A VLSI IMPLEMENTATION OF A LINEAR CELLULAR AUTOMATA TEST-BED

Flávio du Pin Calmon, Felipe Miranda Costa, A. C. A. Nascimento and A. R. S. Romariz

Departamento de Engenharia Elétrica, Universidade de Brasília, CP 4386, Brasília, DF 70904-970, Brazil

ABSTRACT

This paper presents a preliminary prototype hardware implementation of a one-dimensional homogeneous circle Cellular Automaton (CA) system for generating CA-based stream ciphers. The VLSI model was prototyped using the Hardware Description Language (HDL) VHDL and synthesized to a *Xilinx XC3S1000* FPGA. The hardware gives the possibility of choosing the rule of the CA, being able evolve over 3500 cells simultaneously.

1. INTRODUCTION

Since the 1980's, theory of CA has been studied, with applications, for example, in pseudorandom number generation, as reported in [4], and in cryptography, as in [1],[2],[3]. A 1-dimensional Cellular Automaton consists of *n* connected cells which evolve by a simple local rule. As mentioned in [2], one-dimensional CA cells can be displayed in a circle (Circle CA), i.e., the border cells are neighbors to each other. One-dimensional CAs can also have their cells displayed in a line (noncircle CA), meaning that the boundary cell's neighbors are imaginary cells with a constant 0 state.

Each cell's next state depends only on its present state and on its right and left neighbors' state, being determined by a local rule. Consequently, 2^3 possibilities of layouts can appear, and the one-dimensional CA rule can be represented by an 8-bit number (shown in figure 1 as question marks). Usually, the decimal representation of the rule is given. For example, rule 30 means that the values 00011110 would be substituted in the place of the question marks in figure 1. If the rules of all cells in one CA are the same, it is called a "homogeneous" CA; otherwise, it is called "nonhomogeneous".

For the Cellular Automata hardware implementation, Field Programmable Gate Arrays (FPGAs) were used. FPGAs are reconfigurable devices that provide fine-grained logic and interconnections elements whose functions and structures can be programmed to suit a certain application defined by the user [5]. It has been shown [6] that these devices can accelerate certain computations when compared to traditional software implementations, especially when a great deal of parallelism is required. In this work, we present a preliminary FPGA-based test-bed for studying stream-ciphers generated with one-dimensional Cellular Automata. Because of the enormous amount of parallel computation involved with the CA structure, the FPGA implementation results in a larger absolute performance when compared to software based alternatives.

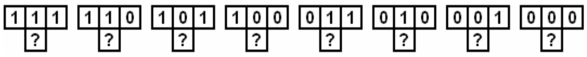

Figure 1 – CA evolution rule. The question marks represent the cells' next state.

This paper is divided in three parts: initially, details about the hardware implementation are presented. The second section presents the simulations and results obtained. Finally, the conclusions are presented.

2. HARDWARE IMPLEMENTATION

The structures used for implementing the FPGA based test-bed were described using VHDL. Initially, we developed a simple cell with a configurable rule and a two-cell neighborhood. In this structure, a 1-bit register stores the cell's current state, which is also presented on a 1-bit output. The evolution rule is determined by an 8-bit input, following the convention presented in the introduction. Two 1-bit inputs (left and right) are used to indicate the values of the neighboring cells.

The cell's next state is found based on a simple look-up procedure, where the 3-bit number formed by the left input, the current state and the right input indicate a bit position of the rule, whose value will determine the following state. The cell's operation is synchronized with an external clock and a reset input reinitializes the cell's current state. The block diagram of the cell is shown in figure 2.

After we created a simple cell structure, a larger entity was described for efficiently connecting together a large number of cells, forming the test-bed's CA core. A periodic boundary condition was used, i.e., the extreme cells of the one-dimensional CA are adjacent (Circle CA). This larger entity also contains a 8-bit register for storing the evolution rule, which can be set externally through a 8-bit input. Along with the rule input, a master reset and a clock input were also included,

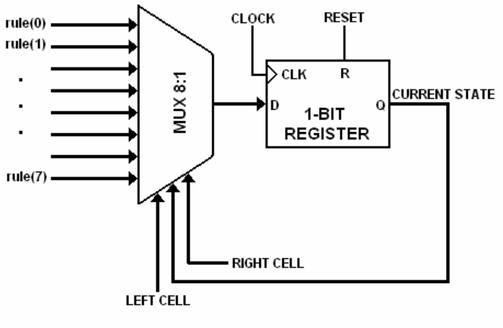

Figure 2 – Block diagram of a cell.

distributing the clock and master reset signals throughout the cells.

For the initial tests, the evolution of a single cell was verified, with the CA core having, consequently, a single output. The number of outputs, however, can be easily changed, resulting in sampling a larger number of cell states. This measure was taken due to the lack of fast I/O resources of the development board used. The structure of the CA core is shown in figure 3.

3. SIMULATION, PROTOTYPING AND RESULTS

The simulations of the simple cell and the CA core were realized with the software *ModelSim XE III 6.0a*, validating the VHDL structures. All the synthesis was done using *Xilinx ISE 7.1.03i* software for a *Xilinx XC3S1000* FPGA.

For the preliminary on-board prototype we used the *Digilent Spartan-3 Starter Kit* board. Other VHDL structures were created to allow I/O interfaces, such as 7-segment converters, switch debouncers and a clock divider. For demonstration purposes and due to the lack of a fast I/O port on the development board used, the CA evolution starts from a single black cell. More sophisticated ways of setting the CA's initial state are being developed. This cell's current state is shown on a 7-segment display, using a low-frequency clock for the CA's evolution. The rule is determined by eight external switches, with the selected value being displayed in hexadecimal form on two 7-segment displays. A reset switch was created along with a led indicating the current reset status.

Although the initial prototype was created only for demonstration purposes, we were able to successfully synthesize and use a CA composed of 3500 cells, along with other structures used for interface purposes. The synthesis results, presented in Table 1, show that a clock of over 145 MHz may be used. This result clearly indicates that the FPGA based test-bed is an efficient platform for testing CA structures, having an enourmous speed gain when compared to software simulations.

TABLE 1 - RESULT OF THE PROTOTYPE SYNTHESIS FOR A FPGA XILINX XC3S1000

Parameter	Result
Number of *Slices*	99%
Number of *Slice Flip Flops*	29%
Number of 4 *input LUTs*	91%
Number of *IOBs*	13%
Number of *GCLKs*	25%
Maximum operation frequency	145.9 MHz

4. CONCLUSION

We were able to successfully implement in VHDL and prototype on hardware a preliminary version of an FPGA based test-bed for generating and realizing cryptanalysis of Cellular Automata based stream ciphers. The synthesis results for a *Xilinx XC3S1000* FPGA indicate the high performance gain of the hardware simulation of the CA when compared to the software simulations. Currently, a fully programmable test-bed with a high-speed I/O port is being developed.

4. REFERENCES

[1] S. Nandi, B. Kar, and P. Chaudhuri, "Theory and Applications of Cellular Automata in Cryptography," *IEEE Trans. Computers*, vol. 43, no. 12, pp. 1346-1357, Dec. 1994.

[2] Feng Bao, "Cryptanalysis of a Partially Known Cellular Automata Cryptosystem," *IEEE Ttrans. Computers*, vol. 53, no. 11, pp. 1493-1497, Nov. 2004.

[3] S. Wolfram, *A New Kind of Science*. Wolfram Media Inc., 2002. 1197 p.

[4] P. D. Hortensius, R. D. McLeod, W. Pries, D. M. Miller, and H. C. Card, "Cellular Automata-Based Pseudorandom Number Generators for Built-In Self-Test," *IEEE Trans. Computer-Aided Design*, vol. 8, pp. 842--859, Aug. 1989.

[5] R. Vemuri, R. Harr, "Configurable Computing: Technology and Applications," *IEEE Computer Magazine*, vol. 33, no. 4, pp. 39–40, Apr. 2000.

[6] A. de Hon, "The Density Advantage of Configurable Computing," *IEEE Computer Magazine*, vol. 33, no. 4, pp. 41–49, Apr. 2000.

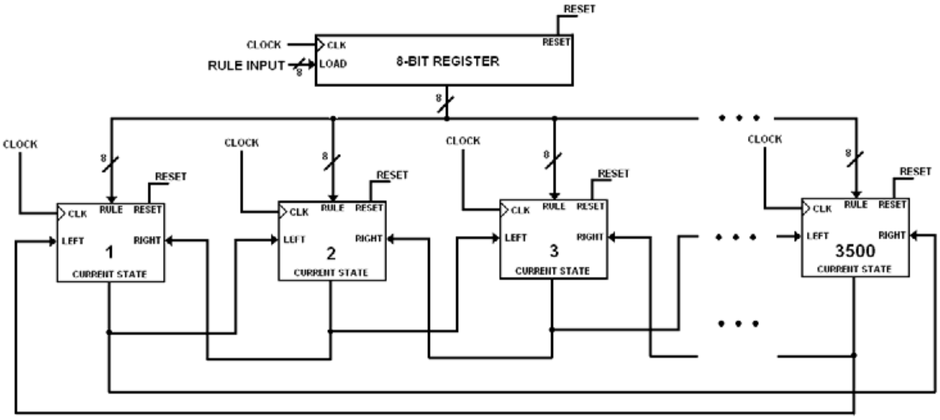

Figure 3 – Block diagram the CA core. The numbered structures represent the cells (1 to 3500).