Heterogeneous PCM array architecture for reliability, performance and lifetime enhancement

Taehyun Kwon*†, Muhammad Imran‡, Jung Min You‡ and Joon-Sung Yang*

*Department of Semiconductor and Display Engineering and ‡Department of Electrical and Computer Engineering,
Sungkyunkwan University, Suwon, Korea
†System LSI Division, Samsung Electronics, Korea
{th.kwon, imran, yugura, js.yang}@skku.edu

Abstract—**Conventional DRAM and flash memory are reaching their scaling limits thus motivating research in various emerging memory technologies as a potential replacement. Among these, phase change memory (PCM) has received considerable attention owing to its high scalability and multi-level cell (MLC) operation for high storage density. However, due to the resistance drift over time, the soft error rate in MLC PCM is high. Additionally, the iterative programming in MLC negatively impacts performance and cell endurance. The conventional methods to overcome the drift problem incur large overheads, impact memory lifetime and are inadequate in terms of acceptable soft error rate (SER). In this paper, we propose a new PCM memory architecture with heterogeneous PCM arrays to increase reliability, performance and lifetime. The basic storage unit in the proposed architecture consists of two single-level cells (SLCs) and one four-level cell (4LC). Using the reduced number of 4LCs compared to conventional homogeneous 4LC PCM arrays, the drift-induced error rate is considerably reduced. By alternating each cell operation between SLC and 4LC over time, the overall lifetime can also be significantly enhanced. The proposed architecture achieves up to 10^5 times lower soft error rate with considerably less ECC overhead. With simple ECC scheme, about 22% performance improvement is achieved and additionally, the overall lifetime is also enhanced by about 57%.**

Index Terms—**Emerging memories, Endurance, Multi-level cell, Heterogeneous cell storage, Phase change memory (PCM), Resistance drift, Reliability**

I. Introduction

Memory density has increased enormously over time, thanks to the continuous technology scaling. However, the mainstream memories such as DRAM and flash memory are reaching their scaling limits. The desire for a highly scalable memory has motivated over a decade of research in emerging memories. The commercial deployment of these memories is still awaiting to overcome challenges related to their performance and reliability.

Phase change memory (PCM) is one of the emerging memory technologies that has received considerable attention as a next generation memory. It offers a number of advantages such as low-leakage power, scalability and high performance of a solid-state memory along with non-volatility and low cost of conventional disks [1]–[3]. In PCM, data is stored by modulating the resistance of the chalcogenide material which is often the $Ge_2Sb_2Te_5$ (GST) alloy [1], [3]–[5]. This material changes to either an amorphous (high resistance) or crystalline (low resistance) state depending on its resistance level. By

programming PCM to either of the states, binary information can be stored. The reset operation consists of melting the GST material by a short high current pulse, followed by instant quenching thus causing the material to become amorphous. The set operation, on the other hand, needs a current pulse of relatively longer duration that heats the material below the melting temperature followed by slow cooling. The slow heating and cooling changes the material to crystalline state. The resistance of a PCM cell in the crystalline phase is known to be around 10^3 ohms while that in the amorphous phase is around 10^6 ohms [6]. This wide resistance gap has motivated the multi-level cell (MLC) operation for PCM. It involves a more precise programming of PCM to intermediate states between the low resistance crystalline and the high resistance amorphous state. MLC operation is also seen as a necessity for PCM to compete with existing technologies such as flash memory in terms of storage density. The MLC programming in PCM requires an iterative write and verify process with multiple current pulses to bring the resistance within some range of the target level resistance.

Reliability is one of the biggest concerns in MLC PCM. While the phase change material is very much stable to the radiation-induced errors, which is a main cause of soft errors in DRAM, it introduces an entirely new mechanism of soft errors. The resistance of phase change material tends to drift over time [1], [3]–[8]. The crystalline phase tends to be relatively stable but the amorphous phase is meta-stable with considerable increase in resistance over time. Experimental results show that the increase in resistance of PCM is directly proportional to the initial resistance. The resistance drift problem is not considered to be a challenge in single level cell (SLC) PCM. It is because the rate of drift is negligible in crystalline state while higher drift in amorphous state cannot cause error in threshold-based sensing. This problem manifests itself more in multi-level cell operation where there's considerably less margin between adjacent states.

Beside resistance drift, another significant challenge in PCM is limited cell endurance. A typical PCM cell is expected to tolerate about 10^8 writes on average [9] after which the heating element can permanently detach itself from the GST material and the cell represents a stuck at fault. With further scaling, the cell lifetimes would suffer considerable variations leading to more cells failing than expected. This problem is exacerbated

by the iterative programming involved in MLC operation of PCM.

Several studies have been introduced to mitigate resistance drift [1], [3]–[8], [10] and address the problem of permanent cell failures in phase change memory [2], [9], [11], [12]. In some cases, the proposed techniques addressing the resistance drift such as use of strong ECC have negative impact on cell endurance. We, therefore, propose a new PCM architecture that trades off a portion of the storage density for increased reliability, performance and lifetime.

The rest of the paper is organized as follows. In Sec. II, we discuss the background and motivation for this research. The proposed architecture is described in Sec. III. In Sec. IV, we evaluate our idea and Sec. V concludes the paper.

II. BACKGROUND AND MOTIVATION

In DRAM, the soft errors are mostly caused by the background radiation. PCM is much robust to the radiation-induced errors. Soft errors in PCM mainly result from the resistance drift over time due to the structural relaxation of the chalcogenide material. In the following subsections, we discuss the soft error rate due to the resistance drift in 4LC PCM, the related works addressing the drift problem and the motivation for the research.

A. Soft Error Rate (SER) for 4LC PCM

MLC PCM consists of multiple resistance levels for storing multi-bit information. For example, a 4LC PCM consists of 4 levels to store 2 bit information. Programming of MLC PCM is an iterative process of write and verify operations to bring the cell's resistance within a target range for a given level. A soft error in MLC PCM is said to occur when a storage level's resistance value crosses a defined threshold due to drift over time. The soft error rate (SER) for MLC PCM can be estimated by mathematically calculating the probability of threshold crossings for different resistance levels. This analysis was proposed in [6] and has been adopted in this research to estimate SER for PCM.

The rate of resistance drift is proportional to the initial resistance of the cell. Based on the experimental data, a mathematical model for resistance drift has been introduced in [1]. In this model, resistance $R(t)$ of a cell at time t is given as:

$$R(t) = R_0(\frac{t}{t_0})^\alpha \qquad (1)$$

where R_0 is an initial resistance and α is a drift exponent which is proportional to R_0. Both R_0 and α approximately follow the Gaussian distribution [1].

To estimate the soft error rate, we assume that $log_{10}R_0$ follows a normal distribution with $\mathcal{N}(\mu_{R_0}, \sigma_{R_0}^2)$ and α follows a normal distribution with $\mathcal{N}(\mu_\alpha, \sigma_\alpha^2)$. Additionally, as in [1], [6], the target resistance in each level is assumed to lie within the range $10^{\mu_{R_0} \pm 2.75\sigma_{R_0}}$ while the thresholds for a given level are set at $10^{\mu_{R_0} \pm 3\sigma_{R_0}}$. A soft error is, therefore, said to occur when

$$R(t) > 10^{\mu_{R_0} + 3\sigma_{R_0}} \qquad (2)$$

TABLE I
DISTRIBUTION PARAMETERS (R_0 AND α) FOR 4LC PCM

Level	Data	$log_{10}R_0$		α	
		μ_{R_0}	σ_{R_0}	μ_α	σ_α
0	00	3.0		0.01	
1	01	4.0	$\frac{1}{6}$	0.02	$0.4 \times \mu_\alpha$
2	11	5.0		0.06	
3	10	6.0		0.10	

TABLE II
SOFT ERROR PROBABILITIES FOR 4LC PCM

Time (s)	Level 1	Level 2	$SER_{average}$ (4 levels)
2	Negligible	5.88E-6%	1.47E-6%
2^2	1.59E-12%	0.02%	5.35E-3%
2^3	5.89E-6%	0.12%	0.03%
2^4	7.50E-4%	0.29%	0.072%

The same parameters for distribution of R_0 and α are used in our analysis, as used in [1] and they are summarized in Table I. We use the analytical model proposed in [6] to calculate the probability of soft errors for each of the four resistance levels in 4LC PCM at a given time. The results are presented in Table II. Note that Table II shows only the error probabilities for two intermediate storage levels. It is because the error probability for storage level 0 is too small to be considered and that of the storage level 3 does not lead to any error. The last column in Table II shows the average SER for all four levels at a given time assuming all levels to be equally probable.

The average SER for DRAM (without using any error correction scheme) has been reported to be $25,000$ to $75,000$ failures in time (FIT) per billion hours of operation per Mbit or equivalently 2.5×10^{-11} to 7.5×10^{-11} per bit hour [13]. From the SER shown in Table II, it is evident that 4LC PCM would not be practical without architectural enhancement for reliability. In the next subsection, we discuss related works addressing this problem and the motivation for this research.

B. Related works and motivation

For PCM reliability, there has been some research done. Xu et al. [1] introduced a time-aware fault tolerance scheme which adaptively adjusts the thresholds utilizing the time duration of the stored information. In [5], additional reference cells store the threshold resistance values thus incorporating drift in threshold values. However, for these methods, the fault tolerance enhancement would be limited by the randomness in initial resistance as well as the drift. The drift-prone data pattern reduction method by data rotation and inversion is proposed in [4]. It also suggests a hybrid or temperature aware page memory allocation scheme. [7] proposes an efficient scrubbing scheme using a strong ECC and on-chip approximate error detection. This can significantly reduce the number of errors, however, SER still remains much higher than DRAM. Moreover, the use of stronger ECC codes results in a reduced cell lifetime and performance. [6] presents a ternary storage array architecture. It can increase noise margins
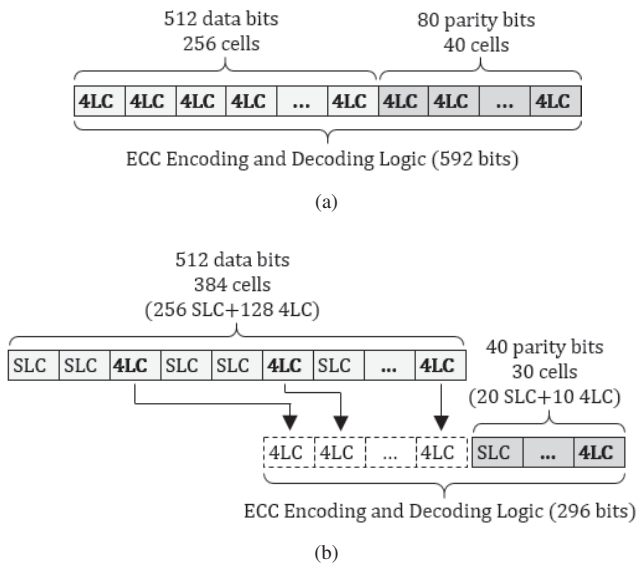
512 data bits
256 cells

80 parity bits
40 cells

| 4LC | 4LC | 4LC | 4LC | ... | 4LC | 4LC | 4LC | ... | 4LC |

ECC Encoding and Decoding Logic (592 bits)

(a)

512 data bits
384 cells
(256 SLC+128 4LC)

| SLC | SLC | 4LC | SLC | SLC | 4LC | SLC | ... | 4LC |

40 parity bits
30 cells
(20 SLC+10 4LC)

| 4LC | 4LC | ... | 4LC | SLC | ... | 4LC |

ECC Encoding and Decoding Logic (296 bits)

(b)

Fig. 1. Comparison of different array architectures for storing 512 bits of data with BCH-8 ECC scheme (a) Conventional uniform 4LC PCM array (b) Proposed heterogeneous (SLC + 4LC) PCM array

SLC	SLC	4LC	Mode = 0
SLC	4LC	SLC	Mode = 1
4LC	SLC	SLC	Mode = 2

Fig. 2. Intra-array wear leveling based Read/Write architecture

and decrease error rates. However, it would be less attractive in terms of cell endurance because it still requires iterative programming.

With high scalability and potential 3-D integration, the storage density in PCM can easily outweigh the existing DRAM technology, however the higher soft error rate due to resistance drift pose a great challenge. Based on these observations, this paper proposes a heterogeneous PCM array architecture based on trading off density for the improved reliability, performance and lifetime. Using the proposed heterogeneous memory array, the SER is reduced up to 10^5 times with significantly less ECC and memory scrubbing overhead. The overall memory lifetime is also enhanced up to 57% as compared to uniform 4LC structure.

III. PROPOSED HETEROGENEOUS PCM ARCHITECTURE

Reliability, performance and lifetime are impacted by MLC operation in PCM. Stronger ECCs with a periodic memory scrubbing scheme are required for acceptable SER. This adds considerable hardware and performance overhead. This paper proposes a heterogeneous PCM array architecture which is composed of a combination of SLC and 4LC. This array has a reduced number of 4LC, thus enhancing the reliability and performance while employing simpler ECC logic. The detailed structure is described in the following section. In Sec. III-B, an intra-array wear leveling scheme for the proposed heterogeneous array architecture is introduced to improve the memory lifetime. Sec. III-C provides SER analysis in detail.

A. Heterogeneous PCM array

The basic storage unit in the proposed architecture consists of two single-level cells and one four-level cell (4LC). Since the resulting architecture has a less number of drift prone cells,
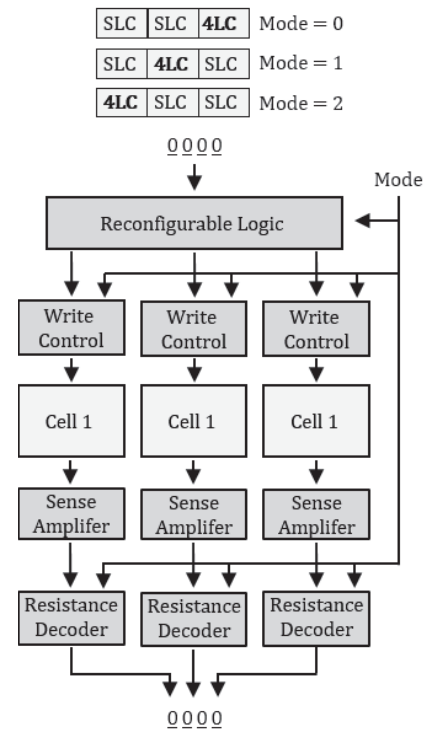
it would considerably reduce the error rate. By using ECC only for 4LCs, a less ECC overhead and latency is achieved.

Fig.1 depicts a PCM row for the conventional 4LC and the proposed array architecture. Assuming the BCH-8 ECC scheme to correct up to 8 errors, the proposed array needs 256 SLCs and 128 4LCs for 512 data bits and 20 SLCs and 10 4LCs for 40 parity bits, whereas the conventional 4LC array requires 256 cells to store 512 data bits and 40 more cells to store 80 parity bits. As can be seen, because the number of 4LC cells is reduced by about 53% than in the conventional array with all 4LCs, the proposed array architecture would achieve considerably higher reliability by trading off some array density. However, it should be noted that the information density (bits/cell) over memory lifetime from the proposed architecture is higher than the conventional array. The information density of 4LC PCM with BCH-8 ECC is 1.73 while that for proposed heterogeneous array is 1.24. As evaluated later in Sec. IV-B, the lifetime of the prosed architecture is enhanced up to 57% as compared to conventional approach. Taking into account the enhancement in lifetime, the effective information density for the proposed array is 1.94 which is roughly 12% higher than that of conventional 4LC. It should also be noted that for the same amount of reliability, the conventional array requires much stronger ECC than the proposed architecture resulting in even higher effective information density for the proposed architecture. The ECC hardware and performance overhead is also considerably reduced because of smaller code word. For example, using BCH-8, we need to encode and

Scrubbing Period (seconds)	Average per-bit SER	Probability of uncorrectable errors for 512 bits / Soft Error Rate									
		No ECC		BCH-8		BCH-16		BCH-24		BCH-32	
		Conventional (512b+0b)	Proposed (512b+0b)	Conventional (512b+80b)	Proposed (512b+40b)	Conventional (512b+160b)	Proposed (512b+80b)	Conventional (512b+240b)	Proposed (512b+120b)	Conventional (512b+320b)	Proposed (512b+160b)
2^3	0.03%	7.39%	3.77%	Negligible	Negligible	Negligible	Negligible	Negligible	Negligible	Negligible	Negligible
2^4	0.07%	16.41%	8.57%	1.49E-10%	2.66E-13%	Negligible	Negligible	Negligible	Negligible	Negligible	Negligible
2^5	0.133%	28.87%	15.66%	3.93E-8%	4.30E-11%	Negligible	Negligible	Negligible	Negligible	Negligible	Negligible
2^6	0.218%	42.80%	24.37%	2.70E-6%	3.31E-9%	Negligible	Negligible	Negligible	Negligible	Negligible	Negligible
2^7	0.325%	56.54%	34.08%	7.45E-5%	1.06E-7%	Negligible	Negligible	Negligible	Negligible	Negligible	Negligible
2^8	0.475%	70.44%	45.63%	1.54E-3%	2.72E-6%	1.27E-10%	Negligible	Negligible	Negligible	Negligible	Negligible
2^9	0.668%	82.02%	57.59%	2.03E-2%	4.68E-5%	2.32E-8%	1.89E-13%	4.11E-13%	Negligible	Negligible	Negligible
2^{10}	0.91%	90.37%	68.97%	0.1773%	5.71E-4%	2.15E-6%	5.38E-12%	2.81E-12%	Negligible	Negligible	Negligible
2^{11}	1.21%	95.57%	78.95%	1.0827%	5.25E-3%	1.10E-4%	4.82E-10%	1.34E-9%	Negligible	Negligible	Negligible
2^{12}	1.57%	98.26%	86.81%	4.6105%	3.61E-2%	3.14E-3%	2.58E-8%	2.66E-7%	8.33E-13%	8.55E-12%	8.22E-13%

decode 296 bits in proposed case as opposed to 592 bits in the conventional scheme.

B. Intra-array wear leveling for lifetime enhancement

One of the advantages of the proposed heterogeneous array architecture is enhancement in memory system lifetime compared to the conventional scheme. The proposed architecture alternates the position of 4LC over the operation time thus allowing an equal or similar amount of wear to all cells.

We can change the position of 4LC in the basic storage unit by using two bits mode selection. In Mode 0, the last cell is 4LC, while in Mode 1 and 2, its position is changed to 2 and 1, respectively. The Mode selection can be sequentially changed over the memory lifetime. Since the mode selection is the same for the entire memory array, the overhead is negligible.

Fig. 2 shows the corresponding wear leveling architecture for implementing mode-based write and read operation. In the conventional 4LC scheme, a uniform programming and sensing logic is used to store and retrieve two bits in each cell. For the write operation in proposed architecture, we specify the programming behavior for each cell based on a selected mode. This requires additional reconfigurable logic. To retrieve data on a read, uniform sense amplifiers are used to measure the resistance level of each cell. Then, the resistance decoder decodes it to retrieve the stored data based on the mode bits.

C. Soft Error Rate (SER) analysis for proposed architecture

As given in [6], the SER for PCM can be evaluated reasonably by mathematically estimating the probability of uncorrectable errors in a data word. The SER analysis is performed by considering various ECC schemes for error correction.

$(72, 64)$ Hamming code is widely used in conventional DRAM for single error correction. It adds 8 parity bits to 64 bit data and corrects 1 error. If we use 4-level cells to store a 72 bit code word, 36 cells are required. With gray encoding,

we expect to have a single bit error whenever there's a state change due to resistance drift. Two errors in 72 bits code word are possible only if two of the 36 cells change state. Therefore, the probability of uncorrectable errors (more than 1) in this case can be found as:

$$
\begin{aligned}
P_{error}(72b) &= 1 - P(\text{no error}) - P(\text{one bit error}) \\
&= 1 - (1 - SER_{average})^{36} \\
&\quad - \binom{36}{1}(1 - SER_{average})^{35}(SER_{average})
\end{aligned}
\tag{3}
$$

Note that $SER_{average}$ is the average SER per 1-bit computed earlier and listed in Table II. By knowing the number of 4-level cells used to store the code word, the Equation 3 can be extended to the cases where stronger ECC schemes are applied. In general, for m 4-level cells, the probability of having at least n errors is:

$$
= 1 - \sum_{k=0}^{n-1} \binom{m}{k}(1 - SER_{average})^{m-k}(SER_{average})^k \tag{4}
$$

Equation 4 is used to compare the SER for the conventional 4LC approach and the proposed method. Let's assume that 512 bit data word with BCH-8 is used. This requires additional 80 parity bits. In the conventional approach, 296 4LCs are needed to represent 592 bit encoded information. The proposed method requires $128 \, (= 512/4)$ 4LCs for data bits and this stores 256 bit information. Since the proposed method requires parity information only for 4LCs, the additional 40 parity bits would need 10 more 4LCs. Hence, the proposed method uses 138 4LCs to represent a code word (512 data word with BCH-8). This finds $m = 296$ for the conventional method and $m = 138$ for the proposed method.

Table III summarizes the SER in terms of probability of uncorrectable errors calculated from Equation 4 for a 512 bit data word with various ECC capabilities. The first column
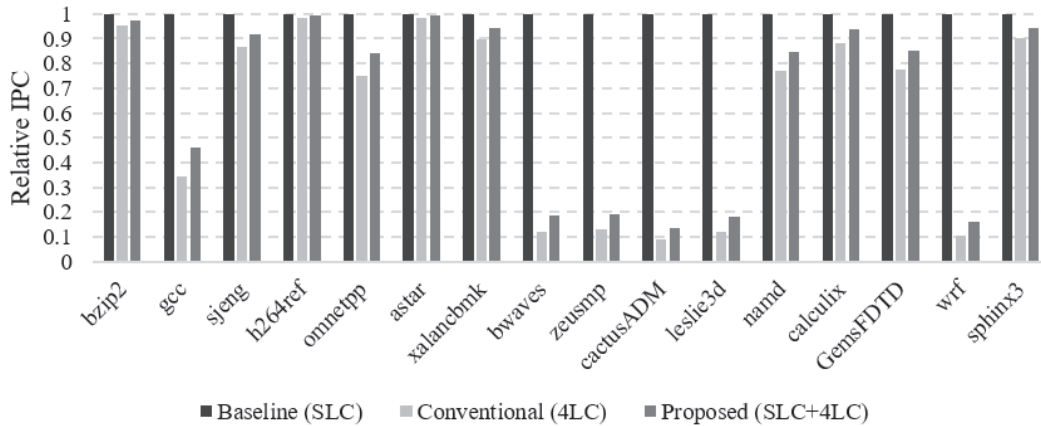
Fig. 3. Performance comparison between different PCM array architectures

shows the various scrubbing periods. For each scrubbing period, the corresponding average SER per 1-bit is listed in the second column. It is used for (4) to calculate the probability of uncorrectable errors. To consider various correcting capabilities, No ECC to BCH-32 is also used. Some results are denoted as 'Negligible' when the result of a calculation in Mathematica 11.2 [14] is negative or less than 10^{-13} due to limited precision (considering baseline DRAM SER of the order of 10^{-11}).

As an example, consider BCH-16 scheme with scrubbing period of 2^{12} seconds. In this case, the uncorrectable error probability for 512 bit data word in conventional 4LC PCM is 3.14×10^{-3}. Using proposed architecture, the error probability is 2.58×10^{-8} showing a marked reduction of about 10^5 times. The results in Table III also demonstrate that, for a given level of reliability, the conventional scheme requires stronger ECC as compared to the proposed architecture.

IV. EVALUATION

From the SER analysis in Sec. III-C, it is evident that the proposed architecture enhances the reliability of 4LC PCM up to 10^5 times of the conventional approach. To evaluate the performance and memory lifetime improvement by the proposed array architecture, we simulated the execution of several benchmark programs from the SPEC2006 benchmark [15] using GEM5 simulator [16]. The simulation is based on a model of 16GB main memory having 8 banks (2GB per bank capacity). The CPU clock frequency is considered to be 1GHz with L1 and L2 caches of sizes 64kB and 4MB, respectively. We assume BCH-24 ECC for the conventional 4LC array architecture and BCH-16 ECC for the introduced heterogeneous architecture keeping in view that both have the same soft error rate for the similar scrubbing period. In our simulations, the scrubbing period for ECC is taken as 2^9 seconds which is considered to be a manageable scrubbing time for PCM [7].

A. Impact on performance

The performance enhancement is evaluated with respect to instructions per cycle (IPC) by executing 16 programs from SPEC2006 benchmark. The estimates are based on executions over a period of 10 years of memory operation. The IPC is affected by scrubbing time in different architectures as well as the latency due to ECC. In our simulations, as mentioned earlier, the scrubbing period for ECC is set as 2^9 seconds. The time for scrubbing a cache line (considering all 4LCs) is given as 1.15μs in [6]. A 2GB 4LC PCM memory bank with 8M cache lines, therefore, requires a total of 9.65 seconds for a scrubbing operation. For 2GB memory, the ratio of 4LCs in the proposed architecture (using BCH-16) to that in the conventional scheme (using BCH-24) is 0.394. Therefore, the scrubbing time for 2 GB memory in case of the proposed architecture is 3.8 seconds. The latency for BCH-24 and BCH-16 is considered to be 1217ns and 772ns in 22nm process technology as shown in [17]. We therefore consider the impact of both scrubbing time and ECC latency to estimate the effective IPC for both schemes. The results are shown in Fig. 3. The graph shows relative IPC with SLC PCM as a baseline for comparison. The average performance improvement in heterogeneous array architecture is found to be 21.9% over the conventional 4LC PCM array.

B. Impact on lifetime

Using the reduced number of drift-prone 4LCs, and changing the programming behavior of each cell over memory lifetime using intra-array wear leveling proposed in Sec. III-B, the proposed architecture achieves considerable improvements in lifetime. The simulation results show that the number of writes due to scrubbing is not a significant portion of the total writes. Using a scrubbing period of 2^9 seconds, the average number of scrubbing related writes for different benchmark programs is 6.16×10^5 out of a total of 5.54×10^{11} writes (1.11×10^{-4}%). Using the number of writes, MLC PCM lifetime can be estimated by taking into consideration the

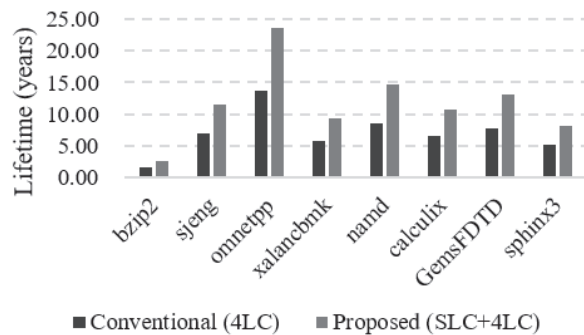*Design, Automation And Test in Europe (DATE 2018)*

Fig. 4. Comparison of memory lifetimes for conventional 4LC PCM and proposed SLC+4LC architecture

number of iterations per write required in different schemes. As given in [18], the number of iterations for MLC PCM has randomness due to process variations. However, 98% of the cells in a 4LC PCM can be programmed by using less than 5 iterations. The average number of iterations is found to be 2.2. For the proposed architecture, every cell will incur an average of 4.2 programming iterations for a write by using 3 different modes of operation over the lifetime. In contrast, in the conventional approach, every 4LC would require an average of 6.6 iterations. We assume that each iteration equally wears a cell with maximum number of iterations to be 10^8 (cell lifetime). The lifetimes are calculated for different benchmark programs and the results are shown in Fig. 4. The average improvement in lifetime over the conventional scheme is estimated to be 57%.

V. CONCLUSION

The reliability and limited endurance are major concerns in MLC PCM. Resistance drift leads to high error rates thus requiring effective architectural enhancements to make MLC PCM reliable. In this work, a new heterogeneous array architecture, comprising of both single-level cells and multi-level cells, has been introduced to achieve improved reliability, performance and lifetime. With reduced number of 4LCs, the soft error rate can be reduced up to 10^5 times. By employing a simpler ECC scheme, the overhead is reduced and performance is improved by about 22%. Additionally, the memory lifetime is improved by 57% thus making up for the loss in information density.

In this paper, we evaluated the idea of using heterogeneous PCM array by using the minimum number of 4LCs. It can be extended to flexibly adjusting the number of 4LCs over lifetime to achieve reasonable reliability without compromising on memory lifetime.

ACKNOWLEDGMENT

REFERENCES

[1] W. Xu and T. Zhang, "A time-aware fault tolerance scheme to improve reliability of multilevel phase-change memory in the presence of significant resistance drift," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 19, no. 8, pp. 1357–1367, Aug 2011.

[2] R. Maddah, R. Melhem, and S. Cho, "Rdis: Tolerating many stuck-at faults in resistive memory," *IEEE Transactions on Computers*, vol. 64, no. 3, pp. 847–861, March 2015.

[3] M. Asadinia, M. Arjomand, and H. Sarbazi-Azad, "Variable resistance spectrum assignment in phase change memory systems," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 23, no. 11, pp. 2657–2670, Nov 2015.

[4] W. Zhang and T. Li, "Helmet: A resistance drift resilient architecture for multi-level cell phase change memory system," in *2011 IEEE/IFIP 41st International Conference on Dependable Systems Networks (DSN)*, June 2011, pp. 197–208.

[5] P. Junsangsri, J. Han, and F. Lombardi, "A system-level scheme for resistance drift tolerance of a multilevel phase change memory," in *2014 IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT)*, Oct 2014, pp. 63–68.

[6] N. H. Seong, S. Yeo, and H.-H. S. Lee, "Tri-level-cell phase change memory: Toward an efficient and reliable memory system," in *Proceedings of the 40th Annual International Symposium on Computer Architecture*, ser. ISCA '13. New York, NY, USA: ACM, 2013, pp. 440–451.

[7] M. Awasthi *et al.*, "Efficient scrub mechanisms for error-prone emerging memories," in *IEEE International Symposium on High-Performance Comp Architecture*, Feb 2012, pp. 1–12.

[8] R. Wang, Y. Zhang, and J. Yang, "Readduo: Constructing reliable mlc phase change memory through fast and robust readout," in *2016 46th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, June 2016, pp. 203–214.

[9] E. Ipek, J. Condit, E. B. Nightingale, D. Burger, and T. Moscibroda, "Dynamically replicated memory: Building reliable systems from nanoscale resistive memories," *SIGPLAN Not.*, vol. 45, no. 3, pp. 3–14, Mar. 2010.

[10] M. Stanisavljevic, A. Athmanathan, N. Papandreou, H. Pozidis, and E. Eleftheriou, "Phase-change memory: Feasibility of reliable multilevel-cell storage and retention at elevated temperatures," in *2015 IEEE International Reliability Physics Symposium*, April 2015, pp. 5B.6.1–5B.6.6.

[11] S. Schechter, G. H. Loh, K. Strauss, and D. Burger, "Use ecp, not ecc, for hard failures in resistive memories," *SIGARCH Comput. Archit. News*, vol. 38, no. 3, pp. 141–152, Jun. 2010.

[12] M. K. Qureshi, "Pay-as-you-go: Low-overhead hard-error correction for phase change memories," in *2011 44th Annual IEEE/ACM International Symposium on Microarchitecture (MICRO)*, Dec 2011, pp. 318–328.

[13] B. Schroeder, E. Pinheiro, and W.-D. Weber, "Dram errors in the wild: A large-scale field study," in *Proceedings of the Eleventh International Joint Conference on Measurement and Modeling of Computer Systems*, ser. SIGMETRICS '09. New York, NY, USA: ACM, 2009, pp. 193–204.

[14] [Online]. Available: https://www.wolfram.com/mathematica/

[15] [Online]. Available: https://www.spec.org/cpu2006/

[16] N. Binkert *et al.*, "The gem5 simulator," *SIGARCH Comput. Archit. News*, vol. 39, no. 2, pp. 1–7, Aug. 2011.

[17] D. Strukov, "The area and latency tradeoffs of binary bit-parallel bch decoders for prospective nanoelectronic memories," in *2006 Fortieth Asilomar Conference on Signals, Systems and Computers*, Oct 2006, pp. 1183–1187.

[18] L. Jiang, B. Zhao, Y. Zhang, J. Yang, and B. R. Childers, "Improving write operations in mlc phase change memory," in *Proceedings of the 2012 IEEE 18th International Symposium on High-Performance Computer Architecture*, ser. HPCA '12. Washington, DC, USA: IEEE Computer Society, 2012, pp. 1–10.