COOPERATIVE ARTIFICIAL IMMUNE SYSTEM AND RECURRENT NEURAL NETWORK ERROR CORRECTION SCHEME FOR GENERATING ROBUST HARDWARE KEY

SHAHRIAR B. SHOKOUHI

School of Electrical Engineering, Iran University of Science & Technology, Tehran, Iran
E-mail: bshokouhi@iust.ac.ir

Abstract- In this paper, a Computational Intelligent (CI) error detection\correction scheme is proposed that is based on cooperation between Artificial Immune System (AIS) and Recurrent Neural Network (RNN). The key idea is to incorporate the search capability, detection, and classification of the AIS algorithms for error characterization for Physically Unclonable Function (PUF) error characterization. AIS also manage the supervised learning of RNNs. The parallel structure of Bidirectional Associated Memory (BAM) neural networks is used to correct the PUF occurred errors. The results demonstrate effectiveness of the proposed structure as an intelligent error correction scheme to have a trusted hardware.

Keywords- Artificial Immune System, Error Detection/Correction, PUF Key Generation, Recurrent Neural Network.

I. INTRODUCTION

Password is an important part of today's life. Measurements from intrinsic specifications of a person or a circuit can provide core knowledge for generating a secure key. This knowledge along with the cryptographic primitives can provide a commercial key for most applications. Error Correction Coding (ECC) is the main step for implementing a cryptographic key. Nature of a secure hardware and its variability, because of the intrinsic and extrinsic parameters such as age, temperature, and voltage, imply to use an ECC to achieve stable response. The accuracy of a typical conventional ECC is mainly related to different parameters such as message length, added redundant bits, and number of occurred error bits in the received message. Error correction limit, failure rate, and complexity are the most important design concerns for ECC algorithms. It has been observed that there are some functional similarities between an error correcting scheme and a recurrent neural network such as: distributed structure, high level of separability between pieces of information, being content associated memory, and resistance to noise. This fact motivates us to revisit the neural computation field with the help of error correction concepts. Also, new area of research on digital signature for intellectual property (IP) has been proposed for securing Field Programmable Gate Array (FPGA), integrated circuits (IC), and embedded systems. Physically Unclonable Functions (PUFs) are promising way to address against the problems of counterfeiting, cloning and reverse engineering. Our main goal of this research is to introduce a combined version of CI tools that works as stability algorithm and also provides a robust and secure ECC for generating a secure hardware PUF key. Therefore, it is one of the most important research topics in this area and has been the focus of only a few research studies. One of the original works for applying NN for ECC is presented by Bruck and Blaum in 1989. It is shown that for a given error correcting code, they can construct a neural network in which every local maximum is a codeword and vice versa. Simultaneously, Yuan et al. explained the behavior of a particular class of neural networks and their application to error control coding. They prove that the neural network is able to perform analog error correction without hanging up at some undesired states. The corresponding code is a balanced code which has significantly higher rate than original Hopfield code. A recent work by Berrou and Gripon illustrate that error correcting codes, combined with sparse data representation, can play in neural networks. ECC is introduced in associative memories based on Hopfield networks in order to increase the learning diversity as well as the recall robustness in presence of erasures and errors.

In this paper, we propose a novel cooperative Artificial Immune System (AIS) and Recurrent Neural Network (RNN) scheme which takes advantage of two sources of learning in order to improve the ECC performance in terms of error detection and correction. One of the most important advantages of the proposed scheme is that it has immune memory, compared to other proposed methods in this field, where provides fast and accurate error detection approach. A RNN with capabilities of learning and memorizing of input/output patterns has been chosen which works as an associated memory. The associated memory allows the recall of information based on partial knowledge of its contents. An associative memory is a Content Addressable Memory (CAM) that maps specific input representations to specific output representations. The weights of the connections between the neurons have to be set that the states of the system corresponding with the patterns which are to be stored in the network are stable. When the network is activated with a noisy or incomplete test pattern, it will render the incorrect or missing data by iterating to a stable state which is close to the main

pattern. Bidirectional Associated Memory (BAM) neural network has been chosen for the correction process and furthermore to generate a secure helper data. It has found that a parallel structure of BAM networks can increase accuracy and security of the generated helper data. Also, the training step of the parallel network will be supervised by the AIS algorithms to provide an adaptive learning process.

AIS inspired from mammals immune system as a new computational intelligence tool for solving complex problems. In recent years, AIS has been applied for solving problems in different areas such as fault detection, computer network and security, hardware implementation, and even hardware layout optimization and problem recognition. The use of AIS algorithms to manage a supervised learning procedure of a hardware response is proposed in this paper. The AIS algorithms have been presented in this paper are based on the mechanisms are called Negative Selection Algorithm (NSA) and Positive Selection Algorithm (PSA). These mechanisms explain the way in which immune cells capability to eliminate self-cells and detect antigens as nonself elements. AIS based systems have shown good error detection result. Furthermore, combination of AIS with parallel BAMs have demonstrated good correction results as well.

II. THE COOPERATIVE AIS-NN

The overall structure of the proposed cooperative AIS-NN are discussed in this section. Applying AIS on a circuit is an original idea that has been revealed in this paper. In the first step, we introduce the AIS algorithms and their tasks for characterization of errors. The classified dataset by AIS has been used during the NN weight training step that is performed in the initialization phase. Then, we introduce a parallel structure of the BAM neural networks that has been employed in both initialization and regeneration phases.

The proposed framework for generating cryptographic key and robust helper data are shown in Fig.1. In Fig.1, the AIS algorithm has responsibility for detecting and classifying the most important PUF hardware occurred errors. The blocks inside the dashed lines, Fig.1, represent the stability method. We have employed one of the main immune methods for characterization of error bits. The applied AIS algorithm is Negative Selection Algorithm (NSA). The combination of NSA with binary representation and Hamming Distance (HD) matching rule achieves smaller detector storage complexity and potentially better detection time. The classified data sets provide source of the training data for the neural network as shown in Fig.1. AIS can work as an intelligent teacher for NNs and prepares the required data for the training step. Helper data is the combination weights of the NNs weight matrices and some classified information from the AIS algorithms, which is prepared for public access and regeneration phase.

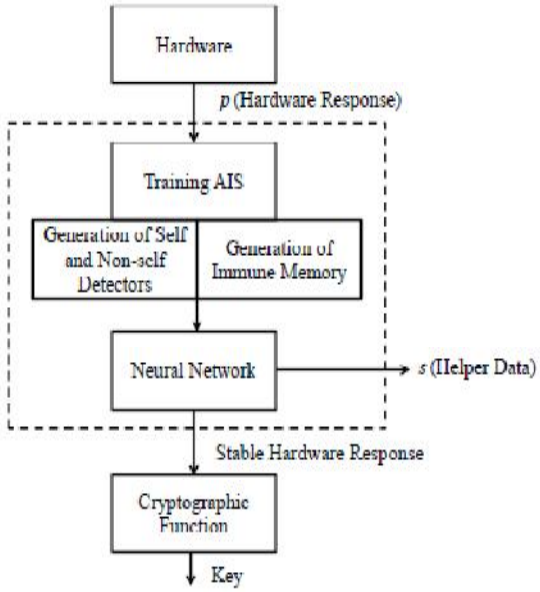


Fig.1. The proposed AIS-NN ECC framework

Basically designing an error characterization system based on AIS is equal to generating efficient detectors and has two major phases including detector generation and class detection. In detector generation step, Detectors are generated and trained to detect any error bits. In our algorithm, two major classes of antibodies are generated to minimize the required memory to store normal and abnormal samples. Every input response is compared to samples of both classes, so each positive antibody is replacement for several normal samples and each negative antibody is replacement for several error samples in dataset.

We take advantage of Negative Selection Algorithm (NSA) in our approach to generate more efficient detectors. Similar to NSA, if we use Positive Selection Algorithm (PSA), then in the detector generation phase, the detector candidates are generated by some random processes and matched against the given self sample set S . The candidates that do not match any element in S are eliminated and the rest are kept and stored in the detector set D. In the detection phase, the collection of detectors are used to distinguish self from non-self. If incoming data instance matches any detector, it is claimed as self. In NSA and PSA, an essential component is the matching rule which determines the similarity between detectors and self samples (in the detector generation phase) and coming data instances (in the detection phase). Obviously, the matching rule is dependent on detector representation. In this paper, we assume binary representation for all detectors and data (Hardware response bits). Since each antibody and antigen in our algorithm is a vector of binary valued fields, HD measurement which is simple to

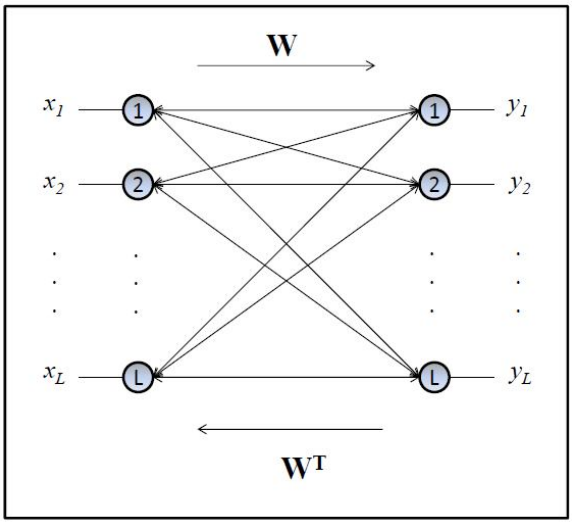
implement and easy to imagine is used for affinity calculation. We also control overlap between detectors by calculating HD between generated detectors and test vector, to make sure that minimum efficient number of detectors are generated. Assuming no overlap between detectors, make some parts of non-self space abandoned without any detector. These parts are called holes.

A parallel structure of RNNs has been employed for error correction process. Different structures of the RNNs have been investigated in this research but we find that BAM has superior performance in comparison with other networks such as bipolar network, supervised learning, easy implementation, and less complexity. BAM NN has capabilities of learning and memorizing of input/output patterns but their memorizing capacity and complexity should be considered during the implementation step. Our aim is to achieve high level of accuracy and security from the implemented AIS-NN that makes our approach being competitive with the classical error correction methods. Therefore, we have chosen a parallel network of BAMs and apply a supervised learning scheme that has been provided by AIS.

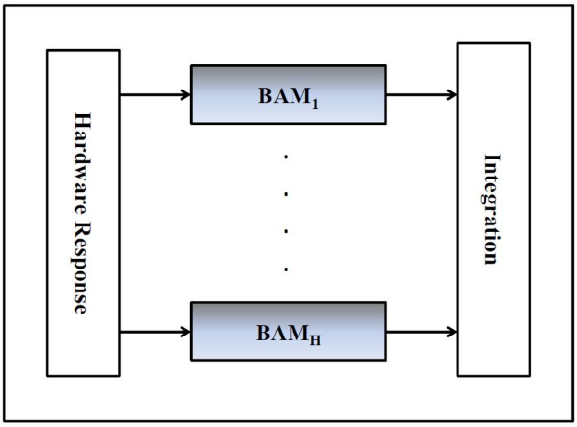
BAM is introduced by Kosko. The simplified model for BAM is shown in Fig.2 (a). The model which is shown in Fig.2 (a) has L nodes and the output of each node i is back to all other nodes j through the weights w_{ij}. Hebbian learning can be used to compute an adequate matrix W. The connection weights are set as,

$$W = \{ w_{ij} \} = \sum_{m=1}^{M} Y_m X_m^T, \qquad (1)$$

where w_{ij} is the connection weight from node i to node j; X_m , and Y_m are the input and output pattern vectors with the model pattern m , and M is the number of patterns. We denote the L×L weight matrix of the network by W and the weights are symmetric, i.e. $w_{ij} = w_{ji}$. Many enhancements to Kosko original network have been proposed on learning convergence, online learning and progressive recall to occur. Moreover, recent works have enabled BAMs to deal with multivalued stimuli in addition to bipolar (binary) stimuli. These developments directly increase the BAM's modeling capacities and range of applications. We apply a parallel structure of H BAMs to increase the accuracy of the error correction scheme. The overall structure for parallel BAMs is demonstrated in Fig.2 (b). In the process of parallelism and integration, two or more bipolar binary vectors are connected in series, and thereby the length of the integrated vectors becomes the sum of the lengths of those individual vectors that are aggregated together. If we have equal length vector for each BAM unit, $N_1 = N_2 = ... = N_H$, then the total length of the overall network will be L=HN accordingly.



(a)



(b)

Fig.2. The architecture of the RNN: (a) Simplified model for BAM. The equal number of nodes has been chosen for the input and output nodes, therefore the weights are symmetrical. (b) Parallel structure of BAMs. It improves storage capacity, error correction ability, reliability, and security of the generated helper data.

There are M bipolar binary vectors stored in the network, and the length of each vector, due to integration, increases to L=HN bits long which is equal to the sum of the lengths of vectors in both sets, X and Y . The increased length increases resolution among the stored vectors, and thereby enhancing the network performance. In the parallel structure of BAMs, the retrieval process is initiated by using simultaneously in parallel multiple probe vectors, and the desired pairs are retrieved simultaneously in parallel. Due to the integration process, the resolution greatly improves among the stored vectors. Therefore, performance of the recall process in Fig.2 (b) is far superior to that of any basic associated memory even if a single probe vector, which may be incomplete or contains some errors, to be used to initiate the retrieval process. We provide test results for the overall cooperative AIS-NN system in the result section to compare the performance of the error correcting scheme using a parallel structure of BAMs. BAMs weight matrices and some classified data from the AIS data set generate a powerful helper data.

Thus, there is no logical connection between this generated helper data and Hardware responses which means that security of the helper data or cryptographic key will be satisfied. The parallel structure of BAMs shows some advantages in comparison with single BAM such as increasing accuracy of the generated Hardware response and security of the generated helper data.

III. RESULTS AND DISCUSSION

In this section, we provide the implementation results for the proposed algorithms that has been employed to provide a secure and immune Hardware key. We discuss about important details regarding the stability of the hardware response bits using the cooperative AIS-NN mechanism to evaluate the system under various environmental conditions and change of supply voltage. We bring a comparative analysis to approve the advantages of the proposed scheme for the purpose of cryptographic key generation.

In the first attempt of our research, we implement the AIS algorithms offline using MATLAB 2013a software. The primary AIS defined task is to collect all of the required datasets for the preprogramming step of the RNN before releasing the key. The error analysis of the sample circuit has been accomplished using AIS algorithm. It means the AIS algorithm has been applied to characterize the hardware from the viewpoints of response occurred errors during change of temperature and supply voltage. The responses are clustered to the requested subsets and then applied for the training step of the neural network. Note that only a binary space for the self and nonself space is considered in this paper. This algorithm requires generating a number of candidate detectors. The memory set contains the most common occurred errors according to the defined affinity number from the collected responses. The number of iterations and affinity in the NSA algorithm are found heuristically to obtain the optimal number of antibodies.

The following experiments are conducted to evaluate the performance of the detector generation algorithm proposed in this paper. Every experiments runs 10 times independently. Also the experiments are conducted to estimate the average matching number for generating one detector. All candidate detectors are generated at random and some of them are removed because of matching one or more self strings. In the introduced algorithm, the basic operator is the matching operator between the self string and the candidate detector (or the candidate detector template). Therefore, the average matching number for generating one detector can reflect their time cost experimentally.

In all experiments, the size of the test set is denoted by N_T. Notably, the test set consists of different anomaly strings, and they are generated randomly one by one. That is to say, if an anomaly string is identical to any one of the test set, it cannot be added into the test set. Suppose the length of string is l. An anomaly string in the test set is generated according to the following steps.

(1) The random function is used to generate an integer between 0 and $2^{l}-1$ directly, then transform this integer into a binary string.

(2) If this binary string matches any self individual or anyone in the test set, go to (1). Or add this binary string into the test set.

When the length of string is l and the matching length is ε; a self string with Hamming distance can cover $\binom{l}{r}$ strings. Therefore, the size of self set is relatively small. Otherwise, the self set is prone to covering the whole space, and both the detector set and the test set are difficult to be generated. In the experiments, G_M represents the matching times between all candidate detectors and the self individuals during the generation of detectors.

The parallel structure of BAMs has been employed as a correction unit. This error correction process is initiated for a 128 bit vector, which means 128 bit vectors are used for training and testing the network. We found that the parallel structure of BAMs has three considerable advantages: lower error failure rate, fast response, and highly secure helper data. For the parallel BAMs, four sets that consist of M vectors (according to the number of patterns that the network has to be memorized) and the length of vectors in each set is N bits long are used. These four sets are integrated together to form a set of M, 32 bit long vectors (L=4N =128). The error is induced at random positions in the original vector using the information that has been collected by applying AIS algorithm. To test the proposed ECC scheme, test vectors are applied to the parallel BAM network. Trained networks have successfully corrected faulty test vectors. The error correcting helper data is calculated for a group of PUF outputs rather than one single output vector. All the vectors used to train the network can use the same error correcting helper data to correct the noisy bits. The weighting matrices of the BAMs is then combined together to generate a secure helper data.

Bit Error Rate (BER) for the implemented BAM is 50%, which means that the network can recover the PUF response when half of the response bits are error bits. We provide the average bit error rate at different voltage and temperature conditions which is summarized in Table 1 for two simple PUFs, Ring Oscillator PUF (ROPUF) and SR Latch PUF. The worst case happens at the lowest supply voltage and high temperature. The failure rate of the overall error correction unit using parallel BAMs is negligible.

When the application of PUF is cryptographic key, another important issue is to have a secure helper data from an ECC. The provided ECC helper data from a parallel BAM can solve this problem. The design strategy for generating ECC helper data which is based on weight matrix from a recurrent neural

network with random and no correlation matrix elements produces the required security for the implemented system. Also, generating a combined helper data from multi-parallel BAMs instead of one BAM increases the security of the proposed scheme accordingly.

Table 1: Stability of the proposed cooperative AIS-NN for two classes of PUF circuits under different conditions of temperature and supply voltage variation tests.

Test Condition	(1.5 V, 25 °C)	(1.1 V, 25 °C)	(1.8 V, 25 °C)	(1.1 V, 80 °C)	(1.8 V, 80 °C)
Average ROPUF Response Bit Error Rate %	6.12	5.12	5.68	7.04	7.25
AIS-NN ECC Failure Rate (ROPUF) %	0.61	0.64	0.64	0.85	0.88
Average SR Latch PUF Response Bit Error Rate %	7.2	7.8	7.6	9.4	9.6
AIS-NN ECC Failure Rate (SR Latch PUF) %	0.71	0.74	0.79	0.92	0.95

CONCLUSIONS

In this paper, the design methodology of an intelligent error detection/correction approach for generating a hardware-based cryptographic key has been explained. It is shown that cooperation of AIS and NN prepares enough accuracy, stability, and security for generating a secure ECC. AIS is an adaptive intelligent method for error detection using NSA and PSA algorithms, and also provides immune memory. Detector generation, classification, and immune memory make AIS a powerful cooperative tool for other computational intelligence methods in order to improve accuracy, performance, security, and convergence. AIS also constructs a supervised learning method for the chosen parallel structure BAMs, thus different classes of errors can be monitored and classified accordingly.

BAM network is used because of its simple implementation structure and ability to be trained by supervised learning. In comparison with the layered neural networks, BAM is better at dealing with discrete binary data and providing associated memory. BAM takes insignificant teaching (weight assignment) time for a huge amount of teaching sample data, and can recall a known pattern accurately and instantaneously. The weighting matrix of BAM prepares a secure error correction helper data. There is no identical specification in the achieved BAM weighting matrix, thus the overall security of the coding scheme is considerable.

REFERENCES

[1]. M.-D. M. Yu and S. Devadas, "Secure and robust error correction for physical unclonable functions." IEEE Design and Test of Computers, vol. 27, no. 1, pp. 48–65, 2010.

[2]. R. Maes, D. Schellekens, and I. Verbauwhede, "A pay-per-use licensing scheme for hardware ip cores in recent sram-based fpgas," IEEE Transactions on Information Forensics and Security, vol. 7, no. 1, pp. 98–108, 2012.

[3]. G. E. Suh and S. Devadas, "Physical unclonable functions for device authentication and secret key generation," in Proceedings of the 44th annual Design Automation Conference, San Diego, CA, Jun. 2007, pp. 9–14.

[4]. C. Berrou and V. Gripon, "Coded hopfield networks," in Proceedings of 6th International Symposium on Turbo Codes and Iterative Information Processing, Brest, France, Sept. 2010, pp. 1–5.

[5]. S. Pappala, M. Niamat, and W. Sun, "Fpga based trustworthy authentication technique using physically unclonable functions and artificial intelligence," in Proceedings of International Symposium on Hardware-Oriented Security and Trust (HOST), San Francisco, CA, Jun. 2012, pp. 59–62.

[6]. M. M. Htay, S. S. Iyengar, and S.-Q. Zheng, "Correcting errors in linear codes with neural network," in Proceedings of the Twenty-Seventh Southeastern Symposium on System Theory, Lexington, VA, Mar. 1995, pp. 386–391.

[7]. J. Bruck and M. Blaum, "Neural networks, error-correcting codes, and polynomials over the binary n-cube," IEEE Transactions on Information Theory, vol. 35, no. 5, pp. 976–987, 1989.

[8]. J. Yuan, V. Bhargava, and Q. Wang, "An error correcting neural network," in Proceedings of IEEE Pacific Rim Conference on Communications, Computers and Signal Processing, BC, Canada, Jun. 1989, pp. 530–533.

[9]. D. Dasgupta and F. Nino, Immunological Computation: Theory and Applications, 1st ed. Boston, MA, USA: Auerbach Publications, 2008.

[10]. A. Graves, Supervised Sequence Labelling with Recurrent Neural Networks. Springer, 2012, vol. 385.

[11]. G. Palm, "Neural associative memories and sparse coding," Elsevier, Neural Networks, vol. 37, pp. 165–171, 2012.

[12]. B. Kosko, "Adaptive bidirectional associative memories," Applied optics, vol. 26, no. 23, pp. 4947–4960, 1987.

[13]. A. P. Engelbrecht, Computational Intelligence: An Introduction, 2nd ed. Wiley Publishing, 2007.

[14]. G. Costa Silva, R. M. Palhares, and W. M. Caminhas, "Immune inspired fault detection and diagnosis: A fuzzy-based approach of the negative selection algorithm and participatory clustering," Expert Systems with Applications, vol. 39, no. 16, pp. 12 474–12 486, 2012.

[15]. S. T. Powers and J. He, "A hybrid artificial immune system and self organising map for network intrusion detection," Information Sciences, vol. 178, no. 15, pp. 3024–3042, 2008.

[16]. C. Yan, G. K. Venayagamoorthy, and K. Corzine, "Hardware implementation of an ais-based optimal excitation controller for an electric ship," IEEE Transactions on Industry Applications, vol. 47, no. 2, pp. 1060–1070, 2011.

[17]. B. Haktanirlar Ulutas and S. Kulturel-Konak, "An artificial immune system based algorithm to solve unequal area facility layout problem," Expert Systems with Applications, vol. 39, no. 5, pp. 5384–5395, 2012.

[18]. S. Forrest, A. S. Perelson, L. Allen, and R. Cherukuri, "Self-nonself discrimination in a computer," in Proceedings of IEEE Computer Society Symposium on Research in Security and Privacy, Albuquerque, NM, May 1994, pp. 202–212.

[19]. Z. Ji and D. Dasgupta, "Revisiting negative selection algorithms," Evolutionary Computation, vol. 15, no. 2, pp. 223–251, 2007.
[20]. X. H. Nguyen, C. M. Luong et al., "A novel combination of negative and positive selection in artificial immune systems," in Proceedings of the International Conference on Computing and Communication Technologies, Research, Innovation, and Vision for the Future (RIVF), Hanoi, Vietnam, Nov 2013, pp. 6–11.
[21]. I. Idris, S. M. Abdulhamid, A. Mohammed, U. D. Suleiman, and N. Akosu, "Email spam detection generation algorithm for negative selection algorithm with hamming distance partial matching rules." Australian Journal of Basic and Applied Sciences, vol. 8, no. 6, pp. 21–29, 2014.
[22]. F. ZHANG and D. QI, "A positive selection algorithm for classification," Journal of Computational Information Systems, vol. 8, no. 1, pp. 207-215, 2012.
[23]. L. N. De Castro and F. J. Von Zuben, "Artificial immune systems: Part i–basic theory and applications," Universidade Estadual de Campinas, Dezembro de, Tech. Rep, vol. 210, 1999.
[24]. I. Idris and A. Selamat, "Improved email spam detection model with negative selection algorithm and particle swarm optimization," Elsevier, Applied Soft Computing, vol. 22, pp. 11–27, 2014.
[25]. B. Kosko, "Bidirectional associative memories," Systems, Man and Cybernetics, IEEE Transactions on, vol. 18, no. 1, pp. 49–60, 1988.
[26]. B. Prasad, P. K. Prasad, S. Yeruva, and P. S. R. Murty, "A study on associative neural memories," International Journal of Advanced Computer Science and Applications, vol. 1, no. 6, 2010.
[27]. A. A. Bhatti, "A model of an intraconnected neural parallel bidirectional memory," in Proceedings of International Joint Conference on Neural Networks, IJCNN 2009, Atlanta, GA, Jun. 2009, pp. 316–323.
[28]. D. Zhang and S. Chen, "A novel multi-valued bam model with improved error-correcting capability," Journal of Electronics (China), vol. 20, no. 3, pp. 220–223, 2003.

★★★