3D DRAM Design and Application to 3D Multicore Systems

Hongbin Sun
Xi'an Jiaotong University

Jibang Liu
Rensselaer Polytechnic

Rakesh S. Anigundi
Qualcomm

Nanning Zheng
Xi'an Jiaotong University

Jian-Qiang Lu, Kenneth Rose, and Tong Zhang
Rensselaer Polytechnic

Editor's note:
From a system architecture perspective, 3D technology can satisfy the high memory bandwidth demands that future multicore/manycore architectures require. This article presents a 3D DRAM architecture design and the potential for using 3D DRAM stacking for both L2 cache and main memory in 3D multicore architecture.

—Yuan Xie, Pennsylvania State University

■ **A VIABLE AND PROMISING** option to address the well-known memory wall problem in high-performance computing systems is 3D integration. Through multiple high-capacity DRAM dies stacked with one or more processor dies and through massive interdie interconnect bandwidth, 3D processor-DRAM integrated systems can feature drastically reduced memory access latency and increased memory access bandwidth. The computer architecture community has recognized the significance of this scenario, and several researchers have recently explored and demonstrated the potential of 3D processor-DRAM integrated computing systems.[1-5]

Most prior research on 3D processor-DRAM integration featured several conventional commodity 2D DRAM dies stacked as main memory.[1-3] Loh, for example,[4] demonstrated that the performance of 3D processor-DRAM integrated systems could be further improved through Tezzaron Semiconductors' nonconventional, so-called "true" 3D DRAM design strategy.[6] This strategy essentially involves locating DRAM cell arrays and DRAM peripheral circuits on separate dies so that designers can use high-performance logic dies to implement DRAM peripheral circuits and, consequently, improve the speed and reduce silicon area. Such an aggressive design strategy, however, tends to demand that through-silicon via (TSV) pitch be comparable to that of DRAM word line or bit line (typically, TSV pitch is on the order of 10× the DRAM word line or bit line pitch), which can result in nontrivial TSV fabrication challenges, particularly as DRAM technology continues to scale down. Moreover, prior research assumed that 3D DRAM has a homogeneous architecture, used either as main memory or as cache, in which the performance gain results from reduced access latency and increased bandwidth between the last-level on-chip cache and 3D stacked DRAM.

This article contributes to the state of the art of 3D processor-DRAM integrated computing systems design in two ways. First, we present a coarse-grained 3D partitioning strategy for 3D DRAM design, to effectively exploit the benefits of 3D integration without incurring stringent constraints on TSV fabrications. The key is to share the global routing of both the memory address bus and the data bus among all the DRAM dies through coarse-grained TSVs, with the pitch ranging in the tens of microns. To demonstrate the effectiveness of this proposed 3D DRAM design approach, we have modified Hewlett-Packard's CACTI 5, an integrated tool that models cache and

0740-7475/09/$26.00 © 2009 IEEE Copublished by the IEEE CS and the IEEE CASS

memory access time, cycle time, area, leakage, and dynamic power.[7]

Second, we demonstrate, through simulations on a target multicore computing system that we conducted, the potential for using a heterogeneous 3D DRAM architecture to implement both L2 cache and main memory within the 3D stacked DRAM dies. Although DRAM is commonly believed to be far slower than SRAM, we show that by using the modified CACTI tool, 3D DRAM L2 cache can achieve comparable or faster speed than 2D SRAM L2 cache, especially for an L2 cache with large capacity (say, 2 Mbytes). We have evaluated a representative heterogeneous 3D DRAM architecture by using Binkert et al.'s M5 full system simulator for a four-core computing system.[8] Our results show that, compared with using a homogeneous 3D DRAM as main memory, a heterogeneous 3D DRAM design strategy improves the normalized harmonic mean instructions per cycle (IPC) by more than 23.9% on average over a wide spectrum of multiprogrammed workloads.

Background: 3D integration

In general, 3D integration refers to a variety of technologies that provide electrical connectivity between stacked, multiple active device planes. Researchers have been investigating three categories of 3D integration technologies:

- *3D packaging technology.* Enabled by wire bonding, flip-chip bonding, and thinned die-to-die bonding, this is the most mature 3D integration technology and is already being used in many commercial products, noticeably in cell phones. Its major limitation is very low interdie interconnect density (for example, only a few hundred interdie bonding wires) compared to the other emerging 3D integration technologies.
- *Transistor build-up 3D technology.* This technology forms transistors layer by layer, on polysilicon films, or on single-crystal silicon films. Although a drastically high vertical interconnect density can be realized, it is not readily compatible with the existing fabrication process and is subject to severe process temperature constraints that tend to degrade the circuit electrical performance.
- *Monolithic, wafer-level, BEOL-compatible (back end of the line) 3D technology.* Enabled by wafer alignment, bonding, thinning, and interwafer interconnections, this technology uses TSVs to realize a die-to-die interconnect. BEOL-compatible 3D technology appears to be the most promising option for high-volume production of highly integrated systems.

The simulation results we describe in this article focus on the use of wafer-level BEOL-compatible 3D integration technology in the design of 3D processor-DRAM integrated computing systems.

Other researchers have considered 3D integration of digital memory previously,[1-4,9] largely because of the potential of 3D processor-memory integration to address and help resolve the looming memory wall problem. One option for 3D memory integration is to directly stack several memory dies connected with high-bandwidth through-silicon vias (TSVs), in which all the memory dies are designed separately using conventional 2D SRAM or commodity DRAM design practice. Such direct memory stacking has been assumed by Liu et al. and Kgil et al.[1,2] Intuitively, although this option requires almost no changes in the memory circuit and architecture design, direct memory stacking might be unable to exploit the potential benefit of 3D integration to its full extent. As previously mentioned, Loh[4] investigated the potential of 3D processor-memory integration,[6] locating DRAM cell arrays and DRAM peripheral circuits on separate dies so that the high-performance logic dies implemented DRAM peripheral circuits. In a different approach, Tsai et al.'s research evaluated two 3D SRAM design strategies that partitioned word lines and bit lines in the 3D domain, respectively, and explored the corresponding 3D SRAM performance space by modifying Hewlett-Packard's CACTI 3 tool.[5] The 3D memory design strategies others have explored essentially used intrasubarray 3D partitioning,[5,6] which requires the fabrication of a relatively large number of TSVs, and the pitch of TSVs must be comparable to the memory word line and bit line pitch.

Coarse-grained 3D DRAM strategy

As semiconductor technology scales down, interconnects tend to play an increasingly important role, particularly in high-capacity DRAMs. The results we achieved, in estimates with the CACTI 5 tool,[7] clearly demonstrate the significant role of the global interconnect in determining the overall DRAM performance; global H-tree routing

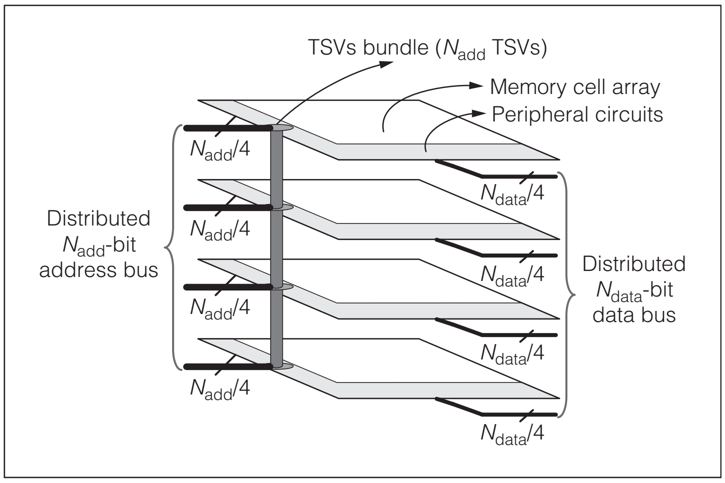

Figure 1. Illustration of a four-layer 3D subarray set.

accounts for 28% of the area, 54% of the access latency, and 69% of the energy consumption. Consequently, we have developed a coarse-grained partitioning strategy for 3D DRAM architecture that mainly aims at leveraging die stacking to substantially reduce the overhead induced by global H-tree routing in DRAM. Compared with prior research,[6,9] which essentially uses intrasubarray 3D partitioning (that is, each individual memory subarray, including the memory cells and peripheral circuits, is split across several dies), this work moves 3D partitioning up to the intersubarray level, leading to far fewer TSVs and less-stringent constraints on TSV pitch. As a result, this design solution accommodates a potentially significant mismatch between the DRAM word line and bit line pitch and TSV pitch.

Similar to current DRAM design practice, our proposed 3D DRAM memory with coarse-grained intersubarray 3D partitioning also has a hierarchical architecture, consisting of banks, subbanks, and subarray sets. Each bank is divided into subbanks, and the data is read to and written from one subbank during each memory access. Each subbank is further divided into 3D subarray sets, which contain n individual 2D subarrays. In particular, each 2D subarray contains the memory cells and its own complete peripheral circuits such as decoders, drivers, and sense amplifiers (SAs). Within each 3D subarray set, all the 2D subarrays share only address and data I/O through TSVs, which clearly requires far fewer TSVs compared to the intrasubarray 3D partitioning. Moreover, the global address and data routing outside and inside each bank can be distributed simply across all the n layers through TSVs. Our proposed intersubarray 3D partitioning possesses two attractive advantages in particular:

- Because the 3D distributed global H-tree routing is shared by all the DRAM dies, the footprint of the H-tree is minimized, leading to a substantially reduced access latency, energy consumption, and DRAM footprint.
- Because each individual memory subarray is retained in one layer, we can keep exactly the same subarray circuit design as is currently done, and the operational characteristics of each subarray are insensitive to the parasitics of TSVs.

Figure 1 shows the design realization of each 3D subarray set. Let N_{data} and N_{add} denote the data access and address input bandwidth of each 3D subarray set, and recall that n denotes the number of memory layers. Without loss of generality, we assume that N_{data} and N_{add} are divisible by n. As Figure 1 shows, the N_{add} bit address bus is uniformly distributed across all n layers and shared by all n 2D subarrays within the same 3D subarray set through a TSV bundle; the N_{data} bit data bus is uniformly distributed across all the n layers outside the subarray set. All n 2D subarrays participate in the read/write operations, that is, each 2D subarray handles the read/write of N_{data}/n bits.

This coarse-grained 3D DRAM partitioning strategy lets us keep exactly the same subarray circuit design as we do currently with 2D DRAM design, and the operational characteristics of each subarray are insensitive to TSV parasitics. Compared with finer-grained 3D partitioning, this design strategy demands far fewer TSVs and a significantly relaxed TSV pitch constraint. To evaluate our proposed partitioning strategy, we modified CACTI 5, as we explain later.

Heterogeneous 3D DRAM

The key element of our proposed 3D processor-DRAM integrated computing systems is to incorporate a *heterogeneous* 3D DRAM architecture to cover more than one memory level within the entire computer memory hierarchy. In contrast, other researchers' work assumed a homogeneous 3D DRAM as either cache or main memory. As mentioned, the work we discuss here considers the use of 3D DRAM to implement a private large capacity L2 cache, such as that in a multicore computing system, for each core and main memory shared by all the cores. Figure 2b further illustrates this approach.

Because L2 cache access latency plays a critical role in determining the overall computing system performance, we could argue that, compared with on-chip SRAM L2 cache, 3D DRAM L2 cache might suffer from much longer access latency and therefore significantly degrade computing system performance. This intuitive argument might not necessarily hold true, however; in particular, as we increase the L2 cache capacity and the number of DRAM dies, the 3D DRAM L2 cache might well have an even shorter access latency than its 2D SRAM L2 cache counterpart. The common impression that DRAM is far slower than SRAM results from the fact that, because it's a commodity, DRAM has always been optimized largely for density and cost, rather than for speed. We can greatly improve DRAM speed by using a dual approach:

- We can reduce the size of each individual DRAM subarray to reduce the memory access latency, at the penalty of storage density. With shorter lengths of word lines and bit lines, a smaller DRAM subarray can directly lead to reduced access latency because of the reduced load of the peripheral decoders and bit lines.
- We can adopt the multiple threshold voltage (multi-V_{TH}) technique that has been widely used in logic circuit design; that is, we still use high-V_{TH} transistors in DRAM cells to maintain a very low cell leakage current, but we use low-V_{TH} transistors in peripheral circuits and H-tree buffers to reduce latency. Such multi-V_{TH} design is typically not used in commodity DRAM because it increases leakage power consumption and, more importantly, complicates the DRAM fabrication process, thereby incurring higher cost.

Moreover, as we increase the L2 cache capacity, H-tree routing plays a bigger role in determining the overall L2 cache access latency. By applying the 3D DRAM design strategy we've described, we can directly reduce the latency incurred by H-tree routing, further reducing 3D DRAM L2 cache access latency compared with a 2D SRAM L2 cache.

One potential penalty when using DRAM to realize an L2 cache is that the periodic DRAM refresh operations might degrade the DRAM-based L2 cache performance. Fortunately, because of the relatively small storage (a few Mbytes) capacity of an L2 cache and the relatively long refresh interval (64 or 128 ms) for each memory cell, refresh might not necessarily induce noticeable L2 cache performance degradation.

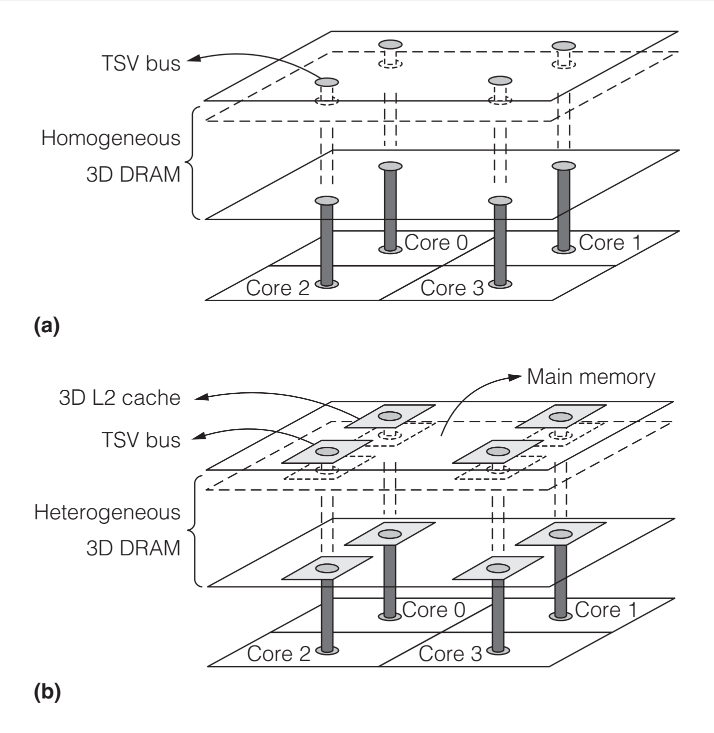

Figure 2. Homogeneous 3D DRAM assumed in earlier work by other researchers (a), and proposed heterogeneous 3D DRAM covering both L2 cache and main memory (b).

For example, consider a 2-Mbyte L2 cache with a 64-byte block size and 65,536 cache blocks. Even for the worst case in which each cache block is refreshed independently, the DRAM L2 cache need only refresh one cache block every 976 ns if we assume the typical DRAM memory cell refresh interval of 64 ms. This refresh rate means the availability of DRAM cache is as high as 99.7%. Although refresh operations tend to increase energy dissipation, L2 cache memory has relatively high access activity, which means the energy overhead induced by refresh might be insignificant. We can, moreover, considerably reduce this energy overhead by using efficient refresh management techniques.[10]

Simulation results

To examine the potential of 3D DRAM, we conducted two simulations that involved 3D DRAM modeling and multicore computing systems.

Table 1. 3D 1-Gbyte main memory configuration parameters.

Page size	Bus width	No. of banks	No. of subbanks or banks	Subarray size
4 Kbytes	512 bits	8	16	2,048 × 1,024

Table 2. 3D 2 MB L2 cache configuration parameters.

Associativity	Block size	Bus width	No. of banks
8-way	64-byte	256 bits	1

3D DRAM modeling

By modifying CACTI 5, we developed a 3D DRAM modeling tool to support our proposed coarse-grained intersubarray 3D DRAM partitioning design approach. For use with the tool, we chose the TSV size to be 10 microns × 10 microns, and we assumed its resistance could be ignored because of its relatively large size. To evaluate our 3D DRAM design approach, we considered a 1-Gbyte DRAM main memory design at the 45-nm technology node. For comparison purposes, we also evaluated two other design options: first, the conventional 2D design, and second, 3D packaging of separate DRAM dies using wire bonding instead of TSVs (referred to as 3D die packaging).

Table 1 shows the configuration parameters used in DRAM modeling, where the CACTI tool automatically chose the subbank number in each bank and subarray size. Figure 3 shows the estimated 3D 1-Gbyte DRAM main memory footprint, access latency, and energy consumption using our proposed 3D DRAM architecture and 3D die packaging in the 45-nm technology node. The obvious advantages of our proposed 3D design over 3D die packaging result from the fact that the global routing is distributed over all the DRAM dies and so result in less overhead induced by global routing.

Using the CACTI-based modeling tool we modified, we further evaluated the effectiveness when using 3D DRAM to implement an L2 cache at the 45-nm technology node. All the DRAM cache access latency values we discuss refer to the random memory access latency. Table 2 lists the basic parameters. As we varied the subarray size from 512 × 512 to 128 × 64, the access latency could be reduced by up to 1.48 ns. During the simulation, the subarray size of 2D SRAM was 256 × 256, and we set the subarray size of 3D DRAM to be 128 × 64. Our results show that, the multi-V_{TH} DRAM L2 cache significantly outperformed its single-V_{TH} DRAM counterpart. In particular, as we increased the capacity of the L2 cache, the multi-V_{TH} DRAM L2 cache already excelled over its SRAM counterpart in terms of access latency. We could additionally improve the access latency with 3D die stacking.

Figure 4 shows the overall comparison of access latency, footprint, and energy consumption of the 2-Mbyte L2 cache using 2D SRAM and various 3D DRAM implementation options. The results show that the access latency advantage of multi-V_{TH} 3D

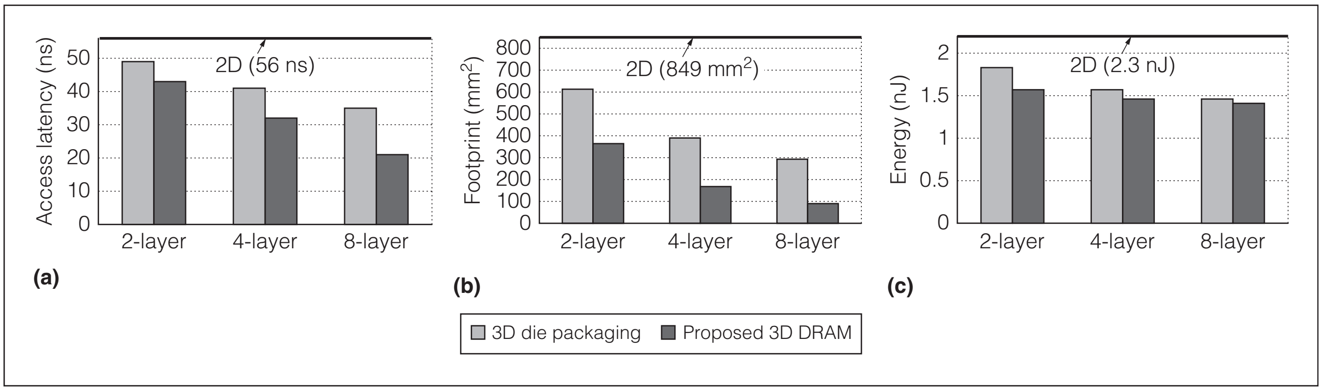

Figure 3. Estimated results of access latency (a), footprint (b), and energy consumption for the 1-Gbyte DRAM design using different design approaches at the 45-nm node based on the *International Technology Roadmap for Semiconductors* projection (c).

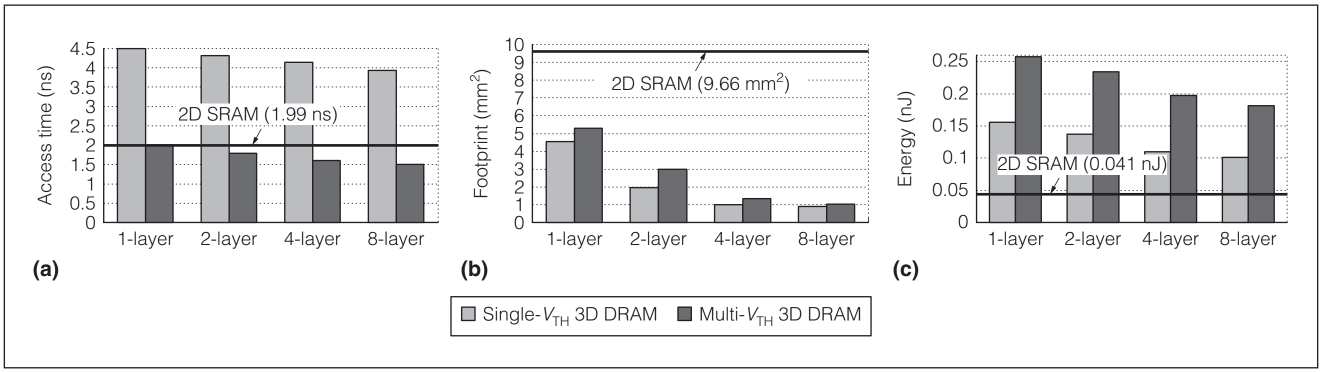

Figure 4. Comparison of access latency (a), footprint (b), and energy consumption of the 2-Mbyte L2 cache with different implementation options (c).

DRAM over 2D SRAM improved further as we increased the number of DRAM dies. This is mainly because, as we stack more DRAM dies, the footprint and hence latency incurred by H-tree routing diminishes accordingly. As Figure 4 shows, the penalty of moving the L2 cache into the DRAM domain is higher energy consumption, due to the inherently high-power operational characteristics of DRAM compared with SRAM. This essentially agrees with the conclusions drawn from recent work on embedded silicon-on-insulator DRAM design that shows embedded SOI DRAM can achieve shorter access latency than its SRAM counterpart at a capacity as small as 256 Kbytes.[11]

Multicore computing systems

We conducted full system simulations to further evaluate the effectiveness of the heterogeneous 3D DRAM architecture. Here, we first outline the configurations of the processor microarchitecture, memory system, and 3D DRAM stacking, using a four-core, 45-nm processor design. We estimated the total die size of four cores without the L2 cache as 100 mm^2.

By using the CACTI-based 3D DRAM modeling tool that we modified, we have found that a 100 mm^2 die size with eight stacked DRAM dies can achieve a total storage capacity of 1 Gbyte at the 45-nm node. We also used the CACTI-based 3D tool to estimate the access time for each memory module. For performance evaluation, we used the M5 full system simulator running a Linux kernel.[8] Each core used an out-of-order issue and executed the Alpha ISA. Table 3 lists the basic configuration parameters we used in our simulation.

We first consider the following three different scenarios in which processor cores are stacked with 3D DRAM:

- *Baseline.* The four-core processor was stacked with an eight-layer homogeneous 3D DRAM shared main memory, and each core had only its own private L1 cache on the logic die.
- *Heterogeneous single-*V_{TH} *3D DRAM.* The four-core processor was stacked with an eight-layer 3D DRAM that implemented private L2 caches for all the cores and a shared main memory in which the L2 caches used single-V_{TH} transistors.
- *Heterogeneous multi-*V_{TH} *3D DRAM.* The four-core processor was stacked with an eight-layer 3D DRAM that implemented private L2 caches for

Table 3. Configuration parameters of the simulated multicore processor.

Configuration parameter	Configuration value
Frequency	4 GHz
No. of cores	4
Die size	100 mm^2
No. of DRAM dies	8
Core type	Out-of-order issue
L1 cache	16 Kbytes, 4-way, 64-byte blocks, private SRAM on the logic die, access latency: 0.26 ns
L2 cache	2 Mbytes, 8-way, 64-byte blocks, private 8-layer 3D DRAM, random access latency: 3.93 ns (single-V_{TH}); 1.51 ns (multi-V_{TH})
Main memory	1 Gbyte, 4-Kbyte page size, shared 8-layer 3D DRAM, access latency: 20.95 ns

Table 4. Memory configuration of the 2D SRAM L2-cache and 2D DRAM L2-cache scenarios. (The access latency is estimated using the CACTI-based 3D DRAM modeling tool.)

Type of 2D L2 cache	L2 cache	Main memory
2D SRAM	2 Mbytes, 8-way, 64-byte blocks, private 2D SRAM on logic die, access latency: 4.32 ns	1 Gbyte, 4-Kbyte page size, 8-layer 3D DRAM, access latency: 20.95 ns
2D DRAM	64 Mbytes, 32-way, 64-byte blocks, shared 2D DRAM stacked on logic die, access latency: 16.4 ns	1 Gbyte, 4 Kbyte-page size, access latency: 300 cyclesoff-chip DRAM; 20.95 ns on-chip DRAM

all the cores and a shared main memory in which the L2 caches used multi-V_{TH} transistors to further reduce L2 cache access latency.

In addition, we also considered the following two scenarios to compare this work with other processor-DRAM integration systems.

- *2D SRAM L2 cache.* The four-core processor was stacked with an eight-layer 3D DRAM shared main memory, and each core had its private L1 and L2 cache on the logic die. This is the most intuitive 3D processor-DRAM integrated architecture assumed by many prior works.
- *2D DRAM L2 cache.* The four-core processor was stacked with a single-layer 2D DRAM die that served as the shared L2 cache, while the main memory was off-chip. This is the 3D processor-DRAM integrated architecture that Black et al. used.[5] In addition, we also simulated this 2D DRAM L2 cache with stacked 3D DRAM main memory to ensure a fair comparison.

The system configuration of these two scenarios was the same as the other three, except for the memory hierarchy parameters, which are listed in Table 4.

Our simulations used multiprogrammed workloads to evaluate the performance of the five different scenarios. We first recorded the L2 cache miss statistics of various benchmarks, in which each benchmark ran on a single core with the 2-Mbyte L2 cache. Next, we selected 24 representative benchmarks (see Table 5) that cover a wide

Table 5. Simulation benchmarks.

Benchmarks		MPKI	Benchmarks		MPKI
Name	Suite	(2 Mbytes)	Name	Suite	(2 Mbytes)
mcf	I'00	43.05	eon	I'06	1.69
tigr	Bio	20.59	libquantum	I'06	1.49
lbm	F'06	14.62	gcc	I'00	1.06
mummer	Bio	3.45	apsi	F'06	1.06
crafty	I'00	3.26	wupwise	F'00	0.93
namd	F'06	3.00	h264	F'06	0.85
swim	F'00	2.91	astar	I'06	0.78
twolf	I'00	2.71	equake	F'00	0.75
vortex	I'00	2.65	omnetpp	I'06	0.62
mesa	F'00	2.54	applu	F'06	0.58
milc	F'06	2.30	bzip2	I'06	0.53
soplex	F'06	1.79	gzip	I'02	0.49

* Bio: BioBench; F: SpecFP; I: SpecInt; '00: cpu2000; '06: cpu2006.

spectrum of L2 cache miss statistics represented as L2 miss per kilo-instructions (MPKI). These benchmarks included SPECcpu's integer and floating-point suites from the Standard Performance Evaluation Corp. in both the 2000 and 2006 editions (see http://www.spec.org), and bioinformatics workloads.[12] We further grouped these benchmarks to form 12 multiprogrammed workloads, as listed in Table 6, falling into three categories: all high-miss (H), all low-miss (L), and a mix of both high and low miss (HL). For each workload, we fast-forwarded the first half billion instructions and ran simulations for the next 500 million instructions.

Table 6. Four-threaded workloads.

High (H) L2 cache MPKI		Normalized harmonic
Workloads	**Benchmark name**	**mean IPC (HMIPC)**
H1	mcf, tigr, 1bm, mummer	0.499
H2	crafty, namd, swim, twolf	0.702
H3	vortex, mesa, milc, soplex	0.489
High-Low (HL) mixes		
Workloads	**Benchmark name**	**(HMIPC)**
HL1	mcf, tigr, gcc, apsi	0.302
HL2	1bm, mummer, eon, libquantum	0.482
HL3	crafty, namd, wupwise, h264	0.618
HL4	swim, twolf, astar, equake	0.912
HL5	vortex, mesa, omnetpp, applu	0.871
HL6	milc, soplex, bzip2, gzip	0.752
Low (L) L2 cache MPKI		
Workloads	**Benchmark name**	**(HMIPC)**
L1	eon, libquantum, gcc, apsi	0.492
L2	wupwise, h264, astar, equake	0.833
L3	omnetpp, applu, bzip2, gzip	0.889

The results in Figure 5 show the performance of the proposed heterogeneous 3D DRAM architecture in a processor-DRAM integration system. To clarify the graph, we show the normalized harmonic mean IPC (HMIPC) improvement of the other two scenarios compared against the baseline scenario. The results clearly show that even the simplest single-V_{TH} design option can largely improve the performance for all the benchmarks by more than 16.3% on average. Switching from single-V_{TH} to multi-V_{TH} can bring another 7.6% performance improvement, on average, as the multi-V_{TH} design directly reduces the L2 cache access latency.

Simulation results shown in Figure 6 demonstrate the performance improvement we obtained by moving the private L2 cache from on-chip SRAM to 3D stacked DRAM. Moreover, using SRAM L2 cache tends to drastically increase the processor die size and, accordingly, the fabrication cost. In the simulated multicore processor, the extra area overhead devoted to on-chip SRAM L2 cache can reach up to 38.64%. If we consider the interconnect between the processor core and the L2 cache, the area overhead could be even higher. This clearly suggests that using SRAM to design a large-capacity cache is area-inefficient, especially when 3D integration technology enables the use of 3D stacked DRAM as the L2 cache. Therefore, implementing lower-level cache memories using 3D stacked DRAM is an appealing option for emerging 3D integrated computing systems.

In Figure 7, simulation results show that a multicore processor using the proposed heterogeneous 3D DRAM integration outperforms the 3D processor-DRAM integration that Black et al. proposed.,[5] even though its performance has been substantially improved by integrating main memory on-chip. This suggests that the direct use of commodity 2D DRAM as a large-capacity shared L2 cache tends to considerably degrade performance. Therefore, the L2 cache should be of a reasonable size, and the DRAM memory's access latency must be reduced by leveraging specific techniques, such as small subarray size and multi-V_{TH}, as we've proposed.

Finally, we analyzed the system configuration sensitivity of the proposed processor-DRAM integrated multicore computing systems. Tables 7 and 8 list the HMIPC improvement against the baseline with different system configurations. The results clearly show that this proposed design strategy can perform very well over a large range of system configurations.

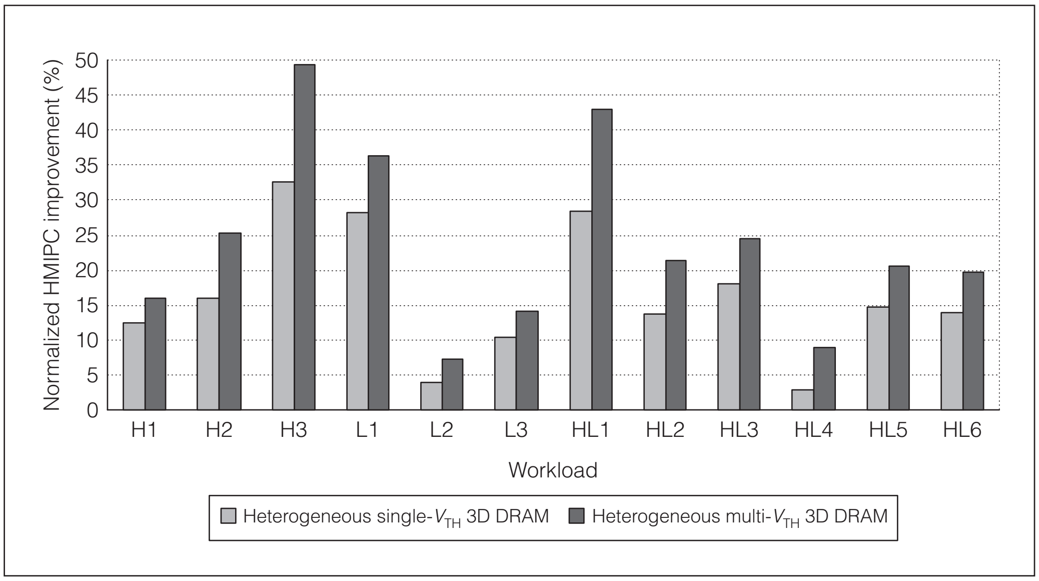

Figure 5. Normalized harmonic mean IPC (HMIPC) improvement comparison among baseline, heterogeneous single-V_{TH} 3D DRAM and heterogeneous multi-V_{TH} 3D DRAM.

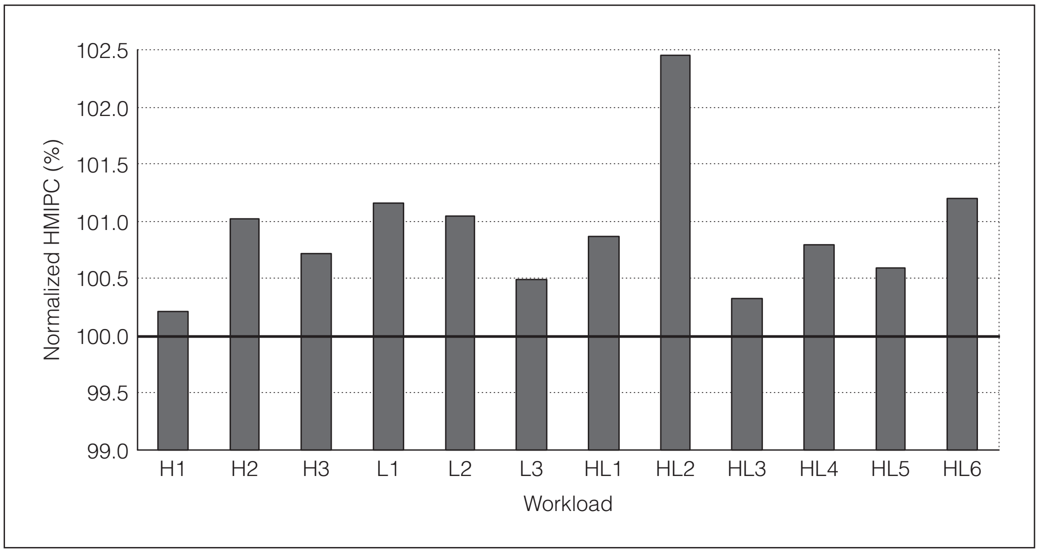

Figure 6. Normalized HMIPC improvement when using a multi-V_{TH} 3D DRAM private L2 cache to replace an on-chip SRAM private L2 cache for each core.

The key to the 3D DRAM design approach we have discussed is to apply a coarse-grained 3D partitioning strategy and effectively exploit the benefits provided by 3D integration without incurring stringent constraints on TSV fabrications. 3D processor-memory integration appears to the most viable approach to fundamentally address the looming memory wall problem and enable the most effective use of future multi- and manycore microprocessors. Certainly, much more research is required in order to fully exploit the potential and understand involved design trade-offs, particularly the thermal, power delivery and cost issues. ■

Acknowledgments

This research was funded in part by the Key Program of the Nation Natural Science Foundation of China under grant 60635050, the National Science Foundation under grant 0823971, and the Semiconductor Research Corp. Global Research Consortium under contract 2008-HJ-1794.

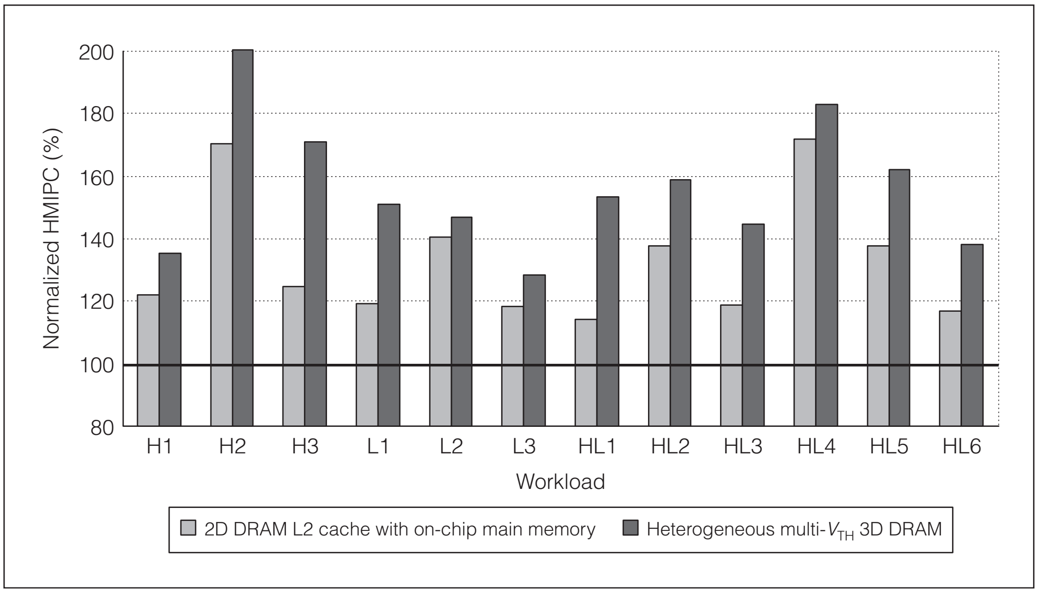

Figure 7. Normalized HMIPC improvement when using a multi-V_{TH} 3D DRAM private L2 cache to replace a 2D DRAM shared L2 cache.

Table 7. System sensitivity to L1 cache size and degree of set associativity.

	Cache size (Kbytes)			Set associativity		
	16	32	64	2-way	4-way	8-way
HMIPC improvement	23.9%	21.2%	20.7%	25.1%	23.9%	22.7%

Table 8. System sensitivity to L2 cache size and degree of set associativity.

	Cache size (Mbytes)			Set associativity		
	1	2	4	8-way	16-way	32-way
HMIPC improvement	23.2%	23.9%	24.7%	23.9%	24.1%	24.9%

References

1. C.C. Liu et al., "Bridging the Processor-Memory Performance Gap with 3D IC Technology," *IEEE Design & Test,* vol. 22, no. 6, 2005, pp. 556-564.
2. T. Kgil et al., ''PicoServer: Using 3D Stacking Technology to Enable a Compact Energy Efficient Chip Multiprocessor,'' *Proc. 12th Int'l Conf. Architectural Support for Programming Languages and Operating Systems* (ASPLOS 06), ACM Press, 2006, pp. 117-128.
3. G. Loh, Y. Xie, and B. Black, "Processor Design in 3D Die-Stacking Technologies," *IEEE Micro,* vol. 27, no. 3, 2007, pp. 31-48.
4. G. Loh, ''3D-Stacked Memory Architecture for Multicore Processors,'' *Proc. 35th ACM/IEEE Int'l Symp. Computer Architecture* (ISCA 08), IEEE CS Press, 2008, pp. 453-464.
5. B. Black et al., ''Die Stacking (3D) Microarchitecture,'' *Proc. Ann. IEEE/ACM Int'l Symp. Microarchitecture,* IEEE CS Press, 2006, pp. 469-479.
6. Tezzaron Semiconductors, "3D Stacked DRAM/Bi-STAR Overview," 2008; http://www.tachyonsemi.com/memory/Overview_3D_DRAM.htm.
7. CACTI: An Integrated Cache and Memory Access Time, Cycle Time, Area, Leakage, and Dynamic Power Model, http://www.hpl.hp.com/research/cacti/.

8. N.L. Binkert et al., ''The M5 Simulator: Modeling Networked Systems,'' *IEEE Micro,* vol. 26, no. 4, 2006, pp. 52-60.
9. Y.-F. Tsai et al., ''Design Space Exploration for 3-D Cache,'' *IEEE Trans. Very Large Scale Integration (VLSI) Systems,* vol. 16, no. 4, 2008, pp. 444-455.
10. M. Ghosh and H.-H.S. Lee, ''Smart Refresh: An Enhanced memory Controller Design for Reducing Energy in Conventional and 3D Die-Stacked DRAMs,'' *Proc. 40th ACM/IEEE Int'l Symp. Microarchitecture,* IEEE CS Press, 2007, pp. 134-145.
11. J. Barth et al., ''A 500 MHz Random Cycle, 1.5 ns Latency, SOI Embedded DRAM Macro Featuring a Three-Transistor Micro Sense Amplifier,'' *IEEE J. Solid-State Circuits,* Jan. 2008, pp. 86-95.
12. K. Albayraktaroglu et al., ''Biobench: A Benchmark Suite of Bioinformatics Applications,'' *Proc. Int'l Symp. Performance Analysis of Systems and Software,* IEEE CS Press, 2005, pp. 2-9.

Hongbin Sun is a postdoctoral student in the School of Electronic and Information Engineering at Xi'an Jiaotong University, China. His research interests include fault-tolerant computer architecture, 3D memory-processor integration, and VLSI architecture for digital video processing. Sun has a PhD in electrical engineering from Xi'an Jiaotong University.

Jibang Liu is a PhD student in the Department of Electrical, Computer, and Systems Engineering at Rensselaer Polytechnic Institute. His research is mainly focused on VLSI design. Liu has an MS in electrical engineering from Rensselaer Polytechnic Institute.

Rakesh S. Anigundi is an ASIC design engineer at Qualcomm. His research interests include 3D SoC system design, 3D EDA tool enablements, and 3D ASIC flow design. Anigundi has an MS in electrical engineering from Rensselaer Polytechnic Institute.

Nanning Zheng is a professor and the director of the Institute of Artificial Intelligence and Robotics at 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. Zheng has a PhD in electrical engineering from Keio University. He is a Fellow of the IEEE.

James Jian-Qiang Lu is an associate professor in the Electrical, Computer, and Systems Engineering Department at Rensselaer Polytechnic Institute, where he leads the 3D hyperintegration technology research programs. His research interests include 3D hyperintegration design and technology, and micro-nano-bio interfaces for future chips and microelectromechanical systems. Lu has a Dr.rer.nat. (PhD) from the Technical University of Munich. He is a senior member of the IEEE and is a member of the American Physical Society, the Materials Research Society, and the Electrochemical Society.

Kenneth Rose is professor emeritus in the Electrical, Computer, and Systems Engineering Department at Rensselaer Polytechnic Institute. His research interests include VLSI performance prediction, mixed-signal design, and error correction in flash memories. Rose has a PhD in electrical engineering from the University of Illinois at Urbana-Champaign.

Tong Zhang is an associate professor in the Electrical, Computer, and Systems Engineering Department at Rensselaer Polytechnic Institute. His research interests include algorithm and architecture codesign 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 and nanodevice electronic systems. Zhang has a PhD in electrical engineering from the University of Minnesota, Twin Cities. He is a senior member of the IEEE.

■ Direct questions and comments about this article to Hongbin Sun, Institute of Artificial Intelligence and Robotics, Xi'an Jiaotong University, Shaanxi, 710049, P.R. China; sunsir@mail.xjtu.edu.cn.

IEEE computer society

PURPOSE: The IEEE Computer Society is the world's largest association of computing professionals and is the leading provider of technical information in the field.
MEMBERSHIP: Members receive the monthly magazine *Computer*, discounts, and opportunities to serve (all activities are led by volunteer members). Membership is open to all IEEE members, affiliate society members, and others interested in the computer field.
COMPUTER SOCIETY WEB SITE: www.computer.org
OMBUDSMAN: To check membership status or report a change of address, call the IEEE Member Services toll-free number, +1 800 678 4333 (US) or +1 732 981 0060 (international). Direct all other Computer Society-related questions—magazine delivery or unresolved complaints—to help@computer.org.
CHAPTERS: Regular and student chapters worldwide provide the opportunity to interact with colleagues, hear technical experts, and serve the local professional community.
AVAILABLE INFORMATION: To obtain more information on any of the following, contact Customer Service at +1 714 821 8380 or +1 800 272 6657:

- Membership applications
- Publications catalog
- Draft standards and order forms
- Technical committee list
- Technical committee application
- Chapter start-up procedures
- Student scholarship information
- Volunteer leaders/staff directory
- IEEE senior member grade application (requires 10 years practice and significant performance in five of those 10)

PUBLICATIONS AND ACTIVITIES
***Computer*:** The flagship publication of the IEEE Computer Society, *Computer*, publishes peer-reviewed technical content that covers all aspects of computer science, computer engineering, technology, and applications.
Periodicals: The society publishes 14 magazines, 12 transactions, and one letters. Refer to membership application or request information as noted above.
Conference Proceedings & Books: Conference Publishing Services publishes more than 175 titles every year. CS Press publishes books in partnership with John Wiley & Sons.
Standards Working Groups: More than 150 groups produce IEEE standards used throughout the world.
Technical Committees: TCs provide professional interaction in over 45 technical areas and directly influence computer engineering conferences and publications.
Conferences/Education: The society holds about 200 conferences each year and sponsors many educational activities, including computing science accreditation.
Certifications: The society offers two software developer credentials.
For more information, visit www.computer.org/certification.

IEEE
Celebrating 125 Years
of Engineering the Future

revised 1 May 2009