



## Article

# CORDIC-Based FPGA Realization of a Spatially Rotating Translational Fractional-Order Multi-Scroll Grid Chaotic System

Wafaa S. Sayed <sup>1</sup>, Merna Roshdy <sup>2</sup>, Lobna A. Said <sup>2</sup>, Norbert Herencsar <sup>3,\*</sup> and Ahmed G. Radwan <sup>1,4</sup>

<sup>1</sup> Engineering Mathematics and Physics Department, Faculty of Engineering, Cairo University, Giza 12613, Egypt

<sup>2</sup> Nanoelectronics Integrated Systems Center, Nile University, Giza 12588, Egypt

<sup>3</sup> Department of Telecommunications, Faculty of Electrical Engineering and Communication, Brno University of Technology, Technicka 12, 616 00 Brno, Czech Republic

<sup>4</sup> School of Engineering and Applied Sciences, Nile University, Giza 12588, Egypt

\* Correspondence: herencsn@ieee.org

**Abstract:** This paper proposes an algorithm and hardware realization of generalized chaotic systems using fractional calculus and rotation algorithms. Enhanced chaotic properties, flexibility, and controllability are achieved using fractional orders, a multi-scroll grid, a dynamic rotation angle(s) in two- and three-dimensional space, and translational parameters. The rotated system is successfully utilized as a Pseudo-Random Number Generator (PRNG) in an image encryption scheme. It preserves the chaotic dynamics and exhibits continuous chaotic behavior for all values of the rotation angle. The Coordinate Rotation Digital Computer (CORDIC) algorithm is used to implement rotation and the Grünwald–Letnikov (GL) technique is used for solving the fractional-order system. CORDIC enables complete control and dynamic spatial rotation by providing real-time computation of the sine and cosine functions. The proposed hardware architectures are realized on a Field-Programmable Gate Array (FPGA) using the Xilinx ISE 14.7 on Artix 7 XC7A100T kit. The Intellectual-Property (IP)-core-based implementation generates sine and cosine functions with a one-clock-cycle latency and provides a generic framework for rotating any chaotic system given its system of differential equations. The achieved throughputs are 821.92 Mbits/s and 520.768 Mbits/s for two- and three-dimensional rotating chaotic systems, respectively. Because it is amenable to digital realization, the proposed spatially rotating translational fractional-order multi-scroll grid chaotic system can fit various secure communication and motion control applications.

**Keywords:** chaos; CORDIC; encryption; fractional-order systems; Grünwald–Letnikov



**Citation:** Sayed, W.S.; Roshdy, M.; Said, L.A.; Herencsar, N.; Radwan, A.G. CORDIC-Based FPGA Realization of a Spatially Rotating Translational Fractional-Order Multi-Scroll Grid Chaotic System. *Fractal Fract.* **2022**, *6*, 432. <https://doi.org/10.3390/fractalfract6080432>

Academic Editor: Manuel Duarte Ortigueira

Received: 27 June 2022

Accepted: 2 August 2022

Published: 7 August 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

Fractional calculus, the non-integer counterpart of the classical integer calculus, has a relatively long history in theory, rather than applications [1,2]. It is known for describing a real object more accurately since it includes memory effects. However, the solution and implementation of fractional-order systems started to flourish only a few decades ago with the advances in digital computers and digital realization technologies [3].

In addition, fractional orders provide more controllability of the governing mathematical relations. For example, in their fractional-order form, chaotic systems exhibit the same aperiodicity, ergodicity, randomness, and sensitivity properties as their integer-order counterparts [1,2]. Both types of chaotic systems were employed in various applications such as encryption [4,5], synchronization [6,7], control [8], secure communication [9], modeling of electronic devices [10], diseases [11,12], and biomedical applications [13]. Most research in this field extended classical single- and double-scroll chaotic systems to the fractional-order domain. However, fractional-order delayed, hyperchaotic, and multi-scroll systems appeared less frequently [1].

Multi-scroll chaotic systems, especially those with grid scroll attractors, exhibit interesting and more complex topologies than systems with fewer scrolls. Various methods of multi-scroll generation have been presented, such as parameter switching, saturated sequence, sawtooth, piecewise-linear, step wave, hysteresis series, switching, sine, and hyperbolic tangent signals [14]. A fractional-order multi-scroll chaotic system with a staircase nonlinearity was formed in [15] by time-shifted versions of the signum function. Fractional-order multi-directional multi-scroll grid chaotic attractors utilized hysteresis [16,17] and staircase functions [18]. In [19], similar systems were generated using nonlinear state feedback controllers with a duality-symmetric multi-segment quadratic function. A piecewise-linear function [20] and a combination of the saturated nonlinear function and the stair nonlinear function series [21] were also utilized for similar purposes. Recently, there has been increasing interest in presenting, analyzing, and implementing integer- [22–26] and fractional-order [27–29] multi-scroll chaotic systems.

For chaotic systems to take part in real-world applications, implementations that generate the chaotic signal are required rather than the computer-simulated numerical form [30]. Several new chaotic systems were implemented in a microcontroller-based hardware platform [31–33]. The digital hardware FPGAs, especially when using fixed-point registers, provide many advantages, including, but not limited to the simple design, fast prototyping, and reduced hardware cost. Additionally, FPGAs provide high speed, reliability, programmability, reconfigurability, reproducibility, and immunity to noise. These advantages strongly encourage utilizing FPGAs for fractional-order chaotic systems' implementation [30,34], as long as accuracy and dynamical degradation limitations are considered in numerical approximations, bit representation, and precision decisions [30]. FPGA implementations of integer-order multi-scroll chaotic systems [35] have been more frequently presented than fractional-order ones. This scarcity in implementations applies to other types of fractional-order systems as well [30], and there is a growing need for researchers to reach high throughputs and low latency for their FPGA implementations. To overcome the challenges of the non-locality property of the fractional calculus operators and their resources and memory requirements, the hardware implementation of the Grünwald–Letnikov (GL) fractional-order operator was presented in [36]. In addition, the hardware of fractional-order chaotic systems was implemented with flexibility in the setup of the values of the parameters without changing the architecture [34]. Among [15–21], only [20,21] provided analog circuit implementation.

A relatively newer approach employed transformations of chaotic systems for different purposes [7,31,37–39]. Translation or offset boosting is achieved using additive parameters and was presented for integer-order [40–43] and fractional-order chaotic systems [44,45]. Rotation, suitable for multi-scroll generation and motion control applications, was presented for integer-order [37] and fractional-order [7] classical chaotic systems. It was proven to preserve the chaotic dynamics independent of the value and variation of the rotation angle [7,37]. Applications of these rotated chaotic systems and their performance in pseudo-random number generation and encryption were discussed in [7]. A cosine transform [31] was applied to chaotic maps for encryption applications. Three grid multi-scroll chaotic attractors based on a one-dimensional multi-scroll Chua system were presented in [46]. The digital implementations of the integer-order systems presented in [37,46] used constant values to replace the sine and cosine terms at design time instead of implementing these functions. The digital design does not provide automation or control capability for the value of rotation angle, such that it can be specified during simulation or real-time application.

The accompanying complicated simulation and implementation comprise the main challenge facing these different approaches to novel and generalized chaotic systems formation [47]. Few researchers presented analog implementations of chaotic systems, including trigonometric and/or hyperbolic functions [48]. However, digital implementation has many advantages, as previously listed. CORDIC is an iterative method to calculate elementary functions such as trigonometric and hyperbolic using add and shift operations. It overpasses other methods, including multiplication and division, such as the Taylor

series and lookup table, which suffer from increased hardware resources and memory requirements. CORDIC was infrequently employed in realizing the rotation of chaotic attractors [39], fractals with trigonometric and exponential nonlinearities [49], and in a reconfigurable manner for a system with several types of nonlinearities [50] on an FPGA. CORDIC-based blocks were also infrequently employed for the hardware approximation of activation functions in Artificial-Neural-Network (ANN)-based chaotic generators [51,52].

This paper proposes a spatially rotating translational fractional-order multi-scroll grid chaotic system and its hardware realization. It employs different approaches to achieve extra controllability and complex, chaotic behavior. First, a multi-scroll grid chaotic system is extended to the fractional-order domain. Then, this system is enriched with rotation angle and translational parameters, which can be dynamic. Two-dimensional and three-dimensional rotations are presented and implemented, where CORDIC is utilized in rotation and the compact representation of GL, which was presented in [36], is utilized for the fractional-order system. An image encryption application utilizing the proposed system as a PRNG is presented, and various evaluation criteria validate its good performance. Generic digital realization of the proposed system is presented for two- and three-dimensional rotations with automated control of real-time dynamic parameters. Experimental results exhibit accuracy and efficiency, and the proposed approach is advantageous compared to recent related works.

The rest of the paper is organized as follows. Section 2 reviews the mathematical background of the utilized numerical methods and algorithms. Section 3 explains the proposed translation and rotation transformation to obtain spatially rotating chaotic translational attractors. In addition, it extends a multi-scroll grid chaotic attractor to the fractional-order domain and uses it to validate the proposed transformation in two- and three-dimensional spaces. Section 4 presents an image encryption application for the proposed systems and validates its good performance by various evaluation criteria. Section 5 presents the hardware design and implementation of the two- and three-dimensional rotation algorithms for the fractional-order system based on the CORDIC algorithm and compact GL. Section 6 presents the experimental results of the digital realization showing good accuracy, hardware resource utilization, and efficiency. In addition, it compares the proposed system and hardware realization to recent related works. Finally, Section 7 concludes the main contributions of the paper and suggests future work directions.

## 2. Mathematical Background

### 2.1. Numerical Solution of Fractional-Order Systems

Grünwald–Letnikov (GL) established a general form to solve the fractional-order derivative [1] given by:

$$D_{\alpha}^t f(t) = \lim_{h \rightarrow 0} \frac{1}{h^{\alpha}} \sum_{j=0}^{\lfloor \frac{t-\alpha}{h} \rfloor} (-1)^j \binom{\alpha}{j} f(t-jh), \quad (1)$$

where  $h$  is the time step of calculation,  $\alpha$  is the fractional-order parameter, and  $(-1)^j \binom{\alpha}{j}$  are binomial coefficients. The general fractional-order differential equation using GL can be described as follows:

$$\begin{aligned} D_{\alpha}^t x(t) &= f(x(t), t), \\ x(t_k) &= f(x(t_{k-1}), t_k) h^{\alpha} - \sum_{j=1}^k c_j^{(\alpha)} x(t_{k-j}), \\ c_j^{(\alpha)} &= \left(1 - \frac{1+\alpha}{j}\right) c_{j-1}^{(\alpha)}, \quad j = 1, 2, 3, \dots, \quad c_0^{(\alpha)} = 1, \end{aligned} \quad (2)$$

which is composed of two terms. First, the aggregation or the GL term and, second, the system description term, which is multiplied by the time step  $h$  to the power of the fractional-order parameter  $\alpha$ .

### 2.2. CORDIC Algorithm

Currently, sine and cosine functions are essential in different applications, such as radio waves, digital signal processing [53], satellites, and software radio [54]. The trigonometric functions can be implemented using the Taylor series, lookup table, and CORDIC techniques.

Taylor series expansion is an old technique that contains many challenges. Take as an example the Taylor series expansion of the cosine:

$$\cos x = \sum_{n=0}^{\infty} \frac{(-1)^n}{(2n)!} x^{2n} = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \dots \tag{3}$$

Implementing such an equation requires a subtractor, multiplier, divider, and exponential, a complex circuit in hardware implementation. This technique requires much hardware resource and time to calculate the sine and cosine.

Furthermore, lookup table techniques depend on storing sine and cosine values. By increasing the lookup table size, the accuracy increases, which means it needs more memory, which is very expensive.

CORDIC calculates trigonometric and hyperbolic functions iteratively using add and shift operations. It is one of the digit-by-digit algorithms. The analog navigation computer was replaced by CORDIC on the B-58 bomber [55], as they need high accuracy and performance. Furthermore, Legendre polynomials were calculated based on the CORDIC algorithm, which was the first step of the International Geomagnetic Reference Field [56]. Besides, an adequate hardware solution for tasks of two mobile robots without applying any division was designed on an FPGA using the CORDIC algorithm [57]. Moreover, the CORDIC algorithm can be used in neural networks' VLSI design, as in [58]: there, it was used to design the activation and net functions. Accordingly, in hardware, it is more efficient to use the CORDIC algorithm as it minimizes the number of employed gates more than any other technique (lookup table, Taylor series).

The CORDIC algorithm has two working modes: rotation mode and vectoring mode. Volder first described rotation mode in 1959, and it was used to compute the rotation of a vector in a Cartesian coordinate system and evaluate the angle and length of the vector. In 1971, the CORDIC algorithm was extended to include multiplication and logarithm and exponential functions found in many applications such as matrix computation, digital image processing, and digital signal processing [59].

Rotation mode is used for calculating the sine and cosine by rotating an initial unit vector. The algorithm rotates the coordinate system by a series step clockwise ( $-\theta$ ) or anti-clockwise ( $+\theta$ ), as shown in Figure 1. The CORDIC algorithm iteration equations can be given by:

$$\begin{aligned} x_{i+1} &= x_i - \text{sign}(z_i)2^{-i}y_i, \\ y_{i+1} &= y_i + \text{sign}(z_i)2^{-i}x_i, \\ z_{i+1} &= z_i - \text{sign}(z_i) \arctan(2^{-i}), \end{aligned} \tag{4}$$

and the expected final output of the iteration equations:

$$\begin{aligned} x_n &= K_n(x_0 \cos z_0 - y_0 \sin z_0), \\ y_n &= K_n(y_0 \cos z_0 + x_0 \sin z_0), \\ z_n &= 0, \end{aligned} \tag{5}$$

where  $K_n$  is a scale factor calculated as follows:

$$K_n = \prod_{i=0}^{n-1} \sqrt{1 + 2^{-2i}}. \tag{6}$$

It represents the increase in the rotating vector during the micro-rotations, as it is not a pure rotation, but a rotation expansion. After a certain number of micro-rotations, the

scale factor  $K_n$  is a constant number approximated to 1.64676 [56]. In other words, as the number of iterations tends to infinity, the scale factor will be a constant number.

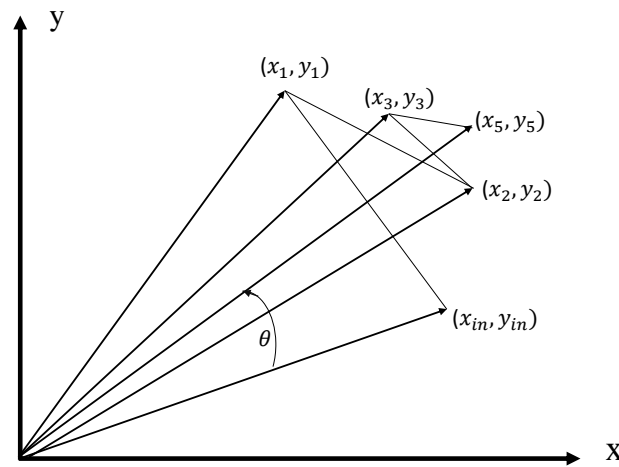


Figure 1. Rotation of a vector using microrotations.

Since the calculation of sin and cos depends on the rotation of unit vector ( $x_0 = 1, y_0 = 0$ ), the outputs will be as follows:

$$\begin{aligned} x_n &= K_n x_0 \cos z_0, \\ y_n &= K_n x_0 \sin z_0, \\ z_n &= 0, \end{aligned} \tag{7}$$

however, to avoid division in the hardware implementation, the initialization of  $x_0$  is modified by  $\frac{1}{K_n}$ .

The CORDIC algorithm is an iterative method; therefore, it is required to know how many iterations are needed to obtain the optimum values of  $\sin(\theta)$  and  $\cos(\theta)$ . The final  $z_n$  is the desired angle; if it reaches zero, as in (7c), the desired angle is achieved. Table 1 shows the results after testing the CORDIC algorithm. Therefore, it was chosen in the proposed algorithm to use the number of iterations as 15 to obtain the highest accuracy without wasting the resources of the FPGA.

Table 1. CORDIC algorithm results  $z_n$ .

$n$	$\theta = 45^\circ$	$\theta = 90^\circ$
13	0.0044	-0.0028
14	-0.0026	0.0042
15	0.000858884	0.00074059
16	0.00088966	-0.001

### 3. Towards a Spatially Rotating Translational Fractional-Order Multi-Scroll Grid Chaotic System

#### 3.1. Fractional-Order Multi-Scroll Grid Chaotic System

A multi-scroll  $2 \times 2$  grid chaotic system was presented in [60]. Its fractional-order counterpart is given by:

$$D^\alpha X = AX + B\Phi(X), \tag{8}$$

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -a & -a & -a \end{bmatrix}, \quad B = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & a \end{bmatrix}, \quad \Phi = \begin{bmatrix} f_1(y) \\ 0 \\ f_2(x) \end{bmatrix}, \tag{9}$$

$$f_1(y) = g(y), \quad f_2(x) = 2g(x), \quad g(\tau) = \begin{cases} 0 & \tau \geq 0.5 \\ -1 & \tau < 0.5 \end{cases} \quad (10)$$

Hence, the final system equations are given by:

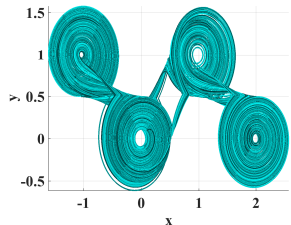
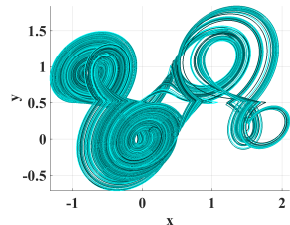
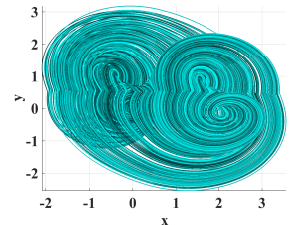
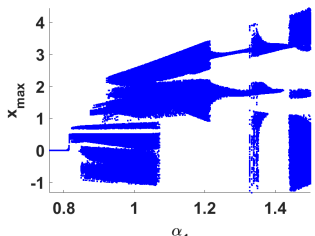
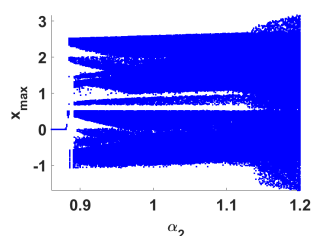
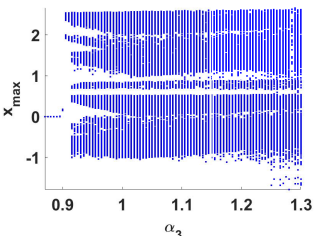
$$\begin{aligned} D^{\alpha_1}x &= y - g(y), \\ D^{\alpha_2}y &= z, \\ D^{\alpha_3}z &= -a(x + y + z - 2g(x)), \end{aligned} \quad (11)$$

and it is solved using (2) as follows:

$$\begin{aligned} x_{i+1} &= (y_i - g(y_i))h^{\alpha_1} - \sum_{j=1}^i c_j^{\alpha_1} x_{i-j+1}, \\ y_{i+1} &= z_i h^{\alpha_2} - \sum_{j=1}^i c_j^{\alpha_2} y_{i-j+1}, \\ z_{i+1} &= (-a(x_i + y_i + z_i - 2g(x_i)))h^{\alpha_3} - \sum_{j=1}^i c_j^{\alpha_3} z_{i-j+1}. \end{aligned} \quad (12)$$

The projections of the strange attractors and bifurcation diagrams shown in Table 2 show that the system exhibits chaotic behavior at different combinations and against ranges of the fractional orders at  $a = 0.81$ .

**Table 2.**  $x - y$  projections and bifurcation diagrams of the solution of (12).

$(\alpha_1, \alpha_2, \alpha_3) =$		
$(1, 0.95, 1)$	$(0.87, 1.15, 0.95)$	$(1.1, 1.1, 1.1)$
		
$(\alpha_1, 0.95, 1)$	$(1, \alpha_2, 1)$	$(1, 0.95, \alpha_3)$
		

### 3.2. Two-Dimensional Translational Rotating System

Rotation can be applied for any pair of the phase space dimensions. For example, if  $\theta$  is the rotation angle in the  $x - y$  plane or about the  $z$ -axis, the equations of the corresponding variables  $u$  and  $v$  in terms of  $x$  and  $y$  are given by:

$$\begin{aligned} D^{\alpha_1}u &= \cos \theta D^{\alpha_1}x + \sin \theta D^{\alpha_1}y, \\ D^{\alpha_2}v &= -\sin \theta D^{\alpha_2}x + \cos \theta D^{\alpha_2}y, \\ D^{\alpha_3}w &= D^{\alpha_3}z. \end{aligned} \quad (13)$$

The fractional derivatives of  $x, y,$  and  $z$  in (13) are replaced by the functions on the right-hand side of the chaotic equations to have a rotation. The inverse transformation represents the equations in terms of  $u, v,$  and  $w$ . The resulting system is given by:

$$\begin{aligned}
 t1_i &= \cos(\theta)u_i - \sin(\theta)v_i + s_u, \\
 t2_i &= \sin(\theta)u_i + \cos(\theta)v_i + s_v, \\
 u_{i+1} &= (\cos(\theta)(t2_i - g(t2_i)) + \sin(\theta)(w_i))h^{\alpha_1} - \sum_{j=1}^i c_j^{\alpha_1}u_{i-j+1}, \\
 v_{i+1} &= (-\sin(\theta)(t2_i - g(t2_i)) + \cos(\theta)(w_i))h^{\alpha_2} - \sum_{j=1}^i c_j^{\alpha_2}v_{i-j+1}, \\
 w_{i+1} &= (-a(t1_i + t2_i + w_i - 2g(t1_i)))h^{\alpha_3} - \sum_{j=1}^i c_j^{\alpha_3}w_{i-j+1}.
 \end{aligned}
 \tag{14}$$

The terms  $s_u$  and  $s_v$  are translational parameters that translate or shift the attractor along the  $u$ - and  $v$ -axes, respectively. The system achieves rotation only when  $s_u = s_v = 0$ . Figure 2 shows examples of the two-dimensional rotating translational system with generally dynamic parameters at  $(\alpha_1, \alpha_2, \alpha_3) = (1, 0.95, 1)$ . From Figure 2a, it can be inferred that the translation parameters  $s_u$  and  $s_v$  perform their role in comparison with the original strange attractor in Table 2. Figure 2b uses dynamic translation parameters, which change values after a specific duration. Figure 2c shows the generation of more multiple scrolls using the dynamic rotation angle generated via the same procedure. Two-dimensional rotation about the  $y$ - or  $x$ -axes can be achieved similarly. For instance, in the  $x - z$  plane.

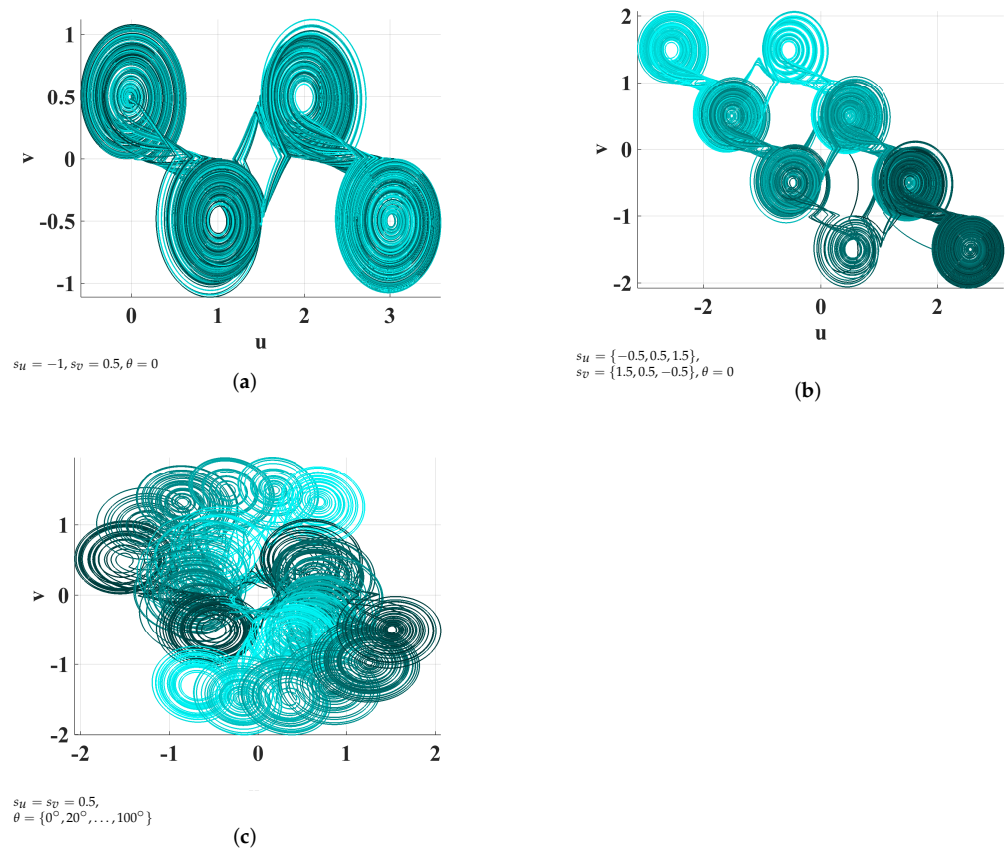


Figure 2. Two-dimensional (a) static translation, (b) dynamic translation, and (c) dynamic rotation.

$$\begin{aligned}
 t1_i &= \cos(\theta)u_i - \sin(\theta)w_i + s_u, \\
 t2_i &= \sin(\theta)u_i + \cos(\theta)w_i + s_w, \\
 u_{i+1} &= (\cos(\theta)(v_i - g(v_i)) - a \sin(\theta)(t1_i + v_i + t2_i - 2g(t1_i)))h^{\alpha_1} - \sum_{j=1}^i c_j^{\alpha_1} u_{i-j+1}, \\
 v_{i+1} &= (t2_i)h^{\alpha_2} - \sum_{j=1}^i c_j^{\alpha_2} v_{i-j+1}, \\
 w_{i+1} &= (-\sin(\theta)(v_i - g(v_i)) - a \cos(\theta)(t1_i + v_i + t2_i - 2g(t1_i)))h^{\alpha_3} - \sum_{j=1}^i c_j^{\alpha_3} w_{i-j+1}.
 \end{aligned}
 \tag{15}$$

### 3.3. Three-Dimensional Rotating System

Two-dimensional rotations can be extended to three-dimensional rotations by constructing elementary three-dimensional rotation matrices, which perform rotations individually about the three coordinate axes  $z$ ,  $y$ , and  $x$  by angles  $\theta_1$ ,  $\theta_2$ , and  $\theta_3$ , respectively. The rotations about the two other axes can be derived similarly and applied to create a general composite three-dimensional rotation as follows:

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = R \begin{bmatrix} x \\ y \\ z \end{bmatrix}, \tag{16}$$

$$R = \begin{bmatrix} \cos \theta_1 \cos \theta_2 & \cos \theta_3 \sin \theta_1 + \cos \theta_1 \sin \theta_3 \sin \theta_2 & \sin \theta_1 \sin \theta_3 - \cos \theta_1 \cos \theta_3 \sin \theta_2 \\ -\cos \theta_2 \sin \theta_1 & \cos \theta_1 \cos \theta_3 - \sin \theta_1 \sin \theta_3 \sin \theta_2 & \cos \theta_1 \sin \theta_3 + \cos \theta_3 \sin \theta_1 \sin \theta_2 \\ \sin \theta_2 & -\cos \theta_2 \sin \theta_3 & \cos \theta_3 \cos \theta_2 \end{bmatrix}. \tag{17}$$

This procedure is used to obtain the equations of the rotating chaotic system, where  $R$  is an orthogonal matrix, through the Algorithm 1.

---

**Algorithm 1:** Three-dimensional rotation algorithm.

---

- Construct the rotation matrix  $R$  and  $R^T$ .
  - Find  $X_i = R^T U_i$ .
  - Apply GL method to solve (12), and find  $X_{i+1}$ .
  - Find  $U_{i+1} = R X_{i+1}$ .
- 

Figure 3 shows an example of the three-dimensional rotating system at rotation angles  $(\theta_1, \theta_2, \theta_3) = (90^\circ, 90^\circ, 45^\circ)$ .

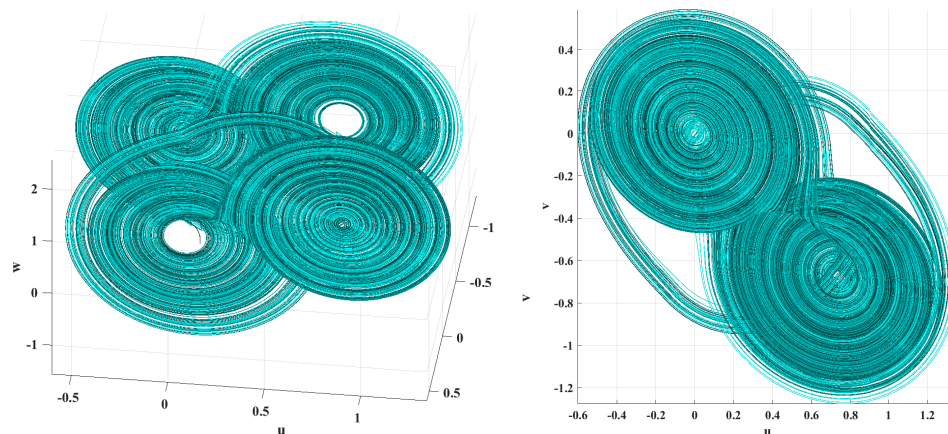
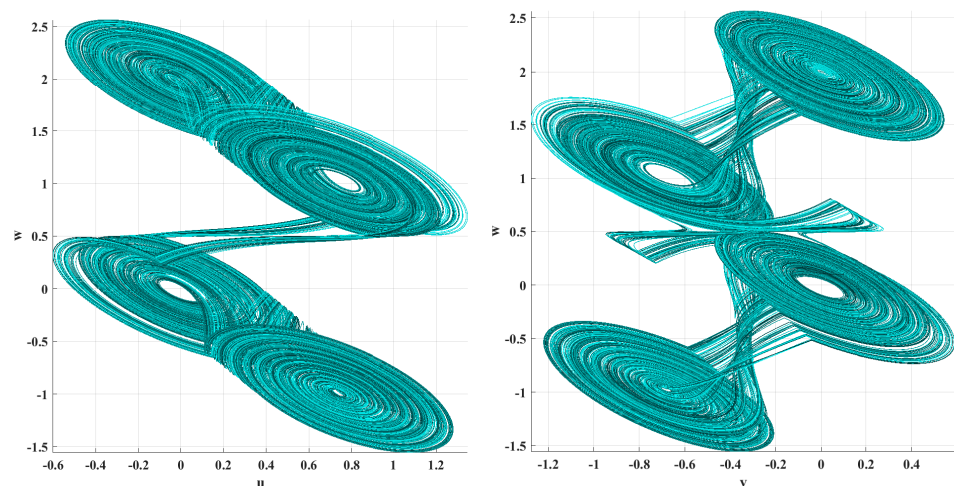


Figure 3. Cont.





**Figure 3.** Three-dimensional rotation example at  $(\theta_1, \theta_2, \theta_3) = (90^\circ, 90^\circ, 45^\circ)$ .

#### 4. Encryption Application

Rotated chaotic systems were shown to theoretically preserve the chaotic dynamics adding extra controllability and sensitivity through stability analysis, bifurcations, and Maximum Lyapunov Exponents (MLEs) [7,37]. This section shows that the same result is valid for practical applications by presenting an image encryption application using the rotated fractional-order multi-scroll grid chaotic system. The advantage of the rotation angle as a system parameter is that, unlike the case for chaotic system parameters, the system remains chaotic and does not drift to stable, periodic, or divergent responses outside a specific range.

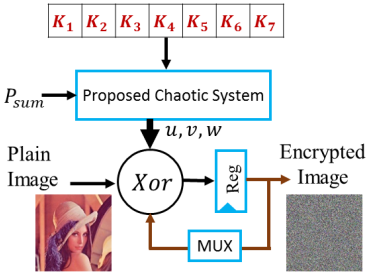
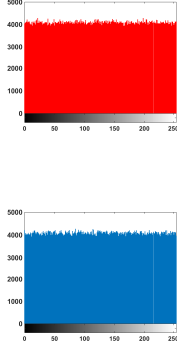
Table 3 shows a simple substitution-based stream cipher scheme with feedback employed for symmetric-key encryption. It uses the rotated fractional-order multi-scroll grid chaotic system as a PRNG. The encryption key is composed of seven sub-keys (18, 18, 18, 18, 18, 19, 19 bits) with a total number of 128 bits that resist brute force attacks where the hacker tries all possible combinations of the encryption key as specified by the Advanced Encryption Standard (AES) [61]. The value  $P_{sum} = R_{sum} + G_{sum} + B_{sum}$  represents the input-dependent term, which enhances the resistance to different attacks, where  $R_{sum}$ ,  $G_{sum}$ , and  $B_{sum}$  are the sums of the red, green, and blue channels of the input image, respectively. The initial system values, rotation angle, and fractional orders are computed from the key, for example:

$$\begin{aligned} u_0 &= u_{fix} + K_1 \times 2^{-24} + \text{mod}(P_{sum}, 10)/1000, \\ \theta &= \theta_{fix} + K_4 \times 2^{-24} + \text{mod}(P_{sum}, 10)/1000, \\ \alpha &= \alpha_{fix} + K_5 \times 2^{-24} + \text{mod}(P_{sum}, 10)/1000, \end{aligned} \quad (18)$$

where the fixed parts are set to values within the ranges corresponding to chaotic behavior.

After discarding a few transient iterations, the outputs  $u$ ,  $v$ , and  $w$  of the chaotic generator are multiplied by a scaling factor of  $10^9$  to be suitable for conversion to an integer value represented in 64 bits. The original color image is decomposed into three channels: red, green, and blue. Each component is XORed with the least significant 8 bits of the outputs of the chaotic generator, respectively, XORed together with a feedback element from a channel of the previously encrypted pixel selected by the least-significant bits of its channels. In the decryption scheme, all operations are reversed.

**Table 3.** Encryption scheme and its performance analysis.

Encryption Scheme		Histograms	NIST		
			Test	PV	PP
			1	✓	0.979
			2	✓	0.958
			3	✓	1
			4	✓	0.917
			5	✓	1
			6	✓	1
			7	✓	1
			8	✓	0.989
			9	✓	1
Horz. corr.	Vert. corr.	Diag. corr.	10	✓	1
$5.5465 \times 10^{-4}$	$2.4273 \times 10^{-4}$	$3.8249 \times 10^{-5}$	11	✓	1
Key Sens.	MSE ( $\times 10^3$ )	Entropy	12	✓	0.969
( $\Delta K_4$ )	8.9265	7.9998	13	✓	0.99
DA	NPCR (%)	UACI (%)	14	✓	1
	99.5607	33.4624	15	✓	0.958

✓: successful

The encrypted image shown in Table 3 is completely random and noisy. Table 3 validates the good performance of the encryption scheme using various perceptual and statistical evaluation criteria. The histogram reveals a uniform intensity distribution compared to the original nonuniform distribution of the plain image. The pixel correlation coefficient  $\rho$  between two adjacent pixels  $(x(i, j)$  and  $y(i, j))$  is given by:

$$\rho = \frac{cov(x, y)}{\sqrt{D(x)}\sqrt{D(y)}} \tag{19}$$

where  $cov(x, y) = \frac{1}{S} \sum_{i=1}^S \left( x_i - \frac{1}{S} \sum_{j=1}^S x_j \right) \left( y_i - \frac{1}{S} \sum_{j=1}^S y_j \right)$ ,  $D(x) = \frac{1}{S} \sum_{i=1}^S \left( x_i - \frac{1}{S} \sum_{j=1}^S x_j \right)^2$ ,  $S = M \times N$ , and  $M$  and  $N$  are the height and width of the image, respectively. The correlation coefficients between pixels of the encrypted image approach zero. More advanced statistical tests are provided by the NIST statistical test suite [62], which is a statistical test suite for PRNGs and cryptographic applications. The following tests are listed: frequency, block frequency, cumulative sums, runs, longest run, rank, FFT, non-overlapping template, overlapping template, universal, approximate entropy, random excursions, random excursions variant, and serial and linear complexity, respectively. They are designed to examine the randomness characteristics of a sequence of bits by evaluating the P-Value distribution (PV) and the Proportion of Passing sequences (PP). The encrypted image successfully passes the tests.

The Mean-Squared Error (MSE) and entropy are used to ensure the randomness of the decrypted image using the wrong key with the Least Significant Bit (LSB) change in each sub-key and are given by:

$$MSE = \frac{1}{M \times N} \sum_{i=1}^N \sum_{j=1}^M (P(i, j) - D(i, j))^2 \tag{20}$$

$$Entropy = - \sum_{i=1}^{2^8} p(s_i) \log_2 p(s_i) \tag{21}$$

Here,  $P(i, j)$  is the original image pixel and  $D(i, j)$  is the wrongly decrypted image pixel, while  $p(s_i)$  is the probability of symbol  $s_i$ . A high MSE value indicates how far the wrongly decrypted image is from the plain image. The entropy value approaching 8 indicates the randomness and unpredictability of the encrypted image samples. An advantage of the encryption system is that perturbation in any parameter affects the three time series and, hence, the three channels, unlike encryption systems based on independent discrete maps for each channel, which require a special key design to overcome their limitation [63,64].

The scheme's sensitivity to plain-text and immunity against differential attacks is asserted through the Number of Pixel Change Rate (NPCR) and the Unified Average Changing Intensity (UACI), given by:

$$\text{NPCR} = \frac{1}{N \times M} \sum_{i=1}^N \sum_{j=1}^M D(i, j) \times 100, \quad (22)$$

$$\text{UACI} = \frac{1}{M \times N} \sum_{i=1}^N \sum_{j=1}^M \left| \frac{C_1(i, j) - C_2(i, j)}{255} \right| \times 100, \quad (23)$$

where  $D(i, j) = \begin{cases} 1, & C_1(i, j) \neq C_2(i, j) \\ 0, & C_1(i, j) = C_2(i, j) \end{cases}$ ,  $C_1$  is the ciphered pixel, and  $C_2$  is the ciphered pixel corresponding to a slightly modified original image. The values of the NPCR and UACI of the three channels are averaged over 20 trials in which one pixel in the original image is changed and found to approach the ideal values 99.60% and 33.46%, respectively [65–67].

## 5. FPGA Implementation

### 5.1. 2D Rotation Algorithm

The hardware implementation of rotating the fractional-order chaotic system in two dimensions is discussed in this section. Figure 4 presents the proposed algorithm's hardware architecture, which is a generic algorithm for rotating in the  $x - z$  plane. First, the block CORDIC is responsible for calculating and changing the values of  $\sin(\theta)$  and  $\cos(\theta)$  in real-time. Therefore, the angle  $\theta$  can be changed in real-time.

Figure 5 shows the hardware architecture of the CORDIC algorithm. To implement the proposed system, three registers each of 32 bits are used to store the values of *Cosine*, *Sine*, and *Angle* divided into 8 bits for the integer part and 24 bits for the fractional part. Additionally, a lookup table is used to store the values of  $\arctan(2^{-j})$ , where  $j$  is a counter.

The input is initialized with the desired angle, and the value is stored in the register *Angle*. Then, the *Cosine* and *Sine* registers are used to calculate the sine and cosine initialized by  $\frac{1}{K}$  and 0, where  $K$  is a scaling factor equal to 1.64676.

The registers *Cosine* and *Sine* each clock cycle are shifted by the counter  $j$ , where  $j$  counts from 0 to 14 every clock cycle. Then, by checking the sign of the *Angle* register, consequently, the ADD/SUB block in Figure 5 determines whether to add or subtract according to (4). After looping on this architecture 15 times (the number of iterations chosen to obtain the optimum results), the values of  $\sin(\theta)$  and  $\cos(\theta)$  will be ready in one clock cycle. To compute the summation of the fractional-order chaotic system in (12), the GL block is used. It is realized based on the previous work in [36], where the upper limit in the summation terms is replaced by a limited window size ( $L$ ) according to the short memory principle.

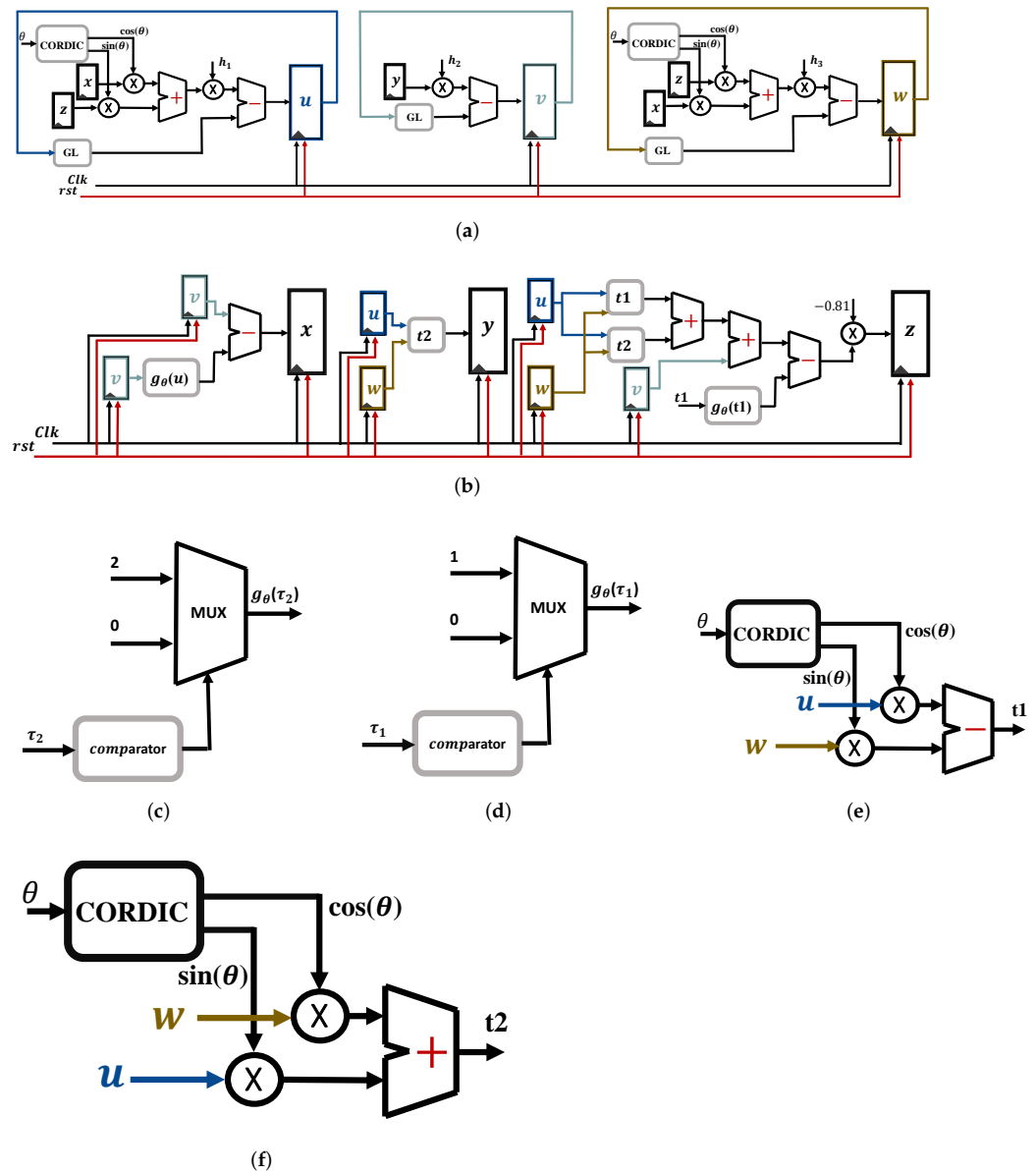


Figure 4. Fractional-order chaotic system 2D rotation: (a) fractional-order chaotic system after applying 2D rotation, (b) chaotic system, (c,d)  $g(\tau)$  architecture, and (e,f)  $t1$  and  $t2$  architectures.

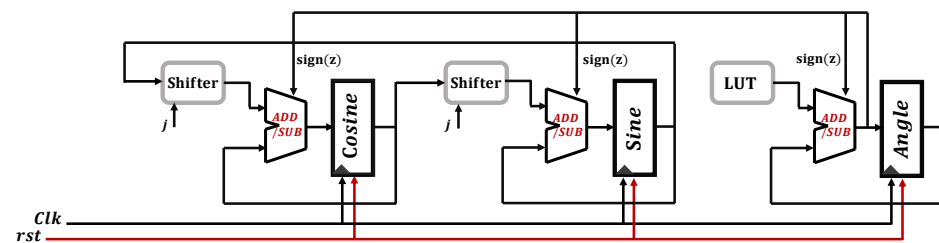
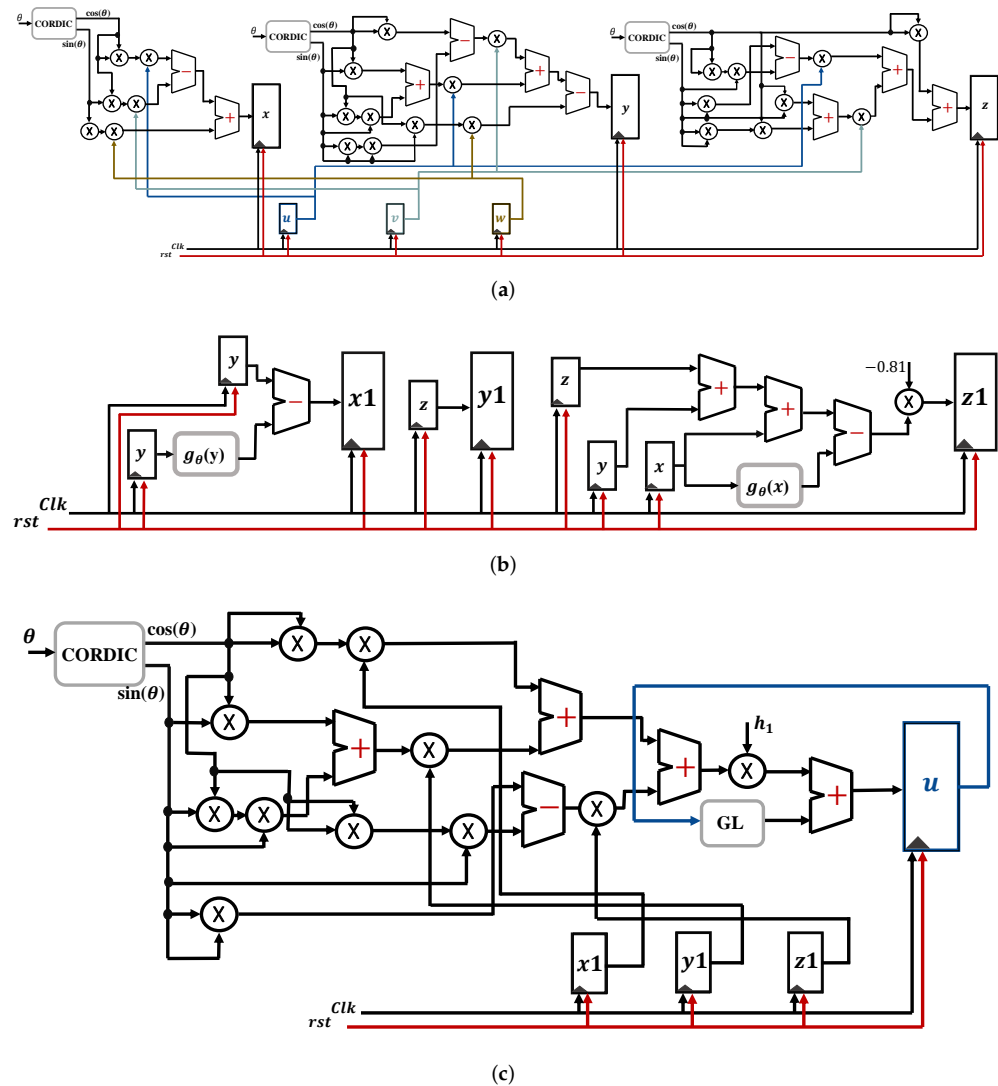


Figure 5. Architecture of the CORIDC algorithm.

Figure 4b shows the calculations of the three  $x, y$ , and  $z$  registers, which are used in calculating the final system registers  $u, v$ , and  $w$ .  $t1$  and  $t2$  are presented in Figure 4e,f. Furthermore,  $g_\theta(t1)$  and  $g_\theta(v)$  are presented in Figure 4c,d; both are comparators to the value of  $\theta$ , as shown in (10). After that, the three registers  $x, y$ , and  $z$  will be ready to be used as illustrated in Figure 4a. This algorithm is a generic one; by changing only the chaotic system Figure 4b, any chaotic system can be rotated in the  $x - z$  plane.

### 5.2. Three-Dimensional Rotation Algorithm

Figure 6 presents the hardware design following Algorithm 1, where it is a generic algorithm for rotating within the three planes  $x - z$ ,  $x - y$ , and  $z - y$ , individually or simultaneously, by changing only the equations of the chaotic attractor in Figure 6b. The 3D algorithm was verified for various systems to ensure its validity. This algorithm is discussed through three steps, as illustrated in Figure 6.



**Figure 6.** Fractional-order chaotic system 3D rotation: (a) first step, (b) second step, and (c) third step.

Figure 6a demonstrates the first step, which is the multiplication between matrices representing the inverse of matrix  $R$  (17), then calculating the three main states of the chaotic attractor present in Figure 6b, which shows the second step; the blocks  $g_\theta(x)$  and  $g_\theta(y)$  are demonstrated in Figure 4c,d. Then, the last and third steps, which are the multiplication of the matrices presented in (17), are realized as shown in Figure 6c, which models one of the three composite dimensional rotations  $u$ . The CORDIC block and GL block were discussed before in the 2D algorithm implementation.

## 6. Experimental Results and Discussion

The proposed algorithms were realized using the Verilog Hardware Description Language (HDL) with software simulation Xilinx ISE 14.7 and FPGA Artix-7 XC7A100T for hardware implementation. The software simulation Xilinx ISE results were verified by

plotting the outputs on MATLAB after extracting them into .dat files. An analog-to-digital converter, Pmod DAC 2, was used to display the chaotic attractor on an oscilloscope, where it allows resolution up to 1 mV. It has two channels, each 12 bit, and takes the output from the FPGA (12 bit) as an input and converts it to an analog signal to display on the oscilloscope. DPO 4104 is the digital oscilloscope that displays the data waveform with the four integrated analog input channels.

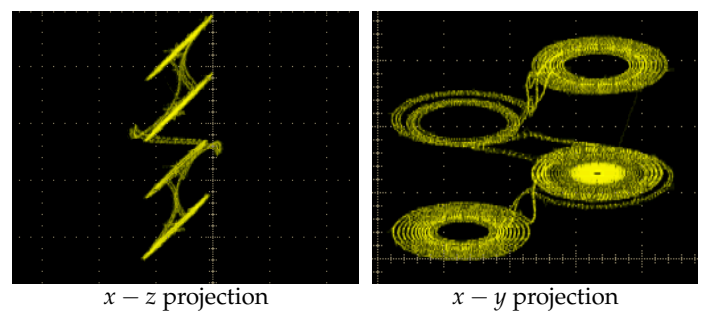
Table 4 also shows the experimental results on the oscilloscope, where the 2D rotation is performed in the  $x - z$  plane with the angle equal to  $90^\circ$ . The 3D rotation is performed in the  $x - y$  plane with angles  $\theta_1 = \theta_2 = \theta_3 = 90^\circ$ . Table 4 shows the hardware resources of the proposed algorithms, with system parameters  $u_1 = 0.1$ ,  $v_1 = 0.1$ ,  $w_1 = 0.1$ , step size  $h = 0.0625$ ,  $\alpha_1 = 1$ ,  $\alpha_2 = 0.9$ ,  $\alpha_3 = 1$ ,  $a = 0.81$ , and  $L = 20$ . The 2D rotation of the fractional-order chaotic attractor can be realized using the 3D algorithm. However, it needs more resources than the 2D algorithm.

The spatially rotating translational fractional-order multi-scroll grid chaotic system and its implementation based on the compact GL and CORDIC algorithms can enrich the fields of fractional chaotic dynamics and their applications. It is advantageous compared to previous related works that proposed multi-scroll chaotic systems and their hardware realizations. The rotation angle had a static value in [46], so a cascade of transformations was applied to achieve a circular grid. The angle variable (register) was set to a specific value during simulation (run)time and was not allowed to vary as time progressed. A dynamic rotation angle was employed in [37] to obtain an increased number of scrolls with a single transformation. However, neither work implemented the sine and cosine functions and only considered the conventional integer-order domain. In the fractional-order domain, the parameter values were still static. None of [37,46] assessed the performance of their proposed systems in PRNG or encryption applications. Our paper combines the fractional-order domain, multi-scroll grid attractors, and two-dimensional and three-dimensional rotation and translation transformations to obtain a chaotic system with controllable complex behavior. The proposed system was employed in an image encryption application that successfully passed the performance tests. To enable complete control and dynamic spatial rotation, it was necessary to have the real-time computation of the sine and cosine functions. A CORDIC-based algorithm was successfully designed and implemented with compact GL, yielding an FPGA realization that balances accuracy and efficiency. The proposed design is generic for rotating any other chaotic system.

**Table 4.** FPGA summary and experimental results for the fractional-order multi-scroll attractor.

Logic Utilization	2D Rotation	3D Rotation
No. of LUT	1833 out of 63,400 (2.8912%)	5636 out of 63,400 (8.8896%)
No. of slice registers	1091 out of 126,800 (0.8604%)	1106 out of 12,6800 (0.8722%)
Clock speed (MHz)	25.685	16.274
Throughput (Mbit/sec)	821.92	520.768

Oscilloscope results



## 7. Conclusions

This paper presented the FPGA realization of a fractional-order multi-scroll  $2 \times 2$  chaotic grid system, solved by the GL technique, dynamically translated and rotated in two and three dimensions using the CORDIC algorithm. The rotation algorithms were implemented as IP-cores to be applicable for any fractional-order chaotic system. The proposed algorithm implementation showed good performance with one-cycle latency. Furthermore, the proposed system provides enhanced complex multi-scroll grid attractor structures and controllability through static and dynamic parameters. It was successfully employed as a PRNG in an image encryption scheme with good performance. In addition, it can be used in other applications such as secure multimedia communication and robotic motion control due to its amenability to digital hardware realization. For future work, a three-dimensional rotating system can be employed in a switched chaotic encryption scheme, as it allows more degrees of freedom and different rotation axes.

**Author Contributions:** Conceptualization, W.S.S., L.A.S. and A.G.R.; methodology, L.A.S. and A.G.R.; software, W.S.S., M.R. and L.A.S.; validation, W.S.S., M.R. and L.A.S.; formal analysis, W.S.S. and A.G.R.; investigation, W.S.S. and N.H.; resources, N.H.; data curation, W.S.S. and M.R.; writing—original draft preparation, W.S.S., M.R., L.A.S. and A.G.R.; writing—review and editing, W.S.S., M.R., L.A.S. and A.G.R.; visualization, M.R., L.A.S. and N.H.; supervision, W.S.S., L.A.S. and A.G.R.; project administration, L.A.S. and N.H.; funding acquisition, L.A.S. All authors have read and agreed to the published version of the manuscript.

**Funding:** This paper is based on work supported by Science, Technology, and Innovation Funding Authority (STIFA) under Grant Number (#38161).

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** All the data are presented in the paper.

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

## References

- Petráš, I. *Fractional-Order Nonlinear Systems: Modeling, Analysis and Simulation*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2011.
- Elwy, O.; Abdelaty, A.; Said, L.; Radwan, A. Fractional calculus definitions, approximations, and engineering applications. *J. Eng. Appl. Sci.* **2020**, *67*, 1–30.
- Sabatier, J.; Agrawal, O.P.; Machado, J.T. *Advances in Fractional Calculus*; Springer: Berlin/Heidelberg, Germany, 2007; Volume 4.
- de la Fraga, L.G.; Torres-Pérez, E.; Tlelo-Cuautle, E.; Mancillas-López, C. Hardware implementation of pseudo-random number generators based on chaotic maps. *Nonlinear Dyn.* **2017**, *90*, 1661–1670. [[CrossRef](#)]
- Elsafty, A.H.; Tolba, M.F.; Said, L.A.; Madian, A.H.; Radwan, A.G. Enhanced hardware implementation of a mixed-order nonlinear chaotic system and speech encryption application. *AEU Int. J. Electron. Commun.* **2020**, *125*, 153347. [[CrossRef](#)]
- Razminia, A.; Baleanu, D. Complete synchronization of commensurate fractional order chaotic systems using sliding mode control. *Mechatronics* **2013**, *23*, 873–879. [[CrossRef](#)]
- Sayed, W.S.; Radwan, A.G. Generalized switched synchronization and dependent image encryption using dynamically rotating fractional-order chaotic systems. *AEU Int. J. Electron. Commun.* **2020**, *123*, 153268. [[CrossRef](#)]
- Caponetto, R. *Fractional Order Systems: Modeling and Control Applications*; World Scientific: Singapore, 2010; Volume 72.
- Kiani-B, A.; Fallahi, K.; Pariz, N.; Leung, H. A chaotic secure communication scheme using fractional chaotic systems based on an extended fractional Kalman filter. *Commun. Nonlinear Sci. Numer. Simul.* **2009**, *14*, 863–879. [[CrossRef](#)]
- Elwakil, A.S. Fractional-order circuits and systems: An emerging interdisciplinary research area. *IEEE Circuits Syst. Mag.* **2010**, *10*, 40–50. [[CrossRef](#)]
- Raza, A.; Rafiq, M.; Awrejcewicz, J.; Ahmed, N.; Mohsin, M. Dynamical analysis of coronavirus disease with crowding effect, and vaccination: A study of third strain. *Nonlinear Dyn.* **2022**, *107*, 3963–3982. [[CrossRef](#)]
- Raza, A.; Chu, Y.M.; Bajuri, M.Y.; Ahmadian, A.; Ahmed, N.; Rafiq, M.; Salahshour, S. Dynamical and nonstandard computational analysis of heroin epidemic model. *Results Phys.* **2022**, *34*, 105245. [[CrossRef](#)]
- Aghababa, M.P.; Borjkhani, M. Chaotic fractional-order model for muscular blood vessel and its control via fractional control scheme. *Complexity* **2014**, *20*, 37–46. [[CrossRef](#)]
- Lü, J.; Chen, G. Generating multiscroll chaotic attractors: theories, methods and applications. *Int. J. Bifurc. Chaos* **2006**, *16*, 775–858. [[CrossRef](#)]

15. Ahmad, W.M. Generation and control of multi-scroll chaotic attractors in fractional order systems. *Chaos Solitons Fractals* **2005**, *25*, 727–735. [[CrossRef](#)]
16. Deng, W.; Lü, J. Design of multidirectional multiscroll chaotic attractors based on fractional differential systems via switching control. *Chaos Interdiscip. J. Nonlinear Sci.* **2006**, *16*, 043120. [[CrossRef](#)] [[PubMed](#)]
17. Deng, W.; Lü, J. Generating multi-directional multi-scroll chaotic attractors via a fractional differential hysteresis system. *Phys. Lett. A* **2007**, *369*, 438–443. [[CrossRef](#)]
18. Deng, W. Generating 3-D scroll grid attractors of fractional differential systems via stair function. *Int. J. Bifurc. Chaos* **2007**, *17*, 3965–3983. [[CrossRef](#)]
19. Zhang, C.; Yu, S. Generation of multi-wing chaotic attractor in fractional order system. *Chaos Solitons Fractals* **2011**, *44*, 845–850. [[CrossRef](#)]
20. Chen, L.; Pan, W.; Wu, R.; Wang, K.; He, Y. Generation and circuit implementation of fractional-order multi-scroll attractors. *Chaos Solitons Fractals* **2016**, *85*, 22–31. [[CrossRef](#)]
21. Chen, L.; Pan, W.; Wu, R.; Machado, J.T.; Lopes, A.M. Design and implementation of grid multi-scroll fractional-order chaotic attractors. *Chaos Interdiscip. J. Nonlinear Sci.* **2016**, *26*, 084303. [[CrossRef](#)]
22. Yu, N.; Wang, Y.W.; Liu, X.K.; Xiao, J.W. 3D grid multi-wing chaotic attractors. *Int. J. Bifurc. Chaos* **2018**, *28*, 1850045. [[CrossRef](#)]
23. Escalante-González, R.J.; Campos-Cantón, E. A class of piecewise linear systems without equilibria with 3-D grid multiscroll chaotic attractors. *IEEE Trans. Circuits Syst. II Express Briefs* **2018**, *66*, 1456–1460. [[CrossRef](#)]
24. Ding, P.F.; Feng, X.Y.; Wu, C.M. Novel two-directional grid multi-scroll chaotic attractors based on the Jerk system. *Chin. Phys. B* **2020**, *29*, 108202. [[CrossRef](#)]
25. Xu, W.; Cao, N. Hardware design of a kind of grid multi-scroll chaotic system based on a MSP430f169 chip. *J. Circuits Syst. Comput.* **2020**, *29*, 2050189. [[CrossRef](#)]
26. Xu, X.L.; Li, G.D.; Dai, W.Y.; Song, X.M. Multi-direction chain and grid chaotic system based on Julia fractal. *Fractals* **2021**, *29*, 1–20. [[CrossRef](#)]
27. Cui, L.; Lu, M.; Ou, Q.; Duan, H.; Luo, W. Analysis and circuit implementation of fractional order multi-wing hidden attractors. *Chaos Solitons Fractals* **2020**, *138*, 109894. [[CrossRef](#)]
28. Ahmad, S.; Ullah, A.; Akgül, A. Investigating the complex behavior of multi-scroll chaotic system with Caputo fractal-fractional operator. *Chaos Solitons Fractals* **2021**, *146*, 110900. [[CrossRef](#)]
29. Yan, M.; Jie, J. Fractional-order multiwing switchable chaotic system with a wide range of parameters. *Chaos Solitons Fractals* **2022**, *160*, 112161. [[CrossRef](#)]
30. Tlelo-Cuautle, E.; Pano-Azucena, A.D.; Guillén-Fernández, O.; Silva-Juárez, A. *Analog/Digital Implementation of Fractional Order Chaotic Circuits and Applications*; Springer: Berlin/Heidelberg, Germany, 2020.
31. Hua, Z.; Zhou, Y.; Huang, H. Cosine-transform-based chaotic system for image encryption. *Inf. Sci.* **2019**, *480*, 403–419. [[CrossRef](#)]
32. Bao, H.; Hua, Z.; Wang, N.; Zhu, L.; Chen, M.; Bao, B. Initials-boosted coexisting chaos in a 2-D sine map and its hardware implementation. *IEEE Trans. Ind. Inform.* **2020**, *17*, 1132–1140. [[CrossRef](#)]
33. Hua, Z.; Zhang, Y.; Bao, H.; Huang, H.; Zhou, Y. n-Dimensional Polynomial Chaotic System With Applications. *IEEE Trans. Circuits Syst. I Regul. Pap.* **2021**, *69*, 784–797. [[CrossRef](#)]
34. Mohamed, S.M.; Sayed, W.S.; Said, L.A.; Radwan, A.G. Reconfigurable FPGA realization of fractional-order chaotic systems. *IEEE Access* **2021**, *9*, 89376–89389. [[CrossRef](#)]
35. Tlelo-Cuautle, E.; Rangel-Magdaleno, J.; Pano-Azucena, A.; Obeso-Rodelo, P.; Nuñez-Perez, J.C. FPGA realization of multi-scroll chaotic oscillators. *Commun. Nonlinear Sci. Numer. Simul.* **2015**, *27*, 66–80. [[CrossRef](#)]
36. Tolba, M.F.; Said, L.A.; Madian, A.H.; Radwan, A.G. FPGA implementation of fractional-order integrator and differentiator based on Grünwald Letnikov’s definition. In Proceedings of the 2017 29th International Conference on Microelectronics (ICM), Beirut, Lebanon, 10–13 December 2017; pp. 1–4.
37. Sayed, W.; Radwan, A.; Elnawawy, M.; Orabi, H.; Sagahyoon, A.; Aloul, F.; Elwakil, A.; Fahmy, H.; El-Sedeek, A. Two-Dimensional Rotation of Chaotic Attractors: Demonstrative Examples and FPGA Realization. *Circuits Syst. Signal Process.* **2019**, *38*, 4890–4903. [[CrossRef](#)]
38. Sayed, W.S.; Radwan, A.G.; Fahmy, H.A.; Elsedek, A. Trajectory control and image encryption using affine transformation of Lorenz system. *Egypt. Inform. J.* **2021**, *22*, 155–166. [[CrossRef](#)]
39. Sayed, W.S.; Roshdy, M.; Said, L.A.; Radwan, A.G. Design and FPGA Verification of Custom-Shaped Chaotic Attractors Using Rotation, Offset Boosting and Amplitude Control. *IEEE Trans. Circuits Syst. II Express Briefs* **2021**, *68*, 3466–3470. [[CrossRef](#)]
40. Li, C.; Sprott, J.C. Variable-boostable chaotic flows. *Optik* **2016**, *127*, 10389–10398. [[CrossRef](#)]
41. Pham, V.T.; Akgul, A.; Volos, C.; Jafari, S.; Kapitaniak, T. Dynamics and circuit realization of a no-equilibrium chaotic system with a boostable variable. *AEU Int. J. Electron. Commun.* **2017**, *78*, 134–140. [[CrossRef](#)]
42. Pham, V.T.; Wang, X.; Jafari, S.; Volos, C.; Kapitaniak, T. From Wang–Chen system with only one stable equilibrium to a new chaotic system without equilibrium. *Int. J. Bifurc. Chaos* **2017**, *27*, 1750097. [[CrossRef](#)]
43. Li, C.; Lei, T.; Wang, X.; Chen, G. Dynamics editing based on offset boosting. *Chaos Interdiscip. J. Nonlinear Sci.* **2020**, *30*, 063124. [[CrossRef](#)]
44. Munoz-Pacheco, J.; Zambrano-Serrano, E.; Volos, C.; Tacha, O.; Stouboulos, I.; Pham, V.T. A fractional order chaotic system with a 3D grid of variable attractors. *Chaos Solitons Fractals* **2018**, *113*, 69–78. [[CrossRef](#)]



45. Gu, S.; He, S.; Wang, H.; Du, B. Analysis of three types of initial offset-boosting behavior for a new fractional-order dynamical system. *Chaos Solitons Fractals* **2021**, *143*, 110613. [[CrossRef](#)]
46. Ai, X.; Sun, K.; He, S.; Wang, H. Design of grid multiscroll chaotic attractors via transformations. *Int. J. Bifurc. Chaos* **2015**, *25*, 1530027. [[CrossRef](#)]
47. Bar-Yam, Y. *Dynamics of Complex Systems*; CRC Press: Boca Raton, FL, USA, 2019.
48. Wang, Z.; Volos, C.; Kingni, S.T.; Azar, A.T.; Pham, V.T. Four-wing attractors in a novel chaotic system with hyperbolic sine nonlinearity. *Optik* **2017**, *131*, 1071–1078. [[CrossRef](#)]
49. AboAlNaga, B.M.; Said, L.A.; Madian, A.H.; Radwan, A.G. Analysis and FPGA of semi-fractal shapes based on complex Gaussian map. *Chaos Solitons Fractals* **2021**, *142*, 110493. [[CrossRef](#)]
50. Mohamed, S.M.; Sayed, W.S.; Radwan, A.G.; Said, L.A. FPGA Implementation of Reconfigurable CORDIC Algorithm and a Memristive Chaotic System With Transcendental Nonlinearities. *IEEE Trans. Circuits Syst. I Regul. Pap.* **2022**, *69*, 2885–2892. [[CrossRef](#)]
51. Koyuncu, İ.; Şahin, İ.; Gloster, C.; Saritekin, N.K. A neuron library for rapid realization of artificial neural networks on FPGA: A case study of Rössler chaotic system. *J. Circuits Syst. Comput.* **2017**, *26*, 1750015. [[CrossRef](#)]
52. Tuna, M. A novel secure chaos-based pseudo random number generator based on ANN-based chaotic and ring oscillator: Design and its FPGA implementation. *Analog. Integr. Circuits Signal Process.* **2020**, *105*, 167–181. [[CrossRef](#)]
53. Hu, Y.H. CORDIC-based VLSI architectures for digital signal processing. *IEEE Signal Process. Mag.* **1992**, *9*, 16–35. [[CrossRef](#)]
54. Harber, R.G.; Hu, X.; Li, J.; Bass, S.C. The application of bit-serial CORDIC computational units to the design of inverse kinematics processors. In Proceedings of the 1988 IEEE International Conference on Robotics and Automation, Philadelphia, PA, USA, 24–29 April 1988; pp. 1152–1157.
55. Volder, J.E. The birth of CORDIC. *J. VLSI Signal Process. Syst. Signal Image Video Technol.* **2000**, *25*, 101–105. [[CrossRef](#)]
56. Vladimirova, T.; Tiggeler, H. FPGA implementation of sine and cosine generators using the CORDIC algorithm. In Proceedings of the 1999 MAPLD International Conference, Washington, DC, USA, 1999; pp. 26–28.
57. Vachhani, L.; Sridharan, K.; Meher, P.K. Efficient FPGA realization of CORDIC with application to robotic exploration. *IEEE Trans. Ind. Electron.* **2009**, *56*, 4915–4929. [[CrossRef](#)]
58. Qian, M. Application of CORDIC algorithm to neural networks VLSI design. In Proceedings of the Multiconference on “Computational Engineering in Systems Applications”, Beijing, China, 4–6 October 2006; Volume 1, pp. 504–508.
59. Walther, J.S. A unified algorithm for elementary functions. In Proceedings of the Spring Joint Computer Conference, Atlantic City, NJ, USA, 18–20 May 1971; pp. 379–385.
60. Yalçın, M.E.; Suykens, J.A.; Vandewalle, J.; Özoğuz, S. Families of scroll grid attractors. *Int. J. Bifurc. Chaos* **2002**, *12*, 23–41. [[CrossRef](#)]
61. Lian, S. *Multimedia Content Encryption: Techniques and Applications*; CRC Press: Boca Raton, FL, USA, 2008.
62. Bassham, L.E., III; Rukhin, A.L.; Soto, J.; Nechvatal, J.R.; Smid, M.E.; Barker, E.B.; Leigh, S.D.; Levenson, M.; Vangel, M.; Banks, D.L.; et al. *Sp 800-22 Rev. 1a. a Statistical Test Suite for Random And Pseudorandom Number Generators for Cryptographic Applications*; National Institute of Standards & Technology: Gaithersburg, MD, USA, 2010.
63. Gotz, M.; Kelber, K.; Schwarz, W. Discrete-time chaotic encryption systems. I. Statistical design approach. *IEEE Trans. Circuits Syst. I Fundam. Theory Appl.* **1997**, *44*, 963–970. [[CrossRef](#)]
64. Mao, Y.; Chen, G. Chaos-based image encryption. In *Handbook of Geometric Computing*; Springer: Berlin/Heidelberg, Germany, 2005; pp. 231–265.
65. Wu, Y.; Noonan, J.P.; Agaian, S. NPCR and UACI randomness tests for image encryption. *Cyber J. Multidiscip. J. Sci. Technol. J. Sel. Areas Telecommun. JSAT* **2011**, *1*, 31–38.
66. Bao, H.; Hua, Z.; Liu, W.; Bao, B. Discrete memristive neuron model and its interspike interval-encoded application in image encryption. *Sci. China Technol. Sci.* **2021**, *64*, 2281–2291. [[CrossRef](#)]
67. Lin, R.; Li, S. An image encryption scheme based on Lorenz Hyperchaotic system and RSA algorithm. *Secur. Commun. Netw.* **2021**, *2021*, 5586959. [[CrossRef](#)]