Sigma Journal of Engineering and Natural Sciences, Vol. 39, No. 1, pp. 1-12, March, 2022
Yildiz Technical University Press, Istanbul, Turkey

A novel application for DC motor-generator cascade system by changing signal densıty of digital chaotic oscillator

Ercan KÖSE[1,*], Aydın MÜHÜRCÜ[2], Serdar COŞKUN[3]

ABSTRACT

Presented is a new method for the realization of a chaotic oscillator in a digital environment. First, a two-stroke sampling mathematical regulation is developed for discrete-time oscillator equations to change signal densities of chaotic signals. This proposed mathematical regulation is applied to Lorenz’s chaotic oscillator, which presents a complex dynamical behavior. An application is shown with simulation through a Matlab-Simulink environment with time-dependent density changes of x, y and z $1-D$ graphics and x,y $2-D$ phase space graphics that are dependent on different density changes. Further to this, in an experimental study, Lorenz’s chaotic oscillator’s signals with variable density is applied to a DC motor as armature voltage via an 8-bit microcontroller based hardware environment. Chaotic supply voltage is applied to the motor rotor to generate a chaotic angular velocity. Time-dependent density change results of x, y and z $1-D$ graphics are obtained and shown on an oscilloscope by converting chaotic rotor angular velocity to electrical signals, through a tacho-generator. The observed results revealed that chaotic signal production with variable density is achieved both in the simulation environment and the experimental environment. Also, it is shown that the proposed program and mathematical equations are feasible in terms of hardware and software implementations.

Keywords: *Chaotic systems, digital chaotic oscillator, microcontroller*

INTRODUCTION

Chaos is the deterministic signal power that is spread to a vast area on frequency axis and, which is quite sensitive to initial conditions. In addition, chaotic signals are described as irregular oscillations and present non-periodic dynamic behaviors. There are many studies in the literature for understanding the natural behavior of chaos systems and interpretation of their dynamics. These studies are primarily focusing on eliminating chaos signals or negative impacts of chaos for developing new theoretical and practical perspectives using the chaos concept and chaotic signals in different areas Ogorzalek [1]. Motivated by the aforementioned nature of the problem, present paper aims to eliminate the negative effects of chaos by proposing a chaotic oscillator and to implement it in a real-time problem.

Chaos’s negative impacts are observed in many systems, thus many control methods and algorithms by Köse and Mühürcü [2], Behera et al. [3], Gholipour et al. [4] have been developed in order to eliminate or limit the signals so-called chaotic noises. Some of these control methods: adaptive control Adomaitien□ et al. [5], feedback control Ontañón-García and Campos-Cantón [6], back-stepping control Farivar et al. [7], fuzzy control Khanesar et al. [8], sliding mode control Aghababa [9], passive control Luo et al. [10], and active control Bhalekar [11] are available in the literature.

This paper was recommended for publication in revised form by Area Editor N. Özlem Ünverdi
[1] Tarsus University, Department of Electrical-Electronics Eng., Tarsus-Mersin, Turkey
[2] Kırklareli University, Department of Mechatronics Eng., Kırklareli, Turkey
[3] Tarsus University, Department of Mechanical Eng., Tarsus-Mersin, Turkey
** E-mail address: ekose@tarsus.edu.tr*
Orcid id: https://orcid.org/0000-0001-9814-6339, 0000-0002-9206-5746, 0000-0002-7080-0340
Manuscript Received 17 December 2021, Accepted 30 March 2022

However, chaos can have positive impacts in many application areas, for instance, anaylysis of multiple system behaviors and Dynamics by Bigdeli and Afshar [12], Liang et al. [13], synchronization of systems Shahverdiev et al. [14], Gao et al. [15] reliable communication applications Yang et al. [16], Yang-Cheng et al. [17], biomedical and medical systems Coyle [18], Taki et al. [19]. In the application, many different noises caused by various sources are considered as chaotic noise that spreads to a vast frequency band, such as fan blade noise depending on its fan motor's angular velocity, absorbing noise of a vacuum cleaner, washing machine's noise caused by its angular motion. A solution to eliminate micro-milling instability and reduce its chaotic noise, a multidimensional time-frequency control method is suggested by Liu et al. [20].

Chaotic systems inherently generate noise-like signals and their power density spectrum are distributed over a certain frequency range. It is known that even a very simple dynamic system can show chaotic behavior, its use for generating broadband signals could be an interesting and cost-effective alternative to investigating the noise effect. The mentioned chaos signals have to be produced for developing a practical method to utilize dynamic behavioral features of chaotic oscillators. As practical production methods, microcontroller and FPGA-based digital platforms are reported in Tlelo-Cuautle et al. [21]. To produce continuous-time chaos oscillator signals in these platforms, we obtain discrete-time chaos oscillator equations and then these equations are converted into software codes. In the digital environment, the generation of chaos signals is easy, flexible, and reliable as mentioned before. In addition to these, signal densities of the generated chaos signals should be regulated. For example, the density of the chaotic signal for the rotational velocity of a chaotic mixer in a milk boiler is pretty smaller than that of the chaotic signal in the chaotic coding of a voice chaotic coder or in the chaotic trajectory of a fired rocket.

Controlling chaotic oscillator signals density in unit time provides an important step to develop and enhance theoretical and practical application areas of chaos theory. The chaotic oscillator signal densities open a door for more flexible production of modulation signals in communication. Therefore, implementing the information in different frequencies will be possible with desired chaotic modulation speed by using the same chaotic oscillator. Chaotic referenced speed control of a DC motor is rendered as another application area. If a chaotic oscillator signal density is wisely regulated, DC motor velocity, and thereby the torque is converted into a structure that has the desired chaotic speed change. Moreover, electro-optical systems are also a new application area where the chaotic signals are produced by chaotic oscillators with regulatable signal densities and they are used as screen scanning signals for camera, television and advertisement boards.

In this study, the chaotic signal density change is demonstrated through Lorenz chaotic oscillator. A Lorenz system exhibits very complex dynamical behavior for analysis, so-called two-scroll butterfly-shaped has been used to model many engineering systems. In this work, output signals of the Lorenz chaotic oscillator are produced simultaneously for 3 different signal densities in the Matlab-Simulink environment and the signal images are monitored. The achievement of chaotic signal production with variable density in the simulation environment is experimentally validated in a microcontroller-based hardware environment and it is shown that signal production with variable density is also possible in a hardware environment. Lorenz chaotic oscillator's signals with variable density are applied to a DC motor as armature voltage in hardware study. The chaotic voltage supply makes the DC motor's rotor angular velocity chaotic. The chaotic rotor angular velocity is converted to electrical signals and shown on an oscilloscope screen through a tacho generator. The innovations are formally stated below:

- By using microcontroller technology instead of Opamp-based analog circuit setup shows that the parameters of the chaotic signal can be successfully controlled with high sensitivity
- Chaotic oscillators based on analog circuit never produces chaotic signals with respect to the predicted initial condition. Because during the operation of the chaotic oscillator, the level of the supply voltage and the type of noise ratio may vary. Because the above points in a digital environment is inevitable, the proposed microcontroller-based oscillator can work more reliable even in nonlinear operating conditions
- The operating conditions of the oscillator can be changed easily with the proposed chaotic microcontroller oscillator only by changing the microcontroller structure
- The used $PIC18F452$ 8-bit microcontroller-based oscillator generator has a low cost and it is easy to implement software platform.

MATERIAL AND METHODS

DISCRETIZATION PROCESS FOR CONTINUOUS TIME CHAOTIC OSCILLATORS

The Lorenz chaotic oscillator has non-linear equations where x, y and z are state variables; a, b and c are positive constant parameters, Eq. (1)

$$\left.\begin{aligned} \dot{x}(t) &= ay(t) - ax(t) \\ \dot{y}(t) &= cx(t) - x(t)z(t) - y(t) \\ \dot{z}(t) &= x(t)y(t) - bz(t) \end{aligned}\right\} \tag{1}$$

Using forward difference method Co [22], Fadali and Visioli [23], the Lorenz time domain equations can be converted to Lorenz Z-domain (discrete time).

Before the conversion, we consider the following preliminaries on discrete time signal theory. The signal $y(t)$ is sampled and we compute the area between two consecutive instants kT and $(k+1)T$. We then have an integral, which can be splitted in two parts as follow:

$$y((k+1)T) = \int_{0}^{(k+1)T} y(\tau)d\tau = \int_{0}^{kT} y(\tau)d\tau = \int_{kT}^{(k+1)T} y(\tau)d\tau, \tag{2}$$

where the first integral is the signal $y(kT) = y(k)$. The second integral is the area of a plot of $y(t)$ and $y((k+1)T)$, and it can be approximated with an area of a rectangle with base of amplitude as T and height of amplitude as $y(kT) = y(k)$, that can be rewritten:

$$y((k+1)T) = y(k) + Tu(k) \tag{3}$$

$$y((k+1)T) - y(k+1) = \int_{kT}^{(k+1)T} y(\tau)d\tau - Ty(k) \tag{4}$$

SIGNAL PRODUCTION USING TWO DIFFERENT SAMPLING PERIOD

A continuous-time signal can be easily created using discrete-time equations in a digital platform with proper sampling time Ts of choice. During signal production, changing the value of Ts period does not change the frequency or phase of the produced signal. The frequency or phase change is possible only through changing frequency or phase parameters in the equation Stojkovi□ and Stanimirovi□ [25]. In this study, using Lorenz chaotic oscillator's x, y and z $1-D$ chaotic signals, two different sampling periods are defined as Ts and $Ts_{(inside)}$, the production of chaotic signal with variable density is achieved in both Matlab-Simulink co-simulation environment and microcontroller-based hardware environment.

$$k_{velocity} = \frac{Ts(inside)}{Ts} \tag{5}$$

The chaotic signal production method with the proposed two sampling periods generates the signals without making any changes to chaotic oscillator equations or equation parameters. Therefore, without changing the process period (Ts) of Lorenz chaotic oscillator algorithm's mathematical equations, a second sampling period $(Ts_{(inside)})$ is used to adjust the density of the signals. The quantity of the produced signal's density is given in Eq. (5).

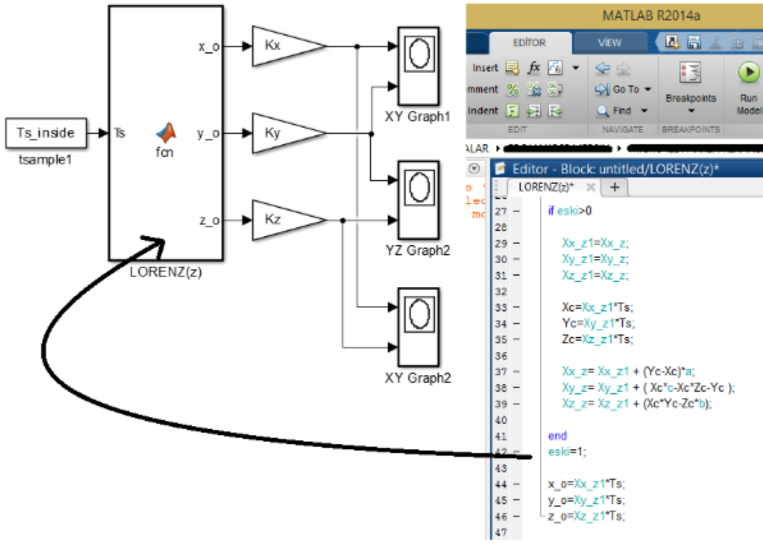


Figure 1. Discrete-time Lorenz chaotic oscillator's circuit and software program in Matlab-Simulink.

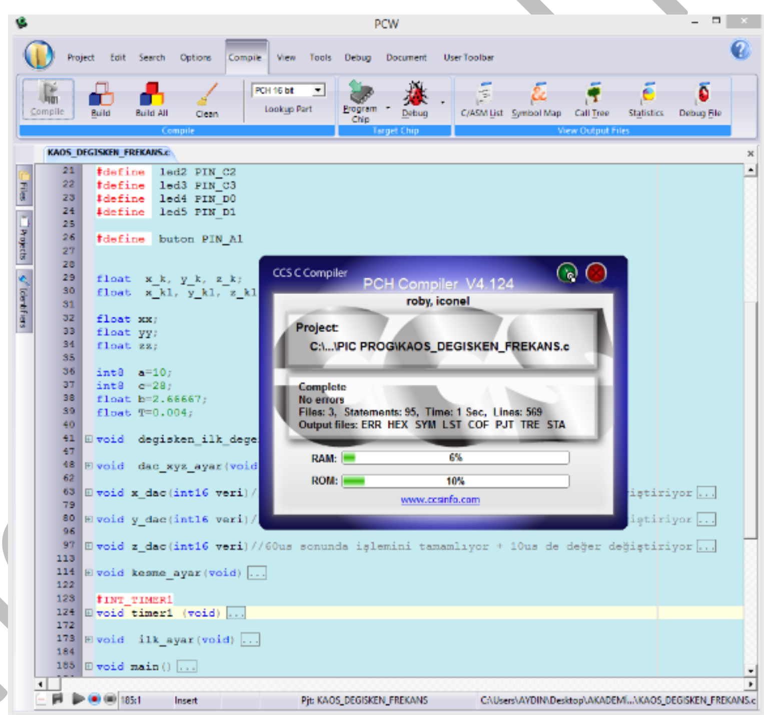


Figure 2. Used microcontroller's memory percentages.

$$Signal_{alternate_{velocity}} = \begin{cases} high, & k_{velocity} > 1 \\ natural, & k_{velocity} = 1 \\ low, & k_{velocity} < 1 \end{cases} \tag{6}$$

FLOW CHART AND SOFTWARE

Using discrete-time block diagram of Lorenz oscillator to produce x, y and z $1-D$ chaotic signals in digital platform, the algorithm flow chart given in Figure 5. Implementation details of two consecutive sampling periods, i.e., Ts and $Ts_{(inside)}$ is seen in Figure 5. Matlab-Function block represents the steps of flow chart of, x, y and z chaotic signals, which are produced in Matlab-Simulink environment for simulation, seen in Figure 1.

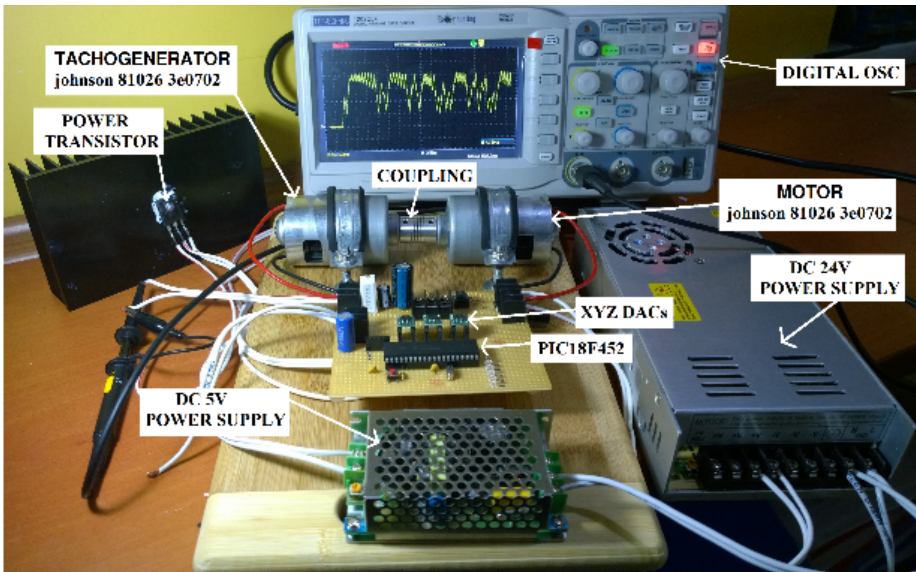


Figure 3. Empirical set for Lorenz chaotic oscillator.

EMPIRICAL HARDWARE MODEL

In this study, in order to create Lorenz chaotic oscillator in a digital platform, a $8-bit$ $PIC18F452$ microcontroller chip belonging to Microchip Technology is used. $PIC18F452$ Microcontroller chip has an ALU (Arithmetic Logic Unit), so it can perform the mathematical operations in a short period of time.

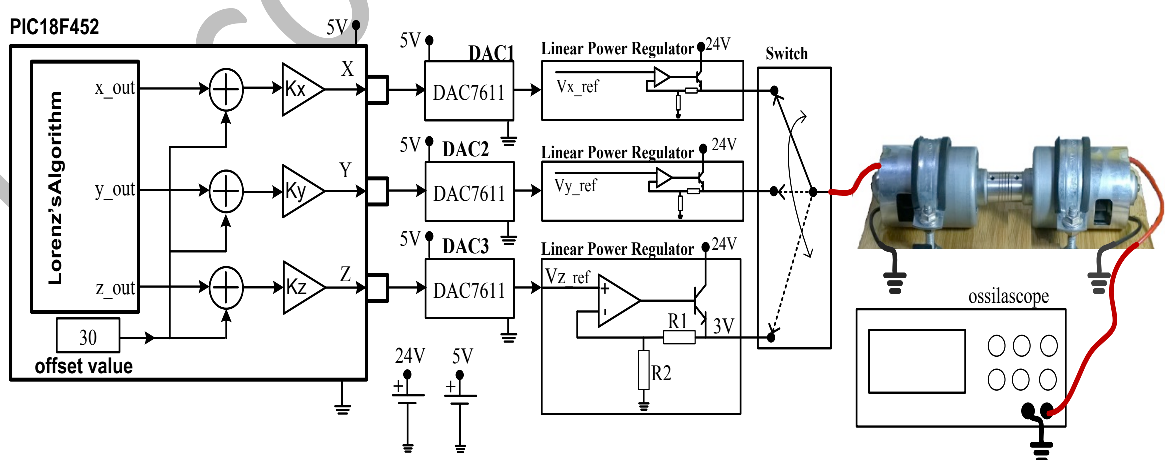


Figure 4. Chaotic oscillator circuit hardware unit block diagram including microcontroller.

In Matlab-Simulink environment, Matlab-Function codes available in Figure 1 are edited for $CCS-C$ compiler and after compiling, they are packed into the the microcontroller, Figure 2. The $CCS-C$ compiles the Lorenz chaotic oscillator software program, for $PIC18F452$ microcontroller, memory cost is measured as 10% ROM and 6% RAM. These low memory values show that more than one chaotic oscillator can be implemented into the microcontroller chip easily, without using complex circuits, as shown in Figure 2. The hardware implementation environment is displayed in Figure 3, the block diagram is shown in Figure 4, as well as the values of the experiment in Table 1.

Table 1. Experimental design elements

Quantity	Name, Serial Number
1	Microcontroller, pic18F452
3	Mono-polar DAC, DAC7611
1	DC-motor, Johnson-81026 3e0702
1	Tacho-generator, Johnson-81026 3e0702
1	DC-5V, smps-power supply
1	DC-24V, smps-power supply
1	BJT power transistor, TIP35C
1	Digital Oscilloscope

The supply voltage that we apply to motor's armature winding in the form of angular velocity on motor's rotor. The tacho-generator produces the voltage which is proportional to shaft speed, is used as a speed transducer for the feedback. After transducing, making the angular velocity's amplitude and shape similar to motor armature winding voltage, the name and serial number of the transducer which we use as tacho-generator and the name and serial number of DC-motor, which the energy is given, are chosen as the same. DC-motor parameters are shown in Table 2.

Before starting the Lorenz chaos algorithm, parameter values (a=10, b=8/3, c=28 and $T_s = 1ms$), the initial value of state variables, sampling period, and signal rate coefficient are determined. The rate coefficient (k_rate) changes the sampling period. The mark formation rate can be selected in the range of k_velocity value (0.1) to be higher than the natural rate (Eq. 5), and also in the range of k_rate coefficient (1,∞) for a lower generation rate. In the algorithm, discrete-time calculation of *X*, *Y* and *Z* discrete-time output variables of the discrete-time Lorenz algorithm have been calculated according to the equations given below.

$$x_{out}(k+1)=\left[x(k-1)+a\left(y(k)-x(k)\right)\right]T_{inside} \tag{7}$$

$$y_{out}(k+1)=\left[y(k-1)+cx(k)-x(k)z(k)-y(k)\right]T_{inside} \tag{8}$$

$$z_{out}(k+1)=\left[z(k-1)+x(k)y(k)-bz(k)\right]T_{inside} \tag{9}$$

$1-D$ x, y and z chaotic signals belonging to discrete-time Lorenz chaotic oscillator algorithm are used in the hardware environment through the microcontroller. It is then applied to the motor as armature voltage through a selection key. The waveform of motor armature winding chaotic voltage causes the motor rotor angular velocity to become chaotic. Motor rotor's angular velocity is converted to electrical signals through tacho-generator, which is fastened to the motor shaft with a coupler and is observed in Figure 4.

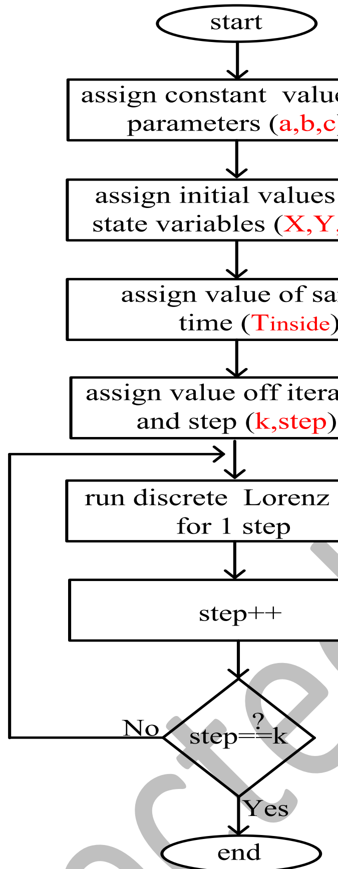


Figure 5. Flow-chart for discrete-time Lorenz algorithm structure with two different sampling periods

Table 2. DC motor parameters

Name, Serial Number	Johnson, $810263e0702$
Max. Efficiency (ME)	%63.36
ME Torque	$123.68 mN-m$
ME Speed	$15998 rpm$
ME Current	$25.14 A$
ME Output Power	$207.05 W$
Torque Constant	$6.3850 m-Nm/Amp$
Armature Resistance	$0.1210 Ohms$
Motor Regulation	$30.3390 Rpm/m-Nm$

SIMULATION RESULTS

Simulations results are shown in graphics for unit time densities of the chaotic signals belonging to Lorenz chaotic oscillator, $kvelocity = 0.5$, , $kvelocity = 1$, and $kvelocity = 2$ in Figures 5,6,7 respectively. Here, $Ts = 1ms$ and the initial conditions are chosen as $x = 1, y = -0.5$ and $z = 1$. The signal density changes seen on $1-D$ graphs do not affect $2-D$ graphics' shape. The affected part is cycle number of the shape dependent to density, given in Figure 8. Each of simulation results has 40 seconds time interval in Matlab-Simulink environment. The only factor which affects density of x, y and z $1-D$ chaotic signals produced for 40 seconds is Tsinside period value.

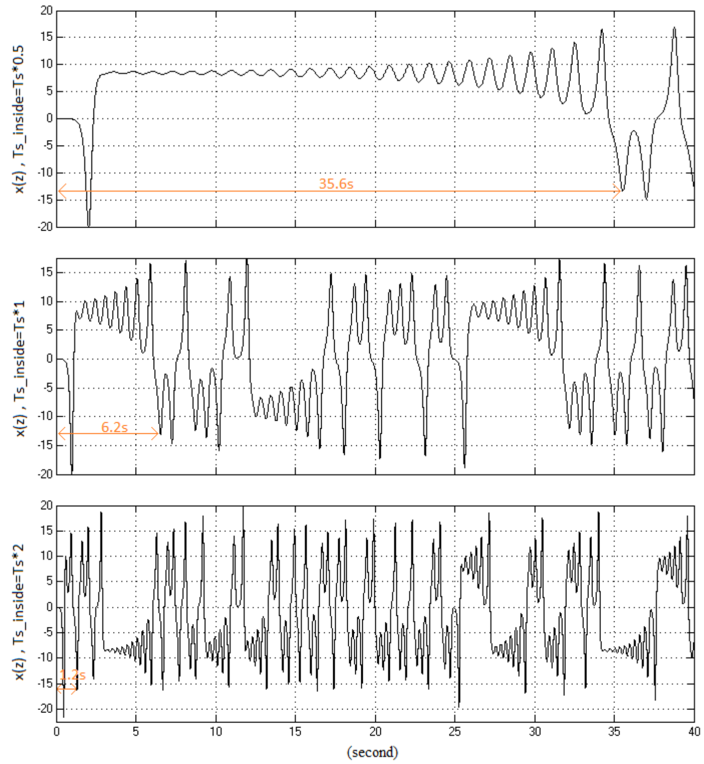


Figure 5. $1-D$ chaos signals produced for different $Ts(inside) = 0.5$ period.

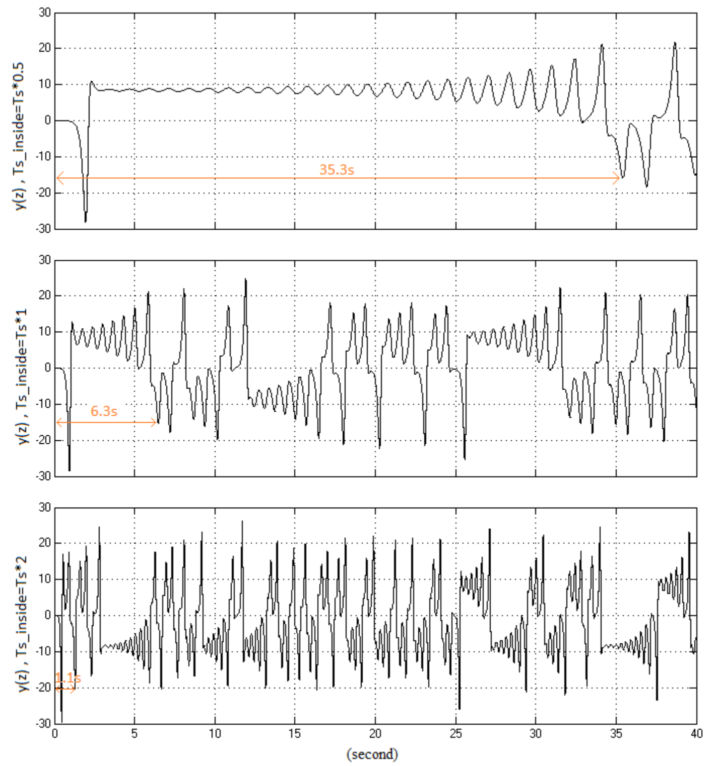


Figure 6. $1-D$ chaos signals produced for different $Ts(inside)=1$ period.

As can be seen from the Table 3, by changing the velocity coefficient, the signal formation rate of the output variables of the chaos algorithm also changes within the framework of the rule given in Eq. (6). While the K velocity coefficient changes the pointer velocity; signal amplitude did not change. As predicted, the velocity coefficient only affected the formation rate of the chaos sign.

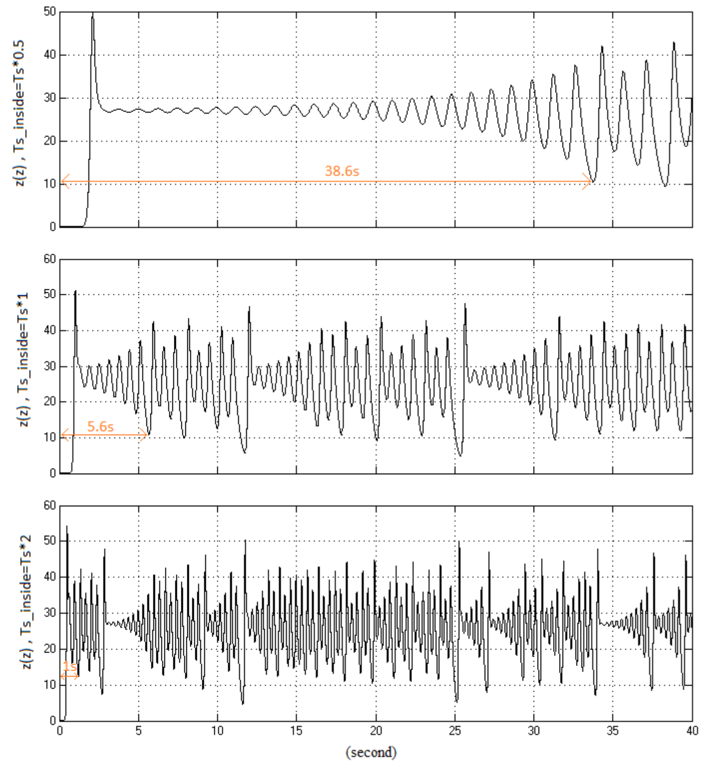
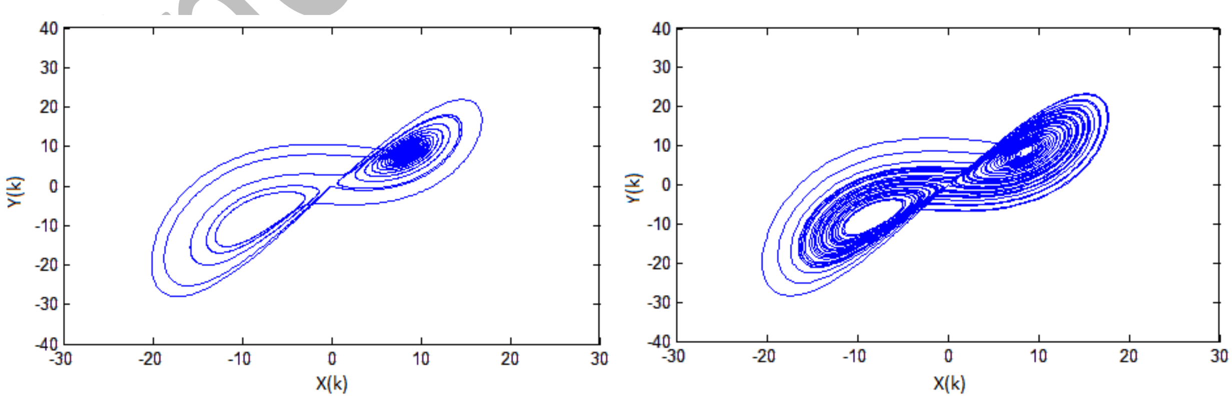


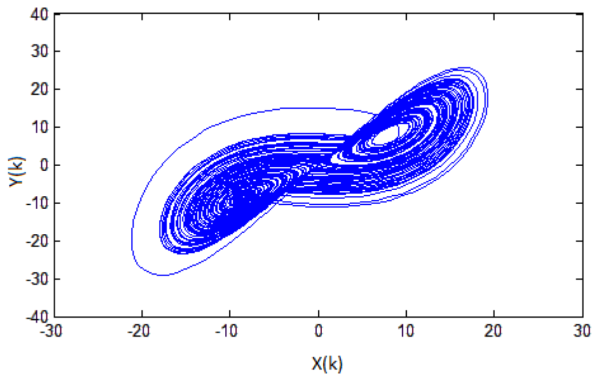
Figure 7. $1-D$ Chaos signals produced for different $Ts(inside) = 2$ period.

Table 3. The speed dependent results

k_velocity	**0.5**	**1**	**2**
Δt_x (ref points)	35.6s	6.2s	1.2s
velocity of state X	low	natural	high
Δt_y (ref points)	35.3s	6.3s	1.1s
velocity of state Y	low	natural	high
Δt_z (ref points)	38.6s	5.6s	1s
velocity of state Z	low	natural	high
amplitude of X,Y,Z	natural	natural	natural



(a) $k_{velocity} = 0.5$ (b) $k_{velocity} = 1$



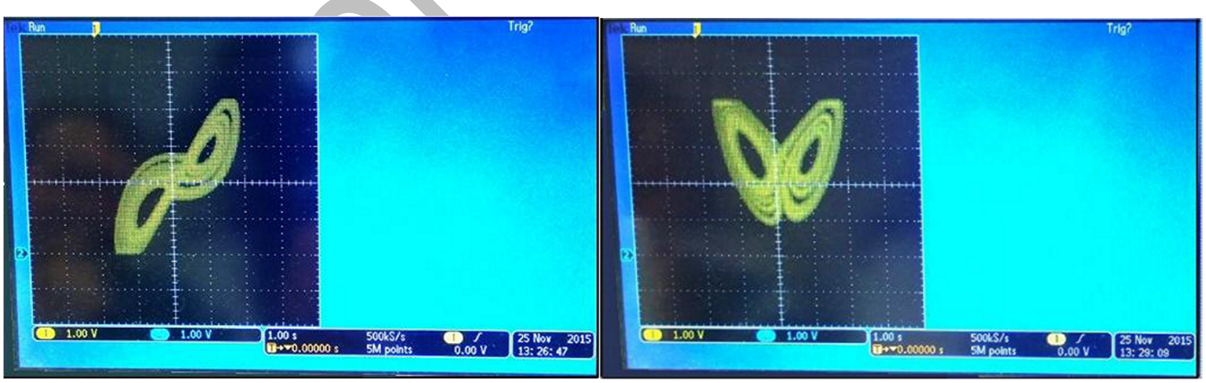
(b) $k_{velocity} = 2$

Figure 9. Simulation results of 40 second $2-D$ chaos signals dependent to $Ts(inside)$ period value.

EXPERIMENTAL RESULTS

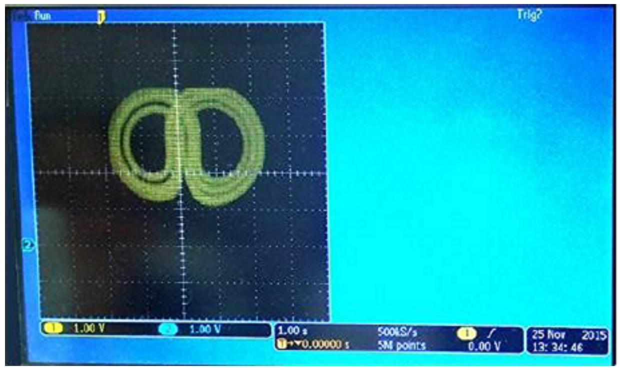
In Figure 9, using $PIC18F452$ microcontroller in the hardware environment, $2-D$ oscilloscope screens are obtained for $Ts_{(inside)}$ is shown. In order to obtain the same chaotic signal form in the simulation environment, Ts period is chosen same as $5ms$. Matlab-Simulink co-simulations are performed in this study and solved based on the $ode45$ solver. In the hardware environment, $PIC18F452$ microcontroller Lorenz chaotic oscillator program is embedded by using $CCS-C$ compiler. PIC compiler are using $IEEE-754$ floating point format for real numbers. The feature of this decimal number format is to be able to save the certainty with 6 numbers after the comma, as demonstrated in Åström and Wittenmark [25], Sarra and Meador [26].

The chaotic signals produced by chaotic oscillators are very sensitive to small parameter changes and calculation format change in digital environment Ogorzalek [1]; Ontañón-García and Campos-Cantón [6]; Khanesar et al. [8]. Two different waveforms from Lorenz chaotic oscillator's are considered as output variables, $X(k*Ts)$, demonstrated in Figure 10. They have the same initail conditions. The reason for the difference between the waveforms is that the floating operation results are edited by multiplying with 1.0000001 coefficient, during calculation of modified $X(k)$.



(a) xy

(b) xz



(c) yz

Figure 9. Outputs from DACxyz, $Ts(inside) = Ts * 1$, Kxyz=1, phase portrait oscilloscope displays for the Lorenz oscillator.

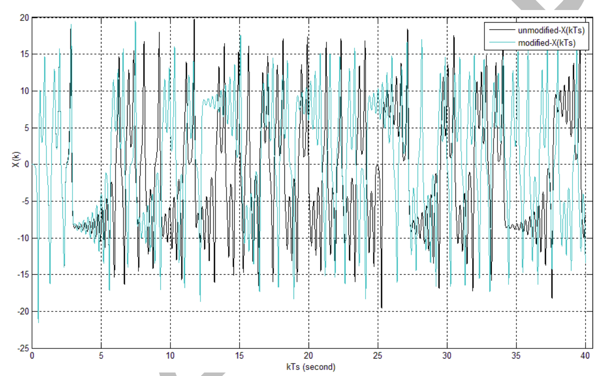
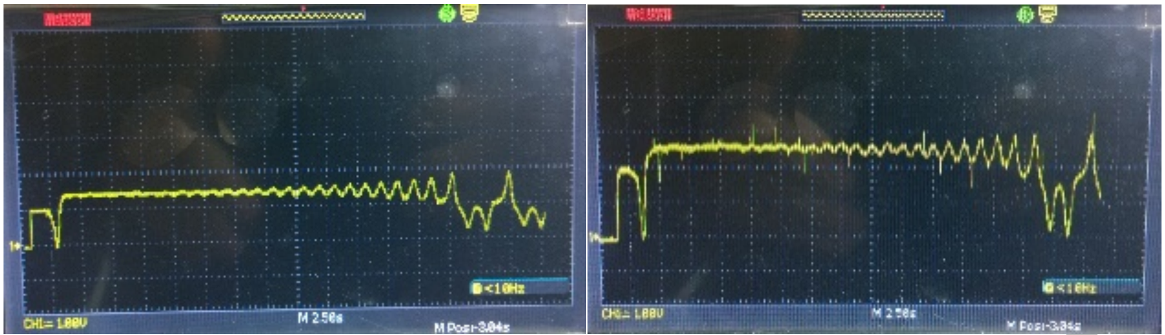
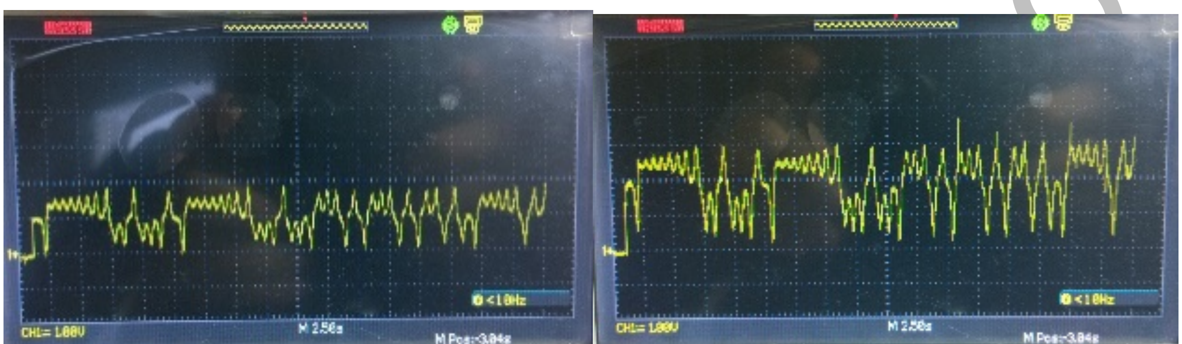


Figure 10. Different waveforms of $X(kTs)$ chaos signal having the same initial conditions.

This small error value and type are modeled for $CCS-C$ compiler's floating operations. It is understood from Figure 10 that the waveforms produced in simulation and hardware environments belonging to chaos signals do not match each other for both two implementation platform seven though they have the same chaos oscillator, the same sampling period, and the same initial conditions. In Figure 10, the reason for the differences between simulation $1-D$ graphs in Figures 5, 6, 7 and Figures 11, 12, 13 hardware $1-D$ graphics can easily be understood by considering the experimental result in simulation environment. The oscilloscope screen images of x, y and z $1-D$ chaotic signals' waveforms and densities produced by Lorenz chaotic oscillator performed in hardware environment are given in Figures 11, 12, 13. The produced waveforms of the chaotic output signals keep their shape even when the densities vary in the the same software platform ($CCS-C$).



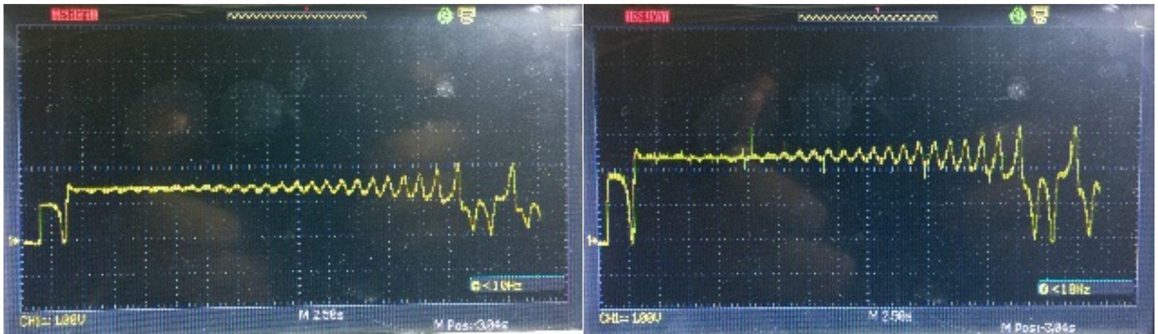
(a) Vx_{ref}, output from DACx, $Ts(inside) = Ts * 0.5$ (b) Vx, output from tacho gen., $Ts(inside) = Ts * 0.5$



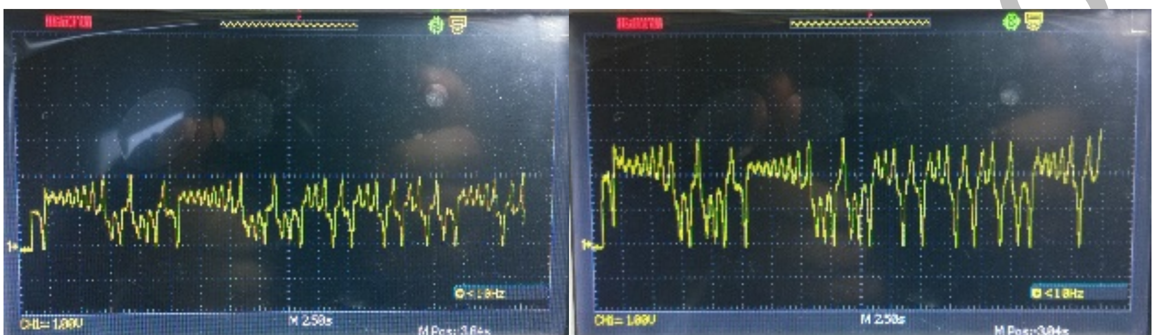
(c) Vx_{ref}, $Ts(inside) = Ts * 1$ (d) Vx, $Ts(inside) = Ts * 1$

(e) Vx_{ref}, $Ts(inside) = Ts * 2$ (d) Vx, $Ts(inside) = Ts * 2$

Figure 11. Lorenz-X signals obtained at different densities in the hardware environment.



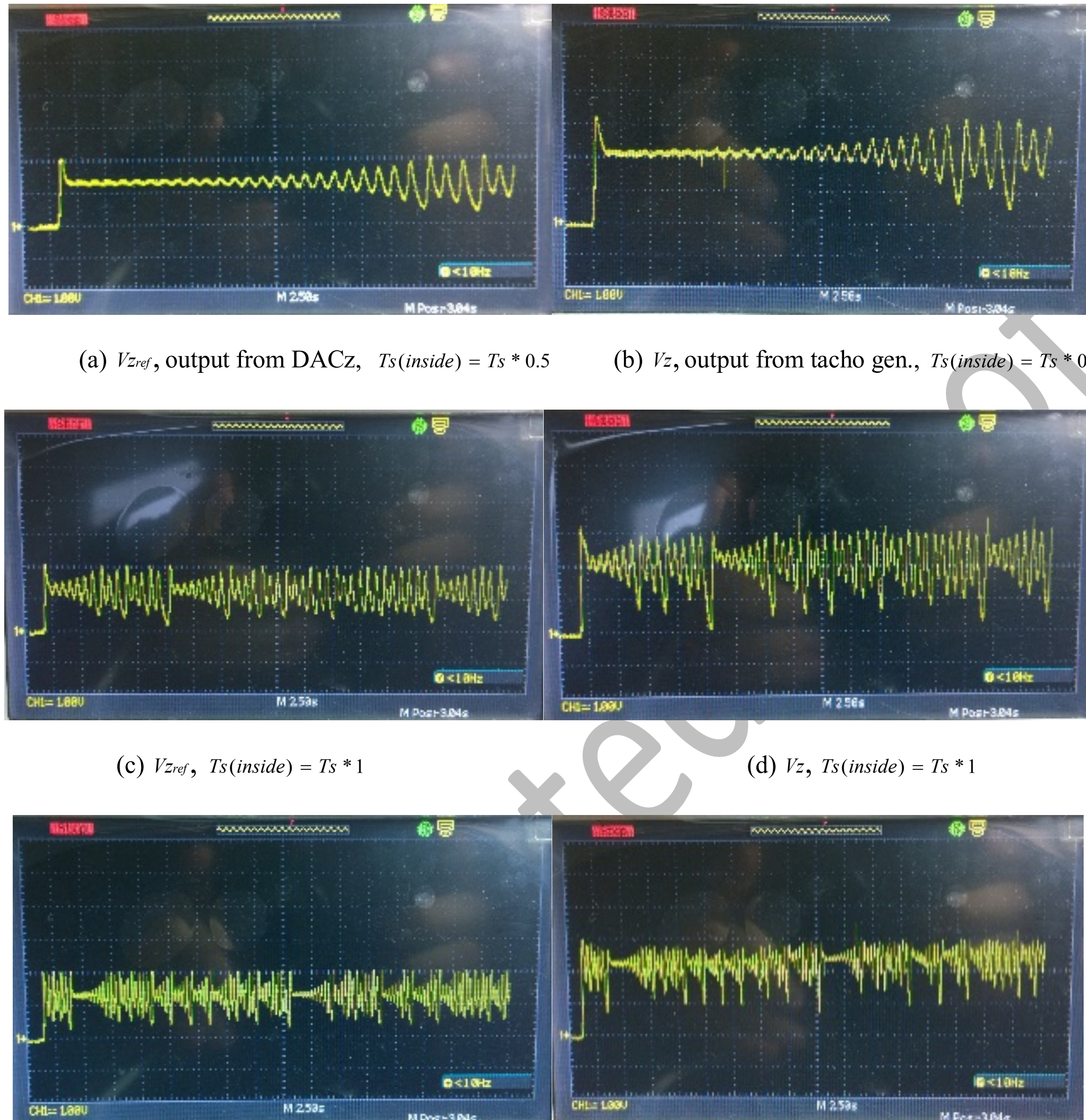
(a) Vy_{ref}, output from DACy, $Ts(inside) = Ts * 0.5$ (b) Vy, output from tacho gen., $Ts(inside) = Ts * 0.5$



(c) Vy_{ref}, $Ts(inside) = Ts * 1$ (d) Vy, $Ts(inside) = Ts * 1$

(e) Vy_{ref}, $Ts(inside) = Ts * 2$ (f) Vy, $Ts(inside) = Ts * 2$

Figure 12. Lorenz-Y signals obtained at different densities in the hardware environment.



(a) Vz_{ref}, output from DACz, $Ts(inside) = Ts * 0.5$ (b) Vz, output from tacho gen., $Ts(inside) = Ts * 0.5$

(c) Vz_{ref}, $Ts(inside) = Ts * 1$ (d) Vz, $Ts(inside) = Ts * 1$

(e) Vz_{ref}, $Ts(inside) = Ts * 2$ (f) Vz, $Ts(inside) = Ts * 2$

Figure 13. Lorenz-Z signals obtained at different densities in the hardware environment.

DISCUSSION AND CONCLUSION

In this study, a method with two sampling periods is proposed to realize a chaotic oscillator and signal density change in a digital environment. This suggested method is applied to Lorenz chaotic oscillator. Lorenz oscillator designed theoretically for discrete-time and simulated in Matlab-Simulink environment. Then, software codes of the oscillator are packed to $PIC18F452$ $8-bit$ microcontroller, belonging to Microchip Company; and x, y and z $1-D$ and $2-D$ Lorenz output signals are obtained. Chaos signals produced by making power transformations through the driver within the microcontroller, which is then applied to a DC motor as armature winding voltage. Since the energy coming into the motor is chaotic, motor shaft is excited with a chaotic angular velocity. In this study, the proposed method is simultaneously implemented in simulation and hardware experiments, and it is shown that natural signal densities belonging to chaotic oscillators can be adjusted. The obtained theoretical and experimental results revealed that the proposed digital oscillator method and the system design with two sampling periods works efficiently with respect to the design objectives. Lastly, this study shows that the chaotic oscillators natural signal densities can be changed, thus the way for many theoretical and practical studies is paved.

AUTHORSHIP CONTRIBUTIONS

Authors equally contributed to this work.

DATA AVAILABILITY STATEMENT

The authors confirm that the data that supports the findings of this study are available within the article. Raw data that support the finding of this study are available from the corresponding author, upon reasonable request.

CONFLICT OF INTEREST

The author declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

ETHICS

There are no ethical issues with the publication of this manuscript.

REFERENCES

[1] Ogorzalek, M. J. (1993). Taming chaos. i. synchronization. IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications, 40(10), 693-699.

[2] Köse, E., Mühürcü, A. (2017). Realization of a digital chaotic oscillator by using a low cost microcontroller. *Engineering Review*, 37(3), 341-348.

[3] Behera, S. K., Das, D. P., & Subudhi, B. (2014). Functional link artificial neural network applied to active noise control of a mixture of tonal and chaotic noise. *Applied Soft Computing*, 23, 51-60.

[4] Gholipour, R., Khosravi, A., & Mojallali, H. (2015). Multi-objective optimal backstepping controller design for chaos control in a rod-type plasma torch system using Bees algorithm. *Applied Mathematical Modelling*, *39*(15), 4432-4444.

[5] Adomaitien□, E., Mykolaitis, G., Bumelien□, S., & Tamaševi□ius, A. (2015). Adaptive nonlinear controller for stabilizing saddle-type steady states of dynamical systems. *Nonlinear Dynamics*, *82*(4), 1743-1753.

[6] Ontañón-García, L. J., & Campos-Cantón, E. (2013). Preservation of a two-wing Lorenz-like attractor with stable equilibria. *Journal of the Franklin Institute*, *350*(10), 2867-2880.

[7] Peng, C., & Zhang, W. (2021). Back-stepping stabilization of fractional-order triangular system with applications to chaotic systems. *Asian Journal of Control*, *23*(1), 143-154.

[8] Balootaki, M. A., Rahmani, H., Moeinkhah, H., & Mohammadzadeh, A. (2021). Non-singleton fuzzy control for multi-synchronization of chaotic systems. *Applied Soft Computing*, *99*, 106924.

[9] Mirrezapour, S. Z., Zare, A., & Hallaji, M. (2021). A new fractional sliding mode controller based on nonlinear fractional-order proportional integral derivative controller structure to synchronize fractional-order chaotic systems with uncertainty and disturbances. *Journal of Vibration and Control*, 1077546320982453.

[10] Sambas, A., Mamat, M., Arafa, A. A., Mahmoud, G. M., Mohamed, M. A., & Sanjaya, W. S. (2019). A new chaotic system with line of equilibria: dynamics, passive control and circuit design. *International Journal of Electrical & Computer Engineering (2088-8708)*, *9*(4).

[11] Handa, H. (2021, November). Active Control Synchronization of Similar and Dissimilar Chaotic Systems. In *2021 Innovations in Power and Advanced Computing Technologies (i-PACT)* (pp. 1-6). IEEE.

[12] Bigdeli, N., & Afshar, K. (2009). Chaotic behavior of price in the power markets with pay-as-bid payment mechanism. *Chaos, Solitons & Fractals*, *42*(4), 2560-2569.

[13] Liang, Z., Liang, J., Zhang, L., Wang, C., Yun, Z., & Zhang, X. (2015). Analysis of multi-scale chaotic characteristics of wind power based on Hilbert–Huang transform and Hurst analysis. *Applied energy*, *159*, 51-61.

[14] Shahverdiev, E. M., Hashimova, L. H., Bayramov, P. A., & Nuriev, R. A. (2015). Chaos synchronization between time delay coupled Josephson junctions governed by a central junction. *Journal of Superconductivity and Novel Magnetism*, *28*(12), 3499-3505.

[15] Gao, L., Wang, Z., Zhou, K., Zhu, W., Wu, Z., & Ma, T. (2015). Modified sliding mode synchronization of typical three-dimensional fractional-order chaotic systems. *Neurocomputing*, *166*, 53-58.

[16] Aliabadi, F., Majidi, M. H., & Khorashadizadeh, S. (2022). Chaos synchronization using adaptive quantum neural networks and its application in secure communication and cryptography. *Neural Computing and Applications*, 1-13.

[17] Nguyen, Q. D., & Huang, S. C. (2021). Synthetic adaptive fuzzy disturbance observer and sliding-mode control for chaos-based secure communication systems. *IEEE Access*, *9*, 23907-23928.

[18] Njitacke, Z. T., Isaac, S. D., Nestor, T., & Kengne, J. (2021). Window of multistability and its control in a simple 3D Hopfield neural network: application to biomedical image encryption. *Neural Computing and Applications*, *33*(12), 6733-6752.

[19] Manera, M. (2021). Perspectives on Complexity, Chaos and Thermodynamics in Environmental Pathology. *International Journal of Environmental Research and Public Health*, *18*(11), 5766.

[20] Taki, F. A., Pan, X., & Zhang, B. (2015). Revisiting Chaos Theorem to Understand the Nature of miRNAs in Response to Drugs of Abuse. *Journal of Cellular Physiology*, *230*(12), 2857-2868.

[21] Liu, M. K., Halfmann, E. B., & Suh, C. S. (2014). Multi-dimensional time-frequency control of micro-milling instability. *Journal of Vibration and Control*, *20*(5), 643-660.

[22] Tlelo-Cuautle, E., Rangel-Magdaleno, J. J., Pano-Azucena, A. D., Obeso-Rodelo, P. J., & Nuñez-Perez, J. C. (2015). FPGA realization of multi-scroll chaotic oscillators. *Communications in Nonlinear Science and Numerical Simulation*, *27*(1-3), 66-80.

[23] Co, T. B. (2013). *Methods of applied mathematics for engineers and scientists*. 12.4, Cambridge University Press.

[24] Fadali, M. S., & Visioli, A. (2013). *Digital control engineering: analysis and design*. Academic Press.

[25] Stojkovi□, N. V., & Stanimirovi□, P. S. (2001). Two direct methods in linear programming. *European Journal of Operational Research*, *131*(2), 417-439.

[26] Åström, K. J., & Wittenmark, B. (2011). *Computer-controlled systems: theory and design*. Third Edition, Lund Institute of Technology, Dover Publications, Inc., New York, , 30-52.

[27] Sarra, S. A., & Meador, C. (2011). On the numerical solution of chaotic dynamical systems using extend precision floating point arithmetic and very high order numerical methods. *Nonlinear Analysis: Modelling and Control*, *16*(3), 340-352.