UNIVERSITY OF PITESTI

Faculty of Electronics, Communications and Computers

Str. Targu din Vale, No. 1, 110040 - Pitesti, ROMANIA
Tel./Fax. : 0348 – 453 217, WEB: http://www.upit.ro

Hardware Implementation of Programmable Cellular Automata Encryption Algorithm

Author: *Petre ANGHELESCU*
petre.anghelescu@upit.ro

Abstract/Overview Summary

The main objective of the paper/project is the design, the implementation and the analysis of a programmable cellular automata (PCA) with properties that make it suitable for use in symmetric key cryptography. Based on PCAs state transitions certain fundamental transformations are defined which represents block ciphering functions of the proposed enciphering scheme. In order to verify the proposed encryption algorithm, an experimental hardware platform based on a reconfigurable FPGA of type Spartan 3E XC3S500E was used. The enciphering scheme provides high speed, good security and it is ideally suit for hardware implementation in FPGA devices.

1. CAs vs. PCAs

CAs represents a particular class of dynamical systems that enable to describe the evolution of **complex systems** with **simple rules**, without using partial differential equations. A CA consists of a regular uniform n-dimensional **array of cells** where every cell can take values either 0 or 1 and **evolve** in **discrete time steps**.

PCAs are modified CA structures, where the combinational logic of each cell is not fixed but controlled by a number of **control signals** such that **different functions** (evolution rules) can be **realized/implemented** on the **same structure**.

2. PCA based encryption algorithm

The encryption method proposed in this paper is based on the fact that the CA from class II (*dynamical systems*) exhibit periodic behaviour (each state lies in some cycles). In these cases, their evolution depends essentially of the initial state, but we can say that after a while the initial state is "forgotten", in sense that the state cannot be retrievable through analyses of the current configuration.

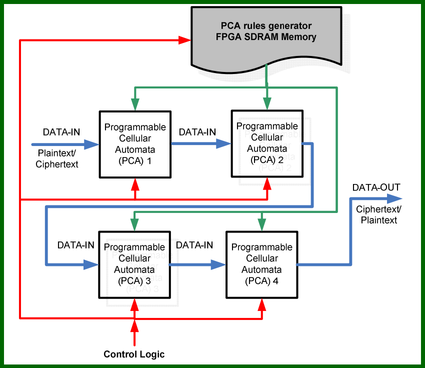
The proposed encryption system it is realised using a combination of 4 PCA with rules **51, 60 and 102 arranged in pipeline**. The PCA evolution rules are stored in a FPGA SDRAM memory (provide real-time keys for the cipher).

Rules	7 111	6 110	5 101	4 100	3 011	2 010	1 001	0 000
51	0	0	1	1	0	0	1	1
60	0	0	1	1	1	1	0	0
102	0	1	1	0	0	1	1	0
	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0

$a_i(t+1)=\overline{a_i(t)}$ *Rule 51*

$a_i(t+1)=a_i(t)\oplus a_{i-1}(t)$ *Rule 60*

$a_i(t+1)=a_i(t)\oplus a_{i+1}(t)$ *Rule 102*


PCA rules generator
FPGA SDRAM Memory
DATA-IN
Plaintext/
Ciphertext
Programmable Cellular Automata (PCA) 1
DATA-IN
Programmable Cellular Automata (PCA) 2
DATA-OUT
Ciphertext/
Plaintext
DATA-IN
Programmable Cellular Automata (PCA) 3
DATA-IN
Programmable Cellular Automata (PCA) 4
Control Logic

The block cipher (decipher) procedure can be defined as follows:

1. Load the PCA1 with one byte plaintext (ciphertext) from I/O. The initial block of the message is the initial state of the PCA1. The global configuration of the PCA4 represents the encrypted (decrypted) message.

2. Load a rule configuration control word from memory rules into the PCA1 … PCA4.

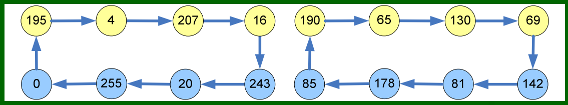
3. Run the PCA (1, 2, 3 and 4) for 1 … 7 cycles.

4. Repeat steps 2 and 3 for four times.

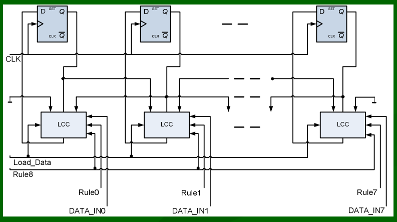
5. Send one byte ciphertext (plaintext) to I/O (from the PCA4). If not end of the plaintext (ciphertext) go to step 1. Otherwise, stop the process.

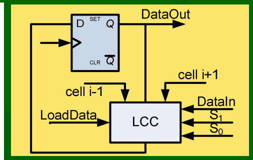
3. PCA evolution rules (51, 60 and 102)

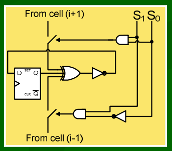
The PCA configured with the rules 51, 60 and 102 has a state-transition diagram that consists of equal cycles of even length. As an example, 8-cell PCA with rule configuration <51, 51, 60, 60, 60, 60, 51 and 51> generates cycles as depicted in below figure.



According with the CA theory, a single basic PCA cell was designed. Then, 8 cells are connected together to compose an 8-cell PCA (in this project we have 32 cells).

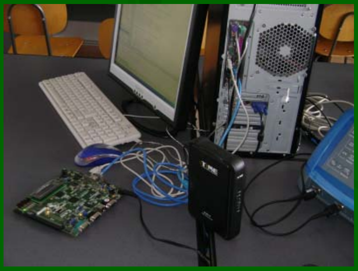


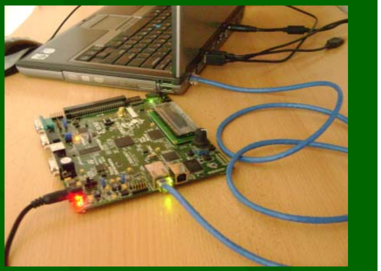




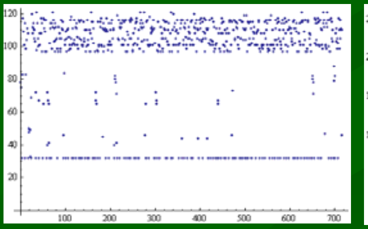
S_1	S_0	Rules applied
0	0	51
0	1	51
1	0	60
1	1	102

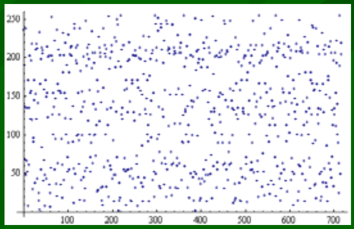
4. Testing and Results





As we depict in the next two figures, the distribution of the encrypted text is uniform in all intervals, i.e. the encrypted text is distributed almost uniformly in all ASCII intervals and not only in alphanumeric intervals.





CONCLUSIONS

In the paper we have reported results of the study on applying PCAs to the secret-key cryptography. As PCAs achieves high parallelism and only local interactions, the proposed PCA based cryptosystem is well suited for high-speed secure network communications and real time applications. Also, the encryption and decryption devices share the same module, and could be implemented efficiently in hardware, in FPGA devices, due to simple structure of PCA.

In the immediate future, the PCA encryption project will be implemented using both communication protocols: TCP/IP and UDP(for increased transmission safety), larger storage memories (for higher speed) and more flexible parameters for system initialization.

This work was supported by CNCS UEFISCDI, project number PN II-RU PD 369, Contract No. 10/02.08.2010.