Asynchronous Asymmetrical Write Termination (AAWT) for a Low Power STT-MRAM

Rajendra Bishnoi, Mojtaba Ebrahimi, Fabian Oboril and Mehdi B. Tahoori
Chair of Dependable Nano Computing (CDNC), Karlsruhe Institute of Technology (KIT), Karlsruhe, Germany
Email: {rajendra.bishnoi, mojtaba.ebrahimi, fabian.oboril, mehdi.tahoori}@kit.edu

***Abstract*—Spin Transfer Torque (STT) memory is an emerging and promising non-volatile storage technology. However, the high write current is still a major challenge which leads to a huge power consumption of the memory. Due to an inherent torque asymmetry of the Magnetic Tunnel Junction (MTJ) device employed in STT memories, the switching time between parallel to anti-parallel and anti-parallel to parallel magnetization is significantly different. Hence, the write latencies for writing '0' and '1' are also considerably different. In this paper, we propose a technique called *Asynchronous Asymmetrical Write Termination* (AAWT) which utilizes this asymmetrical behavior to terminate the write operations asynchronously and as a result significantly reduces the write power consumption. Furthermore, we present two different AAWT implementations to determine the actual write termination times. The first one makes use of a clock signal and the second one employs a self-timing approach based on an internal delay element. As shown by our experimental results, AAWT can reduce the total write energy by 30 % in average with a negligible area overhead.**

I. Introduction

As the continuous downscaling of CMOS becomes more and more challenging, the research community is spending a great deal of efforts to find feasible alternatives. On the side of random access memory (RAM), nano-magnetic storage devices (MRAM) are very promising candidates to replace the traditional CMOS-based memory solutions. In particular, *Spin Transfer Torque* (STT) memory is gaining significant attention as it is non-volatile, scalable, has low read access times and its endurance can reach to that of SRAM [3, 7, 16].

Despite all these advantages, this technology requires a high current to write data into a memory cell, which is a major challenge for the establishment of STT memory. It is shown that STT-MRAM consumes about 10x more energy per write operation than SRAM [4, 14]. Furthermore, experiments revealed that more than 70 % of the dynamic power is consumed by write accesses [20]. In addition, large access transistors are necessary to meet the high current flow, because of which the integration density is degraded. Finally, a high current through the *Magnetic Tunnel Junction*[1] (MTJ) imposes a severe stress for the memory cell. It leads to not only the time dependent degradation of performance parameters such as tunnelling magneto resistance, write current and write time but also lifetime, as the MTJ oxide is threatened by time dependent dielectric breakdown [12, 17]. Hence, techniques are necessary to reduce the write current.

Several approaches have been proposed to address the issue of high write current and write power at various abstraction layers. However, no existing technique addresses the problem that *unnecessary* current flows under certain conditions related to the asymmetrical write behavior of STT-MRAMs. Due to the inherent torque asymmetry in the MTJ cell, the write times and currents are different for a transition from '0'→'1' and '1'→'0'. In more details, the switching time from parallel (P), i.e. both magnetic layers in the MTJ cell have the same field orientation, to anti-parallel (AP) is significantly larger than that from 'AP'→'P' [19]. Therefore, the word line activation period depends on the 'P'→'AP' switching time as it always covers the worst case scenario. As a consequence, the word line cannot be closed immediately, when the faster 'AP'→'P' transition is finished. Hence, there is an unnecessary high current that flows even after the 'AP'→'P' transition is performed.

In this paper, we propose a scheme called *Asynchronous Asymmetrical Write Termination* (AAWT), which addresses the aforementioned problem. We exploit the fact that the write circuitry can be closed asynchronously as soon as the memory cell is in a stable 'P' configuration. To accomplish this, a "delay signal" is required to determine the write termination time. Therefore, we present two different approaches. The first one is a *clock-controlled* scheme which uses a given clock signal to drive the write termination. The other one is a *delay element* scheme which generates the write termination signal using a self-timing approach based on MTJ cells. As a result, AAWT saves a huge amount of current and hence power, if the memory cell configuration is going to be 'P'. In addition, the area overhead of AAWT is negligible. Our experimental results show that AAWT allows to save in average over 30 % of the total write energy with minimal area overhead (at most 0.9 %) and no timing penalties.

In summary, our contributions to this work are as follows:

- We propose a novel approach to reduce the unnecessary current flowing through an MTJ cell called AAWT.
- We present two different AAWT implementation schemes. The first one uses a clock signal to determine the write termination times. The other one is based on a delay element using MTJ cells. We address the impacts of process variation for both implementation schemes.
- We present a comprehensive analysis of AAWT at both circuit- and architecture-level.

The rest of this paper is organized as follows. In Section II the basics of STT are introduced and related work is discussed. Section III explains the write behavior of a bit-cell and our proposed AAWT technique. Two implementation schemes are described in Section IV. In Section V, the experimental results for AAWT at architecture- and circuit-level are presented. Finally, Section VI concludes the paper.

II. Background

A. Basics of STT-MRAM

The storing devices in Spin Transfer Torque (STT) memories are Magnetic Tunnel Junction (MTJ) cells in which data is stored as a resistance state value. An MTJ device, as shown in Fig. 1(a), consists of two independent ferromagnetic layers separated by a *barrier oxide layer* such as *magnesium oxide* (MgO). One of the two ferromagnetic layers is named as *free layer* where magnetization is freely rotated based on the direction of the current flowing through the cell. The other layer is known as *reference layer* or *pinned layer* whose magnetization is always fixed. When the direction of magnetization of the free layer is parallel (P) to the pinned layer, i.e. the magnetic field orientation in both layers are the same, the MTJ cell has a low

[1] memory component consisting of the two magnetic layers and a barrier oxide in between (see Fig. 1)

978-3-9815370-2-4/DATE14/©2014 EDAA

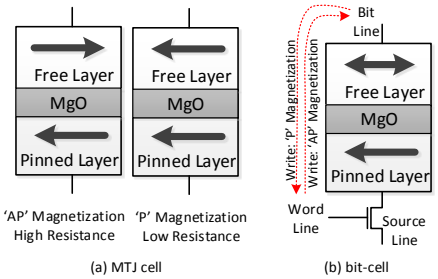


Fig. 1. Spin transfer torque storing device

resistance value. On the contrary, when the direction of magnetization of the free layer is opposite or anti-parallel (AP) to the pinned layer, the MTJ cell has a high resistance value. Depending on the actual memory implementation this high and low resistance values represent either logic '1' and '0' or the inverted values.

An STT-MRAM bit-cell is shown in Fig. 1(b). It has three terminals namely *bit line*, *source line* and *word line* and consists of the aforementioned MTJ device as well as an access transistor. The word line terminal is used to access the required bit-cell during memory operations and the bit line as well as the source line terminals are used for the write and read current flows. The read current is unidirectional and is significantly lower than the write current which is bidirectional and asymmetric, i.e. write time and current are not same for different transitions from one logic state to another. Typically, the switching current of 'P'→'AP' is upto 50 % lager than that of 'AP'→'P' [10].

B. Related Work

To enable the usage of STT-MRAM in low-power application areas, it is necessary to reduce the write current and the write energy. However, only few approaches target this challenge. Early write termination (EWT) is one of these techniques which avoids write operations, if the new value is already stored in the bit-cell [20]. As a result, energy consumption is considerably reduced, since almost the same current flows no matter if the bit-cell value is flipped or not. Two probabilistic design approaches namely, write-then-read with adaptive period and verify-one-while-writing to improve speed and save energy are discussed in [2]. The first one is a read-verify-rewrite technique with an optimal write pulse width and the latter utilizes the asymmetrical behavior to conduct only one write operation. Another architectural-level technique separates the write process for '0' and '1' instead of having a parallel execution [15]. By this means, both write operations can be optimized independently to save power and overcome the challenge of asymmetrical writes in STT memories. Furthermore, there are two circuit-level approaches to reduce write power named bit line voltage clamping and dual source line which are discussed in [8]. In the former technique, the bit-cell is clamped to a lower voltage with the help of a pass transistor which reduces excessive current through the MTJ cell. The later technique uses two access transistors with two source lines to reduce the effective width of the access transistor to deal with excessive currents. Moreover, there is another technique named balance write which lowers the write energy using biasing schemes [9]. In this method, the word line voltage is lowered and a negative voltage is applied to the bit line terminal of the bit-cell.

In summary, the majority of the existing techniques reduce the current flow through the MTJ cell and only the EWT technique tries to minimize the overall time for write current flow. However, EWT does not take care of differential write operations ('0'→'1' or '1'→'0') that are a major contributor to the overall write power, as roughly 25 % of all write operations are differential writes (see Section V). Instead, our proposed technique terminates the write operation asynchronously when the magnetization is going to be in the parallel state and hence also addresses differential as well as non-differential writes.

Please note that all aforementioned techniques are orthogonal to our work and hence can be combined with AAWT.

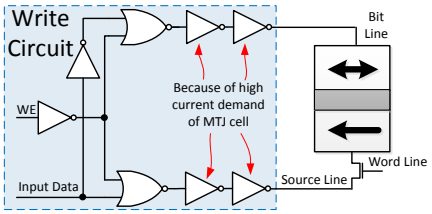
III. Proposed AAWT Technique

In this section the proposed asynchronous asymmetrical write termination (AAWT) technique is presented. In this regard, we explain first the asymmetrical write behavior of MTJ cells using a single bit-cell. Afterwards, the AAWT technique is introduced and we discuss the control circuit which asynchronously closes the write circuitry.

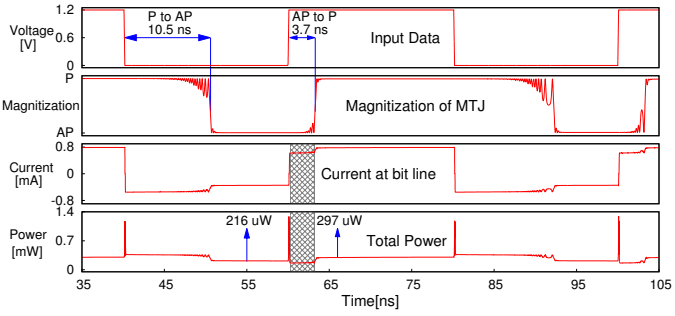
A. Asymmetrical behavior of MTJ cells

The MTJ cell is asymmetric in nature, that means the transition times for 'AP'→'P' and 'P'→'AP' are significantly different. To demonstrate this (write) behavior, we performed SPICE simulations using the experimental setup detailed in Section V-A and the circuit shown in Fig. 2(a). It consists of a single 1T1MTJ[2] bit-cell and only one write circuit. For the experiment the input sequence '1'→'0'→'1'→'0' was applied and the write enable signal (WE) as well as the word line were activated.

The circuit behavior during this evaluation in terms of power, current and magnetic field orientation inside the MTJ cell is shown in Fig. 2(b). As it can be seen, the MTJ conversion from 'P'→'AP' (10.5 ns) takes almost 3x more time than that from 'AP'→'P' (3.7 ns). Therefore, 'P'→'AP' covers the worst case timing scenario and its delay needs to be considered for the memory timing closure. In other words, the word line signal is disabled by considering the 'P'→'AP' timing. In case of an 'AP'→'P' transition, a high current of almost 0.8 mA flows even *after* the transition is completed, as the word line remains active during that period. Because of this unnecessary current, there is a huge unnecessary power consumption. Moreover, this current is even higher (around 150 uA) than that flowing during the 'AP'→'P' transition phase itself, as the MTJ resistance in the



(a) Write circuit diagram driving a single bit-cell



(b) Waveform for write circuit to demonstrate the magnetization of MTJ

Fig. 2. Behavior of bit-cell in case of write accesses

[2] 1T1MTJ = 1 access transistor + 1 MTJ cell

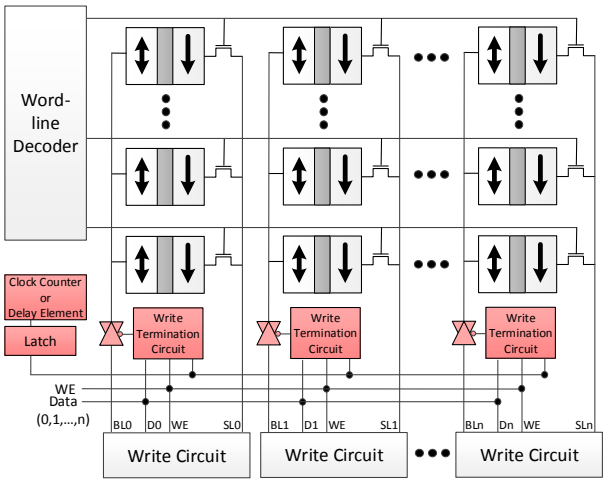
settled 'P' condition is lower than during the actual transition phase (see shaded area in Fig. 2(b)).

B. AAWT Technique

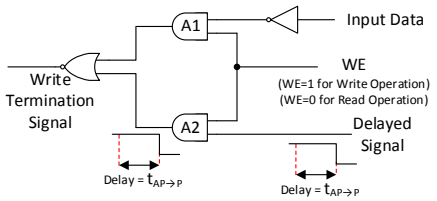
As shown in the previous subsection, the transition from 'AP'→'P' is much faster than that from 'P'→'AP'. Hence, to save current and power, our idea is to terminate the write operation asynchronously as soon as the bit-cell is in a stable 'P' configuration. Therefore, we propose the AAWT technique with which we can cease the current flow after an 'AP'→'P' transition.

The block diagram for the AAWT implementation is shown in Fig. 3(a). Compared to the standard memory implementation, AAWT requires some additional circuitry: *a write termination circuit*, *a delay element* or *a clock counter* to determine the termination time and *a latch*. The write termination circuit is used to close the write circuit after a certain amount of time to save current. The actual point in time, when the write circuit can be closed, is determined by the delayed signal which is generated using the delay element or clock counter. Both schemes are discussed in more detail in the next section.

As shown in Fig. 3(a), the write termination setup needs to be integrated at the column of the memory architecture which controls the termination of the memory write circuit through a transmission switch. The write operations are disabled and enabled when the write termination signals are '1' and '0', respectively. The diagram for the AAWT write termination circuit is shown in Fig. 3(b) and its truth table is given in Table I. The write termination circuit is activated only during write operations when the *Write Enable* (WE) signal is '1'. If, in addition, the input data is '0'[3], the write termination signal becomes '0', irrespective of the state of the delayed signal. Hence, this write operation is not terminated "earlier". Instead, for the input value '1'



(a) Block diagram of AAWT technique



(b) AAWT write termination circuit diagram

Fig. 3. Implementation of the proposed AAWT technique

[3]in this work a logic '0' will be stored using the 'AP' configuration of the MTJ cell

TABLE I. TRUTH TABLE FOR THE OPERATION OF WRITE TERMINATION CIRCUIT USED FOR AAWT IMPLEMENTATION

Write Enable	Input Data	Delayed Signal	Write Termination Signal
0	X	X	1
1	0	X	0
1	1	1	0
1	1	0	1

(here: 'P' configuration), the output of the A1-gate becomes '0' and the output of gate A2 switches from '1'→'0' as the delayed signal is making a '1'→'0' transition after the time required for an 'AP'→'P' write operation. So the write termination signal is '0' for duration of the actual 'AP'→'P' transition and afterwards it switches to '1', i.e. the write circuit will be closed asynchronously. For read operations, when the WE signal is '0', the write termination signal is also '1'. However, as in this case the write circuitry is not active, the write termination signal has no effect. Hence, this technique terminates only write operations resulting in the 'P' configuration and it does not disturb any read or write operation.

IV. IMPLEMENTATION OF AAWT

The most crucial part of the AAWT implementation is the creation of the delayed signal which is required to asynchronously terminate the write circuitry, i.e. whenever the bit-cell is going to be in the 'P' configuration. As explained in the previous section, the delayed signal has to make a '1'→'0' transition to activate the write termination in this case. To save power, this transition should be very close to those of bit-cells making an 'AP'→'P' transition (here: 3.7 ns). However, due to process, voltage and temperature (PVT) variations the real switching delay of the MTJ cells can differ from this value. Hence, the transition of the delayed signal has to include margins to account for PVT variations in the bit-cells. In this section, we present two schemes to create this signal considering PVT variations. One generates the signal through a fine-grained clock and the other uses a delay element designed with MTJ cells.

Please note that for 'AP'/'P'→'AP' transitions the write circuitry is always closed by the clock edge, no matter which implementation is chosen for the asynchronous termination.

A. Clock-Controlled Implementation

The first implementation exploits the clock signal that is fed to the memory to create the delayed signal. This clock signal drives a counter, whose outputs are ORed together to create the delayed signal. Therefore, the counter counts down to 0 and starts with the number of clock cycles that are required to complete an 'AP'→'P' transition. For example, if the clock frequency is 1 GHz, i.e. the clock period is 1 ns, it takes at least four cycles to complete an 'AP'→'P' transition, given that an 'AP'→'P' transition takes 3.7 ns ($\lceil \frac{3.7}{1.0} \rceil = 4$). In case of 10 % variation, the counter will count down from 5 to 0, to ensure that no write failures occur. After 5 cycles the counter result will be 0 and the delayed signal will make a transition from '1' to '0', which will initiate the write termination.

To reduce the power consumption of this approach, a clock gating technique is used. By this means dynamic power consumption during read accesses and late cycles of write accesses during which counting is not required is avoided. In other words, this counter is clocked when a write access is triggered and its clock is gated after instructing the write termination signal.

The main advantage of this scheme is that the clock signal is already available and hence only a light-weight counter needs to be added to the memory unit. Moreover, the clock frequency will be usually around 1 GHz to allow fast read operations (which take 1 ns

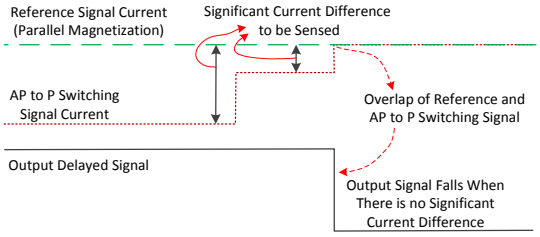
or less [3]). Hence, a three-bit counter should be usually sufficient. In terms of variation this approach needs only to margin for the bit-cell variations as the clock signal itself is very accurate, which will allow significant energy savings. However, the energy savings of the clock-controlled implementation are very sensitive to the clock period, as we will show in Section V. If the clock period is too long the efficiency of this scheme decreases significantly.

An alternative option to the counter-based implementation is to create an artificial clock signal outside of the STT-MRAM, which is then fed into the memory unit to drive the write control circuitry. However, in this case, additional memory pins and a clock multiplier/divider are required for this implementation, which is typically more expensive than the counter-based version.

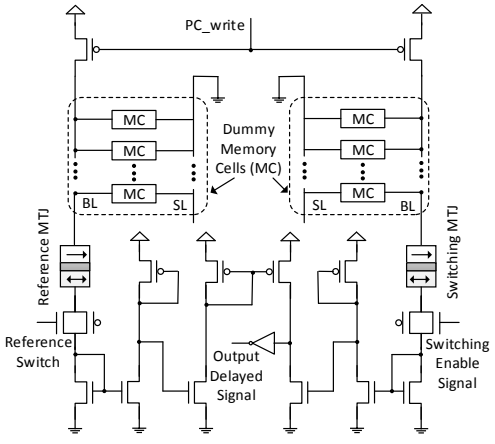
B. MTJ-based Delay Element

An alternative implementation of AAWT uses a self-timing approach based on a special delay element, which is built with an MTJ cell. The flip delay of this cell is used to match the bit-cell delays.

An MTJ cell stores a value in terms of a resistance state and its resistance value changes after the flip. Therefore, the current flowing though the MTJ device also changes after the flip. During an 'AP'→'P' transition, the MTJ cell switches from a high resistance state to a low resistance state. Hence, only a small current flows through the MTJ cell until the flip occurs, and it increases immediately after the flip. This significant difference ($\approx$ 150 uA) can be sensed to detect the flip and to drive the write termination signal. Therefore, a special delay element using a *current comparator* with two MTJ cells is implemented. The first MTJ cell acts as a reference cell which is always in the parallel state (low resistance) and the other one needs to switch from 'AP'→'P'. Hence, at the beginning of the transition, the currents flowing through the two MTJ cells are different, while they become equal after the magnetization of the switching MTJ has changed (as shown in Fig. 4(a)). As soon as both currents are at the same level, the delayed signal (output) will switch from '1' to '0'.



(a) Current behavior of MTJ cell to design delay element



(b) Delay element circuit diagram

Fig. 4. Delay element using MTJ cell

The circuit diagram for the delay element implementation is shown is Fig. 4(b). It works in three phases, as explained below:

- During write operations the *PC_write* signal is activated after the positive clock edge, and both the *reference switch* and the *switching enable signal* are enabled after the activation of the word line. When the *PC_write* signal is active, the current flows through both MTJ cells.
- Then, the current is copied to the "inner" branches using *current mirrors*. Therefore, a proper sizing of the transistors ensures so that the required current is mirrored.
- Finally, the current difference is converted into a voltage value and a stable output, the delayed signal, is taken through an inverter. This delayed signal is '1' as long as the switching MTJ cell is not in the 'P' state, and is '0' otherwise.

To ensure that always the delay of an 'AP'→'P' transition is sensed, it is mandatory that the switching MTJ cell is always in the 'AP' state at the beginning of a write cycle. Therefore, a transmission switch is required to disable the *switching enable signal* and to change the state (back from 'P' to 'AP') of the switching MTJ cell after the delayed signal switched to '0'. This "second" write operation ('P'→'AP') to the switching MTJ cell is performed during the "non-operational" memory phase, during which data and addresses are updated. In contrast, the first write operation ('AP'→'P') is issued in parallel to all bit-cell write operations during the "operational" memory phase. Since both phases have typically the same length, this "additional" write operation to the switching MTJ cell causes no performance penalty. However, as always two writes are performed to the switching MTJ cell, the endurance of this MTJ cell can be limited, although the latest STT-MRAMs promise a similar endurance to SRAM [3]. Therefore, if the endurance is critical, we suggest to put several switching MTJ cells next to each other that are used in a round-robin fashion. Besides reliability, this "additional" write operation also affects the energy overhead. In fact, due to these write operations the energy overhead increases by 0.3 %.

The waveforms for the memory behavior in terms of magnetization, current and power using a delay element to generate the delayed signal are shown in Fig. 5. It can be observed from the figure that the magnetizations of the bit-cell and the delay element change almost at the same time. Nevertheless, the delay element switches slightly later, which is necessary to avoid that the write process is terminated too early. Once the delay element magnetization changes from 'AP'→'P',

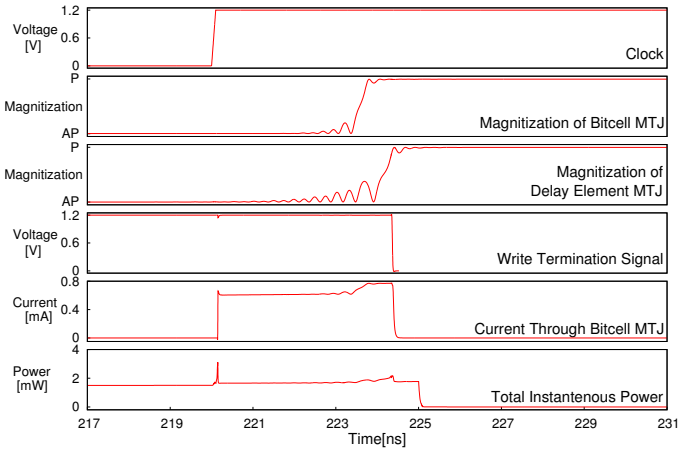


Fig. 5. Waveform for AAWT implementation using the proposed delay element (ideal conditions)

the write termination signal switches immediately from '1' to '0'. Hence, the write circuit is closed at that time and no further current flows through the bit-cell. As a result, there is a sudden drop in instantaneous total power consumption (from 1.79 mW to 1.5 uW).

In order to account for PVT variations, this approach has to consider two different margins. First, a timing margin, e.g. in the form of an inverter chains needs to be included for variations in the bit-cells. Second, the variations of the delay element itself need to be covered by a margin. To minimize the operating condition variability on timing and power of the delay element, we added dummy bit-cells to get similar loading effects as that of the bit-cell array [1]. Hence, this approach can require larger margins and comes with higher implementation costs than the clock-controlled scheme, but is independent from the clock period. Moreover, for long clock periods this technique is more efficient than the clock-controlled implementation (see Section V). This is due to the fact that this scheme always terminates the write operation at the same time, while in the clock-controlled method, the termination time depends on the clock cycle period.

V. Experimental Setup and Results

To evaluate the effectiveness of the proposed technique, a detailed model of AAWT is implemented at circuit-level. Based on the data obtained from the circuit-level analysis, an abstract model of the memory is employed in an architecture-level implementation and its effect on the energy consumption for various workloads is evaluated.

A. Circuit-Level Implementation

The framework from [5] is employed for the circuit-level implementation. We have used a barrier thickness and energy barrier of 1.15 nm and 51 KT, respectively. In our CMOS circuit implementation, we have used TSMC 65 nm general purpose models and supply voltage of 1.2 V.

The implementation of the clock-controlled scheme was straightforward and is done by implementing the clock counter inside the STT-MRAM. For the implementation of the delay element, we used exactly the same parameters for the MTJ cell as for the bit-cell MTJs. We assumed that the variation of the oxide thickness and the cross-sectional area has a standard deviation of 2 % and 5 % of their mean, respectively [18]. Considering these, we obtained 7 % variation in the 'AP'→'P' write delay (3.7 ns). Therefore, a margin of 260 ps and 520 ps are considered in our AAWT implementation with the clock-controlled and with the delay element scheme, respectively. The margin for the latter is larger, since the variation of the delay element itself needs to be considered in addition to the bit-cell variations.

Fig. 6 shows the energy savings for both AAWT implementations with and without margins. This figure is created for different clock periods with the assumption that each type of write operation has an occurrence probability of 25 %. Obviously, the energy saving trends are clock dependent. This is due to the following facts: 1.) Whenever the clock period is a multiple of the 'AP'→'P' delay, the clock-controlled scheme reduces the power at the exact time this write operation terminates. Otherwise, this scheme consumes power until the closest clock edge resulting in less reduction. 2.) Although the delay element scheme closes 'AP'→'P' independent of the clock edge, it still uses the clock edge to close 'P'→'AP' write operations. Hence, the length of the clock period also influences the energy demand of these write operations. This fact is also valid for the clock-controlled based AAWT implementation. Furthermore, Fig. 6 shows that the margins considered for process variation reduce the energy savings. Independent of the margins is the observation, that the

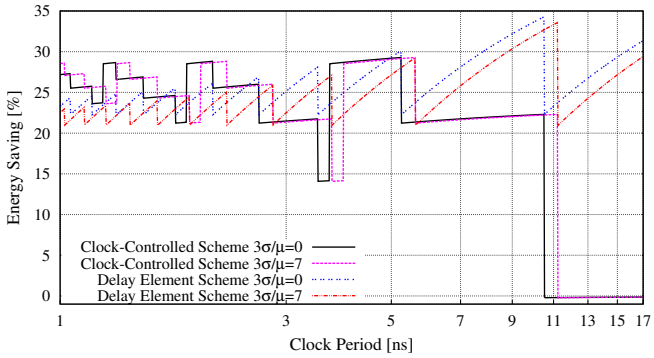


Fig. 6. Overall energy saving for different clock periods with assumption of 25 % of each write operations

clock-controlled scheme offers higher savings when the clock period is small as the required delay for the write termination time can be modeled more accurately. However, the clock-controlled scheme has a negative impact on energy saving for longer clock periods i.e. when it is greater than the maximum write delay (here, 10.5 ns). In such cases, the delay element scheme is more effective in terms of energy saving as it is not dependent on the clock period for the generation of the write termination signal.

We also evaluated the area costs at circuit-level by extrapolating the array sizes and considering the area of additional periphery circuits. According to the obtained results, there is just a negligible area overhead of around 0.9 % and no timing penalties for the AAWT implementation using the delay element. If the clock-based implementation is chosen, the area overhead is even just 0.7 %. In addition, the overhead decreases with increasing bit per word count.

B. Architecture-Level Implementation

In order to show the efficiency of the proposed technique in a real system, an experiment is conducted on a system-on chip including a Leon2 processor and a 32 KByte main memory unit. The processor core is synthesized to obtain the minimum clock period which was 1.24 ns (806 MHz). Considering the delay of the STT-MRAM for read accesses and also the delay from STT-MRAM input/output pins to corresponding registers in the processor core, it is observed that the STT-MRAM can operate with the same clock period. In this manner read and write accesses require 1 and 9 cycles, respectively. Therefore, the main memory is implemented with an abstract model of the STT-MRAM equipped with the proposed AAWT technique. A simple code for analyzing the write operations is added to the description of this memory. This code does a bit-wise comparison between the new and old data and gathers the statistics for differential and non-differential write accesses.

The write energy for a single bit-flip for 8K×32 bit using a clock period of 1.24 ns is shown in Table II. We have taken process variation into account for generating these results. For the delay element implementation, around 60 % of the total 'AP'→'P' energy is saved, since the write circuitry is closed asynchronously which

TABLE II. Energy in pJ for standard STT-RAM and AAWT for 8K×32 bit memory with clock period of 1.24 ns

Magnetization	Standard	Delay Element		Clock-Controlled	
	Energy(pJ)	Energy(pJ)	Reduction	Energy(pJ)	Reduction
'AP'→'P'	914.88	366.83	59.90%	373.76	59.14%
'P'→'AP'	1051.84	1108.27	-5.36%	1052.16	-0.03%
'P'→'P'	664.32	310.19	53.31%	291.20	56.16%
'AP'→'AP'	917.76	945.71	-3.04%	918.08	-0.03%
Average	887.20	682.75	**23.04%**	658.80	**25.74%**

TABLE III. ENERGY REDUCTION FOR LEON2 PROCESSOR MAIN MEMORY WHEN IT IS RUNNING MIBENCH WORKLOADS

Benchmark	Cycles	Write Type Occurrence [%]				Energy Saving	
		P→P	P→AP	AP→P	AP→AP	Delay Element	Clock-Controlled
basicmath	500M+	57.4%	11.7%	11.7%	19.2%	32.4%	35.2%
bitcounts	77M	71.5%	4.7%	4.7%	19.1%	36.5%	39.3%
crc32	500M+	37.5%	12.4%	12.4%	37.6%	21.9%	24.7%
fft	500M+	61.2%	11.1%	11.1%	16.7%	34.3%	37.1%
qsort	6M	34.3%	18.4%	18.3%	29.1%	23.7%	26.5%
sha	500M+	47.2%	14.3%	14.2%	24.2%	28.2%	31.0%
stringsearch	3M	69.5%	8.4%	7.8%	14.4%	38.3%	41.1%
Average		54.1%	11.6%	11.4%	22.9%	30.8%	33.5%

stops unnecessary current flow after the magnetization flip. Moreover, for the case of 'P'→'P' write operations, around 53 % of the total energy is saved, since AAWT can close the write circuit earlier in this case as well. However, for 'P'→'AP' and 'AP'→'AP' write operations the energy demand increases by around 5 % and 3 % due to the additional AAWT circuitry, compared to the standard implementation. Nevertheless, as the savings in the first two cases are much higher than the costs in the last two cases, there is a significant overall energy reduction. Similar savings can be obtained using the clock-controlled AAWT implementation. However, since the energy overhead is smaller than that of the delay element scheme, the overall savings are slightly higher (26 % vs 23 %). Despite less overhead and margins, the energy reduction for 'AP'→'P' is slightly less than that of the delay element scheme, as the write termination for this write operation is dependent on the clock period in case of the clock-controlled implementation, while it is clock-independent for the delay element scheme.

To evaluate the overall energy savings in real applications, we used several memory and computational intensive workloads from Mibench [6] together with the aforementioned Leon2-based platform. Based on the statistics shown in Table II, the write energy for both standard and the AAWT implementations could be easily extracted. For each evaluated workload, we have skipped its initialization phase and ran at most 500M cycles afterwards. Table III shows the summary of the results obtained form the architecture-level analysis. For each switching type, its occurrence rate is multiplied with its energy and the overall write energy for the standard as well as the AAWT implementations are obtained and the total energy reductions are reported. As expected, both implementations of the AAWT technique result in significant energy savings compared to the standard implementation, i.e. the average energy reduction for clock-controlled and delay element implementations were 33.5 % and 30.8 %, respectively.

In addition, the energy saving gap between AAWT and the standard memory scales with the bit per word count as illustrated in Fig. 7. This figure shows the results for the AAWT implementation using the delay element scheme. Similar trends can be observed for the clock-controlled scheme. This trend is due to the fact that in our proposed approach, a single delay element/counter drives all write circuits.

VI. SUMMARY AND CONCLUSIONS

For shrinking technologies, STT-MRAM is a promising non-volatile storage candidate because of its advantageous features. However, the write current is very high which leads to a high dynamic power consumption of the memory. In this work, we proposed the Asynchronous Asymmetrical Write Termination technique to reduce the unnecessary current for write operations that result in a parallel magnetization of the bit-cell. Therefore, these write operations are terminated asynchronously. For this purpose, we also proposed two schemes to generate the required write termination signal. The first approach generates the termination signal using a counter in combination with a fine grained clock. The second technique employs a delay element which is designed with MTJ cells. In both cases, AAWT reduces that average write energy by over 30 % with negligible area overhead (around 0.9 %) and no timing penalties.

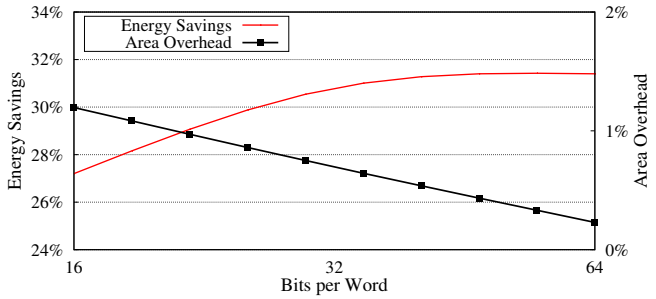


Fig. 7. Energy savings and area overhead for the proposed AAWT technique compared to a standard STT-RAM for different word lengths

VII. 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] B. S. Amrutur and M. A. Horowitz, "A replica technique for wordline and sense control in low-power sram's," *IEEE Journal of Solid-State Circuits*, pp. 1208–1219, 1998.
[2] X. Bi *et al.*, "Probabilistic design methodology to improve run-time stability and performance of stt-ram caches," in *ICCAD*, 2012, pp. 88–94.
[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 *HPCA*, 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 *DAC*, 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 *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 *IEDM*, 2005, pp. 459–462.
[8] Y. Kim *et al.*, "Write-optimized reliable design of stt mram," in *ISLPED*, 2012, pp. 3–8.
[9] D. Lee, S. K. Gupta, and K. Roy, "High-performance low-energy stt mram based on balanced write scheme," in *ISLPED*, 2012, pp. 9–14.
[10] C. Lin *et al.*, "45nm low power cmos logic compatible embedded stt mram utilizing a reverse-connection 1t/1mtj cell," in *IEDM*, 2009, pp. 1–4.
[11] A. Nigam *et al.*, "Delivering on the promise of universal memory for spin-transfer torque ram (stt-ram)," in *ISLPED*, 2011, pp. 121–126.
[12] 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.
[13] R. Sbiaa *et al.*, "Reduction of switching current by spin transfer torque effect in perpendicular anisotropy magnetoresistive devices," *Journal of Applied Physics*, p. 07C707, 2011.
[14] G. Sun *et al.*, "A novel architecture of the 3d stacked mram l2 cache for cmps," in *HPCA*, 2009, pp. 239–249.
[15] G. Sun *et al.*, "Improving energy efficiency of write-asymmetric memories by log style write," in *ISLPED*, 2012, pp. 173–178.
[16] S. A. Wolf *et al.*, "The promise of nanomagnetics and spintronics for future logic and universal memory," *Proceedings of the IEEE*, pp. 2155–2168, 2010.
[17] C. Yoshida *et al.*, "A study of dielectric breakdown mechanism in cofeb/mgo/cofeb magnetic tunnel junction," in *IRPS*, 2009, pp. 139–142.
[18] Y. Zhang, X. Wang, and Y. Chen, "Stt-ram cell design optimization for persistent and non-persistent error rate reduction: a statistical design view," in *ICCAD*, 2010, pp. 471–477.
[19] Y. Zhang *et al.*, "Asymmetry of mtj switching and its implication to stt-ram designs," in *DATE*, 2012, pp. 1313–1318.
[20] P. Zhou *et al.*, "Energy reduction for stt-ram using early write termination," in *ICCAD*, 2009, pp. 264–268.