

Analysis and Experimental Realization of the Logistic Map Using Arduino Pro Mini

Volodymyr Rusyn¹[0000-0001-6219-1031], Sergey Subbotin²[0000-0001-5814-8268], Aceng Sambas³[0000-0002-1623-0770]

¹Yuriy Fedkovych Chernivtsi National University, Kotsybinsky str., 2, Chernivtsi, 58012, Ukraine

rusyn_v@ukr.net

²Zaporizhzhia National Technical University, Zhukovsky str., 64, Zaporizhzhia, 69063, Ukraine

subbotin@zntu.edu.ua

³Universitas Muhammadiyah Tasikmalaya, Jl. Tamansari No. KM 2,5, Mulyasari, Kec. Tamansari, Tasikmalaya, Jawa Barat, 46196, Indonesia

acenx.bts@gmail.com

Abstract. In this paper, we used one-dimensional system that demonstrate chaotic behavior. Equation, analysis and system conditions are presented. Modern software environment LabView was used for analysis of the iteration series with different parameter r . Connection scheme, nominal components and programming code are also presented. For practical realization and visualizing of the logistic map we used Arduino Pro Mini and ten light-emitting diodes (LEDs) with ten resistors for each part of segment of the range $[0;1]$. The Arduino was connected to a computer through the USB port and programmed using a language similar to C++. Sketch was uploaded into Arduino using program software ArduinoIDE.

Keywords. Nonlinear, logistic map, Arduino

1 Introduction

Theory of chaos applied in various fields of research, such as economy [1-3], biology [4], mathematics [5-12], optics [13], robotics [14], memristors [15-20], electronics schemes [21-34], etc.

The logistic map is a one-dimensional discrete-time map that, despite its formal simplicity, exhibits an unexpected degree of complexity. Historically it has been one of the most important and paradigmatic systems during the early days of research on deterministic chaos. The map was popularized in a 1976 paper by the biologist Robert May [35] in part as a discrete-time demographic model analogous to the logistic equation first created by Pierre François Verhulst.

The logistic map is defined by the following equation:

$$x_{n+1} = rx_n(1-x_n) \quad (1)$$

Copyright © 2020 for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

The logistic map is a very simple system, which can produce chaotic behavior with the right values of the parameter r . Sensible values of r range from 0 to 4; also, the values of x range from 0 to 1.

Fig. 1 shows plot the behavior of the orbits for $r = 2.6$, $r = 3.3$, and $r = 3.9$. The software implementation of logistic map using LabView software environment shows in Fig. 2.

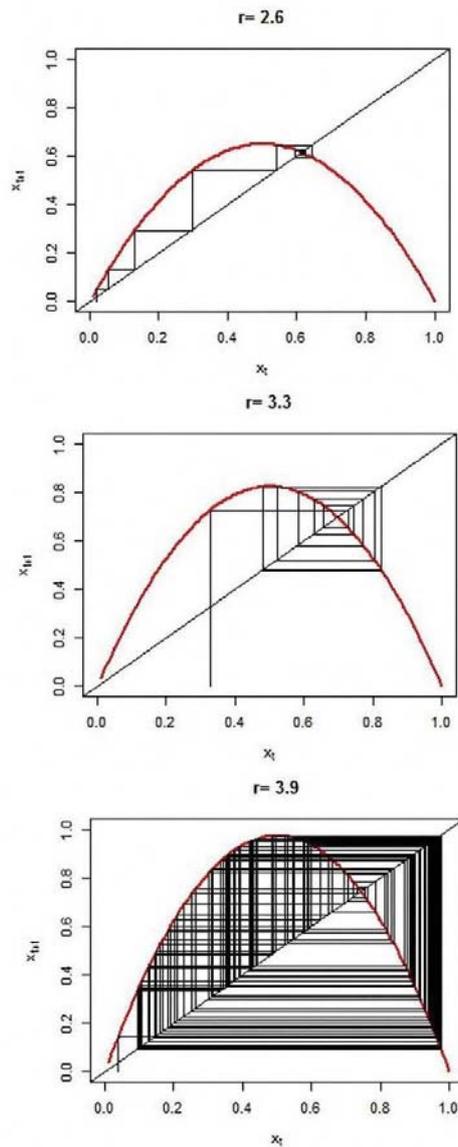


Fig. 1. The behavior of the orbits for different parameter r

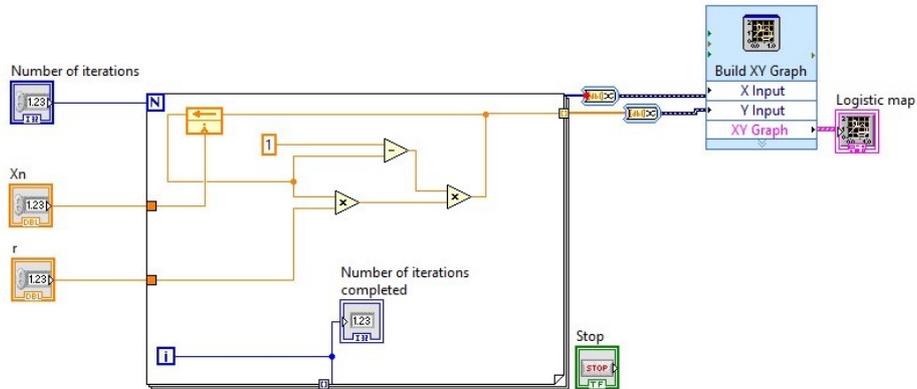


Fig. 2. Block scheme for realization of the logistic map

Fig. 3 and Fig. 4 shows computer modeling results for different initial conditions X_0 . Number of iterations equal 100. Results are presented for two cases: 1 – $X_0 = 0.4$ and 2 – $X_0 = 0.8$.

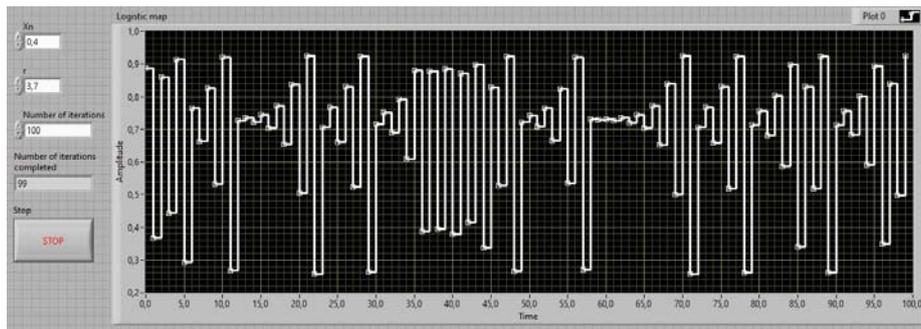


Fig. 3. Iteration series of logistic map for $X_0 = 0.4$

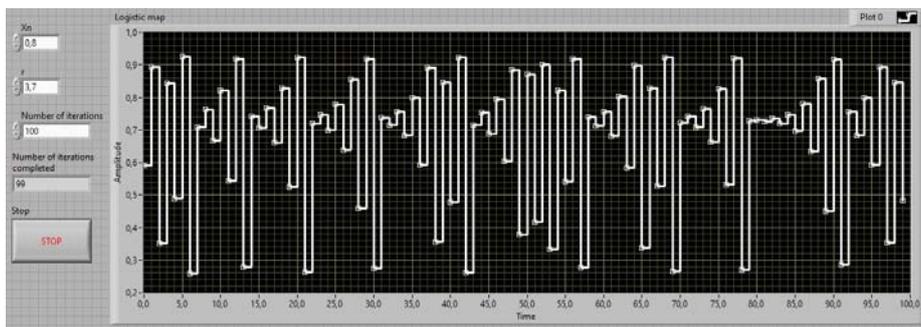


Fig. 4. Iteration series of logistic map for $X_0 = 0.8$

2 ARDUINO board

The top board is Arduino Pro Mini, the bottom – USB-UART / USB-TTL converter and some connectors (Fig. 5).



Fig. 5. Set of the Arduino board, converter and some connectors

Now about the Arduino Pro Mini board / platform. Structurally, it is a board with a microcontroller soldered on it, a RESET button, a power chip, and other peripherals that are not essential for us at this stage.

There are two versions of the Pro Mini: one running at 3.3V at 8 MHz, the other at 5V at 16 MHz. The board is based on the Atmega 168 or Atmega 328 microcontroller – the difference between them is the volume of the system-programmable Flash memory – 16 or 32 kB. This is the so-called "program memory", i.e. the memory in which the program will be recorded and its contents will not change during operation. Atmega is built on the so-called Harvard architecture in which the "program memory" and "data memory" are implemented separately, for greater speed and reliability. The

"data memory" is divided into 2 parts: operational SRAM, which for 168 and 328 is 1 KB, and a permanent EEPROM (Electrically Erasable Programmable Read-Only Memory) of 512 bytes, data from which is not "lost" when the power is turned off.

The connection scheme is quite simple and it's rather difficult to make a mistake (Fig. 6).

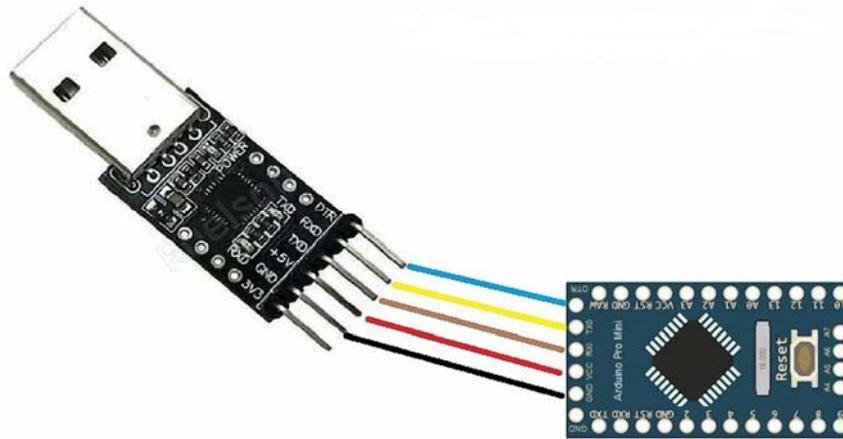


Fig. 6. The connection scheme for programming of the Arduino

3 Program realization

The Arduino integrated development environment is an environment in which an Arduino board can be programmed.

A written program or code is called SKETCH. In this work, the Arduino IDE is used as an environment in which the Arduino Pro Mini program is written, compiled and uploaded on the Arduino board as depicted in Fig. 7.

The Arduino can be connected to a computer through the USB port and programmed using a language similar to C++.

A simple program (sketch) used to operate the microcontroller and visualize the logistic map by looking at the array of blinking LEDs. A glowing LED represents an iteration of the one-dimensional map, and it is linked with a value in the interval $[0;1]$.

As example, the first LED from the left represents the first interval segment $[0;0.1)$, the second LED represents the segment $[0.1; 0.2)$, and so on. The last LED represents the final segment $[0.9; 1.0]$. When the microcontroller iterates the logistic map, a value belonging to one of these ten intervals is returned, and the corresponding LED turns on for 500 ms. This process is repeated infinitely, so we can observe the orbit followed by the logistic map by watching the blinking LEDs sequence.

The initial conditions for the logistic map can be changed and uploaded again into the microcontroller in any time. Thus, we can explore the behavior of the chaotic map when different values of the parameters r and X_0 are chosen.

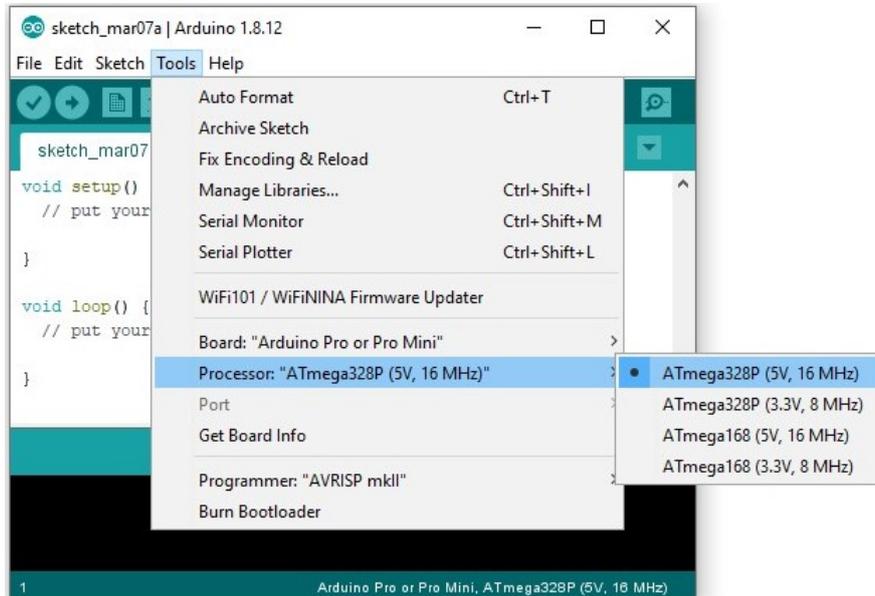


Fig. 7. Arduino IDE

Arduino programs require two mandatory functions: `setup()` and `loop()`. Any variable or constant defined outside these two functions is considered to be global. In the `setup()` function, we tell the microcontroller that there are ten light-emitting diodes connected to the digital pins and that they are intended to be turned on and off. In the `loop()` function, the logistic map is iterated, and the visualization process takes place as we observe the LEDs turning on and off, one after another, following the evolution of the nonlinear system.

The software to connect the LEDs using the following code:

```
// Logistic Map
// Select of the pin for each light-emitting diode (LED)
const int NbrLEDs = 10;
const int LEDpin[] = {4, 5, 6, 7, 8, 9, 10, 11, 12, 13};

const int wait = 500; // Wait for 500 milliseconds

// Parameters of Logistic Map
const double r = 3.8; // Logistic map constant

double X0 = 0.3; // Initial position
double X = X0; // X0 - first calculated point

// For initializes the LED pins use setup ()
void setup() {
```

```

    for (int i = 0; i < NbrLEDs; i++) {
        pinMode(LEDpin[i], OUTPUT);
    }
}

// For turn on/off LEDs and iterates the Logistic Map use
loop ()
void loop ()
{
    if (X < 0.1)
        blinkLED(LEDpin [0]) ;
    else if ((X >= 0.1 ) && (X < 0.2))
        blinkLED(LEDpin[1]);
    else if ((X >= 0.2) && (X < 0.3))
        blinkLED(LEDpin[2]);
    else if ((X >= 0.3) && (X < 0.4))
        blinkLED(LEDpin[3]);
    else if ((X >= 0.4) && (X < 0.5))
        blinkLED(LEDpin[4]);
    else if ((X >= 0.5) && (X < 0.6))
        blinkLED(LEDpin[5]);
    else if ((X >= 0.6) && (X < 0.7))
        blinkLED(LEDpin[6]);
    else if ((X >= 0.7) && (X < 0.8))
        blinkLED(LEDpin[7]);
    else if ((X >= 0.8) && (X < 0.9))
        blinkLED(LEDpin[8]);
    else
        blinkLED(LEDpin[9]);

    // Logistic Map function with iterates
    X0 = X;
    X = r * X0 * (1.0 - X0);
}

// Function for blinkLED
// turn on/off LEDs
void blinkLED(const int pin) {
    digitalWrite(pin, HIGH); // turn LED on
    delay (wait); // wait 500 milliseconds
    digitalWrite(pin, LOW); // turn LED off
}

```

4 Experimental realization

Fig. 8 shows connection scheme for experimental demonstration of chaotic behavior that realized one-dimensional system.

We connected an array of ten light-emitting diodes (HL1-HL10) to the digital pins (D4-D13) of the microcontroller using 220 Ohm resistors (R1-R10).

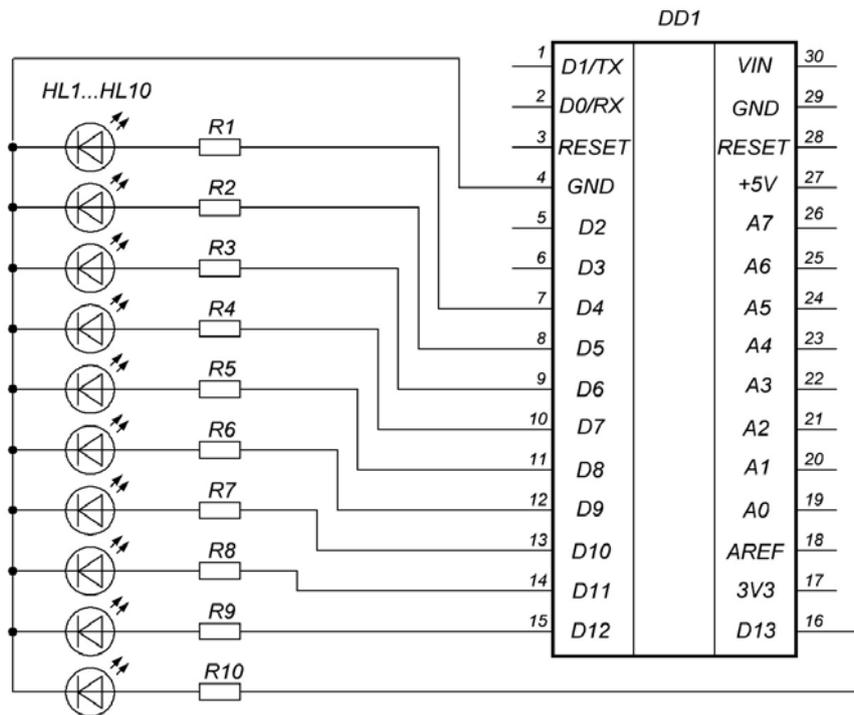


Fig. 8. Connection scheme for experimental demonstration of chaotic behavior

Fig. 9 shows experimental realization of logistic map for demonstration of chaotic behavior.

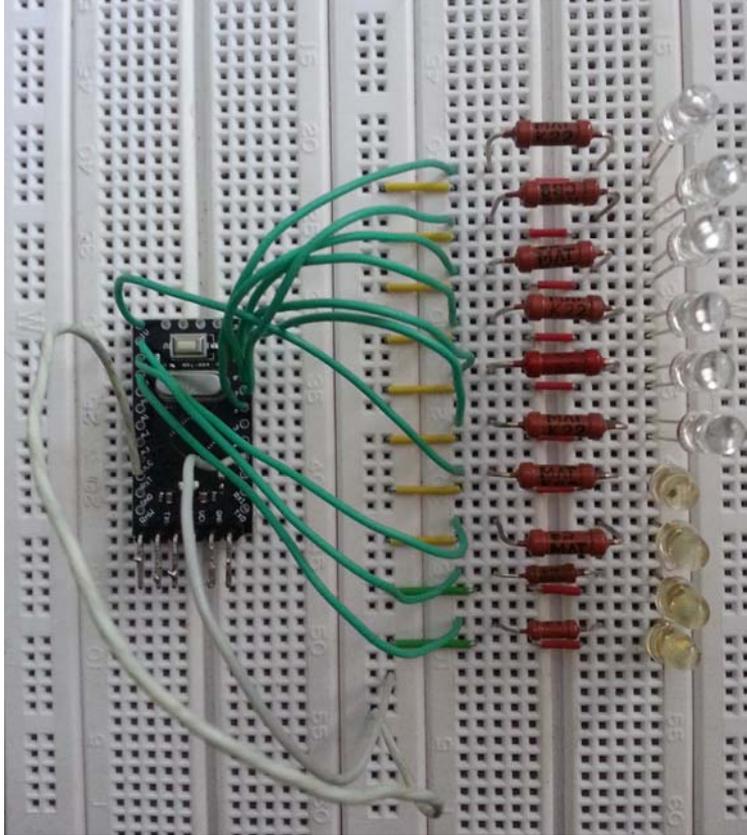


Fig. 9. Experimental realization for demonstration of chaotic behavior

5 Conclusions

The Arduino Pro Mini, ten light-emitting diodes (LEDs) and ten resistors for each part of segment of the range $[0;1]$ was used for practical realization and visualizing of the logistic map. It is one-dimensional system that demonstrate chaotic behavior.

Equation, system conditions and analysis of the iteration series with different parameter r are shown. For this analysis was used modern software environment Lab-View. Connection scheme, nominal components and programming code are also presented.

The Arduino was connected to a computer through the USB port and programmed using a language similar to C++. Sketch was uploaded into Arduino using program software ArduinoIDE.

References

1. Ramazanov, S., Tishkov, B.: Non-linear Forecasting of the State of a Socio-eco-oriented Innovative Economy in the Context of Systemic Crises. *CEUR Workshop Proceedings* **2422**, 181-193 (2019)
2. Pribylova, L.: Bifurcation routes to chaos in an extended Van der Pol's equation applied to economic models. *Electronic Journal of Differential Equations* **53**: 1-21 (2009)
3. Rusyn, V., Savko, O.: Modeling of Chaotic Behavior in the Economic Model. In *8th Chaotic Modeling and Simulation International Conference, Paris, 26-29 May 2015*, pp. 705-712, (2015)
4. Hajnova, V., Pribylova, L.: Two-parameter bifurcations in LPA model. *Journal of Mathematical Biology*. **75**(5): 1235-1251 (2017)
5. Skiadas, C.H. Exact solutions of stochastic differential equations: Gompertz, generalized logistic and revised exponential. *Methodology and Computing in Applied Probability* **12**(2): 261-270 (2010)
6. Skiadas, C.H., Skiadas, C. Chaotic Modelling and Simulation: Analysis of Chaotic Models, Attractors and Forms. In: Taylor & Francis Group, pp. 1-345, LLC (2008)
7. Lingala, N., Namachchivaya, N.S., Pavlyukevich, I., Wedig, W.: Random perturbations of periodically driven nonlinear oscillators. *Procedia IUTAM* **19**, 91-100 (2016)
8. Lingala, N., Namachchivaya, N.S., Pavlyukevich, I.: Random perturbations of a periodically driven nonlinear oscillator: escape from a resonance zone. *Nonlinearity* **30**(4), 1376-1404 (2017)
9. Si, Y., Wang, J., Fečkan, M.: Controllability of linear and nonlinear systems governed by Stieltjes differential equations. *Applied Mathematics and Computation*, **376**, 125139 (2020)
10. Battelli, F., Fečkan, M.: On the Poincaré-Adronov-Melnikov method for the existence of grazing impact periodic solutions of differential equations. *Journal of Differential Equations*, **268**(7), 3725-3748 (2020)
11. Fečkan, M., Sathiyaraj, T., Wang, J.R.: Synchronization of butterfly fractional order chaotic system. *Mathematics*, **8**(3), 446 (2020)
12. Sher, M., Shah, K., Fečkan, M., Khan, R.A.: Qualitative analysis of multi-terms fractional order delay differential equations via the topological degree theory. *Mathematics*, **8**(2), 218 (2020)
13. Rusyn, V., Samila, A., Skiadas, Ch.: Computer modeling and practical realization of chaotic circuit with a light-emitting diode. In: *Fourteenth International Conference on Correlation Optics, Chernivtsi, 16-19 September 2019*, pp. 113690D (2019)
14. Milicka, P., Cížek, P., Faigl, J.: On Chaotic Oscillator-based Central Pattern Generator for Motion Control of Hexapod Walking Robot. *CEUR Workshop Proceedings* **1649**, 131-137 (2016)
15. Chua, L.: Memristor – the missing circuit element. *IEEE Trans. Circuit Theory*. **18**(5): 507-519 (1971)
16. Itoh, M., Chua, L.: Dynamics of Hamiltonian systems and memristor circuits. *International Journal of Bifurcation and Chaos*, **27**(2): 1730005 (2017)
17. Wang, Z.F., Shi, L., Wu, H., Helian, N., Chua, O.L.: Fractional memristor. *Applied Physics Letter* **111**, 243502 (2017)
18. Rusyn, V., Khrapko, S.: Memristor: modeling and research of information properties. *Springer Proceedings in Complexity*. Springer, pp. 229-238 (2019)
19. Yu, D., Zheng, C., Iu, H. H. C., Fernando, T., Chua, L. O.: A new circuit for emulating memristors using inductive coupling. *IEEE Access* **5**, 1284-1295 (2017)

20. Chua, L.: Memristor, Hodgkin–Huxley, and edge of chaos. *Nanotechnology* 24(38): 383001 (2013)
21. Vaidyanathan, S., Sambas, A., Kacar, S., & Çavuşoğlu, Ü.: A new three-dimensional chaotic system with a cloud-shaped curve of equilibrium points, its circuit implementation and sound encryption. *International Journal of Modelling, Identification and Control*, 30(3): 184-196 (2018)
22. Vaidyanathan S, Sambas A, Kacar S and Cavusoglu U.: A new finance chaotic system, its electronic circuit realization, passivity based synchronization and an application to voice encryption. *Nonlinear Engineering*, 8(1): 193-205 (2019)
23. Vaidyanathan, S., Azar, A. T., Rajagopal, K., Sambas, A., Kacar, S., & Çavusoglu, Ü.: A new hyperchaotic temperature fluctuations model, its circuit simulation, FPGA implementation and an application to image encryption. *IJSPM*, 13(3): 281-296 (2018)
24. Rusyn, V.: Modeling and Research Information Properties of Rucklidge Chaotic System Using LabView. In: 10th Chaotic Modeling and Simulation International Conference, Barcelona, 30 May – 2 June 2017, pp. 739-744, (2015)
25. Rusyn V., Kushnir M., Galameiko O.: Hyperchaotic Control by Thresholding Method. In: 11th Modern Problems of Radio Engineering, Telecommunications and Computer Science International Conference, TCSET 2012 - Proceedings, Lviv-Slavske, 21-24 February 2012, art. No. 6192785, p. 67 (2012)
26. Di Marco, M., Innocenti, G., Forti, M., Tesi, A.: Control Design for Targeting Dynamics of Memristor Murali-Lakshmanan-Chua Circuit. In 18th European Control Conference (ECC), pp. 4332-4337 (2019).
27. Vaidyanathan, S., Feki, M., Sambas, A., Lien C. H.: A new biological snap oscillator: its modelling, analysis, simulations and circuit design. *International Journal of Simulation and Process Modelling*, 13(5): 419-432 (2018)
28. Vaidyanathan, S., Azar, A. T., Sambas, A., Singh, S., Alain, K. S. T., Serrano, F. E.: A novel hyperchaotic system with adaptive control, synchronization, and circuit simulation. In: *Advances in System Dynamics and Control*, pp. 382-419, IGI Global (2018)
29. Vaidyanathan, S., Dolvis, L. G., Jacques, K., Lien, C. H., Sambas, A.: A new five-dimensional four-wing hyperchaotic system with hidden attractor, its electronic circuit realisation and synchronisation via integral sliding mode control. *International Journal of Modelling, Identification and Control*, 32(1): 30-45 (2019)
30. Vaidyanathan, S., Sambas, A., Zhang, S.: A new 4-D dynamical system exhibiting chaos with a line of rest points, its synchronization and circuit model. *Archives of Control Sciences*, 29(3): 485-506 (2019)
31. Rusyn, V., Skiadas, Ch.: *Threshold Method for Control of Chaotic Oscillations*. Springer Proceedings in Complexity. Springer, pp. 217-229 (2020)
32. Chua, L.: A zoo of strange attractors from the canonical Chua's circuits. In: *Proceedings of the IEEE 35th Midwest Symposium on Circuits and Systems*, Cat. No. 92CH3099-9, Vol. 2, pp. 916-926 (1992)
33. Rusyn, V., Mohamed, M.A., Purwandari, D., Mamat, M., Titaley, J., Pinontoan, B.: Chaotic and Controlling Regimes of a New Modified Chua's Generator. *Journal of Advanced Research in Dynamical and Control Systems*, 12(2), 556-561 (2020)
34. Rusyn, V., Sadli, M., Mamat, M., Mujiarto, Sanjaya Mada, W.S.: Computer Modelling of a New Simple Chaotic Generator. *Journal of Physics: Conference Series*, 1477, 022010 (2020)
35. May, Robert M.: Simple mathematical models with very complicated dynamics. *Nature*. 261(5560): 459-467 (1976)