Future Cache Design using STT MRAMs for Improved Energy Efficiency: Devices, Circuits and Architecture

Sang Phill Park, Sumeet Gupta, Niladri Mojumder, Anand Raghunathan, Kaushik Roy
Purdue University, West Lafayette, IN 47907
{sppark,guptask,niladri,raghunathan,kaushik}@purdue.edu

ABSTRACT

Spin-transfer torque magnetic RAM (STT MRAM) has emerged as a promising candidate for on-chip memory in future computing platforms. We present a cross-layer (device-circuit-architecture) approach to energy-efficient cache design using STT MRAM. At the device and circuit levels, we consider different genres of MTJs and bitcells, and evaluate their impact on the area, energy and performance of caches. In addition, we propose micro-architectural techniques *viz.* sequential cache read and partial cache line update, which exploit the non-volatility of STT MRAM to further improve energy efficiency of STT MRAM caches. A detailed comparison of STT MRAM caches with SRAM-based caches is also presented. Our results indicate that the proposed optimizations significantly enhance the efficiency of STT MRAM for designing lower level caches.

Categories and Subject Descriptors

B.3.2 [**Hardware**]: Memory Structures—*Cache Memories*

General Terms

Design, Performance, Experiments

Keywords

Cache, Memory, Emerging devices, STT MRAM, Spin

1. INTRODUCTION

The ever-increasing gap between processor speed and main memory latency has driven the demand for larger on-chip caches in processors. Traditionally, on-chip caches in modern processors are implemented using static random access memories (SRAM). However, limited scalability, susceptibility to soft errors and high leakage power of SRAM pose challenges to high-density on-chip cache implementation. In order to address the limited scalability of SRAMs, several recent processors have adopted embedded dynamic RAM (EDRAM) in lower level caches. However, vulnerability to soft errors and significant standby power of EDRAM caches due to high cell leakage are still major bottlenecks in on-chip cache design [4]. To cope with the above problems, there has been significant research directed towards several alternative embedded memory technologies [3].

Among various candidates, spin-transfer torque magnetic RAM (STT MRAM) is considered as a promising technology that can offer desirable memory attributes such as high endurance, non-volatility, soft error immunity, zero standby power and high integration capability. More importantly, its compatibility with CMOS processes makes it an attractive vehicle to realize high-density low-power embedded memories in scaled technologies [6]. However, higher write latency and write energy requirements, compared to the traditional embedded memories such as SRAM, are major issues with STT MRAM [17]. These drawbacks can preclude direct deployment of STT MRAM in level-1 (L1) caches that require fast read and write operations. However, in lower level caches such as the level-2 (L2) or last-level (LL) caches, the low leakage and high density of STT MRAMs can be more effectively utilized to replace SRAMs [7, 17].

Some previous works have explored the use of STT MRAM in the cache hierarchy, primarily through architectural techniques such as hybrid caches, write buffers, *etc.* [7, 18]. While these efforts have proven the potential of STT MRAMs, we believe that deriving highest benefits from STT MRAM requires device/circuit/architecture co-design. In this work, we explore different genres of magnetic tunnel junction (MTJ) stacks and bitcell configurations, and analyze their implications on the energy consumption and performance of STT MRAM caches under different cache utilizations. Furthermore, we present circuit/architecture co-design techniques that exploit the non-volatility of STT MRAM, not only in the standby mode, but also during dynamic cache operations. One of the consequences of non-volatility of STT MRAMs is that during column selection, the unselected columns do not consume any energy. This is unlike SRAM caches, in which the unselected bitcells need to be biased with voltages identical to the for the SRAM read operation to prevent disturb failures (known as half-select problem) [12]. Hence, in STT MRAMs, column selection is half-select-free, as a result of which only the selected columns consume energy. Based on this observation, we present two techniques — sequential tag-data access for reads and partial line update for writes — that significantly improve the energy-efficiency of STT MRAM-based caches.

In summary, we utilize device/circuit/architecture co-design to make STT MRAMs an attractive option for high density on-chip memory and to enhance the energy efficiency of STT MRAM caches. The key contributions of this work are as follows:

- We investigate the impact of various STT MRAM bitcells with different genres of MTJ stacks and bitcell configurations on total cache area, energy consumption and performance. We perform a detailed comparison of STT MRAM caches with respect to SRAM caches, based on *physical layouts* of the STT MRAM and SRAM bitcells.
- Exploiting the non-volatility of STT MRAMs, we propose a cache architecture that performs *partial cache line update* for cache writeback energy reduction. The technique does not incur any extra cache misses since it does not change the data flow between different cache levels.
- We propose a read energy reduction technique exploiting the non-volatility of STT MRAM. The technique is based on *sequential tag-data access*, and does not require signif-

DAC 2012, June 3-7, 2012, San Francisco, California, USA.

icant architectural modification.

- We also analyzed the total energy consumption of STT MRAM caches considering cache utilization during processor operations. We show that for lower level caches, STT MRAM caches are significantly more energy efficient than SRAM-based caches due to low utilization and low standby power.

2. RELATED WORK

Researchers have studied the power and performance characteristics of STT MRAM caches for general purpose processors. Wu *et al.* [18] have proposed SRAM-STT MRAM hybrid-cache architectures based on partitioning of caches into fast SRAM and slow STT MRAM regions. Xu *et al.* [17] have proposed an STT MRAM-based last level cache. In [17], the analysis is limited to the performance benefits arising from the higher integration density of STT MRAM. There have been efforts that address the large write energy of STT MRAM caches. Zhou *et al.* [10] proposed an early termination of write operations to reduce write energy of STT MRAMs. The value stored in the bitcell is sensed at the beginning of the write cycle, and if found to be identical to the new data, the write operation is terminated. In [7], a write biasing technique has been proposed in order to address the high write energy of STT MRAM in cache operation. This technique reduces the number of writebacks from L1 to lower level caches by biasing dirty cache lines to reside in L1 for a longer time. However, write-biasing in a typical size L1 cache can result in noticeable performance penalty due to an increase in L1 read miss rates. In previous work, the reported power and performance evaluations of STT MRAM caches are based on approximate estimation of STT MRAM characteristics such as area, energy and latency. In our work, the total cache area is accurately calculated from bitcell layouts based on λ-based design rules. The performance and energy consumption are evaluated using bitcell and circuit parameters obtained using physics based simulator that can comprehend different types of MTJ stacks [16] along with a circuit simulator. In addition, we consider different genres of MTJs and analyze their impact on cache performance and energy consumption. For addressing excessive write energy of STT MRAMs, we propose *partial cache line update* to avoid unnecessary overwrite of the same data, thus achieving write energy savings. Our technique does not incur cache miss increase or pre-evaluation of the stored data by exploiting the non-volatility of STT MRAMs and data redundancy in a multi-level cache hierarchy. Moreover, we show that read energy reduction can also be achieved by utilizing the non-volatility of STT MRAMs.

3. STT MRAM CACHE DESIGN

In this section, we first describe the basic operation of STT MRAM bitcells. Next, we explore alternative MTJ stacks and bitcell configurations, and evaluate their impact on the area, performance and energy of caches. Finally, we explore the dependence of the energy benefit of STT MRAM based caches on cache utilization, making a case for the use of STT MRAMs in the lower levels of the cache hierarchy.

3.1 STT MRAM Preliminaries

A conventional STT MRAM cell comprises of a magnetic tunnel junction (MTJ) and an access transistor in series (Figure 1 (a-b)). The MTJ contains a pinned layer and a free layer separated by a dielectric layer (e.g. MgO). The

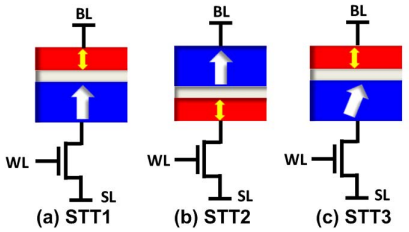

Figure 1: Schematics of an STT MRAM bitcell (a) in the standard-connected configuration (b) in the reverse-connected configuration and (c) with tilted magnetic anisotropy

pinned layer has a fixed magnetization, and the free layer is programmable by changing its magnetic orientation. The resistance of the MTJ depends on the relative magnetization of the free layer with respect to the pinned layer. Parallel magnetization of the free layer with respect to the pinned layer leads to a lower resistance (R_P) compared to the resistance in the anti-parallel state (R_{AP}). The two resistances of the MTJ define the binary states of the memory cell. A read operation is performed by sensing resistance difference of the two binary states. A write operation is performed by passing a current (I_W) through the bitcell that exceeds a critical current (I_C). The direction of (I_W) determines the final magnetization of the free layer (*i.e.*, parallel or anti-parallel states of the MTJ) [16].

3.2 STT MRAM Bitcell Design: Devices and Circuits

Different types of MTJ stacks [1, 9] and bitcell configurations [2] provide several design choices, and can result in substantially different bitcell characteristics. Before exploring these choices, we first discuss design considerations of STT MRAM bitcells. A conventional MTJ [1] has a large switching current density requirement, and the requirement increases dramatically with lower switching delay [2]. The large switching current requirement for fast write operation is one of the major challenges for energy-efficient STT MRAM design. In order to address the excessive switching current requirement, an MTJ with tilted magnetic anisotropy (TMA) has been proposed in [9]. Tilting the direction of the pinned layer, by a larger angle than what stochastic thermal noise can provide, leads to a thermal-noise-independent non-zero initial angle for precessional switching. As a result, the switching current overdrive and switching delay can be reduced significantly [9]. In our work, we consider three different bitcell designs shown in Figure 1: (i) a standard-connected configuration where the access transistor is connected to the pinned layer, (ii) a reverse-connected configuration where the access transistor

Table 1: Bitcell parameters of three STT MRAMs in Figure 1

Bitcell Type	STT1	STT2	STT3
Area*(F^2,$F = 32nm$)	56.0625	44.85	34.5
P('0') read current(μA)	143	144	62
AP('1') read current(μA)	71	82	20
Read voltage(V)	0.19	0.24	0.32
P->AP write current(μA)	367	159	126
P->AP critical current(μA)	140	140	40
AP->P write current(μA)	316	316	90
AP->P critical current(μA)	287	287	82
Access transistor width(nm)	263	202	144**
Bitcell layout aspect ratio*	1.70	1.36	1.04**

* 6T-SRAM bitcell area and aspect ratio are $176F^2$ and 2.75, respectively.

** Cell area is limited by metal to metal pitch

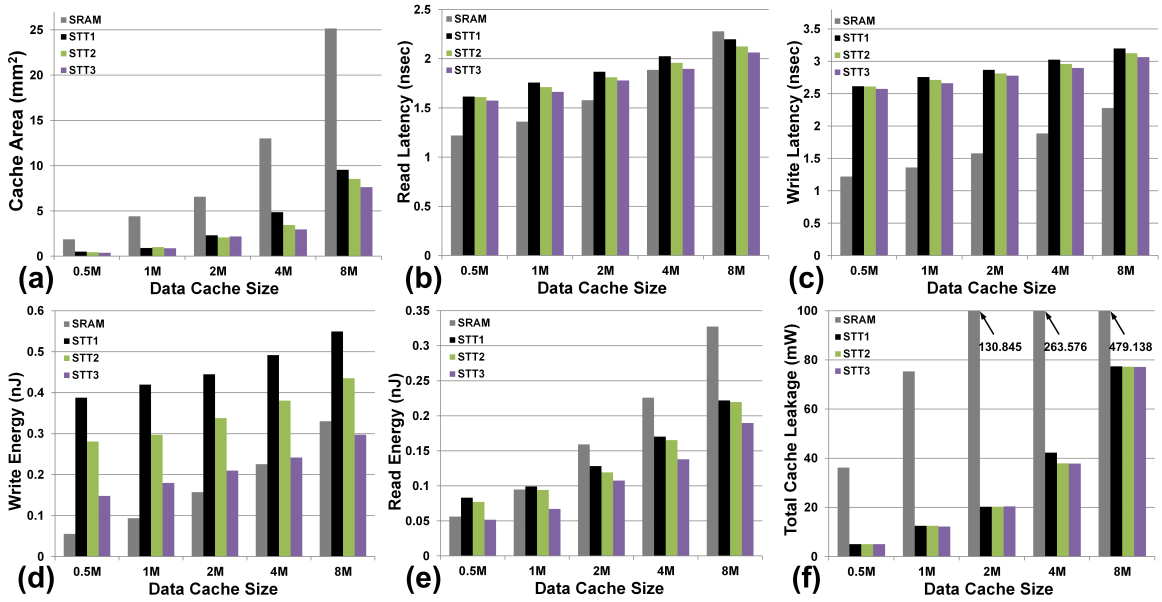

Figure 2: (a) Area requirement of SRAM and STT MRAM based caches (4-way, 64B cache line, B=Byte, M=Mega Byte) in mm^2, (b) read latency and (c) write latency in ns, (d) read energy and (e) write energy in nJ per operation, and (f) leakage power in mW

is connected to the free layer, and (iii) a configuration with tilted magnetic anisotropy. The bitcells are designed to meet the same specification with write error rate of 10^{-9}, switching time of $2ns$, 10% write margin (defined as $(I_W - I_C)/I_C$) and 50% read disturb margin (defined as $(I_C - I_{READ})/(I_C)$). We compare the characteristics of these three design options using manual layout and device simulations. In the simulations, MTJs with a free layer size of 64x64x3 nm^3 are modeled using the Non-equilibrium Green's function (NEGF) formalism [16] to obtain the electronic transport and spin transfer torque characteristics. Modeling of switching dynamics of the MTJ is carried out using the Landau-Liftshitz-Gilbert (LLG) equation with an STT term, and the MTJ models are calibrated against measurements [16, 14]. For the access transistor, $32nm$ predictive technology models (PTM) are used. In addition, a conventional thin-cell layout [11] of an SRAM bitcell is designed for comparison. The SRAM cell is designed to achieve a read access time less than $200ps$ with 128 bitcells per bitline.

Table 1 presents the bitcell parameters obtained from our evaluations. In comparison to the standard-connected bitcell, substantial reduction in write/read current along with reduction in the size of access transistors is observed in the reverse-connected bitcell and the cell with TMA. In the case of the reverse-connected bitcell, a smaller sized access transistor can provide sufficient current drive-ability due to non-source-degenerated transistor operation during P to AP switching [2]. Furthermore, the source degeneration of the transistor operation during AP to P switching reduces excessive overdrive current. In the case of the cell with TMA, the access transistor size can be further reduced due to the significantly lower critical switching current requirement of the MTJ with TMA. The size of the access transistor is a critical parameter in determining the unit cell area of STT MRAM. In the case of the two STT MRAM cells with conventional MTJs, the bitcell width is determined by the width of the access transistor. However, the bitcell width of the STT MRAM with TMA is limited by the metal to metal pitch rather than the size of the access transistor.

3.3 STT MRAM Cache vs. SRAM Cache

In this section, we evaluate STT MRAM caches based on the various bitcells presented in Figure 1 and Table 1. Performance and energy consumption of the arrays can vary, not only with different bitcell characteristics, but also with array parameters, such as capacity, the number of rows and columns, *etc.* [8]. A cache comprises of multiple arrays for storing tags and data bits. In conventional on-chip caches, both the tag and data arrays are implemented using SRAM. On the other hand, in the proposed STT MRAM caches, the tag arrays are implemented using SRAM and data arrays are implemented using STT MRAM. This is due to the fact that the write latency of STT MRAM may not be suitable for tag array operation, which requires frequent and fast updates of status bits and history bits [7]. In order to estimate the overall cache latency, area and energy consumption of the STT MRAM cache, we modified the CACTI 6.5 simulator [8] to consider (i) analog read circuits in STT MRAM data arrays (ii) SRAM-based tag arrays along with STT MRAM data arrays, and (iii) the bitcell layout geometries to optimize the array aspect ratio.

Figure 2 (a) compares the area requirements of caches designed with SRAM and STT MRAM. It is clearly shown that STT MRAM caches have a much higher integration density than SRAM cache. However, the total cache area does not fully reflect the area advantage of STT MRAM bitcells shown in Table 1, due to the area required for SRAM-based tag arrays and peripheral circuits in the STT MRAM caches. For instance, the STT MRAM bitcell with TMA has an approximately 5X smaller footprint in comparison to the SRAM bitcell. However, the 2MB caches based on the three types of STT MRAMs require 2 to $2.3mm^2$, which is slightly larger than the area requirement of 0.5MB SRAM cache ($1.9mm^2$).

The higher integration density of an STT MRAM-based data array can enable improved cache access latency and energy. As the cache area increases with capacity, the impact of wire delay on cache latency becomes larger. The SRAM cache latency increases rapidly with the capacity of

the cache due to longer delays in the metal-lines such as the wordlines, the bitlines, and the data bus. However, the latency increase of STT MRAM-based caches is more graceful due to smaller cache area (Figure 2 (b,c)). It can be seen that the STT MRAM cache can be faster in read access when the cache capacity is larger than 4MB. Similarly, as a result of the graceful increase in the write latency of STT MRAM cache, the write latency gap between STT MRAM cache and SRAM cache becomes smaller with increasing capacity.

A similar trend can be observed in the case of dynamic energy. The energy dissipated in read operations in STT1 and STT2 is higher than that of SRAM due to power dissipation in the analog read circuits, despite 4X smaller total cache area. However, for larger capacity (above 1MB), the energy dissipation due to interconnects becomes dominant. Therefore, read dynamic energy is significantly lower in STT MRAM caches. During write operations, STT MRAM caches with conventional bitcells (STT1,STT2) dissipate significantly larger energy than SRAM based caches. However, the STT MRAM cache using TMA bitcells (STT3) shows significantly lower energy dissipation due to the lower write current requirement of the bitcell. As a result, write energy dissipation of STT3 is comparable to that of SRAM cache at a capacity of 4MB.

Figure 2(f) shows that the leakage power increases with cache capacity for both SRAM and STT MRAM caches. In case of SRAM cache, the leakage power increases drastically due to bitcell leakage. On the other hand, for STT MRAM cache, the increase in leakage is much lower due to zero standby power of the bitcells. The leakage power contribution in STT MRAM cache is primarily due to SRAM-based tag arrays and peripherals.

3.4 Cache Utilization and Energy Consumption

The contribution of active and leakage energy to total energy consumption is different for SRAM- and STT MRAM-based caches. The leakage energy in an STT MRAM cache is smaller than an SRAM cache even with 4 times larger capacity (at iso-area). On the other hand, the dynamic energy for a write operation is higher in an STT MRAM cache compared to an SRAM cache. It is important to note that the total energy dissipation in a cache depends on factors such as cache access patterns (number of read and write operations) and cache utilization (number of times a processor accesses the cache per unit cycle). The cache utilization is lower than 30% in today's processors [13]. Moreover, for lower levels of the cache hierarchy, the cache utilization is significantly lower than 30%. We have measured L2 cache utilizations for various SPEC2000 benchmarks based on the Simplescalar framework [15] with a 32KB L1 cache configuration. Our simulation results also confirm low L2 cache utilization. For a majority of the benchmarks, L2 cache utilization is lower than 3%. The highest utilization, observed for the AMMP benchmark, is about 13%, and the average utilization across 16 benchmarks is only 2.2%.

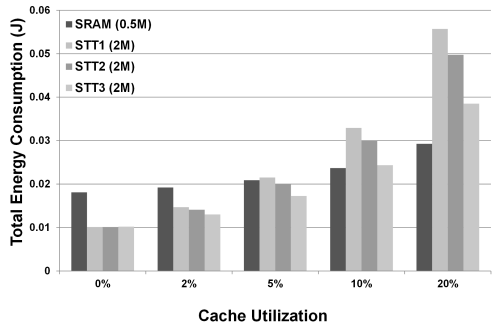

Figure 3: Total energy consumption of L2 caches at iso-area (0.5MB SRAM vs. 2MB STT MRAM)

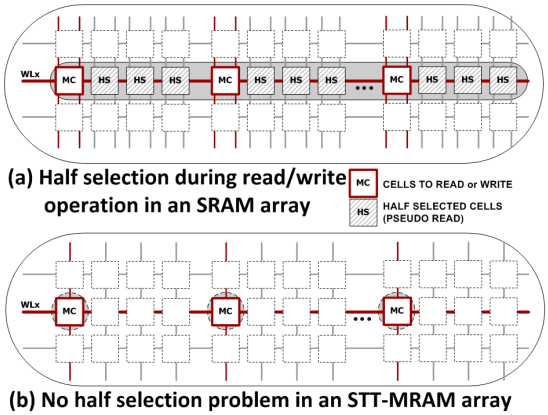

Figure 4: Column-selective read/write operations in SRAM and STT MRAM arrays

As shown in Figure 3, a 2MB STT MRAM cache shows similar or lower energy consumption than a 0.5MB SRAM cache when the utilization is lower than 10%. Although the STT MRAM cache has significantly lower energy consumption at 0% utilization (leakage only), the energy dissipation increases drastically due to excessive write energy as the utilization increases. The results are obtained using the following conditions: read and write operation ratio of 2:1, 2GHz processor speed, and total simulation time of 1 billion processor cycles. Therefore, an STT MRAM cache can achieve high energy-efficiency along with high capacity in comparison to an SRAM cache, especially in lower levels of the cache hierarchy due to the low cache utilization.

4. ENERGY-EFFICIENT STT MRAM CACHE DESIGN

One of the distinct advantages of STT MRAM, compared to SRAM, is *non-volatility* of bitcells. Interestingly, non-volatility can further improve energy efficiency in dynamic operation of STT MRAM caches. In this section, we first investigate the difference in array operations (in particular column selections) for STT MRAM and SRAM arrays. We then propose dynamic energy reduction techniques exploiting the non-volatility for read and write operations of STT MRAM cache, and evaluate their impact on the overall energy consumption.

4.1 Column Selection: SRAM vs. STT MRAM

In a conventional SRAM array, column selection is required for storing multiple words in a single row [12]. Since set associativity is common in modern caches, column selection in SRAM arrays is imperative. Furthermore, bit-interleaving can only be achieved by employing column selection. Bit-interleaving is a commonly adopted technique in SRAM arrays (1) to mitigate soft errors [12], and (2) to increase array density by bitline multiplexing [8]. In the column selection operation of an SRAM array, all unselected bitcells in the accessed row have to be under read mode to prevent unexpected bit flips, when a wordline is asserted. This phenomenon is commonly known as pseudo read or half selection [12]. Note that, in an STT MRAM array, the non-volatility of bitcells can eliminate the half selection problem.

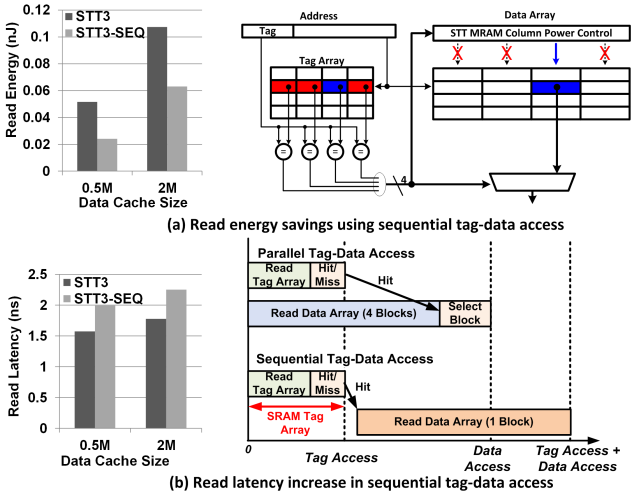

Figure 5: (a) Read energy savings and (b) Read latency increase in sequential tag-data access

As presented in Figure 4, the unselected bitcells can remain in standby mode, and hence, consume no energy during both read and write column selection operations. In the next two sub-sections, we will describe read and write energy saving techniques that are based upon this insight.

4.2 Read Energy Reduction in STT MRAM Cache

One challenge to enable energy-efficient column selection can be to identify the selected column address during cache read operation with minimal performance penalty. We observed that the proposed technique can be easily adopted in a cache implementing sequential tag-data access. Sequential tag-data access is often employed in large, lower-level caches to improve energy-efficiency during operation [8, 19]. In sequential tag-data access, a cache probes the tag array first, and identifies a hit or miss. Access to the data array occurs only when there is a cache hit, and only the sub-array storing the corresponding cache line in the data array is accessed. As a result, significant energy savings can be achieved.

This technique can be more energy-efficient in an STT MRAM cache due to half-selection-free column selection. In general, each sub-array in SRAM-based cache stores multiple cache lines in a row, in order to improve area efficiency or to employ bit-interleaving [8, 12]. Due to the half-selection issue, all cache lines in the row of the SRAM sub-array dissipate dynamic energy during read operations (due to precharging/discharging of bitlines). On the other hand, in a sub-array of an STT MRAM-based cache, only the bit-columns storing a single cache line consume energy as discussed previously. Figure 5 illustrates the proposed sequential tag-data cache access for an STT MRAM-based cache. The column address from the tag array is used to enable the selected bit-columns. The single cache line read access in STT MRAM can substantially lower read dynamic energy as shown in Figure 5 (a) (STT3-SEQ). The sequential tag-data access in cache increases the overall cache access latency [19]. However, the tag array has much smaller latency than the data array due to the smaller size of the tag array. Note that, in our proposed STT MRAM cache, the tag arrays are implemented using SRAMs and are much faster than STT MRAM data arrays. Hence, the overall latency increase due to the sequential access is not significant as shown in Figure 5 (b). Moreover, the latency increase in L2 cache does not have significant impact on the overall processor performance. Our simulation results show that, for a 2MB L2 cache, the increased latency due to sequential tag-data access results in less than 1% IPC (Instructions per cycle) reduction on average for 16 SPEC2000 benchmarks.

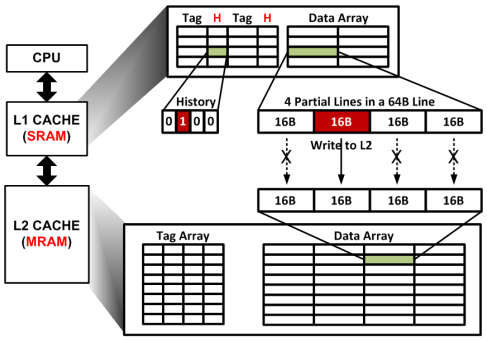

Figure 6: Partial cache line update

4.3 Write Energy Reduction in STT MRAM Cache

Similar to the read energy reduction technique described above, improvement in write energy efficiency of STT MRAM cache can also be achieved by exploiting half-selection-free column selection. We propose partial cache line update (PLU) to reduce cache writeback [5] energy consumption. This technique exploits data redundancy in a multi-level cache hierarchy as well as non-volatility of STT MRAM bitcells. In a writeback cache, writeback is performed when a dirty cache line in the L1 cache needs to be replaced by a new cache line. Hence, the dirty line has to be written into the L2 cache. In general, a cache line consists of multiple processor words in order to take advantage of spatial locality, and the size of a cache line is the unit data size in a cache. As a result, the entire cache line is written during writeback operation, even if there is only a single word that might have changed in the cache line.

Figure 6 presents the proposed PLU STT MRAM cache architecture. Each cache line is partitioned into n partial lines in order to utilize the energy-efficient column selection of STT MRAM arrays ($n = 4$ in the given example). During writeback from the SRAM L1 cache, only the partitions in the cache line that have been updated by the processor (1 out of 4 partitions in the example) are written to the STT MRAM L2 data array. The data in the remaining partitions are identical to the data already stored in the L2 cache. Therefore, writing the unchanged partitions into the L2 cache is unnecessary. The change of partitions can be tracked by using a history bit per partition. In the given example, 4 history bits to support 4 partitions are added into each tag in the L1 SRAM cache. The history data is used and reset whenever the corresponding cache line is written back into the L2 STT MRAM cache. During the PLU in the STT MRAM L2 cache, only the bitcells belonging to the updated partitions are written, while the other bitcells in the unchanged partition remain in standby mode. Note that

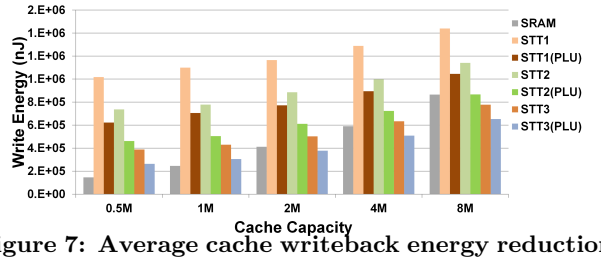

Figure 7: Average cache writeback energy reduction using PLU for 8 SPEC2000 integer benchmarks.

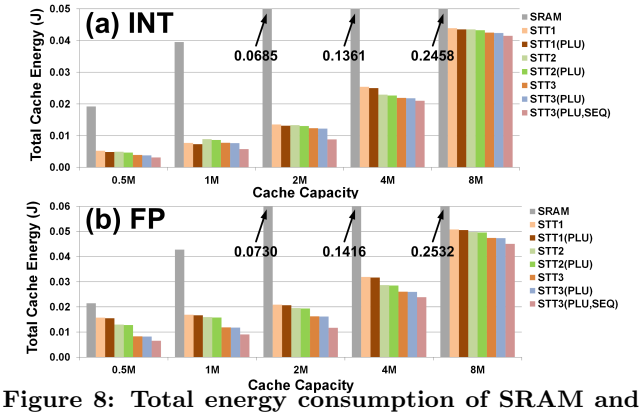

Figure 8: Total energy consumption of SRAM and STT MRAM L2 caches

an SRAM cache may not be able to take advantage of the proposed PLU technique due to the half selection problem.

We performed architectural simulation using 16 SPEC 2000 benchmarks with a processor configuration having 32KB L1 and 512KB L2 cache for 1 billion processor cycles. The results show that, during writeback operations, only 70% of cache line partitions are utilized on average when 4 partitions are used. In the case of 8 partitions per cache line, approximately 60% of the partitions are utilized. Figure 7 presents the total cache writeback energy consumption without and with PLU (for 8 partitions). STT MRAM caches with PLU show significant improvements in writeback energy compared to conventional STT MRAM caches (STT{1,2,3}). The results show that, for large STT MRAM caches (*e.g.*, 4MB and 8MB) exploiting the PLU, the total writeback energy is comparable to an SRAM-based cache.

4.4 Total L2 Energy Consumption

In order to analyze the energy efficiency of STT MRAM caches in comparison to SRAM caches, we measured the total energy consumption of L2 cache including leakage, read and write energy over 1 billion cycles of processor execution. The results presented in Figure 8 are obtained by averaging L2 cache energy consumption across 8 integer and 8 floating point benchmarks. SRAM-based L2 cache shows the largest energy consumption compared to STT MRAM caches with the same capacity, due to the significant leakage energy of SRAM bitcells. The energy difference is further improved for larger cache capacities. Moreover, under iso-area comparison (*e.g.*, 0.5MB SRAM and 2MB STT MRAM caches), STT MRAM caches show significant energy benefit along with larger cache capacity (note that larger capacity improves processor performance by lowering cache misses). Our results show that a processor with 2MB STT MRAM L2 outperforms one with 0.5MB SRAM L2 by 10% in IPC. The energy efficiency is further improved by employing the proposed sequential tag-data access in read and partial-line-update in write operations. In particular, the cache employing the proposed sequential tag-data access in addition to PLU (STT3(PLU,SEQ) in Figure 8) shows substantial total energy reduction ranging from 2% to 28% across various benchmarks.

5. CONCLUSION

In this work, we performed a comprehensive analysis of the performance, energy consumption and integration density of STT MRAM caches in comparison to conventional SRAM cache. We considered different genres of MTJ stacks and STT MRAM bitcell configurations in this study. Based on the detailed analysis of various bitcell characteristics including accurate area estimation from physical layout, we showed that, for large cache capacity, STT MRAM caches can have lower dynamic energy consumption and read latency compared to SRAM caches with the same capacity. Moreover, the low leakage energy consumption and high integration density of STT MRAM are highly beneficial for lower level caches (due to low utilization), and improve energy efficiency and processor performance. We also proposed read and write energy reduction techniques, namely sequential tag-data access in reads and partial cache line update in writes, which exploit the non-volatility of STT MRAM bitcells. The results show that the proposed techniques further improve the energy efficiency of STT MRAM caches.

6. ACKNOWLEDGMENTS

This research was supported in part by NRI, INDEX, Intel Corporation, and Qualcomm.

7. REFERENCES

[1] C. Augustine *et al.* Numerical analysis of typical STT-MTJ stacks for 1T-1R memory arrays. In *Proc. IEDM*, 2010.

[2] C. J. Lin *et al.* 45nm low power CMOS logic compatible embedded STT MRAM utilizing a reverse-connection 1T/1MTJ cell. In *Proc. IEDM*, pages 1 –4, Dec. 2009.

[3] D. Sandre *et al.* A 90nm 4Mb embedded phase-change memory with 1.2V 12ns read access time and 1MB/s write throughput. In *Proc. ISSCC*, Feb. 2010.

[4] K. Itoh. Embedded memories: Progress and a look into the future. *IEEE Design & Test*, 28(1):10 –13, Jan.-Feb. 2011.

[5] J. L. Hennessy *et al. Computer Architecture: A Quantitative Approach*. Morgan Kaufmann, May 2002.

[6] K. Lee *et al.* Development of Embedded STT-MRAM for Mobile System-on-Chips. *IEEE Trans. Magnetics*, 2011.

[7] M. Rasquinha *et al.* An energy efficient cache design using Spin Torque Transfer (STT) RAM. In *Proc. ISLPED*, 2010.

[8] N. Muralimanohar *et al.* Optimizing NUCA Organizations and Wiring Alternatives for Large Caches with CACTI 6.0. In *Proc. MICRO*, pages 3–14, 2007.

[9] N. N. Mojumder. *Design of Hybrid Spintronic Devices at Scaled Technologies for non-Volatile Memory Applications*. PhD thesis, Purdue University, Dec. 2011.

[10] P. Zhou *et al.* Energy reduction for STT-RAM using early write termination. In *Proc. ICCAD*, Nov. 2009.

[11] S. Ohbayashi *et al.* A 65-nm SoC Embedded 6T-SRAM Designed for Manufacturability With Read and Write Operation Stabilizing Circuits. *IEEE JSSC*, Apr. 2007.

[12] S. Park *et al.* Column-selection-enabled 8T SRAM array with ∼1R/1W multi-port operation for DVFS-enabled processors. In *Proc. ISLPED*, pages 303 –308, Aug. 2011.

[13] S. Ramaswamy *et al.* An utilization driven framework for energy efficient caches. In *Proc. HiPC*, pages 583–594, 2008.

[14] S. Yuasa *et al.* Giant room-temperature magnetoresistance in single-crystal Fe/MgO/Fe magnetic tunnel junctions. *Nat. Mater.*, 3, Dec. 2004.

[15] Simplescalar LLC. `http://www.simplescalar.com`.

[16] X. Fong *et al.* Bit-cell Level Optimization for Non-volatile Memories Using Magnetic Tunnel Junctions and Spin-Transfer Torque Switching. *IEEE Trans. Nanotechnology*, 2011.

[17] X. Wei *et al.* Design of Last-Level On-Chip Cache Using Spin-Torque Transfer RAM (STT RAM). *IEEE Trans. VLSI*, 2011.

[18] X. Wu *et al.* Hybrid cache architecture with disparate memory technologies. In *Proc. ISCA*. ACM, 2009.

[19] Z. Chishti *et al.* Distance associativity for high-performance energy-efficient non-uniform cache architectures. In *Proc. MICRO*, pages 55–66, 2003.