Mitigating Process Variability for Non-Volatile Cache Resilience and Yield

Soheil Salehi, Navid Khoshavi, and Ronald F. DeMara

Department of Electrical and Computer Engineering, University of Central Florida, Orlando, FL 32816-2362

Abstract—While inclusion of emerging technology-based Non-Volatile Memory (NVM) devices in on-chip memory subsystems offers excellent potential for energy savings and scalability, their sensing vulnerability creates Process Variation (PV) challenges. This paper presents a circuit-architecture cross-layer solution to realize a radically-different approach to leveraging as-built variations via specific Sense Amplifier (SA) design and use. This novel approach, referred to as a Self-Organized Sub-bank (SOS) design, assigns the preferred SA to each Sub-Bank (SB) based on a PV assessment, resulting in energy consumption reduction and increased read access reliability. To improve the PV immunity of SAs, two reliable and power efficient SAs, called the Merged SA (MSA) and the Adaptive SA (ASA) are introduced herein for use in the SOS scheme. Furthermore, we propose a dynamic PV and energy-aware cache block migration policy that utilizes mixed SRAM and STT-MRAM banks in Last Level Cache (LLC) to maximize the SOS bandwidth. Our experimental results indicate that SOS can alleviate the sensing vulnerability by 89% on average, which significantly reduces the risk of application contamination by fault propagation. Furthermore, in the light of the proposed block migration policy, write performance is improved by 12.4% on average compared to the STT-MRAM-only design.

Index Terms— Magnetic Tunneling Junction (MTJ), Spin-Transfer Torque storage elements, STT-MRAM, Self-referencing MTJ, Reliability, Process Variation, Read/Write Reliability, Sub-banking, Last Level Cache (LLC), Sense Amplifier (SA) design.

1 Introduction

COMPLEMENTARY Metal on Oxide Semiconductor (CMOS) device scaling continues to increase the need to identify viable approaches for reducing leakage power. An alternative to CMOS-based memory devices is offered by emerging technology memory devices that contribute inherent features of Non-Volatile Memory (NVM) capabilities. With attributes of non-volatility, near-zero standby energy, and high density, Spin Transfer Torque Magnetic RAM (STT-MRAM) has emerged as a promising alternative post-CMOS technology for embedded memory applications. In order to practically implement these NVMs, various techniques to mitigate the specific reliability challenges associated with STT-MRAM elements are surveyed, classified, and assessed in [1]. They identify various solutions to the reliability issues within a taxonomy of current and future approaches to reliable STT-MRAM designs [1]. Despite the range of approaches available to mitigate Process Variation (PV), it remains as one of the most negatively influential factors impacting STT-MRAM technology performance from the perspectives of delay and energy consumption [1]. Furthermore, the Sense Margin (SM), which is an important parameter of the tolerance in sensing the resistive state of emerging NVM devices, varies considerably in the presence of PV of the devices which comprise the bit-cell and their associated sensing circuits [1]. SM is also known as the difference between bit-line voltage and reference voltage. These variations may then result in erroneous data sensing operations, read disturbance, readability degradation at scaled technology nodes, and retention failure [1]. These reliability issues have increased the demand for designing advanced low-power approaches with reliable sensing circuits to mitigate and leverage PV for improved performance and reliability of NVMs, including increasing the SM and finding the optimum read current and latency [1].

In an effort to mitigate and leverage the increased effects of PV in deeply-scaled memory devices, the baseline concept of a Self-Organized Sub-bank (SOS) approach was recently proposed in [2]. SOS focuses on mitigating and leveraging PV in order to provide reliable sensing operation by matching the as-built resource performance with the applications' usage demands while taking the energy budget into consideration. In order to achieve these goals, SOS partitions STT-MRAM data arrays into several Sub-Banks (SBs), which are evaluated using a Power-On Self-Test (POST) phase. The POST assesses the PV impact on the SBs, and then, each SB will be assigned an Energy-Aware Sense Amplifier (SA) or a High-Resilience SA with regard to a predefined bit error threshold. Based on the results provided in [2], SOS reduces the risk of contaminating the application's data structure by fault propagation as described herein. Furthermore, several designs have been proposed to address the large incubation delay in writing to STT bit-cells [3-5]. In recent years, several hybrid spintronic-CMOS cache designs have been proposed to improve the write performance while offering much larger cache capacity with low leakage power [5]. Some of these works such as [6], [7], and [8] offer solutions for predicting write-intensive blocks and using migration algorithms,

• *S. Salehi, N. Khoshavi, and R.F. DeMara are with the Department of Electrical Engineering and Computer Science, University of Central Florida, 4328 Scorpius Street, Harris Engineering Center, Bldg#116, Mail-room#345, Orlando, FL 32816-2362. E-mail: soheil.salehi@knights.ucf.edu, navid.khoshavi@knights.ucf.edu, demara@mail.ucf.edu.*

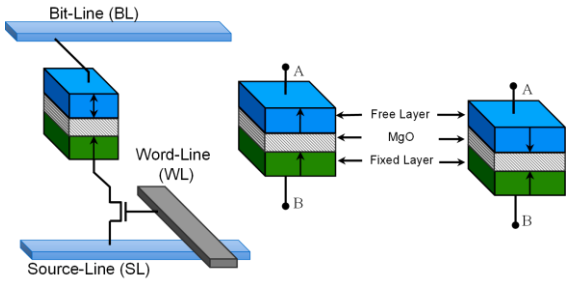
place those write-intensive blocks in the SRAM ways to reduce the energy consumption and delay as well as increase the performance. While the approach proposed in [6] only works for core-write operations, the Access Pattern Predictor (APP) proposed in [7] and the Prediction Hybrid Cache (PHC) proposed in [8] cover all different write operations. Additionally, [8] offers dynamic threshold adjustment that allows the threshold of write intensity to change based on the characteristics of the application. Some of the recently published works such as [9] suggest frequent movement of written cache blocks to other STT-MRAM or SRAM lines to reduce the write variance of STT-MRAM lines, however such approaches often result in unnecessary energy consumption, which can lower the performance.

These methodologies have inspired us to maximize the efficiency and reliability of SOS by proposing a dynamic PV-aware and Energy-aware cache block migration policy as a circuit-architecture solution for hybrid memory devices that utilize a combination of SRAM and STT-MRAM banks in Last Level Cache (LLC). The proposed approach migrates the data among cache blocks within the LLC so that the data with more frequent write operations are moved to SRAM cache blocks, whether they are in high-PV impacted regions or not. Additionally, the proposed approach utilizes SOS to transfer the data with more frequent read operations to STT-MRAM cache blocks that suffer less from PV. As a result, read-intensive operations migrate to low PV regions of the LLC and SBs with less frequent read operations are allocated to high PV regions of the LLC. We identify herein how an SOS-enabled hybrid cache approach can significantly improve cache utilization and bank accessibility while reducing energy consumption and increasing reliability, since SOS allocates the SA with better energy profile to low PV regions and the SA with better reliability profile to high PV regions.

The remainder of the paper is organized as follows: First, background on STT-MRAM and its reliability challenges is provided in Section 2. In Section 3, SOS is elaborated in detail and data-sensing fault models are introduced. Furthermore, in Section 3, new SA circuits are introduced and a comparison is provided. In Section 4, the proposed SOS-enabled hybrid cache is elaborated. Circuit-level simulation results and analysis for the proposed SA designs are provided in Section 5. Architecture-level experimental results and analysis are provided in Section 6. Finally, Section 7 concludes this paper with broader recommendations regarding the proposed circuit and architectural approaches.

2 Overview of MTJ-based NVM Operation

The basic concept of spin-based NVM devices is to control the intrinsic spin of electrons in a ferromagnetic thin film based solid-state nano-device. Fig. 1a shows a STT-MRAM cell structure with a single transistor, known as "one-transistor-one-MTJ (1T-1R)" [1]. Each bit cell is accessed via the corresponding bit-line within the resident word selected by the word line. The non-volatile Magnetic Tunnel Junction (MTJ) consists of two ferromagnetic layers, which are called the fixed layer and the free layer,



(a) (b)

Fig. 1: (a) 1T-1R STT-MRAM cell structure, (b) Right: Anti-parallel (high resistance), Left: Parallel (low resistance).

and one tunneling oxide layer between the two FM layers [1]. FM layers could be aligned in two different magnetization configurations, parallel (P) and antiparallel (AP). Accordingly, the MTJ exhibits a low resistance (R_P) or high resistance (R_{AP}), respectively [1]. Based on STT switching principles, the P or AP state of the MTJ is configured by means of the bidirectional current that passes through it, I_{MTJ}, which could readily be produced by simple MOS based circuits. The states of the MTJ are switched when I_{MTJ} becomes higher than a critical current, I_C. Magnetic Tunnel Junction (MTJ) devices are constructed with layered pillars of ferromagnetic and insulating materials to utilize magnetic orientations that can be controlled and sensed in terms of electrical signal levels as shown in Fig. 1b. The MTJ resistance in P (θ=0°), and AP (θ=180°) states is expressed by the following equations:

$$R(\theta) = 2R_{MTJ} \times \frac{1+TMR}{2+TMR+TMR.cos\theta} = \begin{cases} R_p = R_{MTJ}, & \theta = 0° \\ R_{ap} = R_{MTJ}(1+TMR), & \theta = 180° \end{cases} \quad (1)$$

$$R_{MTJ} = \frac{t_{ox}}{Factor \times Area.\sqrt{\varphi}} \exp(1.025 \times t_{ox}.\sqrt{\varphi}) \quad (2)$$

$$TMR = TMR_0 / 1 + (\frac{V_b}{V_h})^2 \quad (3)$$

where V_b is the bias voltage, V_h = 0.5V is the bias voltage when Tunnel Magneto-Resistance (TMR) ratio is half of the TMR_0, t_{ox} is the oxide thickness of MTJ, Factor is obtained from the resistance-area product value of the MTJ that relies on the material composition of its layers, Area is the surface area of the MTJ, and φ is the oxide layer energy barrier height [10]. Despite all of the merits that STT-MRAM offers, violation of reliability tolerances may result in read and/or write failures [1]. Thermal fluctuations and other issues such as MTJ PV and the CMOS peripheral circuit PV have severely limited the scalability of STT-MRAM devices [1]. Also, as a result of these issues, there is an increased demand for advanced sensing circuits that can provide an adequate SM along with low power operation.

3 Self-Organized Sub-banks (SOS)

3.1 SOS Schematic for SA Assignment

Two of the most frequently used SAs, the Pre-Charge Sense Amplifier (PCSA) as shown in Fig. 2a [11] and the

Separated Pre-Charge Sense Amplifier (SPCSA) as shown in Fig. 2b [12], each have their own benefits and drawbacks. PCSA offers an improved energy profile compared to SPCSA. However, SPCSA offers a more reliable sensing operation. By combining the PCSA and SPCSA, the Merged Sense Amplifier (MSA) [2] is realized to utilize each SA's properties to increase performance and reliability. In order to improve energy efficiency of MSA proposed in [2], the selectors **MUX1** and **MUX2** are included in order to make sure only one SA is operating and to avoid unnecessary energy consumption by gating the **SEN** signal of the offline SA as shown in Fig. 2c. In PCSA, during the pre-charge stage, **SEN** signal is low, turning **MN2** off while turning **MP0** and **MP3** on. This will precharge the output nodes **OUT** and $\overline{\textbf{OUT}}$ to **VDD**. As a result, **MN0** and **MN1** will turn on while **MP1** and **MP2** are still off. As soon as the sensing stage begins, **MP0** and **MP3** turn off and **MN2** turns on. Thus, based on the difference between **MTJ0** and **MTJ1** resistance, which is determined by the magnetization orientation of their free layer compared to their fixed layer, one of the output nodes begins to discharge more rapidly to **GND**, leading either **MP1** or **MP2** to turn on and charge the other output to **VDD**. In SPCSA, during the precharge stage, **SEN** signal is low, turning **MN4** off while turning **MP0**, **MP1**, **MP4**, and **MP5** on. This will precharge the output nodes **OUT**, $\overline{\textbf{OUT}}$, **Node0**, and **Node1** to **VDD**. As a result, **MN0** and **MN1** will turn on while **MP2**, **MP3**, **MN2**, and **MN3** are still off. As soon as the sensing stage begins, **MP0**, **MP1**, **MP4**, and **MP5** turn off and **MN4** turns on. Thus, in the secondary discharge path, based on the difference between **MTJ0** and **MTJ1** resistances, one of the two intermediary output nodes, **Node0** or **Node1** , begins to discharge more rapidly to **GND**. This will lead one of the **INV0** or **INV1** output to turn on **MN2** or **MN3**, respectively, which then will cause the primary discharge path to activate and discharge one of the output nodes **OUT** or $\overline{\textbf{OUT}}$ more rapidly to **GND**, resulting in either **MP2** or **MP3** to turn on and charge the other output to **VDD**.

Herein, we propose an alternative for MSA that further improves energy consumption and reliability due to PV. The Adaptive Sense Amplifier (ASA), as shown in Fig. 3c, has a functionality similar to MSA described in [2]. However, by utilizing the Energy Aware Sense Amplifier (EASA), as shown in Fig. 3a, and the Variation Immune Sense Amplifier (VISA), as shown in Fig. 3b, instead of PCSA and SPCSA, it can achieve better energy and reliability profile respectively [13]. Like MSA, **MUX1** and **MUX2** are included in ASA to reduce energy consumption by gating the **SEN** signal of the offline SA so that only one SA is operating. SPCSA and VISA both increase reliability by reducing the amount of resistance in the MTJ read paths, which increases the SM and voltage headroom of the SA, resulting in a more reliable sensing. Increasing voltage headroom is an important issue in scaled technology nodes since the supply voltage is reduced to 1 volt or below, and even a small voltage drop can result in a sensing error [12]. EASA and VISA were proposed in [13] as alternatives to PCSA and SPCSA, respectively and they offer better performance compared to their counterpart. In order to achieve these improvements, Transmission Gates (TGs) were utilized to improve the voltage headroom [13]. TGs provide near optimal full-swing switching, and as it has been shown in [14], using TGs, can help reduce the vulnerability to reliability issues caused by PV. In addition, using TGs, as presented in [15], can help reduce the energy consumption by reducing the leakage energy. Thus, **TG0**, **TG1**, and **TG2** are added to improve the performance of the PCSA as shown in Fig. 3a [13], and to improve the reliability of SPCSA as shown in Fig. 3b [13]. In EASA, during the pre-charge stage, **TG0**, **TG1**, and **TG2** are off, resulting in a reduction of leakage energy from output nodes, **OUT** and $\overline{\textbf{OUT}}$, that are pre-charged to **VDD**. During the sensing stage, **TG0**, **TG1**, and **TG2** turn on and the output nodes start to discharge to **GND**. Based on the resistance difference

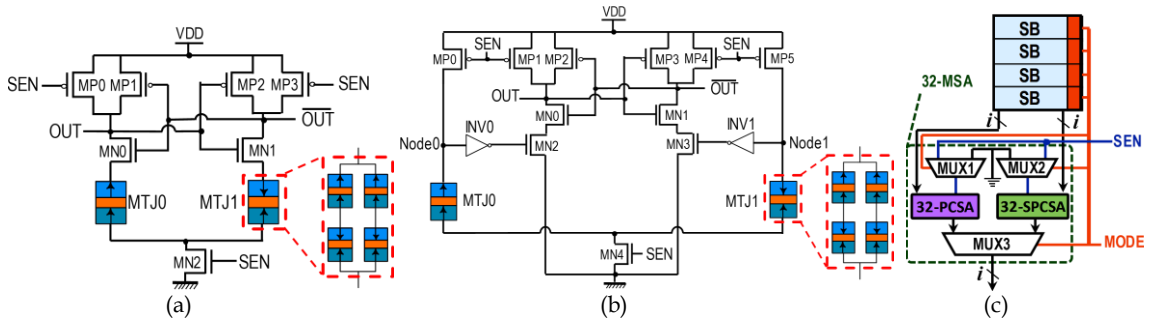


Fig. 2: (a) PCSA (MTJ1: Reference MTJ), (b) SPCSA (MTJ1: Reference MTJ), and (c) MSA (SB: Sub-Bank).

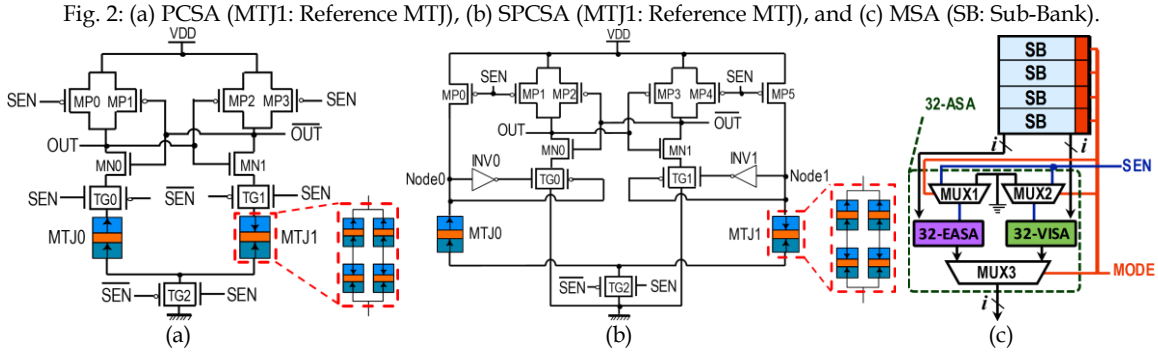


Fig. 3: (a) EASA (MTJ1: Reference MTJ), (b) VISA (MTJ1: Reference MTJ), and (c) ASA (SB: Sub-Bank).

between the two MTJ branches with regard to the MTJs' states, one of the two output nodes begins to discharge more rapidly, leading the other output to charge to **VDD**. EASA offers reduced energy consumption by reducing the leakage, however including the TGs on the path of MTJs results in increased resistance of the branches, which will reduce the SM and may result in decreased reliability.

Similar to EASA, in VISA, during the pre-charge stage all the TGs will be turned off, resulting in reduced leakage energy, and both **OUT**, $\overline{\textbf{OUT}}$, **Node0**, and **Node1** will be charged to **VDD**. During the sensing stage, **TG2** will turn on and in the separated part of the SA, based on the resistance difference between the two branches with MTJs with regard to the MTJs' states, one of the two intermediary output nodes, **Node0** or **Node1**, begins to discharge more rapidly. Then, based on the voltage potential of the intermediary outputs, either **TG0** or **TG1** will turn on faster and one of the main branches of the SA begins to discharge quicker, resulting in the output node of that branch to drop and charge the other branch's output node to **VDD** [13]. **INV0** and **INV1** are used to amplify the voltage difference of **Node0** and **Node1** of the SA. Using **TG0** and **TG1** and utilizing **Node0** and **Node1** as well as their amplified value, the authors have reduced the effects of PV by reducing the chance of failure due to device mismatch in the inverters. Furthermore, by utilizing **TG2**, energy consumption is reduced due to the reduction in the leakage energy [13]. As shown in Fig. 2 and Fig. 3, an alternative referencing configuration is used to further improve the reliability of the SAs. Using $(MTJ_P+MTJ_{AP})||(MTJ_P+MTJ_{AP})$ configuration for the reference MTJ, referred to as MTJ1 in Fig. 2 and Fig. 3, a reference value of $(MTJ_P+MTJ_{AP})/2$ is achieved, which provides increased SM [13]. The schematic of different SOS designs is depicted in Fig. 2c and Fig. 3c, and the process for assigning the preferred SA to each SB is shown in Algorithm 1 for MSA and ASA. As shown in Algorithm 1, SOS starts with a POST function. In both SA designs, after the POST function, an analyzer function is called to determine the preferred SA for that particular SB. A select input is used in the circuit called **MODE** to choose between the two SAs based on the assigned bit set value as shown in Fig. 2c and Fig. 3c. If the logic 1 is assigned to input **MODE**, then the circuit will operate in PCSA mode in MSA or EASA mode in ASA. On the other hand, if logic 0 is assigned to **MODE**, it will change the operation of the SA to SPCSA mode in MSA or VISA mode in ASA. As discussed earlier in this Subsection, in both MSA and ASA, the **SEN** signal is gated for the SA that is not in use to increase energy saving of the SA. In other words, only one SA will turn on, and the other SA's **SEN** signal will be connected to **GND**, which results in **OUT** and $\overline{\textbf{OUT}}$ to be 1 at all times.

Algorithm 1: SOS Approach to Assign Preferred SA to Sub-bank

```
Function SOS() /*SOS Approach for SA Assignment*/
1  for ∀ cache line ∈ LLC do
2      for ∀ sub − bank ∈ cache line do
           begin
3              POST() /*Power-On Self-Test*/
4              Analyzer() /*Evaluate the correctness of the outputs*/
Function POST() /*Power-On Self-Test*/
begin
5      set SEN = 1 /*start the discharge and evaluation stage*/
6      if output != expected-value then
7          ++number-wrong-outputs /*increment number of wrong
             outputs*/
8      set SEN = 0 /*keep the sense signal in pre-charge stage*/
Function Analyzer() /*Evaluate the correctness of the outputs*/
begin
9      if number-wrong-outputs > threshold then
10         set MODE = 0 /*assign MSA-SPCSA or ASA-VISA to
             sub-bank*/
           /*MUX3 takes sensed data from MSA-SPCSA or ASA-VISA
             to output*/
           /*MUX1 selects SEN signal to activate MSA-SPCSA or
             ASA-VISA and deactivate MSA-PCSA or ASA-EASA*/
       else
11         set MODE = 1 /*assign MSA-PCSA or ASA-EASA to
             sub-bank*/
           /*MUX3 takes sensed data from MSA-PCSA or ASA-EASA
             to output*/
           /*MUX2 selects SEN signal to activate MSA-PCSA or
             ASA-EASA and deactivate MSA-SPCSA or ASA-VISA*/
```

3.2 Extracting the PV Parameters

In our PV modeling process, we assume that the cache tag and peripherals (e.g., row decoder, column decoder, row buffer and SAs) are fabricated at the CMOS layer while memory cells are realized through MTJ devices. Since the MTJs are vertically stacked on top of the CMOS layer and these components are tightly coupled to realize the function of STT-MRAM, the SM varies readily based on the effect of PV on that particular region of the die. Accordingly, we consider the same PV parameters to model both CMOS and MTJ variations in VARIUS [16] which is based on static analysis tool R [17] and geoR packages [18]. Table 1 lists the circuit parameters and their standard deviation considered for PV analysis. As listed in Table 1, we choose the standard deviation of the parameters in alignment with the previous measurements reported in [19-21]. Among a large pool of maps that are generated by VARIUS with a resolution of one million (1,000×1,000) sample points, one map is randomly selected. The degree of variation is shown by a range of colors. Each color corresponds to a specific value of sample points as shown in Fig. 4. In our simulation, we consider the amount of PV for each site based on the location of the LLC components within the floorplan and their associated sample points. Thus, VARIUS generates a relatively accurate estimation of the impact of PV on the read SM of each SB.

3.3 Power On Self-Test (POST)

As shown in Fig. 4, the cache bank floorplan of the STT-MRAM layer is superimposed on the map. In our SOS approach, each cache bank is partitioned into 16 SBs. The size of each SB is matched with the word size to maintain the energy consumption of the tag to be as low as possible, e.

Table 1: Technology Parameters

Parameter				*Value*	*Std. Dev.*
PMOS	*V_{th} (Threshold Voltage)*			460mV	10%
	Width /Length $(W/L)_P$			2 & 4	1%
NMOS	*V_{th} (Threshold Voltage)*			500mV	10%
	Width /Length $(W/L)_N$			1 & 2	1%
MTJ	*MgO Thickness*			0.85nm	Effects of variation are applied to TMR
	Shape Area	*main MTJ (MTJ0)*		$(\frac{\pi}{4})$x40x40nm^2	
		reference MTJ (MTJ1)	*MTJ_{AP}*	$(\frac{\pi}{4})$x30x30nm^2	
			$(MTJ_P+MTJ_{AP})/2$	$(\frac{\pi}{4})$x40x40nm^2	
	ϕ (Potential Barrier Height)			0.4 V	N/A
	R·A (Resistance Area Product)			5Ω·μm^2	N/A
	α (Damping Factor)			0.01	N/A
	TMR (Tunnel Magneto Resistance)			100%	1% & 10%
Nominal Voltage				1.0 V	N/A
SEN Signal Period (T)				1ns	N/A

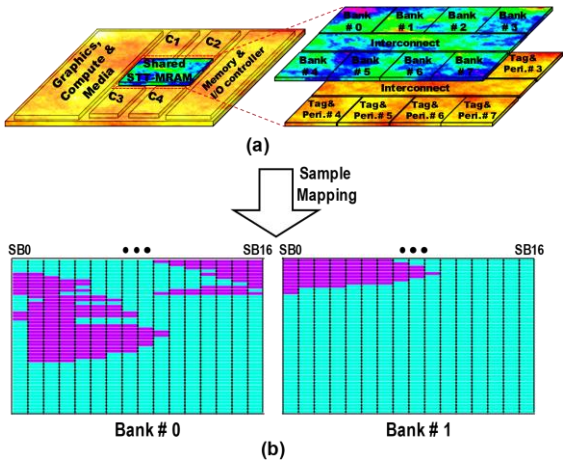


Fig. 4: a) PV map of a 4-core CMP, b) Determining preferred SA based on post-fabrication SB PV resiliency assessment.

g. 32-bits in our case study. We consider one additional bit per SB to identify the preferred SA for that particular SB during post-fabrication resiliency assessment to PV. The POST phase is basically a March Test that targets PV-induced faults in STT-MRAM [22]. Similar to the widely-used March test, during the POST operation, first we write 0 to all memory cells, then we read the memory cells and then we write 1 to all memory cells and then read again. Based on the outcome of all read operations we will be able to find the number of erroneous outputs and based on that is possible to recognize the high-PV regions. We assume the proposed SRAM March Test with O(n) test length can be utilized for our purpose because the tag and peripherals of STT-MRAM are considered to be implemented in the CMOS layer. Thus, variation-induced delay faults in both SRAM and STT-MRAM manifests itself as the same fault model as an insufficient pre-charge period, insufficient discharge and evaluation period, insufficient amplify time, disturbance of sense operation, and simultaneously activation of multiple word lines.

In this regard, PV-aware March Test examines all STT-MRAM data arrays and performs a sequence of operations (e. g., exhaustive pair-wise address transitions) to identify PV-induced delay faults in each cell [22]. If the error rate of the impacted STT-MRAM cells in a SB exceeds the predefined threshold, the extra bit is set to '0' indicating that an array of reliable SAs are required for sensing the data of this SB. Otherwise, the extra bit is set to '1', which indicates that an array of low-power SAs offering reduced delay and power consumption can be considered for that particular SB. Since POST is a one-time operation, it will not impact the performance of the memory as a whole, resulting in a negligible overhead.

3.4 Fault Models Associated with Sensed Data

In the PARSEC suite, when considering the presence of PV, around 27.5% of the sensed data when utilizing a STT-MRAM based LLC has the potential to be incorrect, 6% of which will be overwritten prior to being used by the processor or to be committed to the main memory, on average. Despite the fact that 6% might not be significant, a substantial portion of incorrectly sensed data requires handling before manifesting themselves as wrong outputs, application crashes, or prolonged program executions [23]. To be specific, we classify the outcomes of SA operation to the following categories for broad adaption:

- **True Data Sensing (TDS):** The sensed data value is identical to the value stored in the STT-MRAM cell.
- **Vulnerable False Data Sensing (VFDS):** The sensed data value differs from the value stored in the STT-MRAM cell, which propagates out of cache to be either used by the process or committed to other levels of memory [23].
- **Non-Vulnerable False Data Sensing (NVFDS):** The sensed data value differs from the value stored in the STT-MRAM cell, however the replica copy of the sensed false data in the upper levels of cache will be overwritten by a write operation prior to being used. During a block eviction, replica data becomes written back to the lower levels of cache because it is a dirty victim block. Thus, this benign fault does not threaten the semantic correctness.

Based on these categories, the experiment concentrates on the faults that are caused by incorrectly sensed data rather than alternative fault models that can impact the stored value in STT-MRAM cells [24].

4 Circuit-Architecture Solution for Hybrid Emerging Memory Devices

As described in Section 1, hybrid cache designs have been proposed in the past to improve write performance while offering much larger cache capacities with low leakage power [5]. As mentioned earlier in Section 1, hybrid CMOS/NV cache designs have been proposed in the past to sustain write performance while achieving significantly larger capacities at reduced average leakage power [4, 7, 8, 25]. The previous works are considered from two viewpoints. First, with respect to leveraging referencing behavior in hybrid caches, Wu et al. [25] proposed Read-Write aware Hybrid Cache Architecture (RWHCA). It partitions a hybrid CMOS/STT-MRAM cache into read and write regions. By leveraging proper-ties of intra-cache data movement, RWHCA reduces the power by 55% on average, while providing 5% IPC improvement compared to the baseline SRAM cache across 30 workloads.

Alternatively, Wang et al. [7] proposed Adaptive block Placement and Migration policy (APM) as well as an access pattern predictor. Using the access pattern predictor, APM places a block of data in to SRAM or STT-MRAM lines by adapting to the access pattern of each class. Compared to SRAM-based LLC, their design realized 8% and 20.5% performance improvements on average for single-threaded and multi-threaded workloads, respectively. Furthermore, their results indicate 18.9% and 19.3% reduction in-power dissipation for single-threaded and multi-threaded workloads, respectively. To extend these gains using speculative methods, Ahn et al. [8] pro-posed Prediction Hybrid Cache (PHC), which predicts the write intensity of the data and cache blocks at the time of misses and determine block placement based on the prediction. Moreover, their

dynamic predictor can adapt to the application characteristics. Furthermore, based on the predictor's output, their design places the write-intensive blocks in the SRAM region of the hybrid cache. Their result show 28% and 31% reduction in energy consumption compared to existing hybrid architectures in single-core and multi-core systems, respectively. Most recently, Khoshavi et al. [2] proposed SOS, which balances reliable and energy-efficient SA use by assigning a preferred SA to each SB to maximize energy-efficiency and reliability.

With respect to dealing with PV, Sun et al. [4] proposed Process Variation Aware Non-Uniform Cache Access (PVA-NUCA) to compensate write time variations of STT-MRAM cells due to PV. Moreover, Sun et al. [4] introduced two approaches, namely, conservative promotion and aggressive prediction. Their results offer 26.4% reduced energy consumption and provide 25.29% IPC performance improvement while incurring less than 1% area overhead. In [4], two versions of PVA-NUCA is presented: Static PVA-NUCA (SPVA-NUCA) and Dynamic PVA-NUCA (DPVA-NUCA). In PVA-NUCA, the latency for write operation to cache block is stored using 5 extra bits. In SPVA-NUCA, based on the spatial correlation of cache blocks, nearby cache blocks use the maximum latency among them to avoid erroneous operations. SPVA-NUCA does not have any data migration. On the other hand, DPVA-NUCA has two implementations: DPVA-NUCA-1 which is conservative promotion and DPVA-NUCA-2 which is aggressive prediction. In DPVA-NUCA-1 based on the frequency of write hit and miss, access pattern of the block will be determined and if the block is write-intensive it will be gradually promoted to a block with smaller write latency using swap operation between different banks. On the other hand, the non-write-intensive blocks will be gradually demoted to the blocks with larger write latency. In order to improve the performance even more, DPVA-NUCA-2 was proposed that moves the read-intensive blocks to locations with smaller read latency, moves the write-intensive blocks to locations with smaller write latency, and if the data is not read- or write-intensive, it won't be moved.

While the goals herein are similar to PVA-NUCA, our way of targeting the reliability challenges caused by PV differs from PVA-NUCA. In PVA-NUCA the main focus is on read and write latency and the data migration takes place based on these latencies. Moreover, PVA-NUCA focuses more on improving the read and write performance rather than reliability. Furthermore, PVA-NUCA uses only STT-MRAM devices for LLC. Since read operations are usually more frequent and more critical to the system performance as mentioned in [4], with respect to reliability, our focus is mainly on mitigating the effects of PV during the read operation to reduce the Bit Error Rate (BER). We have adopted the recent hybrid SRAM/STT-MRAM LLC designs and added our migration policy to them to reduce dynamic energy consumption of write operation and included our SOS approach to increase the reliability of read operations by reducing the BER. Alternatively, the work herein builds upon [2] by proposing SOS-enabled hybrid cache.

These methodologies have inspired us to maximize the efficiency of SOS by proposing a dynamic PV/Energy-Aware cache block migration policy that utilizes a mixture of SRAM and STT-MRAM banks in LLC. Even though reliable SAs offer high SMs, which results in a high ratio of error-free read operations, it is still likely that the sensed data value differs from the value stored in the memory cell. To overcome this issue, we propose to transfer vulnerable read-intensive blocks to the ways that belong to low-PV impacted ways located in other STT-MRAM banks. The non-access-intensive blocks can still remain in their STT-MRAM based ways, whether they are high-PV impacted or not. To amortize the energy consumption and long bank service time due to write operations in STT-MRAM data arrays, we propose to allocate write intensive cache blocks from ways in SRAM banks. SRAM offers both low dynamic power and high-performance features for write operation, which significantly improves the cache utilization and bank accessibility.

4.1 Hybrid SRAM and STT-MRAM LLC Design

Fig. 5 illustrates the scheme of a hybrid 8-way set associative SRAM and STT-MRAM LLC design, where way-0 and way-1 are implemented within SRAM-based banks while way-2 through way-7 are built in STT-MRAM-based banks. This configuration is selected based on our experimental results, whereby the average number of write-intensive blocks in each set was approximately 2 across all workloads. Since the peripherals required for read and write operations in NVM arrays occupy a relatively larger portion of the cache footprint than peripherals required by SRAM arrays, it is beneficial to build the tag array with SRAM cells. Thus, we assume that the entire tag array is built with SRAM. With cache tags residing in CMOS, erroneous SRAM-based tags lie outside of the scope of this study. Unlike conventional cache design approaches, where the tag and data array are accessed simultaneously to reduce access latency while incurring significant power overhead, we propose to split the cache access into two stages similar to the work presented in [26], but with adjustments in favor of high SOS throughput. If LLC is accessed with a read operation, the tag array and all STT-MRAM banks are accessed in parallel. Thus, assuming that data is found in STT-MRAM banks, the unnecessary accesses to SRAM banks can be skipped. Upon a LLC miss on STT-MRAM banks, but hit on a tag corresponding to a SRAM bank, the associated SRAM data array of the bank in LLC is accessed. Even though this mechanism incurs additional latency if the data is stored in SRAM banks, we argue that this incident occurs rarely since our insertion/migration policy maintains the read-dominant cache blocks in STT-MRAM banks while write-intensive blocks are transferred to SRAM banks. If the cache set is accessed by a write operation, the tag arrays and SRAM banks are searched in the first stage. If the data is not found in SRAM banks but found in a STT-MRAM bank, the corresponding STT-MRAM banks is accessed in the next stage. Unlike the insertion strategy in [26] where SRAM banks are selected for inserting fetched data from memory upon an LLC miss, our insertion policy allocates a way from either SRAM or

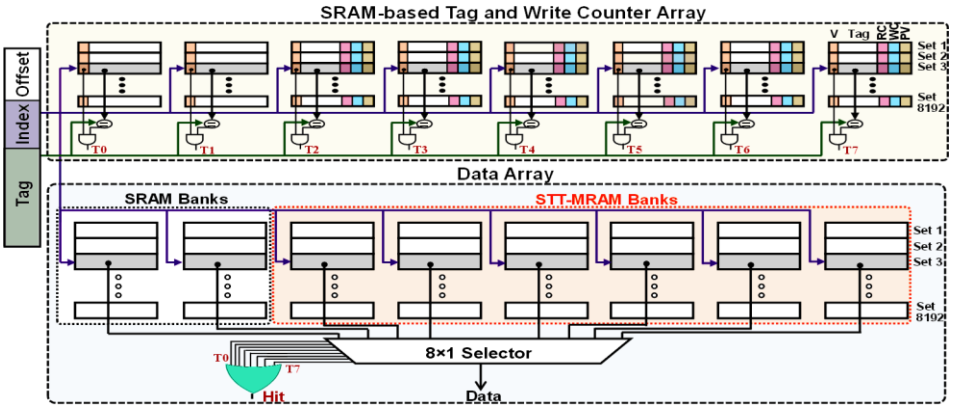


Fig. 5: The scheme of hybrid 8-way set associative SRAM and STT-MRAM cache design, whereby each bank stores a way. In the above configuration, two SRAM-based banks and six STT-MRAM based banks are illustrated.

STT-MRAM banks according to the miss type. In particular, the SRAM and STT-MRAM banks are allocated upon an LLC write miss and read miss, respectively.

Based on our observation presented in [27], a portion of a workload might be re-executed several times, indicating that the read-intensive cache blocks which were brought to LLC once, transferred to low-PV impacted region of a set, and finally evicted need to be re-allocated from low-PV impacted STT-MRAM banks while being re-referenced again. In order to keep track of read-intensive blocks, even after eviction from LLC, we utilize a read-intensive block profiler, which is basically a queue of 16 entries that maintains the address of recent frequently-read blocks. Upon a read miss in LLC, the address of missed data is searched in the profiler. If it is found, a cache block from low-PV impacted STT-MRAM ways based on Least Recently Used (LRU) policy is replaced by fetched data from memory. The dirty victim block is written back into memory while the clean victim block is silently dropped.

4.2 PV/Energy-Aware Cache Migration Policy

Besides considering hybrid SRAM and STT-MRAM designs to accelerate service to write operations and improve bank accessibility, we also propose an efficient block insertion/migration policy to maximize the SOS throughput as shown in Algorithm 2. The tag store associated with STT-MRAM banks are equipped with three fields, Read Counter (RC), Write Counter (WC), and PV status. The main idea behind using RC is to identify vulnerable read-intensive blocks in the set. If a frequently-read block is allocated to a high-PV impacted STT-MRAM array, the cache block must be relocated to a low-PV impacted region of the set to guarantee reliable read operations. We conducted an extensive exploration to evaluate the preferred value for the read threshold level, NR_{th}, within our design. We found that if NR_{th} is small, the ratio of blocks that must be transferred to a low-PV impacted region significantly increases, while if NR_{th} is large, then SOS utilization significantly decreases because only a few read-intensive cache blocks are selected for migration. Thus, we set NR_{th} based on extensive study on block access patterns of under test workloads. In addition, the non-access-intensive cache blocks located in low-PV impacted data arrays in STT-MRAM is selected to be replaced by vulnerable read-intensive blocks, if the corresponding RC of one of the high-PV impacted blocks reaches NR_{th}.

Additionally, WC is a saturating counter to keep track of write access patterns to a cache block. If WC reaches its write threshold level, NW_{th}, it is considered as a write-intensive block. We propose to transfer these blocks to SRAM data arrays in order to amortize the latency and high dynamic energy consumption associated with incoming write operations. The PV status determines whether a cache block is located in low-PV or high-PV impacted data array regions. This bit is set based on a consensus decision-making process in the tag store during the POST phase. Fig. 6 illustrates an example of migration policy for a read-intensive block located in high-PV impacted regions of STT-MRAM cache. Upon a read hit on way-2 in the STT-MRAM bank, the RC reaches its NR_{th}, indicating that it is highly possible that the incoming accesses to this block is a

Algorithm 2: Block Insertion/Migration Policy

Assumptions:
- *RC: Read Counter, WC: Write Counter, PV: Process Variation Status*
- NR_{th}: read threshold levell, NW_{th}: write threshold level
- Way 0-1 and Way 2-7 are built in SRAM and STT-MRAM (NVM), respectively in shared LLC

Function insertion() /*algorithm for inserting requested block*/
begin
1 **if** *LLC miss* **then**
2 **if** *write miss* **then**
3 eviction() /*evict *LRU block* $\in LLC_{SRAM\,Bank}$*/
4 copy *block* $\in$ *memory* into $LLC_{SRAM\,Bank}$
else if *read miss* **then**
5 **if** $\exists$ *block's address* $\in$ *read intensive block profiler* **then**
6 eviction() /*evict *LRU block* $\in$ *NVM-Bank*$_{PV=0}$*/
7 copy *block* $\in$ *memory* into NVM-Bank$_{PV=0}$
else
8 eviction() /*evict *LRU block* $\in LLC_{NVM\,Bank}$*/
9 copy *block* $\in$ *memory* into NVM Bank
if *read hit* **then**
10 **if** $\exists$ *block's address* $\in$ *read intensive block profiler* **then**
11 update LRU status
12 **else if** *read intensive block profiler is full* **then**
13 evict LRU entry and fill the profiler with the new entry's address
else if $RC_{block} < NR_{th}$ **then**
14 $++RC_{block}$
else
15 add new entry's address to read intensive block profiler
16 migration()
if *write hit* **then**
17 **if** *block* $\in LLC_{NVM\,Bank}$ & $WC_{block} < NW_{th}$ **then**
18 $++WC_{block}$
else if *block* $\in LLC_{NVM\,Bank}$ **then**
19 migration()

Function migration()/*algorithm for migrate blocks*/
begin
20 **if** *read from block* $\in Bank_{PV=1}$ **then**
21 swap (block $\in$ Bank$_{PV=1}$, block $\in$ (Bank$_{PV=0}$ & RC $< NR_{th}$))
22 **if** *write into block* $\in LLC_{NVM\,Bank}$ **then**
23 swap (block $\in$ LLC$_{NVM\,Bank}$, block $\in$ LLC$_{SRAM\,Bank}$)

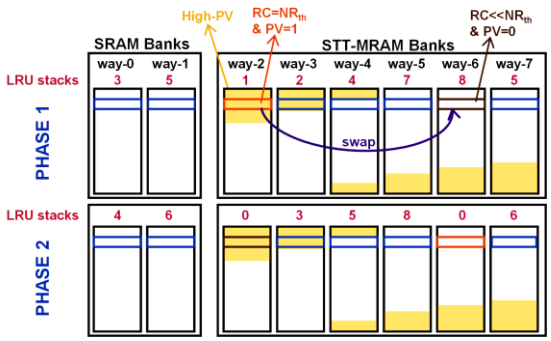


Fig. 6: The migration policy to swap a read-intensive block resided in high-PV impacted region with not access intensive block located in low-PV impacted region.

read-dominant operation. To reduce the probability of incorrectly sensing the stored value in STT-MRAM, the proposed migration policy swaps the selected read-intensive block resided in high-PV impacted region with a non-access-intensive block located in a low-PV impacted region based on LRU stacks in the tag array. A swap buffer is employed to properly enable the block transfer between low-PV impacted regions and high-PV impacted data arrays. This process is completed by updating the LRU stacks associated with each cache block after the swap operation.

5 Circuit-Level Results And Analysis

Extensive circuit-level simulation results and analysis are provided in this Section. The 22nm Predictive Technology Model (PTM) CMOS [28] is used alongside the MTJ model used in [10] to calculate the power and performance of a 1-bit MSA and ASA. In this paper, we have utilized the approach proposed in [10] to model the behavior of STT-MRAM devices, in which a Verilog-AMS model is developed using the aforementioned equations. Then, the model is leveraged in a SPICE circuit simulator to validate the functionality of the designed circuits. Table 1 lists the design parameters and PV values. All PMOS and NMOS transistors are considered minimum size except transistors used in **`INV0`** and **`INV1`**. Since **`INV0`** and **`INV1`** are vital to the reliability of the circuit, we have optimized the size of their transistors to maintain width (W) to length (L) ratio (W/L) of 4 to provide reliable functionality. All of the designs provided in this manuscript are simulated and analyzed in a case where no PV is present and in a case where PV is present. Monte Carlo (MC) simulation methods are utilized to model the PV. Table 2 lists the results for delay, power consumption, and Energy Delay Product (EDP) where no PV is present and the TMR=100% with MTJ_P=3.2 KΩ and MTJ_{ref}=5.7 KΩ. Table 3 lists similar results with MTJ_P=3.2 KΩ and MTJ_{ref}=(MTJ_P+MTJ_{AP})/2=4.8 KΩ.

In order to further investigate the effects of PV on the SAs, 10,000 MC simulations were performed on a single bit memory cell, considering different standard deviations for the CMOS threshold voltage as well as MTJ MgO thickness and surface area. During the simulation, values of V_{th}, W, and L of the CMOS transistors vary in the netlist based on a Gaussian distribution having a mean equal to the nominal model card for PTM and σVth as provided in [29]. For the MTJ variation, the model provided in [10] was used to find the effects of variation on MTJ devices. Overall, 1% to 10% variation is considered for MTJ parameters, which based on the model [10], result in 1% and 10% variation of the MTJs' TMR, respectively. Due to structural limitations of MTJ devices, the TMR ratio is considered 100% as the baseline design herein [11, 12, 30].

5.1 EDP and BER Analysis

Based on the results listed in Table 2 and Table 3, ASA-EASA provides, on average, 2-fold reduced EDP over MSA-PCSA, 7-fold reduced EDP compared to ASA-VISA, and 9-fold reduced EDP compared to MSA-SPCSA. On the other hand, ASA-VISA provides, on average, 1.4-fold reduced EDP compared to MSA-SPCSA. Fig. 7a depicts the EDP distribution of MSA-PCSA and MSA-SPCSA for sensing AP state, respectively. Fig. 7b exhibit similar results for ASA-EASA and ASA-VISA.

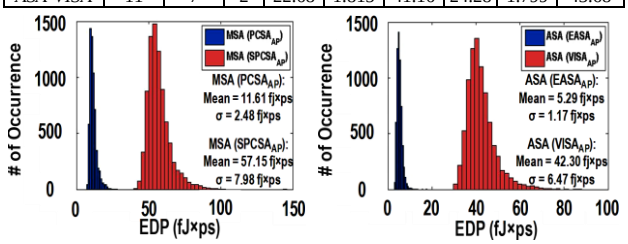
As introduced in this paper, Bit Error Rate (BER) is calculated based on the number of wrong output bits divided by all the input bits applied in both P and AP states. The values provided in Fig. 8 and Fig. 9 are the average BER values of P and AP states' sensed output obtained from simulating a single bit cell. Fig. 8a lists the 10,000 MC simulation results, where MTJ_P=3.2 KΩ, MTJ_{ref}=5.7 KΩ, and MTJ_{AP}=6.4 KΩ for TMR=100%. Considering 10% variation on TMR, the results show that on average ASA-VISA provides 8.3% reduced BER compared to ASA-EASA, 6.1% reduced BER compared to MSA-PCSA, and 1.6% reduced BER compared to MSA-SPCSA considering TMR=100%. The results also exhibit further reliability improvement considering TMR=150% where ASA-VISA provides 10.6% reduced BER compared to ASA-EASA, 7.2% reduced BER

Table 2: Simulation Results with no PV with MTJ_{Ref}=5.7KΩ

Design	*Area (Device Count)*			*Anti-Parallel (6.4 KΩ)*			*Parallel (3.2 KΩ)*		
	PMOS	*NMOS*	*MTJ*	*Delay (ps)*	*Power (μW)*	*EDP (fJ*ps)*	*Delay (ps)*	*Power (μW)*	*EDP (fJ*ps)*
MSA-PCSA	4	3	2	17.79	0.7267	12.93	16.86	0.7026	11.85
MSA-SPCSA	8	5	2	27.26	2.2960	62.59	25.44	2.2690	57.72
ASA-EASA	7	5	2	24.92	0.2445	6.09	27.24	0.2205	6.01
ASA-VISA	11	7	2	25.38	1.8560	47.11	23.29	1.7990	41.90

Table 3: Simulation Results with no PV with MTJ_{Ref}=4.8KΩ

Design	*Area (Device Count)*			*Anti-Parallel (6.4 KΩ)*			*Parallel (3.2 KΩ)*		
	PMOS	*NMOS*	*MTJ*	*Delay (ps)*	*Power (μW)*	*EDP (fJ*ps)*	*Delay (ps)*	*Power (μW)*	*EDP (fJ*ps)*
MSA-PCSA	4	3	2	15.56	0.7139	11.11	17.80	0.7097	12.63
MSA-SPCSA	8	5	2	24.72	2.271	56.14	26.51	2.277	60.36
ASA-EASA	7	5	2	22.73	0.2325	5.28	28.38	0.2274	6.45
ASA-VISA	11	7	2	22.68	1.815	41.16	24.28	1.799	43.68



(a) (b)

Fig. 7: EDP of sensing "1" with MTJ_{Ref}=5.7KΩ and TMR=100%, σTMR=10% for a) MSA in PCSA and SPCSA mode and b) ASA in EASA and VISA mode.

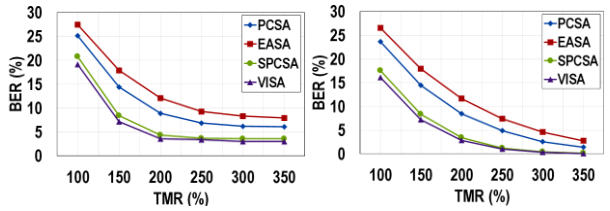


Fig. 8: BER for σTMR=10%, σV_{th}=10%, MTJ_P=3.2 KΩ, a) MTJ_{Ref}=5.7 KΩ, and b) MTJ_{Ref}=(MTJ_P+MTJ_{AP})/2.

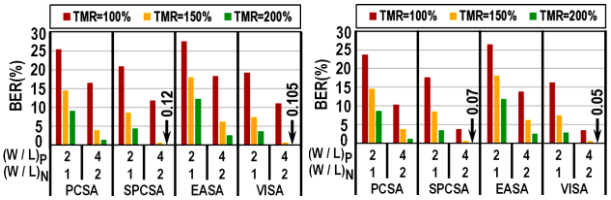


Fig. 9: Average BER for σTMR=1%&10%, σVth=10%, $(W/L)_P$=2&4, $(W/L)_N$=1&2, MTJ_P=3.2 KΩ, a) MTJ_{Ref}=5.7 KΩ, and b) MTJ_{Ref}=(MTJ_P+MTJ_{AP})/2.

compared to MSA-PCSA, and 1.2% reduced BER compared to MSA-SPCSA. Furthermore, Fig. 8b shows 10,000 MC simulation results, where MTJ_P=3.2 KΩ, MTJ_{AP}=6.4 KΩ, and MTJ_{ref}=(MTJ_P+MTJ_{AP})/2=4.8 KΩ for TMR=100%. Considering 10% variation on TMR, the results exhibit that on average ASA-VISA provides 10.3%, 5.7%, and 1.3% reduced BER compared to ASA-EASA, MSA-PCSA, and MSA-SPCSA respectively, considering TMR=100%. The results also indicate additional improvement of reliability for TMR=150% where ASA-VISA provides 10.7%, 7.2%, and 1.1% reduced BER compared to ASA-EASA, MSA-PCSA, and MSA-SPCSA respectively. Fig. 9 shows the 10,000 MC simulation results considering $(W/L)_P$ ratio of 2 and 4, and $(W/L)_N$ ratio of 1 and 2. The results show that in TMR of 100% on average SA designs with increased transistor sizes provide 8.8% and 13.2% reduced BER for MTJ_{ref}=5.7 KΩ as shown in Fig. 9a and MTJ_{ref}= (MTJ_P+MTJ_{AP})/2 as shown in Fig. 9b, respectively, compared to minimally-sized transistors. The results also exhibit further reliability improvement considering TMR of 150% where SAs having increased transistor sizes provide 9.3% and 9.4% reduced BER for MTJ_{ref}=5.7 KΩ as shown in Fig. 9a and MTJ_{ref}= (MTJ_P+MTJ_{AP})/2 as shown in Fig. 9b, respectively, compared to SAs with minimum transistor sizes. Additionally, considering TMR of 200% further improvements in reliability is observed. The BER for SAs with increased transistor sizes is reduced by 6.3% and 5.7% on average for MTJ_{ref}=5.7 KΩ as shown in Fig. 9a and MTJ_{ref}= (MTJ_P+MTJ_{AP})/2 as shown in Fig. 9b, respectively, compared to SAs with minimum transistor sizes. It can be observed that by optimizing the reference MTJ and using (MTJ_P+MTJ_{AP})/2 configuration, the BER can be decreased by 8.9% on average for a TMR of 100% due to increases in the SM for both P and AP states of the MTJ. The distribution of P and AP states of the MTJs and the reference MTJ is depicted in Fig. 10. Based on the results of MC simulations, it is clear that the larger TMR values results in an increased SM, which reduces the impact of PV.

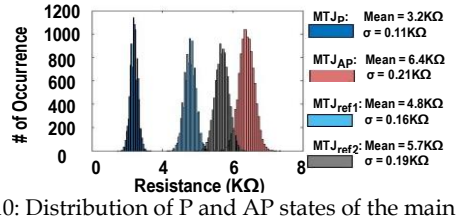


Fig. 10: Distribution of P and AP states of the main MTJ, MTJ_{ref1}=4.8KΩ, and MTJ_{ref2}=5.7KΩ.

6 Architecture-Level Results And Analysis

To comprehensively evaluate the efficacy of SOS, we analyzed SOS on both circuit- and architectural-level simulators. Architectural experimental results are presented in this Section utilizing the evaluation parameters listed in Table 1 and Table 4. The latency and energy usage associated with read and write operations for SRAM and conventional SA cache accesses are provided by NVSim [31]. However, we integrate the obtained results from Section 5 for 1-bit MSA and ASA into NVSim to extract the power and performance parameters for cache accesses in the SOS design. PARSEC 2.1 benchmarks suite is executed on a modified MARSSx86 [32], which supports asymmetric cache read and write from distinct cache banks to extract the evaluation parameters of different cache designs during program execution. We model a Chip Multi-Processor (CMP) with four single-threaded x86 cores. Each core consists of private L1 cache, and shared LLC among all the cores. Eleven workloads are executed for 500 million instructions starting at the Region of Interest (RoI) after warming up the cache for 5 million instructions. The **`simsmall`** input sets are used for all PARSEC workloads.

6.1 Energy Usage Comparison

In order to evaluate the energy benefit of SOS, we compare the energy breakdown of SOS MSA/ASA with LLC built upon SRAM, STT-MRAM, and Hybrid Cache enabled SOS (HC-SOS) with migration policy. Based on the extracted results from NVSim, which are listed in Table 4, SOS neutralizes the high energy consumption of SPCSA/VISA via low-power PCSA/EASA during read operation. The high write energy overhead for storing a value into an STT-MRAM cell incurs significant energy overhead in both SOS MSA/ASA and STT-MRAM based LLC while the SRAM-based LLC design benefits from symmetric acceptable energy consumption for both read and write operations. This incident is conspicuous for write-intensive workloads such

Table 4: Evaluation Parameters

Chip	4-Core CMP
Core	3.3GHz, Fetch / Exec / Commit width 4
L1	Private, 32KB, I/D Separate, 8-way, 64B, SRAM, WB
L2	Shared, 4MB, 8 banks, 8-way, 64B, STT-MRAM, WB
Memory	8GB, 1 channel, 4 ranks/channel, 8 banks/rank

L2 cache bank configuration (32nm, temperature=350K)

L2 Cache Technology	*RL/WL (cycles)*	*RE (nJ)*	*WE (nJ)*	*LP (mW)*	*Area (mm^2)*	*Iso-Area*
1MB SRAM	*7.43/5.78*	*0.161*	0.156	295.58	1.82	Case 1
4MB STT-MRAM	*9.08/25.58*	*0.216*	0.839	18.39	1.86	Case 1
4MB SOS with MSA	*9.08/25.58*	*PCSA=0.209 SPCSA=0.218*	0.839	18.39	2.64	Case 2
4MB SOS with ASA	*9.08/25.58*	*EASA=0.208 VISA=0.217*	0.839	18.39	2.72	Case 2

RL: Read Latency, WL: Write Latency, RE: Read Energy, WE: Write Energy, LP: Leakage Power

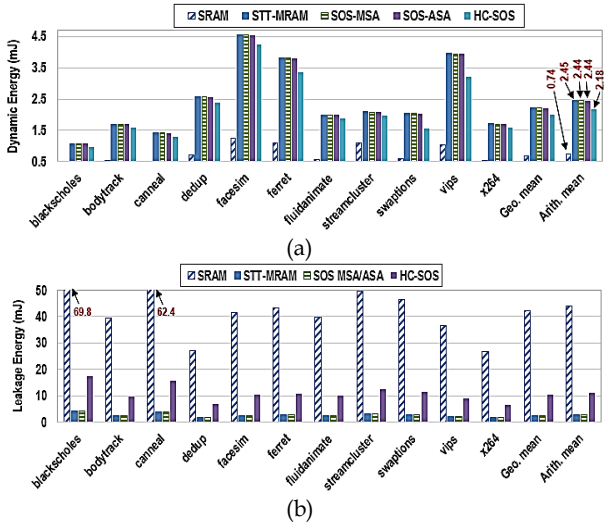


Fig. 11: (a) LLC dynamic energy comparison, and (b) LLC leakage energy comparison for SRAM, STT-MRAM, SOS-MSA, SOS-ASA, and HC-SOS, respectively.

as **facesim**, **ferret**, and **vips** where the ratio of write accesses to the LLC is significantly more than read accesses. We address this issue by proposing HC-SOS where SRAM banks are considered to accommodate write-intensive blocks while read-intensive blocks are maintained in low-PV impacted regions of STT-MRAM. The experimental results indicate that HC-SOS can save up to 10.6%, on average, of dynamic energy consumption compared to STT-MRAM-based LLC. Although SRAM exhibits lower dynamic energy consumption, its high leakage power has worsened the overall consumed energy compared to other designs, as shown in Fig. 11. Both STT-MRAM and SOS-MSA/ASA can conserve 88% on average of the total consumed energy. HC-SOS incurs higher leakage energy compared to STT-MRAM and SOS-MSA/ASA due to leveraging two SRAM-based banks in the design, incurring relatively more leakage energy to the entire cache subsystem.

6.2 Write Performance Analysis

SRAM-based LLC exhibits greater write performance compared to regular STT-MRAM, SOS, and HC-SOS. The main reason for performance degradation in STT-MRAM and SOS designs is the high write latency, while this latency has been amortized in HC-SOS. HC-SOS reduces the latency associated with write operation via allocating write-intensive cache blocks to SRAM ways for a faster write response, which results in improved performance. Additionally, HC-SOS leverages STT-MRAM to maintain read-intensive blocks for a long duration without sacrificing significant energy for preserving data. Fig. 12 shows the cumulative LLC write latency during workload execution. HC-SOS improves the write performance by 12.4%, on average, compared to STT-MRAM. The results indicate that the workloads, such as **vips**, **swaptions** and **ferret**, leverage the full potential of HC-SOS to further diminish the high write latency, which adversely impacts the entire cache sub-system throughput and accessibility.

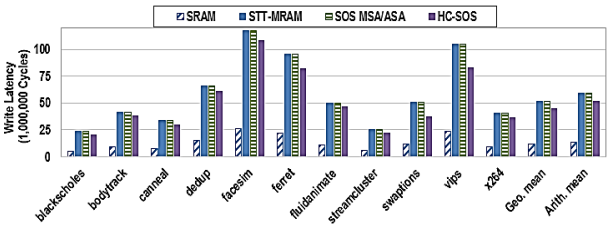


Fig. 12: Write performance comparison for SRAM, STT-MRAM, SOS MSA/ASA, and HC-SOS.

6.3 Empirical Fault Model Analysis

Fig. 13 illustrates the comparison between distributions of sensed data between LLC built by STT-MRAM, SOS-MSA, SOS-ASA, and HC-SOS-ASA. We assume that the PV map for each cache bank is similar to the floorplan of the STT-MRAM layer, shown in Fig. 4. We apply the PV ratio of each accessed SB during fault analysis for each workload. For example, if a SB experiences a high amount of PV, it is highly likely that the data will be sensed incorrectly. Our experimental results indicate that due to the impacts of PV, around one fifth of the overall sensing operations have the potential to contaminate the application's data structure. If this rate of sensed data is not accommodated properly, it may induce application crashes or prolonged program execution. Across all benchmarks suite, the calculated VFDS for some is more than others. For example, in **blackscholes** and **canneal** workloads, the proportion of read operations and dirty victim blocks residing in LLC are more than write operations, which results in the increased VFDS. As another example, the **streamcluster** workload is a read-intensive application in which more than 85% of memory operations are read accesses, which increase the chance for enduring higher VFDS. Additionally, SOS addresses the probability of sensing incorrect data through leveraging PV-resilient SA arrays in the SB architecture whenever the SB's PV ratio is more than a predefined threshold. The proposed PV-/Energy–Aware cache block migration policy further improves the SOS throughput by relocating read/write intensive blocks, which results in enhanced TDS, write performance, and

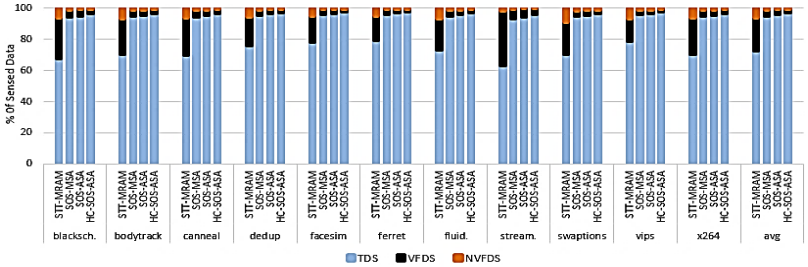


Fig. 13: Distribution of sensed data. SOS is equipped with MSA, ASA, and migration policy for ASA design.

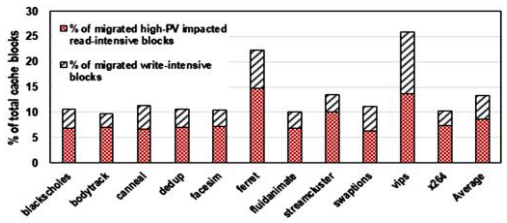


Fig. 14: The ratio of migrated cache blocks in the proposed PV/energy –aware migration policy.

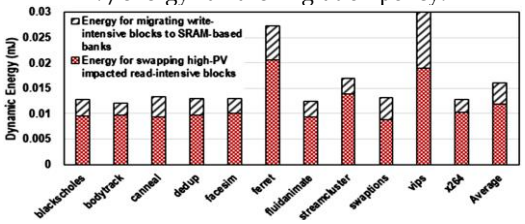


Fig. 15: The dynamic energy consumption associated with PV/energy –aware migration policy.

bank service time. Namely, the VFDS in the HC-SOS-ASA is reduced by 89% on average compared to LLC with STT-MRAM, thus improving the mean TDS from 72.5% to 97% across all workloads.

6.4 Cache Block Migration Frequency Analysis

In order to represent the frequency of cache blocks migration, we extracted the proportion of migrated read/write-intensive blocks for PARSEC benchmark suite and Fig. 14 exhibit these results. After applying PV/energy –aware migration policy, around 13% of total cache blocks are migrated to either improve the read sensing operation or reduce the energy and latency overhead associated with write operation in STT-MRAM banks. We observed that the ratio of migration depends on the behavior of workloads. For example, around 10% of cache blocks in **`streamcluster`** workload are migrated due to experiencing a high ratio of read accesses while placing in a high-PV cache block. However, less than 3.5% of cache blocks in this workload need to be migrated to SRAM-based banks due to frequent write accesses. Thus, the pattern of access and the type of memory operation play an important role in determining whether the migration will be within STT-MRAM banks or between SRAM and STT-MRAM banks. The timing and energy overhead of the proposed migration policy has been included in our experimental results. The energy consumption of PV/energy–aware migration policy is shown in Fig. 15, which demonstrates the dynamic energy consumption breakdown associated with swapping high-PV impacted read-intensive blocks within STT-MRAM-based banks and migrating write-intensive blocks to SRAM-based banks. The corresponding energy overhead for PV/energy–aware migration policy is around 14 µJ which is less than 0.7% of total LLC dynamic energy consumption. This implies that the migration energy overhead is insignificant and incurs a minor energy overhead to the entire system.

7 Conclusion

A novel approach is proposed herein to utilize SOS components for resilience and increased yield. SOS-enabled hybrid cache, utilizing SOS, provides a wide-ranging solution to leverage PV in order to improve the performance and reliability of emerging NVM technologies. Our results indicate both STT-MRAM and SOS using MSA or ASA offer up to 88% conservation of the total consumed energy, on average. ASA offers improved reliability and performance, while maintaining a small footprint of 2.5 μm^2 as depicted in Fig. 16a. Additionally, ASA incurs 0.5-fold, 10.4-fold, 2.3-fold, 3.3-fold, and 1.4-fold area overhead compared to the new MSA shown in Fig. 16b, PCSA [2], SPCSA [2], EASA [13], and VISA [13], respectively. Furthermore, our results exhibit that SOS-enabled hybrid cache improves the write performance by 12.4% on average compared to STT-MRAM design. Moreover, the VFDS is reduced by 89% on average in the SOS-enabled hybrid cache using ASA design compared to LLC with STT-MRAM. This improves the mean TDS from 72.5% to 97% across all workloads. A comparison with previous works is listed in Table 5.

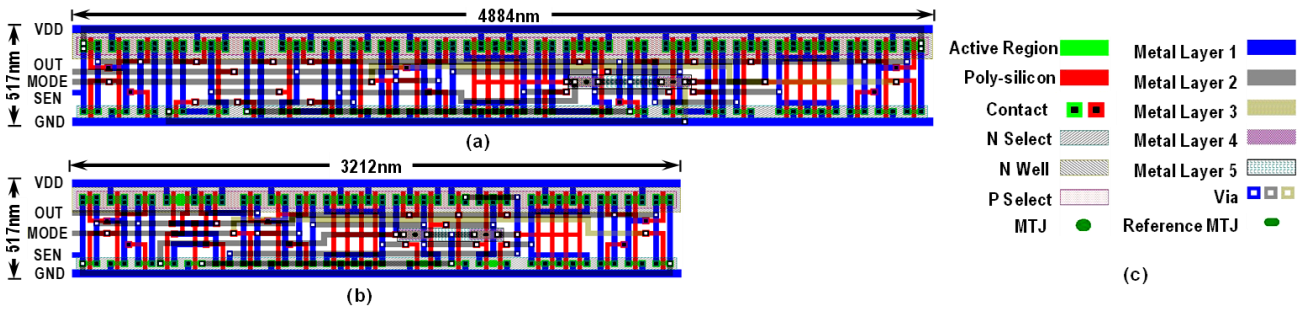


Fig. 16: a) ASA Layout, b) MSA Layout, and c) Layout Legend.

Table 5: Related Work Comparison Table

Design	Circuit-Level/ Architecture-Level	Read Enhancement		Write Enhancement		Contribution
		Reliability	Performance	Reliability	Performance	
RWHCA [25]	Architecture-Level	✗	✓	✗	✓	RWHCA reduces power dissipation by 55% on average, while achieving 5% improvement IPC compared to the baseline SRAM cache across 30 workloads.
APM [7]	Architecture-Level	✗	✗	✗	✓	Provides 18.9% and 19.3% reduction in power dissipation for single-thread and multi-thread workloads, respectively.
PHC [8]	Architecture-Level	✗	✗	✗	✓	Offers 28% and 31% reduction in energy consumption compared to existing hybrid architectures in single-core and multi-core systems, respectively.
PVA-NUCA [4]	Architecture-Level	✗	✓	✗	✓	Offers 26.4% reduced energy consumption and provide 25.29% IPC performance improvement while incurring less than 1% area overhead.
HC-SOS (This Work)	Circuit- and Architecture-Level	✓	✓	✓	✓	SOS-enabled Hybrid Cache improves write performance by 12.4% on average compared to STT-MRAM baseline cache design, improves the mean TDS from 72.5% to 97%, and reduces VFDS by 89% on average across all workloads.

REFERENCES

[1] S. Salehi, D. Fan, and R. F. Demara, "Survey of STT-MRAM Cell Design Strategies: Taxonomy and Sense Amplifier Tradeoffs for Resiliency," J. Emerg. Technol. Comput. Syst., vol. 13, pp. 1-16, 2017.

[2] N. Khoshavi, S. Salehi, and R. F. DeMara, "Variation-Immune Resistive Non-Volatile Memory using Self-Organized Sub-Bank Circuit Designs," in 18th International Symposium on Quality Electronic Design, Santa Clara, CA, USA, 2017.

[3] Y. Zhou, et al., "Asymmetric-access aware optimization for STT-RAM caches with process variations," in Proceedings of the 23rd ACM international conference on Great lakes symposium on VLSI, pp. 143-148, 2013.

[4] Z. Sun, X. Bi, and H. Li, "Process variation aware data management for STT-RAM cache design," in Proceedings of the 2012 ACM/IEEE international symposium on Low power electronics and design, pp. 179-184, 2012.

[5] S. Mittal, J. S. Vetter, and D. Li, "A survey of architectural approaches for managing embedded DRAM and non-volatile on-chip caches," IEEE Transactions on Parallel and Distributed Systems, vol. 26, pp. 1524-1537, 2015.

[6] G. Sun, et al., "A novel architecture of the 3D stacked MRAM L2 cache for CMPs," in High Performance Computer Architecture, 2009. HPCA 2009. IEEE 15th International Symposium on, pp. 239-249, 2009.

[7] Z. Wang, et al., "Adaptive placement and migration policy for an STT-RAM-based hybrid cache," in High Performance Computer Architecture (HPCA), 2014 IEEE 20th International Symposium on, pp. 13-24, 2014.

[8] J. Ahn, S. Yoo, and K. Choi, "Prediction hybrid cache: An energy-efficient STT-RAM cache architecture," IEEE Transactions on Computers, vol. 65, pp. 940-951, 2016.

[9] A. Jadidi, M. Arjomand, and H. Sarbazi-Azad, "High-endurance and performance-efficient design of hybrid cache architectures through adaptive line replacement," in Proceedings of the 17th IEEE/ACM international symposium on Low-power electronics and design, pp. 79-84, 2011.

[10] R. Zand, A. Roohi, D. Fan, and R. F. DeMara, "Energy-Efficient Nonvolatile Reconfigurable Logic using Spin Hall Effect-based Lookup Tables," IEEE Transactions on Nanotechnology, 2016.

[11] W. Zhao, C. Chappert, V. Javerliac, and J.-P. Nozière, "High speed, high stability and low power sensing amplifier for MTJ/CMOS hybrid logic circuits," IEEE Transactions on Magnetics, vol. 45, pp. 3784-3787, 2009.

[12] W. Kang, et al., "Separated Precharge Sensing Amplifier for Deep Submicrometer MTJ/CMOS Hybrid Logic Circuits," IEEE Transactions on Magnetics, vol. 50, pp. 1-5, 2014.

[13] S. Salehi and R. F. DeMara, "Process Variation Immune and Energy Aware Sense Amplifiers for Resistive Non-Volatile Memories," in 50th International Symposium on Circuits and Systems (ISCAS), Baltimore, MD, USA, 2017.

[14] A. Alzahrani and R. F. DeMara, "Process variation immunity of alternative 16nm HK/MG-based FPGA logic blocks," in Proceedings of 58th International Midwest Symposium on Circuits and Systems (MWSCAS), pp. 1-4, 2015.

[15] R. Zand, A. Roohi, S. Salehi, and R. DeMara, "Scalable Adaptive Spintronic Reconfigurable Logic using Area-Matched MTJ Design," IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 63, p. 5, July 2016.

[16] S. R. Sarangi, et al., "VARIUS: A model of process variation and resulting timing errors for microarchitects," IEEE Transactions on Semiconductor Manufacturing, vol. 21, pp. 3-13, 2008.

[17] B. D. Ripley, "The R project in statistical computing," MSOR Connections. The newsletter of the LTSN Maths, Stats & OR Network, vol. 1, pp. 23-25, 2001.

[18] P. J. Ribeiro Jr and P. J. Diggle, "geoR: a package for geostatistical analysis," R news, vol. 1, pp. 14-18, 2001.

[19] X. Liang, R. Canal, G.-Y. Wei, and D. Brooks, "Process variation tolerant 3T1D-based cache architectures," in Proceedings of the 40th Annual IEEE/ACM International Symposium on Microarchitecture, pp. 15-26, 2007.

[20] P. Friedberg, et al., "Modeling within-die spatial correlation effects for process-design co-optimization," in Quality of Electronic Design, 2005. ISQED 2005. Sixth International Symposium on, pp. 516-521, 2005.

[21] R. Teodorescu and J. Torrellas, "Variation-aware application scheduling and power management for chip multiprocessors," in Computer Architecture, 2008. ISCA'08. 35th International Symposium on, pp. 363-374, 2008.

[22] D. Cheng, et al., "A new march test for process-variation induced delay faults in srams," in Test Symposium (ATS), 2013 22nd Asian, pp. 115-122, 2013.

[23] N. Khoshavi, X. Chen, J. Wang, and R. F. DeMara, "Bit-upset vulnerability factor for edram last level cache immunity analysis," in Quality Electronic Design (ISQED), 2016 17th International Symposium on, pp. 6-11, 2016.

[24] A. K. Chintaluri, "Analysis of defects and fault models in embedded spin transfer torque (STT) MRAM arrays," Georgia Institute of Technology, 2016.

[25] X. Wu, et al., "Power and performance of read-write aware hybrid caches with non-volatile memories," in Design, Automation & Test in Europe Conference & Exhibition, 2009. DATE'09., pp. 737-742, 2009.

[26] A. Valero, et al., "Design of hybrid second-level caches," IEEE Transactions on Computers, vol. 64, pp. 1884-1897, 2015.

[27] N. Khoshavi, X. Chen, J. Wang, and R. F. DeMara, "Read-Tuned STT-RAM and eDRAM Cache Hierarchies for Throughput and Energy Enhancement," arXiv preprint arXiv:1607.08086, 2016.

[28] (2008). 22nm Predictive Technology Model (PTM). Available: http://ptm.asu.edu/modelcard/HP/22nm_HP.pm

[29] Y. Ye, F. Liu, S. Nassif, and Y. Cao, "Statistical modeling and simulation of threshold variation under dopant fluctuations and line-edge roughness," in Proceedings of 45th Design Automation Conference (DAC), pp. 900-905, 2008.

[30] E. Deng, et al., "Design optimization and analysis of multicontext STT-MTJ/CMOS logic circuits," IEEE Transactions on Nanotechnology, vol. 14, pp. 169-177, 2015.

[31] X. Dong, C. Xu, N. Jouppi, and Y. Xie, "NVSim: A circuit-level performance, energy, and area model for emerging non-volatile memory," in Emerging Memory Technologies, ed: Springer, pp. 15-50, 2014.

[32] A. Patel, F. Afram, and K. Ghose, "Marss-x86: A qemu-based microarchitectural and systems simulator for x86 multicore processors," in 1st International Qemu Users' Forum, pp. 29-30, 2011.

Soheil Salehi (S'15) is currently working towards Ph.D. degree in Computer Engineering at University of Central Florida (UCF). He received his M.S. from UCF in 2016. His research interests include Reconfigurable and Adaptive Computer Architectures, Spintronic-Based Computing Architectures, Low Power and Reliability-Aware VLSI Circuits and Systems, and Deep Submicron Technology Challenges.

Navid Khoshavi (S'09) is currently a faculty member at Florida Polytechnic Universirty. He received his Ph.D. degree in Computer Engineering at University of Central Florida (UCF) in 2017. His research interests include Online error detection and recovery in multicore processors, Hardware reliability and variability, Energy efficient and high-performance technologies, Emerging technology utilization in memory hierarchy module including emerging spintronic and eDRAM devices.

Ronald F. DeMara (S'87-M'93-SM'05) has been a full-time faculty member at the University of Central Florida since 1993. His interests are in computer architecture, reconfigurable logic, and emerging devices, on which he has published approximately 225 articles and holds one patent. He is a Senior Member of IEEE and has served on the Editorial Boards of IEEE Transactions on VLSI Systems, IEEE Transactions on Computers, and as Associate Guest Editor of ACM Transactions on Embedded Computing Systems. He has been Keynote Speaker at IEEE RAW and IEEE ReConFig conferences, and Guest Editor of IEEE Transactions on Emerging Topics in Computing joint with IEEE Transactions on Computers 2017 Special Section on Innovation in Reconfigurable Fabrics. He received the Joseph M. Bidenbach Outstanding Engineering Educator Award from IEEE in 2008.