

Received January 22, 2019, accepted February 22, 2019, date of publication February 27, 2019, date of current version March 13, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2901738

Improving Energy-Efficiency in Dynamic Memories Through Retention Failure Detection

ROBERT GITERMAN¹, ROMAN GOLMAN², AND ADAM TEMAN¹

¹Telecommunications Circuits Laboratory, Institute of Electrical Engineering, EPFL, 1015 Lausanne, Switzerland

²Emerging Nanoscale Integrated Circuits and Systems Laboratories, Faculty of Engineering, Bar-Ilan University, Ramat Gan 5290002, Israel

Corresponding author: Robert Giterman (robert.giterman@epfl.ch)

This work was supported by the Israel Science Foundation under Grant 996/18 and Grant 2181/18.

ABSTRACT A gain-cell embedded DRAM (GC-eDRAM) is an attractive logic-compatible alternative to the conventional static random access memory (SRAM) for the implementation of embedded memories, as it offers higher density, lower leakage, and two-ported operation. However, it requires periodic refresh cycles to maintain its data which deteriorates due to leakage. The refresh-rate, which is traditionally set according to the worst cell in the array under extreme operating conditions, leads to a significant refresh power consumption and decreased memory availability. In this paper, we propose to reduce the cost of GC-eDRAM refresh by employing failure detection to lower the refresh-rate. A 4T dynamic complementary dual-modular redundancy bitcell is proposed to offer per-bit error detection, resulting in a substantial decrease in the refresh-rate and over 60% power reduction compared with the SRAM. The proposed approach is also compared with the conventional SRAM and GCeDRAM implementations with integrated error correction codes, demonstrating significant area and latency reductions.

INDEX TERMS Error detection and correction, error correcting codes, SRAM, gain cells, logic-compatible eDRAM, GC-eDRAM, low power.

I. INTRODUCTION

As the functionality and complexity of integrated circuit (IC) systems continue to grow, embedded memories are of great interest in modern multiprocessors and other VLSI system-on-chips (SoCs) [1]. The mainstream solution to the implementation of embedded memories is based on six-transistor (6T) static random access memory (SRAM) macrocells due to their high speed and process scalability. However, 6T SRAM incur a large area penalty and suffer from high static power consumption, often dominating the power budget and real-estate of these VLSI systems [1], [2]. Moreover, technology scaling has led to a continuous increase in parametric variations, resulting in reduced SRAM noise margins, which ultimately limit the voltage scaling capabilities of these memories [3], [4].

Logic-compatible gain-cell embedded DRAM (GC-eDRAM) is an attractive alternative to conventional SRAM, as it offers higher density due to a smaller bitcell size, lower leakage, and two-ported operation [5]. However, the dynamic nature of gain-cells requires the application of

periodic refresh operations to maintain the data, which is stored on a parasitic MOSFET gate capacitor. While gain-cell implementations in mature technology nodes were shown to provide sufficiently high data retention times (DRTs) [6], [7], gain-cell embedded DRAM (GC-eDRAM) implementations in 65 nm nodes and below demonstrate significantly lower DRTs due to increased leakage currents and decreased parasitic storage capacitances [8], [9]. Furthermore, the refresh-rate of GC-eDRAM is commonly set according to the worst DRT of all cells in a memory array, simulated under worst-case biasing conditions and PVT variations [6]–[9]. As a result, the retention power of these memories becomes dominated by the refresh power component, and the availability of the memory to processor access is lowered due to frequent refresh cycles.

The DRT of GC-eDRAM arrays highly depends on the sub-threshold (sub- V_T) leakage of MOS transistors, which is exponentially impacted by threshold voltage (V_T) fluctuations due to process variations and device mismatch. As a result, the DRT distribution of GC-eDRAM arrays has a long tail, with the average DRT found to be a few orders-of-magnitude higher than the minimum DRT of the array [10]. This phenomenon has gotten worse with technology scaling

The associate editor coordinating the review of this manuscript and approving it for publication was Tae Hyoung Kim.

due to increased parametric variations [11]. Furthermore, the DRT is measured under worst-case operating sequences, which are highly unlikely, if not infeasible [12]. Therefore, substantial energy savings can be achieved by refreshing *when needed* and not at a predefined rate, set according to a worst-case DRT based on the first failing bitcell, under the worst possible operating conditions and sequences.

Recent work has proposed trading off data accuracy (i.e., bitcell failure rate) with power consumption in cases, where the underlying application has a certain degree of error tolerance [11], [13]. This concept can be extended to non-error tolerant memories that have a back-up copy of their data, such as low-level caches that employ write-through policy [14]–[17]. Based on the fact that the application is either error tolerant or that a backup copy exists, we propose to postpone the refresh operation until one or more errors are detected. This provides a means to significantly increase the energy-efficiency and availability of GC-eDRAM arrays, while bounding the error for error-tolerant applications [18], or signaling that the data is corrupted and should be recovered. An example of such an operation could be signaling a miss in a write-through cache, which would initiate data recovery from a higher level cache.

As an efficient implementation of such an approach, we propose using a 4-transistor complementary dual-modular redundancy (CDMR) gain-cell with inherent per-bit error detection capabilities, previously proposed for space applications due to its enhanced soft-error tolerance [19]. The inherent error-detection scheme does not incur a latency overhead or additional hardware, as required by conventional error correction code (ECC) schemes, and therefore, it is advantageous for the implementation of memories that require high speed operation, such as low-level caches. The proposed approach achieves over 60% energy savings by allowing up-to 5 presumed DRT errors, with a 15% smaller cell size than a conventional 6T SRAM cell. The energy savings can be further increased by tolerating a higher number of DRT errors.

A. CONTRIBUTIONS

The main contributions of this paper are summarized as follows:

- 1) This work presents a methodology for refresh-rate reduction through error detection, based on the large spread of DRT variation in deeply-scaled process nodes.
- 2) We analyze error detection realization using ECC, and provide a qualitative analysis of the power, area, and latency overheads of this approach.
- 3) We propose using a CDMR technique, implemented using a 4-transistor GC-eDRAM bitcell with per-bit error detection, to relax the refresh-rate. This proposal is compared with the ECC approach.
- 4) The proposed CDMR approach achieves over 15% bitcell area decrease, and up-to 60% power savings compared to a 6T SRAM cell, as well as significant power, area, and latency reductions compared to ECC.

B. OUTLINE

The rest of this paper is organized as follows: Section II demonstrates the cost of refresh in terms of the power and availability of GC-eDRAM arrays; Section III introduces concept of DRT extension using error detection; Section IV evaluates the error detection realization using ECC; Section V demonstrates how refresh-rate relaxation can be achieved using CDMR GC-eDRAM with inherent per-bit error detection; Section VI presents the implementation of the proposed memory and compares it with other memory solutions with integrated ECC; and Section VIII concludes the paper.

II. COST OF HIGH REFRESH-RATE

Total power consumption is generally measured as a combination of dynamic and static power components. In the case of static memories, such as SRAM, the dynamic power component (P_{dyn}) is consumed during read and write accesses, while the static power (P_{stat}) is due to leakage during standby (retention) periods. However, when discussing dynamic memories, such as GC-eDRAM, data retention requires the application of periodic refresh operations. Therefore, the static power of dynamic memories is redefined as retention power (P_{ret}) that is composed of both leakage and refresh power components. A refresh operation includes reading out each row and writing it back once every refresh period, such that the total power can be written as:

$$P_{\text{ret}} = P_{\text{leak}} + P_{\text{refresh}} = P_{\text{leak}} + f_{\text{refresh}}(E_{\text{write}} + E_{\text{read}}), \quad (1)$$

where P_{leak} is the average leakage power of the array, P_{refresh} is the refresh power, E_{write} and E_{read} are the write and read energies, respectively, and f_{refresh} is the refresh-rate of the memory. Consequently, lowering the refresh-rate leads to extensive power savings, especially due to the fact that the leakage power component of (1) is often substantially lower than the refresh power component [11].

A different aspect to the cost of refresh-rate is the availability of the array to processor operations, since during refresh operations, the memory cannot be accessed. Defining array availability (Av) as the percentage of time that the memory is available for external access leads to:

$$Av(\%) = (1 - N_{\text{rows}}(t_{\text{write}} + t_{\text{read}})f_{\text{refresh}}) \times 100, \quad (2)$$

where N_{rows} is the number of rows, and t_{write} and t_{read} are the write and read access times, respectively. Again, this leads to the conclusion that refresh-rate is expensive; this time, in terms of array availability.

To understand the need for the application of refresh, Fig. 1 shows a conventional two-transistor (2T) gain-cell, which stores its data as charge on a parasitic capacitor (C_{SN}). The stored charge leaks away over time, primarily through the write transistor (NW), when the voltage difference between the SN and the write bit line (WBL) is high, maximizing the drain induced barrier lowering (DIBL) current. The DRT is defined as the time after write, at which the stored data can no longer be correctly read out, and consequently, the refresh-rate must be set to ensure that the data is rewritten before

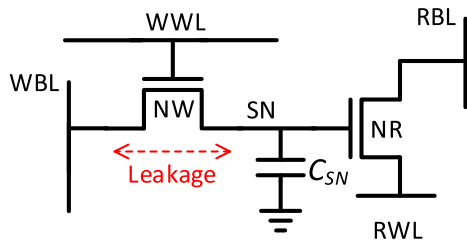


FIGURE 1. Conventional 2T GC-eDRAM with the main leakage path from storage node (SN).

the DRT has passed. While the average DRT of the array is sufficiently high, the DRT distribution has a wide spread across several orders-of-magnitude, primarily due to random dopant fluctuations (RDF), which significantly affect the V_T of NW [7], [10]. In order to ensure that no DRT errors occur, the refresh-rate is usually set according to the cell with the worst data retention, as extracted from high-sigma analysis under the assumption of worst-case operation.

Fig. 2(a) shows the SN voltage following write ‘1’ (blue) and write ‘0’ (red) operations, demonstrating a degradation in the stored voltage due to the leakage currents through the write transistor. The plot is based on 10,000 Monte Carlo (MC) simulations, including both mismatch and global variations, for a 28 nm bulk CMOS technology. The simulations were run under worst-case biasing conditions, with WBL biased at the opposite voltage of SN in SN in order to maximize the DIBL current through NW. This extreme scenario only occurs when a constant value is continuously written to a certain column in the memory array, and this value is opposite to the value stored at the worst-case bit position. Despite this situation having a very low probability of actually occurring, it is generally accepted as the requirement to guarantee 100% error-free operation.

The actual DRT of a GC-eDRAM array depends on the specific array organization and peripheral circuits; however, the simulations of Fig. 2(a) can be used to get a good approximation of the DRT for a general configuration. Estimating the DRT as the time it takes for the difference between the two voltage levels in Fig. 2(a) to decrease beneath 200 mV leads to the distribution of Fig. 2(b), which is approximately log-normal, in correspondence with the model of [10]. While the average estimated DRT of a 2T GC-eDRAM is found to be 24 μ s, the large spread results in a minimum estimated DRT of only 1.5 μ s. The conventional requirement to set the refresh period lower than the worst-case DRT leads to both high retention power and very low array availability, severely impeding the feasibility of using GC-eDRAM in deeply scaled technologies.

III. LOWERING THE REFRESH-RATE WITH ERROR DETECTION

The previous section introduced the disadvantages associated with setting a high refresh-rate. Accordingly, several previous studies have proposed circuits to extend the retention time

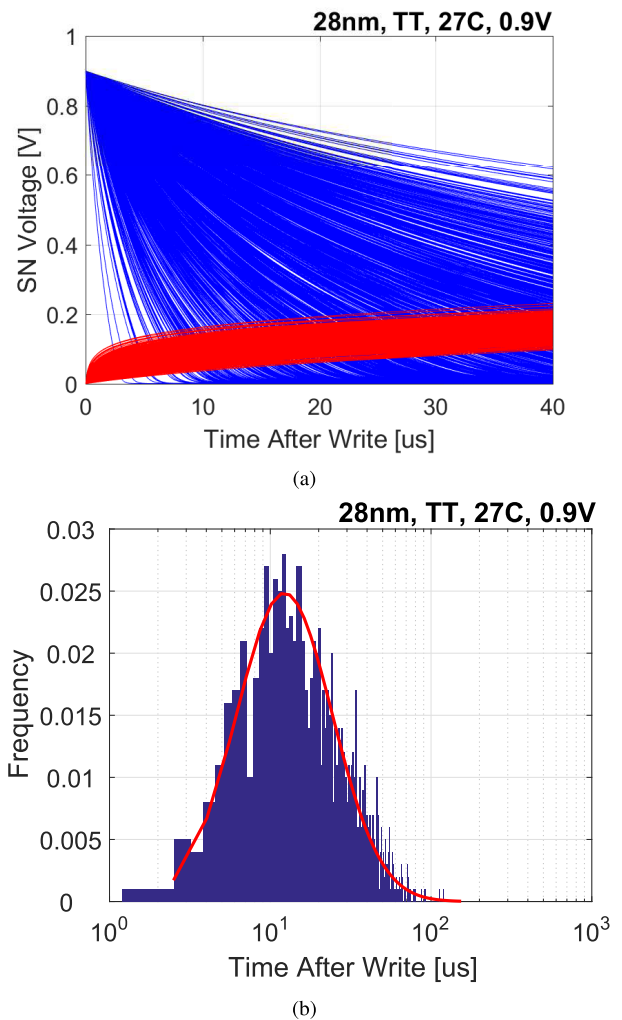


FIGURE 2. (a) SN degradation after write. (b) Data retention time distribution of the 2T GC-eDRAM.

of GC-eDRAM arrays, and consequently reduce the refresh-rate [6], [11], [20], [21]. In [6], a replica circuit was proposed to track both the global corner of the chip, as well as the access statistics of the array, and thereby extend the retention period beyond the traditional worst-case assumption. Finally, the works presented in [11] and [13] propose to relax the 100% correct memory requirement, when operating with error-tolerant applications in order to achieve significant power reduction.

In this work, we advance these ideas and propose to employ error detection to track the errors in the memory in order to dynamically extend the refresh-rate. This approach enables reducing the wide margin due to the worst-case assumption, while still ensuring correct functionality.

A. POWER SAVINGS BY RELAXING THE WORST-CASE ASSUMPTION

The log-normal DRT distribution of Fig. 2(b), points to the fact that the minimum DRT is set by a number of extreme outliers, which suffer from much lower retention times than

the rest of the cells. This observation has led to the assumption that by relaxing the 100% correctness requirement, the refresh-rate could be significantly reduced, as shown by analyzing the failure statistics [11], [22], [23].

Taking the distribution of Fig. 2(b), we can devise the DRT failure probability for a given refresh-rate as:

$$Pr_{\text{failure}} = Pr(t_{\text{DRT}} < 1/f_{\text{ref}}), \quad (3)$$

where t_{DRT} is the DRT of a given bitcell. Assuming an independent distribution of the DRT of all cells [6], [9], the probability that up to k DRT failures occur in a given set of N bits can be expressed according to the cumulative distribution functions (CDFs) equation of the DRT [22]:

$$Pr(x \leq k) = \sum_{i=0}^k \binom{N}{i} Pr_{\text{failure}}^i (1 - Pr_{\text{failure}})^{N-i}. \quad (4)$$

The complementary CDF of the DRT describes the probability of having more than k DRT failures, expressed as:

$$Pr(x > k) = 1 - Pr(x \leq k). \quad (5)$$

Fig. 3 demonstrates the complementary CDFs of more than zero to more than four DRT errors, with N set to 64 bits. The flat area of these curves show that the refresh-rate can be significantly lowered (i.e., a longer refresh period), while still bounding the number of errors in a row to a very low probability. The actual failure probability is lower yet, as this still assumes a worst-case corner with worst-case operating conditions. However, this analysis provides a means to show that even if the refresh-rate was much lower, fewer than k errors are likely to occur. In fact, Fig. 4 plots the potential reduction of f_{ref} while ensuring that there is less than a 0.1% probability that k DRT failures have occurred in a single row. The f_{ref} reduction ranges from 26%–63% by either tolerating or providing error correction capabilities for between 1–5 DRT errors (still, under worst-case operations). The

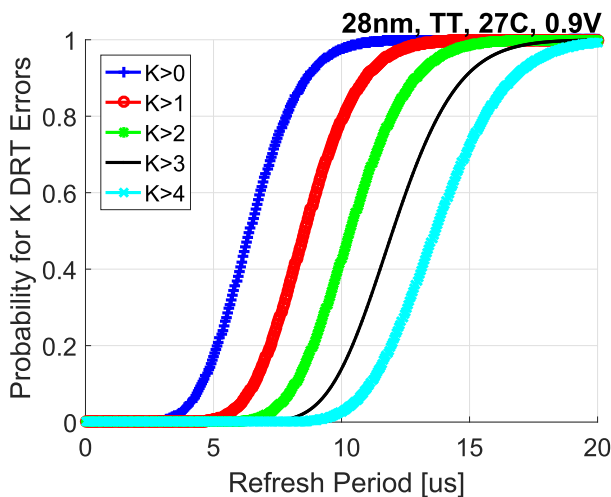


FIGURE 3. Probability for (k) number of errors for different refresh-rates.

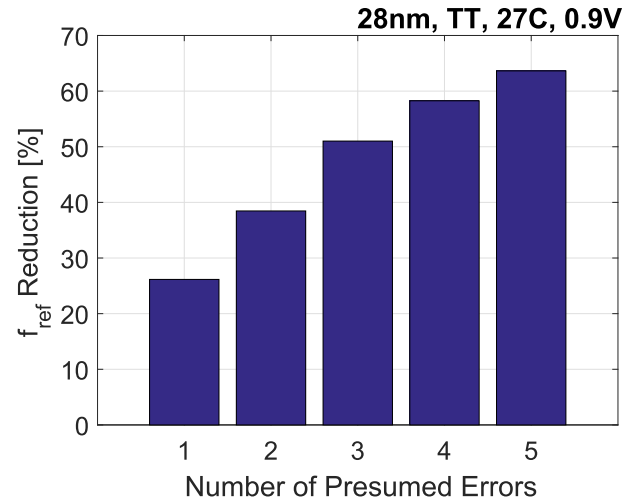


FIGURE 4. Possible f_{ref} reduction by tolerating presumed DRT errors.

lower refresh-rate would result in a reduction in refresh power and improved memory availability, as shown in (1) and (2).

B. METHODOLOGY FOR REFRESH-RATE REDUCTION THROUGH ERROR DETECTION

The previous subsection demonstrated the potential for reducing the refresh-rate by allowing for a certain number of errors to occur in a GC-eDRAM data word, thereby reducing the power consumption of the memory and increasing its availability for access. This concept was previously proposed in the context of approximate computing, where applications that can tolerate a certain number of errors can exploit this trade-off [11], [13]. However, the previous works put no error bound on the DRT failures, and relied solely on the failure probabilities extracted from simulated DRT distributions. We propose to extend the exploitation of this trade-off for:

- 1) Applications with error detection and correction (EDAC) capable memories.
- 2) Applications with a backed up copy of the data, such as write-through caches.

As previously mentioned, the standard approach to setting the refresh-rate of a GC-eDRAM array is according to the DRT of the worst possible cell in the array. However, due to the log-normal distribution of DRT, the probability that the worst simulated cell will fall within a given row is low. By providing even a single bit of error correction capability, a single error in a given word can be corrected. Therefore, the refresh period must be set to ensure that there is a very low (approaching zero) probability that two retention errors occur in the same row. This probability is already much lower than that of a single failure in the array, as previously shown in (4). Thus, the inclusion of even a single bit of EDAC leads to a very large reduction in retention power and increase in array availability. Furthermore, many memories are equipped with such EDAC capabilities to account for soft-errors, and therefore, applying such a methodology can often come at no additional cost.

C. REFRESH-RATE REDUCTION IN WRITE-THROUGH CACHES

Further reduction of f_{ref} can be achieved by integrating higher EDAC capabilities at the cost of their accompanying overheads. These overheads can often be unacceptable, especially when considering high-performance access requirements, such as for low-level caches. In write-through caches and in various other types of memory systems, a backup copy of the data is kept in a slower, larger memory or storage block. Taking cache as an example, we propose to exploit the cache-line validity to achieve further power savings and higher availability. Similar mechanisms could be applied to other memory structures.

Given a write-through cache with an error-detection capability of d -bits per-word and error-correction capability of c bits per word (with $d > c \geq 0$), the refresh period can be set to be just below the DRT of the $(d + 1)$ -th worst cell in a given word. During readout, more than c , but up to d errors are detected, the cache line is considered invalid, resulting in a cache miss. Subsequently, the cache line will be fetched from the lower level cache at the cost of a slight increase in miss-rate that can be accounted for. This approach achieves a very good trade-off between EDAC overhead and f_{ref} reduction. For example, by applying a common single error correction – double error detection (SECDED) code, the refresh-rate is set according to the probability that the DRT of three independent bits were to fail in a given word – resulting in an increase in refresh period by several orders-of-magnitude.

Note that this methodology still assumes worst-case operating conditions and process corners for the GC-eDRAM array. For error-tolerant applications, further savings could be achieved by applying typical conditions at the price of a higher probability of error.

IV. EDAC REALIZATION USING ECC

The most basic approach to error detection is to apply parity to stored data. The addition of a single bit, which is calculated as the XOR of the data word, enables the detection of a bit flip by repeating the XOR operation during readout. While this method is simple and incurs a minimal area cost, it can only detect up to one error per N_{par} bits and lacks the necessary information to correct the error (i.e., $d = 1, c = 0$). The area overhead of including a single parity bit for every N_{par} columns in the memory can be quantified as follows:

$$\text{Area}_{\text{parity}}(\%) = 100 \times \frac{1}{N_{\text{par}}} \quad (6)$$

This overhead is tolerable for a reasonably large value of N_{par} , but application of the methodology of Section III is applicable only to memories with backed up copies of the data, since no error correction capabilities are provided. In such cases, planning according to the worst cell is slightly relaxed, but it is still limited to setting the refresh-rate according to the probability that two errors occur in a given word.

In order to provide both error detection and correction, and thereby reduce the refresh-rate in standard applications, error correction codes can be implemented within the memory. These codes are typically implemented by the addition of extra bits per word and the integration of encoding (during write) and decoding (during read) to detect and correct a predefined number of bits. The most commonly used ECCs are SECDED extended-Hamming or Hsiao codes [16], [24]–[26]. These codes have the ability to correct a single error and detect up-to two errors in a single word, which, as previously discussed, enables setting the refresh-rate according to two DRT failures per-word for standard (error intolerant) applications and three DRT failures per-word for write-through caches. To further lower f_{ref} , multiple-bit EDAC (e.g., double error correction – triple error detection (DECTED)), can be achieved with more powerful ECCs, such as cyclic Bose-Chaundhuri-Hocquenghem (BCH) codes.

While the employment of advanced ECC schemes can detect and correct multiple errors, the addition of extra bits, along with the logic required for fault-detection and correction, incur significant energy, area, and delay overheads. For example, integrating an H(39,32) SECDED code requires the addition of 7 extra columns for storing the parity bits, which is over 20% of the total array area. In addition, complex encoding/decoding logic is required for writing and reading the data, adding approximately 13 gate delays to the read access time of the memory [22], [26]. The power and area overheads of adding more complex schemes is even more significant. For example, Wilkerson, *et al.* [27], proposed using a strong BCH code with an optimized decoder to reduce the refresh-rate of 1T-1C eDRAM implementing a last-level cache. However, this approach achieves high efficacy for lines with zero or one failures, otherwise requiring hundreds of cycles for ECC processing, which is not tolerable for memories with low access latency requirements, such as low-level caches. Alternative parallel [28] and two-dimensional decoding [29] implementations can achieve the necessary latency; however, this comes at the expense of large area and power overheads. To reduce these overheads, the ECCs can be integrated at the granularity of a cache line instead of single word [16], [30]. Per-line ECC reduces the number of redundant bits; however, it requires full-line access and read-before-write operations to check and update the redundant bits.

V. USING CDMR FOR REFRESH-RATE RELAXATION

The previous section overviewed the possibility of implementing various ECC approaches to enable refresh-rate relaxation. However, all of the standard solutions require either a very large area overhead, a very large latency overhead or both, especially when EDAC capabilities higher than SECDED are required. SECDED, or even parity solutions can be sufficient for certain applications, and in fact, are already integrated into many systems to deal with soft-errors. However, certain applications, such as L1 caches, may not be

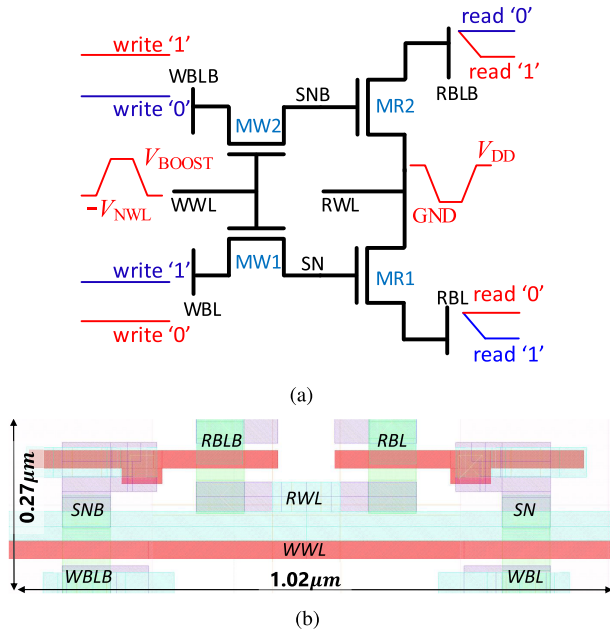


FIGURE 5. Proposed 4T CDMR gain cell. (a) Schematic representation. (b) Bitcell Layout.

able to tolerate the additional latency of even SECDDED solutions. Furthermore, when considering refresh-rate relaxation of GC-eDRAM memories through EDAC, SECDDED may not be sufficient, especially if soft-error tolerance is also required.

In order to resolve these issues, we propose using the recently reported CDMR technique [19] to achieve efficient refresh-rate relaxation for GC-eDRAM arrays. The CDMR technique employs a dense storage cell that, similar to SRAM, stores both the data and its complementary value in each bitcell; however, in this case, the additional information enables per-bit error detection. In addition, an error can be corrected for a set of p bits by adding a single parity bit per set. Furthermore, the error detection is achieved at a very low latency (a single logic gate delay), such that the approach is applicable to high performance memories, such as write-through L1 caches. Finally, the high-degree of EDAC achieved through the CDMR approach can provide refresh-rate relaxation, while also maintaining any required soft-error protection.

A CDMR gain-cell configuration is shown in Fig. 5(a), implemented with four NMOS transistors. The 4-transistor (4T) dynamic bitcell is composed of two write transistors (MW1 and MW2), two read transistors (MR1 and MR2), and two storage nodes (SN and SNB). The data and its complementary are stored on the parasitic capacitances of the storage nodes, mainly composed of the gate capacitances of MR1 and MR2, respectively. A detailed description of the cell can be found in [19]; however, for the intent of this discussion, only a '1' to '0' bit-flip due to a single-event upset (SEU) can reasonably occur. Moreover, the asymmetric DRT characteristics, depicted in Fig. 2(a), demonstrate that a DRT error is much more likely to occur when storing a '1'.

Therefore, an error is detected if both SN and SNB are read out as '0', which can be achieved with a single NAND gate. A parity bit can be used to resolve which of the storage nodes flipped, and the error can be corrected.

The layout of the 4T bitcell is shown in Fig. 5(b), drawn in a 28 nm bulk technology, and composed of minimal-sized NMOS devices. The layout features WWL routed in Poly, RWL routed in M2, RBL/RBLB and WBL/WBLB routed in M3. The cell was measured at $1.02 \mu\text{m} \times 0.27 \mu\text{m}$ ($0.275 \mu\text{m}^2$). For comparison, a 6T SRAM cell was redrawn in the same technology with standard logic design rules and measured at $0.884 \mu\text{m} \times 0.368 \mu\text{m}$ ($0.325 \mu\text{m}^2$). The resulting 4T gain-cell with inherent error detection is 15% smaller than a conventional 6T SRAM without error detection capabilities, and even higher area savings are achieved when taking into account the additional area overhead applied with the SRAM bitcells and logic used for ECC.

The EDAC capabilities of the CDMR approach provides very attractive opportunities. The granularity, p , of the addition of parity bits can be set according to the failure probability requirements of the system to provide the best trade-off between retention power/availability and area overhead, based on (4) with $k = 2$ and $N = p$. More impressively, for applications with a backup copy of the data, such as write-through caches, the memory can be operated without ever refreshing, instead, initiating a cache miss, when *any number* of errors is detected during a read operation. Due to the ability to both detect and correct (through a cache miss) any number of errors, theoretical worst-case failure probabilities become irrelevant, and all unnecessary design margins are eliminated, since a miss is only applied when a cache line with an actual error is accessed. This approach comes at the expense of an increased miss-rate; however, this can be fine-tuned by either adding error correcting parity bits or by setting a (very low) refresh-rate that is defined according to a typical case, which is several orders-of-magnitude better than the worst-case.

VI. COMPARISON

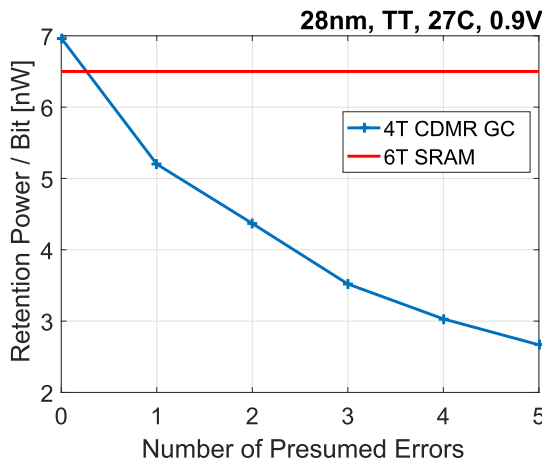
Fig. 6 shows a comparison between the retention power of the 4T CDMR gain-cell, composed of both the leakage and refresh power components, and the static power of a conventional 6T SRAM, composed of the leakage power of the cell, as a function of the number of presumed DRT failures, previously described in Section III. For a fair comparison, the wordline of the SRAM array was also biased to a negative voltage (V_{NWL}) during standby for leakage reduction. As expected, the static power consumption of the 6T SRAM cell is constant as it does not require refresh cycles to maintain its data. On the other hand, as the number of tolerated DRT failures increases, the retention power consumption decreases dramatically, achieving between 20%–60% power reduction over SRAM for a tolerated number of errors between 1–5.

The refresh power of a 256×64 4T CDMR array was evaluated using the DRT extracted from Fig. 2(a) and by extracting the parasitic components of the 4T cell layout with

TABLE 1. Comparison of embedded memories with error detection capabilities.

Memory type	6T SRAM			2T GC-eDRAM			Proposed 4T CDMR
Redrawn memory cell size	0.325 μm^2			0.1375 μm^2			0.275 μm^2
ECC mechanism	Parity bit	SECDED	DECTED	Parity bit	SECDED	DECTED	None
Error detection per word	1-bit	2-bit	3-bit	1-bit	2-bit	3-bit	inherent per-bit
Error correction per word	None	1-bit	2-bit	None	1-bit	2-bit	None
Soft error protection	Included			Not included			Included
Redundancy bits per 64b word	1	8	14	1	8	14	None
Decoder area (μm^2)	69	302	13551	69	302	13551	None
Decoder latency (ns)	0.3	1.4	1.1	0.3	1.4	1.1	None
Encoder area (μm^2)	69	69	141	141	426	426	31
Encoder latency (ns)	0.3	0.5	0.6	0.3	0.5	0.6	0.1
Retention power per 64b word @ 0.9V	416 nW	461 nW	507 nW	338 nW	310 nW	273 nW	180 nW

* Assuming up-to five presumed DRT errors.

**FIGURE 6.** Retention power comparison to 6T SRAM.

256 rows, as it mainly constitutes of the dynamic power used to charge the large bit-line and word-line capacitances. The leakage power of the 4T cell is much smaller than that of a 6T SRAM cell, as it does not contain connections to the supply voltage, but relies solely on its dynamic storage.

In order to demonstrate the reduction in overhead of the proposed 4T CDMR memory array compared to alternative embedded memories with error detection solutions, an area, delay, and power analysis was carried out, compared to 6T SRAM and 2T GC-eDRAM arrays with different error detection and correction schemes. These schemes include a single parity bit per word, providing a 1-bit error detection, a SECDED scheme based on a classical H(72,64) Hamming code, and a DECTED scheme based on Hsiao code. Cadence RTL Compiler was used to synthesize all decoder and encoder circuits to a TSMC 28 nm standard cell library, assuming a 64-bit memory word.

Table 1 shows the comparison of the considered approaches. The 4T CDMR provides an inherent per-bit error detection without ECC, with a 15% smaller bitcell area compared to a redrawn 6T SRAM bitcell. The total array area reduction becomes even more significant when taking into

account the additional bits required for the error detection and correction of conventional ECCs, as shown in the table. For a 2T GC-eDRAM, the usage of ECC for DRT error resilience limits its effectiveness in soft error detection and correction, which requires additional redundancy. On the other hand, the inherent per-bit error detection of the 4T CDMR provides soft error protection as well.

Moreover, the decoder and encoder area overheads become significant with the complexity of the ECC scheme employed in the array. A simple 1-bit error detection per word using a parity bit consumes 69 μm^2 for both the decoder and encoder circuits. For a DECTED scheme the area overhead becomes significantly higher, with 13551 μm^2 for the decoder and 426 μm^2 for the encoder, implemented in a parallel approach for reduced latency [28]. The encoder area of the proposed 4T CDMR consists of a single NAND gate per column for error detection, occupying only 31 μm^2 , which is a much smaller area overhead compared to the ECC schemes.

In terms of latency, the decoder circuits for the different ECCs add between 0.3 ns–1.4 ns, making multiple error detection and correction codes unfeasible for low-level cache implementations. On the other hand, the inherent-per bit detection of the proposed 4T CDMR structure only incurs a 0.1 ns delay due to a single NAND gate at the output of each column.

Finally, assuming 5 presumed DRT errors, the retention power of a 64-bit word of the proposed 4T CDMR array is 180 nW, which is a reduction of 57%–64% over the static power of a 6T SRAM with ECC, and 34%–46% reduction over the retention power of a 2T GC-eDRAM with ECC. Note that the power overhead of the decoder and encoder circuits was not included in this analysis, and it would result in even higher power savings. Moreover, as discussed in Section III-A, a more substantial reduction in power can be achieved by increasing the number of presumed DRT errors.

VII. IMPLEMENTATION AND MISS RATE ANALYSIS

In this section, we analyze the impact of implementing the proposed refresh rate relaxation approach on the performance

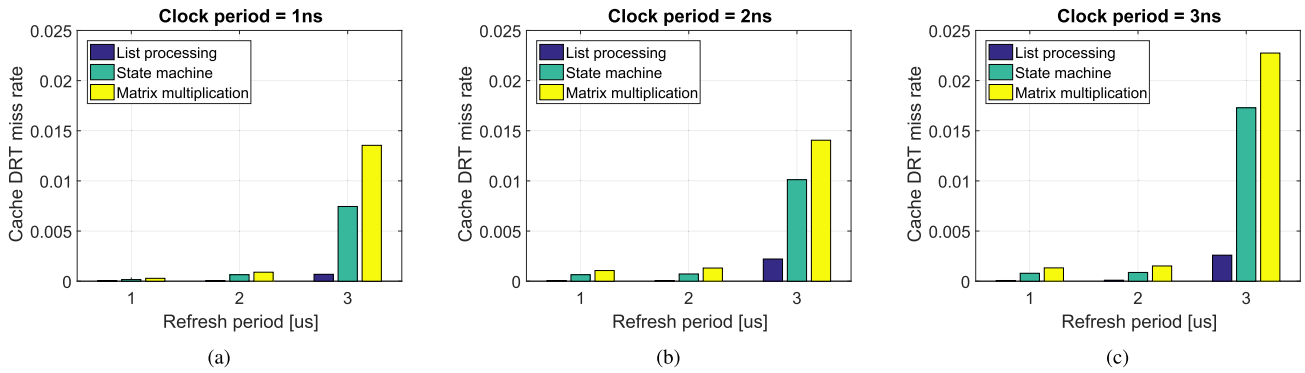


FIGURE 7. DRT miss rate under clock periods of (a) 1 ns, (b) 2 ns, and (c) 3 ns.

of a generic system, employing the proposed GC-eDRAM as a low-level cache. Due to the per-bit error detection capabilities of the CDMR bitcell, we assume that each presumed DRT error can be addressed as a conventional cache miss.

To evaluate the miss rate caused by relaxing the refresh period we captured the L1 cache memory access traces for the CoreMark benchmark [31], applied on a single-threaded RISC-V microprocessor with a 128 kB GC-eDRAM cache. The CoreMark benchmark implements several algorithms, including list processing (find and sort), matrix manipulation, and a state machine determining if an input stream contains valid numbers.

The extracted cache memory accesses were used to find the actual data lifetime in the memory, which is defined as the time between a write and read operation. Based on this data, and the estimated DRT extracted from circuit-level simulations, as described in Section II, the DRT miss rate was computed as a function of the chosen refresh period and the clock period, according to the following equation:

$$\text{Miss}_{\text{DRT}} = \int_0^{t_{\text{ref}}} \text{PDF}_{\text{lifetime}}(t) \cdot \text{CDF}_{\text{DRT}}(t) \cdot dt, \quad (7)$$

where t_{ref} is the chosen refresh period, $\text{PDF}_{\text{lifetime}}(t)$ denotes the empirical probability distribution function (PDF) of the data lifetime, and $\text{CDF}_{\text{DRT}}(t)$ is the CDF of the estimated DRT of the array, as extracted from Fig. 2(b).

The miss rate analysis was made under a worst case scenario, assuming that every data lifetime beyond the minimum DRT of the array and beneath the chosen refresh period results in a cache miss. Fig. 7 depicts the resulting DRT miss rate probability for different refresh and clock periods for list processing, state machine, and matrix multiplication algorithms. The miss rate increases with the refresh period due to a higher probability of a DRT failure, while an increased clock period increases the data lifetime. The resulting DRT miss rate varies between $4 \cdot 10^{-6}$ for list processing at a $5 \mu\text{s}$ refresh period and a 1 ns clock cycle to $2 \cdot 10^{-3}$ for matrix multiplication at a $15 \mu\text{s}$ refresh period and a 3 ns clock cycle.

VIII. CONCLUSIONS

Logic-compatible GC-eDRAM offers an alternative to SRAM due to its higher density and lower leakage power consumption. However, GC-eDRAM relies on dynamic storage, requiring frequent refresh cycles to maintain data. The refresh-rate is commonly set according to the DRT of the worst-cell in the array under extreme process variations and worst-case operating conditions to ensure error-free operation. In this paper, we propose lowering the refresh-rate of GC-eDRAM arrays using error detection, which is especially suitable for low-level caches with backed up copy of the data. To implement a low-cost error detection mechanism, we propose using a 4T CDMR gain-cell array, offering per-bit error detection with low area and latency overheads. The proposed approach was compared to 6T SRAM and GC-eDRAM with integrated ECC, offering significant area and latency reductions, as well as over 60% power reduction over 6T SRAM by allowing up-to five presumed DRT errors.

ACKNOWLEDGEMENTS

The authors would like to thank Udi Kra for his help in the miss-rate analysis of the proposed technique.

REFERENCES

- [1] ITRS. (2015). *International Technology Roadmap for Semiconductors*. [Online]. Available: <http://www.itrs2.net>
- [2] T. Luo et al., "Dadiannao: A neural network supercomputer," *IEEE Trans. Comput.*, vol. 66, no. 1, pp. 73–88, Jan. 2017.
- [3] A. Teman, "Dynamic stability and noise margins of SRAM arrays in nanoscaled technologies," in *Proc. IEEE Faible Tension Faible Consommation*, May 2014, pp. 1–5.
- [4] L. Atias, A. Teman, R. Gitterman, P. Meinerzhagen, and A. Fish, "A low-voltage radiation-hardened 13t SRAM bitcell for ultralow power space applications," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 24, no. 8, pp. 2622–2633, Aug. 2016.
- [5] P. Meinerzhagen, A. Teman, R. Gitterman, N. Edri, A. Burg, and A. Fish, *Gain-Cell Embedded DRAMs for Low-Power VLSI Systems-on-Chip*. Cham, Switzerland: Springer, 2018.
- [6] A. Teman, P. Meinerzhagen, R. Gitterman, A. Fish, and A. Burg, "Replica technique for adaptive refresh timing of gain-cell-embedded DRAM," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 61, no. 4, pp. 259–263, Apr. 2014.
- [7] R. Gitterman, A. Teman, P. Meinerzhagen, L. Atias, A. Burg, and A. Fish, "Single-supply 3T gain-cell for low-voltage low-power applications," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 24, no. 1, pp. 358–362, Jan. 2016.

- [8] P. Meinerzhagen, A. Teman, R. Gitterman, A. Burg, and A. Fish, "Exploration of sub-VT and near-VT 2T gain-cell memories for ultra-low power applications under technology scaling," *J. Low Power Electron. Appl.*, vol. 3, no. 2, pp. 54–72, 2013.
- [9] R. Gitterman, A. Teman, P. Meinerzhagen, A. Burg, and A. Fish, "4T gain-cell with internal-feedback for ultra-low retention power at scaled CMOS nodes," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, Jun. 2014, pp. 2177–2180.
- [10] N. Edri, P. Meinerzhagen, A. Teman, A. Burg, and A. Fish, "Silicon-proven per-cell retention time distribution model of gain-cell based eDRAM," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 63, no. 2, pp. 222–232, Feb. 2016.
- [11] R. Gitterman, A. Fish, N. Geuli, E. Mentovich, A. Burg, and A. Teman, "An 800 MHz mixed-VT 4T gain-cell embedded DRAM in 28 nm CMOS bulk process for approximate computing applications," in *Proc. IEEE Eur. Solid-State Circuits Conf. (ESSCIRC)*, Sep. 2017, pp. 308–311.
- [12] A. Teman, P. Meinerzhagen, R. Gitterman, A. Fish, and A. Burg, "Replica technique for adaptive refresh timing of gain-cell-embedded DRAM," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 61, no. 4, pp. 259–263, Apr. 2014.
- [13] A. Teman, G. Karakonstantis, R. Gitterman, P. Meinerzhagen, and A. Burg, "Energy versus data integrity trade-offs in embedded high-density logic compatible dynamic memories," in *Proc. Design, Autom. Test Eur. Conf. Exhib. (DATE)*, Mar. 2015, pp. 489–494. [Online]. Available: <http://dl.acm.org/citation.cfm?id=2755753.2755864>
- [14] S. Ozdemir, D. Sinha, G. Memik, J. Adams, and H. Zhou, "Yield-aware cache architectures," in *Proc. 39th Annu. IEEE/ACM Int. Symp. Microarchitecture*, Dec. 2006, pp. 15–25.
- [15] K. C. Mohr and L. T. Clark, "Delay and area efficient first-level cache soft error detection and correction," in *Proc. Int. Conf. Comput. Design (ICCD)*, Oct. 2007, pp. 88–92.
- [16] J. Kim, N. Hardavellas, K. Mai, B. Falsafi, and J. Hoe, "Multi-bit error tolerant caches using two-dimensional error coding," in *Proc. 40th Annu. IEEE/ACM Int. Symp. Microarchitecture*, Dec. 2007, pp. 197–209.
- [17] H. Dai, S. Zhao, J. Zhang, M. Qiu, and L. Tao, "Security enhancement of cloud servers with a redundancy-based fault-tolerant cache structure," *Future Gener. Comput. Syst.*, vol. 52, pp. 147–155, Nov. 2015.
- [18] R. Gitterman, A. Fish, N. Geuli, E. Mentovich, A. Burg, and A. Teman, "An 800-MHz mixed-V_T 4T IFGC embedded DRAM in 28-nm CMOS bulk process for approximate storage applications," *IEEE J. Solid-State Circuits*, vol. 53, no. 7, pp. 2136–2148, Jul. 2018.
- [19] R. Gitterman, L. Atias, and A. Teman, "Area and energy-efficient complementary dual-modular redundancy dynamic memory for space applications," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 25, no. 2, pp. 502–509, Feb. 2017.
- [20] R. Gitterman, A. Teman, P. Meinerzhagen, A. Burg, and A. Fish, "A process compensated gain cell embedded-DRAM for ultra-low-power variation-aware design," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2016, pp. 1006–1009.
- [21] R. Gitterman, A. Fish, A. Burg, and A. Teman, "A 4-transistor nMOS-only logic-compatible gain-cell embedded dram with over 1.6-ms retention time at 700 mV in 28-nm FD-SOI," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 65, no. 4, pp. 1245–1256, Apr. 2018.
- [22] S. Ganapathy, G. Karakonstantis, A. Teman, and A. Burg, "Mitigating the impact of faults in unreliable memories for error-resilient applications," in *Proc. 52nd Annu. Design Autom. Conf.*, Jun. 2015, p. 102.
- [23] S. Ganapathy, A. Teman, R. Gitterman, A. P. Burg, and G. Karakonstantis, "Approximate computing with unreliable dynamic memories," in *Proc. Int. New Circuits Syst. Conf. (NEWCAS)*, Jun. 2015, pp. 1–4.
- [24] R. W. Hamming, "Error detecting and error correcting codes," *Bell Syst. Tech. J.*, vol. 29, no. 2, pp. 147–160, Apr. 1950.
- [25] M. Y. Hsiao, "A class of optimal minimum odd-weight-column SEC-DED codes," *IBM J. Res. Develop.*, vol. 14, no. 4, pp. 395–401, Jul. 1970.
- [26] D. Rossi, N. Timoncini, M. Spica, and C. Metra, "Error correcting code analysis for cache memory high reliability and performance," in *Proc. IEEE Design, Autom. Test Eur. Conf. Exhib. (DATE)*, Mar. 2011, pp. 1–6.
- [27] C. Wilkerson, A. R. Alameldeen, Z. Chishti, W. Wu, D. Somasekhar, and S.-L. Lu, "Reducing cache power with low-cost, multi-bit error-correcting codes," *ACM SIGARCH Comput. Archit. News*, vol. 38, no. 3, pp. 83–93, 2010.
- [28] R. Naseer and J. Draper, "Parallel double error correcting code design to mitigate multi-bit upsets in SRAMs," in *Proc. 34th Eur. Solid-State Circuits Conf. (ESSCIRC)*, Sep. 2008, pp. 222–225.
- [29] Y.-H. Yu, P.-H. Wang, S.-J. Tsai, and T.-F. Chen, "A latency-elastic and fault-tolerant cache for improving performance and reliability on low voltage operation," in *Proc. Int. Symp. VLSI Design, Autom. Test (VLSI-DAT)*, Apr. 2015, pp. 1–4.
- [30] H. Farbeh and S. G. Miremadi, "PSP-cache: A low-cost fault-tolerant cache memory architecture," in *Proc. IEEE Design, Autom. Test Eur. Conf. Exhib. (DATE)*, 2014, pp. 1–4.
- [31] *Coremark Benchmark*, Embedded Microprocessor Benchmark Consortium, 2013.

Authors' photographs and biographies not available at the time of publication.

...