Design and Implementation of Error Correcting Codes for Storage and Communication Systems

Thesis submitted
by
RAJ KUMAR MAITY

Doctor of Philosophy (Engineering)

Department of Electronics and Telecommunication Engineering
Faculty Council of Engineering and Technology
Jadavpur University
Kolkata, India
2024

JADAVPUR UNIVERSITY
KOLKATA - 700 032, INDIA

INDEX NO. 152/19/E

Title of the Thesis:

Design and Implementation of Error Correcting Codes for Storage and Communication Systems

Name, Designation and Institution of the Supervisors:

1. Prof. Jaydeb Bhaumik
Professor
Department of ETCE, Jadavpur University
188, Raja S C Mallik Road, Jadavpur, Kolkata-700032

2. Dr. Jagannath Samanta
Associate Professor
Department of ECE, Haldia Institute of Technology
ICARE, Hatiberia, Haldia, Purba Medinipur, West Bengal, 721657

INDEX NO. 152/19/E

List of Publications

International Journals:

1. **R. K. Maity**, S. Tripathi, J. Samanta and J. Bhaumik, "Lower Complexity Error Location Detection Block of Adjacent Error Correcting Decoder for SRAMs" *IET Computers & Digital Techniques*, 14 (05), 210-216, May, 2020. [link]

2. **R. K. Maity**, J. Samanta and J. Bhaumik, "Construction Technique and Evaluation of High Performance t-bit Burst Error Correcting Codes for Protecting MCUs" *Journal of Circuits, Systems and Computers*, 32 (09), 23050142, June, 2023. [link]

3. **R. K. Maity**, J. Samanta and J. Bhaumik "An Improved Single and Double-Adjacent Error Correcting Codec with Lower Decoding Overheads" *Journal of Signal Processing Systems*, 95 (06), 721-733, May, 2023. [link]

4. **R. K. Maity**, J. Samanta and J. Bhaumik,"Single and Double-adjacent Error Correcting Code (SDECC) with Lower Design Overheads and Mis-correction Rate for SRAMs" *Microsystem Technologies*, 29 (06), 823-834, May, 2023. [link]

5. **R. K. Maity**, A. Das, J. Samanta and J. Bhaumik, "Improved ABER Performance of FSO Communication Systems with Soft Decision Decoding of TAEC Code" *Under preparation.*

Book Chapters

1. **R. K. Maity**, J. Samanta, and J. Bhaumik "FPGA-Based Low Delay Adjacent Triple-Bit Error Correcting Codec" Singapore: *Lecture Notes in Electrical Engineering (LNEE, volume 825)*, 419-429, ISBN: 978-1-5225-9767-4,2022. [link]

2. **R. K. Maity**, J. Samanta, and J. Bhaumik "An Efficient Decoding of SEC-DED-DAEC Code with Odd-Column-Weight H-matrix" Singapore: *Lecture Notes in Electrical Engineering (LNEE, volume 825)*, 689-697, ISBN: 978-981-16-9153-9,2022. [link]

List of Presentations in National/ International Conference / Workshops:

1. **R. K. Maity**, J. Samanta, and J. Bhaumik, "An area and power efficient double adjacent error correcting parallel decoder based on (24, 12) extended golay code," Proc. *IEEE International Conference onComputer and Communication Technologies (ICECCT)*, SVS College of Engineering, Tamil Nadu, India, Feb., 2019. [link]

2. **R. K. Maity**, J. Samanta, and J. Bhaumik, "New Compact SEC-DED-DAEC Code for Memory Applications," Proc. *2nd International Conference on Communication, Devices and Computing (ICCDC)*, Haldia Institute of Technology, West Bengal, India, Jan., 2020. [link]

3. **R. K. Maity**, A. Das, J. Samanta, and J. Bhaumik "The Combined ECC Scheme with Outer 3-bit BEC and Inner SEC-DAEC Codes for FSO Communication System" in Proc. ICACA 2024. [link]

PROFORMA - 1

"Statement of Originality"

I, Raj Kumar Maity registered on 18^{th} **June, 2019** do hereby declare that this thesis entitled "Design and Implementation of Error Correcting Codes for Storage and Communication Systems" contains literature survey and original research work done by the undersigned candidate as part of Doctoral studies.

All information in this thesis have been obtained and presented in accordance with existing academic rules and ethical conduct. I declare that, as required by these rules and conduct, I have fully cited and referred all materials and results that are not original to this work.

I also declare that I have checked this thesis as per the "Policy on Anti Plagiarism, Jadavpur University, 2024", and the level of similarity as checked by iThenticate software is 5%.

Raj Kumar Maity

Signature of Candidate:
Date : 30/06/2025

Certified by Supervisor(s):
(Signature with date, seal)

1. Bhaumik 30/06/2025

Dr. Jaydeb Bhaumik
Professor
Dept. of Electronics and Telecommunication Engineering
Jadavpur University
Kolkata - 700032

2. Jagannath Samanta 30/06/2025

Dr. Jagannath Samanta, Ph.D. (Tech.)
Associate Professor
Dept. of Electronics and Communication Engineering
Haldia Institute of Technology

PROFORMA- 2

CERTIFICATE FROM THE SUPERVISOR

This is to certify that the thesis entitled "Design and Implementation of Error Correcting Codes for Storage and Communication Systems" submitted by Shri Raj Kumar Maity, who got his name registered on 18^{th} June, 2019 for the award of Ph. D. (Engg.) degree from Jadavpur University is absolutely based upon his own work under the supervision of Prof. Jaydeb Bhaumik, Professor, Department of ETCE, Jadavpur University, and Dr. Jagannath Samanta, Associate Professor, Department of ECE, Haldia Institute of Technology, and that neither his thesis nor any part of the thesis has been submitted for any degree/diploma or any other academic award anywhere before.

1. Prof. Jaydeb Bhaumik
Professor
Department of ETCE, Jadavpur University

J Bhaumik 30/06/2025

Dr. Jaydeb Bhaumik
Professor
Dept. of Electronics and Telecommunication Engineering
Jadavpur University
Kolkata - 700032

Signature of the Supervisor and date with Office Seal

2. Dr. Jagannath Samanta
Associate Professor
Department of ECE, Haldia Institute of Technology

Jagannath Samanta 30/06/2025

Dr. Jagannath Samanta, Ph.D. (Tech.)
Associate Professor
Dept. of Electronics and Communication Engineering
Haldia Institute of Technology

Signature of the Co-supervisor and date with Office Seal

ACKNOWLEDGEMENTS

I am incredibly grateful to my supervisor Prof. Jaydeb Bhaumik, Professor, Department of ETCE, Jadavpur University, Kolkata, West Bengal, India, and to my co-supervisor Dr. Jagannath Samanta, Associate Professor, Department of ECE, Haldia Institute of Technology, Haldia, West Bengal, India, for their valuable comments, suggestions and guidance throughout my research work.

I also express my heartiest graduate to the members of Research Advisory Committee (RAC), Prof. Mrinal Kanti Naskar, Professor, Department of ETCE, Jadavpur University, Prof. Manotosh Biswas, Professor and Head of the Department of ETCE, Jadavpur University, and the committee chair for their valuable advice.
I am also thankful to Haldia Institute of Technology, ICARE Complex, Hatiberia, Haldia, West Bengal, for providing me the computing facilities for carrying out my research works even on institute's holidays.
I would like to acknowledge and give my warmest thanks to SMDP Laboratory, University of Calcutta, Kolkata, West Bengal, for providing the Cadence tools.
I would also like to give special thanks to Dr. Prabir Saha, HOD and Associate Professor, Department of ECE, NIT, Meghalaya, Dr. Satyajit Das, Assistant Professor, Department of Computer Science and Engineering, IIT Palakkad, and Dr. Akianchan Das, Department of Electronics and Communication Engineering, Haldia Institute of Technology for their valuable cooperation to handle the difficulties faced during my research works.
For their editing assistance and emotional support, I am also appreciative of my colleagues at Department of ECE, Haldia Institute of Technology.
I wish to express my deepest sense of gratitude to my parents, wife, daughter, brothers, sisters and relatives. It is their blessings that enable me to complete the task.
Finally, I would like to thank God, for letting me through all the difficulties. I have experienced your guidance day by day. You are the one who let me finish my degree. I will keep on trusting you for my future.

ABSTRACT

The information storage and transmission are the two most important responsibilities of any modern communication or computing device. The consistent operations of computing or communication device requires reliable operation of both storage and communication systems. The radiation-induced soft errors have been considered as the foremost issues against the reliable operations of storage system. The soft errors are the main cause of cell upsets in storage systems, whereas the performance of communication system is degraded due to channel noise, fading and interference. The Error Correction Codes (ECCs) are the universally accepted solution for enhancing reliability of both the storage and communication systems. The ECCs are preferred mostly for the applications in storage and communication systems which have lower implementation costs in terms of area, delay, power consumption, redundancy, mis-correction rates and Bit Error Rate (BER).

In this thesis, several improved schemes for adjacent and burst error correction codes and their competent Very Large Scale Integration (VLSI) implementations have been introduced for storage and communication systems. In addition, the performance of the soft decision decoding based Triple Adjacent Error Correction (TAEC) code and the combined ECC scheme consists of the 3-bit Burst Error Correction (BEC) and Single Error Correction-Double Adjacent Error Correction (SEC-DAEC) codes have been thoroughly examined for the Free Space Optical (FSO) communication system. This thesis presents the design and efficient VLSI implementation of various SEC-DAEC codes with the goal of lowering the overheads associated with the circuit design of encoders and decoders for memory application. Further, as an effort to reduce design overheads and the rate of mis-correction, a number of SEC-DED-DAEC codes have been developed and implemented in VLSI. Also, SEC-DAEC-TAEC and t-bit BEC codes have been designed and implemented to reduce the overheads associated with codec design. These codes can be used for protecting MCUs in memory applications. Moreover, the performance of a soft decision decoder based TAEC code and the combined ECC scheme with outer 3-bit BEC and inner SEC-DAEC codes for FSO communication channel have been presented in this thesis.

Contents

List of Figures

List of Tables

List of Abbreviations

Abbreviation	Description
ABER	**A**verage **B**it **E**rror **R**ate
AE	**A**djacent **E**rror
AEC	**A**djacent **E**rror **C**orrection
ASIC	**A**pplication **S**pecific **I**ntegrated **C**ircuit
AWGN	**A**dditive **W**hite **G**aussian **N**oise
BCH	**B**ose- **C**haudhuri- **H**ocquenghem
BE	**B**urst **E**rrors
BEC	**B**urst **E**rror **C**orrection
BPSK	**B**inary **P**hase **S**hift **K**eying
CAM	**C**ontent **A**ddresible **M**emory
CM	**C**ache **M**emory
CMOS	**C**omplementary **M**etal **O**xide **S**emiconductor
DAE	**D**ouble **A**djacent **E**rrors
DAAE	**D**ouble **A**lmost **A**djacent **E**rrors
DAEC	**D**ouble **A**djacent **E**rror **C**orrection
DED	**D**ouble **E**rror **D**etection
EC	**E**rror **C**orrection
ECCs	**E**rror **C**orrecting **C**odes
EDA	**E**lectronic **D**esign **A**utomation
ELD	**E**rror **L**ocation **D**etection
FER	**F**rame **E**rror **R**ate

FPGAs	**F**ield **P**rogrammable **G**ate **A**rrays
FSO	**F**ree **S**pace **O**ptical
HDD	**H**ard **D**ecision **D**ecoding
HDL	**H**ardware **D**escription **L**anguage
LBC	**L**inear **B**lock **C**ode
LDPC	**L**ow **D**ensity **P**arity **C**heck
LUTs	**L**ook-up **T**ables
MAP	**M**aximum **A** **P**osteriori
MCUs	**M**ultiple **C**ell **U**psets
MIMO	**M**ultiple **I**nput- **M**ultiple **O**utput
MISO	**M**ultiple **I**nput- **S**ingle**O**utput
ML	**M**aximum **L**ikelihood
PEP	**P**airwise **E**rror **P**robability
QAE	**Q**uadruple **A**djacent **E**rrors
QAEC	**Q**uadruple **A**djacent **E**rror **C**orrection
RS	**R**eed **S**olomon
SC	**S**yndrome **C**omputation
SCU	**S**ingle **C**ell **U**pset
SDD	**S**oft **D**ecision **D**ecoding
SE	**S**ingle **E**rror
SEC	**S**ingle **E**rror **C**orrection
SIMO	**S**ingle **I**nput- **M**ultiple **O**utput
SISO	**S**ingle **I**nput- **S**ingle **O**utput
SRAMs	**S**tatic **R**andom **A**ccess **M**emories
TAE	**T**riple **A**djacent **E**rrors
TAAE	**T**riple **A**lmost **A**djacent **E**rrors
TAEC	**T**riple **A**djacent **E**rror **C**orrection
TCM	**T**rellis **C**oded **M**odulation
VLSI	**V**ery **L**arge **S**cale **I**ntegration

List of Symbols

$f_x(\cdot)$	: Probability Density Function (PDF)
$G_{p\ q}^{m\ n}[\cdot]$	: Meijer's G function
$\Gamma(\cdot)$	: Gamma function
$Q(\cdot)$	: Gaussian Q-function
$\mathrm{erfc}(\cdot)$	: Complementary error function
$\sum(\cdot)$	: Summation of the all variables

Chapter 1

Introduction

Error correcting codes (ECCs) are employed in Forward Error Correction (FEC) technique of error control strategies to detect and correct errors which are commonly occurred due to channel noise, fading and interference. In ECC, redundant bits are appended with the information bits during encoding process and these redundant bits are utilized by the decoder for error detection and correction purpose. The ECC is an integral part of most of the modern storage and communication systems for making these systems reliable.

In Complementary Metal Oxide Semiconductor (CMOS) memory, operating voltages and device dimensions are continuously decreasing due to the rapid advancements of VLSI technology. As a result, the storage systems are becoming more susceptible to the radiation-induced soft errors with the continuous down scaling of technology nodes. Data stored in one or multiple cells of any memory are corrupted by these soft errors. Corruption of stored data in only one memory cell is known as Single Cell Upset (SCU). Whereas Multiple Cell Upsets (MCUs) are the effects of soft errors in multiple cells of a memory. The ECCs are generally employed in memory systems for protecting these systems from soft errors. Single Error Correction (SEC) codes are appropriate to get rid of SCU. But codes with multiple error correction capabilities such as Bose Chaudhuri Hocquenghem (BCH) and Reed Solomon (RS) codes

are applied to mitigate the influences of MCUs in memories. However, for semiconductor technology nodes below 45nm, cell density increases significantly which in turn enhances the probability of multiple adjacent errors in memories. These multiple adjacent errors are corrected by applying Adjacent Error Correcting (AEC) and Burst Error Correcting (BEC) codes such as Single Error Correction-Double Error Detection-Double Adjacent Error Correction (SEC-DED-DAEC), Single Error Correction-Double Error Detection-Triple Adjacent Error Correction (SEC-DED-TAEC), 3-bit BEC and 4-bit BEC codes.

On the other hand, channel coding for error detection and correction helps the communication system designer to mitigate the effects of a noisy transmission. Error control coding has been widely used in all types of wired and wireless communication systems for many years. The main objective of employing ECCs in communication systems is to minimize the effects of channel noise, fading and interference in these systems and hence to improve the Bit Error Rate (BER) performance and maximum utilization of channel capacity. Various ECC schemes like Hamming code, Golay code, BCH code, RS code, Convolutional code, Turbo code etc. have been extensively explored for communication systems over many years. The ECC schemes with Soft Decision Decoding (SDD) have better BER performance and are mostly employed in communication systems. The recent addition of ECC scheme with SDD in the field of communication systems is the Polar Code (PC) which is a near Shannon's capacity approaching code and widely employed in the reverse channel of 5G communication standard. Another application of SDD is found in Concatenated Codes (CC) where two or more ECCs are connected in series for the betterment of BER performance of the communication systems. The coded Free Space Optical (FSO) communication channel are assumed as the backbone of beyond 5G (B5G) communication standards due to their free spectrum utilization, higher speed of operations and lower BER. Although plenty of ECC schemes are available in the literature but still the ECC schemes which are compact, fast, power efficient and provide better BER performance compared to the existing ECC schemes are highly demanded both for storage and communication systems.

1.1 Utilization of Error Correcting Code in Storage and Communication Systems

The Error Correcting Codes (ECCs) are utilized in storage systems mainly for the protection of stored data from soft errors. The block diagram of ECC employed in storage system has been shown in Fig. 1.1. The data bits to be stored in memory is first passed through the ECC encoder to obtain the codewords and then these codewords are kept in memory. Now, one bit or multiple bits in the stored codewords may be affected by soft errors in a memory. The ECC decoder reads the erroneous codewords from a memory location and provides the original data bits as output by detecting and correcting errors.

The error correcting codes are also essential in modern communication systems for

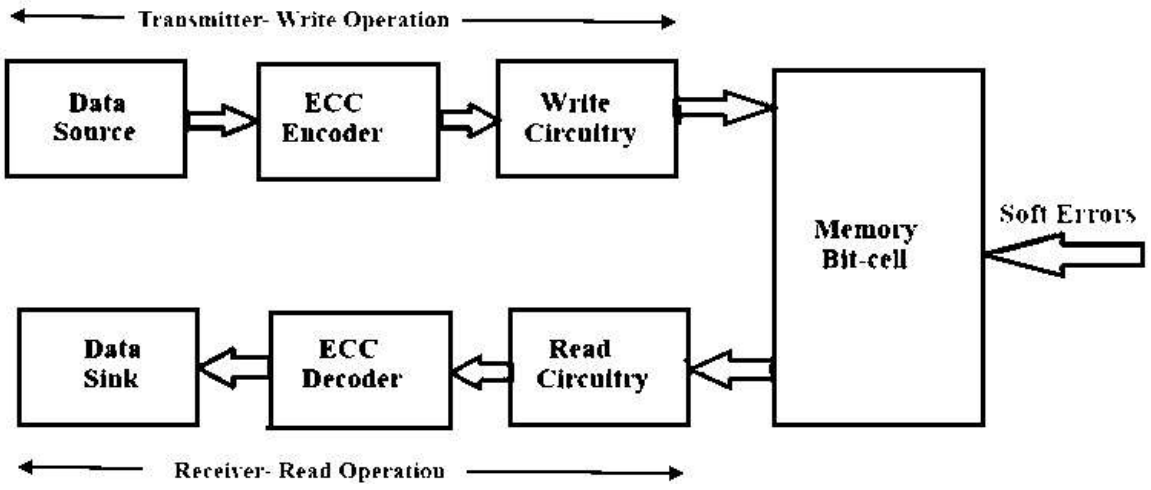

FIGURE 1.1: Typical Block Diagram of Storage System with ECC

detecting and correcting bit errors and hence making these systems reliable. The block diagram of typical communication system has been shown in Fig. 1.2. The information source provides the information to be transmitted over the communication channel. These messages are transformed into data bits by the source encoder.

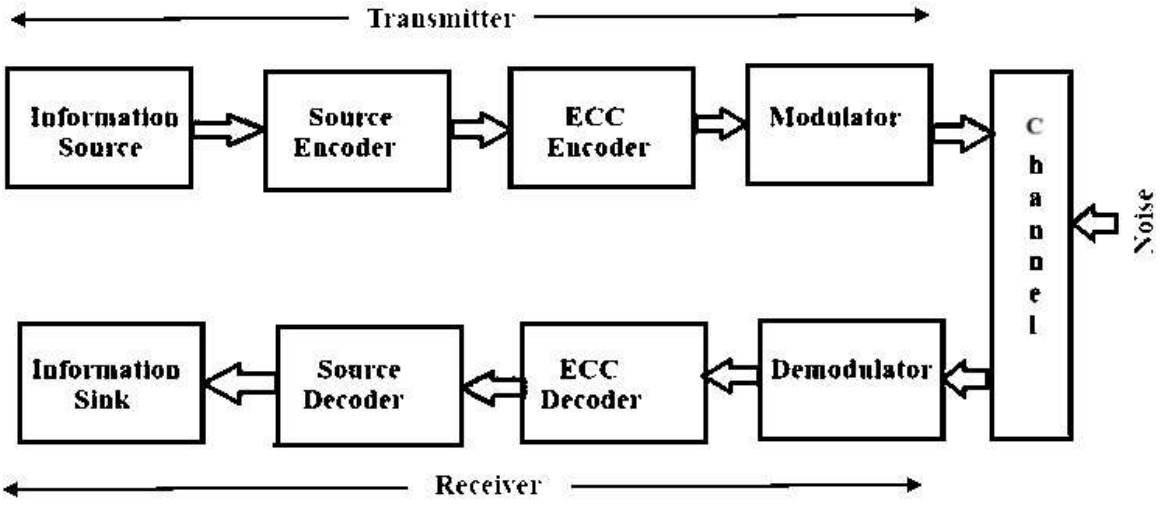

FIGURE 1.2: Typical Block Diagram of Communication System with ECC

These data bits are encoded by the ECC encoder to produce the codewords. The output of the modulator circuit is basically an electrical signal which is transmitted through the communication channel. The noise affects the transmitted signal in the communication channel and transmitted voltage levels are changed in a random manner to produce the received signals at the input of the demodulator. The demodulator converts the electrical received signal to the binary received codewords. These received codewords are decoded by the ECC decoder for recovering the original data bits. These data bits are converted to original messages as produced by the information source by the source decoder. The reconstructed data are received by the information sink.

1.2 Background of Error Correcting Codes for Storage and Communication Systems

Error Correcting Codes (ECCs) [1], [2], [3], [4] have become increasingly important in today's storage and communication systems for error detection and correction. The majority of contemporary electronic devices contain a storage system for the

purpose of storing vital data. Reliability of these storage systems are mostly required for the smooth execution of these equipment. But the soft errors [5], [6] are the main culprit behind the reliability of storage systems. In modern memories, these soft errors are generally induced by the radiation of various energetic particles [7], [8]. The memory cells upsets like Single Cell Upset (SCU) and Multiple Cell Upsets (MCUs) [9], [10] are frequently occurred in memory due to these soft errors. Designing stable and soft errors free memory cells like Static Random Access Memories (SRAMs) is very important for proper implementation of modern electronic devices [11], [12], [13], [14]. In 1984, Chen et al. [15] presented for the first time that Error Correction Codes (ECCs) could be applied in memories to mitigate the effects of soft errors. Accordingly, various Single Error Correction (SEC) and Single Error Correction- Double Error Detection (SEC-DED) codes [16], [17], [18] have been widely employed in memories to get rid of SCU in memory cells. The major advantages of these codes are simple encoder and decoder structures, lower redundancy and higher code rate. Also further simplifications of encoding and decoding operations of SEC and SEC-DED codes have been done by many researchers in recent years [19], [20], [21]. But main limitation of SEC and SEC-DED codes are their error correction capability which is applicable for correcting SCU in memory cells only. On the other hand, the occurrences of MCUs in memory cells have been augmented significantly in recent years due to the down scaling of technology nodes [22], [23], [24], [25], [26], [27], [28], [29].

As the first attempt of protecting MCUs, the interleaving of SEC-DED codes [30], [31], [32], [33] have been used for various memory applications. The interleaved SEC-DED codes have been obtained by physical separation of data and parity bits locations in the codeword such that errors can be corrected beyond the correction capability of original SEC-DED codes. But it has been found that interleaving of SEC-DED code affects the codec design overheads like area, delay and power usage. The interleaving is not realistic for all types of memories specifically for Content Addressable Memory (CAM) [34], [35]. The Built-in Current Sensors (BICS) [36], [37] have been applied along with the SEC-DED codes for handling double bit errors per

codeword caused by MCUs. Further, Matrix Code (MC) [38] and Decimal Matrix Codes (DMCs) [39], [40], [41], [42], [43] have been explored for memory protection against MCUs. But these codes suffer from higher redundancy.

The ECC schemes with superior error correction capabilities like Reed-Solomon (RS) codes [44], [45], [46], [47] and Bose–Chaudhuri–Hocquenghem (BCH) [48], [49], [50], [51] codes have also been tested for defending MCUs in memory application at the cost of excess decoding complexities. As a substitute, the Adjacent Error Correcting (AEC) codes are proficient of rectifying single and multiple adjacent errors with straightforward decoder structures and these codes are mostly applied for protecting MCUs in recent years.

On the other hand, the new era of communication systems has been started with the remarkable invention of Claude E. Shannon [52] in the year 1948. Consequently, various ECC schemes [16], [18], [53], [54], [55], [56], [57] have been developed for enhancing the reliability of communication systems. The ECCs have been applied in modern wireless communication techniques to increase the reliability of data transfer and reduce energy consumption. One of the leading families of error-correcting code for wireless communication systems is known as Low Density Parity Check (LDPC) code. However, the challenges in implementation of LDPC codes are: higher encoding and decoding complexities, long latency and the number of iterations. A novel linear error detection and correction approach for single bit and multiple bit error correcting codes called "Low Complexity Parity Check (LCPC)" codes has been presented in [58]. In [59], RS codes have been used for down link LTE system over LTE-MIMO channel. Short block length codes for ultra reliable low latency wireless communication have been explored in [60]. Also both convolutional and turbo codes are employed for error correction in wireless communication [61], [62], [63], [64]. In case of satellite communication, error detection is normally performed through Cyclic Redundancy Check (CRC) code and error correction is usually performed through linear codes [65], [66]. One of the commonly employed codes for error correction is Reed Solomon (RS) code [67] which is a linear cyclic systematic non-binary block code. The LDPC codes [68], [69] are also employed in satellite

communication. The combined version of CRC-LDPC and CRC-RS codes [70], [71] are also applicable for satellite communication. In optical communication systems, the challenge is to implement codes with low redundancy that are capable of correcting random and burst errors due to noise, dispersion and inter-channel cross talk, with attention for reduction of both complexity and cost [72]. The most recent and efficient codes available in the literature for optical communication systems are: concatenated BCH code [73], [74], Turbo product code with shortened BCH component codes [75], LDPC-based codes [76], staircase codes [77], [78], multidimensional turbo product [79], and super-product BCH code [80].

Various error control coding techniques are used in digital and personal communication standards. In GSM, Cyclic Redundancy Codes (CRC) are used for error detection, block and convolutional codes are applied for error correction [81]. Convolutional codes have been employed in the second generation CDMA systems [82], [83], [84]. In CDMA2000, Convolutional codes and Turbo codes [85], [86], [87] are used for error correction. The LDPC codes are the standard error correcting codes for many wireless communication protocols such as WiMAX (802.16e) [88] and WLAN (802.11) [89], [90]. Convolution codes and Reed-Solomon (RS) codes are mostly used for space communication systems [91], [92].

Also the ECC schemes with Soft Decision Decoding (SDD) exhibits better error performance under noisy channel compared to its Hard Decision Decoding (HDD) counterpart. Various SDD schemes with different decoding algorithms for Linear Block Codes (LBCs) have already been studied throughout the present and last few decades [93], [94], [95], [96], [97], [98]. The soft decoding of cortex code which is half rate systematic LBC with excellent distance property has been reported in [93] with better Bit Error Rate (BER) performance. The Soft Input- Soft Output (SISO) decoding of Hamming codes up to the word lengths of 63-bit have been studied in [94]. A reduced complexity SDD scheme has been applied to LBCs such as BCH and QR codes in [95] for obtaining better error performances. The high-speed, reduced complexity and scalable SDD scheme in [96] has been applied to LBCs for better error performances. The Quantum Maximum Likelihood Decoding (QMLD) scheme

for LBCs has been introduced in [97] with better decoding performance and least amount of asymptotic complexity. A fast and computationally efficient SDD scheme has been presented in [98] for LBCs which employs syndrome search partially. Large numbers of SDD algorithms for Reed-Solomon (RS) codes are also available in the literature for better decoding performance against noise and / or for reducing the computational complexity of decoding. A reduced complexity Maximum Likelihood (ML) decoding scheme for RS codes has been presented in [99] based on trellis diagram. List decoding algorithm based scheme of SDD has been presented in [100] for RS code with larger rates. The computational complexity of Algebraic Soft-decision Decoding (ASD) algorithm for RS codes has been reduced substantially in [101] where Progressive-ASD (PASD) algorithm has been introduced based on module minimization. Low Complexity Chase (LCC) and Progressive-LCC (PLCC) decoding algorithms based on module for reducing the computational complexity of larger rates RS codes have been introduced in [102]. A Viterbi decoder is proposed by Viterbi [103] which employs the Viterbi algorithm for decoding a long bit stream that has been encoded using a convolutional code. This algorithm is developed based on the maximum likelihood decoding. The Polar Codes (PCs) [104], [105] are the near Shannon's capacity approaching codes which have been selected for the control channels of Third Generation Partnership Project (3GPP) [106], [107]. Different SDD schemes [108], [109], [110] for Polar Codes have already been explored for enhancing their error performance and reducing the complexities. Therefore, most of the existing SDD schemes have been intended either for maximizing the decoder error performance or for minimizing the decoder complexity or both of these.

Moreover, the concatenation of two or more ECC schemes have been widely used in different communication systems for obtaining better error performances. The concatenation of RS code with inner BCH code [111] provides better BER performance for Dense Wavelength-Division Multiplexing (DWDM) systems. Various concatenation schemes of Polar codes as inner code and RS [112], [113], BCH [114], LDPC [115] codes as outer codes have been introduced for the application in communication systems. The concatenation of RS and convolutional codes [116] has been applied

in the IEEE802.11b standard for better noise performance. Also various kind of concatenated codes [117], [118], [119], [120] are available in the literature for space and satellite communication applications. The performances of CRC-Polar concatenated code have been analysed in depth for achieving better error performance [121]. Further, concatenation of Polar codes with Hash codes [122] and CRC-Polar codes [123] have been analysed for 5G applications.

1.3 Motivation of the Research Work

The ECCs are employed in almost all kinds of communication and storage systems. The design of suitable codec is very much essential and it is a challenging area in this research field. The ECCs for memories and communication systems suffers from the following weaknesses:

- Lower error detection and correction capability.
- Higher soft error rate with the down scaling of technology nodes.
- Higher design overheads (area, delay and power) in codec design.
- Higher mis-correction rate of adjacent error correcting codes.

Hence modification of existing ECC schemes and new proposal for efficient ECC schemes are necessary to solve the above mentioned problems for memories and communication systems.

1.4 Objectives of the Research Work

The objectives of this research are summarized as follows.

1) Design and implementation of existing random, adjacent and burst error correcting binary codes for semiconductor memories.

2) Modification of different existing schemes to lower hardware resources for implementation.
3) Proposal for improved error correcting codes to achieve lower area, delay, power consumption and mis-correction rate for memory applications.
4) Implementation of existing soft decoding architecture and proposal for modifications.
5) Design and implementation of combined codes for communication systems.

1.5 Thesis Organization

Rest of this thesis has been organized as follows.
In Chapter 2, fundamentals of adjacent and burst error correcting codes are described starting from the construction up to the evaluation of these codes.
In Chapter 3, three schemes for designing and implementing compact SEC-DAEC codes for storage applications have been presented.
In Chapter 4, three schemes for designing and implementing SEC-DED-DAEC codes with lower overheads and mis-correction rate have been introduced in storage system applications.
In Chapter 5, design and implementation of two schemes for SEC-DED-DAEC-TAEC codes and one scheme for t-bit BEC codes have been introduced with the aim of reducing decoder's complexities and mis-correction rates for storage systems.
In Chapter 6, two ECC schemes for FSO communication systems have been proposed and their performances have been analyzed.
In Chapter 7, summary and future directions are presented.

Chapter 2

Fundamentals of Adjacent and Burst Error Correcting Codes

2.1 Introduction

Various Error Correcting Codes (ECCs) are employed in storage systems for enhancing the reliability of these systems against soft errors. The ECC schemes with uncomplicated encoding and decoding operations like Adjacent Error Correcting (AEC) and Burst Error Correcting (BEC) codes are mostly preferred for storage systems. The AEC codes are capable of correcting single and multiple adjacent errors, whereas the BEC codes are capable of correcting single, multiple adjacent and almost adjacent errors. Therefore, the error correction capabilities of BEC codes are superior compared to AEC codes. But the BEC codes require larger numbers of logic gates for designing their decoder circuits compared to AEC codes. Also the circuit design complexities of both AEC and BEC codes increase with the increase of error correction capabilities. In storage systems, the desirable characteristics of AEC and BEC codes are lower redundancy, design overheads and mis-correction rates. Varieties of AEC and BEC codes have already been presented for addressing

these characteristics in the literature. The AEC and BEC codes with lower desirable characteristics compared to existing AEC and BEC codes are more suitable for modern storage systems. In this chapter, the fundamentals of designing AEC and BEC codes have been presented in details.

The rest of this chapter is arranged as follows. In section 2.2, types of errors are presented. The various categories of AEC and BEC codes are summarized in section 2.3. In section 2.4, H-matrix construction rules for AEC and BEC codes have been provided in details. The codec design of AEC and BEC codes are provided in section 2.5. In section 2.6, the process of evaluating AEC and BEC codes have been summarized. And this chapter has been concluded in section 2.7.

2.2 Types of Errors

The errors are induced in the transmitted or stored data by the noise and other forms of interference. These noise and interference are random in nature and change the assign voltage level of the data bits. As a result, the received data bits may differ from the transmitted or stored data bits. Thus transmitted data bit is flipped at the receiver side as the effect of error. The different types of errors which are associated in storage and communication systems are listed in this section.

- Single Error (SE) and Multiple Errors (ME): When a single data bit is found as erroneous bit among all the received data bits then it is termed as Single Error (SE). Similarly, when errors are associated with multiple received data bits then Multiple Errors (ME) occur. Some examples of multiple errors are Double Errors (DE) and Triple Errors (TE).

- Adjacent Errors (AE): The multiple errors which are consecutive or sequential (adjacent) in natures are called Adjacent Errors (AE). Some examples of this types of errors are Double Adjacent Errors (DAE), Triple Adjacent Errors (TAE), Quadruple Adjacent Errors (QAE).

TABLE 2.1: Different Types of Errors

Data bit	**Error Types**	**Erroneous bits**	**Symbolic Representation**
d_i	Single Error (SE)	d_i	**1**
	Double Adjacent Errors (DAE)	d_{i-1}, d_i (DAE_1) d_i, d_{i+1} (DAE_2)	1**1** **1**1
	Triple Adjacent Errors (TAE)	d_{i-2}, d_{i-1}, d_i (TAE_1) d_{i-1}, d_i, d_{i+1} (TAE_2) d_i, d_{i+1}, d_{i+2} (TAE_3)	11**1** 1**1**1 **1**11
	Quadruple Adjacent Errors (QAE)	d_{i-3}, d_{i-2}, d_{i-1}, d_i (QAE_1) d_{i-2}, d_{i-1}, d_i, d_{i+1} (QAE_2) d_{i-1}, d_i, d_{i+1}, d_{i+2} (QAE_3) d_i, d_{i+1}, d_{i+2}, d_{i+3} (QAE_4)	111**1** 11**1**1 1**1**11 **1**111
	Double Almost Adjacent Errors (DAAE) in a 3-bit burst length	d_{i-2}, d_i $(DAAE_1)$ d_i, d_{i+2} $(DAAE_2)$	10**1** **1**01
	Double Almost Adjacent Errors (DAAE) in a 4-bit burst length	d_{i-3}, d_i $(DAAE_1)$ d_i, d_{i+3} $(DAAE_2)$	100**1** **1**001
	Triple Almost Adjacent Errors (TAAE) in a 4-bit burst length	d_{i-3}, d_{i-2}, d_i $(DAAE_1)$, d_{i-1}, d_i, d_{i+2} $(DAAE_2)$ d_i, d_{i+1}, d_{i+3} $(DAAE_3)$ d_{i-3}, d_{i-1}, d_i $(DAAE_4)$ d_{i-2}, d_i, d_{i+1} $(DAAE_5)$ d_i d_{i+2}, d_{i+3} $(DAAE_6)$,	110**1** 11**0**1 **1**101 101**1** 10**1**1 **1**011

- Almost Adjacent Errors (AAE): The multiple errors which are not fully adjacent but nearly (almost) adjacent in nature are known as Almost Adjacent Errors (AAE). Some examples of almost adjacent errors are Double Almost Adjacent Errors (DAAE) and Triple Almost Adjacent Errors (TAAE).

- Burst Errors (BE): The Burst Errors (BE) are the all possibilities of errors which are confined inside a finite length burst of t-bit. Thus the burst errors are the all possibilities of SE, AE and AAE inside the burst length of t-bit. For example 3-bit BE includes SE, DAE and DAAE in a burst length of 3-bit. Similarly, the 4-bit BE includes SE, DAE, DAAE, TAE, TAAE and QAE in a 4-bit burst length.

All of these types of errors can affect data bits in different ways as shown in Table 2.1. As an example, the DAE can affect data bits in two ways i.e. DAE in d_{i-1} and d_i or DAE in d_i and d_{i+1} bits. Also the symbolic representation of each and every types of errors have been listed in the last columns of Table 2.1. In this representation,

'1' and '0' have been used to indicate the data bits with error and without error respectively. Also **'1'** denotes the data bit under consideration.

2.3 Classification of AEC and BEC Codes

In this section, different types of AEC and BEC codes which are available in the literatures for the protection of storage systems have been discussed with their error detection and correction capabilities.

2.3.1 Types of AEC Codes

The types of existing AEC codes for detecting and correcting adjacent errors are as follows.

- SEC-DAEC Code: The simplest form of AEC code is known as Single Error Correction- Double Adjacent Error Correction (SEC-DAEC) code which is proficient to correct the single and adjacent 2-bit errors.
- SEC-DED-DAEC Code: The Single Error Correction- Double Error Detection-Double Adjacent Error Correction (SEC-DED-DAEC) code is proficient for detecting double errors and correcting single error and adjacent double errors.
- SEC-DAEC-TAEC Code: The Single Error Correction- Double Adjacent Error Correction- Triple adjacent Error Correction (SEC-DAEC-TAEC) code is capable of correcting single error and adjacent double and triple bit errors.
- SEC-DED-DAEC-TAEC Code: The Single Error Correction- Double Error Detection Double Adjacent Error Correction-Triple adjacent Error Correction (SEC-DED-DAEC-TAEC) code is the SEC-DAEC-TAEC code with an extra Double Error Detection (DED) capability.

- SEC-DAEC-TAEC-QAEC Code: The Single Error Correction- Double Adjacent Error Correction- Triple adjacent Error Correction- Quadruple adjacent error correction (SEC-DAEC-TAEC-QAEC) is the extension of SEC-DAEC-TAEC code which can also correct adjacent 4-bit errors.

2.3.2 Types of BEC Codes

The different types of existing BEC codes are listed below.

- 2-bit BEC Code: The 2-bit Burst Error Correcting (BEC) codes are basically SEC-DAEC codes which are capable of correcting single and adjacent double bit errors.
- 3-bit BEC Code: The 3-bit BEC codes are capable of correcting single error, adjacent double, triple bit errors and double almost adjacent errors. For 3-bit BEC code, double almost adjacent errors are 2-bit errors which are separated by one correct data bit in a 3-bit burst length.
- 4-bit BEC Code: The 4-bit BEC codes are capable of correcting single bit error, adjacent double, triple and quadruple bit errors, double and triple almost adjacent bits errors. For 4-bit BEC code, double almost adjacent are double bit errors which are separated by one corrected data bit or two corrected data bits in a 4-bit burst length. Also triple almost adjacent errors are the triple bit errors which are separated by one correct data bit in a 4-bit burst length.

2.4 H-matrix Construction Rules for AEC and BEC Codes

The AEC and BEC codes are the class of binary Linear Block Code (LBC) which are entirely specified by parity check matrix (H-matrix). The H-matrix of (n, k)

AEC and BEC codes consist of $(n-k)$ rows and n columns. This H-matrix [2] is expressed in systematic form and provided in equation (2.1).

$$H = [P \quad | \quad I_{n-k}] \tag{2.1}$$

where P is a sub-matrix constructed by employing k numbers of data columns and I represents identity matrix. Generally, the H-matrix for AEC and BEC codes are obtained by employing some heuristic search algorithms where the basic construction rules and the additional conditions are imposed. The basic construction rules provide the required detection and correction capabilities of the H-matrix. The basic H-matrix construction rules for upto QAEC and 4-bit BEC codes have been summarized as follows.

Rule 1: All the columns of H-matrix must be distinct and non-zero.

Rule 2: All the XOR sums of double adjacent columns in H-matrix must be distinct, non-zero and different from any individual column.

Rule 3: All the XOR sums of triple adjacent columns in H-matrix must be distinct, non-zero and different from any individual column and XOR sums of double adjacent columns.

Rule 4: All the XOR sums of double almost adjacent columns in H-matrix must be distinct, non-zero and different from any individual column and XOR sums of double and triple adjacent columns.

Rule 5: All the XOR sums of quadruple adjacent columns in H-matrix must be be distinct, non-zero and different from any individual column and XOR sums of double and triple adjacent columns.

Rule 6: All the XOR sums of triple almost adjacent columns in H-matrix must be distinct, non-zero and different from any individual column and XOR sums of double, triple and quadruple adjacent columns and double almost adjacent columns.

Rule 7: All the XOR sums of any two columns in H-matrix must be distinct, non-zero and different from others error correcting syndromes.

The basic rules 1, 2 and 3 signify Single Error Correction (SEC), Double Adjacent

TABLE 2.2: Basic H-matrix Construction Rules for Different Types of AEC and BEC Codes

AEC and BEC Code	Required Basic Rules
SEC-DAEC	1, 2
SEC-DED-DAEC	1, 2, 7
SEC-DAEC-TAEC	1, 2, 3
SEC-DED-DAEC-TAEC	1, 2, 3, 7
SEC-DAEC-TAEC-QAEC	1, 2, 3, 4
3-bit BEC	1, 2, 3, 5
4-bit BEC	1, 2, 3, 4, 5, 6

Error Correction (DAEC) and Triple Adjacent Error Correction (TAEC) functionalities respectively. Similarly the rules mentioned in 4, 5 and 6 are for Quadruple Adjacent Error Correction (QAEC), Double Almost Adjacent Error Correction (DAAEC) and Triple Almost Adjacent Error Correction (TAAEC) respectively. The Double Error Detection (DED) functionality is ensured by the last rule 7. Therefore, the combinations of two or more of these basic H-matrix construction rules are necessary to obtain the H-matrix of desired AEC and BEC codes and it is provided in Table 2.2.

On the other hand, the additional conditions in the H-matrix search algorithms are employed to reduce either the design overheads, mis-correction rate or both of these. These additional conditions are mainly used to optimized the H-matrix in the following ways.

- *H-matrix optimized to reduce the total counts of 1's*: This provides the benefit of designing encoder and syndrome circuits with lower area requirement [124], [125], [126], [127].
- *H-matrix optimized to reduce the total counts of 1's per row of it*: This makes the encoder and syndrome circuits more faster in their operation [124], [125], [126], [127].
- *Interleaving of data and parity columns in H-matrix*: Interleaving of data and parity columns in the H-matrix provide the parallel decoding of some of the

data bits and reduces the decoding circuitry of AEC and BEC decoders [124], [126], [128].

- *H-matrix with special structure*: The special structure of the H-matrix for SEC-DED-DAEC and SEC-DED-DAEC-TAEC codes as suggested by Neale et al, [129], [130], [131] are efficient for reducing the mis-correction rate of these codes.

- *H-matrix with weight-3 SEC syndrome and weight-4 DAEC syndromes*: The H-matrix of SEC-DED-DAEC code have been optimized by Reviriego et al. [132] with weight-3 SEC syndrome and weight-4 DAEC syndromes. This scheme was proposed to reduce area and delay overheads of the decoder.

Another important parameter for designing an (n, k) AEC and BEC codes is the numbers of parity bits or check bits $(n-k)$ which must be as few as possible for the memory applications. This is due to the fact that excess numbers of parity bits require more area in a memor. But the numbers of parity bits which are required to design various AEC and BEC codes can't be minimized arbitrarily as the numbers of parity bits are bounded by equation (2.2). The left hand side of equation (2.2) represents the available total numbers of non-zero binary combinations of $(n-k)$-bit. And the right hand side of equation (2.2) provides the required numbers of combinations of $(n-k)$-bit for the feasibility of various AEC and BEC codes. Also it is clear from equation 2.2 that AEC and BEC codes with higher correction capabilities require more numbers of parity bits compared to the similar codes with lower error correction capabilities.

$$2^{n-k} - 1 \geq \begin{cases} 2n-1, & \text{for SEC-DAEC} \\ 3n-3, & \text{for SEC-DAEC-TAEC} \\ 4n-6, & \text{for SEC-DAEC-TAEC-QAEC} \\ 4n-5, & \text{for 3-bit BEC} \\ 8n-16, & \text{for 4-bit BEC} \end{cases} \tag{2.2}$$

$$\begin{bmatrix} 10000101 \\ 0100011 \\ 0010111 \\ 0001010 \end{bmatrix}$$

FIGURE 2.1: H-matrix for $(7, 3)$ SEC-DAEC Code

As an illustrative example, one of the possible H-matrix for $(7, 3)$ SEC-DAEC code has been shown in Fig. 2.1.

2.5 Codec Design for AEC and BEC Codes

The codec for any ECC is the concatenated form of the encoder and decoder circuits. The encoding and decoding operations of AEC and BEC codes have been explored in this section.

2.5.1 Encoding of AEC and BEC Codes

The encoder for AEC and BEC codes computes the parity bits based on the H-matrix. The parity expressions are obtained by computing XOR sums of data bits which have 1's per rows of P-sub-matrix in equation (2.1). The parity bits are appended with the original data bits to produce the codewords. These codewords are stored instead of original data in memory as a protective measure against soft errors.

2.5.2 Decoding of AEC and BEC Codes

The AEC and BEC decoder circuits retrieve the original data bits by considering the store codewords in the memory cells as input. The decoder performs three major

tasks while retrieving the original data bits namely: Syndrome Computation (SC), Error Location Detection (ELD) and Error Correction (EC). The syndromes (S) are computed from H-matrix by employing the following expression:

$$S = r.H^T \tag{2.3}$$

where r is the stored codeword in memory and H^T is the transpose of parity check matrix (H) of the ECC. The syndrome generator produces syndrome values according to equation (2.3) and this syndrome circuit is constructed by employing only XOR gates. Error detection is done based on the syndrome values i.e. if all syndromes are zero then no error is occurred, whereas for any non-zero syndrome corresponds to error in received codeword.

The ELD circuit is required in any decoder to determine the accurate location of errors in the received codeword. Various AEC and BEC codes have ELD circuits with different logic requirements based on their respective error correction capabilities. The correctable errors for a particular AEC or BEC code play an important role for the designing of ELD circuit of that code. All the syndromes for these correctable errors are given as the inputs of ELD circuit. And, these syndromes are compared to produce the output of ELD circuit. Error is located in a particular data bit position when the corresponding ELD circuit output is equals to 1. The ELD circuits for locating error in i^{th} data bit (d_i) of SEC-DAEC, SEC-DAEC-TAEC and SEC-DAEC-TAEC-QAEC decoders have been shown in Fig. 2.2. In the case of SEC-DAEC decoder, the ELD circuit for a particular data bit position (d_i) requires highest of three error locating syndromes. These error locating syndromes are correspond to Single Error (SE), Double Adjacent Errors in d_{i-1} and d_i (DAE_1) and Double Adjacent Errors in d_i and d_{i+1} (DAE_2).

The ELD circuit for a particular data bit (d_i) in SEC-DAEC-TAEC decoder require three additional Triple Adjacent Errors (TAE) locating syndromes along with the single and double adjacent errors locating syndromes. These TAE locating syndromes are correspond to TAEs in d_{i-2} , d_{i-1}, d_i (TAE_1), d_{i-1} , d_i, d_{i+1} (TAE_2)

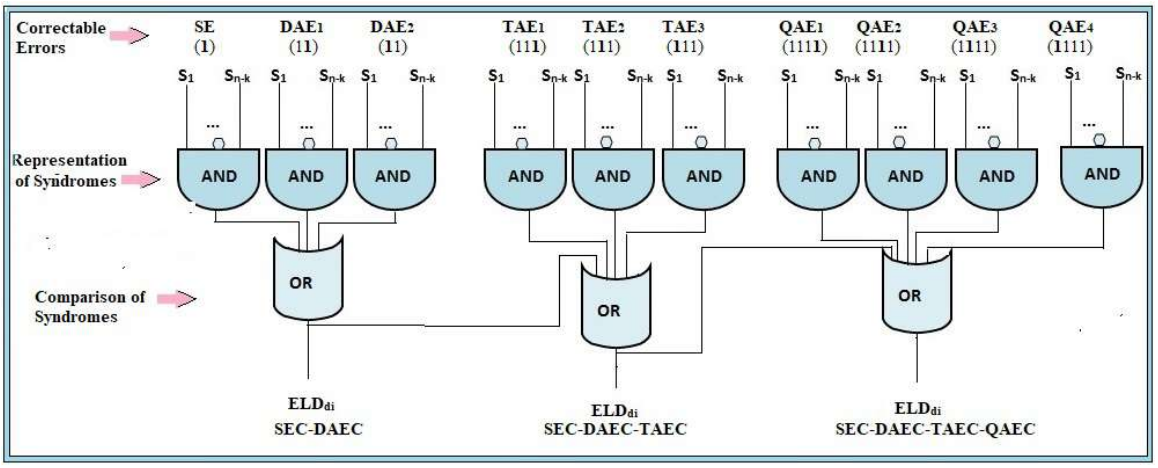

FIGURE 2.2: ELD Circuits for Locating Error in i^{th} Data bit (d_i) of SEC-DAEC, SEC-DAEC-TAEC and SEC-DAEC-TAEC-QAEC Decoders

and d_i , d_{i+1}, d_{i+2} (TAE_3). Therefore, a maximum of 6 numbers of error locating syndromes are involved for locating error in a particular data bit position for SEC-DAEC-TAEC decoder. Due to this reason higher numbers of logic gates are involved in designing the ELD block of SEC-DAEC-TAEC decoder compared to the SEC-DAEC decoders.

The ELD circuit for a particular data bit (d_i) in SEC-DAEC-TAEC-QAEC decoder require four additional Quadruple Adjacent Errors (QAE) locating syndromes along with the single, double and triple adjacent errors locating syndromes. These QAEs locating syndromes are correspond to QAE in d_{i-3} , d_{i-2}, d_{i-1}, d_i (QAE_1), d_{i-2}, d_{i-1}, d_i, d_{i+1} (QAE_2), d_{i-1}, d_i, d_{i+1}, d_{i+2} (QAE_3) and d_i, d_{i+1}, d_{i+2}, d_{i+3} (QAE_4). Therefore, a maximum of 10 numbers of error locating syndromes are involved for locating error in a particular data bit position for SEC-DAEC-TAEC-QAEC decoder. Due to this reason higher numbers of logic gates are involved in designing the ELD block of SEC-DAEC-TAEC-QAEC decoder compared to the SEC-DAEC-TAEC decoders. The ELD circuit of 3-bit BEC decoder for locating error in d_i requires all the error locating syndromes for single, double and triple adjacent errors and highest of two extra error locating syndromes for Double Almost Adjacent Errors (DAAE) as shown in Fig.2.3. The ELD circuit of 3-bit BEC decoder for location error in d_i requires a maximum of 8 error locating syndromes. Therefore, higher numbers of

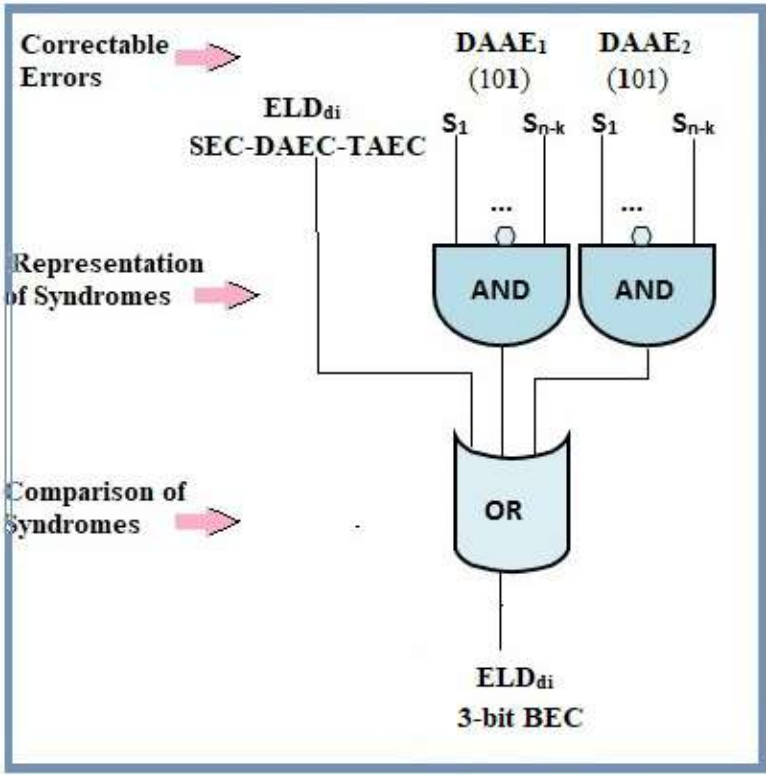

FIGURE 2.3: ELD circuits for Locating Error in i^{th} Data bit (d_i) of 3-bit BEC Decoder

logic gates are required for designing the ELD block of 3-bit BEC decoder compared to the ELD block of SEC-DAEC-TAEC decoder. In a similar way, the ELD circuit of 4-bit BEC decoder for location error in d_i requires a maximum of 10 additional double and triple almost adjacent errors locating syndromes along with all the error locating syndromes for single error, double, triple and quadruple adjacent errors. Thus ELD circuit of 4-bit BEC decoder require a maximum of 20 different error locating syndromes for locating error in a particular data bit. Therefore, the ELD block of 4-bit BEC decoder requires highest numbers of logic gates compared to all the adjacent and burst error correcting decoders discussed so far.

The located errors in the ELD block of AEC and BEC decoders are corrected by the Error Correction (EC) block of the respective decoders. In the EC block of AEC

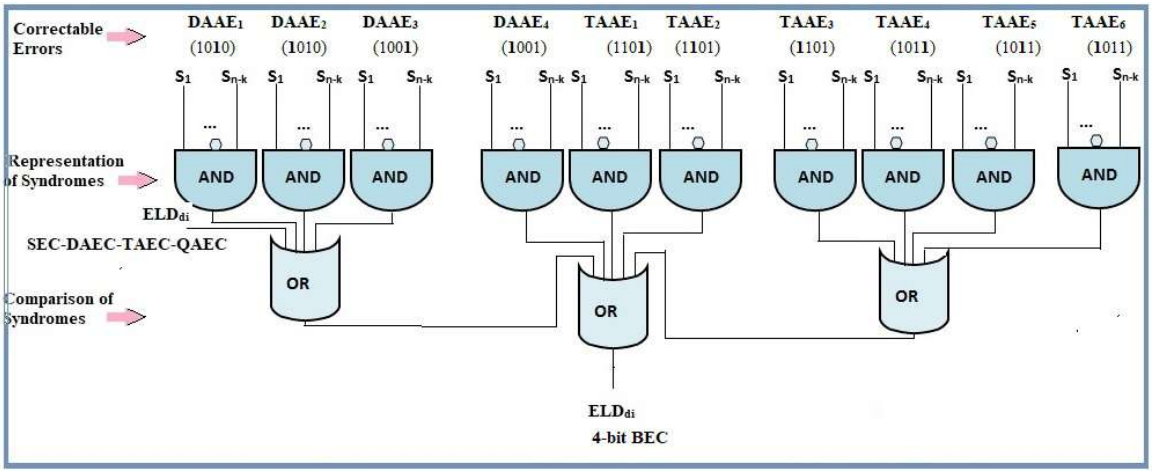

FIGURE 2.4: ELD Circuits for Locating Error in i^{th} Data bit (d_i) of 4-bit BEC Decoder

and BEC decoders, one XOR logic gate for each and every data bit positions is used to flip the bits in errors.

2.5.3 Illustrative Example of a SEC-DAEC Codec

In this section, the design of SEC-DAEC codec has been illustrated for memory applications. The block diagram of encoding and decoding of $(7, 3)$ SEC-DAEC code for memory application has been presented in Fig. 2.5. In this case, the $(7, 3)$ SEC-DAEC encoder generates the codeword $(c) = [0011111]$ from the input data $(d) = [111]$.. This codeword is stored in memory. Now if soft errors occur in the memory with an error pattern (e) equals to $[0000011]$, then the stored codeword (c) changes to received codeword (r) as follows:

$$r = c \oplus e = [0011111] \oplus [0000011] = [0011100] \tag{2.4}$$

This r is the input to the $(7, 3)$ SEC-DAEC decoder. On the other hand, the SEC-DAEC decoder reads out the stored data from the memory and correct the presence of any single or double-adjacent errors. The three functional units of the decoder circuit are the Syndrome Computation (SC) block, the Error Location Detection (ELD) block, and the Error Correction (EC) block. The first block multiplies any

received codeword (r) with the H-matrix's transposed value to arrive at the syndrome value (S) for any linear ECCs. The presence of any errors in r depends on the values of the syndrome, i.e., if S=0, errors remain undetected, otherwise errors are detected. The second functional block of the decoder utilizes the syndrome values to identify the precise locations of errors in the r. The ELD block of a typical SEC-DAEC decoder analyses three syndrome values, namely syndromes for a Single Error (SE) and syndromes for two possibilities of Double Adjacent Errors (DAE). These DAE syndromes are obtained by XORing i^{th} column in H-matrix with $(i-1)^{th}$ or $(i+1)^{th}$ column respectively. Finally, the located error in the ELD block is corrected by error correction block which inverts the bit in error with the help of XOR gate. The syndrome circuit of $(7,3)$ SEC-DAEC decoder generates the syndrome values (S) equals to [1001] for r=[0011100]. Error Location Detection (ELD) expressions are computed for each data bit based on the syndrome values as shown in Fig. 2.5. The output of ELD circuit for data bits d_1, d_2 and d_3 are 0, 1 and 1 respectively. Therefore, the errors are located on the second and third data bits and these errors are corrected by error correction circuit of $(7,3)$ SEC-DAEC decoder.

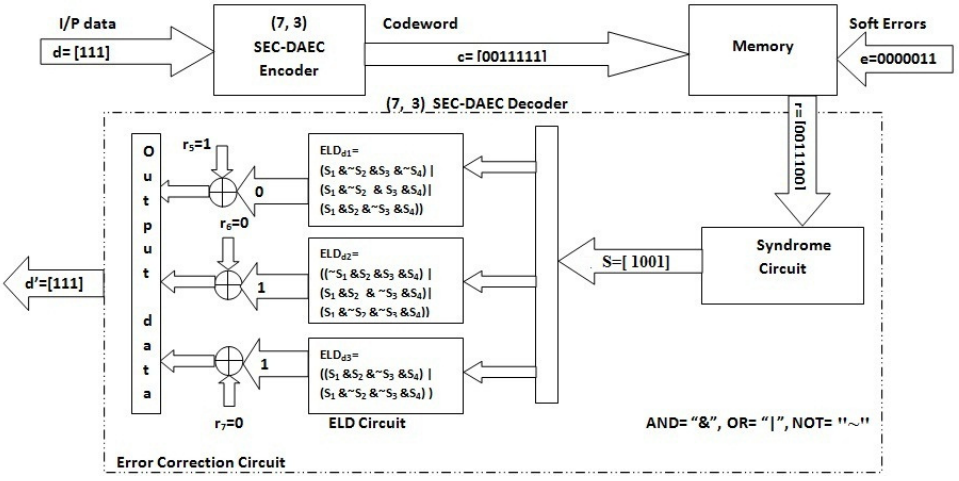

FIGURE 2.5: Block Diagram of $(7,3)$ SEC-DAEC Codec for Memory Application

2.6 Evaluation of AEC and BEC Codes

Once the H-matrix, encoder and decoder circuits of AEC and BEC codes have been constructed completely, then evaluation of these AEC and BEC codes are performed for reporting their respective merits and demerits. The evaluation of any AEC or BEC code can be performed in two ways. Firstly, by evaluating the theoretical code design parameters and then evaluating the synthesis results.

2.6.1 Code Design Parameters

Various theoretical code design parameters which are used to perform the quality of (n, k) AEC and (n, k) BEC codes are listed below.

- Codeword Length (n):
  It represents the total number of bits in the codeword produced at the output of the encoder.

- Data Word Length (k):
  It represents the total number of data bits which are provided at the input of the encoder. In the case of memory applications the k is selected always as the integer power of 2.

- Code Rate (R_c):
  This is the ratio of data word length to the codeword length and expressed as

$$R_c = \frac{k}{n}. \tag{2.5}$$

  The AEC or BEC code designed with higher value of code rate is always preferred as it corresponds to lesser area overhead for storage and better bandwidth utilization during transmission.

- Numbers of parity bits ($n - k$):
  The numbers of parity bits are the numbers of extra (redundant) bits which

are added with the message bits to form the codeword. The AEC or BEC code designed with lower numbers of parity bits is always have the added benefits of lower area requirement for storing the parity bits in memory applications.

- Fractional Redundancy (R):

  It is the ratio of number of parity bits to the numbers of information bits and it is expressed as

  $$R = \frac{n-k}{k}. \tag{2.6}$$

  The R represents the numbers of parity bits per information bit.

- Total Count of 1's in H-matrix:

  The total count of 1's in the H-matrix provide the theoretical area estimations of encoding and syndrome circuits. The H-matrix of any AEC or BEC code which is constructed with reduced total count of 1's has the obvious benefit of lesser number of XOR logic gates in its encoding and syndrome circuits. This makes the encoding and syndrome circuits more compact in size.

- Highest Number of 1's per Row of H-matrix:

  The highest number of 1's per row of H-matrix provides the theoretical delay estimations in encoder and syndrome circuits. The H-matrix of AEC or BEC code designed with lesser number of 1's per row corresponds to lower delay in encoding and syndrome circuits.

- Total Number of Equivalent NAND2 Gates:

  The overall theoretical area of encoder and decoder circuits are estimated by counting the total number of equivalent 2-input NAND (NAND2) gates which are required to construct the respective encoder and decoder circuits.

- Total Count of Equivalent NAND2 Gates in the Critical Path of Encoder and Decoder Circuits:

  The critical path of any logic circuit is the longest path in terms of logic gates from the input to the output of the circuit. The overall theoretical delay of

any encoder or decoder or codec circuit is generally estimated by counting the total numbers of equivalent NAND2 gates which are required to establish the critical path.

- Mis-correction Rate of AEC and BEC Codes:
  Mis-correction rate is the chances of misinterpretation of non-adjacent random errors as adjacent errors. The mis-correction rate of any (n, k) code for e-random errors is denoted by P and is computed based on H-matrix of the code by employing equation (2.7).

$$P = \frac{X}{\binom{n}{e} - Y} \tag{2.7}$$

  where X=number of shareable syndromes i.e. number of possibilities of e-columns in H-matrix which wrongly generate either an error identifying or an all-zero syndrome and Y=number of e-column error identifying syndromes.

2.6.2 Synthesis Results

The synthesis results for any ECC scheme provides the encoder, decoder or codec design overheads like area, delay and power consumption. The AEC and BEC codec circuits are represented in verilog Hardware Description Language (HDL) for the purpose of obtaining the synthesis results of these circuits. Now synthesis results are obtained by two ways namely FPGA-based synthesis results and ASIC-based synthesis results. In the case of FPGA based synthesis, the verilog codes are produced in Field Programmable Gate Array (FPGA) platform with a suitable device family. The FPGA-based synthesis results are measured in terms of Lookup Table (LUTs) for area estimation and delay in ns for timing estimation. The Application Specific Integrated Circuit (ASIC) based synthesis of any circuit provides the area, delay and power overheads with better accuracy compared to FPGA-based synthesis results.

2.7 Conclusions

In this chapter, the fundamentals of AEC and BEC codes are provided in details. Various types of adjacent and burst errors and their corresponding error correcting codes have been mentioned first at the beginning of this chapter. The basic construction rules for the H-matrix of various AEC and BEC codes have been summarized for constructing the respective H-matrix. The design of encoder and decoder for various AEC and BEC codes have been presented based on the H-matrix. Finally, the various parameters which are essential for the evaluation of AEC and BEC codes have been familiarised at the end of this chapter. Design and implementation of three new schemes of SEC-DAEC codes have been introduced in the next chapter for storage systems with the objective of reducing the decoding overheads.

Chapter 3

Design and Implementation of SEC-DAEC Codes

3.1 Introduction

Memories are integral part of modern electronic gadgets for storing data and programs. The proper functionality of these electronic gadgets depends on the reliability of the storage system. But the soft errors mostly induced by radiation are the major challenge against the reliability of storage system. The memory cell upsets like Single Cell Upset (SCU) and Multiple Cells Upsets (MCUs) are caused in memories due to these soft errors. The Error Correction Codes (ECCs) are widely employed for reducing cell upsets. The Single Error Correction (SEC) and Single Error Correction- Double Error Detection (SEC-DED) codes were employed for the protection of memory against SCU in the earlier version of semiconductor memories. But the MCUs are more frequent in modern storage systems due to continuous down scaling of semiconductor technology nodes. The most common errors in multiple memory cells are the single error and double-adjacent errors. These errors are corrected by employing Single Error Correction- Double Adjacent Error Correction (SEC-DAEC) codes. The SEC-DAEC codes are the most straightforward form of

adjacent error correcting codes. The SEC-DAEC codes are popularly employed in memory systems for mitigating the effects of soft errors in memory cells due to their simplest encoder and decoder structures. The SEC-DAEC codes have higher decoding overheads compared to traditional SEC-DED codes and also these codes have higher chances of mis-correction against 2-random and 3-random errors. Therefore, design of SEC-DAEC codes with lower decoding overheads and mis-correction rates are desirable to protect memory against MCUs and it is indeed a challenging task to the researchers. This chapter presents three different schemes for SEC-DAEC codes which have been designed with the aim of reduction in codec design overheads.

The rest of this chapter is structured as follows. The section 3.2 describes related SEC-DAEC and SEC-DED-DAEC Codes. The design of an improved single and double- adjacent codec with lower decoding overheads has been presented in section 3.3. In section 3.4, design of lower complexity error location detection block for existing SEC-DAEC codes has been introduced. The parallel decoding of extended Golay code based SEC-DAEC code has been explored in section 3.5. Finally, section 3.6 concludes the chapter.

3.2 Existing SEC-DAEC and SEC-DED-DAEC Codes

The related SEC-DAEC and SEC-DED-DAEC codes which are available in the literature for memory applications have been summarized in this section.

The first SEC-DED-DAEC code for memories have been introduced by Dutta et al. [133] in the year 2007. The Dutta codes and traditional SEC-DED codes have exactly same redundancy but Dutta codes have higher mis-correction rates with respect to 2-random and 3-random errors. The mis-correction rate of SEC-DED-DAEC code against 3-random errors have been reduced by Richter et al. [134]. The mis-correction rate of SEC-DED-DAEC codes have been further reduced by

utilizing idle standby columns in [135]. The SEC-DED-DAEC codes by Ming et al. [136] have been designed with the minimum redundancy of SEC-DED codes and these codes have lower mis-correction rate against 2-random errors. Total removal of mis-correction within the dispersion window have been explored with lower SEC-DED-DAEC codes design cost in [137]. A new structure of SEC-DED-DAEC codes have been constructed with lower mis-correction rates and scalable adjacent error detection proficiency by Neale et al. [129], [130]. A competent realization of SEC-DED-DAEC code [138] has been presented by considering parity check bits precalculation for achieving lesser design overheads. The SEC-DED-DAEC codes [139] have been explored for the protection of on-chip memory with zero mis-correction. Also the zero mis-correction and lower design overheads SEC-DED-DAEC codes have been introduced for the on-chip memories in [140]. Reliability of stored data bit and parity bit arrays in memory have been enhanced by applying SEC-DAEC codes in [141]. Two decoding schemes for SEC-DED-DAEC codes have been introduced by Reviriego et al. [132] for optimizing the area and delay overheads respectively. The SEC-DED-DAEC codes based on Neale et al. and Reviriego et al. have been produced in [142] to obtain their combined merits. Lower decoding overheads and zero mis-correction rates against 2-random errors are the major characteristics of SEC-DAEC codes in [143] for the protection of on-chip memory. The SEC-DAEC codes have been optimized at the time of their implementation in FPGAs to attain the design constrains [144]. The SEC-DED-DAEC code based on Orthogonal Latin Square (OLS) code has been reported in [145] with zero mis-correction against double non-adjacent errors and lower decoding latency. Also parallel decoding of extended Golay code based SEC-DAEC code [128] reduces the decoding overheads efficiently. But both the codes in [145] and [128] have higher counts of parity bits compared to the existing SEC-DAEC codes. The lower latency SEC-DAEC codes have been presented by Li et al. [124] with minimum redundancy at the cost of higher mis-correction rates. The simultaneous error detection method in SEC-DAEC encoder has been presented in [146] for enhancing the reliability of encoding operations. The use of Karnaugh map (K-map) in the design of adjacent error correcting codes have

been explored in [147] for reducing the codec design overheads. Therefore, the main objectives of designing the SEC-DAEC and SEC-DED-DAEC codes for memory applications are either to reduce decoding overheads or mis-correction rates or both of these.

3.3 Design of Improved SEC-DAEC Codec

The details design technique and implementation of improved SEC-DAEC codec with lower decoding overheads has been presented in this section.

3.3.1 Construction of Proposed SEC-DAEC H-matrix

The H-matrix of proposed SEC-DAEC code has been constructed with the objective that it can be decoded using lower overheads. There are several existing techniques available to reduce decoding overheads like reducing the total number of 1's in H-matrix or reducing the number of 1's in the heaviest row of H-matrix or interleaving of data and parity columns of H-matrix. But the proposed technique concentrates on simplifying the error location detection (ELD) block of conventional SEC-DAEC decoders. For this purpose, additional conditions have been introduced with the basic H-matrix construction rules of SEC-DAEC code. These additional conditions are as follows:

1. The syndromes for correcting single and double-adjacent errors are common for a particular code bit position when half of the total syndrome bits are considered for correcting errors i.e. either the upper or lower half of the total syndrome bits.
2. Also these common upper half syndromes which are used for correcting errors in different code bit positions are distinct from each other and non-zero, so as the common lower half syndromes.

For the feasibility of these conditions, the structure adopted for the proposed H-matrix is shown in Fig. 3.1, where the shaded portion of any column represents distinct non-zero combination of $(n-k)/2$ bits. Also, common syndromes for single and double-adjacent errors occur in the upper $(n-k)/2$ rows of H-matrix for odd code bit positions and the same happen in the lower $(n-k)/2$ rows for even code bit positions. Thus, odd and even positioned code bits can be decoded by utilizing only the upper and lower half of syndrome bits respectively.

The illustration of proposed H-matrix structure for 8-bit parity has been shown in Fig.3.2. In this figure, each column of the H-matrix has been divided into two halves and each half consists of 4-bit. In case of odd positioned columns, the upper half consists of distinct non-zero combination of 4-bit and the lower half consists of four 0's as shown in Fig. 3.2 for column numbers of 1, 3, 5, ..., $(n-1)$. The even positioned columns contains four 0's and distinct non-zero combination of 4-bit in the upper and lower half respectively as shown in Fig. 3.2 for column numbers of 2, 4, ..., n. Therefore, with this H-matrix structure, the upper and lower half of the syndrome bits for single error correction is exactly same as the double-adjacent error correction for odd and even positioned columns respectively. These Common Syndrome (CS) bits of first three columns as shown in Fig. 3.2 for correcting single and double-adjacent errors are expressed in equation (3.1). These syndromes expressions are basically the expressions for Error Location Detection (ELD) logic for these respective bit positions. Hence all the odd or even bit positions can be decoded by employing only the upper or lower half of the syndromes respectively in the proposed H-matrix structure.

$$\begin{aligned} CS_1 &= S_1 \& S_2' \& S_3 \& S_4' \\ CS_2 &= S_5 \& S_6' \& S_7 \& S_8' \\ CS_3 &= S_1 \& S_2' \& S_3' \& S_4 \end{aligned} \tag{3.1}$$

In the proposed H-matrix structure, common upper or lower half of the syndrome bits are employed for correcting single and double-adjacent errors depending on odd

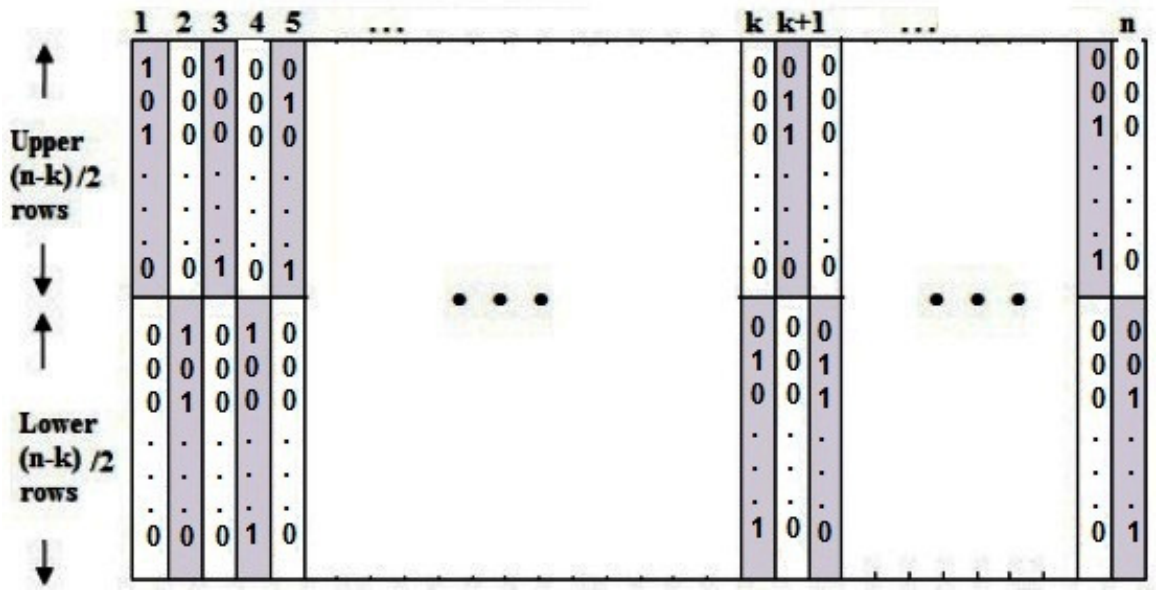

FIGURE 3.1: Structure of Proposed H-matrix

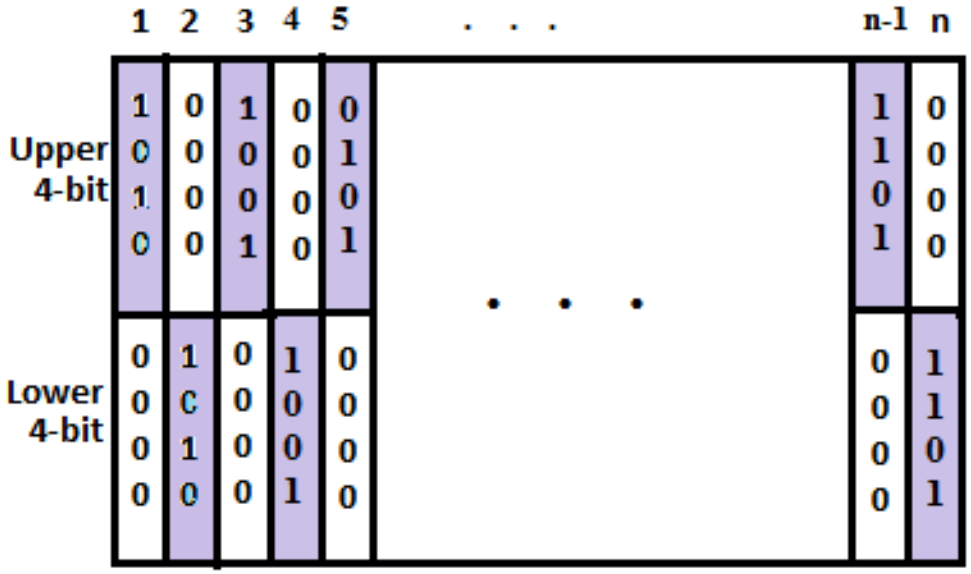

FIGURE 3.2: Illustration of Proposed H-matrix Structure for 8-bit Parity

or even positioned columns. Hence, only n numbers of distinct non-zero combinations of $(n-k)/2$ bits are required to construct proposed (n,k) SEC-DAEC code. Also any non-zero combination of $(n-k)/2$ bits can be utilized at most two times i.e. once in the upper and once in the lower half of any odd and even positioned column respectively. Therefore, only $n/2$ number of distinct non-zero combinations

of $(n-k)/2$ bits are required now for constructing n columns. The $(n-k)$ number of parity bits for the proposed structure of H-matrix for (n,k) code, must satisfy the inequality as shown in equation (3.2).

$$\frac{n}{2} \leq 2^{\frac{n-k}{2}} - 1 \tag{3.2}$$

The minimum number of parity bits which are required to construct the proposed H-matrix for word lengths of 16, 32 and 64 bits are 8, 10 and 12 respectively. The numbers of parity bits in the proposed H-matrices are slightly higher compared to existing SEC-DAEC codes which in turn increases the requirement of extra memory cells for storing these extra parity bits which is the limitation of proposed scheme. The proposed H-matrix for 16, 32 and 64 bits word lengths are presented in Fig. 3.3 to Fig. 3.5.

$$\begin{bmatrix}
101000000010001000001010 \\
001010100000000010000010 \\
000010001010100000001000 \\
000000001000001010101010 \\
\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\
010000000001010000010101 \\
010001000000000101000001 \\
000001010101000000010000 \\
000100010000010001010001
\end{bmatrix}$$

FIGURE 3.3: Proposed H-matrix for $(24,16)$ SEC-DAEC Code

3.3.2 Design of Proposed SEC-DAEC Codec

A SEC-DAEC codec consists of an encoder and a decoder. In SEC-DAEC codec, the encoder computes parity bits based on data bits. These parity bits are also

$$
\begin{bmatrix}
101000000000000101000100000000010001000100\\
001010100000000000000000100010001000101000 10\\
000010001000101000000000000010101000101000\\
000000001010000010100010000000001010001010\\
000000000010000000001000101010001000100010\\
\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\\
010000000000001000101000000010001000100010 0\\
010001000000000000000101010000010001010001\\
000001010000010100000000000001010100010100\\
000000010101000000100000100000000000101000101\\
000100000000100000000100000100010001000100 01
\end{bmatrix}
$$

Figure 3.4: Proposed H-matrix for $(42, 32)$ SEC-DAEC Code

$$
\begin{bmatrix}
1010000000000000010001010001000000000000000010001000100010001000100010001000100000\\
0010101000000000000000000001000101000000010001000100010000010001000100010 10\\
0000100010001000100000000000000000000101000100000100010001010001000100000 1010\\
0000000010100000000101000000001000000000000100010100000100010100000100010 100010\\
0000000000100010000000101000000010001000000010001010000010001010000001010 0000\\
0000000000000010000000000001000100010001010001000100010100000100010100000 1000\\
\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\\
0100000000000000010100010100000001000000000100010001000100010001000100010000\\
0100010000000000000000000001010100010000000100010001000100000100010001000101\\
0000010100000000010001000000000000000001010001000001000100010100010001000001 01\\
0000000100010001000100000000010000000000001000101000001000101000001000101000 1\\
0000000001010100000000001000000001000001000000010001010000010001010000010100 00\\
0001000000000001000000000000100000000010001010001000100010100000100010100000 100
\end{bmatrix}
$$

Figure 3.5: Proposed H-matrix for $(76, 64)$ SEC-DAEC Code

stored along with the data bits in memory cells. Whereas, the decoder takes the stored codewords from the memory as input and produces corrected version of the stored data as output. The parity bits are computed by taking XOR sums of the corresponding data bits which have 1's in each row of H-matrix. Expressions for computing the parity bits for the proposed $(24, 16)$ SEC-DAEC code based on H-matrix shown in Fig. 3.6 are provided in equation (3.3).

p_1	d_1	d_2	p_8	d_3	d_4	p_2	d_5	d_6	p_7	d_7	d_8	p_3	d_9	d_{10}	p_6	d_{11}	d_{12}	p_4	d_{13}	d_{14}	p_5	d_{15}	d_{16}
1	0	1	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	1	0	1	0
0	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0
0	0	0	0	1	0	0	0	1	0	1	0	1	0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	1	0	1	0	1	0	1	0
0	1	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	1	0	1	0	1
0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	1
0	0	0	0	0	1	0	1	0	1	0	1	0	0	0	0	0	0	0	1	0	0	0	0
0	0	0	1	0	0	0	1	0	0	0	0	0	1	0	0	0	1	0	1	0	0	0	1

FIGURE 3.6: Proposed H-matrix for $(24, 16)$ SEC-DAEC Code with Data and Parity Bits Positions

$$
\begin{aligned}
p_1 &= d_2 \oplus d_7 \oplus d_{10} \oplus d_{14} \oplus d_{15} \\
p_2 &= d_2 \oplus d_3 \oplus d_{11} \oplus d_{15} \\
p_3 &= d_3 \oplus d_6 \oplus d_7 \oplus d_{14} \\
p_4 &= d_6 \oplus d_{10} \oplus d_{11} \oplus d_{14} \oplus d_{15} \\
p_5 &= d_1 \oplus d_8 \oplus d_9 \oplus d_{13} \oplus d_{16} \\
p_6 &= d_1 \oplus d_4 \oplus d_{12} \oplus d_{16} \\
p_7 &= d_4 \oplus d_5 \oplus d_8 \oplus d_{13} \\
p_8 &= d_5 \oplus d_9 \oplus d_{12} \oplus d_{13} \oplus d_{16}
\end{aligned} \tag{3.3}
$$

where p_i and d_j represent i^{th} parity bit and j^{th} data bit respectively. Proposed $(24, 16)$ SEC-DAEC encoder circuit generates parity bits by employing equation (3.3).

The proposed decoding steps are summarized as follows.

1. Compute the syndrome (S) as per equation (2.3).

2. Correct the odd positioned data bit in the codeword for which upper $(n-k)/2$ number of rows of H-matrix match with the first $(n-k)/2$ numbers of syndrome bits.

3. Correct the even positioned data bit in the codeword for which lower $(n-k)/2$ number of rows of H-matrix match with the last $(n-k)/2$ numbers of syndrome bits.

The first step of decoding is the conventional one where syndrome bits are generated by syndrome circuit. But the main deference between proposed and conventional SEC-DAEC decoder remains in the last two decoding steps. The Error Location Detection (ELD) block of proposed decoder is much simpler compared to existing decoders. This is due to the fact that the proposed H-matrix is constructed in a manner such that any odd positioned data bit in the codeword has a common syndrome for single and double-adjacent errors for upper half of the syndrome bits. And the same is true for any even positioned data in the codeword for lower half of the syndrome bits. As a result, any error in the odd or even positioned data bits can be located by simply checking the upper or lower half of the syndrome bits only. Thus, comparison of all the syndrome bits for locating errors like traditional decoders are not required in proposed scheme. Also the upper or lower half of syndrome bits which are needed in the proposed decoder are common for single as well as double-adjacent errors for any particular data position. As a result, there are no need of comparing syndromes for single and double-adjacent errors separately for each data bit in proposed decoder like traditional decoders. Therefore, benefits of proposed decoding scheme are two folded: i) half of the syndrome bits are necessary only i.e. reduction in number of AND and NOT gates in ELD block, and ii) common syndrome for single and double-adjacent errors i.e. OR gate is not required in ELD block. The ELD expressions of first four data bits for proposed $(24, 16)$ SEC-DAEC decoder have been summarized in equation (3.4). The ELD expressions for rest of the data bits can be obtained similarly based on proposed decoding steps and the H-matrix as shown in Fig. 3.3.

$$\begin{aligned} ELD_{d_1} &= S_5 \& S_6 \& S'_7 \& S'_8 \\ ELD_{d_2} &= S_1 \& S_2 \& S'_3 \& S'_4 \\ ELD_{d_3} &= S'_1 \& S_2 \& S_3 \& S'_4 \\ ELD_{d_4} &= S'_5 \& S_6 \& S_7 \& S'_8 \end{aligned} \tag{3.4}$$

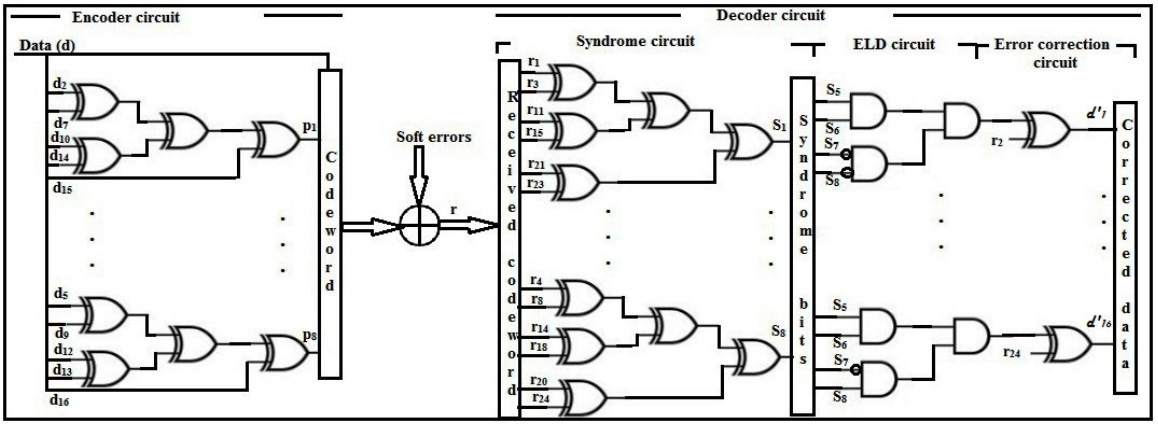

FIGURE 3.7: Gate Level Design of Proposed $(24, 16)$ SEC-DAEC Codec

The gate level design of proposed $(24, 16)$ SEC-DAEC codec is shown in Fig. 3.7.

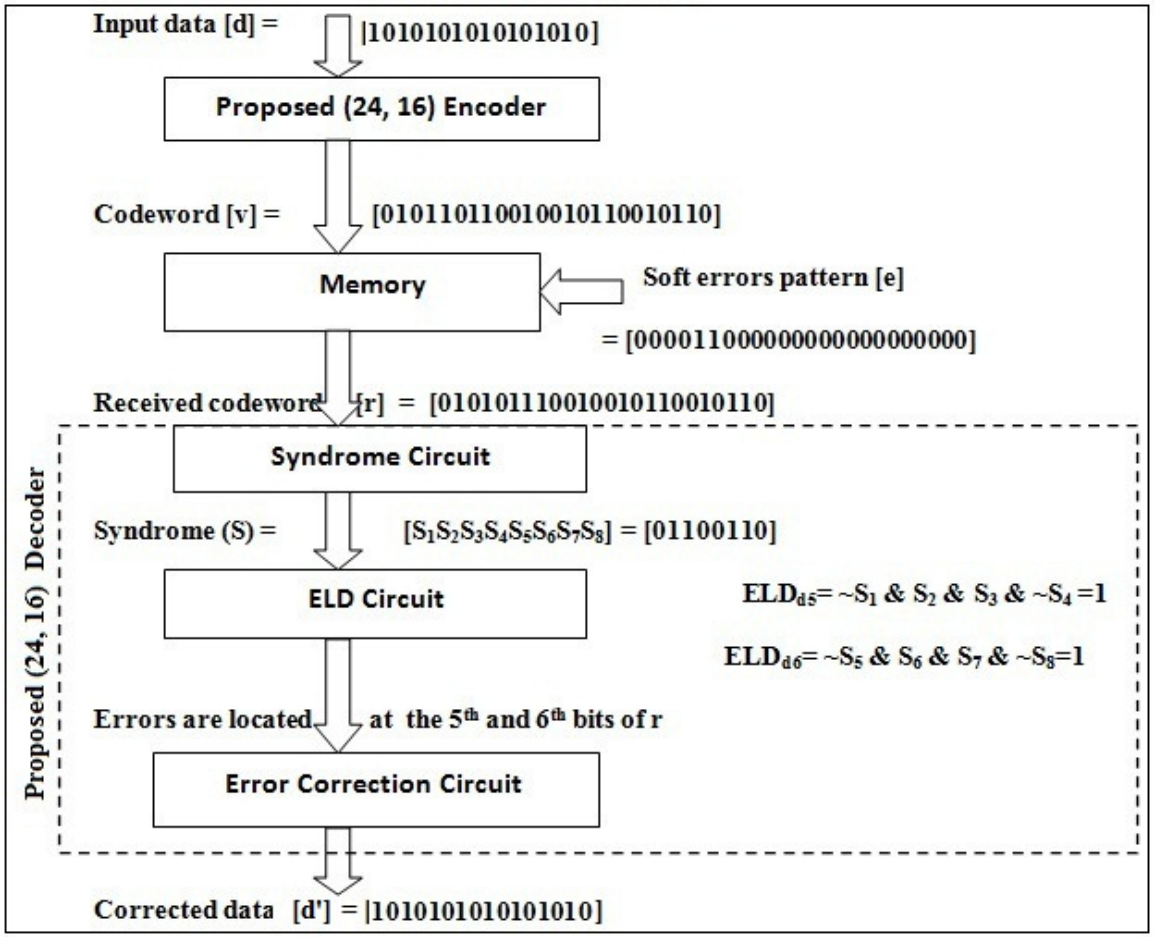

FIGURE 3.8: Illustration of Proposed $(24, 16)$ SEC-DAEC Codec

The proposed $(24, 16)$ codec has been illustrated by an example here. The encoding and decoding operations of proposed $(24, 16)$ codec has been explained by using Fig. 3.8. The proposed $(24, 16)$ encoder takes an arbitrary input data (d)=(1010101010101010) and produces the codeword (v)=(010110110010010110010110) to be stored in the memory as shown in Fig. 3.8. Assume the soft errors pattern (e)=(000011000000000000000000) has been occurred for a particular time instance. The received codeword (r) at the decoder input is expressed as: $r{=}v \oplus e$ =(010101110010010110010110).

The syndromes (S) can be expressed as $S{=}r.H^T$=(01100110) for this received codeword (r). As per the proposed decoding rules, the first four syndrome bits are matching with upper half of fifth column of the proposed $(24, 16)$ H-matrix. Whereas the last four syndrome bits are exactly same as the lower half of the sixth column in the H-matrix. So the errors are located in the 5^{th} and 6^{th} bits of the received codewords (r) and these bits are corrected by employing XOR gates. Finally, the decoder output is the corrected version of the data.

3.3.3 Theoretical Gate Count of Proposed Codec

Proposed codes have been designed for different word lengths based on the H-matrices presented in Fig. 3.3 to Fig. 3.5. Theoretical hardware and delay estimation of proposed codecs have been computed in terms of total number of 1's in the H-matrix and the highest number of 1's in a row of the H-matrix. Also theoretical hardware requirements of proposed codecs have been compared with other related codecs. The comparison of total count of 1's in the proposed H-matrices and highest count 1's in a row of H-matrices are shown in Table 3.1. As shown in Table 3.1, the total number of 1's which are required to design the proposed H-matrices are lower compared to the SEC-DAEC codes presented in [133], [129], [130], [132]. Also it can be observed from Table 3.1 that the heaviest row of proposed H-matrices have the lowest weight compared to all other codes.

TABLE 3.1: Comparison of Total Number of 1's and Highest Number of 1's in a Row of Proposed H-matrix

k	Code	Total count of 1's in H-matrix	Highest count of 1's per row in H-matrix
16	**Dutta (22, 16)** [133]	54	10
	Neale (23, 16) [129], [130]	49	9
	Reviriego (23, 16) [132]	55	11
	Proposed (24, 16)	44	6
32	**Dutta (39, 32)** [133]	103	15
	Neale (39, 32) [129], [130]	105	18
	Reviriego (39, 32) [132]	103	16
	Proposed (42, 32)	86	9
64	**Dutta (72, 64)** [133]	232	32
	Neale (72, 64) [129], [130]	219	30
	Reviriego (73, 64) [132]	201	26
	Proposed (76, 64)	174	15

The total counts of equivalent 2-input NAND gates (NAND2) required to design the proposed codecs are shown in Table 3.2 which provide theoretical area estimation of proposed codecs as a whole. The improvement percentage or Improv. (%) in Table 3.2 has been calculated by employing equation 3.5.

$$\textbf{Improv. (\%)} = \frac{\textbf{EV} - \textbf{PV}}{\textbf{PV}} \times \textbf{100\%}, \tag{3.5}$$

where EV and PV represent existing value and proposed value respectively. The proposed $(24, 16)$ codec has the highest improvement in area against existing 16-bit codecs. The highest area improvement of proposed codecs for 32 and 64 bits word lengths are obtained against the respective word length of Dutta et al.'s codec. The numbers of equivalent NAND-2 gates required to design the proposed and various existing codecs have been compared graphically in Fig. 3.9. From this figure, it can be concluded that proposed codecs require lesser numbers of equivalent NAND2 gates compared to related existing codecs. Thus the proposed codecs for each value of word length are expected to provide lower area, delay and power consumption in synthesis results compared to the existing codecs

TABLE 3.2: Theoretical Area Analysis of Proposed Codes in terms of Logic Gates

k	Codec	XOR2	OR2	AND2	NOT	Equiv. NAND2	Improv. (%)
16	**Dutta (22, 16)** [133]	106	31	235	126	1113	150
	Neale (23, 16) [129], [130]	93	31	282	173	1202	170
	Reviriego (23, 16) [132]	101	31	282	1	1062	139
	Proposed (24, 16)	80	0	48	28	444	-
32	**Dutta (39, 32)** [133]	217	63	570	321	2518	144
	Neale (39, 32) [129], [130]	221	62	564	305	2503	143
	Reviriego (39, 32) [132]	223	62	282	1	1643	59
	Proposed (42, 32)	174	0	128	84	1036	-
64	**Dutta (72, 64)** [133]	504	127	1337	777	5848	147
	Neale (72, 64) [129], [130]	478	126	1330	714	5664	140
	Reviriego (73, 64) [132]	450	127	573	1	3328	41
	Proposed (76, 64)	376	0	320	222	2366	-

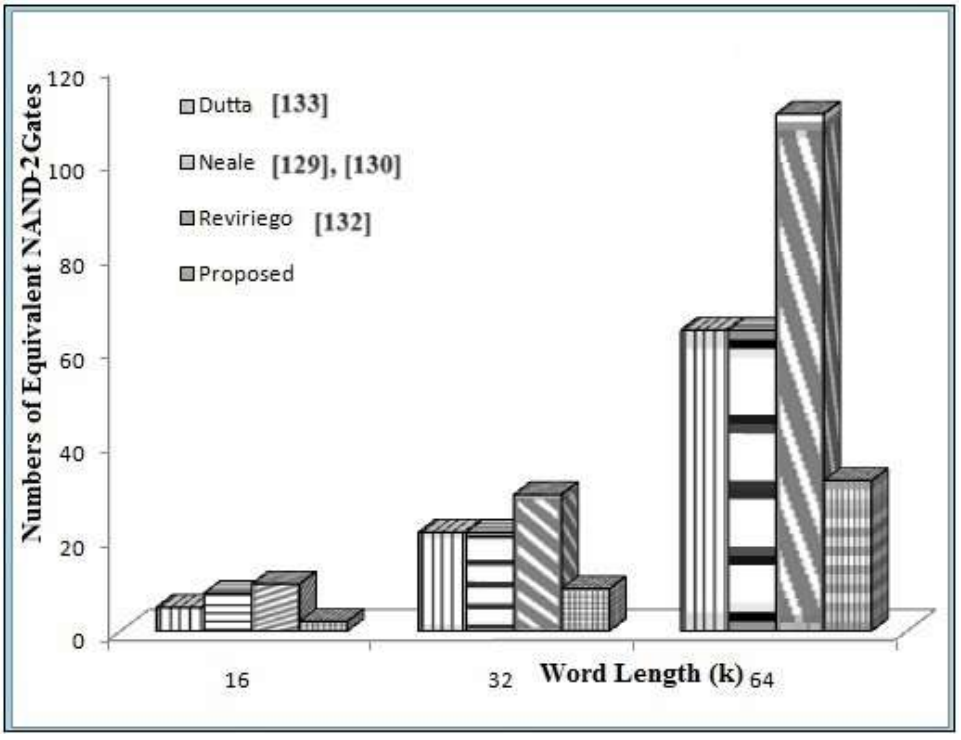

FIGURE 3.9: Graphical Comparison of Area (NAND2 Gates)

3.3.4 Results and Analysis

In this section, the analysis of overall error correction capability of proposed scheme have been presented first for the sake of fair comparisons between proposed and existing codes in terms of circuit overheads. The analysis of synthesis results have also been presented in this section.

3.3.4.1 Overall Error Correction Capability

The proposed and the existing codes in [129]- [130], [132] and [133] have the capability of correcting single and double-adjacent errors. These errors are also called as two-bit Burst Errors (2-bit BE). Thus the proposed and existing codes namely Dutta et al., Neale et al. and Revirigeo et al. are capable of correcting BE of length 2-bit completely. Apart from that, these codes are also capable of correcting certain percentage of burst errors of lengths higher than 2-bit at the cost of additional circuitry. The t-bit Burst Error Correcting Capability (BECC_t) of any code can be computed by employing equation (3.6).

$$BECC_t = \frac{\text{Total Number of } t\text{-bit BE Corrected}}{\text{Total Number of } t\text{-bit BE Injected}} X100\% \tag{3.6}$$

Table 3.3 represents the burst error correcting capabilities of proposed and existing codes which have been computed by employing Matlab simulator based on equation (3.6) for burst lengths of 3, 4 and 5 bits.

The performance of proposed and existing codes based on their burst error correcting capabilities degrades with the increase in burst length for word lengths of 16, 32 and 64 bits as shown in Table 3.3. The main aim of the proposed code is to reduce the decoding overheads while correcting single and double-adjacent errors. Half of the total syndrome bits have been employed in proposed decoding scheme to achieve lower decoding overheads. Due to this, the proposed codes with the proposed Half Length (HL) decoding technique provide the worst burst error correction capability.

TABLE 3.3: Comparison of Burst Error Correcting Capability of Different Schemes

Code	k	Overall Error Correcting Capability (%)				
		Single Error	2-bit Burst Error	3-bit Burst Error	4-bit Burst Error	5-bit Burst Error
Dutta [133]	**16**	100	100	62.65	37.11	21.12
Neale [129], [130]		100	100	86.21	59.28	36.68
Reviriego [132]		100	100	71.23	52.10	34.48
Proposed (HL)		100	100	30.77	16.00	09.55
Proposed (FL)		100	100	78.02	70.71	61.49
Dutta [133]	**32**	100	100	69.23	42.46	23.63
Neale [129], [130]		100	100	63.58	41.02	21.74
Reviriego [132]		100	100	65.56	40.34	21.91
Proposed (HL)		100	100	34.36	17.56	09.63
Proposed (FL)		100	100	81.60	77.55	72.23
Dutta [133]	**64**	100	100	69.23	42.46	23.63
Neale [129], [130]		100	100	63.25	41.32	22.21
Reviriego [132]		100	100	68.69	50.62	33.78
Proposed (HL)		100	100	32.11	16.24	09.26
Proposed (FL)		100	100	80.27	75.25	69.32

But the H-matrices of proposed codes have been constructed such a manner that the same codes are also decodable with Full Length (FL) conventional decoding scheme. And the proposed codes with FL decoding scheme can provide the best performance in term of burst error correcting capability compared to existing schemes in Table 3.3. The proposed codes can be used either with higher overall correction capabilities or with lower decoding overheads based on the needs of the application.

3.3.4.2 ASIC-based Synthesis Results

The proposed SEC-DAEC codecs have been implemented in verilog Hardware Description Language (HDL). These codecs are simulated and synthesized on ASIC environment. The EDA tool "Cadence Genus Synthesis Solution" has been utilized for the purpose of ASIC synthesis. This ASIC synthesis has been carried out in 45nm technology node with the help of "gpdk045bc" technology libraries. The supply voltage of 1.08V and clock frequency of 10 MHz have been applied during the ASIC synthesis. The ASIC-based synthesis results of proposed and related existing codecs have been summarized in Table 3.4. The proposed codecs are more compact with respect to the existing codecs for word lengths: 16, 32, 64 bits as shown in

TABLE 3.4: Comparison of ASIC-based Synthesis Results for Different Schemes

k	Codec	Area (μm^2)	Delay (ps)	Power (μW)	Improvement (%)		
					Area	Delay	Power
16	**Dutta (22, 16)** [133]	537	471	54	27	39	28
	Neale (23, 16) [129], [130]	527	501	67	26	43	42
	Reviriego (23, 16) [132]	533	480	89	27	40	56
	Proposed (24, 16)	390	288	39	-	-	-
32	**Dutta (39, 32)** [133]	1087	498	178	27	23	46
	Neale (39, 32) [129], [130]	1000	538	181	21	29	46
	Reviriego (39, 32) [132]	1038	512	253	24	25	62
	Proposed (42, 32)	793	383	97	-	-	-
64	**Dutta (72, 64)** [133]	2065	576	430	24	21	45
	Neale (72, 64) [129], [130]	2062	571	438	24	20	46
	Reviriego (73, 64) [132]	2027	592	651	23	23	64
	Proposed (76, 64)	1568	456	237	-	-	-

Table 3.4. But the highest benefit in area has been obtained against Dutta et al.'s 16-bit codec. The proposed codecs are faster with respect to related published codecs for all word lengths. The proposed 16-bit codec provides the highest delay advantage with respect to Neale et al.'s 16-bit codec. In case of power consumption, the proposed codecs have the least power requirement for three different word lengths compared to existing codecs as listed in Table 3.4. But the highest improvement in power consumption has been observed against 64-bit Reviriego et al. codec. As a whole, the proposed codecs are compact, faster and consume lower power compared to related existing codecs for all word lengths.

3.3.4.3 *ADPR*-metric

The proposed codecs have significant improvements in area, delay and power consumption but these codecs have higher redundancy in contrast to existing codecs. Therefore, it is apparent that proposed codecs have area, delay and power benefits at the cost of their increased redundancy. An analysis by considering the parameters like area, delay, power and redundancy is very much essential at this point. The *ADPR*-metric is defined which is product of four parameters and which is defined

in equation (3.7).

$$ADPR = \quad ADP \text{ x } R \tag{3.7}$$

$$where \quad ADP = \quad Area \text{ x } Delay \text{ x } Power \tag{3.8}$$

and R is the fractional redundancy which has been defined for an (n, k) code in equation (2.6).

The $ADPR$-metric of proposed and existing codecs have been computed in terms of "attometer square Watt second" ($am^2.W.s$) or "attometer square Joule" ($am^2.J$) based on Table 3.4 and these results are summarized in Table 3.5. The proposed codecs have lowest value of Area, Delay and Power product (ADP) for all word lengths compared to the existing codecs as shown in Table 3.5. This is the main advantage of proposed codecs. But the proposed codes have higher number of parity bits which in turn increases the fractional redundancy compared to those minimum redundancy existing codes as listed in Table 3.5. Therefore, extra memory space is required to store these excess parity bits of proposed codes. As a result, the additional area for these extra parity bits can be treated as the main demerit of the proposed codes. In spite of that, the proposed codecs have lowest value of $ADPR$-metric compared to the existing codecs presented in Table 3.5 for different word lengths. Thus it can be concluded that the main merit of the proposed codecs have been the lower area, delay and power benefits. The graphical comparison of $ADPR$-metric for presented and existing codecs have been shown in Fig. 3.10.

3.4 Design of Compact Error Location Detection Block for DAEC Decoder

This section presents a method to design lower complexity error location detection block for various existing SEC-DAEC decoders. The main objective of the proposed

TABLE 3.5: Comparison of $ADPR$-metric for Different Schemes

k	Code	ADP $(am^2.J)$	R	$ADPR$ $(am^2.J)$
16	**Dutta (22, 16)** [133]	14	0.375	5
	Neale (23, 16) [129], [130]	18	0.438	8
	Reviriego (23, 16) [132]	23	0.438	10
	Proposed (24, 16)	4	0.500	2
32	**Dutta (39, 32)** [133]	96	0.219	21
	Neale (39, 32) [129], [130]	97	0.219	21
	Reviriego (39, 32) [132]	134	0.219	29
	Proposed (42, 32)	29	0.313	9
64	**Dutta (72, 64)** [133]	511	0.125	64
	Neale (72, 64) [129], [130]	516	0.125	64
	Reviriego (73, 64) [132]	781	0.141	110
	Proposed (76, 64)	169	0.188	32

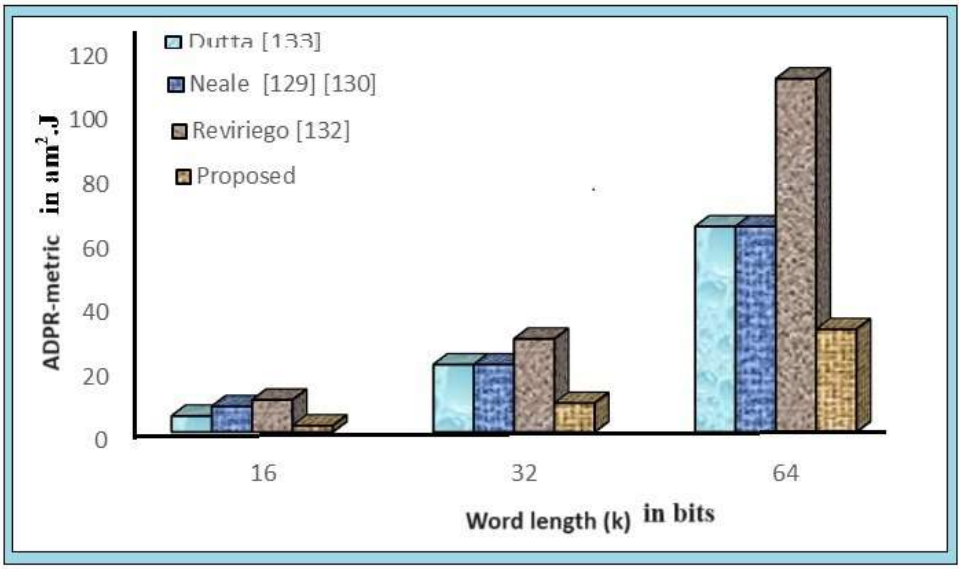

FIGURE 3.10: Graphical Comparison of $ADPR$-metric for Different Schemes

work is to make the existing decoders for SEC-DAEC codes more compact in terms of logic gates. In this work, Karnaugh map (K-map) has been employed to simplify the logical expressions for the ELD block of existing decoders. But syndrome and error correction circuitry for existing and proposed schemes are exactly same. In the proposed scheme, the ELD block of conventional SEC-DAEC (n, k) decoders have been simplified prior to their implementation by using following steps:

Step 1: The syndromes and corresponding error locations are listed in a table which include all possible $(n - k)$ bits syndromes and their respective error positions.

Step 2: Unused syndromes are identified from the syndromes and corresponding error locations table and considered as a don't care conditions for K-map.

Step 3: The K-map is solved for each data bit position using same don't care conditions but different position of 1's according to the affected syndromes for the specific data bit position.

Step 4: Simplified expressions obtained in step 3 for all data positions are employed to design the error location block of the proposed scheme.

3.4.1 An Illustrative Example

c4	d1	c6	d2	d3	d4	d5	d6	d7	c5	d8	c7	d9	d10	c2	d11	c1	d12	d13	c3	d14	d15	d16
0	0	0	0	1	0	1	0	0	0	0	0	0	1	0	1	1	0	1	0	1	1	1
0	1	0	1	0	1	0	1	0	0	0	0	1	0	1	0	0	0	0	0	1	1	1
0	1	0	1	1	0	1	0	1	0	1	0	0	0	0	0	0	1	0	1	0	0	0
1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0
0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0
0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1
0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0

FIGURE 3.11: H-matrix of SEC-DED-DAEC $(23, 16)$ Code

As an example, the design of compact SEC-DAEC $(23, 16)$ decoder based on H-matrix shown in Fig. 3.11 is presented here. The seven bits syndrome are computed using equation (2.3) . The syndromes and corresponding error locations table are enlisted in Table 3.6 using all possible combinations of seven bits syndrome (synd.)

TABLE 3.6: Syndromes and Corresponding Error Locations for (23, 16) Compact Decoder

SL.	**Synd.**	**Error**	**SL.**	**Synd.**	Error	SL.	Synd.	Error	SL.	Synd.	Error
0	0000000	no error	32	0100000	X	64	1000000	X	96	1100000	X
1	0000001	r12	33	0100001	r8	65	1000001	r16	97	1100001	X
2	0000010	r3	34	0100010	r15	66	1000010	r19	98	1100010	r23
3	0000011	X	35	0100011	X	67	1000011	X	99	1100011	r15,r16
4	0000100	r10	36	0100100	r6	68	1000100	r14	100	1100100	r22
5	0000101	X	37	0100101	X	69	1000101	X	101	1100101	X
6	0000110	r22,r23	38	0100110	X	70	1000110	X	102	1100110	r14,r15
7	0000111	X	39	0100111	X	71	1000111	X	103	1100111	X
8	0001000	r1	40	0101000	r13	72	1001000	r17	104	1101000	r21
9	0001001	r16, r17	41	0101001	r12,r13	73	1001001	X	105	1101001	r4,r5
10	0001010	X	42	0101010	X	74	1001010	X	106	1101010	X
11	0001011	X	43	0101011	X	75	1001011	X	107	1101011	X
12	0001100	r21,r22	44	0101100	X	76	1001100	X	108	1101100	r13,r14
13	0001101	X	45	0101101	X	77	1001101	X	109	1101101	X
14	0001110	X	46	0101110	X	78	1001110	X	110	1101110	X
15	0001111	X	47	0101111	X	79	1001111	X	111	1101111	X
16	0010000	X	48	0110000	X	80	1010000	X	112	1110000	X
17	0010001	r20	49	0110001	r4	81	1010001	X	113	1110001	X
18	0010010	r11	50	0110010	X	82	1010010	r7	114	1110010	X
19	0010011	r11,r12	51	0110011	r3,r4	83	1010011	r19, r20	115	1110011	r7,r8
20	0010100	r18	52	0110100	r2	84	1010100	X	116	1110100	X
21	0010101	X	53	0110101	X	85	1010101	X	117	1110101	X
22	0010110	r10,r11	54	0110110	r2,r3	86	1010110	r18,r19	118	1110110	r6,r7
23	0010111	X	55	0110111	X	87	1010111	X	119	1110111	X
24	0011000	r9	56	0111000	X	88	1011000	r5	120	1111000	X
25	0011001	X	57	0111001	r8,r9	89	1011001	X	121	1111001	r20,r21
26	0011010	X	58	0111010	X	90	1011010	X	122	1111010	X
27	0011011	X	59	0111011	X	91	1011011	X	123	1111011	X
28	0011100	r9,r10	60	0111100	r1,r2	92	1011100	r17.r18	124	1111100	r5,r6
29	0011101	X	61	0111101	X	93	1011101	X	125	1111101	X
30	0011110	X	62	0111110	X	94	1011110	X	126	1111110	X
31	0011111	X	63	0111111	X	95	1011111	X	127	1111111	X

and their respective error position in stored codeword. Unused syndromes are identified in the Table 3.6 and utilized as don't care conditions for K-map. The K-map has been solved for 16 times (using Logic Friday software) to obtain the simplified ELD expressions of 16 data bits.

As an example, for first data position d_1 as indicated in Fig. 3.11, the syndrome values correspond to single error in d_1 as well as double adjacent errors that includes d_1 are listed in serial numbers 52, 54 and 60 respectively in the Table 3.6. Now, the K-map is solved to obtain simplified expression of ELD logic for the first data bit

by keeping those respective positions as 1's in the K-map and with all don't care conditions as indicated in Table 3.6 and filling the remaining positions in the K-map by 0's. Similar procedure is repeated for all the data position. Therefore, position of 1's in the K-map are different from one data position to another before solving the K-map. Figure 3.12 shows the K-map grouping to obtain simplified expression of ELD logic for the first data bit. The simplified expressions of ELD logic for each data position are obtained similarly by solving K-maps and these expressions are summarized in equation (3.9).

$$
\begin{aligned}
ELD_{d1} &= (S_1' S_2 S_3 S_7') \\
ELD_{d2} &= (S_1 S_3' S_4 S_7) + (S_1' S_2 S_3 S_4' S_5') \\
ELD_{d3} &= (S_1 S_3' S_4 S_7) + (S_1 S_2' S_3 S_5' S_6') + (S_1 S_2 S_3 S_6' S_7') \\
ELD_{d4} &= (S_1' S_2 S_3' S_5) + (S_1 S_2 S_3 S_7') \\
ELD_{d5} &= (S_1 S_2 S_3 S_4') + (S_1 S_3 S_4' S_5' S_7') \\
ELD_{d6} &= (S_1' S_3 S_4 S_7) + (S_2 S_3' S_4' S_5' S_6') + (S_1 S_2 S_3 S_4' S_5') \\
ELD_{d7} &= (S_1' S_2' S_3 S_4) + (S_1' S_2 S_3 S_4 S_5') \\
ELD_{d8} &= (S_1' S_2' S_3 S_6) \\
ELD_{d9} &= (S_1' S_2 S_3' S_4) + (S_2 S_3' S_4 S_5) \\
ELD_{d10} &= (S_2 S_3' S_5 S_6) + (S_2 S_3' S_4 S_5) + (S_1 S_2' S_3' S_5) \\
ELD_{d11} &= (S_2' S_4 S_7) + (S_1 S_3' S_4' S_7) \\
ELD_{d12} &= (S_1 S_2' S_3 S_5) + (S_2' S_3 S_4' S_5 S_6') \\
ELD_{d13} &= (S_1 S_2' S_3' S_6) + (S_1 S_2' S_6 S_7) + (S_1 S_2' S_5 S_6) \\
ELD_{d14} &= (S_1' S_3' S_4 S_5) + (S_1 S_3 S_6' S_7) + (S_1 S_2 S_5' S_6' S_7') \\
ELD_{d15} &= (S_1' S_3' S_5 S_6) + (S_1' S_3' S_4 S_5) + (S_1 S_2 S_4' S_6') \\
ELD_{d16} &= (S_1' S_3' S_5 S_6) + (S_1 S_2 S_4' S_5' S_7')
\end{aligned}
\tag{3.9}
$$

Now these simplified expressions are used to design the ELD block of the compact

decoder. Finally, the error associated with each bit is corrected by error correction block which perform XOR operation between ELD block output and received data. The error correction logic for the first data bit (d_1) which has received as (r_2) in compact decoder is expressed as:

$$d_{c1} = r_2 \oplus (S'_1 S_2 S_3 S'_7) \tag{3.10}$$

The expression for error correction in the first data bit of same $(23, 16)$ code without K-map i.e. by using conventional decoding method for variable column weight codes is given by:

$$d_{c1} = r_2 \oplus ((S'_1 S_2 S_3 S'_4 S_5 S'_6 S'_7) + (S'_1 S_2 S_3 S_4 S_5 S'_6 S'_7) + (S'_1 S_2 S_3 S'_4 S_5 S_6 S'_7)) \tag{3.11}$$

The equations (3.10) and (3.11) clearly show that when K-map is used then there is significant reduction in numbers of gates required to implement ELD block of the decoder. The compact decoder for SEC-DAEC $(23, 16)$ code is shown in Fig. 3.13.

The degree of logic simplification totally depends on grouping of 1's corresponds to the syndrome values for a particular data bit and don't care conditions. Logic simplification is maximum for a data bit when the syndrome values correspond to that data bit forms maximum possible groups among them with the help of don't care conditions in K-map. As a result, the degree of simplification in ELD block of a decoder can vary from one data bit to another and from one code to another code. The K-map based simplification method has been applied to minimize the decoding logic expressions of different DAEC codes such as SEC-DED-DAEC-5AED $(23, 16)$ code, SEC-DED-DAEC-3AED $(39, 32)$ code, and SEC-DED-DAEC-3AED $(72, 64)$ code by Neale et al. [129], [130] and SEC-DED-DAEC codes by Reviriego et al. [132].

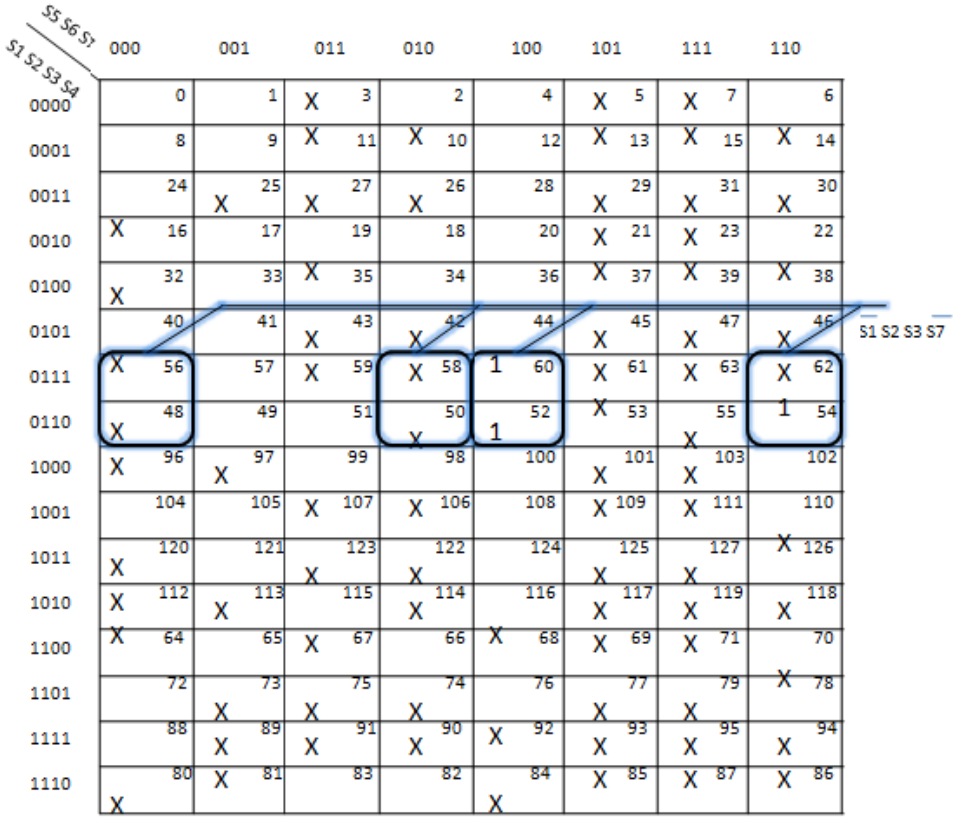

FIGURE 3.12: Illustration of 7 Variables K-map for obtaining ELD_{d1} of SEC-DED-DAEC $(23, 16)$ Decoder

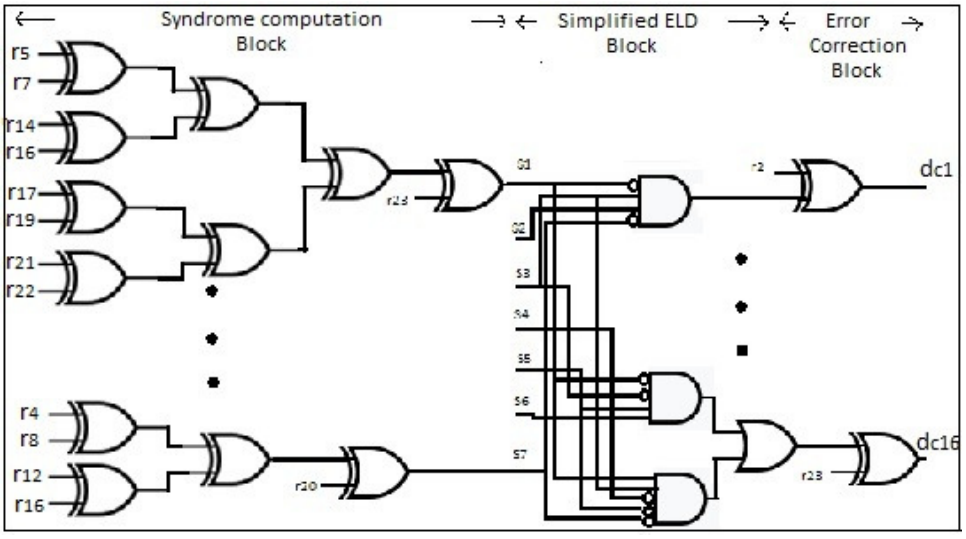

FIGURE 3.13: Schematic of Compact $(23, 16)$ SEC-DAEC Decoder

TABLE 3.7: Area of Existing and Compact Decoders in terms of Logic Gates

Scheme	k	Code	Existing Decoders					CompactDecoder					Improv. (%)
			XOR2	OR2	AND2	NOT	Equiv. NAND2	XOR2	OR2	AND2	NOT	Equiv. NAND2	
DAEC	16	Neale $(23,16)$ [129], [130]	58	31	282	173	1062	58	20	117	67	593	44.16
		Reviriego $(23,16)$ [132]	64	31	189	64	791	64	25	136	56	659	16.69
	32	Neale $(39,32)$ [129], [130]	130	62	564	305	2139	130	55	358	198	1599	25.25
		Reviriego $(39,32)$ [132]	128	62	381	128	1588	128	60	338	123	1491	6.11
	64	Neale $(72,64)$ [129], [130]	275	126	1330	714	4852	275	114	892	509	3735	23.02
		Reviriego $(73,64)$ [132]	256	127	893	384	3575	256	116	671	278	2992	16.31

3.4.2 Theoretical Gate Count of Compact Decoder

Proposed decoders require lesser number of AND, OR and NOT gates compared to traditional decoder for designing the ELD block. There is no change in logic to compute syndromes for both the decoders. The number of gates required to implement Neale et al. and Reviriego et al. SEC-DED-DAEC decoders using conventional and proposed approaches are shown in Table 3.7 .

The improvement in terms of number of logic gates is highest for Neale et al. SEC-DED-DAEC $(23,16)$ code and it reduces gradually for $(39,32)$ and for $(72,64)$ codes as shown in Table 3.7. The SEC-DED-DAEC codes with second scheme of optimized decoding by Reviriego et al. exhibits better improvement for $(23,16)$ and $(73,64)$ codes compared to $(39,32)$ code. Graphical representation of area improvements in terms of number of equivalent NAND2 logic gates is presented in Fig. 3.14. Here, compact SEC-DAEC decoders are compared with existing decoders. Area improvement is maximum for $(23,16)$ Neale et al. code among all SEC-DED-DAEC codes as shown in Fig. 3.14. Also the performance of compact decoder is better for SEC-DAEC presented by Neale et al. compared to the codes proposed by Reviriego et al.. This is due to the fact that the SEC-DAEC code in [132] applies optimized decoding scheme.

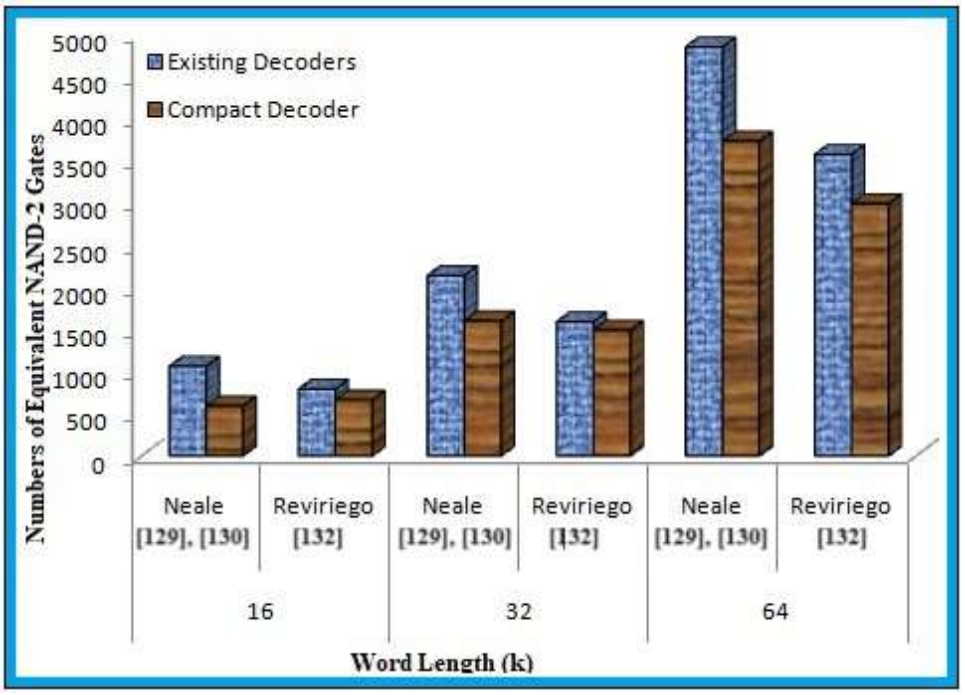

FIGURE 3.14: Graphical Comparison of Equivalent NAND-2 Gate Count for Different Decoders

3.4.3 Synthesis Results and Comparisons

The proposed and existing SEC-DAEC decoders are simulated and synthesized on FPGA and ASIC platforms. In this section, the obtained synthesis results have been analyzed in details.

3.4.3.1 FPGA-based

All the functional blocks are simulated and synthesized using FPGA based virtex 6 (6vcx75tff484-2) device. The FPGA based synthesis results have been shown in Table 3.8. The proposed compact decoder requires lesser area compared to the existing SEC-DAEC decoder as listed in Table 3.8. But the highest improvement in terms of LUTs for SEC-DED-DAEC code is achieved with respect to $(23, 16)$ Neale et al. decoder. Similarly, the delay performances of proposed compact decoders are

TABLE 3.8: FPGA-based Synthesis Results of SEC-DAEC Decoders

Scheme	k	Code	Existing Decoders		Compact Decoder		Improv (%)	
			Area (LUTs)	Delay (ns)	Area (LUTs)	Delay (ns)	Area	Delay
DAEC	16	**Neale** $(23, 16)$ [129], [130]	51	3.31	44	3.04	21.57	8.16
		Reviriego $(23, 16)$ [132]	53	3.48	43	3.28	18.87	5.75
	32	**Neale** $(39, 32)$ [129], [130]	108	5.29	88	4.51	18.52	14.74
		Reviriego $(39, 32)$ [132]	102	3.96	92	3.67	9.8	7.32
	64	**Neale** $(72, 64)$ [129], [130]	210	5.4	191	4.95	9.05	8.33
		Reviriego $(73, 64)$ [132]	216	4.08	205	4.2	5.09	-2.94

better compared to Neale et al. and Reviriego et al. decoders for the word lengths of 16-bit and 32-bit. The Reviriego et al $(73, 64)$ SEC-DED-DAEC decoder provides slight delay improvements in FPGA-based synthesis results. Whereas, the maximum improvement in terms of delay is achieved against Neale et al. SEC-DED-DAEC $(39, 32)$ decoder.

3.4.3.2 ASIC-based

The existing and proposed compact SEC-DAEC decoders have been further synthesized in ASIC platform using Cadence based Genus synthesis solution (TSMC18) tool. The ASIC-based synthesis results have been presented in Table 3.9 which shows similar trends of improvements in terms of area and delay of proposed decoder as shown in FPGA-based synthesis result. But power consumption for the compact decoder is slightly higher compared to existing decoder for almost all SEC-DED-DAEC codes listed in Table 3.9 except SEC-DED-DAEC $(23, 16)$ and $(73, 64)$ codes [18] for which a slight improvement in power has also been observed. In brief, the proposed compact decoders provide considerable reduction in area and delay of decoder at the cost of slight increase in power consumption.

TABLE 3.9: ASIC-based Synthesis Results of SEC-DAEC Decoders

Scheme	k	Code	Existing Decoders			Compact Decoder			Improvement (%)		
			Area (μm^2)	Delay (ps)	Power (mW)	Area (μm^2)	Delay (ps)	Power (mW)	Area	Delay	Power
DAEC	16	**Neale** $(23, 16)$ [129], [130]	2089	389	0.37	1556	350	0.38	25.50	10.07	-2.70
		Reviriego $(23, 16)$ [132]	2089	471	0.45	1963	452	0.42	6.05	4.03	6.67
	32	**Neale** $(39, 32)$ [129], [130]	3719	617	0.81	3059	449	0.94	17.74	27.17	-16.05
		Reviriego $(39, 32)$ [132]	3998	620	1.08	3885	611	1.11	2.83	1.45	-2.78
	64	**Neale** $(72, 64)$ [129], [130]	7468	919	1.8	6554	872	2.02	12.23	4.33	-12.22
		Reviriego $(73, 64)$ [132]	7681	712	2.78	7512	674	2.63	2.20	5.34	4.71

3.5 Design of an Area and Power Efficient DAEC Decoder based on $(24, 12)$ Extended Golay Code

In this section, the hardware complexity of parallel decoding scheme [128] for double adjacent error correction based on $(24, 12)$ extended Golay Code has been further reduced by combining the parallel decoding scheme with the area optimized decoding scheme [132].

3.5.1 Overview of $(24, 12)$ Extended Golay Code

The $(23, 12)$ Golay code is only another binary perfect code like the Hamming codes [2]. This code has a minimum distance of 7 and capable of correcting three or fewer random errors. The $(24, 12)$ Golay code is the extended version of $(23, 12)$ code in which an overall parity bit is added. This $(24, 12)$ extended Golay code has a minimum distance of 8 but it is not a perfect code. The generator matrix (G) for the extended Golay code is in the form given in equation (3.12).

$$G = [P \quad I_{12}] \tag{3.12}$$

where P and I represents parity matrix and identity matrix respectively. The P matrix is a diagonally symmetric matrix i.e. $P = P^T$. The parity check matrix (H) of extended Golay code can be written from equation (3.12) and which is shown in

equation (3.13).

$$H = [I_{12} \quad P^T] = [I_{12} \quad P] \tag{3.13}$$

The H-matrix of $(24, 12)$ extended Golay code is shown in Fig. 3.15. Encoder circuit of $(24, 12)$ extended Golay code compute the 12 parity bits with the help of data bits and H-matrix in Fig. 3.15. These 12 parity bits are combined with 12 data bit to form codewords which are stored in memory for its protection against soft errors. Decoder circuit for $(24, 12)$ extended Golay code read the stored codewords in memory and produce corrected version of original data with the help of syndrome (S). The syndromes are computed based on H-matrix by employing equation (2.3).

$$H = \left[\begin{array}{l} 100000000000110111000101 \\ 010000000000011011100011 \\ 001000000000101101110001 \\ 000100000000010110111001 \\ 000010000000001011011101 \\ 000001000000000101101111 \\ 000000100000100010110111 \\ 000000010000110001011011 \\ 000000001000111000101101 \\ 000000000100011100010111 \\ 000000000010101110001011 \\ 000000000001111111111110 \end{array}\right]$$

FIGURE 3.15: H-matrix of $(24, 12)$ Extended Golay Code

3.5.2 Proposed $(24, 12)$ SEC-DAEC Code

The main objective of present work is to derive a $(24, 12)$ SEC-DAEC code from the $(24, 12)$ extended Golay code. It is observed that Hamming weight of all the columns in H-matrix of $(24, 12)$ extended Golay code as shown in Fig. 3.15 is not constant. Also it is well known that decoding complexity for a ECC with constant column weight in H-matrix is lesser compared to H-matrix with variable weight columns. Thus an effort has been made here to obtain a new H-matrix of $(24, 12)$

SEC-DAEC code with constant column weight.

The proposed H-matrix of $(24, 12)$ SEC–DAEC code has been constructed using

$$H = \begin{bmatrix} 100000000000110111000101 \\ 010000000000011011100010 \\ 001000000000101101110000 \\ 000100000000010110111000 \\ 000010000000001011011101 \\ 000001000000000101101110 \\ 000000100000100010110110 \\ 000000010000110001011011 \\ 000000001000111000101101 \\ 000000000100011100010111 \\ 000000000010101110001011 \\ 000000000001111111111111 \end{bmatrix}$$

FIGURE 3.16: Proposed H-matrix of $(24, 12)$ SEC-DAEC Code

the following rules:

1. All the columns in H-matrix are nonzero and distinct.

2. All the data columns in H-matrix have constant Hamming weight of 7.

3. The XOR sum of any two adjacent columns involving a data column is distinct and different from any column in H-matrix and has a constant Hamming weight of 6.

The Single Error Correction (SEC) feature of H-matrix is confirmed by condition 1 and 2. Condition 3 provides the Double Adjacent Error Correction (DAEC) feature. Also the each data column has a constant odd Hamming weight 7 and each parity column has a constant odd weight is equals to 1. Therefore, if single error occurs in any bit of received codeword then corresponding syndrome has odd weight but the syndrome has even weight when double error occurs. This confirm the DED functionality of the code. Hence, the proposed $(24, 12)$ SEC-DAEC code is an odd column weight code and the H-matrix of the code is given in Fig. 3.16.

The proposed H-matrix of $(24, 12)$ SEC-DED-DAEC code has been derived from the H-matrix of $(24, 12)$ extended Golay code but the proposed code does not hold the

properties of Golay code. Also the minimum distance of the proposed SEC-DAEC code is 6 i.e. the proposed code can correct any combination of two or fewer error in a word length of 24.

3.5.3 Parallel Decoding of Proposed $(24, 12)$ SEC-DAEC Code

The proposed $(24, 12)$ SEC-DAEC decoder can be implemented in serial or parallel design technique. But the parallel decoding technique is more compact and fast compared to serial design technique. Hence the decoding of proposed $(24, 12)$ SEC-DAEC code is based on parallel decoding of $(24, 12)$ extended Golay code as presented in [128]. First of all, the proposed H-matrix is bit replaced by arranging alternate parity and data column. Bit replacement version of proposed H-matrix is shown in Fig. 3.17.

The twelve bits syndrome value correspond to single error in a particular data bit,

$$H = \begin{bmatrix} 110100010101000000010001 \\ 001101000101010000000100 \\ 010011010001010100000000 \\ 000100110100010101000000 \\ 000001001101000101010001 \\ 000000010011010001010100 \\ 010000000100110100010100 \\ 010100000001001101000101 \\ 010101000000010011010001 \\ 000101010000000100110101 \\ 010001010100000001001101 \\ 010101010101010101010111 \end{bmatrix}$$

FIGURE 3.17: Proposed H-matrix of $(24, 12)$ SEC-DAEC Code with Bit Replacement

double adjacent errors which include that data bit and triple adjacent errors with that data bit in middle position are same in ten bits but differ only in two bits. This is true for first eleven data bits in the codeword but for the last data bit the

corresponding syndromes are same for first eleven bits. Therefore, syndromes correspond to all three types of error (single, double adjacent and one triple adjacent) for any particular data bit differ in two or one positions only. The two or one positions where the syndrome values are different depend on particular data position i.e. in case of first and second data bit the respective positions are S_1, S_2 and S_2, S_3. Hence error location for any data position can be detected by considering the 10 or 11 bit window where the syndrome for single, two double adjacent and one triple adjacent errors are exactly same for that data position like the parallel decoding of $(24, 12)$ extended Golay code in [128]. But, in proposed $(24, 12)$ SEC-DAEC decoder error locations are detected considering only positions of 1's in the particular 10 or 11 bit window. Decoding of proposed $(24, 12)$ SEC-DED-DAEC code is based on the following rules.

1. Compute the syndrome.

2. Correct the data bits (first eleven) whose position of 1's in the corresponding 10-bit window match with the syndrome value (if any).

3. Correct the last data bit if the syndrome matches with the positions of 1's in 11-bit window (if any).

Now syndrome is computed using equation (2.3) and the syndrome expressions for the proposed $(24, 12)$ SEC-DED-DAEC code are shown in equation (3.14).

$$
\begin{aligned}
S_1 &= r_1 \oplus r_2 \oplus r_4 \oplus r_8 \oplus r_{10} \oplus r_{12} \oplus r_{20} \oplus r_{24} \\
S_2 &= r_3 \oplus r_4 \oplus r_6 \oplus r_{10} \oplus r_{12} \oplus r_{14} \oplus r_{22} \\
S_3 &= r_2 \oplus r_5 \oplus r_6 \oplus r_8 \oplus d_{12} \oplus r_{14} \oplus r_{16} \\
S_4 &= r_4 \oplus r_7 \oplus r_8 \oplus r_{10} \oplus d_{14} \oplus r_{16} \oplus r_{18} \\
S_5 &= r_6 \oplus r_9 \oplus r_{10} \oplus r_{12} \oplus r_{16} \oplus r_{18} \oplus r_{20} \oplus r_{24} \\
S_6 &= r_8 \oplus r_{11} \oplus r_{12} \oplus r_{14} \oplus r_{18} \oplus r_{20} \oplus r_{22} \\
S_7 &= r_2 \oplus r_{10} \oplus r_{13} \oplus r_{14} \oplus r_{16} \oplus r_{20} \oplus r_{22} \\
S_8 &= r_2 \oplus r_4 \oplus r_{12} \oplus \oplus r_{15} \oplus r_{16} \oplus \oplus r_{18} \oplus r_{22} r_{24} \\
S_9 &= r_2 \oplus r_4 \oplus r_6 \oplus r_{14} \oplus r_{17} \oplus r_{18} \oplus r_{20} \oplus r_{24} \\
S_{10} &= r_4 \oplus r_6 \oplus r_8 \oplus r_{16} \oplus r_{19} \oplus r_{20} \oplus r_{22} \oplus r_{24} \\
S_{11} &= r_2 \oplus r_6 \oplus r_8 \oplus r_{10} \oplus d_{18} \oplus r_{21} \oplus r_{22} \oplus r_{24} \\
S_{12} &= r_2 \oplus r_4 \oplus r_6 \oplus r_8 \oplus d_{10} \oplus ... \oplus r_{22} \oplus r_{24}
\end{aligned} \tag{3.14}
$$

Error location detection logic for each data bit in the proposed code is listed in Table 3.10. From Table 3.10, it is obvious that lesser number of 2-input AND (AND-2) gates are required to implement the error location detection part in the proposed parallel decoder of $(24, 12)$ SEC-DAEC code compared to the parallel decoder of $(24, 12)$ extended Golay code for single and double adjacent errors correction. Also the proposed parallel decoder do not requires any NOT gate in its error location detection part.

The parallel decoder for proposed $(24, 12)$ SEC-DED-DAEC code is shown in Fig. 3.18. The decoder circuit is composed of three function parts. The first part is the syndrome circuit which computes the syndrome values as per equation (2.3) by employing XOR gates. If error occurs in any data bit then the location of the error is detected by the second part of the decoder known as ELD circuit. This part of the decoder is designed using AND gates with the help of ELD logic expressions which

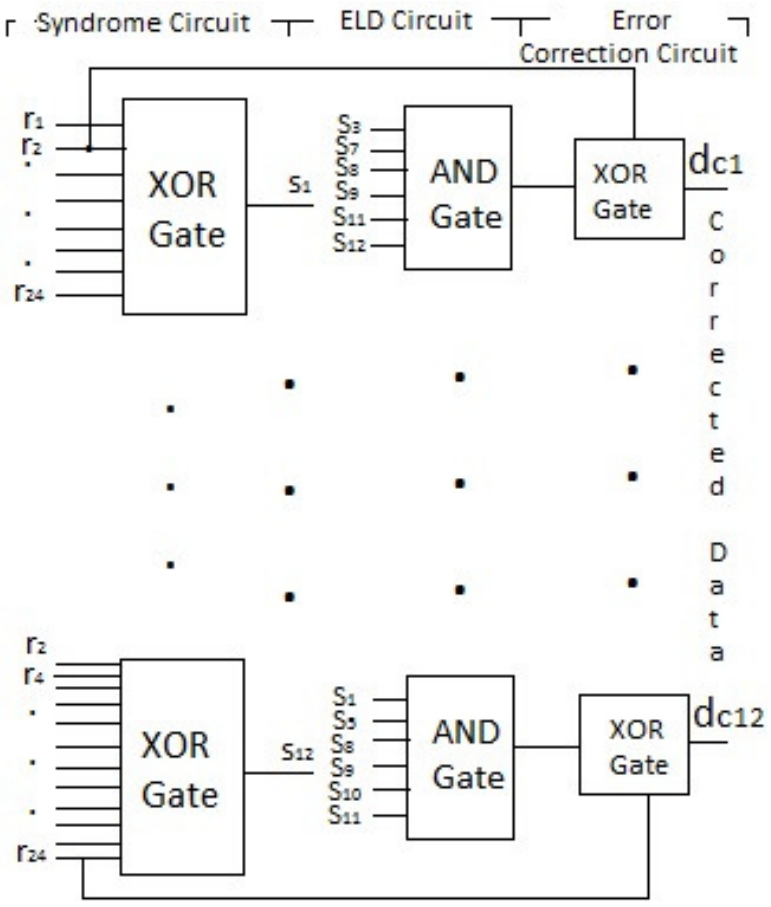

FIGURE 3.18: Parallel Decoder Circuit of Proposed $(24, 12)$ SEC-DAEC Code

are listed in Table 3.10. The last part of the decoder is the error correction circuit where XOR gates are employed to flip the bits in error.

3.5.4 Theoretical Analysis

In this section, the theoretical results in terms of number of logic gates required to implement the proposed parallel decoder of $(24, 12)$ SEC-DAEC code and number of logic gates required to establish the critical path of the decoder have been summarized. Theoretically, the proposed parallel decoder of $(24, 12)$ SEC-DAEC code requires lesser number of AND gates and does not needs any NOT gate compared to the parallel decoder in [128] for designing the ELD block. The number of gates required to implement the proposed parallel decoder of $(24, 12)$ SEC-DAEC code

TABLE 3.10: Error Positions and Corresponding Error Location Detection Logic

Sl. No.	Data Bit	Error Location Detection Logic
1	d_1	$s_3\ s_7\ s_8\ s_9\ s_{11}\ s_{12}$
2	d_2	$s_1\ s_4\ s_8\ s_9\ s_{10}\ s_{12}$
3	d_3	$s_2\ s_5\ s_9\ s_{10}\ s_{11}\ s_{12}$
4	d_4	$s_1\ s_3\ s_6\ s_{10}\ s_{11}\ s_{12}$
5	d_5	$s_1\ s_2\ s_4\ s_7\ s_{11}\ s_{12}$
6	d_6	$s_1\ s_2\ s_3\ s_5\ s_8\ s_{12}$
7	d_7	$s_2\ s_3\ s_4\ s_6\ s_9\ s_{12}$
8	d_8	$s_3\ s_4\ s_5\ s_7\ s_{10}\ s_{12}$
9	d_9	$s_4\ s_5\ s_6\ s_8\ s_{11}\ s_{12}$
10	d_{10}	$s_1\ s_5\ s_6\ s_7\ s_9\ s_{12}$
11	d_{11}	$s_2\ s_6\ s_7\ s_8\ s_{10}$
12	d_{12}	$s_1\ s_5\ s_8\ s_9\ s_{10}\ s_{11}$

and the parallel decoder of $(24, 12)$ SEC-DAEC extended Golay code in [128] are shown in Table 3.11. The critical path analysis of the proposed parallel decoder in terms of number of logic levels required to implement the largest path from the input to output of the decoder has also been presented in Table 3.12.

TABLE 3.11: Gate Count of Proposed $(24, 12)$ Parallel Decoder

Decoder	**XOR2**	**OR2**	**AND2**	**NOT**	**Equiv. NAND2**	**Improv. (%)**
Parallel (24, 12) Golay Decoder [128]	**100**	**0**	**109**	**45**	**663**	**24.28**
Proposed Parallel (24, 12) SEC-DAEC Decoder	**96**	**0**	**59**	**0**	**502**	

TABLE 3.12: Critical Path Analysis of Proposed $(24, 12)$ Parallel Decoder

Decoder	**XOR-2**	**OR2**	**AND-2**	**NOT**	**Equiv. NAND2**	**Improv. (%)**
Parallel (24, 12) Golay Decoder [128]	**12**	**0**	**9**	**5**	**71**	**12.68**
Proposed Parallel (24, 12) SEC-DAEC Decoder	**13**	**0**	**5**	**0**	**62**	

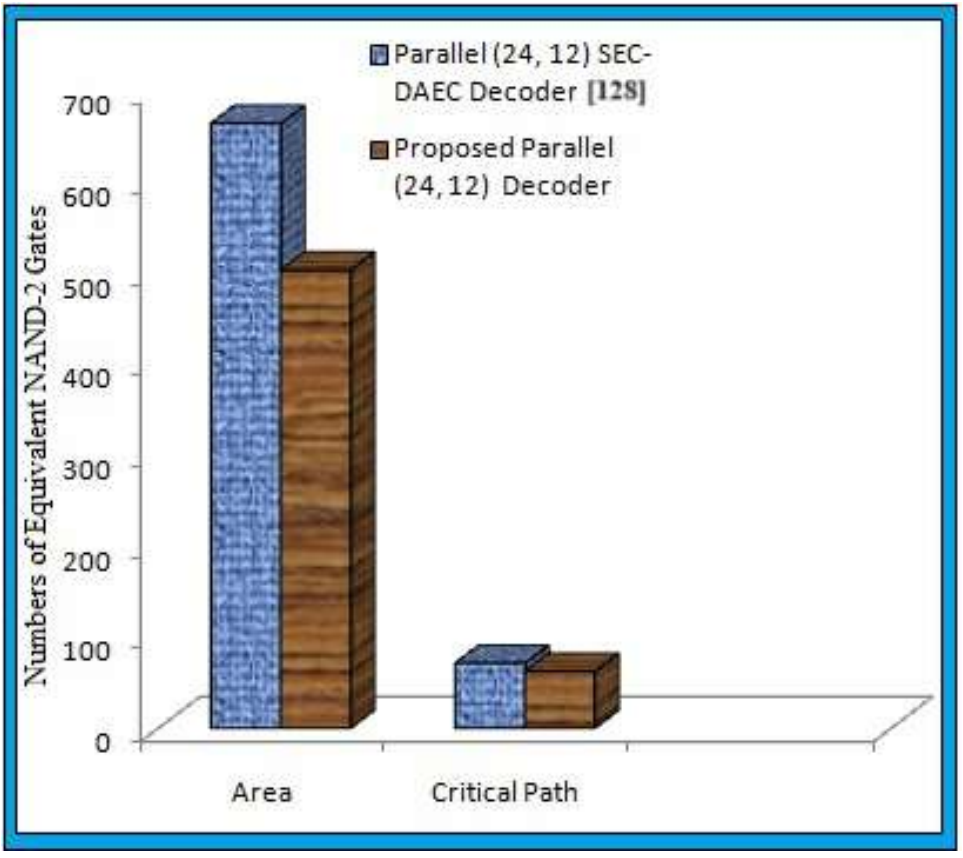

FIGURE 3.19: Graphical Comparison of Area and Critical Path of Proposed $(24, 12)$ Parallel Decoder

The proposed parallel decoder of $(24, 12)$ SEC-DAEC code requires around 24% lesser number of equivalent NAND2 gates compared to parallel decoding of $(24, 12)$ SEC-DAEC extended Golay code [128]. Also the proposed parallel decoder shows 12.68% improvement in critical path.

Graphical comparison of area and critical path of proposed $(24, 12)$ parallel decoder with the parallel decoder of $(24, 12)$ SEC-DAEC extended Golay code in [128] have been shown in Fig. 3.19. The proposed parallel decoder of $(24, 12)$ SEC-DAEC code

TABLE 3.13: FPGA-based Synthesis Results of Proposed $(24, 12)$ Parallel Decoder

Decoder	**Area (LUTs)**	**Delay (ns)**	**Improv. (%)**	
Parallel (24, 12) Golay Decoder [128]	65	8.288	Area	Delay
Proposed Parallel (24, 12) SEC-DAEC Decoder	52	7.408	20.00	10.62

shows better performance both in area and critical path.

3.5.5 Synthesis Results

Synthesis results of proposed parallel decoder have been reported both in FPGA and ASIC platforms and has been compared with the parallel decoder presented in [128].

3.5.5.1 FPGA-based Results

All the functional blocks of proposed and existing decoders are simulated and synthesized using FPGA based virtex 4 (xc4vfx12-12sf363) device. The FPGA based synthesis results have been summarized in Table 3.13. The proposed parallel decoder of $(24, 12)$ SEC-DAEC code exhibits 20% and 10.62% improvement in number of LUTs and delay respectively.

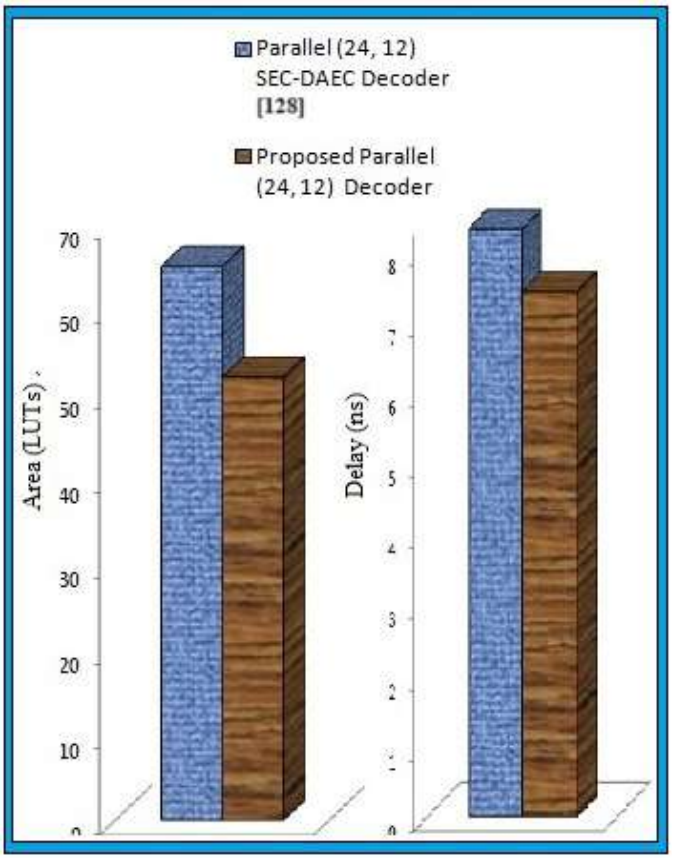

FIGURE 3.20: Graphical comparison of FPGA-based Synthesis Results of Proposed $(24, 12)$ Parallel Decoder

Figure 3.20 shows the graphical comparison of FPGA synthesis results of proposed $(24, 12)$ parallel decoder. The proposed parallel decoder require lesser number of LUTs and it is faster in its operation.

3.5.5.2 ASIC-based Results

The proposed parallel decoder has been further synthesized in ASIC platform using cadence based Genus synthesis solution (TSMC18) tool. The performance of proposed parallel decoder and the parallel decoder presented in [128] have been

TABLE 3.14: ASIC-based Synthesis Results of Proposed (24, 12) Parallel Decoder

Decoder	Area (μm^2)	Delay (ps)	Power (nW)	Power-Delay Product (fJ)
Parallel (24, 12) Golay Decoder [128]	2242.00	421.30	352.44	0.148
Proposed Parallel (24, 12) SEC-DAEC Decoder	1949.27	416.60	305.23	0.127

measured in terms of area, delay and power in ASIC platform. These ASIC based synthesis results have been summarized in Table 3.14. The proposed parallel decoder shows better performance in terms of area, delay as well as in power consumption compared to the parallel decoder for single and double errors correction [128] as shown in Table 3.14. Proposed parallel decoder exhibits maximum improvement in area (13.1%) and power (13.4%). Also the power delay (PD) product is shown in Table 3.14. An improvement of 14.19% has been achieved by the proposed parallel decoder in PD product.

3.6 Conclusions

In this chapter, three new schemes of SEC-DAEC codes have been introduced and these schemes have been designed with the aim of lowering the area, delay and power usages of SEC-DAEC decoders. In the first scheme, the improved SEC-DAEC codec provides the highest improvement of 27% in area, 43% in delay and 64% in power respectively. The methodology presented in designing the improved SEC-DAEC codes can be applied to design burst error correcting codes with different burst lengths in future. The second scheme in this chapter presents the design of lower complexity ELD block for existing SEC-DAEC decoder using K-map based simplification technique. This scheme provides 2.20–26.81% reduction in area and 0.30–28.96% reduction in delay compared to existing related works in ASIC-based synthesis results. This presented scheme of K-map based simplification can be applied to the other existing ECC schemes with more complex ELD blocks in future. The third scheme in this chapter combines the parallel decoder and area optimized decoder

for further reduction of decoding overheads of $(24, 12)$ extended Golay code based SEC-DAEC decoder. Therefore, the three schemes of SEC-DAEC codes explored in this chapter are mainly for lowering the decoder or codec design overheads. The next chapter presents three schemes of SEC-DED-DAEC codes for storage systems which are designed with the aim of reducing both the codec design overheads and mis-correction rates.

Chapter 4

Design and Implementation of SEC-DED-DAEC Codes

4.1 Introduction

The Single Error Correction- Double Error Detection- Double Adjacent Error Correction (SEC-DED-DAEC) codes are the extension of conventional SEC-DED codes in which the feature of double adjacent error correction is included. These codes are proficient of correcting single and double adjacent errors and detecting any combinations of 2-bit errors. Although both SEC-DAEC and SEC-DED-DAEC codes have exactly similar correction capability, but double error detection capability of SEC-DED-DAEC codes make these codes different from SEC-DAEC codes. The SEC-DED-DAEC codes are widely used in storage systems for protecting data bits from single and double adjacent errors and for detecting the presence of double errors in data bits. The single and double adjacent errors are induced by soft errors in the storage systems and tends of these errors are more common now a days due to the rapid advancement of semiconductor technology nodes. The main design issue of SEC-DED-DAEC codes have been their higher mis-correction rates and decoding overheads. Varieties of SEC-DED-DAEC codes which are available in the literature

have already been attempted to resolve these design issues in their SEC-DED-DAEC codes. But SEC-DED-DAEC codes with lower mis-correction rates, compact in area, faster in speed, efficient in power consumption are still required for the protection of storage systems against Multiple Cell Upsets (MCUs). In this chapter, three new schemes of SEC-DED-DAEC codes have been presented which have been designed with the objectives to reduce both the mis-correction rate and decoding overheads. The rest of this chapter is organized as follows.

A brief overview of related schemes are discussed in section 4.2. The section 4.3 describes the design of single and double- adjacent error correcting codes with lower design overheads and mis-correction rate for SRAMs. Design of compact SEC-DED-DAEC codec for memory applications has been presented in section 4.4. The design of an efficient decoding scheme of SEC-DED-DAEC codes with odd weighted data columns has been presented in section 4.5 and this chapter is concluded in section 4.6.

4.2 Brief Overview of Existing Schemes

Various SEC-DED-DAEC and SEC-DAEC codes for memories are available in the literature. Among those some of the well known schemes in [133], [129]-[130], [132], [124] and [148] are briefly discussed in this section.

- **Dutta et al. Scheme [133]**:

  The Dutta et al. code [133] is considered as the first SEC-DED-DAEC code for memory application. The main features of Dutta codes are that no additional parity bit is required and have nearly equal codec design overheads like existing SEC-DED codes. Also Dutta codes can be employed in all types of memories even for smaller size of memories where interleaving is not feasible. But the main limitation of this scheme is that it has higher mis-correction rate

against non-adjacent double and triple errors.

- **Neale et al. Scheme [129]-[130]:**

  In [129]-[130], Neale et al. explored a different construction of SEC-DED-DAEC codes. The structure of H-matrix has been partitioned into two parts: a bottom part contains repeated identity matrices placed one after another in horizontal direction and a top part contains data columns which have been arranged as sub-matrices. Also some additional conditions have been imposed in the basic H-matrix searching algorithm of SEC-DED-DAEC code. These conditions were imposed to reduce the mis-correction rate and to facilitate adjacent error detection feature. Thus considerable reduction in mis-correction rate and scalable Adjacent Error Detection (xAED) are the main features of Neale et al. SEC-DED-DAEC codes. But these codes are decoded by employing conventional decoding scheme and hence don't provide substantial improvement over area and power consumption.

- **Reviriego et al. Scheme [132]:**

  An attempt to reduce decoding overheads of SEC-DED-DAEC code has been observed in [132]. Reviriego et al. have constructed their H-matrices with the additional constrains that all the data columns in H have a fixed odd weight and it is equal to three. Whereas all the double-adjacent error correcting syndromes for data bits have a constant even weight and it is equal to four. With these modification in H-matrices, the decoding schemes for SEC-DED-DAEC codes have been optimized in two ways. In the first method of decoding, single and double-adjacent errors were located by checking only 1's in the syndrome values and employing a $notDED$ signal for single error. The $notDED$ signal was expressed as the XOR sum of all the syndrome bits. In the second method, single error was located by checking both 1's and 0's in the syndrome values

without employing $notDED$ signal and double-adjacent errors are located exactly in the same manner as in the first method. Considerable reduction in decoder's area and delay have been reported in the first and second optimization methods respectively.

- **Li et al. Scheme [124]**:
  Recently, Li et al. have introduced a low delay ECC scheme to correct single error and double-adjacent errors [124]. In this scheme, the delay has been minimized by incorporating three techniques: i) reducing the total number of 1's in the H-matrix and it's heaviest row, ii) interleaving of data and parity columns in H, and iii) optimizing syndrome comparison based decoding technique. The H-matrix has been constructed employing first two techniques together and the same proposed codes have been decoded twice based on conventional and optimized syndrome comparison based decoding techniques respectively. Lower value of delay has been observed in Li et al. scheme at the cost of increased mis-correction rate with respect to 2-random and 3-random errors.

Therefore, the major limitations of existing SEC-DAEC codes are their higher mis-corrections rates and codec design overheads. Under this situation, the proposed schemes of SEC-DED-DAEC codes aim to minimize the codec design overheads and mis-correction rates with respect to existing SEC-DED-DAEC codecs.

4.3 Design of Proposed SDECC

In this section, a new SEC-DED-DAEC code has been introduced which is proficient of correcting single error and double-adjacent errors in SRAMs. The newly proposed code has been termed as Single and Double-adjacent Error Correcting Code (SDECC). Proposed codecs have been designed by employing new parity check

matrices for different word lengths which are frequently applied in memory. The miscorrection rate of proposed SDECC codes are up to 88.24% and 80.67% lesser for 2-random and 3-random bit errors respectively with respect to related existing designs. Proposed codecs have been replicated and synthesized in ASIC environment. It is observed that area and power consumption of the proposed codecs are reduced up to 8.93% and 21.47% compared to the recently published results respectively.

4.3.1 Generation of Proposed Parity Check Matrix

The H-matrix is constructed by placing k numbers of data columns and $(n-k)$ numbers of parity columns. The k numbers of data columns for the proposed Single and Double-adjacent Error Correcting Codes (SDECCs) are constructed by employing the following rules:

1. All the data and parity columns must be non-zero.
2. All the data and parity columns must be dissimilar.
3. All the data columns must have a fixed Hamming weight and which is equal to three (w=3).
4. The XOR sum of any two adjacent data columns and XOR sum of last data column with the first parity column must be unique and has a fixed Hamming weight which is equal to four (w=4).
5. All the data columns must have '1' as its last element.

Conditions 1, 2 and 3 ensure the proficiency of SEC-DED features. DAEC feature is guaranteed by condition 4. Condition 5 is included in the proposed scheme to minimize the decoding overheads of proposed scheme of SDECC.

Once the k-data columns are determined by applying the above mentioned rules then these are combined with $(n-k)$ parity columns to form the entire H-matrix of proposed scheme. Now the parity columns for the proposed scheme are obtained in two different ways. The first scheme (scheme1) of assigning parity columns is conventional one where for an (n,k) code, $(n-k)\times(n-k)$ identity matrix is used

to form the parity. In second scheme (scheme2), parity columns are generated by employing following rules:

1. All the parity columns are non-zero, dissimilar and have a varying Hamming weight $w \leq 3$.
2. No parity column matches with any of the data columns.
3. All the parity columns have one ('1') as its last element.

These two different schemes of parity column assignment for the same data columns produces two different H-matrices for proposed SDECCs. Encoding and decoding of proposed codes with two different parity representation differ slightly from each other. As a result, there is a slight difference in the circuit complexities as well as mis-correction rate of two proposed codes. Number of parity bits required to construct the proposed H-matrices based on first scheme (scheme1) of assigning parity columns are bounded by equation (4.1).

$$\binom{n-k-1}{2} \geq k+1 \quad \text{and} \quad \binom{n-k}{4} \geq k \tag{4.1}$$

whereas proposed H-matrices based on second scheme (scheme2) of assigning parity columns need number of parity bits which are bounded by following equation. (4.2).

$$\binom{n-k-1}{2} \geq k+2 \quad \text{and} \quad \binom{n-k}{4} \geq k \tag{4.2}$$

The total number of combinations of weight two in $(n-k-1)$ bits have been employed in equations (4.1) and (4.2) respectively due to the fact that the data columns in proposed H-matrices have last bit as 1's. Also k and $k+1$ represent the required numbers of double weighted columns in the proposed H-matrices based on first scheme (scheme1) and second scheme (scheme2) of assigning parity columns respectively. The smallest number of parity bits requires to satisfy equations (4.1) and (4.2) of proposed SEC-DED-DAEC codes based on both the parity generation schemes are 8, 10 and 13 for word lengths 16, 32 and 64 respectively. Hence number of parity bits which are required to construct proposed SDECCs are slightly higher

compared to traditional SEC-DED-DAEC codes.

The H-matrix for proposed (24, 16) SDECC based on first parity scheme (scheme1) is shown in Fig. 4.1. The parity check matrix for proposed (24, 16) SDECC based on second parity scheme (scheme2) is presented in Fig. 4.2. In this H-matrix, the weight of parity columns are not fixed. The weight of first parity column is equal to 3 and weight of the last parity column is 1. The remaining parity columns have same constant weight and is equal to 2 and all the parity columns have '1' as their last bit.

The H-matrices for 32 bit and 64 bit data length of proposed SDECCs based on both the parity schemes are presented in Fig. 4.3 to Fig. 4.6 respectively.

$$\begin{bmatrix} 000100010101001010000000 \\ 101010000100100001000000 \\ 010000101010001000100000 \\ 010010000001010100010000 \\ 000001010010100000001000 \\ 001000100000010000000100 \\ 100101001000000100000010 \\ 111111111111111100000001 \end{bmatrix}$$

FIGURE 4.1: Proposed H-matrix for (24, 16) SDECC Based on Scheme1

$$\begin{bmatrix} 000100010101001010000000 \\ 101010000100100001000000 \\ 010000101010001000100000 \\ 010010000001010100010000 \\ 000001010010100000001000 \\ 001000100000010010000100 \\ 100101001000000100000010 \\ 111111111111111111111111 \end{bmatrix}$$

FIGURE 4.2: Proposed H-matrix for (24, 16) SDECC Based on Scheme2

$$\begin{bmatrix}
0000001010000000000000010101010101000000000\\
0000001000100010101010000000000000100000000\\
0100010010000000000101010100000000010000000\\
0101000000000000010010001001010100010000000\\
0001010100000101001000000000000100000100000\\
0010000001010010000100100001000000000010000\\
1010000000101001000001000010000010000001000\\
1000100000010100100000010000010000000000100\\
0000100101001000010000000100100000000000010\\
1111111111111111111111111111111110000000001
\end{bmatrix}$$

FIGURE 4.3: Proposed H-matrix for (42, 32) SDECC Based on Scheme1

$$\begin{bmatrix}
0000001010000000000000010101010101000000000\\
0000001000100010101010000000000000100000000\\
0100010010000000000101010100000000010000000\\
0101000000000000010010001001010100010000000\\
0001010100000101001000000000000100000100000\\
0010000001010010000100100001000000000010000\\
1010000000101001000001000010000010000001000\\
1000100000010100100000010000010010000000100\\
0000100101001000010000000100100000000000010\\
1111111111111111111111111111111111111111111
\end{bmatrix}$$

FIGURE 4.4: Proposed H-matrix for (42, 32) SDECC Based on Scheme2

$$\begin{bmatrix}
0001001000000000000000010101010000000000000000000000000001010101010000000000000\\
0001000000010100101010101010101000000000000000000000000000000000001000000000000\\
0000000010100101010101000000000000000000000000000000000001001010100100000000000\\
0010000010001000001000000000000010000000000000101010101000100000000010000000000\\
0000000100101010100000000000000000100010101010000000000000001000000010000000000\\
0000010100010000000100000000000010000000000101010100000001000010000001000000000\\
1000010000000001000010000000010000000000010010001001010100000000000000010000000\\
1000000001000000000001001001000100000101001000000000001001000000000000001000000\\
0100000001000000010000100100000001010010000100100001000000000000000000000010000\\
0100101000000000000000000000010000010100100000100001000010000000100000000001000\\
0010100000000000000000010000001000010100100000010000010000001000000000000100000\\
0000000000000010000000000010000101001000010000000100100000010000000000000010000\\
1111111111111111111111111111111111111111111111111111111111111111111110000000000001
\end{bmatrix}$$

FIGURE 4.5: Proposed H-matrix for (77, 64) SDECC Based on Scheme1

$$\left[\begin{array}{l}
0001001000000000000000010101010000000000000000000000000001010101010000000000\\
0001000000010100101010101010101000000000000000000000000000000000100000000000\\
0000000010100101010101000000000000000000000000000000000010010101001000000000\\
0010000010001000001000000000000010000000000000101010101000100000000100000000\\
0000000100101010100000000000000000100010101010000000000000001000000010000000\\
0000010100010000000100000000000010000000000101010100000001000010000001000000\\
1000010000000001000010000000010000000000010010001001010100000000000000010000\\
1000000001000000000001001001000100000101001000000000010010000000000000100000\\
0100000001000000010000100100000001010010000100100001000000000000000000010000\\
0100101000000000000000000000010000010100100000100001000010000000100000000001000\\
0010100000000000000000100000010000101001000000100000100000001000000000000100\\
0000000000000010000000000010000101001000010000000100100000010000100000000010\\
1111111111111111111111111111111111111111111111111111111111111111111111111111
\end{array}\right]$$

FIGURE 4.6: Proposed H-matrix for (77, 64) SDECC Based on Scheme2

4.3.2 Design of Proposed SDECC Codec

A SDECC codec is a combination of an encoder and a decoder. In memory applications, the encoder for an ECC determines the coded version of data (codewords) to be stored in memory cells, whereas the corrected version of the stored data are read out from memory by the decoder. The encoding and decoding processes for the proposed SDECCs are described in the following subsections.

4.3.2.1 Encoding of Proposed SDECC

The proposed SDECC encoder based on scheme1 determines the parity bits by employing H-matrices given in Fig. 4.1, Fig. 4.3 and Fig. 4.5. Parity bits for proposed $(24, 16)$ code with the H-matrix given in Fig. 4.1, are expressed in terms of data bits and these are as follows:

$$
\begin{aligned}
cb_1 &= d_4 \oplus d_8 \oplus d_{10} \oplus d_{12} \oplus d_{15} \\
cb_2 &= d_1 \oplus d_3 \oplus d_5 \oplus d_{10} \oplus d_{13} \\
cb_3 &= d_2 \oplus d_7 \oplus d_9 \oplus d_{11} \oplus d_{15} \\
cb_4 &= d_2 \oplus d_5 \oplus d_{12} \oplus d_{14} \oplus d_{16} \\
cb_5 &= d_6 \oplus d_8 \oplus d_{11} \oplus d_{13} \\
cb_6 &= d_3 \oplus d_7 \oplus d_{14} \\
cb_7 &= d_1 \oplus d_4 \oplus d_6 \oplus d_9 \oplus d_{16} \\
cb_8 &= d_1 \oplus d_2 \oplus d_3 \oplus d_4 \oplus d_5 \oplus d_6 \oplus d_7 \oplus d_3 \oplus d_8 \oplus d_9 \oplus d_{10} \\
&\quad \oplus d_{11} \oplus d_{12} \oplus d_{13} \oplus d_{14} \oplus d_{15} \oplus d_{16}
\end{aligned} \tag{4.3}
$$

In proposed SDECCs based on scheme2, the parity bits are computed by taking combinations of data bits and parity bits based on H-matrix. The expressions of parity bits for proposed $(24, 16)$ code with H-matrix given in Fig. 4.2 are as follows:

$$
\begin{aligned}
cb_1 &= d_4 \oplus d_8 \oplus d_{10} \oplus d_{12} \oplus d_{15} \\
cb_2 &= d_1 \oplus d_3 \oplus d_5 \oplus d_{10} \oplus d_{13} \\
bc_3 &= d_2 \oplus d_7 d_9 \oplus d_{11} \oplus d_{15} \\
cb_4 &= d_2 \oplus d_5 \oplus d_{12} \oplus d_{14} \oplus d_{16} \\
cb_5 &= d_6 \oplus d_8 \oplus d_{11} \oplus d_{13} \\
cb_6 &= d_3 \oplus d_7 \oplus d_{14} \oplus cb_1 \\
cb_7 &= d_1 \oplus d_4 \oplus d_6 \oplus d_9 \oplus d_{16} \\
cb_8 &= d_1 \oplus d_2 \oplus \cdots \oplus d_{15} \oplus d_{16} \oplus cb_1 \oplus cb_2 \oplus cb_3 \oplus cb_4 \oplus cb_5 \oplus cb_6 \oplus cb_7
\end{aligned} \tag{4.4}
$$

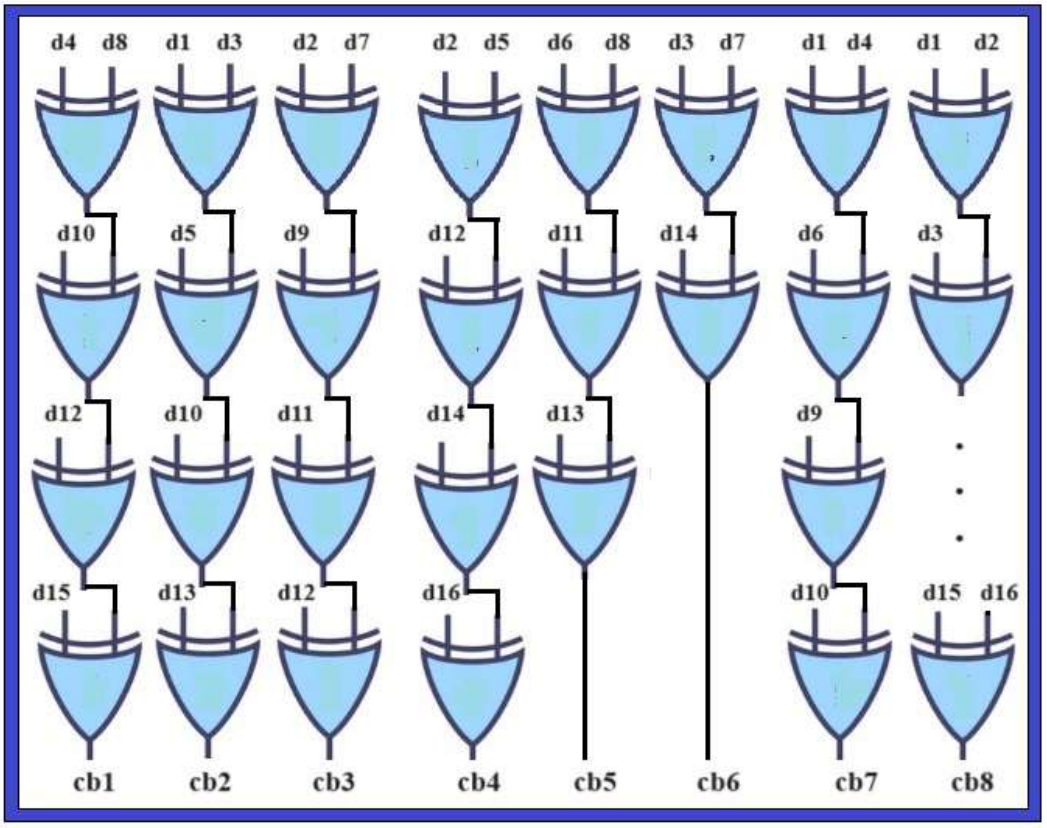

FIGURE 4.7: Encoder Circuit for (24,16) SDECC Based on Scheme1

4.3.2.2 Decoding of Proposed SDECC

The decoder circuits of proposed SDECCs are based on the first area optimized decoding scheme of Reviriego et al. [132]. Proposed decoder circuits have been implemented using the following steps:

step-1: compute the syndrome.

step-2: correct the data column in which first two positions of 1's matches with the first two 1's in the syndrome and satisfy the $notDED$.

step-3: correct the double-adjacent data columns whose module-2 sum matches with the 1's in the syndrome.

The syndrome is generated in step-1 by employing equation (2.3). Single error location determination and its correction is done by using step-2 of decoding logic. The

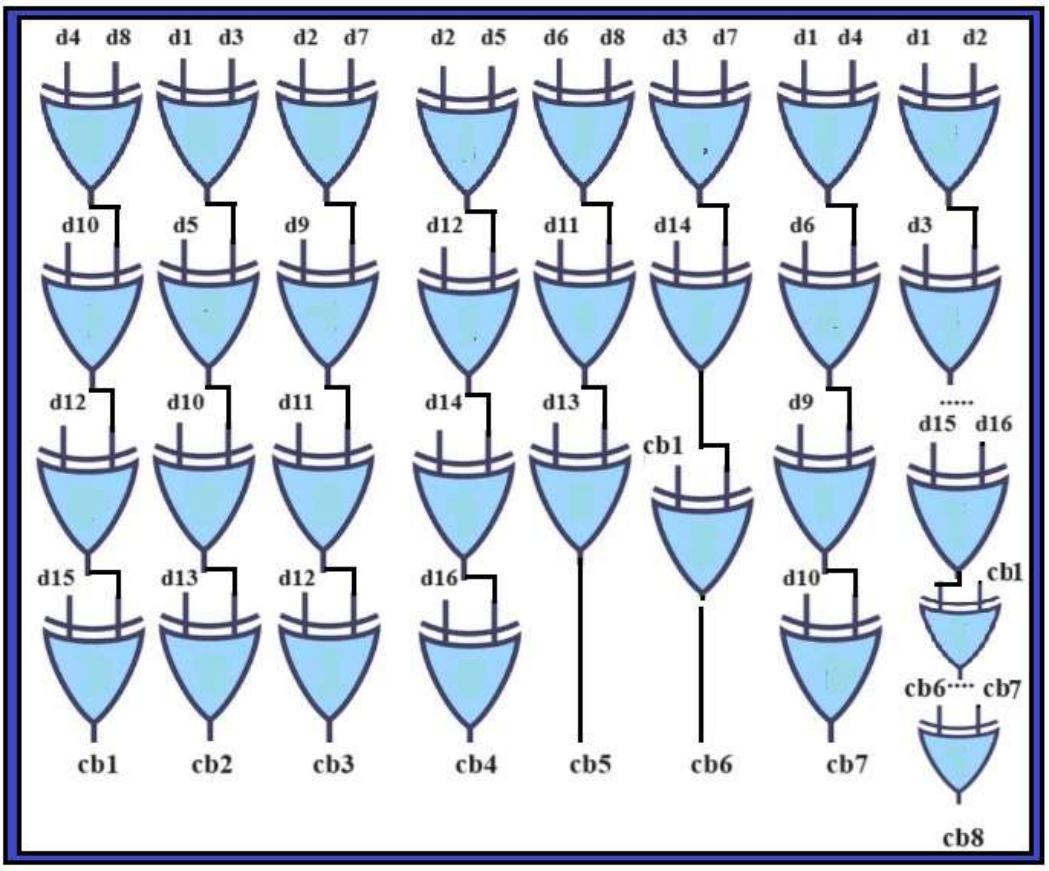

FIGURE 4.8: Encoder Circuit for (24,16) SDECC Based on Scheme2

H-matrices of proposed codes have distinct data columns with constant hamming weight equal to 3 but all the data columns have one as its last element. So even if we don't consider the last bit of each data column still the data columns are distinct. This concept is used to locate the position of single error in any of the data column i.e. if position of first two 1's in any data column matches with the syndrome value and $notDED$ is satisfied then single error is located on that position. To locate the single error in any of the k data position with the proposed SDECCs, only k number of 3-input AND gates are required. whereas, to perform the same functionality k number of 4-input AND gates are required for the codes presented in [132]. Double-adjacent errors are located and corrected in step-3 of decoding logic.

The syndrome expressions for proposed $(24, 16)$ SDECC based on scheme1 are listed

in equation (4.5).

$$\begin{aligned}
S_1 &= r_4 \oplus r_8 \oplus r_{10} \oplus r_{12} \oplus r_{15} \oplus r_{17} \\
S_2 &= r_1 \oplus r_3 \oplus r_{10} \oplus r_{13} \oplus r_{18} \\
S_3 &= r_2 \oplus r_7 \oplus r_9 \oplus r_{11} \oplus d_{15} \oplus r_{19} \\
S_4 &= r_2 \oplus r_5 \oplus r_{12} \oplus r_{14} \oplus d_{16} \oplus r_{20} \\
S_5 &= r_6 \oplus r_8 \oplus r_{11} \oplus r_{13} \oplus r_{21} \\
S_6 &= r_3 \oplus r_7 \oplus r_{14} \oplus r_{22} \\
S_7 &= r_1 \oplus r_4 \oplus r_6 \oplus r_9 \oplus r_{16} \oplus r_{23} \\
S_8 &= r_1 \oplus r_2 \oplus \cdots \oplus r_{15} \oplus r_{16} \oplus r_{24}
\end{aligned} \tag{4.5}$$

On the other hand, the $notDED$ signal which is required for the determination of single error location, can be computed conventionally for constant odd column weight codes by taking XOR of all the syndrome bits. The expression for $notDED$ signal for proposed $(24, 16)$ SDECC based on scheme1 as shown in Fig. 4.1 can be computed conventionally as:

$$notDED = S_1 \oplus S_2 \oplus S_3 \oplus S_4 \oplus S_5 \oplus S_6 \oplus S_7 \oplus S_8 \tag{4.6}$$

Decoder circuit for proposed $(24, 16)$ SDECC based on scheme1 has been indicated in Fig. 4.9.

The syndrome expressions based on H-matrix provided in Fig. 4.2, are as follows:

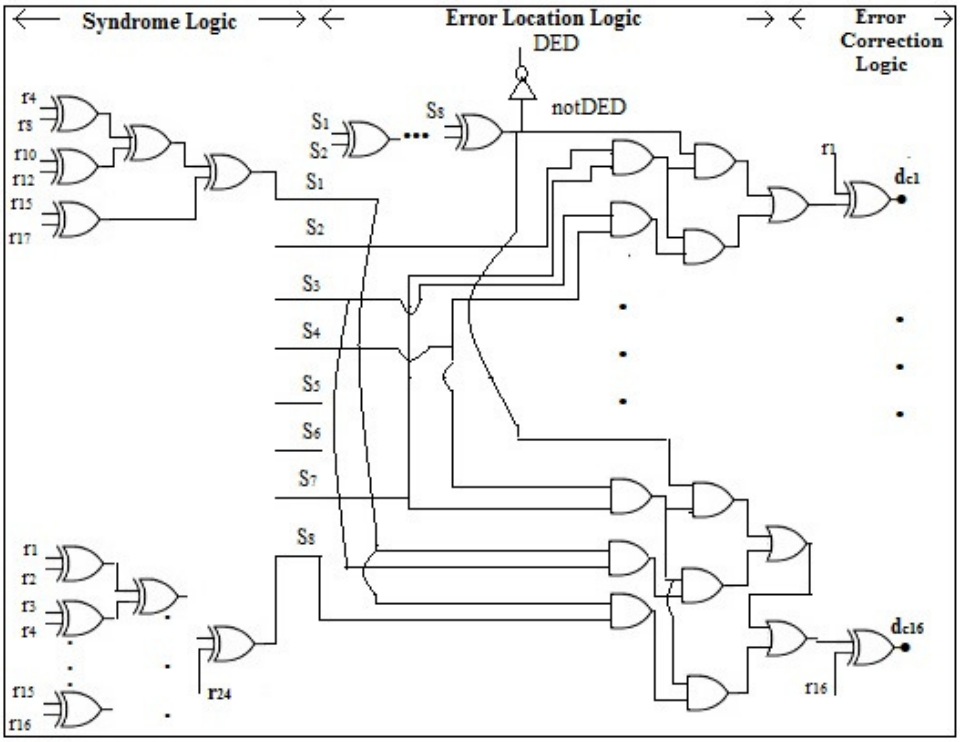

FIGURE 4.9: Decoder Circuit of Proposed (24, 16) SDECC Based on Scheme1

$$
\begin{aligned}
S_1 &= r_4 \oplus r_8 \oplus r_{10} \oplus r_{12} \oplus r_{15} \oplus r_{17} \\
S_2 &= r_1 \oplus r_3 \oplus r_{10} \oplus r_{13} \oplus r_{18} \\
S_3 &= r_2 \oplus r_7 \oplus r_9 \oplus r_{11} \oplus d_{15} \oplus r_{19} \\
S_4 &= r_2 \oplus r_5 \oplus r_{12} \oplus r_{14} \oplus d_{16} \oplus r_{20} \\
S_5 &= r_6 \oplus r_8 \oplus r_{11} \oplus r_{13} \oplus r_{21} \\
S_6 &= r_3 \oplus r_7 \oplus d_{13} \oplus r_{17} \oplus r_{22} \\
S_7 &= r_1 \oplus r_4 \oplus r_6 \oplus r_9 \oplus r_{16} \oplus r_{23} \\
S_8 &= r_1 \oplus r_2 \oplus \cdots \oplus r_{16} \oplus r_{17} \oplus \cdots \oplus r_{24}
\end{aligned} \tag{4.7}
$$

From equations (4.6) and (4.7), it is observed that the syndrome expressions for codes as presented in Fig. 4.1 and Fig. 4.2 are almost similar except that there is a slightly higher hardware requirements for computation of some syndrome bits based

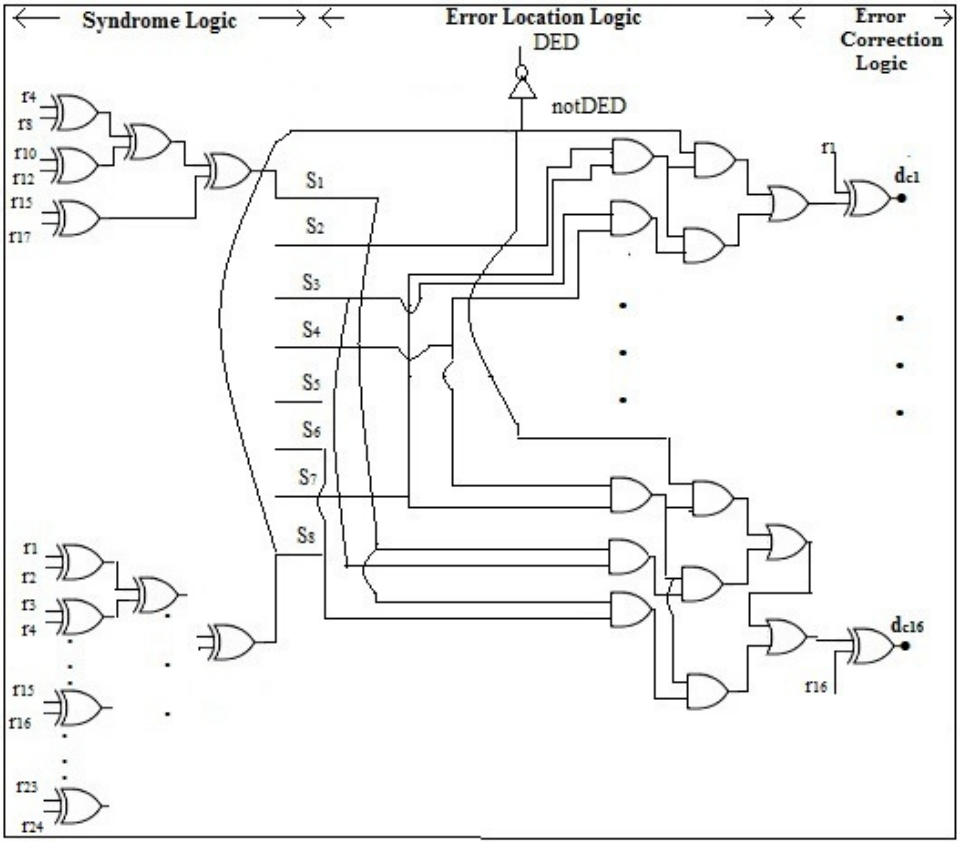

FIGURE 4.10: Decoder Circuit of Proposed (24, 16) SDECC Based on Scheme2

on scheme2. Apart from that there exist another alternative option for $notDED$ signals for the proposed SDECCs based on scheme2. This assignment of $notDED$ signal is based on the fact that there are no non-zero element in the last row of proposed H-matrices based on scheme2. Thus, when a single error occurs the last bit of the syndrome will be 1 and if double errors occur then the last syndrome bit will be 0 i.e. single or double errors can be identified by observing the last syndrome bit in this scheme. The expression for $notDED$ signal for the code presented in Fig. 4.10 can be expressed as:

$$notDED = S_8 \tag{4.8}$$

This simple $notDED$ expression of proposed SDECCs based on scheme2 provides

significant reduction in decoder complexity compared to conventional $notDED$ expression of the proposed SDECCs based on scheme1. Even though the encoder circuits and syndrome circuits for the proposed scheme2 have higher delay compared to the proposed scheme1, but the second scheme exhibits better performance when encoder and decoder circuits are combined to form a codec.

Decoder circuit for proposed $(24, 16)$ SDECC based on scheme2 has been indicated in Fig. 4.10. In both the decoders, as indicated in Fig. 4.9 and Fig. 4.10, syndrome is computed by syndrome logic part of the decoders which is implemented using XOR gates. Error location logic of the decoders determines the position of single or double-adjacent errors based on syndrome values and this part is implemented using AND and OR gates. The final stage in both decoders is the error correction logic which is implemented using XOR gates and this stage flips the bits in error as identified by the error location logic. Two decoder circuits are almost similar except few changes in syndrome logic and one change in double-adjacent error location logic. The most important difference between two decoder circuits are the implementation of $notDED$ logic.

4.3.3 ASIC-based Synthesis Results

The proposed SDECCs based on two different parity generations schemes are simulated and synthesized using Application Specific Integrated Circuit (ASIC) environment. The area, delay and power of existing and proposed codes have been measured in ASIC. Also the chances of misinterpreting nonadjacent random errors as adjacent errors known as mis-correction rate has been computed using MATLAB simulation and results are compared with the other existing codes. The proposed SDECC codecs have been synthesized in ASIC using 180 nm technology nodes. An operating voltage of 1.62V has been chosen during the ASIC synthesis. These synthesis results are mentioned in Table 4.1. In this table, the performances of two proposed codecs have been evaluated with respect to five existing related schemes

TABLE 4.1: ASIC Synthesis Results for Proposed SDECC Codecs

k	Codes	(n, k)	Area (μm^2)	Delay (ps)	Power (mW)
16	**Dutta [133]**	(22, 16)	2967	346	0.49
	Reviriego [132]	(23, 16)	2831	349	0.56
	Neale [129], [130]	(23, 16)	2764	348	0.51
	Li [124]	(22, 16)	2781	340	0.55
	Tripathi [148]	(24, 16)	2848	412	0.55
	Proposed scheme1	(24, 16)	2807	356	0.54
	Proposed scheme2	(24, 16)	2741	399	0.50
32	**Dutta [133]**	(39, 32)	5815	524	1.07
	Reviriego [132]	(39, 32)	5625	443	1.63
	Neale [129], [130]	(39, 32)	5385	478	1.36
	Li [124]	(39, 32)	5496	412	1.37
	Tripathi [148]	(41, 32)	5567	573	1.42
	Proposed scheme1	(42, 32)	5442	415	1.38
	Proposed scheme2	(42, 32)	5296	570	1.28
64	**Dutta [133]**	(72, 64)	10967	588	2.74
	Reviriego [132]	(73, 64)	10984	856	4.20
	Neale [129], [130]	(72, 64)	11039	555	3.66
	Li [124]	(72, 64)	10884	553	3.53
	Tripathi [148]	(75, 64)	10893	688	3.65
	Proposed scheme1	(77, 64)	10518	565	3.54
	Proposed scheme2	(77, 64)	10322	726	3.46

in [133], [132], [129]- [130], [124] and [148]. The proposed codecs occupy lower area with respect to the existing codecs as listed in Table 4.1. Also proposed codecs based on scheme2 are more area efficient than that of scheme1. The delay performance of proposed codecs for 16 bits word length are not significant compared to most of the existing codecs as listed in Table 4.1. But the proposed 16-bit codec based on scheme2 provides a slight delay improvement compared to 16-bit codec [148]. Whereas the proposed codecs based on scheme1 exhibits better delay performance for 32 bits word length compared to the existing codecs presented in [133], [132] and [148]. Also proposed 64-bit codec based on scheme1 has better delay performance compared to the existing Dutta et al., Reviriego et al., and Tripathi et al. codecs of same word length. The proposed codec based on scheme2 has lesser delay for 64 bits word length compared to the scheme in [132]. Proposed codecs based on both the parity schemes consume lesser power compared to the codecs presented in [132], [124] and [148] for all the word lengths. Also the proposed codecs based on scheme2 have lower power consumption with respect to Neale et al. codecs for all the word lengths. From Table 4.1, it is found that the proposed codecs based on scheme2 consumes lesser power compared to the proposed scheme1. The power consumption

of the proposed codecs based on both the parity schemes are higher compared to the codes presented in [133]. This is due to the fact that the H-matrix construction rules as well as decoding mechanism for the proposed codes and the codes in [132] are quite similar but different from the codes in [133]. The simplified error location detection logic of proposed codecs provide significant improvement over the codecs presented in [148]. But the proposed codecs have higher total numbers of 1's in their H-matrices compared to the codecs in [148]. Therefore, the overall improvements of the proposed codecs in error location detection logic are relatively nullified due to their requirements of higher numbers of logic gates in the encoding and syndrome circuits compared to [148]. Due to this fact, the improvements of the proposed codecs are not much against the codecs in [148].

Area improvement for the proposed codecs based on both the parity schemes have been represented graphically in Fig. 4.11. From Fig. 4.11, it is observed that area improvement for proposed codes based on scheme2 is the highest for 32 bits word length against the code proposed in [133]. Whereas area improvement is nominal for the proposed code based on scheme1 for 16 bits word length against the code in [132]. Delay improvement for proposed codecs have been plotted in Fig. 4.12. The utmost progress in delay is accomplished by the proposed code based on scheme1 for 64 bits word length against [132]. The power consumption for the proposed code have been graphically represented in Fig. 4.13. As shown in Fig. 4.13, the power savings is utmost for proposed code based on scheme2 for 32 bits word length against [132]. However, the utmost power savings is exhibited by proposed code based on scheme1 for 16 bits word length compared to the results reported in [132].

4.3.4 Decoder Performance in Terms of Mis-correction Rate

The mis-correction rate of proposed and existing SEC-DED-DAEC codes have been computed based on H-matrix of the respective codes by employing equation (2.7). The mis-correction rate for proposed codes have been computed using MATLAB

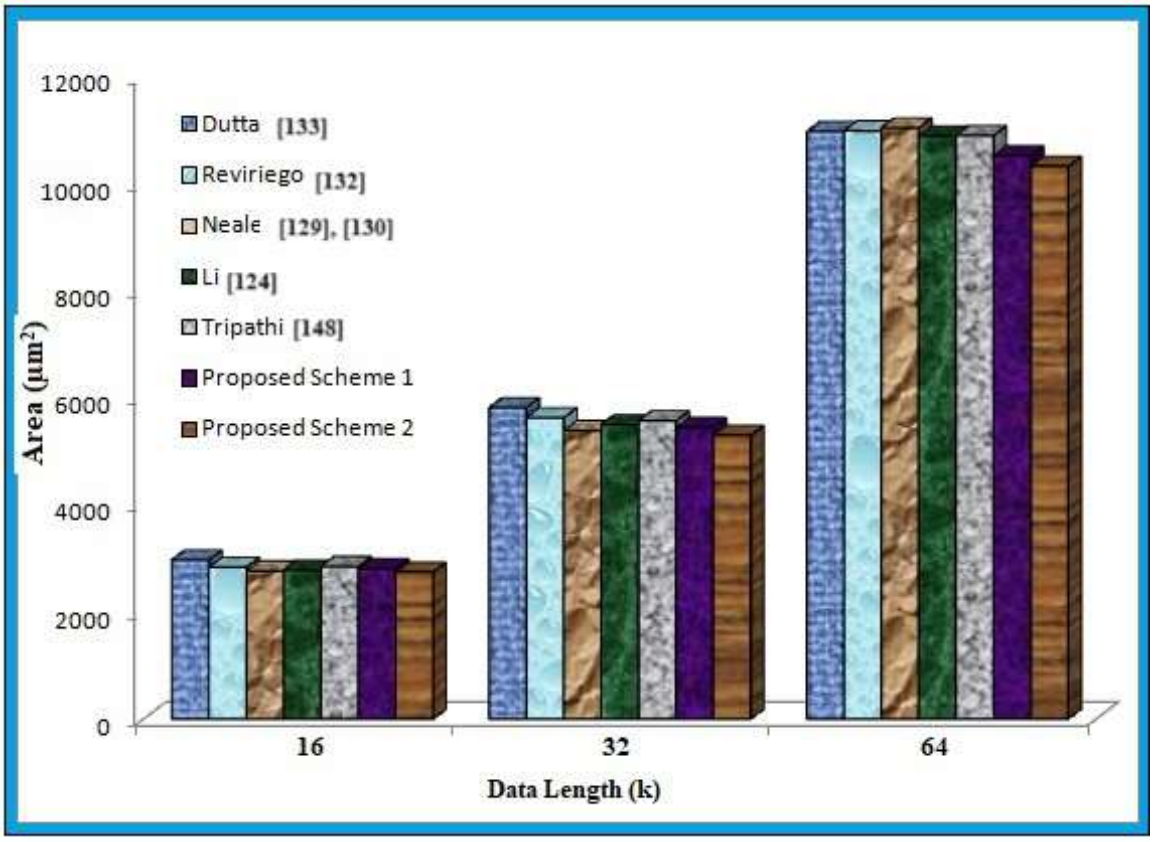

FIGURE 4.11: Graphical Representation of Improvement in Area of Proposed Codes w.r.t. Existing Codes

and compared with five other existing codes in [133], [132], [129]- [130], [124] and [148] which are reported in Table 4.2. The mis-correction rates of the proposed codes based on both the parity schemes are superior compared to all of existing codes as listed in Table 4.2. Therefore, the proposed codes are suitable to correct double-adjacent errors with a significantly lower chances of being misinterpreted as nonadjacent double and triple bit errors. The graphical representation of mis-correction rate of the proposed codes have been shown in Fig. 4.14 against 2-random and 3 random errors. This figure shows that the mis-correction rate for proposed codes based on both the parity schemes are nearly equal and decreases with the increase in word length. The highest value of mis-correction rate for the proposed codes is 17% and the lowest value is 8.46% for 2-random errors and the respective values are 23.72% and 11.98% for 3-random errors.

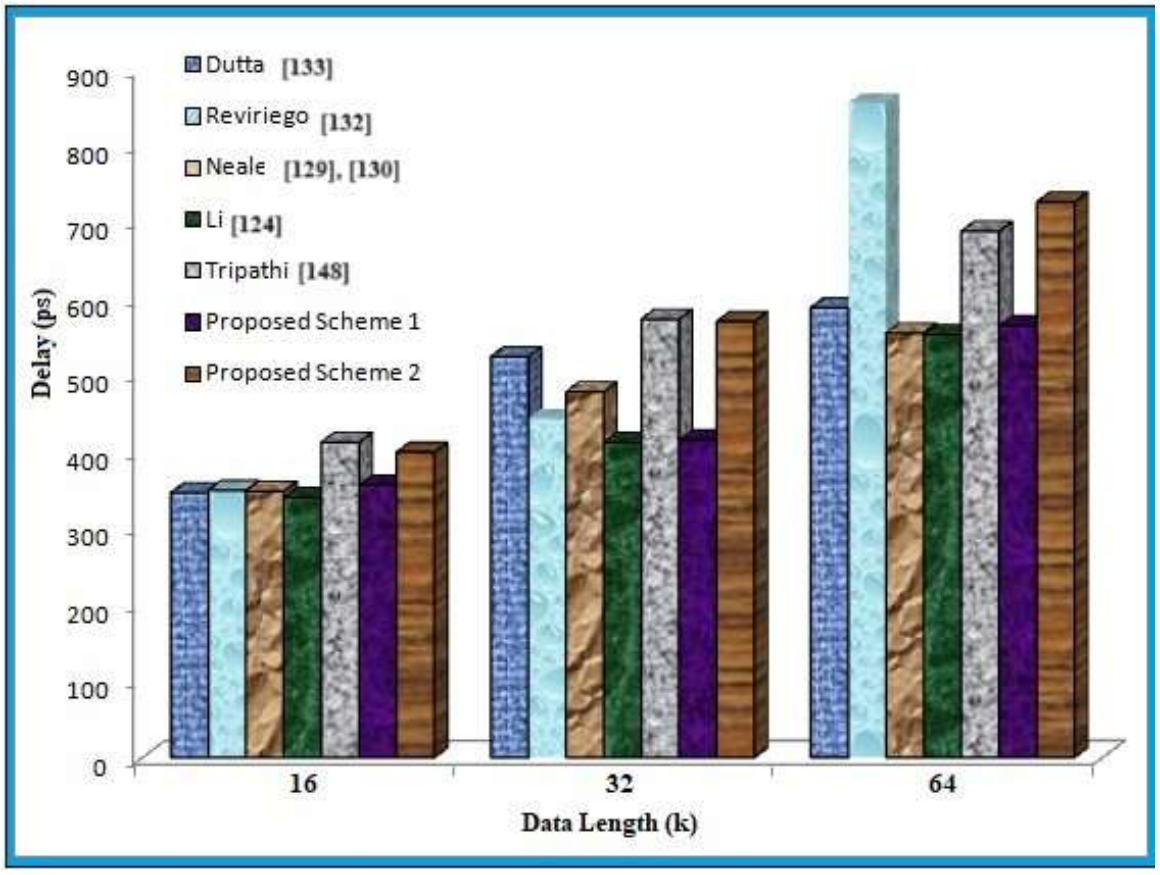

FIGURE 4.12: Graphical Representation of Improvement in Delay of Proposed Codes w.r.t. Existing Codes

4.3.5 Evaluation of R-metric for ECCs

A number of metrics have already been explored in [38] to evaluate the performance of ECC schemes. The most widespread metric for an ECC is termed as *cost* and which is the product of area delay and power (ADP) as described in equation (3.8) Thus, *cost*-metric of an ECC is computed based on only encoder and decoder or codec design constraints like area, delay and power consumption. But this metric does not reflect the effect of other two essential parameters for designing ECCs for SRAMs applications which are mis-correction rate and redundancy. We have modified the *Cost*-metric with the involvement of two more important factors: the average mis-correction rate ($P_{avg.}$) and fractional redundancy (R). The reason behind the inclusion of average mis-correction rate is that lower value of this parameter

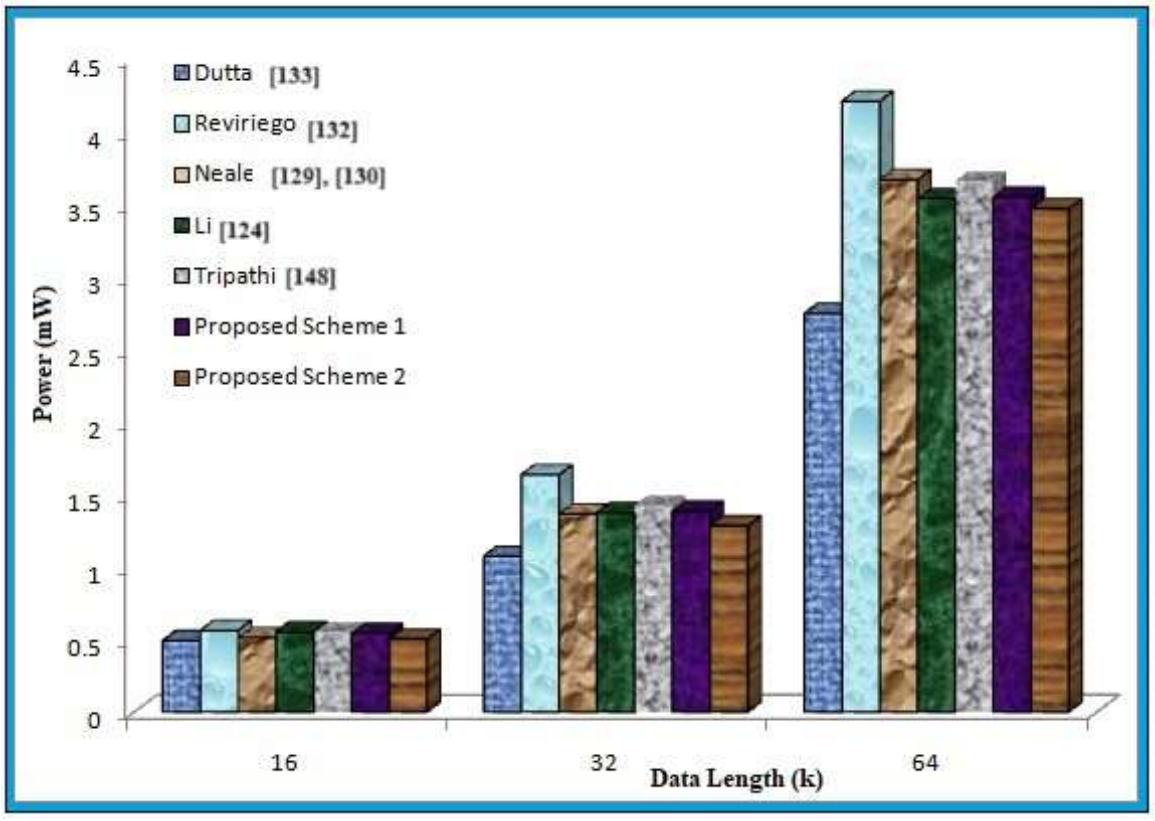

FIGURE 4.13: Graphical Representation of Improvement in Power of Proposed Codes w.r.t. Existing Codes

reduces the occurrence of non-adjacent double and triple errors as adjacent errors. Also lower value of fractional redundancy is desired to minimize the parity area overhead. Therefore, with the inclusions of average mis-correction rate and fractional redundancy in the *cost*-metric, we present a new metric for ECCs with non-zero mis-correction probabilities, denoted by R-metric and which is expressed in equation (4.9).

$$R - metric = \quad Area \text{ x } Delay \text{ x } Power \text{ x } P_{avg.} \text{ x } R \tag{4.9}$$

where $P_{avg.}$=Average mis-correction rate

$$= \frac{P_{\text{2-Random Errors}} + P_{\text{3-Random Errors}}}{2}$$

TABLE 4.2: Comparison of Mis-correction Rates for Proposed SDECCs

k	Codes	(n, k)	Mis-correction Rate (P) in %	
			2-Random Errors	3-Random Errors
16	Dutta [133]	(22, 16)	64.29	65.20
	Reviriego [132]	(23, 16)	47.19	50.37
	Neale [129], [130]	(23, 16)	28.60	38.60
	Li [124]	(22, 16)	77.14	70.07
	Tripathy [148]	(24, 16)	18.21	25.30
	Proposed scheme1	(24, 16)	17.00	23.72
	Proposed scheme2	(24, 16)	17.00	22.33
32	Dutta [133]	(39, 32)	57.33	59.66
	Reviriego [132]	(39, 32)	57.33	59.92
	Neale [129], [130]	(39, 32)	49.80	60.80
	Li [124]	(39, 32)	71.27	63.72
	Tripathy [148]	(42, 32)	20.90	27.31
	Proposed scheme1	(42, 32)	13.17	17.84
	Proposed scheme2	(42, 32)	13.29	17.84
64	Dutta [133]	(72, 64)	54.57	55.59
	Reviriego [132]	(73, 64)	36.23	37.17
	Neale [129], [130]	(72, 64)	47.80	57.00
	Li [124]	(72, 64)	71.91	61.97
	Tripathy [148]	(75, 64)	11.37	18.02
	Proposed scheme1	(77, 64)	8.60	12.03
	Proposed scheme2	(77, 64)	8.46	11.98

and R is the fraction redundancy as described in equation (2.6).

The R-metric expressed in equation (4.9) is consists of five design constraints of an ECC and it can be termed as the overall cost of an ECC. Expected values of all these five constraints should be as lower as probable. But this is hard to realize in practical scenarios. Therefore, an ECC scheme which is designed with a smaller value of R-metric is the ideal alternative for protecting SRAMs against MCUs. The R-metrics for proposed SDECC codes based on both the parity schemes and existing codes in [133], [132], [129]- [130], [124] and [148] have been computed based on Tables 4.1, 4.2 and plotted in Fig. 4.15. Proposed SDECC based on scheme2 provide lowest

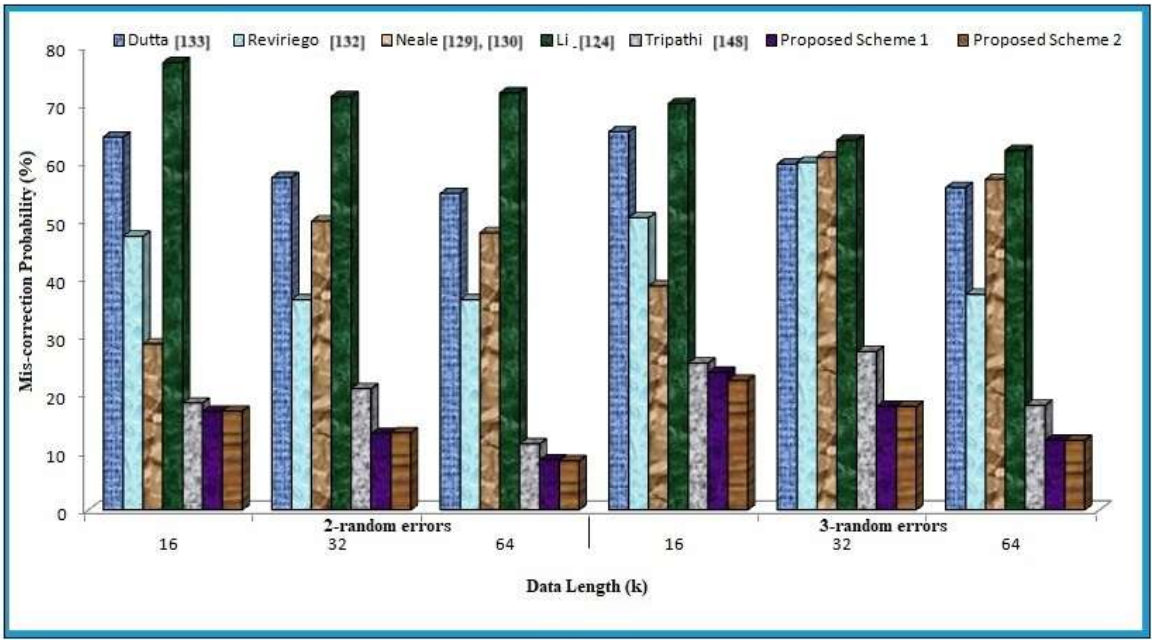

FIGURE 4.14: Graphical Representation of Improvement in Mis-correction Rate for Proposed Codes w.r.t. Existing Codes

value of R-metric for 16-bit data length compared to all other codes as shown in Fig. 4.15. But R-metric of proposed SDECC based on scheme1 for 32-bit and 64-bit data lengths are lowest compared to existing codes. As a whole, R-metric of proposed SDECCs based on both the parity schemes are superior compared to existing related codes.

4.4 Design of New Compact SEC-DED-DAEC Code for Memory

In this section, design of a new SEC-DED-DAEC code has been proposed for memory applications. The proposed codecs have been designed and synthesized in FPGA platform for some common word lengths which are frequently used in memory applications. The performance of proposed codecs have been compared with other related works. The proposed codecs require lesser area compared to existing codecs.

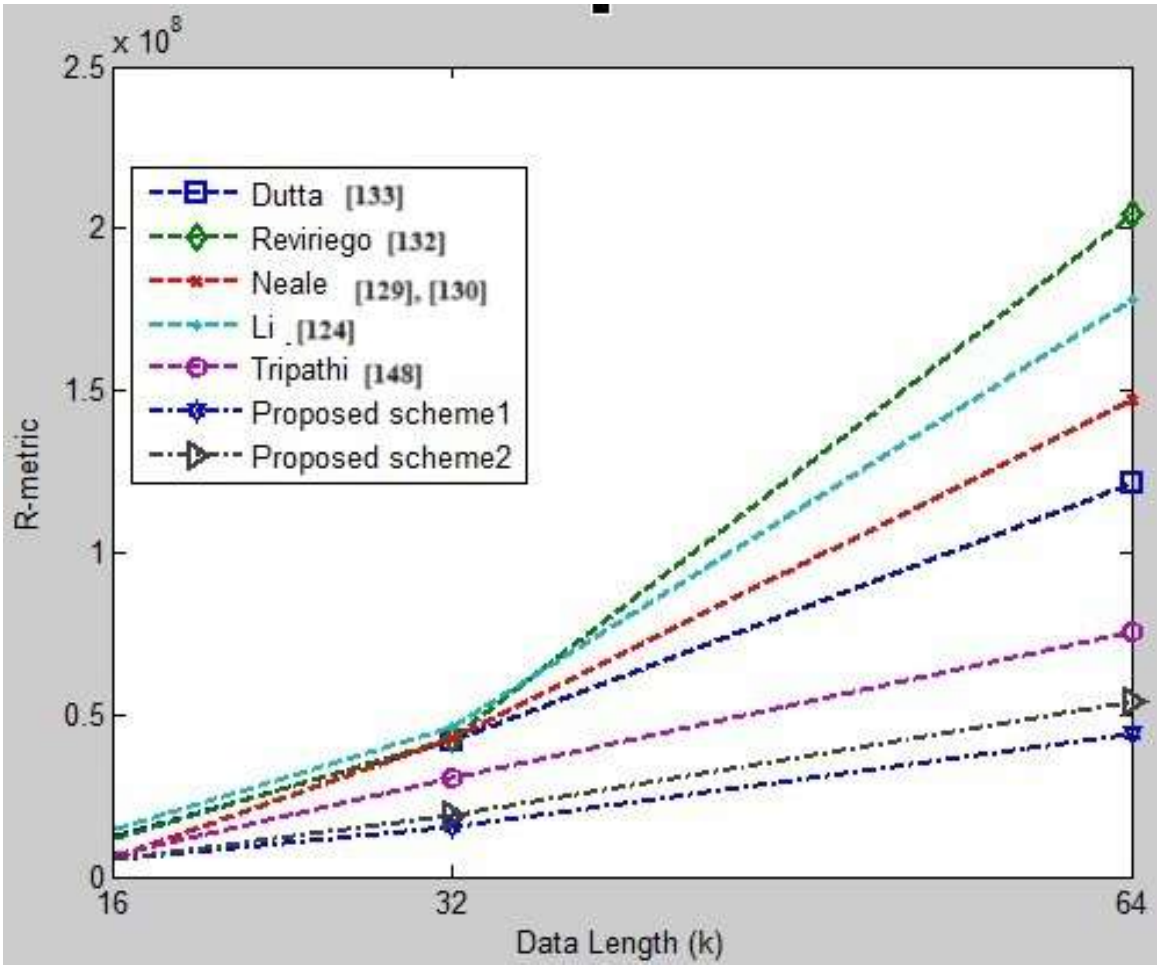

FIGURE 4.15: *R*-metric vs. Data Lengths of Proposed and Existing Codes

4.4.1 *H*-matrix Construction of Proposed SEC-DED-DAEC Codes

The proposed SEC-DED-DAEC *H*-matrices are created by employing the following conditions:

1. All the columns in *H*-matrix must be nonzero and distinct.

2. All the data columns in *H*-matrix must have constant Hamming weight equals to 3.

3. The XOR sum of any two adjacent columns in *H*-matrix should be unique and has a constant Hamming weight equal to 2.

Condition 1 provides the single error correction (SEC) feature. Double error detection (DED) feature of the proposed codes are confirmed by condition 2 as single error produces a odd weight syndrome whereas weight of the syndrome is even for double errors. The double adjacent error correction (DAEC) feature of the proposed codes are maintained by condition 3.

The H-matrices of proposed SEC-DED-DAEC codes have been constructed for some common word lengths used in memory i.e. for k=16, 32 and 64 bits. For the proposed (n, k) SEC-DED-DAEC codes the length of codewords (n) and number of parity bits $(n-k)$ are determined by employing equation (4.10) and these values are listed in Table 4.3.

$$\binom{n-k}{3} \geq k \quad \text{and} \quad \binom{n-k}{2} \geq n-1 \tag{4.10}$$

The H-matrices of proposed SEC-DED-DAEC are shown in Fig.4.16 to Fig.4.18

TABLE 4.3: Number of Parity Bits and Codeword Lengths of Proposed Codes

Number of Data Bits (k)	**Number of Parity Bits**$(n-k)$	(n, k)
16	8	$(24, 16)$
32	10	$(42, 32)$
64	13	$(77, 64)$

$$H = \begin{bmatrix} 000011100000011110000000 \\ 011110000001110001000000 \\ 110000001011100000100000 \\ 000000011110000100010000 \\ 000001000111000000001000 \\ 000111110000001100000100 \\ 101100000000111000000010 \\ 111000111100000000000001 \end{bmatrix}$$

FIGURE 4.16: H-matrix of Proposed $(24, 16)$ SEC-DED-DAEC Code

for 16, 32 and 64-bit word lengths respectively.

$$H = \begin{bmatrix}
0111000000011000000000000011101110000000000 \\
1111110000000000111010001000000001000000000 \\
1100000010000000000000111101000000010000000 \\
0001111000000001000000011111110000001000000 \\
1000001100001111110000000000000010000100000 \\
0000000000111111100001000000000000000010000 \\
0000100001110000000111100000000000000001000 \\
0010000000000100001111110000000000000000100 \\
0000000111100000011100000000111000000000010 \\
0000011111000010000000000110011100000000001
\end{bmatrix}$$

FIGURE 4.17: H-matrix of Proposed $(42, 32)$ SEC-DED-DAEC Code

$$\begin{bmatrix}
0000111110000011111100000001000000000000000000000000000000010101110000000000 \\
0000000000000001000000000011111000100000000000000000000000100000010100000000000 \\
0001110000011100000000000000000010000000000000011111101000000000001000000000000 \\
1100000000000000000000000000001000000000000111110000010000000001110001000000000 \\
0000000011111000000000010000000000000000000000010111100000111100000100000000 \\
0000000000110000000110001111111110000000000101000000000000000000000000010000000 \\
1000000000000000000011111100001100000000010000001100000000000000000000001000000 \\
0000011111100000001000000000100000011111000001110000000000000000000000100000 \\
0011001000000000000001111000000000000100001111111000000000000000000000010000 \\
1110000001000000011111100000000001110001100000000001111000000000000000001000 \\
0000000000000111110000000000011011100011111000000000000100000000000000000100 \\
0000000100001110000110000000000000000111100000000000011111110000000000000010 \\
0111100000000000100000001000000000001100000000000000000011111110000000000001
\end{bmatrix}$$

FIGURE 4.18: H-matrix of Proposed $(77, 64)$ SEC-DED-DAEC Code

4.4.2 Design of Proposed SEC-DED-DAEC Codecs

Encoding and decoding of any ECC can be performed by codec which is a combination of an encoder and a decoder. The decoder detects or corrects error in stored codewords caused by soft errors to retrieve the original data.

The expressions for parity bits of proposed $(24, 16)$ SEC-DED-DAEC code based on its H-matrix shown in Fig. 4.16 are presented as follows

$$
\begin{aligned}
c_1 &= d_5 \oplus d_6 \oplus d_7 \oplus d_{14} \oplus d_{15} \oplus d_{16} \\
c_2 &= d_2 \oplus d_3 \oplus d_4 \oplus d_5 \oplus d_{12} \oplus d_{13} \oplus d_{14} \\
c_3 &= d_1 \oplus d_2 \oplus d_9 \oplus d_{11} \oplus d_{12} \oplus d_{13} \\
c_4 &= d_8 \oplus d_9 \oplus d_{10} \oplus d_{11} \oplus d_{16} \\
c_5 &= d_6 \oplus d_{10} \oplus d_{11} \oplus d_{12} \\
c_6 &= d_4 \oplus d_5 \oplus d_6 \oplus d_7 \oplus d_8 \oplus d_{15} \oplus d_{16} \\
c_7 &= d_1 \oplus d_3 \oplus d_4 \oplus d_{13} \oplus d_{14} \oplus d_{15} \\
c_8 &= d_1 \oplus d_2 \oplus d_3 \oplus d_7 \oplus d_8 \oplus d_9 \oplus d_{10}
\end{aligned} \tag{4.11}
$$

The encoder circuit of proposed $(24, 16)$ SEC-DED-DAEC code is revealed in Fig. 4.19 which computes the parity bits employing equation (4.11). In Fig. 4.19, the logic design of first (c_1) and last parity (c_8) bits have been shown. The encoder circuit consists of only 2-input XOR (XOR-2) gates.

Decoder circuit in proposed SEC-DED-DAEC codec retrieves back the original data stored in memory by correcting the errors caused by soft errors. Syndrome (S) for any ECC is computed based on the H-matrix of the code and r (stored codeword in memory) by employing equation (2.3).

The syndrome expressions for proposed $(24, 16)$ SEC-DED-DAEC code with H-matrix as indicated in Fig. 4.16 are as follows.

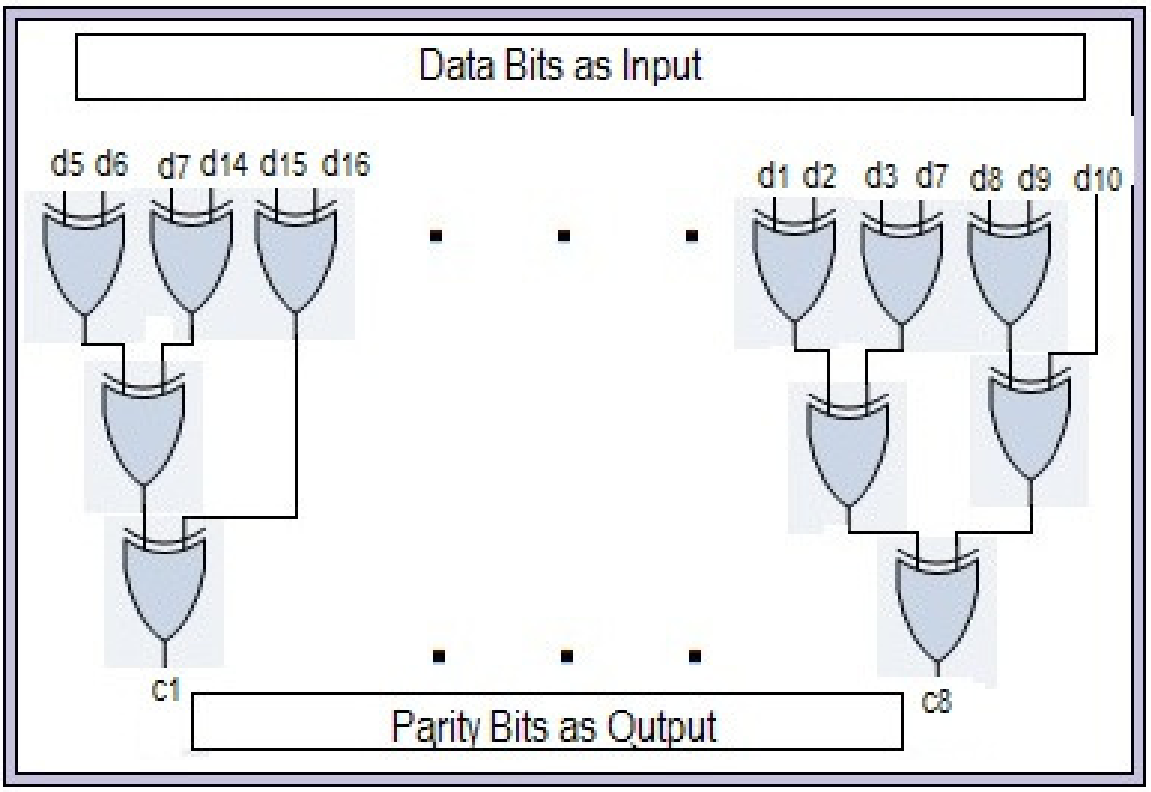

FIGURE 4.19: Encoder Circuit of Proposed $(24, 16)$ SEC-DED-DAEC Code

$$
\begin{aligned}
S_1 &= r_5 \oplus r_6 \oplus r_7 \oplus r_{14} \oplus r_{15} \oplus r_{16} \oplus r_{17} \\
S_2 &= r_2 \oplus r_3 \oplus r_4 \oplus r_5 \oplus r_{12} \oplus r_{13} \oplus r_{14} \oplus r_{18} \\
S_3 &= r_1 \oplus r_2 \oplus r_9 \oplus r_{11} \oplus r_{12} \oplus r_{13} \oplus r_{19} \\
S_4 &= r_8 \oplus r_9 \oplus r_{10} \oplus r_{11} \oplus r_{16} \oplus r_{20} \\
S_5 &= r_6 \oplus r_{10} \oplus r_{11} \oplus r_{12} \oplus r_{21} \\
S_6 &= r_4 \oplus r_5 \oplus r_6 \oplus r_7 \oplus r_8 \oplus r_{15} \oplus r_{16} \oplus r_{22} \\
S_7 &= r_1 \oplus r_3 \oplus r_4 \oplus r_{13} \oplus r_{14} \oplus r_{15} \oplus r_{23} \\
S_8 &= r_1 \oplus r_2 \oplus r_3 \oplus r_7 \oplus r_8 \oplus r_9 \oplus r_{10} \oplus r_{24}
\end{aligned} \tag{4.12}
$$

If all the syndrome expressions are zero then there is no error in r to be detected. Otherwise if syndrome corresponds to a nonzero value then error is detected. So

whether error has occurred or not in r can be determined by checking the syndrome values. Locations of detected errors are also determined based on syndrome of r i.e. if syndrome values match any of data column or XOR sum of two adjacent data columns in H-matrix then single or double adjacent errors are located in respective position of data-bit in r. This is the conventional way of error location detection method for SEC-DED-DAEC codes with unequal column weight in H-matrices.

In the H-matrices of proposed SEC-DED-DAEC codes, the data columns have a constant odd weight equal to 3 and XOR sum of two adjacent columns have a constant even weight equal to 2. So error location detection part of the proposed SEC-DED-DAEC decoders can be simplified as follows.

i) If weight of the syndrome is odd and 1's in the syndrome matches with the 1's in any data column then single error is located on that particular data column.

ii) Else if weight of the syndrome is even and 1's in syndrome matches with the 1's in XOR sum of any two adjacent data columns then double adjacent errors are located on respective data columns.

This simple error location detection logic for proposed SEC-DED-DAEC decoders requires lesser number of AND gates and don't require any NOT gate in their implementation compared to conventional error location detection logic.

The located errors are corrected by error correction logic part of the decoder which employ XOR gates to flip the bits in errors. The decoder circuit of proposed $(24, 16)$ SEC-DED-DAEC code has been indicated in Fig. 4.20. As shown in Fig. 4.20, the syndrome computation part of the decoder generates syndrome bits and this part is designed using only XOR-2 gates. The error location detection part of the decoder identify the locations of errors and this part is implemented using 2-input AND (AND2) and 2-input OR (OR2) gates. Finally, the located errors are corrected by error correction part which employ XOR2 gates to flip the bits in errors.

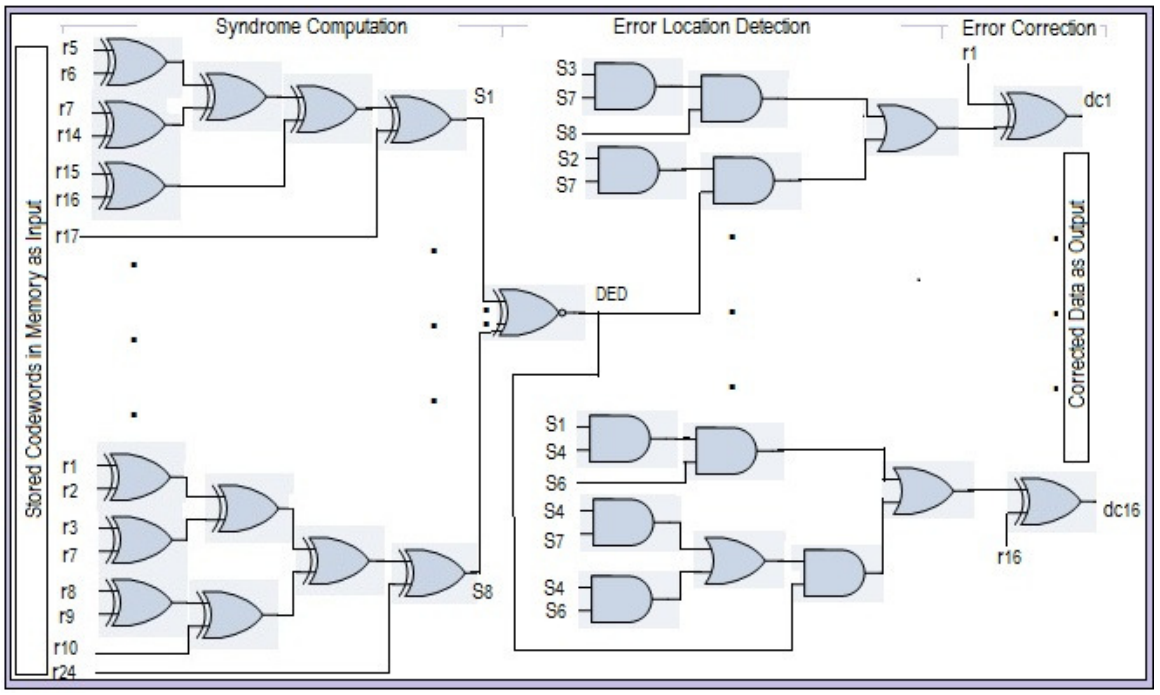

FIGURE 4.20: Decoder Circuit of Proposed (24, 16) SEC-DED-DAEC Code

TABLE 4.4: Comparison of FPGA-based Synthesis Results for Proposed Codecs

Codecs	Area (LUTs)	Delay (ns)
Neale (24, 16) [129], [130]	**70**	**4.931**
Reviriego (23, 16) [132]	**65**	**3.634**
Proposed (24, 16)	**58**	**4.084**
Neale (41, 32) [129], [130]	**129**	**6.278**
Reviriego (39, 32) [132]	**114**	**4.970**
Proposed (42, 32)	**101**	**5.338**
Neale (75, 64) [129], [130]	**260**	**6.727**
Reviriego (73, 64) [132]	**214**	**5.904**
Proposed (77, 64)	**197**	**6.651**

4.4.3 Synthesis Results

The proposed SEC-DED-DAEC codecs have been simulated and synthesized on FPGA based Artix7 (XC7A100T-2csg324) device family. The FPGA based synthesis results of proposed codecs have been compared with other related codecs in [129]-[130], [132]. These comparisons are summarized in Table 4.4. As shown in Table 4.4, proposed codecs require lesser number of LUTs compared to both the existing

codecs presented in [129], [130] and [132] with first scheme of optimized decoding. Also proposed codecs are more area (LUTs) efficient than the codecs in [129], [130] compared to the codecs in [132]. The delay performance of proposed codecs are better than the codecs presented in [129], [130]. But delay of proposed codecs are worse compared to the codecs in [132]. Graphical comparison of improvements in

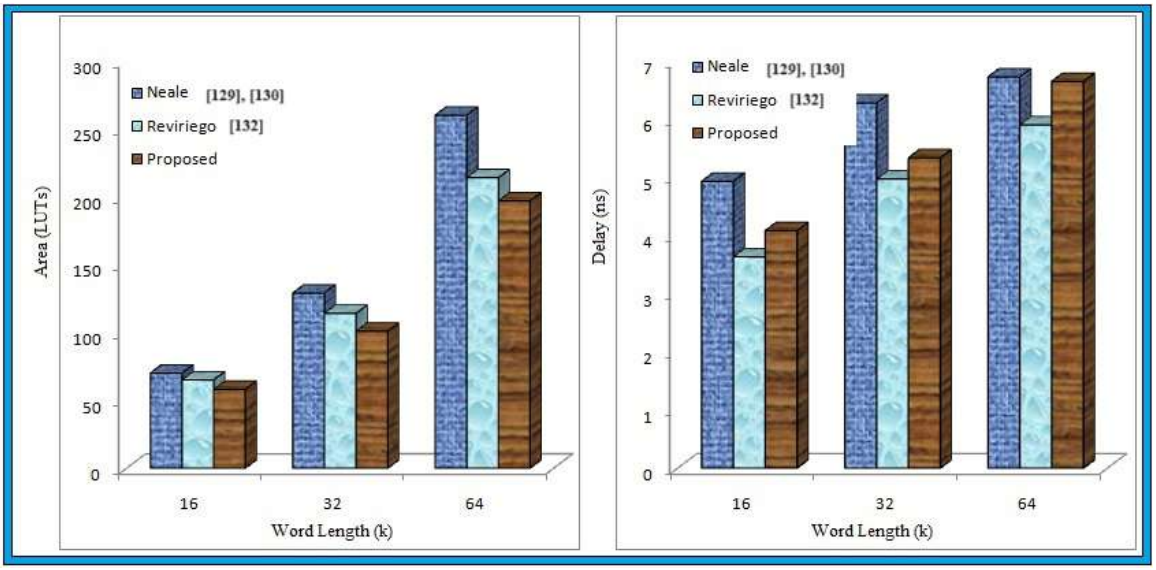

FIGURE 4.21: Graphical Comparison of Improvements in LUTs and Delay of Proposed Codecs

LUTs and delay have been indicated in Fig. 4.21. Proposed codecs exhibit better performance in terms of LUTs compared to the existing codecs. Also improvement in terms of LUTs for proposed codecs compared to the codecs in [129], [130] increases with the increase in word length. But improvement diminishes with increase in word length against codecs in [132]. Also delay performances of proposed codecs are better with respect to codecs by Neale et al. only and these improvement decreases with the increase in word length. The proposed $(77, 64)$ codec exhibits the highest improvement of 24.23% in LUTs against Neale et al. $(75, 64)$ codec. Whereas the highest improvement of 17.18% has been attained by proposed $(24, 16)$ codec against Neale et al. $(24, 16)$ codec.

4.5 Design of an Efficient Decoding Scheme for SEC-DED-DAEC Code with Odd Column Weight H-matrix

In this section, an efficient decoding technique has been presented to minimize the area and delay requirement of SEC-DED-DAEC codes with odd column weight H-matrix.

4.5.1 Proposed Decoding Scheme

In this section, the proposed decoding scheme for SEC-DED-DAEC code with odd column weight H-matrix has been presented in details. The proposed decoding rules are as follows:

step-1: The syndrome bits are computed.

step-2: The single error is located for each data bit position by considering both 1's and 0's in the syndrome value but with the properly selected variable length syndrome bits.

step-3: The adjacent double errors are located in each data bit position by considering only the 1's the corresponding syndrome bits.

step-4: The located single and adjacent double errors are corrected for each data bit positions.

The main difference between the proposed and existing second scheme of decoding of SEC-DED-DAEC code by Reviriego et al. lies in the step-2 of the proposed decoding rules. All the syndrome bits have been utilized in the second scheme of decoding by Reviriego et al. for locating single error. Whereas the proposed scheme of decoding utilizes a variable length syndrome bits for locating single error in each data bit. These variable length syndrome bits for locating single error in each data

TABLE 4.5: Comparison of Single Error Locating Logical expressions of $(23, 16)$ SEC-DED-DAEC Decoder by Reviriego et al. and Proposed Schemes

Data bit	**Second scheme by Reviriego et al. [132]**	**Proposed scheme**
d_1	$s'_1\&s_2\&s'_3\&s_4\&s'_5\&s_6\&s'_7$	$s_2\&s_4\&s'_5\&s_6\&s'_7$
d_2	$s_1\&s'_2\&s'_3\&s'_4\&s'_5\&s_6\&s_7$	$s_1\&s'_5\&s_6\&s_7$
d_3	$s'_1\&s'_2\&s_3\&s_4\&s'_5\&s_6\&s'_7$	$s_3\&s_4\&s'_5\&s_6\&s'_7$
d_4	$s'_1\&s_2\&s'_3\&s_4\&s'_5\&s'_6\&s_7$	$s'_1\&s_2\&s'_3\&s_4\&s'_5\&s'_6\&s_7$
d_5	$s'_1\&s'_2\&s_3\&s'_4\&s_5\&s'_6\&s_7$	$s'_2\&s_3\&s'_4\&s_5\&s'_6\&s_7$
d_6	$s'_1\&s_2\&s'_3\&s'_4\&s'_5\&s_6\&s_7$	$s_2\&s'_3\&s'_4\&s_6\&s_7$
d_7	$s'_1\&s'_2\&s_3\&s'_4\&s_5\&s_6\&s'_7$	$s'_2\&s_3\&s'_4\&s_5\&s_6\&s'_7$
d_8	$s'_1\&s'_2\&s'_3\&s_4\&s_5\&s'_6\&s_7$	$s'_2\&s'_3\&s_4\&s_5\&s'_6\&s_7$
d_9	$s'_1\&s_2\&s'_3\&s'_4\&s_5\&s_6\&s'_7$	$s_2\&s'_3\&s'_4\&s_5\&s_6$
d_{10}	$s'_1\&s'_2\&s'_3\&s_4\&s'_5\&s_6\&s_7$	$s'_2\&s'_3\&s_4\&s'_5\&s_6\&s_7$
d_{11}	$s'_1\&s_2\&s'_3\&s'_4\&s_5\&s'_6\&s_7$	$s_2\&s'_3\&s'_4\&s_5\&s_7$
d_{12}	$s'_1\&s'_2\&s_3\&s_4\&s_5\&s'_6\&s'_7$	$s'_2\&s_3\&s_4\&s_5\&s'_6\&s'_7$
d_{13}	$s'_1\&s'_2\&s_3\&s'_4\&s'_5\&s_6\&s_7$	$s'_2\&s_3\&s'_4\&s'_5\&s_6\&s_7$
d_{14}	$s'_1\&s'_2\&s'_3\&s_4\&s_5\&s_6\&s'_7$	$s'_2\&s'_3\&s_4\&s_5\&s_6\&s'_7$
d_{15}	$s'_1\&s'_2\&s_3\&s_4\&s'_5\&s'_6\&s_7$	$s'_1\&s'_2\&s_3\&s_4\&s'_5\&s'_6\&s_7$
d_{16}	$s'_1\&s'_2\&s'_3\&s'_4\&s_5\&s_6\&s_7$	$s'_1\&s'_3\&s'_4\&s_5\&s_6\&s_7$

bit have been selected properly by avoiding any conflicts from other single and double adjacent errors locating syndromes. Therefore, these SEC syndromes are unique from each others and from DAEC syndromes. The logical expressions for locating single error in each data bit for the proposed and the second scheme of decoding by Reviriego et al. have been summarized in Table. 4.5 for $(23, 16)$ SEC-DED-DAEC decoder in [132]. As shown in Table 4.5, the expressions for locating single error are more compact in the proposed scheme of decoding. Hence, lesser number of AND and NOT logic gates are required for designing the proposed decoding scheme compared to the second decoding scheme by Reviriego et al.. The proposed decoder circuit for correcting the second data bit of $(23, 16)$ SEC-DED-DAEC code in [132] has been shown in Fig. 4.22.

The comparison of theoretical area complexities in terms of equivalent 2-input NAND (NAND2) gates for proposed and existing related decoding schemes have been presented in Table 4.6. As shown in Table 4.6, the first scheme of decoding by Reviriego et al. and the decoding scheme by Neale et al. require lowest and highest number of equivalent NAND2 gates respectively for all the word lengths. The

TABLE 4.6: Comparison of Area Requirements in Terms of Equivalent NAND2 Gates

k	Decoder	XOR2	OR2	AND2	NOT	Equiv. NAND2
16	**Reviriego et al. Scheme1 [132]**	70	31	151	0	675
	Reviriego et al. Scheme2 [132]	64	31	189	64	791
	Neale et al. Scheme [129], [130]	58	31	282	173	1062
	Proposed Scheme	64	31	170	42	731
32	**Reviriego et al. Scheme1 [132]**	134	62	285	0	1292
	Reviriego et al. Scheme2 [132]	128	62	381	128	1588
	Neale et al. Scheme [129], [130]	130	62	564	305	2139
	Proposed Scheme	128	62	362	112	1534
64	**Reviriego et al. Scheme1 [132]**	264	127	573	0	2583
	Reviriego et al. Scheme2 [132]	256	127	893	384	3575
	Neale et al. Scheme [129], [130]	275	126	1330	714	4852
	Proposed Scheme	275	126	717	207	3119

theoretical area requirement of the proposed decoding scheme is lower compared to Neale et al. scheme and the second scheme of decoding by Reviriego et al. for all the word lengths. But the proposed scheme of decoding have slightly higher theoretical area requirement compared to first scheme of decoding by Reviriego et al..

4.5.2 FPGA-based Synthesis Results

The proposed decoding scheme has been applied to the odd column weight H-matrices presented in [132] for SEC-DED-DAEC codes. The proposed SEC-DED-DAEC decoders for the word length of 16, 32 and 64 bits have been simulated and synthesized on FPGA based Virtex6 (XC6VCX75T-2ff484) device family. The FPGA based synthesis results of proposed and existing SEC-DED-DAEC decoders in [129], [130], [132] have been compared in Table 4.7.

The area requirement of proposed decoding scheme with 16-bit word length is lower compared to the other existing schemes in Table 4.7. Also the proposed scheme with word lengths 32 and 64 bits have lower area requirement compared to the Neale et al. scheme and second scheme of decoding by Reviriego et al. for the same word lengths. But the area requirement of proposed scheme with 64-bit word length is slightly higher with respect to the first scheme of decoding of Reviriego et al.. So

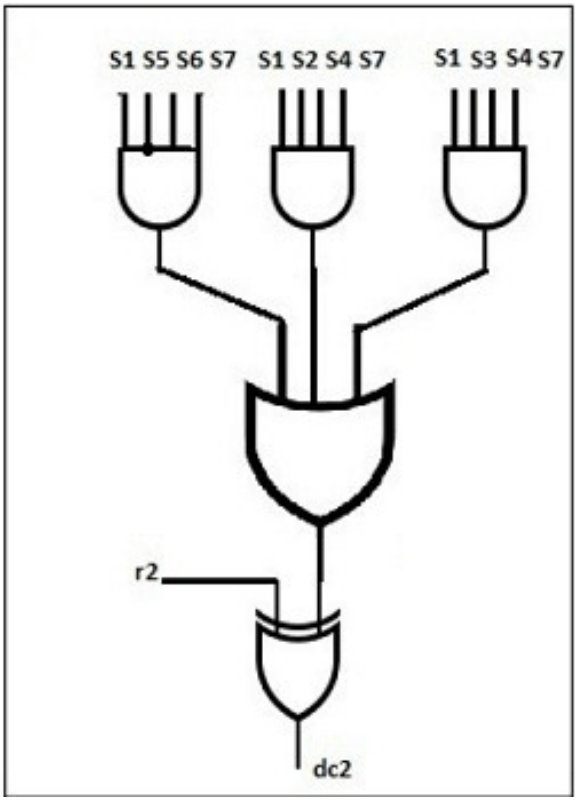

FIGURE 4.22: Proposed Scheme for Decoding the 2^{nd} Data Bit of $(23, 16)$ SEC-DED-DAEC Code

the area requirement of the proposed decoding scheme lies in between the area and delay optimized decoding schemes by Reviriego et al. for the world length of 32 and 64 bits. The proposed scheme of decoding exhibits the delay requirement which stands in between the delay requirement of both the existing schemes by Reviriego et al. for the word lengths of 16 and 32 bits. The existing decoding scheme by Neale et al. provides slightly better delay performance compared to proposed scheme for these word lengths. But the proposed scheme with 64-bit word length has the lowest delay requirement compared to all the existing schemes in Table 4.7.

TABLE 4.7: Comparison of FPGA-based Synthesis Results

Word length (k)	Decoding Schemes	Area (LUTs)	Delay (ns)
16	**Reviriego et al. Scheme1 [132]**	55	3.20
	Reviriego et al. Scheme2 [132]	53	3.44
	Neale et al. Scheme [129], [130]	51	3.31
	Proposed Scheme	47	3.39
32	**Reviriego et al. Scheme1 [132]**	93	4.39
	Reviriego et al. Scheme2 [132]	102	3.96
	Neale et al. Scheme [129], [130]	108	5.29
	Proposed Scheme	97	4.02
64	**Reviriego et al. Scheme1 [132]**	175	5.15
	Reviriego et al. Scheme2 [132]	216	4.05
	Neale et al. Scheme [129], [130]	210	5.40
	Proposed Scheme	202	3.95

4.6 Conclusions

In this chapter, three different SEC-DED-DAEC codes have been introduced for protecting storage systems from MCUs. Reductions in decoding overheads and mis-correction rate have been the major objectives of these SEC-DED-DAEC codes. The key novelty of work presented in section 4.3 is the construction of two new schemes of SDECC-based on two different parity columns assigning rules for reducing the mis-correction rate and codec design overheads like area and power consumption. The mis-correction rates of proposed SDECC codes are up to 88.24% and 80.67% lesser for 2-random and 3-random bit errors respectively with respect to related existing designs. It is observed that area and power consumption of the proposed codecs are reduced up to 8.93% and 21.47% compared to the recently published results respectively. The compact SEC-DED-DAEC code (presented in section 4.4) for memory application has lower area and delay requirements compared to related works. The efficient decoding scheme for SEC-DED-DAEC codes with odd column weight H-matrix (presented in section 4.5) have moderate area and delay overheads compared to other related decoders. The SEC-DED-DAEC codes and their respective decoding techniques which are introduced in this chapter can be further extended for the

design of adjacent ECCs with higher error correction capabilities in future. The design of ECC schemes with higher error correction capabilities like SEC-DAEC-TAEC and Burst Error Correcting (BEC) codes have have been reported in the next chapter for application in storage systems.

Chapter 5

Design and Implementation of SEC-DAEC-TAEC and Burst Error Correcting Codes

5.1 Introduction

The Single Error Correction-Double Adjacent Error Correction-Triple Adjacent Error Correction (SEC-DAEC-TAEC) codes are another form of adjacent error correcting codes which are capable of correcting single error, double adjacent and triple adjacent errors. The extension of SEC-DAEC-TAEC codes are termed as SEC-DED-DAEC-TAEC codes which have an extra Double Error Detection (DED) capability. Both the SEC-DAEC-TAEC and SC-DED-DAEC-TAEC codes have been mostly employed in storage systems for reducing the effects of soft errors. The main advantage of TAEC codes is their better error correction capability compared to SEC, SEC-DAEC and SEC-DED-DAEC codes. But the decoding complexity of TAEC decoders increase as a consequence of added error correction capability. Therefore, minimization of decoding complexity is the main challenge in SEC-DAEC-TAEC

code design. Also mis-correction rate of SEC-DAEC-TAEC code is another factor which must be reduced as far as possible.

On the other hand, the t-bit Burst Error Correcting (BEC) code is capable of correcting all possible errors confined to a code of window size t-bit. Thus BEC codes are proficient of correcting all possibilities of errors within the burst size. These error possibilities are single error, adjacent errors and almost adjacent errors. The 3-bit BEC code is capable of correcting single error, double adjacent, triple adjacent and double almost adjacent errors. Similarly, the 4-bit BEC code is proficient of correcting single error, double adjacent, triple adjacent, quadruple adjacent, double almost adjacent and triple almost adjacent errors. The BEC codes have superior error correction proficiency compared to adjacent ECCs. Due to this fact the BEC codes are employed in storage systems for correcting error burst. But BEC codes have more complex decoder circuits as compared to adjacent ECCs. Also the complexity of BEC decoder circuit grows exponentially with the increase in correction capability. Therefore, lower complexity BEC code design is most desirable for memory application and it is a challenging task to the designers.

In this chapter, two SEC-DAEC-TAEC coding schemes have been introduced for storage systems. The first scheme of SEC-DAEC-TAEC code reduces the decoding complexity and mis-correction rate while correcting single error, double and triple adjacent errors. Whereas the second scheme of SEC-DAEC-TAEC decoder reduces the decoding complexity of existing SEC-DAEC-TAEC codes for SRAMs applications. In this chapter, the construction technique and evaluation of high performance t-bit BEC code has been presented for correcting MCUs in memory applications.

The remaining of this chapter is structured as follows. The related SEC-DAEC-TAEC codes are presented in section 5.2. Section 5.3 presents the design of FPGA-based low delay 3-bit adjacent errors correcting codec. Lower complexity error location detection block for 3-bit adjacent errors correcting decoder has been presented in section 5.4. The section 5.5 provides construction techniques and evaluation of t-bit burst error correcting codes for protecting MCUs. Finally, this chapter is concluded in section 5.6.

5.2 Existing SEC-DAEC-TAEC and BEC Codes

The existence of varieties of SEC-DAEC-TAEC codes [149], [131], [125], [150] [148], [151] are found in the literature for the protection of memories against MCUs. These codes are proficient of rectifying single and up to triple adjacent errors. The SEC-DED-DAEC codes by Neale et al. [129], [130] have been modified further in [130], [131] for the inclusion of TAEC feature. The Neale et al.'s SEC-DAEC-TAEC codes have the lower mis-correction rate, scalable adjacent errors and double error detection capability like SEC-DED-DAEC codes. But Neale et al.'s SEC-DED-DAEC and SEC-DAEC-TAEC codes have not focused on the reduction of encoding and decoding circuits design overheads. The SEC-DAEC-TAEC codes have been designed with minimum number of parity bits and reduced design constrains by Adalid et al. [125] for SRAMs. But Adalid et al.'s SEC-DAEC-TAEC codes suffer from higher mis-correction rates with respect to 2-random and 3-random errors. The Karnaugh map based design of multiple adjacent ECCs have been introduced in [150] for faster encoding and decoding operations at the cost of increased redundancy. The latest addition of TAEC code in the literature is the power efficient and high speed SEC-DAEC and SEC-DAEC-TAEC codes by Tripathi et al. [148], [151]. But Tripathi et al. codes have higher redundancy compared to other related existing TAEC codes. Thus, the main concerns in designing the existing SEC-DAEC-TAEC codes are the reduction of mis-correction rate and codec circuit design overheads. The Quadruple Adjacent Error Correction (QAEC) codes [152], [153], [154] are also found in the literature for memory applications. These codes are capable of correcting from single error up to 4-bit adjacent errors. But these codes have the highest decoding circuit complexity among all others existing AEC codes available in the literatures [133], [132], [129], [130], [131], [125], [148].

In addition, different BEC codes have been employed for memory protection against MCUs. These codes are proficient of rectifying single error, adjacent and almost adjacent errors caused by MCUs. The 3-bit BEC codes have been introduced in [155] for enhancing the reliability of memories. Two schemes of optimization have

been applied for constructing lower redundancy 3-bit BEC codes [125]. The BEC codes [156] have been designed for the detection of burst errors. Two schemes of 3-bit BEC codes [157], [158] have been presented by Klockmann et al. for memory protections. In [159], a new BEC code has been presented by modifying the general BEC Hamming code to achieve considerable lower area of the decoder. A lower delay 3-bit BEC code has been introduced by Li et al. in [126]. The protection of larger word lengths memories have been performed by employing 3-bit BEC codes in [160]. The parallel decoding of lower redundancy BEC codes have been introduced in [161] for reducing the design overheads of decoding circuits. Recently, a 3-bit BEC code [162] has been presented based on region selection code for achieving the better correction capability. Two methods of implementing 4-bit BEC codes have been introduced in [127]. Lower decoding complexity has been achieved by the first method of designing 4-bit BEC code [127] compared to conventional second method of designing 4-bit BEC code. The 5-bit BEC code [163] has been introduced for improving the memory reliability against MCUs. Therefore, the concern of existing BEC codes are the reductions in decoding overheads as these overheads increase with error correction capability of any BEC code.

5.3 Design of FPGA-based Low Delay TAEC Codec

In this section, a new SEC-DED-DAEC-TAEC code has been proposed for the word lengths of 16, 32 and 64 bits. The proposed codecs for three different word lengths have been simulated and synthesized in FPGA platform. The proposed codecs have lower delay compared to the existing related works. Also mis-correction rate of the proposed codes is lower compared to the related works.

5.3.1 Proposed SEC-DED-DAEC-TAEC Code

In this section, details of proposed SEC-DED-DAEC-TAEC codes are presented. The H-matrices of proposed SEC-DED-DAEC-TAEC codes have been constructed by considering the following rules:

1. All the columns must be different from each other, non-zero and have a constant weight equals to 3.

2. All the XOR sums of any two adjacent columns which include at least one data column must be different from each other and have a constant weight equals to 4.

3. All the XOR sums of any three adjacent columns which include at least one data column must be different from each other as well as from any column of H-matrix and have a constant weight equals to 3.

The first condition provides SEC functionality of the proposed codes. The second and third conditions are for the double and triple adjacent errors correction capabilities respectively. Also the DED feature of the proposed codes have been confirmed by the first and second conditions jointly as the DED syndrome is even weighted in proposed scheme. The required number of parity bits for the feasibility of the proposed (n, k) SEC-DED-DAEC-TAEC code in systematic form is bounded by the following equation (5.1).

$$\binom{n-k}{3} \geq (k+n-2) \quad \text{and} \quad \binom{n-k}{4} \geq k \tag{5.1}$$

where $(n-k)$ is the number of parity bits. In equation (5.1), k and $(n-2)$ number of weight three syndromes which are required for SEC and TAEC functionality respectively. Hence, the total number of combinations of weight three in $(n-k)$ bits must be greater than equals to required number of weight three syndromes. Similarly, total number of combinations of weight four syndromes in $(n-k)$ bits must be greater than or equals to the required number of weight four syndromes. The required numbers of parity bits for the proposed SEC-DED-DAEC-TAEC codes with different word lengths have been listed in Table 5.1 based on equation (5.1).

The proposed H-matrices with word lengths 16, 32 and 64 bits have been presented in Fig. 5.1, Fig. 5.2 and Fig. 5.3 respectively.

$$\begin{bmatrix}
010110011010101000000001 \\
100100000101010100000000 \\
011010101000000010000000 \\
110000110110000001000010 \\
000011010000110000100000 \\
001001000001100000010011 \\
000000001100010000001010 \\
101101100011000000000101
\end{bmatrix}$$

FIGURE 5.1: Proposed H-matrix for (24, 16) SEC-DED-DAEC-TAEC Code

$$\begin{bmatrix}
00100000010100010110000001001101000000000 \\
11000000110000001010101100000010100000001 \\
10110001101101011000110000000110010000000 \\
01011000000110000011000101100000001000000 \\
00001101000001100000000011011010000100000 \\
00000110100000000010101011000000000010001 \\
10000011011000110000011000101000000001000 \\
00010100001011000001100000010100000000101 \\
01101010000010101100001010110000000000010
\end{bmatrix}$$

FIGURE 5.2: Proposed H-matrix for (41, 32) SEC-DED-DAEC-TAEC Code

$$\begin{bmatrix}
0000000000000000000100010000110000000000010110001010001000001010000000000001 \\
0000100101101101010000000010010101100000000000101101100000000101000000000000 \\
0011010110110110110110000000000110000110110000000000011011000000100000000000 \\
1010000000010000001000001101100000010101100000000011000000010100010000000000 \\
1101101100000001100001011000000011011000000000011000000101100000001000000010 \\
0000000000001011000000000111010101010110000000000000101011000000000010000010 \\
0100011000000100000010101011011000000000000101100000000010110000000100000000 \\
0001000001011000011011000000000001000011000010101101100000000000000001000000 \\
0000001011000000000000110100000000110000001011000000000001011000000000100101 \\
0110110000100000101101100000000000000101001100000100000010110000000000010110 \\
1000000010000010000000000001001100000001010110000001011000000000000000001011
\end{bmatrix}$$

FIGURE 5.3: Proposed H-matrix for (75, 64) SEC-DED-DAEC-TAEC Code

TABLE 5.1: Required Number of Parity Bits for Proposed Codes

k	$(n-k)$	Number of weight three syndromes		Number of weight four syndromes	
		Available	**Required**	**Available**	**Required**
16	8	56	38	70	16
32	9	84	71	136	32
64	11	165	137	330	64

TABLE 5.2: Comparison of Mis-correction Rate of Proposed Codec

Different Codec Schemes	**Miscorrection Rate (%)**	
	2-random errors	**3-random errors**
Proposed (24, 16) SEC-DED-DAEC-TAEC	13.83	38.71
Neale (24, 16)-I5 SEC-DED-DAEC-TAEC [130], [131]	16.20	39.70
Adalid (22, 16) SEC-DAEC-TAEC [125]	98.57	98.22
Proposed (41, 32) SEC-DED-DAEC-TAEC	16.03	36.92
Neale (41, 32)-I5 SEC-DED-DAEC-TAEC [130], [131]	21.80	43.70
Adalid (39, 32) SEC-DAEC-TAEC [125]	89.76	89.65
Proposed (75, 64) SEC-DED-DAEC-TAEC	12.74	24.36
Neale (75, 64)-I5 SEC-DED-DAEC-TAEC [130], [131]	11.40	25.30
Adalid (72, 64) SEC-DAEC-TAEC [125]	84.71	83.79

5.3.2 Mis-correction Rate of Proposed Codes

The mis-correction rates of proposed and existing codes in [130], [131] and [125] have been computed for double random and triple random errors respectively and summarized in Table 5.2. The mis-correction rate against double random errors of proposed codes are slightly lower compared to Neale et al. codes for all the word lengths except for 64-bit word length. But in case of triple random errors, the proposed and Neale et al. [130], [131] codes have nearly equal mis-correction rate for different word lengths. The Adalid et al. codes have the highest mis-correction rates against both types of random errors for all the word lengths compared to proposed and Neale et al. codes [130], [131].

5.3.3 Design of Proposed Codec

In proposed codec, the parity bits are computed using encoder circuit. The expressions for the parity bits of proposed (24, 16) SEC-DED-DAEC-TAEC code are shown in equation (5.2).

$$\begin{aligned}
c_1 &= d_2 \oplus d_4 \oplus d_5 \oplus d_8 \oplus d_9 \oplus d_{11} \oplus d_{13} \oplus d_{16} \\
c_2 &= d_1 \oplus d_4 \oplus d_{10} \oplus d_{12} \oplus d_{14} \\
c_3 &= d_2 \oplus d_3 \oplus d_5 \oplus d_7 \oplus d_9 \\
c_4 &= d_1 \oplus d_2 \oplus d_7 \oplus d_8 \oplus d_{10} \oplus d_{11} \oplus d_{15} \\
c_5 &= d_5 \oplus d_6 \oplus d_8 \oplus d_{13} \oplus c_{14} \\
c_6 &= d_3 \oplus d_6 \oplus d_{12} \oplus d_{13} \oplus d_{15} \oplus d_{16} \\
c_7 &= d_9 \oplus d_{10} \oplus d_{14} \oplus d_{15} \\
c_8 &= d_1 \oplus d_3 \oplus d_4 \oplus d_6 \oplus d_7 \oplus d_{11} \oplus d_{13} \oplus d_{16}
\end{aligned} \tag{5.2}$$

where c_i for i=1, 2, . . ., 7, 8, and d_j for j=1, 2, . . ., 15, 16 represent parity bits and data bits respectively. These parity equations are implemented by employing XOR logic gates in the encoder circuit. On the other hand, the proposed decoder accepts the received codeword (r) as input and produces the corrected version of data as output. The flow diagram of proposed decoding technique has been shown in Fig. 5.4. The first task of proposed decoder is to compute the syndrome bits (S). The expressions for eight syndrome bits of proposed (24, 16) SEC-DED-DAEC-TAEC

code have been shown in equation (5.3).

$$\begin{aligned}
S_1 &= r_2 \oplus r_4 \oplus r_5 \oplus r_8 \oplus r_9 \oplus r_{11} \oplus r_{13} \oplus r_{15} \oplus r_{24} \\
S_2 &= r_1 \oplus r_4 \oplus r_{10} \oplus r_{12} \oplus r_{14} \oplus r_{16} \\
S_3 &= r_2 \oplus r_3 \oplus r_5 \oplus r_7 \oplus r_9 \oplus r_{17} \\
S_4 &= r_1 \oplus r_2 \oplus r_7 \oplus r_8 \oplus r_{10} \oplus r_{11} \oplus r_{18} \oplus r_{23} \\
S_5 &= r_5 \oplus r_6 \oplus r_8 \oplus r_{13} \oplus r_{14} \oplus r_{19} \\
S_6 &= r_3 \oplus r_6 \oplus r_{12} \oplus r_{13} \oplus r_{20} \oplus r_{23} \oplus r_{24} \\
S_7 &= r_9 \oplus r_{10} \oplus r_{14} \oplus r_{21} \oplus r_{23} \\
S_8 &= r_1 \oplus r_3 \oplus r_4 \oplus r_6 \oplus r_7 \oplus r_{11} \oplus r_{12} \oplus r_{22} \oplus r_{24}
\end{aligned} \tag{5.3}$$

The second task of decoder is to locate the errors in case of non-zero syndrome bits. In proposed decoder two complementary signals namely $notDED$ and DED are generated based on odd and even numbers of errors respectively. As the proposed H-matrices have odd weighted columns, thus weight of syndrome is also odd in case of odd numbers of errors. Similarly even weighted syndrome implies even numbers of errors. Therefore, non-zero syndrome value implies that either $notDED$ or DED has been satisfied. In the proposed decoder, single and triple adjacent errors are corrected when $notDED$ signal is satisfied and this signal is expressed as the XOR sum of all the syndrome bits. Also the DED signal which is the complement of $notDED$ signal is used for correcting double adjacent errors. Also errors are located by considering only the 1's in the syndrome values. The error location determination (ELD) logic for the first data bit of proposed (24, 16) decoder has been shown in equation (5.4).

$$ELD_1 = ((((S_2 \& S_4 \& S_8) | (S_1 \& S_2 \& S_6)) \& nD) | (S_1 \& S_2 \& S_3 \& S_8 \& D)) \tag{5.4}$$

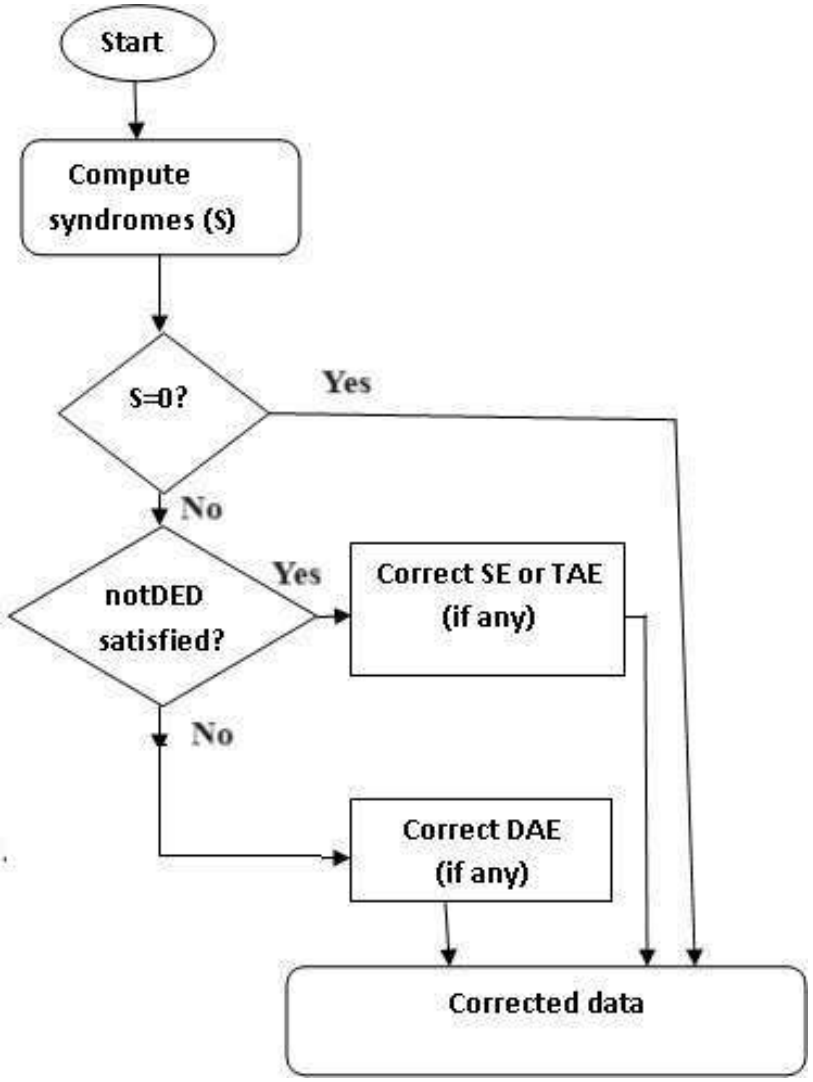

FIGURE 5.4: Flow Chart of Proposed Decoding Technique

(5.5)

where '&' indicates AND operation, nD and D are the $notDED$ and DED signals respectively which have been presented in equations (5.6) and (5.7).

$$nD = S_1 \oplus S_3 \oplus \cdots \oplus S_7 \oplus S_8 \tag{5.6}$$

and

$$D = nD' \tag{5.7}$$

Thus, the ELD expressions of proposed decoder are more simplified compared to traditional SEC-DED-DAEC-TAEC decoder [125], [130]- [131] . In proposed decoder, only one NOT gate is required to generate DED signal from $notDED$ signal. The third and final task of the proposed decoder is to correct the located errors and this is performed by employing XOR logic.

5.3.4 FPGA-based Synthesis Results

The proposed SEC-DED-DAEC-TAEC codecs have been simulated and synthesized on FPGA based virtex 7 (xc7vx330t-3ffg1157) device families. These synthesis results of proposed and existing related works by Neale et al. and Adalid et al. have been summarized in Table 5.3. As shown in Table 5.3, the SEC-DAEC-TAEC codec by Adalid et al. requires the lowest number of Lookup Tables (LUTs) for 16-bit word length. But the proposed 16-bit codec needs lesser number of LUTs compared to Neale et al. 16-bit codec. In case of 32-bit word length, the LUTs requirement of the proposed and existing codecs are the same. The LUTs requirement of proposed and Adalid et al. 64-bit codecs are nearly equal and slightly lower compared to Neale et al. 64-bit codec. The proposed codecs experience lower delay compared to all the existing codecs for all three word lengths as indicated in Table 5.3. But the highest improvement in delay has been achieved against Neale et al. 16-bit codec.

TABLE 5.3: Comparisons of FPGA-based Synthesis Results

Different Codec Schemes	Area (LUTs)	Delay (ns)	Improvement (%)	
			Area	Delay
Proposed (24, 16) SEC-DED-DAEC-TAEC	83	2.82	-	-
Neale (24, 16)-I5 SEC-DED-DAEC-TAEC [130], [131]	94	3.53	13.20	25.17
Adalid (22, 16) SEC-DAEC-TAEC [125]	57	3.35	-31.33	18.80
Proposed (41, 32) SEC-DED-DAEC-TAEC	163	3.99	-	-
Neale (41, 32)-I5 SEC-DED-DAEC-TAEC [130], [131]	163	4.20	0	5.26
Adalid (39, 32) SEC-DAEC-TAEC [125]	165	4.41	–1.22	10.53
Proposed (75, 64) SEC-DED-DAEC-TAEC	338	4.01	-	-
Neale (75, 64)-I5 SEC-DED-DAEC-TAEC [130], [131]	364	4.74	7.70	18.20
Adalid (72, 64) SEC-DAEC-TAEC [125]	329	4.60	-2.67	14.71

5.4 Design of Compact ELD Block for SEC-DAEC-TAEC Decoders

The K-map based simplification method presented in chapter 3, section 3.4 has also been applied to the existing SEC-DAEC-TAEC codes for the simplification of ELD block of these decoders. The obtained theoretical and synthesis results are summarized in this section.

5.4.1 Theoretical Gate Count

The theoretical gate count of existing SEC-DAEC-TAEC decoders by Neale et al. and Adalid et al. have been computed by counting the total number of equivalent 2-input NAND (NAND2) gates. The comparison of theoretical area over heads of existing and proposed compact decoders has been presented in Table 5.4. The proposed compact decoder require lesser numbers of equivalent NAND2 gates compared to the both existing TAEC decoders introduced by Neale et al. and Adalid et al. for all the word lengths. The proposed compact decoder provides the highest improvement of 39.83%, 40.55% and 44.77% in estimated area for the word lengths of 16, 32 and 64 bits respectively against the corresponding word length decoders by Neale et al.. But the theoretical area improvements of modified compact Adalid et al.'s decoder is lower compared to Neale et al.'s decoder. The theoretical gate count

TABLE 5.4: Area Requirements in terms of Logic Gates

Scheme	k	Code	Existing Decoders	Compact Decoder	Improv. (%)
			Equiv. NAND2		
TAEC	16	Neale $(23, 16)$ [130], [131]	1848	1112	39.83
		Adalid $(22, 16)$ [125]	1493	1342	16.39
	32	Neale $(40, 32)$ [130], [131]	4197	2495	40.55
		Adalid $(39, 32)$ [125]	3744	3234	13.62
	64	Neale $(74, 64)$ [130], [131]	10888	6013	44.77
		Adalid $(72, 64)$ [125]	8636	7347	14.93

of proposed and existing decoders have been presented by employing a bar graph in Fig. 5.5. It is clear from this figure that theoretical area in terms of NAND2 gates of proposed compact decoder is lesser compared to both the existing TAEC decoders by Neale et al. and Adalid et al..

5.4.2 Synthesis Results

In this section, the synthesis results of proposed and existing SEC-DAEC-TAEC decoders on FPGA and ASIC platforms have been presented.

5.4.2.1 Synthesis Results in FPGA Platform

All the functional blocks of proposed and existing SEC-DAEC-TAEC decoders are simulated and synthesised using FPGA-based virtex 6 (6vcx75tff484-2) device family. The FPGA-based synthesis results have been shown in Table 5.5. The SEC-DAEC-TAEC (74, 64) code has the highest improvement in term of LUTs but SEC-DAEC-TAEC (39, 32) Adalid and co-workers code exhibits the highest improvement in decoding delay. Among all SEC-DED-TAEC codes the highest improvement in terms of LUTs is achieved in Neale et al.'s SEC-DAEC-TAEC (74, 64) code.

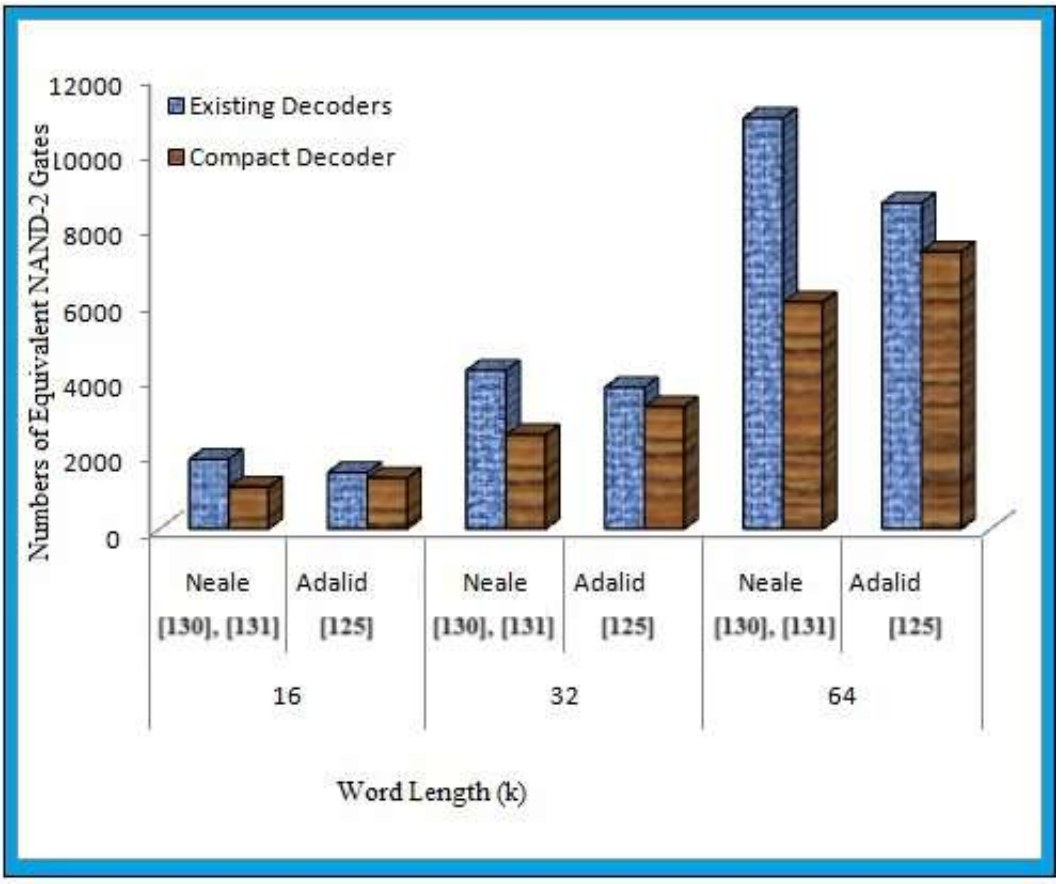

FIGURE 5.5: Graphical Compassion of Theoretical Gate Count

5.4.2.2 Synthesis Results in ASIC Platform

The SEC-DAEC-TAEC decoders have been further synthesised in ASIC platform using Cadence based Genus synthesis solution (TSMC18) tool. The ASIC-based synthesis results have been presented in Table 5.6 which shows similar trends of improvement in terms of area and delay of proposed decoder as observed in FPGA-based synthesis result. However, power consumption for the compact decoder is slightly higher compared to existing decoder for almost all SE-DAEC-TAEC codes

TABLE 5.5: FPGA-based Synthesis Results of Proposed and Existing SEC-DAEC-TAEC Decoders

Scheme	k	Code	Existing Decoders		Compact Decoder		Improv. (%)	
			Area (LUTs)	Delay (ns)	Area (LUTs)	Delay (ns)	Area	Delay
TAEC	16	Neale $(23, 16)$ [130], [131]	66	4.19	54	3.89	18.18	7.16
		Adalid $(22, 16)$ [125]	45	3.31	42	2.93	6.67	11.48
	32	Neale $(40, 32)$ [130], [131]	151	4.85	133	4.57	11.92	5.77
		Adalid $(39, 32)$ [125]	146	4.46	144	3.86	1.37	13.45
	64	Neale $(74, 64)$ [130], [131]	324	5.08	232	4.9	28.40	3.54
		Adalid $(72, 64)$ [125]	289	5.12	280	5.25	3.11	-2.44

TABLE 5.6: ASIC-based Synthesis Results of SEC-DAEC-TAEC Decoders

Scheme	k	Code	Existing Decoders			Compact Decoder			Improv. (%)		
			Area (μm^2)	Delay (ps)	Power (mW)	Area (μm^2)	Delay (ps)	Power (mW)	Area	Delay	Power
TAEC	16	Neale $(23, 16)$ [130], [131]	2601	398	0.38	2165	375	0.39	16.78	5.78	-2.63
		Adalid $(22, 16)$ [125]	2279	465	0.44	2215	350	0.48	2.78	24.65	-9.09
	32	Neale $(40, 32)$ [130], [131]	4956	484	1.35	4003	478	1.40	19.23	1.24	-3.70
		Adalid $(39, 32)$ [125]	4534	521	0.88	4360	370	1.01	3.83	28.96	-14.77
	64	Neale $(74, 64)$ [130], [131]	8792	857	1.62	6435	852	1.80	26.81	0.58	-11.11
		Adalid $(72, 64)$ [125]	8785	973	2.22	8522	970	2.30	3.00	0.30	-3.60

listed in Table 5.6. In brief, the proposed compact decoders provide considerable reduction in area and delay of TAEC decoder at the cost of slight increase in power consumption.

5.5 Construction Technique and Performance Evaluation of t-bit Burst Error Correcting Code for Protecting MCUs

In this section, a new class of high performance t-bit BEC code has been introduced for protecting MCUs. Parity check matrices (H-matrices) have been proposed for 3-bit and 4-bit BEC codes with word lengths of 16-bit, 32-bit and 64-bit. Also a simplified decoding scheme has been introduced for these codes. The proposed codecs have been designed and implemented in FPGA and ASIC platforms. Proposed codecs are compact, fast and power efficient compared to existing related schemes.

But these lower design constrains are achieved at the cost of increase in redundancy. Therefore the proposed codecs can be employed in applications where redundancy is not the only constrain for correcting t-bit burst errors caused by MCUs.

5.5.1 Construction of Proposed H-matrices for t-bit BEC Codes

In this section, the general construction methodology of proposed H-matrices for t-bit BEC codes are presented in details. Also proposed H-matrices for 3-bit and 4-bit BEC codes with world lengths of 16, 32 and 64 bits are provided.

The proposed H-matrices for t-bit BEC codes have been constructed in a manner such that it facilitates the low overheads decoding. For this purpose, $(n-k)$ rows of proposed t-bit BEC H-matrix are divided into t equal parts with $(\frac{n-k}{t})$ numbers of rows in each part. Proposed H-matrices for t-bit BEC codes are constructed by incorporating the following conditions:

1. All the t-bit burst error correcting syndromes must be exactly same for a particular code bit position in any of the t equal part of rows.

2. All the t-bit burst error correcting syndromes of length $(\frac{n-k}{t})$ must be non-zero and different from each other.

The structure adopted for the proposed t-bit BEC H-matrix has been shown in Fig. 5.6 to incorporate the above mentioned conditions. In this figure, the $(n-k)$ rows of H-matrix have been partitioned into t equal groups where each group contains exactly $(\frac{n-k}{t})$ rows. Now the rows of first group are assigned with different non-zero combination of $(\frac{n-k}{t})$ bit starting from the first column of H-matrix and then others columns in the same group which are separated by a multiple of t from the first column. The same process is repeated for other groups of rows by starting from second, third,...., t^{th} columns of H-matrix respectively. In Fig. 5.6, the highlighted portions indicate the non zero combinations of $(\frac{n-k}{t})$ bits. Also a particular combination of $(\frac{n-k}{t})$ bits can be utilized at most t times in t different divisions of rows

as these have different syndrome indexes for decoding. So decoding can be done by employing only one division of rows where all the t-bit BEC syndromes are exactly same.

The $(n-k)$ number of parity bits for designing the proposed t-bit (n,k) BEC code

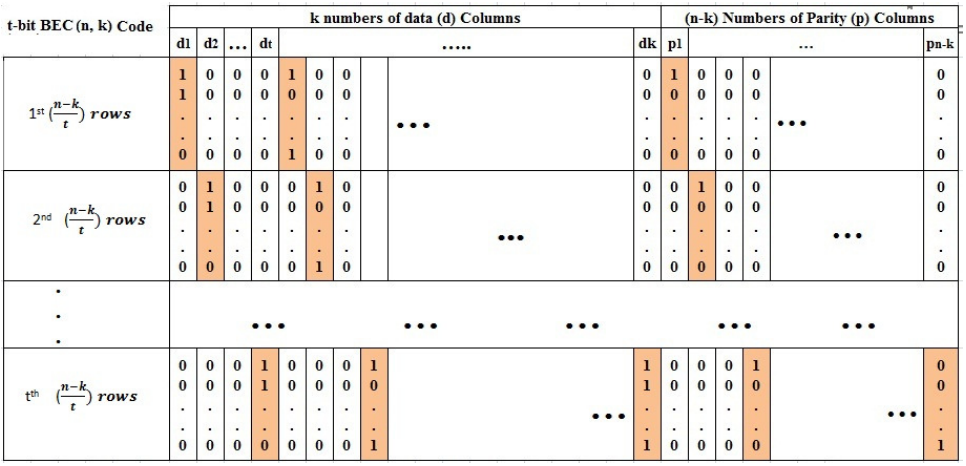

FIGURE 5.6: Structure of Proposed t-bit BEC H-matrix

must satisfy the equation (5.8).

$$(2^{\frac{n-k}{t}} - 1) \geq \frac{n}{t} \tag{5.8}$$

where the left and right hand sides of equation (5.8) indicate the available and required numbers of non-zero combinations of $(\frac{n-k}{t})$ bits respectively for constructing the proposed t-bit BEC codes. Also the proposed scheme of (n,k) t-bit BEC code requires n numbers of distinct error locating syndrome of length $(\frac{n-k}{t})$ bits. But the same non-zero combination of $(\frac{n-k}{t})$ bits can be reused at most t different divisions of rows as shown in Fig. 5.6. Considering this fact, the required numbers of non-zero combinations of $(\frac{n-k}{t})$ bits for the proposed scheme now reduces to $(\frac{n}{t})$ which has been indicated on the right hand side of equation (5.8). The minimum number of parity bits which are required to design proposed 3-bit BEC codes for word lengths (k) equal to 16, 32 and 64 bits are 12, 12 and 15 bits correspondingly. The respective numbers of parity bits for proposed 4-bit BEC codes are 12, 16 and 20 bits. These numbers of parity bits are slightly higher for proposed 3-bit and 4-bit BEC codes

compared to existing 3-bit and 4-bit BEC codes. As a result the parity area overhead of proposed BEC codes are higher compared to existing BEC codes. The proposed H-matrix of 3-bit and 4-bit BEC codes for different word lengths are shown in Fig. 5.7 to Fig. 5.12.

$$\begin{bmatrix}
1000000001001000001000000000 \\
1001001000000000000001000000 \\
0001000001000001000000001000 \\
0000001000001001000000000001 \\
0100000000100100100000000000 \\
0100100100000000000100000000 \\
0000100000100000000000100000 \\
0000000100000100000000000100 \\
0010000000010010010000000000 \\
0010010010000000000010000000 \\
0000010000010000000000010000 \\
0000000010000010000000000010
\end{bmatrix}$$

FIGURE 5.7: Proposed H-matrix for 3-bit BEC Code with 16-bit Word Length

$$\begin{bmatrix}
10000000010010000010000010010010010000000000 \\
10010000000000010010010010000010000010000000 \\
00010010010000000010010000010010000000010000 \\
00000010000010010000010010010010000000000010 \\
01000000010010000010000010010010010000000000 \\
01001000000000010010010010000010000010000000 \\
00001001001000000001001000001001000000001000 \\
00000010000010010000010010010010000000000001 \\
00100000000100100000100000100100100000000000 \\
00100100000000000100100100100000000100000000 \\
00000100100100000000100100000100000000100000 \\
00000000100000100100000100100100000000000100
\end{bmatrix}$$

FIGURE 5.8: Proposed H-matrix for 3-bit BEC Code with 32-bit Word Length

$$\begin{bmatrix}
1000000000001001001000000000001000000001000001001000001001001000001000000000000\\
1001000000000000000001001000001001000001001000001001000000001001000001000000000\\
0001001000001000000000000001001001001000001001000000001000001001000000001000000\\
0000001001000001000001000000000001001001000000000001001001001001000000000001000\\
0000000001000000001000001001000000001000001001001001000001000001000000000000001\\
0100000000000100100100000000000100000001000001001000001001001001000000000000000\\
0100100000000000000001001000001001000001001000001001000000001000001000000000000\\
0000100100000100000000000001001001001000001001000000001000001000000001000000000\\
0000000100100000100000100000000001001001000000000001001001001000000000000100000\\
0000000000100000000100000100100000000100000100100100100000010000000000000000100\\
0010000000000010010010000000000010000000100000100100000100100100100000000000000\\
0010010000000000000001001000001001000001001000001001000000001000001000000000000\\
0000010010000010000000000000001001001001000001001000000001000001000000001000000\\
0000000010010000010000010000000000001001001000000000001001001001000000000010000\\
0000000000010000000010000010010000000010000010010010010000010000000000000000010
\end{bmatrix}$$

FIGURE 5.9: Proposed H-matrix for 3-bit BEC Code with 64-bit Word Length

$$\begin{bmatrix}
1000000010001000100000000000\\
1000100000001000000010000000\\
0000100010001000000000001000\\
0100000001000100010000000000\\
0100010000000100000001000000\\
0000010001000100000000000100\\
0010000000100010001000000000\\
0010001000000010000000100000\\
0000001000100010000000000010\\
0001000000010001000100000000\\
0001000100000001000000010000\\
0000000100010001000000000001
\end{bmatrix}$$

FIGURE 5.10: Proposed 4-bit BEC H-matrix for k=16 bits

5.5.2 Design of Proposed t-bit BEC Codec

The encoder computes parity bits based on data bits and append them with the data bits to construct codewords. Whereas, the decoder takes the stored codewords from the memory as the input and produces corrected version of the data as the output. The encoder circuits for the proposed codecs are designed based on their respective H-matrices by employing XOR gates. The proposed decoding operation consists of following two steps.

$$\begin{bmatrix}
1000000000001000100000001000000010000000000000000\\
1000100000000000000010001000100000001000000000000\\
0000100010001000000000001000100000000000010000000\\
0000000010000000100010000000100000000000000001000\\
0100000000000100010000000100000001000000000000000\\
0100010000000000000010001000100000001000000000000\\
0000010001000100000000000100010000000000001000000\\
0000000001000000010001000000010000000000000000100\\
0010000000000010001000000010000000100000000000000\\
0010001000000000000001000100010000000100000000000\\
0000010001000100000000000100010000000000001000000\\
0000000001000100000001000000010000000000000000010\\
0001000000000001000100000001000000010000000000000\\
0001000100000000000000010001000100000001000000000\\
0000000100010001000000000001000100000000000010000\\
0000000000010000000100010000000100000000000000001
\end{bmatrix}$$

FIGURE 5.11: Proposed 4-bit BEC H-matrix for k=32 bits

$$\begin{bmatrix}
10000000000000001000100010000000000000001000000000010000000100010000000000000000\\
10001000000000000000000000010001000000010001000000000001000100000001000000000000\\
00001000100000001000000000000000000010001000100010001000000000000000000010000000000\\
00000000100010000000100000001000000000000000100010000000100010000000000000010000000\\
00000000000010000000000010000000100010000000000100010001000000000000000000001000\\
01000000000000000100010001000000000000000100000000001000000010001000000000000000\\
01000100000000000000000000010001000000010001000000000001000100000001000000000000\\
00000100010000000100000000000000000010001000100010001000000000000000000010000000000\\
00000000010001000000010000000100000000000000100010000000100010000000000000001000000\\
00000000000001000000000001000000010001000000000010001000100000000000000000000100\\
00100000000000000100010001000000000000000100000000001000000010001000000000000000\\
00100010000000000000000000000010001000000010001000000000001000100000001000000000000\\
00000100010000000100000000000000000010001000100010001000000000000000000010000000000\\
00000000010001000000010000000100000000000000100010000000100010000000000000001000000\\
00000000000001000000000001000000010001000000000010001000100000000000000000000100\\
00100000000000000100010001000000000000000100000000001000000010001000000000000000\\
00010000000000000010001000100000000000000100000000001000000010001000000000000000\\
00010001000000000000000000000010001000000010001000000000001000100000001000000000000\\
00000001000100000010000000000000000010001000100010001000000000000000000010000000000\\
00000000000100010000000100000001000000000000001000100000001000100000000000000100000\\
00010000000000000010001000100000000000000100000000001000000010001000000000000000\\
00010001000000000000000000000010001000000010001000000000001000100000001000000000000\\
00000001000100000010000000000000000010001000100010001000000000000000000010000000000\\
00000000000100010000000100000001000000000000001000100000001000100000000000000100000\\
00000000000001000000000001000000010001000000000010001000100000000000000000000100
\end{bmatrix}$$

FIGURE 5.12: Proposed 4-bit BEC H-matrix for k=64 bits

1. Calculate the syndrome (S) using equation (2.3).
2. Correct the data column for which any consecutive ($\frac{n-k}{t}$) bits match with the

respective positions of syndrome bits.

First step is conventional one where syndrome bits are computed. The main difference between existing and proposed decoding of t-bit BEC codes stands in the second step of proposed decoding scheme. In case of conventional decoding of t-bit BEC codes, the required numbers of error locating syndromes per data bit depends on t. Also the numbers of error locating syndromes increases exponentially when t is increased. This makes the ELD block of conventional t-bit BEC decoders more complex as more circuitry are needed to design these ELD blocks. Whereas, the proposed H-matrices for t-bit BEC codes have been constructed in such a manner that all the syndrome patterns for correcting t-bit burst errors are exactly similar to any $(\frac{n-k}{t})$ number of syndrome bits. So the benefits of proposed decoding for the correction of t-bit burst errors are as follows:

a) Instead of all the $(n-k)$ numbers of syndrome bits, only $(\frac{n-k}{t})$ numbers of syndrome bits are employed to locate error in each data bits. This requires lesser number of AND and NOT gates in gate level design.

b) Instead of separating syndrome patterns for different burst errors, only one syndrome pattern is employed to locate error in each data bit position. This will save the necessity of numbers of OR gates in our proposed codec in gate level design.

c) Instead of exponential increases, the decoding complexity will gradually increases with t.

Gate level design of proposed codec for 3-bit BEC $(28, 16)$ code has been shown in Fig. 5.13 with considering the H-matrix as indicated in Fig. 5.7. It can be noted that hardware complexity of ELD block of proposed 3-bit BEC $(28, 16)$ decoder for any data bit is much simplified compared to conventional decoding as shown in Fig. 2.3.

5.5.2.1 Theoretical Overheads of Proposed Codes

The theoretical overheads of proposed H-matrices have been calculated in terms of number of parity bits needed, total number of 1's in H-matrix and the highest

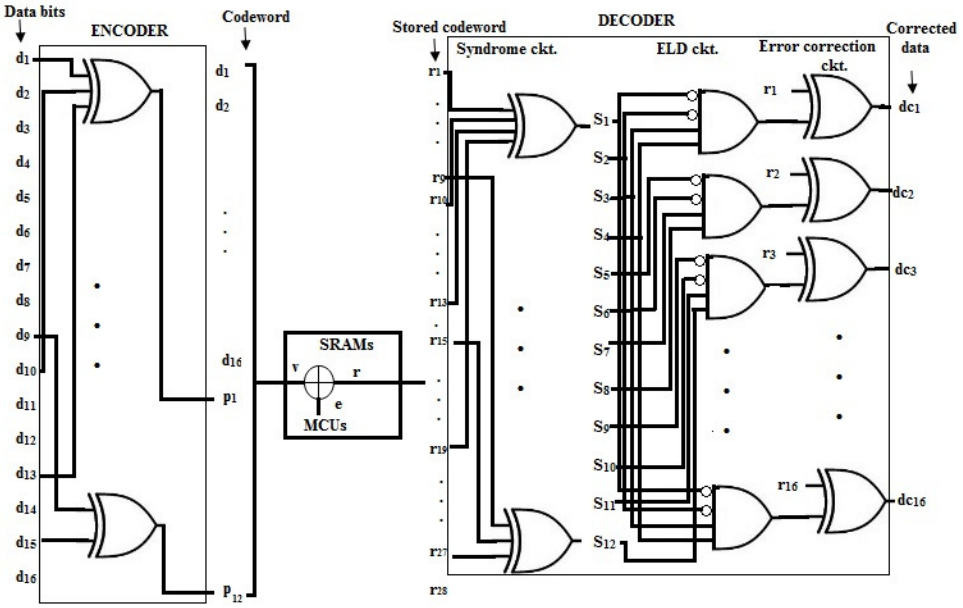

FIGURE 5.13: Gate Level Design of Proposed 3-bit BEC Codec

number of 1's in a row. Analysis of all these parameters are essential as mentioned in section 2.6.1 of chapter 2 and this has been presented in Table 5.7. The parity overhead of proposed 3-bit and 4-bit BEC codes are comparatively higher than the existing SEC-DAEC-TAEC [125], [164], 3-bit BEC [126] and 4-bit BEC [127] codes. The total number of 1's of proposed H-matrices are nearly compatible to the existing 3-bit and 4-bit BEC codes. But the proposed H-matrices have the smallest number of 1's in their heaviest rows compared to all the existing codes in Table 5.7.

Another popular way for estimating the overall theoretical area requirements of any ECC scheme is carried out by counting the total numbers of logic gates which are needed for constructing the codec. The overall theoretical area and delay estimations have been performed for the proposed and existing codecs which are compared in Table 5.8 and Table 5.9 respectively. From these tables, it has been observed that proposed 3-bit codecs require lesser number of equivalent 2-input NAND (NAND2) gates as compared to the related schemes in [125], [164], [126] for all word lengths. The similar trends have been observed for proposed 4-bit BEC codecs compared to existing 4-bit BEC codecs in [127] for different word lengths. Also the theoretical delay of proposed 3-bit BEC and 4-bit BEC codes are smaller with respect to the

TABLE 5.7: Comparison of Theoretical Overheads for Proposed Codes

k	Codes	No. of parity bits $(n-k)$	Total count of 1's in H-matrix	Highest count of 1's per row in H-matrix
16	**SEC-DAEC-TAEC [125]**	6	56	10
	SEC-DAEC-TAEC [164]	7	49	9
	3-bit BEC [126]	7	42	7
	4-bit BEC [127]	9	45	6
	Proposed 3-bit BEC	12	44	4
	Proposed 4-bit BEC	12	48	4
32	**SEC-DAEC-TAEC [125]**	7	93	14
	SEC-DAEC-TAEC [164]	8	85	12
	3-bit BEC [126]	8	76	10
	4-bit BEC [127]	9	108	14
	Proposed 3-bit BEC	12	92	8
	Proposed 4-bit BEC	16	88	6
64	**SEC-DAEC-TAEC [125]**	8	196	25
	SEC-DAEC-TAEC [164]	10	187	25
	3-bit BEC [126]	9	169	19
	4-bit BEC [127]	10	184	22
	Proposed 3-bit BEC	15	181	13
	Proposed 4-bit BEC	20	172	9

existing codes as shown in Table 5.9.

5.5.3 Synthesis Results

The proposed 3-bit and 4-bit BEC codecs have been simulated and synthesized on FPGA and ASIC platforms to obtain corresponding hardware requirements. These synthesis results have been presented in this section.

5.5.3.1 FPGA-based Synthesis Results

The FPGA based virtex 7 (xc7vx330t-2ffg1157) device has been chosen for simulation and synthesis of all the functional blocks of proposed and existing codecs. The FPGA based synthesis results have been presented in Table 5.10.

In Table 5.10, the LUTs and delay requirement of proposed 3-bit BEC codecs

TABLE 5.8: Area Comparison in terms of Logic Gates

k	Codecs	No. of 2-input Gates Required to Design Codec				No. of Equivalent NAND2 Gates
		XOR2	OR2	AND2	NOT	
16	**SEC-DAEC-TAEC [125]**	110	76	460	235	1823
	SEC-DAEC-TAEC [164]	103	54	277	164	1292
	3-bit BEC [126]	79	100	696	369	2377
	4-bit BEC [127]	79	280	2368	1106	6998
	Proposed 3-bit BEC	68	0	48	32	400
	Proposed 4-bit BEC	76	0	32	12	380
32	**SEC-DAEC-TAEC [125]**	197	156	1128	548	4060
	SEC-DAEC-TAEC [164]	200	116	652	363	2815
	3-bit BEC [126]	160	212	1708	920	5612
	4-bit BEC [127]	222	581	4904	2442	14881
	Proposed 3-bit BEC	180	0	96	48	960
	Proposed 4-bit BEC	160	0	96	56	888
64	**SEC-DAEC-TAEC [125]**	432	316	2660	1360	9356
	SEC-DAEC-TAEC [164]	476	254	1663	961	6953
	3-bit BEC [126]	375	436	4000	2294	13102
	4-bit BEC [127]	409	1191	11295	5929	33728
	Proposed 3-bit BEC	154	0	256	154	1282
	Proposed 4-bit BEC	348	0	256	168	2072

TABLE 5.9: Comparison of Critical Paths

k	Codecs	No. of Logic Gates Present in the Critical Path				No. of Equivalent NAND2 Gates
		XOR-2	OR-2	AND-2	NOT	
16	**SEC-DAEC-TAEC [125]**	18	5	30	14	161
	SEC-DAEC-TAEC [164]	18	4	19	11	133
	3-bit BEC [126]	13	7	48	24	193
	4-bit BEC [127]	10	19	160	72	489
	Proposed 3-bit BEC	5	0	3	2	28
	Proposed 4-bit BEC	6	0	2	1	29
32	**SEC-DAEC-TAEC [125]**	26	5	36	16	207
	SEC-DAEC-TAEC [164]	30	5	42	28	247
	3-bit BEC [126]	18	7	56	33	238
	4-bit BEC [127]	26	19	160	74	555
	Proposed 3-bit BEC	14	0	3	2	64
	Proposed 4-bit BEC	10	0	3	2	48
64	**SEC-DAEC-TAEC [125]**	37	5	25	13	306
	SEC-DAEC-TAEC [164]	30	5	42	28	247
	3-bit BEC [126]	36	7	64	41	334
	4-bit BEC [127]	42	19	180	89	674
	Proposed 3-bit BEC	24	0	4	3	107
	Proposed 4-bit BEC	16	0	4	2	74

for different word lengths have been compared with the existing SEC-DAEC-TAEC [125], [164] and 3-bit BEC [126] codes. The numbers of LUTs and delay requirements for proposed 3-bit BEC codecs are the lowest for all the word lengths compared to the existing SEC-DAEC-TAEC and 3-bit BEC codecs as shown in Table 5.10. The Table 5.10 also presents the comparison of FPGA based synthesis results for proposed and existing 4-bit BEC codecs. The number of LUTs and amount of delay which are needed to design the proposed 4-bit BEC codecs are lower than that of

TABLE 5.10: FPGA-based Synthesis Results

k	Codecs	Area (LUTs)	Delay (ns)	Improvement (%)	
				Area	Delay
16	**SEC-DAEC-TAEC [125]**	57	2.8	143	154
	SEC-DAEC-TAEC [164]	59	3.89	148	213
	3-bit BEC [126]	55	3.37	138	185
	4-bit BEC [127]	173	4.3	433	242
	Proposed 3-bit BEC	40	1.82	100	100
	Proposed 4-bit BEC	40	1.78	100	100
32	**SEC-DAEC-TAEC [125]**	165	4.41	217	193
	SEC-DAEC-TAEC [164]	155	4.62	204	202
	3-bit BEC [126]	160	5	211	218
	4-bit BEC [127]	356	4.93	556	254
	Proposed 3-bit BEC	76	2.29	100	100
	Proposed 4-bit BEC	64	1.94	100	100
64	**SEC-DAEC-TAEC [125]**	329	4.6	244	165
	SEC-DAEC-TAEC [164]	251	5.12	186	184
	3-bit BEC [126]	389	5.38	288	193
	4-bit BEC [127]	713	5.64	495	230
	Proposed 3-bit BEC	135	2.79	100	100
	Proposed 4-bit BEC	144	2.45	100	100

existing 4-bit BEC codecs [127] for all the word lengths.

5.5.3.2 ASIC-based Synthesis Results

The Cadence based Genus synthesis solution (TSMC18) tool has been employed for further synthesising the proposed 3-bit and 4-bit BEC codecs in ASIC platform. Table 5.11 compares the ASIC based synthesis results of proposed and existing codecs in terms of area, delay and power consumption. Conventionally, a SEC-DAEC- TAEC codec requires lesser number of syndrome patterns in its ELD block compared to a 3-bit or 4-bit BEC codec. Hence with conventional decoding scheme, codec design overheads in terms of area, delay and power of a SEC-DAEC-TAEC codec should be lesser compared to 3-bit and 4-bit BEC codecs. But the proposed codes for all the word lengths exhibit significantly lower codec design overheads not only against existing 3-bit and 4-bit BEC codecs but also against the existing SEC-DAEC-TAEC codecs as shown in Table 5.11. The area, delay and power consumption of proposed 3-bit BEC codecs are lowest compared to the existing SEC-DAEC-TAEC [125], [164]

TABLE 5.11: ASIC-based Synthesis Results

k	Codecs	Area (μm^2)	Delay (ps)	Power (mW)	Improvement (%)		
					Area	Delay	Power
16	**SEC-DAEC-TAEC [125]**	2927	499	0.54	145	145	256
	SEC-DAEC-TAEC [164]	2875	428	0.41	144	124	195
	3-bit BEC [126]	2931	450	0.43	145	131	204
	4-bit BEC [127]	4239	557	0.83	203	148	377
	Proposed 3-bit BEC [126]	2019	344	0.21	100	100	100
	Proposed 4-bit BEC [127]	2116	377	0.22	100	100	100
32	**SEC-DAEC-TAEC [125]**	5971	678	1.16	143	187	193
	SEC-DAEC-TAEC [164]	5785	515	1.51	138	142	252
	3-bit BEC [126]	6324	488	1.07	151	135	178
	4-bit BEC [127]	9478	728	1.94	219	182	298
	Proposed 3-bit BEC	4188	362	0.6	100	100	100
	Proposed 4-bit BEC	4328	401	0.65	100	100	100
64	**SEC-DAEC-TAEC [125]**	11922	877	3.04	138	217	204
	SEC-DAEC-TAEC [164]	9723	860	3.12	112	213	209
	3-bit BEC [126]	12727	688	2.61	147	170	175
	4-bit BEC [127]	19065	923	3.67	205	227	205
	Proposed 3-bit BEC	8629	404	1.49	100	100	100
	Proposed 4-bit BEC	9281	450	1.58	100	100	100

and 3-bit BEC [126] codecs for all the word lengths. The proposed 3-bit BEC codec with word length 32-bit exhibits the highest area improvement of 51% with respect to the existing 3-bit BEC codec for same word length. The highest power and delay improvements for proposed 3-bit BEC codec have been reported as 156% and 117% against existing SEC-DAEC-TAEC codecs [125] with word lengths 16-bit and 64-bit respectively. Also the codec design overheads for proposed 4-bit BEC code are much lesser compared to the existing conventional 4-bit BEC code [127] for all the word lengths as shown in Table 5.11.

The plots of area, delay and power of proposed and existing codecs versus the word lengths have been presented in Fig. 5.14, Fig. 5.15 and Fig. 5.16 respectively. Among the proposed and existing 3-bit BEC and SEC-DAEC-DAEC codecs, the highest area and delay overheads have been observed for existing 3-bit BEC codec by Li et al. and SEC-DAEC-TAEC codec by Adalid et al. respectively. Similarly the comparisons between proposed and existing 4-bit BEC codecs show that the existing scheme provides maximum overheads in terms of area, delay and power

consumption as indicated in Fig. 5.14 to Fig. 5.16. Thus, the area, delay and power overheads of proposed 4-bit BEC codec is much lesser compared to existing 4-bit BEC codec for all the word lengths.

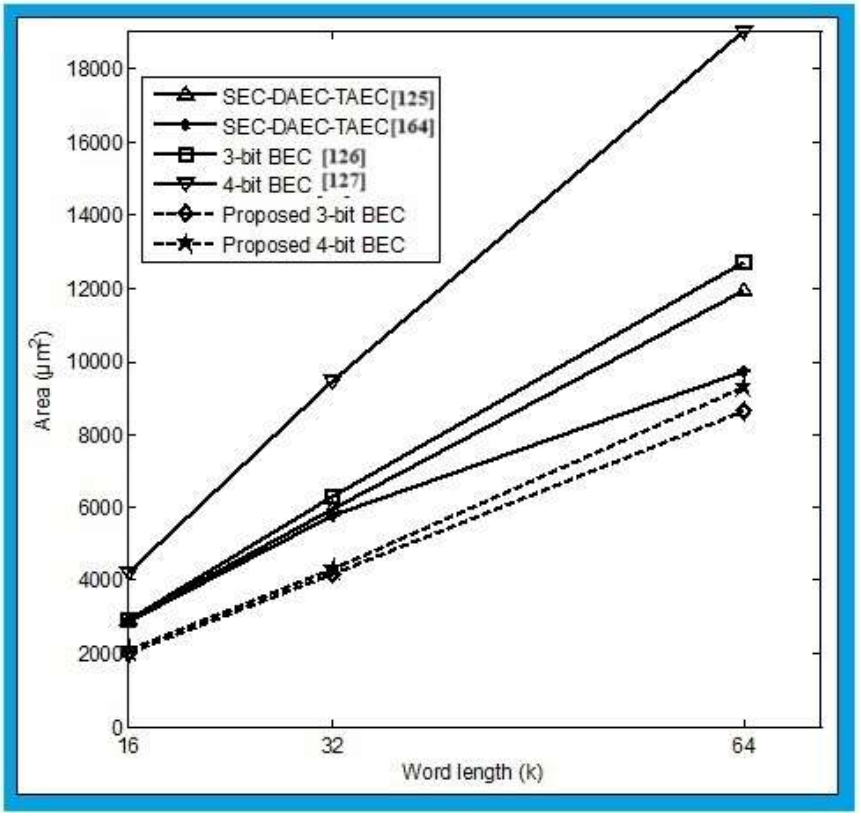

FIGURE 5.14: Area vs. Word Length

5.5.3.3 Evaluation of *ADPR*-metric for Proposed Codecs

It is difficult to present a global evaluation of the proposed codecs based on the synthesis results obtained from previous section. The most common metric for evaluating the overall performance of an ECC is known as *ADPR* metric which is

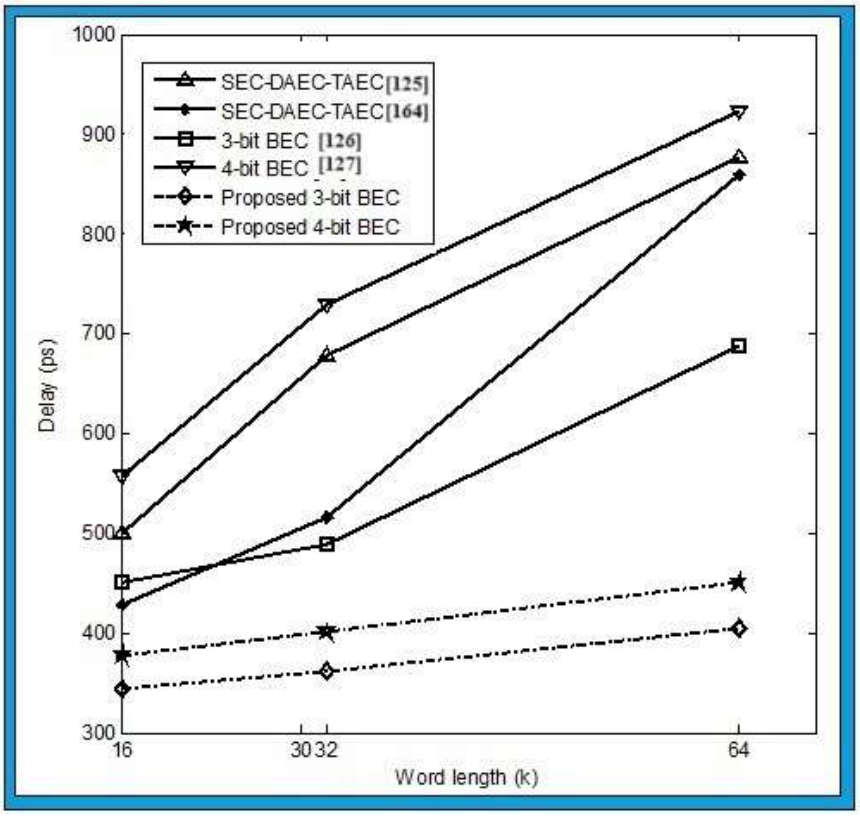

FIGURE 5.15: Delay vs. Word Length

defined in equation (3.7) The $ADPR$ metric of an ECC depends not only on the codec design parameters namely area, delay and power consumption but also on fractional redundancy (R). The desired values of all these parameters should be as low as possible. But this is difficult to achieve in practice. Therefore, an ECC scheme with a lower value of $ADPR$ metric is the preferred choice. The $ADPR$ metric for proposed and existing codecs has been computed using the values presented in Table 5.7 and Table 5.11, and these are summarized in Table 5.12. The highest improvement in $ADPR$-metric has been achieved by proposed 3-bit BEC codec against the corresponding SEC-DAEC-TAEC codec by Maity et al. [164]. Also from Table 5.12, it is observed that the proposed 3-bit BEC codecs have the lowest value of the $ADPR$-metric in spite of their reasonably higher value of fractional redundancy (R) compared to existing codecs. The proposed 4-bit BEC codec has lowest value of $ADPR$-metric compared to the existing 4-bit BEC codec. Hence the proposed

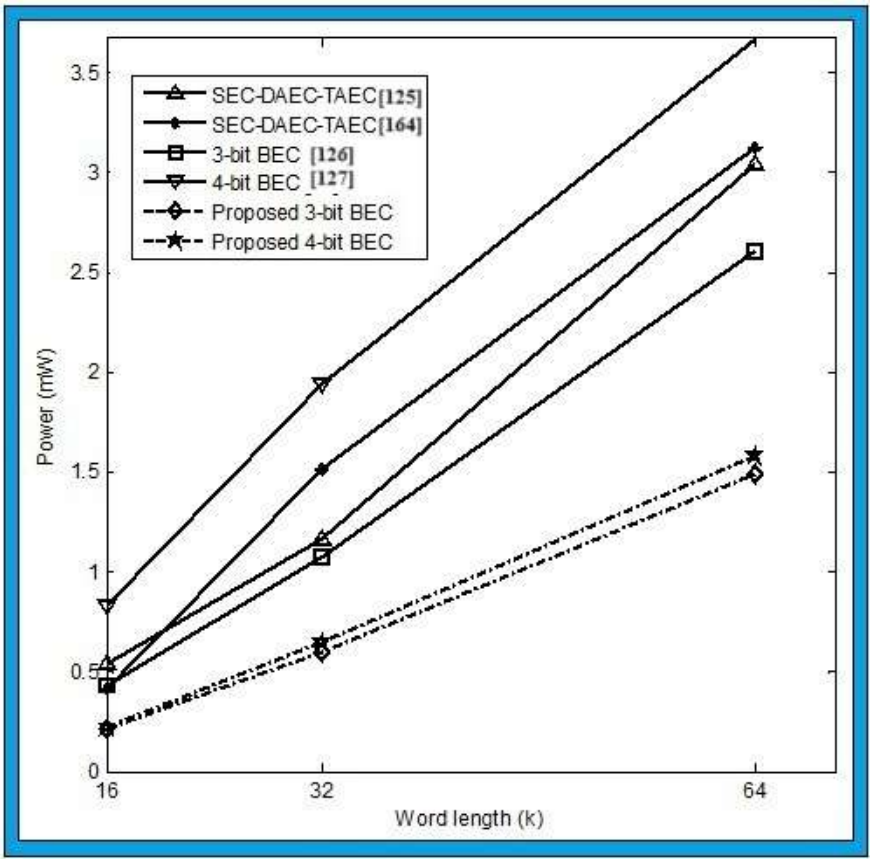

FIGURE 5.16: Power vs. Word Length

codec exhibits lower design overheads in the expense of increased redundancy. The $ADPR$-metric of proposed and existing codecs have been plotted against the word length in Fig. 5.17. The existing Adalid et al.'s SEC-DAEC-TAEC codec attains the highest value of $ADPR$-metric among all the codecs as shown in Fig. 5.17 but the $ADPR$-metric of proposed 3-bit BEC and 4-bit BEC codecs have lower values.

5.6 Conclusions

In this chapter, design of FPGA-based low delay 3-bit adjacent errors correcting codec has been presented first. This SEC-DED-DAEC-TAEC code is faster in speed and it has lower mis-correction rate. This work can be extended in future for the design of quadruple adjacent error correcting codes for memories with lower delay

TABLE 5.12: Comparison of *ADPR*-metric

k	Codecs	ADP	R	ADPR	Improvement (%)
16	**SEC-DAEC-TAEC [125]**	788709	0.375	295766	270
	SEC-DAEC-TAEC [164]	504505	0.438	220973	202
	3-bit BEC [126]	567149	0.438	248411	227
	4-bit BEC [127]	1959732	0.56	1097450	834
	Proposed 3-bit BEC	145853	0.75	109389	100
	Proposed 4-bit BEC	175501	0.75	131626	100
32	**SEC-DAEC-TAEC [125]**	4696072	0.219	1028440	301
	SEC-DAEC-TAEC [164]	4498705	0.25	1124676	330
	3-bit BEC [126]	3302140	0.25	825535	242
	4-bit BEC [127]	13385969	0.281	3761457	667
	Proposed 3-bit BEC	909634	0.375	341113	100
	Proposed 4-bit BEC	1128093	0.5	564047	100
64	**SEC-DAEC-TAEC [125]**	31785006	0.125	3973126	327
	SEC-DAEC-TAEC [164]	26088754	0.157	4095934	337
	3-bit BEC [126]	22853619	0.141	3222360	265
	4-bit BEC [127]	64580972	0.156	10074632	488
	Proposed 3-bit BEC	5194313	0.234	1215469	100
	Proposed 4-bit BEC	6598791	0.313	2065422	100

and mis-correction rate. The design of lower complexity error location detection block for existing SEC-DAEC-TAEC codes is also presented in this chapter. The ELD block of two existing TAEC decoders have been simplified by employing K-map based approach. The ASIC-based synthesis results provide the highest of 26.81% and 28.96% reduction in area and delay correspondingly. The K-map based simplification technique can be applied to existing burst error correcting decoders to simplify their ELD blocks.

Further, a new method for constructing the t-bit Burst Error Correcting (BEC) code has been presented in this chapter. This construction has been adopted mainly to reduce the decoding overheads of BEC code by increasing the redundancy. The proposed decoding scheme for t-bit BEC code requires lesser number of logic gates in it's gate level design. This is due to the fact that all burst errors patterns which are associated to a particular code bit has common syndrome bits for any $(\frac{n-k}{t})$ number of successive syndrome bits. The H-matrices of 3-bit and 4-bit BEC codes have been

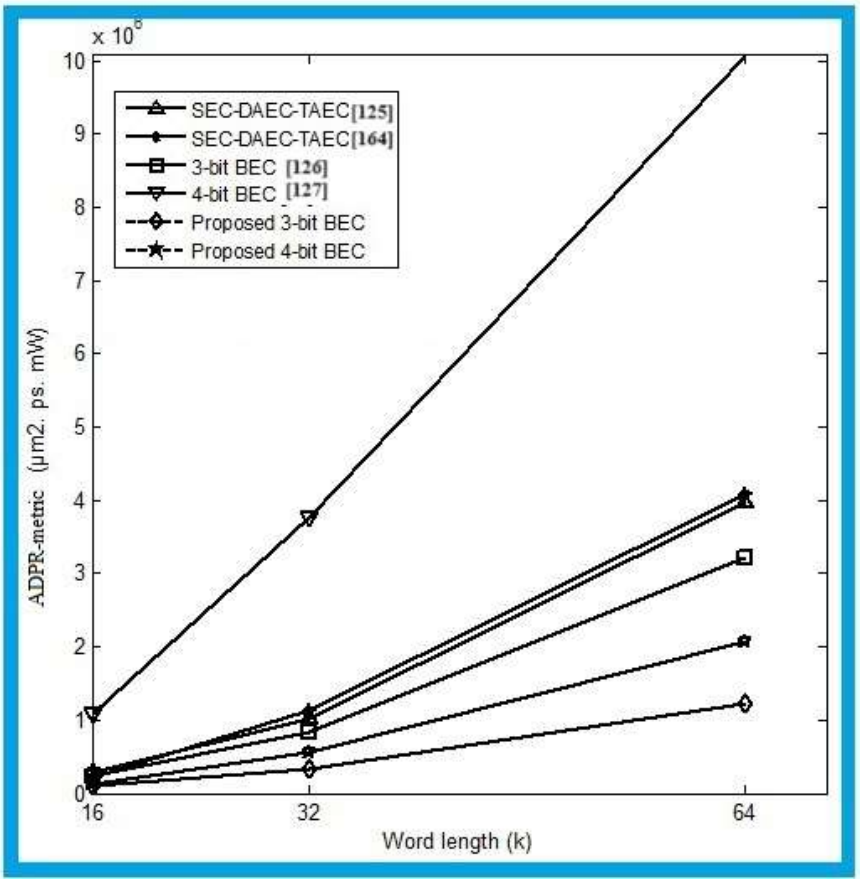

FIGURE 5.17: *ADPR* vs. Word Length

generated by employing the proposed technique as an illustration. Also codecs for these H-matrices have been designed and synthesized in FPGA and ASIC platforms. The performances of proposed 3-bit and 4-bit BEC codecs have been compared with the existing codecs of SEC-DAEC-TAEC, 3-bit BEC and 4-bit BEC codes. This comparison exhibits that proposed codecs require lower area, delay and power. Thus the proposed codecs can be employed to correct t-bit errors burst caused by MCUs with lower decoding overheads. This work an be extended in future for constructing double-bit and triple-bit random errors correcting codes with lower design overheads. The design and implementation of different schemes of adjacent and burst error correcting codes have been discussed so far in this thesis for storage systems with the objectives of lowering the design overhead and mis-correction rate. In the next

chapter, two new schemes of ECCs have been presented for Free Space Optical (FSO) communication systems with the aim of improving the error performance of these systems.

Chapter 6

Performance Analysis of Coded-FSO Communication Systems

6.1 Introduction

Reliability of communication systems is hugely degraded due to the channel noise induced errors in transmitted information sequences. The Error Correcting Codes (ECCs) are consistently applied in communication systems for ensuring system reliability by detecting and correcting errors in the received information sequences. In case of communication systems, the ECC encoder generates the coded information sequences by appending redundant information sequences with the original information sequences at the transmitter end. These coded information sequences are modulated with the help of suitable digital modulator and then transmitted through the communication channel where errors may occurred. At the receiver side of communication system, these erroneous received signals are demodulated to

form received sequences. The ECC decoder retrieves the original information sequences from the received information sequences.

Various ECC schemes like Hamming, Hsiao, LDPC, BCH, RS, Convolutiobnal, Turbo and Polar etc. have already been applied in communication systems for maintaining their reliability against channel noise. Moreover, the Soft Decision Decoding (SDD) of various ECC schemes have been explored for combining the demodulator and ECC decoder. The SDD of ECC schemes have been incorporated in modern communication systems for their better error performances. Also concatenation of two or more ECC schemes have a wide range of applications in communication systems due to their superior error performances. But both the SDD and concatenation of ECC schemes suffer from higher design complexities.

Further, utilization of unlicensed frequency spectrum, higher data rate and lower implementation cost make the Free Space Optical (FSO) communication systems attractive for the beyond 5G and upcoming communication standards. But these FSO communication systems are susceptible to Atmospheric Turbulence (AT) in the propagation path which cause degradation of FSO system performance. Various ECC schemes have already been employed in FSO systems for improving their error performance against different AT scenarios.

In this chapter, the Average Bit Error Rate (ABER) performance of FSO communication system has been improved with the SDD of Triple Adjacent Error Correction (TAEC) code. Also the combined ECC scheme with 3-bit Burst Error Correction (BEC) outer code and Single Error Correction- Double Adjacent Error Correction (SEC-DAEC) inner code have been introduced for application in FSO communication systems.

The rest of this chapter is structured as follows. In section 6.2, fundamentals of FSO communication system has been presented. Basics of SDD and combined ECC have been elaborated in section 6.3. In section 6.4, related coded-FSO systems have been discussed in brief. The improved ABER performance of FSO communication systems with the soft decision decoding of TAEC code has been presented in section

6.5. The combined ECC scheme with 3-bit BEC outer code and SEC-DAEC inner code for FSO communication systems have been illustrated in section 6.6. And finally, the conclusions are presented in section 6.7.

6.2 Fundamentals of FSO Communication System

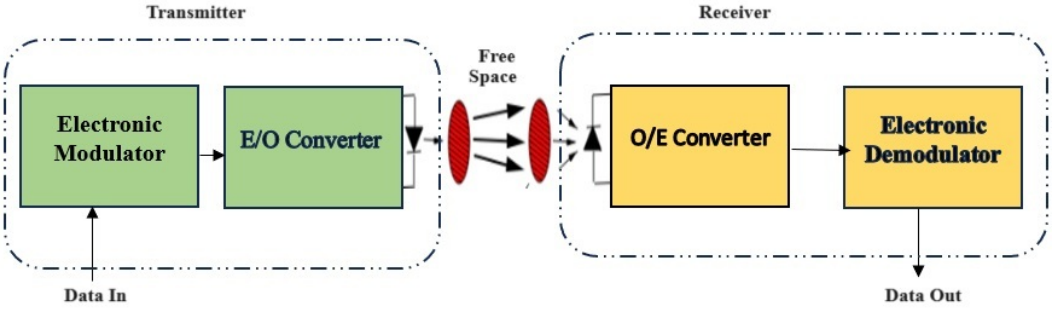

FIGURE 6.1: Block Diagram of a Typical SISO-FSO Communication System

The block diagram of a typical Single Input Single Output- Free Space Optical (SISO- FSO) communication system has been presented in Fig. 6.1. At the transmitter end, the data to be transmitted is converted to electrical signal by utilizing a suitable electronic modulator. The electrical to optical (E/O) converter provides optical signal as output with the help of optical modulator and optical source. The optical signal propagates through free space channel and affected by noise and AT. At receiver end, the received optical signal is applied to optical to electrical (O/E) converter which utilizes photo detector to produce electrical signal output. The electronic demodulator provides the estimation of transmitted data to the end users. The mathematical representation of the received optical signal can be presented as follows.

$$\mathrm{Y} = Xh + N \tag{6.1}$$

where, X is the transmitted signal, h is the propagation path coefficient, and N is the channel noise present in the propagation path. On the other hand, the Gamma-Gamma statistical distribution is usually considered to characterize the harsh channel conditions. The mathematical expression of considered statistical distribution scheme may represented as in [165].

$$f_I(I) = \frac{2\alpha_t\beta_t}{\Gamma(\alpha_t)\Gamma(\beta_t)} (I)^{\frac{\alpha_t+\beta_t}{2}-1} K_{\alpha_t-\beta_t}\left(2\sqrt{\alpha_t\beta_t I}\right) \tag{6.2}$$

where, the term $K_{\alpha_t-\beta_t}(\cdot)$ known as $(\alpha_t - \beta_t)^{th}$ order second kind modified Bessel function, and $\Gamma(\cdot)$ is the Gamma function. Atmospheric turbulence has been represented by the terms α_t and β_t, those are associated with the Rytov variance σ_R^2 as follows.

$$\begin{cases} \alpha_t = \left[\exp\left[\frac{0.49\sigma_R^2}{\left(1+1.11\sigma_R^{\frac{12}{5}}\right)^{\frac{7}{6}}}\right] - 1\right]^{-1} \\ \\ \beta_t = \left[\exp\left[\frac{0.51\sigma_R^2}{\left(1+0.69\sigma_R^{\frac{12}{5}}\right)^{\frac{5}{6}}}\right] - 1\right]^{-1} \end{cases} \tag{6.3}$$

Next, the above said statistical distribution in Eq.(6.2) can be expressed in terms of corresponding average SNR ($\bar{\gamma}$) and instantaneous SNR (γ) as follows.

$$f_\gamma(\gamma) = \frac{(\alpha_t\beta_t)^{\frac{(\alpha_t+\beta_t)}{2}}}{\Gamma(\alpha_t)\Gamma(\beta_t)\sqrt{\gamma\bar{\gamma}}} \left(\sqrt{\frac{\gamma}{\bar{\gamma}}}\right)^{\frac{(\alpha_t+\beta_t)}{2}} \mathrm{K}_{\alpha_t-\beta_t}\left(2\sqrt{\alpha_t\beta_t\sqrt{\frac{\gamma}{\bar{\gamma}}}}\right) \tag{6.4}$$

where $\gamma = \eta^2\mathrm{I}^2/\mathrm{N}_0$ and $\bar{\gamma} = \eta[\mathrm{E(I)}]^2/\mathrm{N}_0$, η denotes the optical-to-electrical conversion ratio. We can rewrite the above equation using the Meijer's G form of $\mathrm{K_v}(x)$ [166, Eq.(8.4.23.1)] as

$$\mathrm{f}_\gamma(\gamma) = \frac{(\alpha_t\beta_t)^{\frac{(\alpha_t+\beta_t)}{2}}}{2\Gamma(\alpha_t)\Gamma(\beta_t)(\bar{\gamma})^{\frac{(\alpha_t+\beta_t)}{4}}} (\gamma)^{\left(\frac{\alpha_t+\beta_t}{4}\right)-1} \mathrm{G}_{0\ 2}^{2\ 0}\left[\alpha_t\beta_t\sqrt{\frac{\gamma}{\bar{\gamma}}}\middle|_{\frac{\alpha_t-\beta_t}{2}, \frac{\beta_t-\alpha_t}{2}}^{-}\right] \tag{6.5}$$

TABLE 6.1: Values of α_t and β_t for Three Different AT Scenarios

AT Scenarios	**AT Parameters**	
	α_t	β_t
Weak AT	4.20	2.72
Moderate AT	3.99	1.65
Strong AT	2.20	0.65

where, $K_v(2\sqrt{x}) = \frac{1}{2}G_{0\ 2}^{2\ 0}\left[x\middle|_{\frac{v}{2},-\frac{v}{2}}^{-}\right]$, and $G[\cdot]$ is the Meijer's G function defined in [167, Eq.(9.301)]. Moreover, three distinct AT scenarios have been chosen as weak, moderate, and strong, depending on the severity of AT in the communication channel as listed in Table 6.1 to develop our numerical analysis. Generally, three set of AT parameters are considered [168], $\{\alpha_t, \beta_t\} = \{4.20, 2.72\}$ to indicate the weak AT condition, for moderate $\{\alpha_t, \beta_t\} = \{3.99, 1.65\}$, and $\{\alpha_t, \beta_t\} = \{2.20, 0.65\}$ for strong AT regime.

6.2.1 Average Bit Error Rate for Uncoded-FSO (U-FSO) Communication System

The ABER is an essential performance metric to claim the system's capabilities concerning the erroneous bit received at the receiver end during the entire data transmission period. The ratio of received erroneous bits to total transmitted bits is generally defines the ABER. Still, it is sometimes challenging to evaluate the accurate ABER of the system when the transmitted signal propagates through a noisy channel. In general, under noisy medium, the probability of error function ($P_e(\gamma)$) is considered for computing the analytical expression of ABER, and this error function intensely depends on the modulation technique. The probability of error function is associated with the normalized SNR, where SNR $(\gamma) = E_b/N_0$, E_b is the normalized energy per bit and N_0 is the noise power. The ABER of the system under fading environment is typically defined as

$$P_e = \int_0^\infty P_e(\gamma) f_\gamma(\gamma) d\gamma \tag{6.6}$$

where $f_\gamma(\cdot)$ is the PDF of the turbulence-induced fading channel as shown in Eq.(6.5). Next, the analytical derivation of the considered metric will be expressed under the uncoded-FSO (U-FSO) scenario. A standard Binary Phase Shift Keying (BPSK) modulation scheme has been assumed to evaluate the analytical derivation. With the help of Eq.(6.6), and the probability error function of BPSK under Additive White Gaussian Noise (AWGN) channel, the integral form of ABER under U-FSO communication system may be derived as

$$\mathrm{ABER}_{\mathrm{U-FSO}} = \int_0^\infty \mathrm{Q}(\sqrt{2\gamma})\mathrm{f}_\gamma(\gamma)\mathrm{d}\gamma. \tag{6.7}$$

Now, from the relationship between $\mathrm{Q}(\cdot)$ and $\mathrm{erfc}(\cdot)$ [169], and Eq.(6.6), above integral can be expressed as

$$\begin{aligned}\mathrm{ABER}_{\mathrm{U-FSO}} = \mathcal{K}\int_0^\infty (\gamma)^{\left(\frac{\alpha_\mathrm{t}+\beta_\mathrm{t}}{4}\right)-1}\,\mathrm{erfc}(\sqrt{\gamma})\\ \times \mathrm{G}_{0\ 2}^{2\ 0}\left[\alpha_\mathrm{t}\beta_\mathrm{t}\sqrt{\frac{\gamma}{\bar{\gamma}}}\Big|_{\frac{\alpha_\mathrm{t}-\beta_\mathrm{t}}{2},\frac{\beta_\mathrm{t}-\alpha_\mathrm{t}}{2}}^{-}\right]\mathrm{d}\gamma,\end{aligned} \tag{6.8}$$

where, $\mathcal{K} = \frac{(\alpha_\mathrm{t}\beta_\mathrm{t})^{\frac{(\alpha_\mathrm{t}+\beta_\mathrm{t})}{2}}}{4\Gamma(\alpha_\mathrm{t})\Gamma(\beta_\mathrm{t})(\bar{\gamma})^{\frac{(\alpha_\mathrm{t}+\beta_\mathrm{t})}{4}}}$. Next, by using the relationship between erfc $(\cdot)$ and the Meijer's G function [166, Eq.(8.4.14.2)] as, $\mathrm{erfc}(\sqrt{x}) = \frac{1}{\sqrt{\pi}}\mathrm{G}_{1\ 2}^{2\ 0}\left[x\big|_{0,\frac{1}{2}}^{1}\right]$; the above integral can be expressed as

$$\begin{aligned}\mathrm{ABER}_{\mathrm{U-FSO}} = \frac{\mathcal{K}}{\sqrt{\pi}}\int_0^\infty (\gamma)^{\left(\frac{\alpha_\mathrm{t}+\beta_\mathrm{t}}{4}\right)-1}\,\mathrm{G}_{1\ 2}^{2\ 0}\left[\gamma\Big|_{0,\frac{1}{2}}^{1}\right]\\ \times \mathrm{G}_{0\ 2}^{2\ 0}\left[\alpha_\mathrm{t}\beta_\mathrm{t}\sqrt{\frac{\gamma}{\bar{\gamma}}}\Big|_{\frac{\alpha_\mathrm{t}-\beta_\mathrm{t}}{2},\frac{\beta_\mathrm{t}-\alpha_\mathrm{t}}{2}}^{-}\right]\mathrm{d}\gamma.\end{aligned} \tag{6.9}$$

Finally, with the help of [170, Eq.(26)], the closed-from equation of the ABER under U-FSO communication system is expressed as

$$\mathrm{ABER}_{\mathrm{U-FSO}} = \frac{\mathcal{K}}{2\pi\sqrt{\pi}}\mathrm{G}_{2\ 5}^{4\ 2}\left[\frac{\alpha_\mathrm{t}^2\beta_\mathrm{t}^2}{16\bar{\gamma}}\Big|_{\frac{\alpha_\mathrm{t}-\beta_\mathrm{t}}{4},\frac{\alpha_\mathrm{t}-\beta_\mathrm{t}+2}{4},\frac{\beta_\mathrm{t}-\alpha_\mathrm{t}}{4},\frac{\beta_\mathrm{t}-\alpha_\mathrm{t}+2}{4},-\frac{\alpha_\mathrm{t}+\beta_\mathrm{t}}{4}}^{1-\frac{\alpha_\mathrm{t}+\beta_\mathrm{t}}{4},\frac{1}{2}-\frac{\alpha_\mathrm{t}+\beta_\mathrm{t}}{4}}\right] \tag{6.10}$$

6.3 Basics of Soft Decision Decoding and Concatenated ECC

In this section, the basics of soft decision decoding and combined (concatenated) ECCs have been presented in brief.

6.3.1 Basics of Soft Decision Decoding

The Soft Decision Decoding (SDD) consists of two fundamental optimal decoding schemes namely Maximum Likelihood (ML) decoding and Maximum A Posteriori (MAP) decoding [2]. In this section, the ML decoding scheme has been described in brief. The block diagram of ML decoding has been presented in Fig. 6.2.

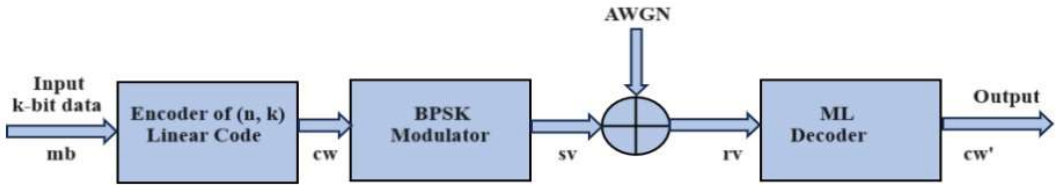

FIGURE 6.2: Block Diagram of ML Decoding

The encoder of an (n, k) binary linear code takes k-bit data (mb) as input and produces n-bit codewords (cw) at the output. These codewords are BPSK modulated with the transformation of 1 to -1 and 0 to +1. Thus symbol vectors (sv) are produced at the output of BPSK modulator and these symbol vectors are transmitted through the communication channel. The white Gaussian noise with zero mean and variance σ^2 is added with the symbol vector to form the received vectors (rv). The ML decoder is an optimal decoder which produces the estimation of transmitted codewords (cw') at the output in the following manners.

$$cw^{'} = arg.[max_{u\epsilon C} f(rv|cw = u)] \tag{6.11}$$

In equation (6.11), $f(rv|cw = u)$ is the conditional probability density function (pdf) of received vector (rv) provided that the codeword (cw) equals to u has been transmitted. Now,

$$f(rv|cw = u) = f(rv|sv = 1 - 2u) = f([rv_1 rv_2 \cdots rv_n]|[sv_1 sv_2 \cdots sv_n] \tag{6.12}$$

In equation (6.12), the symbol vectors $[sv_1\ sv_2\ ...\ sv_n]$ are known and thus the received vectors (rv) are conditionally independent. Hence the conditional probability density functions can be written as the product of individual pdf as expressed in equation (6.13).

$$f([rv_1 rv_2 \cdots rv_n]|[sv_1 sv_2 \cdots sv_n] = \prod_{i=1}^{n} f(rv_i|sv_i) \tag{6.13}$$

Now,

$$f(rv_i|sv_i) = \frac{1}{\sqrt{2\pi}\sigma} e^{\frac{-(rv_i - sv_i)^2}{2\sigma^2}} \tag{6.14}$$

Substituting equation (6.14) in equation (6.13),

$$f(rv|cw = u) = (\frac{1}{\sqrt{2\pi}\sigma})^n e^{-\frac{1}{2\sigma^2}\sum_{i=1}^{n}(rvi - sv_i)^2} \tag{6.15}$$

Now, maximization of left hand side of equation (6.15) equivalent to maximization of right hand side of the same equation. And this lead to minimization of exponential term in the right hand side of equation (6.15). The summation term in equation (6.15) is termed as the Euclidean distance between the vectors rv and sv and it is denoted by $||rv - sv||^2$. Thus,

$$max_{u\epsilon C} f(rv|cw = u) = min_{sv\epsilon 1-2u} ||rv - sv||^2 \tag{6.16}$$

The ML decoder compares the Euclidean distance between the received vectors (rv)

and all possible symbol vectors (sv) and the symbol vector which provides the highest Euclidean distance is treated as the estimation of the transmitted codeword.

6.3.2 Basics of Combined ECC

The combined or concatenated ECC is the series (cascade) connection of two or more ECCs and it is utilized to enhance the error performance of communication systems [2]. A simple concatenated ECC scheme with two different error correcting codes has been shown in Fig. 6.3. As shown in Fig. 6.3, the data bits to be transmitted are encoded by the encoder of the first ECC (outer encoder) and the output codewords of the this outer encoder are provided to the encoder of the second ECC (inner encoder). The codewords generated by the inner encoder are modulated and transmitted over the communication channel. At the receiver side of the communication systems, the inner decoder estimates the binary codewords as transmitted by the inner encoder. Generally, the inner decoder applies SDD for this purpose. The outer decoder which is based on Hard Decision Decoding (HDD) retrieves the original data bits. In this way, the concatenated ECC scheme utilizes the correction capabilities of both the inner and outer error correcting codes for better error correction coverage. Moreover, the use of interleaver and deinterleaver between two encoders and decoders respectively makes the communication systems more reliable. If the outer and inner ECC schemes of any combined ECC have the code dimension of (n_1, k_1) and (n_2, k_2) respectively. Then the overall code dimension of combined ECC scheme is given by (n, k)=$(n_1 n_2, k_1 k_2)$. Similarly, the overall code rate of the combined ECC scheme is R_c=$\frac{k_1.k_2}{n_1.n_2}$.

6.4 Existing Related Coded-FSO Systems

The FSO communication systems have been attracted the attention of researchers for the upcoming communication standards in recent years. This is due to the fact that

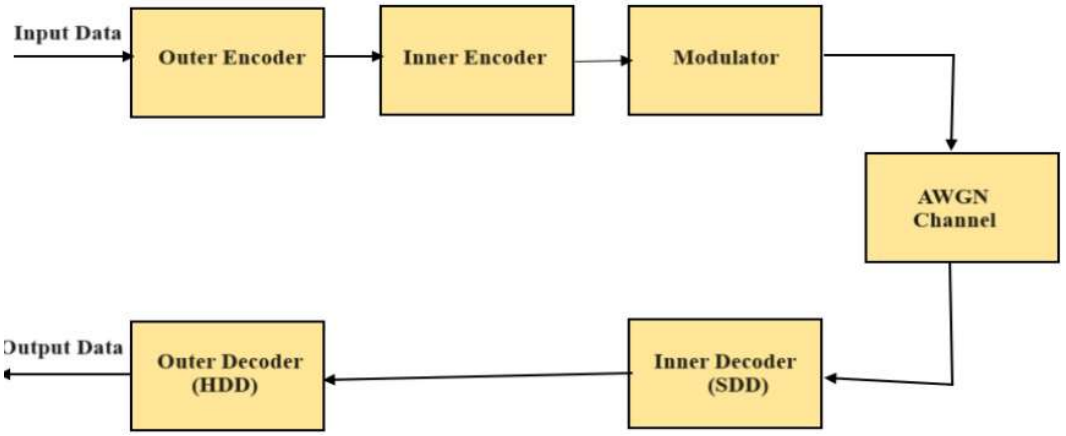

FIGURE 6.3: Block Diagram of Combined ECC Scheme

FSO communication links have enormous merits like unlicensed frequency spectrum utilization, higher speed of transmission and lower cost of implementation compared to RF communication links [171]. But FSO communication systems are susceptible to AT and performance of these systems are deteriorated as AT conditions degrade [172]-[173]. The Gamma-Gamma fading distribution [174] is typically employed in various FSO communication systems [165], [175]-[176] as it accurately describes the different AT conditions.

Varieties of ECCs have previously been investigated in FSO communication systems for improving their performances under different AT scenarios. The Pairwise Error Probability (PEP) of uncoded-FSO (U-FSO) and convolutional coded-FSO (C-FSO) systems have been analysed in [177]. Performance comparisons between the hard-decision decoded Reed- Solomon (RS) and soft-decision decoded Turbo codes for a range of atmospheric link conditions have been presented in [178]. The regular and irregular Low Density Parity Check (LDPC) C-FSO systems [179] have been analysed for the betterment of error performances. The performance of the FSO system model has been improved by applying RS, Turbo, and Convolutional coding schemes [180]. The improvement of capacity and error performance of LDPC and

BCH C-FSO systems have been reported in [181].

Further, various concatenated or combined ECC schemes have been presented for improving the error performance of C-FSO systems in different AT scenarios. The concatenation of LDPC and Vertical Bell Labs layered Space-Time (VBLAST) codes [182] have been investigated for improving the ABER performance of FSO communication systems. The combination of LDPC as outer and Trellis Coded Modulation (TCM) as inner code have been studied by Gupta et al. [183] for FSO communication systems. The concatenated C-FSO system with BCH and repetition codes has been presented in [184] with improved ABER performance under different AT scenarios.

6.5 Improved ABER Performance of FSO Communication Systems with Soft Decision Decoding of TAEC Code

In this section, the ABER improvement of FSO communication system with soft decision decoding of proposed TAEC code has been reported in details.

6.5.1 Proposed TAEC Code

The TAEC code as mentioned in chapter 2, is proficient of detecting and correcting single-bit error, double-bit and triple-bit adjacent errors. The H-matrix of proposed $(9, 3)$ TAEC code with code rate 1/3 and minimum distance (d_{min}) of 4 has been provided in Table 6.2.

In communication system, the TAEC encoder generates the codewords by combining data bits and parity bits as indicated in Fig. 6.4. These codewords are electronically modulated and transmitted over the communication channel. The noise affects

TABLE 6.2: Proposed H-matrix for $(9,3)$ TAEC Code

Parity bits						**Data bits**			**Parity equations**
p_1	p_2	p_3	p_4	p_5	p_6	d_1	d_2	d_3	
1	0	0	0	0	0	0	1	0	p_1=d_2
0	1	0	0	0	0	1	0	1	p_2=$d_1 \oplus d_3$
0	0	1	0	0	0	0	1	1	p_3=$d_2 \oplus d_3$
0	0	0	1	0	0	1	0	0	p_4=d_1
0	0	0	0	1	0	0	1	1	p_5=$d_2 \oplus d_3$
0	0	0	0	0	1	1	0	1	p_6=$d_1 \oplus d_3$

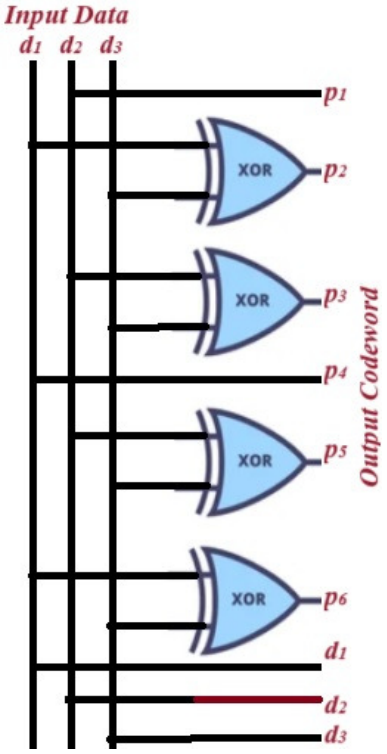

FIGURE 6.4: Proposed $(9,3)$ TAEC Encoder

the transmitted signal mostly in channel and noisy version of the transmitted signal is termed as received signal which arrives at the receiver. The SDD is the class of optimal decoding scheme frequently used in the receiver end of communication systems. The Maximum Likelihood (ML) decoder is the popular form of SDD scheme which is mostly utilized as the integral part of modern communication systems. The ML decoder takes the raw (soft) signals from the communication channel and directly estimates the transmitted data bits. The ML decoding scheme as mentioned

in section 6.3, has been applied in this work for SDD of proposed TAEC code. The SDD considers received signal blocks one by one and computes the Euclidean distance of these blocks with all the codewords of TAEC code. The codeword with maximum Euclidean distance is considered as the transmitted codeword.

6.5.2 Proposed TAEC Coded-FSO (C-FSO) System Model

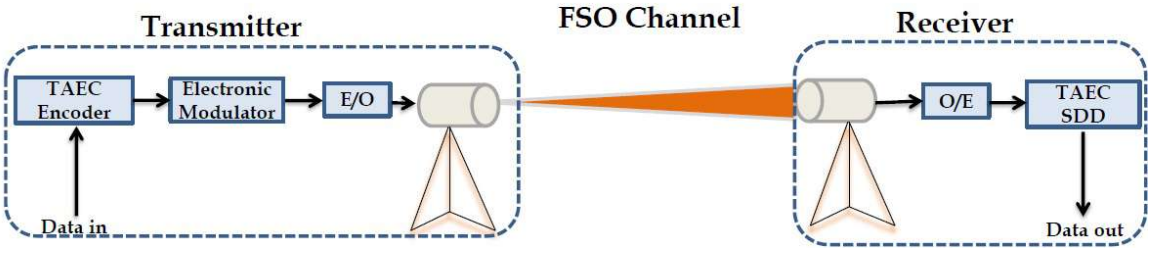

FIGURE 6.5: Proposed TAEC Coded-FSO Communication Architecture with Single Transmitting and Receiving Antenna.

In this section,proposed TAEC Coded-FSO (C-FSO) system model has been discussed. A typical FSO architecture consisting of single transmitter (T_x) and single receiver (R_x) has been considered as shown in Fig. 6.5. At the T_x terminal, raw data is encoded by proposed $(9, 3)$ TAEC encoder to produce the codewords. After that, these codewords are passed through the electronic modulator block to produce electronic modulated wave and then electrical to optical (E/O) converter generates the high-frequency optical modulated wave. The R_x terminal does the opposite process after receiving the optical wave by the photo detector. The output electrical signal of optical to electrical (O/E) converter is applied to SDD of proposed TAEC code to get the estimation actual transmitted data.

6.5.3 Analytical ABER Expression of Proposed TAEC C-FSO System

In this section, we have provided analytical derivation of considered performance metrics in terms of ABER for proposed TAEC C-FSO communication system. Considering the (R_c) as the code rate of proposed TAEC code, the probability of bit error (P_e) [2] of linear block coded communication system with BPSK modulation and ML-SDD can be written as

$$\mathrm{P_e} \simeq \frac{\mathrm{d_{min}}}{\mathrm{n}} \mathrm{A_{d_{min}}} Q\left(\sqrt{2\mathrm{d_{min}} R_c \gamma}\right) \tag{6.17}$$

where, $\mathrm{d_{min}}$ is known as minimum distance of the code, and $\mathrm{A_{d_{min}}}$ is the number of code words of the code with weight equal to $\mathrm{d_{min}}$. Finally, the ABER under C-FSO link can be approximated by utilizing the generalized BER formulation for linear block code with ML decoding expressed as [184]

$$\mathrm{ABER_{C-FSO}} \leq \sum_{\mathrm{i=d_{min}}}^{\mathrm{n}} \binom{\mathrm{n}}{\mathrm{i}} \mathrm{P_{ABER}^{i}} \left(1 - \mathrm{P_{ABER}}\right)^{\mathrm{n-i}} \tag{6.18}$$

where, $\mathrm{P_{ABER}}$ is expressed as

$$\begin{aligned} \mathrm{P_{ABER}} = {} & \frac{\mathcal{K}}{2\pi \mathrm{n}\sqrt{\pi}} \left(\frac{\mathrm{m}}{2}\right)^{-\frac{\alpha_t+\beta_t}{4}} \mathrm{A_{d_{min}}} \mathrm{d_{min}} \\ & \times \mathrm{G}_{2\ 5}^{4\ 2}\left[\frac{\alpha_t^2 \beta_t^2}{8\bar{\gamma}\mathrm{m}}\middle| \begin{matrix} 1-\frac{\alpha_t+\beta_t}{4}, \frac{1}{2}-\frac{\alpha_t+\beta_t}{4} \\ \frac{\alpha_t-\beta_t}{4}, \frac{\alpha_t-\beta_t+2}{4}, \frac{\beta_t-\alpha_t}{4}, \frac{\beta_t-\alpha_t+2}{4}, -\frac{\alpha_t+\beta_t}{4} \end{matrix}\right] \end{aligned} \tag{6.19}$$

where, $\mathrm{m} = 2\mathrm{d_{min}} R_c$.

6.5.4 Simulation Results and Discussions

In this section, ABER performance of proposed C-FSO, existing U-FSO [165] and $(3, 1)$ repetition C-FSO [184] systems have been provided for different AT scenarios

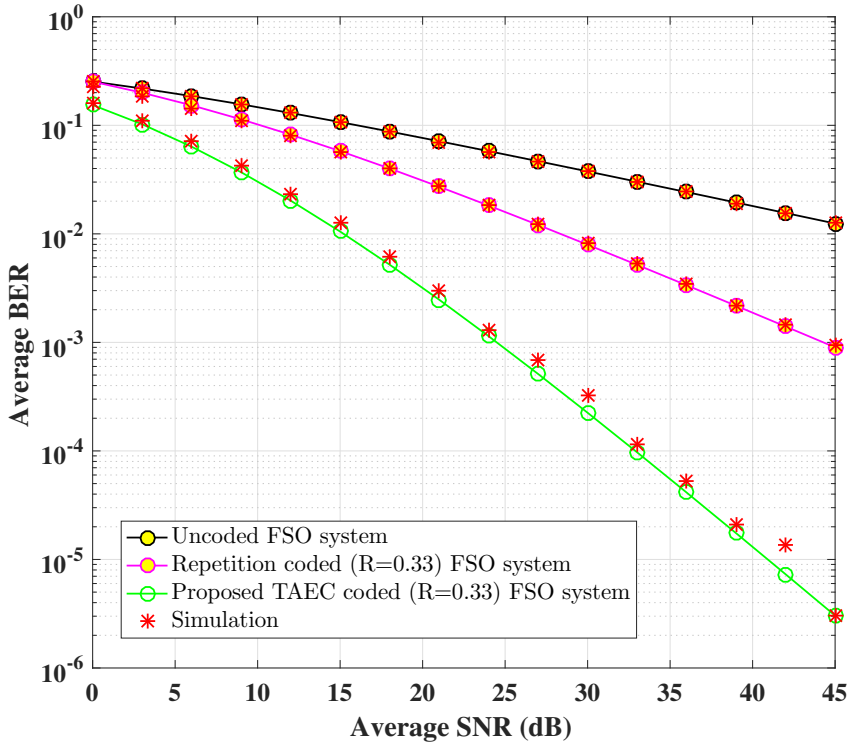

FIGURE 6.6: ABER Comparison for Uncoded and Coded FSO Communication Systems under Strong AT Condition with AT Parameters $\alpha_t = 2.20$ and $\beta_t = 0.65$

as mentioned in Table 6.1. The accuracy of the estimated results from the derived mathematical expression is validated through Monte Carlo simulation results. The ABER of U-FSO [165], existing repetition C-FSO [184] and proposed TAEC C-FSO systems have been presented in Fig. 6.6 under strong AT condition. The plot implies that both the C-FSO systems provide more noticeable outcomes than the U-FSO link. The U-FSO system delivers only 3.76×10^{-2} ABER at an average SNR of 30 dB. But the existing repetition C-FSO and proposed TAEC C-FSO systems provide the ABER values of 8.17×10^{-3} and 3.25×10^{-4} respectively at the same average SNR of 30 dB. Thus the ABER performance of proposed TAEC C-FSO system is 25.14 times better compared to Repetition C-FSO systems at 30 dB average SNR in strong AT regime. In Fig. 6.7, ABER of U-FSO [165], existing Repetition C-FSO [184] and proposed TAEC C-FSO systems have been presented under moderate AT condition. The ABER values of 2.18×10^{-3}, 7×10^{-5} and 3.33×10^{-8} have been observed in average SNR of 30 dB for U-FSO, Repetition C-FSO and proposed TAEC C-FSO

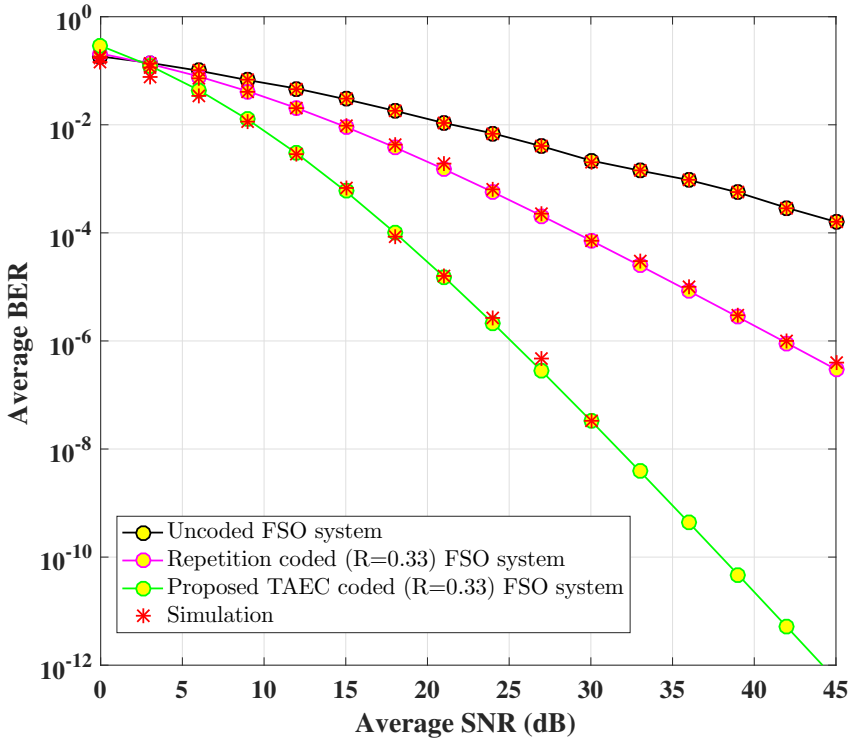

FIGURE 6.7: ABER Comparison for Uncoded and Coded FSO Communication Systems under Moderate AT condition with AT Parameters $\alpha_t = 3.99$ and $\beta_t = 1.65$

systems respectively. As a whole, ABER performance of proposed TAEC C-FSO system is better compared to the U-FSO and Repetition C-FSO systems for the entire average SNR as depleted in Fig. 6.7. For the weak AT condition, ABER of U-FSO [165], existing Repetition C-FSO [184] and proposed TAEC C-FSO systems have been plotted in Fig. 6.8. The ABER performance of proposed TAEC C-FSO system in weak AT condition is more improved compared to strong and moderate AT conditions. In summary, the ABER performance of proposed TAEC C-FSO system outperforms the subsisting U-FSO and Repetition C-FSO systems in all three AT regimes (strong, moderate and weak).

In Fig. 6.9.A to Fig. 6.9.C ABER improvement (%) of proposed TAEC C-FSO and existing Repetition C-FSO systems with respect to U-FSO system concerning average SNR under the strong, moderate and weak AT scenarios respectively have

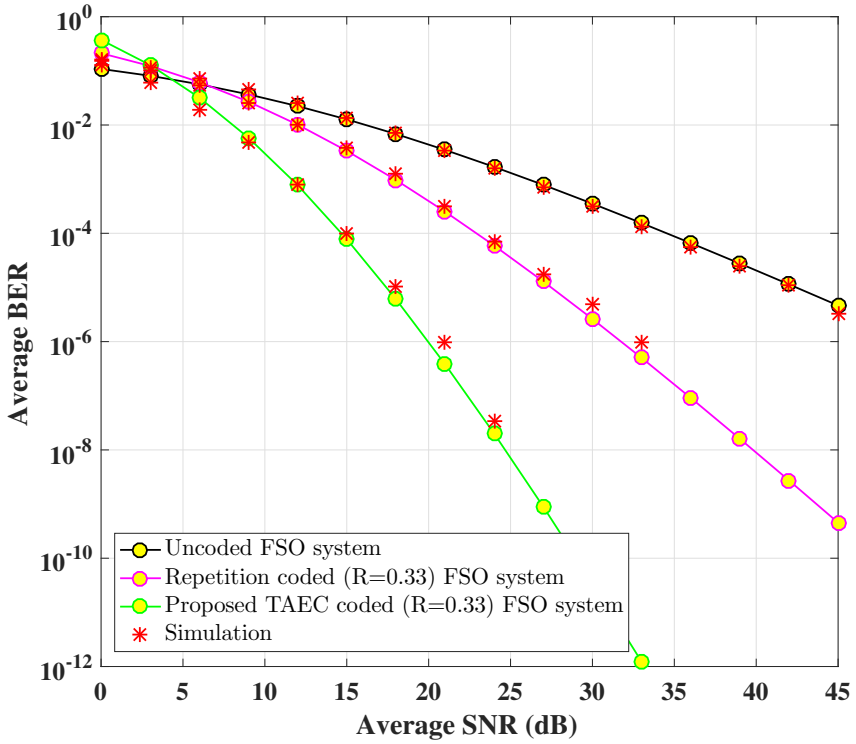

FIGURE 6.8: ABER Comparison for Uncoded and Coded FSO Communication Systems under Moderate AT Condition with AT Parameters $\alpha_t = 4.20$ and $\beta_t = 2.72$

been displayed. In strong AT condition, highest ABER improvements of 189.70% and 59.90% has been observed for proposed TAEC C-FSO and Repetition C-FSO systems correspondingly against U-FSO system. Similarly, the proposed TAEC C-FSO system exhibits highest ABER improvements of 223.20% and 228.30% against U-FSo system for moderate and weak AT conditions respectively. Thus ABER improvement of proposed TAEC C-FSO system increases as AT conditions change from strong to moderate to weak. Also the ABER improvements of proposed TAEC C-FSO systems are superior compared to the Repetition C-FSO systems for all the AT conditions under consideration. Therefore the proposed TAEC C-FSO system is the best candidate for obtaining huge ABER improvements under all three AT conditions when compared to U-FSO and Repetition C-FSO communication systems.

The coding gain of any coded communication system is the difference in SNR (dB) between uncoded and coded communication system to achieve a fixed target ABER

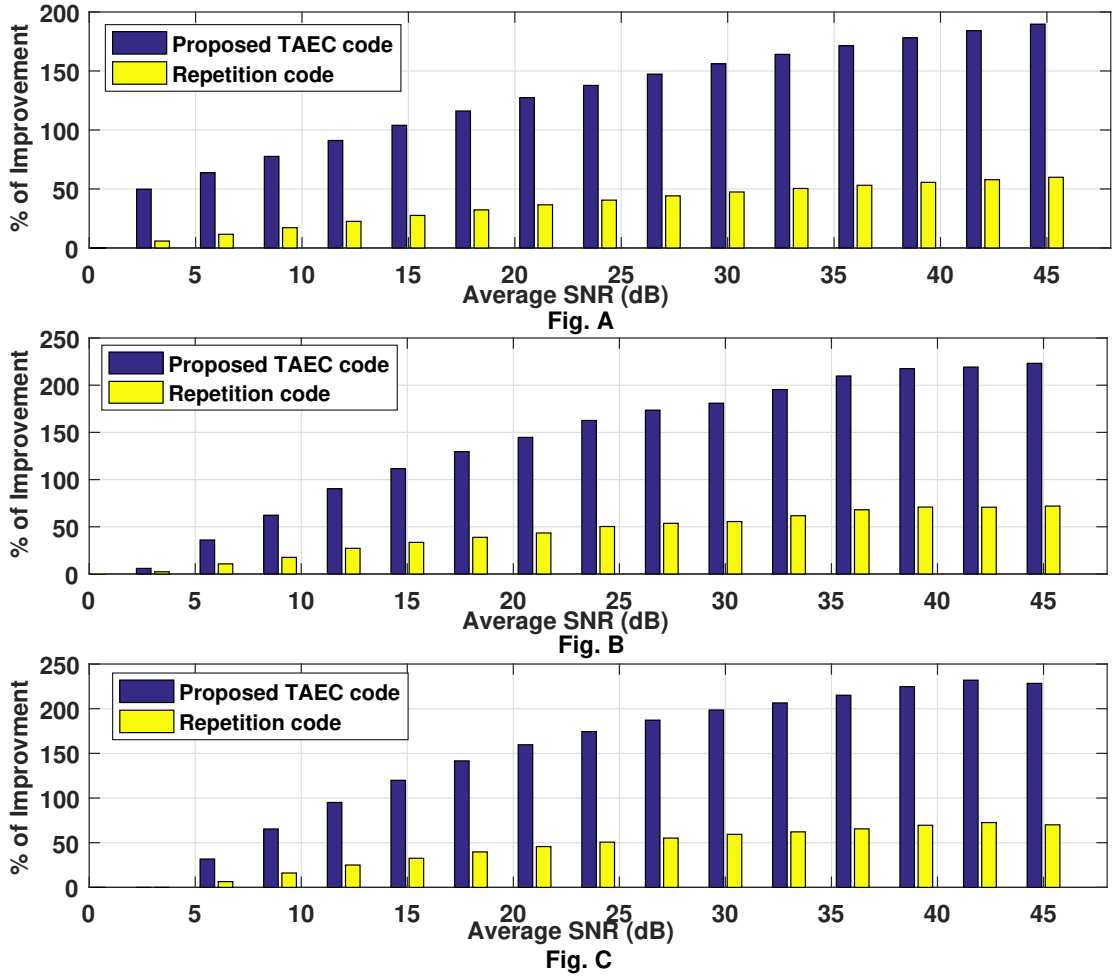

FIGURE 6.9: Improvement of ABER for C-FSO w.r.t. U-FSO Communication under Strong, Moderate and Weak AT Conditions

TABLE 6.3: Comparison of Coding Gain of Coded FSO Channel

Scheme	Coding Gain (dB)		
	Strong AT	Moderate AT	Weak AT
U-FSO System [165]	-	-	-
Repetition C-FSO System [184]	56	22.22	16.09
Proposed TAEC C-FSO System	89.19	37.22	26.96

value. An ABER of 10^{-5} is generally preferred as the target ABER for computing the coding gain of C-FSO systems. Coding gains of proposed TAEC C-FSO and existing Repetition C-FSO [184] have been computed with respect to U-FSO communication system [165] and compared in Table 6.3. Proposed TAEC C-FSO system achieves the highest coding gains with respect to U-FSO system for all three AT conditions as indicated in Table 6.3. Also a highest of 89.19 dB coding gain has been achieved by proposed TAEC C-FSO system at strong AT and this coding gain of proposed TAEC C-FSO system gradually reduces from moderate to weaker AT conditions. Further the coding gains of proposed TAEC C-FSO system are improved by 33.19dB, 15dB and 10.87dB in contrast with existing Repetition C-FSO system for strong, moderate and weak AT conditions respectively. Therefore the coding gain of proposed TAEC C-FSO system is better against existing Repetition C-FSO system as a whole.

6.6 The Combined ECC Scheme with Outer 3-bit BEC and Inner SEC-DAEC Codes for FSO Communication Systems

In this section, the proposed combined ECC scheme with outer 3-bit BEC and inner SEC-DAEC codes for FSO communication systems has been presented in depth.

6.6.1 Proposed 3-bit BEC, SEC-DAEC Codes and Their Combined ECC Scheme

The 3-bit burst error correcting (BEC) code as mentioned in chapter 2, is proficient of correcting any possibility of errors with in a burst length of 3-bit. The error possibilities inside the burst length of 3-bit are single-bit error, double-bit adjacent,

almost adjacent errors and triple-bit adjacent errors. On the other hand, the single error correction– double adjacent error correction (SEC-DAEC) code as mentioned in chapter 2, is proficient of correcting single-bit and double-bit adjacent errors. The H-matrices of proposed $(11, 5)$ 3-bit BEC and $(17, 11)$ SEC-DAEC codes have been presented in Fig.6.10 and Fig.6.11 respectively.

$$H = \begin{bmatrix} 10000010110 \\ 01000001010 \\ 00100011001 \\ 00010010101 \\ 00001001011 \\ 00000111001 \end{bmatrix}$$

FIGURE 6.10: Proposed H-matrix of $(11, 5)$ 3-bit BEC code

$$H = \begin{bmatrix} 10000000110010111 \\ 01000011101001110 \\ 00100010001110111 \\ 00010010010001010 \\ 00001011111010010 \\ 00000110100110101 \end{bmatrix}$$

FIGURE 6.11: Proposed H-matrix of $(17, 11)$ SEC-DAEC code

The proposed combined ECC scheme for FSO communication systems has been constituted by the concatenation of $(11, 5)$ 3-bit BEC and $(17, 11)$ SEC-DAEC codes as outer and inner codes respectively. In transmitter side of FSO communication system, two-stage encoding of input data bits are performed in combined ECC encoder block as indicated in Fig. 6.12. In the first stage of encoding, every 5-bit data block is converted to 11-bit codeword by outer $(11, 5)$ 3-bit BEC encoder. The 11-bit codeword at the output of outer encoder is provided as the input to the inner $(17, 11)$ SEC-DAEC encoder to produce 17-bit codeword.

At the receiver side of FSO communication system, the combined ECC decoder as shown in Fig. 6.13, provides the estimation of original transmitted data bits. The

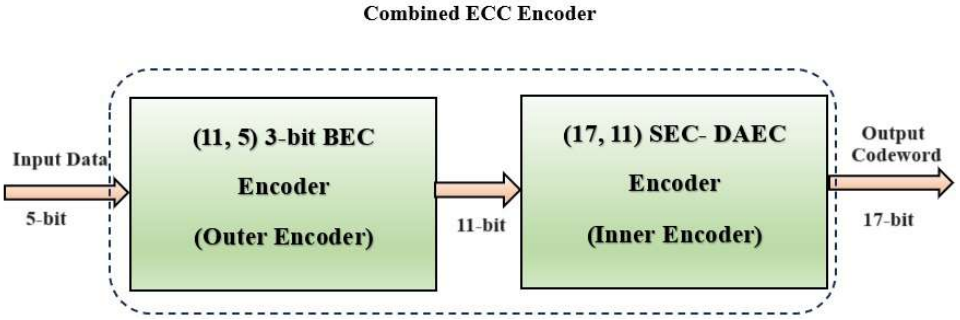

FIGURE 6.12: Proposed Combined ECC Encoder

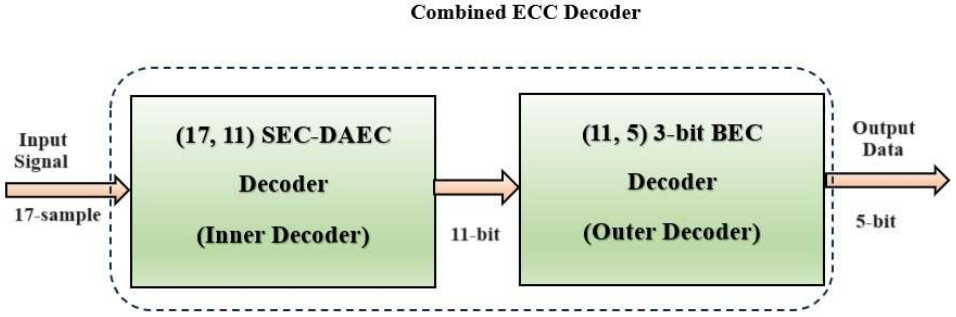

FIGURE 6.13: Proposed Combined ECC Decoder

inner $(17, 11)$ SEC-DAEC decoder accepts the electrical signal as input to produce 11-bit data output. This inner decoder employs Maximum Likelihood- Soft Decision Decoding (ML-SDD) for retrieving the binary data bits from received electrical signal. The outer 3-bit BEC decoder is a hard decision decoder (HDD) which produces the final estimation of original data bits. The proposed combined ECC scheme for FSO communication system has an overall code dimension of $(187, 55)$ and an overall code rate of 0.29.

6.6.2 Proposed C-FSO System Model

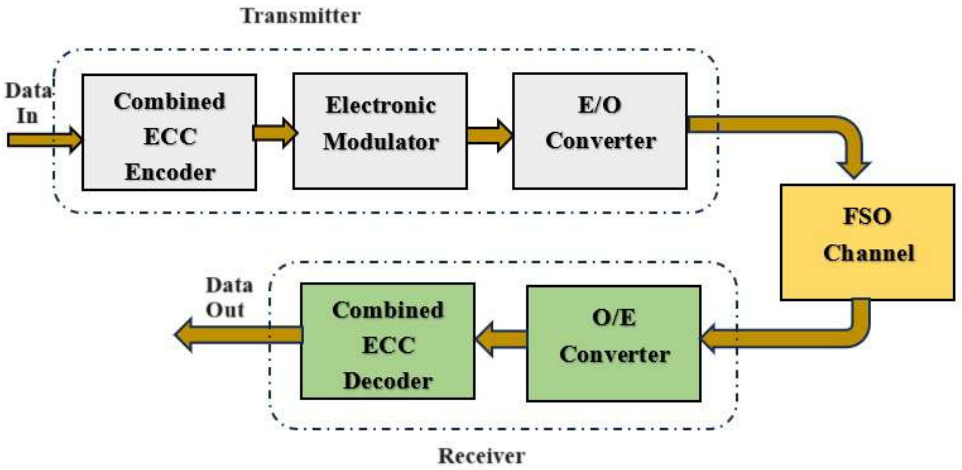

FIGURE 6.14: Proposed Combined ECC C-FSO Communication System Model

In this section, the proposed combined ECC C-FSO system model has been discussed. For this particular, architecture of proposed combined ECC C-FSO system model has been shown in Fig. 6.14. At the transmitter side, raw data is encoded by proposed combined ECC encoder prior to the data transmission to produce the codewords. After that, these codewords are passed through the electronic modulator block to produce electronic modulated wave and then alliance with electrical to optical converter to generate the high-frequency optical modulated wave. Before reaching the original information at the retailer user, the receiver terminal works the opposite process after receiving the optical wave by the photo detector. The electrical output signal is applied to the proposed combined ECC decoder for providing the fair estimations of transmitted data.

6.6.3 Analytical ABER Expression for Proposed C-FSO System

Here, the mathematical framework of the ABER under the proposed coded-FSO communication system has been produced based on our proposed system model. The

proposed 3-bit BEC code, SEC-DAEC code and combination of these two codes have been considered to carry out the mathematical formulation of the ABER. An (n, k) linear block code considers k-bit blocks of information to generate n-bit codewords. These n-bit codewords are transmitted one after another through the communication channel. The probability of error function of linear block coded BPSK modulation can be expressed by Marcum Q function written as [2]

$$\mathrm{P_e} \simeq Q\left(\sqrt{2R_c\gamma}\right) \tag{6.20}$$

Finally, the ABER under C-FSO link can be approximated by utilizing the generalized BER formulation for linear block code with hard decision decoding expressed as

$$\mathrm{ABER_{C-FSO}} \leq \sum_{\mathrm{i=t+1}}^{\mathrm{n}} \frac{\mathrm{i}}{\mathrm{n}} \binom{\mathrm{n}}{\mathrm{i}} \mathrm{P}_{\mathrm{ABER}}^{\mathrm{i}} \left(1 - \mathrm{P_{ABER}}\right)^{\mathrm{n-i}} \tag{6.21}$$

where, t, is the error correction capability of the code, and $\mathrm{P_{ABER}}$ of the above equation may derived as

$$\mathrm{P_{ABER}} = \frac{\mathcal{K}(\mathrm{R})^{-\frac{\alpha_t+\beta_t}{4}}}{2\pi\sqrt{\pi}} \mathrm{G}_{2\ 5}^{4\ 2}\left[\frac{\alpha_t^2\beta_t^2}{16\bar{\gamma}\mathrm{R}}\middle|_{\frac{\alpha_t-\beta_t}{4}, \frac{\alpha_t-\beta_t+2}{4}, \frac{\beta_t-\alpha_t}{4}, \frac{\beta_t-\alpha_t+2}{4}, -\frac{\alpha_t+\beta_t}{4}}^{1-\frac{\alpha_t+\beta_t}{4}, \frac{1}{2}-\frac{\alpha_t+\beta_t}{4}}\right] \tag{6.22}$$

6.6.4 Simulation Results and Discussions

Based on the discussion of analytical derivation in section 6.6.3 for the considered performance metrics, some numerical results under different types of AT scenarios have been provided in the current section. In this work three set of AT parameters have been considered as listed in Table 6.1. On the other hand, we have taken proposed $(11, 5)$ 3-bit BEC code, $(17, 11)$ SEC-DAEC code and combination of these two codes to pursue the performance analysis of coded-FSO (C-FSO) communication systems. Several 2D graphical plots have been provided to discuss this investigation's

outcomes. The accuracy of the estimated results from the derived mathematical expression is validated through Monte Carlo simulation results.

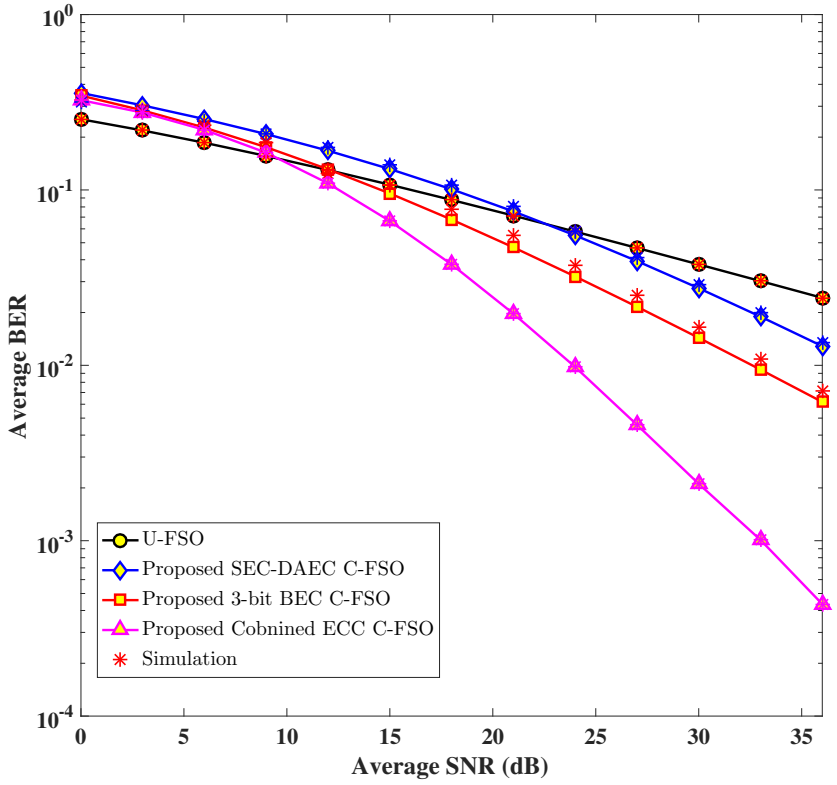

FIGURE 6.15: ABER vs. SNR Plots for Uncoded and Coded FSO Communication systems under Strong AT Condition

TABLE 6.4: Comparison of ABER Improvement for Proposed C-FSO Channel

Average SNR (dB)	ABER Improvement (%) at Strong AT			ABER Improvement (%) at Moderate AT			ABER Improvement (%) at Weak AT		
	Proposed SEC-DAEC C-FSO	Proposed 3-bit BEC C-FSO	Proposed Combined ECC C-FSO	Proposed SEC-DAEC C-FSO	Proposed 3-bit BEC C-FSO	Proposed Combined ECC C-FSO	Proposed SEC-DAEC C-FSO	Proposed 3-bit BEC C-FSO	Proposed Combined ECC C-FSO
0	-25.11	-22.58	-18.22	-40.82	-39.01	-17.35	-45.93	-44.38	-16.35
3	-21.61	-17.27	-15.40	-32.34	-27.17	-8.24	-34.62	-28.76	-2.05
6	-18.46	-11.80	-10.04	-24.48	-14.57	7.92	-23.64	-11.77	14.91
9	-15.46	-6.17	-2.31	-16.17	-1.82	27.88	-11.30	5.13	39.34
12	-12.42	-0.49	8.60	-7.05	10.24	49.89	2.10	20.36	64.16
15	-9.21	5.14	21.34	2.48	21.03	70.17	15.17	33.21	80.92
18	-5.76	10.59	34.79	11.77	30.33	88.48	26.87	43.65	104.95
21	-2.10	15.78	48.77	20.35	38.19	109.25	36.81	52.00	114.26
24	1.73	20.66	62.37	28.00	44.76	119.31	45.02	58.65	126.76
27	5.62	25.19	75.79	34.67	50.25	129.26	51.72	63.95	135.14
30	9.52	29.38	87.81	40.42	54.86	139.63	57.20	68.21	147.84
33	13.35	33.22	97.13	45.37	58.74	144.24	61.69	71.67	157.39
36	17.06	36.75	108.11	49.63	62.03	147.91	65.40	74.50	162.76

The ABER of U-FSO and three proposed C-FSO systems have been presented in Fig. 6.15 for strong AT conditions. The plot implies that the proposed coded-FSO

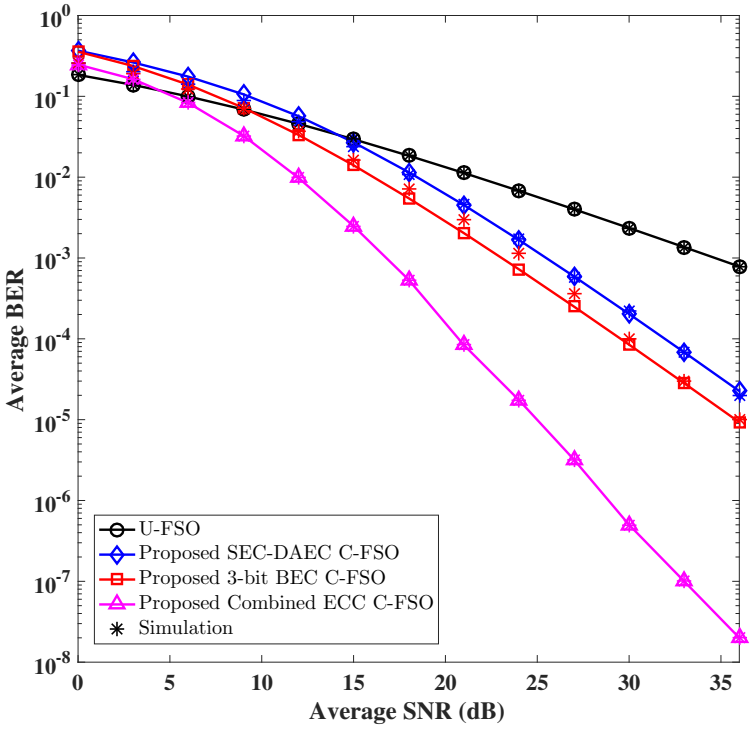

FIGURE 6.16: ABER vs. SNR Plots for Uncoded and Coded FSO Communication systems under Moderate AT Condition

systems provides more noticeable outcomes than the U-FSO link. It has been examined that at 30 dB average SNR, the U-FSO system delivers only 3.76×10^{-2} ABER; whereas by applying the same average SNR, the proposed SEC-DAEC, 3-bit BEC and combined ECC C-FSO systems provide an ABER of 2.75×10^{-2}, 1.66×10^{-2} and 2.11×10^{-3} respectively. Therefore the proposed SEC-DAEC and 3-Bit BEC C-FSO systems exhibit a slight improvements in ABER compared to U-FSO system for higher SNR region at strong AT condition. But the proposed combined ECC scheme provides a maximum of 55.75 times lower ABER compared to U-FSO system at 36 dB SNR. The ABER of U-FSO and three proposed C-FSO systems have been plotted in Fig. 6.16 for moderate AT condition. The ABER performance of of proposed C-FSO systems are superior compared to the U-FSO system for higher ranges of SNR as shown in Fig. 6.16. Further, the proposed combined ECC C-FSO system exhibits the best ABER performance when compared to other C-FSO and

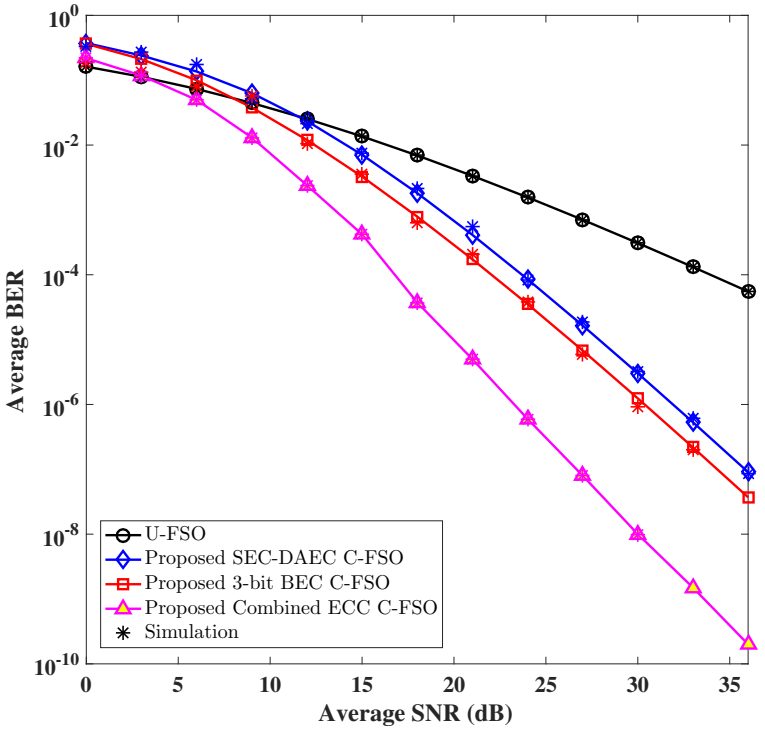

FIGURE 6.17: ABER vs. SNR Plots for Uncoded and Coded FSO Communication systems under Weak AT Condition.

U-FSO systems in moderate AT condition. The ABER performances of U-FSO and three proposed C-FSO systems have been compared in Fig. 6.17 for weak AT condition. The U-FSO system exhibits only 3.08×10^{-4} ABER at an average SNR of 30 dB as shown in Fig. 6.17. Whereas, the proposed SEC-DAEC, 3-bit BEC and combined ECC C-FSO systems provide an ABER of 3.18×10^{-4}, 1.24×10^{-6} and 1×10^{-8} respectively by utilizing the same average SNR. Therefore, the ABER performances of three proposed C-FSO systems are superior compared to U-FSO system for all the AT regimes and the best ABER performance has been provided by the proposed combined ECC C-FSO system. Also the ABER improvements of all the proposed C-FSO systems increase as AT condition changes from strong to weak through moderate.

The ABER improvements percentage for proposed SEC-DAEC, 3-bit BEC and combined ECC C-FSO systems have been computed with respect to U-FSO system and

reported in Table 6.4. As listed in Table 6.4, the proposed SEC-DAEC and 3-bi BEC C-FSO systems exhibit ABER improvement at higher average SNR compared to U-FSO system at strong AT conditions. Whereas in at the same AT conditions, the proposed combined ECC C-FSO system provides an utmost ABER improvement of 108.11% at an average SNR of 36 dB. In moderate AT conditions, highest ABER improvements of 49.63%, 62.03% and 147.91% have been observed for proposed SEC-DAEC, 3-bit BEC and combined ECC C-FSO systems correspondingly. An utmost ABER improvement of 162.76% has been achieved for proposed combined ECC C-FSO system with respect to U-FSO system at the weak AT conditions. As a whole, all the proposed C-FSO systems provide better ABER improvements with increase in average SNR levels and as AT conditions change from strong to weak through moderate. But the proposed combined ECC C-FSO system has been observed to provide maximum ABER improvement percentages at all the AT regimes when compared to other two proposed C-FSO systems.

6.7 Conclusions

In this chapter, the improved ABER performance of FSO communication systems with soft decision decoding of TAEC code has been presented. The proposed $(9, 3)$ TAEC code has been decoded by applying Maximum Likelihood-Soft Decision Decoding (ML-SDD) method for improving the ABER performance of FSO communication systems. Moreover, a combined ECC scheme has also been suggested for FSO communication systems. The proposed combined ECC scheme has been constructed with outer 3-bit BEC and inner SEC-DAEC codes for the betterment of error performance of FSO communication systems. Both the works presented in this chapter can be extended by incorporating different diversity schemes to design the architecture of Multiple Input Single Output (MISO), Single Input Multiple Output

(SIMO), and Multiple Input Multiple Output (MIMO) FSO communication systems in future. An additional direction for this work is the application to the field of vehicular technology research.

Chapter 7

Summary and Future Scope

The research works in this thesis addresses the design and implementation of various adjacent and burst error correcting codes for storage systems with the aim of reducing the codec design parameters namely area, delay and power as well as mis-correction rate. Also in this thesis triple adjacent error correcting code and a combination of double adjacent and burst error correcting codes have also been introduced for improving the ABER performance of FSO communication channel. This chapter highlights the key contributions and provides a summary of the thesis. It also provides a few suggestions for extending the current work in future.

7.1 Summary

Design and efficient VLSI implementation of various adjacent and burst error correcting codes for storage and FSO communication systems have been introduced in this thesis. The main objectives of these implementations are the reduction of area, delay, power and mis-correction rate of adjacent and burst error correcting codes for storage systems. Moreover, soft decision decoding of TAEC code and a combined ECC scheme where 3-bit BEC outer code and SEC-DAEC inner code are employed

for improving the ABER performances of FSO communication channel.

An overview, a literature survey of related works, and a list of significant contributions are included in the first chapter. In chapter 2, fundamentals of adjacent and burst error correcting codes are discussed starting from the construction up to the evaluation of these codes.

Three new schemes for SEC-DAEC codes have been introduced in the first contributory chapter i.e. chapter 3 and these schemes have been designed with the aim of lowering the area, delay and power usages of SEC-DAEC decoders. In the first scheme, the improved SEC-DAEC codec provides the highest improvements of 27% in area, 43% in delay and 64% in power against Dutta et al., Neale et al. and Reviriego et al. schemes respectively. Moreover, the Area-Delay-Power-Redundancy (ADPR) product of the proposed SEC-DAEC code is the highest of 70.91% lower compared to Reviriego et al. scheme. The second scheme in this chapter presents the design of lower complexity ELD block for existing SEC-DAEC decoder using K-map based simplification technique. This scheme provides 2.20–26.81% reduction in area and 0.30–28.96% reduction in delay compared to existing related ASIC-based synthesis results. The third scheme in this chapter combines the parallel decoder and area optimized decoder for further reduction of decoding overheads in $(24, 12)$ extended Golay code based SEC-DAEC decoder.

In chapter 4, three different SEC-DED-DAEC codes have been introduced for protecting storage systems from MCUs. Reductions in decoding overheads and mis-correction rates have been the major objectives of these SEC-DED-DAEC codes. The first SEC-DED-DAEC code presented in this chapter has been constructed for reducing the mis-correction rate and codec design overheads namely area and power consumption. The mis-correction rate of proposed SDECC codes are up to 88.24% and 80.67% lesser with respect to related existing designs by Li et al. [124] for 2-random and 3-random bit errors respectively. It is observed that area and power consumption of the proposed codecs are reduced up to 8.93% and 21.47% compared to the related published results [133], [132]. The compact SEC-DED-DAEC code

for memory application has been presented as the second contribution of this chapter. Implementation of said code requires lower area and delay compared to related works. The efficient decoding scheme for SEC-DED-DAEC codes with odd column weight H-matrix have moderate area and delay overheads compared to other related decoders proposed by Neale et al. and Reviriego et al..

In chapter 5, design of FPGA-based low delay triple adjacent bits error correcting codec has been presented first. This TAEC code is faster and it has lower miscorrection rate compared to Neale et al. and Adalid et al. schemes with respect to 2-bit and 3-bit random errors. The design of lower complexity error location detection block for existing triple adjacent error correction codes is also presented in this chapter. The ELD block of two existing TAEC decoders have been simplified by employing K-map based approach. The ASIC-based synthesis results provide the highest of 26.81% and 28.96% reduction in area and delay, respectively. Further, a new method for constructing the t-bit Burst Error Correcting (BEC) code has been presented in this chapter. This construction has been adopted mainly to reduce the decoding overheads of BEC code by increasing the redundancy. The proposed decoding scheme of t-bit BEC code requires lesser number of logic gates for designing. The ASIC synthesis results show that the proposed 3-bit BEC codec has the highest reductions of 1.51, 2.17, and 2.56 times in area, delay, and power consumption respectively with respect to related existing SEC-DAEC-DAEC and 3-bit BEC codecs. Whereas, the highest reductions of 2.19 times in area, 2.27 times in delay and 2.98 times in power has been obtained by proposed 4-bit BEC codec against the Li et al. 4-bit BEC codec [127].

In chapter 6, an improved ABER performance of soft decision decoding based TAEC code for FSO communication channel has been presented. The proposed $(9, 3)$ TAEC code has been decoded by using Maximum Likelihood- Soft Decision Decoding (ML-SDD) method for improving the ABER performance of FSO communication channel. The ABER performance of proposed TAEC C-FSO channel is superior compared to U-FSO and repetition code based C-FSO channels for all three AT conditions (strong, moderate and weak). Also the proposed TAEC C-FSO channel provides

the highest coding gain of 89.19 dB in strong AT condition with respect to U-FSO channel. Moreover, a combined ECC scheme has also been suggested for FSO communication channel. The proposed combined ECC scheme has been constructed with 3-bit BEC outer code and SEC-DAEC inner code for the betterment of error performance of FSO communication channel. The highest ABER improvements of 108.11%, 147.91% and 162.76% have been achieved by the proposed combined ECC scheme with respect to U-FSO channels in strong, moderate and weak AT conditions respectively.

7.2 Future Scope

The future scopes of the works presented in this thesis are enumerated as follows.

- The design and implementation of adjacent and burst error correcting codes elaborated in this thesis for storage systems which can be extended further to enhance error correction capability, lower codec design complexity and miscorrection rate.

- Further, construction of modified H-matrix for adjacent and burst error correcting codes considering K-map grouping in advance for obtaining lower complexity ELD block is another scope of work.

- The works presented for FSO communication systems can be extended by incorporating different diversity schemes to design the architecture of Multiple Input Single Output (MISO), Single Input Multiple Output (SIMO), and Multiple Input Multiple Output (MIMO) FSO communication systems in future.

- Moreover, design and implementation of random 2-bit and 3-bit errors correcting codes for storage and communication systems is another future scope of work.

Bibliography

[1] R. E. Blahut, *Theory and practice of error control codes.* Addison-Wesley Reading, 1983.

[2] S. Lin and D. J. Costello, *Error control coding: fundamentals and applications.* Prentice-Hall, 1983.

[3] T. R. Rao and E. Fujiwara, *Error-control coding for computer systems.* Prentice-Hall, Inc., 1989.

[4] E. Fujiwara, *Code design for dependable systems: theory and practical applications.* John Wiley & Sons, 2006.

[5] R. C. Baumann, "Soft errors in advanced computer systems," *IEEE Design & test of Computers*, vol. 22, no. 3, pp. 258–266, Jun. 2005.

[6] R. C. Baumann, "Radiation-induced soft errors in advanced semiconductor technologies," *IEEE Transactions on Device and Materials Reliability*, vol. 5, no. 3, pp. 305–316, Sep. 2005.

[7] L. Lantz, "Soft errors induced by alpha particles," *IEEE Transactions on Reliability*, vol. 45, no. 2, pp. 174–179, Jun. 1996.

[8] T. C. May and M. H. Woods, "Alpha-particle-induced soft errors in dynamic memories," *IEEE Transactions on Electron Devices*, vol. 26, no. 1, pp. 2–9, Jan. 1979.

[9] H. Fuketa, M. Hashimoto, Y. Mitsuyama, and T. Onoye, "Alpha-particle-induced soft errors and multiple cell upsets in 65-nm 10T sub-threshold SRAM," in *2010 IEEE International Reliability Physics Symposium*. IEEE, May. 2010, pp. 213–217.

[10] H. Fuketa, M. Hashimoto, Y. Mitsuyama and T. Onoye, "Neutron-induced soft errors and multiple cell upsets in 65nm 10T subthreshold SRAM," *IEEE Transactions on Nuclear Science*, vol. 58, no. 4, pp. 2097–2102, Aug. 2011.

[11] V. Sharma, N. Gupta, A. P. Shah, S. K. Vishvakarma, and S. S. Chouhan, "A reliable, multi-bit error tolerant 11t SRAM memory design for wireless sensor nodes," *Analog Integrated Circuits and Signal Processing*, vol. 107, no. 2, pp. 339–352, May. 2021.

[12] A. Sachdeva and V. K. Tomar, "A multi-bit error upset immune 12T SRAM cell for 5G satellite communications," *Wireless Personal Communications*, vol. 120, no. 3, pp. 2201–2225, Oct. 2021.

[13] V. Sharma, J. E. Kim, Y.-J. Jo, Y. Chen, and T. T. Kim, "AND8T SRAM macro with improved linearity for multi-bit in-memory computing," in *2021 IEEE International Symposium on Circuits and Systems (ISCAS)*. IEEE, May. 2021, pp. 1–5.

[14] A. A. Sachdeva and V. K. Tomar, "Characterization of stable 12T SRAM with improved critical charge," *Journal of Circuits, Systems and Computers*, vol. 31, no. 02, p. 2250023, Jan. 2022.

[15] C. L. Chen and M. Y.Hsiao, "Error-correcting codes for semiconductor memory applications: A state-of-the-art review," *IBM Journal of Research and Development*, vol. 28, no. 2, pp. 124–134, Mar. 1984.

[16] R. W. Hamming, "Error detecting and error correcting codes," *The Bell system technical journal*, vol. 29, no. 2, pp. 147–160, Apr. 1950.

[17] N. Sloane and D. Whitehead, "New family of single-error correcting codes," *IEEE Transactions on Information Theory*, vol. 16, no. 6, pp. 717–719, Nov. 1970.

[18] M. Y. Hsiao, "A class of optimal minimum odd-weight-column SEC-DED codes," *IBM Journal of Research and Development*, vol. 14, no. 4, pp. 395–401, Jul. 1970.

[19] S. S. Liu, P. Reviriego, A. S. Macian, L. Xiao, and J. A. Maestro, "Odd-weight-column SEC-DED-TAED codes," *Electronics Letters*, vol. 52, no. 2, pp. 119–120, Jan. 2016.

[20] J. Gracia-Morn, L. J. Saiz-Adalid, D. Gil-Toms, J. C. Baraza, and P. J.Gil-Vicente, "A very fast Single Error Correction-Double Error Detection Code for short data words," in *Workshop on Innovation on Information and Communication Technologies (ITACA-WIICT 2016)*, Jan. 2016, pp. 151–159.

[21] J. Samanta, J. Bhaumik, and S. Barman, "Compact and power efficient SEC-DED codec for computer memory," *Microsystem Technologies*, vol. 27, pp. 359–368, Feb. 2021.

[22] R. C. Baumann, "The impact of technology scaling on soft error rate performance and limits to the efficacy of error correction," in *Digest. International Electron Devices Meeting,*. IEEE, Dec. 2002, pp. 329–332.

[23] N. Seifert, X. Zhu, and L. W. Massengill, "Impact of scaling on soft-error rates in commercial microprocessors," *IEEE Transactions on Nuclear Science*, vol. 49, no. 6, pp. 3100–3106, Dec. 2002.

[24] V. Chandra and R. Aitken, "Impact of technology and voltage scaling on the soft error susceptibility in nanoscale CMOS," in *2008 IEEE International Symposium on Defect and Fault Tolerance of VLSI Systems*. IEEE, Oct. 2008, pp. 114–122.

[25] E. Ibe, H. Taniguchi, Y. Yahagi, K. Shimbo, and T. Toba, "Impact of scaling on neutron-induced soft error in SRAMs from a 250nm to a 22nm design rule," *IEEE Transactions on Electron Devices*, vol. 57, no. 7, pp. 1527–1538, May. 2010.

[26] A. Dixit and A. Wood, "The impact of new technology on soft error rates," in *2011 International Reliability Physics Symposium*. IEEE, Apr. 2011, pp. 5B–4.

[27] C. Slayman, "Soft error trends and mitigation techniques in memory devices," in *Proceedings-Annual Reliability and Maintainability Symposium*. IEEE, Jan. 2011, pp. 1–5.

[28] B. D. Sierawski, R. A. Reed, M. H. Mendenhall, R. A. Weller, R. D. Schrimpf, S. Wen, R. Wong, N. Tam, and R. C. Baumann, "Effects of scaling on muon-induced soft errors," in *2011 International Reliability Physics Symposium*. IEEE, Apr. 2011, pp. 3C–3.

[29] L. W. Massengill, B. L. Bhuva, W. T. Holman, M. L. Alles, and T. D. Loveless, "Technology scaling and soft error reliability," in *2012 IEEE International Reliability Physics Symposium (IRPS)*. IEEE, Apr. 2012, pp. 3C–1.

[30] S. Baeg, S. Wen, and R. Wong, "SRAM interleaving distance selection with a soft error failure model," *IEEE Transactions on Nuclear Science*, vol. 56, no. 4, pp. 2111–2118, Aug. 2009.

[31] P. Reviriego, J. A. Maestro, S. Baeg, S. Wen, and R. Wong, "Protection of memories suffering mcus through the selection of the optimal interleaving distance," *IEEE Transactions on Nuclear Science*, vol. 57, no. 4, pp. 2124–2128, Aug. 2010.

[32] S. Venkataraman, R. Santos, A. Das, and A. Kumar, "A bit-interleaved embedded hamming scheme to correct single-bit and multi-bit upsets for SRAM-based FPGAs," in *2014 24th International Conference on Field Programmable Logic and Applications (FPL)*. IEEE, Sep. 2014, pp. 1–4.

[33] M. Maniatakos, M. K. Michael, and Y. Makris, "Multiple-bit upset protection in microprocessor memory arrays using vulnerability-based parity optimization and interleaving," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 23, no. 11, pp. 2447–2460, Nov. 2014.

[34] K. Pagiamtzis and A. Sheikholeslami, "Content-addressable memory (CAM) circuits and architectures: A tutorial and survey," *IEEE journal of Solid-State Circuits*, vol. 41, no. 3, pp. 712–727, Feb. 2006.

[35] S. Baeg, S. Wen, and R. Wong, "Minimizing soft errors in TCAM devices: A probabilistic approach to determining scrubbing intervals," *IEEE Transactions on Circuits and Systems I*, vol. 57, no. 4, pp. 814–822, Jun. 2009.

[36] P. Reviriego and J. A. Maestro, "Efficient error detection codes for multiple-bit upset correction in SRAMs with BICS," *ACM Transactions on Design Automation of Electronic Systems (TODAES)*, vol. 14, no. 1, pp. 1–10, Jan. 2009.

[37] C. Argyrides, R. Chipana, F. Vargas, and D. K. Pradhan, "Reliability analysis of H-tree random access memories implemented with built in current sensors and parity codes for multiple bit upset correction," *IEEE Transactions on Reliability*, vol. 60, no. 3, pp. 528–537, Jul. 2011.

[38] A. Costas, D. K. Pradhan, and T. Kocak, "Matrix codes for reliable and cost efficient memory chips," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 19, no. 3, pp. 420–428, Dec. 2009.

[39] J. Guo, L. Xiao, Z. Mao, and O. Zhao, "Enhanced memory reliability against multiple cell upsets using decimal matrix code," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 22, no. 1, pp. 127–135, Mar. 2013.

[40] A. Ahilan and P. Deepa, "Modified decimal matrix codes in FPGA configuration memory for multiple bit upsets," in *International Conference on Computer Communication and Informatics (ICCCI)*. IEEE, Jan. 2015, pp. 1–5.

[41] S. Kamalakannan, S. Karthikeyan, and K. Sathyamoorthy, "Implementation of error correction technique based on decimal matrix code," *International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)*, vol. 2, no. 4, pp. 1–6, Apr. 2015.

[42] B. Anitha and B. Jeevidha, "Low overhead decimal matrix code with dynamic network on chip against multiple cell upsets," in *International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS)*. IEEE, Mar. 2015, pp. 1–6.

[43] N. K. Yedere and V. K. Pamula, "Performance analysis of decimal matrix code and modified decimal matrix code," in *2016 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC)*. IEEE, Dec. 2016, pp. 1–5.

[44] S. Kaneda and E. Fujiwara, "Single byte error correcting double byte error detecting codes for memory systems," *IEEE Transactions on Computers*, vol. 31, no. 07, pp. 596–602, Jul. 1982.

[45] G. Neuberger, F. G. de Lima, and R. Reis, "An automatic technique for optimizing reed-solomon codes to improve fault tolerance in memories," *IEEE Design & Test of Computers*, vol. 22, no. 1, pp. 50–58, Mar. 2005.

[46] S. Pontarelli, P. Reviriego, M. Ottavi, , and J. A. Maestro, "Low delay single symbol error correction codes based on reed solomon codes," *IEEE transactions on computers*, vol. 64, no. 5, pp. 1497–1501, May. 2014.

[47] J. Samanta, J. Bhaumik, and S. Barman, "Compact ca-based single byte error correcting codec," *IEEE Transactions on Computers*, vol. 67, no. 2, pp. 291–298, Aug. 2017.

[48] F. Sun, K. Rose, and T. Zhang, "On the use of strong BCH codes for improving multilevel NAND flash memory storage capacity," in *IEEE Workshop on Signal Processing Systems (SiPS): Design and Implementation.* Citeseer, Oct. 2006, pp. 1–8.

[49] P. Reviriego, C. Argyrides, and J. A. Maestro, "Efficient error detection in double error correction BCH codes for memory applications," *Microelectronics Reliability*, vol. 52, no. 7, pp. 1528–1530, Jul. 2012.

[50] S. Cho, D. Kim, J. Choi, and J. Ha, "Block-wise concatenated BCH codes for NAND flash memories," *IEEE Transactions on Communications*, vol. 62, no. 4, pp. 1164–1177, Feb. 2014.

[51] K. Namba, S. Pontarelli, M. Ottavi, and F. Lombard, "A single-bit and double-adjacent error correcting parallel decoder for multiple-bit error correcting BCH codes," *IEEE Transactions on Device and Materials Reliability*, vol. 14, no. 2, pp. 664–671, Mar. 2014.

[52] C. E. Shannon, "A mathematical theory of communication," *The Bell system Technical Journal*, vol. 27, no. 3, pp. 379–423, Jul. 1948.

[53] M. Y. Rhee, *Error-correcting coding theory.* McGraw-Hill, Inc., 1989.

[54] V. Pless, *Introduction to the theory of error-correcting codes.* John Wiley & Sons, 1998, vol. 48.

[55] S. Tong, D. Lin, A. Kavcic, B. Bai, and L. Ping, "On short forward error-correcting codes for wireless communication systems," in *2007 16th International Conference on Computer Communications and Networks.* IEEE, Aug. 2007, pp. 391–396.

[56] B. P. Smith, *Error-correcting codes for fibre-optic communication systems*. University of Toronto, 2011.

[57] G. C. C. Jr and J. B. Cain, *Error-correction coding for digital communications*. Springer Science & Business Media, 2013.

[58] S. A. Alabady, M. F. Salleh, and F. Al-Turjman, "A novel approach of error detection and correction for efficient energy in wireless networks," *Multimedia Tools and Applications*, vol. 78, pp. 1345–1373, Jan. 2019.

[59] G. A. Hussain and L. Audah, "RS codes for downlink LTE system over LTE-MIMO channel," *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 16, no. 6, pp. 2563–2569, Dec. 2018.

[60] M. Shirvanimoghaddam, M. S. Mohammadi, R. Abbas, A. Minja, C. Yue, B. Matuz, G. Han, Z. Lin, W. Liu, and Y. Li, "Short block-length codes for ultra-reliable low latency communications," *IEEE Communications Magazine*, vol. 57, no. 2, pp. 130–137, Dec. 2018.

[61] T. H. Liew and L. Hanzo, "Space-time codes and concatenated channel codes for wireless communications," *Proceedings of the IEEE*, vol. 90, no. 2, pp. 187–219, Feb, 2002.

[62] C. R. Berger, S. Zhou, Y. Wen, P. Willett, and K. Pattipati, "Optimizing joint erasure-and error-correction coding for wireless packet transmissions," *IEEE Transactions on Wireless Communications*, vol. 7, no. 11, pp. 4586–4595, Nov. 2008.

[63] R. M. Islam, "Error correction codes in wireless sensor network: An energy aware approach," *International Journal of Electronics and Communication Engineering*, vol. 4, no. 1, pp. 26–31, Jan. 2010.

[64] N. A. Alrajeh, U. Marwat, B. Shams, and S. S. Shah, "Error correcting codes in wireless sensor networks: an energy perspective," *Applied Mathematics & Information Sciences*, vol. 9, no. 2, p. 809, Mar. 2015.

[65] W. Wu, D. Haccoun, R. Peile, and Y. Hirata, "Coding for satellite communication," *IEEE Journal on Selected Areas in Communications*, vol. 5, no. 4, pp. 724–748, May. 1987.

[66] J. Cantillo, J. Lacan, and I. Buret, "A CRC usefulness assessment for adaptation layers in satellite systems," in *24th AIAA International Communications Satellite Systems Conference*, Jun. 2006, p. 5358.

[67] Y. Liu, Y. Guan, J. Zhang, G. Wang, and Y. Zhang, "Reed-Solomon codes for satellite communications," in *2009 IITA International Conference on Control, Automation and Systems Engineering (CASE 2009)*. IEEE, Jul. 2009, pp. 246–249.

[68] E. W. Ryan *et al.*, "An introduction to LDPC codes," *CRC Handbook for Coding and Signal Processing for Recording Systems*, vol. 5, no. 2, pp. 1–23, Sep. 2004.

[69] C. Zhang, X. Mu, J. Yuan, H. Li, and B. Bai, "Construction of multi-rate quasi-cyclic LDPC codes for satellite communications," *IEEE Transactions on Communications*, vol. 69, no. 11, pp. 7154–7166, Aug. 2021.

[70] B. Wang, P. Chen, Y. Fang, and F. C. Lau, "The design of vertical RS-CRC and LDPC code for ship-based satellite communications on-the-move," *IEEE Access*, vol. 7, pp. 44 977–44 986, Jan. 2019.

[71] K. Liu, M. Zhu, M. Jiang, C. Zhao, and X. Zhang, "Efficient decoding of parity-check concatenated RS codes," in *2022 IEEE 8th International Conference on Computer and Communications (ICCC)*. IEEE, Dec. 2022, pp. 1676–1681.

[72] F. Q. Kareem, S. R. M. Zeebaree, H. I. Dino, M. A. Sadeeq, N. Z. Rashid, D. A. Hasan, and K. H. Sharif, "A survey of optical fiber communications: challenges and processing time influences," *Asian Journal of Research in Computer Science*, vol. 7, no. 4, pp. 48–58, Apr. 2021.

[73] K. Lee, H. G. Kang, J. Park, and H. Lee, "100gb/s two-iteration concatenated BCH decoder architecture for optical communications," in *2010 IEEE Workshop On Signal Processing Systems*. IEEE, Oct. 2010, pp. 404–409.

[74] K. Lee and H. Lee, "A high-performance concatenated bch code and its hardware architecture for 100 Gb/s long-haul optical communications," in *2010 International SoC Design Conference*. IEEE, Nov. 2010, pp. 428–431.

[75] L. Lopacinski, J. Nolte, S. Buechner, M. Brzozowski, and R. Kraemer, "Improved Turbo product coding dedicated for 100 Gbps wireless terahertz communication," in *2016 IEEE 27th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*. IEEE, Sep. 2016, pp. 1–6.

[76] K. Sugihara, Y. Miyata, T. Sugihara, K. Kubo, H. Yoshida, W. Matsumoto, and T. Mizuochi, "A spatially-coupled type LDPC code with an NCG of 12 dB for optical transmission beyond 100 Gb/s," in *Optical Fiber Communication Conference*. Optical Society of America, Mar. 2013, pp. OM2B–4.

[77] B. P. Smith, A. Farhood, A. Hunt, F. R. Kschischang, and J. Lodge, "Staircase codes: Fec for 100 Gb/s OTN," *Journal of Lightwave Technology*, vol. 30, no. 1, pp. 110–117, Nov. 2011.

[78] M. L. Zhang and F. R. Kschischang, "Staircase codes with 6% to 33% overhead," *Journal of Lightwave Technology*, vol. 32, no. 10, pp. 1999–2002, May. 2014.

[79] I. B. Djordjevic, L. Xu, and T. Wang, "Multidimensional Turbo product and generalized LDPC codes with component rs codes suitable for use in beyond 100 Gb/s optical transmission," in *2009 9th International Conference on Telecommunication in Modern Satellite, Cable, and Broadcasting Services*. IEEE, Oct. 2009, pp. 402–405.

[80] Z. Wang, "Super-FEC codes for 40/100 Gbps networking," *IEEE Communications Letters*, vol. 16, no. 12, pp. 2056–2059, Nov. 2012.

[81] P. M. Shah, P. D. Vyavahare, and A. Jain, "Modern error correcting codes for 4G and beyond: Turbo codes and LDPC codes," in *Radio and Antenna Days of the Indian Ocean (RADIO)*. IEEE, Sep. 2015, pp. 1–2.

[82] R. F. Ormondroyd and J. J. Maxey, "Performance of low-rate orthogonal convolutional codes in DS-CDMA applications," *IEEE Transactions on Vehicular Technology*, vol. 46, no. 2, pp. 320–328, May. 1997.

[83] D. R. Cideciyan, E. Eleftheriou, and M. Rupf, "Concatenated Reed-Solomon/convolutional coding for data transmission in CDMA-based cellular systems," *IEEE Transactions on Communications*, vol. 45, no. 10, pp. 1291–1303, Oct. 1997.

[84] P. K. Frenger, P. Orten, T. Ottosson, and A. B. Svensson, "Rate-compatible convolutional codes for multirate DS-CDMA systems," *IEEE Transactions on Communications*, vol. 47, no. 6, pp. 828–836, Jun. 1999.

[85] D. J. Shah, V. K. Patel, and H. A. Patel, "Performance analysis of Turbo code for CDMA 2000 with convolutional coded IS-95 system in wireless communication system," in 2^{nd} *International Conference on Electronic Computer Technology*. IEEE, May. 2010, pp. 42–45.

[86] P. Zhang and P. Luukkanen, "Performance of Turbo codes in 3/sup rd/generation cdma2000 mobile system," in *IEEE* 49^{th} *Vehicular Technology Conference (Cat. No. 99CH36363)*, vol. 2. IEEE, May. 1999, pp. 1674–1677.

[87] Q. Li and N. S. Ramesh, "Channel coding performance in cdma2000 systems," in *IEEE Emerging Technologies Symposium on Broadband, Wireless Internet Access. Digest of Papers (Cat. No. 00EX414)*. IEEE, Apr. 2000, p. 5.

[88] S. H. Gupta and B. Virmani, "LDPC for Wi-Fi and WiMAX technologies," in *2009 international conference on emerging trends in electronic and photonic devices & systems*. IEEE, Dec. 2009, pp. 262–265.

[89] T. Brack, M. Alles, T. Lehnigk-Emden, F. Kienle, N. Wehn, N. E. L'Insalata, F. Rossi, M. Rovini, and L. Fanucci, "Low complexity LDPC code decoders for next generation standards," in *2007 Design, Automation & Test in Europe Conference & Exhibition*. IEEE, Apr. 2007, pp. 1–6.

[90] I. Tsatsaragkos and V. Paliouras, "A reconfigurable LDPC decoder optimized for 802.11 n/ac applications," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 26, no. 1, pp. 182–195, Sep. 2017.

[91] R. J. McEliece and L. Swanson, "Reed-Solomon codes and the exploration of the solar system," 1994.

[92] L. Chen, "Iterative soft decoding of Reed-Solomon convolutional concatenated codes," *IEEE Transactions on Communications*, vol. 61, no. 10, pp. 4076–4085, Sep. 2013.

[93] P. Adde, C. Jego, B. R. Le, and J. E. Chamorro, "Design and implementation of a soft-decision decoder for cortex codes," in *2010* 17^{th} *IEEE International Conference on Electronics, Circuits and Systems*. IEEE, Dec. 2010, pp. 663–666.

[94] B. Müller, M. Holters, and U. Zölzer, "Low complexity soft-input soft-output hamming decoder," in *2011* 50^{th} *FITCE Congress-" ICT: Bridging an Ever Shifting Digital Divide"*. IEEE, Aug. 2011, pp. 1–5.

[95] M. E. K. Alaoui, S. Nouh, and A. Marzak, "A low complexity soft decision decoder for linear block codes," *Procedia Computer Science*, vol. 127, pp. 284–292, Jan. 2018.

[96] C. Choi and J. Jeong, "Fast and scalable soft decision decoding of linear block codes," *IEEE Communications Letters*, vol. 23, no. 10, pp. 1753–1756, Jul. 2019.

[97] H. Jung, J. Kang, and J. Ha, "Quantum maximum likelihood decoding for linear block codes," in *2020 International Conference on Information and Communication Technology Convergence (ICTC)*. IEEE, Oct. 2020, pp. 227–232.

[98] C. Choi and J. Jeong, "Fast soft decision decoding of linear block codes using partial syndrome search," in *2020 IEEE International Symposium on Information Theory (ISIT)*. IEEE, Jun. 2020, pp. 384–388.

[99] B. Vucetic, V. Ponampalam, and J. Vuckovic, "Low complexity soft decision decoding algorithms for Reed-Solomon codes," *IEICE Transactions on Communications*, vol. 84, no. 3, pp. 392–399, Mar. 2001.

[100] S. Egorov, "A soft decoding for high rate Reed-Solomon codes," in *Optoelectronic Equipment and Devices in Systems of Pattern Recognization, Image and Symbol Information Processing. Recognition-2017*, May. 2017, pp. 29–36.

[101] J. Xing, L. Chen, and M. Bossert, "Progressive algebraic soft-decision decoding of Reed–Solomon codes using module minimization," *IEEE Transactions on Communications*, vol. 67, no. 11, pp. 7379–7391, Jul. 2019.

[102] J. Xing, L. Chen and M. Bossert, "Low-complexity chase decoding of Reed-Solomon codes using module," *IEEE Transactions on Communications*, vol. 68, no. 10, pp. 6012–6022, Jul. 2020.

[103] A. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," *IEEE Transactions on Information Theory*, vol. 13, no. 2, pp. 260–269, Apr. 1967.

[104] E. Arikan, "Systematic polar coding," *IEEE Communications Letters*, vol. 15, no. 8, pp. 860–862, Jun. 2011.

[105] E. Arika, "A performance comparison of polar codes and Reed-Muller codes," *IEEE Communications Letters*, vol. 12, no. 6, pp. 447–449, Jun. 2008.

[106] Z. B. K. Egilmez, L. Xiang, R. G. Maunder, and L. Hanzo, "The development, operation and performance of the 5G polar codes," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 1, pp. 96–122, Dec. 2019.

[107] S. Huilgol, "Channel coding techniques for 5G using polar codes," Ph.D. dissertation, 2017.

[108] O. Afisiadis, A. Balatsoukas-Stimming, and A. Burg, "A low-complexity improved successive cancellation decoder for polar codes," in *2014* 48^{th} *Asilomar Conference on Signals, Systems and Computers*. IEEE, Nov. 2014, pp. 2116–2120.

[109] T. Tal and A. Vardy, "List decoding of polar codes," *IEEE Transactions on Information Theory*, vol. 61, no. 5, pp. 2213–2226, Mar. 2015.

[110] P. Trifonov, "Efficient design and decoding of polar codes," *IEEE Transactions on Communications*, vol. 60, no. 11, pp. 3221–3227, Aug. 2012.

[111] J. Yuan, Z. Jiang, Y. Mao, and W. Ye, "Forward error correction concatenated code in DWDM systems," *Frontiers of Optoelectronics in China*, vol. 1, pp. 20–24, Sep. 2008.

[112] H. Mahdavifar, M. El-Khamy, J. Lee, and I. Kang, "Performance limits and practical decoding of interleaved Reed-Solomon polar concatenated codes," *IEEE Transactions on Communications*, vol. 62, no. 5, pp. 1406–1417, May. 2014.

[113] J. Zhao, W. Zhang, Y. Liu, J. Gao, and R. Zhang, "A rate-matching concatenation scheme of polar codes with outer Reed-Solomon codes," *IEEE Wireless Communications Letters*, vol. 10, no. 3, pp. 459–463, Oct. 2020.

[114] Y. Wang, K. R. Narayanan, and Y. C. Huang, "Interleaved concatenations of polar codes with BCH and convolutional codes," *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 2, pp. 267–277, Nov. 2015.

[115] X. Li, Q. Yu, Z. Shi, Y. Li, and Q. Yan, "Concatenations of polar codes with outer nonbinary LDPC codes," in *2017 IEEE* 17^{th} *International Conference on Communication Technology (ICCT)*. IEEE, Oct. 2017, pp. 117–121.

[116] S. Zengyou and L. Huanhuan, "Application of concatenated codes in the IEEE802. 11b channel coding," in 3^{rd} *International Conference on Computer Science and Service System*. Atlantis Press, Jun. 2014, pp. 386–389.

[117] R. S. Elagooz, A. Mahran, S. Gasser, and M. Aboul-Dahab, "Efficient low-complexity decoding of CCSDS Reed–Solomon codes based on Justesen's concatenation," *IEEE Access*, vol. 7, pp. 49 596–49 603, Apr. 2019.

[118] M. E. Jourmi, H. Ouahmane, and H. E. Ghazi, "Performance evaluation of concatenated codes in GEO satellite link at Ku band," *International Journal of Electrical Engineering and Technology (IJEET)*, vol. 11, no. 9, pp. 188–200, Nov. 2020.

[119] J. Kang, S. Jun, and B. Wang, "An efficient FEC encoder core for VCM LEO satellite-ground communications," *IEEE Access*, vol. 8, pp. 125 692–125 701, Jul. 2020.

[120] J. Ke, X. Lu, X. Wang, X. Chen, and S. Tang, "Concatenated coding for GNSS signals in urban environments," *Applied Sciences*, vol. 10, no. 18, p. 6397, Sep. 2020.

[121] G. Ricciutelli, T. Jerkovits, M. Baldi, F. Chiaraluce, and G. Liva, "Analysis of the block error probability of concatenated polar code ensembles," *IEEE Transactions on Communications*, vol. 67, no. 9, pp. 5953–5962, Jun. 2019.

[122] P. Chen, B. Bai, Z. Ren, J. Wang, and S. Sun, "Hash-polar codes with application to 5G," *IEEE Access*, vol. 7, pp. 12 441–12 455, Jan. 2019.

[123] J. King, A. Kwon, H. Yang, W. Ryan, and R. D. Wesel, "CRC-aided list decoding of convolutional and polar codes for short messages in 5G," in *ICC 2022-IEEE International Conference on Communications*. IEEE, May. 2022, pp. 92–97.

[124] J. Li, P. Reviriego, L. Xiao, Z. Liu, L. Li, and A. Ullah, "Low delay single error correction and double adjacent error correction (SEC-DAEC) codes," *Microelectronics Reliability*, vol. 97, pp. 31–37, Jun. 2019.

[125] L. J. Saiz-Adalid, P. Reviriego, P. Gil, S. Pontarelli, and J. A. Maestro, "MCU tolerance in SRAMs through low-redundancy triple adjacent error correction," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 23, no. 10, pp. 2332–2336, Oct. 2014.

[126] J. Li, P. Reviriego, and L. Xiao, "Low delay 3-bit burst error correction codes," *Journal of Electronic Testing*, vol. 35, pp. 413–420, Jun. 2019.

[127] J. Li, L. Xiao, P. Reviriego, and R. Zhang, "Efficient implementations of 4-bit burst error correction for memories," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 65, no. 12, pp. 2037–2041, Mar. 2018.

[128] P. Reviriego, S. Liu, L. Xiao, and J. A. Maestro, "An efficient single and double-adjacent error correcting parallel decoder for the (24, 12) extended golay code," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 24, no. 4, pp. 1603–1606, Aug. 2015.

[129] A. Neale and M. Sachdev, "A new SEC-DED error correction code subclass for adjacent MBU tolerance in embedded memory," *IEEE Transactions on Device and Materials Reliability*, vol. 13, no. 1, pp. 223–230, Dec. 2012.

[130] A. Neale, "Design and analysis of an adjacent multi-bit error correcting code for nanoscale SRAMs," 2014.

[131] A. Neale, M. Jonkman, and M. Sachdev, "Adjacent MBU tolerant SEC-DED-TAEC-yAED codes for embedded SRAMs," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 62, no. 4, pp. 387–391, Nov. 2014.

[132] P. Reviriego, J. Martínez, S. Pontarelli, and J. A. Maestro, "A method to design SEC-DED-DAEC codes with optimized decoding," *IEEE Transactions on Device and Materials Reliability*, vol. 14, no. 3, pp. 884–889, Jul. 2014.

[133] A. Dutta and N. A. Touba, "Multiple bit upset tolerant memory using a selective cycle avoidance based SEC-DED-DAEC code," in 25^{th} *IEEE VLSI Test Symposium (VTS'07).* IEEE, May. 2007, pp. 349–354.

[134] M. Richter and K. Oberlaender and M. Goessel, "New linear SEC-DED codes with reduced triple bit error miscorrection probability," in *2008* 14^{th} *IEEE International On-Line Testing Symposium.* IEEE, Jul. 2008, pp. 37–42.

[135] R. Datta and N. A. Touba, "Exploiting unused spare columns to improve memory ECC," in *2009* 27^{th} *IEEE VLSI Test Symposium.* IEEE, May. 2009, pp. 47– 52.

[136] Z. Ming, L. Y. Xiao, and L. H. Wei, "New SEC-DED-DAEC codes for multiple bit upsets mitigation in memory," in *IEEE/IFIP* 19^{th} *International Conference on VLSI and System-on-Chip,.* IEEE, Oct. 2011, pp. 254–259.

[137] A. Dutta, "Low cost adjacent double error correcting code with complete elimination of miscorrection within a dispersion window for multiple bit upset tolerant memory," in *2012 IEEE/IFIP* 20^{th} *International Conference on VLSI and System-on-Chip (VLSI-SoC).* IEEE, Oct. 2012, pp. 287–290.

[138] S. Cha and H. Yoon, "Efficient implementation of single error correction and double error detection code with check bit precomputation for memories," *Journal of Semiconductor Technology and Science*, vol. 12, no. 4, pp. 418–425, Apr. 2012.

[139] H. Jun and Y. Lee, "Single error correction, double error detection and double adjacent error correction with no mis-correction code," *IEICE Electronics Express*, vol. 10, no. 20, pp. 20 130 743–20 130 743, Oct. 2013.

[140] H. Jun, Y. Lee, "SEC—DED—DAEC codes for on-chip memory protection against multiple cell upsets," *International Journal of Computer and Information Technology*, Nov. 2014.

[141] S. Cha and H. Yoon, "Single-error-correction and double-adjacent-error-correction code for simultaneous testing of data bit and check bit arrays in memories," *IEEE Transactions on Device and Materials Reliability*, vol. 14, no. 1, pp. 529–535, Jan. 2014.

[142] S. Tripathi, R. K. Maity, J. Jana, J. Samanta, and J. Bhaumik, "FPGA based low area multi-bit adjacent error correcting codec for sram application," *Radioelectronics and Communications Systems*, vol. 63, no. 10, pp. 543–552, Oct. 2020.

[143] H. Jun and Y. Lee, "Protection of on-chip memory systems against multiple cell upsets using double-adjacent error correction codes," *IEICE Transactions on Electronics*, vol. 98, no. 3, pp. 267–274, Mar. 2015.

[144] A. Sanchez-Macian, P. Reviriego, and J. A. Maestro, "Optimizing the implementation of SEC–DAEC codes in FPGAs," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 24, no. 12, pp. 3538–3542, May. 2016.

[145] P. Reviriego, S. Pontarelli, A. Evans, and J. A. Maestro, "A class of SEC-DED-DAEC codes derived from orthogonal latin square codes," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 23, no. 5, pp. 968–972, May. 2014.

[146] J. Li, P. Reviriego, C. Argyrides, and L. Xiao, "Efficient concurrent error detection for SEC-DAEC encoders," in *2019 IEEE* 25^{th} *International Symposium*

on On-Line Testing and Robust System Design (IOLTS). IEEE, Jul. 2019, pp. 165–170.

[147] M. Tabandeh, "Application of karnaugh map for easy generation of error correcting codes," *Scientia Iranica*, vol. 19, no. 3, pp. 690–695, Jun. 2012.

[148] S. Tripathi, J. Jana, and J. Bhaumik, "New low power and fast SEC-DAEC and SEC-DAEC-TAEC codes for memories in space application," *Integration*, vol. 89, pp. 56–72, Mar, 2023.

[149] S. Vijayakumaran and D. Pal, "On the minimum redundancy of SEC-DAEC-TAEC binary linear block codes," *IEEE Communications Letters*, vol. 20, no. 4, pp. 652–655, Feb. 2016.

[150] L. J. Saiz-Adalid, J. Gracia-Moran, D. Gil-Tomas, J. C. Baraza-Calvo, and P. J. Gil-Vicente, "Ultrafast codes for multiple adjacent error correction and double error detection," *IEEE Access*, vol. 7, pp. 151 131–151 143, Oct. 2019.

[151] S. B. Maity, R. K. Maity, J. Samanta, and C. K. De, "Comments on "new low power and fast SEC-DAEC and SEC-DAEC-TAEC codes for memories in space application"," *Integration*, vol. 92, pp. 35–37, Sep. 2023.

[152] J. Li, P. Reviriego, L. Xiao, C. Argyrides, and J. Li, "Extending 3-bit burst error-correction codes with quadruple adjacent error correction," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 26, no. 2, pp. 221–229, Nov, 2017.

[153] J. Gracia-Moran, L. J. Saiz-Adalid, D. Gil-Tomas, and P. J. Gil-Vicente, "Improving error correction codes for multiple-cell upsets in space applications," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 26, no. 10, pp. 2132–2142, May. 2018.

[154] S. Prasad, P. B. Natarajan, and L. B. Shankar, "Broadening 3-bit burst error-correction codes with quadruple adjacent error correction," in *2021 International Conference on Intelligent Technologies (CONIT).* IEEE, Jun. 2021, pp. 1–5.

[155] X. She, N. Li, and D. W. Jensen, "SEU tolerant memory using error correction code," *IEEE Transactions on Nuclear Science*, vol. 59, no. 1, pp. 205–210, Jan. 2012.

[156] L. J. Saiz-Adalid, P. Gil, J. Baraza-Calvo, J. Ruiz, D. Gil-Tomás, and J. Gracia-Morán, "Modified Hamming codes to enhance short burst error detection in semiconductor memories," in *2014* 10^{th} *European Dependable Computing Conference.* IEEE, May. 2014, pp. 62–65.

[157] A. Klockmann and G. G. M. Goessel, "A new 3-bit burst-error correcting code," in *2017 IEEE* 23^{rd} *International Symposium on On-Line Testing and Robust System Design (IOLTS).* IEEE, Jul. 2017, pp. 3–4.

[158] A. Klockmann, G. Georgakos, and M. Goessel, "Systematic design of a new 3-bit-burst-error correction code with minimal number of check bits," in *Reliability by Design; 9. ITG/GMM/GI-Symposium.* VDE, Sep. 2017, pp. 1–6.

[159] A. Das and N. A. Touba, "Low complexity burst error correcting codes to correct MBUs in SRAMs," in *Proceedings of the 2018 on Great Lakes Symposium on VLSI*, May. 2018, pp. 219–224.

[160] J. Li, P. Reviriego, L. Xiao, and A. Klockmann, "Protecting large word size memories against MCUs with 3-bit burst error correction," in *2019 IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT).* IEEE, Oct. 2019, pp. 1–4.

[161] A. Das and N. A. Touba, "A new class of single burst error correcting codes with parallel decoding," *IEEE Transactions on Computers*, vol. 69, no. 2, pp. 253–259, Oct. 2019.

[162] F. Silva, A. Pinheiro, J. Silveira, and C. Marcon, "A triple burst error correction based on region selection code," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 31, no. 8, pp. 53–64, May. 2023.

[163] J. Q. Li, L. Y. Xiao, L. He, and H. Wu, "A method to design 5-bit burst error correction code against the multiple bit upset (mbu) in memories," in *2019 IEEE 13th International Conference on ASIC (ASICON).* IEEE, Oct. 2019, pp. 1–4.

[164] R. K. Maity, S. Tripathi, J. Samanta, and J. Bhaumik, "Lower complexity error location detection block of adjacent error correcting decoder for SRAMs," *IET Computers & Digital Techniques*, vol. 14, no. 5, pp. 210–216, Sep. 2020.

[165] A. Das, B. Bag, C. Bose, and A. Chandra, "ABER of an FSO link in Gamma-Gamma turbulence with SSK and SEC," in *2021 IEEE 93rd Vehicular Technology Conference (VTC2021-Spring).* IEEE, 2021, pp. 1–5.

[166] A. P. Prudnikov, Y. A. Brychkov, and O. I. Marichev, *Integrals and Series Volume 3 : More Special Functions.* Gordon and Breech Science, 1990.

[167] I. Gradshteyn and I. Ryzhik, *Table of Integrals, Series, and Products Seventh Edition.* Elsevier Academic Press, 2007.

[168] A. Das, B. Bag, C. Bose, and A. Chandra, "Performance analysis of MIMO FSO link with Alamouti coding and switch-and-examine combining," *Photonic Network Communications*, vol. 36, no. 3, pp. 350–360, Aug. 2018.

[169] M. Jana and S. Kumar, "Performance analysis of IRS-assist wireless communication system with Alamouti transmit diversity scheme." *Int. J. Electron. Commun.*, vol. 171, pp. 1–10, Nov. 2023.

[170] V. S. Adamchik and O. I. Marichev, "The algorithm for calculating integrals of hypergeometric type functions and its realization in REDUCE system," in *Proc. of the international symposium on Symbolic and algebraic computation.* Tokyo, Japan: ACM, Aug. 1990, pp. 212–224.

[171] R. A. Alsemmeari, S. T. Bakhsh, and H. Alsemmeari, "Free space optics vs radio frequency wireless communication," *Int. J. Inf. Technol. Comput. Sci*, vol. 8, no. 9, pp. 1–8, Sep. 2016.

[172] H. Henniger and O. Wilfert, "An introduction to free-space optical communications." *Radioengineering*, vol. 19, no. 2, pp. 203–212, Jun. 2010.

[173] H. Kaushal, V. Jain, and S. Kar, *Free space optical communication*. Springer, Feb. 2017.

[174] S. Al-Ahmadi, "The Gamma-Gamma signal fading model: A survey," *IEEE Antennas and Propagation Magazine*, vol. 56, no. 5, pp. 245–260, Oct. 2014.

[175] K. Prabu, "Analysis of FSO systems with SISO and MIMO techniques," *Wireless Personal Communications*, vol. 105, pp. 1133–1141, Apr. 2019.

[176] T. N. Khajwal, A. Mushtaq, and S. Kaur, "Performance analysis of FSO-SISO and FSO-WDM systems under different atmospheric conditions," in *2020 7th International Conference on Signal Processing and Integrated Networks (SPIN)*. IEEE, Jan. 2020, pp. 312–316.

[177] W. Gappmair and M. Flohberger, "Error performance of coded FSO links in turbulent atmosphere modeled by Gamma-Gamma distributions," *IEEE transactions on wireless communications*, vol. 8, no. 5, pp. 2209–2213, May. 2009.

[178] S. S. Muhammad, T. Javornik, I. Jelovčan, Z. Ghassemlooy, and E. Leitgeb, "Comparison of hard-decision and soft-decision channel coded M-ary PPM performance over free space optical links," *European Transactions on Telecommunications*, vol. 20, no. 8, pp. 746–757, Dec. 2009.

[179] S. Garg, A. Dixit, and V. K. Jain, "Analysis of LDPC codes in FSO communication system under fast fading channel conditions," *IEEE Open Journal of the Communications Society*, vol. 2, pp. 1663–1673, Jul. 2021.

[180] B. D. Ajewole, P. A. Owolawi, and V. M. Srivastava, "Error performance of coded BPSK OFDM-FSO system under atmospheric turbulence," *J. Commun.*, vol. 14, no. 10, pp. 936–944, Oct. 2019.

[181] S. Garg, N. Gupta, A. Dixit, and V. K. Jain, "Capacity and BER analysis of BCH and LDPC coded FSO communication system for different channel conditions," *Optical and Quantum Electronics*, vol. 53, pp. 1–25, May. 2021.

[182] H. Wang, F. Wang, and M. Cao, "A novel concatenated coding scheme combined LDPC and VBLAST for FSO links," in *2015 IEEE International Conference on Communication Software and Networks (ICCSN)*. IEEE, Jun. 2015, pp. 192–195.

[183] R. Gupta, T. S. Kamal, and P. Singh, "Concatenated LDPC-TCM codes for better performance of OFDM-FSO system using Gamma–Gamma fading model," *Wireless Personal Communications*, vol. 106, pp. 2247–2260, Jun. 2019.

[184] N. Gupta, A. Dixit, and V. K. Jain, "Performance analysis of BCH and repetition codes in Gamma-Gamma faded FSO link," in *2019 National Conference on Communications (NCC)*. IEEE, Feb. 2019, pp. 1–5.