IOSR Journal of VLSI and Signal Processing (IOSR-JVSP)
Volume 4, Issue 6, Ver. I (Nov - Dec. 2014), PP 28-33
e-ISSN: 2319 – 4200, p-ISSN No. : 2319 – 4197
www.iosrjournals.org

Optimization of 1D and 2D Cellular Automata for Pseudo Random Number Generator.

P.Sudhakar[1], B.Chinnarao[2], Dr. M. Madhavi Latha [3]

[1]*(Department of E.C.E, Aditya Institute of Technology And Management,Tekkali, Srikakulam, AP, India)*
[2]*(Department of E.C.E, Aditya Institute of Technology And Management,Tekkali, Srikakulam, AP, India)*
[3]*(Department of E.C.E, JNTUH College of Engineering Kukatpally, Hyderabad, Telangana, India)*

Abstract: *In this paper we have implemented 1D binary cellular automata with wrap around at the edges (i.e. a ring). The default update rule used is rule 30 discovered by Stephen Wolfram. Rule 30 is an update rule that when applied to the CA will produce a class III, a periodic, chaotic behavior. The response with respect to rule 2 and rule 90 is also verified on Xilinx Spartan 3E FPGA and this can be applied for modeling PRNG. This paper also discusses the correlation between 1D and 2D cellular automata.*

Cellular automata concept was first introduced by von Neumann von Neumann for the proposal of modeling biological self-reproduction. The primary interest was to derive a computationally universal cellular space with self-reproduction configurations. Afterward, a new phase of activities was started by Wolfram who pioneered the investigation of Cellular automata as a mathematical model for self-organizing statistical systems. Wolfram was proved that the randomness of the patterns generated by maximum-length Cellular automata is significantly better than other widely used methods, such as linear feedback shift registers. The intensive interest in this field can be attributed to the phenomenal growth of the VLSI technology that permits cost-effective realization of the simple structure of local-neighborhood Cellular automata Wolfram.

Keywords: *Cellular automata, chaotic, FPGA, PRNG-pseudo random number generator.*

I. Introduction

The present day communication systems require low power consumption and low complexity of their implementation solutions. Applications like remote transmission, surveillance, sensors networks for different kind of systems, are very good candidates for an efficient hardware realization. The study of Cellular automata (CA) dates back to John von Neumann in the early 50's. Neumann [1] framed CA as a cellular space capable of self reproduction. Since then, many researchers have taken interest in the study of CA for modeling the behavior of complex system. Wolfram [2] studied one dimensional CA with the help of polynomial algebra. Pries et al. [3] studied one dimensional CA exhibiting group properties based on a similar kind of polynomial algebra. Later Das [4] extended the characterization of one dimensional CA with the help of Matrix Algebra. Many applications of one dimensional CA [9-17] have been reported. On the other hand, 2D CA is not yet a well studied area. Packard [5] reported some empirical studies on 2D CA depending on five neighborhoods CA. Chowdhury et.al. [6] Extended one dimensional CA built around matrix algebra for characterization of 2D CA. However, emphasis was laid on special class of additive 2D CA, known as Restricted Vertical Neighborhood (RVN) CA. In this class of 2D CA, the vertical dependency of a site is restricted to either the sites on its top or bottom, but not both. Khan et. al. [7] studied the 9 neighborhood 2D CA. He developed the basic mathematical model to study the entire nearest neighborhood 2D CA and presented a general framework for state transformation. Nayak [8] used color graphs to model 2D CA linear rules. Algebraic Matrix formulae for few 2D CA have been studied in [18]. Modeling techniques for fundamental image transformations have been studied in [19]. An analytical frame work to study a restricted class of 2D CA has been reported in [20]. The concept of CA PRNG is a fast, compact pattern generator capable of providing user selectable patterns at very high speed. The CA PRNG module is suitable for FPGA-accelerated verification, on-chip testing as well as for applications that needs random patterns or specific sets of patterns generated.

II. One-Dimension CA

The CA structure investigated by Wolfram can be viewed as discrete lattice of sites (cells) where each cell can assume either the value 0 or 1. The next state of a cell is assumed to depend on itself and on its two neighboring cells for a 3 neighborhood dependency. The cells evolve in discrete time steps according to some deterministic rule that depends only on local neighborhood. In effect, each cell as shown in Fig 1, consists of a storage element (D- Flip Flop) and a combinational logic implementing the next state.

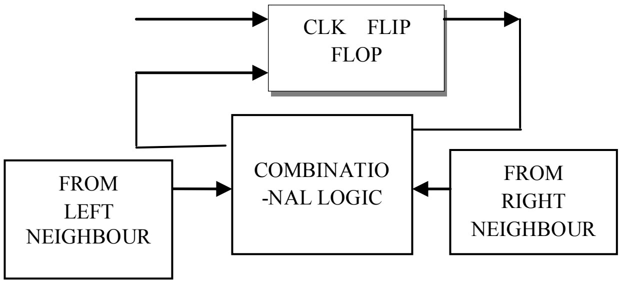


Fig 1: Basic CA Block

If the next state function of a cell is expressed in the form of a truth table, then the decimal equivalent of the output is conventionally called the rule number for the cell. Thus for a 3 neighborhood CA, the next state function for cell i is represented as follows for each of 256 rules. From table 1 the top rows gives all the possible states of the neighborhood cells at the time instant (t), while the 2nd and 3rd rows give the corresponding states of the ith cell at the time instant (t+1) for two illustrative CA rules. The Second Row taken as binary number and converted into decimal representation is the rule no. 90. Similarly, the third row corresponds to rule no. 2. The expression for a rule can be obtained from its truth table. The minimized expression for rule 90 & rule 2 is

Neighbor State	**111**	110	101	100	011	010	001	000	RULE
Next State	**0**	**1**	**0**	**1**	**1**	**0**	**1**	**0**	**90**
Next State	**1**	**0**	**0**	**1**	**0**	**1**	**1**	**0**	**2**

Table 1: 1D CA rule example

q_i (t+1)=$q_{i-1}(t) \oplus q_{t+1}(t)$

q_i (t+1)=$q_{i-1}(t) \oplus q_i(t) \oplus q_{t+1}(t)$

Definitions:

1. If same rule is applied to all the cells in a CA, then the CA is said to be Uniform or Regular CA.
2. If different rules are applied to different cells in a CA, then the CA is said to be Hybrid CA.
3. The CA is said to be a Periodic CA if the extreme cells are adjacent to each other.
4. The CA is said to be a Null boundary CA if the extreme cells are connected to logic 0-state.
5. If in CA the neighborhood dependence is on XOR or XNOR only, then the CA is called additive CA, specifically, a linear CA employs XOR rules only.
6. A CA whose transformation is invertible (i.e. all the states in the state transition diagram lie in some cycle) is called a Group CA, otherwise it is called a Non Group CA.

2D Cellular Automata:

In 2D cellular automata the cells are arranged in two dimensional grids with connections among the neighborhood cells. The state of CA at any time instant can be represented by an *m* x *n* binary matrix. The neighborhood function specifying the next state of a particular cell of the CA is affected by the current state of itself and eight cells in its nearest neighborhood. Mathematically, the next state q of the (i,j)th cell of a 2D CA is given by

q_{ij} (t+1)=$f[q_{i-1,j-1}(t) \oplus q_{i-1,j}(t), q_{i-1,j+1}(t), q_{i,j-1}(t), q_{i,j+1}(t), q_{i+1,j-1}(t), q_{i+1,j}(t)'_{q}{}_{\substack{i+1,\\ j+1(t)}}$]

Where f is the Boolean function of 9 variables. To express a transition rule of 2D CA, a specific rule convention proposed in [17] is noted below.

64	128	256
32	1	2
16	8	4

The Central box represents the current cell (that is the cell being considered) and all other boxes represent the eight nearest neighbors of that cell. The number within each box represents the rule number associated with that particular neighbor of the current cell – that is, if the next state of a cell is dependent only on its present state, it is referred to as rule 1. If the next state depends on the present state of itself and its right neighbor, it is referred to as rule 3(=1+2). If the next state depends on the present state of itself and its right, bottom, left, and top neighbors, it is referred to as rule 171 (=1+2+8+32+128) and so on.

III. Mathematical Model

This 2D CA behavior can be analyzed with the help of an elegant mathematical model where we use two fundamental matrices to obtain row and column dependencies of the cells. Let the two dimensional binary information matrix be denoted as X_t that represents the current state of a 2D CA configured with a specific rule. The next state of any cell will be obtained by XOR operation of the states of its relevant neighbors associated with the rule. The global transformation associated with different rules can be made effective with following fundamental matrices referred to as T1 and T2 in the rest of the paper.

$$T1=\begin{pmatrix} \mathbf{0} & \mathbf{1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{1} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{pmatrix} \qquad T2=\begin{pmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{1} & \mathbf{0} \end{pmatrix}$$

The following lemma specifies the value of the next state of a 2D CA referred to as X_{t+1} given that its current state is X_t. The CA is assumed to be configured with primary rule only—that is it depends on only one of its nine neighbors.

The next state transition of all the primary rules (1, 2, 4, 8, 16, 32, 64,128, 256) with null boundary condition, can be represented as

Rule 1 $\longrightarrow$ $[\boldsymbol{X_{t+1}}] = [\boldsymbol{X_t}]$
Rule 2 $\longrightarrow$ $[\boldsymbol{X_{t+1}}] = [\boldsymbol{X_t}][\boldsymbol{T_2}]$
Rule 4 $\longrightarrow$ $[\boldsymbol{X_{t+1}}] = [\boldsymbol{T_1}][\boldsymbol{X_t}][\boldsymbol{T_2}]$
Rule 8 $\longrightarrow$ $[\boldsymbol{X_{t+1}}] = [\boldsymbol{T_1}][\boldsymbol{X_t}]$
Rule 16 $\longrightarrow$ $[\boldsymbol{X_{t+1}}] = [\boldsymbol{T_1}][\boldsymbol{X_t}][\boldsymbol{T_1}]$
Rule 32 $\longrightarrow$ $[\boldsymbol{X_{t+1}}] = [\boldsymbol{X_t}][\boldsymbol{T_1}]$
Rule 64 $\longrightarrow$ $[\boldsymbol{X_{t+1}}] = [\boldsymbol{T_2}][\boldsymbol{X_t}][\boldsymbol{T_1}]$
Rule 128 $\longrightarrow$ $[\boldsymbol{X_{t+1}}] = [\boldsymbol{T_2}][\boldsymbol{X_t}]$
Rule 255 $\longrightarrow$ $[\boldsymbol{X_{t+1}}] = [\boldsymbol{T_2}][\boldsymbol{X_t}][\boldsymbol{T_2}]$

The next state transition of a CA configured with a secondary rule can be represented as modulo 2 sum of matrices of the concerned primary rules.

For Example:

1) Rule 3 = Rule 1 + Rule 2, the next state transition can be represented as

$$[\boldsymbol{X_{t+1}}]=[\boldsymbol{X_t}]+[\boldsymbol{X_t}][\boldsymbol{T_2}]$$

2) Rule 90 = Rule 2 + Rule 8 + Rule 16 + Rule 64, the next state transition for rule 90 can be represented as

$$[\boldsymbol{X_{t+1}}]=[\boldsymbol{X_t}][\boldsymbol{T_2}]+[\boldsymbol{T_2}][\boldsymbol{X_t}][\boldsymbol{X_t}][\boldsymbol{T_1}]+[\boldsymbol{T_1}][\boldsymbol{X_t}][\boldsymbol{T_1}]+[\boldsymbol{T_2}]\ [\boldsymbol{X_t}][\boldsymbol{T_1}]$$

IV. VLSI Applications

As the semiconductor technology is moving towards the submicron era, the system designers try to embed complex functions from software to hardware blocks on the silicon floor. At the same time for keeping the design complexity within a feasible limit, the designers are forced to look for simple, regular, modular cascadable and reusable building blocks for implementing various complex functions. The homogeneous structure of Cellular Automata (CA) is a right candidate to fulfill all the above objectives. Moreover, the demand for parallel processing architectures has gained importance with ever increasing need for faster computing. To this end, we are motivated to investigate the two dimensional Cellular Automata (2D CA) to arrive at the easily implementable parallel processing architecture in VLSI. Several researchers have proposed VLSI applications of 1D CA. One of the major contributions in this area is the BIST (built in self test) structure of CALBO (CA based Logic Block Observer), which is analogous to the BILBO structure designed around LFSR. It was shown that the patterns generated by 1D CA exhibit better randomness as compared to those generated by LFSR. Applications of CA in BIST schemes have also been extensively investigated in [9-10].

Another potential application area of CA is in signature analysis [11]. The concept of programmable CA has been introduced in [12]. A scheme for generation and decoding of bit error correcting codes, using CA has been proposed [13]. This is referred to as CAECC (CA based Error Correcting Codes). A Novel scheme for hashing in hardware has been proposed in [14]. Implementation of Finite State Machines (FSM) has been proposed in [15]. In [16] a scheme for CA based cipher system design has been proposed. Better randomness of the patterns generated by 2D CA over that of 1D CA and LSFR has been established and accordingly applications of 2D CA as a pseudo-random pattern generator was established in [6].

A cellular Automata Machine (CAM) proposed around the parallel architecture of 2D CA has been reported in [7]. Analysis of 2D images can be undertaken in such a CAM. In addition theory of 2D CA was employed for other vital application areas like text compression [17] VLSI testing and cryptography.

V. Simulation and Synthesis Results of CA PRNG.

The RTL modules of CA PRNG are shown in Fig2, Fig3 and Fig.4 respectively.

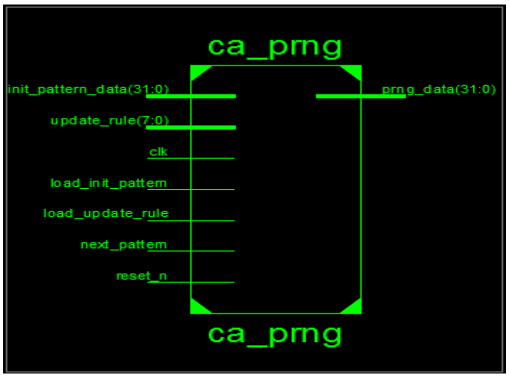


Fig 2: CA PRNG top module.

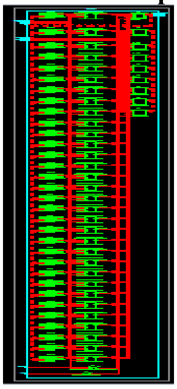


Fig 3: CA PRNG RTL schematic.

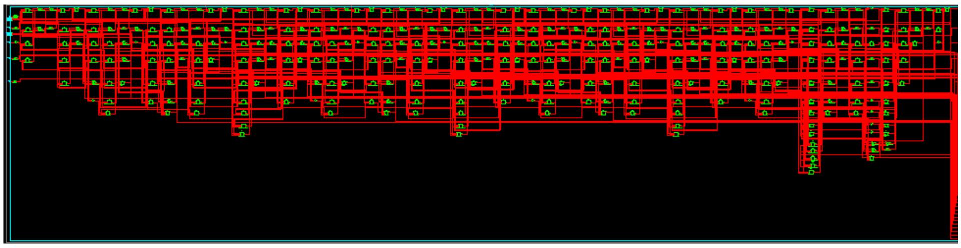
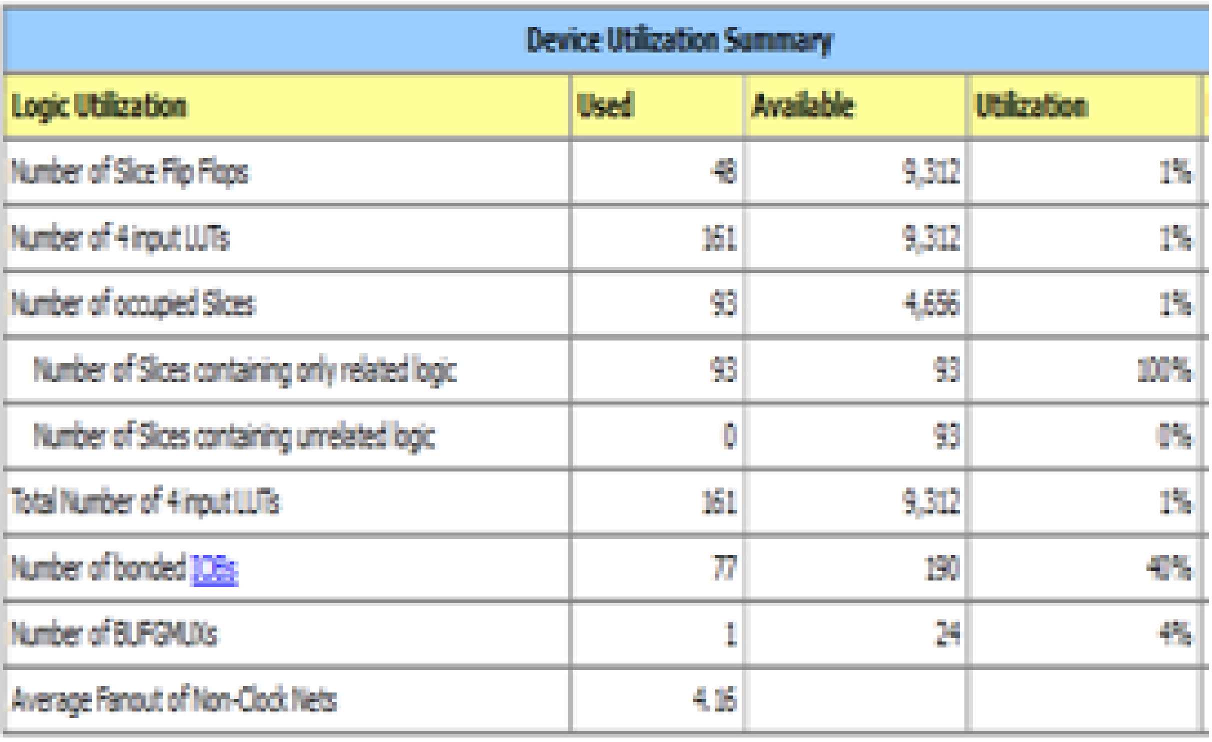


Fig 4: Technology schematic

We have implemented the module on Xilinx XC3S500E fpga and the design summary was given below. It was found that the static power consumption is 81mW at clock frequency of 246 MHz and the area utilization is minimum and it may vary in 2D scenario. The top module and its RTL schematic are presented in Fig 2 and 3.Fig 5 shows the simulated output The CA PRNG design is simulated and synthesized in Xilinx ISE 12.1.



Device Utilization Summary			
Logic Utilization	Used	Available	Utilization
Number of Slice Flip Flops	48	9,312	1%
Number of 4 input LUTs	161	9,312	1%
Number of occupied Slices	93	4,656	1%
Number of Slices containing only related logic	93	93	100%
Number of Slices containing unrelated logic	0	93	0%
Total Number of 4 input LUTs	161	9,312	1%
Number of bonded IOBs	77	190	40%
Number of BUFGMUXs	1	24	4%
Average Fanout of Non-Clock Nets	4.16		

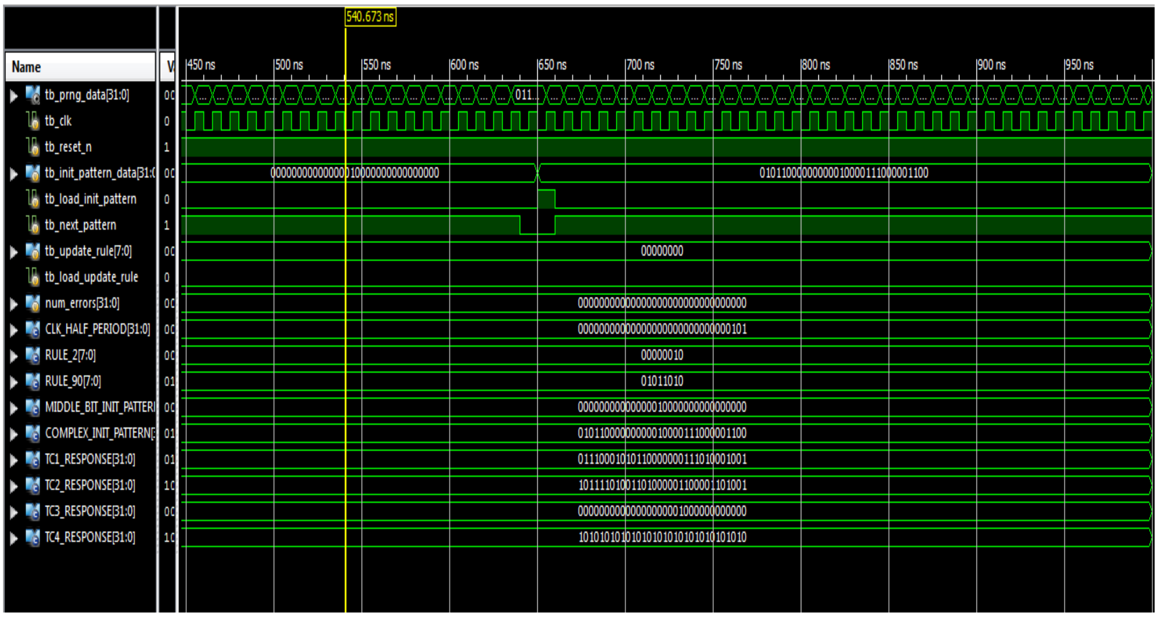


Fig 5: Simulation waveform.

VI. Conclusion

Presently less research has been done to analyze and characterize 2D Cellular Automata. In addition, the randomness properties of 1D CA have been explored at lot. Keeping in view that the extended neighborhood in 2D CA, it is possible to explore the randomness properties of 2D CA. One of the major objectives of this article is to motivate researchers to study the mathematical model of 2D CA and extend it to all group and non group CA. To characterize group and non group 2D CA and identify cycle length of 2D CA group rules and depth and inner cycle length of non group 2D CA. Presently we are studying to extend the application area of 2D CA particularly in digital image processing and testing. Based on our analysis it is possible to apply 2D CA theoretically extend and update other application areas.

.

References

[1]. J. Von Neumann, “The theory of self reproducing Automata” (edited by A. W. Burks), University of Illinois Press, Urbana, (1966).

[2]. S. Wolfram, “Statistical Mechanics of Cellular Automata” Rev. Mod. Physics Vol. 55 No. 3 pp 601-644 (July 1983)

[3]. W. Pries, A. Thanailakis and H.C. Card, “Group Properties of Cellular Automata and VLSI applications” IEEE Trans. On Computers Vol. 35 No 12, pp 1013 -1024 (1986).

[4]. A. K. Das, “Additive Cellular Automata: Theory and applications as a Built in self test structure and VLSI applications”, Ph. D. Thesis, IIT Kharagpur, India (1990)

[5]. N. H. Packard and S. Wolfram, “Two dimensional Cellular Automata” Journal of Statistical physics Vol. 38 No 5/6 pp 901-946 (1985).

[6]. D. R. Chowdhury, I. S. Gupta and P. P. Choudhuri, “A Class of two dimensional cellular automata and applications in random pattern testing”, Journal of Electronic Testing: Theory and Applications, Vol. 5 pp 65-80. (1994)

[7]. A. RaoufKhan , P. P. Choudhury et.al. “VLSI architecture of a cellular automata machine”, Int. Journal Computers and Mathematics with applications, Vol. 33 No. 5 pp79-94, (1997)

[8]. B. K. Nayak, et al. “Colour Graph: An efficient model for two dimensional cellular automata” Private correspondence.

[9]. International Journal of Electrical & Computer Sciences IJECS-IJENS Vol: 10 No: 06 129102805-06-9393 IJECS-IJENS © December 2010 IJENS I J E N S

[10]. A. Albicki and M .Khare, “Cellularautomata used for test generation”,Proc. ICCD, pp 56-59, (1987).

[11]. P. D. Hortensius et al. “Cellular automata based pseudo randomnumber generator for Built in Self Test”, IEEE Trans. On CAD, Vol. 8,pp 842-859, (1989).

[12]. A. K. Das and P. P. Choudhuri,“Efficient Characterization of cellularautomata,” Proc. IEE (part E), Vol.137 pp 81-87, (1990).

[13]. S. Misra et al. “Synthesis of selftestablesequential logic using programmable cellular automata”, inProc. VLSI’92, pp 193-198, (1992).

[14]. D. Roy Chowdhuryet.al., “Design of CAECC- Cellular automata based error correcting codes”, IEEE Trans.on Computers, Vol. 43 No3, pp 371-382, (1994).

[15]. D. Roy Chowdhuryet. al., “A low cost high capacity associative memory design using cellular automata”, IEEE Trans on Computers, Vol.44 No10 (1995).

[16]. B. Mitra et.al, “A flexible scheme for state assignment based on characteristics of FSM”, in proc. ICCAD, pp 226-229 (1991).

[17]. S. Nandi et. al. “Theory and applications of Cellular automat in cryptography”, IEEE Trans. on computers, Vol. 43.No12, pp 1346- 1357 (1994).

[18]. A. RaoufKhan , P. P. Choudhury et.al. “Text Compression using two dimensional cellular automata”, Int. Journal Computers and Mathematics with applications, Vol. 43 No. 6 pp115-127, (1999)

[19]. P.P. Choudhury, K. Dihidar, Matrix Algebraic formulae concerning some special rules of two-dimensional Cellular Automata, International journal on Information Sciences, Elsevier publication, Volume 165, Issue 1-2.

[20]. P.P.Choudhury, B.K. Nayak, S. Sahoo, Efficient Modelling of some Fundamental Image Transformations. Tech.Report No. ASD/2005/4, 13 May 2005.

[21]. S Munshiet.al.,Änalalytical framework for characterizing restricted two dimensional cellular automata evolution”, Journal of Cellular Automata, Vol. 3 No2, pp 313-335 (2008).