Exploring MRAM Technologies for Energy Efficient Systems-On-Chip

Sophiane Senni, Lionel Torres, Gilles Sassatelli, Abdoulaye Gamatié, Bruno Mussard

Exploring MRAM Technologies for Energy Efficient Systems-On-Chip

Sophiane Senni, Lionel Torres, Gilles Sassatelli, Abdoulaye Gamatie, and Bruno Mussard

Abstract—**It has become increasingly challenging to respect Moore's well-known law in recent years. Energy efficiency and manufacturing constraints are among the main challenges to current integrated circuits today. The energy efficiency issue is mainly due to the high leakage current from the CMOS transistors that are used to build almost all logic devices. As a result, performance is limited to a few gigahertz due to high power dissipation. A significant proportion of total power is spent on memory systems due to the increasing trend of embedding volatile memory into systems-on-chip devices. New non-volatile memory technologies are one possible way to solve the energy efficiency issue. Among these technologies, magnetic memory is a promising candidate to replace current memories since it combines non-volatility, high density, low latency and low leakage. This paper describes an approach to obtain large, fine-grained exploration of how magnetic memory can be included in the memory hierarchy of processor-based systems by analyzing both performance and energy consumption.**

Index Terms—**Memory hierarchy, MRAM, non-volatile memories, systems-on-chip, VLSI.**

I. Introduction

INTENSIVE investigations are underway to resolve the most critical problem of current nano-electronic systems: energy efficiency. Major issues encountered in today's integrated circuits (ICs) include high leakage current, performance saturation, increased device variability and process complexity. For battery-powered applications, energy consumption is unquestionably the most critical metric. In dynamic mode, fast switching at low power is targeted. In static mode, low leakage power is desired. Current systems embed volatile devices such as flip-flops, static random access memories (SRAM) and dynamic random access memories (DRAM), which lose information when powered off. Circuit design techniques, such as clock and power gating, are currently used to reduce the power consumed during standby mode. Although these techniques can reduce the consumption of static energy, it is not so easy to manage the total power consumption. First, systems-on-chip (SoC) are becoming more and more complex with the increasing number of transistors per die. Regarding on-chip memories, as they are mostly volatile, several power modes are required such as active, standby, retention, deep sleep and power down for various application demands [1]. The possibility of integrating non-volatile memories (NVM) would greatly facilitate the power-saving techniques implementation.

That is why one possible way to overcome the energy efficiency issue is non-volatile SoCs using non-volatile devices. In this case, a complete power down is possible with no loss of data or logic states. A promising candidate for non-volatile SoCs is magnetic memory (MRAM) based on magnetic tunnel junction (MTJ) component. Both academia and industry regard MRAM as a suitable technology to become a universal memory as it combines low leakage, high density and has low access time compared to other existing and emerging NVMs such as FLASH, Phase-Change RAM (PCRAM) or Resistive RAM (ReRAM). However, despite the many attractive features of MRAM, two challenges are still under intensive investigation. First, MTJ switching requires a significant amount of current. Second, even if MTJ is orders of magnitude faster than conventional NVMs, e.g., FLASH or embedded FLASH, it is slower than typical 6-transistor-based SRAM, especially for write operations. However, Toshiba recently published very encouraging results [2] on a perpendicular MTJ technology with an access time of 3 ns and read/write bit energy that is almost equivalent to SRAM. MRAM has attracted many researchers, and many studies have been conducted to evaluate integration of MRAM in the memory hierarchy of processor architecture.

A. Contribution of This Paper

We present a fine-grained exploration to evaluate the performance and energy impacts of including MRAM in the memory hierarchy of processor architecture. The exploration is essentially focused on the L2 cache. Previous papers limited their analysis of the performance and energy of MRAM-based cache to a direct comparison with that of SRAM-based cache. In other words, they did not sufficiently analyze the memory traffic to better understand the gain or loss in performance/energy using MRAM-based cache. Although comparing direct performance and energy enables evaluation of the potential of MRAM-based cache, analysis of the results remains too superficial, especially for complex systems such as multi-core architectures. In the present study, useful information about the memory traffic are extracted, such as the cache miss rate and the cache bandwidth. This information is monitored over time to better understand the

Manuscript received June 26, 2015; revised September 02, 2015 and October 31, 2015; accepted December 03, 2015. Date of publication April 07, 2016; date of current version September 09, 2016. This paper was recommended by Guest Editor S. Ghosh.

S. Senni and B. Mussard are with the Crocus Technology, 38025 Grenoble, France (e-mail: ssenni@crocus-technology.com; bmussard@crocus-technology.com).

L. Torres, G. Sassatelli, and A. Gamatie are with the Microelectronics Department of Montpellier Laboratory of Informatics, Robotics and Microelectronics (LIRMM), UMR 5506, University of Montpellier–CNRS, 34095 Montpellier, France (e-mail: torres,sassatelli,gamatie@lirmm.fr).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JETCAS.2016.2547680

behavior of the workloads in terms of memory access. Hence, a clear vision of the influence of these parameters on performance and energy is possible. In addition, a set of data including the read/write ratio, the static/dynamic energy ratio and L1/L2 access ratio are also extracted to better analyze the impact of the different read/write latencies, of the high dynamic energy and of the low leakage of MRAM, thus enabling fine-grained analysis of the performance and the total energy consumption of MRAM-based cache.

A preliminary version of this work was published in [3]. But, this paper gives a more detailed circuit-level and architecture-level analysis of MRAM-based cache memory, including a detailed state-of-the-art of existing MRAM technologies. In addition, another benchmark suite is considered for architecture-level evaluation. The main contributions of this work are as follows.

1) A state-of-the-art review of all existing MRAM technologies highlighting for each one the advantages and limitations compared to its competitors including the scaling aspects. There are not so many papers with a detailed overview and comparison between all existing MRAM technologies.
2) A fine-grained performance/energy analysis of MRAM-based cache extracting important information on memory hierarchy activity, including the cache miss rate, the cache bandwidth, the read/write ratio, the static/dynamic energy ratio, the ratio of the number of accesses between different levels of cache (e.g., L1 and L2).

It is planned to extend this study to evaluate other levels of the memory hierarchy such as main memory and scratchpad memory. In the present study, the exploration flow only supports evaluation of cache memory. The rest of this paper is organized as follows: Section II provides basic information on MRAM, then describes and compares current MRAM technologies. Section III describes the NVM exploration flow used in this paper to allow the large fine-grained evaluation of including MRAM in the memory hierarchy of processor architecture. Section IV analyzes and compares MRAM and SRAM caches at circuit level. Section V explores MRAM-based cache at architecture level for L2 as last-level-cache (LLC). Section VI discusses related work on MRAM-based cache memory. Section VII concludes this paper.

II. MRAM Technologies: State-of-the-Art

A. Basics

A MRAM bit is a MTJ consisting of two ferromagnetic layers separated by a thin insulating barrier. The information is stored as the magnetic orientation of one of the two layers, called the free layer (FL). The other layer, called the reference layer (RF), provides the fixed reference magnetic orientation required for reading and writing. The tunnel magnetoresistance (TMR) effect [4] causes MTJ resistance to depend significantly on the relative orientation of the two magnetic layers: the antiparallel state provides much larger resistance than the parallel state. It enables the magnetic state of the FL to be sensed thanks to a current flowing through the MTJ. Hence, stored information can be read. Five methods have been proposed to switch the orientation of the FL: toggle [5], thermally assisted switching (TAS) [6], spin transfer torque (STT) [7], voltage-induced switching and the most recent method is called spin orbit torque (SOT) [8].

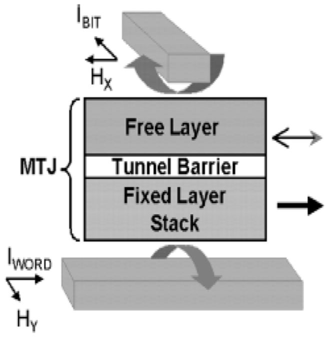


Fig. 1. Conventional.

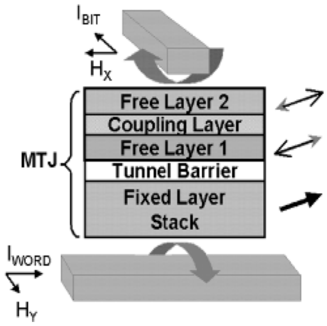


Fig. 2. Toggle MRAM.

A conventional MRAM, shown in Fig. 1, uses a simple way to program the MTJ where sufficient magnetic field is generated thanks to a combination of two current flows applied simultaneously through a row and a column of an MTJ array. Two problems arose with this method. First, large current is needed to generate sufficient magnetic field to reverse the magnetization of the FL. Second, this approach suffers from selectivity problem: some of the bits sharing the same row or column of the cell being programmed might be exposed to sufficient magnetic field and be switched unintentionally. This effect is one consequence of process variability. The magnetic field necessary to reverse the magnetization is not exactly the same for all the bits [9].

B. Toggle

This technology is currently commercialized by Everspin.[1] Fig. 2 illustrates a standard toggle MRAM MTJ. A CMOS access transistor provides a current through the MTJ needed for the read operation. Each MTJ is located at the intersection of two conductive lines, I_{WORD} and I_{BIT} in Fig. 2. Toggle MRAM was proposed to deal with the selectivity problem observed in the standard MRAM. The toggle MRAM adds a second FL and an anti-ferromagnetic coupling layer above the first FL, as shown in Fig. 2. In addition, a specific timing sequence of the write-current pulses is used to switch only the MTJ at the intersection

[1] Available online: http://www.everspin.com/

of the conductive lines [10]. These changes improve the stability of the magnetic orientation of the bit cell and avoid the selectivity issue. Unlike other MRAM technologies, the toggle MRAM scheme does not drive the bit cell to a predetermined state, instead it always reverses the current magnetization of the FL. As a result, a read of the current state of the MTJ is required before a write if the opposite state is desired.

The toggle MRAM has certain limitations. Although it resolves the selectivity issue of the conventional MRAM, it still needs a significant amount of current to switch the bit cell, thereby limiting the upper bound of the write speed. Moreover, the amount of current needed for writing remains almost the same even when the size of the bit cell is reduced. Consequently, the selectivity issue can appear again when scaling the MTJ. In addition, as the switching current does not shrink scaling the technology node, the area of the peripheral circuits of the MTJ array also remains the same, thus limiting the density [9]. Toggle MRAM is not predicted to be suitable at nodes less than 90 nm.

C. *Thermally Assisted Switching*

The aim of the TAS concept was to improve the downsize scalability of MRAM. The concept was developed by the Spintec laboratory and TAS-based devices are commercialized by Crocus Technology.[2] TAS-based MTJ uses an anti-ferromagnetic layer to block the magnetic orientation of the FL under a threshold temperature. To switch the bit cell, a select transistor provides a flow of current to heat the MTJ above the blocking temperature thereby enabling storage of new information thanks to application of a magnetic field. Heating the FL allows TAS-MRAM to use a smaller magnetic field and hence less current than toggle MRAM to write the bit cell, since a single conductive line is sufficient to generate the required magnetic field. Blocking the FL's state using a coupling anti-ferromagnetic layer also significantly improves data stability, even scaling the technology node. As a result, TAS-MRAM makes it possible to reduce the switching energy while ensuring excellent data retention. This new method also solves the selectivity issue, since the MTJ has to be heated before writing.

Fig. 3 shows a complete TAS write operation. Assuming the MTJ stores a "0" state (parallel state), the first step in the TAS method is to heat the FL by flowing a current through the MTJ to reach the blocking temperature (heating step in Fig. 3). The second step is to generate an external magnetic field to switch the FL while heating the MTJ (switching step in Fig. 3). Once the FL switches to the "1" state, the CMOS transistor responsible for the heating process is switched off whereas the MTJ remains under the external magnetic field (cooling step in Fig. 3).

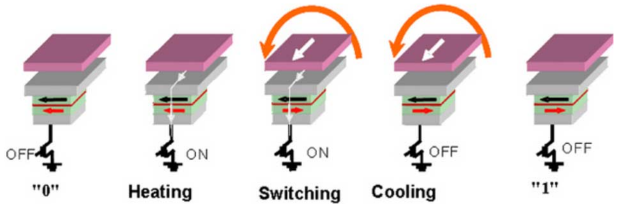
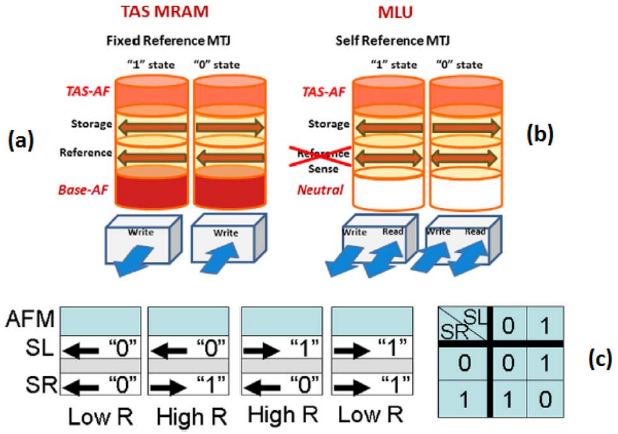


Fig. 3. Thermally assisted switching MRAM [11].



SR \ SL	0	1
0	0	1
1	1	0

Fig. 4. Magnetic logic unit. (a) Magnetic stack of TAS-MRAM. (b) Magnetic stack of MLU. (c) Virtual XOR logic gate of MLU [11].

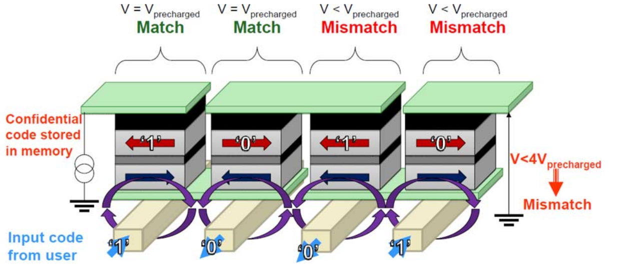


Fig. 5. Match-in-place principle.

Crocus technology designed another implementation of TAS-based MTJs, called Magnetic Logic Unit (MLU) [12], in which the reference layer (RL) is replaced by a self-reference layer (SRL). As a result, the SRL can easily be switched by applying an external magnetic field because the magnetic orientation of the SRL is not fixed like that of the RL, as shown in Fig. 4(b). While the write scheme remains the same, the read operation is quite different. In this case, reading consists of two steps: the MTJ resistance is measured while the magnetization of the SRL is set in one direction. The MTJ resistance is then measured again when the magnetization of the SRL is reversed to the opposite direction. The resistance variation between the two measurements provides information about the magnetization of the FL. The new approach increases the read time, but tolerance to process variation is clearly improved since each bit cell is self-referenced. Moreover, this approach significantly reduces reading errors. Usually, the two resistance states need to be well separated, which can lead to manufacturing problems when scaling the technology node. For MLU, using the difference in the two states for reading is not sensitive to this manufacturing problem.

The innovative MLU concept also leads to another interesting feature: the device can also act as an exclusive-OR logic gate (XOR). Assuming that the two magnetic layers are the inputs and the MTJ's resistance is the output, the truth table of a XOR logic gate can be built [Fig. 4(c)]. This makes MLU a particularly useful component of security applications. An example of application introduced by Crocus technology is the match-in-place [12], shown in Fig. 5. If the current direction applied on the

[2] Available online: http://www.crocus-technology.com/

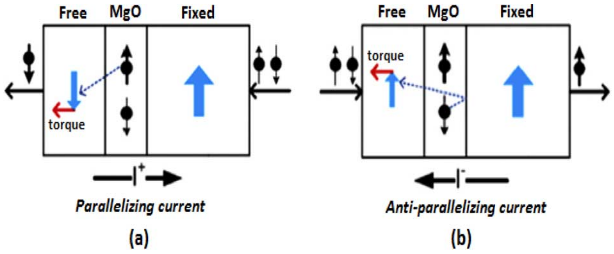


Fig. 6. Spin transfer torque effect: (a) illustration of the transition from an antiparallel to a parallel state, and (b) the transition from a parallel to an antiparallel state [7].

field line during a read operation is considered as an information input, the device can use its XOR logic capability to compare in situ the data stored in memory with the input and control if there is a match or a mismatch comparing the output resistance with a reference. Other possible applications of MLU are: content-addressable memory, NOR-MRAM, NAND-MRAM [11].

Although the structure of TAS-MRAM means it has better scalability than toggle-MRAM, TAS-MRAM needs a non-negligible time to complete its write operation due to the heating/cooling processes. Moreover, since an external magnetic field is used to switch the MTJ, the amount of current is still high, even if it is lower than for the toggle. One possible way to reduce the switching energy is to combine the TAS method with the STT effect. TAS-MRAM is expected to be scalable down to 45 nm [11]. However, the aim of combining TAS with STT is to further improve scalability.

D. Spin Transfer Torque

STT-MRAM appeared with the need to reduce the switching energy consumption of MRAMs. Unlike the previous MRAMs, which use an external magnetic field to program a bit cell, STT-MRAM write operations are based on another physical phenomenon to switch the magnetic orientation of the FL called STT. The idea is that the FL can be switched by direct transfer of the spin angular momentum from spin-polarized electrons. In this way, a highly spin-polarized current flowing through the MTJ causes a "torque" applied by the injected electron spins on the magnetization of the FL. Applying sufficient current will cause sufficient torque to switch the bit cell, thereby enabling information to be written. Fig. 6 depicts the STT effect.

Fig. 6(a) shows the transition from an antiparallel to a parallel state. In this case, electrons go through the fixed layer first, and the fixed layer acts as a polarizer. Thus, electrons are spin-polarized in the magnetic orientation of the fixed layer. Once the insulating barrier (MgO) is crossed, the spin-polarized electrons exert torque on the magnetization of the FL until a magnetic orientation reversal occurs. A similar effect is depicted in Fig. 6(b) for the transition from a parallel to an antiparallel state. In this case, electrons go through the FL first. While the majority of the electrons will be spin-polarized in the magnetic orientation of the FL, a minority of electrons will still be spin-polarized in the opposite direction of the FL. These minority electrons will be reflected at the barrier interface and will exert torque on the magnetization of the FL.

Two kinds of magnetization of the magnetic layers can be found in STT-MRAM: in-plane and perpendicular. In-plane magnetization is also used in toggle-MRAM and TAS-MRAM, in which the magnetic orientation is parallel to the plan of the MTJ, whereas in perpendicular magnetization, the magnetic orientation is perpendicular to the plan of the MTJ. Perpendicular STT-MRAM was introduced to further reduce the switching current of the MTJ and to improve scalability.

State-of-the-art showed that STT-MRAM read access time is similar and sometimes better than its SRAM equivalent [13]–[15]. Concerning write operations, despite the fact that STT-MRAM considerably reduces switching energy compared to the previous MRAM technologies, some limitations were observed. Some of them were mitigated or eliminated while others still remain.

First, read and write operations use the same path, which can lead to unexpected writes when reading is underway, particularly with advanced technology nodes. To mitigate this issue, a solution was proposed at device level designing a three-terminal dual-pillar MTJ structure with two spatially and electrically independent ports for writes and reads [16].

Second, the current needed to switch the MTJ from the parallel to the antiparallel state (and vice-versa) is not symmetrical [17]. Switching from a parallel to an antiparallel state requires more current than the reverse. This is because switching from an antiparallel to a parallel state is performed by spin-polarized electrons going through the MTJ (majority of the electrons), whereas switching from a parallel to an antiparallel state is performed by reflected spin-polarized electrons (a minority of the electrons). A solution was also proposed to eliminate this problem by adding a complementary polarizer [18], [19]. In this proposed device, the MTJ has two pinned layers instead of one, with opposite magnetic orientations. Depending on the information to write, the switching current will flows through the corresponding pinned layer.

Third, STT-MRAM is confronted to scalability issue. When a STT-MRAM cell is scaled, the thermal stability factor scales down linearly with the area, and can cause unreliability due to retention failure [20]. Moreover, although its switching energy remains low compared to Toggle and TAS-MRAM, STT-MRAM needs access transistor sizes larger than the minimum size at advanced node (32 nm and below) [21], limiting thus the memory density . This is an issue also for high performance applications which require high write speed, since the switching current of STT-MRAM increases when the write pulse width decreases.

E. Voltage Induced Switching

In order to improve the scalability and reduce the switching energy observed with STT-MRAM, a voltage-controlled MTJ were proposed ([22]–[26]), also known as magnetoelectric random access memory (MeRAM). This approach uses voltage rather than current to reverse the magnetization of the free layer thanks to the recently demonstrated voltage-controlled magnetic anisotropy effect (VCMA) [27]. The free layer has a magnetic anisotropy that can be changed by voltage. Hence, voltage-induced switching of the magnetization can be performed modifying the magnetic anisotropy of the MTJ.

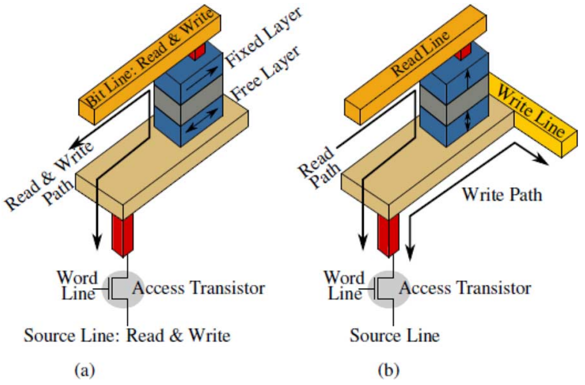


Fig. 7. SOT-MRAM. (a) Conventional STT-MRAM. (b) SOT-MRAM [33].

The voltage-controlled MTJ (VMTJ) structure uses materials commonly used by previous MRAM technologies, thus maintaining manufacturability [22]. VMTJ has an unipolar voltage-controlled behavior, i.e., switching is performed by set/reset voltages of different amplitudes but same polarity, whereas STT-MRAM uses opposite current polarities to switch the bit cell.

Since switching for VMTJ is performed via voltage, the barrier thickness can be increased to reduce the parasitic conductance and hence the effect of current-induced torques (i.e., STT effect). Moreover, a high TMR (greater than 100%) is possible allowing the readout of the magnetization of the free layer [28]. Although it is still at experimental level and needs further improvements in the design, MeRAM is expected to improve the scalability by eliminating the need for large currents, which is currently a real issue with STT-MRAM for advanced technology nodes.

F. *Spin Orbit Torque*

SOT-MRAM is the most recent technology for MRAMs. It was developed to mitigate the issues observed in STT-MRAM. Contrary to STT-MRAM, this new technique uses a three-terminal structure to separate the read and write paths, as shown in Fig. 7(b). The physical effect responsib le for the reversal of magnetization of the FL is not yet fully understood. According to some authors, the Rashba effect [35] or the spin Hall effect [36] could explain the switch in magnetization of the storage layer.

Unlike STT-MRAM, SOT-MRAM intrinsically separates the read and write paths and allows symmetrical switching current between the two states of the MTJ. Hence, read stability is improved, strongly reducing the possibility of a bit flip (the bit changes its state) during a read operation. Also, designers can optimize the read and write separately. On the other hand, SOT-MRAM has a bigger cell size than STT-MRAM because of its three-terminal structure. As SOT-MRAM is a young technology compared to other MRAMs, further research is needed to optimize the SOT-based MTJs. Like MeRAM, a great potential is expected from this technology to reach same performance as SRAM.

G. *MRAM Technologies: Summary*

Table I summarizes the MRAM technologies above to give an overview of the main differences between them.

For the evaluation of MRAM-based cache memory in Sections IV and V, we decided to evaluate TAS-MRAM and STT-MRAM technologies. As mentioned above, toggle MRAM has very high switching energy and its scalability is limited, which is why we decided not to include it in our study. Even though MeRAM is a very promising candidate for fast and ultra-low power applications, it is always at experimental level and requires additional development, hence it was not evaluated in this work. We also decided not to include SOT-MRAM in our evaluation because it is still a young technique that has not yet been fully explored and is not yet completely understood, as already mentioned in Section II-F. Even though TAS-MRAM may currently not appear to be competitive with SRAM, this technology is quite mature, hence we decided to evaluate a TAS-MRAM-based L2. We also decided to evaluate a STT-MRAM-based L2 cache, as its performance is close to that of SRAM.

III. NVM EXPLORATION FLOW

A. *Overview*

As seen above, MRAM has attractive features such as low leakage, high density, and non-volatility. However, MRAM still suffers from high write latency and high write energy. To evaluate the impact of including MRAM in the memory hierarchy of processor architecture, a framework based on both circuit-level and architecture-level tools is needed. A circuit-level tool needs to provide characteristics of a complete memory circuit (i.e., including cell array and peripheral circuits). An architecture-level tool simulates a complete processor-based system with its memory hierarchy. For area, performance, and energy evaluations, the minimum information required is:

- **Circuit-level requirements:** access latency, access energy, static power, area;
- **Architecture-level requirements:** execution time of the simulated applications, amount of memory transactions for each level of the memory hierarchy.

Another important point is that the framework needs flexibility (i.e., extension or modifications should be possible) to make it possible to model any kind of architecture.

In this section, we propose an exploration flow based on gem5 [37], a processor architecture simulator widely used by the research community. gem5 currently supports most commercial instruction set architectures (ISA) including ARM, ALPHA, MIPS, Power, SPARC, and x86. gem5 is able to simulate a complete processor-based system with devices and operating system in full system mode (i.e., nothing is emulated). The use of gem5 makes it possible to define the total processor system architecture, including memory hierarchy specifications: cache size, cache and main memory latencies, etc. Execution time and memory transactions can be extracted for a given application, i.e., cache read/write accesses including cache hits and misses. In addition, the cache miss rate, the cache miss latency, and the memory bandwidth can be monitored over time to better understand the activity of the memory. Hence, a fine-grained analysis

TABLE I
MRAM TECHNOLOGIES

Technology	**Cell size** (F^2)	**Access time read/write**	**Write current**	**Endurance**	**Maturity**	**Advantages/Drawbacks**
Toggle MRAM [5], [11], [10]	50	35 ns / 35 ns	$>$30 mA	10^{15}	Commercialized	(+) Selectivity (+) Scalability (+) Reliability
TAS-MRAM [6], [12], [13]	$<$50	30 ns / 30 ns	A few mA	10^{15}	Test chip [12]	(++) Selectivity (++) Scalability (+++) Reliability
STT-MRAM [7], [19]	$<$50	2-20 ns / 2-20 ns	50 uA	$> 10^{16}$	Test chip [31], [32], [33]	(++) Selectivity (++) Scalability (++) Reliability
MeRAM [34], [30]	$<$10	$<$10 ns	very low	$> 10^{16}$	Prototype	(++) Selectivity (+++) Scalability (+++) Reliability
SOT-MRAM [15], [35], [36]	$<$50	A few ns	$<$100 uA	$> 10^{16}$	Prototype	(++) Selectivity (++) Scalability (++) Reliability

of performance and energy results for each simulated workload is possible.

Using gem5 is a judicious choice for processor architecture researchers for three main reasons. First, it is open source. Second, it is a community-supported tool, i.e., extension of this tool is done by gem5 users from both industry and academia, making gem5 a sustainable solution. Third, the flexibility of gem5 allows users to easily model new architectures, new cache management policies, or any new optimization techniques at architecture level. In addition, gem5 is potentially able to allow exploration of manycore architecture including more than one hundred cores applying a trace-driven approach proposed in [38].

B. Description

In this sub-section, we provide details on the NVM exploration flow used in this paper. Evaluating performance/energy/area of NVM-based memory hierarchy can be divided into 5 steps:

1) Defining the architecture: Single or multi-core, ISA choice (e.g., ARM, x86), defining the memory hierarchy (e.g., level of cache, memory size).
2) Obtaining memory characteristics at circuit level (Area analysis possible at this step).
3) Calibrating each level of the memory hierarchy with the access latencies obtained in step 2.
4) Extracting the outputs of the gem5 simulation (Performance analysis possible at this step)
5) Calculating the energy consumption of each level of the memory hierarchy (Energy analysis possible at this step).

To calibrate the memory hierarchy in terms of access latency (step 3), NVSim [39] was used, a circuit-level model for NVM performance, energy (considering both cell array and peripheral circuits), and area estimation, which supports different NVM technologies including planar STT-MRAM, RRAM, PCRAM. It also models the volatile SRAM memory. NVSim should be used for a rapid estimation of electrical features of a complete memory chip. The estimation error is less than or equal to 24% [39]. For more precise evaluations, the results from SPICE simulation of a design or the electrical features of a real prototype are more appropriate.

In step 5, the total cache energy consumption is calculated as shown in the (1). P_{static} is the leakage power of the cache, N_r (N_w) is the number of reads (writes), and E_r (E_w) is the energy per access for a read (write) operation

$$\begin{aligned} E_{total} &= E_{static} + E_{dynamic} \\ &= P_{static}.Runtime + N_r.E_r + N_w.E_w. \end{aligned} \tag{1}$$

IV. MRAM-BASED CACHE: CIRCUIT-LEVEL ANALYSIS

In this section, we provide a first comparative analysis of the performance/energy/area of MRAM-based and SRAM-based caches at circuit level (i.e., a direct comparison of the memory characteristics). STT-MRAM-based, TAS-MRAM-based and SRAM-based caches are analyzed. Table II shows the cache parameters (latency, energy per access, static power, area) of a 512 kB L2 for the three memory technologies concerned. Table III shows the same parameters for a 32 kB L1. Because TAS-MRAM is not suitable for use in L1 (because of its slow access time), we only evaluated this technology for the L2 cache. Note that results of both SRAM and STT-MRAM come from NVSim, while for TAS-MRAM, outcomes from a real prototype are used thanks to support provided by Crocus Technology. To take into account the state-of-the-art of MRAM technology and to evaluate performance and energy fairly, we compared 45 nm STT-MRAM-based cache results with a baseline 45 nm SRAM-based cache, and 130 nm TAS-MRAM-based cache results with a baseline 120 nm SRAM-based cache.

Regarding results in Tables II and III, we can observe that SRAM-based cache is a little faster in write than in read operation. This is understandable because a read is quite different from a write at circuit level. Unlike write, a read operation needs to read out data through a sense amplifier. The readout delay depends on the load capacitance of the bit line. Therefore, a read operation has additional time and energy compared to a write

TABLE II
512 kB L2 CACHE FEATURES

	Latency		Energy			Cache area	
Technology	**Read**	**Write**	**Read**	**Write**	**Leakage**	**Total**	**Cell**
	(ns)	(ns)	(nJ)	(nJ)	(mW)	(mm^2)	(F^2)
45 nm SRAM	4.28	2.87	0.27	0.02	320	1.36	146
45 nm STT	2.61	6.25	0.28	0.05	23	0.82	57
120 nm SRAM	5.95	4.14	1.05	0.08	82	9.7	146
130 nm TAS	35	35	1.96	4.62	10	11.7	35

TABLE III
32 kB L1 CACHE FEATURES

	Latency		Energy			Cache area	
Technology	**Read**	**Write**	**Read**	**Write**	**Leakage**	**Total**	**Cell**
	(ns)	(ns)	(nJ)	(nJ)	(mW)	(mm^2)	(F^2)
$45nm$ SRAM	1.25	1.05	0.024	0.006	22	0.091	146
$45nm$ STT	1.94	5.94	0.095	0.04	3.3	0.117	57

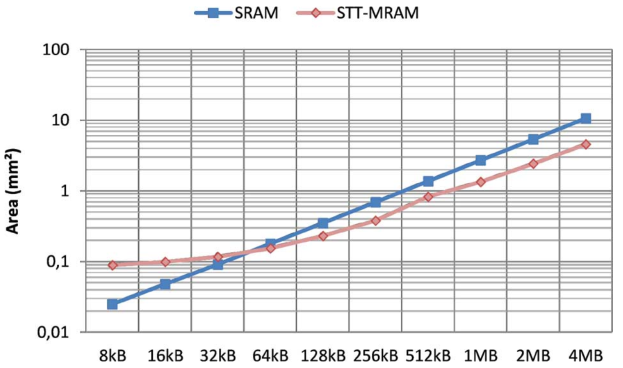


Fig. 8. Cache area scaling behavior for SRAM and MRAM (log scale for area).

operation due to data sensing. This is why STT-MRAM write also shows smaller energy than read at circuit-level. However, the main issue of MRAM, i.e., its write latency and energy are higher than those of SRAM, is still present.

A. Analysis of 512 kB L2 Cache

As expected, both MRAM technologies have higher write latency than SRAM. Regarding read latency, STT-MRAM is faster than SRAM (45 nm). This is due to a smaller load capacitance for STT-MRAM as it is denser than SRAM. Hence, when the cache capacity increases, SRAM read latency increases much more than that of STT-MRAM [33]. Concerning the difference in cache area between the two technologies, a SRAM bit cell consists of six CMOS transistors whereas a STT-MRAM bit cell consists of one CMOS transistor and a MTJ. As a result, for the same capacity, the area of the cell array for STT-MRAM is smaller than for SRAM resulting in smaller total L2 cache area (Table II). These two advantages of STT-MRAM in terms of read latency and cache area are only noticeable in the case of large cache capacity, when the area of the cell array occupies a large proportion of the total cache area compared to the area occupied by the peripheral circuitry. Fig. 8 shows the scaling behavior of the cache area for both SRAM and STT-MRAM. As observed in this figure, STT-MRAM-based cache is more area efficient than SRAM-based cache for capacities of 64 kB and above.

TAS-MRAM write and read latencies are respectively 8.5 and 6 times higher than those of SRAM (120 nm). As explained in Section II-C, the long write latency of TAS-MRAM in mainly due the heating/cooling stages required to switch the MTJ. The total cache area of TAS-MRAM is slightly larger than that of SRAM (120 nm), for which there are two explanations. First, contrary to STT-MRAM, the TAS-MRAM bit cell is written by a magnetic field generated by a current flowing through a conductive line. Hence, conductive lines have to be added to TAS-MRAM-based memory. Second, as explained in Section II-C, TAS-MRAM requires a high write current (higher than STT-MRAM). As a result, the write drivers have to be large enough to generate sufficient current for a write operation.

Concerning the cell size values in Table II, note that the field line is not considered for TAS-MRAM. This MRAM technology shows a smaller cell area in terms of feature size (F) compared to the STT-MRAM model in NVSim. This is explained by a smaller access transistor (in terms of F) for TAS-MRAM when compared to STT-MRAM in NVSim.

Regarding L2 energy consumption, using MRAM instead of SRAM results in higher write energy for both STT-MRAM and TAS-MRAM. STT-MRAM read energy is very similar to that of SRAM, whereas a TAS-MRAM read consumes around 2× more energy than SRAM. However, in terms of leakage power, MRAM has a considerable advantage over SRAM: a 45 nm STT-MRAM-based L2 consumes over one order of magnitude less power than 45 nm SRAM-based L2, while a TAS-MRAM-based L2 consumes around 8× less power than a 120 nm SRAM-based L2. This is because most of the static power of large capacity memories comes from cell arrays. Since MRAM cell has zero standby power and the CMOS access transistor does not require a power supply (to retain data), all the static power in MRAM-based memory is due to peripheral circuitry such as address decoding, drivers, and sense amplifiers.

B. Analysis of 32 kB L1 Cache

As shown in Table III, STT-MRAM and SRAM read latencies are similar. But a higher latency is still observed for STT-MRAM writes. STT-MRAM consumes around 4× and 7× more energy than SRAM for read and write operations, respectively. A significant gain in static power is obtained by replacing SRAM with STT-MRAM due to the zero leakage of the MTJ. The total cache area of a STT-MRAM-based L1 is slightly larger than that of a SRAM-based L1 cache. This is because the capacity of the cache is small (32 kB), and so, the area occupied by the peripheral circuits is not negligible compared to the area of cell array. Since MTJ writes need a large amount of current, the transistors of the write circuitry for STT-MRAM have to be large enough to generate sufficient write current. As a result, the peripheral circuits for STT-MRAM-based cache occupy more area than their SRAM equivalents.

C. Summary

The comparison of MRAM and SRAM at the circuit level showed that the high density of MRAM may be advantageous in terms of read latency for large cache capacity. In addition, considerable static power can be saved using MRAM instead

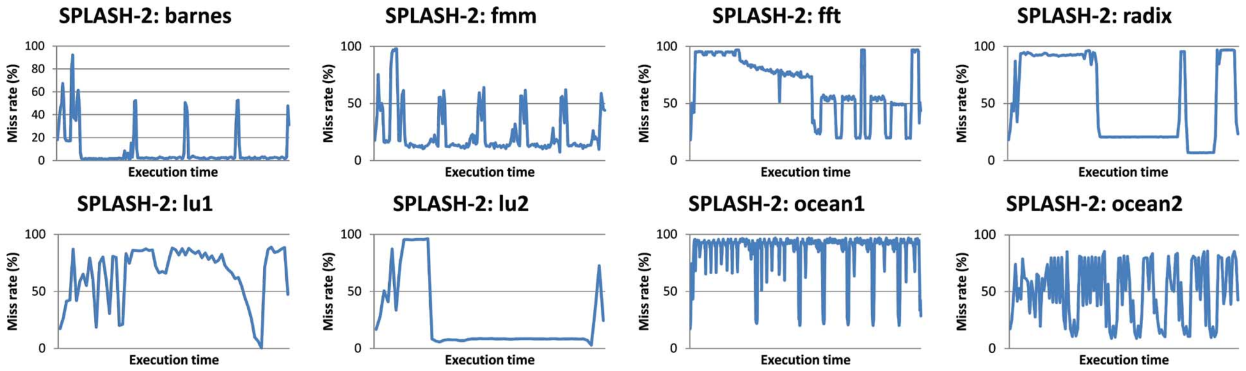


Fig. 9. L2 cache miss rate.

TABLE IV
ARCHITECTURE CONFIGURATION

Hierarchy level	Configuration	
Processor	4-core, 1 GHz, 32-bit RISC ARMv7 (Linux OS)	
L1 I/D cache	Private, 32kB, 4-way associative, 64B cache line	
L2 cache	Shared, 512kB, 8-way associative, 64B cache line	
SRAM baseline	45nm SRAM (read: 5, write: 3)	120nm SRAM (read: 6, write: 5)
MRAM cache	45nm STT (read: 3, write: 7)	130nm TAS (read: 35, write: 35)
Main Memory	DRAM, 512MB, DDR3, 100-cycle latency	

TABLE V
BENCHMARKS

Splash-2 benchmark	Input set
barnes	16K Particles, Timestep = 0.25, Tolerance 1.0
fmm	16K Particles, Timestep = 5
ft	2^{20} total complex data points
lu1	Contiguous blocks, 512x512 Matrix, Block = 16
lu2	Non-contiguous blocks, 512x512 Matrix, Block = 16
ocean1	Contiguous partitions, 514x514 Grid
ocean2	Non-contiguous partitions, 258x258 Grid
radix	4M Keys, Radix = 4K

Parsec benchmark	Input set
blacksholes	4,096 options
bodytrack	4 cameras, 1,000 particles, 5 layers, 1 frame
ferret	3,544 images, 16 queries
fluidanimate	35,000 particles, 5 frames
streamcluster	4,096 points per block, 32 dimensions, 1 block
x264	640 x 360 pixels, 8 frames

of SRAM. On the other hand, our results clearly reveal the disadvantage of using MRAM in terms of write latency and write energy compared to its SRAM equivalents. The difference between the latency parameter in SRAM and MRAM will of course depend on the frequency used by the processor. In this study, the frequency used for the processor is 1 GHz. Table IV shows the access latencies in terms of CPU cycle for L1 and L2.

Regarding L1 cache, results clearly show that STT-MRAM is currently not competitive to directly replace SRAM according to the three metrics speed, energy, and area. As a result, it is necessary to propose optimizations at architecture level, such as in [40], or at circuit level to allow use of MRAM in such a cache level. Therefore, for architecture-level exploration, the following section will only focused on the L2 cache.

V. MRAM-Based Cache: Architecture-Level Analysis

A. Experimental Setup

A few workloads of *SPLASH-2* [41] and *PARSEC* [42] benchmark suites were used to explore STT-MRAM and TAS-MRAM based caches for a quad-core processor ARM architecture. While *SPLASH-2* workloads are mostly focused on high performance computing (HPC), *PARSEC* includes emerging workloads in many different areas such as computer vision, financial analytics, data mining, animation physics, image processing and video encoding. According to [43], *PARSEC* includes workloads that handle a huge amount of data compared to *SPLASH-2*. Table IV shows the architecture configuration we used for simulation and Table V provides details on the simulated workloads. The gem5 simulator source code was modified to allow different read and write latencies to be configured (by default, gem5 assumes the same latency for reading and writing).

B. Analysis of Cache Memory Activity

Here we provide prior results on the workload behavior in terms of memory activity. Information including the read/write ratio, static/dynamic energy ratio, L1/L2 access ratio, the cache miss rate and the cache bandwidth are analyzed. The results are shown for the baseline architecture (SRAM-based cache). The aim of this preliminary study was to obtain useful information for a more comprehensive analysis of subsequent results concerning the impacts of incorporating MRAM into cache memory on performance and energy.

1) Read/Write Ratio: Since access time and energy requirements for reading and writing differ considerably in MRAM, it is important to analyze the read/write ratio of the workloads. As explained in Sections I and II, a MRAM write operation consumes more energy and is slower than its SRAM equivalent. Fig. 11 depicts the read/write ratio for L2.

Our results showed that majority of the accesses are reads for both L1 and L2 caches. In L2, three workloads (lu1, lu2, blackscholes) have more than 30% writes, and two workloads (ferret, fluidanimate) have more than 40% writes.

2) Static/Dynamic Energy Ratio: The static/dynamic energy ratio in memory is helpful when studying different memory technologies. As mentioned in Section I, the main challenge

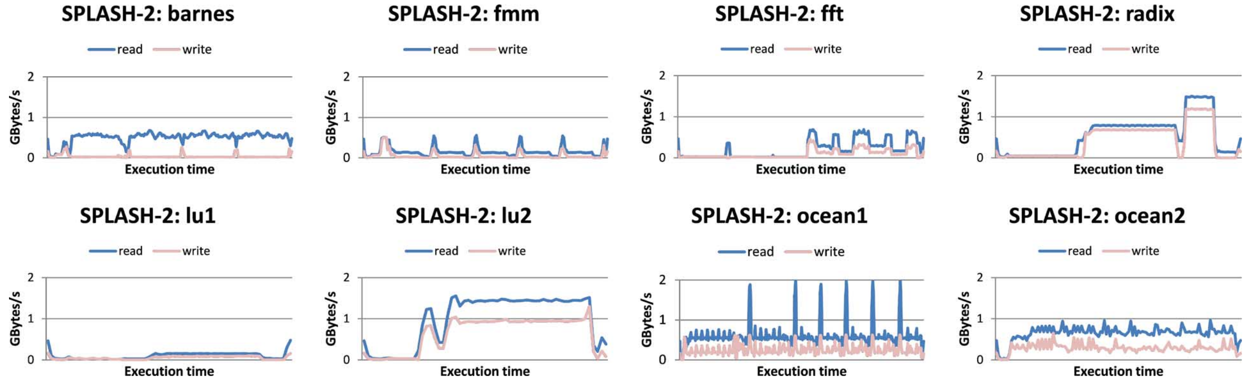


Fig. 10. L2 cache bandwidth.

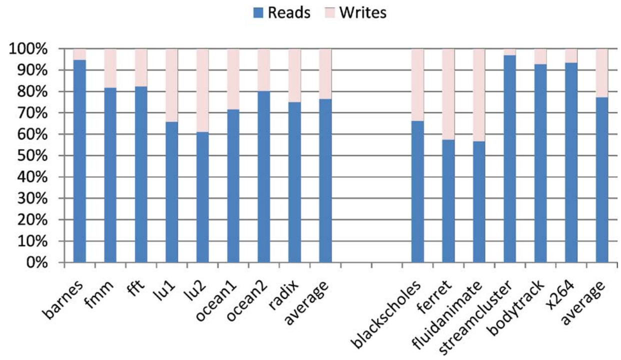


Fig. 11. L2 read/write ratio.

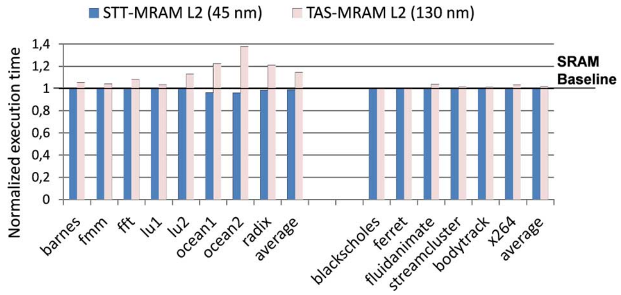


Fig. 12. Execution time with MRAM-based L2 cache.

TABLE VI
L1/L2 ACCESS RATIO

Benchmark	Number of accesses	
	L1	L2
Splash-2	2 billion (0.5 billion/CPU)	26 million
Parsec	12 billion (3 billion/CPU)	16 million

for existing and future ICs is energy efficiency. Due to the high leakage current of SRAM, a memory device like MRAM with ultra-low leakage is very attractive. Our results showed that more than 90% of the total energy consumption in the L2 cache is static. In Section V-C, we analyze the results concerning the MRAM-based cache to see whether the notable gain in static power consumption in MRAM compensates for the high dynamic energy loss of MRAM in L2.

3) L1/L2 Access Ratio: The L1/L2 access ratio provides a rough idea of the impact of the L2 (and L1) cache on performance (i.e., execution time). For instance, if there is a big difference between the number of L1 accesses and the number of L2 accesses, L2 will have little impact on the execution time. Table VI lists the number of accesses in L1 and L2 caches. Note that each value is the average of all the workloads.

For both benchmarks (*SPLASH-2* snd *PARSEC*), L1 is much more accessed than L2. The L1/L2 ratio for *PARSEC* is clearly higher than for *SPLASH-2*. As a result, if MRAM is used in L2 when executing *PARSEC* workloads, the drop in performance due to the long write latency of MRAM will be less visible.

4) Cache Miss Rate and Cache Bandwidth: Other interesting results concerning memory activity are related to the cache miss rate and the cache bandwidth. The cache miss rate reveals the number of times data is not found in the cache, thus requiring access to a lower level of the memory hierarchy to update (write) the corresponding cache block. A high cache miss rate can have a negative effect on performance due to the long write latency of MRAM. This is the case if the cache bandwidth is also high. The cache bandwidth is the number of bytes accessed per second. Therefore, if both cache miss rate and cache bandwidth are high, using MRAM will likely have a negative effect on performance. The L2 cache miss rate and bandwidth for each workload of *SPLASH-2* are listed in Figs. 9 and 10 respectively.

Although the same results as Figs. 9 and 10 are available for the *PARSEC* workloads, we do not show them here for the sake of brevity.

C. Exploration of the L2 Cache

1) Performance Evaluation: Fig. 12 shows the execution time of *SPLASH-2* and *PARSEC* workloads for both STT-MRAM and TAS-MRAM based L2 caches. Fig. 12 shows that the performance of STT-MRAM-based L2 scenario is similar and sometimes better (*ocean1*, *ocean2*) than the baseline. This is because STT-MRAM has a smaller read latency than its SRAM equivalent and also to the fact that, as mentioned in Section V-B1, L2 is more accessed in read.

TAS-MRAM-based L2 performance penalties of 14% and 2% on average were observed for *SPLASH-2* and *PARSEC* workloads respectively. In the case of *ocean2*, 38% of performance degradation was observed using TAS-MRAM. The cache memory traffic analysis (Section V-B) showed that the difference in the number of L1 and L2 accesses was larger for *PARSEC* than for *SPLASH-2*. Hence, L2 has less impact on

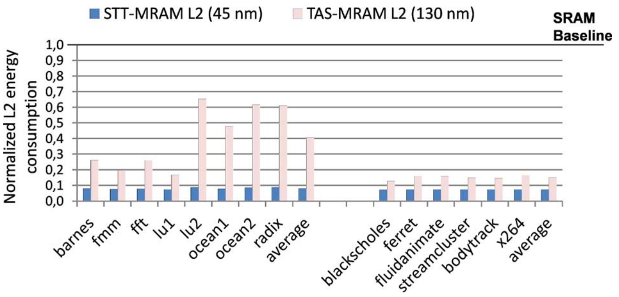


Fig. 13. MRAM-based L2 energy consumption.

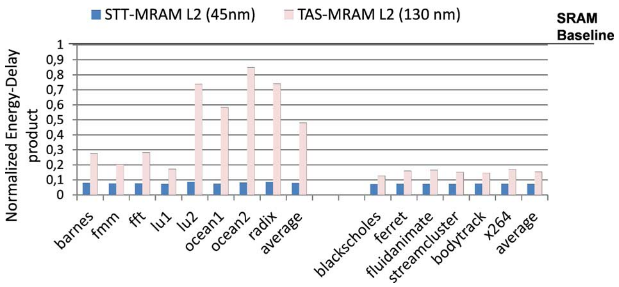


Fig. 14. Energy-Delay product of MRAM-based L2 cache.

the execution time for *PARSEC* than for *SPLASH-2* explaining why the decline in performance is very small for *PARSEC* workloads, even using TAS-MRAM.

The highest penalties in execution time using TAS-MRAM were observed for *ocean1*, *ocean2* and *radix* workloads. This is understandable when the cache miss rate and cache bandwidth in Figs. 9 and 10 are analyzed: all three workloads have both a high cache miss rate and a high cache bandwidth compared with other *SPLASH-2* workloads. As explained in Section V-B4 above, for memory activity, this kind of behavior does not favor MRAM due to its long write latency.

2) Energy Evaluation: Fig. 13 shows total L2 energy consumption (including dynamic and static energy). Simulation results showed using STT-MRAM is 92% more energy efficient on average than SRAM for both *SPLASH-2* and *PARSEC* workloads. Using TAS-MRAM, 63% and 84% energy gains on average were observed for *SPLASH-2* and *PARSEC* respectively. This notable difference in energy consumption between the two technologies is explained by the low leakage power of MRAM compared to SRAM. As we noticed during the analysis of the cache memory activity (Section V-B), more than 90% of the total L2 energy consumption is static when using SRAM. As a result, replacing SRAM with MRAM can dramatically reduce the total energy consumption in L2. This makes MRAM-based cache memory an attractive alternative for energy efficient systems because despite the reduction in energy consumption, the performance remains reasonable.

Fig. 13 shows that *lu2*, *ocean1*, *ocean2*, and *radix* are the workloads with the lowest energy gain for TAS-MRAM-based L2. This makes sense given the previous results on the cache bandwidth in Fig. 10, showing that L2 is more frequently accessed for these four workloads. As seen in Table II, TAS-MRAM consumes more energy than SRAM, not only for writes, but also for reads in the L2 cache. As a result, less energy is gained for these four workloads because of the loss in dynamic energy when SRAM is replaced by TAS-MRAM.

3) Energy-Delay Product: To evaluate the trade-off between performance and energy consumption, the energy-delay product (EDP) is an interesting figure of merit. Fig. 14 gives the EDP results of STT-MRAM and TAS-MRAM based L2 caches compared to the baseline. As observed in this figure, using STT-MRAM and even TAS-MRAM in L2 cache is better than using SRAM if a trade-off between performance and energy is targeted. Use of STT-MRAM results on a gain of more than 90% in terms of EDP compared to SRAM. Whereas for TAS-MRAM, the gain is from 14% to 87% depending on the workload.

D. Summary

Regarding performance, no penalty in execution time is observed when SRAM is replaced by STT-MRAM in L2 (LLC) because STT-MRAM has a lower read latency than its SRAM equivalent. The low read latency of STT-MRAM is due to its high density compared to SRAM, resulting in a smaller cache area than SRAM, which is observed only for large cache capacity. For TAS-MRAM, the reduction in performance depends on the number of accesses in L2 compared to L1 (L1/L2 access ratio). If L2 has a small impact on the execution time, TAS-MRAM-based L2 is better than SRAM-based L2 in terms of a trade-off between performance and energy consumption.

Concerning energy, the low leakage of MRAM is extremely beneficial for lower levels of cache in the memory hierarchy, since leakage accounts for an important part of the total energy consumption of SRAM cache, as seen in Section V-B2.

VI. Related Work

Many studies have been conducted on integrating of MRAM into the memory hierarchy of single-core and multi-core architectures. All these studies explored a hybrid cache hierarchy using SRAM and STT-MRAM technologies, but a few studies also explored DRAM and PCRAM. Most of the studies evaluated the use of STT-MRAM for LLC. A few authors explored MRAM for upper levels of cache such as L1.

A. 3D-Stacking MRAM

Some authors studied the benefit of the 3D-stacking ability of MRAM combined with its high density to evaluate 3D-processor architecture. They analyzed the performance and energy impacts of having a MRAM-based LLC on top of a 2D-processor architecture. [44] and [45] evaluated a 3D-stacked STT-MRAM-based L2 on top of a 2D-processor architecture. [46] explored three different memory hierarchy configurations using MRAM, PCRAM and embedded DRAM in an 8-core processor considering a 3D chip integration.

B. MRAM-Based Non-Uniform Cache Architecture

Other authors explored non-uniform cache architectures (NUCA) using both SRAM and MRAM in one cache level. [47] and [46] proposed a hybrid cache consisting of a large but slow MRAM-based region and small but fast SRAM-based region. Using data migration policies, the objective is to write mostly in the SRAM region (because it is faster) and to read data from the MRAM region. In this way, performance degradation due to the high write latency of MRAM can be mitigated. In addition, larger cache capacity is possible thanks to the high density of MRAM. [48] proposed a hybrid cache architecture for chip-multiprocessors. In addition, micro-architectural mechanisms were introduced to reduce the number of writes in STT-MRAM regions.

C. Novel Management Policies for MRAM-Based Cache

Several cache management techniques have been proposed to mitigate the two main drawbacks of MRAM (high write latency and high write energy). [49] proposed a novel technique called early write termination (EWT). EWT aims at removing unnecessary writes (i.e., writing same value) to reduce the write energy consumption in the cache. Because MTJ does not switch gradually but abruptly at the end of a STT-based write, this technique proposed to read the stored value during a write operation and to stop the write if it is redundant. [45] introduces the read-preemptive write buffer technique in which write buffers are used to mitigate the long write latency of MRAM. In addition, when there is a conflict between a read (from the upper level cache) and a write (from the write buffer), a read-preemptive policy gives priority to the read in order to prevent write operations from blocking read operations due to the long write latency of writes. [15] proposed a similar technique called the obstruction-aware policy (OAP) for cache management of a single-port STT-MRAM-based L3 (LLC). In addition to preventing the delay of several read operations caused by a long write operation, OAP can mitigate the performance degradation of the MRAM-based LLC while significantly reducing total energy consumption thanks to the ultra-low leakage of MRAM. [50] proposed a new STT-MRAM cache architecture called asymmetric write architecture with redundant blocks (AWARE) to reduce the average cache write latency. This is done by taking advantage of the asymmetric write characteristics of STT-MRAM.

D. Other Studies on MRAM-Based Cache

Among studies on MRAM-based cache, [51] evaluated the performance/energy impacts of a STT-MRAM-based L2 when the retention time of the MTJ is reduced. Reducing the retention time of the MTJ can reduce both switching energy and switching latency. 10+ years, 1 s and 10 ms retention times for STT-MRAM were explored. In addition, a cache revive policy was proposed for the 10-ms-retention-time-based STT-MRAM to refresh data if necessary. [52] proposed fine-grained power gating on STT-MRAM peripheral circuits to further reduce the total energy consumption of STT-MRAM-based LLC. [13], [33] made a first study of using the new SOT-MRAM technology in caches. Both SOT-MRAM-based L1 and L2 were explored and compared with SRAM-based and STT-MRAM-based caches. Although this work is close to our study, some important aspects on the architecture and the memory behavior were not taken into account for relevant analysis. First, the exploration were only made for a single-core architecture. It is also important to analyze the impact of having a multi-core architecture because critical information such as the memory bandwidth can significantly influence the results (especially for a cache shared between the cores). Second, device and circuit-level information were mostly used to analyze performance and energy results at architecture level. As proposed in this paper, it is necessary to take into account the memory activity of the cache, such as the miss rate and bandwidth since our results have demonstrated the high influence of these parameters on the overall performance and energy consumption. [40] also explored MRAM-based L1 cache (Data-Cache) using an advanced perpendicular STT-MRAM (ap-STT-MRAM) proposed in [53]. This work highlighted that the read latency of STT-MRAM is the new bottleneck when it is used in upper level cache (i.e., L1). After demonstrating the major performance penalty of replacing SRAM by STT-MRAM in L1-Data cache, micro-architectural modifications by means of an intermediate buffer placed between the processor and the L1-Data cache have been proposed to overcome the read limitations of the STT-MRAM. In addition, appropriate data allocations schemes coupled with code transformations and optimizations are performed to reduce the performance penalty introduced by the STT-MRAM to extremely tolerable levels (8%) compared to a SRAM L1-Data cache design.

E. Discussion

Many works on MRAM-based caches point to the real interest of this memory technology for future ICs. The common trend in these studies was to take advantage of the non-volatility, high density, low leakage, and 3D-stacking capability of MRAM while mitigating its drawbacks, which are high write energy and latency. Results showed that for large cache capacity (e.g., LLC), systems can significantly benefit from the high density and the ultra-low leakage of MRAM with no or with very little performance degradation. [54] showed that for the big.LITTLE system [55], the L2 cache area is about 40% and 30% of the total area of the cortex-A7 cluster (4-core) and of the cortex-A15 cluster (4-core), respectively. An energy evaluation showed that cache energy consumption (including L1 and L2) represents around 50% and 25% of the total energy consumption of the cortex-A7 cluster and the cortex-A15 cluster, respectively, when only one core is active. For write-intensive workloads, cache management needs to be optimized to mitigate the high write energy and latency of MRAM. In the case of small cache capacity, such as L1, SRAM is always better than MRAM, especially for L1-Data cache. However, if we consider the three metrics speed/energy/area, studies showed that the best trade-off is clearly a hybrid cache hierarchy that uses both SRAM and MRAM.

In this paper, we proposed an exploration flow based on architecture-level and circuit-level tools allowing large exploration of MRAM-based cache including the studies presented in this section. Compared to previous papers, this paper adds the extraction of a lot of useful information about the cache memory activity (cache miss rate, cache bandwidth, static/dynamic energy ratio, etc.) to enable a fine-grained performance/energy analysis of MRAM-based cache. This is especially helpful when evaluating the impact of a MRAM-based cache for multi-core architectures.

VII. Conclusion

For several years, new NVMs technologies have been under intensive investigation in an era where SoCs design has to be reconsidered to face the most critical challenge: energy efficiency. This paper proposes an exploration flow that allows large and fine-grained exploration of integrating MRAM into the memory hierarchy (cache) of processor architecture. Performance and energy consumption are analyzed using useful information on memory traffic such as the cache miss rate and the cache bandwidth. Simulations show that the total energy consumption of LLC can be significantly reduced (up to 90% for certain applications) thanks to the low leakage power of MRAM. However, for upper levels of cache, e.g., L1, current MRAM characteristics do not make it the best choice for direct replacement of SRAM. Even if energy reduction is possible thanks to the low leakage of MRAM, performance degradation is still an issue.

As perspectives, it is envisaged to explore use of STT-MRAM in L1 cache considering the last improvements at device and circuit levels. In addition, modifications at architecture level may be done to reduce the performance penalty caused by the high write latency of STT-MRAM. Finally, one of the most interesting feature of MRAM to reduce the total energy consumption of a system is non-volatility, which will certainly help to propose new energy optimizations based on instant-on/off computing.

Acknowledgment

The authors wish to acknowledge all people from ADAC team at LIRMM and people from Crocus technology for their support in this work.

References

[1] F. Shearer, *Power Management in Mobile Devices*. Oxford, U.K.: Newnes, 2011.

[2] E. Kitagawa *et al.*, "STT-MRAM cuts power use by 80%," [Online]. Available: http://www.eetimes.com/document.asp?doc_id=1280753

[3] S. Senni, L. Torres, G. Sassatelli, A. Bukto, and B. Mussard, "Exploration of magnetic ram based memory hierarchy for multicore architecture," in *Proc. 2014 IEEE Comput. Soc. Annu. Symp. IEEE VLSI*, 2014, pp. 248–251.

[4] J. S. Moodera, L. R. Kinder, T. M. Wong, and R. Meservey, "Large magnetoresistance at room temperature in ferromagnetic thin film tunnel junctions," *Phys. Rev. Lett.*, vol. 74, no. 16, p. 3273, 1995.

[5] B. Engel *et al.*, "A 4-mb toggle MRAM based on a novel bit and switching method," *IEEE Trans. Magn.*, vol. 41, no. 1, pp. 132–136, Jan. 2005.

[6] I. Prejbeanu *et al.*, "Thermally assisted MRAM," *J. Phys. Condensed Matter*, vol. 19, no. 16, p. 165218, 2007.

[7] A. Khvalkovskiy *et al.*, "Basic principles of STT-MRAM cell operation in memory arrays," *J. Phys. D, Appl. Phys.*, vol. 46, no. 7, pp. 74001–74020, 2013.

[8] P. Gambardella and I. M. Miron, "Current-induced spin-orbit torques," *Phil. Trans. R. Soc. London A, Math. Phys. Eng. Sci.*, vol. 369, no. 1948, pp. 3175–3197, 2011.

[9] K. Lewotsky, "Tech trends: Details on Everspin's ST-MRAM," [Online]. Available: http://www.eetimes.com/document.asp?doc_id=1280267

[10] T. W. Andre *et al.*, "A 4-mb $0.18-\mu$m 1t1mtj toggle MRAM with balanced three input sensing scheme and locally mirrored unidirectional write drivers," *IEEE J. Solid-State Circuits*, vol. 40, no. 1, pp. 301–309, Jan. 2005.

[11] I. Prejbeanu *et al.*, "Thermally assisted MRAMs: Ultimate scalability and logic functionalities," *J. Phys. D, Appl. Phys.*, vol. 46, no. 7, p. 074002, 2013.

[12] B. Cambou, Match in place. A novel way to perform secure and fast user's authentication [Online]. Available: www.crocus-technology.com.

[13] R. Bishnoi, M. Ebrahimi, F. Oboril, and M. B. Tahoori, "Architectural aspects in design and analysis of sot-based memories," in *Proc. 2014 19th Asia South Pacific Design Automat. Conf.*, 2014, pp. 700–707.

[14] S. Lee, K. Kang, and C.-M. Kyung, "Runtime thermal management for 3-d chip-multiprocessors with hybrid SRAM/MRAM l2 cache," *EEE Trans. Very Large Scale (VLSI) Syst.*, vol. 23, no. 3, pp. 520–533, Mar. 2015.

[15] J. Wang, X. Dong, and Y. Xie, "OAP: An obstruction-aware cache management policy for STT-RAM last-level caches," in *Proc. Conf. Design, Automat. Test Eur. Consort.*, 2013, pp. 847–852.

[16] N. N. Mojumder, S. K. Gupta, S. H. Choday, D. E. Nikonov, and K. Roy, "A three-terminal dual-pillar STT-MRAM for high-performance robust memory applications," *IEEE Trans. Electron Devices*, vol. 58, no. 5, pp. 1508–1516, May 2011.

[17] S. Kang and K. Lee, "Emerging materials and devices in spintronic integrated circuits for energy-smart mobile computing and connectivity," *Acta Materialia*, vol. 61, no. 3, pp. 952–973, 2013.

[18] X. Fong and K. Roy, "Low-power robust complementary polarizer STT-MRAM (CPSTT) for on-chip caches," in *Proc. 2013 5th IEEE Int. IEEE Memory Workshop*, 2013, pp. 88–91.

[19] X. Fong and K. Roy, "Complimentary polarizers STT-MRAM (CPSTT) for on-chip caches," *IEEE Electron Device Lett.*, vol. 34, no. 2, pp. 232–234, 2013.

[20] H. Naeimi, C. Augustine, A. Raychowdhury, S.-L. Lu, and J. Tschanz, "STT-MRAM scaling and retention failure," *Intel Technol. J.*, vol. 17, no. 1, pp. 54–75, 2013.

[21] P. Khalili and K. Wang, "Voltage-controlled MRAM: Status, challenges and prospects," [Online]. Available: http://www.eetimes.com/document.asp?doc_id=1280508&page_number=1

[22] J. G. Alzate *et al.*, "Voltage-induced switching of nanoscale magnetic tunnel junctions," in *Proc. IEEE Int. Electron Devices Meet.*, 2012, pp. 29–35.

[23] S. Kanai, M. Yamanouchi, S. Ikeda, Y. Nakatani, F. Matsukura, and H. Ohno, "Electric field-induced magnetization reversal in a perpendicular-anisotropy CoFeB/MgO magnetic tunnel junction," *Appl. Phys. Lett.*, vol. 101, no. 12, p. 122403, 2012.

[24] Y. Shiota *et al.*, "Induction of coherent magnetization switching in a few atomic layers of FECO using voltage pulses," *Nature Mater.*, vol. 11, no. 1, pp. 39–43, 2012.

[25] Y. Shiota *et al.*, "Pulse voltage-induced dynamic magnetization switching in magnetic tunneling junctions with high resistance-area product," *Appl. Phys. Lett.*, vol. 101, no. 10, p. 102406, 2012.

[26] P. K. Amiri, P. Upadhyaya, J. Alzate, and K. Wang, "Electric-field-induced thermally assisted switching of monodomain magnetic bits," *J. Appl. Phys.*, vol. 113, no. 1, p. 013912, 2013.

[27] W.-G. Wang, M. Li, S. Hageman, and C. Chien, "Electric-field-assisted switching in magnetic tunnel junctions," *Nature Mater.*, vol. 11, no. 1, pp. 64–68, 2012.

[28] K. Wang, J. Alzate, and P. K. Amiri, "Low-power non-volatile spintronic memory: STT-RAM and beyond," *J. Phys. D, Appl. Phys.*, vol. 46, no. 7, p. 074003, 2013.

[29] H. Noguchi *et al.*, "A 250-mHz 256b-i/o 1-mb STT-RAM with advanced perpendicular MTJ based dual cell for nonvolatile magnetic caches to reduce active power of processors," in *Proc. Symp. IEEE VLSI Technol.*, 2013, pp. C108–C109.

[30] K. Ikegami *et al.*, "A 4 ns, 0.9 v write voltage embedded perpendicular STT-RAM fabricated by MTJ-Last process," in *Proc. Tech. Program-2014 Int. Symp. VLSI Technol. Syst. Appl.*, 2014, pp. 1–2.

[31] H. Noguchi *et al.*, "7.5 a 3.3 ns-access-time 71.2 μw/mhz 1 mb embedded STT-RAM using physically eliminated read-disturb scheme and normally-off memory architecture," in *Proc. IEEE 2015 Int. Solid-State Circuits Conf.*, 2015, pp. 1–3.

[32] R. Dorrance *et al.*, "Diode-MTJ crossbar memory cell using voltage-induced unipolar switching for high-density MRAM," *IEEE Electron Device Lett.*, vol. 34, no. 6, pp. 753–755, Jun. 2013.

[33] F. Oboril, R. Bishnoi, M. Ebrahimi, and M. B. Tahoori, "Evaluation of hybrid memory technologies using SOT-MRAM for on-chip cache hierarchy," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 34, no. 3, pp. 367–380, Mar. 2015.

[34] K. Jabeur, L. Buda-Prejbeanu, G. Prenat, and G. Pendina, "Study of two writing schemes for a magnetic tunnel junction based on spin orbit torque," *Int. J. Electron. Sci. Eng.*, vol. 7, no. 8, pp. 501–507, 2013.

[35] I. M. Miron *et al.*, "Current-driven spin torque induced by the Rashba effect in a ferromagnetic metal layer," *Nature Mater.*, vol. 9, no. 3, pp. 230–234, 2010.

[36] L. Liu *et al.*, "Spin-torque switching with the giant spin hall effect of tantalum," *Science*, vol. 336, no. 6081, pp. 555–558, 2012.

[37] N. Binkert *et al.*, "The gem5 simulator," *ACM SIGARCH Comput. Architect. News*, vol. 39, no. 2, pp. 1–7, 2011.

[38] A. Butko *et al.*, "A trace-driven approach for fast and accurate simulation of manycore architectures," in *Proc. IEEE 20th Asia South Pacific Design Automat. Conf.*, 2015, pp. 707–712.

[39] X. Dong, C. Xu, Y. Xie, and N. P. Jouppi, "NVSim: A circuit-level performance, energy, and area model for emerging nonvolatile memory," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 31, no. 7, pp. 994–1007, Jul. 2012.

[40] M. P. Komalan, C. Tenllado, J. I. G. Pérez, F. T. Fernández, and F. Catthoor, "System level exploration of a STT-MRAM based level 1 data-cache," in *Proc. 2015 Design, Automat. Test Eur. Conf. Exhibit. EDA Consortium*, 2015, pp. 1311–1316.

[41] S. C. Woo, M. Ohara, E. Torrie, J. P. Singh, and A. Gupta, "The SPLASH-2 programs: Characterization and methodological considerations," *ACM SIGARCH Comput. Architect. News*, vol. 23, no. 2, pp. 24–36, 1995.

[42] C. Bienia, S. Kumar, J. P. Singh, and K. Li, "The parsec benchmark suite: Characterization and architectural implications," in *Proc. 17th Int. Conf. Parallel Architect. Compilat. Tech.*, 2008, pp. 72–81.

[43] C. Bienia, S. Kumar, and K. Li, "PARSEC vs. SPLASH-2: A quantitative comparison of two multithreaded benchmark suites on chip-multiprocessors," in *Proc. IEEE Int. Symp. IEEE Workload Characterization*, 2008, pp. 47–56.

[44] X. Dong *et al.*, "Circuit and microarchitecture evaluation of 3d stacking magnetic ram (MRAM) as a universal memory replacement," in *Proc. 45th ACM/IEEE. IEEE Design Automat. Conf.*, 2008, pp. 554–559.

[45] G. Sun, X. Dong, Y. Xie, J. Li, and Y. Chen, "A novel architecture of the 3d stacked MRAM l2 cache for CMPs," in *Proc. IEEE 15th Int. Symp. IEEE High Performance Comput. Architect.*, 2009, pp. 239–249.

[46] X. Wu *et al.*, "Hybrid cache architecture with disparate memory technologies," *ACM SIGARCH Comput. Architect. News*, vol. 37, no. 3, pp. 34–45, 2009.

[47] X. Wu, J. Li, L. Zhang, E. Speight, and Y. Xie, "Power and performance of read-write aware hybrid caches with non-volatile memories," in *IEEE Design, Automat. Test Eur. Conf. Exhibit.*, 2009, pp. 737–742.

[48] J. Li, C. J. Xue, and Y. Xu, "STT-RAM based energy-efficiency hybrid cache for CMPs," in *Proc. IEEE/IFIP 19th Int. Conf. IEEE VLSI System-on-Chip*, 2011, pp. 31–36.

[49] P. Zhou, B. Zhao, J. Yang, and Y. Zhang, "Energy reduction for STT-RAM using early write termination," in *Proc. IEEE/ACM Int. Conf. IEEE Comput.-Aided Design-Digest Tech. Papers*, 2009, pp. 264–268.

[50] K.-W. Kwon, S. H. Choday, Y. Kim, and K. Roy, "AWARE (asymmetric write architecture with redundant blocks): A high write speed STT-MRAM cache architecture," *IEEE Trans. Very Large Scale (VLSI) Syst.*, vol. 22, no. 4, pp. 712–720, Apr. 2014.

[51] A. Jog *et al.*, "Cache revive: Architecting volatile STT-RAM caches for enhanced performance in CMPs," in *Proc. 49th Annu. Design Automat. Conf.*, 2012, pp. 243–252.

[52] E. Arima *et al.*, "Fine-grain power-gating on STT-MRAM peripheral circuits with locality-aware access control," in *Memory Forum*, 2014.

[53] H. Noguchi *et al.*, "Highly reliable and low-power nonvolatile cache memory with advanced perpendicular STT-MRAM for high-performance CPU," in *Proc. 2014 Symp. IEEE VLSI Circuits Dig. Tech. Papers*, 2014, pp. 1–2.

[54] F. A. Endo, D. Couroussé, and H.-P. Charles, "Micro-architectural simulation of embedded core heterogeneity with GEM5 and McPAT," in *Proc. 2015 Workshop Rapid Simulat. Performance Evaluat., Methods Tools*, 2015, p. 7.

[55] P. Greenhalgh, "Big. little processing with arm cortex-a15 & cortex-a7," *ARM White Paper*, pp. 1–8, 2011.

[56] W. Zhao and G. Prenat*, Spintronics-Based Computing*. New York: Springer, 2015.

[57] D. Apalkov *et al.*, "Spin-transfer torque magnetic random access memory (STT-MRAM)," *ACM J. Emerg. Technol. Comput. Syst.*, vol. 9, no. 2, p. 13, 2013.

[58] S. P. Park, S. Gupta, N. Mojumder, A. Raghunathan, and K. Roy, "Future cache design using STT MRAMs for improved energy efficiency: Devices, circuits and architecture," in *Proc. ACM 49th Annu. Design Automat. Conf.*, 2012, pp. 492–497.

[59] C. Augustine *et al.*, "Spin-transfer torque MRAMs for low power memories: Perspective and prospective," *IEEE Sensors J.*, vol. 12, no. 4, pp. 756–766, 2012.

[60] Y. Kim, S. K. Gupta, S. P. Park, G. Panagopoulos, and K. Roy, "Write-optimized reliable design of STT MRAM," in *Proc. 2012 ACM/IEEE Int. Symp. Low Power Electron. Design*, 2012, pp. 3–8.

[61] S. Chatterjee, M. Rasquinha, S. Yalamanchili, and S. Mukhopadhyay, "A scalable design methodology for energy minimization of STT RAM: A circuit and architecture perspective," *IEEE Trans. Very Large Scale (VLSI) Syst.*, vol. 19, no. 5, pp. 809–817, May 2011.

[62] W.-K. Cheng, Y.-H. Ciou, and P.-Y. Shen, "Architecture and data migration methodology for l1 cache design with hybrid SRAM and volatile STT-RAM configuration," *Microprocessors Microsyst.*, 2015.

[63] J.-H. Choi and G.-H. Park, "Demand-aware NVM capacity management policy for hybrid cache architecture," *Comput. J.*, p. Bxv103, 2015.

Sophiane Senni received the M.S. and Ph.D. degrees from the University of Montpellier, Montpellier, France, in 2012 and 2015, respectively.

Currently, he is a postdoctoral researcher at the Montpellier Laboratory of Informatics, Robotics and Microelectronics (LIRMM), France. His research interests are focused on new architectures for non-volatile computing based on magnetic RAM technology.

Lionel Torres received the M.S. and Ph.D. degrees from the University of Montpellier, Montpellier, France, in 1993 and 1996, respectively.

From 1996 to 1997 he was in ATMEL company as IP core methodology R&D Engineer. From 1997 to 2004 he was Assistant Professor at the University of Montpellier, Polytech Montpellier (microelectronic design) and LIRMM laboratory (The Montpellier Laboratory of Informatics, Robotics and Microelectronics—joint unit between CNRS and University of Montpellier). Since 2004 he is full Professor and was at the Head of the Microelectronic Department of the LIRMM from 2007 to 2011. He is now Deputy Head of Polytech Montpellier (engineering school of Montpellier) in charge of research and industrial relationship. Since 2015 he leads the LABEX NUMEV (Laboratory of Excellence) for digital and hardware solutions, modelling for the environment and life sciences. His research interests and skills concern emerging technologies for adaptive architecture. He is involved in different major conferences as DATE, VLSI, FPL, ISVLSI, DAC and is (co)author of more than 40 journal papers and 150 conference publications, and around 10 patents.

Gilles Sassatelli is a Senior CNRS Scientist at LIRMM, a CNRS-University of Montpellier joint research unit. He conducts research in the area of adaptive multiprocessor architectures for embedded systems in the adaptive computing group he leads. He is the author of more than 200 publications in a number of renowned international journals and international conferences. He regularly serves as Track or Topic Chair in major conferences in the field of reconfigurable computing (IEEE FPL, IEEE Reconfig, Worldcomp ERSA, etc.). Most of his research is conducted in collaboration with international partners; over the past five years he has been involved in several national and European research projects including PERPLEXUS (FP6) and MONT-BLANC projects (FP7 and H2020).

Abdoulaye Gamatié received the Ph.D. degree in computer science from Université de Rennes 1, France, in 2004.

He is currently a Senior Researcher at CNRS/LIRMM, France. His research activity focuses on the design of energy-efficient multicore/multiprocessor architectures for embedded and high-performance computing. He is the author of a reference book on synchronous programming of embedded applications with Signal language. He co-authored more than 50 articles in refereed international conferences and journals.

Bruno Mussard graduated in physics from Centrale Marseille, in 1993, and received the Executive MBA degree from Aix-Marseille Graduate School of Management.

He is currently Field Application Engineer at Crocus Technology, promoting and supporting magnetic sensors developed and sold by his company. From 1992 to 2001, he developed general purpose and secure microcontrollers at STMicroelectronics as a Design Engineer. From 2001 to 2010, he managed ARM-based ASICs and microcontrollers hardware development at Atmel Corporation. In 2011, he joined Oberthur Technologies as a Semiconductor Program Manager and in 2012 he became Technical Leader at Crocus Technology, focusing on MRAM-based secure element development.