Architecting Phase Change Memory as a Scalable DRAM Alternative

Benjamin C. Lee† Engin Ipek† Onur Mutlu‡ Doug Burger†

†Computer Architecture Group
Microsoft Research
Redmond, WA
{blee, ipek, dburger}@microsoft.com

‡Computer Architecture Laboratory
Carnegie Mellon University
Pittsburgh, PA
onur@cmu.edu

ABSTRACT

Memory scaling is in jeopardy as charge storage and sensing mechanisms become less reliable for prevalent memory technologies, such as DRAM. In contrast, phase change memory (PCM) storage relies on scalable current and thermal mechanisms. To exploit PCM's scalability as a DRAM alternative, PCM must be architected to address relatively long latencies, high energy writes, and finite endurance.

We propose, crafted from a fundamental understanding of PCM technology parameters, area-neutral architectural enhancements that address these limitations and make PCM competitive with DRAM. A baseline PCM system is 1.6x slower and requires 2.2x more energy than a DRAM system. Buffer reorganizations reduce this delay and energy gap to 1.2x and 1.0x, using narrow rows to mitigate write energy and multiple rows to improve locality and write coalescing. Partial writes enhance memory endurance, providing 5.6 years of lifetime. Process scaling will further reduce PCM energy costs and improve endurance.

Categories and Subject Descriptors

B.3.3 [**Memory Structures**]: Performance Analysis and Design Aids—*Simulation*; B.6.1 [**Logic Design**]: Design Styles—*Memory control and access*

General Terms

Design

Keywords

PCM, phase change memory, DRAM alternative, scalability, performance, power, energy, endurance

1. INTRODUCTION

Memory technology scaling drives increasing density, increasing capacity, and falling price-capability ratios. Memory scaling, a first-order technology objective, is in jeopardy for conventional technologies. Storage mechanisms in prevalent memory technologies require inherently unscalable charge placement and control. In the non-volatile space, Flash memories must precisely control the discrete charge placed on a floating gate. In volatile main memory, DRAM must not only place charge in a storage capacitor but must also mitigate sub-threshold charge leakage through the access device. Capacitors must be sufficiently large to store charge for reliable sensing and transistors must be sufficiently large to exert effective control over the channel. Given these challenges, manufacturable solutions for scaling DRAM beyond 40nm are unknown [1].

Phase change memory (PCM) provides a non-volatile storage mechanism amenable to process scaling. During writes, an access transistor injects current into the storage material and thermally induces phase change, which is detected during reads. PCM, relying on analog current and thermal effects, does not require control over discrete electrons. As technologies scale and heating contact areas shrink, programming current scales linearly. This PCM scaling mechanism has been demonstrated in a 20nm device prototype and is projected to scale to 9nm [1, 23]. As a scalable DRAM alternative, PCM could provide a clear roadmap for increasing main memory density and capacity.

To realize this vision, however, we must first overcome PCM's disadvantages relative to DRAM. Access latencies, although tens of nanoseconds, are several times slower than those of DRAM. At present technology nodes, PCM writes require energy intensive current injection. Moreover, writes induce thermal expansion and contraction within the storage element, degrading injection contacts and limiting endurance to hundreds of millions of writes per cell at current processes. These limitations are significant, which is why PCM is currently positioned only as a Flash replacement; in this market, PCM properties are drastic improvements. For a DRAM alternative, however, we must architect PCM for feasibility in main memory within general-purpose systems.

Current prototype designs are not designed to mitigate PCM latencies, energy costs, and finite endurance. This paper rethinks PCM subsystem architecture to bring the technology within competitive range of DRAM. Since area translates directly into memory manufacturing cost, we ensure proposed solutions are area neutral. Drawn from a rigorous survey of PCM device and circuit prototypes published within the last five years (Section 2) and comparing against modern DRAM memory subsystems (Section 3), we examine the following:

ISCA'09, June 20–24, 2009, Austin, Texas, USA.

	Horri [11]	Ahn [2]	Bedeschi [6]	Oh [20]	Pellizer [21]	Chen [8]	Kang [12]	Bedeschi [7]	Lee [15]	Parameters [this work]
Year	2003	2004	2004	2005	2006	2006	2006	2008	2008	**
Process (nm, F)	**	120	180	120	90	**	100	90	90	90
Array Size (Mb)	**	64	8	64	**	**	256	256	512	**
Material	GST, N-d	GST, N-d	GST	GST	GST	GS, N-d	GST	GST	GST	GST,N-d
Cell Size (μm^2)	**	0.290	0.290	**	.097	60 sq-nm	0.166	0.097	0.047	0.065-0.097
Cell Size (F^2)	**	20.1	9.0	**	12.0	**	16.6	12.0	5.8	9.0-12.0
Access Device	**	**	BJT	FET	BJT	**	FET	BJT	diode	BJT
Read T (ns)	**	70	48	68	**	**	62	**	55	48
Read I (uA)	**	**	40	**	**	**	**	**	**	40
Read V (V)	**	3.0	1.0	1.8	1.6	**	1.8	**	1.8	1.0
Read P (uW)	**	**	40	**	**	**	**	**	**	40
Read E (pJ)	**	**	2.0	**	**	**	**	**	**	2.0
Set T (ns)	100	150	150	180	**	80	300	**	400	150
Set I (uA)	200	**	300	200	**	55	**	**	**	150
Set V (V)	**	**	2.0	**	**	1.25	**	**	**	1.2
Set P (uW)	**	**	300	**	**	34.4	**	**	**	90
Set E (pJ)	**	**	45	**	**	2.8	**	**	**	13.5
Reset T (ns)	50	10	40	10	**	60	50	**	50	40
Reset I (uA)	600	600	600	600	400	90	600	300	600	300
Reset V (V)	**	**	2.7	**	1.8	1.6	**	1.6	**	1.6
Reset P (uW)	**	**	1620	**	**	80.4	**	**	**	480
Reset E (pJ)	**	**	64.8	**	**	4.8	**	**	**	19.2
Write Endurance	1E+07	1E+09	1E+06	**	1E+08	1E+04	**	1E+05 (MLC)	1E+05	1E+08

Table 1: Technology Survey. ** denotes information not available in cited publication. The last column identifies parameters derived for this work.

- **Buffer Organization:** We examine PCM buffer organizations that satisfy DRAM imposed area constraints. Narrow buffers mitigate high energy PCM writes. Multiple buffer rows exploit locality to coalesce writes, hiding their latency and energy. PCM buffer reorganizations reduce application execution time from 1.6x to 1.2x and memory energy from 2.2x to 1.0x, relative to DRAM-based systems. (Section 4)
- **Partial Writes:** We propose partial writes, which track data modifications and write only modified cache lines or words to the PCM array. Using an endurance model to estimate lifetime, we expect write coalescing and partial writes to deliver an memory module average lifetime of 5.6 years. Scaling improves PCM endurance, extending lifetimes by four orders of magnitude at 32nm. (Section 5)

Collectively, these results indicate PCM is a viable DRAM alternative, with architectural solutions providing competitive performance, comparable energy, and feasible lifetimes.

2. PCM TECHNOLOGY

Given the still speculative state of PCM technology, researchers have made several different manufacturing and design decisions. We survey device and circuit prototypes published within the last five years (Table 1). From the survey, we derive conservative PCM technology parameters, which are identified in the last column.

2.1 Memory Cells

The storage element is comprised of two electrodes separated by a resistor and phase change material, which is typically a chalcogenide (Figure 1L). $Ge_2Sb_2Te_5$ (GST) is most commonly used, but other chalcogenides offer higher resistivity and improve the device's electrical characteristics. Nitrogen doping increases resistivity and lowers programming current while GS offers lower latency phase changes [8, 11]. We derive parameters for Nitrogen-doped GST, given its widespread adoption.

Phase changes are induced by injecting current into the resistor-chalcogenide junction and heating the chalcogenide to 650 °C. Current and voltage characteristics of the chalcogenide are identical regardless of its initial phase, which lowers programming complexity and latency [14]. The amplitude and width of the injected current pulse determines the programmed state.

Phase change memory cells are 1T/1R devices, comprised of the resistive storage element and an access transistor (Figure 1C). Access is typically controlled by one of three devices: field-effect transistor (FET), bipolar junction transistor (BJT), or diode. In future, FET scaling and large voltage drops across the cell will adversely affect gate oxide reliability for unselected wordlines [22]. BJTs are faster and expected to scale more robustly without this vulnerability [7, 22]. Diodes occupy smaller areas and potentially enable greater cell densities, but require higher operating voltages [15]. We derive parameters with BJT access devices, given their balance between speed and scalability.

2.2 Writes

Phase change memory typically operates in two states. The SET and RESET states are defined as the crystalline (low-resistance) and amorphous (high-resistance) phases of the chalcogenide, respectively. Illustrated in Figure 1R, the storage element is RESET by a high, short current pulse. The short pulse abruptly discontinues current flow, quickly quenching the heat generation and freezing the chalcogenide into the amorphous state. In contrast, the storage element is SET by a moderate, long current pulse, which ramps down over the duration of the write. The ramp down gradually cools the chalcogenide and induces crystal growth.

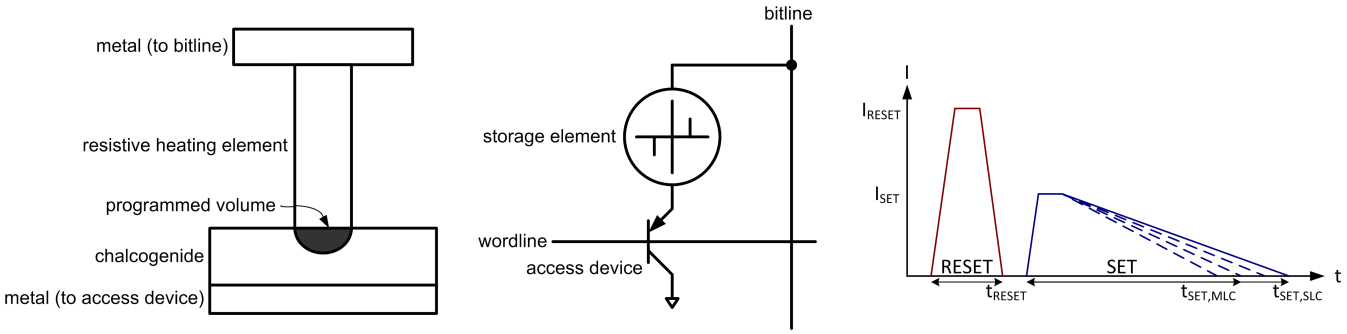

Figure 1: Phase Change Memory. Storage element with heating resistor and chalcogenide between electrodes (L). Cell structure with storage element and BJT access device (C). RESET to an amorphous (high resistance) state with a high, short current pulse. SET to a crystalline (low resistance) state with moderate, long current pulse. Slope of SET current ramp down determines the state in multi-level cells (R).

As the longer of the two, SET latency determines write performance. We derive a SET latency of 150ns as separately demonstrated by Ahn et al. and Bedeschi et al. [2, 6]. We extrapolate across process generations to identify a SET current and voltage of 150 μA and 1.2V.[1] We compute average power by integrating current over time and multiplying by voltage. A SET dissipates 90 μW for 150ns, consuming approximately 13.5pJ.

RESET latency is a determinant of write energy. We derive a RESET latency of 40ns as demonstrated by Bedeschi et al. [6]. We further determine RESET requires 300 μA at 1.6V by extrapolating across process generations using current scaling rules (Section 2.5, [6, 7]). RESET dissipates 480 μW for 40ns and consumes approximately 19.2 pJ.

We derive conservative write latencies and currents although other prototypes demonstrate more aggressive parameters. Shorter SET latencies of 80 and 100ns are demonstrated for emerging cell technologies and not for array prototypes [8, 11]. Longer SET latencies of 180 to 400ns arise from a choice of dense but slow access devices [12, 15, 20]. Chen et al, demonstrates a 90 μA RESET, which uses a new, speculative phase change material [8].

Although most PCM prototypes consider only two states per storage element (*i.e.*, crystalline and amorphous) to produce single-level cells (SLC), recent research demonstrates additional intermediate states, which enables multi-level cells [7, 19]. Multi-level cells (MLC) store multiple bits by programming the cell to produce intermediate resistances. Smaller current slopes (*i.e.*, slow ramp down) produce lower resistances and larger slopes (*i.e.*, fast ramp down) produce higher resistances. Varying slopes induce partial phase transitions and/or change the size and shape of the amorphous material produced at the contact area, giving rise to resistances between those observed from fully amorphous or fully crystalline chalcogenide. The difficulty of differentiating between a large number of resistances typically constrains MLC cells to two bits per cell.

2.3 Write Endurance

Writing is the primary wear mechanism in phase change memory. When injecting current into a volume of phase change material, thermal expansion and contraction degrades the electrode-storage contact, such that programming currents are no longer reliably injected into the cell. Since material resistivity is highly dependent on current injection, current variability causes resistance variability. This greater variability degrades the read window, the difference between programmed minimum and maximum resistance.

Write endurance, the number of writes performed before the cell cannot be programmed reliably, ranges from 1E+04 to 1E+09. Write endurance depends on manufacturing techniques and differs across manufacturers. 1E+04 is likely an outlier specific to a speculative, new cell technology [8]. 1E+05, the low end of surveyed endurance, is comparable to Flash endurance [13]. However, PCM is more likely to exhibit greater write endurance by several orders of magnitude (*e.g.*, 1E+07 to 1E+08). The ITRS roadmap projects improved endurance of 1E+12 writes at 32nm [1], but we conservatively model 1E+08. We propose differential writes (Section 5), which can combine with previously proposed techniques for Flash memories [10, 13], so that write limits are not exposed to the system during a memory's lifetime.

2.4 Reads

Prior to reading the cell, the bitline is precharged to the read voltage. The wordline is active low when using a BJT access transistor. If a selected cell is in a crystalline state, the bitline is discharged with current flowing through the storage element and access transistor. Otherwise, the cell is in an amorphous state, preventing or limiting bitline current.

We derive a cell read latency of 48ns as demonstrated by Bedeschi et al. [6]. This latency includes bitline precharge and assumes BJT access and current sensing. This same prototype requires 40 μA of read current at 1.0V. In this implementation, a cell read dissipates 40 μW for 48ns, consuming approximately 2pJ of energy. Other prototypes demonstrate higher read latencies, which range from 55 to 70ns. However, these other prototypes implement FET or diode access devices, which produce slower response times.

2.5 Process Scaling

PCM scaling reduces required programming current injected via the electrode-storage contact. As the contact area decreases with feature size, thermal resistivity increases and the volume of phase change material that must be melted to

[1]Bedeschi et al, produce two prototypes at 180 and 90nm, illustrating linear write current and voltage scaling [6, 7]. We apply this scaling to obtain 90nm SET parameters from those at 180nm.

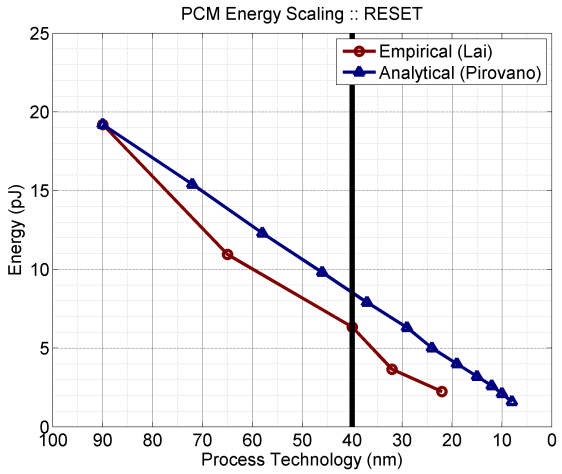

Figure 2: PCM RESET energy scaling. Lai examines prototypes [14]. Pirovano et al. examines scaling rules [22]. PCM is projected to scale to 9nm, while DRAM is projected to scale to 40nm [1].

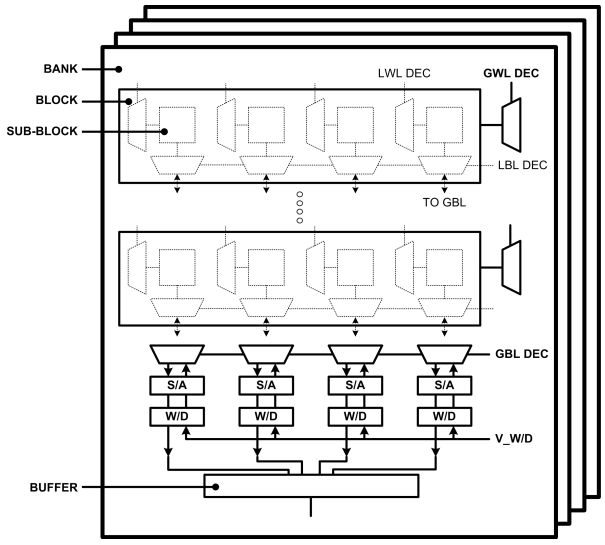

Figure 3: Array Architecture. A hierarchical memory organization includes banks, blocks, and sub-blocks with local, global decoding for row, column addresses. Sense amplifiers (S/A's) and word drivers (W/D's) are multiplexed across blocks.

completely block current flow decreases. These effects enable smaller access devices for current injection. As shown for energy in Figure 2, Pirovano et al., outline PCM scaling rules [22], which are confirmed empirically in a survey by Lai [14]. Specifically, as feature size scales down by k, contact area decreases quadratically ($1/k^2$). Reduced contact area causes resistivity to increase linearly (k), which causes programming current to decrease linearly ($1/k$).

These trends assume SET/RESET voltage does not scale [22]. SET current is typically 40 to 80 percent of RESET current and these currents scale together [23]. Process scaling does not impact read and write latencies. Write latencies, in particular, are determined primarily by the phase change material [8].

Operational issues arise with aggressive PCM technology scaling. As contact area decreases, lateral thermal coupling may cause programming currents for one cell to influence the states of adjacent cells. Lai's survey of the state of PCM finds these effects negligible in measurement and simulation [14]. Temperatures fall exponentially with distance from programmed cell, suggesting no appreciable impact from thermal coupling. Increasing resistivity from smaller contact areas may reduce signal strength (*i.e.*, smaller resistivity difference between crystalline and amorphous states). However, these signal strengths are well within the sense circuit capabilities of modern memory architectures [14].

2.6 Array Architecture

As shown in Figure 3, phase change memory array structures are similar to those for existing memory technologies. PCM cells might be hierarchically organized into banks, blocks, and sub-blocks. Row and column addresses are often decoded at local sub-blocks. Peripheral circuitry, such as sense amplifiers and write drivers are shared among blocks. Despite similarities to conventional memory array architectures, PCM-specific design issues must be addressed.

Choice of bitline sense amplifiers affect the read access time of the array. Voltage sense amplifiers are cross-coupled inverters which require differential discharging of bitline capacitances. In contrast, current sense amplifiers rely on current differences to create a differential voltage at the amplifier's output nodes. Although current sensing is faster since it does not discharge bitline parasitic capacitances, these amplifier circuits are larger [25]. We use current sensing to derive 48ns PCM reads for this work. These larger sense amplifiers affect PCM area (Section 4).

Within this memory architecture, a row is activated by reading it from the array and latching it in a buffer. Memory accesses read data from and write data to the buffer. Accesses that require an unbuffered row must evict the current row and read the desired row. Destructive DRAM reads require array writes during every eviction to restore buffered data. In contrast, PCM reads are non-destructive and array writes are required to update the array only when evicting dirty buffer contents.

In DRAM, sense amplifiers both sense and buffer data using cross-coupled inverters. In contrast, we explore PCM architectures with separate sensing and buffering; sense amplifiers drive banks of explicit latches. These latches provide greater flexibility in row buffer organization by enabling multiple buffered rows. However, these latches incur area overheads, which affect PCM area (Section 4).

Separate sensing and buffering enables multiplexed sense amplifiers. Local wordline decoders activate lines across multiple sub-blocks. A subset of these sub-blocks' data pass through local and global bitline decoders for sensing and buffering. This distributed bitline decode enables buffer widths narrower than the total number of bitlines. Buffer width is a critical design parameter, determining the required number of expensive current sense amplifiers.

3. BASELINE PCM/DRAM COMPARISON

We express PCM device and circuit characteristics within conventional DDR timing and energy parameters, thereby quantifying PCM in the context of more familiar DRAM parameters while facilitating a direct comparison.

	PCM	DRAM
Delay & Timing		
tRCD (cy)	22	5
tCL (cy)	5	5
tWL (cy)	4	4
tCCD (cy)	4	4
tWTR (cy)	3	3
tWR (cy)	6	6
tRTP (cy)	3	3
tRP (cy)	60	5
tRRDact (cy)	2	3
tRRDpre (cy)	11	3
Energy		
Array read (pJ/bit)	2.47	1.17
Array write (pJ/bit)	16.82	0.39
Buffer read (pJ/bit)	0.93	0.93
Buffer write (pJ/bit)	1.02	1.02
Background power (pJ/bit)	0.08	0.08

Table 2: Memory subsystem parameters.

3.1 Experimental Methodology

We evaluate a four-core chip multiprocessor using the SESC simulator [24]. The 4-way superscalar, out-of-order cores operate at 4.0GHz. This datapath is supported by 32KB, direct-mapped instruction and 32KB, 4-way data L1 caches, which may be accessed in 2 to 3 cycles. A 4MB, 8-way L2 cache with 64B lines is shared between the four cores and may be accessed in 32 cycles.

Below the caches is a 400 MHz SDRAM memory subsystem modeled after Micron's DDR2-800 technical specifications [16]. We consider one channel, one rank, four x16 chips per rank to achieve the standard 8B interface. Internally, each chip is organized into four banks to facilitate throughput as data are interleaved across banks and accessed in parallel. We model a burst length of eight blocks. The memory controller has a 64-entry transaction queue.

We consider parallel workloads from the SPLASH-2 suite (fft, radix, ocean), SPEC OpenMP suite (art, equake, swim), and NAS parallel benchmarks (cg, is, mg) [3, 4, 27]. Each application is simulated to completion. Regarding input sets, we use 1M points for FFT, 514x514 grid for ocean, and 2M integers for radix. SPEC OpenMP workloads run MinneSpec-Large data set and NAS parallel benchmarks run with Class A problem sizes. All applications are compiled using gcc and Fortran compilers at the `O3` optimization level. Particular applications in each benchmark suite are chosen for their memory intensity. We did not consider a benchmark if system performance or energy was not impacted by replacing DRAM with PCM.

Delay and Timing. DDR defines its command interface with a series of timing constraints, which dictate when a command can issue. In Table 2, DRAM timing parameters are provided by Micron specifications [16] and analogous PCM timing parameters are derived from Table 1.

- `tRCD` specifies the delay between an array read and buffer read/write command. This parameter is determined by the 60ns array read latency, which includes 48ns read (Table 1) and 7.5ns row decode [15]. At 400MHz, `tRCD` for PCM is 22 cycles, 4.4x greater than the DRAM value of 5 cycles.

- `tCL`, `tWL`, `tCCD`, and `tWTR` constrain consecutive buffer commands and are independent of memory cell technology. `tWR`, `tRTP` specify the delay between buffer read/write commands and an array write of that buffered data. `tWR`, `tRTP` ensure data stability in the cross-coupled inverters that feed array write drivers and are independent of memory cell technology.

- `tRP` specifies the delay between an array write and a following array read. Since an array read proceeds only after previously buffered data is successfully written back to the array, `tRP` quantifies array write latency. The longer SET delay of 150ns determines PCM write latency (Table 1) and `tRP` is 60 cycles at 400MHz.

- `tRRDact`, `tRRDpre` specify constraints on the frequency of PCM array accesses to meet power budgets. The parameters distinguish between array read (`tRRDact`) and write (`tRRDpre`) since a read is non-destructive and a write is required only when a read evicts dirty buffer contents. Furthermore, given asymmetric read and write energy costs, no single timing constraint can satisfy both read and write power budgets. PCM read energy and delay is 2.1x and 4.4x greater than that of DRAM. Because power is energy divided by delay, a PCM read dissipates 0.47x the power of DRAM reads, which produces a `tRRDact` of 2 cycles (0.47x of 3 cycle `tRRD` in DRAM). Similarly, `tRRDpre` is 11 cycles for PCM writes.

Thus, we estimate PCM read, write latencies are approximately 4.4x, 12.0x greater than those for DRAM. PCM array reads may occur 2.1x more frequently and array writes must occur 3.6x less frequently than those for DRAM.

Energy. DRAM energy costs are calculated according to Micron technical notes and specifications [17]. However, these notes do not explicitly differentiate read and write energy since writes must follow every destructive DRAM read. A current diagram in the technical note indicates array read current is much greater than array write current. From this current diagram, we derive array write and read energy costs, which are 25 and 75 percent of the total 1.56pJ per DRAM bit.

From Table 1, PCM array reads consume 2.0 pJ of energy per bit. Furthermore, we use CACTI to estimate energy consumed by peripheral circuitry to obtain approximately 2.5pJ of total array read energy per bit [18]. Array writes consume 13.5pJ or 19.2pJ when writing a zero or one. On average, zeros and ones are equally likely and writes require 16.35pJ of energy in addition to 0.53pJ peripheral circuit energy. Thus, we derive PCM array read and write energies, which are 2.1x and 43.1x greater than those for DRAM.

Reads and writes to buffered data will consume similar energy costs for PCM and DRAM since mechanisms for buffer access are independent of memory technology. Although various power modes exist for DRAM, in practice, we observe only one power mode while an application executes; there are no opportunities to enter low power modes during computation. This mode consumes 0.08pJ per buffered bit per memory cycle while clocks are enabled and memory is ready to receive commands [17]. This background energy is consumed by peripheral circuitry common to both PCM and DRAM.

3.2 Evaluation: Baseline

We consider a PCM baseline architecture, which implements DRAM-style buffering with a single 2048B-wide buffer. Figure 4L illustrates end-to-end application performance when

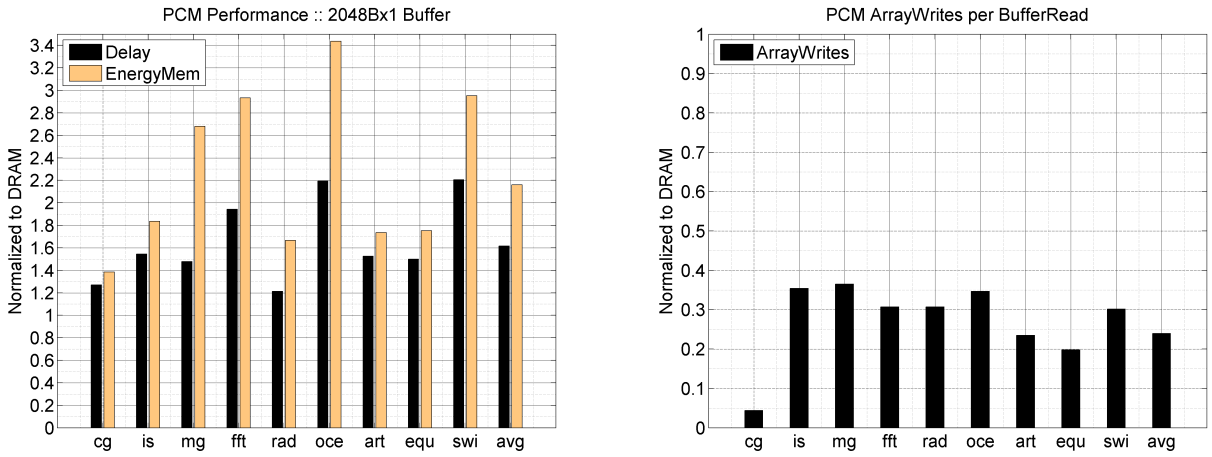

Figure 4: Application delay and energy when using PCM as a DRAM replacement (L). With non-destructive PCM reads, only a fraction of reads first require a write for dirty data evicted from buffer (R).

PCM replaces DRAM as main memory. Application delay increases with penalties relative to DRAM ranging from 1.2x (radix) to 2.2x (ocean, swim). On average, we observe a 1.6x delay penalty. The energy penalties are larger, ranging from 1.4x (cg) to 3.4x (ocean), due to the highly expensive array writes required when buffer contents are evicted. On average, we observe a 2.2x energy penalty.

The end-to-end delay and energy penalties are more modest than the underlying technology parameters might suggest. Even memory intensive workloads mix memory accesses with computation. Furthermore, the long latency, high energy array writes manifest themselves much less often in PCM than in DRAM; non-destructive PCM reads do not require subsequent writes whereas destructive DRAM reads do. Figure 4R indicates only 28 percent of PCM array reads first require an array write of a dirty buffer.

To enable PCM for use below the lowest level processor cache in general-purpose systems, we must close the delay and energy gap between PCM and DRAM. Figure 4 indicates non-destructive PCM reads help mitigate underlying delay and energy disadvantages by default. We seek to eliminate the remaining PCM-DRAM differences with architectural solutions. In particular, the baseline analysis considers a single 2048B-wide buffer per bank. Such wide buffering is inexpensive in DRAM, but incurs unnecessary energy costs in PCM given the expensive current injection required when writing buffer contents back into the array.

4. BUFFER ORGANIZATION

We examine whether PCM subsystems can close the gap with DRAM application performance and memory subsystem energy using area-neutral buffer reorganizations. To be a viable DRAM alternative, buffer organizations must hide long PCM latencies, while minimizing PCM energy costs. Effective organizations would also mitigate PCM wear mechanisms (Section 5).

To achieve area neutrality, we consider narrower buffers and additional buffer rows. The number of sense amplifiers decreases linearly with buffer width, significantly reducing area as fewer of these large circuits are required. We utilize this area by implementing multiple rows with latches much smaller than the removed sense amplifiers. Narrow widths reduce PCM write energy but negatively impact spatial locality, opportunities for write coalescing, and application performance. However, these penalties may be mitigated by the additional buffer rows. We examine these fundamental trade-offs by identifying designs that meet a DRAM-imposed area budget before optimizing delay and energy.

We consider buffer widths ranging from the original 2048B to 64B, which is the line size of the lowest level cache. We consider buffer rows ranging from the original single row to a maximum of 32 rows. At present, we consider a fully associative buffer and full associativity likely becomes intractable beyond 32 rows. Buffers with multiple rows use a least recently used (LRU) eviction policy implemented in the memory controller.

4.1 Area and Density Analysis

We estimate the net area impact of every buffer organization, considering only array components affected by the organization. Specifically, we consider the bitline peripheral circuitry (e.g., sense amplifiers), additional decoding required by multiple buffer rows, and explicit latches. We neglect area of components unrelated to the buffer, such as wordline decode.

Table 3 summarizes the parameters in our area model, which this section details. We consider area in units of transistors (T) converted to square feature sizes (F^2), which makes our analysis independent of process technology.[2] We estimate the number of transistors in each circuit and estimate transistor density using guidelines from Weste and Harris [26]. These guidelines differentiate between dense datapath circuits (e.g., 250 λ^2/T) and sparse control circuits (e.g., 1000 λ^2/T). Datapaths operate on multi-bit data words and perform the same function for every bit. As a result, they consist of identical circuits repeated in one dimension. In contrast, control circuits have less structure and, therefore, lower density.

DRAM Area Model. $8F^2$ DRAM cells provide a sufficiently wide pitch to enable a folded bitline architecture,

[2]In a 90nm process, feature size F is 90nm and layout design $\lambda = F/2$ is 45nm. For example, a 14T sense amplifer implemented with a density of $250\lambda^2$/T occupies $3500\lambda^2$ area. Converting to F^2, each amplifier occupies $875F^2$ area.

		PCM	DRAM
Array			
A	bank size (MB)	16	16
C	cell size (F^2)	9MLC, 12MLC	6
Periphery			
S	sense amplifer (T @ $250\lambda^2$/T)	44	14
	sense amplifer (F^2)	2750	875
L	latch (T @ $250\lambda^2$/T)	8	0
	latch (F^2)	500	0
D	decode 2-AND (T @ $1000\lambda^2$/T)	6	0
	decode 2-AND (F^2)	250	0
Buffer Organization			
W	buffer width (B)	64::2x::2048	2048
R	buffer rows (ea)	1::2x::32	1

Table 3: Area parameters where T refers to transistor counts, λ refers to layout density, and F^2 refers to square feature sizes. Sense amplifier transistor count from Sinha et al. [25]. Transistor densities from Weste and Harris, Table 1.10 [26]. Notation i::j::k describes range of i to k in steps of j.

which is resilient against bitline noise during voltage sensing. However, since manufacturers often choose the density of $6F^2$ DRAM cells, we derive models using $6F^2$ as a conservative point of reference that favors DRAM. The narrow pitch in $6F^2$ designs preclude folded bitlines, increasing vulnerability to noise and requiring unconventional array designs. For example, Samsung's $6F^2$ implements array blocks with 320 wordlines, which is not a power of two, to improve reliability [9].

$$\hat{A}_D = \underbrace{A{\cdot}C_D}_{\texttt{array}} + \underbrace{W_D{\cdot}S_D}_{\texttt{sense}} \quad (1)$$

Equation (1) estimates array and sense amplifier area for the baseline DRAM organization. $\hat{A}_D$ defines the area budget and PCM buffer organizations exceeding this budget are not considered. We consider a conservative DRAM cell size of $C_D = 6F^2$. Given the engineering effort required to optimize $6F^2$ DRAM arrays, we expect a buffer reorganization for DRAM to be prohibitively expensive. Thus, the buffer width W_D and rows R_D are fixed at 2048B and 1; $R_D = 1$ does not impact $\hat{A}_D$ and does not appear in Equation (1). This derivation uses a $S_D = 14T$ voltage sense amplifier, which consists of cross-coupled inverters with supporting transistors for precharge and equalization [25].

PCM Area Model. As surveyed in Table 1, PCM cells occupy between 6-20F^2. Part of this spread is due to differences in design and fabrication expertise for the new technology. However, we also observe a correlation between cell size and access device (e.g., the $6F^2$ cell uses the relatively small diode). In contrast, we favor larger BJTs for their low access times. Cells with BJTs occupy between 9-12F^2. For 9-12F^2 PCM to be as dense as $6F^2$ DRAM, multi-level cells are necessary. Such cells have an effective density of 4.5-6.0F^2 per bit, which we use for PCM cell area C_P.

Equation (2) estimates PCM area $\hat{A}_P$ from array, sense, latch, and decode contributions given buffer widths W_P and rows R_P. We model a $S_P = 44T$ MLC current sense amplifier, which is larger than the $S_D = 14T$ voltage sense amplifier in the DRAM model [25]. A larger 22T current amplifier is necessary for low read times in PCM and two-bit MLC requires two amplifiers for the least and most significant bits [5]. Amplifiers feed explicit L_P=8T latches, which are two inverters in series (4T) accessed via two pass gates (4T) that select the input from either the sense amplifier or the previously latched value. Both amplifiers and latches are regular datapath circuitry, capable of high transistor density (e.g., 250 λ^2/T).

$$\hat{A}_P = \underbrace{A{\cdot}C_P}_{\texttt{array}} + \underbrace{W_P{\cdot}S_P}_{\texttt{sense}} + \underbrace{R_P{\cdot}W_P{\cdot}L_P}_{\texttt{latch}} + \underbrace{R_P{\cdot}G(log_2R,2){\cdot}D_P}_{\texttt{decode}} \quad (2)$$

A buffer with multiple rows requires a decoder to direct sense amplifier outputs to the correct row. A buffer with R_p rows requires R_p control signals each generated by an AND gate with log_2R inputs. Avoiding slow, large fan-in gates, we compute the number of gates in an equivalent 2-AND implementation. $G(n,k)$ computes the number of k-AND gates required to implement a single n-AND gate. Each 2-AND occupies $D_p = 6T$ area, arising from a 4T NAND and a 2T inverter. The low density of decoders arise from its relatively irregular tree structure.

This area analysis favors multiple narrow buffers. We illustrate with a qualitative example, which is supported by our detailed equations. Suppose widths are reduced by a factor of R and the number of rows is increased by a factor of R. The width reduction means expensive sense amplifier area ($2750F^2$ each) is reduced by a factor of R while inexpensive latches ($500F^2$ each) are increased by a factor of R. Given negligible decoder overhead for modest R, such buffer reorganizations produce a net reduction in buffer area. However, we find these buffer reorganizations significantly impact the degree of exploited memory access locality and the costs of array writes.

4.2 Buffer Design Space

Figure 5 illustrates the delay and energy characteristics of the buffer design space for representative benchmarks. The triangles illustrate PCM and DRAM baselines, which implement a single 2048B buffer. Circles illustrate various buffer organizations. Open circles indicating organizations that require less area than the DRAM baseline when using $12F^2$ cells. Closed circles indicate additional designs that become viable when considering smaller $9F^2$ cells. By default, the PCM baseline (green triangle) does not satisfy the area budget due to larger current sense amplifiers and explicit latches.

Figure 5L illustrates the delay and energy trends for the ocean benchmark. Reorganizing a single, wide buffer into multiple, narrow buffers reduces both energy costs and delay. Examining the Pareto frontier, we observe Pareto optima shift PCM delay and energy into the neighborhood of the DRAM baseline. Furthermore, among these Pareto optima, we observe a knee that minimizes both energy and delay. For ocean, this organization is four 512B-wide buffers. Such an organization reduces the PCM delay, energy disadvantages from 2.2x, 3.4x to more modest 1.2x, 1.05x.

These observations generalize to the average across all benchmarks illustrated in Figure 5R. Although smaller $9F^2$ PCM cells provide opportunities for wider buffers and additional rows, the associated energy costs are not justified. In general, diminishing marginal reductions in delay suggest

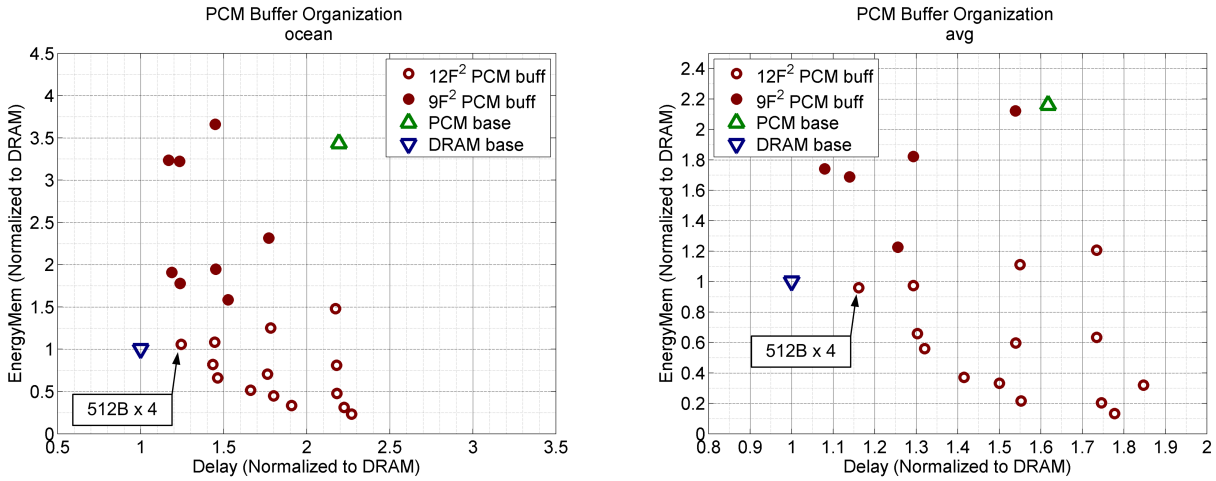

Figure 5: Pareto analysis for ocean (L) and benchmark average (R). Open circles indicate designs satisfying area constraints assuming $12F^2$ PCM multi-level cells. Closed circles indicate additional designs satisfying area constraints assuming smaller $9F^2$ PCM multi-level cells.

area savings from $9F^2$ cells should go toward improving density, not additional buffering.

Figure 6 illustrates memory subsystem effects from reorganized buffers averaged across all workloads. Although our design space considers R_p up to 32 rows, we illustrate trends up to $R_p = 4$ since we observe diminishing marginal delay and energy effects beyond $R_p = 4$. Figure 6UL illustrates the number of array reads, which increases very slowly as buffer width decreases exponentially from 2048B to 64B. For a single row, a 32x reduction in buffer width produces only a 2x increase in array reads, suggesting very little spatial locality within wide rows for the memory intensive workloads we consider. The single row is evicted too quickly after its first access, limiting opportunities for spatial reuse. However, we do observe significant temporal adaptivity. A 2048B-wide buffer with two rows requires 0.4x the array reads as a 2048B-wide buffer with only a single row.

Figure 6UR illustrates increasing opportunities for write coalescing, quantified by the number of array writes per buffer write. As the number of rows in a 2048B-wide buffer increases from one to two and four rows, array writes per buffer write falls by 0.51x and 0.32x, respectively; the buffers coalesce 49 and 68 percent of memory writes. Coalescing opportunities fall as buffer widths narrow beyond 256B. Regarding 64B-wide buffers, since we use 64B lines in the lowest level cache, there are no coalescing opportunities from spatial locality within a row buffered for a write. Increasing the number of 64B rows has no impact since additional rows exploit temporal locality, but any temporal locality in writes are already exploited by coalescing in the lowest level cache.

Figure 6LL illustrates delay trends. The delay trends confirm a lack of spatial locality within a row; row widths may be narrowed with little impact on application performance. Increasing the number of rows exploits temporal locality with great effect. The sensitivity of buffer width for $R_p > 1$ suggests a temporal component to spatial reuse. If we only buffer one row, that row is likely evicted by a buffer conflict before any spatial reuse. However, if we buffer multiple rows, buffer conflicts decrease, opportunities for spatial reuse increase, and buffer width becomes more important.

Figure 6LR illustrates energy trends. We observe sub-linear energy growth as we increase the number of rows. For example, energy costs for one and two rows are comparable. Although two rows consume twice the background energy, the second row significantly reduces the number of array reads/writes (Figure 6UL) and the effectiveness of write coalescing in the buffer (Figure 6UR). These effects on array accesses reduce dynamic energy much more rapidly than any increase in background energy. We observe near-linear energy reductions as buffer width narrows since array read and write energy is directly proportional to width.

4.3 Evaluation: Buffer Organization

Optimizing average delay and energy across the workloads, we find four 512B-wide buffers most effective. Figure 7L illustrates the impact of reorganized PCM buffers. Delay penalties are reduced from the original 1.60x to 1.16x. The delay impact ranges from 0.88x (swim) to 1.56x (fft) relative to a DRAM-based system. Executing on effectively buffered PCM, more than half the benchmarks achieve within 5 percent of their DRAM performance. Benchmarks that perform less effectively exhibit low write coalescing rates. For example, buffers cannot coalesce any writes in the fft workload.

Buffering and write coalescing also reduces memory subsystem energy from the original 2.2x of Figure 4L to 1.0x parity with DRAM. Although each PCM array write requires 43.1x more energy than a DRAM array write, these energy costs are mitigated by narrow buffer widths and additional rows, which reduce the granularity of buffer evictions and expose opportunities for write coalescing, respectively.

Thus, we demonstrate area-neutral buffering that mitigates fundamental PCM constraints and provide competitive performance and energy characteristics relative to DRAM-based systems. Narrow buffers mitigate high energy PCM writes and multiple rows to exploit locality. This locality not only improves performance, but also reduces energy by exposing additional opportunities for write coalescing. We evaluate PCM buffering using technology parameters at 90nm. As PCM technology matures, baseline PCM latencies may improve. Moreover, process technology scaling will drive linear reductions in PCM energy.

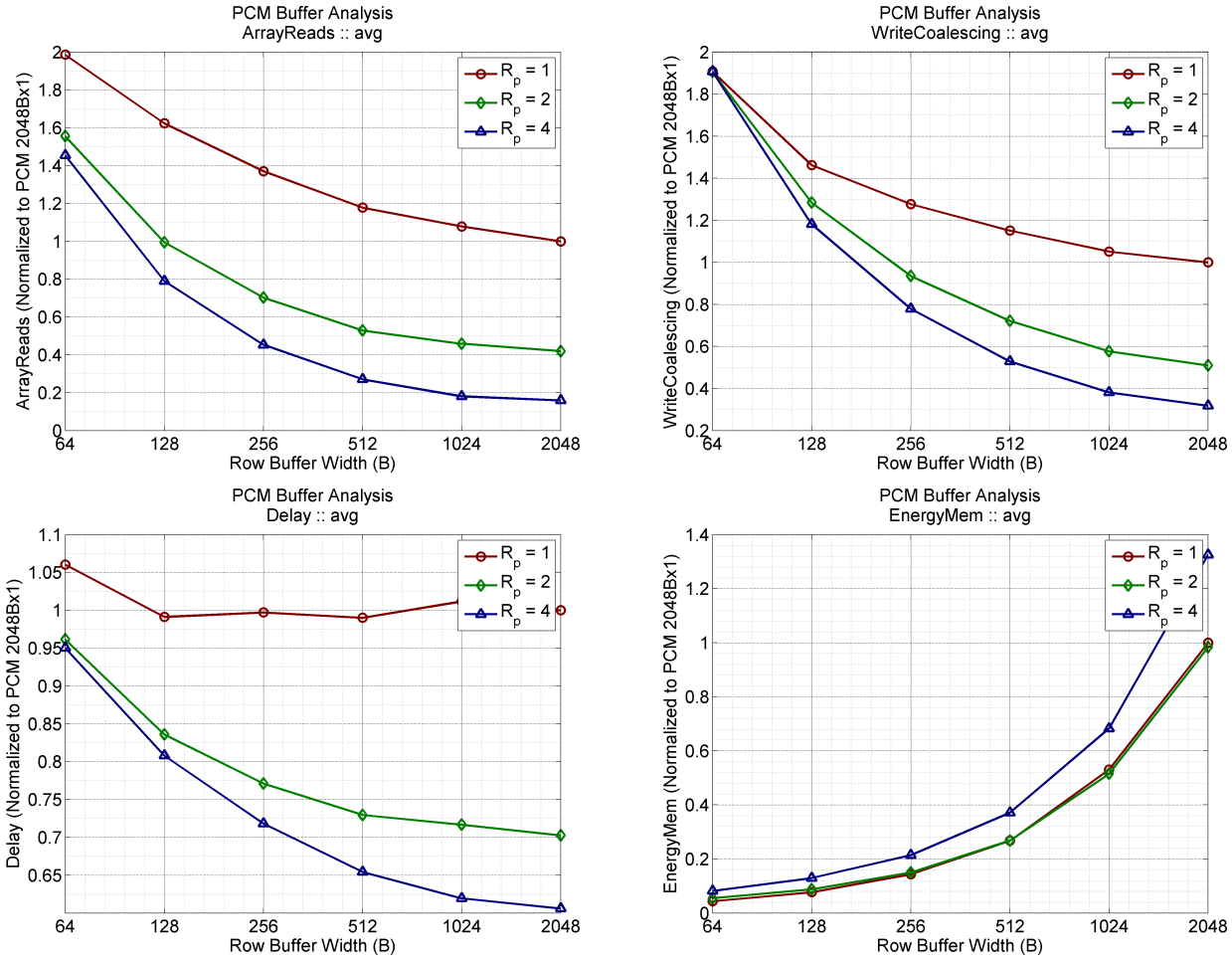

Figure 6: Memory subsystem effects and trends from PCM buffer organizations. Array reads increase sub-linearly buffer width (UL), while array write coalescing opportunities are greater with additional rows (UR). Diminishing marginal reductions in delay (LL), energy (LR) are observed at a width of 512B.

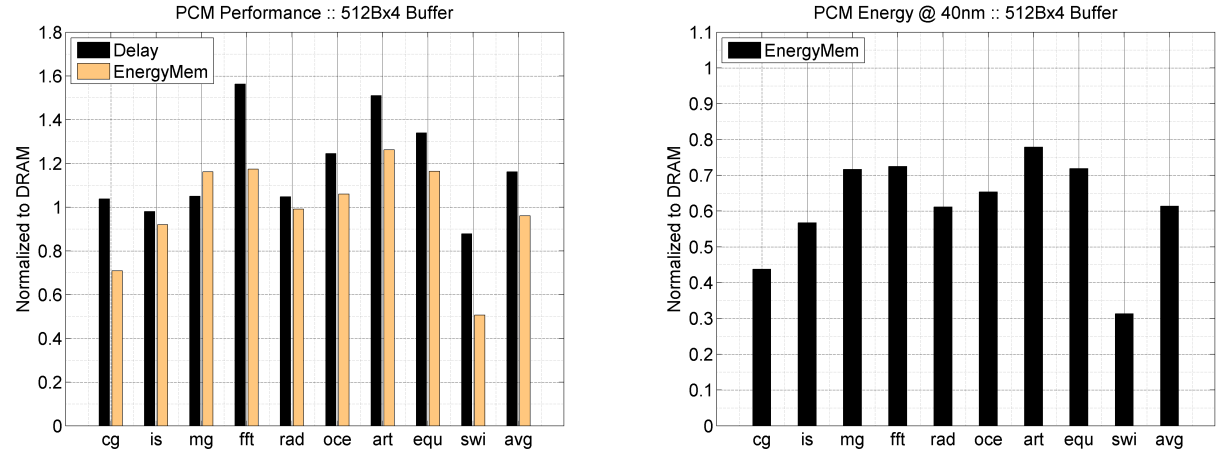

Figure 7: Application delay and energy when using PCM with optimized buffering as a DRAM replacement (L). Memory subsystem energy projections for 40nm (R).

4.4 Evaluation: Scaling Comparison

DRAM scaling faces many significant technical challenges as scaling attacks weaknesses in both components of the one transistor, one capacitor (1T1C) cell. Capacitor scaling is constrained by the DRAM storage mechanism, which requires maintaining charge on a capacitor. In future, process scaling is constrained by manufacturing small capacitors that store sufficient charge for reliably sensing despite large parasitic capacitances on the bitline.

The scaling scenarios are also bleak for the access transistor. As this transistor scales down, increasing sub-threshold leakage will make it increasingly difficult to ensure DRAM retention times. Not only is less charge stored in the capacitor, that charge is stored less reliably. These trends impact the reliability and energy efficiency of DRAM in future process technologies. According to ITRS, "manufacturable solutions are not known" for DRAM beyond 40nm [1].

In contrast, ITRS projects PCM scaling mechanisms will extend to 9nm, after which other scaling mechanisms might apply [1]. PCM scaling mechanisms have already been demonstrated up to 20nm with a novel device structure fabricated by Raoux [23]. Although both DRAM and PCM are expected to be viable at 40nm technologies, energy scaling trends strongly favor PCM. Figure 2 projects a 2.4x reduction in PCM energy from 80 to 40nm. In contrast, ITRS projects DRAM energy falls by only 1.5x at 40nm [1], which reflects the technical challenges of DRAM scaling.

Since PCM energy scales down 1.6x faster than DRAM energy, PCM subsystems significantly outperforms DRAM subsystems at 40nm. Figure 7R indicates PCM subsystem energy is 61.3 percent that of DRAM averaged across workloads. Switching from DRAM to PCM reduces energy costs by at least 22.1 percent (art) and by as much as 68.7 percent (swim). Note this analysis does not account for refresh energy, which would further increase DRAM energy costs. Although ITRS projects constant retention time of 64ms as DRAM scales to 40nm [1], less effective access transistor control may reduce retention times. If retention times fall, DRAM refresh energy will increase as a fraction of total DRAM energy costs.

5. PARTIAL WRITES

In addition to architecting PCM to offer competitive delay and energy characteristics relative to DRAM, we must also consider PCM wear mechanisms. To mitigate these effects, we propose partial writes, which reduce the number of writes to the PCM array by tracking dirty data from the L1 cache to the memory banks. This architectural solution adds a modest amount of cache state to drastically reduce the number of bits written. We derive an analytical model to estimate memory module lifetime from a combination of fundamental PCM technology parameters and measured application characteristics. Partial writes, combined with an effective buffer organization, increases memory module lifetimes to a degree that makes PCM below the lowest level processor cache feasible.

5.1 Mechanism

Partial writes track data modifications, propagating this information from the L1 cache down to the buffers at the memory banks. When a buffered row is evicted and writes content to the PCM array, only modified data is written. We consider partial writes at two granularities: lowest level cache line size (64B) and word size (4B).

These granularities are least invasive since dirty words are tracked by store instructions from the microprocessor pipeline. In contrast, bit-level granularity requires knowledge of previous data values and expensive comparators. We analyze a conservative implementation of partial writes, which does not exploit cases where stores write the same data values already stored. Detecting such cases would require comparators.

Partial writes are supported by adding state to each cache line, tracking stores using fine-grained dirty bits. At the dirty line granularity, 64B modifications are tracked beginning at the lowest level cache and requires only 1b per 64B L2 line. At the dirty word granularity, 4B modifications are tracked beginning at the L1 cache with 8b per 32B L1 line and propagated to the L2 cache, which requires 16b per 64B L2 line. Overheads are 0.2 percent and 3.1 percent of each cache line when tracking dirty lines and words, respectively.

When the L2 cache issues writebacks to memory, it must communicate its state to the memory controller and across the memory bus. Delay overheads for transmitting 16b of state is no more than one cycle in a DDR interface. Latched at the addressed bank, this state controls pass gates placed before write drivers. Latched state for 64B and 4B partial writes require $W_P R_P/64$ and $W_P R_P/4$ latches where W_P is buffer width and R_P is the number of buffered rows. The cost of pass gates for word drivers is $W_P \cdot 2T$.

Our buffer reorganizations achieve a net area savings, which accommodate these overheads. We reduce large sense amplifiers ($2750F^2$ each) by a factor of R and increase the number small latches ($500F^2$ each) by a factor of R. Partial writes require a 3.1 percent latch overhead, increasing the effective area cost for every latched bit ($525F^2$ each). However, these overheads are dwarfed by area reductions from using narrower buffers and eliminating large sense amplifiers.

5.2 Endurance

Equation (3) estimates the lifetime of a memory module driven with access patterns observed in our memory intensive workloads. Table 4 summarizes the model parameters. The model estimates the number of writes per second $\hat{W}$ for any given bit. We first estimate memory bus occupancy, which has a theoretical peak command bandwidth of $f_m \cdot (B/2)^{-1}$. Each command requires $B/2$ bus cycles to transmit its burst length B in a DDR interface, which prevents commands from issuing at memory bus speeds f_m. We then scale this peak bandwidth by application-specific utilization. Utilization is computed by measuring the number of memory operations $N_w + N_r$ and calculating the processor cycles spent on these operations $(B/2) \cdot M_f$. The processor is M_f faster than f_m. The time spent on memory operations is divided by total execution time T.

$$\hat{W} = \underbrace{\frac{f_m}{B/2} \cdot \frac{(N_w + N_r) \cdot (B/2) \cdot M_f}{T}}_{\texttt{memBusOcc}} \times \underbrace{\frac{N_w}{N_w + N_r}}_{\texttt{writeIntensity}} \times \underbrace{8W_P \cdot \left(\frac{N_{wa}}{N_{wb}}\right) \cdot \delta}_{\texttt{bufferOrg}} \times \underbrace{\frac{1}{C/2}}_{\texttt{capacity}} \quad (3)$$

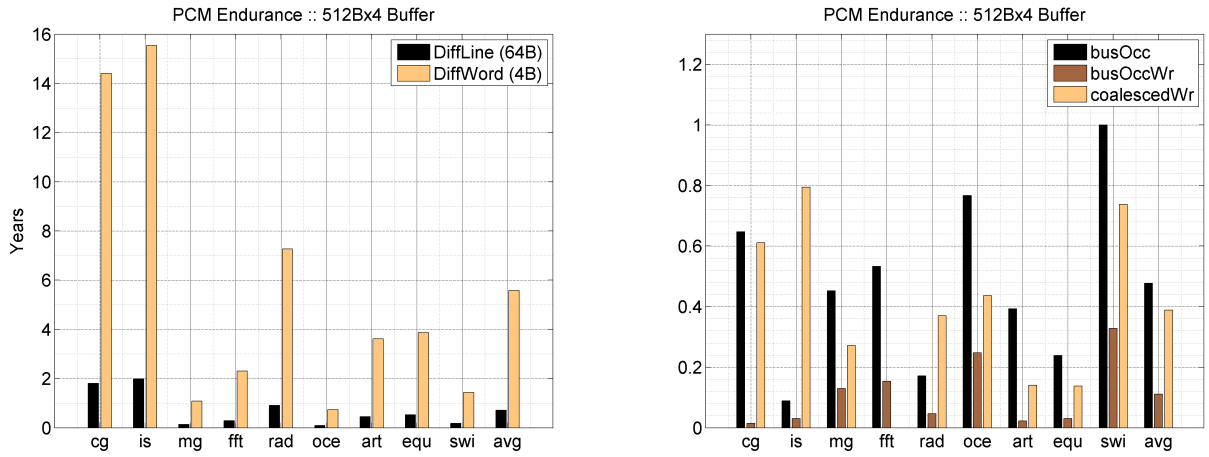

Figure 8: PCM endurance with effective buffer organization (four 512B-wide buffers) and partial writes.

Endurance		
$\hat{W}$	writes per second per bit	calc
$\hat{L}$	memory module lifetime (s)	calc
E	write endurance	1E+08
Memory Module		
C	logical capacity (Gb)	2
Memory Bus Bandwidth		
f_m	memory bus frequency (MHz)	400
M_f	processor frequency multiplier	10
B	burst length (blocks)	8
Application Characteristics		
N_w, N_r	number of writes, reads	sim
T	execution time (cy)	sim
Buffer Characteristics		
W_P, R_P	buffer width (B), rows	512, 4
N_{wb}, N_{wa}	buffer, array writes	sim
δ	fraction of buffer written to array	sim

Table 4: Endurance model parameters with simulated application-specific characteristics.

Since only a fraction of memory bus activity reaches the PCM to induce wear, we scale occupancy by write intensity to estimate the number of write operations arriving at the row buffers. In the worst case, the entire buffer must be written to the array. However, not all buffer writes cause array writes due to coalescing. N_{wa}/N_{wb} measures the coalescing effectiveness of the buffer, which filters writes to the array. Lastly, partial writes mean only the dirty fraction δ of a buffer's $8W_P$ bits are written to the array. Assuming effective wear-leveling as done in Flash [13], writes will be spread across the $C/2$ physical bits in the module, which is half the logical bits in two-bit multi-level PCM. Given $\hat{W}$ writes per second, a bit will fail in $\hat{L} = E/\hat{W}$ seconds, where E is the characterized endurance of PCM cells.

5.3 Evaluation: Partial Writes

In a baseline architecture with a single 2048B-wide buffer, average module lifetime is approximately 525 hours as calculated by Equation (3). For our memory intensive workloads, we observe 32.8 percent memory bus utilization. Scaling by application-specific write intensity, we find 6.9 percent of memory bus cycles are utilized by writes. At the memory banks, the single 2048B buffer provides limited opportunities for write coalescing, eliminating only 2.3 percent of writes emerging from the memory bus. Frequent row replacements in the single buffer limit opportunities for coalescing.

Figure 8 indicates significant endurance gains from reorganized buffers and partial writes. 64B and 4B partial writes improve endurance to 0.7 and 5.6 years, respectively. On average, the four 512B-wide buffers coalesce 38.9 percent of writes emerging from the memory bus, which is 47.0 percent utilized. Writes alone utilize 11.0 percent of the bus. Buffers use partial writes so that only a fraction of the buffer's bits is written to the array. As shown in Figure 9, only 59.3 and 7.6 percent of the buffer must be written to the array for 64B and 4B partial writes.

Considering only memory intensive workloads, this analysis is conservative. PCM subsystems would more likely experience a mix of compute and memory intensive workloads. Expected lifetimes would be higher had we considered, for example, single-threaded SPEC integer workloads. However, such workloads are less relevant for a study of memory subsystems. Moreover, within memory intensive workloads, we would expect to see a mix of read and write intensive applications, which may further increase lifetimes.

Endurance might be further improved by using single-level instead of multi-level cells. This would improve the expected lifetime since more physical bits deliver the same number of logical bits. Endurance would also benefit from more fine-grained partial writes. Partial bit writes would require additional overheads for shadow buffers to track previous data values and comparators to determine the difference, improving endurance at the cost of density.

Scalability is projected to improve PCM endurance from the current 1E+08 writes per bit to 1E+12 writes per bit at 32nm with known manufacturable solutions [1]. This higher endurance increases lifetime by four orders of magnitude in our models. ITRS also anticipates 1E+15 PCM writes at 22nm although the source of these projections is less clear since manufacturable solutions are currently unknown.

6. CONCLUSION

We provide a rigorous survey and derivation of phase change memory properties to drive architectural studies and enhancements. Architecturally relevant parameters are expressed within a DDR framework to facilitate a DRAM com-

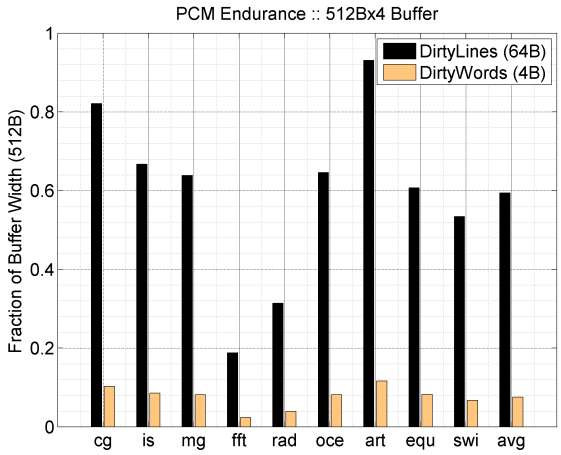

Figure 9: Partial writes and fraction of dirty data in four 512B-wide buffers.

parison. This comparison found that PCM's long latencies, high energy, and finite endurance can be effectively mitigated. Effective buffer organizations and partial writes make PCM competitive with DRAM at current technology nodes. Moreover, these complexity effective solutions are area neutral, a critical constraint in memory manufacturing.

The proposed memory architecture lays the foundation for exploiting PCM scalability and non-volatility in main memory. PCM scalability implies lower main memory energy and greater write endurance. Furthermore, non-volatile main memories will fundamentally change the landscape of computing. Software cognizant of this newly provided persistance can provide qualitatively new capabilities. For example, system boot/hibernate will be perceived as instantaneous; application checkpointing will be inexpensive; file systems will provide stronger safety guarantees. Thus, the analysis in this work is a step towards a fundamentally new memory hierarchy with deep implications across the hardware-software interface.

7. REFERENCES

[1] Process integration, devices & structures. *International Technology Roadmap for Semiconductors*, 2007.

[2] S. Ahn et al. Highly manufacturable high density phase change memory of 64Mb and beyond. In *International Electron Devices Meeting*, 2004.

[3] V. Aslot and R. Eigenmann. Quantitative performance analysis of the SPEC OMPM2001 benchmarks. *Scientific Programming*, 11(2), 2003.

[4] D. Bailey et al. NAS parallel benchmarks. In *Technical Report RNR-94-007, NASA Ames Research Center*, March 1994.

[5] M. Bauer et al. A multilevel-cell 32mb flash memory. In *International Solid-State Circuits Conference*, 1995.

[6] F. Bedeschi et al. An 8Mb demonstrator for high-density 1.8V phase-change memories. In *Symposium on VLSI Circuits*, 2004.

[7] F. Bedeschi et al. A multi-level-cell bipolar-selected phase-change memory. In *International Solid-State Circuits Conference*, 2008.

[8] Y. Chen et al. Ultra-thin phase-change bridge memory device using GeSb. In *International Electron Devices Meeting*, 2006.

[9] Y. Choi. Under the hood: DRAM architectures: 8F2 vs. 6F2. *EE Times*, February 2008.

[10] R. Hamming. Error detecting and error correcting codes. *Bell System Technical Journal*, 29(2), April 1950.

[11] H. Horii et al. A novel cell technology using N-doped GeSbTe films for phase change RAM. In *Symposium on VLSI Technology*, 2003.

[12] S. Kang et al. A 0.1um 1.8V 256Mb 66MHz synchronous burst PRAM. In *International Solid-State Circuits Conference*, 2006.

[13] T. Kgil and T. Mudge. FlashCache: A NAND flash memory file cache for low power web servers. In *International Conference on Compilers on Compilers, Architecture, and Synthesis for Embedded Systems*, October 2006.

[14] S. Lai. Current status of the phase change memory and its future. In *International Electron Devices Meeting*, 2003.

[15] K.-J. Lee et al. A 90nm 1.8V 512Mb diode-switch PRAM with 266 MB/s read throughput. *Journal of Solid-State Circuits*, 43(1), January 2008.

[16] Micron. 512Mb DDR2 SDRAM component data sheet: MT47H128M4B6-25. In *www.micron.com*, March 2006.

[17] Micron. Technical note TN-47-04: Calculating memory system power for DDR2. In *www.micron.com*, June 2006.

[18] N. Muralimanohar et al. Optimizing NUCA organizations and wiring alternatives for large caches with CACTI 6.0. In *International Symposium on Microarchitecture*, December 2007.

[19] T. Nirschl et al. Write strategies for 2 and 4-bit multi-level phase-change memory. In *International Electron Devices Meeting*, 2008.

[20] H. Oh et al. Enhanced write performance of a 64mb phase-change random access memory. In *International Solid-State Circuits Conference*, 2005.

[21] F. Pellizzer et al. A 90nm phase change memory technology for stand-alone non-volatile memory applications. In *Symposium on VLSI Circuits*, 2006.

[22] A. Pirovano et al. Scaling analysis of phase-change memory technology. In *International Electron Devices Meeting*, 2003.

[23] S. Raoux et al. Phase-change random access memory: A scalable technology. *IBM Journal of Research and Development*, 52(4/5), Jul/Sept 2008.

[24] J. Renau et al. SESC simulator. In *http://sesc.sourceforge.net*, 2005.

[25] M. Sinha et al. High-performance and low-voltage sense-amplifier techniques for sub-90nm sram. In *International Systems-on-Chip Conference*, 2003.

[26] N. Weste and D. Harris. *CMOS VLSI Design*. Pearson Education, 3 edition, 2005.

[27] S. Woo et al. The SPLASH-2 programs: Characterization and methodological considerations. In *International Symposium on Computer Architecture*, June 1995.