Parameter	Heun-based	RK4-based					
Maximum frequency (MHz)	359.71	359.71					
No. of DSP	2	4					
Number of 4 input LUTs	2912	2637					
Number of bonded IOBs	32	32					
Number of Slice Flip Flops	4977	4692					
Total On-chip Power(W)	0.178	0.179					

 Table 1

 Final report of the resources consumption

## **V CONCLUSION**

The Heun and RK4 algorithm basedPBS Chaotic generator have been synthesized using the Nexus 4 DDR XC7A100TCSG-1 (Artix7) and Basys3 (Artix7) from the Xilinx Vivado v.2017.3 design suite. RK4 based chaotic generator gives optimize result with the use of 2637 LUT's and 4692 registers with set clock period 2.78 ns which corresponds to maximum frequency achieved 359.71 MHz.The attracter of the system is generated by the data setare given in fig. 10(a-c) which are similar to PBSCS designed on analog platform.

## REFERENCES

 Ismail K., A. Tuaran O, IhsanPehlivan, "Implementation of FPGA-based real time novel chaotic oscillator", Nonlinear dynamics (2014); pp. 49-59.

- [2] Murat Tunaa, Can BülentFidan, "Electronic circuit design, implementation and FPGAbasedrealization of a new 3D chaotic system with singleequilibrium point", Optic ElsevierOptik(2016) pp. 11786–11799.
- [3] S. Banerjee, J. Kurths, "Chaos and cryptography: a new dimension in secure communications", Eur. Phys. J. Spec. Top., (2014) pp. 1441–1445.
- [4] I. Koyuncu, A.T. Ozcerit, I. Pehlivan, "An analog circuit design and FPGA-based implementation of the Burke-Shaw chaotic system", Optoelectron. Adv. Mater. Rapid Commun. (2013) pp. 635–638.
- [5] L. Merah, A. Ali-pacha, N.H. Said, "A pseudo random number generator based on the chaotic system of Chua's circuit and its real time", FPGA Implementation (2013) pp. 2719–2734.
- [6] S. C, ic, ek, A. Ferikog'lu, I'. Pehlivan, "A new 3D chaotic system: dynamical analysis, electronic circuit design, active control synchronization and chaotic masking communication application", Optik – Int. J. Light Electron Opt. (2016) pp. 4024–4030.
- [7] M. Tuna, I. Koyuncu, C.B. Fidan, I. Pehlivan, "Real time implementation of a novel chaotic generator on FPGA", in: 2015 23rd Signal Processing andCommunications Applications Conference (SIU), IEEE, 2015, pp. 698–701.
- [8] Alpana Pandey, R. K. Baghel, R.P. Singh, "Analysis and Circuit Realization of a NewAutonomous Chaotic System", International Journal of Electronics and Communication Engineering.ISSN 0974-2166 Volume 5, Number 4 (2012), pp. 487-495.

## **FPGA Implementation of Chaotic Generator Using Numerical Algorithms**

Subodh Kumar Pandey<sup>1</sup>, Dr. Sanjeev Kumar Gupta<sup>2</sup>

<sup>1,2</sup>Dept. of ECE, Rabindranath Tagore University, Raisen (M.P.) India.

#### ABSTRACT

Now a day's chaotic systems have an important role in secure communication and cryptography. As FPGA implementation have certain advantages over analog one, different chaotic system like chaotic oscillator, True random number generators and chaotic systems used in image processing, optical circuits for secure communications were successfully realized in FPGA. This paper presents methodology of FPGA implementation of any chaotic system using different numerical algorithm. In study the Numerical solution of Differential equations given in Pandey-Baghel-Singh system were obtained and coded in Verilog and tested with XilinxVivado 17.3 design suites in Artix-7 Nexus 4 DDR and Basys3. Performance of the FPGA based chaotic generator using Heun and RK4 algorithmsare analyzed using 10<sup>6</sup> data sets with the maximum operating frequency achieved up to 359.71MHz.

Key Words - Chaotic Generators, Heun, RK4 algorithm, FPGA

### **I INTRODUCTION**

Chaos generator is a fundamental block of any chaos based system. Basically chaos based system are used in secure communication and cryptography.Recently implementation of FPGA based real time chaotic systems were presented. Due to parallel processing capabilities the processing speed of FPGA is much higher. Analog based chaotic generatorsare sensitive to initial conditions and acquires a large chip area hence it may be interesting to see the performance of FPGA based chaotic generators using FPGA can be implemented as FPGA implementation is more flexible architecture and have low cost test cycle and found more useful in chaos based engineering applications [1-7].

In II section of the paper presented the Pandey-Baghel-Singh Chaos System (PBSCS) is described along with their x,y and z signals and their attractors [8]. In the III section the mathematical models of PBSCS is numerically obtained using Heun and RK4 algorithms and FPGA models of PBSCS is introduced. In sectionIV simulation results of different numerical algorithm based design has been presented and analyzed. In section V conclusion is given.

## II INTRODUCTION TO PANDEY-BAGHEL-SINGH CHAOS SYSTEM

Pandey-Baghel-Singh Chaos System (PBSCS) is defined by the set of differential equation (1).

$$\dot{x} = y$$
  
$$\dot{y} = z \qquad (1)$$
  
$$\dot{z} = -ax - by - cz - x^{2}$$

In the system two equilibrium points as (0, 0, 0) and (-1, 0, 0) were shown for the constants a = 1, b = 1.1, and c = 0.4. The equilibrium point (0, 0, 0) have the Eigen values

-0.745, 0.162+j1.147and 0.162-j1.147. For the equilibrium point (-1, 0, 0) the Eigen values shown are 0.589, -0.504+j1.20, and -0.504-j1.20. The initial condition for the system is taken x = 0.1, y = 0 and z = 0. The time domain representation of x, y and z waveform are given in Fig.1 and attractors generated are given in Fig. 2 (a-c).



Fig 1: Time domain representation of x, y and z signals of PBSCS.



## III NUMERICAL MODELS OF PBSCSAND THEIR FPGAIMPLEMENTATION USING DIFFERENT NUMERICAL ALGORITHMS

For FPGA implementation of the system the numerical model of PBSCSis obtained using Heun and RK4algorithm and coded in Verilog.

## (a) Numerical model using Heun algorithm

For the numerical model using Heun algorithm initial value of x (n), y (n) and z (n) are taken as x (t<sub>0</sub>) = x (n) = 0.1, y (t<sub>0</sub>) = y (n) = 0 and z(t<sub>0</sub>) = z(n) = 0. The Heun algorithm have two successive stages. In the first stage x(n<sup>0</sup> + 1) is calculated and x(n + 1) the value after steps h is calculated using previous values x(n<sup>0</sup> + 1) andx(n). The mathematical model of PBS chaotic system is described by the following Equation (2).

 $x(n^{0} + 1) = x(n) + h.y(n)$ 

$$x(n + 1) = x(n) + h\{y(n) + x(n^{0} + 1)\}/2(2)$$

$$y(n^{0} + 1) = y(n) + h.z(n)$$

$$y(n + 1) = y(n) + h\{z(n) + y(n^0 + 1)\}/2$$

$$z(n^{0} + 1) = z(n) + h\{-a.x(n) - b.y(n) - c.z(n) - x(n)^{2}\}$$

$$z(n + 1) = z(n) + h. [\{-a.x(n) - b.y(n) - c.z(n) - x(n)^2\} + z(n^0 + 1)]/2$$

#### (b) Numerical model using RK4 algorithm

To construct the mathematical model of the PBSCS using RK4 algorithm, the system equation are represented as a function of f, gand $\delta$  as equation (3)  $\dot{x} = f(t, x, y, z) = y$  $\dot{y} = g(t, x, y, z) = z$  (3)  $\dot{z} = \delta(t, x, y, z) = -ax - by - cz - x^2$ With respect to above equation the mathematical model

of the system using RK4 algorithm is given in equation (4). The parameter K,  $\lambda$  and  $\xi$  in equation (5) defined as the coefficients of the first, second and third equations given in equation (3) and are placed in equation (4) to calculate x(k + 1), y(k + 1) and z(k + 1) which are the values of the system afterh steps.

The valuesx(k + 1), y(k + 1) and z(k + 1) are the output of the system after each interval which are used as initial conditions of the algorithm to calculate the values for the next cycle.

$$\begin{aligned} x(n+1) &= x(n) + \frac{1}{6}h[k_1(n) + 2k_2(n) + 2k_3(n) + k_4(n)] \\ y(n+1) &= y(n) + \frac{1}{6}h[\lambda_1(n) + 2\lambda_2(n) + 2\lambda_3(n) + \lambda_4(n)] \\ z(n+1) &= z(n) + \frac{1}{6}h[\xi_1(n) + 2\xi_2(n) + 2\xi_3(n) + \xi_4(n)] (4) \\ k_1 &= f[x(n), y(n), z(n)] \end{aligned}$$

$$\begin{split} \lambda_{1} &= g[x(n), y(n), z(n)] \\ \xi_{1} &= \delta[x(n), y(n), z(n)] \\ k_{2} &= f[x(n) + \frac{1}{2}hk_{1}. y(n) + \frac{1}{2}h\lambda_{1}. z(n) + \frac{1}{2}h\xi_{1}] \\ \lambda_{2} &= g[x(n) + \frac{1}{2}hk_{1}. y(n) + \frac{1}{2}h\lambda_{1}. z(n) + \frac{1}{2}h\xi_{1}] \\ \xi_{2} &= \delta[x(n) + \frac{1}{2}hk_{1}. y(n) + \frac{1}{2}h\lambda_{1}. z(n) + \frac{1}{2}h\xi_{1}] \\ k_{3} &= f[x(n) + \frac{1}{2}hk_{2}. y(n) + \frac{1}{2}h\lambda_{2}. z(n) + \frac{1}{2}h\xi_{2}] \\ \lambda_{3} &= g[x(n) + \frac{1}{2}hk_{2}. y(n) + \frac{1}{2}h\lambda_{2}. z(n) + \frac{1}{2}h\xi_{2}] \\ \xi_{3} &= \delta[x(n) + \frac{1}{2}hk_{2}. y(n) + \frac{1}{2}h\lambda_{2}. z(n) + \frac{1}{2}h\xi_{2}] \\ k_{4} &= f[x(n) + hk_{3}y(n) + h\lambda_{3}z(n) + h\xi_{3}] \\ \lambda_{4} &= g[x(n) + hk_{3}y(n) + h\lambda_{3}z(n) + h\xi_{3}] \\ \xi_{4} &= \delta[x(n) + hk_{3}y(n) + h\lambda_{3}z(n) + h\xi_{3}] \end{split}$$

# (c) FPGA Implementation of Autonomous Chaotic Generator

The PBSCS which have been modeled usingHeun andRK4 algorithm are implemented with 32- bit IEEE 754-1985 standard on FPGA. Mathematicalmodeling is done in VerilogusingVivado design suite. The Top-level diagramwhich is same for both models using Heun and RK4algorithm have been shown in Fig. 3. For the synchronization purpose one bit start, reset and clock signal is used. A 32-bit input has been used and initial conditions are set in the beginning phase. The 32-bit signal are used as input parameter. There is three 32-bit output signals (Xn\_out), (Yn\_out) and (Zn\_out)and ready signal is taken as one bit control signals for the proposed chaotic generator.



## Fig.3 Top level diagram of PBS Chaotic System based on FPGA

The second level block diagram of the chaotic generator is presented in Fig. 4 It have one multiplexer and a chaotic generator unit which is FPGA based. The multiplexer is used to provide initial condition signals. For successive operation it is provided by the output signals. When enable is at logic high, the output generates chaotic signal.



Fig.4 second level diagram of PBSCS based on FPGA

The third level block diagram of the Heun based chaotic generator is given in Fig.5. The proposed generator consist of multiplexer, function f<sup>0</sup>, multiplier, adder, f, Divider and filter stages. The PBSCS equation are calculated by  $f^0$  stage with the help of MUX unit which provides control signal. After multiplication with h the output is added with the previous generated signal x(n), y(n) and z(n) by the generator unit. The output of this adder stage is applied to f stage which calculate the equation of PBSCS. The output of this stage and output of f<sup>0</sup> are adder-II stage. Further the output of the adder-II stage divided in the divider stage. In adder-III stage output of the chaotic generator from MUX stage and divider stage are added. The Heun based chaotic generator works in sequential order which generates the first value after 118 clock cycles.



Fig. 5 Third Level diagram of Heun based PBSCS Generator Unit

The thirdlevel block diagram of the RK4 based chaotic generator is given in Fig. 6. The proposed chaotic generator consist of multiplexer,  $K_s$  units,  $Y_s$  block and filter stage. $K_s$  units calculate  $k_s$ ,  $\lambda_s$  and  $\xi_s$  where s varies  $\frac{X_s - out[b]}{2}$  etween 1 to 4.

Y\_out[3]: The x(k + 1), y(k + 1) and z(k + 1) given in equation Z\_out[3:] are Calculatedat Y<sub>s</sub> block. The first value is generated after 142 clock pulses and a feedback system Ready is to be employed so that output is feedback to MUX after 142 clock pulses to generate next cycle. Filter unit stops undesired signal to reach output if generator does not generate any result.



Fig. 6 Third Level diagram of RK4 based PBSCS Generator Unit

## IV SIMULATION RESULTS OF PBS CHAOTIC GENERATOR

The numerically modelled (Heun and RK4) PBS Chaotic generator have been synthesized on Nexus-4 DDR XC7A100TCSG-1 (Artix7) and Basys-3 (Artix7) from the Xilinx Vivado v.2017.3 design suite. The simulation results of numerically modelled PBSCS and FPGA chip related Parameters and clock speed of the system is presented in the Fig. 7and Fig. 8. The summery of the FPGAchip speed and other statistics which are obtained for both the algorithm based system is given in table 1. Among the two numerically modelled system the RK4 based chaotic generator gives optimize result with the use of 2637 LUT's and 4692 registers with set clock period 2.78 ns which corresponds to maximum frequency achieved 359.71 MHz.The attracter of the system is generated by the data set are given in fig. 9 (ac) which are similar to PBSCS designed on analog platform

## Shrusandhan-AISECT University Journal Vol. VII/Issue XIV September 2018 p-ISSN : 2278-4187, e-ISSN: 2457-0656

		0.000000	us																	
Name	Valu	e 0 us	50	0 us	100	us	150 us	2	:00 us	2	50 us	300 us	5	350 us	400	us	450 us	500	us	550 us
> 1 RN_out[20:0	) 0000	X X	100000		8aa10	1489£3	X 029	887	008019	X 008	55fe	007682	00343	30 X 00e35	5a	001cle	003747	001c94	0011	5 001384
> 📲 RS[20:0]	20000	X X00000X	0020	180 160	( aba	070 050	040	040 0	30 030	030	030 (	020 020 0	020 0	20 020 0	020 02	20 020	010 01	010 011	) ( 010 ) ( O	010 010 0
1 rst	Z																			
Ck	Z	1000						_										_		
> ~ X_in[31:0]	00000	0000			=		=				(									
> ¥ 7 in[31:0]	00199	1999			=		-					00000000			==					
> 📢 h[31:0]	00199	9999			=		=					001999999								
> 💘 Xn[31:0]	10000	000X 0000	0000	000 000	( 000 )	000 000	3 ( 000 )	000 ) 0	00 / 000	( 000		) 000 X 000	000 0		000 01		( opa ) ( oo	a X 000 X 000	3 X 000 X 0	
> 📢 Yn[31:0]	30000	1000X 1000E	0000	000 000	000	000 000	3 ( 000 )	000 0	00 000	000	000 (0		000 0		000 00	000	00 00	000 000	3 \ 000 \ 0	00 000 0
> 🐕 Zn[31:0]	30000	D0000 X000	000	000 000	000	000 000	000	000 0	00 000	000	000	000 000	000 0		000 0	000	000 00	a <b>( 000 ) ( 00</b>	000	000 000 000
₩ RS_H	х									J										
₩ Ready_w	Х																			
H RN_Ready	0																			
IR RN_Ready_																				
W RN_Ready_	12 1																			
Sources N	etlist ×				2		P	roiect	Summa	rv	× Sc	hematic	×	Chaotic C	Oscilla	tor.v	×			206
T H						*		_   _				5,7	0	o Lu		0			o Deste	» – –
X M						*			U.	C	i in	- 193	Ψ	-0-   T			60 C	ens / I/	0 Pons	
R Top						1														
> 📄 Nets (3	40)																F	-p,	L	
> 🗁 Leaf Ce	ells (57)																E			
> I BRAM_	IP (blk_mem_	gen_0)								Г				L					<u> </u>	
_ \	<ul> <li>Opeillator (C</li> </ul>	baotic Oscilla	ator)				1 <u>1</u>		╞╒Ũ╴		E	-		╘╍╌╓	0	P			╺╴┼╤╸	
Source File Pr	operties			?	_ □	ı c x				Ľ							b			<b>---</b>
Chaotic Os	scillatory					a a	ы.										ų.			
						· · · ·												4_3		
General	roperties																			
Tcl Console	Messages	Log Rej	ports	Desig	an Run	s ×													?.	
Q	- ●    4   ≪		+	%																
me	Constraints	Status		10		WNS	TNS	WH	а тн	в т	PWS	Total Po	wer	Failed Ro	utes	LUT	FF	BRAMs	LIRAM	DSP
synth 1	constrs 1	synth desig	n Com	npletel						- 1		- otarr 0				3002	5049	28.00	0.0.01	2
✓ impl 1	constrs 1	route desig	n Com	pletel		0 826	0.0	0.01	5 00		0 0 0 0	0	178		0	2912	4977	28.00	0	2
*		.cato_acoly				0.020	0.0	. 0.01	0.0			0				2012	4011	20.00	Ŭ	-
1																				

Fig.7 Simulation result of Heun based PBSCS on Vivado 17.3

0.01			and the second second second	testimed at the file	1000 C 1000 C 100	and the second sec	Tana (2) at Charles Carl
r Di L			al_operation,v A	Junded 1 × 110	10115.V (2) × 0	ator (z) ×   minut	Topo (2) A Chaone_Osch
				Fe ef H	t tr +F F	- I I H 🔁	Q 📕 Q Q 💥
40 us.   160 us.   180 us.   200 us.   220 us.   240 us.	120 us.  140 us.	100 us	s. 180 us	us . 160 us	. 140 us	Value	Name
1eb X 0004da71 X 00045a68 X 0003f604 X 0003a3f8 X 00035f4b XD	0007096c X 000591eb X 0	00022309	X 00163a3a	X 00188a43	00002053	0000f232	> M Xn[31:0]
teb X 0004da71 X 00045a68 X 0003f604 X 0003a3f8 X 00035f4b XD	0007096c 000591eb 0	00022309	00163838	00188443	00028185	0000f232	> 🖬 Yn[31:0]
1a71 X 00045a68 X 0003f604 X 0003a3f8 X 00035f4b X 000324c9 X0	000591eb 0004da71	00070960	000553c9	00163a3a	00188443	0000edd4 3000	> 🖬 Zn[31:0]
	00000000					00000000	> 📲 X_in[31:0]
50	0000000					00000000	> 📲 Y_in[31:0]
99	00199999					00199999	> 📲 Z_in[31:0]
99	00199999					00199999	> 🎫 h[31:0]
						1	1 ready
فتنصبها متقاصي والأحدي والأدعي والأحي والقراب						0	1a rst
						1	le dk
1eb X 00044da71 X 00045a68 X 0003f604 X 0003a3f8 X 00035f4b X0	0007096c 000591eb	00000309	00163a3a	00188a43	00002053	00001232 0000	> 📢 X[31:0]
1eb X 0004da71 X 00045a68 X 00032604 X 0003a328 X 0003524b X0	0007096c 000591eb	X 000bb3c9	00163a3a	00188443	00028115	0000f232 0000	> 🙀 Y[31:0]
1m71 X 00045m68 X 0003f604 X 0003m3f8 X 00035f4b X 000324c9 D	000\$91eb 000\$da71	X 0007096c	000ър3с9	00163a3a	0018843	0000edd4 0010	> 📲 Z[31:0]
1eb X 0004da71 X 00045a68 X 00032604 X 0003a328 X 0003524b X0	X 0009096c X 000591eb X	X 000bb3c9	00163a3a	00188843	00002053	00001232 0000	> 🖼 X_r[31:0]
1eb X 00044da71 X 00045a68 X 0003f604 X 0003a3f8 X 00035f4b X0	X 0007096c X 000591eb X	0000003c9	00163a3a	00188843	00028115	0000f232 0000	> 📲 Y_r[31:0]
1a71 X 00045a68 X 00032604 X 0003a328 X 0003524b X 000324c9 XD	( 000\$91eb ) 000\$da71 )	0007096c	000553c9	00163a3a	00188843	0000edd4 0010	> 🛀 Z_r[31:0]
						01800000	> M fixed_i1[31:0]
						0000fee6	> M fixed_i2[31:0]
						mme6	> M fixed_o[31:0]
						2	> 📲 mode[1:0]
					vvvvvv	4 1 2	> M ESM(2:01
1ab         000(44.71)         00(45.669)         00036604         00133763         00003564b           1ab         000(45.71)         00545.660         00032604         00333263         0003564b           1ab         000(45.660)         000345.660         00032634         0003564b         0003564b           1ab         000(45.660)         00053636         00053636         0003564b         0003564b           1ab         000(45.71)         000456.660         000536204         00033270         0003564b           1ab         000(45.71)         000456.660         000326204         00033270         0003564b           1ab         000(45.71)         000456.660         00032604         00033270         0003564b           1ab         000(45.72)         000456.660         00033300         0003264b         000326420           1ab1         000(45.660)         00033300         0003264b         000326420         0003264b           1ab2         000(45.660)         00033300         0003264b         000326400         000326400           1ab2         000(45.660)         00033300         0003264b         000326200         000326400           1ab2         000(45.660)         00033360         00326400 <t< th=""><th>0007094C 000591ab 000591ab 00004471 000591ab 00004471 0007094C 000591ab 0007094C 000591ab 0007094C 000591ab</th><th>000003259 00003529 0007096c 00003529 00003529 00003529 00005096c</th><th>0016 9434 0016 9434 0006 945 0016 9434 0016 9434 0016 9434 0006 9359</th><th>00188443 00165433 00165433 00165433 0018843 0018843 00165433</th><th>000020153 00020145 0010843 000020153 00020115 0010843</th><th>1 00001232 0000 00001232 0000 00001232 00001232 00001 00001232 00001 00001232 0000 0000</th><th>Te dk &gt; Mt (131.0) &gt; Mt (13</th></t<>	0007094C 000591ab 000591ab 00004471 000591ab 00004471 0007094C 000591ab 0007094C 000591ab 0007094C 000591ab	000003259 00003529 0007096c 00003529 00003529 00003529 00005096c	0016 9434 0016 9434 0006 945 0016 9434 0016 9434 0016 9434 0006 9359	00188443 00165433 00165433 00165433 0018843 0018843 00165433	000020153 00020145 0010843 000020153 00020115 0010843	1 00001232 0000 00001232 0000 00001232 00001232 00001 00001232 00001 00001232 0000 0000	Te dk > Mt (131.0) > Mt (13



Fig.8 Simulation result of RK4 based PBSCS on Vivado 17.3



Fig. 9 (a) x-y attractor, (b) y-z attractor, (c) x-z attractor

Parameter	Heun-based	RK4-based					
Maximum frequency (MHz)	359.71	359.71					
No. of DSP	2	4					
Number of 4 input LUTs	2912	2637					
Number of bonded IOBs	32	32					
Number of Slice Flip Flops	4977	4692					
Total On-chip Power(W)	0.178	0.179					

 Table 1

 Final report of the resources consumption

## **V CONCLUSION**

The Heun and RK4 algorithm basedPBS Chaotic generator have been synthesized using the Nexus 4 DDR XC7A100TCSG-1 (Artix7) and Basys3 (Artix7) from the Xilinx Vivado v.2017.3 design suite. RK4 based chaotic generator gives optimize result with the use of 2637 LUT's and 4692 registers with set clock period 2.78 ns which corresponds to maximum frequency achieved 359.71 MHz.The attracter of the system is generated by the data setare given in fig. 10(a-c) which are similar to PBSCS designed on analog platform.

## REFERENCES

 Ismail K., A. Tuaran O, IhsanPehlivan, "Implementation of FPGA-based real time novel chaotic oscillator", Nonlinear dynamics (2014); pp. 49-59.

- [2] Murat Tunaa, Can BülentFidan, "Electronic circuit design, implementation and FPGAbasedrealization of a new 3D chaotic system with singleequilibrium point", Optic ElsevierOptik(2016) pp. 11786–11799.
- [3] S. Banerjee, J. Kurths, "Chaos and cryptography: a new dimension in secure communications", Eur. Phys. J. Spec. Top., (2014) pp. 1441–1445.
- [4] I. Koyuncu, A.T. Ozcerit, I. Pehlivan, "An analog circuit design and FPGA-based implementation of the Burke-Shaw chaotic system", Optoelectron. Adv. Mater. Rapid Commun. (2013) pp. 635–638.
- [5] L. Merah, A. Ali-pacha, N.H. Said, "A pseudo random number generator based on the chaotic system of Chua's circuit and its real time", FPGA Implementation (2013) pp. 2719–2734.
- [6] S. C, ic, ek, A. Ferikog'lu, I'. Pehlivan, "A new 3D chaotic system: dynamical analysis, electronic circuit design, active control synchronization and chaotic masking communication application", Optik – Int. J. Light Electron Opt. (2016) pp. 4024–4030.
- [7] M. Tuna, I. Koyuncu, C.B. Fidan, I. Pehlivan, "Real time implementation of a novel chaotic generator on FPGA", in: 2015 23rd Signal Processing andCommunications Applications Conference (SIU), IEEE, 2015, pp. 698–701.
- [8] Alpana Pandey, R. K. Baghel, R.P. Singh, "Analysis and Circuit Realization of a NewAutonomous Chaotic System", International Journal of Electronics and Communication Engineering.ISSN 0974-2166 Volume 5, Number 4 (2012), pp. 487-495.