Improving Min-sum LDPC Decoding Throughput by Exploiting Intra-cell Bit Error Characteristic in MLC NAND Flash Memory

Wenzhe Zhao*, Hongbin Sun*, Minjie Lv*, Guiqiang Dong†, Nanning Zheng*, and Tong Zhang‡
* Institute of AI&R, Xi'an Jiaotong University, P.R. China
† Skyera Inc., USA
‡ ECSE Department, Rensselaer Polytechnic Institute, USA

***Abstract*—Multi-level per cell (MLC) technique significantly improves storage density, but also poses new challenge to data integrity in NAND flash memory. Therefore, low-density parity-check (LDPC) code and soft-decision memory sensing have become indispensable in future NAND flash-based solid state drive design. However, these more powerful technologies inevitably increase the memory read latency and hence degrade the decoding throughput. Motivated by intra-cell unbalanced bit error probability and data dependency in MLC NAND flash memory, this paper proposes two techniques, i.e. intra-cell data placement interleaving and intra-cell data dependency aware min-sum decoding, to effectively improve the throughput of LDPC decoding. Experimental results show that, the proposed techniques used in an integrated way can improve the LDPC decoding throughput by up to 85% when the MLC NAND flash chip is heavily cycled, compared with conventional design practice.**

I. Introduction

Due to the aggressive technology scaling and multi-level per cell (MLC) technique, the storage reliability and performance of NAND flash memory have been largely degraded [1]. Therefore, low-density parity-check (LDPC) code [2] that has more powerful error correction capability is believed to be indispensable for future solid state drive (SSD). Recently, how to employ LDPC code in SSD has attracted much attention and is extensively evaluated. Prior works have demonstrated that, with soft-decision memory sensing, the LDPC code can achieve the noticeable coding gain over conventional BCH code [3], [4].

However, since NAND flash memory read latency is proportional to the memory sensing precision, the simple use of soft-decision memory sensing tends to directly cause the significantly increased NAND flash memory read latency, leading to unacceptable storage system read response time degradation [5]. Therefore, the practical use of LDPC in SSDs should employ a two-step trial-and-error procedure [6]: upon a read request, SSDs always first execute hard-decision memory sensing and decoding, and only when hard-decision LDPC code decoding fails, execute soft-decision memory sensing and decoding. Although the two-step trial-and-error procedure significantly reduces the soft-decision memory sensing induced read latency, the decoding throughput degradation is still noticeable especially when the NAND flash memory chip is heavily cycled. As the throughput of LDPC decoding has significant impact on the performance of overall storage system, great effort has been made to investigate the throughput improvement techniques for LDPC decoding [7], [8].

978-1-4799-5671-5/14/$31.00

Previously proposed LDPC decoders generally assume all the data bits are independent to each other and have the equal error probability. However, MLC NAND flash memory actually possesses interesting intra-cell bit error characteristics: (1) the bit error probability in one flash memory cell is quite unbalanced, that the lower page bit is much more prone to errors. (2) the probability that one bit is prone to error could be largely dependent on data value of the other bit in the same flash memory cell. Since the min-sum LDPC decoding is essentially a belief propagation procedure, being aware of these intra-cell bit error characteristics is potentially beneficial for LDPC decoding design. This useful observation motivates us to investigate possible design techniques that can exploiting intra-cell bit error characteristics for improving the throughput of LDPC decoding. Firstly, we propose to use intra-cell data placement interleaving to put the upper page bit and lower page bit that belong to the same flash cell into one LDPC codeword. This can effectively enable us to leverage the aforementioned intra-cell data dependency as additional belief information during LDPC decoding. Accordingly, we further propose an intra-cell data dependency aware min-sum LDPC decoding technique to exploit the additional belief information during the decoding procedure and improve the decoding throughput. The effectiveness of proposed techniques has been demonstrated in a case study of 2b/cell MLC NAND flash memory. By using the proposed techniques in an integrated way, the LDPC decoding throughput can be improved by up to 85% when the MLC NAND flash chip is heavily cycled.

II. Background and Motivations

A. Program/read operation of MLC NAND flash memory

Compared with single-level cell (SLC), MLC NAND flash allows more bits to be stored in a single cell. Hence, MLC NAND flash cell provides much less margin between two adjacent storage states and is thereby much more prone to

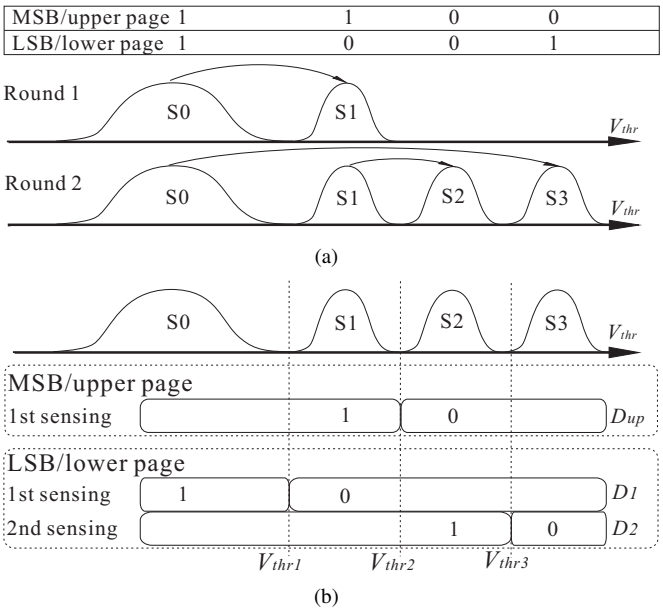

Fig. 1. Program and read operations for 2 bits/cell NAND flash memory (a) program operation, (b) read operation.

errors. In order to reduce its raw bit error probability, Gray-code is commonly used when mapping information bits to voltage levels in the MLC cells, so that neighboring levels only differ in one bit. Fig.1 illustrates an example for 2b/cell NAND flash memory. The voltage levels S0, S1, S2, and S3 denote information bits '11', '10', '00', and '01' respectively. And the most significant bit (MSB) and the least significant bit (LSB) are associated to upper page and lower page respectively.

As more than one bit are stored in a single cell, the program/read operations of MLC NAND flash are also complicated. As illustrated in Fig. 1(a), 2 bits are associated to the four read threshold distribution, and the multilevel program is achieved in two distinct rounds, one for each bit to be stored. In the first round, the LSB is programmed. If the bit is '1', the voltage threshold of the cell V_{thr} does not change and the cell remains in the erased state S0. If the bit is '0', the threshold is increased until it reaches the S1 state. In the second round, the MSB is programed. If the bit is '1', the voltage threshold does not change and the cell remains either in the S0 state or the S1 state, depending on the value of LSB. When MSB is '0', the voltage threshold is programed as follows: 1) if the cell remains in S0 state, then the voltage threshold is increased to S3 state. 2) if the cell was programed to S1 state, then the voltage threshold reaches S2 state.

During the read operation, if the sensing threshold is lower than V_{TH}, the cell denotes bit '1'; else, the cell denotes bit '0'. As shown in Fig. 1(b), when reading upper page data from 2b/cell NAND flash memory, the read voltage threshold is set as V_{thr2}, and then upper page data can be read out by flash memory sensing. For lower page data, the flash cell needs to be sampled two times by changing the read voltage threshold as V_{thr1} and V_{thr3}, representing the result as $D1$ and $D2$ respectively. And finally, the lower page data can be calculated as $D_1 + \overline{D_2}$

The above-mentioned read operation is so called hard-decision memory sensing, which uses only one quantization level between two adjacent storage states; While if more than one quantization levels are used between two adjacent storage states, it is called soft-decision memory sensing. As belief propagation algorithm inherently prefers high-precision information, the use of soft-decision memory sensing is able to significantly improve the error correction strength of LDPC decoding. However, since NAND flash memory read latency is proportional to the memory sensing precision, soft-decision memory sensing will directly cause the increased NAND flash memory read latency, leading to read response time degradation in storage system. Therefore, it is reasonable for current LDPC-based SSDs to employ a two-step trial-and-error procedure, and only execute soft-decision memory sensing and decoding when necessary.

B. Intra-cell bit error characteristics

Due to the specific data structure and operation inside memory cell, MLC NAND flash memory actually possesses very useful intra-cell bit error characteristics that can be exploited for LDPC decoding design. Firstly, due to the Gray mapping, the bit error probability of lower page is significantly higher than lower page, which is referred to as *intra-cell unbalanced bit error probability* in this paper. We can intuitively see intra-cell unbalanced bit error probability from Fig.1. For upper page, there is only one adjacent threshold voltage level that can cause bit $1 \rightarrow 0$ or $0 \rightarrow 1$; while for lower page, there are two. In addition, we can give an approximate probability estimation as follows.

As the great majority of errors in MLC NAND flash memory are caused by the obscure area between two adjacent levels and hence the dominate errors are single-bit errors after Gray mapping, we can safely assume one cell reading failure only results in a single-bit error. Moreover, in flash device cell, as all levels are tuned to have almost the same level error rate (LER), we simply assume all levels have the same error rate p, and each bit has equal *a priori* probability of being 0 and 1, i.e. each cell has equal probability to stay in every storage level. Denote the expected (correct) level/state of a cell as s and this cell is sensed to stay in level s'. For 2b/cell flash with Gray mapping, the bit error rate (BER) of upper page is

$$p_u = \frac{1}{4}(p(s'=2|s=1) + p(s'=1|s=2)) = \frac{p}{2} \quad (1)$$

and the BER of lower page is

$$p_l = \frac{1}{4}(p(s'=1|s=0) + p(s'=0|s=1) + p(s'=3|s=2) + p(s'=2|s=3)) = p \quad (2)$$

According to the above analysis, the BER of lower page is almost twice larger than upper page, leading to significantly unbalanced bit error probability in each MLC NAND flash cell.

Besides intra-cell unbalanced bit error probability, the probability that one bit is prone to error could be largely dependent

on the data value of the other bit in the same flash memory cell, which is referred to as *intra-cell data dependency*. Again, we take the 2b/cell flash memory shown in Fig.1 as example. The following three intra-cell data dependency are exploited in following LDPC decoding design: (1) When data value of lower page bit equals to '1', the probability that the corresponding upper page bit is correct will be very large. (2) When the lower page bit falls to the obscure area between two adjacent levels, the probability that the corresponding upper page bit is correct will be very large. While (3) when the upper page bit falls to the obscure area between two adjacent levels, data value of the corresponding lower page bit has a very large probability to be to '0'. NAND flash memory with more than 2 bits per cell also has the similar intra-cell data dependency characteristics.

The intra-cell bit error characteristics exploited in this paper can be summarized as:

- intra-cell unbalanced bit error probability: The bit error probability of the upper page and lower page in one flash memory cell is largely unbalanced, that the lower page is much higher than upper page.
- intra-cell data dependency: The probability that one bit is prone to error could be largely dependent on the data value of the other bit in the same flash memory cell.

Previously proposed LDPC decoders generally assume the equal error probability for all bits and omit the data dependency between upper page bits and lower page bits. However, since the LDPC decoding is essentially a belief propagation procedure, it is natural for us to exploit the use of these intra-cell bit error characteristics as additional belief input to further improve the throughput of LDPC decoding and hence reduce the read response time of flash-based solid state storage system.

III. Proposed Techniques

This section presents two techniques to exploit intra-cell bit error characteristics for the throughput improvement of LDPC decoding design. The description is base on the case of 2b/cell NAND flash. However, these techniques can be similarly extended to other MLC NAND flash memory.

A. Intra-cell data placement interleaving

Whether we can exploit intra-cell bit error characteristics in LDPC decoding is largely dependent on the data organization in flash memory page. Although we know intra-cell data dependency is beneficial to LDPC dedoding, the previously proposed decoding algorithm stored the codewords in each separate upper or lower page, and there is no interaction between upper page bit and lower page bit during the decoding procedure. To exploit the intra-cell data dependency, firstly, we need to put the upper page bit and lower page bit that belong to the same cell into one codeword. Accordingly, we propose a intra-cell data placement interleaving techniques to enable the use of data dependency in LDPC decoding.

The proposed intra-cell data placement interleaving technique in detail is illustrated in Fig. 2. To make one codeword contain both the upper page bit and lower page bit belonging to the same flash memory cell, we divide the codeword into several sub-blocks, and place the sub-blocks into upper page and lower page alternately in interleaved form. In particular, the size of each sub-block is chosen to be half of the parallel processing bits of LDPC decoder, thus the decoder could process intra-cell mutually dependent data simultaneously.

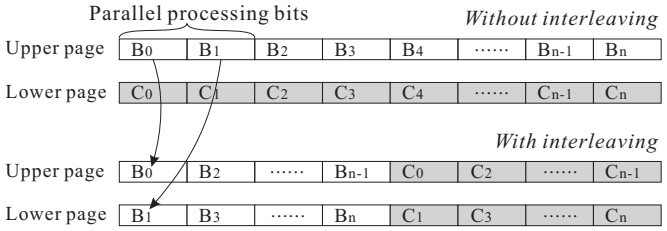

Fig. 2. upper/lower page interleaving diagram in 2 bits/cell NAND flash memory.

To support the proposed intra-cell data placement interleaving only requires slight modification at architecture level. The major overhead is the increase of memory sensing delay during read operation. As for each codeword read, three memory sensing operations are required, i.e. the first memory sensing for upper page bits and the second&third for lower page bits. However, since the data transfer time is usually longer than memory sensing time in current commercial products, we can 'hide' the second&third memory sensing delay by conducting the memory sensing during the data transfer for upper page bits, as shown in Fig. 3. The overlap of consecutive flash memory read can be easily realized by using cache read mode in commercial flash chip. In addition, in case that flash memory chip integrates high speed I/O interface and data transfer becomes much faster, we can set logical page from system level twice the size of physical page, containing both the upper and lower pages, to eliminate the memory sensing induced overhead.

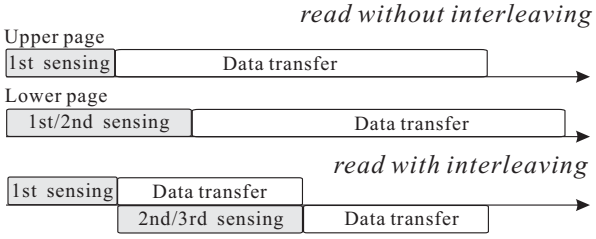

Fig. 3. Illustration of overlap read operation of MLC NAND flash memory.

In addition, data placement interleaving not only provides extra belief information for LDPC decoding, but also improves the decoding throughput itself. This is mainly because the interleaving induced bit error probability average can reduce the probability of high-precision soft-decision memory sensing being invoked. As without data placement interleaving, the bit error probability of LDPC codeword in lower page is much higher than upper page and hence tends to frequently invoke the very cost soft-decision memory sensing, leading to the

overall system throughput degradation especially when the flash memory chip is heavily cycled.

B. Intra-cell data dependency aware min-sum decoding

Inherently, min-sum LDPC decoding algorithm utilizes the iterative belief propagation to correct errors. And conventional min-sum LDPC decoding design merely relies on the belief propagation between variable bit and check bit. By employing intra-cell data placement interleaving, we can have additional belief information inside the variable bits, as the error probability of one bit largely depends on the value of the other bit in the same cell. Accordingly, we propose a intra-cell data dependency aware min-sum LDPC decoding technique that is able to improve both the error correction strength and the decoding convergence speed.

In essence, the procedure of min-sum decoding is to compute the maximum posteriori probability for each codeword bit D_i, which is the probability that the i-th codeword bit is '1' if all parity check constraints are satisfied. Fig. 4 illustrates the decoding procedure of the proposed intra-cell data dependency aware decoding technique. The initial probabilities R of variable bits are stored in upper and lower pages, alternatively. Firstly, while estimating the satisfied probability D_i of the variable bit i, all constraint bits that connect to this variable bit provide probabilities that variable bit i is correct. Then the decoder checks whether all the hard-decision results d of the estimated variable bits are satisfied. If not, the probability M that constraint bits cause information bits should be updated by the satisfied probability of variable bits, and start another iteration. The major difference between the proposed and conventional min-sum LDPC decoding can be summarized as follows. Firstly, when estimating D_i, the corresponding variable bit R_c provides additional probability K_i that variable bit i is correct, due to the intra-cell data dependency. Hence, K_i is considered when calculating L_i in the proposed decoding strategy. Secondly, when estimating $M_{j,i}$, which constraint bit j causes variable bit i, K_i also provides extra information that helps to improve the estimation accuracy. The proposed technique invokes K_i to realize accurate $M_{j,i}$.

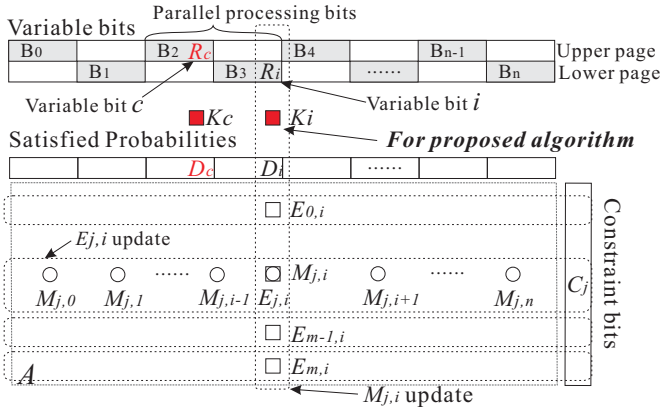

Fig. 4. $E_{j,i}$ and $M_{j,i}$ update diagram in the proposed Min-sum LDPC decoding algorithm.

To simplify the calculation, all the probabilities are represented as log likelihood ratios (LLR) to transform the multiplication of probabilities into add operation. If the LLR of probability is smaller than 0, the bit is more likely to be '1', otherwise the bit is more likely to be '0'. The value of additional probability K_i depends on the variable bit R_c and the hard-decision of variable bit satisfied probability d_c in the corresponding upper or lower page. According to the relationship between R_c and d_c, the K_i calculation has the following three possible cases:

Case 1: For the variable bit i in upper page, if R_c and d_c are both smaller than 0, it indicates that the variable bit c is probably equal to bit '1'. Thus, the initial variable bit R_i is probably correct. We set K_i as αR_i to enhance the probability that the initial variable bit is correct, where α is usually set as 0.75.

Case 2: For the variable bit i in upper page, if the sign of R_c is different with d_c, it also indicates the variable bit c is probably equal to bit '1', but the probability is less than Case 1. Thus, we set K_i as $0.5\alpha R_i$.

Case 3: For the variable bit i in lower page, if the sign of R_c is different with d_c, it indicates the corresponding variable bit c is probably in S1 and S2, which is shown in Fig. 1. In this scenario, the variable bit i is probably equal to bit '0'. Thus, we set K_i as a positive value to make the variable bit i prone to bit '0'.

For concise description, K_i is calculated as

$$K_i = \begin{cases} \alpha R_i & \text{if } R_c < 0 \&\& d_c < 0, i \in \textit{upper page} \\ 0.5\alpha R_i & \text{else if } R_c d_c < 0, i \in \textit{upper page} \\ 2 & \text{else if } R_c d_c < 0, i \in \textit{lower page} \\ 0 & \text{else} \end{cases}$$

In summary, the intra-cell data dependency aware min-sum LDPC decoding flow can be described as follows:

Step 1: Set $M_{j,i} = R_i$, for each $j \in codeword$.

Step 2: Compute the variable-to-check message $E_{j,i}$ for each parity-check bit with $M_{j,i}$, by

$$E_{j,i} = \alpha(\prod_{i'} sign(M_{j,i'}))Min_{i'}|M_{j,i'}|. \quad (3)$$

Step 3: Calculate the check-to-variable message $M_{j,i}$ for each variable bit with $E_{j,i}$ and K_i, by

$$M_{j,i} = r_i + \sum_{j' \in A_i, j' \neq j} Ej', i + K_i, \quad (4)$$

where A is the none-zero node of parity-check matrix.

Step 4: Concurrent with step 3, compute the LLR message D_i for each variable bit, and check whether all bits d are satisfied. D_i is expressed as follows:

$$D_i = R_i + \sum_{j' \in A_i} Ej', i + K_i. \quad (5)$$

If all the bits are satisfied, decoding is successful. Otherwise, go to Step 2 to start a new decoding iteration until the maximum iteration number is reached.

By exploiting the intra-cell data dependency in hard-decision LDPC decoding, the proposed technique can reduce average iteration number and meanwhile increase error correction strength. Moveover, due to the improved error correction strength, the frequency of soft decision memory sensing can be largely reduced, leading to significantly improved LDPC decoding throughput especially when the flash memory is heavily cycled. The overhead of proposed intra-cell data dependency aware technique is mainly due to the computation of K. Since K can be easily calculated by R and d, both of which already exist in conventional min-sum decoding, the computation of K only requires additional small algebra logic and its overhead tends to be negligible.

IV. Experimental Results

A. Measurement of intra-cell unbalanced bit error probability

To quantitatively evaluate the throughput improvement of proposed techniques demands the availability of MLC NAND flash memory bit error probability and intra-cell unbalanced bit error probability statistics. Using commercial sub-22nm MLC NAND flash memory chips, we firstly carry out bit error probability statistics measurements following the same experiment flow as we did in our prior work [6].

Fig. 5 shows the distribution histogram of page raw bit error probabilities under four different P/E cycling numbers. The measurement results show that, with the increase of P/E cycling, the average page raw bit error probability in MLC NAND flash memory increases significantly. Moreover, the raw bit error probabilities among different pages vary largely, especially when the NAND flash chip comes to the end of its lifetime.

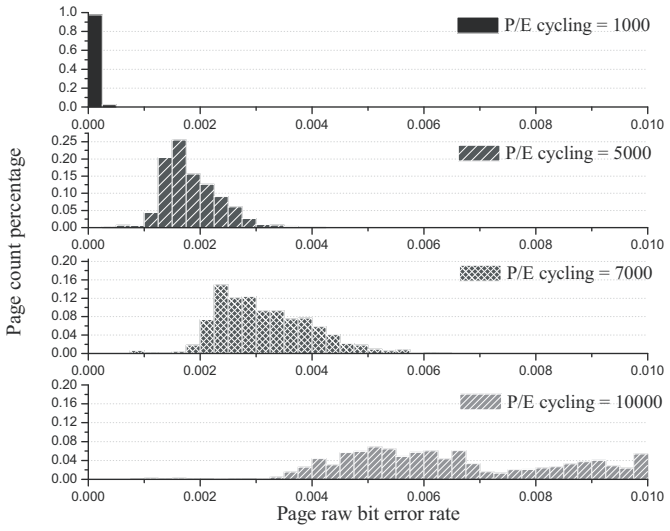

Fig. 5. Distribution histogram of measured raw bit error probabilities using sub-22nm MLC NAND flash memory.

We further conduct the bit error probability statistics in lower and upper pages to analysis the intra-cell unbalanced bit error probability. As shown in Fig.6. the raw bit error probability of lower page is significantly different from that of upper page. Moreover, the difference increases with the PE cycling number, and it reaches up to 76% at the PE cycling of 9000. The experimental results clearly justify the previous analysis for intra-cell unbalanced bit error probability in MLC NAND flash memory.

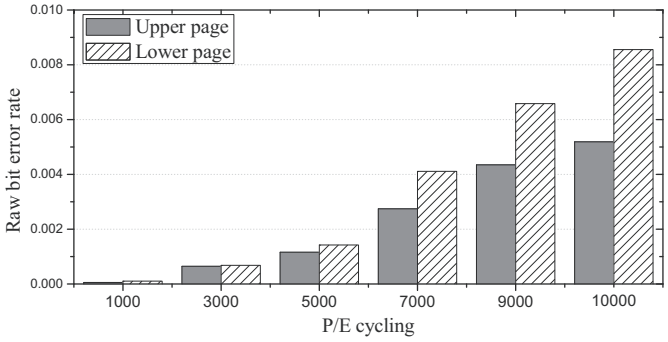

Fig. 6. Measured raw bit error probability comparison of upper and lower pages using sub 22nm MLC NAND flash memory.

B. Throughput improvement by proposed techniques

To evaluate the throughput improvement by proposed techniques, we use the MLC NAND flash memory model that is developed in our prior work [9]. In addition, we use the rate-8/9, length-2KB, QC-LDPC codes with the column weight cw of 4. We assume the codewords pass through diverse AWGN (adding white godson noise) channels. For the purpose of comparison, we set the baseline scenario as conventional min-sum LPDC decoding without exploiting any intra-cell bit error characteristics. And Fig. 7 shows the hard-decision decoding failure rate comparison between the baseline and proposed intra-cell data dependency aware min-sum LDPC decoding at different raw bit error rate. Experimental results show that, by leveraging the intra-cell data dependency, the proposed LPDC decoding technique has the much lower hard-decision decoding failure rate than baseline. In particular, the error correction strength of proposed decoding technique with hard-decision memory sensing (codeword precision is 1) is even comparable to the baseline algorithm with soft-decision memory sensing when the codeword precision is 2.

Besides the increase of error correction strength, the hard-decision decoding convergence speed of proposed LDPC

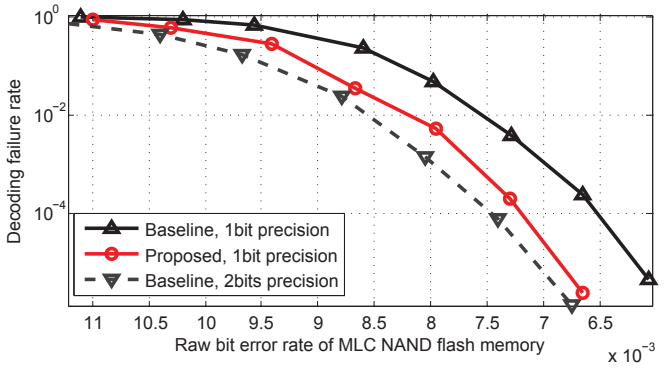

Fig. 7. Hard-decision LDPC decoding failure rate comparison between the baseline and proposed technique.

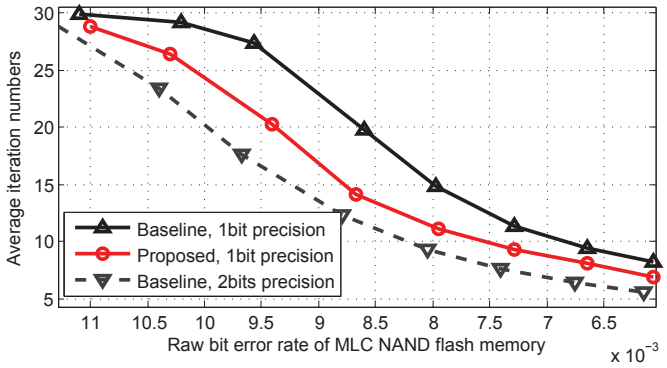

Fig. 8. Average iteration number comparison during hard-decision LDPC decoding between the baseline and proposed technique.

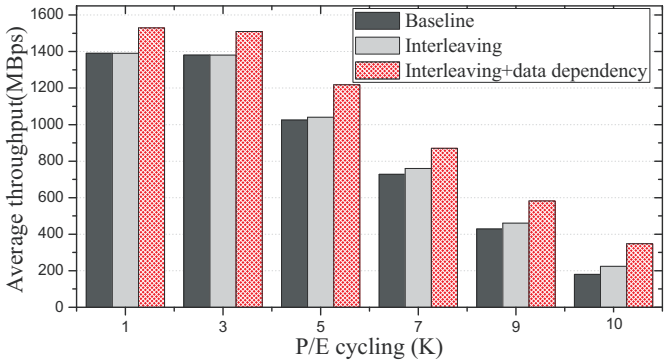

Fig. 9. Simulated throughput comparison between baseline and proposed techniques under different P/E cycling.

decoding is also improved. The average iteration number comparison at different raw bit error rate is illustrated as Fig.7, which clearly suggests that the proposed technique demands the much lower iteration numbers than baseline.

By employing the proposed techniques to hard-decision LDPC decoding, the decoding throughput tends to be higher due to the reduction of average iteration numbers. Moreover, the increased error correction strength during hard-decision decoding can reduce the probability of high-precision soft-decision memory sensing being invoked and further improve the decoding throughput. The throughput of hard-decision LDPC decoder Th_{avg} can be calculated as

$$Th_{avg} = \int_0^1 \frac{f_q \cdot l_{inf} \cdot (1 - dfr(x))}{c_{iter}(x) \cdot num + c_{load}} d(x) \qquad (6)$$

Where f_q, l_{inf}, dfr, c_{iter}, num, c_{load} and x denote the working frequency, information length, decoding failure rate, average iteration number, clock cycle/iteration, decoding preparation time and the NAND flash raw bit error rate, respectively. To evaluate the decoding throughput, we employ the same min-sum LDPC decoder as published in [10], which works at 100MHz and consumes 76 cycles for one iteration, 2 clock cycles for preparation. In case of hard-decision decoding failure, the soft-decision memory sensing and decoding will be invoked, consuming another 100μs delay penalty. According to the raw bit error probability statistics under different P/E cycling as shown in Fig. 5, we can simulate the corresponding average LDPC decoding throughput and illustrate the simulation result in Fig. 9. The experimental results clearly demonstrates that the throughput improvement of proposed techniques increases when the P/E cycling number is getting higher. In particular, the throughput improvement of proposed LDPC decoding can reach up to 85% when the MLC NAND flash chips are heavily cycled and their P/E cycling number approach to 10K.

V. Conclusion

This work is motivated by the observation that the bit error probability of upper page bit and lower page bit in one cell is largely unbalanced and has data dependency on each other in MLC NAND Flash memory. By exploiting these intra-cell bit error characteristics, we propose two techniques that are able to reduce the iteration number, increase the error correction strength and alleviate the invoke of soft-decision memory sensing, leading to significantly improved decoding throughput for LDPC code. Experimental results demonstrate the effectiveness of proposed design techniques that, compared with the conventional min-sum LDPC decoding, the throughput improvement can reach up to 85% when the MLC NAND flash memory chips are heavily cycled.

Acknowledgment

This research is funded in part by grants from the National Natural Science Foundation of China (No. 61274028) and the National High Technology Research & Development Program of China (No. 2011AA010405).

References

[1] C. Compagnoni, S. Spinellet, and et.c, "First evidence for injection statistics accuracy limitations in NAND Flash constant-current fowler-nordheim programming," *IEDM Tech. Dig*, pp. 165–168, 2007.

[2] D. J. C. MacKay, "Good error-correcting codes based on very sparse matrices," *IEEE Transactions on Information Theory*, vol. 45, pp. 399–431, Mar. 1999.

[3] N. Duann, "Error correcting techniques for future NAND flash memory in SSD applications," *Technology*, p. 1, 2009.

[4] Z. Wang, Z. Cui, and J. Sha, "VLSI design for low-density parity-check code decoding," *IEEE Circuits and Systems Magazine*, vol. 11, pp. 52–69, 2011.

[5] W. Zhao, G. Dong, H. Sun, and et.c, "Reducing latency overhead caused by using LDPC codes in NAND flash memory," *EURASIP Journal on Advances in Signal Processing*, vol. 2012, pp. 1–9, 2012.

[6] K. Zhao, W. Zhao, H. Sun, and et.c, "LDPC-in-SSD: making advanced error correction codes work effectively in solid state drives," in *Proceedings of USENIX Conference on File and Storage Technologies (FAST)*, 2013.

[7] J. Kim, J. Cho, and W. Sung, "A high-speed layered min-sum LDPC decoder for error correction of NAND flash memories," in *Proceedings of IEEE 54th International Midwest Symposium on Circuits and Systems (MWSCAS)*, Aug 2011, pp. 1–4.

[8] D. Lee and W. Sung, "Estimation of NAND flash memory threshold voltage distribution for optimum soft-decision error correction," *IEEE Transactions on Signal Processing*, vol. 61, pp. 440–449, 2013.

[9] G. Dong, N. Xie, and T. Zhang, "On the use of soft-decision error-correction codes in NAND flash memory," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 58, pp. 429–439, 2011.

[10] H. Zhong and T. Zhang, "Block-LDPC: a practical LDPC coding system design approach," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 52, pp. 766–775, 2005.