Eur. Phys. J. Special Topics
© EDP Sciences, Springer-Verlag 2014
DOI: 10.1140/epjst/e2014-02144-8

THE EUROPEAN
PHYSICAL JOURNAL
SPECIAL TOPICS

Regular Article

Synchronization in coupled Ikeda delay systems

Experimental observations using Field Programmable Gate Arrays

D. Valli[1,a], B. Muthuswamy[2,b], S. Banerjee[3,c], M.R.K. Ariffin[3,4], A.W.A. Wahab[5], K. Ganesan[1,a,d], C.K. Subramaniam[1,a], and J. Kurths[6,d]

[1] TIFAC-CORE, Vellore Institute of Technology, India
[2] Milwaukee School of Engineering, USA
[3] Institute for Mathematical Research, Universiti Putra Malaysia, Malaysia
[4] Mathematics Department, Faculty of Science, Universiti Putra, Malaysia
[5] Faculty of Computer Science & Information Technology, University of Malaya, Malaysia
[6] Humboldt University, Berlin and Potsdam Institute for Climate Impact Research, Germany

Received 17 February 2014 / Received in final form 21 March 2014
Published online 21 April 2014

Abstract. In this work, we demonstrate the use of a Field Programmable Gate Array (FPGA) as a physical platform for realizing chaotic delay differential equations (DDE). Moreover, using our platform, we also experimentally study the synchronization between two time delayed systems. We illustrate two different experimental approaches – one is hardware co-simulation (using a Digilent Atlys with a Xilinx Spartan-6 FPGA) and the other is analog output (using a Terasic DE2-115 with an Altera Cyclone IV E FPGA).

1 Introduction

Time-delay is inherent in many physical systems and could be caused by (for example) lag between the sensing of disturbance and the triggering of an appropriate response [20]. Such time-delay systems are be modeled by delay differential equations that are infinite dimensional [20]. The attractiveness of chaotic DDEs is that due to the infinite dimensionality, even first order systems can exhibit chaos [20]. Classic example are the Mackey-Glass system [6,7] and the Ikeda DDE [9,10,20] shown in Eq. (1). Parameter values and simulation results will be provided in Section 3.

$$\dot{x} = \mu \sin(x(t-\tau)) - \alpha x(t), \; x(t \leq 0) = 0.1. \tag{1}$$

However, physical realization of time-delay systems pose challenges due to the implementation of the time delay. Although physical realization of delays using analog electronic circuits such as tunable delay and RC filter have been studied [4,15], an

[a] e-mail: vallinandha@gmail.com; {kganesan,subramaniam}@vit.ac.in
[b] e-mail: muthuswamy@msoe.edu
[c] e-mail: santoban@gmail.com; rezal@putra.upm.edu.my
[d] e-mail: Juergen.Kurths@pik-potsdam.de

attractive solution is to implement the differential equation on an FPGA. These devices are massively parallel architectures whose hardware functionality can be appropriately configured. The attractiveness of the FPGA stems from the fact that these devices can be configured with any nonlinearity. One can also implement numerical algorithms (such as Euler's method) and specify time delays quite easily (without the burden of a processor) using a hardware description language (HDL). Hence these platforms can serve as engineering solutions for conducting scientific experiments. Moreover, 100 MHz clock frequencies are quite common on modern (21st century) FPGAs. Such high frequencies, coupled with a variable data-path (32-bit, 64-bit, 128-bit etc.) specification, provide us with a platform that is capable of handling large parallel blocks of data transfer, unlike a traditional processor. Thus, this work is a natural followup to our previous result [21], where we utilized a soft-processor on the FPGA to sequentially emulate chaotic DDEs.

We chose to focus on the Ikeda DDE since this differential equation involves a trigonometric nonlinearity and therefore, physical analog realization of this system is difficult. To our knowledge, this work is the first paper to report on analog chaotic waveforms from the Ikeda DDE, with a tunable bandwidth for the underlying chaotic system.

The paper is organized as follows: in Section 2, we will begin with a general discussion of the system block diagram. In Section 2, we will also discuss the two different approaches to FPGA hardware design – utilizing hardware co-simulation and for an FPGA board equipped with an ADA (Analog-to-Digital and Digital-to-Analog) converter[1], we show how one could obtain analog results.

Section 3 shows results from both numerical simulation and FPGA realization of the Ikeda DDE. To illustrate the robustness of our approach, we utilize two different FPGA platforms [22,23] from two different FPGA [24,25] manufacturers.

The observation of chaos [3] and also its synchronization in physical systems [1,2, 5–8] has led to significant research results [16–18]. We show complete synchronization results (unidirectional and bidirectional coupling) for the Ikeda DDE in Section 4 for both FPGA platforms.

We also chose to experimentally demonstrate via hardware co-simulation Delay Time Modulation (DTM) since this can lead to more secure chaotic systems [19]. We also include details of parameter mismatch experiments. Section 5 concludes the paper with suggestions for future work.

2 An overview of the proposed FPGA design

An overall system block diagram is shown in Fig. 1. For simplicity, we have considered only first-order systems with a single constant delay [11]. The block diagram in Fig. 1 is simply a graphical representation of the delay differential equation in Eq. (2).

$$\dot{x} = f(x, x(t-\tau)). \tag{2}$$

But, the robustness of the FPGA platform is that each block in Fig. 1 can be mapped into hardware. The nonlinearities and delays can be implemented using a high-level specification (such as Simulink) and, for the integrator, the appropriate numerical algorithm (Euler's method, Runge-Kutta methods etc.) can also be specified. There are two primary methods for mapping Fig. 1 onto an FPGA.

[1] Audio codecs that have onchip ADAs are very common on FPGA boards. In fact, the Digilent Atlys platform does provide an on-board audio codec. One could also simply connect an external ADA board to an FPGA.

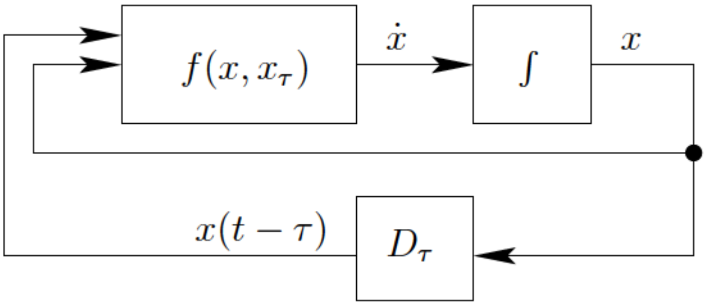


Fig. 1. The overall block diagram for specifying the delay differential equation. Initial conditions are specified in the integrator block.

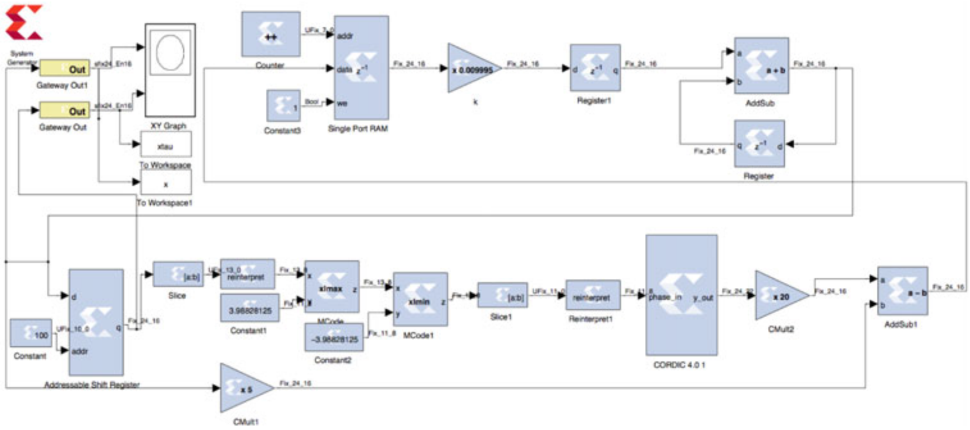


Fig. 2. We have chosen to implement the entire block diagram in hardware and simply utilize the software (Simulink) interface to display the chaotic waveform(s).

The first method is hardware co-simulation. As the name suggests, we execute selected (resource-intensive) components of the algorithm as hardware on the FPGA. The rest of the algorithm is executed in a high-level environment such as Simulink and communication is established between hardware and software via protocols such as ethernet. Figure 2 shows the Ikeda DDE specified using hardware co-simulation for execution on a Digilent Atlys platform.

The advantage of hardware co-simulation is in simplicity of the interface: the underlying communication between hardware and software is abstracted from the user. Nevertheless, the disadvantage is the very low frequency bandwidth available for examining the chaotic signal, due to data buffering between hardware and software.

The second method simply implements the entire design on the FPGA hardware and utilizes an ADA converter to interface the FPGA to external peripherals. This allows us to view the chaotic waveforms on an analog scope and also study synchronization behavior by using external forcing inputs. Figure 3 shows a high level block diagram for implementing the system in Eq. (2). The analog output (using an audio codec D/A, for example) solution does not suffer from the low frequency bandwidth of the hardware co-simulation approach. The simple justification is that there is no need for data transfer from the FPGA to a system running MATLAB. The entire design executes on the FPGA.

Nevertheless, implementing either hardware co-simulation or analog output using an FPGA based development platform requires the user to have an intimate

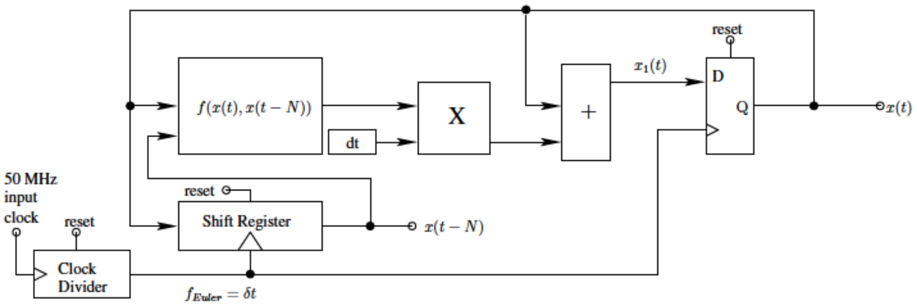


Fig. 3. Hardware block diagram for specifying DDEs, using forward-Euler's method. Note that one could specify more advanced integration methods, such as Runge-Kutta. The clock divider block is configured to divide the input 50 MHz (board clock) such that delays introduced by the shift register etc. can be accommodated. This clock (with frequency given by f_{Euler}) is used for the shift register and the output D flip-flop. The shift amount is related to dt by $N = \frac{1}{dt}$. Since design is synchronous, all sequential components have a well-defined reset state. The nonlinear function f, the multiplier and adder blocks can be implemented using core generators or specified in Simulink. $x(t)$ and $x(t-N)$ also serve as outputs to be send to the digital-to-analog converter.

knowledge of FPGA digital design practices. The upcoming book [13] would serve as an excellent reference for those interested in implementing chaotic systems on FPGAs.

In the next section, we will first discuss numerical simulation and then illustrate both FPGA realization approaches.

3 Numerical simulation and experimental results

3.1 Numerical simulation results

Consider the Ikeda DDE, reproduced in Eq. (3). Parameter values are $\mu = 6, \tau = 1, \alpha = 1$ with the infinite set of initial conditions: $x(t \leq 0) = 0.1$.

$$\dot{x} = \mu \sin(x(t-\tau)) - \alpha x(t). \tag{3}$$

The Ikeda DDE models a passive optical bistable resonator [9,10]. Experimentally, it was observed [9,10] that when the intensity of incident light is varied, the electric field strength in the nonlinear dielectric medium placed in a ring cavity undergoes state transitions from stationary to periodic and then to non-periodic states. If the length of the cavity is sufficiently large, time delayed feedback occurs and the nonperiodic state is chaotic in nature. Physically, in Eq. (3), x represents the phase lag of the electric field across the resonator, μ is the light intensity injected into the system, τ is the feedback delay time in the resonator and α is the relaxation coefficient.

Figure 4 shows a Simulink block diagram that utilizes forward-Euler's method along with 32-bit single-precision floating-point quantization to perform a discrete simulation of Eq. (3). We resorted to a discrete-simulation since the FPGA design is a sampled and quantized specification of our differential equation.

3.2 Experimental results: Hardware co-simulation

Figure 5 shows the phase-plot obtained from the hardware co-simulation in Fig. 2. However, the block diagram in Fig. 2 can only be sampled at 100 Hz because the full bandwidth of our FPGA is unavailable since we have to communicate waveform data to the design running in Simulink.

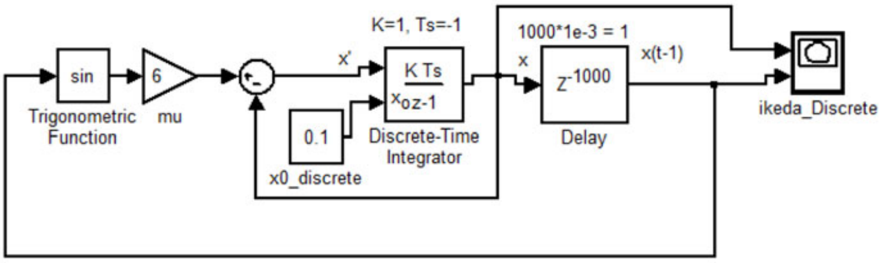


Fig. 4. We are using a fixed-step ode1 (Euler) solver in Simulink, with a step size of 0.001. We run the simulation for 25 seconds. Since sampling time of the discrete-integrator block is specified to be −1 (inherited), we use K = 1.

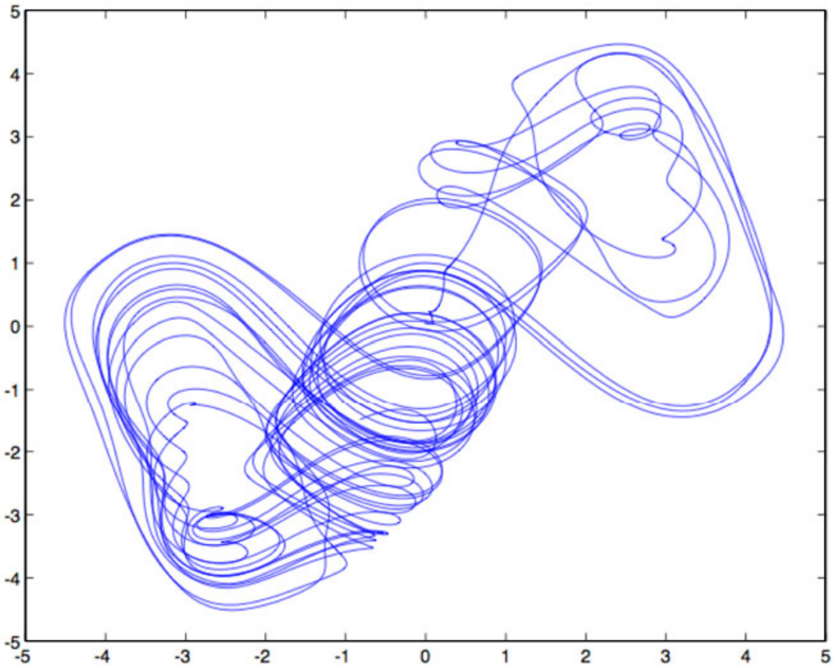


Fig. 5. Result from hardware co-simulation, plotted using XY graph in Simulink. y-axis is $x(t)$, $x-axis$ is $x(t-\tau)$.

3.3 Experimental results: Analog output

Figure 6 shows the block diagram of the VHDL specification, which is Fig. 3 adapted to the Ikeda DDE. Figure 7 shows the phase plot from specifying the Ikeda DDE in hardware, for the Terasic DE2-115.

The sampling frequency for Euler's method is approximately 762 Hz. This can be computed as $dt \cdot \delta t = \frac{1}{1024}\frac{50 \cdot 10^6}{64}$. We have chosen $dt = \frac{1}{1024}$ and utilized a 64× clock divider to slow down the 50 MHz board clock. The ADA converter is configured to run at 96 KHz and hence we can substantially increase the sampling frequency for Euler's method.

The reference designs in the links for both hardware co-simulation and analog output also include synchronization results. The hardware co-simulation reference design also includes DTM and parameter mismatch.

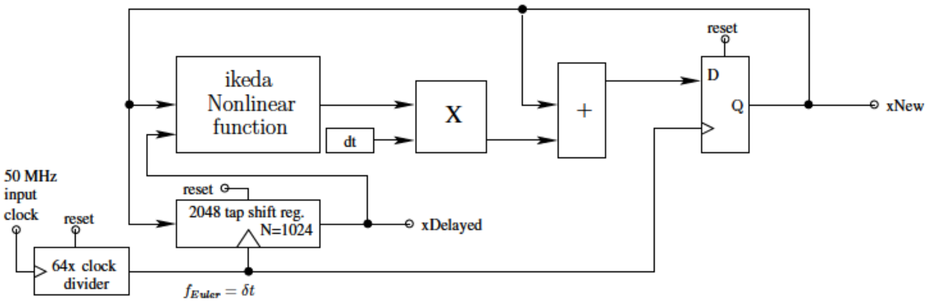


Fig. 6. The block diagram shows specific signal names from the FPGA HDL design specification. $dt = \frac{1}{1024}$ and internal bus width is 32-bit (for single-precision IEEE 754 floating point). Output is converted to 5.27 fixed point format and the high 16-bits are extracted for the 16-bit data input to the audio codec.

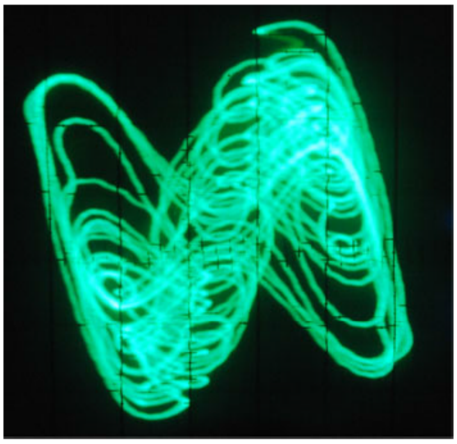


Fig. 7. $x(t)$ (Y-input) vs. $x(t-\tau)$ (X-input) displayed on an oscilloscope. Scales for both channels are 0.5 V/div.

A compilation report result for the analog output realization on the Terasic board is shown in Fig. 8.

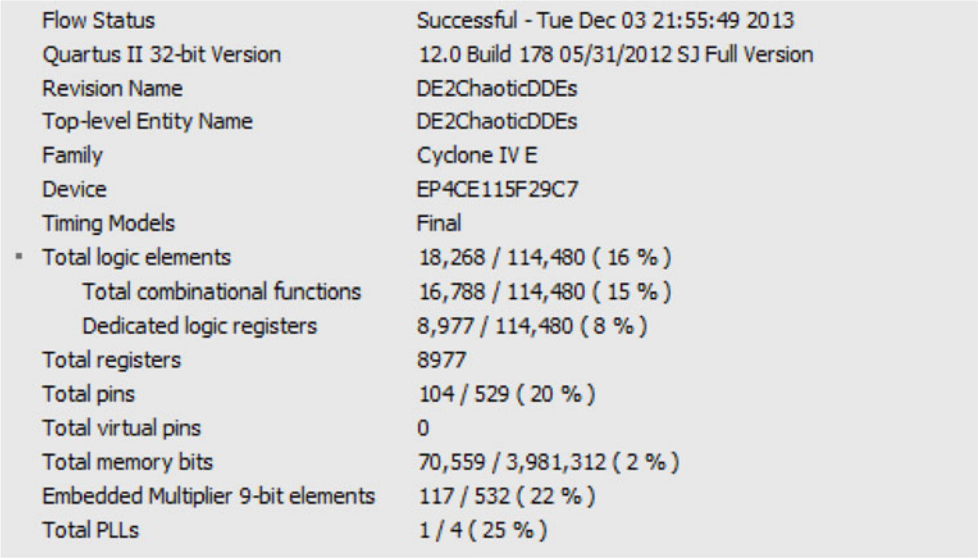
4 Synchronization experiments

Synchronization of chaotic systems is of interest because its possible applications in the domain of cryptography and secure communications [14]. Identical synchronization is the very effective and the simplest type of synchronization that is characterized by perfect follow-up of two chaotic trajectories. Synchronization is achieved by means of a coupling factor. We consider linearly coupled Ikeda systems as the drive and response systems, described by Eq. (4) and Eq. (5) respectively.

$$\dot{x} = -\alpha x + \mu \sin x(t-\tau) \tag{4}$$

$$\dot{y} = -\alpha y + \mu \sin y(t-\tau) + k(t)(x-y) \tag{5}$$

$k(t)$ is the coupling factor between drive and response system [5,12]. In the following subsections, we illustrate complete synchronization using unidirectional and



Flow Status	Successful - Tue Dec 03 21:55:49 2013
Quartus II 32-bit Version	12.0 Build 178 05/31/2012 SJ Full Version
Revision Name	DE2ChaoticDDEs
Top-level Entity Name	DE2ChaoticDDEs
Family	Cyclone IV E
Device	EP4CE115F29C7
Timing Models	Final
▪ Total logic elements	18,268 / 114,480 (16 %)
Total combinational functions	16,788 / 114,480 (15 %)
Dedicated logic registers	8,977 / 114,480 (8 %)
Total registers	8977
Total pins	104 / 529 (20 %)
Total virtual pins	0
Total memory bits	70,559 / 3,981,312 (2 %)
Embedded Multiplier 9-bit elements	117 / 532 (22 %)
Total PLLs	1 / 4 (25 %)

Fig. 8. The report above shows how many logic elements our design occupies on the Cyclone IV E FPGA. Note that we are only 16% of the FPGA, even though we have two Ikeda DDEs (drive and response, for synchronization). Hence our FPGA can accommodate more chaotic DDEs. Note that such a report is difficult to obtain for the hardware co-simulation since part of the design does not run on the FPGA.

bidirectional coupling via the FPGA. The parameters are taken as $\mu = 20, \alpha = 5$, $\tau = 1$. Figure 9 shows the analog output with the parameters for synchronization.

4.1 Unidirectional coupling

In unidirectional coupling, the drive system is free to evolve and the response system is influenced by the coupling factor $k(t)$. Due to this, the dynamics evolution of the response system is to follow the dynamics of the drive system in the course of time. We investigated chaos synchronization in unidirectionally coupled Ikeda systems with two types of coupling factors. In the first type, $k(t)$ is a square wave coupling represented by

$$(t_0, k_1), (t_1, k_2), (t_2, k_1), (t_3, k_2)....... \qquad (6)$$

where $t_j = t_0 + (j-1)\tau$, $j \geq 1$ with $k_1 \neq k_2$. It is observed that the amplitude of the control parameter $k(t)$ is the key factor to achieve synchronization between drive and response, larger the amplitude quicker the convergence into synchronization, provided that the conditional Lyapunov exponents of the response systems are all negative. The threshold value is chosen as $k_1 = 0$ and $k_2 = 50$ for square wave coupling. We also used as a second type of coupling factor defined by

$$k(t) = -\alpha + 2\mu|\cos(y(t-\tau))|. \qquad (7)$$

Figures 10 and 11 depicts the experimental setup of complete synchronization for the above two types of coupling factors.

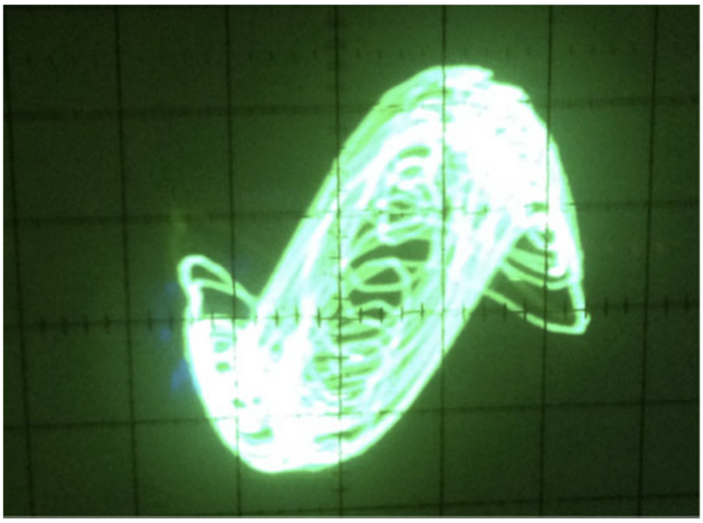


Fig. 9. $x(t-\tau)$ (X-input) vs. $x(t)$ (Y-input) displayed on an oscilloscope for the Ikeda attractor with parameters for synchronization experiments. Scales for both channels are 0.2 V/div.

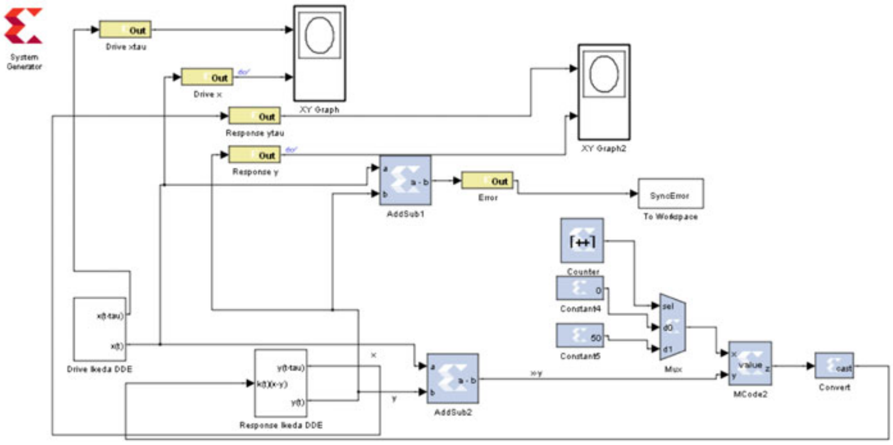


Fig. 10. Complete synchronization of Ikeda DDE for unidirectional square wave coupling using Xilinx Blockset.

4.2 Bidirectional coupling

In bidirectional coupling, both drive and response systems are coupled with each other by a coupling factor $k(t)$ that induces mutual synchronization. For this bidirectional coupling, the drive (Eq. (8)) and response (Eq. (9)) systems are considered as

$$\dot{x} = -\alpha x + \mu \sin x(t-\tau) + k(t)(y-x) \tag{8}$$

$$\dot{y} = -\alpha y + \mu \sin y(t-\tau) + k(t)(x-y). \tag{9}$$

The synchronization error is computed by $e(t) = x(t) - y(t)$, which is the measure for convergence of two chaotic trajectories. Figures 12a through 12d show results

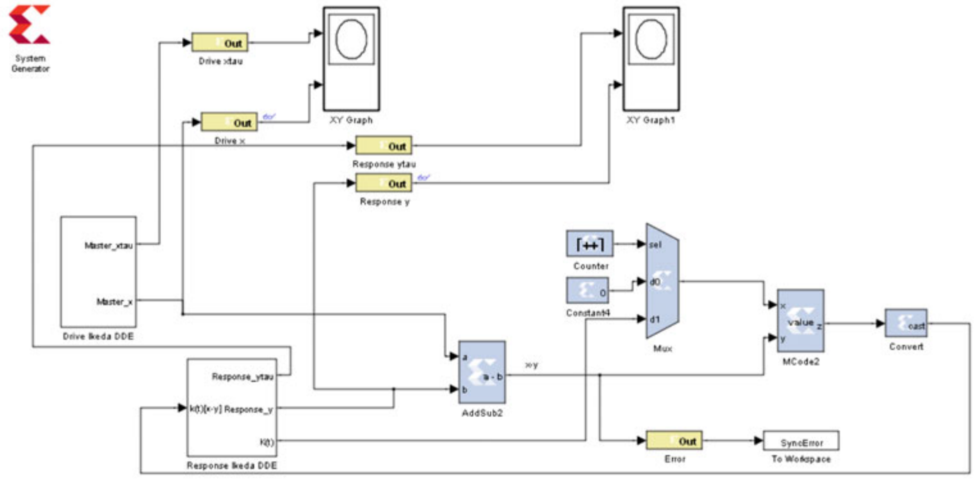


Fig. 11. Complete synchronization of Ikeda DDE for for unidirectional coupling function $k(t) = -\alpha + 2\mu|\cos(y(t-\tau))|$ using Xilinx Blockset.

from the analog output synchronization experiment. In order to generate the analog output, we simply passed $x(t) - y(t)$ into a DAC channel. It is observed that the synchronization is quicker in bidirectional coupling compared to the unidirectional coupling for the same parameters.

In order to shed more light on synchronization, we have used the analog oscilloscope to obtain X-Y plots where X is $x(t)$ (drive) and Y is $y(t)$ (response). Figure 13 first shows the X-Y plot when the drive and response systems are not synchronized.

The X-Y plots as a result of synchronization are shown in Figs. 14a through 14d.

4.3 Effect of DTM and synchronization results

To observe effect of modulated time delay on the chaotic dynamics of the Ikeda DDE, delay $\tau(t) = 1.5|\sin(t)|$ is considered in Eq. (10). To generate the sinusoidal function, DDS compiler is used in Xilinx Blockset with the sample time chosen as 0.001.

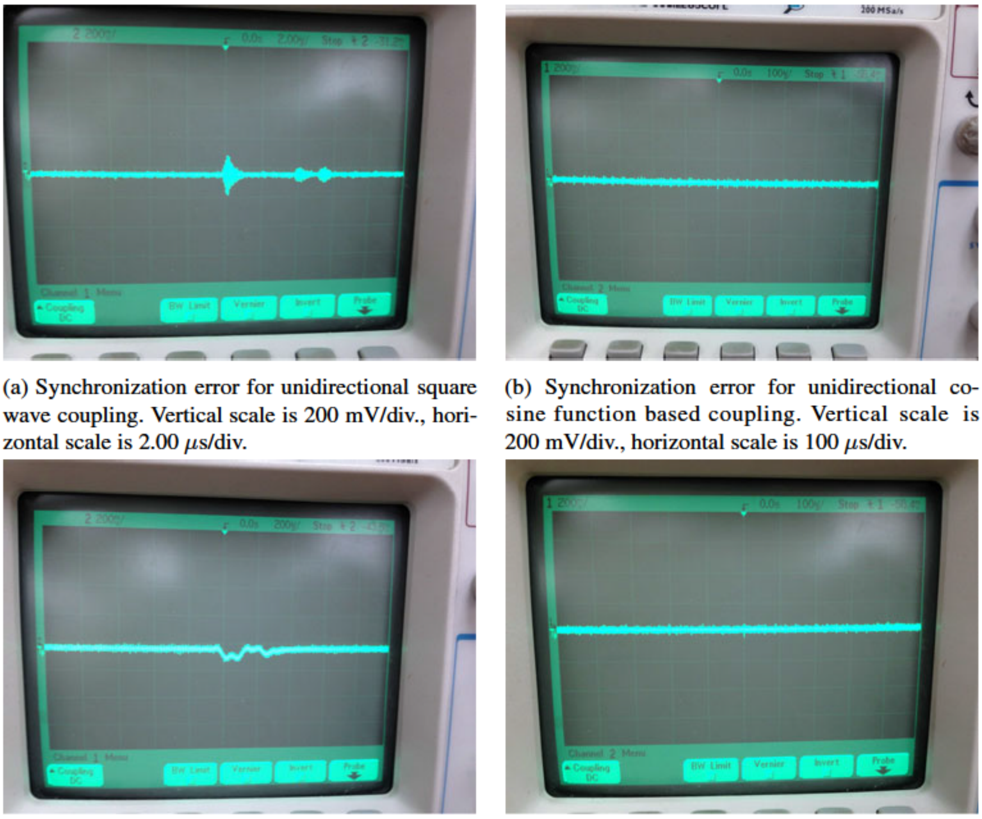
$$\dot{x} = \mu \sin(x(t-\tau(t)) - \alpha x. \tag{10}$$

The corresponding FPGA block diagram is shown in Fig. 15. The resultant phase-plot is shown in Fig. 16. The plot shows that modulation in delay time has significant effect on the chaotic dynamics of the system.

The two unidirectional coupled chaotic systems with time varying delay in Eqs. (11) and (12) are considered for synchronization. The observed chaotic time series synchronization error is shown in Fig. 17 for two types of coupling functions.

$$\dot{x} = -\alpha x + \mu \sin(x(t-\tau(t)) \tag{11}$$

$$\dot{y} = -\alpha y + \mu \sin(y(t-\tau(t)) + k(t)(x-y). \tag{12}$$



(a) Synchronization error for unidirectional square wave coupling. Vertical scale is 200 mV/div., horizontal scale is 2.00 μs/div.

(b) Synchronization error for unidirectional cosine function based coupling. Vertical scale is 200 mV/div., horizontal scale is 100 μs/div.

(c) Synchronization error for bidirectional square wave coupling. Vertical scale is 200 mV/div., horizontal scale is 200 μs/div.

(d) Synchronization error for bidirectional cosine function based coupling. Vertical scale is 200 mV/div., horizontal scale is 100 μs/div.

Fig. 12. Error signal, obtained using trigger functionality of a digital oscilloscope. We have not used any averaging or bandwidth limit on the channels, noise level is around 50 mV. Since the error signal is actually analog, we don't know what initial conditions are present at the D/A channel output. Hence we obtain arbitrary fluctuations in the analog signals.

4.4 Effect of parameter mismatch

The parameter mismatch plays an important role in synchronization. Any two systems, may be from the same wafer, must have some mismatch in terms of the parameters. It is one of the most effective tool to find out the robustness of the synchronization. Here we investigated the one dimensional correlation C, with respect to the relative mismatch parameter p. The robustness of the synchronization in presence of maximum 20% mismatch of two parameters μ, τ and their simultaneous mismatch are investigated. To observe the effect of this on synchronization, the cross correlation coefficient is determined for both unidirectional and bidirectional coupling. Figures 18 and 19 show the cross correlation measure C for a percentage of parameter mismatch P.

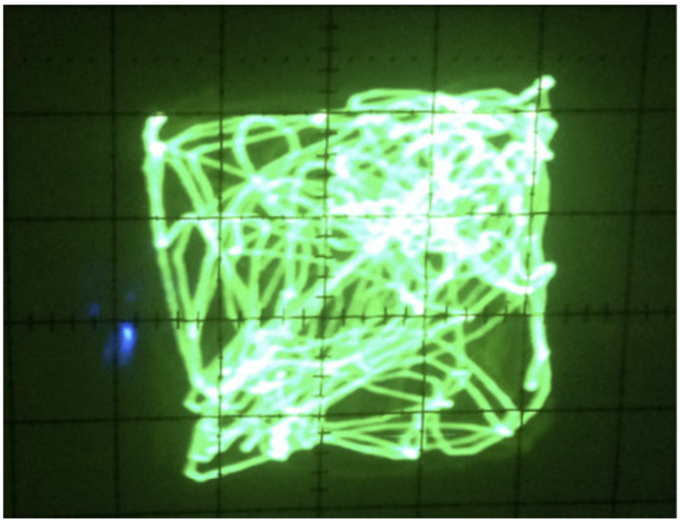
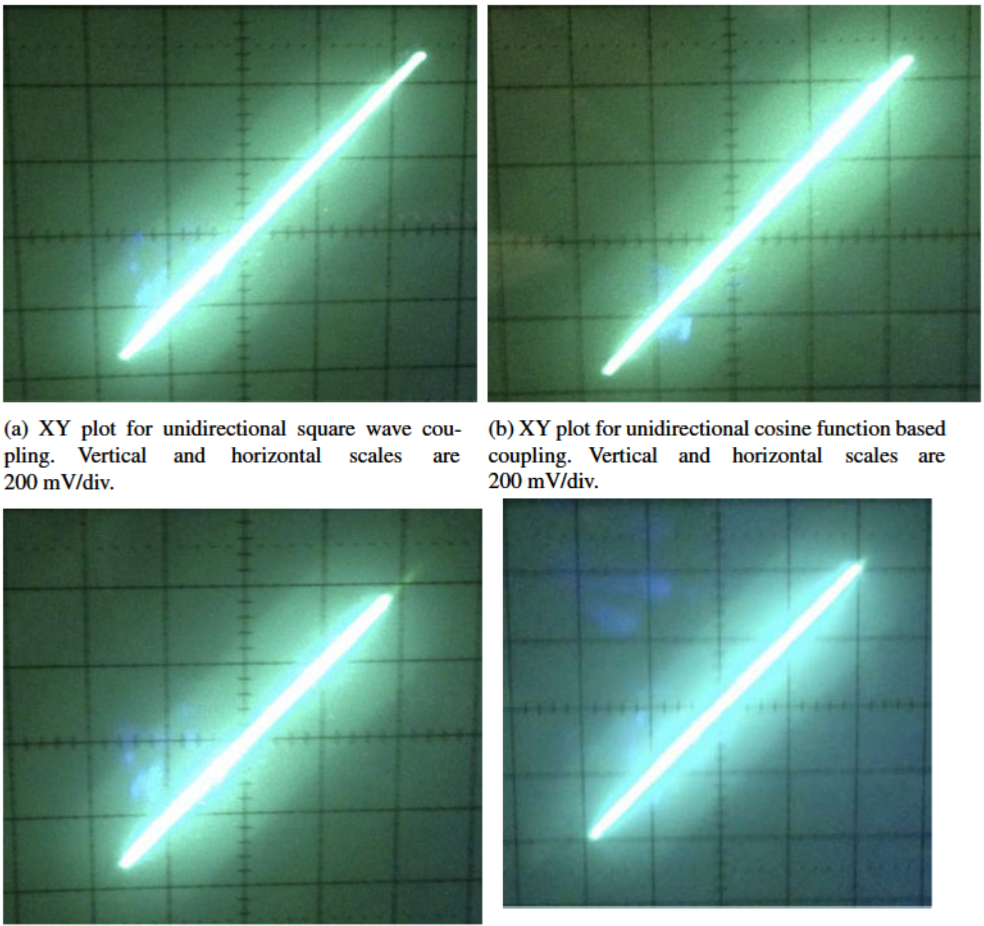


Fig. 13. The fact that the drive and response systems are not synchronized and the outputs are analog waveforms is an experimental evidence of “sensitive dependence on initial conditions”, one of the hallmarks of chaotic systems. Although the FPGA Ikeda DDE drive and response systems are both digital specifications, the output waveform is analog. Thus any noise on the analog line is going to ensure that the drive and response systems do not have the same initial conditions and this leads to the X-Y plot shown.

It is observed that bidirectional coupling exhibits better synchronization and for simultaneous parameter mismatch, synchronization decays slowly for unidirectional coupling.

4.5 Possible communication protocol

It has been accepted that the possibility of deploying a communication protocol based on synchronized chaotic systems does have potential. Moreover, deployment of such high dimensional on basic programming circuits such as the FPGA is highly desirable. This is because, the ability to re – produce high dimensional structures on a limited capacity “real – life” circuit increase the potential of such experimental results to be utilized by man. As demonstrated in previous sections of this article, our experiment methodology has realized the coupled Ikeda delay differential equation on the FPGA. Hence, an encoding (as opposed to encrypting – encoding is a terminology not related to ciphering information, rather encoding is a “particular language for a particular machine to communicate”) can be derived. Upon the setting up of two or more end points with FPGA’s with the coupled Ikeda delay differential equation programmed upon it, an efficient mechanism to relay information can be executed. After time t_k (i.e. threshold synchronization time), all data transmitted can be retrieved at all end points. Such communication infrastructure is relative similar with quantum information delivery mechanism setup (minus the security features enabled upon the quantum schemes). Nevertheless, by realizing the potential on FPGA’s, further research can be done upon the chaotic schemes in order for it to provide security relative to the quantum mechanisms. Achieving this goal would result in secure information transmission with synchronized chaotic systems mimicking the quantum setup (i.e. by having pre-assigned hardware’s at each end points) – but with much lower cost.



(a) XY plot for unidirectional square wave coupling. Vertical and horizontal scales are 200 mV/div.

(b) XY plot for unidirectional cosine function based coupling. Vertical and horizontal scales are 200 mV/div.

(c) XY plot for bidirectional square wave coupling. Vertical and horizontal scales are 200 mV/div.

(d) XY plot for bidirectional cosine function based coupling. Vertical and horizontal scales are 200 mV/div.

Fig. 14. The slope for all plots is one, implying synchronization.

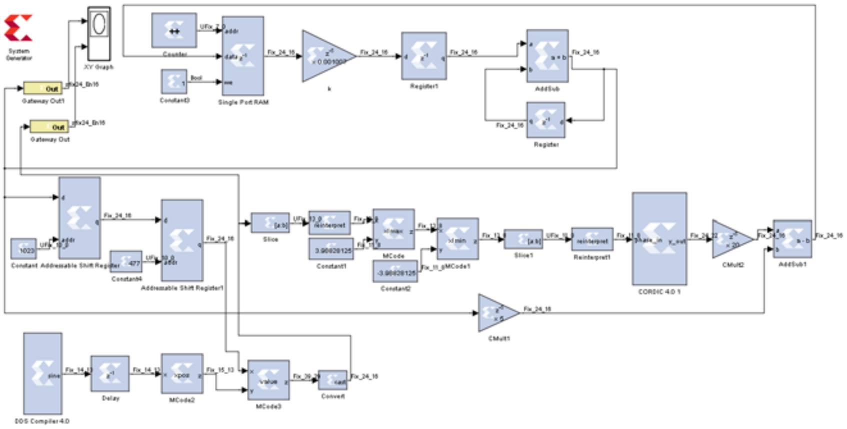


Fig. 15. Delay time modulated Ikeda model using Xilinx blockset.

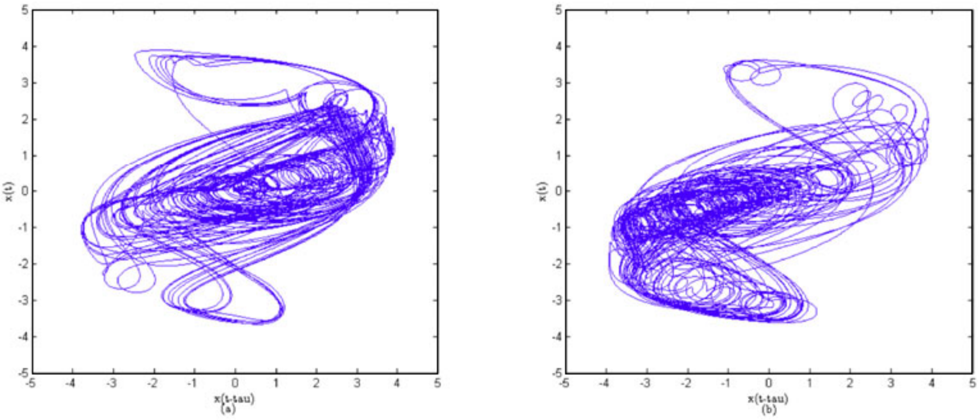


Fig. 16. Chaotic dynamics of DTM Ikeda model for (a) $\tau(t) = 1.5|\sin(t)|$ and (b) $\tau(t) = 1|\sin(t)|$.

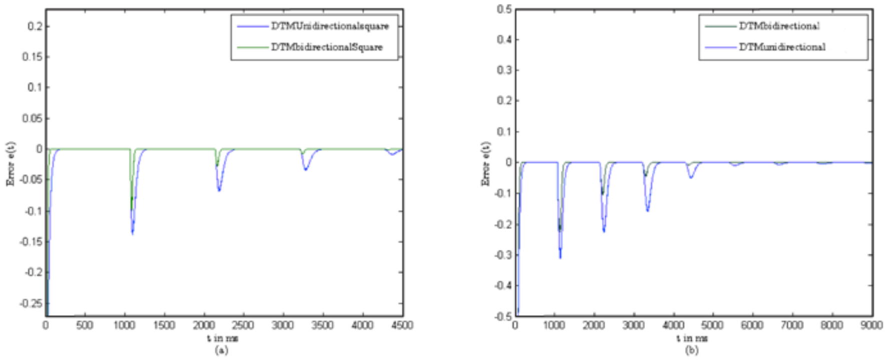


Fig. 17. Error dynamics of DTM Ikeda systems for delay $\tau(t) = 1|\sin(t)|$. (a) Square wave coupling. (b) Coupling function $K(t) = -\alpha + 2\mu|\cos(y(t-\tau))|$.

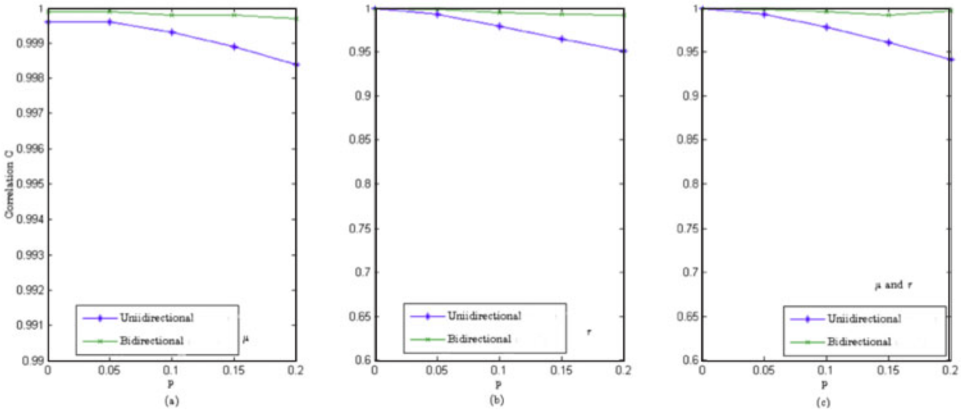


Fig. 18. Effect of parameter mismatch for square wave coupling function. Maximum correlation coefficient for (a) μ (b) τ (c) simultaneous mismatch of μ and τ.

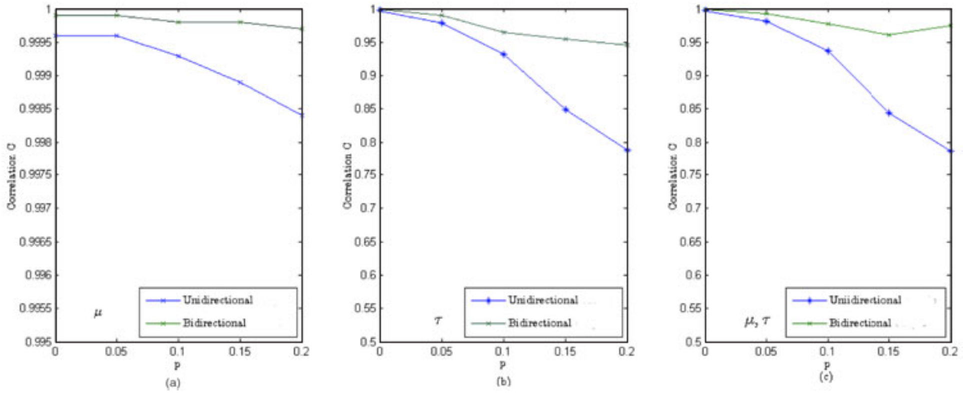


Fig. 19. Effect of parameter mismatch for coupling function $k(t) = -\alpha + 2\mu|\cos(y(t-\tau))|$. Maximum correlation coefficient for (a) μ (b) τ (c) simultaneous mismatch of μ and τ.

5 Conclusion

The FPGA has been proposed as a robust physical platform for realizing time delayed chaotic systems. Two approaches – hardware co-simulation and specification of the chaotic DDE on the FPGA – were illustrated.

For future work, from a mathematical standpoint, a rigorous framework for quantifying the effects of sampling frequency on the underlying chaotic behavior is necessary. This framework should also address the effects of the numerical algorithm used.

From the engineering standpoint, one should implement a timing-driven synthesis procedure that provides timing closure. That is, constraints for the clock frequencies and associated delay parameters are specified so the place and route tools can attempt to fit the design on the FPGA, such that our constraints are satisfied. Note that timing closure is more of an art that usually requires multiple iterations of the constraints before timing can be satisfied. DTM specification and parameter mismatch results via analog output are also ripe areas for future work. Moreover, in order to completely decouple the drive and response systems on FPGAs, one should utilize the LogicLock feature of synthesis tools.

Currently, we are investigating all the topics above but using the more "complex" Lang-Kobayashi (L-K) chaotic DDE [1].

Many thanks to Jeff Nigh and Greg Nash from Altera corporation for generous donation of hardware.

References

1. S. Banerjee, M. Banerjee, Opt. Comm. **285**, 2402 (2012)
2. P. Saha, S. Banerjee, A.R. Chowdhury, Chaos, Solitons Fractals **14**, 1083 (2002)
3. S. Banerjee, P. Saha, A.R. Chowdhury, Phys. Lett. A **291**, 103 (2001)
4. T. Banerjee, et al., Nonlinear Dyn. **70**, 721 (2012)
5. S. Boccalettia, et al., Phys. Rep. **366** (2002)
6. L. Glass, Nature **410** (2001)
7. L. Glass, et al., Math. Biosci. **90**, 111 (1988)
8. H.U. Voss, Phys. Rev. E **61** (2000)
9. K. Ikeda, K. Matsumoto, Physica D **29**, 223 (1987)
10. K. Ikeda, H. Daido, O. Akimoto, Phys. Rev. Lett. **709**, 45 (1980)

11. M. Lakshmanan, D.V. Senthilkumar, *Dynamics of Nonlinear Time-Delay Systems*, 1st edition (Springer Series in Synergetics, 2011)
12. M. Chen, K. Kurths, Phys. Rev. E **036212**, 76 (2007)
13. B. Muthuswamy, S. Banerjee, *A Route to Chaos Using Integrated Circuits: The FPGA Approach*, 1st edition (Springer Series in Emergence, Complexity and Computation, 2014)
14. S. Banerjee, Chaos synchronization and cryptography for secure communications: applications for encryption. IGI Global, USA (2011)
15. A. Namajunas, et al., Int. J. Bif. Chaos **7**, 957 (1997)
16. L.M. Pecora, T.L. Carroll, Phys. Rev. Lett. **821**, 64 (1990)
17. D.V. Senthilkumar, M. Lakshmanan, J. Kurths, Phys. Rev. E **035205(R)**, 74 (2006)
18. D.V. Senthilkumar, M. Lakshmanan, Phys. Rev. E **016211**, 71 (2005)
19. D.V. Senthilkumar, M. Lakshmanan, Int. Conf. Contr. Synchr. Dyn. Syst. **23**, 300 (2005)
20. J.C. Sprott, *Elegant Chaos, Algebraically Simple Chaotic Flows* (World Scientific, 2010), p. 221
21. D. Valli, et al., Chaotic Time Delay Systems and Field Programmable Gate Array Realization. International Symposium on Chaos, Complexity and Leadership (2012), p. 9
22. Digilent Corporation, "Atlys Spartan-6 Design Platform", available, online: http://www.digilentinc.com/Products/Detail.cfm?NavPath=2,400,836&Prod=ATLYS&CFID=2379218&CFTOKEN=48534838, Last accessed: November 19th (2013)
23. Terasic, "DE2-115 Development and education board", available, online: http://www.terasic.com.tw/_sub/de2-115/ Last accessed: November 19th (2013)
24. Xilinx Inc. "Spartan-6 FPGA Family", available, online: http://www.xilinx.com/products/silicon-devices/fpga/spartan-6/ Last accessed: November 19th (2013)
25. Altera Inc, "Cyclone IV FPGA Family: Lowest cost, Lowest Power, Integrated Transceivers", available, online: http://www.altera.com/devices/fpga/cyclone-iv/cyiv-index.jsp, last accessed: November 19th (2013)