ISSN 1749-3889 (print), 1749-3897 (online)
International Journal of Nonlinear Science
Vol.7(2009) No.4,pp.467-474

An FPGA Real-time Implementation of the Chen's Chaotic System for Securing Chaotic Communications

Said Sadoudi *, Mohamed Salah Azzaz, Mustapha Djeddou, Mustapha Benssalah
Communications Systems Laboratory
Military Polytechnic School, Algiers, Algeria
(Received 29 November 2008, accepted 19 March 2009)

Abstract: In this paper, we present a new approach for real-time implementation of the Chen's chaotic system on Field Programmable Gate Array (FPGA). The approach consists on using structural hardware description language (VHDL) with a fixed-point representation, contrary to some previous works which use somewhat non optimal VHDL code generation using automatic code generation tools. The obtained real-time chaotic signals are compared with those obtained using ModelSim and Matlab. The hardware implementation exhibits good performances of throughput and cost in term of resources used. The aim of this implementation is to use this chaotic generator in secure digital chaotic communication system.

Keywords: Chen's chaotic system; chaotic communication; FPGA; VHDL

1 Introduction

Since Pecora and Carroll have discovered in 1990 that the chaotic systems can be synchronized [1], there has been a growing interest in digital chaotic communications over the past several years. In fact, researches in the application of chaos in communication have been greatly motivated [2-4]. Therefore, the communication systems based on chaos theory have exceeded the stage of the laboratory simulations. Practical tests are conducted in order to send digital message at Gbps speeds over one kilometer using commercial optical fiber [5]. This experience shows that chaotic generator can be controlled and used in communication systems [2-7].

However, there are many methods that use analogue circuits for chaotic communication, and to implement chaotic generators switched capacitor or analogue CMOS technology are used [8-9]. But, those methods exhibit some practical difficulties since the component values are varying with age, temperature etc [6-7]. And we must build both the transmitter and the receiver with very high components accuracy to insure information recovery, since the recovery characteristics are very sensitive to parameter mismatch between the transmitter and the receiver. Then, it is very difficult to deal with the problem of the chaotic synchronization. Hence, analogue implementation is very difficult thought it is not impossible to overcome this difficulties to some extent.

So, a hardware implementation of the chaotic generators can be a solution to overcome this problem. In fact, in this case the problem of parameter mismatch between the transmitter and the receiver does not exist. Nevertheless, the difficulty of the information recovery depends only on the channel noise sensibility. Thus, the chaotic synchronization becomes less difficult then the case of analogue implementation. In this context, advances in VLSI process technology have been applied to the manufacturing of reconfigurable logic including FPGA chips and helped their rapid growth in logic capacity, performance and popularity [10]. The extreme flexibility and the widespread acceptance of hardware description languages such as VHDL and VERILOG have made FPGA based systems the privileged choice for the hardware development and

*Corresponding author. *E-mail address*: sadoudi_said@yahoo.fr

IJNS.2009.06.30/250

implementation of high-performance compute-intensive applications. In the chaotic system implementation case, some previous works use somewhat non optimal VHDL code generation using automatic code generation tools as in [6,7]. However, the "high level" aspect of this method maintains the user far away from realities of the physical implementation. So, the result in terms of performance and density of resources used remains out of designer reach. If the final assessment does not agree perfectly with the requirements of the schedule conditions, it is impossible to intervene and to optimize. The quality and the effectiveness of these tools thus bring an undeniable comfort for the user but certain strategic points of the implementation are likely to escape to his control [11].

In this paper, we present a new approach for real-time implementation of the Chen's chaotic system in the FPGA circuit, Virtex-II type of Xilinx. Our approach consists on using the implemented forth order Rung-Kutta method (RK-4) to resolve the differential equations system of Chen. For that raison, we use directly VHDL language for the hardware description with a fixed point representation of data on 32 bits ($16Q16$) contrary to the method presented in [6,7]. The fixed-point arithmetic is simple and takes minimum chip area and has more advantages than disadvantages. The method presented in this paper is very simple, exhibits good performances of throughput and cost in term of resources used, and can be used for the implementation of all the others Chen's like chaotic systems such as Lorenz, Chua, Lü, Rössler etc. Therefore, this work will permit to use in choice these chaotic systems in secure digital chaotic communication systems.

This paper is organized as follows. In section 2 we introduce the hardware implementation of the Chen's chaotic system. We also present a short description of the Chen's chaotic generator, a description of the RK-4 method and the architecture of our solution. Section 3 is devoted to the implementation. Where, first of all a ModelSim simulation results are presented, secondly the synthesis results of the real time implementation are presented and finally the real-time results are given. Concluding remarks are given in Section 4.

2 Hardware Chen's chaotic system implementation

2.1 Chen's chaotic model

Since Chen has found another chaotic attractor from the Lorenz chaotic system, the study about Chen's system has attracted many researchers attention in these lasts years [3,12,13]. It has been experienced that Chen's chaotic system is relatively difficult to control compared to the Lorenz system due to the prominent three-dimensional and complex dynamic property [11]. Its equations system model is described by the following equations system:

$$\begin{aligned} \frac{dx}{dt} &= a(y-x) \\ \frac{dy}{dt} &= (c-a)x - xz - cy \\ \frac{dz}{dt} &= xy - bz \end{aligned} \tag{1}$$

2.2 Numerical resolution (RK-4)

The mathematical model of the Chen's system given in (1) is a simple three-dimensional autonomous system, which nevertheless is not topologically equivalent to the Lorenz one [12]. So, to implement it in the FPGA circuit we must use one of the existing numerical resolution methods since most ordinary differential equations are note soluble analytically. There are a handful of methods known and widely used (i.e., Runge-Kutta, Adams-Bashforth-Moulton and Backward Differentiation Formulae methods). All these methods discretize the differential system to produce a difference equation or map. For our problem, we have used the RK-4 method because it is one of the well-known numerical methods for differential equations, produces a far more accurate estimate of the solution and it is the most used for the resolution of the chaotic systems [14]. This method is given by the following equations.

Let us consider a first order differential equation:

$$F(x,y) = \frac{dy}{dx} \tag{2}$$

The RK-4 method uses several intermediate points to calculate the value of y_{i+1} starting from the value of y_i, h being the step length in x. It defines two series:

1. A first series that define the values of x :
   - Initial term: x_0
   - Relation of recurrence:
$$x_{i+1} = x_{i+h} \tag{3}$$
2. A second series that define the values of y :
   - Initial term: y_0
   - Relation of recurrence:
$$y_{i+1} = y_i + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4) \tag{4}$$

with

$$k_1 = h \times f(x_i, y_i) \tag{5}$$

$$k_2 = h \times f(x_i + \frac{h}{2}, y_i + \frac{k_1}{2}) \tag{6}$$

$$k_3 = h \times f(x_i + \frac{h}{2}, y_i + \frac{k_2}{2}) \tag{7}$$

$$k_4 = h \times f(x_i + h, y_i + k_3) \tag{8}$$

The original classical Chen's chaotic system as it is described in (1) has such parameters values a = 35, b = 3, c = 28 and the initial conditions x_0 = y_0 = z_0 = 1. The numerical resolution of the system (1) with the RK-4 method using Matlab gives the corresponding chaotic signals x, y and z presented in the figure 1, the value of the step length used is h = 0.01. The figure 2 and 3 show two different strange attractors of the chaotic system. The first one is in the phase plane $(x - z)$ and the second one in the phase plane $(x - y)$. Next, these results will be useful as references for the implementation results.

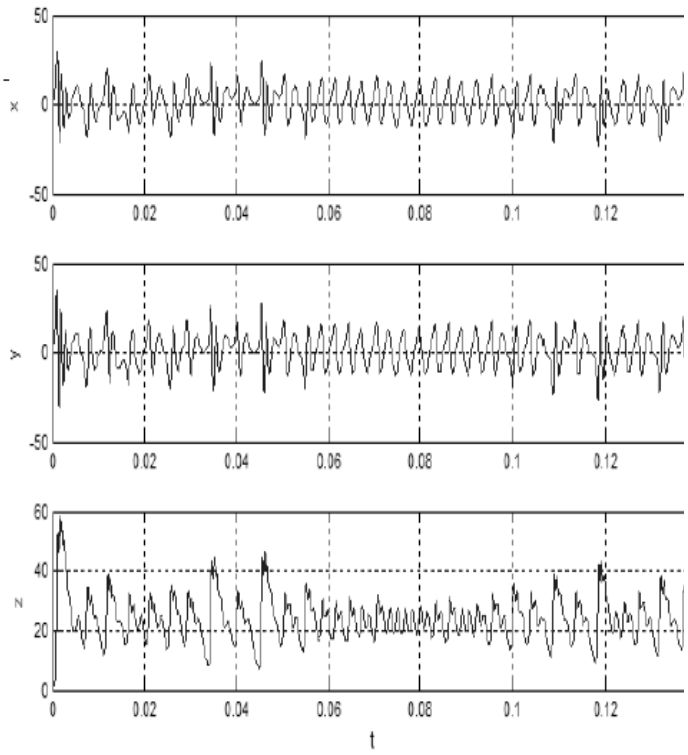

Figure 1: Chen's chaotic signals.

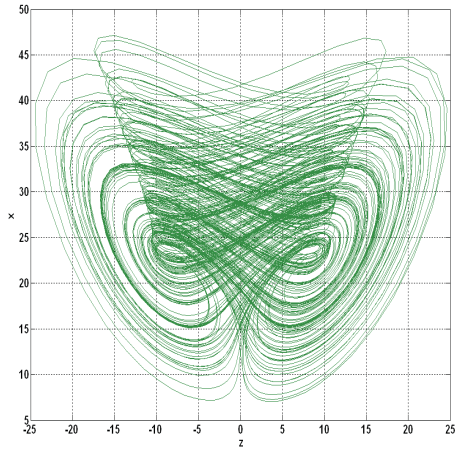

Figure 2: Chen's strange attractor (phase plane $x - z$).

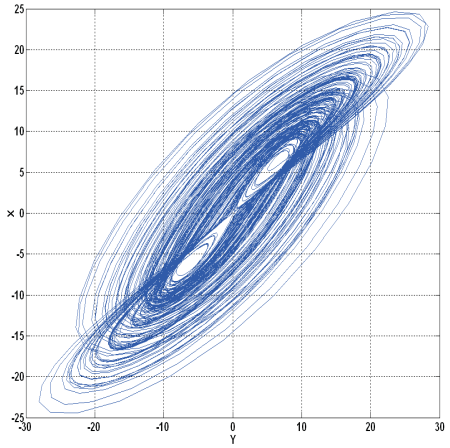

Figure 3: Chen's strange attractor (phase plane $x - y$).

2.3 Architecture of the solution

In this section, we present the architecture of our solution proposed for the hardware implementation of the Chen's chaotic generator in the circuit FPGA Virtex-II type of Xilinx. This solution consists first of all on describing directly the RK-4 method with the hardware description languages VHDL. Then, the optimal VHDL code obtained is used to resolve the Chen's chaotic system given by (1). Contrary to some recent techniques which use somewhat non optimal VHDL code generation using automatic code generation tools as in [6,7]. However, it should be noted that the continuous chaotic signals are real. But, in the VHDL language, we cannot treat real signals, variables or parameters with infinite precision. Only finite resolution numbers using binary representation are treated. To get around this problem, we have adopted the implementation with a fixed point representation of the real data on 32 bits ($16Q16$), i.e. all data is fixed point format with 16 bits integer and 16 bits fraction. The general diagram of the digital implementation is depicted in the figure 4 where:

- Clk : Clock system.
- Reset: Asynchronous re-initialization of the system.
- DAC: Digital/Analog Converter.
- Wr: Control signal, to write the 32 bits data on the DAC.
- X.32bits: The Chen's chaotic signal x represented on 32 bits.

From the diagram, we note that we have two principal blocks. The Automat block and the Chen block. The first one represents the brain of the system. In fact, it directs the totality of the system tasks. It orders the Chen block to resolve the system (1) with the RK-4 method. Once the chaotic signals (x, y and z on 32 bits) are obtained, they are sent to the DAC one after one and the process is repeated in time. Therefore, the real-time chaotic signals obtained at the output of the DAC can be visualized on the digital oscilloscope.

It should be noted that our solution, based on the implementation of the RK-4 method, is simple and can be used for the implementation of all the others like Chen's chaotic systems such as Lorenz, Chua, Lü, Rössler ...etc.

3 Implementation results

3.1 ModelSim Simulator results

To test the effectiveness of our solution, we simulate the resolution of the Chen's chaotic system by the adopted RK-4 method with ModelSim simulator. The obtained results are presented in the figure 5 where the chaotic signals x, y and z are represented on 32 bits.

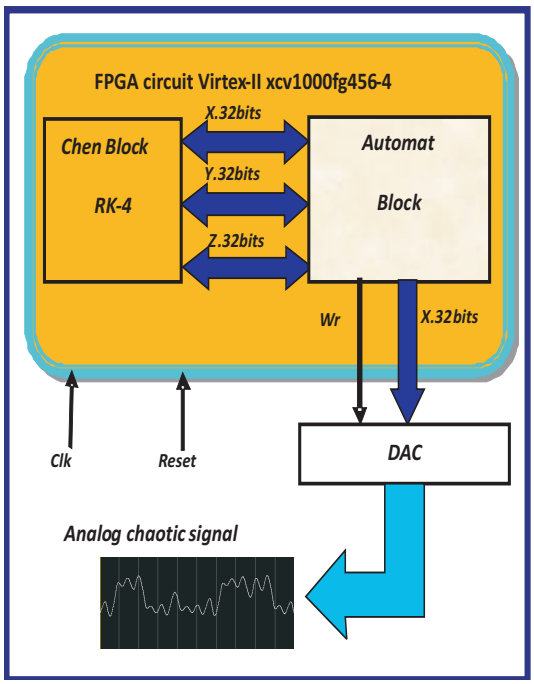

Figure 4: General diagram of the digital implementation of Chen's chaotic system in the FPGA circuit.

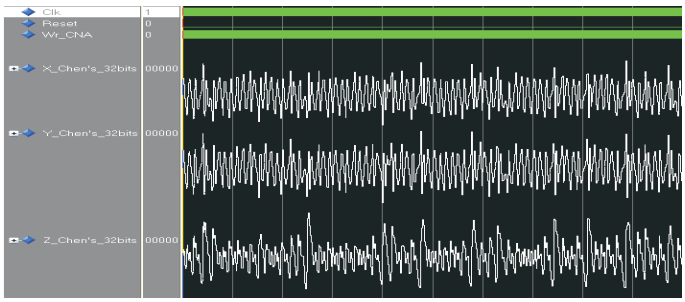

Figure 5: Simulation results of the Chen's chaotic generator (x, y and z), obtained by ModelSim.

3.2 Synthesis results

After the generation of the VHDL code, a functional verification must be realized to carry out a verification of the FPGA design. This step is very important in the FPGA implementation. In fact, it allows a verification of the capacity overshooting in the resources consummations and then test if the implementation in the FPGA circuit, Virtex-II xcv1000fg456-4 of Xilinx, is possible or not. The final report of the resources consummations is presented in table 1.

Table 1 shows that there is no problem of capacity overshooting. Our implemented method has used all the multipliers of the FPGA circuit but we notice that we have gained in area in term of slices (22 % of slices). So, the Chen's chaotic system can be implemented in the FPGA circuit Virtex-II xcv1000fg456-4 of Xilinx.

Another criterion of performances is the calculation of the throughput T at the output of the FPGA circuit. This depends on three parameters: size of the storage bloc S, the algorithm frequency F and the necessary clock cycles number C. It is calculated according to the following formula:

$$T = \frac{S \times F}{C} \tag{9}$$

However, the implementation of the Chen's chaotic requires 6 cycles of clock to treat data on 32 bits, with a maximum frequency of 22.850 MHz, which gives then a throughput of 121.86 Mbits/s.

The real-time chaotic signals of the Chen's chaotic generator, obtained by the direct implementation in the FPGA circuit Virtex-II xcv1000fg456-4 of Xilinx, are given by the figures 6,7 and 8. This figures show

Device utilization summary FPGA: 2v1000fg456-4		
Number of Slices	1138 out of 5120	22%
Number of Slice Flip Flops	885 out of 10240	8%
Number of 4 input LUTs	1969 out of 10240	19%
Number of bonded IOBs	11 out of 324	3%
Number of MULT18X18s	40 out of 40	100%
Number of GCLKs	2 out of 16	12%
Maximum frequency	22.850 MHz	

Table 1: Final report of the resources consummations.

the real-time chaotic signals x, y and z seen in the digital oscilloscope respectively. One can compare these results and those obtained using Matlab simulation (figure 1) and ModelSim (figure 5), to see that all of them have the same form. The measured real-time attractors $(x-z)$ and $(x-y)$ are presented in the figures 9 and 10 respectively. These results confirm that the implemented chaotic system work well in his chaotic mode. Then, our implemented method is validated.

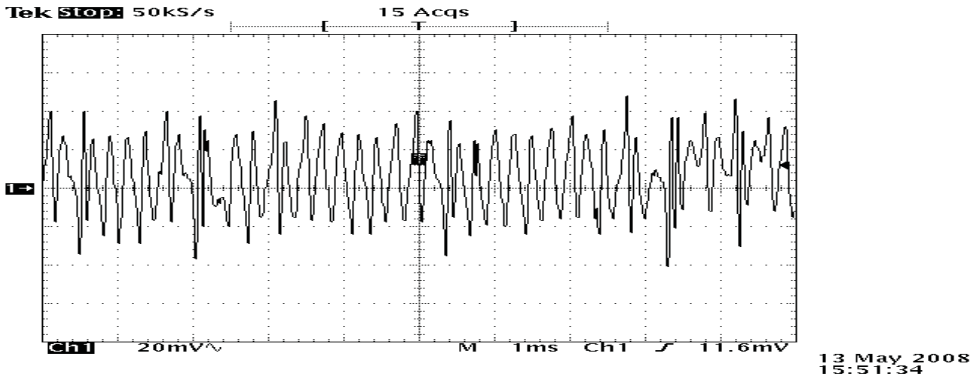

Figure 6: Real-time Results of the implemented Chen's chaotic generator: chaotic signal x.

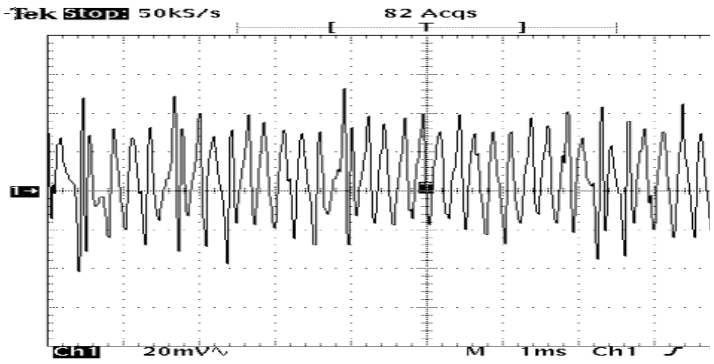

Figure 7: Real-time Results of the implemented Chen's chaotic generator: chaotic signal y.

It should be noted that, the real-time chaotic signals of the Chen's chaotic system, obtained by our implemented method in the FPGA circuit of Xilinx, are obtained for the first time to the best of our knowledge. In addition, our implemented method is more optimal then the existing method [6,7], because we use directly the hardware description language VHDL to generate the bit file [11], of the RK-4 method, needed to resolve the chaotic system and then for the implementation on FPGA.

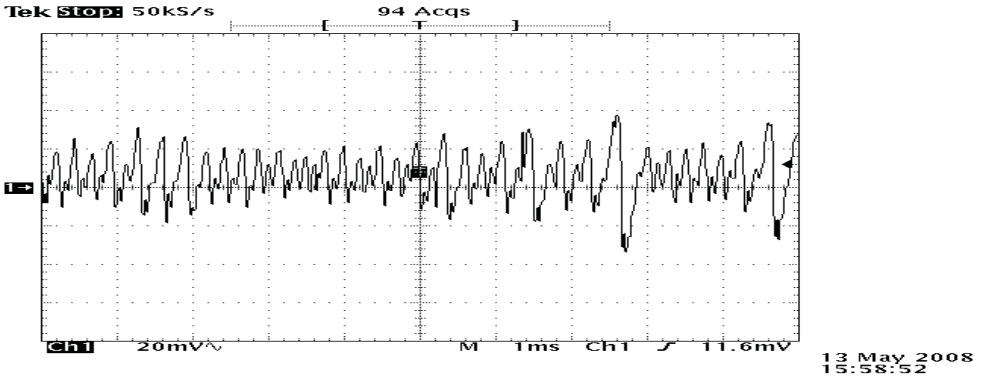

Figure 8: Real-time Results of the implemented Chen's chaotic generator: chaotic signal z.

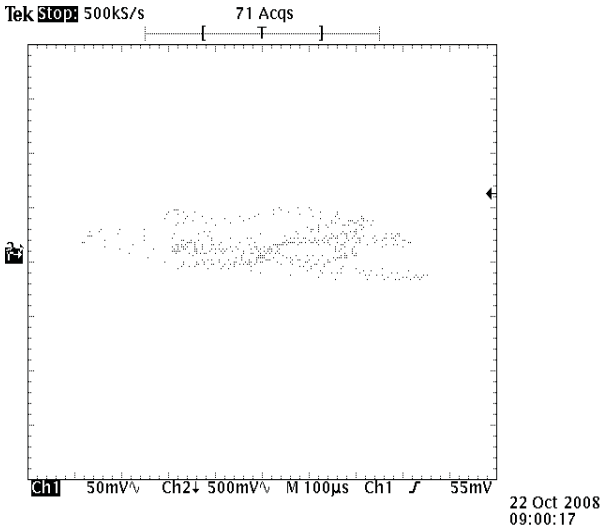

Figure 9: Measured attractor $(x - z)$ of the real-time Chen's generator.

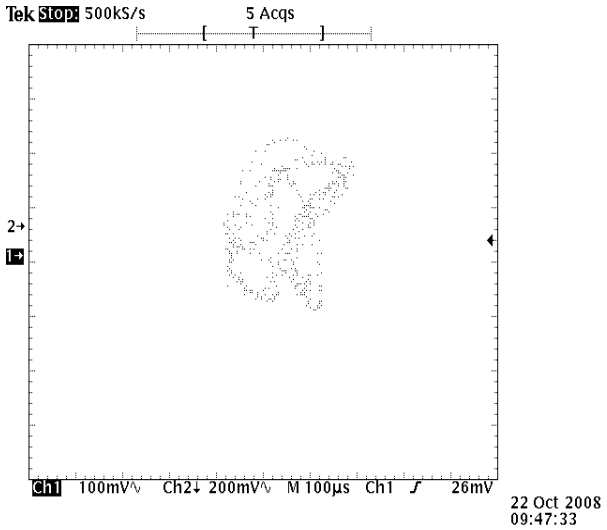

Figure 10: Measured attractor $(x - y)$ of the real-time Chen's generator.

4 Conclusion

A new optimal and simple approach for real-time implementation on FPGA, of continuous chaotic generator systems, is developed. The developed approach consists on using the implemented forth order Rung-Kutta method to resolve the differential equations system of the Chen's chaotic generator. To accomplish this, we use directly the VHDL language for the hardware description with a fixed point representation of data on 32 bits ($16Q16$ format). Contrary to some recent techniques, which use somewhat non optimal VHDL code generation using automatic code generation tools in order to generate the bit file needed to the implementation in the FPGA circuit.

The real-time Chen's chaotic signals obtained are almost identical in form to those of the simulations ones and then validated. In addition, the implemented solution exhibits good performances of throughput and cost in term of resources consummations. In fact, we have obtained low consummations in area with 22 % of slices and we have achieved a maximal throughput of 121.86 Mbits/s. Finally, our solution can be used for the hardware implementation of all the others Chen like chaotic systems such as Lorenz, Chua, Lü, Rössler ...etc. Therefore, it is possible to use in choice these different chaotic systems in secure digital chaotic communication system.

Acknowledgements

The authors wish to thank Dr. Mustapha Djeddou and Dr. Camel Tanougast. This work was supported by the Military Polytechnic School of Algeria.

References

[1] T L Carroll, L M Pecora: Synchronization in chaotic systems. *Phy. Rev. Lett*. 64(8):821-824(1990)

[2] K M Cuomo, A V Oppenheim, Steven H Strogatz: Synchronization of Lorenz-Based Chaotic Circuits with Applications to Communications. *IEEE transactions on circuits and systems: analog and digital signal processing*. 40(10):(1993)

[3] K Fallahi, R Raoufi, H Khoshbin: An application of Chen system for secure chaotic communication based on extended Kalman filter and multi-shift cipher algorithm. *Communications in onlinear Science and Numerical Simulation* 13:763-781(2008)

[4] T Yang:A survey of chaotic secure communication systems. *Inter. J. of Computational Cognition*. 2(2):81-130(2004)

[5] IEEE Spectrum. (2006)

[6] M I Sobhy, M A Aseeri, A E R Shehata: Real Time Implementation Of Continuous (Chua And Lorenz) Chaotic Generator Models Using Digital Hardware. (1999)

[7] M A Aseeri, M I Sobhi and P Lee: Lorenz Chaotic Model Using Field Programmable Gate Array (FPGA). *Midwest Symposium on Circuit and Systems*. (2002)

[8] C Y Cha, S G Lee: Complementary Colpitts Oscillator in CMOS Technology. *IEEE Transaction on microwave theory and techniques*. 53(3):(2005)

[9] T Matsumoto: Chaos in electronic circuits. *Proc. IEEE*. 75(8):1033-1046(1987)

[10] Celoxica, RC200 platform deform developer's kit.(2005)

[11] E Garcia: Implanter des fonctions DSP sur FPGA de façon simple et efficace. *Electronique- Décembre*. 164:(2005)

[12] J Lü, G Chen: Anew chaotic attractor coined. *Inter. J. of Bifurcation and Chaos*. 12(3):659-661(2002).

[13] X Wang, L Tian, L Yu: Linear Feedback Controlling and Synchronization of the Chen's Chaotic system. *Inter. J. of Nonlinear Science*. 2(1):43-49(2006).

[14] J H E Cartwright, O Piro: The Dynamics of Runge-Kutta Methods. *Int. J. Bifurcation and Chaos*. 2:427-449(1992)

[15] X Shi, Q Zhu: An Exact Linearization Feedback Control of CHEN Equation. *Inter. J. of Nonlinear Science*. 3(1):58-62(2007).

[16] G Cai, J Huang, L Tian, Q Wang: Adaptive Control and Slow Manifold Analysis of a New Chaotic System. *Inter. J. of Nonlinear Science*. 2(1):50-60(2006).

[17] L Tian, S Zhang, G Yang: Some Results of a Class of Systems and Their Application in Chaotic Synchronization. *Inter. J. of Nonlinear Science*. 2(3):159-165(2006).