Magma: A Monolithic 3D Vertical Heterogeneous ReRAM-based Main Memory Architecture

Farzaneh Zokaee
Indiana University
fzokaee@iu.edu

Mingzhe Zhang
ICT, CAS
zhangmingzhe@ict.ac.cn

Xiaochun Ye
ICT, CAS
yexiaochun@ict.ac.cn

Dongrui Fan
ICT, CAS
fandr@ict.ac.cn

Lei Jiang*
Indiana University
jiang60@iu.edu

Abstract

3D vertical ReRAM (3DV-ReRAM) emerges as one of the most promising alternatives to DRAM due to its good scalability beyond 10*nm*. Monolithic 3D (M3D) integration enables 3DV-ReRAM to improve its array area efficiency by stacking peripheral circuits underneath an array. A 3DV-ReRAM array has to be large enough to fully cover the peripheral circuits, but such large array size significantly increases its access latency. In this paper, we propose Magma, a M3D stacked heterogeneous ReRAM array architecture, for future main memory systems by stacking a large unipolar 3DV-ReRAM array on the top of a small bipolar 3DV-ReRAM array and peripheral circuits shared by two arrays. We further architect the small bipolar array as a direct-mapped cache for the main memory system. Compared to homogeneous ReRAMs, on average, Magma improves the system performance by 11.4%, reduces the system energy by 24.3% and obtains > 5-year lifetime.

CCS Concepts

• **Hardware → 3D integrated circuits**; **Memory & storage**;

Keywords

Monolithic 3D Integration, 3D Vertical ReRAM

ACM Reference Format:
Farzaneh Zokaee, Mingzhe Zhang, Xiaochun Ye, Dongrui Fan, and Lei Jiang. 2019. Magma: A Monolithic 3D Vertical Heterogeneous ReRAM-based Main Memory Architecture. In *The 56th Annual Design Automation Conference 2019 (DAC '19), June 2–6, 2019, Las Vegas, NV, USA.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3316781.3317858

1 Introduction

In deep nano-regimes, Moore's law continues by integrating more cores into a chip, e.g., Intel 72-core Xeon Phi [24]. An increasing number of cores enable more threads to run concurrently in a processor. Emerging workloads such as BigData analytics [1, 9] substantially enlarge the working set size of each thread. So a modern processor requires a high density main memory system to maintain scalable performance for all concurrent threads. However, traditional DRAM suffers from large refresh power consumption and key timing parameter (e.g., tWR [35]) degradation. Moreover, DRAM stops scaling and stays with $7F^2$ cell size [33] since 20*nm*.

To overcome the DRAM scaling crisis, prior works [6, 29, 36] adopt emerging nonvolatile memory technologies, e.g., PCM, STT-MRAM and ReRAM, to build future huge capacity memory systems. Although PCM successfully achieves the $4F^2$ cell size at 20*nm*, it suffers from large write power and write disturbances [10]. STT-MRAM enjoys DRAM-comparable access latency, but the large cell size (e.g., $50F^2$ [6]) hinders its practical deployment in terabyte-scale chips. Recent research efforts [12, 27, 29, 34] advocate building high density main memories by ReRAM. Unlike PCM and STT-MRAM, ReRAM [29] has the $4F^2$ cell size, 10^{12}-write cell endurance [7], low write power consumption and short access latency.

Besides the $4F^2$ cell size, ReRAM can use multi-level cell (MLC) [29], 3D XPoint structure [3, 11, 30] and 3D vertical array [4, 16] to further reduce its cell area in an array. As a result, peripheral circuits, e.g., sense amplifiers, decoders, and multiplexers, dominate the array area. Recently, due to its Back-End-of-Line (BEOL) compatibility, ReRAM uses the M3D stacking technology [1, 11, 13, 23, 30] to improve its array area efficiency by stacking a 3D memory array upon peripheral circuits. To cover all peripheral circuits, a 3D ReRAM array has to be large enough. However, a large capacity array significantly prolongs the access latency. Large sneak currents also tightly restrict the size of a bipolar ReRAM array.

Prior works focus on only optimizing the lifetime, access latency, power and area of ReRAM 2D arrays [12, 27, 29, 34] and 3D arrays [4, 16], but fail to consider the impact of 3D monolithically stacked peripheral circuits. No prior work identifies the 3D stacked peripheral circuits underneath an array may greatly increase the array access latency. In this paper, we propose Magma, a M3D stacked heterogeneous ReRAM array architecture, to implement scalable main memory systems with short latency, high density and long endurance. Our contributions are summarized as follows.

- We studied the trade-off between endurance, latency, power and area on M3D stacked ReRAM arrays by comprehensively exploring the design space of 3D arrays and their 3D stacked peripheral circuits. We found that the 3D ReRAM array size has to be large enough to fully cover its peripheral circuits, but the large sneak path current seriously limit the size of a bipolar ReRAM array.
- To obtain short access latency, low access power, long enough array endurance, and high area efficiency, we propose Magma, a 2-layer M3D stacking architecture, by stacking a unipolar large ReRAM array on the top of a bipolar small ReRAM array and peripheral circuits shared by two arrays.
- To leverage small arrays, we architect the small bipolar array as a direct-mapped cache for the M3D stacked ReRAM-based main memory system. Because of the large capacity, the direct-mapped cache hit rate is similar to that of a set-associative cache.

*This work was supported in part by Indiana University FRSP Award 2018.

DAC '19, June 2–6, 2019, Las Vegas, NV, USA

ACM ISBN 978-1-4503-6725-7/19/06…$15.00
https://doi.org/10.1145/3316781.3317858

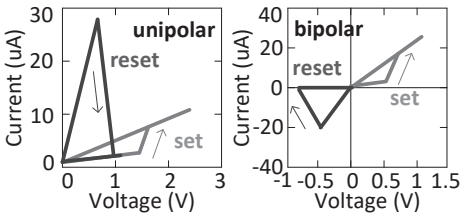

Figure 1: Unipolar & bipolar ReRAMs.

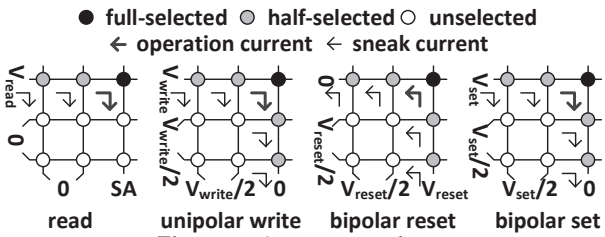

Figure 2: Array operations.

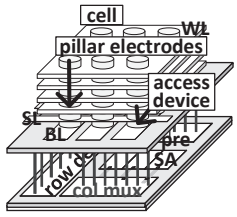

Figure 3: A 3D-VA.

However, unlike a set-associative cache, the direct-mapped cache avoids complex cache structures and extra tag lookup latency.

- We evaluated and compared our proposed techniques against various M3D stacked memory baselines. Compared to M3D stacked homogeneous ReRAMs, on average, Magma improves the system performance by 11.4%, reduces the system energy by 24.3% and obtains > 5-year lifetime. Compared to M3D stacked PCMs and STT-MRAMs, Magma averagely increases the system performance by 8%-10%.

2 Background and Related Work

2.1 ReRAM Technology

Cell. A ReRAM cell [29] records data by a thin metal-oxide (e.g., HfO_x) layer sandwiched by a top electrode and a bottom electrode. In the metal-oxide layer, a SET produces conductive filaments resulting in a low resistance state (LRS) cell indicating "1", while a RESET yields a high resistance state (HRS) cell representing "0" by rupturing conductive filaments.

Table 1: The unipolar and bipolar ReRAM comparison.

ReRAM	W energy	W latency	endurance	costs	sneak current
unipolar	10×	200ns	10^8	cheap	tiny
bipolar	1×	50ns	10^{12}	expensive	large

Switching. As Figure 1 shows, ReRAM relies on two methods of resistance switching [8] that differ by the polarity of SET and RESET. We present the comparison between unipolar and bipolar ReRAMs in Table 1. In *unipolar* switching, both SET and RESET occur under positive voltage, while the polarities of SET and RESET must be alternated in *bipolar* switching. Unipolar switching is triggered by the Joule heating acceleration of redox transitions at the basis of conductive filaments formation and rupture in the gap region, while bipolar switching is explained in terms of ionic migration assisted by the electric field. Because of its switching mechanism, a bipolar ReRAM cell requires smaller write current and voltage, thereby consuming less write energy. Bipolar switching ReRAMs exhibit better endurance [18] owing to the less material loss during bipolar writes. A bipolar switching cell can tolerate 10^{12} writes [7], while a unipolar switching cell stands for only 10^8 writes [19]. Moreover, bipolar switching [8] completes much faster than unipolar switching on a ReRAM cell. However, bipolar switching ReRAM cannot rely on a simple unidirectional diode and requires a more sophisticated selector [17] with nonlinear selectivity at both polarities, e.g., a metal-amorphous si-metal selector or a silicon NPN access device. It also requires a more complicated write control circuit.

ReRAM arrays and sneak currents. The array biasing schemes during a read and a write are shown in Figure 2. A read drives the selected word-line (WL) to V_{read} and senses current changes on the selected bit-line (BL). During a unipolar write, the selected WL and BL are driven to 0 and V_{write} respectively, while unselected WLs and BLs are set to $\frac{V_{write}}{2}$. On the contrary, during a bipolar write on a memory line, only after all RESETs finish, SETs can start [29], since their polarities must be alternated. To RESET/SET a cell in a bipolar array, the selected WL and BL are set to $0/V_{set}$ and $V_{reset}/0$ respectively. All unselected WLs and BLs are charged to $\frac{V_{reset}}{2}/\frac{V_{set}}{2}$. Besides that the operation voltage is fully applied across the *fully-selected* cells, the *half-selected* cells also confront partial voltage causing the *sneak current*. Large sneak currents significantly decrease read margins and normal write currents in an array. By turning their unidirectional diodes OFF, half-selected cells in a unipolar switching ReRAM array can keep sneak currents in check. However, compared to a unidirectional diode with > 10^8 ON/OFF current ratio, the nonlinear selectivity of the bipolar selectors is only ~ 10^5 [17]. It is difficult to use bipolar switching ReRAMs to construct a large capacity array, since the ReRAM write latency exponentially increases due to the voltage drop [27, 29, 34] introduced by sneak currents. A detailed design space exploration on building bipolar arrays by various selectors is shown in [17] highlighting no prior bipolar selector supports >4K-bit array size because of the large sneak current.

2.2 Monolithic 3D Vertical ReRAMs

3D vertical ReRAM array. There two types of 3D ReRAM array: 3D *XPoint* [11, 30] and 3D *vertical* [4, 16] arrays. Compared to a 3D XPoint array, the same capacity 3D vertical ReRAM array is more scalable, since it can stack the same number of memory layers with less peripheral overhead [3]. As Figure 3 shows, to reduce the average cell size beyond $4F^2$, a 3D vertical ReRAM (3DV-ReRAM) array [3, 4, 16] stacks multiple array layer vertically in a 3D structure. The 3DV-ReRAM cells are sandwiched between perpendicular pillar electrodes and WL layers. At the bottom of pillar electrodes, there is a 2D arrays of access devices that are connected to BLs and controlled by select-lines (SLs). With appropriate bias schemes on WLs, BLs and SLs, each cell in a 3D vertical array can be individually accessed. During a read or write, a selected SL is biased to turn ON the access devices connected to this SL while all the other access devices remain OFF by grounding the unselected SLs.

Monolithic 3D stacking. M3D stacking technology [1, 2, 20, 23, 31] emerges to overcome the integration and connectivity limitations of TSVs with monolithic integration through high density nano-scale inter-layer vias (ILVs). M3D ILVs are scalable with process technology and efficient use of 3D layers. At 14*nm*, ILV has only a 50*nm* diameter, while the diameter of a TSV is 2μm [20]. The extra power and latency introduced by a M3D ILV are insignificant [20], because of its tiny parasitic capacitance and resistance. A M3D SRAM cell [25] built by ILVs reduces the area overhead by 33% and the energy delay product by 1.6× over a 6-transistor 2D SRAM cell. Recent research efforts [1, 13, 23, 30] monolithically

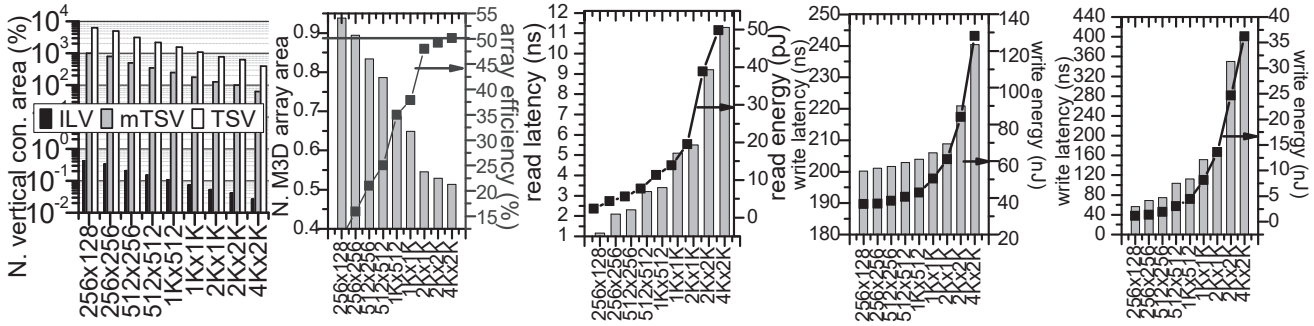

Figure 4: ILV & TSV. Figure 5: Area reduction. Figure 6: Read latency. Figure 7: Unipolar write. Figure 8: Bipolar write.

stack peripheral circuits underneath a ReRAM array to improve the array area efficiency. As Figure 3 shows, row decoders, column multiplexers, prechargers and sense amplifiers on the bottom layer are connected to a ReRAM array on the top layer by M3D ILVs. To fully cover the peripheral circuits on the bottom layer, the array on the top layer has to be large enough thereby greatly increasing access latency. Particularly, such large bipolar arrays suffer from strong sneak currents resulting in long write latencies.

2.3 Related Work

With MLC, 3D XPoint structure and 3D vertical array, the ReRAM cell area in an array greatly shrinks. As a result, the peripheral circuits of an array dominates the array area. The emerging M3D stacking technology [1, 11, 13, 23, 30] decreases the peripheral overhead by stacking a 3D ReRAM array upon its peripheral circuits. Prior works heavily optimize array endurance, access latency, power and area for both 2D [12, 27, 29, 34] and 3D [4, 16] bipolar switching ReRAM arrays. But no prior work considers the influence of M3D stacked peripheral circuits on its ReRAM array. In this paper, we identified only a large ReRAM array can fully cover its peripheral circuits in a M3D structure. However, a large array prolongs its access latency and makes bipolar switching ReRAM suffer from large sneak currents decreasing both read margin and normal write currents. Several techniques including double-sided ground biasing, multi-phase write [29], array level write scheduling [34], LRS cell reduction [27] and complementary resistive switch [12] are proposed to mitigate negative impacts of sneak current. However, it is still difficult to construct large memory arrays by bipolar switching ReRAMs, because of the weak selectivity of bidirectional access devices [17].

3 M3D Array Design Space Exploration

3.1 Low-cost M3D Stacking Technology

A 2D array [15, 28] is partitioned into multiple pieces that can be stacked vertically by 3D stacking technologies, so that the array area is greatly reduced. Although the 3D memory array was first proposed by [15, 28], traditional TSVs are too large to implement it. Unlike the original concept [15, 28] fabricating both memory cells and peripheral circuits on each layer of a 3D memory array, the latest M3D ReRAM [1, 11, 13, 23, 30] separates memory cells and peripheral circuits into two groups, and fabricates each group on one individual layer. So vertical links are required to connect an array on the top layer to decoders, multiplexers, and sense amplifiers on the bottom layer. The area overhead comparison of various 3D vertical links in 3D arrays with various capacities [1] is exhibited in Figure 4, where each bar is normalized to the array area. Each 3D array configuration has a 8-bit data path including 8 sense amplifiers and 8 write drivers. We considered three types of 3D vertical links: M3D ILVs, TSVs and mini-TSVs (mTSVs). The diameter of a mini-TSV is 40% [20] of that of a TSV. Our experimental methodology can be found in Section 5. Because of its small diameter, M3D ILVs cost only 42%-3% of the array area in various array configurations. On the contrary, the area of TSVs or mTSVs is at least 60× larger than the array area, because of their μm-level diameters. Therefore, only M3D ILVs can enable the *fine-grained array-level* 3D integration, while TSVs are used for the *coarse-grained chip/die-level* 3D integration [15, 28]. As Figure 5 shows, a M3D array reduces the area of various array configurations by 6.3%-48.7%.

3.2 Detailed Design Space Exploration

We perform the design space exploration of a 2-layer M3D array with various capacities, and report the array area efficiency in Figure 5, the read latency & energy in Figure 6 and the unipolar / bipolar write latency & energy in Figure 7 / 8. The 2-layer M3D array adopts an 8-bit data path balancing the trade-off between array activation power and array area [29]. We assume the array is built on the top layer while all peripheral circuits are fabricated on the bottom layer in a 2-layer M3D array. In Figure 5, we used the metric of area efficiency to evaluate the peripheral circuit area of each M3D array configuration. The array area efficiency is defined as $\frac{area_{cell}}{area_{cell}+area_{peri}}$, where $area_{cell}$ is the area of ReRAM cells while $area_{peri}$ is the area of peripheral circuits. If the area efficiency is 50%, the top layer array can fully cover all peripheral circuits on the bottom layer in a 2-layer M3D array. Only large M3D arrays ($> 2K \times 2K$) approaching $\sim$ 50% area efficiency can fully take advantage of the M3D stacking technology to hide the peripheral overhead. In contrast, the peripheral circuits still dominate the area of small M3D arrays, since their area efficiencies are very low. The M3D stacking technology cannot obviously reduce the peripheral overhead for these small arrays. However, both read and write latencies are significantly prolonged by long BLs and WLs in large M3D arrays. Particularly, as Figure 7 and 8 exhibit, the write latencies in both unipolar and bipolar switching ReRAM arrays increase as the array capacity enlarges. Because of sneak currents, the bipolar array write latency degrades more severely than that of a unipolar array with the same capacity.

[1] In a 3DV-ReRAM array, one WL is a layer. We use the WL length of a 3DV-ReRAM to indicate the array dimension. $L_{WL} = A \times B$, where L_{WL} is the WL length; A is the number of BLs; and B is the BL length.

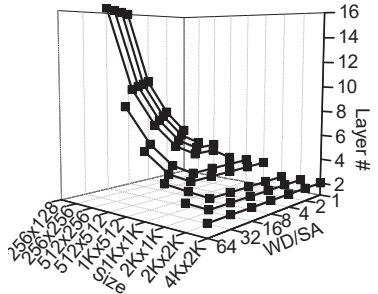

Figure 9: Many-layer peri. stacking.

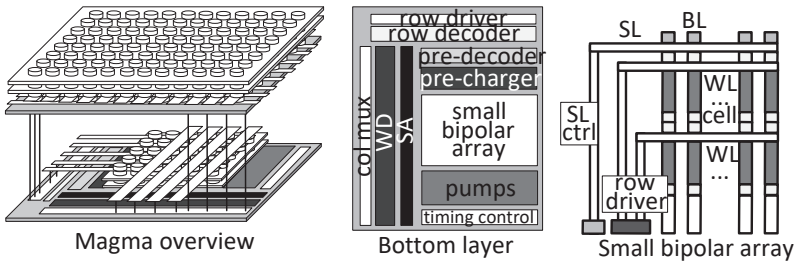

Figure 10: The Magma architecture.

3.3 M3D Many-layer Integration

To reduce the peripheral overhead in small M3D arrays, it is possible to partition the peripheral circuits into many parts and then stack them vertically in a M3D array. The number of peripheral circuit layers in such a M3D array is highlighted in Figure 9. A 256×256 array with a 8-bit data path requires 5 layers to include all peripheral circuits. A wider data path in the same array increases the M3D layer number, since more sense amplifiers and write drivers have to be integrated. The state-of-the-art M3D stacking technology [2] successfully integrates only 3 layers into a 3D structure. Therefore, we do not anticipate the 5-layer M3D stacked peripheral is a practical solution for small M3D arrays to hide the peripheral overhead.

4 Magma

We use large M3D arrays to reduce the peripheral costs and propose Magma to improve the access latency of large M3D arrays. The idea of Magma is straightforward. It uses a large unipolar 3DV-ReRAM array on the top layer and integrates a small bipolar 3DV-ReRAM array to the bottom layer containing all peripheral circuits. Magma enables two arrays to share all peripheral circuits. In Magma, the top large array provides large capacity, while the bottom small array supplies short access latency.

Overall architecture. The overall architecture of Magma is shown in Figure 10, where two layers are connected by vertical M3D ILVs in a 3D structure. On the top layer, Magma includes only a large unipolar switching ReRAM array to obtain high density and avoid large sneak currents, due to the high selectivity of unidirectional diodes [17]. We set the unipolar array dimension as $4K \times 2K$, so that it is large enough to fully cover all peripheral circuits. On the bottom layer, besides peripheral circuits, a small bipolar switching ReRAM array is integrated to reduce the access latency and energy of Magma. Based on the latency explorations in Figure 6, 7 and 8, we set the bipolar array dimension as $1K \times 1K$ WLs. There is a $100nm$ SiO_x inter-layer dielectric (ILD) [23] between two M3D layers. The height of a ReRAM cell is only $15nm$ [32], so a 4-layer vertical ReRAM array can be easily fabricated on a layer of the M3D structure. But we assume only a 2-layer vertical structure for both top unipolar and bottom bipolar arrays. Two arrays on different layers share peripheral circuits on the bottom layer including row drivers and decoders, column multiplexers, charge pumps, sense amplifiers and write drivers. To flexibly supply write voltages for both unipolar and bipolar arrays, charge pumps adopt a cascaded stage structure [11] at the cost of slightly increased chip area. The top unipolar array is connected to peripheral circuits by ILVs. By situating the bottom bipolar array close to pre-chargers, charge pumps and sense amplifiers, we can minimize the IR drop along both its BLs and WLs. The bipolar array has one eighth of the unipolar array capacity, so it uses only a half of the row decoder and one forth of the column multiplexer. Moreover, the bottom bipolar array builds SLs and BLs on its top, so that they can connect to remote row drivers and column multiplexers by vertical wires.

Area, leakage, access latency and energy. At $32nm$, in Magma the top layer occupies $0.0344mm^2$ chip area and consumes $26.4mW$ leakage power, while the bottom layer increases the area by 3% and the leakage power by 5%. A Magma array occupies $0.035mm^2$ chip area and costs $54.1mW$ leakage power. Compared to the array design composed of a $4K \times 2K$ array and a $1K \times 1K$ array having two independent 8-bit data paths, due to the peripheral circuits shared by two arrays, Magma reduces the chip area by 10% and the leakage power by 13%. When accessing only the bottom bipolar array, it is possible to power gate 50% of the row decoder and 75% of the column multiplexer to reduce power consumption. The access latency and energy values can be viewed in Figure 6, 7 and 8.

Array endurance. We use a Magma configuration composed of 4GB bottom bipolar arrays and 32GB top unipolar arrays to explain the main memory system endurance. A bipolar (unipolar) ReRAM cell can stand for 10^{12} [7] (10^8 [19]) writes. We aggressively set the bipolar (unipolar) ReRAM write latency as $150ns$ ($240ns$). Since we have 8 banks in our main memory baseline, 8 writes can happen simultaneously. A write occurs on 64-byte, but 50% of cells in a line change. We adopted Flip-N-Write [5], security fresh [22] and error correcting pointers (ECPs) [21] in our main memory baseline. We modeled the process variation by following the methodology in [21] with $\sigma = 0.1$. As a result, under constantly arriving writes, the 4GB bottom bipolar arrays last > 100 years, while the 32GB top unipolar arrays stand for 5.1 years.

Leveraging Magma. The bottom bipolar arrays can be exposed to the operating system that is able to allocate physical memory frames located in bottom bipolar arrays to critical benchmarks or even key data structures in an application. However, in this paper, we architect bottom bipolar arrays as a hardware-managed cache for top unipolar arrays. Although this approach decreases the overall available memory capacity, it reduces the software design complexity. We organize all bottom bipolar arrays as a directly mapped cache. Each cache line has 64-byte data and 8-byte tag. An access reads both the cache line and its tag, totally 72-byte data, from 72 bipolar arrays in a bank. Each bipolar array has a 8-bit data path. A cache controller on the bridge chip checks whether the access gets a hit or not. When a hit, the cache controller returns the line to the CPU; otherwise, it reads the target memory line from top unipolar arrays, returns the data to the CPU and starts a cache replacement procedure.

The thermal issue and data transfers. An access goes to a bottom bipolar array first. Only if a miss occurs, it will be re-directed to the top unipolar array. Two arrays can only be accessed sequentially, so there is no thermal problem caused by multiple writes. When a cache replacement happens, the dirty data are read from the bottom bipolar array by sense amplifiers, and then written to the top unipolar array directly. The data installations and transfers are kept within the NVDIMM without interrupting the CPU.

CPU	8 3.2GHz, OoO cores, 4-wide, 8MSHRs/core, 128-entry instruction window
I/D cache	private, I/D 32KB each/core, 4-way, LRU, 64B line, 1-cycle hit
L2 cache	private, 2MB/core, 8-way, LRU, 64B line, write back, 1-cycle tag, 5-cycle data hit
DRAM	private, on-die 3D, 32MB/core, 16-way, LRU, write back, 96-cycle hit
MC	40-entry R/W queues, MC to bank 64-cycle
Main Memory	64B line, NVDIMM, 36GB unipolar arrays, $t_{RCD} = 20ns$, $t_{WR} = 150ns$, $t_{CL} = 15ns$, 4.5GB bipolar arrays, $t_{RCD} = 27ns$, $t_{WR} = 240ns$, $t_{CL} = 15ns$

Table 2: Baseline configuration.

Benchmark	Description
Terasort	1GB input, 1GB working set, 3GB footprint
JoinQuery	2GB working set, 6.5GB footprint
Kmean	2GB input, 2.3GB working set, 7GB footprint
PageRank	0.8GB working set, 2GB footprint
SPEC2006	mcf, bwaves, lbm, libquantum, zeusmp
x_m	8 copies of x, a copy on a core
mix_1	2-bwaves,2-Kmean,2-stream,2-Terasort
mix_2	2-lbm,2-JoinQuery,2-Kmean,2-zeusmp
mix_3	2-bwaves,2-mcf,2-zeusmp,2-PageRank

Table 3: Simulated benchmark.

Design Overhead. We added a cache controller in the bridge chip. By Synopsys design compiler, we synthesized the cache control logic costing 552K transistors into $0.066mm^2$ at $32nm$. A tag comparison costs 1.7ns and 1.59pJ.

5 Experimental Methodology

Simulator. We evaluated our proposed designs by a PIN-based CPU simulator Sniper that is configured to model the CPU processor and all cache hierarchies. And we implemented Magma main memory system by NVMain.

Baseline configuration. Our baseline processor is an eight-OoO-core CPU. Each core can be operated at 3.2GHz and has a private 32MB on-die 3D stacked DRAM cache. Our memory controller (MC) prioritizes reads and schedules writes only when there is no read. Once the write queue is full, the MC issues a write burst, where all pending reads are stalled until the write queue is empty. We formulated a Magma configuration consisting of 36GB $4K \times 2K$ unipolar arrays for memory entries and ECPs, and 4.5GB $4K \times 2K$ bipolar arrays for cache entries and tags. Magma relies on a NVDIMM that has one channel, one rank per channel and eight banks per rank. A bank spreads across eight 8-bit wide chips, so eight banks share eight chips in a rank. Magma adopts Flip-N-Write [5], security fresh and error correcting pointers (ECPs) [21]. The detailed baseline configuration can be found in Table 2.

Chip and array modeling. We modeled unipolar and bipolar cells by models in [30] and [11]. We used NVsim to compute chip and array parameters in Figure 5, 6, 7 and 8. We adopted the models of ILVs, mini-TSVs and TSVs from [20]. We simulated the latency and energy overhead of ILVs through HSPICE. We applied power gating [36] on peripherals when an array is idle and used the LRS cell reduction technique [27] to mitigate sneak currents.

Simulated benchmarks. In Table 3, we chose a subset of programs from SPEC-CPU2006, STREAM and Intel BigData HiBench [9] suites to construct multi-programmed workloads covering different memory access characteristics. Most applications from HiBench cost > $2GB$ maximum physical memory (footprint) and actively use > $1GB$ physical memory (working set). We collected traces of HiBench benchmarks on a single working node.

Simulation and evaluation. The representative portions of benchmark was determined by PinPoints. We simulated 5 billion instructions to obtain performance results. For our results, we define *speedup* as: $Speedup = \frac{IPC_{tech}}{IPC_{baseline}}$, where $IPC_{baseline}$ and IPC_{tech} are the instruction number per cycle of our baseline setting and the setting with scheme $tech$, respectively.

Schemes. We implemented and compared the following schemes in our experiments.

- *B-36GB* is built by 36GB 2-layer M3D $4K \times 2K$ bipolar arrays, where the top layer includes only arrays and the bottom layer is composed of all peripheral circuits.
- *U-36GB* is the same as *B-36GB* except it is built by unipolar arrays.
- *U-X/YGB* is built by Magma arrays, where the top layer includes Y-GB $4K \times 2K$ unipolar arrays and the bottom layer consists of X-GB $1K \times 1K$ bipolar arrays and peripheral circuits.

6 Results and Analysis

Performance. The performance comparison is shown in Figure 11, where all results are normalized to *B-36GB*. Compared to *B-36GB*, *U-36GB* has shorter write latency and thus improves the performance by 10.3%. *M-2.25/36GB* improves the performance by 13.7% over *B-36GB* by adding a 2GB directly mapped cache. When enlarging this cache to 4GB (*M-4.5/36GB*), the performance improvement over *B-36GB* increases to 23%, because the 4GB cache covers almost all working sets of BigData applications including *joi_m*, *ter_m*, *kme_m* and *pag_m*. However, compared to *M-2.25/36GB*, *M-4.5/36GB* does not obviously improve the performance of SPEC2006 applications, since *M-2.25/36GB* has enough capacity to buffer their small working sets. Compared to *M-4.5/36GB*, *M-9/36GB* only slightly improves the performance by 1%. Therefore, we select *M-4.5/36GB* as our default Magma configuration. *str_m* is only sensitive to the memory bandwidth, so all Magma configurations achieve similar performance on stream.

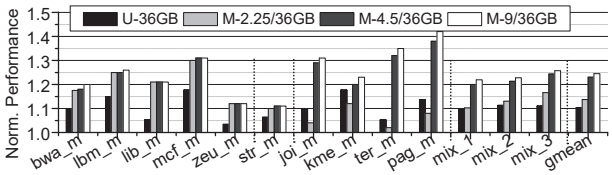

Figure 11: The performance comparison (norm. to B-36GB).

Energy. The energy comparison is exhibited in Figure 12, where all bars are normalized to *B-36GB*. Because of the power gating on peripheral circuits, the leakage energy is proportional to the memory access latency. Therefore, compared to *U-36GB*, *M-4.5/36GB* reduces the leakage energy by 10.3%. Due to small bipolar arrays, *M-4.5/36GB* further decreases the read energy by 48% and the write energy by 74.2% over *U-36GB*. In summary, *M-4.5/36GB* spends only 61.5% of the total energy used by *B-36GB* in completing the same benchmarks. Compared to *U-36GB*, *M-4.5/36GB* reduces the system energy by 24.3% averagely.

Cache schemes. We also implemented Buffered Way Predictor (BWP) [26] and on-chip cTLB [14] to architect bipolar arrays as a set-associative cache and a tag-less fully-associative cache, respectively. The performance comparison of various cache schemes is

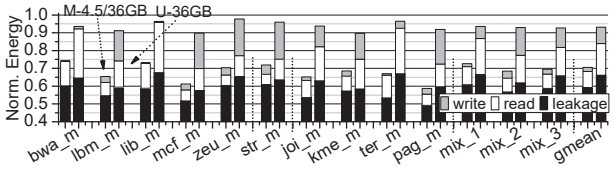

Figure 12: The energy comparison (norm. to B-36GB).

shown in Figure 13, where all bars are normalized to *B-36GB*. For a 2GB cache, BWP has the best performance, because of its high hit rate. The fully-associative cache manages the data at 4KB page granularity, so the 2GB capacity is too small for such coarse-grained cache replacement policy. For 4GB and 8GB caches, Magma and the fully-associative cache run faster, since they have no tag lookup. Compared to Magma, the fully-associative cache improves the system performance by only <3%. But it requires heavy modifications on the CPU TLB.

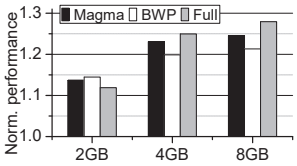

Figure 13: Cache schemes.

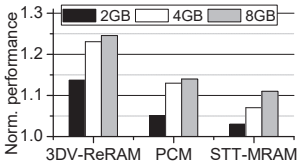

Figure 14: NVM techs

Other NVM technologies. We can build the bottom small arrays in Magma by other NVM technologies, i.e., PCM and STT-MRAM, with the same area constraint (32*nm*). The performance comparison of various NVM technologies is shown in Figure 14, where *X*GB indicates the cache configuration costing the same area of *X*GB 3DV-ReRAM and built by a NVM technology (e.g., 3DV-ReRAM, PCM and STT-MRAM). PCM has $4F^2$ cell size and can also use 3D vertical array structure, so it can achieve the same cache capacity as 3DV-ReRAM with the same area overhead. Compared to 3DV-ReRAM, PCM suffers from longer pump charging latency and larger t_{FAW} (four-bank activation window), because of its high write voltage and huge write power. As a result, the PCM write bandwidth is smaller than that of 3DV-ReRAM. Compared to PCM, 3DV-ReRAM improves the performance by at least 8% with various cache capacities. Compared to 3DV-ReRAM, STT-MRAM has much larger cell area and cannot use 3D vertical array structure. So it can obtain only 30% of the 3DV-ReRAM cache capacity with the same chip area. Although STT-MRAM enjoys much shorter memory access latency, compared to STT-MRAM, 3DV-ReRAM improves the performance by 10%-15%. Because STT-MRAM has too small cache capacity to hold the entire working sets of most simulated benchmarks.

7 Conclusion

M3D integration hides the peripheral overhead by stacking peripheral circuits underneath an array. However, to fully cover the peripheral circuits, a 3DV-ReRAM array has to have a large capacity significantly increasing the access latency. In this paper, we propose Magma, a M3D stacked heterogeneous ReRAM array architecture, by stacking a large unipolar 3DV-ReRAM array on the top of a small bipolar 3DV-ReRAM array and peripheral circuits shared by two arrays. We also architect the bipolar array as a direct-mapped cache for the main memory system. Averagely, compared to prior 3DV-ReRAM-based main memory systems, Magma improves the system performance by 11.4%, reduces the system energy by 24.3% and obtains > 5-year lifetime.

References

[1] M. M. S. Aly, *et al.*, "Energy-efficient abundant-data computing: The N3XT 1,000 x," *Computer*, 2015.
[2] F. Andrieu, *et al.*, "A review on opportunities brought by 3D-monolithic integration for CMOS device and digital circuit," in *ICICDT*, 2018.
[3] I. G. Baek, *et al.*, "Realization of vertical resistive memory (VRRAM) using cost effective 3D process," in *IEDM*, 2011.
[4] P. Chen, *et al.*, "Design Tradeoffs of Vertical RRAM-Based 3-D Cross-Point Array," *TVLSI*, 2016.
[5] S. Cho and H. Lee, "Flip-N-Write: A simple deterministic technique to improve PRAM write performance, energy and endurance," in *MICRO*, 2009.
[6] J. DeBrosse, *et al.*, "A fully-functional 90nm 8Mb STT MRAM demonstrator featuring trimmed, reference cell-based sensing," in *CICC*, 2015.
[7] C.-W. Hsu, *et al.*, "Self-rectifying bipolar TaO_x RRAM with superior endurance over 10^{12} cycles for 3D high-density storage-class memory," in *VLSIT*, 2013.
[8] D. Ielmini, "Resistive switching memories based on metal oxides: mechanisms, reliability and scaling," *Semiconductor Science and Technology*, 31(6), 2016.
[9] Intel, "HiBench: the bigdata micro benchmark suite," https://github.com/intel-hadoop/HiBench, 2015.
[10] L. Jiang, *et al.*, "Mitigating Write Disturbance in Super Dense Phase Change Memory," in *DSN*, 2014.
[11] A. Kawahara, *et al.*, "An 8 Mb Multi-Layered Cross-Point ReRAM Macro With 443 MB/s Write Throughput," *JSSC*, 2013.
[12] M. A. Lastras-Montano, *et al.*, "A low-power hybrid reconfigurable architecture for resistive random-access memories," in *HPCA*, 2016.
[13] H. D. Lee, *et al.*, "Integration of 4F2 selector-less crossbar array 2Mb ReRAM on transition metal oxides for high density memory applications," in *VLSIT*, 2012.
[14] Y. Lee, *et al.*, "A fully associative, tagless DRAM cache," in *ISCA*, 2015.
[15] G. H. Loh, *et al.*, "Processor Design in 3D Die-Stacking Tech." *IEEE Micro*, 2007.
[16] M. Mao, *et al.*, "Design and Analysis of Energy-Efficient and Reliable 3-D ReRAM Cross-Point Array System," *TVLSI*, 2018.
[17] P. Narayanan, *et al.*, "Circuit-Level Benchmarking of Access Devices for Resistive Nonvolatile Memory Arrays," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, 2016.
[18] F. Nardi, *et al.*, "Complementary switching in metal oxides: Toward diode-less crossbar RRAMs," in *IEDM*, 2011.
[19] F. Pan, *et al.*, "Recent progress in resistive random access memories: Materials, switching mechanisms, and performance," *MSE Reports*, 2014.
[20] S. K. Samal, *et al.*, "Monolithic 3D IC vs. TSV-based 3D IC in 14nm FinFET technology," in *S3S*, 2016.
[21] S. Schechter, *et al.*, "Use ECP, Not ECC, for Hard Failures in Resistive Memories," in *ISCA*, 2010.
[22] N. H. Seong, *et al.*, "Security Refresh: Prevent Malicious Wear-out and Increase Durability for Phase-change Memory with Dynamically Randomized Address Mapping," in *ISCA*, 2010.
[23] M. M. Shulaker, *et al.*, "Monolithic 3D integration of logic and memory: Carbon nanotube FETs, resistive RAM, and silicon FETs," in *IEDM*, 2014.
[24] A. Sodani, "Knights Landing (KNL): 2nd Generation Intel® Xeon Phi processor," in *IEEE Hot Chips Symposium*, 2015.
[25] S. Srinivasa, *et al.*, "A Monolithic-3D SRAM Design with Enhanced Robustness and In-Memory Computation Support," in *ISLPED*, 2018.
[26] Z. Wang, *et al.*, "Building a Low Latency, Highly Associative DRAM Cache with the Buffered Way Predictor," in *SBAC-PAD*, 2016.
[27] W. Wen, *et al.*, "Speeding up crossbar resistive memory by exploiting in-memory data patterns," in *ICCAD*, 2017.
[28] Y. Xie, *et al.*, "Design Space Exploration for 3D Architectures," *JETC*, 2006.
[29] C. Xu, *et al.*, "Overcoming the challenges of crossbar resistive memory architectures," in *HPCA*, 2015.
[30] T. y. Liu, *et al.*, "A 130.7-mm^2 2-Layer 32-Gb ReRAM Memory Device in 24-nm Technology," *JSSC*, 2014.
[31] C.-C. Yang, *et al.*, "Logic/memory hybrid 3D sequentially integrated circuit using low thermal budget laser process," in *S3S*, 2015.
[32] S. Yu, *et al.*, "3D vertical RRAM-scaling limit analysis and demonstration of 3D array operation," in *VLSIT*, 2013.
[33] W. J. Yun, *et al.*, "A digital DLL with hybrid DCC using 2-step duty error extraction and 180 phase aligner for 2.67Gb/S/pin 16Gb 4-H stack DDR4 SDRAM with TSVs," in *ISSCC*, 2015.
[34] H. Zhang, *et al.*, "Leader: Accelerating ReRAM-based main memory by leveraging access latency discrepancy in crossbar arrays," in *DATE*, 2016.
[35] X. Zhang, *et al.*, "Restore truncation for performance improvement in future DRAM systems," in *HPCA*, 2016.
[36] P. Zhou, *et al.*, "A Durable and Energy Efficient Main Memory Using Phase Change Memory Technology," in *ISCA*, 2009.