

Contents

1	Introduction	13
1.1	Cellular Automata Creating Artificial Reality	14
1.2	Background of this book	15
1.3	Overview of the Book	16
2	A Comprehensive Survey On Cellular Automata	18
2.1	Cellular Automata - Structural Variations	19
2.2	Cellular Automata (<i>CA</i>) Behavior - Local to Global Mapping	21
2.2.1	Linear/Additive <i>CA</i>	22
2.2.2	Non-Linear <i>CA</i>	24
2.3	The Inverse Problem - Global to Local Mapping	24
2.4	<i>CA</i> Applications	26
2.4.1	<i>CA</i> Games	26
2.4.2	<i>CA</i> as Parallel Computing Machine	27
2.4.3	Modeling Non-linear Dynamics	27
2.4.4	<i>CA</i> For Modeling Physical and Biological Systems	28
2.4.5	<i>CA</i> Application in Social Sciences	28
2.4.6	Diverse Applications in VLSI era	29
2.4.7	Pattern Recognition	31
2.5	Conclusion	32
3	Analysis of <i>CA</i> State Space - A Vector Space Theoretic Approach	35
3.1	Cellular Automata Preliminaries	36
3.1.1	<i>CA</i> Terminologies	37
3.1.2	Cellular Automata (<i>CA</i>) Characterization	38
3.1.3	Group and Non-Group <i>CA</i>	39

3.2	Vector Space Theoretic Analysis of Linear CA (LCA)	42
3.2.1	Derivation of Elementary Divisors From T Matrix (Step 1 of Algorithm 3.1)	44
3.2.2	Generation of Cyclic Sub-space of Elementary Divisors (Step 2 of Algorithm 3.1)	46
3.2.3	Generation of Cycle Structure of an LCA having Multiple Elementary Divisors (Step 3 of Algorithm 3.1)	50
3.3	Vector Space Theoretic Analysis of Additive CA (ACA)	59
3.3.1	Method to determine the presence of cycle of length (k) in an ACA (Property P_1)	60
3.3.2	Special class of C' for which cycle structure is identical to that of C (Property P_2)	61
3.3.3	Class of ACA with cycle structure different from that of LCA (Property P_3)	63
3.3.4	Algorithm for Enumerating Cycle Structure of an ACA	69
3.4	Conclusion	71
4	Cellular Automata Synthesis	79
4.1	Synthesis of a Linear CA (LCA)	79
4.1.1	Generation of $EFCS$ from cycle structure (CS) - Step A of Algorithm 4.1	81
4.1.2	Generation of ECS from $EFCS$ - Step B of Algorithm 4.1	86
4.1.3	Find Elementary Divisors (EDs) for ECS - Step C of Algorithm 4.1	89
4.1.4	LCA Synthesis from Elementary Divisors - Step D of Algorithm 4.1	91
4.2	Synthesis of Additive Cellular Automata (ACA)	94
4.2.1	Generation of $EFCS'$ from Cycle Structure (CS') - Step A of Algorithm 4.1	97
4.2.2	Generation of ECS from $EFCS'$ - Step B of Algorithm 4.1	102
4.2.3	Synthesis of Inversion Vector - Step E of Algorithm 4.1	103
4.3	The Cellular Automata Toolkit (CAT)	104
4.4	Conclusion	106
5	Analysis and Synthesis of Multiple Attractor Cellular Automata ($MACA$)	115
5.1	Multiple Attractor Cellular Automata ($MACA$)	116
5.2	$MACA$ Based Hamming Hash Family (HHF)	120
5.2.1	Computation of Expected Occurrence (EO)	122

5.3	<i>MACA</i> Synthesis - A <i>GA</i> (Genetic Algorithm) Formulation	127
5.3.1	The Encoding Scheme - A Pseudo Chromosome Format	128
5.3.2	Random Generation of the Initial Population	129
5.3.3	Mutation Algorithm	131
5.3.4	Crossover Algorithm	132
5.3.5	Selection, Crossover and Mutation Probability	133
5.4	Dependency Vector/String for <i>MACA</i> Characterization	134
5.4.1	Dependency Vector and Dependency String for Characterizing a Vector Subspace	135
5.4.2	Legal Dependency Vector and String	136
5.4.3	Application of Dependency String (DS) for Identification of PEF of an attractor basin	138
5.4.4	Synthesis of Dependency Matrix From Dependency String	140
5.5	Genetic Algorithm (GA) for Evolution of Dependency Vector and String . .	143
5.5.1	Chromosome Format	143
5.5.2	Random Generation of Initial Population	143
5.5.3	Crossover Algorithm	144
5.5.4	Mutation Algorithm	145
5.6	Conclusion	146
6	<i>MACA</i> Based Associative Memory Model for Pattern Classification/Clustering	157
6.1	Pattern Recognition - A Brief Survey	159
6.1.1	Definition, Categorization & Applications	159
6.1.2	Similarity Measures	160
6.1.3	Template Matching, Syntactic and Statistical Pattern Recognition .	161
6.1.4	Decision Tree	162
6.1.5	Neural Networks	163
6.1.6	Hopfield Network	163
6.1.7	Clustering Techniques	164
6.2	<i>MACA</i> Modeling Associative Memory	165
6.2.1	Design Guidelines	166
6.2.2	Pattern Recognition	168
6.2.3	Evolutionary Synthesis of <i>MACA</i> Modeling Associative Memory . .	169
6.2.4	Theoretical Analysis and Experimental Results	170
6.2.5	Multiple <i>MACA</i> Based Associative Memory Model	179

6.3	Design of <i>MACA</i> Based Pattern Classifier	185
6.3.1	Two-Class Classifier	185
6.3.2	Performance Analysis of <i>MACA</i> based Classifier	189
6.3.3	Two Stage Two Class Classifier (TCC)	193
6.4	Scheme I: Hierarchical Two Class Classifier (HTCC)	196
6.5	Scheme II - <i>MACA</i> Based Tree-Structured Classifier (MTSC)	199
6.5.1	Design of <i>MACA</i> Tree	200
6.5.2	<i>MACA</i> Selection to build <i>MACA</i> tree	202
6.6	Application of <i>MACA</i> Based Pattern Classifier	203
6.6.1	<i>MACA</i> Based Pattern Classifier For Data Mining	204
6.6.2	Experimental Set-up	204
6.6.3	Performance of <i>MACA</i> based pattern classifiers	206
6.7	Pattern Clustering	208
6.7.1	<i>MACA</i> Based Pattern Clustering Model	209
6.7.2	Experimental Results	214
6.8	Conclusion	216
7	Hierarchical Cellular Automata (HCA)	217
7.1	Extension Field	218
7.1.1	Extension Field $GF(2^p)$	218
7.1.2	$GF(2^p)$ Field Elements	219
7.1.3	Operations in Extension Field $GF(2^p)$	221
7.2	$GF(2^p)$ CA	222
7.2.1	$GF(2^p)$ CA Structure	222
7.2.2	Hardware Realization of $GF(2^p)$ CA	224
7.2.3	Characterization of $GF(2^p)$ CA	224
7.2.4	Characterization of Group $GF(2^p)$ CA	226
7.2.5	Non-Group $GF(2^p)$ CA	233
7.3	Additive $GF(2^p)$ CA	236
7.3.1	Cycle Structure and Depth	237
7.3.2	Group ACA Characterization	238
7.3.3	Characterization of $GF(2^p)$ Non-group ACA	241
7.4	Analysis of $GF(2^p)$ CA State Transition Behavior	244
7.4.1	Characteristic polynomial and Elementary Divisor	244

7.4.2	Enumerating Cycle Structure of an <i>LCA</i> having a single Elementary Divisor	245
7.4.3	Terminologies	245
7.4.4	Vector Space Theoretic Analysis Based on Elementary Divisor . . .	246
7.4.5	Deriving Elementary Divisors From T Matrix	247
7.4.6	Enumerating Cycle Structure and Depth from Elementary Divisors	248
7.5	Vector Space Theoretic Analysis of Additive $GF(2^p)$ <i>CA</i>	251
7.6	Advantages of $GF(2^p)$ <i>CA</i>	259
7.7	Extension of Extension Field	260
7.8	Hierarchical Cellular Automata (HCA)	261
7.8.1	Hierarchical <i>CA</i> Structure	263
7.8.2	Hardware Implementation of <i>HCA</i>	264
7.8.3	Reduction of Overhead due to XOR gates	266
7.9	Characterization of <i>HCA</i>	266
7.9.1	Group HCA	266
7.9.2	Non-Group <i>HCA</i>	270
7.9.3	Enumerating Cycle Structure and Depth of <i>HCA</i> from Elementary Divisors	270
7.9.4	Algorithm to Find Cycle structure	271
7.10	Synthesis of Hierarchical Cellular Automata	273
7.10.1	Conversion of Cycle Structure to Elementary Component	273
7.10.2	Conversion of Elementary Components into Elementary Divisor . . .	274
7.10.3	Generation of the <i>CA</i> from the Elementary Divisor	275
7.11	Conclusion	277
8	CA Based BIST Structure For Testing VLSI Circuits	278
8.1	BIST Structure - A Brief Survey	279
8.1.1	Test Pattern Generation	279
8.1.2	Response Analysis	280
8.1.3	BIST Structures - the Current Trends	281
8.2	$GF(2^p)$ CABIST	283
8.2.1	Design of $GF(2^p)$ <i>CATPG</i>	283
8.2.2	Design of $GF(2^p)$ <i>CABIST</i>	297
8.2.3	Experimental Results	302

8.3	Hierarchical <i>TPG</i>	307
8.3.1	<i>HCATPG</i> For a CUT with Hierarchical Structural Description . . .	308
8.3.2	<i>HCATPG</i> For a Flat Circuit without Hierarchical Structural Description	316
8.4	Fault Diagnosis	321
8.4.1	Overview of Diagnosis Scheme	322
8.4.2	<i>MACA</i> Revisited	324
8.4.3	Design of <i>MACA</i> Based Pattern Classifier	325
8.4.4	<i>MACA</i> Based Fault Diagnosis	338
8.4.5	Experimental Results	343
8.5	Design of Test Pattern Generator Without Prohibited Pattern Set	351
8.5.1	$GF(2)$ <i>CA</i> Theory Revisited	352
8.5.2	Design of PRPG Without PPS	353
8.5.3	Experimental Results	360
8.5.4	Conclusion	363
9	Programmable Cellular Automata(PCA)	364
9.1	Programmable CA cells	364
9.1.1	Programmable Cell of Linear CA(LCA)	365
9.1.2	Programmable Cell of Additive CA(ACA)	366
9.1.3	Universal Programmable CA(UPCA) cell	367
9.2	Program Instruction of a PCA	368
9.3	Cellular Automata Processor (CAP)	369
9.3.1	Functional Blocks of CAP Architecture	369
9.3.2	CAP Instructions	371
9.3.3	Execution of CAP Instruction	372
9.4	Programmable Hierarchical CA (PHCA)	372
9.4.1	$GF(2^p)$ CA Revisited	373
9.4.2	Programmable HCA (PHCA) cell	376
9.4.3	Program Structure for PHCA	380
10	Data Encompression	385
10.1	Data Compression and Encryption - A Brief Overview	386
10.1.1	Data Compression	386
10.1.2	Data Encryption	389

10.2	Encompression - An overview	389
10.2.1	CA Based Model for Compression Operation in Encompression . . .	389
10.2.2	CA Based Model for Encryption Operation in Encompression	391
10.3	Cellular Automata (CA) Transforms	391
10.3.1	Linear/Additive CA	392
10.3.2	Group CA: Reversible Transform	393
10.3.3	Non Group CA: Non-reversible Transform	393
10.3.4	Multiple Attractor Cellular Automata (MACA)	394
10.3.5	Non Linear CA	398
10.4	MACA - As A Pattern Classifier to Model Compression Operation	399
10.4.1	MACA Based Two Stage Classifier	400
10.5	CA Based Model of Vector Quantization Scheme	401
10.5.1	Codebook Design	402
10.5.2	Image Analysis and Modeling	403
10.5.3	Codebook Generation	405
10.5.4	CA Based Model of PTSVQ	406
10.5.5	Experimental Results of CA based Compression	406
10.6	CA Based Cryptosystem (<i>CAC</i>)	409
10.6.1	Experimental Results of CA based Cryptography (<i>CAC</i>)	410
10.7	Architecture of Encompression Technology	410
10.8	Conclusion	412
11	Cellular Automata Models for Authentication and Watermarking	416
11.1	Authentication and Watermarking - A Brief Survey	417
11.2	Single Attractor Cellular Automata (<i>SACA</i>)	418
11.2.1	$GF(2^p)$ <i>CA</i> Preliminaries	418
11.2.2	<i>SACA</i> Characterization	419
11.2.3	<i>SACA</i> as a unique address generator	422
11.3	Cellular Automata based Authentication (<i>CAA</i>)	423
11.3.1	Protocol for Message Authentication	424
11.3.2	Hash Function for Generation of Message Digest	424
11.3.3	Security Analysis	427
11.4	<i>CAA</i> for Fragile Watermarking	431
11.4.1	Watermark Insertion	431

11.4.2	Watermark Extraction	433
11.4.3	Experimental Results	436
11.5	<i>CAA</i> Based Robust Watermarking For Image Data	437
11.5.1	Watermarking for Image Compressed with <i>DCT</i>	437
11.5.2	Watermarking for Image Compressed with Wavelet Transform . . .	448

List of Figures

1	Associative Memory Model	5
2	Classification Model	6
3	Associative Memory Model	6
3.1	Configuration of a 3-neighborhood CA cell	36
3.2	A 5-cell group CA with rule vector $\langle 102, 240, 102, 150, 240 \rangle$ and its cycle structure $CS = [1(1), 1(3), 1(7), 1(21)]$	40
3.3	State Transition Diagram of a 4-cell non-group CA with rule vector $\langle 102, 60, 90, 60 \rangle$	41
3.4	State Transition Diagram, T matrix, characteristic polynomial, minimal polynomial, and elementary divisors of a 7-cell non-group CA Note: As per the formulation noted in <i>Section 3.2</i> , both the T matrices in <i>Fig. 3.4 (a) & (b)</i> have identical characteristic polynomial expressed in Elementary Divisor form and generate identical cycle structure	75
3.5	An example LCA with its elementary divisors and corresponding elementary cycle structures (ECS s)	76
3.6	Formation of final cycle structure	77
3.7	Additive CA with its rule vector $\langle 195, 195, 195, 15 \rangle$, T matrix and inversion vector along with its cycle structure	78
5.1	Associative Memory Model	116
5.2	Classification Model	116
5.3	State transition diagram of a 5-cell $MACA$ with 4 attractor basins having Characteristic Matrix T , Rule Vector $\langle 204, 170, 90, 240, 204 \rangle$, and characteristic polynomial $x^3(1+x)^2$	117
5.4	Concatenation of m number of 2-attractor $MACA$ to form an $MACA(n, m)$	119
5.5	Expected Distribution of $MACA$ with two attractors ($m = 1$)	124
5.6	Expected Occurrence ($EO(w, n, 2)$) with 8 attractors ($m = 3$)	127
5.7	Expected Occurrence ($EO(w, n, 2)$) with 8 attractors ($m = 3$)	127

5.8	Encoding of <i>MACA</i> in pseudo-chromosome format derived out <i>MACA</i> rule vector, <i>T</i> matrix, elementary divisors and corresponding sub-matrices . . .	128
5.9	<i>MACA</i> rule vector, Characteristic Matrix <i>T</i> and its <i>pseudo-chromosome format</i>	129
5.10	Different methods to arrange two matrices	130
5.11	Illustration of Mutation Algorithm	131
5.12	An Example of Cross-over Technique	133
5.13	Illustration of Identical attractor basins of three <i>MACAs</i>	135
5.14	State transition diagram of a 5 cell <i>MACA</i> with 4 attractor basins with attractors 0(00000), 3(00011), 4(00100), and 7(00111). Note: (i) <i>PEF</i> (as explained in Section 5.4.3 and Example 5.9) covers 3rd and 4th bit position of an attractor with Most Significant Bit (<i>MSB</i>) counted as 1st bit. (ii) Dependency matrix <i>T</i> formed out of block diagonal placement of two sub-matrices <i>T</i> ₁ and <i>T</i> ₂	137
5.15	Dependency String (DS) of Dependency Matrix (<i>T</i>) of Fig.5.14 formed through concatenation of two Dependency Vectors. Note : (i) While the first Dependency Vector contributes the first bit of the <i>PEF</i> (pseudo-exhaustive field), second Dependency Vector specifies the second bit of 2 bit <i>PEF</i> . (ii) Dependency String of <i>T</i> Matrix is obtained through concatenation of two Dependency Vectors of <i>T</i> ₁ and <i>T</i> ₂ respectively - which are placed in non-overlapping positions.	138
5.16	An <i>n</i> -bit Dependency String (DS) derived out of <i>m</i> Dependency Vectors (DVs). Note: Each Dependency Vector (DV) contributes one bit (0 or 1) of the <i>PEF</i> (pseudo-exhaustive field) of an attractor basin.	139
5.17	An example chromosome for current GA formulation	143
5.18	An example of crossover technique	144
5.19	An example of mutation technique	145
6.1	Associative Memory Model	157
6.2	State transition diagram, Characteristic Matrix <i>T</i> (Rule Vector < 204, 170, 90, 240, 204 >) of a 5 cell <i>MACA</i> having 4 attractor basins and depth <i>d</i> as 3.	165
6.3	Division of State Space around the given pattern set $\mathcal{P} = \langle 01010, 01111, 10010, 10001 \rangle$ referred to as pivot points.	167
6.4	<i>MACA</i> based Associative Memory	168
6.5	Experimental Results of noise recovery capacity of <i>MACA</i> Note: <i>w</i> denotes noise in number of bits and $p(w)$ reflects the noise recovery capacity, while <i>k</i> is the number of <i>n</i> bit patterns to be memorized.	173

6.6	Theoretical Estimation (Relation 6.9) of noise recovery capacity of <i>MACA</i> at memorizing capacity for $n = 10, 30, 40, 60$. Note: While w denotes noise in terms of number of bits, $p(w)$ reflects the noise recovery capacity.	174
6.7	Experimental data and theoretical evaluation of recovery capacity $p(w)$ of noise (w) for $n =$ number of bits in a pattern $= 40$, and $k =$ number of patterns to be learnt $= 22$	175
6.8	Gaussian Probability Distribution Function of the relation $N(n, m) = f(1, n, m, x)$ and its corresponding Standard Normal form obtained through scaling . . .	177
6.9	Ideal basins of attraction and Noise Recovery Graph	180
6.10	Enhancement of noise recovery capacity achieved through use of Multiple <i>MACA</i>	182
6.11	Noise Recovery capacity of ($n = 40$) Multiple <i>MACA</i> with $\mathcal{N} =$ number of <i>MACAs</i> $=1,6,20$	183
6.12	<i>MACA</i> based Classification Strategy Note: MSB and LSB of attractor states constitute PEF (Pseudo Exhaustive Field)	186
6.13	Distribution of patterns in Class 1 and Class 2 (<i>HD</i> denotes Hamming Distance)	190
6.14	Cluster detection by Two-class classifier	192
6.15	Two Stage Classifier	193
6.16	Hierarchical Two Class Classifier (HTCC) modeling a multi-class classifier .	197
6.17	Decision tree	200
6.18	Basic architecture of <i>MACA</i> based tree-structured pattern classifier	201
6.19	PEF (Pseudo Exhaustive Field) of <i>MACA</i> attractors storing cluster information. Note: Most Significant Bit (MSB) and Least Significant Bit (LSB) constitute the PEF	210
6.20	An <i>MACA</i> -tree for a dataset= $\{00000, 01000, 11000, 11001, 10010, 10011, 11111, 01111\}$	211
7.1	The field elements of $GF(2^2)$	220
7.2	General structure of a $GF(2^p)$ CA.	222
7.3	A 3-cell $GF(2^2)$ CA.	223
7.4	T matrix of the example 3-cell $GF(2^2)$ CA with its generator polynomial and corresponding plus and star tables.	224
7.5	T matrix and state transitions of a 3-cell $GF(2^2)$ group CA.	227
7.6	T matrix of a 3-cell $GF(2^2)$ non-group CA and its state transitions.	228

7.7	A 2-cell $GF(2^2)$ <i>non – group CA</i> state transition	234
7.8	A 3-cell $GF(2^2)$ group <i>ACA</i> along with its characteristic matrix (T) and the inversion vector (F) with cycle structure [4(4), 4(12)] Note: Cyclic state transitions of two representative cycles are only shown	238
7.9	State transition diagram of a 3-cell non-group $GF(2^2)$ <i>ACA</i>	241
7.10	State transition diagram of a 3-cell non-group $GF(2^2)$ <i>LCA</i>	243
7.11	State Transition Diagram of a 3-cell non-group $GF(2^2)$ <i>ACA</i>	243
7.12	Characteristic Matrix (T) and Inversion Vector (F) of a 5 cell $GF(2^2)$ <i>EAM</i> ($C'(T, F)$). [For the corresponding <i>ELM</i> , F is the all 0s vector]	246
7.13	Hierarchical structures of field elements.	262
7.14	General structure of an n -cell Hierarchical <i>CA</i>	263
7.15	Structure of a $GF(2^{p^r})$ Hierarchical <i>CA</i> cell.	264
7.16	A 3-Cell $GF(2^{2^3})$ <i>HCA</i> (a group <i>CA</i>).	265
7.17	270
8.1	A generic n -cell null boundary <i>CATPG</i>	284
8.2	General structure of a $GF(2^p)$ <i>CATPG</i> cell.	285
8.3	High-level model for s349 4-bit multiplier.	286
8.4	Pseudo control input of a <i>CUT</i>	286
8.5	Construction of dependency matrix <i>D</i>	289
8.6	A 3-cell $GF(2^3)$ uniform <i>CATPG</i>	290
8.7	A 3-cell $GF(2^3)$ hybrid <i>CATPG</i>	291
8.8	Construction of dependency matrix <i>D</i> in 3-neighborhood.	293
8.9	Extraction of disjoint paths from dependence graph.	295
8.10	Cone point extraction to detect HFs.	296
8.11	Transition count.	296
8.12	Folding of a $GF(2^p)$ <i>CATPG</i>	298
8.13	Folding of <i>SignatureAnalyzer</i>	300
8.14	High-level model for modified c432 bench circuit & the <i>HCATPG</i> of the <i>CUT</i>	310
8.15	The example <i>CUT</i> ‘L’.	324
8.16	State Transition diagram of a 3-cell non-group $GF(2^2)$ <i>CA</i>	325
8.17	0-basin of a 3-cell $GF(2^2)$ <i>D1-MACA</i>	326
8.18	The 2-Cell $GF(2^2)$ <i>D1-MACA</i> based classification.	327
8.19	Three class classification with 2-Cell $GF(2^2)$ <i>D1-MACA</i>	328
8.20	The $n \times n$ tri-diagonal <i>T</i> matrix.	330

8.21	T matrix and tri-graph of a 5-cell $GF(2^2)$ D1-MACA.	331
8.22	Multi-class classification to identify the class i of an element S_{ij}	332
8.23	Partitioning of pattern set in disjoint classes.	332
8.24	The 2-cell $GF(2^2)$ D1 – MACA.	334
8.25	Module diagnosis of an example CUT ' L_1 '.	340
8.26	Diagnosis of an example CUT ' L'_2 '.	342
8.27	Hierarchical classification.	343
8.28	Fault partitioning.	347
8.29	Valid nodes of the tri-graph in $GF(2^p)$	350
8.30	$PRPG$ without the prohibited patterns.	354
9.1	Programmable cell of LCA with control vector $\langle c_{i-1} \ c_i \ c_{i+1} \rangle$	365
9.2	Programmable cell of Additive(ACA) CA with control	367
9.3	Structure of a Universal Programmable CA (UPCA) cell	367
9.4	PCA operated by program	368
9.5	The structure of programmed instruction executed on a PCA	369
9.6	CAP(Cellular Automata Processor) Architecture	370
9.7	Instruction Format of CAP	371
9.8	$GF(2^p)$ CA structure	373
9.9	Hardware realization of 3 cell $GF(2^P)$ CA	376
9.10	HCA cell structure with its generator polynomial generating weight $w=2$. .	379
9.11	Matrix representation and cell structure of a $GF(2^4)$ HCA cell with two different generator polynomials generating weighted output with weight values 3 and 4	380
9.12	A $GF(2^p)$ PHCA cell structure with $p=3$ and generator polynomial as $x^3 + x^2 + 1$	382
9.13	A $GF(2^p)$ PHCA cell with $p=4$	383
9.14	Subfield structure of Field 1	383
9.15	Program Instructions of n cell PHCA	384
10.1	Basic principle of Encompression	389
10.2	Architecture of Encompression- Encoder	390
10.3	Block Diagram of Encryption Scheme	391
10.4	State transition diagram of a maximum length group CA	393
10.5	State transition diagram of a non-maximum length group CA	393

10.6 State Transition Diagram of a 5-cell non-group CA (MACA) with 4 attractor basins	394
10.7 T_1, T_2, \dots , etc. in Block Diagonal Form. T_i (T_j) has no dependency on T_j (T_i)	395
10.8 An n-bit Dependency String (DS) consists of m number of Dependency Vectors (DVs).Note: Each Dependency Vector (DV) contributes the value of each pseudo-exhaustive bit (either 0 or 1) of an attractor basin.	396
10.9 Dependency String (DS) of Dependency Matrix (T) of Fig.10.3.3 which is formed through concatenation of two Dependency Vectors. Note : (i) While the first Dependency Vector contributes the value of first pseudo-exhaustive bit, second Dependency Vector gives the value of another bit. (ii) Dependency String of T Matrix is obtained by concatenating two Dependency Vectors of T_1 and T_2 respectively - which are placed in non-overlapping positions.	397
10.10MACA based two class classification strategy	400
10.11Two Stage Classifier	401
10.12Encoder and Decoder	402
10.13Block Diagram of Codebook Generation scheme	403
10.14Curve of Standard Deviation against different block size of human portrait images	404
10.15The logical Structure of multi-class Classifier equivalent to the PTSVQ . . .	407
10.16Original image and Decompressed image with compression ratio=98.82% . .	407
10.17Original image and Decompressed image with compression ratio=98.73% . .	408
10.18Original and Decompressed image of brain MRI with compression ratio 97.36%	408
10.19Original and Decompressed image of brain MRI with compression ratio 98.35%	408
10.20PSNR of all test images for CA based compression and other standard algorithms when performing lossy encoding at 0.25, 0.5 1 and 2 bpp and other standard	409
10.21Programmable CA(PCA) with XOR/XNOR rule	411
10.22Architecture of Encompression- Decoder	411
11.1 General Structure of a $GF(2^p)$ CA (For p=1, It's a Conventional $GF(2)$ CA)	417
11.2 State Transition Diagram of 3-cell $GF(2^2)$ SACA and Its Dual	420
11.3 Unique Hash Addressing Using 3-cell $GF(2^2)$ SACA	423
11.4 Message Authentication	424
11.5 Watermark Extraction for Fragile/Robust to JPEG Method	435
11.6 Original and Watermark Embedded Host Image Lenna	435
11.7 Watermark Insertion and Extraction for Robust to JPEG Watermarking . .	439

11.8 NC vs. MSE for True and False Watermarking	444
11.9 Watermarking	445
11.10 NC vs. Quality in JPEG Lossy Compression	448
11.11 NC vs. Quality in JPEG2000 Lossy Compression	449
11.12 Watermark Insertion for Robust to JPEG2000 Watermarking	450
11.13 Derive_Data (Step 1.)	451
11.14 Derive_Data (Function of Step 2.)	452
11.15 Derive_Data (Result of Step 2.)	453
11.16 Derive_Data (Step 3. and Step 4.)	454
11.17 Derive_Image (Function of Step 1.)	455
11.18 Derive_Image (Result of Step 1.)	456
11.19 Derive_Image (Step 2.)	457
11.20 Embedding in Brief	458
11.21 Derive_Data (Step 1.)	459
11.22 Derive_Data (Function of Step 2.)	460
11.23 Derive_Data (Result of Step 2.)	461
11.24 Derive_Data (Step 3. and Step 4.)	462

List of Tables

3.1	Additive <i>CA</i> Rules	37
4.1	Computation of elementary cycle structure (<i>ECS</i>)	109
5.1	System of Linear Equations representing a set of Vectors	135
6.1	Applications of Pattern Recognition	160
6.2	Memorizing Capacity (Experimental Result) at different values of n	170
6.3	Fraction of <i>MACA</i> Pool Required to Memorize Patterns	176
6.4	Experimental Results on Classification Accuracy of <i>MACA</i> based classifier .	190
6.5	Cluster Detection by <i>MACA</i> based Classifier	192
6.6	Comparison of Memory Overhead Between Single Stage (SS) and Two Stage (TS) Two Class Classifier	195
6.7	Description of Datasets and Experimental Setup	205
6.8	Comparison of Classification Accuracy between Different Algorithms	207
6.9	Comparison of Memory Overhead between Different Algorithms	208
6.10	Comparison of <i>MACA</i> Based Tree-Structured Classifier and Decision Tree (C4.5)	209
6.11	Performance on Synthetic and Real-life Binary Dataset	215
6.12	Performance on synthetic and real-life real-Valued dataset	216
7.1	The star_table and plus_table for $GF(2)$	219
7.2	The star_table and plus_table for $x^2 + x + 1$	221
7.3	The star_table and plus_table for $x^3 + x + 1$	221
7.4	Cycle structure: primitive vs non-primitive weights	231
7.5	Hierarchy in <i>CA</i>	263
7.6	Example converting cycle structure to elementary component	274
8.1	Test results with $GF(2^p)$ <i>CATPG</i>	304

8.2	Test results on running hard fault (<i>HF</i>) detection algorithms	305
8.3	Comparison of test results	306
8.4	Overhead reduction in CABIST	307
8.5	Circuit descriptions	315
8.6	Test results of customized design	316
8.7	Randomness Test I	318
8.8	Randomness Test II	319
8.9	Test results of HCA	320
8.10	Comparison of test results	321
8.11	Test results of dictionary method for ISCAS benchmarks	344
8.12	Test results of diagnosis scheme I	345
8.13	Circuits with hierarchical descriptions	346
8.14	Results of hierarchical fault diagnosis I	347
8.15	Results of hierarchical fault diagnosis II	348
8.16	<i>follow – nodes</i> of valid-node in $GF(2^2)$	351
8.17	<i>start</i> and <i>final</i> nodes in $GF(2^2)$	351
8.18	Success rate of the <i>PRPG</i> design	361
8.19	Randomness Test I	362
8.20	Randomness Test II	362
8.21	Comparison of Test Results	363
9.1	CA rules	365
9.2	Control bits of an LCA cell	366
9.3	Control bits of XNOR rules of an ACA cell with $c' = 1$	366
9.4	Matrix Representation of extension field element for generator polynomial $x^4 + x + 1$	375
9.5	Matrix Representation of extension field element for generator polynomial $x^4 + x^3 + 1$	377
9.6	Matrix Representation of $GF(2^3)$ elements with generator polynomial $x^3 + x^2 + 1$	378
9.7	16 bit Control Vector(input to 16 bit control register in Fig 9.13) generating variable weights $w=0$ to 15 for 2 different generator polynomials	378
10.1	Results of human face Image	413
10.2	Results of brain-MRI	413

10.3	Differential Cryptanalysis of CAC scheme and Comparative Study with <i>DES</i> and <i>AES</i>	414
10.4	Measurement of Shannon's Security Quotient and comparative study with <i>DES</i> and <i>AES</i>	414
10.5	Comparison of Execution Time of Software Version of <i>CAC</i> and <i>AES</i> . . .	415
11.1	Speed of Generation of <i>SACA</i>	427
11.2	Results of Avalanche Effect on <i>CAA</i> and MD5	428
11.3	Results of Related-Key Cryptanalysis on <i>CAA</i> and MD5	429
11.4	Differential cryptanalysis on <i>CAA</i> and MD5	430
11.5	Comparative Performance regarding speed in WindowsNT 4.00-1381,IBM .	431
11.6	Comparison of PSNR at Fragile Watermarking	436
11.7	Comparative PSNR and NC at Robust watermarking	443
11.8	Different NC values for Different Compression Ratio in JPEG	447
11.9	Different NC values for 0.25 bpp in JPEG2000	447
11.10	Different NC values for 0.50 bpp in JPEG2000	450
11.11	Different NC values for 0.25 bpp in JPEG2000	460