ASPECTS REGARDING CHAOTIC MAPS HARDWARE IMPLEMENTATIONS

CRISTIAN-IULIAN RÎNCU1, VASILE-GABRIEL IANA2

Key words: Communications, Dynamical chaotic systems, Logistic map, Field programmable gate array (FPGA).

In the recent years, many new communication systems have been based on chaos theory. Chaotic systems have been widely used to design analog and digital systems that are used to improve security level or to achieve spread spectrum systems in communication networks. In this paper we present some aspects regarding the implementation of digital chaotic maps that are used to achieve dynamical chaotic system. We present the case of using field programmable logic areas to implement chaotic generators based on well-known digital chaotic maps that are used for designed the communication networks using dynamical systems. We show that is a major difference between the values obtained using Matlab and a FPGA simulator. We discuss also about the problems regarding the degradation of digital chaotic maps that are still present in real digital implementation using digital processors not only in the simulation.

1. INTRODUCTION

The fields of communications, generally speaking, and the section containing the mobile telephony in particular, have greatly expanded during the last time. We nowadays live in the context of the globalization of economy and of the continuously increasing development of multinational companies, which release high-tech products on the market, products that require a more and more specialized manlabour. Therefore, people feel the need to communicate, much stronger every day, the need to exchange ideas and pieces of information under all shapes.

Likewise, the necessity of making data transfers on a very short notice has become one of the world's major issues. Different system types, such as standard telephony, computer networks, television, and other systems have been integrated

1 Communications and military electronic systems Department, Military Technical Academy, 81–83 George Coşbuc blvd., 75275, Bucharest 5, Romania, E-mail: r_iulian@mta.ro
2 University of Piteşti, str. Targu din Vale, no.1, 10040, Piteşti, Romania, fax: +40 248 222949 phone: +(40) 745 90 50 49, E-mail: gbiana2000@yahoo.com

in a sole universal system. The importance of the Internet has increased with the speed of light, as well as the importance of all related applications. This is why companies have felt the need to subordinate all these systems to the client's more and more complex and sophisticated requirements. All these changes have also played an important part in the development witnessed in the field of communications.

The prominent development of digital communications is a exceedingly important strategically priority, both for a modern economy and for a country's national security. At present, digital communications are especially based on the linear theory and technology. The linear devices and circuits have been and still are very successful in the digital networks and communication systems. However, the conventional linear technologies, used in communications, appear as limited. For instance, the functioning of amplifiers in linear conditions requires inefficient polarization techniques, which limit the activity of the devices only in a field reduced from their entire dynamical series [1].

Nowadays, low energy consumption circuits and communication systems, with a great transmission capacity, increased flexibility and security, at a low production price, are regular techno-economic desiderata.

The scientific research of the past decade has proved the possibility of creating digital communication systems based on the non-linear functioning system of the electronic devices and circuits. The activity of the latter has been labeled as chaotic, because there was sensed an intense sensitivity of their activity, given the incremental variations of the initial conditions and their parameters [2]. The advantages offered by the devices, circuits and non-linear communication systems, may they be virtual or already made, refer to: an increased efficiency, reduced dimensions and weight, a greater capacity of functioning for the informational channels, the possibility of creating a larger number of informational channels and of using the devices at higher power levels, a lower cost and a more unlikely contingency of being intercepted [1].

![Fig. 1 – Chaos-based communication system.](image)

The general form for a simple communication system that is based on chaos theory can be considered like in Fig. 1. An information signal is applied to the transmitter system, which produces a chaotic output signal. This signal contains the
information, but it is invisible to the observer who has no knowledge about the parameters and initial conditions of the transmitter system. The right receiver, who possesses this knowledge, then reproduces the input signal or at least an approximation of it [3].

Over past ten years, new chaos-based spreading techniques have been developed: chaotic masking, chaotic modulation, chaos shift keying (CSK) and predictive Poincare control (or symbolic) modulation. The methods mentioned above represent techniques that are used to implement communication networks in order to protect the information and also to use efficient the available spectrum.

2. DIGITAL IMPLEMENTATION OF THE LOGISTIC MAP

The generators that have a chaotic behavior are the kernel of all the communication systems designed based on chaos theory. There has been significant interest in recent years in exploiting chaotic dynamics for these generators in communications, because chaotic processes exhibit several natural characteristics beneficial to communications systems regarding information security. Two key features of chaos include a noise-like time series, termed deterministically random motion, and a sensitive dependence on initial conditions, known colloquially as the butterfly effect.

In order to obtain a chaotic generator on a FPGA (Field-Programmable Gate Array), we have chosen a well-known one dimensional chaotic map. This digital function is the logistic map that was chosen because its properties are good to prove the utility of chaotic maps in protection of communication and in the same time because its simple form makes it easy to implement.

The logistic map is defined by a hyperbolic characteristic, and is described by the following equation:

$$x[n+1] = k \cdot x[n] \cdot (1 - x[n]).$$ (1)

$k$ is the parameter of logistic map that determines the chaotic behavior, with $x \in (0, 1)$ and $k \in (3,57; 4]$ for good chaotic properties.

The hardware description for the implementation of the chaotic generator based on the logistic equation is made using VHDL (Very High Speed Integrated Circuit Hardware Description Language) [4]. The hardware and software design have a close relationship because the specification method software organization and algorithm design can be directly implemented in hardware. This is possible because VHDL contains all the elements for algorithm description common in programming languages.

The implementation model entity for the chaotic generator is presented in Fig. 2. The input ports are reset and clk.
The output for the chaotic generator uses 8, 16 or 32 bits data buses. The update is performing at the rising edge of the $clk$ signal. The $reset$ signal initializes the internal registers of the generator with 0 and the output value with $x_0=0.34$. The input data have fixed point representation with 1:31 format. Only fractional numbers can be represented with the most significant bit used for sign representation. For data acquisition at maximum frequency, the program has instruction blocs implemented in parallel hardware structures. This structures work independently trough separate processes. The data is sampled and processed only when on specific input ports are detected changes. These also have one extra advantage, that the structures are synchronous and give the possibility for data processing right after it is available. The clock signal is used for synchronization with systems external to the FPGA structures.

The schematic block of the implementation model in FPGA hardware for the chaotic generator is presented in Fig. 3. The U2 performs the multiplication with a constant for the output signal $k$ (numerical value). The U3 block gives the binary two complement for the subtraction operation $(1-x[n])$ so that in the end using the U1 block results ($x[n+1] = k x[n] (1-x[n])$). These three blocks are implemented asynchronously. The U2 is used for obtaining a single clock delay, synchronously.
with the \( clk \) signal. These blocks are concurrent processes. They are implemented in hardware structures.

For functional circuit verification the test model was designed. This is an integrated environment where the projects (also called unit used for testing – UUT) are verified by applying stimulus signals and monitoring the systems responses.

For functional circuit verification the test model was designed. This is an integrated environment where the projects (also called unit used for testing – UUT) are verified by applying stimulus signals and monitoring the systems responses.

In other words, the simulation model substitutes the environment where the design generator will function so that we can observe and analyze its behavior. Project testing is performing in two steps. First, the project initialization is verified by applying stimulus using the test simulation module. As long as \( \text{reset} \) signal is ‘1’, the output \( \text{out} \_\text{port} \) is 0 and doesn’t depend on \( clk \) signal. The reset is synchronously with the clock signal. The first data from the output signal (a numeric sample) is obtained of the first rising edge of the \( clk \) signal.

In the second step is performed the functional verification of the whole project. The next figures present the graphics behavior of the logistic map for other two \( k \) values and the same value for \( x_0 \).

The obtained values for the logistic map implemented on 32 bits are presented in the Fig. 7.

At the implementation of the chaotic generator using FPGA 2S200PQ208-5 were used: 923 slices (two slices is a CLB – configurable logic block) that represent 39%, 32 D bitable, 1317 LUT (look up table) that represent 27% and a clock signal of 4 disposable on the device.
Fig. 6 – Logistic map simulation for $k = 3.50$.

The maximum frequency estimate that can be applied on clock signal is 24.026 MHz, with a period of 41.621 ns where 23.031 ns (55.3%) for combinational logics and 18.590 ns (44.7%) internal signals.

3. COMPARATIVE ANALYSIS BETWEEN MATLAB AND DIGITAL IMPLEMENTATION (VHDL)

At our testing of logistic map using Matlab and VHDL we have observed that are differences between the values obtained in both situations. To show the inequalities between the values obtained for the studied chaotic map using Matlab simulations and the values obtained using a FPGA simulator program we consider for digital chaotic map representation on 8, 16 and 32 bits.
For logistic map we have considered the initial condition $x_0 = 0.34$ and the value of the parameter $k = 3.996$.

![Fig. 8 – Differences between Matlab and VHDL values (on 32 bits).](image)

Although we have used the same representation on bits for the values of the logistic map, we can see that the differences are very important beginning with the 30th iteration of the logistic map. When we have compared the values obtained on bits we had seen that it is a small difference on bits for the first values, and these different bits are propagated from the LSB to MSB that bring to a big difference for a significant number of iteration. We have obtained even values of 0.99 that is quite big for the domain function that is (0, 1).

![Fig. 9 – Differences between Matlab and VHDL values (on 16 bits).](image)
Figure 10 is also illustrative for the inequality between the values even we use for representation only 8 bits and the number of possible values are limited. We can see also that it is maintained the disadvantage that appear when the chaotic map are implemented in digital form represented of the dynamical degradation of digital chaotic maps. The number of the values that are taken by the logistic map is equal to 4 (with one more than in Matlab case) even the possible distinct values are 255. For this reason is clear that we need to use perturbing algorithms [5] when we want to implement chaotic maps using digital devices.

4. CONCLUSIONS

The communication systems that use chaos theory have exceeded the stage of the laboratory simulations using Matlab or another designing tool and they have become the physical implementation step with component subsystems and certain applications. This paper has pointed out the moment of the introduction for some new technologies and has presented some hardware and software achievements and a comparison between these two methods. We have presented some aspects regarding the digital implementation of one well-known digital chaotic map that can be used in communication systems to develop signal generators or in cryptosystems. We have shown that there are differences between the values obtained using the FPGA and the others obtained using Matlab. Although we have used in both situations the same number of bits for the number representation, the difference is increasing and it is quite big for a number of iteration more than 50.

We have shown that the behavior of the digital implemented maps is also chaotic like in the Matlab simulation although the values are not the same. For this
reason the digital implemented chaotic maps can be used in digital systems that use the properties of the dynamical systems.

We have underlined for digital implementation using FPGA, a very important problem that exists also in the results obtained using Matlab simulations, the periodicity of the chaotic maps when are design in digital mode. If we need to design dynamical chaotic system using digital processors it will be necessary to use a precision big enough or, when this thing is not possible there will be used methods for perturbing the digital chaotic maps. The perturbing algorithms can assure good statistical properties for the binary sequence that will be obtained.

A secure data transfer using few resources can be made, with performances easy to improve. An important factor in data transfer is the speed that is closely connected to the speed of the chaotic generator. A higher throughput of the chaotic generator can be obtained by using dedicated circuits (ASICs), field programmable logic arrays (FPGAs) or faster general purpose processors.

ACKNOWLEDGEMENT

This work was supported by The National University Research Council, under grant Chaos-Based Cryptosystems For The Information Security In Communication Networks.

Received on 16 July 2006

REFERENCES