Microprocessors and Microsystems 35 (2011) 371–381

Contents lists available at ScienceDirect

Microprocessors and Microsystems

journal homepage: www.elsevier.com/locate/micpro

Architecting high-performance energy-efficient soft error resilient cache under 3D integration technology

Hongbin Sun [a,*], Pengju Ren [a], Nanning Zheng [a], Tong Zhang [b], Tao Li [c]

[a] Institute of AIAR, Xi'an Jiaotong University, Xian, Shaanxi 710049, China
[b] ECSE Department, Rensselaer Polytechnic Institute, Troy, NY 12180, USA
[c] IEDAL Lab, University of Florida, Gainesville, FL 32611, USA

ARTICLE INFO

Article history:
Available online 17 February 2011

Keywords:
3D integration
Cache memory
Soft error resilient

ABSTRACT

Radiation-induced soft error has become an emerging reliability threat to high performance microprocessor design. As the size of on chip cache memory steadily increased for the past decades, resilient techniques against soft errors in cache are becoming increasingly important for processor reliability. However, conventional soft error resilient techniques have significantly increased the access latency and energy consumption in cache memory, thereby resulting in undesirable performance and energy efficiency degradation. The emerging 3D integration technology provides an attractive advantage, as the 3D microarchitecture exhibits heterogeneous soft error resilient characteristics due to the shielding effect of die stacking. Moreover, the 3D shielding effect can offer several inner dies that are inherently invulnerable to soft error, as they are implicitly protected by the outer dies. To exploit the invulnerability benefit, we propose a soft error resilient 3D cache architecture, in which data blocks on the soft error invulnerable dies have no protection against soft error, therefore, access to the data block on the soft error invulnerable die incurs a considerably reduced access latency and energy. Furthermore, we propose to maximize the access on the soft error invulnerable dies by dynamically moving data blocks among different dies, thereby achieving further performance and energy efficiency improvement. Simulation results show that the proposed 3D cache architecture can reduce the power consumption by up to 65% for the L1 instruction cache, 60% for the L1 data cache and 20% for the L2 cache, respectively. In general, the overall IPC performance can be improved by 5% on average.

1. Introduction

Radiation-induced soft errors has become an emerging reliability threat to high performance microprocessor design. As CMOS processing technologies continue to scale down, the soft error rate on future microprocessors are expected to grow rapidly. Cache memories constitute a significant portion of the transistor budget in current microprocessors, thereby playing a key role in processor reliability. Previous studies have concluded that unprotected memory elements are the most vulnerable system component to soft errors in current systems. Conventional error protection mechanisms for cache, such as radiation-harden technology or error detection/correction circuits, introduce undesirable draw backs in performance, power and area. Therefore, the design for an efficient soft error resilient cache memory is becoming significantly more important.

Recently, three-dimensional (3D) integration has emerged as an attractive technique for microprocessor design. The 3D technologies provide the opportunity to stack multiple active dies vertically and connect different dies through silicon vias (TSVs). Because of the high inter-die bandwidth enabled by die stacking, 3D integration has promised to bridge the processor and memory gap, and hence will have a significant impact on processor design. The potential performance benefits of 3D die stacked microarchitecture are extensively investigated [1–3]. Moreover, memory hierarchy can also take advantage of the 3D integration to improve performance and reduce energy dissipation as well. Sun et al. [4] presented a coarse-grained inter-sub-array partitioning 3D memory design that can be used to implement the overall memory hierarchy on the chip.

In addition, 3D microarchitecture provides another interesting advantage, as circuits on different dies may exhibit heterogeneous soft error vulnerabilities due to the shielding effect of die stacking. Zhang and Li [5] characterized microarchitecture soft error vulnerabilities across the 3D-stacked chip dies and concluded that the inter-dies can be shielded by the outer-dies from particle strikes. To leverage the 3D-stacked reliability benefit, Zhang and Li proposed to combine reliability harden circuits with reliability aware resource allocation to improve the reliability of 3D processor. While the idea in [5] is attractive, their investigation mainly

* Corresponding author.
E-mail address: sunsir@mail.xjtu.edu.cn (H. Sun).

0141-9331/$ - see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.micpro.2011.01.004

focuses on processor core. In this paper, we re-visit heterogeneous soft error vulnerabilities of die stacking and propose a soft error resilient cache architecture which can significantly improve the performance and energy efficiency of the overall cache hierarchy.

Due to the 3D shielding effect, the inner dies are implicitly protected by the outer dies from the particle strikes. By analyzing the simulation models presented in [5], we conclude that the most inner die in a 3D chip that have four or more dies stacked is actually invulnerable to soft error. This advantage provides an opportunity to intuitively allocate the most vulnerable component of cache memory onto the soft error invulnerable dies (SIDs) to improve reliability. Moreover, we can make a further step to eliminate the soft error protection circuits for data blocks on the SIDs without degrading reliability. As a result, access to data blocks on the SID introduces a considerably reduced access latency and energy dissipation with respect to vulnerable dies. This soft error resilience nonconformity further motivates us to leverage the dynamic scheduling strategy to maximize the data access on the SIDs, resulting in a significant performance and energy efficiency improvement.

The major contributions of this paper are followings:

- We developed a soft error resilient 3D cache architecture that follows a coarse grained inter-sub-array 3D partitioning strategy, in which the tag array is located on the SIDs while the data array is spread out among all the dies. More importantly, data blocks on the SIDs are not protected by any error correction circuits. To maximize the fast yet low-energy accesses on the SIDs as much as possible, we make the attempt to dynamically move the recently used data blocks onto SIDs. By analyzing the access mode differences between L1 and low level caches, we conclude that the scheduling strategies of dynamic data block movement for different cache level should be specified correspondingly.
- For L1 cache, we propose a SID direct mapping cache architecture to maximize the accesses on the SIDs and to avoid the energy waste on the useless data accesses. Simulation results show that the proposed cache architecture reduces the energy consumption of the L1 instruction and data cache by up to 65% and 60%, respectively. Moreover, this SID direct mapping approach in L1 cache can achieve a IPC performance improvement of up to 20%.
- For low level caches, we propose a cache architecture to decouple the tag entry from data block to compensate the relatively poor locality characteristics in low level caches. By leveraging additional forward and reverse pointers, we can substantially increase the flexibility of data block locations with reasonable overhead. The proposed tag-data decoupled cache can reduce the energy dissipation by 20% on average while maintaining the same or even better IPC performance.

The rest of this paper is organized as follows. Section 2 provides the background on soft error and conventional cache protection techniques. Section 3 briefly describes the 3D shielding effect on soft error vulnerability. Section 4 proposes the high performance and low power soft error resilient 3D cache architecture. Section 5 describes our experimental setup including the simulated processor configuration, benchmarks, modeling tools and target scenarios. Section 6 presents the evaluation results. Section 7 discusses the related work. Finally, Section 8 concludes the paper.

2. Background: soft error and conventional cache protection techniques

2.1. Soft error in computer system

As semiconductor process technology continues to scale down, soft errors are becoming an involving burden to high-performance microprocessors. Through the past decades, microprocessor industry has encountered with many soft error induced system crashes. When energetic particles, i.e. alpha and neutron particles, pass through a semiconductor device, they generate electron–hole pairs, which can be collected by transistor source and diffusion nodes. A sufficient amount of accumulated charge may invert the state of a logic device, thereby introducing a logical fault into the circuit's operation. At sea level, alpha particles are the major cause of the total transient failures [7]. In general, the primary source of alpha particle is the package materials (mold compound, underfill, solder, etc.) and not the materials used to fabricate the semiconductor device [6]. Therefore, our soft error resilient cache design focuses on the packaging materials of alpha particles.

To address the soft error vulnerability in microprocessor, a variety of techniques are proposed, from robust fabrication technologies (e.g., SOI) to special radiation-hardened circuit designs (e.g., [8]) and architectural redundancy (e.g., [9,10]). Unfortunately all of them come with undesirable penalty in performance, power and area. For instance, SOI wafer adds to the manufacturing cost of 10–15% compared with bulk silicon technology [11], and architectural redundancy increases the system overhead in terms of both area and energy consumption. As cache memory has become the dominate component in modern processor, efficient soft error resilient cache design is becoming increasingly crucial from both the reliability and the performance point of view.

2.2. Cache memory and conventional protection techniques

High performance processor is generally integrated with a two/three-level cache hierarchy. The L1 cache mainly targets for fast memory access time, while the low-level (L2/L3) cache aims at increasing the overall on-chip hit rate. As a result, L1 cache is substantially different with low-level caches in terms of capacity, set-associativity and access method.

- *L1 cache*: To reduce the access latency, L1 cache is always small, and lowly associative or even direct mapping. In addition, upon a L1 cache access, the tag and data arrays are probed simultaneously, referred to as *parallel access* [12,13]. Parallel access reduces the access latency at the cost of extra energy waste. As shown in Fig. 1, all of the N way data are read out with tag access, whereas one way at most will be matched. Reading useless data will inevitably cause unexpected energy dissipation.
- *L2/L3 cache*: Low-level cache is often designed as the relatively large and highly associative cache, to increase the overall hit rate. Upon a L2/L3 cache access, the cache waits until the tag array determines the matching way, and then accesses only the matching way of the data array, referred to as *sequential access* [14]. Sequential access cache tends to dissipate much less energy than parallel access cache, yet induces a longer access latency.

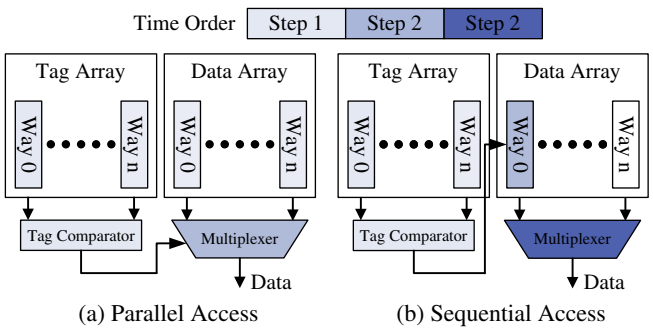

Fig. 1. Access mode options in cache memory.

Error detection codes (EDC) and error correction codes (ECC) are the most prevalent soft error protection techniques used in modern processors [15]. ECC protected cache memory is able to tolerate one or more errors in each data block, hence will be much more robust against soft error. However, previous research has shown that, implementing ECC protection circuits in caches already increases the cache access time by up to 95% [17] and power consumption by up to 22% [18]. Taking into consideration that the upcoming 3D integration may substantially reduce the latency and energy dissipation of interconnect, the cost of error protection circuits in cache will become increasingly unacceptable.

3. The soft error shielding effect provided by 3D die stacking technology

3D vertical die stacking not only brings performance and power consumption advantages, but also provides shielding effect to soft errors: the outer-dies may block energetic particles from striking the inner-dies. This 3D shielding effect was first modeled and characterized in [5]. Here, we briefly summarize our simulation results and conclusions.

Fig. 2 shows a *n*-die stacked 3D microprocessor. The thickness of each die is around 27 μm with 2 μm spacing between two dies [3]. Our analysis focuses on alpha particles from the package materials, since they are the primary source of alpha particle [6]. For the packaging emission source, we only consider particles emitted from the large surfaces of the package which are parallel to the chip. This is because the possibility of emitted alpha particles from surrounding packing material with directions that are vertical to the die is significantly smaller than that of those alpha particles emitted with directions that are parallel to the dies. We assume that all emitted particles strike the active device surface from a vertical direction with a zero angle in order to estimate the maximal distance they can penetrate into the chip. At the ground level, alpha particles in packaging material typically possess a kinetic energy between 4 MeV and 10 MeV [19]. Using the analytical models developed in [5], the alpha particles have the capability of passing through a distance between 20 μm and 70 μm into a chip. Alpha particles emitted from the top package need to pass through the bulk silicon, which is several hundred micrometers in thickness before striking transistors on the most inner die (die *n*). Alpha particles emitted from the bottom package can be completely stopped by the three outer-dies (e.g. dies 1–3), assuming the thickness of each die is around 29 μm. The above analysis indicates that the inner-dies (die 4–*n*) is inherently invulnerable to soft errors due to 3D shielding effect.

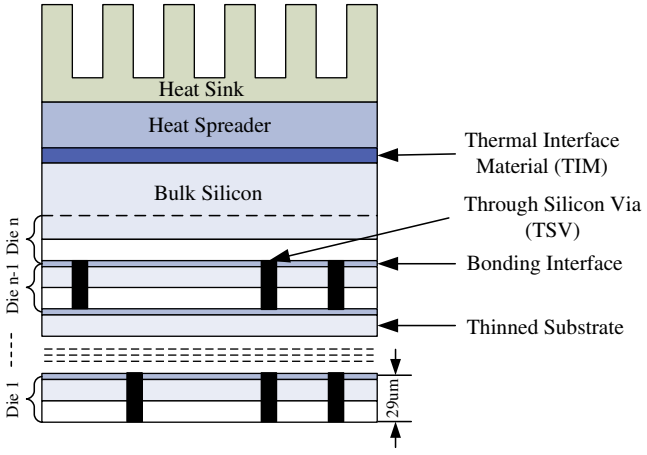

Fig. 2. A cross-section view of the *n*-die stacked 3D processor.

The shielding effect of 3D technology opens a wide range of opportunities for cost-effective soft error reliability optimization. For example, we may allocate all types of vulnerable components in microprocessor onto the inner die to improve the reliability. Alternatively, since the shielding effect provides protection for inner dies, reliability-hardening techniques only need to be selectively deployed at the vulnerable layers. In [5], Zhang and Li proposed 3D shielding aware processor core microarchitecture design. Their work, however, largely ignores cache hierarchy, which accounts for a dominant fraction of on-chip transistor real estate. In this paper, we propose novel architectures that exploit 3D soft error shielding effect to mitigate the ECC-induced performance and power overhead in cache memory.

4. The proposed soft error resilient cache architecture

In this section, we propose our soft error resilient cache architecture (SERCA) that can significantly improve the performance and energy efficiency by exploiting the die stacking shielding effect we described in Section 3.

4.1. The proposed 3D cache architecture

Due to the 3D shielding effect, the inner dies that are protected by more than three outer dies are invulnerable to soft errors. This intuitively enables us to improve the soft error reliability by reallocating the resources of cache memory in vertical dimension. Moreover, we can further eliminate the soft error protection circuits on the SIDs to achieve additional performance and energy efficiency improvement. Fig. 3 shows the proposed soft error resilient 3D cache architecture, which is distinguished from all the prior 3D cache work in the essence that (1) tag array is allocated on the soft error invulnerable dies while data array is spread out among all the dies with the coarse-grained 3D partitioning strategy [4] and (2) data array on the soft error invulnerable dies is not protected by any error correction/detection circuits while data array on the soft error vulnerable dies is applied with conventional soft error protection approaches. Note that we apply the same 3D cache architecture to both L1 instruction/data cache and low level caches, where L1 instruction cache is protected by error detection code while L1 data and L2 caches are protected by error correction code.

In contrast to data array which can be easily protected by error detection/correction techniques, tag array is difficult to adopt the sophisticated ECC techniques, whether it is the SRAM-based or CAM-based. As a result, tag array is actually the most vulnerable component in planar cache design. 3D shielding effect offers a great opportunity to address the involving reliability concern of tag array. As shown in Fig. 3, tag array in the proposed 3D cache

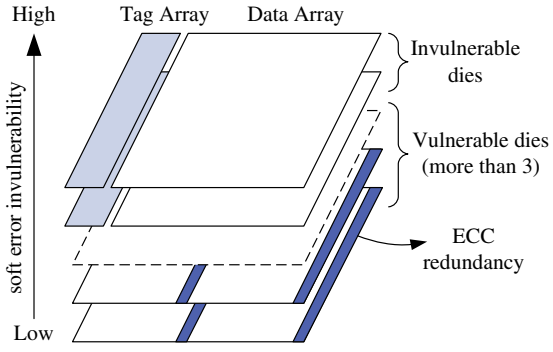

Fig. 3. The proposed soft error resilient 3D cache architecture.

is only allocated on the soft error invulnerable dies. As the SIDs can be shielded by outer-dies and is inherently invulnerable against soft error, the proposed tag allocation scheme can substantially improve the soft error resilience for tag array.

Data array in the proposed cache architecture is implemented by applying the coarse-grained inter-sub-array 3D partitioning [4] strategy, in which only the address and data I/O wires of each memory sub-array associate through silicon vias (TSVs) and the entire memory sub-array including cell array and peripheral circuits remain exactly the same as in the 2D design. During a cache access, only one 2D sub-array in the 3D sub-array set is activated and handles the read/write of all data bits at a time. The coarse-grained inter-sub-array 3D partitioning strategy is originally proposed to relax the TSV fabrication constraints in 3D cache memory, as it requires much less amount of TSVs with respect to other 3D cache partitioning strategies. It however enables us to apply different soft error protection schemes or even no protection on different dies in this work. Different soft error protection schemes among different dies in turn result in different access latency and energy consumption upon a cache access.

Similar to the conventional cache memory, each tag entry in the proposed 3D cache is coupled with a data block on the specific die. Hence, upon a tag hits, cache data can always be acquired at the location of coupled data block. The underlying difference is that data blocks on the SIDs have no soft error protection while data blocks on the soft error vulnerable dies are protected by ECC/EDC. As a result, the latency and energy consumption when accessing the data block on the SID will be considerably reduced with respect to that of the vulnerable dies. Due to this attractive advantage provided by die stacking, we can intuitively figure out that the proposed architecture is able to improve the performance and energy efficiency of the overall cache hierarchy, even if it is managed/operated the same way as conventional planar cache. Moreover, we can further exploit the efficiency of the proposed cache architecture by dynamically placing frequently accessed data on the SID dies, and less important yet still cached-data on the vulnerable dies, which is referred to as the dynamic soft error resilient cache architecture (dynamic-SERCA).

The key to the dynamic-SERCA is to place frequently-accessed data on the SIDs, which implicitly allows the data block to move or exchange among different dies. To manage the data block movement in dynamic-SERCA, we should take into consideration two important questions. (1) *vulnerability associativity*: in what associativity can the data be exchanged with data on other dies. (2) *vulnerability replacement*: under what conditions the data should be migrated from one die to another. The associativity of data movement is important to the efficiency of dynamic-SERCA. At one extreme is the static-SERCA strategies, in which a block of data can not move among dies and has to stay at the location where it is initially placed. At the other extreme, a data block could be move onto any die at any location. While the latter approach maximizes placement flexibility, the overhead of locating the data block tends to increase dramatically in terms of both area and energy consumption. Vulnerability replacement policy also has a substantial impact on the efficiency of dynamic-SERCA. Potential replacement policy could be least recently used (LRU), first in first out (FIFO) and random, each of which represents different performance vs. overhead trade-off.

As the set-associativity and access mode of L1 and low level caches are substantially different from each other, we have to exploit the appropriate *vulnerability associativity* and *replacement* policies for them respectively. In the following subsections, we make our architectural explorations assuming a four-die stacked 3D processor, where only die 4 is invulnerable to soft errors while other dies have to be protected by error correcting circuits.

4.2. SID direct-mapping in L1 cache

The intuitive vulnerability associativity and replacement policy is to use the same associativity and replacement policy as that of cache memory, where data blocks are ordered by the least recently used (LRU) and can only be exchanged within the same cache set, with SID holds the most recently used data blocks. Fig. 4 shows an example of tag-data mapping strategy, where the most recently used data blocks should be located on position A or C. However, a direct use of this intuitive policy in L1 cache may be infeasible, since the L1 cache usually chooses a small associativity that may be less than the number of the dies. On one hand, maintaining a small set will restrict the flexibility of vulnerability replacement and may impact the overall performance. On the other hand, keeping a large set will certainly waste more power by probing unnecessary data blocks and result in the unacceptable energy efficiency. More importantly, energy consumed by ECC circuits in a parallel accessed L1 cache is less critical, as the majority of power waste is caused by probing the useless data blocks. To address the above problems, we investigate the SID direct-mapping approach to dramatically reduce the energy consumption while keeping a large set.

Researchers in [20–22] have proposed several approaches to improve the energy efficiency of highly associative cache. However these approaches mitigate the impact of high-associativity by predicting either the approximate data address in the pipeline or the direct-mapping way before tag probing, almost all of which tend to considerably complicate the cache design. In this section, we propose a SID direct-mapping approach for L1 cache to leverage the soft error resilience nonconformity in 3D cache memory as much as possible. The underlying idea is to always locate the most recently used data blocks on the soft error invulnerable die and probe the tag that is coupled with the SID data block first during a cache access.

The operation flow of the proposed SID direct-mapping L1 cache is illustrated in Fig. 5. We make sure that the SID always holds the most recently used (MRU) data blocks in each set by the following two operations: (1) upon a SID miss, cache has to swap the SID data block with the currently accessed data block; (2) upon a cache miss, data block loaded from L2 cache has to be located on the SID while the previous SID data block has to take the position of evicted data block. For each L1 cache access, only the SID tag and data are probed first and the other ways are probed in case of SID misses. As over 95% of cache hits are concentrated on the MRU blocks, the proposed SID direct-mapping approach can dramatically mitigate the energy waste on the unnecessary data probe. Moreover, as the access latency and energy consumption on the SID is much less than that on the vulnerable dies, the proposed approach can achieve additional performance and energy efficiency improvement. We note that the similar selective direct-mapping approach can also be used in conventional planar

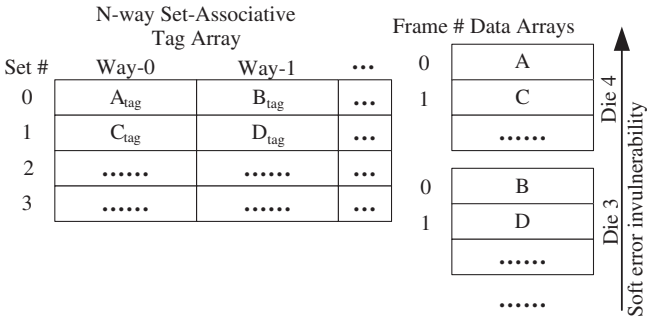

Fig. 4. The proposed tag-data mapping strategy for L1 cache.

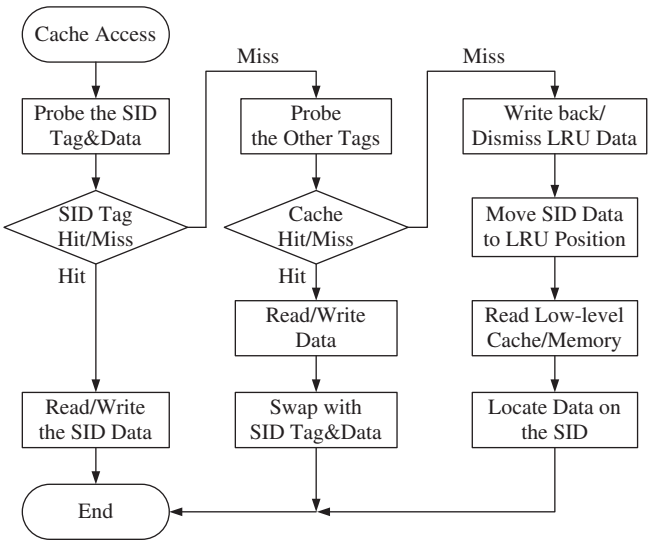

Fig. 5. Operation flow chart of the SID direct-mapping L1 cache.

cache where the access latency of all the ways in one set is equal, nevertheless it risks degrading the performance if the direct-mapping hit rate is not high enough. In the proposed 3D cache, however, access to the SID data incurs much less latency, which gives us enough confidence to take the risk. The effectiveness of SID direct-mapping approach is extensively evaluated in Section 6.

4.3. Tag-data decoupling in low level cache

It seems easy for low level cache to organize the vulnerability replacement with the same set, as it commonly maintains a relatively large set and chooses the sequential tag-data access mode as well. However, accesses to low level cache usually have relatively poor locality with respect to L1 cache. Take a 8-way associative L2 cache for example, there are two specific ways in each set may be placed on the soft error invulnerable die. If a "hot" set has more than two frequently-accessed ways, it may incur frequent data block swap between SID and other die within the same set. While other "cool" sets also hold SID data blocks even though they may not be accessed recently. The undesirable frequent data block swaps due to poor locality may considerably degrade the performance and energy efficiency of low level cache.

The sequential tag-data access in low level cache creates a new opportunity to decouple data placement from tag placement [12] to make better use of the SID. Because sequential tag-data access probes the tag array first, the exact location in the data array may be determined even if there is no implicit coupling between tag and data locations. This decoupling enables an extremely large resilience associativity, which allows a completely flexible choice of the SID for vulnerability replacement, as opposed to set associativity in conventional cache memory. Because of this flexibility, the decoupled cache can place *all* recently accessed cache blocks on the soft error invulnerable die without demoting a member of the same set to another die. This flexible vulnerability associativity is implemented by introducing a *forward pointer* and a *reverse pointer*. A forward pointer, which allows an entry in the tag array to point to an arbitrary position on an arbitrary die. One option is to store the forward pointers in each tag entry [12] that the location of data block can be obtained right after the tag probe. However this may dramatically increase the energy consumption by probing the tag, especially for the highly associative cache. In the proposed low level cache, we store the forward pointers in a small buffer named forward pointer array, as shown in Fig. 6. A reverse pointer is also necessary to determine the corresponding tag entry when accessing a data block. And we propose to store the reverse pointer with each data block, as shown in Fig. 6. By leveraging the forward and reverse pointers, we can always get the position of its data block counterpart when reading the tag entry and vice versa.

Fig. 7 shows the operation flow of the tag-data decoupled low level cache. The key point is to swap the data block between soft error invulnerable die and vulnerable dies upon a SID miss, where the data block swap should follow an appropriate vulnerability replacement policy. While using LRU as the vulnerability replacement policy is desirable for performance, its implementation may be too complex. Random replacement provides a simpler alternative but risks accidental demotion of frequently-accessed blocks. In Section 6 we show that using random replacement over LRU for vulnerability replacement has minimal impact on the performance of low-level cache.

Extra cost of additional pointers in this tag-data decoupled cache is a matter of concern. The forward and inverse pointers may be unexpected overhead for the cache memory. For example, in an 2 MB cache with 64 B blocks, 15-bit forward and reverse pointers would be required for complete flexibility. This amounts to around 64 KB pointers. Although the area overhead is only 3% with respect to the total cache size, such overhead may be undesirable in some situation. Moreover, reading the forward pointer array before reading data block tends to increase the access latency and energy dissipation, which may even offset the performance and energy benefits by removing ECC circuits. One promising solution is to place small restriction on vulnerability associativity, which can substantially reduce the pointer overhead. In our example of the 2 MB 16-way cache, if we restrict the vulnerability associativity only which includes 32 cache sets, the pointer size will be reduced to 9 bits. In Section 6, we show that the overhead of the

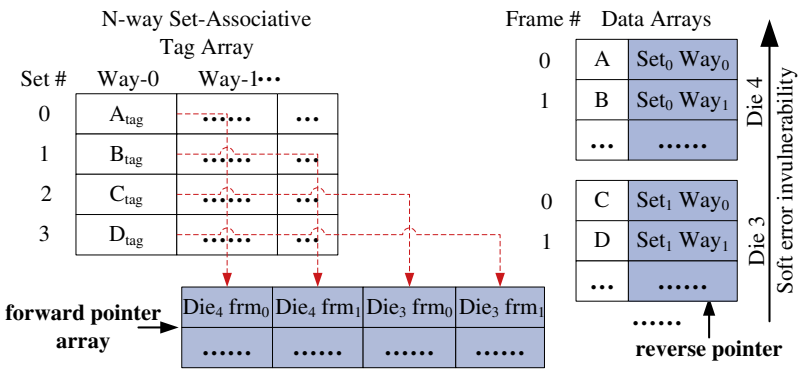

Fig. 6. Tag-data decoupled mapping strategy for low level cache.

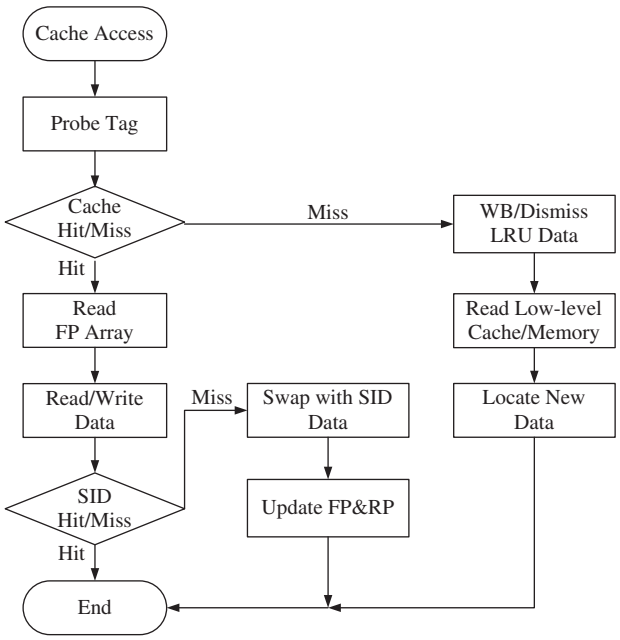

Fig. 7. Operation flow chart of the tag-data decoupled low level cache.

extra pointers is completely acceptable compared with that of ECC circuits.

5. Experimental methodology

In this section, we briefly describe our experimental methodology for evaluating the performance and energy benefits of the proposed soft error resilient 3D cache architecture. The simulated 3D processor in this paper is chosen to be 4-die stacked, and our simulations mainly focus on cache hierarchy while assuming that the processor core is implemented as a true 3D design [5] that can be seamlessly integrated with the proposed 3D cache. Table 1 lists the base configuration for the simulated system. We perform all our simulations for 65 nm technology, with a clock frequency of 3 GHz. The basic cache hierarchy configuration has a 64 KB, 4-way L1 instruction cache, a 64 KB, 4-way L1 data cache and a 2 MB, 16-way L2 unified cache, all of which have 64-byte blocks.

Table 1
Default configuration parameters used in simulated system.

Configuration parameters	Value
Processor	
Frequency	3 GHz
Functional units	4 Integer ALUs, 1 integer multiplier/divider 4 FP ALUs, 1 FP multiplier/divider
LSQ/RUU Size	8/16 Instructions
Fetch/decode/issue/commit width	4/4/4/4 Instructions/cycle
Fetch queue size	4 Instructions
Branch logic	
Predictor	Combined, bimodal 2 KB table, two-level 1KB table 8 bit history
BTB	512 Entry, 4-way
Miss-prediction penalty	3 Cycles
Cache and memory hierarchy	
L1 instruction cache	64 KB, 4-way, 64-byte blocks
L1 data cache	64 KB, 4-way, 64-byte blocks
L2 cache	2 MB unified, 16-way, 64-byte blocks
Main memory	300 Cycle latency

In particular, the access latency varies with different soft error protection schemes.

We modified Cacti 5, the latest version of a widely used cache modeling tool Cacti [23], to estimate the characteristic of cache memory with various configurations and protection schemes. The estimation assumes the 65 nm 3D integration technology and the memory is modeled as coarse-grained inter-sub-array 3D partitioning strategy. Regarding to the logic circuit, we designed the circuits to implement the parity check and SEC-DED encoding and decoding using TSMC 65 nm standard cell technology.

To evaluate the performance of the proposed 3D cache architecture, we carry out simulations using the *SimpleScalar 3.0* simulator [24]. And we use the whole integer and floating point benchmarks of the SPEC2000 suite [25] in this simulation. For each benchmark, we use *ref* inputs, fast-forward 2 billion instructions, and run for 1 billion instructions. During the fast-forward phase, we warm-up both L1 and L2 caches.

Regarding to L1 data cache, we consider the following four different scenarios:

- *Basic*: The L1 data cache is four-way set associative write-back cache, and locates data blocks in the same set onto four separate dies. Upon a cache access, the tag and data arrays are probed in parallel. To tolerate the soft error, each 64-bit data sub-block in data array is uniformly protected by SEC-DED code (8 check bits for each sub-block).
- *Static-SERCA*: The L1 data cache is designed almost the same as *Basic* scenario except that data blocks on the soft error invulnerable die (die 4 in the simulated 3D processor) are not protected by any ECC/EDC. In addition, the position of data block is determined by the initial location and the movement of data block among different dies is not allowable.
- *Dynamic-SERCA*: This scenario is the same as *static-SERCA* except that it allow data block movement among different dies to swap the most recently used data blocks onto the SID.
- *SID-DM-SERCA*: The dynamic soft error resilient cache uses the SID direct-mapping approach when accessing the tag and data arrays.

Similarly for L1 instruction cache, we also take into account the above four scenarios, i.e. basic, static-SERCA, dynamic-SERCA and SID-DM-SERCA, except that the instruction cache uses parity check code to protect the data block against soft error.

We evaluate the following four scenarios for L2 cache.

- *Basic*: The L2 cache employs the same ECC protection scheme as L1 cache. While L2 cache probes the tag and data arrays sequentially during a cache access.
- *Static and dynamic-SERCA*: The static and dynamic soft error resilient L2 cache scenarios are similar to their L1 cache counterparts, respectively.
- *TD-decoupled-SERCA*: This scenario is designed as the dynamic soft error resilient cache architecture. In particular, its tag and data blocks are decoupled to enable a greater flexibility to move most recently used data blocks onto the soft error invulnerable die.

6. Evaluation results

6.1. Cache modeling results

Table 2 shows the simulation results of cache memories estimated by our modified Cacti tool. Note that each cache memory is implemented with coarse-grained inter-sub-array 3D partitioning strategy. We focus on the dynamic energy and access delay while do not estimate the area overhead reduction by removing

Table 2
Energy consumption and delay of L1 instruction/data cache and L2 cache.

Cache configuration	Dynamic energy (nJ)	Delay (ns)	Access latency (cyc)
Basic L1I cache			
No ECC	0.0188	0.53	2
Parity	0.0194	0.92	3
SID-DM L1I cache:			
No ECC	0.0065	0.53	2
Parity	0.0071	0.92	3
Basic L1D cache:			
No ECC	0.0188	0.53	2
SEC-DED	0.0212	1.14	4
SID-DM L1D cache:			
No ECC	0.0065	0.53	2
SEC-DED	0.0089	1.14	4
Basic L2 cache:			
No ECC	0.0254	3.19	10
SEC-DED	0.0360	4.10	13
TD-Decoupled L2 cache:			
No ECC	0.0275	3.65	11
SEC-DED	0.0381	4.56	14

the ECC. The latency and dynamic energy of parity and SEC-DED codes are estimated by Synopsys tool under TSMC 65 nm technology. We assume that reading/writing L2 cache incurs an eight times larger dynamic energy with respect to that of L1 cache as L2 cache is usually accessed at the data block level. And we ignore the latency and power differences between decoding and encoding to simplify our simulation.

We can clearly figure out that cache memories with error protection increase their access latency and dynamic energy whether they are protected by parity or ECC code. For L1 cache, the proposed SID direct mapping can dramatically reduce the energy consumption although they may increase the dynamic energy upon a SID miss. The SID direct mapping may also have positive impact on the access latency, we however ignore the access latency reduction to make our simulation simple. For L2 cache, we assume the 8-bit forward and reverse pointer to estimate the overhead of the proposed tag-data decoupled approach. Modeling result shows the energy reduction of removing the ECC circuits in L2 cache more than offset the overhead due to additional forward and backward pointers, as the forward pointer array is quite small, i.e. 16 KB. However accessing the forward pointer array adds an extra cycle, which has the potential to degrade performance. Nevertheless, simulation in the Section 6 shows that its negative impact on access speed is negligible.

6.2. L1 cache simulation results

We first investigate the performance and energy efficiency of our soft error resilient architecture in both L1 instruction and data cache. Besides removing the EDC/ECC circuits on the SID, we propose to use the SID direct mapping approach to avoid the energy waste due to parallel access. To evaluate the effectiveness of SID direct mapping approach, we use SID hit rate as the comparison metric, where SID hit rate represents the percentage of cache hits in which data blocks are located on the soft error invulnerable die. The key point to the success of SID direct mapping approach is to maintain a high SID hit rate. Otherwise extra energy dissipation due to the data block swaps upon SID misses may worsen the energy efficiency. Hence we simulate the SID hit rate for both the static and dynamic scenarios. Figs. 8 and 9 show the simulated SID hit rates for L1 instruction and data cache respectively. Compared with static-SERCA, dynamic-SERCA can significantly increase the SID hit rate for both instruction and data caches. For L1 instruction cache, the average SID hit rate of dynamic-SERCA can reach up to 99.3%. The 'gcc' benchmark shows the best SID hit rate of 99.7% and the 'crafty' benchmark shows the worst case of 97.1%. The average SID hit rate for the dynamic-SERCA scenario of L1 data cache shown in Fig. 9 is 96.8%, which is a little bit lower than that of instruction cache. And the 'gcc' and 'galgel' benchmarks show the best and worst SID hit rate, which are 99.7% and 83.5% respectively. As a result, most of the L1 cache accesses will finish after the

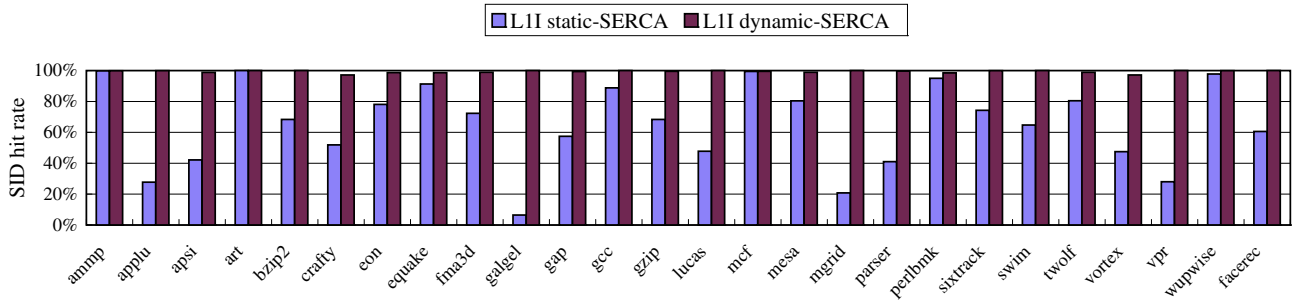

Fig. 8. The soft error invulnerable die hit rates of the static and dynamic-SERCA in L1 instruction cache.

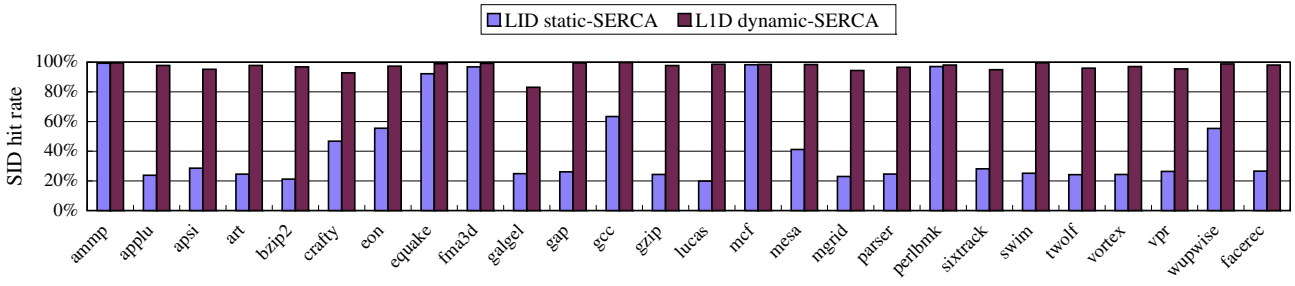

Fig. 9. The soft error invulnerable die hit rates of the static and dynamic-SERCA in L1 data cache.

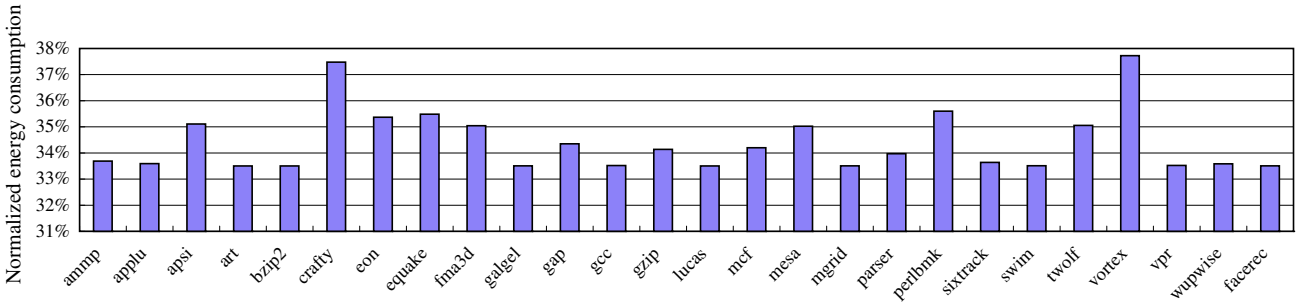

Fig. 10. Normalized power consumption of soft error resilient L1 instruction cache.

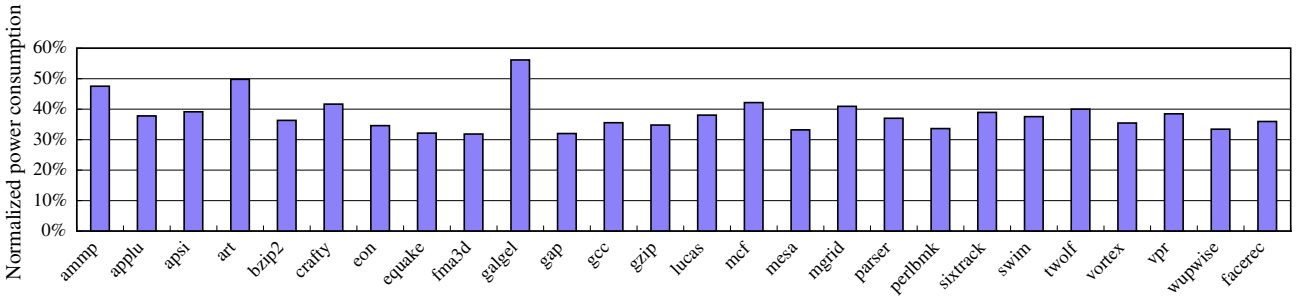

Fig. 11. Normalized power consumption of soft error resilient L1 data cache.

SID hits and read/write the data without incurring the explicit ECC decoding/encoding.

Figs. 10 and 11 show the power consumption of SID direct mapping L1 instruction and data cache respectively. The power consumption is calculated by leveraging the dynamic energy per access from *Cacti* and cache hit/miss number from *Simplescalar*, and is normalized against its basic scenario counterpart. Due to the high SID hit rate, the SID direct mapping approach dramatically reduces the energy consumption. As a SID miss costs much more energy for probing the extra tag entries and swapping the data blocks among different dies, a few benchmarks show the relatively high energy consumption. Nevertheless, for the great majority of benchmarks, energy consumption of the SID direct mapping cache can be reduced by up to 65% for instruction cache and 60% for data cache respectively, with respect to the basic 4-way set associative cache. In addition, the access latency of the data blocks on the soft error invulnerable die is much lower than that on the vulnerable dies. The increase of access speed can accordingly improve the performance, as shown in Fig. 12. For most of the simulated benchmarks, the IPC improvement can reach up to 5%. In particular, the benchmark "equake" can even achieve a IPC improvement of as high as 30%.

6.3. L2 cache simulation results

Fig. 13 shows the SID hit rates of the L2 static and dynamic-SERCA over all of the spec2000 benchmarks. The SID hit rate of L2 dynamic-SERCA is still unsatisfactory, for a few benchmarks, e.g. "ammp" and "perlbmk", and it is even worse than its static counterparts. Therefore a direct use of dynamic-SERCA in L2 cache is infeasible and may even degrade the performance. To increase the SID hit rate to a desirable level, we apply the proposed tag-data decoupling approach, where the size of forward and reverse pointer is chosen to be 8 bits. Fig. 14 shows the effectiveness of the tag-data decoupling approach, where the soft error invulnerable die hit rates of all the benchmark are higher than 95%.

To calculate the power consumption of the tag-data decoupled L2 cache, we take into account the following additional energy dissipations due to forward and reverse pointers: (1) upon a cache hit, L2 cache has to access the forward pointer array to get the location of the corresponding data block; (2) upon a SID miss and cache hit, L2 cache has to swap the accessed data block with a SID data block and update the relevant forward and reverse pointers simultaneously. Although those additional pointers cost extra energy, simulation results show that the energy save due to the increase of SID

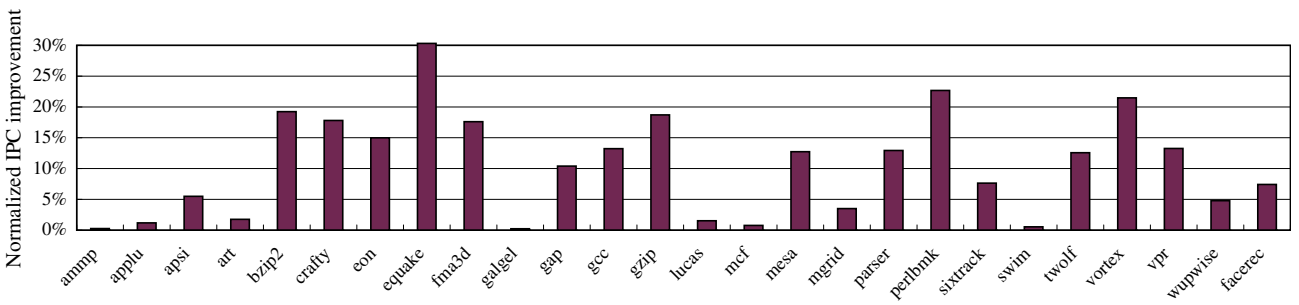

Fig. 12. Normalized IPC improvement of the proposed soft error resilient L1 cache.

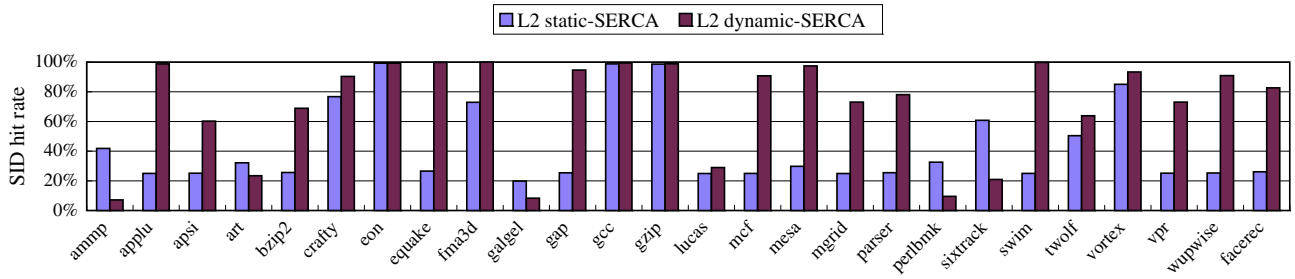

Fig. 13. The soft error invulnerable die hit rates of the L2 static and dynamic-SERCA.

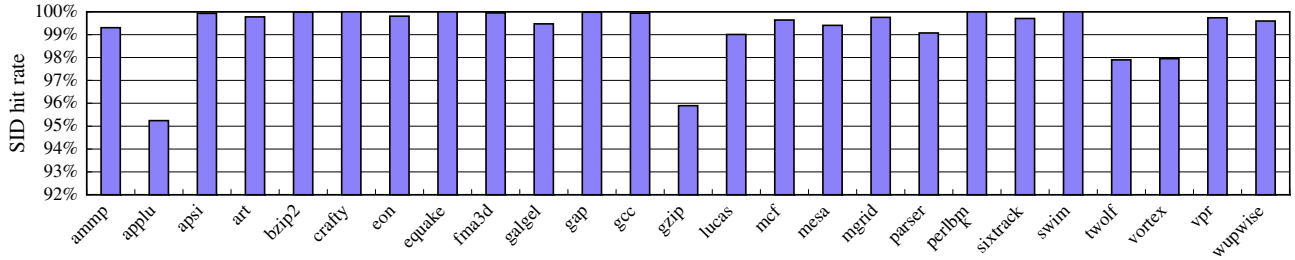

Fig. 14. The soft error invulnerable die hit rates of the L2 TD-decoupled-SERCA.

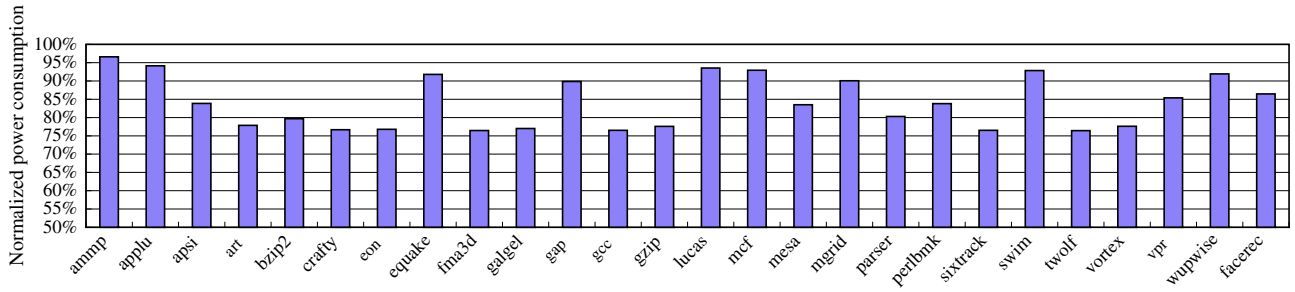

Fig. 15. Normalized power consumption of the tag-data decoupled L2 cache.

hit more than offset the extra energy cost. Fig. 15 illustrates the normalized power consumption of the proposed tag-data decoupled L2 cache with respect to the conventional cache. For the majority of benchmarks, the TD-decoupled-SERCA can reduce the power consumption by over 15%, except for a few benchmarks that have a very high L2 cache miss rate and most of the energy is dissipated on handling the cache misses. Accessing the forward pointer array incur an extra cycle upon a cache hit. On the one hand, the extra cycle on the critical path tends to degrade performance; on the other hand, the forward pointer increases the SID hit rate and in turn reduces the overall access latency. Fig. 16 shows that the tag-data decoupled L2 cache maintains the same level of or even better IPC performance.

6.4. Sensitivity analysis

We also conduct experiments to examine the sensitivity of the proposed architecture to different design practices. For L1 cache, we only simulate for different cache sizes, as L1 cache is always designed as low degree of set associativity. Table 3 shows the average

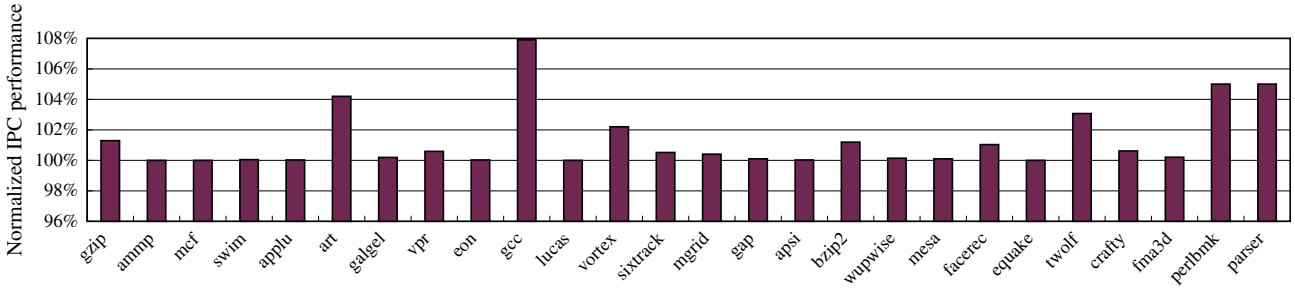

Fig. 16. Normalized IPC performance of the tag-data decoupled L2 cache.

Table 3
The L1 instruction and data cache's sensitivity to cache size.

	Instruction cache			Data cache		
	32 KB	64 KB	128 KB	32 KB	64 KB	128 KB
SID hit rate	99.2%	99.3%	99.3%	96.7%	96.8%	96.9%

Table 4
The L2 cache's sensitivity to cache size and degree of set associativity.

	Cache size			Set associativity		
	1 MB	2 MB	4 MB	8 way	16 way	32 way
SID hit rate	97.6%	99.2%	99.7%	99.2%	97.6%	96.3%

SID hit rates with the size of L1 instruction/data cache varying from 32 KB to 128 KB. As the cache size increases, the average SID hit rate stays nearly constant. This indicates that the SID direct mapping approach is insensitive to cache size variations. For L2 cache, we evaluate its sensitivity to both cache size and degree of set associativity. Table 4 shows the average SID hit rates with cache size varying from 1 MB to 4 MB and number of set associativity varying from 8-way to 32-way, respectively. As the L2 cache size increases, the SID hit rate also increases. And even for the 1 MB cache size, the proposed L2 cache can achieve a hit rate of as high as 97.6%. Similarly, the SID hit rate decreases with the increase of set associativity number, but still be able to maintain a desirable level. The above results indicate that the proposed tag-data decoupling approach is robust to different L2 cache design practices.

The simulated 3D microprocessor is four-die stacked. The die number of 3D microprocessor may also influence the efficiency of the proposed soft error resilient cache design. Increasing the stacked dies to more than 4 will enable more soft error invulnerable dies that do not need to be protected by ECC, hence can certainly achieve a better performance and energy efficiency. Therefore, we did not give the quantitative comparison in this paper.

Although thermal issue is a major concern for 3D chip design, we do not conduct detailed thermal simulation to analysis the impact of the proposed cache architecture on 3D chip temperature. This is mainly because that the hot spots in microprocessor are not located in cache memory but processor core [5]. However the proposed cache architecture will certainly outperform the prior 3D caches in terms of temperature, as the proposed cache architecture consumes a much less dynamic energy and keeps the great majority of cache accesses on the dies close to the heat sink.

7. Related work

Many recent research efforts have explored the performance benefits, thermal issues and SER characteristics of 3D integration techniques. [1–4] investigated the performance impact of placing main memory or cache on top of the processor, referred to as 3D processor-memory integration. [26,27] targeted for the thermal management in 3D microarchitecture. [5] modeled the 3D soft error shielding effect of vertically stacked dies and further exploited the architectural optimization for processor core. This work differs from prior works in that we focus on the soft error protection techniques in 3D cache memory and develop a more efficient soft error resilient 3D cache architecture by leveraging the 3D soft error shielding effect.

To improve the soft error invulnerability of cache memory, a variety of techniques have been proposed [15], among which SEC-DED code [16] is widely used in modern processors. Moreover, several studies have investigated to mitigate the power consumption and coding latency of ECC. [28] proposed architectural techniques to reduce the dynamic energy of error protection components without impacting the reliability. [29] proposed a energy efficient implementation of SEC-DED. In this paper, we propose to remove the error protection circuits on the soft error invulnerable dies to take advantage of the soft error shielding effect offered by die stacking. Moreover, by leveraging several techniques for different cache levels, we maximize the data access on the soft error invulnerable die to avoid the ECC encoding/decoding as much as possible.

8. Conclusions

With the approaching of nano-scale transistor processing technology and rapid growth of SoC integration scale, future microprocessor-based systems will be more susceptible to soft errors. Therefore, reliability is becoming an increasing design challenge and concern. With current and future processor design, cache dominates transistor real estate and its design is often subject to performance and power constraints.

A conventional method to enhance cache resilience to soft error is to apply error correction code. However, such method incurs area, performance and power overhead. This paper explores a novel soft error resilient cache architecture to mitigate the error correction circuits induced performance and energy overhead by leveraging the soft error shielding effect in 3D microarchitecture. In the proposed 3D cache architecture, tag array is only located on the soft error invulnerable dies while data array is spread out among four separate dies. In particular, data blocks on the soft error invulnerable dies have no protection against soft error. Hence the access latency and energy dissipation of data blocks on the soft error invulnerable dies can be considerably reduced. In addition, to maximize the data accesses on the invulnerable dies, we propose several approaches for different cache levels to enable a dynamic data movement strategy, including soft error invulnerable die direct mapping approach for L1 cache, and tag-data decoupling approach for low level cache. Simulation results show that the soft error resilient cache architecture can achieve a significant performance and energy efficiency improvement. The power consumption is reduced by up to 65% for L1 instruction cache, 60% for L1 data cache and 20% for the L2 cache. The overall IPC performance is also improved by 5% on average.

Acknowledgments

This research was funded in part by grants from the Nation Natural Science Foundation of China (No. 60905007), the National High Technology Research and Development Program of China (863 Program) (No. 2009AA011709) and the National Science Foundation for Post-doctoral Scientists of China (No. 20090461299).

References

[1] T. Kigl et al., PicoServer: using 3D stacking technology to enable a compact energy efficient chip multiprocessor, in: Proceedings of 12th International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS), October 2006, pp. 117–128.
[2] F. Li, C. Nicopoulos, T. Richardson, Y. Xie, Design and management of 3D chip multiprocessors using network-in-memory, in: Proceedings of the 33rd International Symposium on Computer Architecture, 2006, pp. 130–141.
[3] G.H. Loh, 3D-stacked memory architecture for multi-core processors, in: Proceedings of the 35th ACM/IEEE International Conference on Computer Architecture, June 2008, pp. 453–464.
[4] H. Sun, J. Liu, R. Anigundi, J. Lu, K. Rose, T. Zhang, 3D DRAM design and application to 3D Multicore systems, IEEE Design & Test of Computers 26 (5) (2009) 36–47.
[5] W. Zhang, T. Li, Microarchitecture soft error vulnerability characterization and mitigation under 3D integration technology, in: Proceedings of the 41th Annual International Symposium on Microarchitecture (MICRO), December 2008, pp. 453–446.

[6] R. Baumann, Soft error in advanced semiconductor devices, part I: the three radiation sources, IEEE Transaction on Device and Materials Reliability 1 (1) (2001) 17–22.
[7] T. Heijmen, P. Roche, G. Gasiot, K.R. Forbes, A comparative study on the soft-error rate of flip-flops from 90-nm production libraries, in: Proceedings of IEEE 44th Annual International Reliability Physics Symposium, 2006, pp. 204–211.
[8] T. Calin, M. Nicolaidis, R. Velazco, Upset hardened memory design for submicron cmos technology, IEEE Transaction on Nuclear Science 43 (6) (1996) 2874–2878.
[9] S.S. Mukherjee, M. Kontz, S.K. Reinhardt, Detailed design and implementation of redundant multithreading alternatives, in: Proceedings of the 29th Annual International Symposium on Computer Architecture (ISCA), May 2002, pp. 99–110.
[10] S.K. Reinhardt, S.S. Mukherjee, Transient fault detection via simultaneous multithreading, in: Proceedings of the 27th Annual International Symposium on Computer Architecture (ISCA), June 2000, pp. 25–36.
[11] IBM touts chipmaking technology. <http://www.news.com/IBM-touts-chipmaking-technology/2100-1001_3-254983.html>.
[12] Z. Chishti, M. Powell, T. Vijaykumar, Distance associativity for high-performance energy-efficient non-uniform cache architectures, in: Proceedings of 36th Annual IEEE/ACM International Symposium on Microarchitecture, December 2003, pp. 55–66.
[13] J. E et al., Internal organization of the Alpha 21164, a 300-MHz 64-bit quad-issue CMOS RISC microprocessor, Digital Technical Journal 7 (1) (1995) 119–135.
[14] D. Weiss, J. Wuu, V. Chin, The on-chip 3-MB subarray-based third-level cache on an Itanium microprocessor, IEEE Journal of Solid-State Circuits 37 (11) (2002) 1523–1529.
[15] C. Chen, M. Hsiao, Error-correcting codes for semiconductor memory applications: a state-of-the-art review, IBM Journal of Research and Development 28 (2) (1984) 124–134.
[16] J.L. Hennessy, D.A. Patterson, Computer Architecture a Quantitative Approach, fourth ed., Morgan Kaufman, 2006.
[17] J.-F. Li, Y.-J. Huang, An error detection and correction scheme for RAMs with partial-write function, in: Proceedings of IEEE International Workshop on Memory Technology, Design and Testing (MTDT), 2005, pp. 115–120.
[18] R. Phelan, Addressing soft errors in arm core-based designs, Technical report, ARM, 2003.
[19] M. Gordon, K. Rodbell, D. Heidel, C. Cabral, E. Cannon, D. Reinhardt, Single-event-upset and alpha-particle emission rate measurement techniques, IBM Journal of Research and Development 52 (3) (2008) 265–273.
[20] B. Batson, T. Vijaykumar, Reactive-associative caches, in: Proceedings of Parallel Architectures and Compilation Techniques, 2001, pp. 49–60.
[21] K. Inoue, T. Ishihara, K. Murakami, Way-predicting set-associative cache for high performance and low energy consumption, in: Proceedings of International Symposium on Low Power Electronics and Design, 1999, pp. 273–275.
[22] M. Powell, A. Agarwal, T. Vijaykumar, B. Falsafi, K. Roy, Reducing set-associative cache energy via way-prediction and selective direct-mapping, in: Proceedings of the 34th International Symposium on Microarchitecture (MICRO), 2001, pp. 54–65.
[23] CACTI: an integrated cache and memory access time, cycle time, area, leakage, and dynamic power model. <http://www.hpl.hp.com/research/cacti/>.
[24] http://www.simplescalar.com, 2008.
[25] Standard Performance Evaluation Corporation, 2000. <http://www.spec.org>.
[26] G. Loh, B. Agarwal, N. Srivastava, S. Lin, T. Sterwood, A thermally-aware performance analysis of vertically integrated (3-D) processor-memory hierarchy, in: Proceedings of the 43rd Annual Conference on Design Automation, 2006, pp. 991–996.
[27] K. Puttaswamy, G. Loh, Thermal herding: microarchitecture techniques for controlling hotspots in high-performance 3d-integrated processors, in: Proceedings of 13th International Symposium on High-Performance Computer Architecture(HPCA), 2007, pp. 193–204.
[28] L. Li, V. Degalahal, N. Vijaykrishnan, M. Kandemir, M. Inwin, Soft error and energy consumption interactions: a data cache perspective, in: Proceedings of the international symposium on Low power electronics and design, 2004, pp. 132–137.
[29] S. Ghosh, S. Basu, N. Touba, Reducing power consumption in memory ecc checkers, in: Proceedings of International Test Conference, 2004, pp. 1322–1331.

Hongbin Sun received B.S. degree in electrical engineering from Xi'an Jiaotong University, Xian, China in 2003. He is currently a Ph.D. student in the school of electronic and information engineering at Xi'an Jiaotong University. He was a visiting scholar in electrical, computer and systems engineering department at Rensselaer Polytechnic Institute, Troy, NY, from November 2007 to October 2008. His current research interests include fault tolerant computer architecture, 3D memory-processor integration and VLSI architecture for digital video processing.

Pengju Ren received B.S. degree in electrical engineering from Xi'an Jiaotong University, Xi'an, China in 2004. He is currently a Ph.D. student in the school of electronic and information engineering at Xi'an Jiaotong University. He is now a visiting scholar in Computer Science and Artificial Intelligence Laboratory (CSAIL) at Massachusetts Institute of Technology, Cambridge, US from October 2009. His current research interests include networking on multicore chip and VLSI architecture for digital video processing.

Nanning Zheng graduated from the Department of Electrical Engineering, Xi'an Jiaotong University, Xi'an, China, in 1975, and received the M.S. degree in information and control engineering from Xi'an Jiaotong University in 1981 and the Ph.D. degree in electrical engineering from Keio University, Yokohama, Japan, in 1985. He jointed Xi'an Jiaotong University in 1975, and he is currently a Professor and the Director of the Institute of Artificial Intelligence and Robotics, Xi'an Jiaotong University. His research interests include computer vision, pattern recognition, machine vision and image processing, neural networks, and hardware implementation of intelligent systems. Dr. Zheng became a member of the Chinese Academy of Engineering in 1999 and has been the chief scientist and the director of the Information Technology Committee of the China National High Technology Research and Development Program since 2001. He was General Chair of the International Symposium on Information Theory and its Applications and General Co-Chair of the International Symposium on Nonlinear Theory and Its Applications, both in 2002. He is a member of the Board of Governors of the IEEE ITS Society and the Chinese Representative on the Governing Board of the International Association for Pattern Recognition. He also serves as an executive deputy editor of the Chinese Science Bulletin. He is a fellow of IEEE.

Tong Zhang received the B.S. and M.S. degrees in electrical engineering from the Xi'an Jiaotong University, Xian, China, in 1995 and 1998, respectively. He received the Ph.D. degree in electrical engineering from the University of Minnesota, Minneapolis, in 2002. Currently he is an Associate Professor in electrical, computer and systems engineering department at Rensselaer Polytechnic Institute, Troy, NY. His current research interests include algorithm and architecture co-design for communication and data storage systems, variation-tolerant signal processing IC design, fault-tolerant system design for digital memory, and interconnect system design for hybrid CMOS/nanodevice electronic systems. Currently he serves as an Associate Editor for the IEEE Transactions on Circuits and Systems – II and the IEEE Transactions on Signal Processing.

Tao Li received the PhD degree in computer engineering from the University of Texas at Austin in 2004. He has been an assistant professor in the Department of Electrical and Computer Engineering at the University of Florida since August 2004. His research interests including high-performance, low-power and dependable processor and memory architectures, application-specific embedded systems, the impact of emerging technologies & applications on computer architecture design, and the interactions between computer architecture, operating systems, compilers and run-time systems. He is a recipient of the 2009 National Science Foundation Faculty Early CAREER Award, 2008, 2007, 2006 IBM Faculty Awards, 2008 Microsoft Research Safe and Scalable Multi-core Computing Award, 2006 Microsoft Research Trustworthy Computing Curriculum Award.