CHAPTER I

ERROR DETECTION AND CORRECTION IN NANOSCALE MEMORIES

1.1 INTRODUCTION

Communication system is meant for exchange of information from source to destination, generally over long distances using either wired or wireless channels. There is every possibility that the signal transmitted gets corrupted by the noise existing in the channel. Hence, for error free communication along with the minimum power consumption as desired in the wireless channels, channels coding should be employed. This adds the bits and increases the length of the message. Hence, the coded message is called as code vector. At the receiver, these redundant bits are to be removed and the original message should be extracted, after using these bits for error checking. In this context, there are many types of coding techniques for error control for different type of channels for different types of applications. The recent trends in communication technology is to design the hardware with minimum chip area, should consume least power along with high speed decoding, best performance, approaching the maximum channel capacity.

Coding Theory uses either a Hard-decision or Soft-decision based decoding for detecting and correcting data which has been encoded at the transmitter. The Hard-decision decoding uses data that has fixed values (‘0’ and ‘1’ in case of binary values) and compares against a threshold value to arrive at a decision of the data value whereas
Soft-decision decoder uses a complete range of values between the fixed values also to arrive for better estimation of the data. Hence, this scheme is better than hard decision scheme. In this research, a soft decision based decoding has been employed. The Turbo codes and LDPC codes are soft decision based error correcting codes.

This research focuses on the error detection and correction in nanoscale memories for which the suitable scheme is developed keeping the following factors in mind: complexity, suitability of VLSI implementation and progressive decoding. In nanoscale memory, while retrieving the data, the random burst type of errors require controlling with the proposed Projective Geometry based LDPC code, satisfying the above criteria. Hence, the relevant topics are briefed.

1.2 LOW DENSITY PARITY CHECK CODES

LDPC code introduced by gallager[1] is a linear block code, suites to the requirements mentioned above i.e., high performance, simple hardware complexity, maximum channel capacity. Shu Lin and Costello Jr in [2] discussed in detail about the construction of the linear block codes.

The three most important advantages of LDPC codes over turbo codes are:

1. Absence of low-weight code words.
2. Iterative decoding of lower complexity.
3. Allow parallel implementation-enabling high data rates.
LDPC codes are efficient used at very low SNRs. Generally, like in Satellite downlinks, are constrained by the power. The heavy weight of the components such as On-board batteries, the solar cells adds to the costs of the launching. Traditionally, evolutonal codes are used for error control in these kind of applications. Recently, the Turbo, LDPC codes have become alternatives. Since, the power consumption is a serious problem in wireless communications, for example, in the cell phones, the personal digital Assistants, laptops etc., one needs a better code such as LDPC to improve the life of the battery.

Carefully constructed rate $\frac{1}{2}$ irregular LDPC code of length $10^7$ achieves a BER (bit error rate) of $10^{-6}$ approximately. The applications of these codes extend from optical, DVB, magnetic storage etc.

1.3 PROJECTIVE GEOMETRY BASED LDPC CODES

Codes based on Projective Geometry of dimension 2 have appreciable minimum distance properties very importantly, the bipartite graph has a better girth of 6 as against to 4 in simple LDPC codes. The proof for this is available in the book authored by Shu Lin & Daniel J Costello [2]. This family of LDPC codes are cyclic in nature and as such can be encoded using a series of shift registers with appropriate feedback. They can also be orthogonalized to have simple decoding scheme using majority logic strategy, which is suitable for certain types of cyclic codes. Projective geometry based codes originate from the incidence relationship between the nodes and edges of the Tanner graph or in other words, the points and lines (as...
shown in fig.1.1) of a 2-dimensional projective space over GF(p) or its extension GF(p^t) where p is prime and t>0 [2].

![Fig.1.1 Points and lines in Projective Geometry](image)

A binary PG code C over GF(2^s) with m dimensions is represented as PG(m, 2^s). The length of the code, n = (2^(m+1)s - 1)/(2^s - 1) is the cyclic code having its null space, the incidence matrix H of PG(m, 2^s). Since column weight of H is (2^{ms} - 1)/(2^s - 1), the minimum distance of C = (2^{ms} - 1)/(2^s - 1) + 1. Since the H matrix is sparse, the code C is called PG-LDPC code.

### 1.4 p-adic EXTENSIONS OF PG-LDPC CODES OVER PG(2, 2^s)

The p-adic (poly adic) PG codes are derived from the PG-LDPC codes over PG(2, 2^s) and are designated as PG(2, 2^s, 2^t) codes. In PG(2, 2^s, 2^t) codes, the first parameter is 2 indicating 2-Dimensional projective geometry is being considered. The parameter 2^s indicates that subfield of binary PG code (parent code) is GF(2^s). The parameter 2^t indicates that p-adic integers are from the ring mod(2^t).

Any cyclic code over Z_q, with length n, is a linear code, having the property that if (c_0, c_1, ..., c_{n-1}) belongs to C, then the cyclically shifted set (c_1, c_2, ..., c_{n-1}, c_0) also belongs to C. Generally codeword’s are
represented as polynomials. Hence, cyclic codes are the ideal in the ring $R = \mathbb{Z}_q[x]/(x^n-1)$. To study the $p$-adic codes over $\mathbb{Z}_q$ as explained by Calderbank in [3] and Gorgui Naguib in [4] on $p$-adic numbers, the properties of the ring $R$ need to be studied and also certain Galois rings $\text{GR}(q^m)$. This ring $\text{GR}(q^m)$ can be cyclic multiplicative sub-group of the order $m$ over $\mathbb{Z}_q$ in $R$.

The lifting method adopted, is a generalization of Graeffe’s method used for finding the primitive polynomial for codes over $\mathbb{Z}_q$ is demonstrated by W.C.Huffman and Vera Pless in the page 479 of the book [5]. This method is recursively applied $(t-1)$ times to obtain the generator polynomial and the parity check polynomial for the $p$-adic PG codes over $\text{PG}(2, 2^s, 2^t)$. Parity check polynomial is the polynomial obtained by lifting the first row of the incidence matrix of $\text{PG}(2, 2^s)$ expressed in polynomial form (LSB first) $(t-1)$ times. The error detection and correction in Nanoscale memories requires a byte (symbol) correction using $p$-adic extension of PG based LDPC codes for handling burst type of errors.

1.5 NANOSCALE MEMORY

The photolithographic based fabrication of components below 32 nanometers is challenged by technology and economic barriers as reported by Shalini Ghosh and Lincoln D in [6]. Silicon nano wires or carbon nano tubes (self-assembled devices) achieve these dimensions, addresses power and other issues in system architecture, but the fabrication costs goes up. It is really a challenging task to assemble
the best-quality, large-scale circuits of nanoelectronic type. The bigger challenges are too many defects along with high fault rates take place in these devices and are susceptible to the soft type of errors than microscale devices. Shalini Ghosh and Lincon [6] in the year 2007, came up with suggestions to reduce the soft errors happening at the memories.

1.6 EXISTING TECHNOLOGY

Today’s micro scale devices, for example gates, memories and Programmable Logic arrays etc., built with a technique of ‘top-down lithographic’ has error rates around 1%. But, the components made using nanoscale elements such as the bistable, carbon nanotubes, wires do have an error rate of a large value, close to 10%. Soft error correction is critical for Storage devices (nano memory), for Computation (nano-ALU), Communication (nanoscale signal transmitters and receivers).

For static defects, there can be an alternative solution. Fig.1.2 shows the Nano PLA architecture as detailed by Shalini Ghosh and D Lincoln.
1.7 FORMULATION OF THE PROBLEM

The architecture that is suitable for this purpose is nano-PLA based blocks and simple gates of nanotype construction, for example the universal gates and majority gates as the design components. Error correction at nanoscale memories suffers from the following problems:

- Low error detection and correction
- High soft error rates
- High computational overhead
- No modular approach for the transmitter’s encoder and the receiver’s decoder blocks for simplification to minimize the hardware.
• Currently, there is no provision for selective switching of enabling blocks as well as the disabling of the decoder blocks to gain the important power savings.

• Dynamically, the error-correction should be done to adopt the proper trade-off between the required error correction level and the performance of the system. Such provision is currently not available.

For nanoscale memories possessing high fault rates, a new scheme of error correction is required, since the existing conventional ECC techniques are not suitable because of their huge decoding complexity and non-scalability. For example, conventional codes like Reed-Solomon or BCH to achieve higher error-correcting capability require more sophisticated decoding algorithms. These algorithms need complex Galois field arithmetic, not a practical idea for the kind of the nano scale size that one is looking for in these PLAs. Moreover, they are not scalable. The complexity of Reed Solomon codes grows exponentially with the code length as detailed in the patent in [7]. Another prerequisite for the ECC applicable for nano scale memories is reconfigurability.

1.8 STRATEGY

The system architecture of the proposed nanomemory using PG-LDPC as Error Correction Codes is shown in the fig.1.3. During a write operation, the incoming word is encoded and the code word is buffered. While retrieving a code word from the memory during a
read operation, is being checked using the checker unit. The logic unit based on the majority uses the syndrome to decode for appropriate error detection and error correction. The micro level control circuit optimizes the decoding circuitry for power saving.

![System architecture of the Nanoscale memory with PG-LDPC ECC](image)

The behaviour of the circuitry is such that the there will be more fault rates in the infant mortality and when the circuitry gets aged as the parts become more unreliable. But during the normal operation i.e., when the middle age part of the parts of the circuitry, the fault rate decreases. One would like to have higher capacity of error detection and correction during the infant and the aged stages of the parts such that the unused part is disabled as the rate of faults are significantly less and hence save the power. The PG(2, $2^s$, $2^t$) codes are amenable for reconfigurability. The analysis of the algorithm for PG(2, $2^s$, $2^t$) codes suggests that they efficiently sort out the above issues and therefore become good candidates for being used in burst and
random error corrections in nano-scale memories, in place of Reed Solomon Codes

1.9 OBJECTIVES OF THE PROPOSED WORK:

The research work focuses on the PG(2, 2^s, 2^t). Since these codes are cyclic in nature, encoding can be implemented using simple shift registers as in the case of Reed Solomon codes. The difference between PG and RS codes is that instead of complex Galois field arithmetic, PG uses arithmetic over 2-adic integers which is very simple to implement. The decoding architecture is based on the MLD strategy. The thesis addresses the following objectives:

1. To highlight the various LDPC decoding methodologies.
2. To describe procedure to construct 2-dimensional projective space over GF(2^t) where t >0 and extend it to construct LDPC codes over PG(2, 2^s, 2^t).
3. To design Encoder and Decoder (using majority logic decoding strategy) and propose a new bit serial architecture for symbol/byte decoding (for burst errors).
4. To develop a generic conflict free PG based LDPC decoder as an application to fault tolerance in computer network systems that adopts to both regular and irregular LDPC codes.

1.10 METHODOLOGY

1. The above objectives are based on lifting the basic PG(2, 2^s, 2^t) LDPC codes using Graeff’s technique. The Graeff’s technique
is well documented in the literature, page 479 of [5]. Lifting can be done on the fly to account for different duration of error bursts, which is the unique method.

2. The construction procedures for the objective 2, includes getting points and lines of \( \text{PG}(2, 2^s, 2^t) \) as given in [2] and then constructing the incidence matrix. The elements of the incidence matrix are over \( \text{GF}(2) \), Generator polynomial, Generator matrix and H-matrix for the codes.

3. For objective 3, newly proposed Bit Serial Decoding architecture (well suited for VLSI implementation) is proposed. It handles symbol / byte instead of bit and is independent of the amount of ‘lifting’ done. Progressive decoding is possible. (Suitable for Audio/Video applications)

4. The Decoder design is simplified to be of linear complexity as the exponent ‘t’ of the 2-adic integers grows linearly. This is in contrast to the exponential complexity of the RS Decoders for the burst type of error detection and correction as applied in nanoscale memory. [7].

5. The generic conflict free PG based LDPC decoder (as an application to fault tolerance in computer network systems that adopts to both regular and irregular LDPC code) as explained by Balaji Janikiram, Girish, B. Suryanarayana Adiga and others in [8] adopts a newly proposed concept called zero–free Perfect Difference Networks (PDN).
1.11 Organization of The Thesis

- The **Chapter 1** provides a brief introduction of error detection and correction in the context of Nanoscale memories with PG LDPC Codes, problem definition, objectives, methodology and the organization of the dissertation.

- The **Chapter 2** gives an introduction to Literature survey on LDPC codes, PG LDPC codes, Majority logic decoding, p-adic concept, Reed Soloman codes, complexity analysis of RS codes, Perfect difference Networks etc.

- The **Chapter 3** describes the LDPC codes, the encoding of LDPC codes and various decoding algorithms.

- The **Chapter 4** focuses on the Projective Geometry. Concepts of points and lines to construct Tanner Graph on the lines of check nodes and variable nodes of LDPC codes. Describes procedure to construct 2-dimensional projective space over GF(2^t) where t>0. Construction of Generator polynomial, Generator matrix and H-matrix for the PG LDPC codes. The working of algorithm 2DPG and its output is listed.

- The **Chapter 5** describes one-step Majority logic decoding, suitable for decoding p-adic based PG LDPC codes using the majority voting. The algorithm of the majority logic decoding and its output is listed.

- In the **Chapter 6**, introduction of p-adic arithmetic, lifting scheme to covert binary bits to symbols with the help of ring size are discussed. It describes procedure to design (newly
proposed) decoder using majority logic strategy to decode P-adic extended PG based LDPC codes.

- The **Chapter 7** discusses simulation of encoder and decoder and explains the operation of decoder for symbol error detection and correction.

- The **Chapter 8** proposes conflict free generic PG LDPC decoder for fault tolerance in single node/link failure in computer networks of diameter 2 based on Perfect Difference Network(PDN) using a concept called 0-free PDN.

- The **Chapter 9** describes the results, lists the conclusions, limitations and future proposal of the overall work.

- The **Appendix A** covers the detailed complexity analysis of the RS codes. The nonlinearity complexity of these codes is established.

The research work is implemented in MATLAB 7.6.0 software. It uses communication tool box to select the minimum polynomials to generate the generator polynomial of PG based LDPC codes. Simulation has been done to verify the functionality.

### 1.12 Conclusions

The relevant topics such as LDPC, PG LDPC, majority logic decoding, p-adic numbers and error detection and correction in Nanoscale memory are briefed. The existing problem (Problem definition), Objectives, methodology and the organization of the dissertation are discussed. The next chapter deals with the literature survey of the above and related topics.