DJ Journal of Advances in Electronics and Communication Engineering, Vol. 5(1) 2019, pp. 1-9

RESEARCH ARTICLE

FPGA Implementation of Reduced Complexity LDPC Codes

[*]N Murali Krishna[1]

[1]Department of ECE Sreyas Institute of Engineering &Technology, Hyderabad, India.

Received- 20 February 2018, Revised- 6 October 2018, Accepted- 11 October 2018, Published-15 October 2018

ABSTRACT

Error detection is the process of detecting an error for the purpose of reliable communication between transmitter and receiver. This paper proposes an Error Control Coding (ECC) technique in flash memories using FPGA implementation. Due to several advantages, NAND based Multi-Level Cell (MLC) flash memory is used in this paper. In flash memories, there is an increased amount of soft errors and thus the encoder and decoder circuits of this memory must be protected. For this purpose, fault tolerant system based on Low Density Parity Check (LDPC) product code scheme is introduced. The main requirement of this scheme is to reduce the area. These product code scheme uses LDPC code for performing encoding operation and thus the designed fault tolerant system for flash memories is implemented using the programming language Verilog HDL, and the area covered can also be synthesized using the software XILINX ISE. The obtained synthesis result proves that the hardware complexity of this fault tolerant system of MLC NAND based flash memory is reduced.

Keywords: ECC, FPGA, MLC, LDPC, Memories.

1. INTRODUCTION

Due to the improvement in computer and microelectronics technology, the demand for storing the information is also drastically increased. High storing capacity in memories can be obtained by reducing the size of memory cell. Among the non-volatile memories, flash memory offers better performance and there is an increased demand for flash memory in solid state memories [1]. The term flash refers to the process in which erasing the contents of the memory sector can be done in one step. Flash memories have good non-volatile property and high density and rewritable characteristics of flash memory. Flash memory finds application in many areas such as accumulating and storing data, storage in look up table as well as code embedding. Flash memory is used as a prominent storage media in computer because of its key advantage of increased capacity and low cost. Flash memory is considered to be inexpensive when compared to the non-volatile memories [2, 3].

Flash memory plays an important role in the architecture of computer system. It can be used instead of RAM and also considered as a layer between magnetic disk drive and RAM [4]. In flash memory, erasing can be performed in blocks and the writing operation is performed in pages. Data should be erased first and then it is written into the block. Flash memory arranged in blocks is written for a specified number of times. Writing operation is performed incrementally or sequentially.

Flash memory is divided into two types based on logic gate used. They are named as NAND and NOR based flash memory. [5] has proposed the structure and the characteristics of both NAND and NOR based memory. NOR based flash memory arranges the cell in parallel connection between the two bit lines whereas the NAND memory arranges the cell in series connection. NOR based flash memory consists of control and floating gates.

*Corresponding author. Tel.: +919885837535
Email address: omkrish29@gmail.com (N.M.Krishna)
Double blind peer review under responsibility of DJ Publications
https://dx.doi.org/10.18831/djece.org/2019011001

The floating gate is insulated by an oxide material and is present between the control gate and MOSFET channel. But the NAND based memory consists of a transistor with floating gate. During write operation, the charges will be injected into the floating gate and while performing erase operation, the charges will be released. NAND cell reduces the overhead that occurs during decoding process. NOR based memory is designed to provide efficient random process and consists of separate data and address buses. These two flash memories have several advantages as well as disadvantages. But when compared to NOR, the NAND flash memory is a better option to store data. Hence NAND based flash memory is mainly focused in this paper. The stacked string structure of NAND flash memory is explained briefly in [6].

1.1. NAND based MLC flash memory

NAND flash memory is implemented by using floating gate transistor. The threshold voltage and its characteristics depend mainly on the charge stored in the floating gate [7]. The NAND cell can be programmed by applying the bias voltage to the control gate and the drain of the floating gate. NAND flash memory consists of flash plans. The plan composed of buffer to hold the data, which results in high performance. A number of blocks are present in plans and the block is divided into subclass of pages and each page is associated with a spare and main data area [8]. The main characteristics of NAND based flash memory are described as follows: low memory cost, high performance and reliable medium for storage purposes[9].

To reduce the storage cost of memory, manufacturers move from Single Level Cell (SLC) technology to the MLC technology [10]. SLC technology can be used to store the single bit of data information whereas MLC technology can store multiple bit of cells and hence requires narrow threshold ranges. Due to larger storage capacity, MLC is cheaper than SLC and thus finds promising solution in large scale memory systems [11]. Programming is done first on the LSB bit and then on the MSB bit. Due to the usage of multiple cells, several issues arise in NAND flash memory on the basis of noise, reliability and endurance. To overcome these limitations, error control techniques are proposed. ECC can detect and correct multiple errors, which is also considered as cost efficient. It is important to select a correct ECC technique and is a trade-off between reliability and complexity of memory. This paper proposes the design of product based code scheme for NAND based memory by using LDPC codes.

1.2. Error correcting code

ECC technique is used to correct errors occuring in digital communication and provide reliable communication over the channel. Errors are introduced in the data while transmitting from source to receiver due to channel noise [12]. The general idea behind the correction technique is to add redundancy bits to the message signal to determine the consistency of the signal. ECC schemes are described as systematic or non-systematic scheme. In systematic, the transmitter sends the original message by attaching the parity bits whereas in systematic, the original message is converted to encoded message which has bits same as that of original message signal. [13] has developed the error correcting procedures for the conventional codes. In this paper, fault tolerant system based on product code scheme is proposed. The product code uses linear block code named as low density parity check codes.

1.3. LDPC

LDPC codes are a subclass of block code, in which the block of data can be encrypted. It is also considered as one of the classes of channel capacity coding technique. Also it is considered as a prominent one because of its better performance when compared to other codes. One important feature of this code is that the parity check matrix denoted as [H] reduces the complexity of LDPC code thus resulting in the decrease of latency [14]. Iterative algorithm can be used to decode the LDPC codes. The performance of the LDPC decoder mainly depends on the parity check matrix [H]. This performance depends directly on the bit error rate performance and the hardware complexity. Depending on parity check matrix, there are several LDPC codes used named as regular and irregular codes, random and pseudo-random codes [15]. Assuming the code word comprising (N-M) variable nodes and M check

nodes, the code rate of LDPC code is defined as in (1) [16].

$$r = \frac{N-M}{N} \times 100\% \quad (1)$$

LDPC of class of Euclidean geometric is proposed, which are constructed by using the lines and points of a Euclidean geometry whereas encoding can be performed using shift registers [17]. An error problem does not occur usually in these codes when compared to other codes. In wireless communication networks, LDPC codes are widely used because of its several advantages [18]. Because of the variation in code length and code rate, LDPC covers wide range in many applications. Some of the application that uses LDPC are GEO-mobile radio, LTE Mobile Communication, wireless regional area network and Bit patterned media recording [19. The remainder of the work is organized as follows: Section II explains some of the error correction techniques used in the existing methods. Section III includes the proposed fault tolerant architecture and the product code scheme. Section IV describes the result of the proposed architecture. Section V concludes the paper.

2. LITERATURE SURVEY

[20] has proposed the error control technique for retention error that occurs in MLC based NAND flash memory. If leakage occurs in the floating gate of the flash memory cell, then retention error arises. In this, read-retry technique is proposed to avoid the retention error by adjusting the threshold voltage up or down so that error can be detected and minimized. [21] has developed an error recovery strategy word line program disturbance, in which the leakage can be compensated by inducing an electron. In this, the leakage mainly occurs due to the long retention time of the floating gate. Thus this method is used to reduce the retention error rate and to improve the performance of the NAND flash memories. [22] has used an adaptive rate error correction scheme to tackle the reliability problem of flash memories. Adaptive rate refers to the changing correction capability. In this, BCH code is used for encoding and decoding. The syndrome is calculated for the data retrieved from memory. If the syndrome is zero then it assumes that there is no error. If the syndrome is nonzero, then it is sent to the error location block to determine the polynomial. Thus corrected messages can be obtained, which improves the reliability. [23] has employed a new error correcting approach known as programming initial step only). In this, the electrons which are injected to the cell can compensate the charge loss over time. It can also reduce the error with minimum overhead and is more efficient on the basis of read operation. This technique has the capability to share the same memory cell hence applied to either MSB or LSB. By using this technique, over programmable issues can be compensated. [24] has put forward a REAL scheme, in which the propagation user detects and corrects the error in the received code words. The objective of this scheme is to add the information bits into the decoding process of LDPC so that read performance can be improved by reducing the decoding iterations. [25] has stated a CooECC technique to reduce the latency of MSB pages. This minimizes the decoding rate of MSB pages. The data error characteristics caused due to retention error is combined with the decoding of LSB pages. Also in this, the information obtained is accurate and convergence speed can be enhanced to have low latency. Pattern aware write strategy is proposed in [26] to reduce the bit error occurring in the flash memory. This can allocate certain blocks to store the data and perform bit operations on these data. This technique also improves the reliability of flash memory drives.

A multilevel FEC technique is proposed for multiple bits used in flash cell. This technique can obtain larger coding gain, thus reducing the power consumption [27]. In this after performing erase operation, the threshold voltage will be in error state and it consists of lowest value. Also, the threshold voltage can be increased by the programming operation hence moved to the programmed states. This scheme increases the performance of NAND flash memory. A multi strategy ECC technique known as pre-check mechanism is suggested. This scheme is based mainly on the polar code thus the endurance of memory can be increased [28]. By performing bitwise operation, the efficiency can be improved thereby eliminating the hardware cost. Also, in this, the polar encoder is used to

encode the binary bits from the information system and the decoding operation is performed by the decoder thus the performance can be improved. [29] has employed systematic error correcting code technique. To perform write operation, the information is given to the input of an encoder. The encoder consists of EXOR tree which is used to compute the check symbol and these are written together with the data symbol. These two symbol forms a code word. For reading operation, syndrome pattern is calculated and the same encoding setup can be used in decoder. This method reduces the area of flash MLC memory. Flash correct and refresh FCR scheme is proposed [30] to tolerate the retention error that occurs in the memory. The basic idea of this technique is to read, correct and reprogram the data periodically before it gets affected by the error. FCR are of different types; hybrid FCR is mainly used to reduce the data overhead mainly caused due to reprogramming. The adaptive rate FCR reduces the unnecessary refresh operation that occurs in a memory. Remapping FCR reduces error by performing remapping operation thus resulting in high performance. A Neighbor Assisted Error Correction (NAC) scheme is proposed to correct error. The most important objective of this is to re-read a memory pages and it improves the life time by 33%. To improve the capability of correcting error, NAC based blocks are added to the controller (i.e) NAC based engine and the NAC buffer. NAC buffer size is considered as an important parameter in this technique [31]. After performing NAC iterations, if error occurs, it cannot be corrected. Gradual error correction code is used [32] that offers a best level of correcting errors when compared to other ECC techniques. In this method, storing of parity bits consumes more space. It consists of five blocks and each block is comprised of eight pages with data as well as spare regions. By using the parity bits, the error in this flash memory will be corrected, and thus lifetime can be extended.

3. FAULT TOLERANT SYSTEM

Fault tolerant system is an architecture in which the transient fault can be tolerated in logic and storage circuits. Even if failure arises in the component of the system, fault tolerant makes the system to work properly. ECC guarantee a fault tolerant detector to handle multiple errors hence this type of fault tolerance detector is named as FSD-ECC.

3.1. Memory system

The fault tolerant memory system is shown in figure 1. This system consists of encoder, corrector and fault secured detector circuit [33]. The brief descriptions of these components are illustrated in the following subclasses.

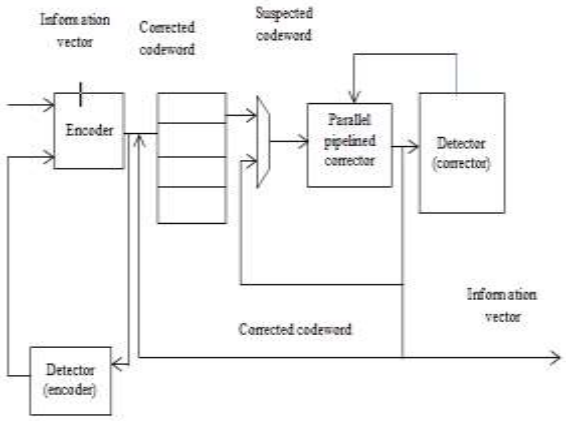

Figure 1.Fault tolerant memory system

3.1.1. Encoder

First, the information bits to be encoded are given to the encoder circuit. After encoding, it is given to the fault secure detector. This detector circuit monitors the output of encoder. If an error is detected, the same operation is repeated until it produces the correct suitable output.

3.1.2. Memory

If the error is not detected by the detector circuit, the particular code word will be stored in memory. These code words will be present in the memory for a long time. During this time, transient error may occur. To avoid these, a process known as memory scrubbing is introduced. It is the process in which the code words can be read to detect the errors and it is written back to the memory.

3.1.3. Corrector

After performing scrubbing operation, some code words are considered as a susceptible one and these are given to the corrector circuit to detect the error and correct it. Similar to encoding unit, a detector is also present to monitor the operation of the

corrector. After performing all these operations, the output information vector is displayed. The fault tolerant system is designed using LDPC product code along the rows and column. Since it consists of cross parity checking, it is considered as one of the strong error correcting technique. The hardware overhead of this system is low and hence finds application in many areas.

3.2. LDPC Product code schemes

Product code scheme is a method in which a small length code can be grouped or constituted to form a long length LDPC code with high ECC capability. [34, 35] These product code schemes offer better performance in terms of cross parity check matrix, and also provide low circuit overhead when compared to long length plain codes. The main reason behind these advantages is that the LDPC code words used in the product code scheme has low capability to correct the error. The general architecture of product code scheme is given in figure 2.

Let C1 be a linear LDPC code with size (n1, k1) and let C2 be a linear code with size (n2, k2), then, a linear code (n1n2, k1k2) is formed in which each code word is arranged in the form of rectangular array with n1 columns and n2 rows assumed that each row is considered as a code word in C1, and each column is considered as a code word in C2 as shown in figure 2.

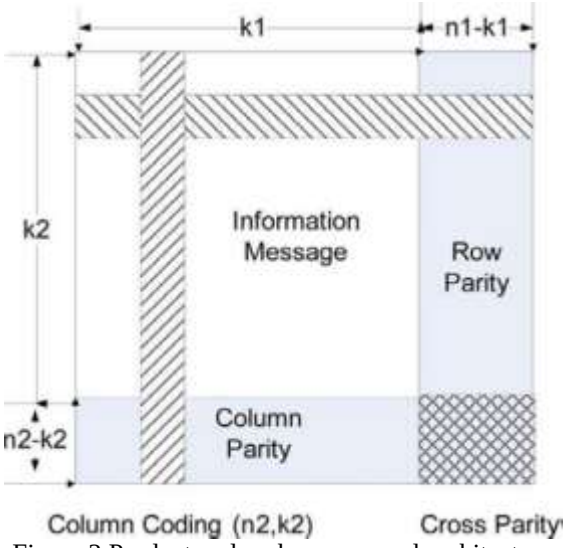

Figure 2.Product code scheme general architecture

An n-bit code word denoted as C, k-bit information vector denoted as i and k x n bit generator matrix denoted as G is considered. The purpose of n-bit code word C is to encode the k-bit information vector i and thus can be obtained by the product of i and matrix G which is given in (2).

$$C = i \,.\, G \tag{2}$$

The systematic format of the generator matrix for the linear code (15, 7, 5) is given in (3).

$$\begin{bmatrix} 1\,0\,0\,0\,0\,0\,0\,1\,0\,0\,1\,1\,1\,0\,1 \\ 0\,1\,0\,0\,0\,0\,0\,1\,1\;\;0\,0\,1\,1\,1\,0 \\ 0\,0\,1\,0\,0\,0\,0\,0\,1\,1\,1\,0\,0\,0\,1 \\ 0\,0\,0\,1\,0\,0\,0\,1\,0\,1\,1\,1\,0\,0\,0 \\ 0\,0\,0\,0\,1\,0\,0\,0\,1\,0\,1\,1\,1\,0\,0 \\ 0\,0\,0\,0\,0\,1\,0\,0\,0\,1\,0\,1\,1\,1\,0 \\ 0\,0\,0\,0\,0\,0\,1\,0\,0\,0\,1\,0\,1\,1\,1 \end{bmatrix} \tag{3}$$

This generator matrix can be obtained by first encoding the data array with size (k1 x k2) along the row (column) and then along the column (row). The cross parity matrix of size (n1 – k1) x (n2 - k2) which is present in the bottom right side of generator matrix is obtained by performing encoding operation along the row (column) and then along the column (row).

Here let hamming distance of C1 be d1, hamming distance of C2 be d2 and hamming distance of the product code C1C2 be d1d2. The number of error pattern can be enhanced by increasing the minimum weight assigned to each code. This error pattern can be corrected in the code array. For this purpose, the flash memory should have high error correcting capability. To provide this feature, a code which can handle multiple errors is proposed along any one of its dimension. In memory, the data will be stored in rows and it is necessary to use these stronger ECC schemes along the row dimension, by which burst and random errors can be dealt efficiently. Further, a long code word is used in this dimension so that better coding performance can be obtained.

4. RESULTS

The code LDPC is not a systematic one and decoding must be performed on the encoded information bits. These conditions are not necessary for fault tolerant system, due to the presence of noises and complication in its

operation. Figure 3 shows the encoder circuit for the linear code (15, 7, 5).

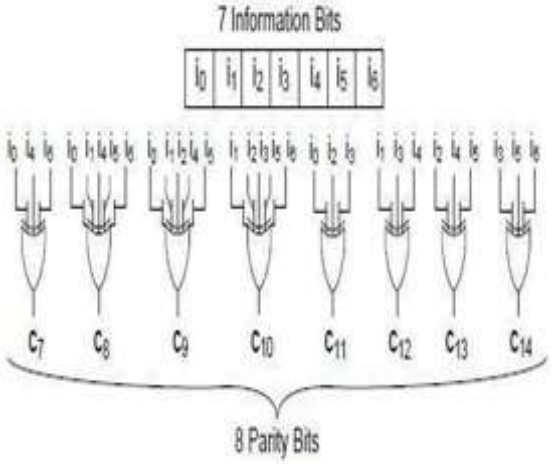

Figure 3.Encoder circuit for the linear LDPC code (15, 7, 5)

In this, i (i0 , i1, i2 ,....., i6) is defined as the information vector and c (c0,....., c6) is defined as the encoded vector. First the information vector is copied to the encoded vector and then the remaining encoded vectors which are denoted as the parity bits are considered as the linear sum of the information vector. The XOR gate used consists of two inputs; hence the encoder circuit nearly takes 22 XOR gates. Table 1 describes the area of EG-LDPC code on the basis of generator matrix. Each XOR gate produces one parity output to produce the code word which consists of seven information bit along with eighth parity bit.

Table 1.Area covered by encoder, decoder and corrector circuit on the basis of 2-i/p gate

Code	(15,7,5)	(63,37,9)	(275,175,17)
Encoder	45	501	3825
Decoder	22	355	6577
Serial corrector	19	83	331
Parallel corrector	285	5229	84405

The ECC based NAND MLC product code scheme is designed and its output waveform can be obtained by implementing on FPGA. Hardware description language such as Verilog and VHDL is commonly used for programming FPGA. After simulation process, the output waveform obtained is shown in figure 4.

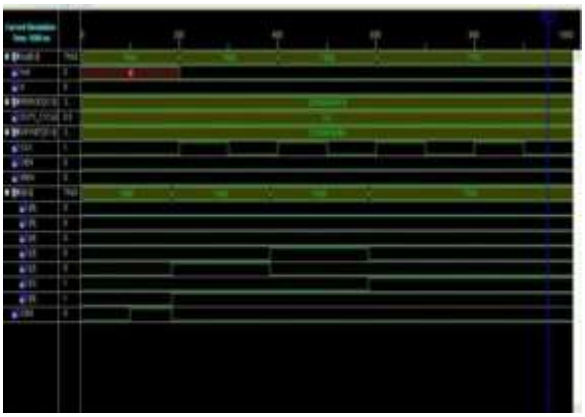

Figure 4.Simulation result of ECC based NAND MLC product code scheme of flash memory

4.1. RTL schematic

After obtaining simulation results, RTL schematic representation is generated. RTL schematic refers to the representation of design in terms of symbols. The symbol includes the logic gates, counters, adders etc. Figure 5 shows the RTL schematic of this proposed scheme.

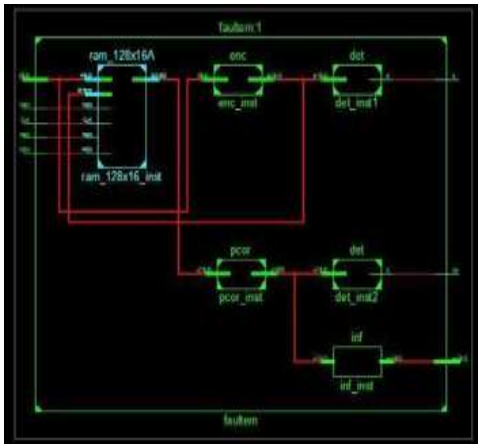

Figure 5.RTL schematic of the proposed scheme

5. CONCLUSION

In this, ECC technique is proposed on the basis of product code schemes. The hardware resources used to implement this technique get reduced in large amount. This is considered as one of the advantages of this technique. In this, fault tolerant system based on LDPC product coding scheme is designed that tolerates the error present in memory as well as the supporting logic circuit. Loss Less data transmission can be achieved using this technique. Additionally, other advantages offered by ECC based product code scheme are security, safety and flexibility. Due to these advantages, this technique is widely in use in

flash memories for real time transmission of data. The fault tolerant system is simulated based on its behavior and thus there is no data loss in MLC NAND for designing 128 x 16 flash memories.

REFERENCES

[1] Zhu Jun, Flash Memory Technology Development, International Conference on Solid-State and Integrated-Circuit Technology, China, 2001.

[2] N.Murali Krishna, FPGA Implementation of On-Chip Network, DJ Journal of Advances in Electronics and Communication Engineering, Vol. 4, No. 2, 2018, pp. 1-10, https://dx.doi.org/10.18831/djece.org/2018021001.

[3] Sanghyuk Jung and Yong Ho Song, Data Loss Recovery for Power Failure in Flash Memory Storage Systems, Journal of Systems Architecture, Vol. 61, No. 1, 2015, pp. 12-27, https://dx.doi.org/10.1016/j.sysarc.2014.11.002.

[4] Yi Wang, Lisha Dong, Zhong Ming, Yong Guan and Zili Shao, Virtual Duplication and Mapping Prefetching For Emerging Storage Primitives In NAND Flash Memory Storage Systems, Microprocessors and Microsystems, Vol. 50, 2017, pp. 54-65, https://dx.doi.org/10.1016/j.micpro.2017.02.008.

[5] Yuhui Deng and Jipeng Zhou, Architectures and optimization Methods of Flash Memory based Storage Systems, Journal of Systems Architecture, Vol. 57, No. 2, 2011, pp. 214-227, https://dx.doi.org/10.1016/j.sysarc.2010.12.003.

[6] Do-Bin Kim, Dae Woong Kwon, Seunghyun Kim, Sang-Ho Lee and Byung-Gook Park, Program Scheme using Common Source Lines in Channel Stacked NAND Flash Memory with Layer Selection by Multilevel Operation, Solid-State Electronic, Vol. 140, 2018, pp. 46-50, https://dx.doi.org/10.1016/j.sse.2017.10.014.

[7] Jan Petervan Zandwijk, A Mathematical Approach to NAND Flash-Memory Descrambling and Decoding, Digital Investigation, Vol. 12, 2015, pp. 41-52, https://dx.doi.org/10.1016/j.diin.2015.01.003.

[8] Soojun Im and Dongkun Shin, ComboFTL: Improving Performance and Lifespan of MLC Flash Memory using SLC Flash Buffer, Journal of Systems Architecture, Vol. 56, No. 12, 2010, pp. 641-653, https://dx.doi.org/10.1016/j.sysarc.2010.09.005.

[9] Gyu Sang Choi, Ingyu Lee, Mankyu Sung and Choongjae Im, A Hybrid SSD with PRAM and NAND Flash Memory, Microprocessors and Microsystems, Vol. 36, No. 3, 2012, pp. 257-266, https://dx.doi.org/10.1016/j.micpro.2011.11.001

[10] Antonios Prodromakis, Stelios Korkotsides and Theodore Antonakopoulos, MLC NAND Flash Memory: Aging Effect and Chip/Channel Emulation, Microprocessors and Microsystems, Vol. 39, No. 8, 2015, pp. 1052-1062, https://dx.doi.org/10.1016/j.micpro.2015.06.007.

[11] M.Fabiano, M.Indaco, S.Di Carlo and P.Prinetto, Design and Optimization of Adaptable BCH Codecs for NAND Flash Memories, Microprocessors and Microsystems, Vol. 37, No. 4-5, 2013, pp. 407-419, https://dx.doi.org/10.1016/j.micpro.2013.03.002.

[12] Nellipudi Surekha, Implementation of Low-Complexity LDPC Codes for Lossless Applications, International Journal of Engineering Research and Science & technology, Vol. 3, No. 4, 2014, pp. 294-301.

[13] Robert Redinbo, Real number convolutional code correction and reliability calculations in fault-tolerant systems, IEEE International Symposium on Information Theory, Canada, 1995.

[14] Sribarki Srinath and Bantupalli Divakar, Hardware Complexity Reduced ECC with FSD in MLC NAND Flash Memories, International Journal of Advanced Research in Electrical, Electronics and Instrumentation

Engineering, Vol. 2, No. 10, 2013, pp. 5101-5110.

[15] Beomkyu Shin, Seokbeom Hong, Hosung Park, Jong-Seon No and Dong-Joon Shin, New Decoding Scheme for LDPC Code based on Simple Product Code Structure, Journal of Communications and Network, Vol. 17, No. 4, 2015, pp. 351-361.

[16] Debao Wei, Libao Deng, Mengqi Hao, Liyan Qiao and Xiyuan Peng, A joint-LDPC Decoding Scheme based on Retention Error Characteristics for MLC NAND Flash Memory, Microprocessors and Microsystems, Vol. 60, 2018, pp. 65-76, https://dx.doi.org/10.1016/j.micpro.2018.04.001.

[17] Vikram Arkalgud Chandrasetty and Syed Mahfuzul Aziz, Overview of LDPC Codes, Resource Efficient LDPC Decoders, 2018, pp. 5-10, https://dx.doi.org/10.1016/B978-0-12-811255-7.00002-2.

[18] Vikram Arkalgud Chandrasetty and Syed Mahfuzul Aziz, Prospective LDPC Applications, Resource Efficient LDPC Decoders, 2018, pp. 117-122, https://dx.doi.org/10.1016/B978-0-12-811255-7.00008-3.

[19] Seongkwon Jeong and Jaejin Lee, LDPC Product Coding Scheme with Extrinsic Information for Bit Patterned Media Recoding, American Institute of Physics, Vol. 7, No. 5, 2017, https://dx.doi.org/10.1063/1.4978219 .

[20] Aya Fukami, Saugata Ghose, Yixin Luo, Yu Cai and Onur Mutlu, Improving the Reliability of Chip-off Forensic Analysis of NAND Flash Memory Devices, Digital Investigation, Vol. 20, 2017, pp. S1-S11, https://dx.doi.org/10.1016/j.diin.2017.01.011

[21] Haozhi Ma, Liyang Pan, Changlai Song, Zhongyi Gao, Dong Wu and Jun Xu, Word Line Program Disturbance based Data Retention Error Recovery Strategy for MLC NAND Flash, Solid-State Electronics, Vol. 109, 2015, pp. 1-7, https://dx.doi.org/10.1016/j.sse.2015.03.009

[22] Wei Wang, Tao Xie, Antoine Khoueir and Youngpil Kim, Reducing MLC Flash Memory Retention Errors through Programming Initial Step Only, IEEE Symposium on Mass Storage System and Technologies, USA, 2015, pp. 1-8.

[23] Meng Zhang, Fei Wu, Xubin He, Ping Huang and Shunzhuo Wang, Changsheng Xie, REAL: A Retention Error Aware LDPC Decoding Scheme to Improve NAND Flash Read Performance, IEEE Symposium on Mass Storage System and Technologies, USA, 2017.

[24] Meng Zhang, Fei Wu, Yajuan Du, Chengmo Yang, Changsheng Xie and Jiguang Wan, CooECC: A Cooperative Error Correction Scheme to Reduce LDPC Decoding Latency in NAND Flash, IEEE International Conference on Computer Design, USA, 2017.

[25] Tseng-Yi Chen, Yuan-Hao Chang, Yuan-Hung Kuan and Pi-Cheng Hsiu, A Pattern-Aware Write Strategy to Enhance the Reliability of Flash-Memory Storage Systems, Proceedings of the Symposium on Applied Computing, USA, 2017, pp. 1460-1466, https://dx.doi.org/10.1145/3019612.3019680 .

[26] Zhiqiang Cui, Zhongfeng Wang and Xinming Huang, Multilevel Error Correction Scheme for MLC Flash Memory, IEEE International Symposium on Circuits and Systems, Australia, 2014.

[27] Haochuan Song, Chuan Zhang, Shunqing Zhang and Xiaohu You, Polar Code-Based Error Correction Code Scheme for NAND Flash Memory Applications,IEEE International Conference on Communications and Signal Processing, China, 2016.

[28] T.Chen, Y.Hsiao, Y. Hsing and C.Wu, An Adaptive-Rate Error Correction Scheme for NAND Flash Memory, IEEE VLSI Test Symposium, USA, 2009, pp. 53-58.

[29] Yu Cai, Gulay Yalcin, Onur Mutlu, Erich F.Haratsch, Adrian Cristal, Osman S.Unsaland and Ken Mai, Flash Correct-and-Refresh: Retention-Aware Error Management for Increased Flash Memory Lifetime, IEEE International Conference on Computer Design, Canada, 2012, pp. 94-101.

[30] Yu Cai, Gulay Yalcin, Onur Mutlu, Erich F.Haratsch, Osman Unsal, Adrian Cristal and Ken Mai, Neighbor-Cell Assisted Error Correction for MLC NAND Flash Memories, ACM international conference on Measurement and modeling of computer systems, USA, 2014, pp. 491-504, https://dx.doi.org/10.1145/2637364.2591994.

[31] Chanha Kim, Chanik Park, Sungjoo Yoo and Sunggu Lee, Extending Lifetime of Flash Memory using Strong Error Correction Coding, IEEE Transactions on Consumer Electronics, Vol. 61, No. 2, 2015, pp. 206-214.

[32] Helia Naeimi and Andre DeHon, Fault Secure Encoder and Decoder forNanoMemory Applications, IEEE Transactions on Very Large Scale Integration Systems, Vol. 17, No. 4, 2009, pp. 473-486.

[33] P.Kannan, S.Karthiga, Euclidean Gemetry LDPC Codes for Error Correction and Error Detection in Memory Application, International Journal of Computer Science and Mobile Computing, Vol. 3, No. 4, 2014, pp. 964-972.

[34] C.Yang, Y.Emre, C.Chakrabarti and T.Mudge, Flexible Product Code-Based ECC Schemes for MLC NAND Flash Memories, IEEE Workshop on Signal Processing Systems, Lebanon, 2011, pp. 255-260.

[35] Chengen Yang, Yunus Emre and Chaitali Chakrabarti, Product Code Schemes for Error Correction in MLC NAND Flash Memories, IEEE, Transactions on Very Large Scale Integration Systems, Vol. 20, No. 12, 2012, pp. 2302-2314.