Logic and Memory Design using Spin-based Circuits

Zhaoxin Liang, Meghna Mankalale, Brandon Del Bel, and Sachin S. Sapatnekar
Department of Electrical and Computer Engineering
University of Minnesota
Minneapolis, MN 55455
e-mail: {zxliang, manka018, sachin}@umn.edu

Abstract— The design of logic and memory circuits in emerging spintronics technology offers fertile ground for new ideas and innovations. We first describe methods for optimizing spintronic logic circuits at the level of physical design, including systematic approaches for building standard cell libraries to enable the design of large circuits. Next, we examine issues in the design of spintronic memories and present methods that trade off volatility with error correction to create dense memory arrays.

I. Introduction

As Moore's law approaches its limits, several post-CMOS technologies are being explored as direct replacements for CMOS, as augmentations to CMOS technologies to build heterogeneous systems, and as platforms for alternative computational models. Spintronic technologies show promise on all of these fronts, with structures that enable logic computations, memory platforms that are dense and nonvolatile, and substrates for neural computation.

In this paper, we will overview several directions in logic and memory design for spintronic circuits. Spintronic devices have been proposed as logic primitives, first by using the idea of differential resistances to translate input states to an output state, and then by transmitting spin-polarized current from a set of inputs, along a channel, to alter an output. In Section II, we outline methods for building and optimizing spin-based logic. Next, in Section III, we present methods for constructing compact spin-based memories, where compact, nonvolatile memory cells can be built by using a structure known as a magnetic tunneling junction that stores the state of a bit cell in the form of a spin orientation and uses a current to change the state of the bit cell. We conclude the paper in Section IV.

II. Logic design

A. Basic logic structures

The essential idea of logic design using spintronics is to perform a spin-based logic operation on the states of a set of inputs, and to set the output to the resulting state. Several schemes for such operations have been proposed using various structures, and they largely rely on the notion of a spin-majority gate. A majority gate delivers an output that corresponds to the majority of its inputs (and therefore, it must have an odd number of inputs). Such gates map more naturally on to some logic functions, and less so to others. For example, the carry output is a majority function of the three inputs of the full adder, and it is possible to build a full adder using two majority gates and five magnets [1, 2].

Fig. 1(a) shows how an AND gate can be implemented using a majority gate. In general, an n-input AND gate is built by augmenting these inputs with $n-1$ inputs fixed to logic 0, for a total of $2n-1$ inputs. The output attains a majority of 1 only when all n gate inputs are at logic 1 and is 0 otherwise. Functionalities such as OR are easily built using a similar scheme, but setting $n-1$ fixed inputs to logic 1, so that the gate output is 1 except when all n inputs are at logic 0. Inversions are easy to implement, as we will soon show, and therefore, since NAND and NOR gates can be built using this method, the logic family is universal.

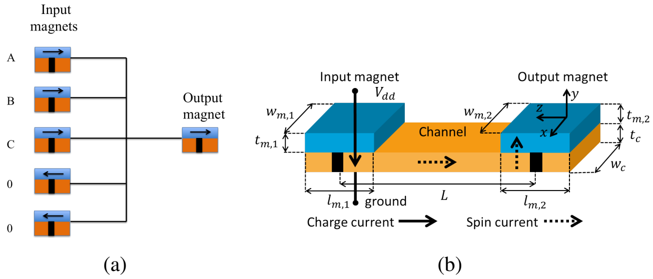


Fig. 1. (a) A three-input AND as a majority gate. (b) A basic ASL inverter.

Several mechanisms for building spin-majority gates have been proposed in the literature. Many of these capture the input logic state in a ferromagnet at the gate input and transmit this state to the output along a nonmagnetic channel. While early approaches performed spin-to-charge conversions to transmit logic states, such conversions were found to be prohibitively power-intensive. More recent approaches attempt to avoid any such conversion: a typical example is the all-spin logic (ASL) structure, illustrated in Fig. 1(b). Other physical mechanisms may be used to achieve the spin-majority structure: for example, the methods in [3] transmit information from the input to the output using the spin-Hall effect, while approaches such as [4] use a domain wall as the communication medium.

A basic ASL gate [5] consists of three components as shown in Fig. 1(b): an input magnet at left that polarizes the charge current and injects spin current into the channel, a channel that transfers the spin current from input magnet to output magnet, and an output magnet that sets its state based on the incoming spin torque. To allow a magnet to serve both as output to its previous magnet and input to its following magnet, an isolation layer is added under it, separating the channel beneath the magnet equally into an input and an output side to ensure minimal interaction between the input and output spin currents.

At the input side of the inverter, a charge current (solid arrow) flows from V_{dd} to ground. The polarizing action of the input magnet results in a spin accumulation, opposite to the magnet spin, at the input end of the gate and this diffuses towards the output (dotted arrow). This creates a spin torque at the output end that sets the output magnet state. A buffer is very similar in structure, except that the role of V_{dd} and ground are interchanged: this ensures that the polarizing action of the input magnet introduces a spin current of the same polarity into the channel. A spin majority gate connects multiple channels, each driven by a different input magnet, to the output magnets. Based on the scheme in Fig. 1(a), the majority spin current delivered to the output depends on the algebraic sum of spin currents from each magnet, and the output is accordingly set. Multioutput gates may also be built using the same principles.

To model the delay and power dissipation, the operation of an ASL gate can be divided into three segments:

- An input magnet converts the charge current between V_{dd} and ground to a spin current in the channel below the magnet. The efficiency of this conversion depends on factor such as the polarization factor of the magnet and its spin accumulation resistance.
- The spin current then diffuses down the channel towards the output magnet(s). The current decays according to a factor known as the spin diffusion length, and the spin current that reaches the end of the channel is a fraction, η, of the injected charge current, I_c: η is referred to as the spin injection efficiency and a typical value in current technologies is below 0.2.
- The polarity of the net spin current at the end of the channel corresponds to the majority of all injected spin currents, and this switches the output magnet, if necessary. The switching mechanism is described by the Landau-Lifschitz-Gilbert (LLG) equation.

B. Performance modeling

This overall process of switching the output logic gate can be modeled by replacing the input and output magnets, as well as elements of the channel, by impedances in the form of π models [6, 7]. This network can be solved to determine waveforms and delays at the output magnet. Simpler models for switching time are also available [8].

Computing circuit delays from gate delays is a relatively straightforward process. As in static timing analysis for CMOS circuits, once the delays of each logic stage (i.e., a gate and its fanout interconnect) are computed, a topological traversal from the primary inputs to the primary outputs can be used to find the delay of the circuit. Given the delay of a gate, the circuit delay can be determined using routine static timing analysis methods.

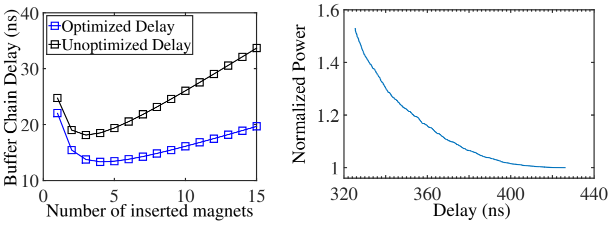
We observe that power is delivered to the system by the V_{dd} source in the form of charge current. Therefore, the power dissipation per gate is V_{dd} times the charge current.

C. Optimizing ASL gates

The timing behavior of an ASL gate can be optimized by sizing magnets in the gate. Specifically, increasing the size of the input magnet of an ASL gate injects a larger amount of charge current, and hence delivers more spin current to the output, increasing the switching speed of this stage. However, this acts as a larger load to the previous stage: a larger number of Bohr magnetons must be switched in the larger magnet, implying that the driving stage becomes slower. This inherent tradeoff can be resolved to obtain an optimal set of magnet sizes.

Further, the switching time has an exponential dependence on the length of the nonmagnetic channel that conveys the spin current. As a result, interconnect is a significant bottleneck in ASL circuits, and adding buffers can help in reducing both the delay and the power dissipation on a long wire.

Fig. 2 shows the delay and corresponding energy of a 1800nm interconnect wire, with input and output magnets of size 20nm, under three cases for a specified number of equally-spaced buffers (n): (i) The length of each magnet is sized individually for optimal delay; (ii) The lengths of all the inserted magnets are assumed to be the same; (iii) The lengths of all inserted magnets are of minimum length 20nm, the buffers are inserted but not optimized.



(a) (b)

Fig. 2. The (a) delay and (b) energy of the buffer chain under three different cases vs. number of inserted buffers.

As shown in Fig. 2(a), the minimum delay occurs when four magnets are inserted, corresponding to a delay of 13.4ns for the case where each inserted magnet is sized individually. As a comparison, the delay with the same number of unsized magnet insertion is 18.5ns, implying that the optimization provides a 27.6% improvement, and the corresponding energy overhead is very small. It is also observed that when all magnets are identically sized, the delay curve virtually coincides with that for the individually-sized case, which implies this closed form could be very useful in predicting the optimal delay. However, the energy in this case is noticeably larger.

On the problem of optimizing general circuits, we present the results of sizing for the C6288 benchmark in Fig. 2(b), assuming a spin diffusion length of 1000nm, a supply voltage of 20mV, a polarization factor of 0.6, and channel widths and thicknesses of 15nm and 25nm, respectively. A clear energy-delay tradeoff can be seen.

D. Creating a design methodology for spin-based logic

The design of large circuits using any logic family requires the availability of fundamental building blocks. For CMOS

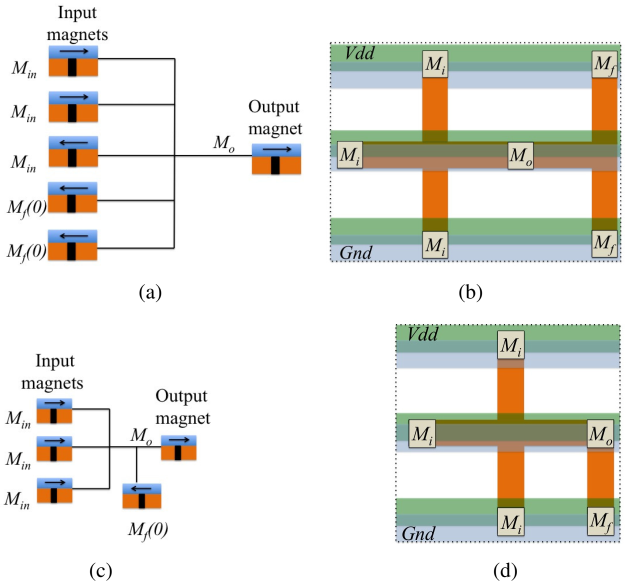


Fig. 3. An ASL AND3 gate using (a) the conventional design (b) its layout. (c) Concept and (d) layout of an improved AND3 ASL gate that optimizes the fixed magnet.

logic circuits, this building block is typically a set of standard cells that implement a set of basic logic functions, and these are typically available in multiple driving powers. The same requirement for a diversity of logic styles is equally applicable to ASL circuits, and as seen in the previous section, sizing the cells helps provide a selection of driving powers, enabling area/delay/power tradeoffs in designing a circuit. A primary difference between CMOS and ASL standard cell functionalities is that the most natural basis for creating logic functions in ASL is the spin-majority gate. Therefore, ASL libraries will depend on this structure and develop functionalities such as NANDs and NORs using the notion presented in Fig. 1(a). Further, just as majority gates cannot be built in a simple way in CMOS, some functionalities that map on easily to CMOS, such as AOI or OAI gates, do not have natural counterparts in ASL.

Standard cells in ASL must obey a set of design constraints. *Geometrical constraints* relate to obeying design rule requirements, ensuring adequate separation between magnets to avoid unintentional dipolar coupling, and ensuring that all standard cells have the same height. *Functional constraints* pertain to ensuring that the gate performs its intended logic function, and relates to ensuring the right contribution of spin currents from the input and fixed magnets.

A careful examination of the task of designing majority gates in ASL shows that it is predicated on a very careful balance between spin current injection, where cell layout choices can significantly impact logical correctness. For example, a structure such as that shown in Fig. 1(a) acts as a majority gate if the spin current contribution through each channel to the output is equal. This requires a careful balance of the geometric parameters of the layout in implementing functional constraints, thus underlining the importance of layout decisions in standard cell design for ASL.

Figs. 3(a) and 3(b) show the layout of an AND3 cell using a strict majority scheme, equalizing the spin currents from three input and two fixed magnets. A closer study of the layout indicates that there are several flexibilities available to the layout designer:

- The large number of fixed magnets ($n-1$ such magnets are required to implement an AND of n values) implies that the layout must be large. As a result, the lengths of the channels can grow large: since the spin signal weakens exponentially with length, this can deliver a major speed disadvantage to gates with many inputs.
- The role of the fixed magnets is to deliver a biasing current that forces all of the three main inputs to be at logic 1 for the output to be 1. Such a bias can be delivered by a single magnet of appropriate size. In fact, if only a single magnet is used, the length of the channel can be significantly decreased, implying that the size of this fixed magnet driving a shorter channel can be smaller than the original fixed magnets.

Based on the above observations, we see that this structure should be considered not as a majority gate with integer inputs, but as a set of competing currents, the problem maps over to the form of a threshold gate [9]. The role of the bias from the fixed magnet is to ensure that the gate switches when n inputs are at logic 1, but not when $n-1$ or fewer inputs are at 1. Therefore, the equivalent switching current has to be of strength equivalent to $n-1$ magnets, plus a safety margin. For a majority gate, this margin corresponds to one input, but the gate can function even when this is smaller. This can yield further savings in the cell size.

The layout of an AND3 ASL gate using this improved scheme is shown in Figs. 3(c) and 3(d), where a single fixed magnet is used and the overall footprint of the layout is significantly decreased, and this also translates to a speed improvement. On average, our approach yields 7.0% faster AND2 and XOR2 devices, 37.0% faster AND3, and 63.3% AND4 devices over conventional structures. Consequently, AND2, AND3 and AND4 are an average of 2.40%, 49.9% and 84.4% more energy efficient as a result of our optimization, and our layouts occupy 19.1% smaller area on average. Since the three-input majority gate (MAJ3) and five-input majority gate (MAJ5) use no fixed magnets, there is no area improvement over their conventional implementations, and their layouts are similar to AND2 and AND4, respectively.

A standard logic flow (e.g., using SIS, ABC, or Synopsys Design Compiler) may be leveraged with this set of logic gates to map a circuit to the library. The resulting circuits, on average, are 19.6% faster, consume 17.7% less energy and 33.5% more area efficient compared to the conventional approach. These improvements can largely be credited to the elimination of the additional channel segments for the multiple fixed magnets used in the conventional approach.

III. Memory design

A. Spintronic memories: an overview

The principal element of an STT-MRAM is the magnetic tunnel junction (MTJ), which consists of two ferromagnetic layers, a free layer and a fixed layer, separated by a thin tunneling barrier. By passing a charge current through this structure, the fixed layer creates a spin-polarized current that transmits a spin-transfer torque (STT) to set the magnetization in the free layer: the direction of the current and the polarization of the fixed layer determine the direction of free layer magnetization. The resulting configuration may be have parallel or anti-parallel magnetizations for the free and fixed layers, corresponding to distinct states that correspond to different resistances. Thus, the structure shows two states depending on the magnetization of the free layer, it can be programmed through an STT-inducing current, and the state can be read by building structures that sense the differential between the resistances for the parallel and anti-parallel states. Further, since the MTJ retains its magnetization in the absence of a power source, it has the attractive property of nonvolatility.

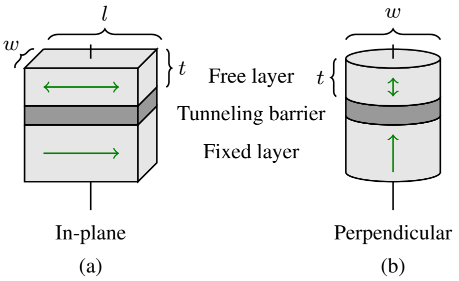


Fig. 4. Schematic of an (a) in-plane and (b) perpendicular MTJ.

MTJs come in three flavors, depending on the nature of their anisotropy and the orientation of their magnetization relative to the substrate. The earliest applications of MTJs were to memory structures: the resulting STT-MRAMs are attractive because of their compact design, nonvolatility, low leakage power, and their potential for scalability. For memory applications, all three types of MTJs are similarly deployed in bit cells, but they differ in attributes such as volume, aspect ratio, switching current, and scalability in future technologies [10–12].

Several architectures have been proposed for STT-MRAM cells, but fundamentally all consist of an MTJ, an access transistor, and contacts for the word line (WL), bit line (BL), and source line (SL). The differences arise from the way the MTJ is connected to the transistor, the number of transistors used, and the cell layout. Common designs use one access transistor TX and an MTJ (1T1MTJ) (Fig. 5(a)) or two transistors TX1, TX2, and an MTJ (2T1MTJ) (Fig. 5(b)) with the free layer connected to the bit line BL [13, 14]. The MTJ stores state using free layer polarization relative to a fixed layer. The two states – parallel and antiparallel – have different resistances, allowing for read operations with voltage sense amplifiers after passing a small current pulse through the access transistor TX. The fixed layer is also used to write the free layer by either passing a larger current or applying a longer pulse (as compared to the read operation) through TX.

Initial work on STT-MRAMs has focused on the use of *in-plane* MTJs (IMTJs), where the direction of magnetization is in the plane of the magnet, as shown in Fig. 4(a)). The magnet is in the shape of a rectangular cuboid with thickness t, width w, and aspect ratio AR in the orthogonal in-plane dimension.

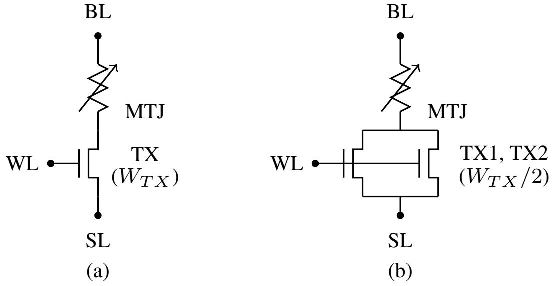


Fig. 5. 1T1MTJ (a) and 2T1MTJ (b) bit cell schematics.

For the magnet to function correctly, it is crucial for the aspect ratio to be sufficiently different from 1 in order to enable an appropriate demagnetizing field.

Recent work has also addressed the less mature technology based on *crystalline anisotropy* and *interfacial anisotropy perpendicular* MTJs (c-PMTJs and i-PMTJs), which have lower switching current densities than IMTJs at existing technology nodes. Perpendicular MTJs have a magnetization direction that is perpendicular to the plane of the magnet and are typically cylindrical in shape, with diameter w and thickness t, as shown in Fig. 4(b).

B. Nonvolatility and thermal stability

Typically, MTJs are designed to provide high levels of nonvolatility due to a significant energy barrier between the logic 0 and logic 1 states that is maintained even when the system is powered off. The robustness and nonvolatility of the memory can be enhanced by maintaining the energy barrier at a high level. This optimization involves an increase the size of the memory cell by building a larger MTJ and a larger access transistor that provides the switching current required to surmount the barrier during an intentional write operation.

In the absence of spin current, the magnetization of the free layer may take one of two stable states, separated by an energy barrier, as illustrated in Fig. 6. In order to change state, sufficient energy has to be provided to the MTJ to overcome the energy barrier and cross over to the opposite state.

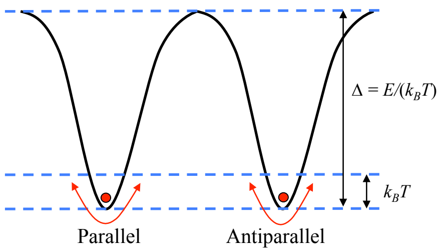


Fig. 6. Thermal fluctuations in an MTJ free layer [14].

The height E of the energy barrier is a key attribute of an MTJ, and is referred to as its thermal stability factor, Δ (often referred to informally as the thermal stability). This is typically expressed in multiples of $k_B T$, where k_B is the Boltzmann constant and T is the absolute temperature, as $\Delta = E/k_B T$.

C. Errors in MTJs

A low error rate is an important property of a reliable memory array. There are three types of errors based on unintentional changes to the free layer magnetization orientation in an MTJ – retention errors, write errors, and read errors – and all of them bear a relation to the thermal stability. As we will see in the ensuing discussion, a higher value of Δ is favorable for reducing standby and read errors, while a lower value facilitates low-energy write operations.

Random *retention errors* occur in the absence of MTJ current and are caused by thermal noise. When the memory is in standby mode, a higher energy barrier Δ indicates that it is less likely that random thermal noise can generate enough energy to surmount the barrier and reverse the state of the MTJ, reducing the probability of retention errors and improving its nonvolatility properties.

Write errors are caused by passing too little current while writing the state of the memory. The mechanism of a write operation is that by passing a current through the MTJ, a spin torque is generated. The magnitude of the current influences the magnitude of spin torque. To surmount the energy barrier and effect a change in the state of the MTJ, a low energy barrier is favored.

Read errors are caused by passing too much current while reading the state of the memory. During a read operation, a current is applied to determine the state of the MTJ, using a sense amplifier to distinguish between the resistance of the MTJ in the parallel and antiparallel states. The supplied current must be sufficient to detect the resistance value, but not so much that it applies a spin torque that inadvertently reverses the state of the memory element. Therefore, safe read operations are favored by a high value of Δ. However, read error probabilities can be made negligible by suitably choosing the read current profile and characteristics [14, 15].

D. Trading off error probabilities and ECC overheads

It is possible to save cell area and increase memory density by using smaller cells. However, by itself such an approach results in unsatisfactory designs since the reduction in size has the immediate consequence of raising the error rate due to a reduced levels of nonvolatility. For some designs this is acceptable: e.g., on-chip caches have relaxed retention time requirements [16] since data does not reside in them for very long, but given any baseline retention rate, density can be increased by using smaller cells at the cost of reduced nonvolatility.

To recover from retention errors due to this factor, we use error-correcting codes (ECCs) to inoculate the memory from such errors and maintain the overall nonvolatility of the memory array by expending some of the saved area into ECC bits. Such codes require additional bits, and we present a scheme that shows that the increase in area due to the additional ECC bits is far outweighed by the area reduction as cell sizes are reduced. In particular, unlike traditional error correction for memories that captures single-bit errors, we consider strong error correction that can potentially correct for multiple errors.

An ECC can be characterized by the following parameters: the number of symbols prior to encoding, k, the number of symbols after encoding, n, and the number of correctable symbols, c, for the encoded data. For normal binary codes such as those used to store data in memory, each bit is a symbol. As c increases, more additional symbols are required for correction, and therefore n increases [17]. We consider block error correcting codes, based on Bose–Chaudhuri–Hocquenghem (BCH) codes, which function by transforming a fixed-size data block into another fixed-size data block by adding error-correcting bits that can potentially correct for multibit errors.

Consider a memory with m bits and a block size of n bits, the number of blocks $b = m/n$. Typically, m/n is an integer because data blocks correspond to the fundamental elements of a memory, such as words, lines, and pages. Each such fundamental element consists of k data bits and $(n - k)$ ECC bits that are used to correct up to c errors in the block. A block is error-free under a BCH ECC with c correctable symbols if the number of errors in the block does not exceed c. If $P_{bit}(t)$ is the error probability in a single bit cell t seconds after it was last written into, then the probability, P_{blk}, that a block is error-free is given by:

$$P_{blk}(t) = \sum_{i=0}^{c} \binom{n}{i} (1 - P_{bit}(t))^{i} (P_{bit}(t))^{n-i} \qquad (1)$$

For an STT-MRAM array, the probability that none of the b blocks experiences an error is given by $(P_{blk}(t))^{b}$. For a given value of the retention time, t_r, the failure rate, λ, is given by the complement of this value. The corresponding failure rate for a period of t_r seconds (which can be translated from a FIT specification for the memory) is:

$$\lambda = \frac{1 - (P_{blk}(t_r))^{b}}{t_r} \qquad (2)$$

E. Tradeoff between retention failures and write failures

At a constant bit error probability P_{bit}, there is a family of allowable values that constitute a tradeoff between P_{wr} and P_{ret}. The value of P_{ret} is related to the thermal stability, Δ, and a higher energy barrier implies a greater retention probability. At a fixed P_{bit}, this value translates to a specific value of P_{wr}, and this maps on to a precessional angle, θ, illustrated in Fig. 7. A smaller value of P_{wr} corresponds to a larger θ, and as seen from Fig. 7, this means that part of the energy barrier has already been scaled by precession. However, this smaller value of P_{wr} also translates to a larger P_{ret}, which implies a larger energy barrier, Δ, to be surmounted. Depending on which of these two contributions is stronger, the write current may be larger or smaller as P_{wr} is reduced, and this is the inherent tradeoff between the retention and write failures. Since the write current is proportional to the cell size, this tradeoff can be used to determine the compactness of the memory cell.

Therefore, there is a three-way tradeoff between retention failures, ECC overheads, and write failures that can be explored to optimize the STT-RAM memory. The idea of retention-ECC tradeoffs have been explored in [18], and significant density improvements are available. For example, for a 32MB on-chip memory with 512b blocks, a retention time requirement of 10 years, and failure rate of 1 FIT, an area reduction of over 40%

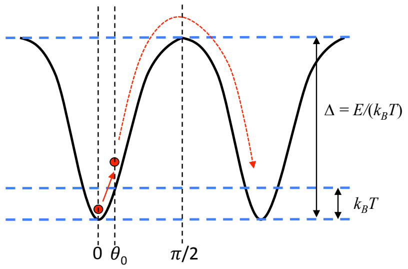


Fig. 7. The tradeoff between the write and retention error probability: for two initial precession angle θ of 0 and θ_0 illustrated here, under a constant write current I and switching time t_{sw}, the former has a higher write failure probability.

is seen over conventional STT-RAM design without the benefit of multibit ECC; these savings account for the overhead of additional ECC bits and the BCH codec area. Considering that STT-RAM cells are roughly 4× as dense as SRAMs, this is a significant advantage over SRAMs. For a 32Gb memory with 4kb blocks, the area savings exceed 50%.

IV. Conclusion

Spintronic technologies show considerable promise as a post-CMOS alternative, and they are already viable for memory applications. Today's spin-based logic is not as fast and energy-efficient as CMOS [8] (but at this time, no other post-CMOS device is). However, multidisciplinary efforts are actively being pursued to close this gap. There are numerous ongoing challenges and opportunities in various aspects of this field, ranging from physics to materials to circuits to architectures. For example, new materials and physical mechanisms, some of which are currently under exploration, have the potential to provide improvements of several orders of magnitude in energy; new circuit schemes utilizing alternative mechanisms such as the magnetoelectric effect or voltage-controlled magnetic anisotropy could be leveraged to build faster or more energy-efficient gates; new architectures that use logic-in-memory computations could be used to accelerate memory-intensive computations.

V. Acknowledgment

This work was supported in part by C-SPIN, one of the six SRC STARnet Centers, sponsored by MARCO and DARPA.

References

[1] H. M. Martin, "Threshold logic for integrated full adder and the like," 1971. US Patent 3,609,329.

[2] C. Augustine, G. Panagopoulos, B. Behin-Aein, S. Srinivasan, A. Sarkar, and K. Roy, "Low-power functionality enhanced computation architecture using spin-based devices," in *Proceedings of the IEEE/ACM International Symposium on Nanoscale Architectures*, pp. 129–136, 2011.

[3] S. Datta, S. Salahuddin, and B. Behin-Aein, "Non-volatile spin switch for boolean and non-boolean logic," *Applied Physics Letters*, vol. 101, no. 25, pp. 252411–1 – 252411–5, 2012.

[4] X. Yao, J. Harms, A. Lyle, F. Ebrahimi, Y. Zhang, and J.-P. Wang, "Magnetic tunnel junction-based spintronic logic units operated by spin transfer torque," *IEEE Transactions on Nanotechnology*, vol. 11, pp. 120–126, Jan 2012.

[5] B. Behin-Aein, D. Datta, S. Salahuddin, and S. Datta, "Proposal for an all-spin logic device with built-in memory," *Nature Nanotechnology*, vol. 5, no. 4, pp. 266–270, 2010.

[6] S. Srinivasan, V. Diep, B. Behin-Aein, A. Sarkar, and S. Datta, "Modeling multi-magnet networks interacting via spin currents," 2013. arXiv preprint arXiv:1304.0742.

[7] S. Manipatruni, D. E. Nikonov, I. Young, *et al.*, "Modeling and design of spintronic integrated circuits," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 59, no. 12, pp. 2801–2814, 2012.

[8] J. Kim, A. Paul, P. Crowell, S. J. Koester, S. S. Sapatnekar, J.-P. Wang, and C. H. Kim, "Spin-based computing: Device concepts, current status, and a case study on a microprocessor," *Proceedings of the IEEE*, vol. 103, pp. 106–130, Jan. 2015.

[9] S. Muroga, *Threshold Logic and its Applications*. New York, NY: Wiley, 1971.

[10] K. C. Chun, H. Zhao, J. D. Harms, T.-H. Kim, J.-P. Wang, and C. H. Kim, "A scaling roadmap and performance evaluation of in-plane and perpendicular MTJ based STT-MRAMs for high-density cache memory," *IEEE Journal of Solid-State Circuits*, vol. 48, pp. 598–610, 2013.

[11] R. Dorrance, F. Ren, Y. Toriyama, A. A. Hafez, C.-K. K. Yang, and D. Markovic, "Scalability and design-space analysis of a 1T-1MTJ memory cell for STT-RAMs," *IEEE Transactions on Electron Devices*, vol. 59, pp. 878–887, 2012.

[12] J. Kim, H. Zhao, Y. Jiang, A. Klemm, J.-P. Wang, and C. H. Kim, "Scaling analysis of in-plane and perpendicular anisotropy magnetic tunnel junctions using a physics-based model," in *Proceedings of the Device Research Conference*, pp. 155–156, 2014.

[13] S. P. Park, S. Gupta, N. Mojumder, A. Raghunathan, and K. Roy, "Future cache design using STT MRAMs for improved energy efficiency," in *Proceedings of the ACM/EDAC/IEEE Design Automation Conference*, pp. 492–497, 2012.

[14] R. Takemura, T. Kawahara, K. Miura, H. Yamamoto, J. Hayakawa, N. Mitsuzata, K. Ono, M. Yamanouchi, K. Ito, H. Takahashi, S. Ikeda, H. Hasegawa, H. Matsuoka, and H. Ohno, "A 32-Mb SPRAM with 2T1R memory cell, localized bi-directional write driver and '1'/'0' dual-array equalized reference scheme," *IEEE Journal of Solid-State Circuits*, vol. 45, pp. 869–879, 2010.

[15] A. Raychowdhury, D. Somasekhar, T. Karnik, and V. K. De, "Modeling and analysis of read disturb (RD) in 1T-1STT MTJ memory bits," in *Proceedings of the Device Research Conference*, pp. 43–44, 2010.

[16] C. W. Smullen, IV, V. Mohan, A. Nigam, S. Gurumurthi, and M. R. Stan, "Relaxing non-volatility for fast and energy-efficient STT-RAM caches," in *Proceedings of the IEEE Symposium on High Performance Computer Architecture*, pp. 50–61, 2011.

[17] R. H. Morelos-Zaragoza, *The Art of Error Correcting Coding*. New York, NY: John Wiley, 2006.

[18] B. Del Bel, J. Kim, C. H. Kim, and S. S. Sapatnekar, "Improving STT-MRAM density through multibit error correction," in *Proceedings of the Design, Automation & Test in Europe*, 2014.