

# Avoiding Unnecessary Write Operations in STT-MRAM for Low Power Implementation

Rajendra Bishnoi, Fabian Oboril, Mojtaba Ebrahimi and Mehdi B. Tahoori

Chair of Dependable Nano Computing (CDNC), Karlsruhe Institute of Technology (KIT), Karlsruhe, Germany

Email: {rajendra.bishnoi, fabian.oboril, mojtaba.ebrahimi, mehdi.tahoori}@kit.edu

**Abstract**—Spin Transfer Torque (STT) is a promising emerging memory technology because of its various advantages such as non-volatility, high density, virtually infinite endurance, scalability and CMOS compatibility. Despite all these features, high write current is still a challenge for its widespread use. When writing a value that is already stored, a significant current flows through the Magnetic Tunnel Junction (MTJ) cell which is almost the same as that required to flip the stored data. This increases the total power consumption of the memory. To address this issue, we propose a technique which can avoid unnecessary write operations with bit-level granularity. Our technique can save 68.9 % of the total write power consumption with a minor area overhead (0.68 %) and only a small timing penalty (1.33 %).

**Keywords**—STT-MRAM, non-volatile memory, low power, write avoidance

## I. INTRODUCTION

As memory is a key component of computing systems, it has stringent design requirements as it faces various design challenges such as scalability and high leakage power [14]. An emerging storage device that fits well to these requirements is *nano-magnetic random access memory* (MRAM). Especially the *Spin Transfer Torque* memory (STT-MRAM) is very promising due to its advantageous features such as non-volatility, scalability, high density, low read latency, and CMOS-compatibility [3], [7], [20]. In a STT-MRAM, data is stored in a *Magnetic Tunnel Junction* (MTJ) cell by exploiting the magnetic orientation of two independent ferromagnetic layers. However, a high write current of several hundred  $\mu\text{A}$  is required to flip that magnetization [7] which is a major challenge for the establishment of STT-based storage devices in universal memories. The high write current infers a high power consumption of the memory (write energy is about 10x higher than in SRAM [4], [18]) and also severe stress is imposed on the MTJ cell. As a result, various STT performance parameters such as the *Tunnelling Magneto Resistance* (TMR), write current, and latency as well as lifetime of the cell are degrading over time [16], [22].

In general, write operations in memories can be split into two categories: *necessary write* operations which flip the bit-cell value, and *unnecessary write* operations which write the same data as stored into the bit-cell (i.e. no flip). In reality, typical applications show that 20 %–68 % of all write operations fall into the second category [12], [26]. In addition, more than 70 % of the dynamic power of STT-MRAM is consumed by write accesses [26]. Hence, as write current in STT-MRAM is very high, no matter whether there is a necessary or unnecessary write operation, it is crucial to avoid the latter in order to reduce the overall power consumption.

To tackle this issue, the *Early Write Termination* (EWT) technique is proposed in [26]. This technique uses special

sensing circuits to detect the value which is currently stored, while the write process is performed. If the present data is the same as the one that is currently written to the memory cell, the write operation is *terminated* immediately. This way, for unnecessary write operations the write process is terminated at an early stage. Hence, the EWT technique can save a large amount of power. However, beside this advantage, the EWT technique has also major drawbacks. First, the additional circuitry required for sensing the stored data value is power hungry and significantly contributes to the overall write power consumption (up to 28.1 % overhead according to our investigation). In addition, the area overhead is considerable. Furthermore, the proposed EWT implementation can lead to a reduced write current which may increase the write latency.

In this paper, we propose a novel circuit-level technique to *completely avoid* unnecessary write operations in STT-MRAM. Our proposed technique reuses the existing read circuitry and hence has a much smaller area penalty than the aforementioned EWT technique (0.7 % vs. 2.4 %). In general, the read time of STT-MRAM is comparable to that of SRAM and it is very less compared to the write time of STT-MRAM [3]. Consequently, there is just a small timing penalty (1.33 %) when a read operation is performed before write. In addition, the power overhead for the necessary control circuits is very small (at most 2.4 %). Therefore, our technique is much more efficient than the EWT technique. It allows to save in average 68.9 % of the total write power consumption as well as up to 95 % considering only unnecessary write operations compared to 46.7 % and 71 %, respectively, for the EWT technique.

In summary, our contributions in this paper are as follows:

- We design an efficient technique to completely avoid unnecessary write operations which saves 68.9 % of write power, with a bare minimal area overhead of 0.68 % and a timing penalty of only 1.33 %.
- We design and implement an efficient technique to sense data using a pre-charge sense amplifier.
- We perform a comprehensive analysis of our proposed technique at both circuit- and architecture-level.

The rest of this paper is organized as follows. In Section II, the basics of STT-MRAM, related work and drawbacks of the EWT technique are discussed. Section III explains our proposed technique as well as the sensing schemes used in our implementation. In Section IV, the results are discussed, and finally Section V concludes the paper.

## II. BACKGROUND

### A. Spin Transfer Torque MRAM

In STT-MRAM, MTJ cells are the storing devices. These are build out of two ferromagnetic layers separated by a thin *barrier oxide layer* (e.g. magnesium oxide, MgO) as shown in

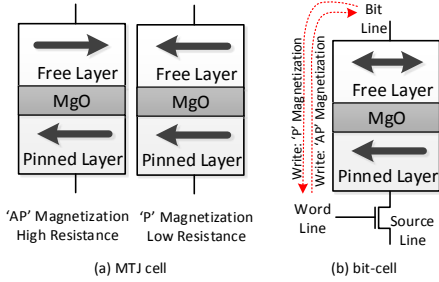


Fig. 1. Spin transfer torque storing device

Figure 1(a). One of the ferromagnetic layers has always a fixed magnetic orientation and is known as *pinned layer* or *reference layer*. The other ferromagnetic layer is referred to as *free layer* where the magnetic orientation can be freely rotated based on the direction of the current flowing through the MTJ cell. When the magnetic orientation of the free layer is in *parallel* (P) or *anti-parallel* (AP) to that of the reference layer, the MTJ cell has a low resistance or high resistance, respectively. The two resistance states of the MTJ cell can store either a logic '1' or '0' depending on the memory architecture implementation. In this work, the 'AP' state is used to store a logic '0' and a logic '1' is stored in form of the 'P' state.

One of the important physical parameters of the MTJ device is the *Tunneling Magneto Resistance (TMR)*, which is defined as:

$$TMR = (R_{AP} - R_P) / R_P,$$

where  $R_{AP}$  and  $R_P$  are the resistances in the 'AP' and 'P' state of the MTJ cell, respectively [9]. In STT-MRAM, a high TMR is of special interest, as a higher TMR improves the read margin, i.e. it is "easier" for the sense amplifier used in the read circuitry to distinguish between the 'AP' and 'P' state. Hence, the probability for a read failure decreases with increasing TMR.

The bit-cell which we use in this work is build on 1T1MTJ<sup>1</sup> and has three terminals namely *source line* (SL), *bit line* (BL) and *word line* (WL) as shown in Figure 1(b). It consists of a single MTJ cell and a single access transistor. The source of the access transistor is connected to the source line and the drain is connected to the pinned layer of the MTJ cell. The gate of the access transistor is connected to the word line to select the required bit-cell for the memory operations. The free layer of the MTJ cell is connected to the bit line terminal.

In STT-MRAM, the read current is unidirectional and flows through the bit-cell to sense the stored value based on the resistance state. In contrast, the write current is bi-directional and the value to be written in the bit-cell is determined by the direction of the current as illustrated in Figure 1(b). As a consequence, the write current and the switching time are different depending on the value stored in the bit-cell, because of an inherent torque asymmetry of the MTJ device [23].

## B. Related Work

A major challenge for STT-MRAM is the high current of several hundred  $\mu\text{A}$  that is required to flip the magnetization of the free layer of the MTJ cell [7]. Hence, the overall power consumption of the memory is negatively affected. Therefore,

several research work introduced approaches to reduce this high write current.

At architecture-level, write-aware scheduling and re-computation can be used to reduce the total number of write operations [8]. Furthermore, a bit inversion technique to reduce the number of ones being written is proposed in [15]. Therefore, the number of ones are counted before the write operation is initiated and if it is greater than half of the total word size, all bits are inverted to decrease the total write energy. Another architectural-level technique separates the write process for '0' and '1' instead of having a parallel execution [19]. By this means, both write operations can be optimized independently to save power and overcome the challenge of asymmetrical writes in STT-MRAM.

Apart from these architecture-level methods, there are a few circuit- and device-level techniques to reduce the write current in STT-MRAM. Asynchronous Asymmetrical Write Termination (AAWT) is proposed in [2] to reduce the current for write operations resulting in a 'P' state. Bit-line clamping is an approach to lower the voltage with the help of a pass transistor [10]. Another technique named "balanced write scheme" lowers the word-line voltage and a negative voltage is applied at the bit-line terminal to improve the write current [11]. A device-level technique named Perpendicular Magnetic Anisotropy (PMA) reduces not only the switching current but also the switching time [17]. However, it has some drawbacks such as a large damping constant, and potential lattice mismatch for high TMR ratios [13], [21].

All of the aforementioned techniques are orthogonal to our proposed technique, as it is intended to detect and avoid unnecessary write operations at bit-level. Hence, it can be combined with any of the other solutions.

A few techniques exist in the literature that combine read and write operations for various purposes. In [1] a write-then-read method and a verify-one-while-writing technique are proposed to enhance the reliability of STT-MRAM. These techniques can be also used to reduce the energy consumption by using more aggressive write latencies. In [25], a read-before-write scheme is presented to remove redundant write operations in *phase change memories* (PCM). However, the objective is not to reduce the energy consumption but to enhance the endurance of the bit-cells. Because of this a very complex implementation is chosen, which even increases the overall energy consumption. In summary, although the combination of read and write operations is already proposed for different objectives, there exists no technique that uses an efficient read-compare-write method for STT-MRAM for energy reduction, which is our proposal in this paper. Therefore, our implementation is intended to maximize power savings with low performance and area costs.

## C. Early Write Termination (EWT)

Besides the methods mentioned in the previous subsection, there is also a technique named *Early Write Termination (EWT)* that is designed to reduce the power consumption of STT-MRAM. The EWT technique presented in [26] terminates write operations as soon as these are detected to be unnecessary. To implement such a behavior, several additional circuits such as two conversion circuits using basic differential amplifiers, one sense amplifier, one latch, three multiplexers and one inverter are required. Due to these additional circuits, the proposed EWT architecture not only imposes a significant

<sup>1</sup>1T1MTJ = 1 access transistor + 1 MTJ cell

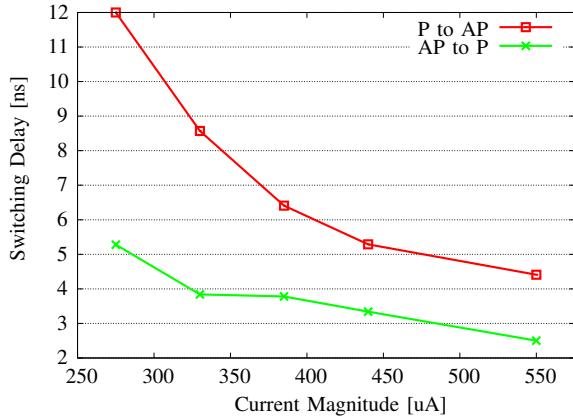


Fig. 2. Switching delay of an MTJ cell for different current values

area overhead but also contributes to the total (write) power consumption of the memory. Furthermore, these additional circuits are part of each column of the memory bit-cell array, so the ratio of the periphery area to that of the array circuits increases in direct relation with the number of bits per word.

To avoid any write timing penalty, the bit-cell value detection and the write operation run in parallel. Therefore, the conversion circuits used for EWT are biased by the write current. However, this will reduce the current flowing through the MTJ cell. As a result, the switching time of the magnetic orientation can increase and hence can infer some latency penalty. To illustrate this behavior, we performed an experiment with a single MTJ cell which was connected to a configurable current source. The results of our SPICE simulations are depicted in Figure 2. Especially the 'P' → 'AP' transition is very sensitive to the amount of current flowing through the MTJ device. Hence, already small deviations in current can lead to a considerable delay penalty. Moreover, the current reduction can even lead to write failures if the current is not high enough to flip the magnetization at all. To avoid such reliability issues and performance penalties, designers should increase the strength of the write driver which will further increase the dynamic power of the memory. Another challenge associated with the proposed EWT implementation is that for a short pulse width, the bit-cell value detection may be incorrect [15].

Although our proposed technique also targets unnecessary write operations, the idea behind our method is completely different. While the EWT technique, as explained before, just *terminates* write operations as soon as these are classified as unnecessary, our approach is to completely *avoid* such write operations. As a result, using our method the write process is not even initiated, while it is processed for some time using EWT. In addition, our implementation reuses existing circuits for the sensing phase. As a consequence, our technique is much more efficient in terms of area and power (see Section IV).

### III. PROPOSED IMPLEMENTATION

In this section, our proposed technique to avoid unnecessary write operations for STT-MRAM is presented. In Section III-A, the write behavior of a single bit-cell is explained to motivate our proposed technique. The technique itself is explained in Section III-B, followed by the sense amplifier circuit required to sense the current state of a bit-cell.

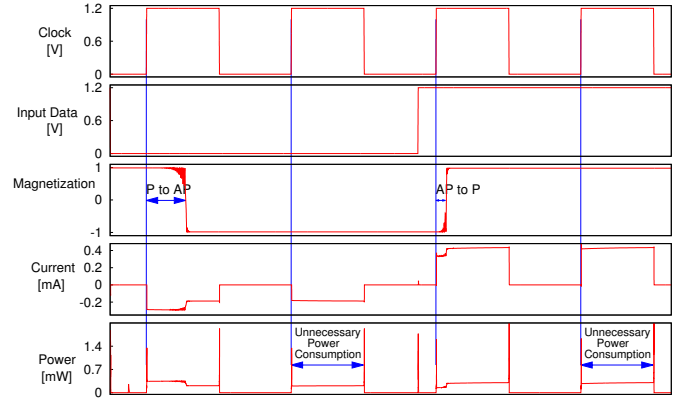


Fig. 3. Waveform to demonstrate the real behavior of an MTJ cell

#### A. Write Behavior of an STT-MRAM Bit-Cell

To demonstrate the write behavior of a bit-cell, we have conducted an experiment with a single 1T1MTJ bit-cell. The waveform for this setup is shown in Figure 3. In this experiment, the sequence "0011" is written to a bit-cell that stored the value '1' to show the effect of necessary and unnecessary write operations on the total power consumption of the memory. The first write operation is considered to be a necessary write operation as the bit-cell value flips. The bit-cell value remains unchanged during the second write operation as the same value is re-written which is considered to be an unnecessary write operation. The third and fourth write operations are also treated to be necessary and unnecessary write operations, respectively. As shown in Figure 3 the power consumption for both types of unnecessary write operations ('0' → '0' and '1' → '1') is considerable (around 0.3 mW). Hence, these write operations, which occur frequently, need to be addressed to achieve a low power solution.

#### B. Avoiding Unnecessary Write Operations

As explained in the previous subsection, the unnecessary write operations significantly contribute to the total power consumption of the STT-MRAM. Therefore, we propose a technique which does not initiate the write operation for a bit-cell if it is detected to be unnecessary.

For our technique, the process of the write operation is divided into two parts:

- 1) *Bit-cell value detection*: To detect what value is already stored in the bit-cell, it is required to sense the resistance state of the bit-cell by passing current through it. This operation is performed using a memory read operation and will be referred to as *internal read operation*.
- 2) *Actual write operation*: When the bit-cell value is known, it is compared with the value to be written. If these values are different, the actual write operation is performed otherwise the write operation is avoided. This conditional implementation is done using a special *write activation circuit*.

At circuit-level, the desired behavior can be implemented using the aforementioned write activation circuit and parts of the existing read circuitry (sense amplifier and latch) as shown in Figure 4. In this implementation, the access transistor is driven through the word line decoder. The memory read operations are performed by passing current through the MTJ

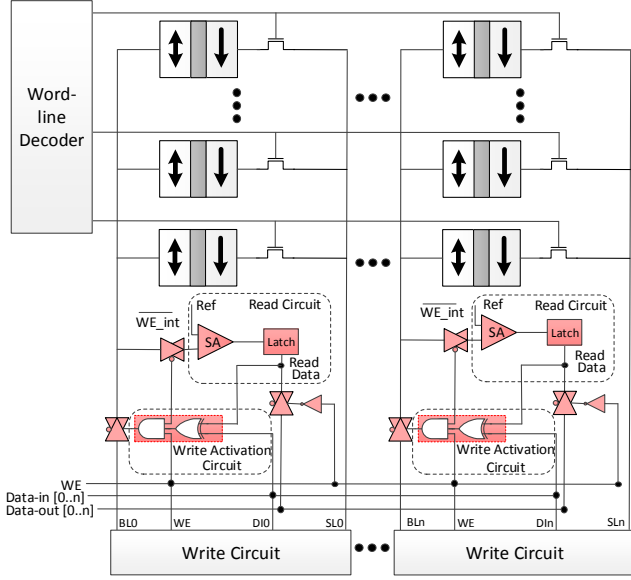


Fig. 4. Block diagram showing the implementation of our proposed technique to avoid unnecessary write operations

cell and sensing its resistance state using a sense amplifier. The memory write operations are conducted by passing current through the BL and SL terminal of the bit-cell.

The *write enable* (WE) signal is the control signal to distinguish between non-internal read and write operations. Consequently, the WE signal is '1' and '0' corresponding to write and read operations, respectively. To separate the internal read from the actual write operation the WE\_int signal is used. Initially, the WE\_int signal is '0' as in our case all memory operations start with an internal read. As a result, the stored value of the addressed bit-cell is sensed using the sense amplifier of the read circuitry and then latched. As soon as this internal read operation is over, the WE\_int signal changes to '1'. If the WE signal is '0' the latched data is sent to the memory output ports, since in this case the latched data corresponds to a non-internal read operation. Otherwise is '1', i.e. a write operation is ongoing (internal read) and the latched value is used as one of the inputs of the write activation circuit. This circuit compares the latched value with new input value using an XOR gate as shown in Figure 4. If both values are the same, the write activation signal is set to '0' and the write activation circuit does not initiate the actual write process. In the opposite case, the write activation signal is set to '1' and the write process is started. This whole process with all corresponding signal states is also shown in Table I.

The functionality of our proposed implementation is depicted as a waveform in Figure 5. It shows all four possible cases of the bit-cell behavior for STT-MRAM similar to one shown in Figure 3, i.e. '1'→'0', '0'→'0', '0'→'1'

TABLE I. TRUTH TABLE FOR THE OPERATION OF THE PROPOSED WRITE ACTIVATION CIRCUIT

Input Data	Sensed Data	WE	WE_int	Write Activation Signal	Operation
X	X	0	0	0	Read
X	X	1	0	0	Write with internal read
0	0	1	1	0	Write avoided
0	1	1	1	1	Normal write
1	0	1	1	1	Normal write
1	1	1	1	0	Write avoided

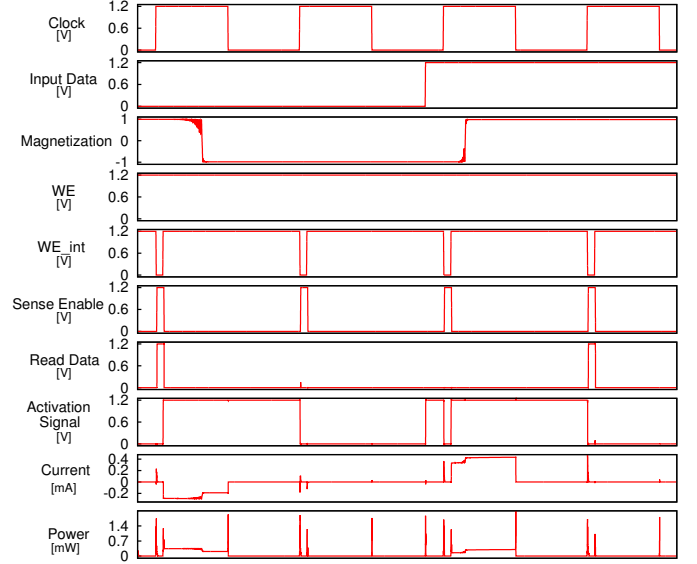


Fig. 5. Waveform to demonstrate the real behavior of an MTJ cell using our proposed technique

and '1'→'1'. Compared to Figure 3, it is observed that the current waveform for our proposed circuit remains '0' for the unnecessary write operations and it also consumes just a minor amount of power for these durations. In our implementation, the (internal) read sensing takes around 226 ps to detect the stored value in the bit-cell. In total, it takes just around 319 ps to initiate the write operation.

The normal read and write operations are not disturbed by our implementation as these are controlled by the WE and WE\_int signals. Moreover, the write activation is simple and can be easily integrated with the write circuit of the STT-MRAM. The additional circuit components for this implementation are few (one 2-input XOR, one 3-input AND) and hence it occupies very little area (0.68%) and also just has a low power overhead (2.4%). The timing penalty is also very small (1.33%) since the read operation is much faster than a write operation (upto 11 ns) due to the asynchronous behavior of STT-MRAM.

### C. Sense Amplifier for Bit-Cell Sensing

To achieve a low power solution, we use a pre-charge sense amplifier to sense the data for read (both internal and normal), similar to the one designed in [24]. The circuit diagram for this pre-charge sense amplifier is shown in Figure 6. As discussed in Section II, in STT-MRAMs the logic data is stored in terms of resistance states of the bit-cell. Let us assume that  $R_P$  and  $R_{AP}$  are the resistance values for the bit-cell when it is in the 'P' and 'AP' magnetization state, respectively. Since we use MTJ cells with a high TMR, a reference node with the resistance value  $(R_P + R_{AP})/2$  simplifies the sensing process. This resistance can be achieved by connecting two series connections of  $R_{AP}$  and  $R_P$  in parallel as shown in Figure 6. The effective resistance ( $R_{ER}$ ) value of that structure can be obtained as follows:

$$\begin{aligned}
 R_{ER} &= (R_{AP} + R_P) \parallel (R_{AP} + R_P) \\
 &= (R_{AP} + R_P)/2
 \end{aligned}$$

A pre-charge sense amplifier works in two phases, one is the pre-charge phase and the other one is the sensing phase.



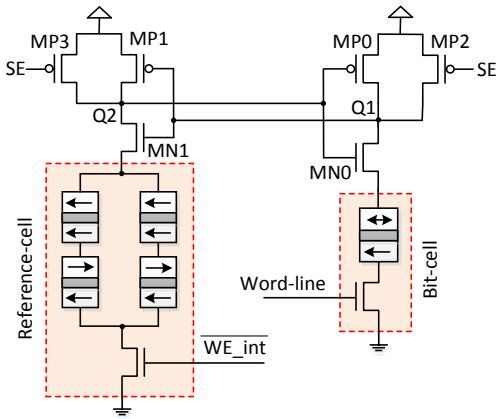


Fig. 6. Circuit diagram for the used pre-charge sense amplifier

In the pre-charge phase, the *sense enable* signal (SE) is '0' to equalize the two nodes (Q1 and Q2). In the sensing phase, the word-line is activated, the SE signal is '1' and the current is flowing through both the reference and the bit-cell terminals. If the resistance state of the bit-cell is  $R_{AP} (> R_P)$ , less current will flow through the bit-cell branch compared to the reference branch, since the latter has a resistance of  $R_{ER} < R_{AP}$ . This in turn raises the potential at the Q1 node and reduce the potential at the Q2 node. Here two inverters are connected back to back, to attain a stable value quickly, i.e. Q1 goes to '1' and Q2 to '0'. Similarly, if the resistance state of the bit-cell is  $R_P$  (i.e. a logic '1' is stored in the bit-cell), then the values of the nodes Q1 and Q2 settle at '0' and '1', respectively. Hence, we used the node Q2 as output of the sense amplifier which is connected to a latch.

We have chosen this pre-charge sense amplifier for our implementation as it is very fast (sensing takes just 226 ps) and consumes small amount of power. Nevertheless, our technique is not limited to this specific sense amplifier and any other sense amplifier can be used in the read circuitry.

#### IV. EXPERIMENTAL SETUP AND RESULTS

In order to evaluate the effectiveness of the proposed technique, a detailed model is implemented at circuit-level. Based on the information obtained from the circuit-level analysis, an abstract model of the memory is employed in an architecture-level implementation and its effect on the overall power consumption for several workloads is shown.

##### A. Circuit-Level Analysis

For the circuit-level implementation of our proposed technique, we have employed the TSMC 65nm general purpose standard cell library. For the MTJ cells we have used the model from [5] with a energy barrier of 51 kT and a barrier thickness of 1.15 nm. We have used a supply voltage of 1.2 V for both read and write circuits. All of the following results

TABLE II. POWER CONSUMPTION FOR STANDARD STT-MRAM AND WITH OUR PROPOSED TECHNIQUE FOR 32 BITS PER WORD

Magnetization	Standard (in uW)	Proposed (in uW)	Reduction (in %)
'AP' → 'P'	8879.3	8894.1	-0.2
'P' → 'AP'	9510.7	9746.5	-2.5
'AP' → 'AP'	6685.5	442.3	93.4
'P' → 'P'	9423.2	437.4	95.3

were obtained using detailed SPICE simulations (with Cadence Spectre).

The total power consumption for a single memory column consisting of 32 bit-cells (32 bits per word) for the standard and our proposed technique is given in Table II. As shown, our technique can reduce the power consumption for unnecessary writes by around 94%. Furthermore, the additional circuits required by our technique infer a very small power overhead (2.4%) when the bit-cell flips.

We have also estimated the area required for our implementation at circuit-level. The total area is calculated by extrapolating the array sizes and estimating the area of the periphery circuits. As shown in Table III, the area overhead for our technique is rather low (around 0.68%). However, as our technique is intended to completely avoid unnecessary write operations, a read operation has to be performed before every write operation. As a consequence, there is a timing penalty of around 1.33%. This timing penalty can be overcome by using a higher write current. In this case some part of the energy savings will be sacrificed, but the performance will improve. Using the same experimental setup, with which we extracted Figure 2, we found out that an increase of 0.37% for the write current is sufficient to compensate the delay penalty. Hence, with negligible power and energy costs the delay penalty of our technique can be eliminated.

To compare our proposed technique with the EWT technique [26], we added the latter in our design environment and analyzed its power consumption and area overhead. The corresponding results are shown in Table III. For the EWT technique, the energy saving amount is the same as the power saving, as this technique is supposed to have no timing penalty. Since our proposed technique infers a timing penalty the energy savings are slightly worse than the power savings. Nevertheless in all conditions our technique is superior compared to the EWT technique in terms of energy savings. Furthermore, the implementation of the EWT technique uses more additional circuits than our proposed implementation and hence the EWT has a higher area overhead (2.33%) than our proposed technique (0.68%).

Please note that the EWT technique is intended to have no timing penalty as it is terminating unnecessary write operations in an early phase. However, as discussed in Section II-C its implementation can reduce the write current and hence, depending on the MTJ device, also cause a longer write latency.

TABLE III. COMPARISON OF EWT WITH OUR PROPOSED TECHNIQUE

	EWT [26]				Proposed			
	'AP' → 'P'	'P' → 'AP'	'AP' → 'AP'	'P' → 'P'	'AP' → 'P'	'P' → 'AP'	'AP' → 'AP'	'P' → 'P'
Power Saving	-28.2 %	-9.0 %	70.9 %	70.0 %	-0.2 %	-2.5 %	93.4 %	95.4 %
Energy Saving	-28.2 %	-9.0 %	70.9 %	70.0 %	-3.2 %	-5.5 %	93.2 %	95.2 %
Area Overhead	2.33 %				0.68 %			
Timing Penalty	0 %				1.33 %			

TABLE IV. POWER REDUCTION FOR LEON2 PROCESSOR MAIN MEMORY WHEN IT IS RUNNING MiBENCH WORKLOADS FOR OUR PROPOSED TECHNIQUE AND FOR EWT

Benchmark	Cycles	Write Type Occurrence Rate				Power Reduction	
		AP→AP	AP→P	P→AP	P→P	Proposed	EWT[26]
basicmath	500M+	57.4 %	11.7 %	11.7 %	19.2 %	68.7 %	44.7 %
bitcounts	77M	71.5 %	4.7 %	4.7 %	19.1 %	83.0 %	60.3 %
crc32	500M+	37.6 %	12.5 %	12.5 %	37.6 %	68.2 %	45.9 %
fft	500M+	61.2 %	11.1 %	11.1 %	16.7 %	68.7 %	46.9 %
qsort	6M	34.3 %	18.5 %	18.2 %	29.1 %	56.0 %	34.6 %
sha	500M+	47.2 %	14.3 %	14.3 %	24.2 %	63.1 %	41.4 %
stringsearch	3M	69.5 %	8.4 %	7.8 %	14.4 %	74.9 %	53.0 %
Average		54.1 %	11.6 %	11.5 %	22.9 %	<b>68.9 %</b>	<b>46.7 %</b>

## B. Architecture-Level Analysis

To evaluate the effectiveness of our proposed technique considering a real system executing typical applications, we have employed the circuit-level power model in an architecture-level implementation. In this implementation an experiment is conducted using the Leon2 processor and a 32 KByte<sup>2</sup> main memory. This is implemented with an abstract model of STT-MRAM equipped with our proposed scheme. We have used several workloads from MiBench [6] for our analysis. The corresponding results are shown in Table IV. As it can be seen, our proposed technique can save in average 68.9 % of the total write power consumption. In contrast the EWT method can only save 46.7 %. Hence, when it comes to low power solutions our proposed technique is superior.

## V. CONCLUSIONS

Spin Transfer Torque (STT) is a promising emerging memory technology because of its various advantages such as non-volatility, high density, virtually infinite endurance, scalability and CMOS compatibility. However, it suffers from a high write current. In this work, we demonstrated that there is a significant write current flowing through the bit-cell even if the bit-cell value does not change. As this current considerably affects the total (write) power consumption of STT-MRAMs, we proposed a low power technique to avoid such unnecessary write operations i.e. when bit-cell value remains unchanged. Our results show that the proposed technique can save in average almost 70 % of the total write power consumption. Moreover, the area overhead of our technique is very low (0.68 %). Hence, our solution is also superior compared to previous work [26], that can only save 47 % in terms of power and increases area by 2.3 %.

## VI. ACKNOWLEDGEMENT

This work was partly supported by the European Commission under the Seventh Framework Program as part of the spOt project (<http://www.spot-research.eu/>).

## REFERENCES

- [1] X. Bi *et al.*, “Probabilistic design methodology to improve run-time stability and performance of stt-ram caches,” in *ICCAD*, 2012, pp. 88–94.
- [2] R. Bishnoi *et al.*, “Asynchronous asymmetrical write termination (AAWT) for a low power STT-MRAM,” in *Design, Automation and Test in Europe*, Mar. 2014.
- [3] M.-T. Chang *et al.*, “Technology comparison for large last-level caches (l3cs): Low-leakage sram, low write-energy stt-ram, and refresh-optimized edram,” in *Symposium on High Performance Computer Architecture*, 2013, pp. 143–154.

- [4] X. Dong *et al.*, “Circuit and microarchitecture evaluation of 3d stacking magnetic ram (mram) as a universal memory replacement,” in *Design Automation Conference*, 2008, pp. 554–559.
- [5] W. Guo *et al.*, “SPICE modelling of magnetic tunnel junctions written by spin-transfer torque,” *Journal of Physics D: Applied Physics*, p. 215001, May 2010.
- [6] M. R. Guthaus *et al.*, “Mibench: A free, commercially representative embedded benchmark suite,” in *International Workshop on Workload Characterization*, 2001, pp. 3–14.
- [7] M. Hosomi *et al.*, “A novel nonvolatile memory with spin torque transfer magnetization switching: Spin-ram,” in *International Electron Devices Meeting*, 2005, pp. 459–462.
- [8] J. Hu *et al.*, “Minimizing write activities to non-volatile memory via scheduling and recomputation,” in *Symposium on Application Specific Processors*, 2010, pp. 101–106.
- [9] A. Khvalkovskiy *et al.*, “Basic principles of stt-mram cell operation in memory arrays,” *Journal of Physics D: Applied Physics*, pp. 74001–74020, 2013.
- [10] Y. Kim *et al.*, “Write-optimized reliable design of stt mram,” in *Proceedings of the International Symposium on Low Power Electronics and Design*, 2012, pp. 3–8.
- [11] D. Lee, S. K. Gupta, and K. Roy, “High-performance low-energy stt mram based on balanced write scheme,” in *Proceedings of the International Symposium on Low Power Electronics and Design*, 2012, pp. 9–14.
- [12] K. M. Lepak and M. H. Lipasti, “On the value locality of store instructions,” in *ACM SIGARCH Computer Architecture News*, vol. 28, no. 2, 2000, pp. 182–191.
- [13] H. Meng and J.-P. Wang, “Spin transfer in nanomagnetic devices with perpendicular anisotropy,” *Applied physics letters*, pp. 172506–172506, 2006.
- [14] M. Meterelliyo, H. Mahmoodi, and K. Roy, “A leakage control system for thermal stability during burn-in test,” in *International Test Conference*, 2005, pp. 10–pp.
- [15] A. Nigam *et al.*, “Delivering on the promise of universal memory for spin-transfer torque ram (stt-ram),” in *Proceedings of the International Symposium on Low Power Electronics and Design*, 2011, pp. 121–126.
- [16] G. Panagopoulos, C. Augustine, and K. Roy, “Modeling of dielectric breakdown-induced time-dependent stt-mram performance degradation,” in *Device Research Conference*, 2011, pp. 125–126.
- [17] R. Shbiaa *et al.*, “Reduction of switching current by spin transfer torque effect in perpendicular anisotropy magnetoresistive devices,” *Journal of Applied Physics*, p. 07C707, 2011.
- [18] G. Sun *et al.*, “A novel architecture of the 3d stacked mram l2 cache for cmps,” in *Symposium on High Performance Computer Architecture*, 2009, pp. 239–249.
- [19] G. Sun *et al.*, “Improving energy efficiency of write-asymmetric memories by log style write,” in *Proceedings of the International Symposium on Low Power Electronics and Design*, 2012, pp. 173–178.
- [20] S. A. Wolf *et al.*, “The promise of nanomagnetism and spintronics for future logic and universal memory,” *Proceedings of the IEEE*, pp. 2155–2168, 2010.
- [21] C. Xu *et al.*, “Device-architecture co-optimization of stt-ram based memory for low power embedded systems,” in *Proceedings of the International Conference on Computer-Aided Design*, 2010, pp. 463–470.
- [22] C. Yoshida *et al.*, “A study of dielectric breakdown mechanism in cofeb/mgo/cofeb magnetic tunnel junction,” in *International Reliability Physics Symposium*, 2009, pp. 139–142.
- [23] Y. Zhang *et al.*, “Asymmetry of mtj switching and its implication to stt-ram designs,” in *Proceedings of the Conference on Design, Automation and Test in Europe*, 2012, pp. 1313–1318.
- [24] W. Zhao *et al.*, “Design considerations and strategies for high-reliable stt-mram,” *Microelectronics Reliability*, pp. 1454–1458, 2011.
- [25] P. Zhou *et al.*, “A durable and energy efficient main memory using phase change memory technology,” in *ACM SIGARCH Computer Architecture News*, vol. 37, no. 3, 2009, pp. 14–23.
- [26] P. Zhou *et al.*, “Energy reduction for stt-ram using early write termination,” in *International Conference on Computer-Aided Design*, 2009, pp. 264–268.

<sup>2</sup>The main memory configured to the minimum size needed for execution of selected benchmarks.