**Towards Developing a limited Magnitude Error Correction Methodology**

**By**

**Mahmudur Rahman Limon**

Roll: 1607049



**Department of Computer Science and Engineering**

**Khulna University of Engineering & Technology**

**Khulna 9203, Bangladesh**

**February, 2021**

**Towards Developing a limited Magnitude Error Correction Methodology**

**By**

**Mahmudur Rahman Limon**

Roll: 1607049

A thesis submitted in partial fulfilment of the requirements for the degree of

“Bachelor of Science in Computer Science and Engineering”

**Supervisor:**

**Dr. Muhammad Sheikh Sadi**

Professor

Department of Computer Science and Engineering

Khulna University of Engineering & Technology …...…………………

Khulna, Bangladesh Signature

**Department of Computer Science and Engineering**

**Khulna University of Engineering & Technology**

**Khulna 9203, Bangladesh**

**February, 2021**

# Acknowledgement

With the blessings and limitless mercy of Almighty, i am able to do this. I express my heartiest gratitude to Almighty Allah. Then i express our indebtedness to our honorable supervisor Prof. Dr. Muhammad Sheikh Sadi for his helpful contribution, necessary guidance, suggestions and encouragement to me. I would also like to thank all the teachers and staffs of CSE department of KUET who helped me providing guidelines to perform the work.

# Abstract

Multilevel cell (MLC) memories have been advocated for increasing density at low cost in next generation memories. Several bits in a cell can reduce the distance between levels. So the sorted margin makes the margin more vulnerable to defective and that’s leading to an error in store data. To protect the memories from such errors and ensure that the stored data is not corrupted, Error correction codes (ECC) are commonly used. The main scheme is mainly based by combining ECCs that are commonly used to protect traditional memories. The IP (interleaved parity) bits identifying the remaining erroneous bits in the memory cell. The scheme is also competitive in terms of number of parity check bits and memory redundancy. [Multilevel cell (MLC) memory depend mostly on the mapping of levels to bits and the magnitude of the errors.Magnitude-1 errors affect a single bit and thus they can be corrected by an SEC code. Magnitude-2 errors only affect two bits. Magnitude-1 errors always corrupt the lowest bit (the second lowest bits) of the memory cell, only a single parity bit (two parity bits) that covers all lowest bits. The magnitude and sign of the error are known properly then it can be corrected with relatively simple code. Basically One-Bit Parity (OBP) scheme that can detect any magnitude-1 error and the Two-Bit Parity (TBP) scheme that can detect any magnitude-2 error. So the efficient scheme that has magnitude errors correction capability for MLC memories.SEC-DAEC code in the two lowest bits of cell with the IP bits to correct up magnitude-3 errors.](#_Toc63085786)

# Contents

Contents

[Acknowledgement ii](#_Toc63121268)

[Abstract iii](#_Toc63121269)

[Contents iv](#_Toc63121270)

[Chapter 1 1](#_Toc63121271)

[Introduction 1](#_Toc63121272)

[1.1 Introduction 1](#_Toc63121273)

[1.2 Problem Statement 1](#_Toc63121274)

[1.3 Objectives of the Thesis 2](#_Toc63121275)

[1.4 Scope of the Thesis 2](#_Toc63121276)

[1.5 Contribution of the Thesis 2](#_Toc63121277)

[1.6 Thesis Organization 3](#_Toc63121278)

[Chapter 2 4](#_Toc63121279)

[Literature Review 4](#_Toc63121280)

[2.1 Introduction 4](#_Toc63121281)

[2.2 Magnitude Errors and operations Briefly: 4](#_Toc63121282)

[2.3 Limited Magnitude Errors in MLC Memories? 5](#_Toc63121284)

[2.4 Why Magnitude Error a Matter of Concern in MLC? 5](#_Toc63121285)

[2.5 Interleaved Parity (IP) & Syndrome 6](#_Toc63121286)

[2.6 SEC-DAEC codes & Scheme 7](#_Toc63121287)

[2.7 Encoder Theory 8](#_Toc63121288)

[2.8 Decoder Theory 8](#_Toc63121289)

[2.10 Summary 8](#_Toc63121290)

[Chapter 3 9](#_Toc63121291)

[Methodology 9](#_Toc63121292)

[3 Introduction 9](#_Toc63121293)

[Chapter 4 10](#_Toc63121294)

[Experimental Analysis 10](#_Toc63121295)

[4.1 Introduction 10](#_Toc63121296)

[4.2 Experimental Setup 10](#_Toc63121297)

[4.3 Results Analysis 11](#_Toc63121298)

[4.4 Summary 11](#_Toc63121299)

[Chapter 4 12](#_Toc63121300)

[Conclusions 12](#_Toc63121301)

[5.1 Concluding Remarks 12](#_Toc63121302)

[5.2 Future Work 12](#_Toc63121303)

[Bibliography 12](#_Toc63121304)

List of Figures

[Figure 1.3-bit memory cell with binary mapping. 5](#_Toc63122085)

[Figure 2.t = 3 IP bits calculated for k = 8 data bits. 6](#_Toc63122086)

[Figure 3.Syndrome bits calculated as per the read-out codeword: (a) error free 7](file:///C:\Users\limon28\Desktop\Towards%20Developing%20a%20limited%20Magnitude%20Error%20Correction%20Methodology.docx#_Toc63122087)

[Figure 4.Implementation of the proposed scheme to protect a 32-bit MLC memory with 3-bit cells. 8](file:///C:\Users\limon28\Desktop\Towards%20Developing%20a%20limited%20Magnitude%20Error%20Correction%20Methodology.docx#_Toc63122088)

[Figure 5.H matrix of the (28, 22) SEC-DAEC code. 9](file:///C:\Users\limon28\Desktop\Towards%20Developing%20a%20limited%20Magnitude%20Error%20Correction%20Methodology.docx#_Toc63122089)

# Chapter 1

# Introduction

## 1.1 Introduction

In this chapter, some formal statements are described. Problem statement, Objective of the thesis, scope of the thesis, contribution of the thesis and thesis organization are stated. In the thesis organization, the chapters are described briefly.

## 1.2 Problem Statement

Most of devices has dramatically grown together with requirements of large volume in storage data. Many technologies have been proposed for the next generation of memories, such as phase change (PC) memories, magneto-electric (ME) memories and  
memristor-based memories [2]–[5]. In multilevel cell (MLC) design is to reduce  
the physical size of the non-volatile (NV) memory element.

A common technique to protect memories is to use Error Correction Codes (ECCs) that add redundancy to the data in the form of parity bits that are then used to detect and/or  
correct errors. Codes for protecting memories have focused on single bit error that detect or correct. A single parity (SP) bit which also says about Single Error Correction (SEC) and this is the most relevant error patterns [14]. Another approach is to use ECCs that can deal with adjacent bit errors; for example, interleaved parity (IP) bits can detect multiple adjacent bit errors, or codes that achieve single error correction and double adjacent error correction (SEC-DAEC) can be constructed [14], [17], [18].

The proposed scheme combines the use of a low redundancy SEC-DAEC code in the two lowest bits of the cell with IP bits to correct up to magnitude-3 symmetric errors. The SEC-DAEC code is used to locate the cell in error and correct errors on some bits. Moreover, ECC cannot check all words in parallel in TCAM easily. For the ECC, using a decoder per word is costly and leads to a huge memory and high-power consumption.

## 1.3 Objectives of the Thesis

In this thesis, an efficient coding method has been proposed by which SE-DAE can be detected and corrected easily. The main objectives of this thesis are as follows.

* To propose a magnitude error detection and correction method for MLC.
* To minimize information redundancy and other overheads without sacrificing error detection ability.
* To make an efficient technique that has higher accuracy rate than other technique for MLC.

## 1.4 Scope of the Thesis

As it was mentioned earlier, the primary objective of this thesis is to develop an efficient technique to detect & correct magnitude errors in MLC. The proposed method has higher error detection and correction rate. In fact, the proposed method increases the detection and correction rate. The vision was to research and develop systems that can detect and correct multi bit error of any large data block.

In computing system, error detection and correction approach is much needed. Traditional approach towards error detection and correction has some limitation which needs to be addressed. To overcome these problems, we proposed a new method. Feature of this proposed method is as follows:

* Detect and correct multiple bit errors with 100% accuracy.
* Reducing overhead.

## 1.5 Contribution of the Thesis

The proposed method shows how to use parity bits in MLC for magnitude error can be efficiently detected and corrected. The major contributions of this paper are: it can correct a hundred percent of errors for the given 62-bit datawords (for all possible combinations), it requires relatively lesser redundant bits in comparison to existing dominant approaches, and it can detect and correct soft errors in both sequential and random datawords in.

## 1.6 Thesis Organization

**Chapter 1** describes the problem statements, objective, scope and contribitoin of the thesis. A brief discussion is given here.

**Chapter 2** introduces the formal structures and terminologies used in this thesis. Related works in this field by other researchers are also discussed in a brief.

**Chapter 3** Methodology of the thesis.

**Chapter 4** draws the conclusion of our proposed method. It also states some future works that can be done for improving the system.

# Chapter 2

# Literature Review

## 2.1 Introduction

In this chapter some formal statements and terminologies related to this thesis will be discussed. Some specifications will also be discussed which will be used to describe the proposed method. This statements and terminologies will be elaborated using some example and pictorial representation. Error detection and correction scheme will also be discussed. Some problems of the existing method will also be explained. Many of the existing methods have some problems. Some methods are used efficiently and some are not. In this chapter some existing method are described.

## 2.2 Magnitude Errors and operations Briefly:

## Multilevel cell (MLC) memory depend mostly on the mapping of levels to bits and the magnitude of the errors.Magnitude-1 errors affect a single bit and thus they can be corrected by an SEC code. Magnitude-2 errors only affect two bits. Magnitude-1 errors always corrupt the lowest bit (the second lowest bits) of the memory cell, only a single parity bit (two parity bits) that covers all lowest bits. The magnitude and sign of the error are known properly then it can be corrected with relatively simple code. Basically One-Bit Parity (OBP) scheme that can detect any magnitude-1 error and the Two-Bit Parity (TBP) scheme that can detect any magnitude-2 error. So the efficient scheme that has magnitude errors correction capability for MLC memories. SEC-DAEC code in the two lowest bits of cell with the IP bits to correct up magnitude-3 errors.

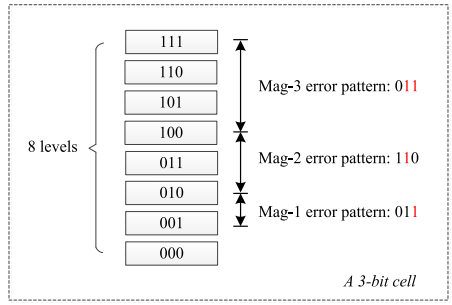


Figure 1.3-bit memory cell with binary mapping.

## 2.3 Limited Magnitude Errors in MLC Memories?

In MLC memories (implemented using emerging technologies), the range of the physical parameter of a cell is divided by levels based on the feature of the storage medium. Figure 1 shows a 3-bit MLC memory cell with binary mapping from eight states (i.e. an octal memory cell).However, a potential issue of MLC memories is that when a cell is programmed, or after a number of switching cycles, the margins between levels shift, or overlap as a result of the  
deterioration of the storage medium. This shift can occur in either one or both directions (e.g.,  
unidirectional in PCM cells [25], [26] and bidirectional in  
memristor-based memory cells[11].The errors in MLC memories are always a limited magnitude.

## 2.4 Why Magnitude Error a Matter of Concern in MLC?

An important observation is that when using binary mapping from levels to bits, magnitude-1 and magnitude-2 errors always affect the lowest or the second lowest bit. This occurs because  
the difference in values between maginitude-1 levels is “01” while between magnitude-2 levels it is “10”. In Figure 1 (where “mag” denotes “magnitude”), a magnitude-  
1 error causes a pattern of “1” on the lowest bit, and the magnitude-2 error causes a “1” on the second lowest bit. The observation propose several limited magnitude error correction schemes.

## 2.5 Interleaved Parity (IP) & Syndrome

Conventional binary memories to detect errors that which affects adjacent bits [14] and IP are used here. To detect *t*-bit adjacent errors on a *k*-bit data, *t* IP bits are needed to cover all data bits; they are calculated by:  
pi = di ⊕ di+t ⊕ di+2t ⊕ . . . ⊕ di+j·t(1)

So from the theory we can say that if t = 3 IP bits calculated for k=8 data bits “10010111”shown in Figure 2.

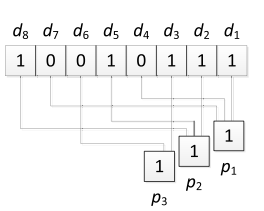


Figure 2.t = 3 IP bits calculated for k = 8 data bits.

The calculation process for the IP bits (i.e., Eq. (1)) is implemented by an encoder. Then in a write operation, the calculated IP bits are stored together with the data bits into a memory word, forming an *n*-bit codeword (i.e., n = k + t).

IP bits are recalculated first as per the read out data bits d’, and then compared with read out data bits p’ to generate the syndrome bits S;

Pi′′ = di ⊕ di′ + t ⊕ di′ + 2t ⊕ . . . ⊕ di′ + j·t (2)   
Si = pi′ ⊕ pi′′ (3)

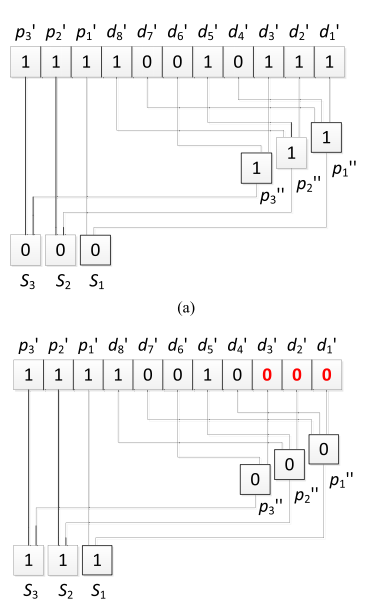


Figure 3.Syndrome bits calculated as per the read-out codeword: (a) error free

Figure 3 shows the cases of error free in (a) and erroneous bits *d*1’, *d*2’, *d*3’ in (b). In the first case *S* = “000” and in the second case *S* = “111”. Therefore, an error detection signal can be obtained if the syndrome bits are not all zero. As per Eqs. (1)-(3), the error pattern is reflected on the syndrome bits.

The use of an IP-based scheme can only determine the error pattern, but it cannot locate the position of the erroneous bits and thus, it is unable to achieve error correction. For example,  
in the case shown in Figure 3 (b), an error affecting d4’, d5’, d6’ generates the same syndrome bits of “111”. To correct errors, stronger ECCs are therefore required.

## 2.6 SEC-DAEC codes & Scheme

SEC codes are able to correct single bits errors. And SEC-DAEC codes have developed to correct also double adjacent bit errors [17], [18].

Symmetric magnitude errors will always corrupt at least one of the lowest and second lowest bits. an SECDAEC code that covers the lowest and second lowest bits per cell can locate the erroneous cell, as well as correcting errors on those bits. However, magnitude-3 errors can also affect some upper bits; for example, in Figure 1 a magnitude-  
2 error affects the second and third lowest bits causing a pattern of “110”.

Combining IP with SEC-DAEC codes, the proposed scheme can always guarantee correct data under symmetric magnitude-3 errors.

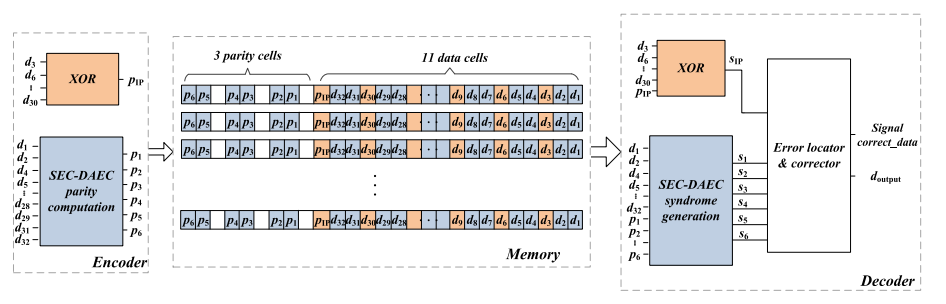


Figure 4.Implementation of the proposed scheme to protect a 32-bit MLC memory with 3-bit cells.

## 2.7 Encoder Theory

In the proposing scheme combines two ECC techniques that are IP and SEC-DAEC. The IP bits are calculated and stored in the MLC memory cells as we target the single cell errors. And the next SEC-DAEC parity bits are need to calculate.

## 2.8 Decoder Theory

In decoder we follow the H matrix and two syndrome generation blocks, an error locator and an error locator. In error locator SEC-DAEC syndrome bits are compared with all correctable error patterns to determine the erroneous memory cell.

## 2.10 Summary

In this chapter, we have discussed about errors and MLC. There are many method existing to detect and correct errors in MLC. Here, in this chapter some of them are mentioned and discussed.

# Chapter 3

# Methodology

## 3 Introduction

Here we discussed about the working progress of error detection in briefly. Here I have taken a 32 bit data. Which is

11111111111111111111111111111011

All the position of the bit are defined as d1 to d32.And a H matrix also need to be taken for finding the Parity. In the 32-bit MLC memory with 3-bit cells shown in Figure 4, In H matrix 28 single bit errors and 27 double adjacent bit errors can be corrected by the (28, 22) SEC-DAEC code; For more efficient SEC-DAEC codes with low redundancy is to design the Hmatrices.

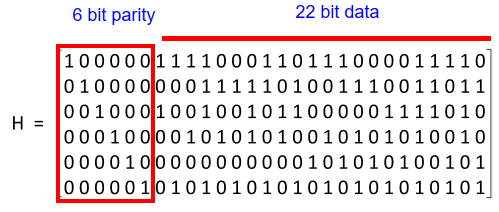


Figure 5.H matrix of the (28, 22) SEC-DAEC code.

In the H matric we can see, from the main data we have taken 22 bits except the 10 bit. The extra 10bit (d3,d6,d9,d12,d15,d18,d21,d24,d27,d30) are separated because as the discussion of previous chapter 2.So,from that theory I separated them and XOR them.

IP parity Pip = d30⊕……..⊕d9⊕d6⊕d3 = 1

This value can help us to find out the syndrome. When we properly able to find out the syndrome and the Sip then we can able to find the error and also the types of error, SEC-DAEC.

SEC-DAEC Parity:

P1 = d32 ⊕ d31 ⊕……..⊕ d5 ⊕ d4 ⊕ d2 = 1

P2 = d28 ⊕ d26 ⊕……..⊕ d5 ⊕ d2 ⊕ d1 = 1

P3 = d32 ⊕ d28 ⊕……..⊕ d7 ⊕ d5⊕ d2 = 0

P4 = d28 ⊕ d26 ⊕……..⊕ d10 ⊕ d7 ⊕ d2 = 1

P5 = d17 ⊕ d14 ⊕……..⊕ d8 ⊕ d4 ⊕ d1 = 0

P6 = d31 ⊕ d28 ⊕……..⊕ d7 ⊕ d4 ⊕ d1 = 1

Here I find out the value of Pip,(P1-P6) and we also check this performing the coding implementation.

# Chapter 4

# Experimental Analysis

## 4.1 Introduction

The experimental analysis of this paper is divided into two sections: Experimental Setup and Results Analysis. These are outlined shortly as follows.

## 4.2 Experimental Setup

To implement the proposed methodology, we used a system with the following configuration.

* Intel(R) Core (TM) i7-7700HQ CPU @ 2.80GHz
* 16 GB RAM
* C++ Language

## 4.3 Results Analysis

Here, total data bits 32 are chosen for evaluating the effectiveness of the method .And the H matrix properly need to take and calculation need to proper also. So, i properly taken the value solve them properly so that Syndrome and all the calculation will be perfect.

When the message is sent, and encoder portion are done properly.

## 4.4 Summary

In this chapter, experimental set up and result analysis are discussed.

# Chapter 4

# Conclusions

## 5.1 Concluding Remarks

In this paper has shown the main calculation part finding out the errors. In MLC finding out the error and solve the problem this theory is very important. We can recover the loss data through the scheme.

## 5.2 Future Work

In the thesis, i proposed the method with the calculation. And the future we have done all the theorem with implements and find out error for 64 bit data and the easiest way to find the magnitude error.

# 

# Bibliography

[1] Codes for Limited Magnitude Error Correction in Multilevel Cell Memories Shanshan Liu , *Member, IEEE*, Pedro Reviriego , *Senior Member, IEEE*, and Fabrizio Lombardi , *Fellow*, IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS–I: REGULAR PAPERS, VOL. 67, NO. 5, MAY 2020.  
[2] G. Atwood, S.-I. Chae, and S. S. Y. Shim, “Next-generation memory,”  
*Computer*, vol. 46, no. 8, pp. 21–22, Aug. 2013.  
[3] N. Papandreou *et al.*, “Multilevel phase-change memory,” in *Proc. IEEE  
Int. Conf. Electron. Circuits Syst.*, Dec. 2010, pp. 1017–1020.  
[4] G. S. Sandhu, “Emerging memories technology landscape,” in *Proc. 13th  
Non-Volatile Memory Technol. Symp.*, Jul. 2014, pp. 1–5.  
[5] X. P. Wang, Z. Y. Wang, and Y. Shen, “A high reliable design of  
memristor-based multilevel memory,” in *Proc. IEEE 34th Chin. Control  
Conf.*, Sep. 2015, pp. 5615–5618.  
[6] M. Stanisavljevic, H. Pozidis, A. Athmanathan, N. Papandreou,  
T. Mittelholzer, and E. Eleftheriou, “Demonstration of reliable triplelevel-cell (TLC) phase-change memory,” in *Proc. IEEE 8th Int. Memory  
Workshop (IMW)*, May 2016, pp. 1–4.  
[7] ExtremeTech. (Aug. 2014). *Western Digital’s HGST Division Creates  
New Phase-Change SSD that’s Orders of Magnitude Faster than any  
NAND Flash Drive on the Market*. [Online]. Available: https://www.  
extrem-etech.com/extreme/187577-hitachis-new-phase-change-ssd-isorders-of-magnitude-faster-than-any-nand-flash-drive-on-the-market  
[8] (Nov. 2019). *Intel Optane Memory*. [Online]. Available: https://www.  
intel.com/content/www/us/en/support/products/99745/memory-andstorage/intel-optane-memory.html  
[9] A. Sebastian *et al.*, “Temporal correlation detection using computational  
phase-change memory,” *Nature Commun.*, vol. 8, no. 1, p. 1115,  
Oct. 2017.  
[10] J. Li, B. Luan, and C. Lam, “Resistance drift in phase change memory,”  
in *Proc. IEEE Int. Rel. Phys. Symp. (IRPS)*, Apr. 2012, p. 6C-1.  
[11] J. J. Yang *et al.*, “High switching endurance in TaO*x* memristive  
devices,” *Appl. Phys. Lett.*, vol. 97, no. 23, Dec. 2010, Art. no. 232102.  
[12] S. Lin and D. J. Costello, *Error Control Coding*, 2nd ed.  
Englewood Cliffs, NJ, USA: Prentice-Hall, 2004.  
[13] E. Fujiwara, *Code Design for Dependable Systems: Theory and Practical  
Applications*. Hoboken, NJ, USA: Wiley, 2006.  
[14] Z. N. Wu, P. A. McEwan, K. K. Fitzpatrick, and J. M. Cloffi, “Interleaved  
parity check codes and reduced complexity detection,” in *Proc. IEEE  
Int. Conf. Commun.*, Aug. 2002, pp. 1648–1652.  
[15] M. Y. Hsiao, “A class of optimal minimum odd-weight-column  
SEC-DED codes,” *IBM J. Res. Develop.*, vol. 14, no. 4, pp. 395–401,  
Jul. 1970.  
[16] E. Ibe, H. Taniguchi, Y. Yahagi, K. Shimbo, and T. Toba, “Impact of  
scaling on neutron-induced soft error in SRAMs from a 250 nm to  
a 22 nm design rule,” *IEEE Trans. Electron Devices*, vol. 57, no. 7,  
pp. 1527–1538, Jul. 2010.  
[17] A. Dutta and N. A. Touba, “Multiple bit upset tolerant memory using a  
selective cycle avoidance based SEC-DED-DAEC code,” in *Proc. IEEE  
VLSI Test Symp.*, May 2007, pp. 349–354.  
[18] Z. Ming, X. L. Yi, and L. H. Wei, “New SEC-DED-DAEC codes for  
multiple bit upsets mitigation in memory,” in *Proc. IEEE/IFIP 19th Int.  
Conf. VLSI Syst.-Chip*, Aug. 2011, pp. 254–259.  
[19] Y. Cassuto, M. Schwartz, V. Bohossian, and J. Bruck, “Codes for  
asymmetric limited-magnitude errors with application to multilevel flash  
memories,” *IEEE Trans. Inf. Theory*, vol. 56, no. 4, pp. 1582–1595,  
Apr. 2010.  
[20] A. Das and N. A. Touba, “Limited magnitude error correction using  
OLS codes for memories with multilevel cells,” in *Proc. IEEE Int. Conf.  
Comput. Design (ICCD)*, Nov. 2017, pp. 391–394.  
[21] S. S. Liu, P. Reviriego, K. Namba, S. Pontarelli, L. Y. Xiao, and  
F. Lombardi, “Low redundancy double error correction spotty codes  
combined with gray coding for 64 data bits memories of 4-bit multilevel  
cells,” in *Proc. 32nd IEEE Int. Symp. Defect Fault Tolerance (DFT) VLSI  
Nanotechnol. Syst.*, Oct. 2019, pp. 1–4.  
[22] A. Das and N. A. Touba, “Efficient non-binary Hamming codes for  
limited magnitude errors in MLC PCMs,” in *Proc. IEEE Int. Symp.  
Defect Fault Tolerance VLSI Nanotechnol. Syst. (DFT)*, Oct. 2018,  
pp. 1–6.