Heterogeneous Die Stacking of SRAM Row Cache and 3-D DRAM: An Empirical Design Evaluation

Dong Hyuk Woo[†]
dong.hyuk.woo@intel.com

Nak Hee Seong
nhseong@ece.gatech.edu

Hsien-Hsin S. Lee
leehs@gatech.edu

[†]Platform Architecture Research, Intel Labs
Santa Clara, CA 95054

School of Electrical and Computer Engineering
Georgia Institute of Technology
Atlanta, GA 30332

ABSTRACT

As DRAM scaling becomes more challenging and its energy efficiency receives a growing concern for data center operation, an alternative approach— stacking DRAM die with thru-silicon vias (TSV) using 3-D integration technology is being undertaken by industry to address these looming issues. Furthermore, 3-D technology also enables heterogeneous die stacking within one DRAM package. In this paper, we study how to design such a heterogeneous DRAM chip for improving both performance and energy efficiency, in particular, we propose a novel floorplan and several architectural techniques to fully exploit the benefits of 3-D die stacking technology when integrating an SRAM row cache into a DRAM chip. Our multi-core simulation results show that, by tightly integrating a small row cache with its corresponding DRAM array, we can improve performance by 30% while saving dynamic energy by 31% for memory intensive applications.

1. INTRODUCTION

DRAM industry is facing several imminent challenges from the limitation posed by fundamental physics and also from increasing needs by consumers. First of all, the DRAM industry is facing a scaling challenge. As the device feature size continues to shrink, the capacitance of the DRAM cell also decreases, at the same time, the junction leakage current drastically increases [8]. Therefore, maintaining enough capacitance and reducing leakage current become a significant challenge, making DRAM feature size scaling impractical. On the other hand, the need from DRAM consumers also continues to evolve as the industry rapidly embraces the *cloud computing* paradigm. In addition to density optimization, the emerging trend of cloud computing also drives the DRAM vendors to increase their power efficiency.

In response to the above challenges, the DRAM industry is undertaking novel approaches. One innovative solution is to integrate multiple DRAM die using 3-D die stacking technology, which increases the DRAM density without paying the cost of using a finer lithography technology. For example, Samsung has demonstrated an 8Gb 3-D stacked DDR3 DRAM chip that consists of four DRAM layers [6]; in which three layers are slave layers without any I/O-related circuit while one layer is the master layer that has shared I/Os. Such sharing is enabled by TSVs that allow high bandwidth, low latency, and low power data communication across layers. Such a TSV-based design can effectively reduce a significant amount of standby and active power compared to an SiP-based design [6]. More recently, Elpida announced an 8Gb 3-D DDR3 SDRAM that stacks eight 1Gb DRAM layers and one logic interface layer together [4, 7]. Implementing DRAM cells and interface circuits on separate, heterogeneous layers allows each of them to perform better and to consume less power [4].

Not being satisfied with those benefits, in this paper, we investigate and propose a 3-D stacked DRAM array design called *folded bank* for achieving higher performance and better energy efficiency. This folded bank design enables the integration of a small SRAM row cache on a logic layer in a cost-effective way, which was infeasible in a conventional 2-D DRAM design.

2. MOTIVATIONS

Unlike a conventional, planar DRAM architecture, small, fast, and short TSVs allow one to integrate multiple DRAM die vertically providing high bandwidth, low latency, and low power interconnect across the stack. Furthermore, such a multiple die design allows the accommodation of one unique logic die within the 3-D DRAM stack. for off-chip interface circuits. However, dedicating the logic layer only to interface logic is likely to under-utilize the entire die area, leaving much white space available for implementing other enhancement circuits to further improve the performance and energy efficiency of a 3-D DRAM chip without paying too much additional cost.

Clearly, such a heterogeneous 3-D DRAM chip will enable many possible, interesting designs in the future. In this paper, we investigate a heterogeneous memory architecture as the first step to exploit the opportunities for performance and energy. The motivation of our heterogeneous memory architecture is as follows. According to a recent study [13], although applications may have very good spatial locality, a conventional DRAM architecture cannot exploit such spatial locality because each bank relies only on a single row buffer which causes row conflict misses. Moreover, such conflicts become even more severe as the number of cores on a single die grows [12]. Thus, it is very likely that one process running on one core will close a row opened by another process running on a different core before the latter process could fully utilize its opened row buffer.

Obviously, such redundant DRAM row open operations also consume DRAM energy considerably. To address the problem of such wasted energy, we need an associative SRAM cache in a DRAM chip to keep several active DRAM rows. Although such technique was considered in a conventional DRAM design, it failed to succeed in commodity DRAM due to high integration overhead. We will then demonstrate how the feasibility of a row cache will change with 3-D IC technology by studying the circuit-level design issues with detailed DRAM and TSV models. We will also address several circuit-level design challenges with a few low-cost architectural solutions.

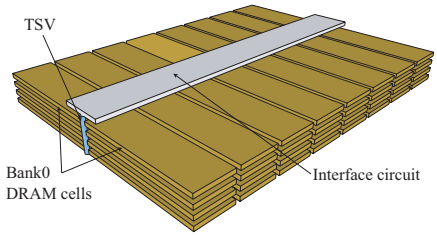
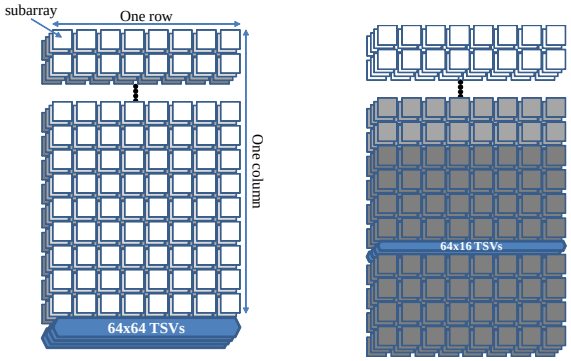


Figure 1: Baseline 3-D Chip (flipped)



(a) Naïve Wide TSV Bus (b) Folded Bank

Figure 2: Different 3-D Design of four Half-Banks (different colors mean different banks.)

3. DESIGNING A TIGHTLY INTEGRATED SRAM AND DRAM STACK WITH TSV

Recently, the emerging 3-D die stacking and TSV technologies have renewed the interest and feasibility of integrating an SRAM row cache into a DRAM chip, which was considered impractical in terms of cost in a 2-D design. In this research, we re-investigated new design issues and studied how computer architects can overcome these prior unresolved obstacles with a novel TSV-enabled heterogeneous 3-D DRAM chip. To perform this study, we first define our baseline 3-D DRAM design (Table 1) that consists of four DRAM die and one logic die as shown in Fig. 1. Here, we assume that each DRAM die implements 8 banks of DDR3 DRAM, similar to Samsung's implementation [6]. On the other hand, we assume a dedicated interface layer similar to Elpida's stack [7]. These DRAM layers and the interface layer are connected through a TSV bus located in the middle of a chip.[1]

Table 1: Baseline DRAM Chip Configuration

DRAM chip capacity	4Gb
# of data pins	8 (x8 chip)
# of banks	8
row size per chip	8Kb

First of all, we try to increase the data width up to the size of a row. To achieve this goal, we place a TSV bus per bank so that four banks stacked vertically can share the TSV bus as shown in Fig. 2(a). (Note that this figure only depicts four 256Mb half-banks.) Below these four banks, we have an SRAM row cache in the logic layer. However, such a naïve design cannot deal with an increased wire count and its corresponding energy inefficiency. To evaluate such a design (and other designs throughout our paper), we modified CACTI 5 [11].[2] According to our modified CACTI, the energy consumption for reading the entire 8Kb from an open row to the logic die is 63.5 times higher than that for reading 64 bits in our baseline. Although such a wide bus design does not need I/O gating circuits and its corresponding column select signals reduce energy consumption in these parts, it consumes a significant amount of energy in the bus between the sense amplifiers and TSVs.

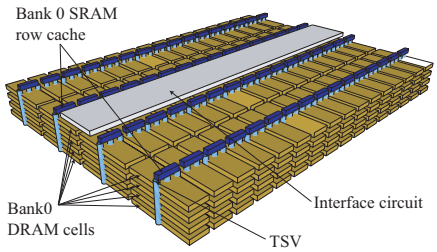


Figure 3: Final Floorplan (flipped)

To solve the energy inefficiency issue, a possible solution is to reduce the bus length by using multiple TSV buses. However, we should consider the trade-off between the dynamic energy consumption and the area overhead, because those multiple TSV buses may require considerably large area. To address this trade-off, we propose to make subarrays of one bank to share the same set of TSVs by folding each bank vertically as shown in Fig. 2(b). By folding each bank, we can reduce the length of wires between the sense amplifiers and the TSV bus. Here, note that this is a scalable design. As the number of DRAM layers increases in the future, we can fold one bank into more layers, which reduces the wire length of each die.

Furthermore, we carefully calculated the wire complexity so that our design does not need more metal layers than our baseline. In our baseline (x8 DRAM) design, eight subarrays of a half-bank form a half row (4Kb) while we fetch 32 bit data from the half-bank.[3] In other words, we fetch 4-bit data from each subarray, which outputs 512 bit-lines. Consequently, the required number of bitline select signals is 128 (= 512 / 4). On the other hands, to allow massive data transfer, our design uses 128 wires between the sense amplifier and the TSV bus. As a result, we need four select signals between the column decoder and each subarray. By using this design, we can equalize the number of wires between the baseline and our design. However, due to such a narrower bus design, the TSV bus width of each half-bank is now 1Kb (128 wires from each subarray). This reduced bus bandwidth forces us to fetch one half row (4Kb) over four cycles. This circuit-level limitation necessitates an architectural technique, which will be detailed later. As a result of such optimization, we are able to design a well-balanced stacked DRAM layers. The detailed results and analysis will be discussed in Section 4.

Our final floorplan based on such a folded bank architecture is shown in Fig. 3. As shown in the figure, we align each bank of our SRAM row cache with its corresponding DRAM bank. By placing an SRAM bank right next to TSVs, we minimize the energy consumed in transferring an entire row to an SRAM bank. Note that if an application does not consume entire row data, the energy consumed to transfer those unused bits to the SRAM row cache is completely wasted because our baseline does not bring those data to the interface circuits at all. This is why we want to minimize the wire length of the bus between the sense amplifiers and the SRAM

[1]Note that the data width of our baseline TSV bus is 64-bit, which is needed in a x8 chip for supporting the minimum burst length of eight by the DDR3 standard.

[2]We assume that DRAM is designed with 32nm technology, and the TSV pitch is 3.6μm in year 2013 [1].

[3]Each subarray is 256Kb with 512 wordlines and 512 bit-lines [2].The size of a subarray does not rapidly change across different DRAM generations [2].

Table 2: tRCD Breakdown

	Baseline	Folded bank
inter-bank address bus	20%	30%
intra-bank address bus	39%	9%
row decoder / wordline	17%	17%
bitline / sense amplifier	24%	24%
total	100%	82%

Table 3: tCL Breakdown

	Baseline	Folded bank
inter-bank address bus	14%	24%
column select	30%	9%
I/O gating / output driver	14%	2%
intra-bank data-out bus	28%	5%
inter-bank data-out bus	14%	0%
total	100%	40%

row cache.

On the other hand, such floorplan necessitates long, cross-chip communication between an SRAM bank and the interface circuits. Note that this long bus is comparable to the bus between the sense amplifiers and the interface circuits in our baseline. In spite of such a long bus between the SRAM row cache and the interface logic, we opted for such design because the bus between the SRAM bank and the interface circuits is just 64-bit, which does not consume much energy and is anyway used by demand requests.

4. EVALUATION

4.1 Circuit-Level Evaluation

For circuit-level modeling, we modified the DRAM model of CACTI 5 [11] as explained in Section 3. With our modified CACTI, we modeled area, delay, and dynamic energy of DRAM. First of all, we evaluate the area overhead of our scheme. Compared to our baseline (Fig. 1), we found that the area overhead of our proposal (Fig. 3) (in DRAM layers) is 5%. To understand the difference, we performed an in-depth analysis and found that the TSV area accounts for most of this overhead. Other than the area occupied by TSVs, we observed minor differences in various components such as the output drivers and column select signal related wires and circuits. These differences are negligibly small compared to the TSV area overhead.

Despite our slightly larger die, we found that our proposed design can actually decrease the access latency of DRAM. In particular, we found that the row-to-column delay (tRCD) and the row precharge delay (tRP) are reduced by 18% and 14%, respectively. Such reduced delay is found to be the result of the reduced wire length within a bank due to the folded bank architecture. This effect is well represented in Table 2. As shown in the table, our folded bank architecture suffers from longer inter-bank bus latency because our new floorplan (Fig. 3) now has 16x4 half-banks instead of 8x2 half-banks of our baseline (Fig. 1), which makes the worst-case inter-bank bus wire longer. However, intra-bank bus latency is found to be reduced significantly because one half-bank is folded across four layers.

On the other hand, the column access strobe latency (tCL) was significantly reduced. As shown in Table 3, the latency of the inter-bank address bus increases, but the latencies of the column select bus and the intra-bank data-out bus decrease due to our new floorplan. Such trend is similar to tRCD. However, one interesting result is that the inter-bank data-bus delay of our new design is zero. This is because our SRAM cache is located right next to the TSV bus

Table 4: Read Energy Breakdown

	Baseline	Folded bank
inter-bank address bus	8%	13%
column select	58%	1%
I/O gating / output driver	4%	64%
intra-bank data-out bus	25%	741%
inter-bank data-out bus	5%	0%
total	100%	818%

Table 5: Memory System Configurations

FR-FCFS scheduling policy, 8B-wide bus, DDR3-1600	
DRAM	tCL-tRCD-tRP: 7-9-8, tRAS: 35 ns, tWR: 15 ns
SRAM row cache + DRAM	tCL-tRCD-tRP: 5-4-7, tRAS: 35 ns, tWR: 15 ns, SRAM access latency: 4 bus clk

storing the entire row data. Note that we suffer from this latency when we read data from our SRAM cache.

Although we have a win in the access latency, our design consumes a significant amount of energy in moving the row data between the sense amplifiers and the SRAM cache. As shown in Table 4, in spite of our reduced wire length due to folding, bringing an entire row data into an SRAM cache consumes a significant amount of energy. Nonetheless, this higher energy consumption of moving rows do not occur frequently in dynamic scenario as the majority of the accesses will be satisfied by the SRAM cache. We will detail the results later. Similar to the read energy, we found that the write energy of the folded bank architecture is 4.6 times higher than that of the baseline. On the other hand, we found that activation energy increases by 18% mainly due to the inter-bank bus energy in our new floorplan.

4.2 Architecture-Level Simulation

In addition to such circuit-level modeling, we also performed architecture-level study using SESC [10]. We extended the SESC simulator to model detailed memory backend. In our simulation, we modeled a quad-core processor with one memory channel. Our quad-core processor has a 4MB L2 cache and a DDR3-1600 like memory interface. When we model the DRAM latency, we used the estimated latency from our modified CACTI. Detailed parameters are listed in Table 5.

Throughout this paper, we simulated 12 sets of multi-programmed workload that consists of four memory-intensive applications from the SPEC2006 benchmark suite.[4] With these 12 sets of workload, we evaluated five memory designs. The first two designs have a conventional DRAM system with an open-row policy and a closed-row policy, respectively. The reason why we evaluated the closed-row policy is to compare the closed-row policy against our scheme that precharges bitlines right after a row is moved to an SRAM row cache. Also, we evaluated our heterogeneous 3-D DRAM chip design where each bank maintains an SRAM row cache that can hold 8, 16, and 32 rows.[5]

Although we simulated all the 12 workloads, we only show their average values due mainly to their similar trends. Also note that, throughout this paper, all reported relative numbers are normalized to that of a conventional DRAM system with an open-row policy, unless otherwise mentioned. First of all, we found that the open-

[4]We defined a memory-intensive application as an application whose number of L2 cache misses per thousand instructions (MPKI) is higher than five when it runs on a single-core processor with a 1MB L2 cache.

[5]In each chip, the capacity of one DRAM bank is 512Mb while that of a 32-entry row cache is 256Kb.

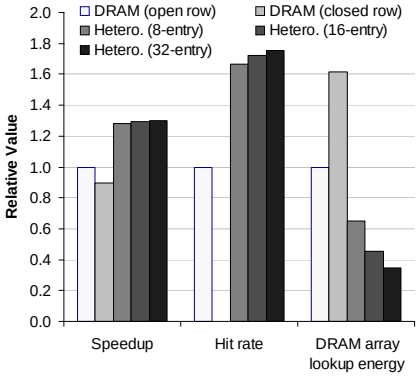


Figure 4: Relative Simulation Results

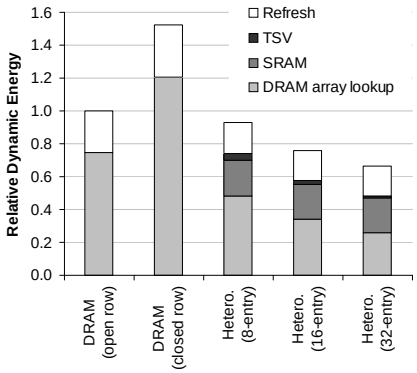


Figure 5: Dynamic Energy Breakdown

row policy is still useful compared to the closed-row policy. Such effect is well explained by the relative hit rate in Fig. 4, which shows the hit rate of a row buffer.[6] Not surprisingly, as we increase the capacity of our SRAM row cache, the hit rate increases leading to higher performance. For example, a heterogeneous 3-D DRAM chip with a 32-entry SRAM row cache can improve performance by 30%, on average. Such improvement was observed by past studies [3, 5, 9, 14], and we just confirm that a row cache is still useful even if we have multiple cores that compete with the shared row cache space.

Fig. 4 also shows relative dynamic energy consumed by DRAM array lookup operations such as row activations, reads, writes, and bit-line precharges. As shown in the figure, we can significantly reduce the DRAM lookup energy. Compared to our baseline, a DRAM chip with a 32-entry SRAM row cache consumes only 35% of energy on average. This result suggests that, even though our new 3-D DRAM design consumes significantly higher read or write energy (Section 4.1), our SRAM row cache can filter out lots of DRAM lookup operations so that the overall DRAM lookup energy is significantly reduced.

However, such reduced DRAM lookup energy does not come for free; we are also spending energy in other additional circuits of our proposed scheme. Thus, to model dynamic energy consumption of the entire chip, we also modeled energy consumption of SRAM lookup operations, TSVs, and refresh operations as shown in Fig. 5. This evaluation suggested that a heterogeneous DRAM chip with a 32-entry SRAM row cache can save dynamic energy of a DRAM chip by 31%, on average. When breaking down energy consumption down further, the TSV energy contributes very little energy while DRAM lookup, SRAM lookup, and refresh operations consume significant energy. Another interesting observation is that the SRAM lookup energy accounts for only 2% of SRAM energy consumption while the other 98% was consumed in the address bus and data bus, which are not sensitive to the capacity of our SRAM row cache. From this observation, we concluded that filtering out more DRAM lookup operations with a larger SRAM cache is more helpful.

5. CONCLUSION

As the DRAM industry starts to revolutionize the conventional planar DRAM design with heterogeneous 3-D stacking technology that integrates DRAM and logic on a single die package, it is also a timely moment for computer architects to contemplate about how to take advantage of these new enabling technologies for improving the DRAM architecture and the overall memory hierarchy. In this paper, we proposed a TSV-enabled, energy-efficient SRAM row cache that is tightly integrated with its corresponding 3-D DRAM array. To evaluate our proposal, we studied its feasibility from the circuit perspective as well as the architectural perspective. Our evaluation with memory intensive applications shows that well-balanced heterogeneous 3-D DRAM chips can improve system performance by 30% while saving dynamic energy by 31%, on average.

[6]The actual hit rate of a DRAM system with the open-row policy was 53%, on average.

6. REFERENCES

[1] "International Technology Roadmap for Semiconductors," 2007. [Online]. Available: http://public.itrs.net

[2] R. Baker, *CMOS: Circuit Design, Layout, and Simulation*. Wiley-IEEE Press, 2007.

[3] V. Cuppu, B. Jacob, B. Davis, and T. Mudge, "A performance comparison of contemporary DRAM architectures," in *Proceedings of the International Symposium on Computer Architecture*, 1999, p. 0222.

[4] Elpida, "Elpida Completes Development of Cu-TSV (Through Silicon Via) Multi-Layer 8-Gigabit DRAM," http://www.elpida.com/pdfs/pr/2009-08-27e.pdf.

[5] H. Hidaka, Y. Matsuda, M. Asakura, and K. Fujishima, "The Cache DRAM architecture: A DRAM with an on-chip cache memory," *IEEE MICRO,*, pp. 14–25, 1990.

[6] U. Kang, H. Chung, S. Heo, D. Park, H. Lee, J. Kim, S. Ahn, S. Cha, J. Ahn, D. Kwon *et al.*, "8 Gb 3-D DDR3 DRAM Using Through-Silicon-Via Technology," *IEEE Journal of Solid-State Circuits*, vol. 45, no. 1, pp. 111–119, 2010.

[7] M. Kawano, S. Uchiyama, Y. Egawa, N. Takahashi, Y. Kurita, K. Soejima, M. Komuro, S. Matsui, K. Shibata, J. Yamada *et al.*, "A 3D packaging technology for 4 Gbit stacked DRAM with 3 Gbps data transfer," in *International Electron Devices Meeting*, 2006, pp. 1–4.

[8] K. Kim *et al.*, "Technology for Sub-50 nm DRAM and NAND Flash Manufacturing," *IEDM Tech. Dig*, pp. 323–326, 2005.

[9] R. Koganti and G. Kedem, "WCDRAM: A Fully Associative Integrated Cached-DRAM with Wide Cache Lines," in *Proc. Fourth IEEE Workshop Architecture and Implementation of High Performance Comm. Systems*, 1997.

[10] J. Renau, B. Fraguela, J. Tuck, W. Liu, M. Prvulovic, L. Ceze, S. Sarangi, P. Sack, K. Strauss, and P. Montesinos, "SESC simulator," January 2005, http://sesc.sourceforge.net.

[11] S. Thoziyoor, N. Muralimanohar, J. Ahn, and N. Jouppi, "CACTI 5.1," *HP Laboratories, Palo Alto, Technical Report*, vol. 20, 2008.

[12] A. Udipi, N. Muralimanohar, N. Chatterjee, R. Balasubramonian, and N. Jouppi, "Rethinking DRAM Design and Organization for Energy-Constrained Multi-Cores," in *Proceedings of the International Symposium on Computer Architecture*, 2010.

[13] D. H. Woo, N. H. Seong, D. L. Lewis, and H.-H. S. Lee, "An Optimized 3D-Stacked Memory Architecture by Exploiting Excessive, High-Density TSV Bandwidth," in *Proceedings of the International Symposium on High Performance Computer Architecture*, 2010.

[14] Z. Zhang, Z. Zhu, and X. Zhang, "Cached DRAM for ILP Processor Memory Access Latency Reduction," *IEEE Micro*, vol. 21, no. 4, pp. 22–32, 2001.