Evaluating STT-RAM as an Energy-Efficient Main Memory Alternative

Archana kumari,Pooja singh,Lakshay sachdeva
aydav.yadav667@gmail.com, pooja7.ps@gmail.com,sachdevlaksh@gmail.com

Abstract—In this paper, we use the possibility of STT-RAM technology to entirely replace DRAM in main memory. Our goal is to make STT-RAM performance comparable to DRAM while providing significant power savings. Towards this goal, we first analyze the performance and energy of STT-RAM, and then identify key optimizations that can be employed to improve its characteristics. Specifically, using partial write and row buffer write bypass, we show that STT-RAM main memory performance and energy can be significantly improved. Our experiments indicate that an optimized, like capacity STT-RAM main memory can provide performance comparable to DRAM main memory, with an usual 60% reduction in main memory energy.

I. INTRODUCTION

The memory wall problem continues to outbreak the design, implementation, and performance of computer systems. As the increasing degree of on-chip multiprogramming puts more pressure on the memory system, main memory serves a critical role lying between the processing cores and peripheral storage devices that have several orders of magnitude of higher latencies compared to DRAM. Consequently, there is continuing demand for DRAM capacity in order to maintain low page miss rates while serving ever-increasing frequency of requests within acceptable latencies. This has resulted in memory power itself becoming a significant contributor to overall system power. Several studies [2], [7], [11], [13], , [30] have shown that main memory now accounts for as much as 30% of overall system power and is a large contributor to operational cost. When memory power becomes such a large concern, one must inevitably start considering alternative technologies that can potentially reduce the total cost of ownership of the system. Towards this goal, solutions that exploit trade-offs between operating and acquisition costs can be employed. Specifically, for memory systems, a technology that has not been considered as a main memory replacement due to its higher acquisition cost can have a significantly better operational cost to reach a lower total cost of ownership.Scalable main memory system [5], [30], [49], [52], [66], [64]. However, it is both much slower (about 2-4X read, 10-100X write) and much more power hungry (about 2-4X read, 10-50X write), compared to DRAM [30], [50], [55], [61]. In addition, a PCRAM cell another competing technology that has also come under much scrutiny recently [8], [15], [23], [54]. STT-RAM does not necessarily have a density benefit over DRAM. While its read performance (latency and energy) is comparable to that of DRAM, its write performance (latency and energy) is worse (1.25-2X in latency, 5-10X in energy [9], [33]) than that of DRAM. However, STTRAM has two major advantages over DRAM: non-volatility and decoupled sensing and buffering. When compared to PCRAM, STT-RAM has much better read/write performance and energy characteristics as well as much better write endurance. Yet, STT-RAM technology has so far only been explored as an SRAM substitute for on-chip caches [27], [53], [56], [57], [62], but has not been considered as a candidate for main memory. In this paper, we ask (and give a positive architectural answer to) the question: Can STT-RAM be used to completely replace DRAM main memory? We set out to make STTRAM main memory performance comparable to DRAM main memory while providing substantial power savings. If we can achieve this goal, then the power savings can allow us to reach much lower total cost of ownership for equal capacity memory and even enable us to boost main memory capacities and several higher number of requests without suffering from a system wide power increase. Towards this goal, this paper starts with a detailed analysis of the performance and energy consumption of several workloads (both single-threaded and multi-programmed workloads) using various memory technologies. Specifically: We give a detailed breakdown of the DRAM power consumption of several applications, Similarly, we analyze the STT-RAM power consumption of these applications, assuming STT-RAM is main memory. We show that the energy and performance of STT-RAM, without any optimizations, is not competitive with DRAM. An in-depth examination of these results leads to two key observations: (i) actual data to be updated in a memory row constitutes only a small fraction of the row, and (ii) row buffer locality of reads is higher than that of writes. This analysis leads us to two optimizations in STT-RAM operation – tracking dirty blocks within rows for partial writes, and writes bypassing the row buffer. We find that these improvements substantially improve STT-RAM characteristics as main memory. In particular ,we show that :An STT-RAM main memory can achieve performance comparable to DRAM main memory, and An STT-RAM main memory can bring 60% reduction in average memory subsystem energy over DRAM main memory.

II. DRAM Organization

In this section, we discuss DRAM, the state-of-the-art main memory technology, along with its basic operations and the peripheral circuitry needed to perform these operations. DRAM operation is described in more detail in [25], [28], [31], [45]. A. How DRAM Fits in the Overall Memory System In this work, we assume a modern computing system with a state-of-the-art two-level private cache hierarchy. An L2 cache miss is sent to one of multiple on-chip memory controllers. Each memory controller is responsible for a separate memory channel which has one or more memory modules (DIMMs) connected to it. Each DIMM has a number of DRAM chips that are accessed in parallel to have a high bitwidth.

B. Memory Chip Organization

A DRAM chip consists of multiple banks that can be accessed independently in parallel. The only restriction in parallel access of banks is that all banks share the external data /address /command buses, so requests to different banks must be scheduled not to cause any conflicts on these buses. A DRAM bank, shown in Figure 1, comprises an array of storage cells, organized as rows and columns, row/column selection logic, sense amplifiers, and read/write latch and drivers [25]. An address provided to a memory bank consists of two parts: a row address and a column address. The row/column addresses and the memory burst length uniquely identify a particular cache block sized data in the array.

C. DRAM Operations

There are four fundamental DRAM operations that are responsible for a significant fraction of the access latency and the dynamic power consumption in DRAM. These operations , triggered by the memory controller through the memory bus, are: row activate (ACT), precharge (PRE), row buffer read (RD), and row buffer write (WR). An ACT operation enables access to a row in the memory array and connects this row with the sense amplifiers in the DRAM peripherals. After sensing, coupled inverters in the DRAM sense amplifiers retain the received data, serving as a row buffer (RB). All read/write operations to DRAM must be performed from the row buffer, and therefore, require an ACT operation to be performed before them. RD and WR operations operate in a block granularity and are served from the row buffer. Which block in the row buffer will be read or written is controlled by the column decoder. A RD operation selects a block in the row buffer and copies its data to the read latch which later transmits the data serially over the DQ pins of the memory chip in a number of bursts. A WR operation receives data from the DQ pins of the memory chip and uses write drivers to overwrite one block in the row buffer. As long as a row is active in the row buffer, the row buffer (i.e., sense amplifiers) remains connected to the row in the memory array. Therefore, a WR operation updates the data stored in the row buffer and the row in the array, simultaneously. For each memory channel, there is a corresponding memory controller that is responsible for deciding which queued request will be scheduled next for each bank. If the next request that is scheduled by the memory controller for a bank is a part of the currently active row, then the request can be directly served using the row buffer (a.k.a., a row buffer hit). When a request to read/write data from/to a row other than the currently active row occurs (a.k.a., row buffer conflict), the row in the array and the bitlines must be disconnected (by turning the access transistor off) and the bitlines must be reset to the sensing voltage (Vcc=2) before the other row can be activated. This operation of preparing the bitlines for activation of another row is called PRE. Since a row buffer conflict involves precharging one row and activating another, its latency is larger than the latency of a row buffer hit. Figure 2 illustrates the timing of two consecutive read operations received by one DRAM bank, These reads access different rows, namely, rows A and B, in the array. To retrieve row A from DRAM, first, the bitlines are precharged so that the array will be ready for activation. The first activate command starts the sensing process for row A. Towards the end of sensing, a column read command is issued so that the column access can start immediately when row sensing is finished. After the column access latency, a burst of data will be available at the output pins. As the next request in this bank is to a different row (i.e., row B), there is a row buffer conflict and an additional precharge time is needed before proceeding. After this delay, the array is ready for activating row B. In addition to the four basic DRAM operations explained above, DRAM also has a refresh operation due to its volatile nature. Figure 1 also illustrates the DRAM cell storage capacitor and the access transistor. The charge stored in this capacitor slowly leaks through the access transistor and, if not refreshed, gets lost over time. The refresh operation in DRAM is performed at a row granularity by reading one row at a time into the row buffer (i.e., activation) which restores the degraded voltage stored in the cell capacitors. A typical refresh period for today's DRAM memory chips is 64ms. For a detailed description of DRAM refresh, we refer the reader to [36]

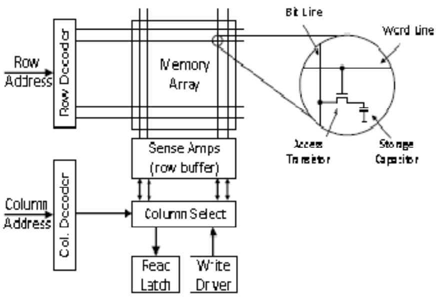


Fig. 1. DRAM bank organization.

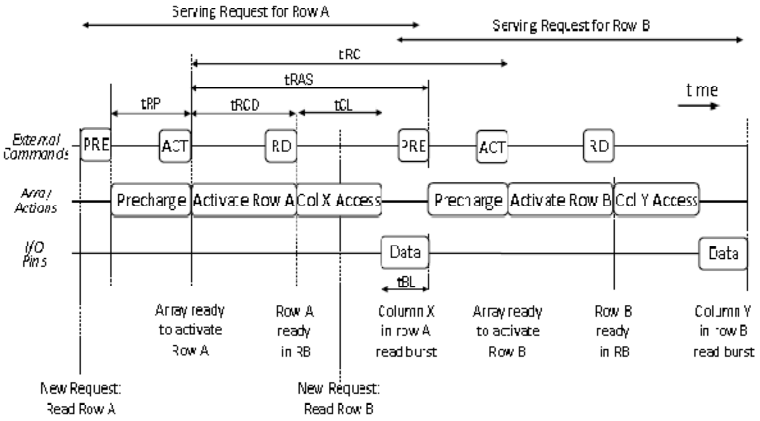


Fig. 2. Timing of two read accesses to distinct rows of a DRAM bank (drawn not to scale) [25], [31].

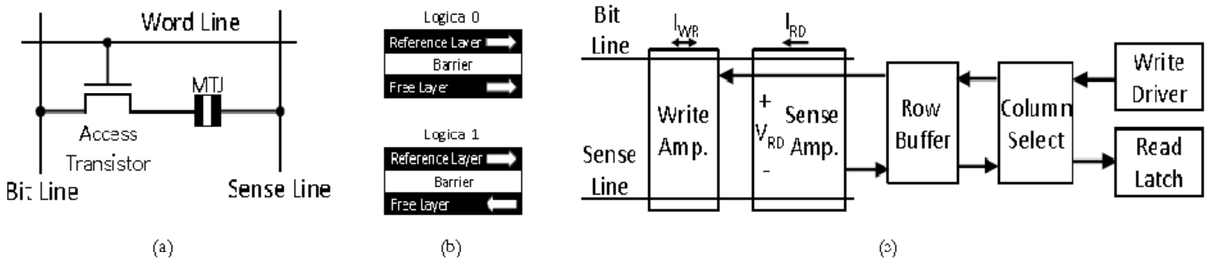


Fig. 3. (a) An STT-RAM cell, (b) an STT-RAM MTJ in parallel (top) and anti-parallel (bottom) alignments, and (c) the organization of the sense and write circuitry for STT-RAM bitlines. (Note the existence of separate sense amplifiers and row buffer storage.)

IV. EXPERIMENTAL SETUP

A. Simulation Framework and Target System We used an in-house instruction-trace-based cycle-level multicore simulator with a front end similar to Pin [38]. This simulator can model the memory subsystem of a multicore system in detail. It models the execution in an out-of-order core including the instruction window, and on the memory side, it enforces channel, rank, bank, and bus conflicts, thereby capturing all the bandwidth limitations and modeling memory performance characteristics accurately. The memory model in the simulator is verified using DRAMSim [60] and its parameters are set to DDR3 memory timing parameters [41]. Table I shows our major processor and memory parameters. Our simulator also employs a resource utilization model similar to [6] which counts the occurrences of various memory activities to estimate the overall memory system energy consumption. For DRAM and STT-RAM, we

3) WB: In STT-RAM, an array write-back must be performed when a row buffer conflict occurs. This component involves the excess energy needed for changing the magnetic orientation of STT-RAM MTJs. characteristics, including the number of pages accessed by the application, the pressure they put on the memory system in

terms of level-2 cache misses per kilo-instructions (MPKI), level-2 cache write-backs per kilo-instructions (WBPKI) and their memory row buffer hit rates. We executed target benchmarks for 5 billion cycles, which corresponds to a real execution time of 2 seconds at 2:5 GHz.

used CACTI [44] and modified it to model STT-RAM cells and peripherals and provide us with accurate estimates for dynamic energy cost of individual DRAM and STT-RAM operations. We also calibrated our DRAM model using the Micron power calculator [42]. The normalized energy values for all memory operations modeled in this work (DRAM and STT-RAM) are given in Table II. These energy values correspond to the basic per-bit hardware events in the evaluated DRAM and STT-RAM main memories. In DRAM, the array read/write energy components involve charging/discharging the bitlines and the cell capacitances through the access transistors and the precharge component measures the energy cost of driving the bitlines to Vcc=2. In STT-RAM, the read component includes the energy of driving a small amount of current into the cell and sensing the voltage difference, whereas the write component drives a much larger current into the cells using the write drivers. The total energy for each of the high level memory commands (i.e., precharge, activate, refresh, read/write) are obtained using (i) the number of bits involved in each event, and (ii) which combination of events are triggered with each command. The granularity of individual operations is common to both types of memories: row activation, precharge, and refresh (DRAM-only) commands

RAM differ in the way array writes are performed, we provide a breakdown of total DRAM and STT-RAM energy into different components. For DRAM, we provide a breakdown of total energy into the following three components: operate on 4KB data, and read and write commands operate on 64B data. However, since DRAM and STT

1) ACT+PRE: A row can be activated only after a precharge which prepares bitlines for activation. Therefore, we merge the energy of activation and precharge and report them together.

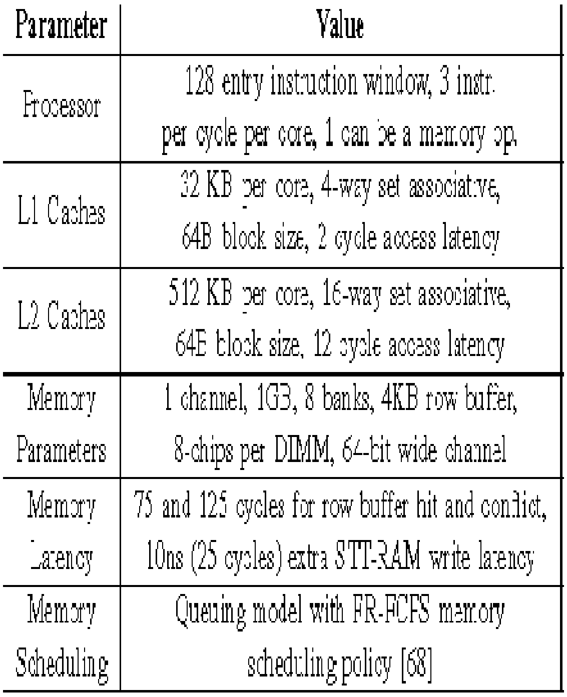
2) RD+WR: We report the total energy for DRAM read and write energy. Note that in DRAM, read and write operations both access the row buffer, but write operation also charges/discharges the bitlines and DRAM cells.

3) REF: We also report refresh energy, which is the biggest component in DRAM background energy.

For STT-RAM, we provide a breakdown of total energy into the following three components:

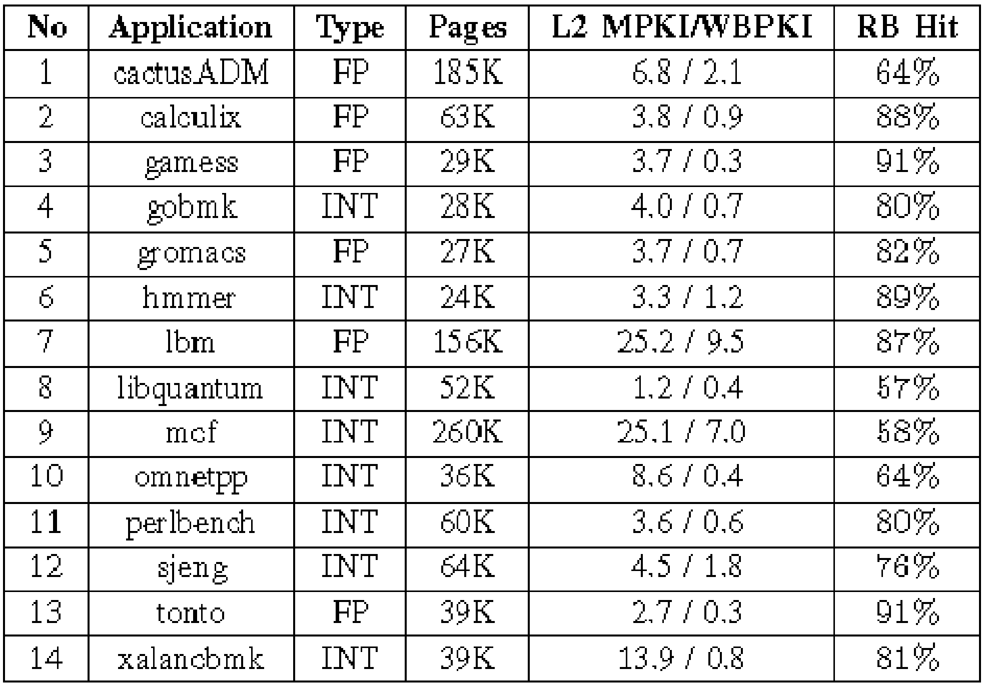
1) ACT+PRE: Similar to DRAM, we provide the total energy for row activation and bitline precharge together.

2) RB: STT-RAM read and write operations are both done on the row buffer. Unlike DRAM, STT-RAM write operation does not involve bitlines or memory cells.



Parameter	Value
Processor	128 entry instruction window, 3 instr. per cycle per core, 1 can be a memory op.
L1 Caches	32 KB per core, 4-way set associative, 64B block size, 2 cycle access latency
L2 Caches	512 KB per core, 16-way set associative, 64B block size, 12 cycle access latency
Memory Parameters	1 channel, 1GB, 8 banks, 4KB row buffer, 8-chips per DIMM, 64-bit wide channel
Memory Latency	75 and 125 cycles for row buffer hit and conflict, 10ns (25 cycles) extra STT-RAM write latency
Memory Scheduling	Queuing model with FR-FCFS memory scheduling policy [68]

TABLE I. Major processor and memory system parameters.



No	Application	Type	Pages	L2 MPKI/WBPKI	RB Hit
1	cactusADM	FP	185K	6.8 / 2.1	64%
2	calculix	FP	63K	3.8 / 0.9	88%
3	gamess	FP	29K	3.7 / 0.3	91%
4	gobmk	INT	28K	4.0 / 0.7	80%
5	gromacs	FP	27K	3.7 / 0.7	82%
6	hmmer	INT	24K	3.3 / 1.2	89%
7	lbm	FP	156K	25.2 / 9.5	87%
8	libquantum	INT	52K	1.2 / 0.4	57%
9	mcf	INT	260K	25.1 / 7.0	58%
10	omnetpp	INT	36K	8.6 / 0.4	64%
11	perlbench	INT	60K	3.6 / 0.6	80%
12	sjeng	INT	64K	4.5 / 1.8	76%
13	tonto	FP	39K	2.7 / 0.3	91%
14	xalancbmk	INT	39K	13.9 / 0.8	81%

TABLE III. Evaluated applications and their characteristics.

[1] International technology roadmap for semiconductors. In ITRS, 2012.
[2] N. Aggarwal et al. Power-Efficient DRAM Speculation. In HPCA, 2008.
[3] T. W. Andre et al. A 4-Mb 0.18-.m 1T1MTJ Toggle MRAM With Balanced Three Input Sensing Scheme and Locally Mirrored Unidirectional
Write Drivers. JSCC, 40(1), 2005.
[4] R. Ausavarungnirun et al. Staged Memory Scheduling: Achieving High
Performance and Scalability in Heterogeneous Systems. In ISCA, 2012.
[5] R. Bheda et al. Energy Efficient Phase Change Memory based Main Memory for Future High Performance Systems. In IGCC, 2011.
[6] D. Brooks et al. Wattch: A Framework for Architectural-level Power
Analysis and Optimizations. In ISCA, 2000.
[7] J. Carter and K. Rajamani. Designing Energy-Efficient Servers and Data Centers. Computer, 43(7), July 2010.
[8] E. Chen et al. Advances and Future Prospects of Spin-Transfer Torque Random Access Memory. Magnetics, IEEE Transactions on, 46(6), June
2010.
[9] S. Chung et al. Fully Integrated 54nm STT-RAM with the Smallest Bit
Cell Dimension for High Density Memory Application. In IEDM, 2010.
[10] J. Coburn et al. NV-Heaps: Making Persistent Objects Fast and Safe
with Next-Generation, Non-Volatile Memories. In ASPLOS, 2011.
[11] H. David et al. Memory Power Management via Dynamic Voltage/
Frequency Scaling. In ICAC, 2011.
Memories on High-Performance, IO-Intensive Computing. In SC, 2010.
[13] Q. Deng et al. MemScale: Active Low-power Modes for Main Memory.
In ASPLOS, 2011.
[14] G. Dhiman et al. PDRAM: A Hybrid PRAM and DRAM Main Memory
System. In DAC, 2009.
[15] Z. Diao et al. Spin-transfer Torque Switching in Magnetic Tunnel
Junctions and Spin-transfer Torque Random Access Memory. Journal of
Physics: Condensed Matter, 19, 2007.
[16] B. Diniz et al. Limiting the Power Consumption of Main Memory. In
ISCA, 2007.
[17] X. Dong et al. Circuit and microarchitecture evaluation of 3D stacking
magnetic RAM (MRAM) as a universal memory replacement. In DAC,
2008.

[18] P. Emma et al. Rethinking Refresh: Increasing Availability and Reducing Power in DRAM for Cache Applications. Micro, IEEE, 28, 2008.

[19] M. Ghosh and H.-H. S. Lee. Smart Refresh: An Enhanced MemoryController Design for Reducing Energy in Conventional and 3D Die-Stacked DRAMs. In MICRO, 2007.

[20] X. Guo et al. Resistive Computation: Avoiding the Power Wall withLow-leakage, STT-MRAM based Computing. In ISCA, 2010.

[21] J. L. Henning. SPEC CPU2006 benchmark descriptions. SIGARCHComput. Archit. News, 34(4):1–17, 2006.

.