Exploring the Vulnerability of CMPs to Soft Errors with 3D Stacked Non-Volatile Memory

Guangyu Sun
Center for Energy-Efficient
Computing and Applications
Peking University
sunguangyu@gmail.com

Eren Kursun
Jude A. Rivers
IBM Thomas J. Watson
Research Center
{ekursun, jarivers}@us.ibm.com

Yuan Xie
Computer Science
and Engineering Department
Pennsylvania State University
yuanxie@cse.psu.edu

***Abstract*—Spin-transfer Torque Random Access Memory (STT-RAM) emerges for on-chip memory in microprocessor architectures. Thanks to the magnetic field based storage STT-RAM cells have immunity to radiation induced soft errors that affect electrical charge based data storage, which is a major challenge in SRAM based caches in current microprocessors. In this study we explore the soft error resilience benefits and design trade offs of 3D-stacked STT-RAM for multi-core architectures. We use 3D stacking as an enabler for modular integration of STT-RAM caches with minimum disruption in the baseline processor design flow, while providing further interconnectivity and capacity advantages. We take an in-depth look at alternative replacement schemes in terms of performance, power, temperature, and reliability trade-offs to capture the multi-variable optimization challenges microprocessor architectures face. We analyze and compare the characteristics of STT-RAM, SRAM, and DRAM alternatives for various levels of the cache hierarchy in terms of reliability.**

I. Introduction

Due to the continuously reduced feature size, supply voltage, and increased on-chip density, modern microprocessors are projected to be more susceptible to soft error strikes [1], [2], [3]. Consequently, the majority of the on-chip memory components (such as SRAM based structures) face exacerbating challenges. As soft error rates continue to grow traditional protection techniques such as ECC show short comings, especially in multi-bit error cases. In recent years non-volatile memory technologies, such as STT-RAM, have emerged as candidates for future universal memory. The prior work on NVM mainly focuses on the density, power, and non-volatility advantages[4], [5], [6], [7], [8]. In order to explore performance advantages several approaches have been proposed to use NVMs as the replacement of DRAM for the main memory [4], [5], or as the replacement of SRAM for on-chip last-level caches (LLCs) [7]. Ipek *et al* propose the "resistive computation", which explores STT-RAM based on-chip memory and combinational logic in processors to avoid the power wall [8].

Yet the main focus has been on its density, power advantages as well as non-volatility, the advantage of NVM's immunity to soft error strikes, however, is not yet well studied at the architectural level. Since STT-RAM storage does not rely on an electrical charge, the state of its basic storage block is not altered by an emissive particle. Recent research show that the soft error rate of STT-RAM, caused by particle strikes, is several orders lower than that of SRAM. [9], [10], [11]. Sun *et al* proposed a error-resilient L1 Cache using STT-RAM [11]. The work, however, only focuses on L1 caches in a single core processor. The impact of using STT-RAM caches on the reliability of the whole cache hierarchy in a multi-core system is not studied.

In this work, we leverage the advantages of 3D integration and NVM to improve the vulnerability of CMPs to soft errors. In particular, we focus this work on inherent SER and endurance advantages of STT-RAM based caches. We explore replacing various levels of on-chip memory with 3D stacked STT-RAM to improve the soft-error vulnerability. The contributions of this work are as follows:

- We quantitatively model the vulnerability of STT-RAM to various soft errors and compare it to traditional memory technology such as SRAM.
- We utilize the low access latency through layers in 3D integration, and propose different configurations of L2/L3 caches with SRAM, eDRAM and STT-RAM. We compare these configurations, in respect of performance, power consumption, and reliability, to explore the benefits of using STT-RAM.
- We define and use a metric/method for evaluating soft error rate (SER) that evaluates vulnerability together with performance.
- We analyze the thermal characteristics of the resulting stacked configurations to indicate that the temperature profiles are within manageable ranges.

II. Preliminaries

In this section, we provide a brief overview of the STT-RAM and its immunity to soft errors.

A. STT-RAM

The basic difference between the STT-RAM and the conventional RAM technologies (such as SRAM/DRAM) is that the information carrier of STT-RAM is a Magnetic Tunnel Junction (MTJ) instead of electric charges [12]. Each MTJ contains *two ferromagnetic layers* and *one tunnel barrier layer*. One of the ferromagnetic layer (reference layer) has fixed magnetic direction while the other one (free layer) can change its magnetic direction by an external electromagnetic field or a spin-transfer torque. If the two ferromagnetic layers have different directions, the MTJ resistance is high, indicating a "1" state; if the two layers have the same direction, the MTJ resistance is low, indicating a "0" state.

The STT-RAM cell structure is composed of one NMOS transistor as the access device and one MTJ as the storage element. The MTJ is connected in series with the NMOS transistor. The NMOS transistor is controlled by the wordline (WL) signal. When a write operation happens, a large positive voltage difference is established for writing "0"s or a large negative one for writing "1"s. The current amplitude required to ensure a successful status reversal is called the threshold current. The current is related to the material of the tunnel barrier layer, the writing pulse duration, and the MTJ geometry [13].

B. Soft Errors of STT-RAM

When a particle strikes the transistor, the accumulated charge generates a pulse of current, which may cause the switching of state in tradition SRAM/DRAM. The strength and duration of the

978-1-4577-1954-7/11/$26.00 ©2011 IEEE

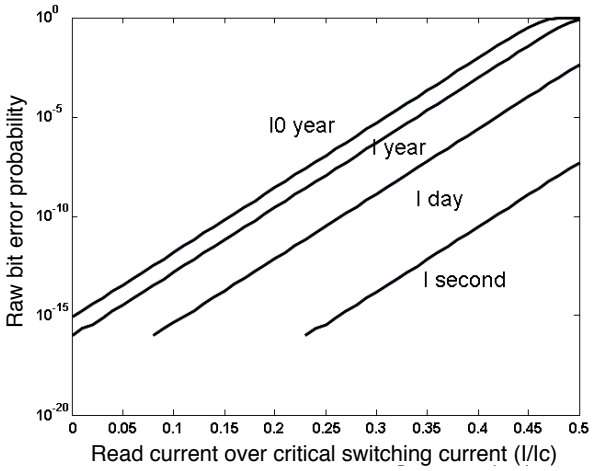

Fig. 1. The raw bit error probability of STT-RAM. Thermal stability: 75.

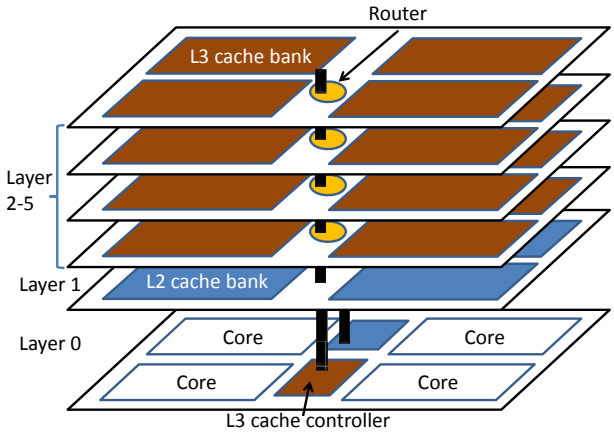

Fig. 2. An illustration of the proposed 3D NUCA structure.

pulse depend on the energy of particle. Prior research has shown the distribution of particle energy observed under different altitudes [14]. With the spice simulations, we observe that the amplitude of current caused by particle is much lower that of switching the state of a MTJ under 65nm technology. More importantly, the duration of current pulse generated by a particle strike is too short to switch a MTJ [15]. *Therefore, even if the energy of a particle strike is high enough to enable the transistor in the cell, the current cannot change the status of the MTJ*

Besides particle strikes, soft errors may also caused by thermal fluctuations in STT-RAM. Based on prior research [16], [17], we model and simulate the switching probability of STT-RAM cells (65nm technology) under thermal fluctuation, as shown in Figure 1. The error probability is explored for different simulation duration (from 1 second to 10 years) under working temperature. The thermal stability factor of STT-RAM is set to 75 in the experiments. The thermal stability factor of STT-RAM is a character related to different parameters including, transistor size, material and geometric ratio of MTJ, etc [17], which can be controlled under specified processing technology. From the results, we find that the switching probability of a STT-RAM cell under thermal fluctuation is less than 10^{-15} in a year, which is much lower than that of a SRAM/DRAM cell caused by particle strikes.

III. Architecture Modification

In this section, we first introduce the baseline configurations of our 3D CMPs. Then, we propose various replacement strategies for different levels of memory hierarchy in the CMPs.

A. Baseline Architecture

Figure 2 shows the baseline structure of this work. There are four cores located in the layer 1. The L2 cache is located in the layer 2, which is stacked above the core layer. The four cores share the same L2 cache controller. The L2 cache controller is connected to the L2 caches by way of the through-silicon-vias between layer 1 and layer 2. There are four more cache layers stacked over the L2 cache layer because the L3 cache is normally much larger than the L2 cache. The four cores also share the same L3 cache controller. The communication between multiple L3 cache layers and the cache controller is through a bus structure, which is also implemented with TSVs. There is a router located in each layer, which connects the cache bank to the bus. This bus structure has the advantage of short inter-connections provided by 3D integrations.

In this work, all replacements follows the constraint of the same area, i.e., we need to keep the similar form factor or cover the similar on-chip real estate. We estimate the area of both SRAM and STT-RAM cache banks with our extension model of CACTI [18]. We observe that the area of the STT-RAM data cell is about $1/4$ of the SRAM cell area. With the same area constraint, the capacity of STT-RAM L2 cache is increased to about 4.6 times of the SRAM one after removing the ECC code. Note that the number of tag cells are also increased as we integrate more STT-RAM cache lines.

B. Replacing L2 and L3 caches with STT-RAM

TABLE I
THREE STRATEGIES OF REPLACING SRAM CACHES WITH STT-RAM .

	L2	L2-ECC	L3	L3-ECC
(a)	STT-RAM	–	STT-RAM	STT-RAM
(b)	SRAM	SRAM	STT-RAM	STT-RAM
(c)	SRAM	STT-RAM	STT-RAM	STT-RAM

We propose three different replacement configurations for L2/L3 caches, which are listed in Table I. The details are discused next.

a) L2 STT-RAM + L3 STT-RAM: The intuitive wisdom is to replace the whole L2 and L3 SRAM caches with STT-RAM. This implies that the ECC and corresponding circuitry in both the L2 and L3 caches will be removed, adding to the potential area and capacity for more cache lines in the resulting STT-RAM caches. Similar to the previous case, both the L2 and L3 cache capacities are increased to about 4.6 times.

The first obvious advantage is that all the data cells/arrays are immune to soft error strikes. The second advantage is that the processor can have the maximum capacity of on-chip memories allowing it to achieve the lowest cache miss rates. For applications with large working set and low to moderate cache write intensities, we anticipate improved performance since the STT-RAM's limitations on write latency would be partially masked. Since the L2 cache is not the LLC, the penalty of each L2 cache miss is much reduced because of the existence of the L3 cache. The performance benefit from the lower L2 cache miss rate is reduced, compared to what we would expect in the previous configuration. Since the L1 cache is write through, for applications with intensive updates to memory, the performance could be degraded with STT-RAM L2 caches.

b) L2 SRAM + L3 STT-RAM: With the potential of performance degradation in the "L2 STT-RAM + L3 STT-RAM" configuration, we propose another configuration where we only replace the L3 SRAM cache with STT-RAM. Such strategy adapts the advantages from both SRAM and STT-RAM caches. We want to achieve a fast access speed from the L2 cache and get a low miss rate from the LLC (L3) cache. The L2 cache is write-back, hence the write intensity of the L3 cache would be much lower compared to that of the L2 cache. Therefore, the effect of long write latency can be effectively hidden in the L3 caches. Compared to the pure SRAM caches, the anticipated low LLC miss rate promises a general improvement of performance. With the L3 cache being the largest on-chip memory component, the raw FIT of the whole cache hierarchy is also greatly improved.

c) L2 SRAM + L3 STT-RAM, L2 ECC code in L3: Due to the high density of STT-RAM, the capacity of the L3 cache would be greatly increased. As we discussed earlier, 3D technology integration provides the same transparent latency for access to different layers. Therefore, we propose to implement most ECC of the L2 SRAM cache with STT-RAM and move it to the L3 cache layer to make room for enlarging the capacity of the L2 cache. A small part of ECC is kept in the L2 cache layer and implemented with SRAM. Only the ECC code of recent accessed data is stored in the SRAM part for fast access, and the rest is kept in STT-RAM. This idea is similar to that of off-chip ECC [2]. Our STT-RAM based ECC, however, induces much less overhead of performance, due to the short access latency of TSVs.

A small space in the lowest STT-RAM L3 cache layer is saved for storing the ECC of the L2 SRAM cache. We add one more set of TSVs which connect between the ECC and the L2 cache controller. Since all L2 cache lines would now be used for storing data instead of storing ECC, we expect the resulting performance to be further improved.

C. Replacing L1 Caches with STT-RAM

In modern processors, L1 caches are normally protected and monitored by parity checking codes. Such simple mechanism does not consume much area overhead, however, it can only detect soft error events but cannot correct any of them. When the L2 cache is exclusive, the data in L1 is not backuped in L1. Consequently, the L1 caches are the largest on-chip memories that may cause SEUs under particle strikes. Even if the L2 cache is inclusive, recovering data from L2 cache induces extra overhead, which can be saved by using STT-RAM L1 cache.

We propose to separate the L1 caches from the core layer and place them onto STT-RAM layers with fast access via TSVs. According to the state-of-the-art tool McPAT [19], which estimates the area of processors. In order to simplify the design, we place the L1 caches in together with L2 caches in the same layer. The L1 and L2 cache controllers can still be located in the core layer. A major objective is to keep the footprint of the processor the same. We show such a placement in Fig 3.

The access intensity to L1 caches is much higher than that to L2 caches, hence, replacing SRAM L1 caches with 3D stacked STT-RAM L1 caches has an impact on the performance of CMPs. First, the access latency may increase due to traversing on the TSVs. However, prior work has shown that the latency for traversing the TSV is trivial [20]. Second, the long write latency of STT-RAM may degrade the performance of L1 caches. Prior research has shown that a SRAM buffer can help mitigate the write overhead [11]. For the same footprint, however, the capacity of the L1 cache increases by 3x. When running applications with large working sets, the increased L1 caches can help reduce the L1 miss rates thereby improving the performance.

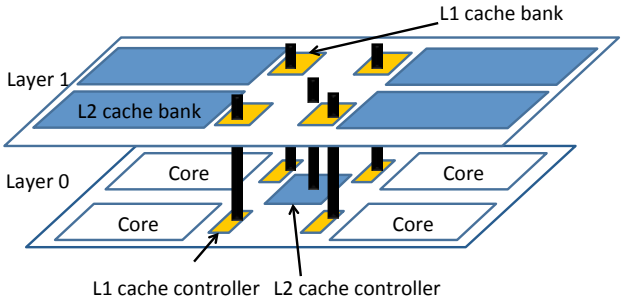

Fig. 3. Replace L1 caches with STT-RAM.

TABLE III
TECHNOLOGY PARAMETERS USED IN THE EXPERIMENTS

Parameter	Value
TSV size/pitch	$10\mu m/20\mu m$
Avg. TSVs per core	< 1024
Average core area	10 mm^2
Silicon thickness	$100\mu m$ thin Si

IV. METHODOLOGY

In this section, we present the evaluation setup. We introduce how we evaluate the vulnerability of the CMPs to soft errors. We also discuss our thermal modeling infrastructure.

A. Evaluation Setup

Our baseline configuration for our analysis is a 4-core out-of-order CMP using the Ultra SparcIII ISA. We estimate the area of the four processing cores to be about $40mm^2$, based on the study of industrial CMP examples[21][22]. We assume that one cache layer fits either the $1MB$ SRAM or a $4MB$ STT-RAM cache. The configurations are detailed in Table II and Table IV. Note that the cache access time includes the latencies of cache controllers and routers. We use the Simics toolset and its extension models GEMS [23] for performance simulations. We simulate multi-threaded benchmarks from the *OpenMP2001* and *PARSEC* [24] suites. We pin one thread on each core during the simulation. For each simulation, we fast forward to warm up the caches, and then run ROI (region of interest[24]) code in the detailed mode.

For our 3D setup, we assume a heterogeneous 3D stack in order to incorporate the STT-RAM and SRAM layers. The device layers are assumed to be thinned to 100um (with 10-15um for inter-layer interconnect and wiring layers) and integrated in a Face-to-Back (F2B) fashion. Final thickness of the 3D stack is similar to the starting 2D version due to the thinning. The TSV sizes are 10um at 20um pitch. The detailed technology parameters are listed in Table III.

B. Metric for Soft Error Vulnerability

We use *mean fault per instruction* (MFPI) as the metric for vulnerability analysis, which is defined in the following equation:

$$MFPI = \frac{number\ of\ errors\ encountered}{number\ of\ committed\ instructions} \quad (1)$$

We define *fault* to include only the errors caused by soft errors which cannot be recovered directly by the affected component. The MFPI does not only represent the vulnerability of the whole system but also shows the impact of each component and its contribution to the total soft error rates. In addition, it also exposes the delicate

TABLE II
AREA, ACCESS TIME AND ENERGY COMPARISON OF SRAM, STT-RAM, AND EDRAM CACHES INCLUDING ECC (65NM TECHNOLOGY) [15].

Cache size	Area	Read Latency	Write Latency	Read Energy	Write Energy	Standby Power
1M SRAM	36.2 mm^2	2.252 ns	2.244 ns	1.074nJ	0.956nJ	1.04W
4M STT-RAM	36.0 mm^2	2.318 ns	6.181 ns	0.858nJ	2.997nJ	0.125W
4M eDRAM	35.1 mm^2	4.053 ns	4.015ns	0.790nJ	0.788nJ	1.20W

balance between performance and soft error reliability. In this work, we trace the processing of each instruction, and calculate the time that the data of each instruction is exposed to soft error strikes. If we assume that each data bit has r soft errors in a unit time when it is exposed to particle strikes, the total number of errors that may happen in each instruction is expressed by the following equation:

$$number\ of\ errors = r \times \sum_{n} data\_size_i \times expose\_time_i \quad (2)$$

where the $data\_size_i$ represents the i_{th} data used in the instruction and its exposed time to particle strikes is $expose\_time_i$. (Note that we assume that errors in any instruction can result in an SUE).

TABLE IV
BASELINE CONFIGURATION PARAMETERS

Processors	
Number of cores = 4	Frequency = 2GHz
In-order Fetch/Decode/Retire; Out-of-Order Issue/LD/ST;	
Fetch Width = Decode Width = Issue Width = Retire Width = 4; IQ : 32 Entries, RAT & RF : 416 Entries, RUU : 32 Entries LSQ: 128 Entries, ROB : 32 Entries	
Memory Parameters	
SRAM L1 (private) STT-RAM L1 (private)	16+16KB, 2-way, 64B/cache line, 2-cycle write-through, read: 2-cycle, write: 16-cycle
SRAM L2 (shared) STT-RAM L2 (shared)	1MB, 8-way, 64B/cache line, 8-cycle write-back, read 8-cycle, write: 20-cycle
SRAM L3 (shared) STT-RAM L3 (shared)	4MB, 8-way, 64B/cache line, 18-cycle write-back, read 18-cycle, write: 30-cycle
L1 Protection	Parity codes
L2/L3 Protection	ECC, 8B/cache line
Main Memory	4GB, 300-cycle latency

V. EXPERIMENTAL RESULTS

In this section we present our experimental results on the SER vulnerability improvement, performance evaluation, power and thermal analysis of our proposed architecture configurations.

A. Performance Evaluation

The IPCs for the different configurations are compared in Figure 4(a). The benefit of replacing L2 SRAM caches with STT-RAM is reduced when there are L3 caches. This is because the penalty of an L2 cache miss is greatly mitigated by the L3 cache. In Figure 4(a), the performance increases for most benchmarks when we replace both L2 and L3 caches with STT-RAM. For four of the benchmarks (*mgrid, canneal, galgel, and equake*) the performance, however, degrades with both L2 and L3 STT-RAM caches. The benefits of reducing the L2 miss rates are offset by the overhead of long write latency to the STT-RAM L2 caches. This conclusion is further supported when we only replace L3 caches with STT-RAM. The results show that the performance increases for all benchmarks. The last set of results in the figure shows that, after we move the ECC of L2 caches to the L3 layer, we further improve the performance due to increasing capacity in the L2 caches. The fast access speed of TSVs ensures little timing overhead for accessing the ECC in the L3 layer. Since the capacity of the STT-RAM L3 cache is large, we get more benefits by placing the ECC in the L3 layers.

In Figure 5(a), we compare the performance of using SRAM, STT-RAM, and DRAM L1/L2 caches (Due to the page limit, we assume that L2 is the LLC. The cases with L3 caches show similar trend.) When we only replace the SRAM L2 caches, the configuration using STT-RAM L2 caches has the best performance for all benchmarks. The results of using DRAM L2 caches are even worse than using SRAM caches for some benchmarks because the DRAM has lower access speed for both read and write operations, as shown in Table II. In addition, the DRAM would suffer higher access latency due to the need for constant refreshes, which we do not model in this work.

The results of replacing both the L2 and L1 caches are also shown in Figure 5(a). Replacing the L1 has significant impact on the performance. For some benchmarks with large working sets, we get more performance benefits by increasing the capacities of the L1 caches with STT-RAM. For other benchmarks, the performance is degraded because the long write latency of the STT-RAM offsets such benefits. The results of using DRAM L1 caches show similar trend as that of using STT-RAM L1 caches. However, the performance of using DRAM L1 caches is worse than that of using STT-RAM L1 caches. This is because the L1 cache has very high read access intensity and the slow read speed of DRAM L1 caches introduces more overhead.

B. Soft Error Vulnerability Analysis

Figure 4(b) shows the normalized MFPI of the CMPs with different configurations of L2 and L3 caches. As we mentioned, errors recoverable in SRAM L2/L3 caches are not counted as faults when there is ECC. Hence, the vulnerabilities of our L2/L3 caches themselves are not affected by replacing SRAM with STT-RAM. However, the MFPIs of the L1 caches and the pipelines are related to the performance of L2/L3 caches. For most of the benchmarks, the replacement of both L2 and L3 caches reduces the period during which the data in L1 caches or pipelines are exposed to particle strikes. The MFPIs of the CMPs decrease for these benchmarks. On the other hand, the MFPIs of the last four benchmarks (*mgrid, canneal, galgel, and equake*) increase with STT-RAM L2/L3 caches.

If we compare Figure 4(b) and Figure 4(a), we find that the MFPI and performance are strongly correlated. When the errors of L2 and L3 caches are not counted, the MFPI of CMPs shows the opposite trend to that of IPCs. When the IPC of CMPs decrease, the period of processing data through the pipelines increases. At the same time, the period that data is exposed in the L1 caches also increases so that the MFPI increases. We can draw similar conclusions for other configurations. Consequently, when we replace the originally ECC protected caches with STT-RAM caches, we observe that the performance does not degrade while the vulnerability of the whole system improves.

The results of MFPIs for different configurations of L1 caches are shown in Figure 5(b). Different from the cases of replacing L2/L3 caches, the errors of L1 caches are counted in the MFPI because they are normally just protected by parity checking codes. Therefore, replacing the L1 caches with STT-RAM can greatly reduce the number of MFPI by eliminating errors in the L1 caches. In addition, the higher L1 cache hit ratios can help reduce the data exposure

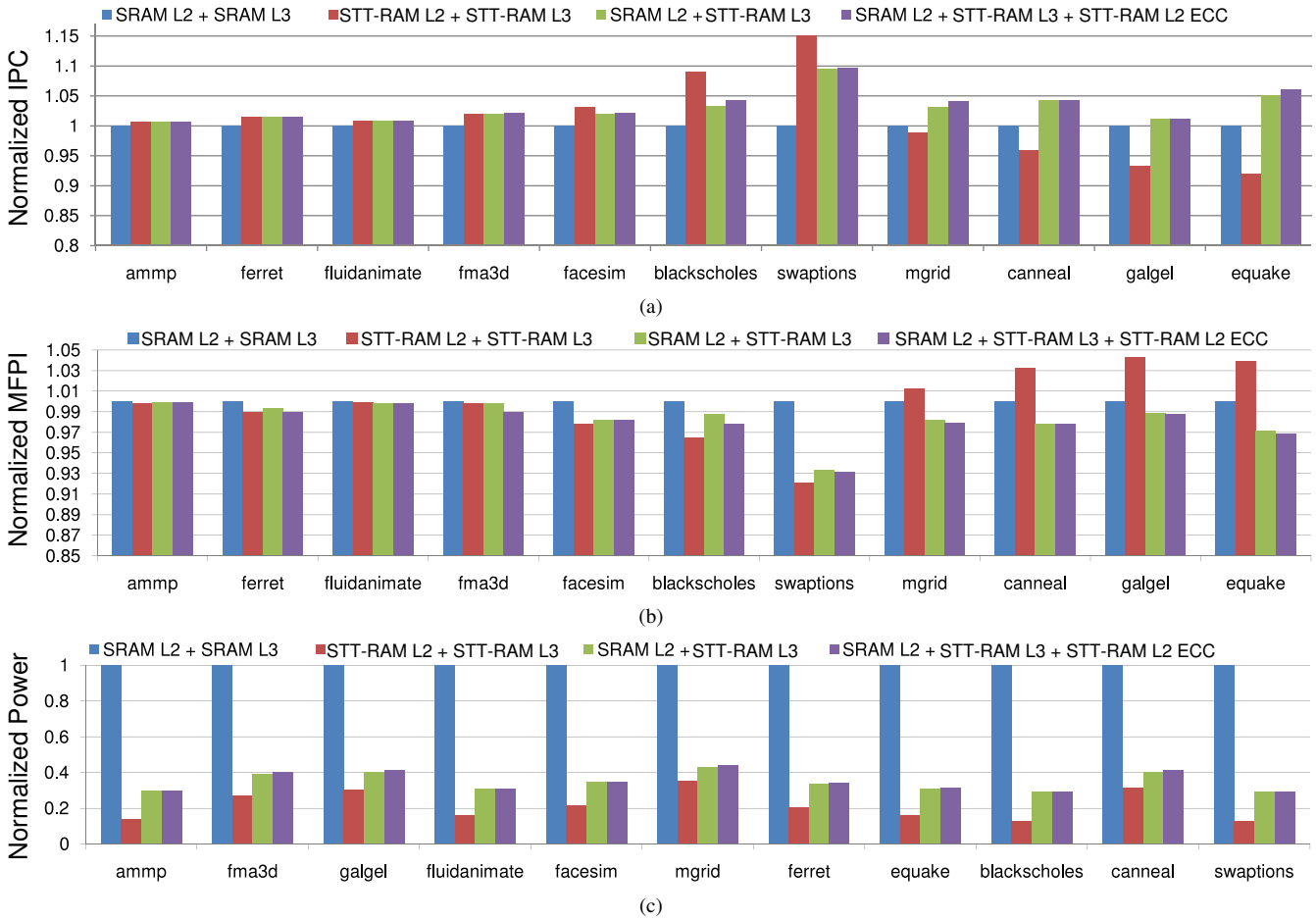

Fig. 4. Comparison between STT-RAM and SRAM for configurations of L2 and L3 caches including (a) IPCs, (b) vulnerability, and (c) power consumption.

period in the pipeline. When the performance degrades with STT-RAM L1 caches, the MFPI of the pipeline increases. The MFPI of the whole CMP, however, is still greatly reduced because the number of errors in L1 caches is much higher than that of the pipeline. The MFPIs of using DRAM caches are also shown in the Figure 5(b). The results show that using DRAM L1 caches increases the MFPI greatly for almost all the benchmarks. The data is exposed for a longer time to particle strikes in the DRAM L1 caches, which have larger capacity but is only protected by parity check code.

C. Power Consumption

As shown in Table II, the STT-RAM has the advantage of low leakage power, but the write energy is higher than those of SRAM or DRAM. In this section, we quantitatively analyze the power consumption of CMPs using STT-RAM memory technologies. The power consumption of the caches and the processing cores are compared separately in order to show the impact on different components.

When we account for the energy overhead of protection mechanisms in SRAM, replacing SRAM with STT-RAM may reduce both leakage and dynamic power. In L2 or L3 caches, the power consumption of the ECC is composed of two parts. The first part comes from the access power to the extra ECC bits; and the second part is introduced by the ECC bits' encoding and decoding. For example, there are $8Bytes$ of ECC codes for each $64Bytes$ cache line in the caches we model. We can save about 20% of power for each operation if we remove the ECC from the SRAM caches. Although the power of write operations increase when using STT-RAM, the total power consumption can still be reduced where the number of read operations dominates. For the parity check code, the power consumption overhead is mainly caused by the encoding and decoding operations. Our evaluation shows that removing parity check code can save about 5% of power consumption.

The comparison of power consumption for different cache configurations are shown in Figure 4(c) and Figure 5(c). When there are L3 caches in the processor, the leakage power of the SRAM dominates because of the large capacity of caches. As shown in Figure 4(c), after both L2 and L3 caches are replaced with STT-RAM, the power consumption is greatly reduced. This is because the leakage power of STT-RAM caches is much lower than that of SRAM caches. When only L3 caches are replaced with STT-RAM, the total power increases because more dynamic power is introduced in the SRAM L2, which is still lower than that of pure SRAM caches. When the ECC of the L2 cache is moved to the L3 cache layers, the leakage power of caches is kept the same. The total power, however, increases slightly because of the higher power consumption of updating STT-RAM ECC.

Figure 5(c) compares the power consumption of using SRAM, STT-RAM, and DRAM for L2 and L1 caches. The results show that the total power consumption decreases when SRAM L2 caches are replaced with STT-RAM caches. Since there are no L3 caches in the processor, the leakage power becomes less dominant. For some

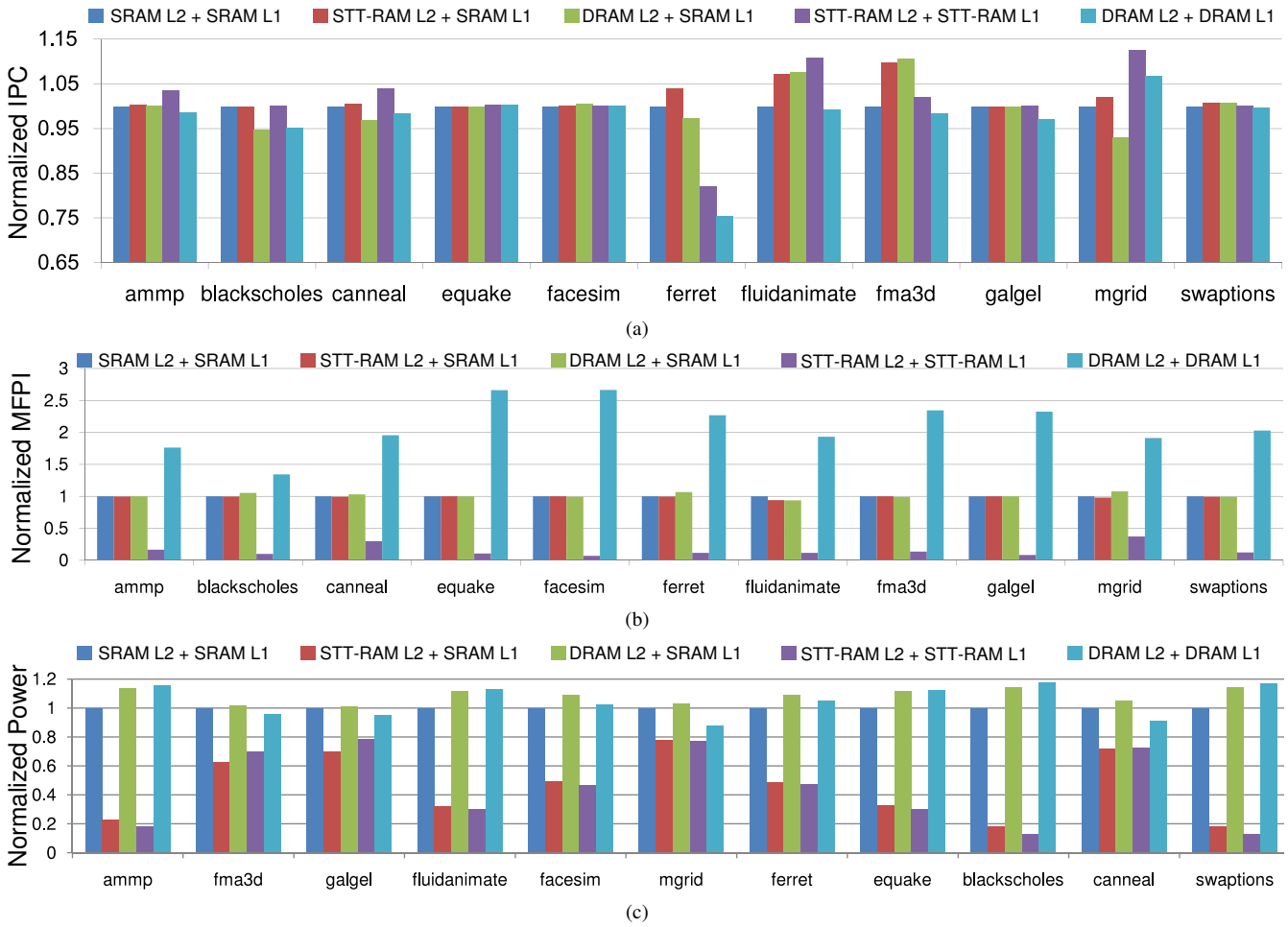

Fig. 5. Comparison STT-RAM, SRAM and DRAM for configurations of L2 and L3 caches including (a) IPCs, (b) vulnerability, and (c) power consumption.

benchmarks, when the intensity of write operations is very high, the power consumption of using SRAM and STT-RAM caches are comparable because of large energy of writing to STT-RAM. The power consumption of using DRAM L2 caches is also shown in the figure. A DRAM memory cell has lower leakage power than that of an SRAM cell. For the similar area, however, the total leakage power (including the refresh power) of a DRAM cache is higher than that of an SRAM cache. Consequently, the total power increases with using DRAM L2 caches.

We show the basic floorplan of each of the cores in our 4-core CMP in Figure 6(a). Figure 6(b) shows the detailed thermal map of the processor core layer. As the figure indicates, the hotspots center on the register file and the execution units. Peak temperatures have high-correlation with the processor layer power densities, since the power density of the stacked L2/L3 layers are lower in both SRAM and STT-RAM alternatives. Furthermore, as the relatively higher power processor layer is placed close to the heat sink and the SRAM/STT-RAM layers are placed closer to the board, the resulting peak temperatures are within manageable ranges.

The thermal model of our stack alternatives includes a detailed model of the device, wiring and inter-layer interconnect layers, full package, and a cooling solution. We used both ANSYS and Flotherm to model the different granularities of the stack (TSV/wiring components were modeled using ANSYS and the full-stack simulations were carried out in Flotherm). Ambient temperature is 25°C for

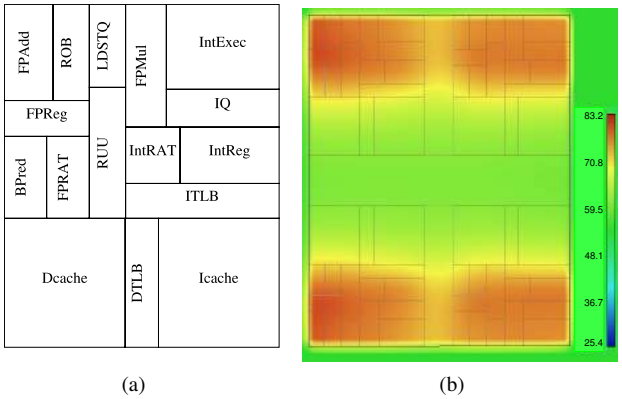

Fig. 6. (a) Basic floorplan of a core; (b) thermal map of the processor core layer.

the simulations. We explored a range of thermal conductivities for the back-end/wiring-layers and the inter-layer interconnect layers for various 3D alternatives, and reported average values of the explored range. We assumed a cooling solution based on product specifications for blade systems with similar power ranges.

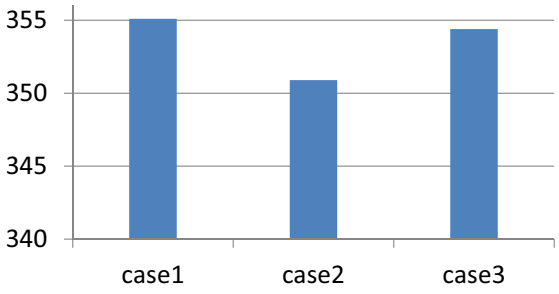

Fig. 7. Peak temperatures.

Figure 7 summarizes the peak temperatures for three stacking alternatives: (1) In the first configuration, the L1/L2/L3 hierarchy is fully implemented in SRAM; (2) An STT-RAM L3 is used to replace the SRAM L3; (3) L1/L2/L3 are all implemented in STT-RAM. As the figure shows, configuration (2) has the lower temperature than (1) (the original full SRAM stack) does, mostly due to the reduced power dissipation. Similarly, (3) has slightly lower temperature than the original SRAM stack. However, the peak temperature of (3) is higher than that of case (2). It is because replacing L1 cache with STT-RAM induces more dynamic energy

VI. Conclusion

In this work, we leveraged the emerging 3D integration and STT-RAM technologies to improve the vulnerability of CMPs to soft errors. We explored various configurations where different levels of the cache hierarchy were implemented in SRAM, STT-RAM or DRAM alternatives and evaluated these alternatives with respect to soft error reliability, performance, power, and temperature characteristics. Our experimental results show the trade-offs between performance and reliability using 3D stacked STT-RAM. For the average workload, replacing all levels of the memory hierarchy with STT-RAM virtually eliminates all soft errors on-chip, improves the performance by 14.5%, and reduces power consumption by 13.44%. The thermal characterization indicates that the resulting peak temperatures are within manageable ranges, especially with proper planning for temperatures.

Acknowledgment

This work was supported in part by NSF 0903432, 1017277, 1017391, and SRC grants.

References

[1] S. S. Mukherjee, C. Weaver, J. Emer, S. K. Reinhardt, and T. Austin, "A systematic methodology to compute the architectural vulnerability factors for a high-performance microprocessor," in *Proceedings of MICRO*, 2003, p. 29.

[2] D. H. Yoon and M. Erez, "Memory mapped ECC: low-cost error protection for last level caches," *In Proceedings of ISCA*, pp. 116–127, 2009.

[3] S. Kim, "Reducing area overhead for error-protecting large L2/L3 caches," *IEEE Trans. Comput.*, vol. 58, no. 3, pp. 300–310, 2009.

[4] P. Zhou, B. Zhao, J. Yang, and Y. Zhang, "A durable and energy efficient main memory using phase change memory technology," in *Proceedings of ISCA*, 2009, pp. 14–23.

[5] B. C. Lee, E. Ipek, O. Mutlu, and D. Burger, "Architecting phase change memory as a scalable dram alternative," in *Proceedings of ISCA*, 2009, pp. 2–13.

[6] G. Sun, X. Dong, Y. Xie, J. Li, and Y. Chen, "A novel architecture of the 3D stacked MRAM L2 cache for CMPs," in *Proceedings of HPCA*, 2009, pp. 239–249.

[7] X. Wu, J. Li, L. Zhang, E. Speight, R. Rajamony, and Y. Xie, "Hybrid cache architecture with disparate memory technologies," in *Proceedings of ISCA*, 2009, pp. 34–45.

[8] X. Guo, E. Ipek, and T. Soyata, "Resistive computation: avoiding the power wall with low-leakage, STT-MRAM based computing," in *Proceedings of ISCA*, 2010, pp. 371–382.

[9] Freescale, "Document number: Brmramslscltrl–freescale mram technology," 2007.

[10] S. Tehrani, "Status and prospect for mram technology," August 2010.

[11] H. Sun, C. Liu, W. Xu, J. Zhao, N. Zheng, and T. Zhang, "Using magnetic RAM to build low-power and soft error-resilient L1 cache," *IEEE Transactions on VLSI*, vol. PP, no. 99, p. 1, 2010.

[12] W. Zhao, E. Belhaire, Q. Mistral, C. Chappert, V. Javerliac, and *et al*, "Macro-model of Spin-Transfer Torque based Magnetic Tunnel Junction Device for Hybrid Magnetic-CMOS Design," in *IEEE International Behavioral Modeling and Simulation Workshop*, 2006, pp. 40–43.

[13] Z. Diao, Z. Li, S. Wang, Y. Ding, A. Panchula, and *et al*, "Spin-Transfer Torque Switching in Magnetic Tunnel Junctions and Spin-Transfer Torque Random Access Memory," *Journal of Physics: Condensed Matter*, vol. 19, no. 16, p. 165209 (13pp), 2007.

[14] S. Mukherjee, "Architecture design for soft errors," *published by Elsevier, Inc.*, 2008.

[15] W. J. Gallagher and S. S. P. Parkin, "Development of the magnetic tunnel junction mram at ibm: From first junctions to a 16-Mb MRAM demonstrator chip," *IBM Journal of Research and Development*, vol. 50, no. 1, pp. 5–23, 2006.

[16] X. Wang, Y. Zheng, H. Xi, and D. Dimitrov, "Thermal fluctuation effects on spin torque induced switching: Mean and variations," *JOURNAL OF APPLIED PHYSICS*, vol. 103, 034507, 2008.

[17] X. Wang, Y. Chen, H. Li, D. Dimitrov, and H. Liu, "Spin torque random access memory down to 22nm technology."

[18] N. Muralimanohar, R. Balasubramonian, and N. Jouppi, "Optimizing NUCA Organizations and Wiring Alternatives for Large Caches with CACTI 6.0," in *Proceedings of MICRO*, 2007, pp. 3–14.

[19] S. Li, R. D. S. J. H. Ahn, J. B. Brockman, and D. M. T. N. P. Jouppi, "McPAT: An Integrated Power, Area, and Timing Modeling Framework for Multicore and Manycore Architectures," in *Proceedings of MICRO*, 2009.

[20] G. L. Loi, B. Agrawal, N. Srivastava, S.-C. Lin, T. Sherwood, and K. Banerjee, "A Thermally-Aware Performance Analysis of Vertically Integrated (3-D) Processor-Memory Hierarchy," in *Proceedings of DAC*, 2006, pp. 991–996.

[21] J. A. Kahle, M. N. Day, H. P. Hofstee, C. R. Johns, T. R. Maeurer, and D. Shippy, "Introduction to the Cell Multiprocessor," *IBM Journal of Research and Development*, vol. 49, no. 4/5, pp. 589–604, 2005.

[22] P. Kongetira, K. Aingaran, and K. Olukotun, "Niagara: A 32-Way Multithreaded SPARC Processor," *IEEE Micro*, vol. 25, no. 2, pp. 21–29, 2005.

[23] M. M. K. Martin, D. J. Sorin, B. M. Beckmann, M. R. Marty, M. Xu, and *et al*, "Multifacet's general execution-driven multiprocessor simulator (gems) toolset," *SIGARCH Comput. Archit. News*, vol. 33, no. 4, pp. 92–99, 2005.

[24] C. Bienia, S. Kumar, J. P. Singh, and K. Li, "The parsec benchmark suite: Characterization and architectural implications," in *Proceedings of PACT*, October 2008.

[25] Y. Xie, G. Loh, B. Black, and K. Bernstein, "Design Space Exploration for 3D Architecture."; *ACM Journal of Emerging Technologies for Computer Systems, Vol. 2. No. 2, pp.65-103*, April 2006.

[26] Y-F. Tsai, Yuan Xie, N. Vijaykrishnan, and Mary J. Irwin, "Three-Dimensional Cache Design Exploration Using 3DCacti." in *Proceedings of ICCD, pp. 519-524*, Oct. 2005

[27] P.Emma, E. Kursun: Is 3D chip technology the next growth engine for performance improvement? *IBM Journal of Research and Development 52(6): 541-52*, 2008

[28] E. Kursun, J. Wakil, "Analysis of spatial and temporal behavior of threedimensional multi-core architectures towards run-time thermal management", *IEEE ITherm, pp.1-8*, 2010

[29] J. A. Rivers, P. Bose, P. Kudva, J.-D. Wellman, P. N. Sanda, and et al, "Phaser: phased methodology for modeling the system-level effects of soft errors," *IBM Journal of Research and Development, vol. 52, no. 3, pp. 293306*, 2008