Analog Integrated Circuits and Signal Processing (2022) 113:145–161
https://doi.org/10.1007/s10470-022-02073-4

Synchronization of hyperchaotic Wang-Liu system with experimental implementation on FPAA and FPGA

Gülnur Yılmaz[1] · **Kenan Altun**[2] · **Enis Günay**[1]

Received: 21 February 2022 / Revised: 24 May 2022 / Accepted: 23 June 2022 / Published online: 6 July 2022

Abstract
This paper is concerned with the synchronization of hyperchaotic Wang-Liu system by using the Pecora-Carroll complete replacement method. One of the most important features of this method is no need for a controller on the slave system which is the basic requirement of other synchronization methods. Initially in this study, dynamic analyses are given to describe the structure of the system. Then, the synchronization is investigated and master and slave systems that have two different dynamics are synchronized. At this point, the hyperchaotic master system forces the quasi-periodic slave system to behave as itself and they demonstrate the same hyperchaotic dynamics after a while. Besides the simulations, synchronization of the two systems is also verified by the experimental realization. FPAA and FPGA implementations of hyperchaos synchronization are provided and results are considered. In the end, it is specified that the experimental results are well-coincides with the simulations.

Keywords Hyperchaotic systems · Synchronization · Complete replacement · Field Programmable Analog Arrays · Field Programmable Gate Arrays

1 Introduction

Chaos is a term that simply defines the complexity of nonlinear dynamical systems. Chaotic systems are extremely sensitive to their system parameters and initial conditions and the dynamics of the system can change even small variations happen. First most-known observation of chaos was realized by Lorenz in his study about the weather forecast in 1969 and sensitivity to initial conditions is proved in that study [1]. Chaotic systems should also have at least one positive Lyapunov exponents and three system dimensions to be able to exhibit nonlinear chaotic dynamics.

Hyperchaotic systems are the type of chaotic systems that demonstrate more complex dynamics. Having more than one positive Lyapunov exponents and at least four system dimensions are the characteristics of hyperchaotic systems. Since Rössler [2] first invented the hyperchaos, several hyperchaotic systems have been proposed [3–8] and they intensively implemented in various areas such as communication, cryptology, nonlinear systems, and control [9–21].

Chaos synchronization has received great attention because of being a challenging problem in nonlinear systems [22]. Since Pecora and Carroll introduced the famous synchronization method in 1990 [23], several approaches have been proposed for three decades. Adaptive synchronization [24–27], complete synchronization [7, 28], projective synchronization [29–31], finite-time synchronization [32, 33] and hybrid synchronization [21, 34] are some special impressive methods used for the synchronization of nonlinear dynamical systems.

In the chaos synchronization phenomena, there are two nonlinear systems called master (drive) and slave (response) systems. The idea of synchronization is to control the slave system by using the signals coming from the master system [35]. Trajectories of the slave system can thus converge to the output of the master's. One method used for the synchronization is the complete replacement that is proposed by Pecora and Carroll [36]. This method is implemented by coupling the master and slave systems with a common signal. Therefore, design of a controller on the slave system side is not required. That feature makes

✉ Gülnur Yılmaz
gulnur.yilmaz@erciyes.edu.tr

[1] Department of Electrical & Electronics Engineering, Erciyes University, Kayseri, Turkey

[2] Sivas Vocational College, Cumhuriyet University, Sivas, Turkey

the complete replacement method as an efficient and simple way of synchronization.

Chaotic and hyperchaotic systems can be experimentally implemented on electronic circuits using passive elements, and integrated circuits including FPAAs (Field Programmable Analog Arrays) and FPGAs (Field Programmable Gate Arrays) [37]. The process of designing, implementing, and testing an electronic circuit causes lots of problems such as cost, time, complexity, efficiency, and feasibility of the system. At this point, integrated circuits namely FPAA and FPGA are an alternative and practical way to implement chaotic and hyperchaotic system [38]. Experimental realization based on FPAA has the advantages of simple and fast design for complex models, reduced cost, reconfigurability even during implementation, fast response and programmable structure [39–41]. On the other hand, an FPGA has the similar advantages with FPAA while providing higher performance, fast prototyping and flexibility [42, 43].

The rest of this paper is organized as follows: Sect. 2 includes the system description and simple dynamical analyses of the Wang-Liu system. In Sect. 3, synchronization of the master and slave systems is realized by using the complete replacement method and simulation results are provided. Methodology for the FPAA implementation is described in Sect. 4. The implementation results of the synchronized systems on FPAA and FPGA are given in Sects. 5 and 6, respectively. Finally, the motivation of this study, advantages of the complete replacement method for the synchronization and the simulation and experimental results are discussed in Sect. 7.

2 System description

Four dimensional hyperchaotic system proposed by Wang and Liu [5] is given in Eq. (1):

$$\begin{aligned} c\dot{x} &= a(y-x) \\ \dot{y} &= bx - kxz + w \\ \dot{z} &= -cz + hx^2 \\ \dot{w} &= -dx \end{aligned} \tag{1}$$

where x, y, z, w are the state variables and a, b, c, d, h and k are the constant parameters of the dynamic model. For the values of $(a, b, c, d, h, k) = (10, 40, 2.5, 10.6, 1, 4)$ and the initial conditions of $(x_0, y_0, z_0, w_0) = (0.1, 0.1, 0.1, 0.1)$, Lyapunov exponents of this dynamical system are calculated as:

$\lambda_{L1} = 1.1498$ $\lambda_{L2} = 0.0276$ $\lambda_{L3} = 0.0185$ $\lambda_{L4} = -13.665$

Since there exist more than one positive Lyapunov exponents, the system exhibits hyperchaotic behavior. For different values of the d parameter, system (1) demonstrates also periodic, quasi-periodic and chaotic dynamical behaviors. Phase portraits for different dynamics are obtained on Matlab and depicted in Fig. 1.

Dynamical structure of the Wang-Liu system is specified by the Lyapunov exponents and bifurcation diagrams on Matlab. The results are given in Fig. 2(a), (b), respectively. System (1) is analyzed by increasing the parameter d gradually while others are constant as $a = 10$, $b = 40$, $c = 2.5$, $h = 4$, and $k = 1$. In the Lyapunov exponents graph shown in Fig. 2(a), the system has two positive Lyapunov exponents for the values of d greater than 0. However, different types of behaviors are observed such as periodic, quasi-periodic and chaotic as the value of d is increased up to 80. Furthermore, bifurcation diagram of the system has the same dynamics seen in the Lyapunov exponents. Thus, dynamics of the Wang-Liu system are verified, and consistent results are obtained. More detailed analysis of the system is provided in the relative references [5, 44].

3 Synchronization with complete-replacement method

In chaos synchronization, there are two systems, master and slave systems, to be synchronized. Master system defined with the subscript m is given in Eq. (2):

$$\begin{aligned} c\dot{x}_m &= a(y_m - x_m) \\ \dot{y_m} &= bx_m - kx_m z_m + w_m \\ \dot{z}_m &= -cz_m + h{x_m}^2 \\ \dot{w_m} &= -d_m x_m \end{aligned} \tag{2}$$

and the slave system with subscript s is described in Eq. (3):

$$\begin{aligned} c\dot{x}_s &= a(y_s - x_s) \\ \dot{y}_s &= bx_s - kx_s z_s + w_s \\ \dot{z}_s &= -cz_s + h{x_s}^2 \\ \dot{w}_s &= -d_s x_s \end{aligned} \tag{3}$$

Both master and slave systems have the constant parameters as $(a, b, c, h, k) = (10, 40, 2.5, 1, 4)$. The only difference between the master and slave systems is the value of the d parameter. d_m equals 10.6 for the master system while d_s is 77 for the slave system. Defining d_m parameter as 10.6 makes the master system hyperchaotic. On the other hand, slave system exhibits quasi-periodic dynamics with d_s equals 77.

Before the synchronization, these two systems are analyzed separately in terms of phase portraits and time series. Phase portraits for the master and slave systems are shown in Fig. 3(a), (b), respectively. In addition, time series for the x and z state variables of master and slave systems are provided and given in Fig. 3(c), (e). As obviously seen that these two systems have different behaviors on time even though only one parameter changes. Moreover, the state

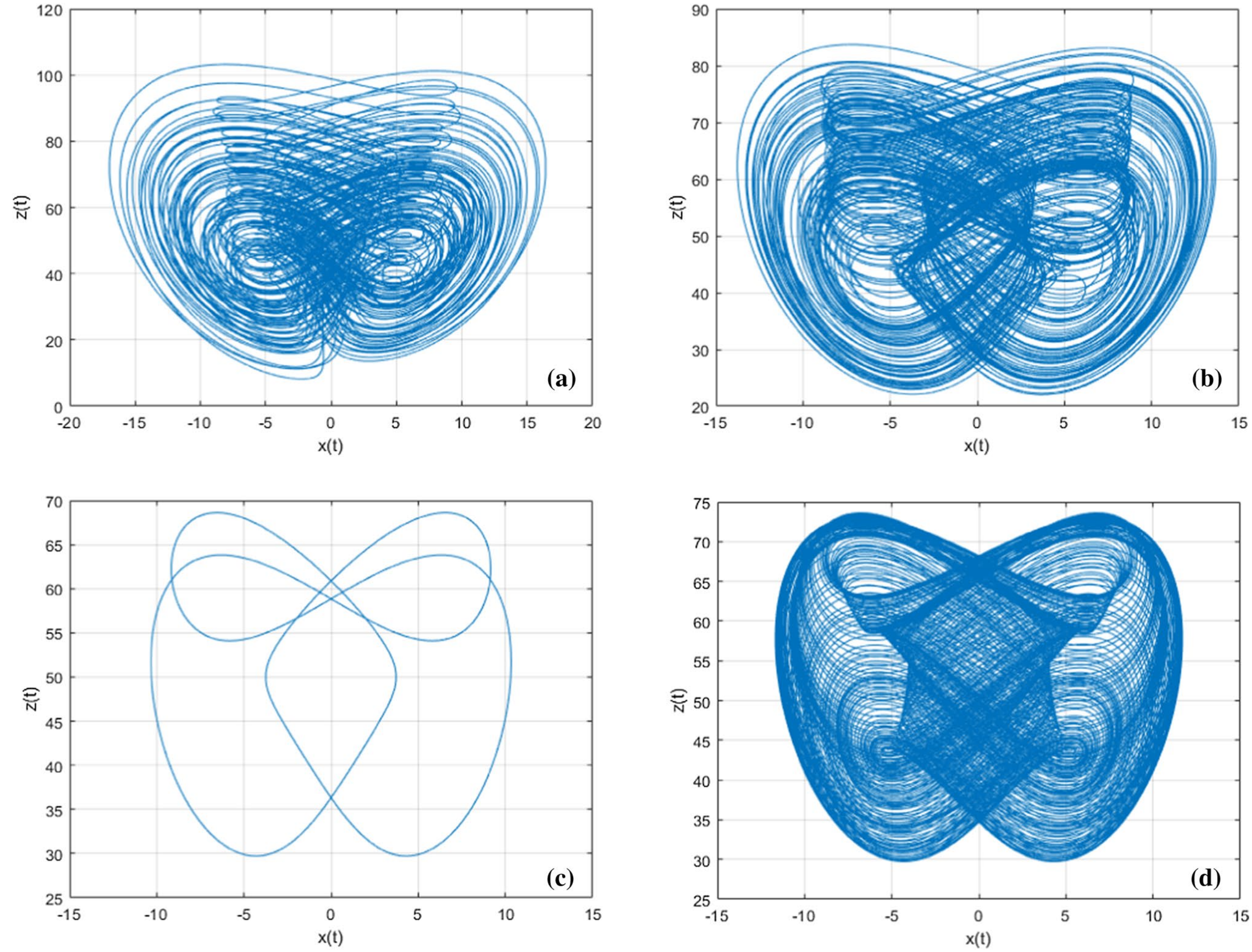


Fig. 1 Phase-space representation of Wang-Liu system for increasing d parameter while (a, b, c, k, h) equals (10, 40, 2.5, 1, 4). **a** hyper-chaotic behavior when d = 10.6; **b** chaotic behavior when d = 50; **c** periodic behavior when d = 55; **d** quasi-periodic behavior when d = 77 (Color figure online)

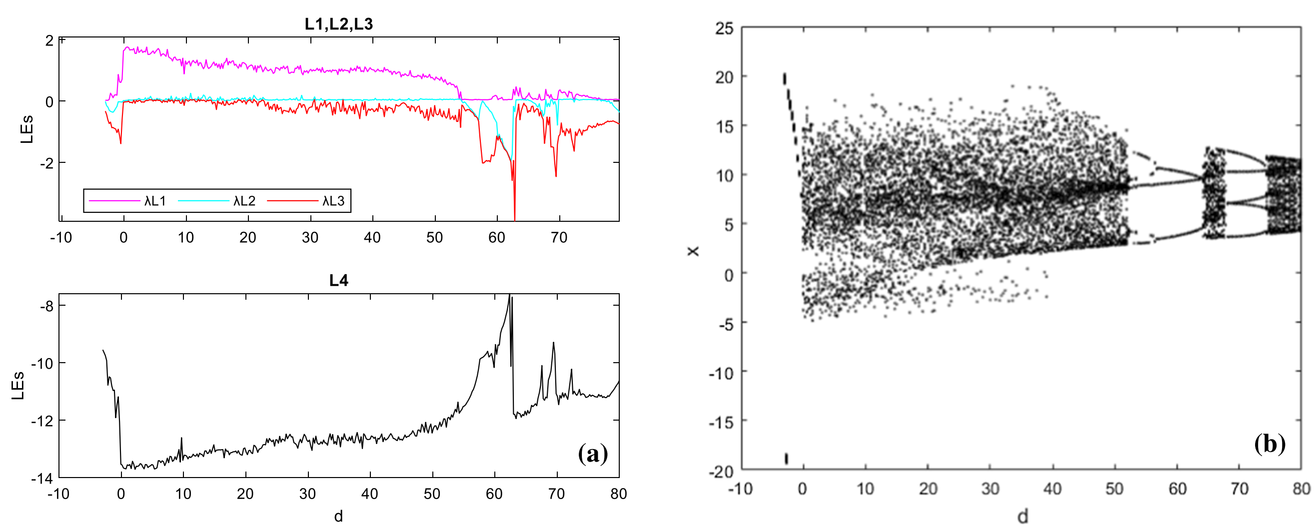


Fig. 2 **a** Lyapunov exponents for increasing *d* value; **b** bifurcation diagrams of the system (Color figure online)

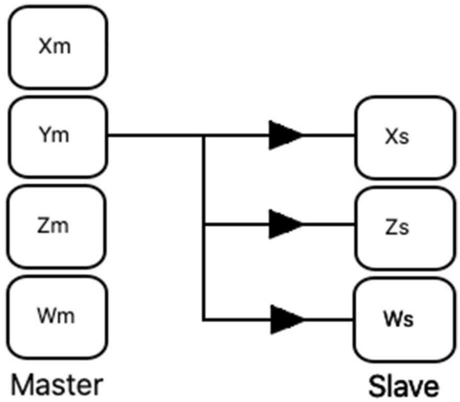


Fig. 3 Simulation results of not synchronized master and slave systems **a** phase portraits for the x–z planes of master system; **b** phase portraits for the x–z planes of slave system; **c** time series of x_m and x_s state variables; **d** phase portraits of x_m vs x_s state variables; **e** time series of z_m and z_s state variables; **f** phase portraits of z_m vs z_s state variables

of x and z variables for the master and slave systems are observed and given in Fig. 3(d), (f). Because the systems are out of synchronization, the graph of x_m vs x_s and z_m vs z_s has a complex shape.

In order to achieve synchronization of system (2) and (3), complete replacement method is adopted. The concept of this method is coupling/linking master and slave systems with a common signal and synchronizing them. There is no need to design a controller for the slave system, coupling is sufficient for the synchronization. The structure of the complete replacement method is illustrated in Fig. 4.

Let the common signal be the y component of the master system. In this case, y_m signal is transmitted to the slave system as the common signal and applied to every point where y_s variable is seen. Therefore, y_s equation is eliminated and all y_s terms are replaced by the y_m on the slave system side. New form of the systems can be expressed in seven dimensions as given below:

$$\begin{aligned}
c\dot{x}_m &= a\left(y_m - x_m\right)\\
\dot{y}_m &= bx_m - kx_m z_m + w_m\\
\dot{z}_m &= -cz_m + hx_m{}^2\\
\dot{w}_m &= -d_m x_m\\
\dot{x}_s &= a\left(y_m - x_s\right)\\
\dot{z}_s &= -cz_s + hx_s{}^2\\
\dot{w}_s &= -d_s x_s
\end{aligned} \tag{4}$$

In system (4), y_m signal forces the slave system to behave as itself. The slave system thus can no longer exhibit quasi-periodic behavior and trajectories of it converge to the master system. Hence, these two systems demonstrate the same dynamics after a while and be synchronized. Phase portraits for the master and slave systems after synchronization are depicted in Fig. 5(a), (b), respectively. Furthermore, exact same graphs of time series for the x_m - x_s and z_m - z_s variables are shown in Fig. 5(c), (e), respectively. Besides these results, synchronization is also observed on the x_m variable and its counterpart x_s variable. The graph given in Fig. 5(d) includes a linear line with 45° of slope which indicates and demonstrates the perfect synchronization between two systems. Fig. 5(f) also confirms the synchronization between the z_m and z_s signals.

4 FPAA implementation methodology

Field Programmable Analog Arrays (FPAA) are the programmable platforms with producing analog outputs. This feature makes suitable FPAAs for the experimental realization of nonlinear dynamical systems. The FPAA implementation steps need to be pursued are given as follows:

- Let $\dot{x}_n = F(x_n)$ is at least 3-dimensional ($n \geq 3$) autonomous nonlinear system. This chaotic or hyperchaotic system is first constructed on Simulink to define voltage levels of the state variables$x_1, x_2, \ldots, x_n$. If the voltage levels are more than $\pm 1.5\ V$, then the system is rescaled to $\pm 1.5\ V$ because of the voltage level limitation of FPAA board.
- Laplace Transform of the rescaled system is obtained to study in frequency domain instead of time domain as follows:
  $s.x_n(s) = F(x_n(s))$ n = 3, 4, …, n.
- SumFilter blocks which are used for modeling the state variables of the dynamical system have the output function as given in Eq. (5):

$$V_{out}(s) = \frac{2\pi f_0\left[\pm G_1 V_{in1}(s) \pm G_2 V_{in2}(s) \pm G_3 V_{in3}(s)\right]}{s + 2\pi f_0}$$

In order to the Laplace-transformed system be compatible with the output function of SumFilter block, $\varepsilon x_n(s)\varepsilon, n = 3, 4, \ldots, n$ variables are added to the set of equations in the system.

$$s.x_n(s) + x_n(s) = F(x_n(s)) + x_n(s)$$

Therefore, $(s+1)$ terms are obtained in the denominator and equations of the system can be modelled by using SumFilter blocks.

$$x_n(s) = \frac{F(x_n(s)) + x_n(s)}{s + 1}$$

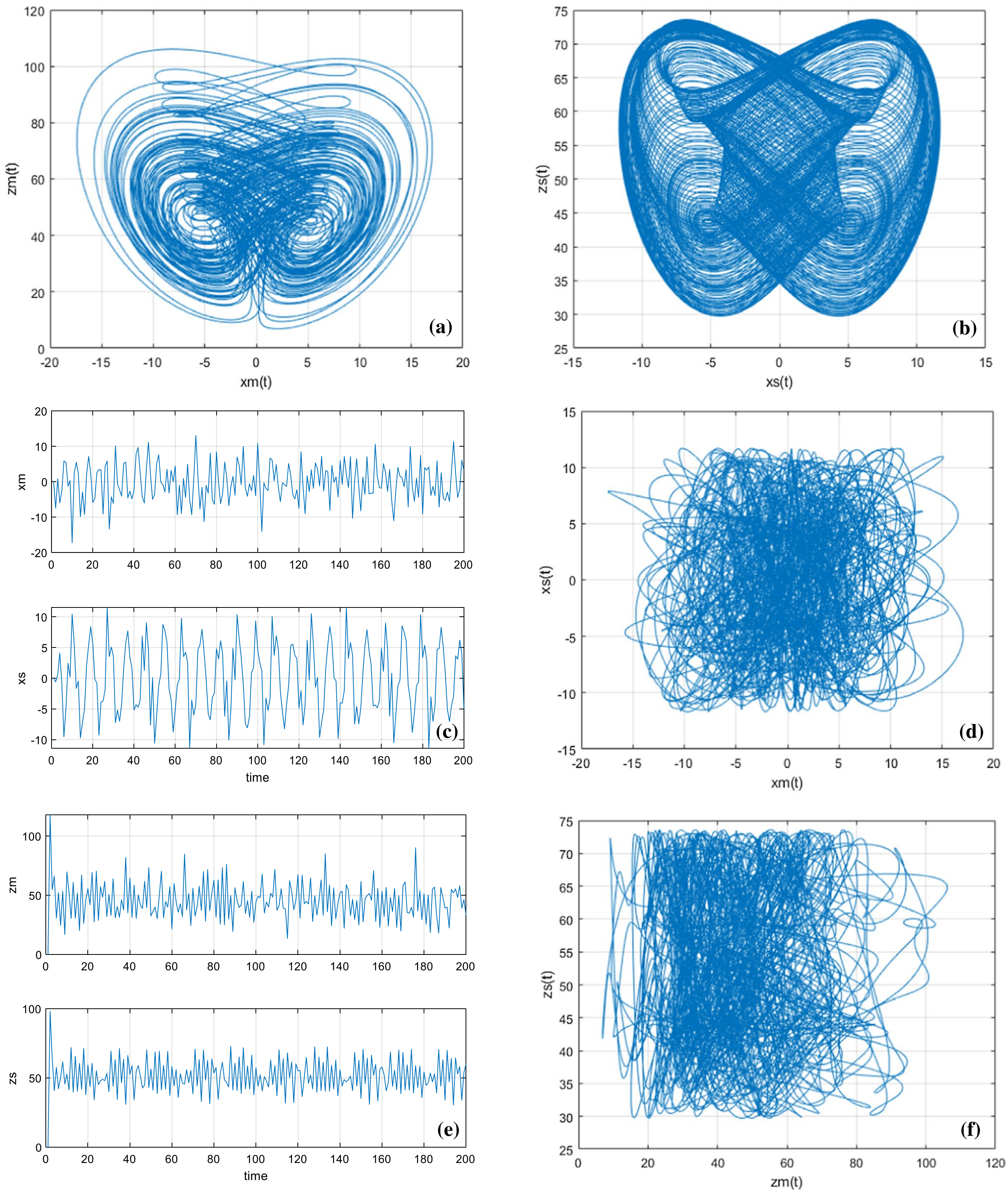


Fig. 4 Structure of the master and slave system for the complete replacement synchronization (Color figure online)

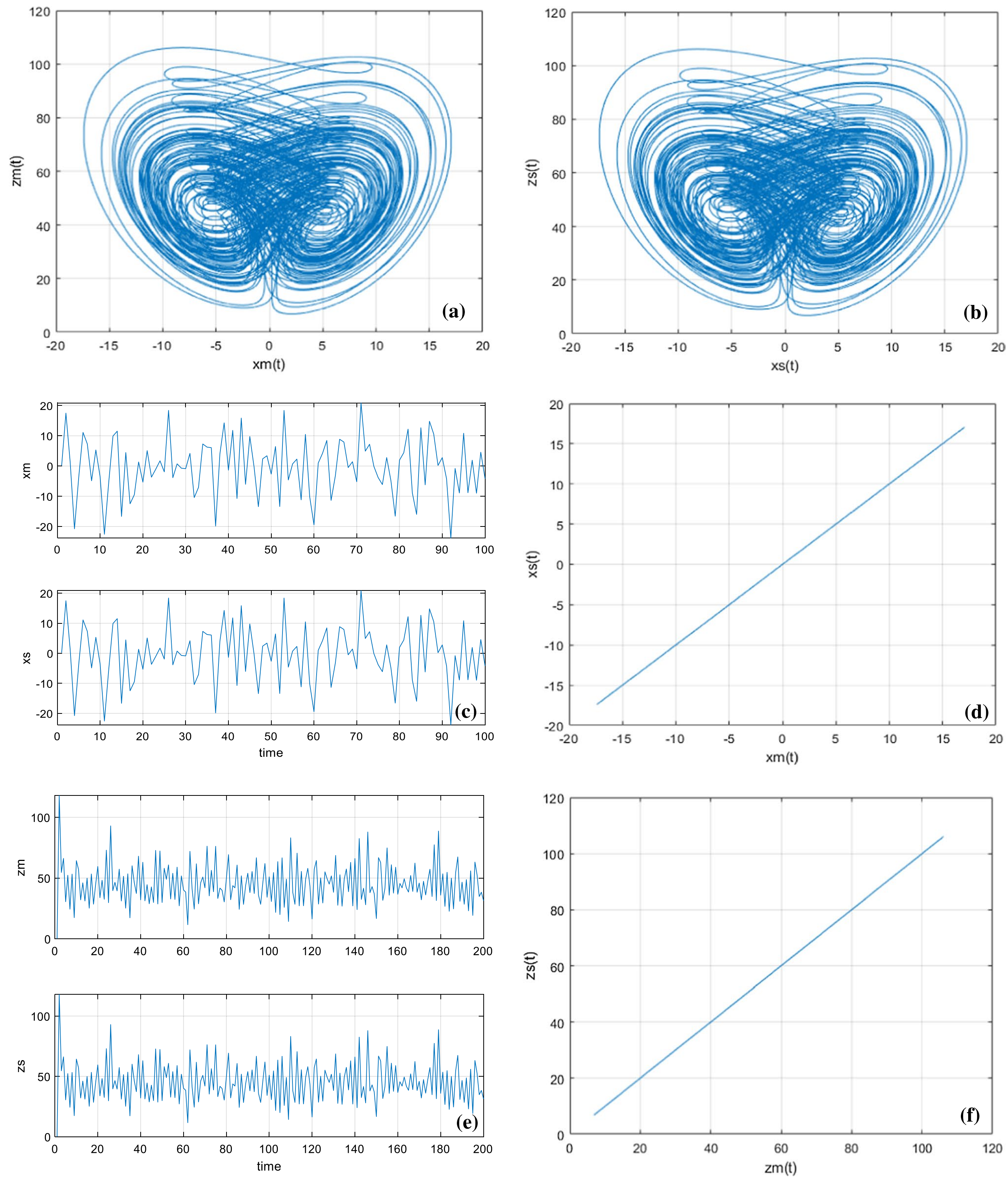


Fig. 5 simulation results of synchronized master and slave systems **a** phase portraits for the x–z planes of master system; **b** phase portraits for the x–z planes of slave system; **c** time series of x_m and x_s state variables; **d** phase portraits of x_m vs x_s state variables; **e** time series of z_m and z_s state variables; **f** phase portraits of z_m vs z_s state variables (Color figure online)

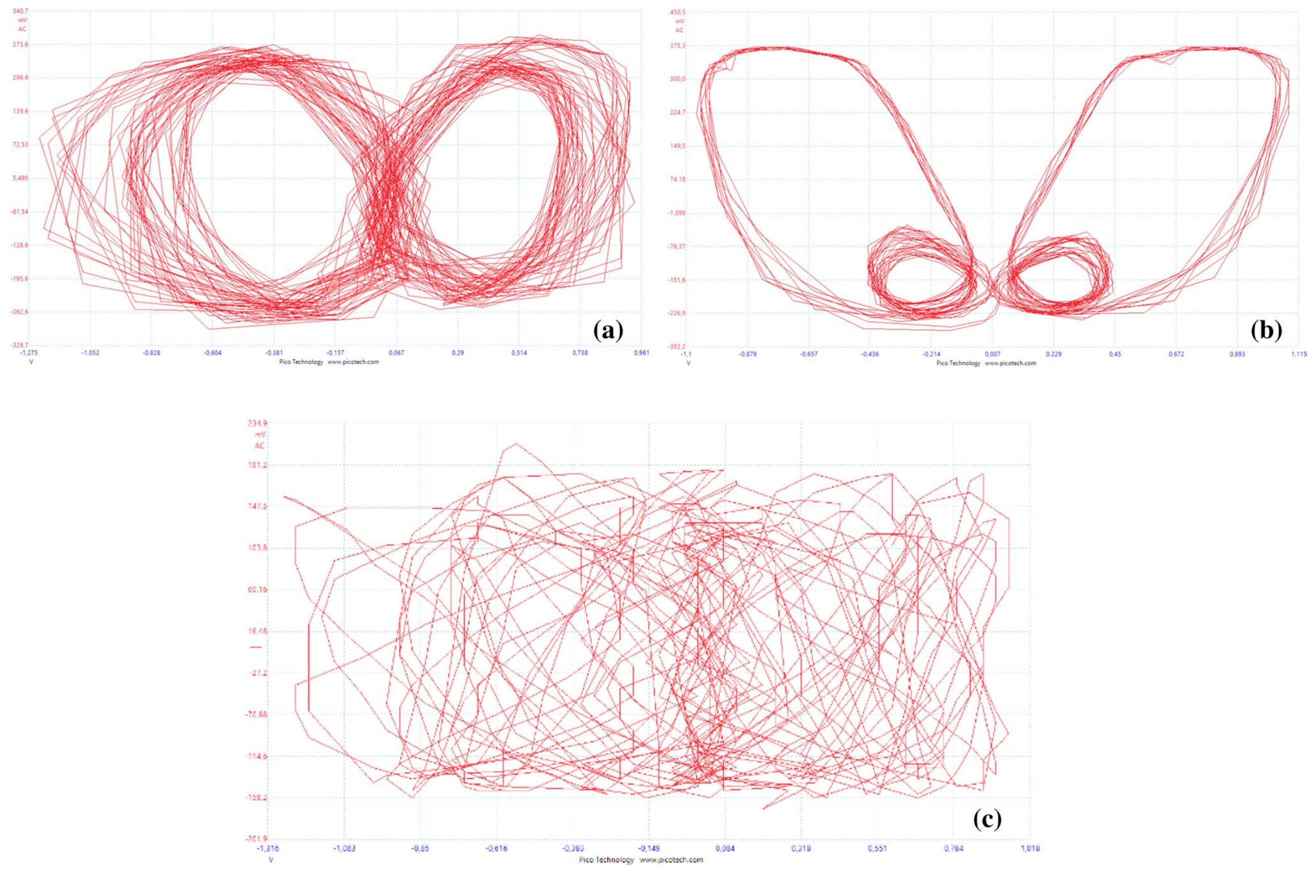


Fig. 6 FPAA implementation results for the not synchronized master and slave systems **a** phase portraits of x–z plane in master system; **b** phase portraits of x–z plane in slave system; **c** graph of x_m vs x_s state variables (Color figure online)

- New form of the system is now modelled on Anadigm Designer 2 which is the computer interface of the FPAA board.
- When the model is ready, experimental setup is prepared and input–output pins are connected as defined in the model.
- Finally desired results are observed by using an oscilloscope.

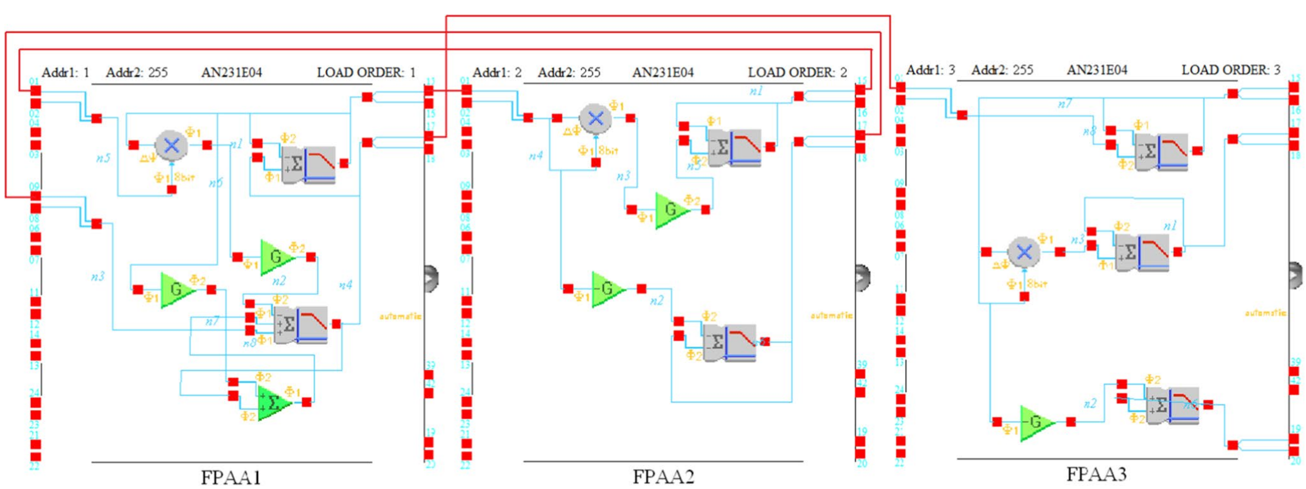


Fig. 7 FPAA model for the synchronized system on Anadigm Designer 2 (Color figure online)

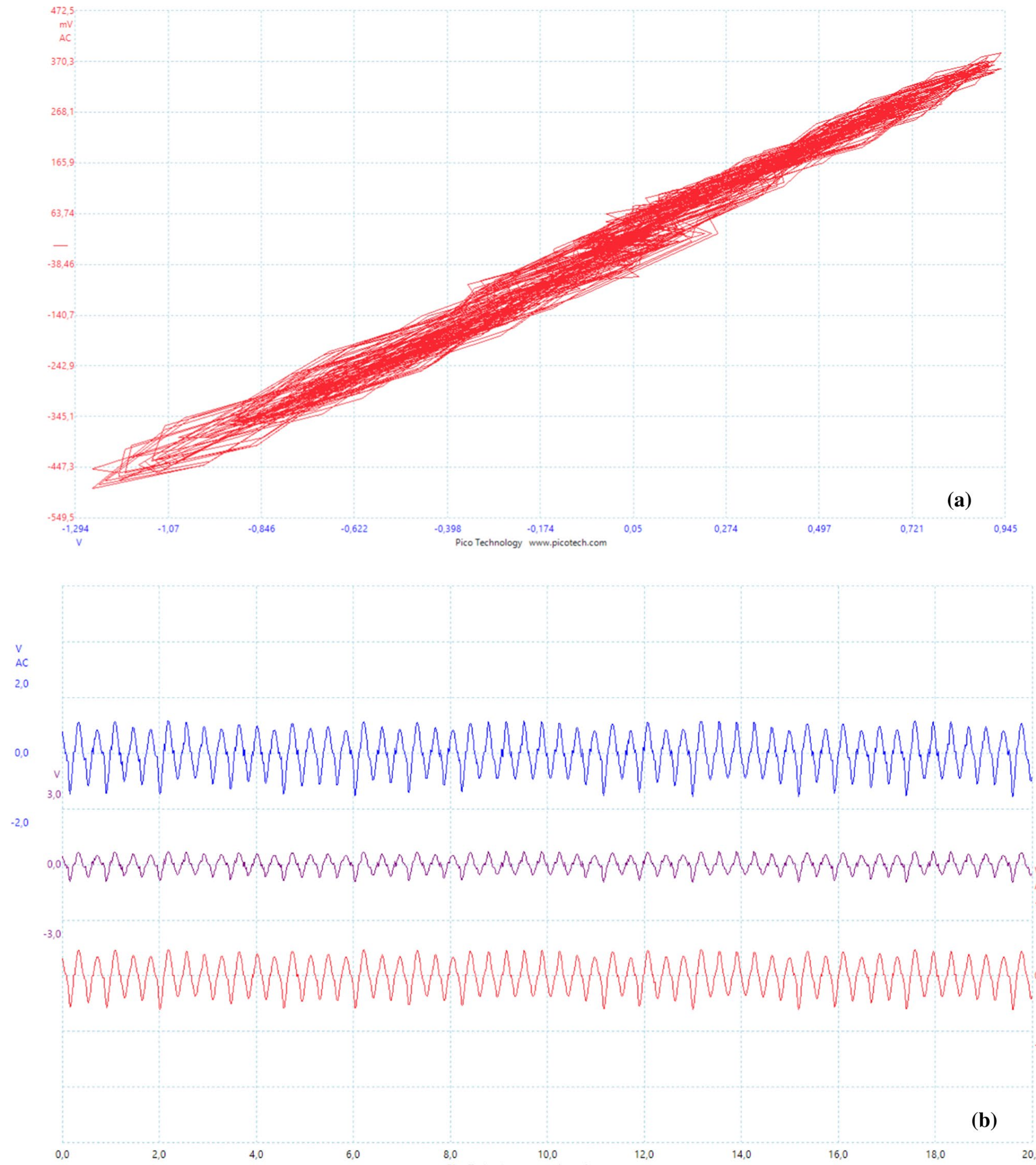


Fig. 8 FPAA implementation results for the synchronized master and slave systems **a** graph of x_m vs x_s state variables; **b** time series of x_m and x_s state variables: blue represents x_m, red represents x_s and purple is the x_m-x_s(difference signal) (Color figure online)

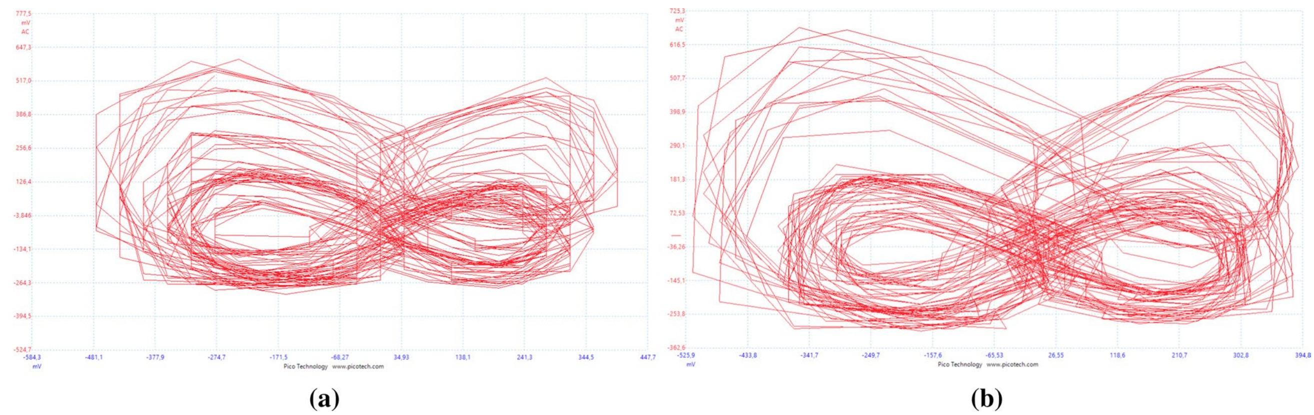


Fig. 9 FPAA implementation results for the synchronized master and slave systems **a** phase portraits of x–z plane in master system; **b** phase portraits of x–z plane in slave system (Color figure online)

5 FPAA implementation of the synchronized Wang-Liu system

FPAA implementation of hyperchaotic Wang-Liu system was realized in our previous study and system behavior was observed under the effect of corner frequency change [44]. In this Section, experimental realization of two synchronized Wang-Liu systems is implemented on FPAA. Before the synchronization, master and slave systems given in Eqs. (2) and (3) are performed individually. Initial conditions of a nonlinear system cannot be defined in FPAA model. For this reason, different dynamic behaviors are obtained by changing the corner frequency values of SumFilter blocks. Therefore, corner frequencies of all state variables are defined as 0.83 kHz to obtain hyperchaotic dynamics for the master system. On the other hand, state variables of the slave system have the corner frequencies as $f_x = f_y = 0.5\ kHz$, $f_z = 0.3\ kHz$ and $f_w = 0.2\ kHz$ and 2-T periodic behavior is observed. Phase portraits of *x-z* plane for the master and slave systems are given in Fig. 6(a), (b), respectively. These graphs demonstrate that two systems exhibit different dynamical behaviors. Not-synchronized structure of the systems is also obtained by the phase portraits of x_m *vs* x_s variables as shown in Fig. 6(c). When the complex shape of this graph is compared to the simulations, expected results are observed.

Synchronized system described in Eq. (4) is then implemented on FPAA. The implementation model constructed on Anadigm Designer 2 is given in Fig 7. First two FPAA chips demonstrate the modelled master system. As seen in the Fig. 7, y_m signal obtained from the output of first FPAA chip is connected to the input of third chip where the slave system is defined. This modelled system is then uploaded to the AN231E04 FPAA board and outputs are measured with a digital oscilloscope. In order to specify whether two systems are synchronized, the synchronization curve is obtained by using the x_m and x_s state variables. The graph given in Fig. 8(a) exhibits a similar x_m vs x_s curve obtained in the simulations. Even if the result has noise because of the FPAA board and do not show a perfect linear line, it indicates the existence of synchronization. In addition, time series of the state variables is observed and depicted in Fig. 8(b). x_m and x_s signals exhibit almost the same behavior as time goes on and synchronization can be seen clearer in this way. However, due to the existence of noise, difference between the x_m and x_s signals is not equal to zero. Phase portraits for the synchronized master and slave systems is depicted in Fig. 9.

According to the results obtained from the FPAA implementation, it is observed that the graphs given in Fig. 8 are noisy. In order to eliminate this noise and further idealize the results, state variables of Eq. (4) are passed through a low-pass filter. The cut-off frequency of the low-pass filter

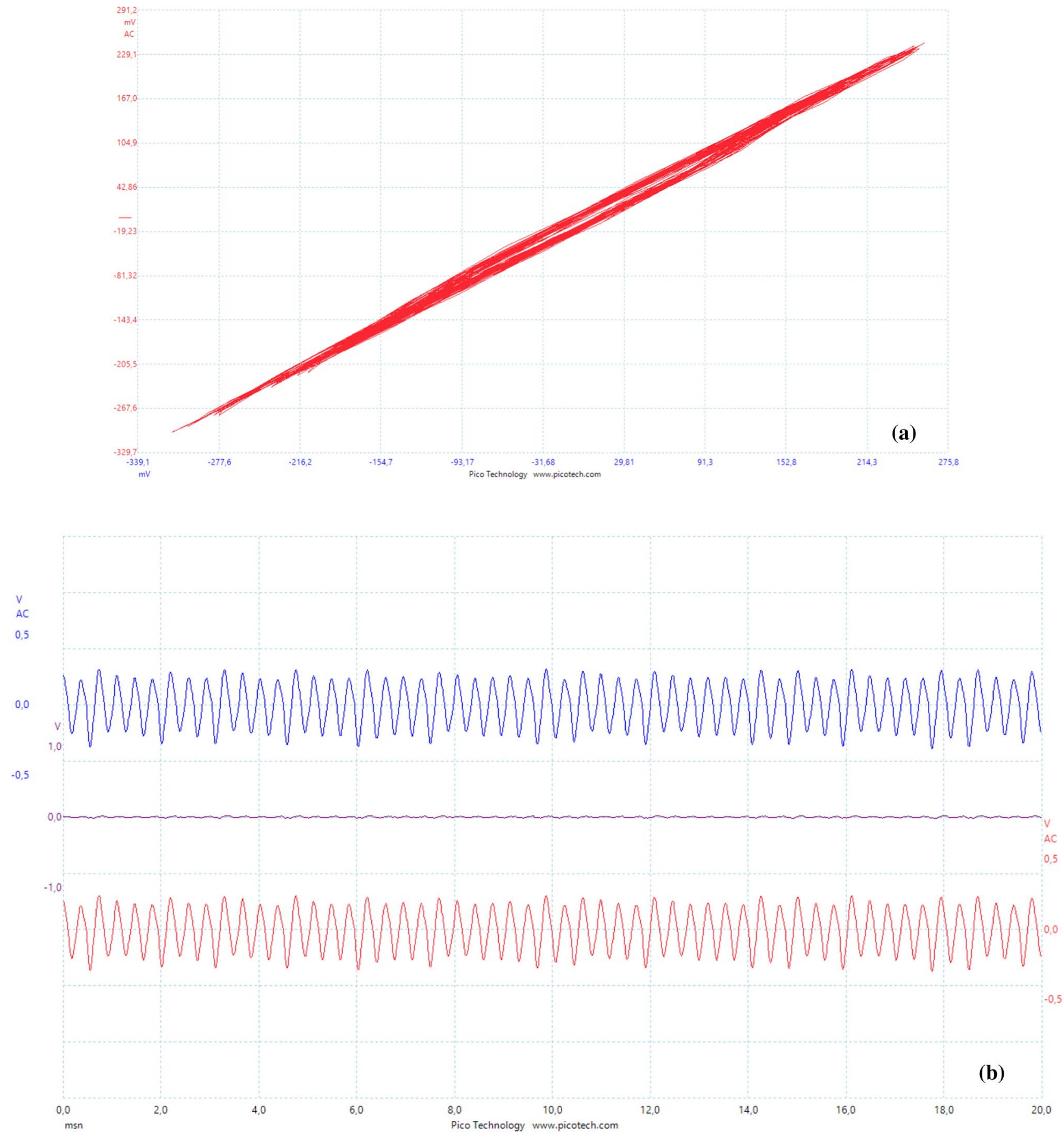


Fig. 10 FPAA implementation results for the synchronized master and slave systems after filtering: **a** synchronization curve (x_m vs x_s) **b** time series of x_m and x_s state variables: blue represents x_m, red represents x_s and purple is the $x_m - x_s$ (difference signal) (Color figure online)

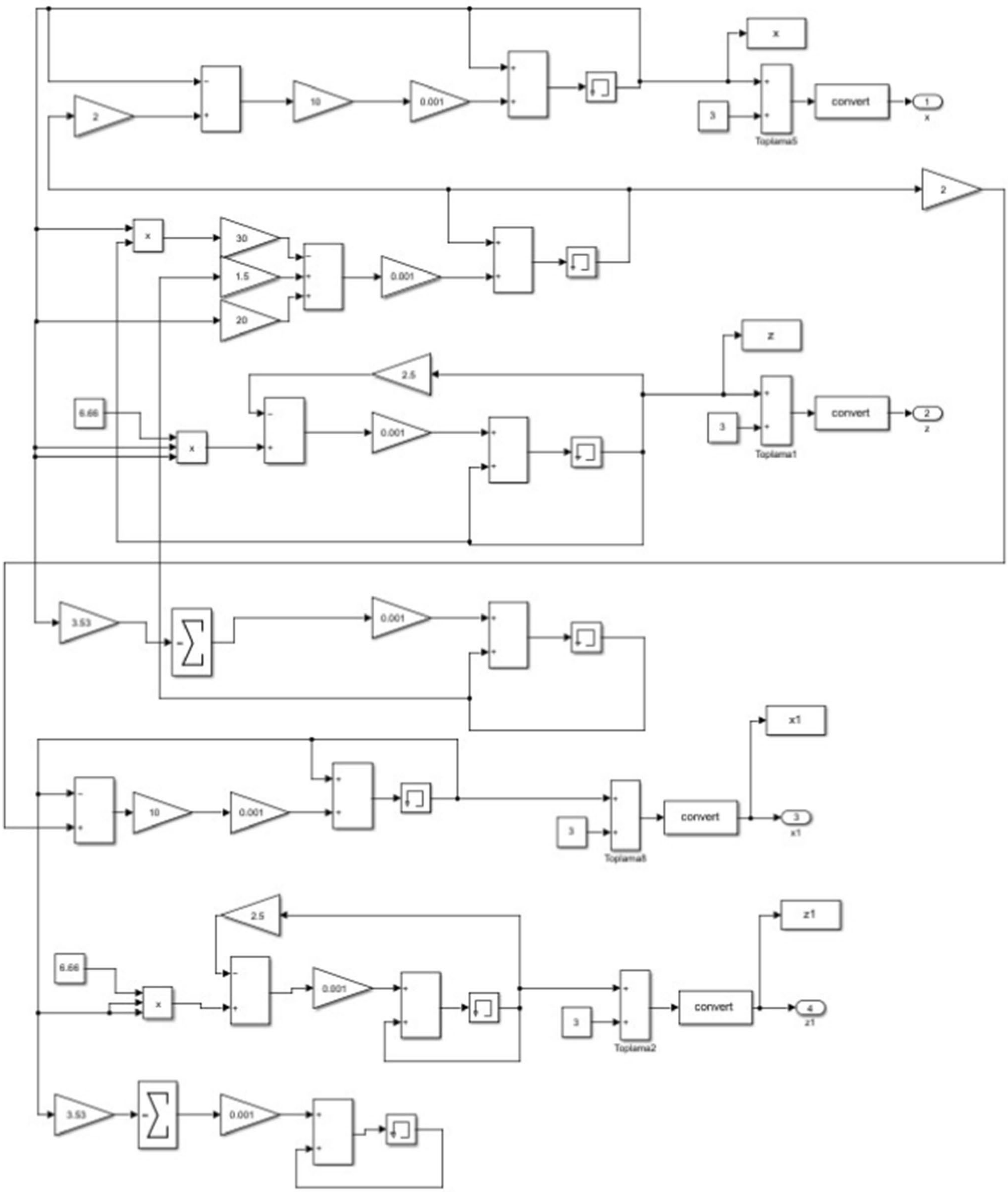


Fig. 11 Simulink model of the system for the HDL code

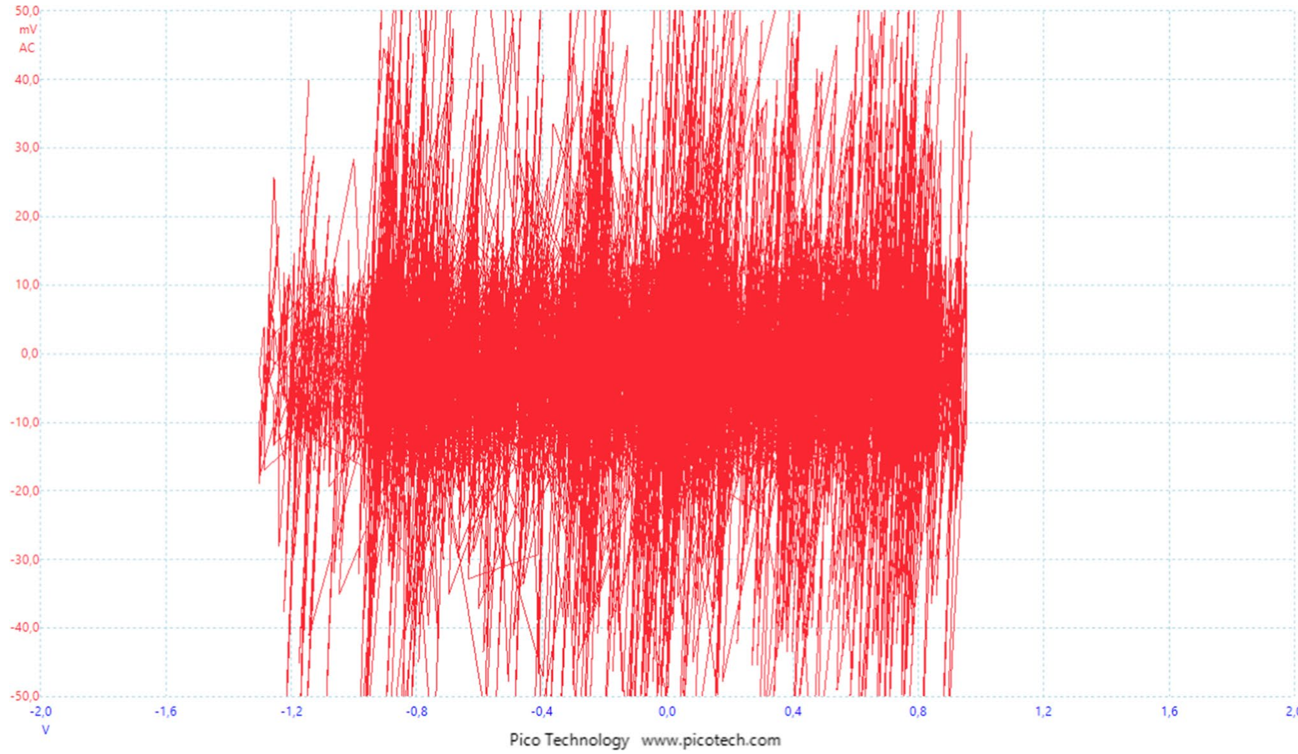


Fig. 12 FPGA implementation results for the not synchronized master and slave systems: graph of x_m vs x_s state variables (Color figure online)

is specified as 1 kHz. The synchronization curve and related time series are shown in Fig. 10. These results are obviously revealed the positive effect of filtering when compared to the Fig. 8(a).

6 FPGA implementation of the synchronized Wang-Liu system

Field Programmable Gate Arrays (FPGA) are another efficient and practical way of implementing nonlinear dynamical systems. This section of study includes the experimental realization of synchronized and not synchronized master-slave systems on FPGA. System (4) is first constructed on Simulink and HDL code is obtained by using Xilinx System Generator. Simulink model of the system is demonstrated in Fig 11. Then the HDL code is compiled by using Quartus 2 which is the computer interface of Altera DE2-115 FPGA board and master-slave systems are implemented individually. The state of x_m and x_s variables is demonstrated in Fig. 12. This graph has a similar shape with Figs. 3(f) and 6(c) which indicates that the master and slave systems are out of synchronization.

System (4) is then performed in the same way and synchronization results on FPGA are provided. The linear line given in Fig. 13(a) represents the synchronization between the x variables of master and slave systems. Furthermore, time series of x_m and x_s is observed and same dynamics is obtained for the variables as seen in Fig. 13(b). The difference between x_m and x_s signals is also given to observe the synchronization error. On the other hand, synchronization is confirmed by the phase portraits. Fig. 14(a), (b) exhibit that the x_m vs z_m and x_s vs z_s graphs have similar behaviors on phase-space, nevertheless they are different before synchronization.

FPGA implementation of synchronized system has also some noise as in the FPAA implementation. However, since the FPGA platform is in a digital world, it does not have the opportunity to design filters in the computer interface. Therefore, a passive low-pass filter is designed by using R-C components and implemented to the output of state variables to eliminate the noisy part. Resistor and capacitor values for the filter are determined as 390 Ohm and 10 nF, respectively. The synchronization curve and time series obtained after filtering are shown in Fig. 15. Comparison of Figs. 13 and 15 proves the advantage of the filtering and demonstrates that more accurate results are achieved.

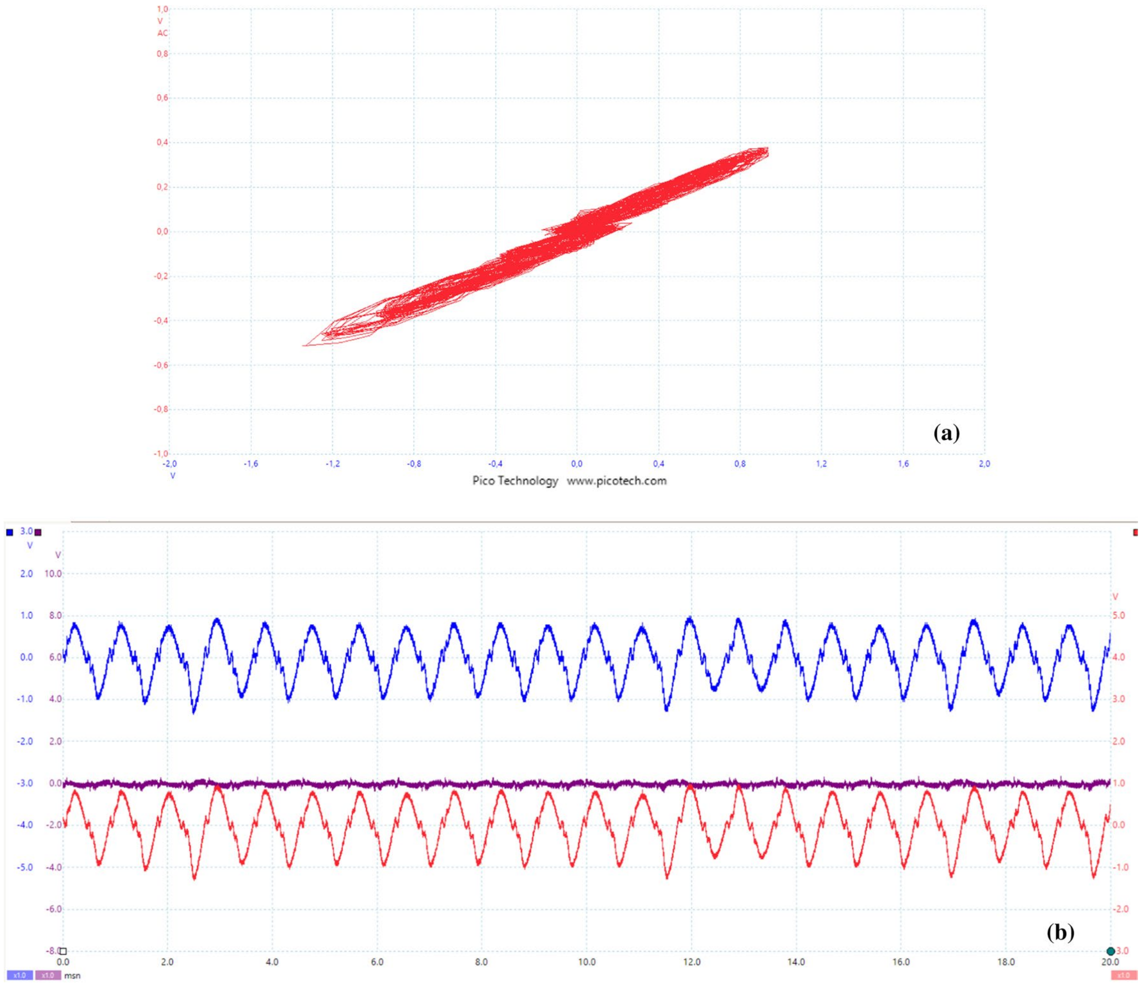


Fig. 13 FPGA implementation results for the synchronized master and slave systems **a** graph of x_m vs x_s state variables; **b** time series of x_m and x_s state variables: blue represents x_m, red represents x_s and purple is the x_m-x_s(difference signal) (Color figure online)

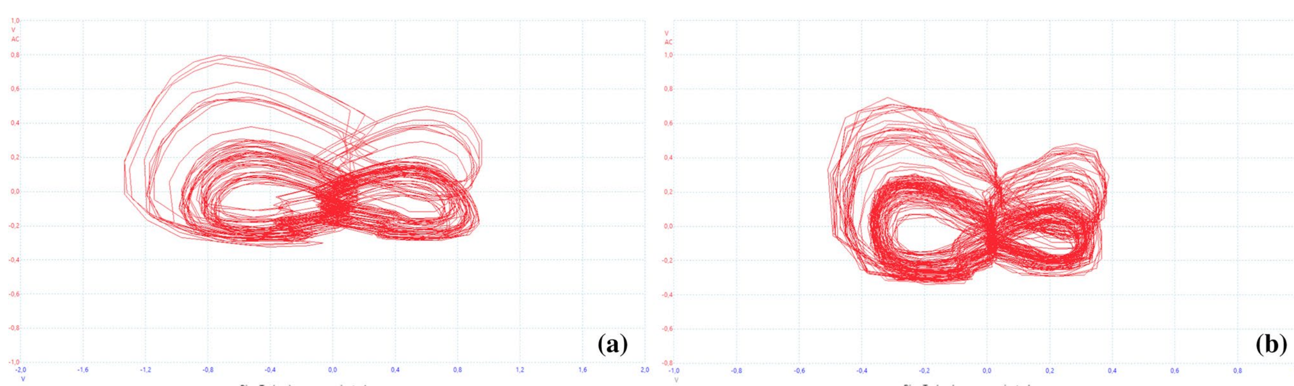


Fig. 14 FPGA implementation results for the synchronized master and slave systems **a** phase portraits of x–z plane in master system; **b** phase portraits of x–z plane in slave system (Color figure online)

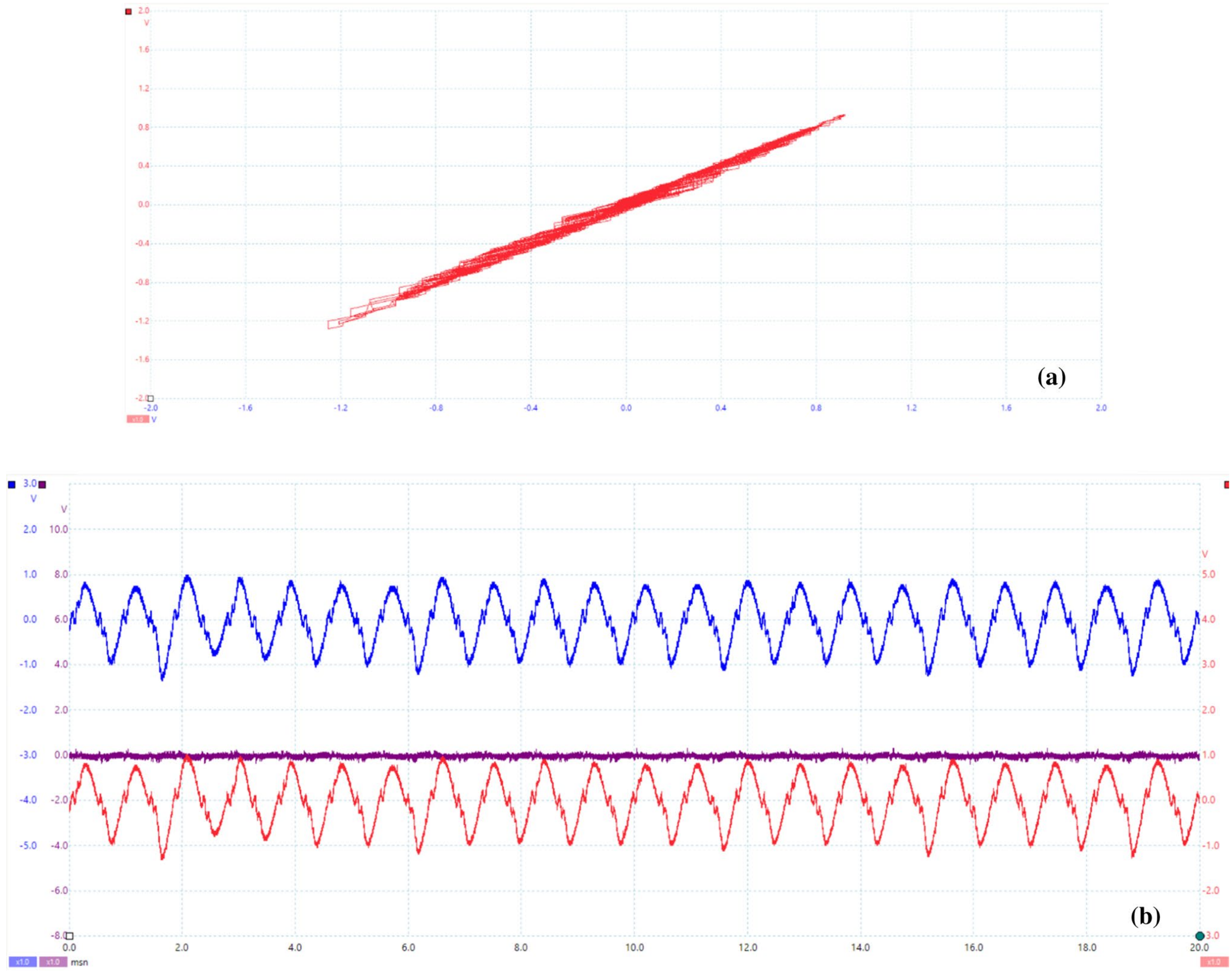


Fig. 15 FPGA implementation results for the synchronized master and slave systems after filtering: **a** synchronization curve (x_m vs x_s) **b** time series of x_m and x_s state variables: blue represents x_m, red represents x_s and purple is the $x_m - x_s$ (difference signal) (Color figure online)

7 Discussion and conclusions

The motivation point of this study is to implement the hyperchaos synchronization on the programmable analog and digital platforms which is the FPAA and FPGA, respectively. As a consequence of the experimental realization, it is observed that both of two integrated circuit devices have their own advantages. FPAA has a user-friendly interface and a dynamical system can be modelled with configurable blocks. However, FPGA interface offers coding as well as modeling with blocks. Additionally, FPGAs produce digital outputs while FPAA has the analog one. Hence, Digital to Analog Converter (DAC) is required for the FPGA implementation of an analog system. On the other hand, even if FPGA demonstrates higher performance in general, FPAA is a more appropriate platform for the realization of analog circuits. Therefore, more accurate and reliable results can be obtained with FPAA implementation when the experimental results are considered.

Synchronization performance of two different implementation method also needs to be discussed. Table 1 demonstrates the measurement results of difference signal obtained by subtracting x_m and x_s time series values. The difference

Table 1 A quantitative comparison between difference signals obtained from FPAA and FPGA implementations

	FPAA results		FPGA results	
	Not filtered	Filtered	Not filtered	Filtered
Max. of difference signal	1,382 V	51,81 mV	797,2 mV	755,1 mV
Min. of difference signal	1,311 V	43,89 mV	715,7 mV	638,7 mV
Avg. of difference signal	1,354 V	45,04 mV	747,3 mV	700,4 mV

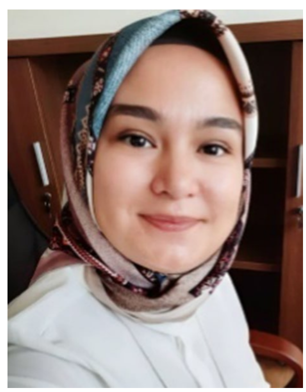
between two signals should be zero in order to perform a perfect synchronization ideally. However, the best performance is observed in the filtered FPAA implementation with the average peak-to-peak voltage level of 45,04 mV. This means that including a low-pass filter to the FPAA implementation enhance the synchronization performance compared to the not filtered results. On the other hand, filtering does not have a significant effect on FPGA performance and in both cases of FPGA implementation, voltage levels of difference signal are higher than the results of FPAA.

In this paper, synchronization of Wang-Liu system based on the complete replacement method is introduced. Lyapunov exponents and the bifurcation diagrams are studied first as dynamical analysis. Then two systems with different dynamics are selected for the synchronization. Structure of the master system is specified as hyperchaotic and the slave system is quasi-periodic. For the synchronization of master and slave systems complete replacement method is used. At this point, master system forced the slave system to demonstrate hyperchaotic behavior. Simulation results proved the perfect synchronization of two systems. Moreover, experimental realization of the synchronized systems is provided using FPAA and FPGA boards. Both implementation results confirm the results obtained from simulations.

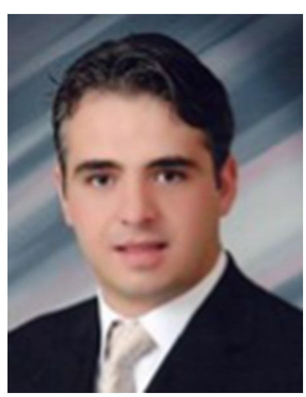
References

1. Lorenz, E. N. (1963). Deterministic nonperiodic flow. *Journal of Atmospheric Science, 20*(2), 130–141. https://doi.org/10.1175/1520-0469(1963)020%3c0130:dnf%3e2.0.co;2
2. Rossler, O. E. (1979). An equation for hyperchaos. *Physics Letters A*. https://doi.org/10.1016/0375-9601(79)90150-6
3. Chua, L. O., & Kobayashi, K. (1986). Hyperchaos: Laboratory experiment and numerical confirmation. *IEEE Transactions on Circuits and Systems, 33*(11), 1143–1147. https://doi.org/10.1109/TCS.1986.1085862
4. Wang, X., & Wang, M. (2008). A hyperchaos generated from Lorenz system. *Physica A: Statistical Mechanics and its Applications, 387*(14), 3751–3758. https://doi.org/10.1016/j.physa.2008.02.020
5. Wang, F. Q., & Liu, C. X. (2006). Hyperchaos evolved from the Liu chaotic system. *Chinese Physics, 15*(5), 963–968. https://doi.org/10.1088/1009-1963/15/5/016
6. Gao, T., Chen, Z., Yuan, Z., & Chen, G. (2006). A hyperchaos generated from Chen's system. *International Journal of Modern Physics C, 17*(4), 471–478. https://doi.org/10.1142/S0129183106008625
7. Vaidyanathan, S., Dolvis, L. G., Jacques, K., Lien, C. H., & Sambas, A. (2019). 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. https://doi.org/10.1504/IJMIC.2019.101959
8. Yujun, N., Xingyuan, W., Mingjun, W., & Huaguang, Z. (2010). A new hyperchaotic system and its circuit implementation. *Communications in Nonlinear Science and Numerical Simulation, 15*(11), 3518–3524. https://doi.org/10.1016/j.cnsns.2009.12.005
9. Brahim, A. H., Pacha, A. A., & Said, N. H. (2021). A new image encryption scheme based on a hyperchaotic system & multi specific S-boxes. *The International Journal of Information Security*. https://doi.org/10.1080/19393555.2021.1943572
10. Chen, X., et al. (2020). Pseudorandom number generator based on three kinds of four-wing memristive hyperchaotic system and its application in image encryption. *Complexity*. https://doi.org/10.1155/2020/8274685
11. Setoudeh, F., & Sedigh, A. K. (2021). Nonlinear analysis and minimum L2-norm control in memcapacitor-based hyperchaotic system via online particle swarm optimization. *Chaos, Solitons & Fractals, 151*, 111214. https://doi.org/10.1016/J.CHAOS.2021.111214
12. Xiu, C., Zhou, R., Zhao, S., & Xu, G. (2021). Memristive hyperchaos secure communication based on sliding mode control. *Nonlinear Dynamics*. https://doi.org/10.1007/s11071-021-06302-9
13. Chen, Y., Zhang, H., & Kong, X. (2021). A new fractional-order hyperchaotic system and its adaptive tracking control. *Discrete Dynamics in Nature and Society*. https://doi.org/10.1155/2021/6625765
14. Hui, Y., Liu, H., & Fang, P. (2021). A DNA image encryption based on a new hyperchaotic system. *Multimedia Tools and Applications*. https://doi.org/10.1007/s11042-021-10526-7
15. Wang, X., & Zhao, M. (2021). An image encryption algorithm based on hyperchaotic system and DNA coding. *Optics & Laser Technology, 143*, 107316. https://doi.org/10.1016/J.OPTLASTEC.2021.107316
16. Zhou, Y., Bi, M., Zhuo, X., Lv, Y., Yang, X., & Hu, W. (2021). Physical layer dynamic key encryption in OFDM-PON system based on cellular neural network. *IEEE Photonics Journal*. https://doi.org/10.1109/JPHOT.2021.3059369
17. Luo, J., Qu, S., Chen, Y., Chen, X., & Xiong, Z. (2021). Synchronization, circuit and secure communication implementation of a memristor-based hyperchaotic system using single input controller. *Chinese Journal of Physics, 71*, 403–417. https://doi.org/10.1016/j.cjph.2021.03.009
18. Bian, Y., & Yu, W. (2021). A secure communication method based on 6-D hyperchaos and circuit implementation. *Telecommunication Systems, 77*, 73. https://doi.org/10.1007/s11235-021-00790-1
19. Yu, W., et al. (2019). Design of a new seven-dimensional hyperchaotic circuit and its application in secure communication. *IEEE Access, 7*, 125586–125608. https://doi.org/10.1109/ACCESS.2019.2935751
20. Benkouider, K., Bouden, T., Yalcin, M. E., & Vaidyanathan, S. (2020). A new family of 5D, 6D, 7D and 8D hyperchaotic systems from the 4D hyperchaotic Vaidyanathan system, the dynamic analysis of the 8D hyperchaotic system with six positive Lyapunov exponents and an application to secure communication design. *International Journal of Modelling, Identification and Control, 35*(3), 241–257. https://doi.org/10.1504/IJMIC.2020.114191
21. Singh, S., Han, S., & Lee, S. M. (2021). Adaptive single input sliding mode control for hybrid-synchronization of uncertain hyperchaotic Lu systems. *Journal of the Franklin Institute*. https://doi.org/10.1016/J.JFRANKLIN.2021.07.037
22. Vaidyanathan, S., & Rasappan, S. (2011). Global chaos synchronization of hyperchaotic Bao and Xu systems by active nonlinear control. *Communications in Computer and Information Science, 198*, 10–17. https://doi.org/10.1007/978-3-642-22555-0_2
23. Pecora, L. M., & Carroll, T. L. (1990). Synchronization in Chaotic systems.
24. Lai, Q., Wan, Z., Kuate, P. D. K., & Fotsin, H. (2021). Dynamical analysis, circuit implementation and synchronization of a new memristive hyperchaotic system with coexisting attractors. *Modern Physics Letters B*. https://doi.org/10.1142/S0217984921501876

25. Sajjadi, S. S., Baleanu, D., Jajarmi, A., & Pirouz, H. M. (2020). A new adaptive synchronization and hyperchaos control of a biological snap oscillator. *Chaos Solitons and Fractals*. https://doi.org/10.1016/j.chaos.2020.109919
26. Vaidyanathan, S., Pham, V. T., Volos, C., & Sambas, A. (2018). A novel 4-D hyperchaotic rikitake dynamo system with hidden attractor, its properties, synchronization and circuit design. In *Studies in systems, decision and control* (Vol. 133, pp. 345–364). Berlin: Springer.
27. Liao, T. L., Wan, P. Y., & Yan, J. J. (2022). Design and synchronization of chaos-based true random number generators and its FPGA implementation. *IEEE Access*. https://doi.org/10.1109/ACCESS.2022.3142536
28. Wang, P., Wen, G., Yu, X., Yu, W., & Huang, T. (2019). Synchronization of multi-layer networks: From node-to-node synchronization to complete synchronization. *IEEE Transactions on Circuits and Systems I: Regular Papers, 66*(3), 1141–1152. https://doi.org/10.1109/TCSI.2018.2877414
29. Al-Obeidi, A. S., & Al-Azzawi, S. F. (2019). Projective synchronization for a cass of 6-D hyperchaotic lorenz system. *Indonesian Journal of Electrical Engineering and Computer Science, 16*(2), 692–700. https://doi.org/10.11591/IJEECS.V16.I2.PP692-700
30. Wu, X., Fu, Z., & Kurths, J. (2015). A secure communication scheme based generalized function projective synchronization of a new 5D hyperchaotic system. *Physica Scripta*. https://doi.org/10.1088/0031-8949/90/4/045210
31. Gularte, K. H. M., Alves, L. M., Vargas, J. A. R., Alfaro, S. C. A., De Carvalho, G. C., & Romero, J. F. A. (2021). Secure communication based on hyperchaotic underactuated projective synchronization. *IEEE Access, 9*, 166117–166128. https://doi.org/10.1109/ACCESS.2021.3134829
32. Zhou, C., Yang, C., Xu, D., & Chen, C. Y. (2019). Dynamic analysis and finite-time synchronization of a new hyperchaotic system with coexisting attractors. *IEEE Access, 7*, 52896–52902. https://doi.org/10.1109/ACCESS.2019.2911486
33. Sangpet, T., & Kuntanapreeda, S. (2020). Finite-time synchronization of hyperchaotic systems based on feedback passivation. *Chaos, Solitons & Fractals, 132*, 109605. https://doi.org/10.1016/J.CHAOS.2020.109605
34. Al-Obeidi, A. S., Al-Azzawi, S. F., Hamad, A. A., Thivagar, M. L., Meraf, Z., & Ahmad, S. (2021). A novel of new 7D Hyperchaotic system with self-excited attractors and its hybrid synchronization. *Computational Intelligence and Neuroscience*. https://doi.org/10.1155/2021/3081345
35. Sarasu, P., & Sundarapandian, V. (2011). The generalized projective synchronization of hyperchaotic lorenz and hyperchaotic Qi systems via active control. *International Journal of Soft Computing, 6*(5), 216–223. https://doi.org/10.3923/IJSCOMP.2011.216.223
36. Pecora, L. M., Carroll, T. L., Johnson, G. A., Mar, D. J., & Heagy, J. F. (1997). Fundamentals of synchronization in chaotic systems, concepts, and applications. *Chaos, 7*(4), 520–543. https://doi.org/10.1063/1.166278
37. Wang, F., Wang, R., Iu, H. H. C., Liu, C., & Fernando, T. (2019). A novel multi-shape chaotic attractor and its FPGA implementation. *IEEE Transactions on Circuits and Systems II: Express Briefs, 66*(12), 2062–2066. https://doi.org/10.1109/TCSII.2019.2907709
38. Kiliç, R., Alçi, M., & Günay, E. (2004). A SC-CNN-based chaotic masking system with feedback. *International Journal of Bifurcation and Chaos in Applied Sciences and Engineering, 14*(1), 245–256. https://doi.org/10.1142/S0218127404009120
39. Günay, E., & Altun, K. (2018). Lorenz-like system design using cellular neural networks. *Turkish Journal of Electrical Engineering and Computer Science, 26*(4), 1812–1819. https://doi.org/10.3906/elk-1706-309
40. Salih, T. A. (2021). Design and implementation of a low power consumption of ASK, FSK PSK, and QSK Modulators based on FPAA technology. *International Journal on Advanced Science, Engineering and Information Technology, 11*(4), 1288. https://doi.org/10.18517/ijaseit.11.4.11299
41. Diab, M. S., & Mahmoud, S. A. (2020). Field programmable analog arrays for implementation of generalized nth-order operational transconductance amplifier-C elliptic filters. *ETRI Journal, 42*(4), 534–548. https://doi.org/10.4218/ETRIJ.2020-0104
42. Vaidyanathan, S., et al. (2021). A 5-D multi-stable hyperchaotic two-disk dynamo system with no equilibrium point: Circuit design, FPGA realization and applications to TRNGs and image encryption. *IEEE Access, 9*, 81352–81369. https://doi.org/10.1109/ACCESS.2021.3085483
43. Yu, F., et al. (2020). Multistability analysis, coexisting multiple attractors, and FPGA implementation of Yu-Wang four-wing chaotic system. *Mathematical Problems in Engineering*. https://doi.org/10.1155/2020/7530976
44. Yılmaz, G., & Günay, E. (2021). FPAA Implementation of Wang-Liu system. In *2021 13th international conference on electrical and electronics engineering (ELECO)* (pp. 167–171). IEEE. https://doi.org/10.23919/ELECO54474.2021.9677754.



Gülnur Yılmaz is currently working as Research Assistant at Erciyes University. She obtained her B.S degree in Electrical & Electronics Engineering from Abdullah Gül University in 2018 and M.S degree in the same department of Erciyes University in 2022. She is pursuing her Ph.D. degree at Erciyes University. Her main areas of research are chaotic and hyperchaotic systems, electronic circuits and embedded systems.



Kenan Altun received the B.S. and M.S. degrees in Electrical and Electronics Engineering from the Cumhuriyet University, Turkey, in 2003 and 2006, respectively. He received a Ph.D. degree in Electrical - Electronics Engineering from Erciyes University, Kayseri, Turkey, in 2019. He is currently Assistant Professor of Electronics and Automation at the University of Sivas Cumhuriyet. His current research interests include embedded systems, nonlinear circuits, chaotic systems and electronic circuits.

Enis Günay received his B.S and Ph.D. degrees in Electrical & Electronics Engineering from Erciyes University in 1999 and 2006, respectively. He also got his M.S. degree from Niğde Ömerhalis Demir University in 2001. He is presently working as a Professor in the Department of Electrical & Electronics Engineering in Erciyes University. The area of interest of the author is circuits-systems theory, electronics and nonlinear dynamical systems.