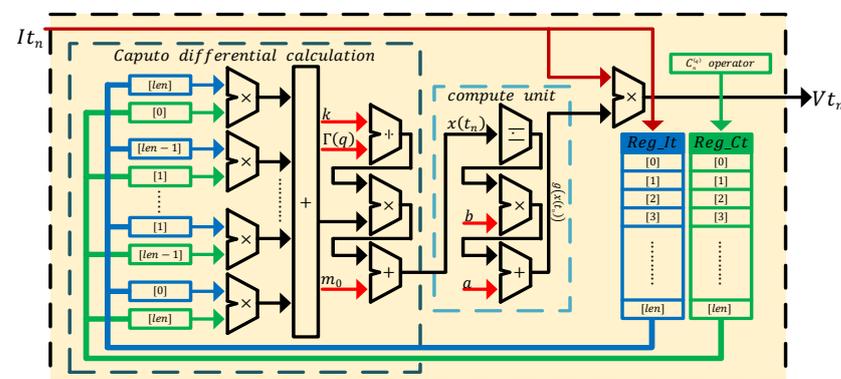


Figure 16. Digital circuit of the Caputo fractional-order discrete memristor.

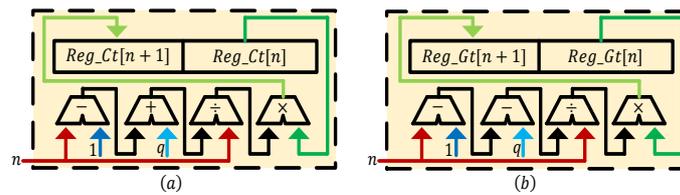


Figure 17. (a) Digital circuit of the C_n^q operator module; (b) Digital circuit of the G_n^q operator module.

According to Equation (40), the digital circuit of the G–L fractional-order discrete memristor is shown in Figure 18. In Figure 18, Reg_Xt is the queue to store $x(t_n)$ and Reg_Gt is the array to store G–L differential operator, where $Reg_Xt[0] = x(t_0)$. The definition of this module is given in Equation (22) and its digital implementation is presented in Figure 17b, where n and q have the same meanings as above. In the G–L differential calculation part of Figure 18, the values in Reg_Xt and Reg_Gt are multiplied in the same manner as above. The multiplied results are accumulated and combined with It_n to obtain $x(t_n)$. On the one hand, the value of $x(t_n)$ is stored in Reg_Xt ; on the other hand, the value of $x(t_n)$ is transmitted to compute the unit part to obtain $g(x(t_n))$, which multiplies with It_n to obtain the voltage output Vt_n , which also is $y(t_n)$. Moreover, when the number of stored values of Reg_Xt reaches the maximum storage length len , it will also perform the same operation as Reg_It above.

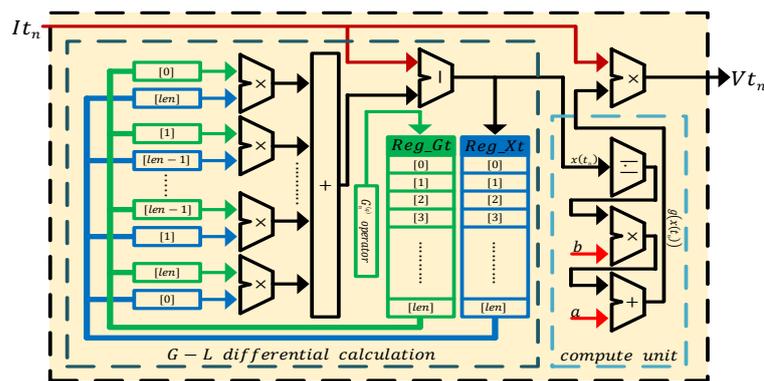


Figure 18. Digital circuit of the G–L fractional-order discrete memristor.

The model for the discrete memristive Sine map is shown in Figure 19. The designed Caputo fractional-order IP core and G–L fractional-order IP core are employed. Since the integer-order memristor is a special case of fractional-order memristor, we designed the IP core of the integer-order memristor directly in the memristive chaotic map, and have not presented the details. There is a selector, thus one of the three memristors works for the system and the proposed scheme can generate three different kinds of chaotic system.

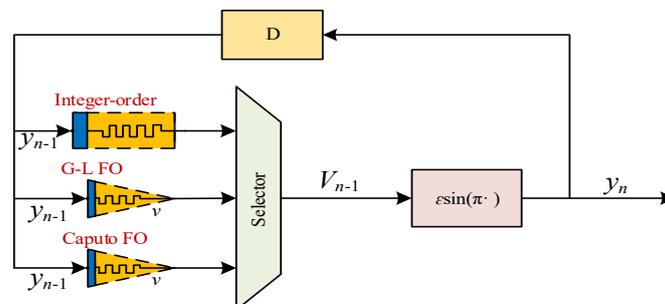


Figure 19. The data path circuit of the discrete memristive Sine map.

To compare our models with Matlab simulation results, we designed a memristor chaotic test system that contains multiple types of discrete memristors and discrete memristive chaotic systems. Figure 20 shows the designed system. It shows that the current and

generated voltage are converted from floating-point numbers to fixed-point numbers. As a result, these signals can be displayed in the oscilloscope.

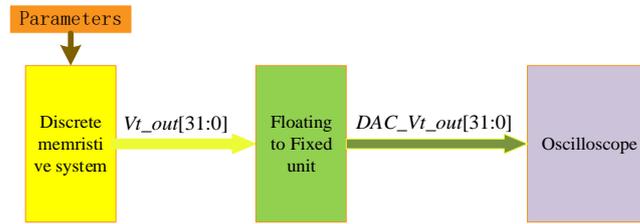


Figure 20. The designed system test circuit.

6.2. FPGA Implementation Results

The designed discrete memristor and memristive chaotic maps are realized in Altera DE2-115 with EP4CE115F29C7. It is shown in Figure 21 that the physical implementation platform for the discrete memristor systems is carried out. The output of the Altera DE2-115 contains two 16-bit current and voltage signals, which are converted using the DAC8552 chip. Finally, the obtained results are displayed in the oscilloscope.

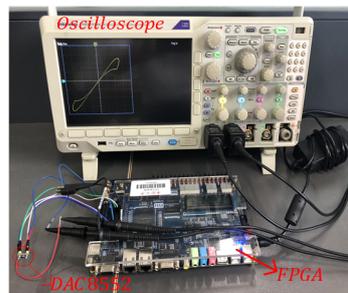


Figure 21. The physical implementation platform of the discrete memristive system.

In the discrete memristor IP core, we set $a = 0.667$ and $b = 1.4828 \times 10^{-4}$. The pinched hysteresis loops of the integer-order discrete memristor are presented in Figure 22. Meanwhile, the pinched hysteresis loops of the Caputo fractional-order discrete memristor are presented in Figure 23, and the pinched hysteresis loops of the G–L fractional-order discrete memristor are presented in Figure 24. They show that the pinched hysteresis loops displayed in the oscilloscope agree well with the corresponding Matlab simulation results and indicate that the designed memristor digital circuit can be used in the engineering application field.

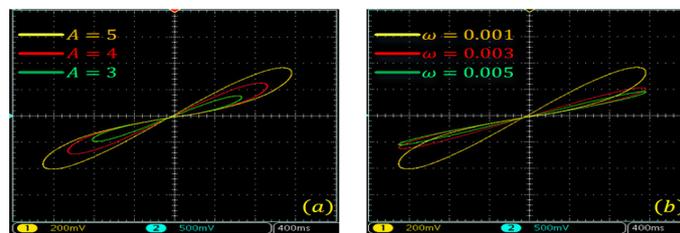


Figure 22. Pinched hysteresis loops of the integer-order discrete memristor. (a) Different A; (b) Different ω .

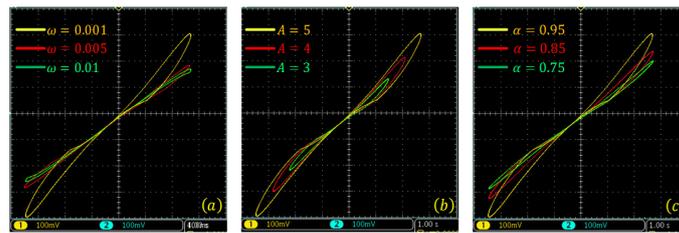


Figure 23. Pinched hysteresis loops of the Caputo-like difference-based-discrete memristor. (a) Different ω ; (b) Different A ; (c) Different α .

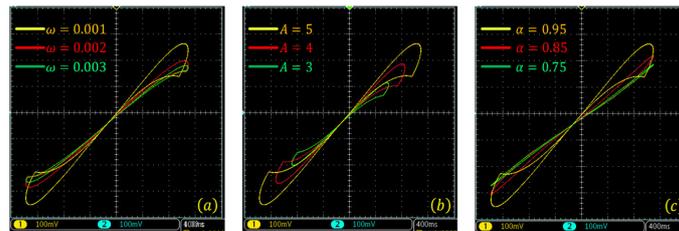


Figure 24. Pinched hysteresis loops of the G-L difference-based-discrete memristor. (a) Different ω ; (b) Different A ; (c) Different α .

Meanwhile, the discrete memristive chaotic maps are realized in the FPGA digital circuit. Let $\varepsilon = 3$ and b take values including $b = 1.4828 \times 10^{-4}$ and $b = 1.4828$, the results are shown in Figure 25. This shows that the digital circuit implementation results agree well with the corresponding Matlab simulation results. The proposed fractional-order discrete memristive Sine map can be realized in the hardware circuit for further applications.

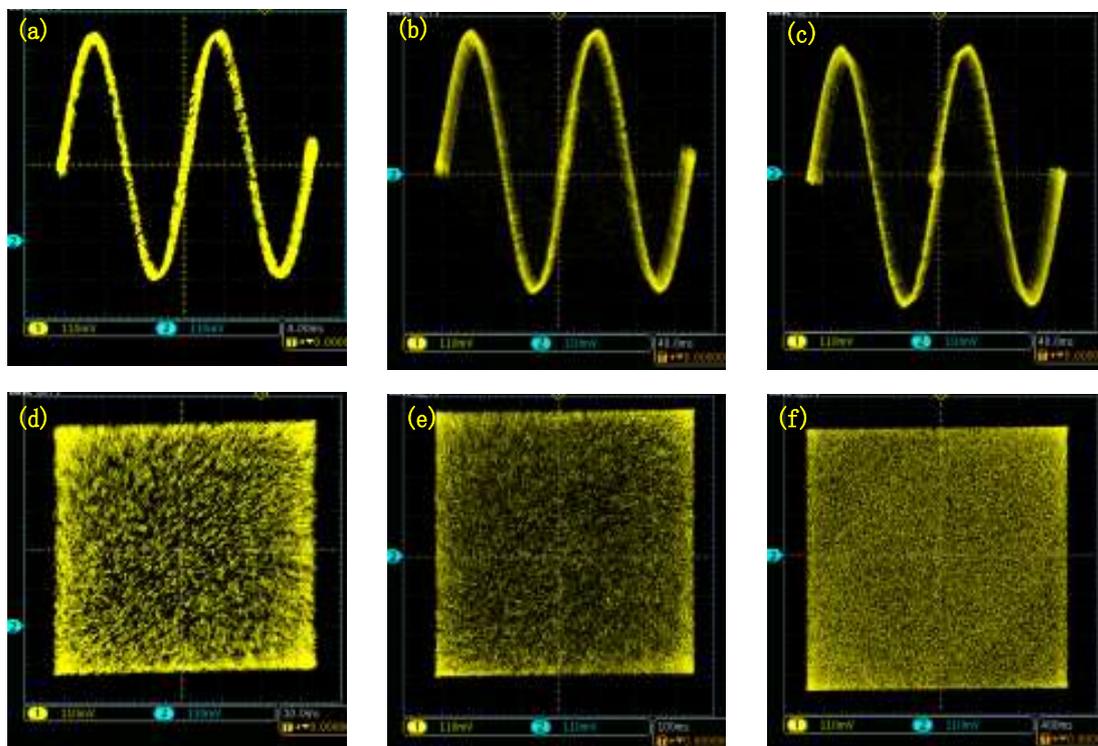


Figure 25. Phase diagrams of the chaotic maps from the FPGA digital circuit. (a) Integer-order system with $b = 1.4828 \times 10^{-4}$; (b) G-L fractional-order system with $b = 1.4828 \times 10^{-4}$; (c) Caputo fractional-order system with $b = 1.4828 \times 10^{-4}$; (d) Integer-order system with $b = 1.4828$; (e) G-L fractional-order system with $b = 1.4828$; (f) Caputo fractional-order system with $b = 1.4828$.

7. Conclusions

In this paper, we discussed the mathematical models for discrete memristors based on Caputo-like difference and G–L difference. Numerical simulations show that both kinds of discrete memristor satisfy the three fingerprints of a memristor device. The integer-order memristor can be deduced from both fractional-order memristors, although they are built through different fractional-order differences. Compared with the Caputo difference-based memristor, the G–L difference-based memristor works in a much lower frequency domain. The analysis results show that the integer-order memristor has the perfect memory effect, and the fractional-order memristor does not. The solution of the fractional-order memristor has a series of coefficients related to the fractional-order difference. Since the fractional-order discrete memristor is related all the historical data, to improve computational efficiency and to realize the discrete memristor in a digital circuit, discrete memristors with a short-term memory effect are built and analyzed. As a result, the characteristics of discrete memristors have some differences with the continuous memristors. Thus, it is a new way to model memristors, which can be used in discrete systems.

By introducing discrete memristors to the Sine map, three memristive chaotic maps are obtained. This shows that the system has higher complexity than the original Sine map. Moreover, FPGA digital circuit implementation of the fractional-order discrete memristors and chaotic maps are carried out. This shows that the proposed memristive systems can be realized in digital circuits. Finally, applications of discrete memristors and memristive chaotic systems will comprise our future work.

Author Contributions: Conceptualization, S.H. and H.W.; methodology, S.H. and H.W.; software, D.Z.; validation, S.H. and K.S.; formal analysis, K.S.; investigation, S.H.; resources, D.Z.; data curation, D.Z. and Y.P.; writing—original draft preparation, S.H.; writing—review and editing, H.W. and Y.P.; visualization, D.Z.; supervision, K.S.; project administration, K.S.; funding acquisition, H.W. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Natural Science Foundation of China (Nos. 61901530, 62071496, 62061008), the Natural Science Foundation of Hunan Province (No. 2020JJ5767).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: All data and models during the study appear in the submitted article.

Acknowledgments: The authors would like to thank the three anonymous reviewers for their constructive comments and insightful suggestions.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Chua, L. Memristor—the missing circuit element. *IEEE Trans. Circuit Theory* **1971**, *18*, 507–519. [[CrossRef](#)]
2. Strukov, D.; Snider, G.S.; Stewart, D.; Williams, R. The missing memristor found. *Nature* **2008**, *453*, 80–83. [[CrossRef](#)]
3. Adhikari, S.P.; Sah, M.; Kim, H.; Chua, L. Three Fingerprints of Memristor. *IEEE Trans. Circuits Syst. I Regul. Pap.* **2013**, *60*, 3008–3021. [[CrossRef](#)]
4. Pal, S.; Bose, S.; Ki, W.H.; Islam, A. Design of Power-and Variability-Aware Nonvolatile RRAM Cell Using Memristor as a Memory Element. *IEEE J. Electron Devices Soc.* **2019**, *7*, 701–709. [[CrossRef](#)]
5. Serb, A.; Khat, A.; Prodromakis, T. Seamlessly fused digital-analogue reconfigurable computing using memristors. *Nat. Commun.* **2018**, *9*, 2170. [[CrossRef](#)] [[PubMed](#)]
6. chul Kim, K.; Williams, R. A Family of Stateful Memristor Gates for Complete Cascading Logic. *IEEE Trans. Circuits Syst. I Regul. Pap.* **2019**, *66*, 4348–4355.
7. Chandrasekaran, S.; Simanjuntak, F.; Saminathan, R.; Panda, D.; Tseng, T.Y. Improving linearity by introducing Al in HfO₂ as memristor synapse device. *Nanotechnology* **2019**, *30*, 445205. [[CrossRef](#)]
8. Yao, P.; Wu, H.; Gao, B.; Tang, J.; Zhang, Q.; Zhang, W.; Yang, J.J.; Qian, H. Fully hardware-implemented memristor convolutional neural network. *Nature* **2020**, *577*, 641–646. [[CrossRef](#)]
9. Xu, C.; Wang, C.; Sun, Y.; Hong, Q.; Deng, Q.; Chen, H. Memristor-based neural network circuit with weighted sum simultaneous perturbation training and its applications. *Neurocomputing* **2021**, *462*, 581–590. [[CrossRef](#)]

10. Pershin, Y.V.; Di Ventra, M. Solving mazes with memristors: A massively parallel approach. *Phys. Rev. E* **2011**, *84*, 046703. [[CrossRef](#)]
11. Wang, G.Y.; He, J.L.; Yuan, F.; Peng, C.J. Dynamical Behaviors of a TiO₂ Memristor Oscillator. *Chin. Phys. Lett.* **2013**, *30*, 110506. [[CrossRef](#)]
12. Xu, Q.; Tan, X.; Zhang, Y.; Bao, H.; Hu, Y.; Bao, B.; Chen, M. Riddled Attraction Basin and Multistability in Three-Element-Based Memristive Circuit. *Complex* **2020**, *2020*, 4624792:1–4624792:13. [[CrossRef](#)]
13. Liang, Z.; He, S.; Wang, H.; Sun, K. A novel discrete memristive chaotic map. *Eur. Phys. J. Plus* **2022**, *137*, 309. [[CrossRef](#)]
14. Chew, Z.J.; Li, L. A discrete memristor made of ZnO nanowires synthesized on printed circuit board. *Mater. Lett.* **2013**, *91*, 298–300. [[CrossRef](#)]
15. Mickel, P.R.; Lohn, A.; Choi, B.J.; Yang, J.J.; Zhang, M.X.; Marinella, M.J.; James, C.D.; Williams, R.S. A physical model of switching dynamics in tantalum oxide memristive devices. *Appl. Phys. Lett.* **2013**, *102*, 223502. [[CrossRef](#)]
16. Khanal, G.M.; Cardarilli, G.; Chakraborty, A.; Acciarito, S.; Mulla, M.Y.; Di Nunzio, L.; Fazzolari, R.; Re, M. A ZnO-rGO composite thin film discrete memristor. In Proceedings of the 2016 IEEE International Conference on Semiconductor Electronics (ICSE), Kuala Lumpur, Malaysia, 17–19 August 2016; pp. 129–132.
17. Ordóñez-Miranda, J.; Ezzahri, Y.; Tiburcio-Moreno, J.A.; Joulain, K.; Drevillon, J. Radiative thermal memristor. *Phys. Rev. Lett.* **2019**, *123*, 025901. [[CrossRef](#)] [[PubMed](#)]
18. Merrikh-Bayat, F.; Parvizi, M. Practical method to make a discrete memristor based on the aqueous solution of copper sulfate. *Appl. Phys. A* **2016**, *122*, 1–10. [[CrossRef](#)]
19. Jo, S.; Chang, T.; Ebong, I.; Bhadviya, B.B.; Mazumder, P.; Lu, W. Nanoscale memristor device as synapse in neuromorphic systems. *Nano Lett.* **2010**, *10*, 1297–1301. [[CrossRef](#)]
20. Kim, H.; Sah, M.; Yang, C.; Cho, S.; Chua, L. Memristor Emulator for Memristor Circuit Applications. *IEEE Trans. Circuits Syst. I Regul. Pap.* **2012**, *59*, 2422–2431.
21. Song, H.; Kim, Y.; Park, J.; Kim, K. Designed Memristor Circuit for Self-Limited Analog Switching and its Application to a Memristive Neural Network. *Adv. Electron. Mater.* **2019**, *5*, 1800740. [[CrossRef](#)]
22. Wang, J.; Mou, J.; Yan, H.; Liu, X.; Ma, Y.; Cao, Y. A three-port switch NMR laser chaotic system with memristor and its circuit implementation. *Eur. Phys. J. Plus* **2021**, *136*, 1112. [[CrossRef](#)]
23. Jiang, Y.; Li, C.; Zhang, C.; Zhao, Y.; Zang, H. A Double-Memristor Hyperchaotic Oscillator With Complete Amplitude Control. *IEEE Trans. Circuits Syst. I Regul. Pap.* **2021**, *68*, 4935–4944. [[CrossRef](#)]
24. Tolba, M.F.; Fouda, M.; Hezayyin, H.G.; Madian, A.H.; Radwan, A.G. Memristor FPGA IP Core Implementation for Analog and Digital Applications. *IEEE Trans. Circuits Syst. II Express Briefs* **2019**, *66*, 1381–1385. [[CrossRef](#)]
25. Wang, H.; Zhan, D.; Wu, X.; He, S. Dynamics of a fractional-order Colpitts oscillator and its FPGA implementation. *Eur. Phys. J. Special Topics* **2022**. doi: [[CrossRef](#)]
26. Yang, Q.; Chen, D.; Zhao, T.; Chen, Y. Fractional calculus in image processing: A review. *Fract. Calc. Appl. Anal.* **2016**, *19*, 1222–1249. [[CrossRef](#)]
27. Kelley, W.G.; Peterson, A.C. *Difference Equations: An Introduction with Applications*; Academic Press: Cambridge, MA, USA, 2001.
28. Bruzzone, L.; Prieto, D.F. Automatic analysis of the difference image for unsupervised change detection. *IEEE Trans. Geosci. Remote Sens.* **2000**, *38*, 1171–1182. [[CrossRef](#)]
29. Qu, L.; Lin, J. A difference resonator for detecting weak signals. *Measurement* **1999**, *26*, 69–77. [[CrossRef](#)]
30. Bai, D.; Wang, G. A Memristive Chaotic Mapping Based on FPGA. *J. Hangzhou Dianzi Univ.* **2013**, *33*, 9–12.
31. He, S.; Sun, K.; Peng, Y.; Wang, L. Modeling of discrete fracmemristor and its application. *AIP Adv.* **2020**, *10*, 015332. [[CrossRef](#)]
32. Peng, Y.; Sun, K.; He, S. A discrete memristor model and its application in Hénon map. *Chaos Solitons Fractals* **2020**, *137*, 109873. [[CrossRef](#)]
33. Peng, Y.; He, S.; Sun, K. A higher dimensional chaotic map with discrete memristor. *AEU-Int. J. Electron. Commun.* **2021**, *129*, 153539. [[CrossRef](#)]
34. Bao, B.; Li, H.; Wu, H.; Zhang, X.; Chen, M. Hyperchaos in a second-order discrete memristor-based map model. *Electron. Lett.* **2020**, *56*, 769–770. [[CrossRef](#)]
35. Bao, H.; Hua, Z.; Li, H.; Chen, M.; Bao, B. Discrete Memristor Hyperchaotic Maps. *IEEE Trans. Circuits Syst. I Regul. Pap.* **2021**, *68*, 4534–4544. [[CrossRef](#)]
36. Bao, H.; Hua, Z.; Li, H.; Chen, M.; Bao, B.C. Memristor-based hyperchaotic maps and application in AC-GANs. *IEEE Trans. Ind. Informatics* **2021**, *18*, 5297–5306. [[CrossRef](#)]
37. Xu, Q.; Ju, Z.; Ding, S.; Feng, C.; Chen, M.; Bao, B. Electromagnetic induction effects on electrical activity within a memristive Wilson neuron model. *Cogn. Neurodyn.* **2022**, 1–11. [[CrossRef](#)]
38. Fu, L.; He, S.; Wang, H.; Sun, K. Simulink modeling and dynamic characteristics of discrete memristor chaotic system. *Acta Phys. Sin.-Chin. Ed.* **2022**, *71*, 030501. [[CrossRef](#)]
39. Atici, F.M.; Eloe, P.W. A transform method in discrete fractional calculus. *Int. J. Differ. Equ.* **2007**, *2*, 165–176.
40. Holm, M. Sum and difference compositions in discrete fractional calculus. *Cubo (Temuco)* **2011**, *13*, 153–184. [[CrossRef](#)]
41. Chua, L.; Kang, S.M. Memristive devices and systems. *Proc. IEEE* **1976**, *64*, 209–223. [[CrossRef](#)]
42. Abdeljawad, T.; Baleanu, D. Fractional Differences and Integration by Parts. *J. Comput. Anal. Appl.* **2011**, *13*, 574–582.

43. Huang, L.; Wang, L.; Shi, D. Discrete fractional order chaotic systems synchronization based on the variable structure control with a new discrete reaching-law. *IEEE/CAA J. Autom. Sin.* **2016**, 1–7. [[CrossRef](#)]
44. Wu, X.; He, S.; Tan, W.; Wang, H. From Memristor-Modeled Jerk System to the Nonlinear Systems with Memristor. *Symmetry* **2022**, *14*, 659. [[CrossRef](#)]
45. Wu, F.; Gao, R.; Liu, J.; Li, C. New fractional variable-order creep model with short memory. *Appl. Math. Comput.* **2020**, *380*, 125278. [[CrossRef](#)]
46. Wu, G.; Luo, M.; Huang, L.; Banerjee, S. Short memory fractional differential equations for new memristor and neural network design. *Nonlinear Dyn.* **2020**, *100*, 3611–3623. [[CrossRef](#)]
47. Wu, J.; Wang, G.; Qiu, R. Design and implementation of digital simulator for memristor. *J. Hangzhou Dianzi Univ. Nat. Sci.* **2018**, *38*, 1–6.
48. Delgado-Bonal, A.; Marshak, A. Approximate entropy and sample entropy: A comprehensive tutorial. *Entropy* **2019**, *21*, 541. [[CrossRef](#)] [[PubMed](#)]
49. Yan, B.; He, S.; Sun, K. Design of a network permutation entropy and its applications for chaotic time series and EEG signals. *Entropy* **2019**, *21*, 849. [[CrossRef](#)]