

# A Variation-Tolerant MRAM-Backed-SRAM Cell for a Nonvolatile Dynamically Reconfigurable FPGA

A. Vatankehaghadim, *Student Member, IEEE*, W. Song, *Member, IEEE*, and A. Sheikholeslami, *Senior Member, IEEE*

**Abstract**—Adding a spin-transfer-torque (STT) magnetoresistive random-access memory (MRAM) to a static random-access memory (SRAM) cell to produce an MRAM-backed SRAM cell for a nonvolatile field-programmable gate array (FPGA) is proposed. The proposed cell reduces the time to reconfigure the FPGA following a power-down and enables fast wake-ups and power gating. With the proposed restore operation, data are recalled with no error even in the presence of mismatch. Simulation results confirm that data can be stored in the proposed cell in 80 ns and restored in less than 1 ns.

**Index Terms**—Field-programmable gate arrays (FPGAs), magnetic tunnel junction (MTJ), magnetoresistive random-access memory (MRAM), nonvolatile (NV), spin-transfer-torque (STT), static random-access memory (SRAM).

## I. INTRODUCTION

MOST of the current field-programmable gate arrays (FPGAs) use static random-access memory (SRAM) cells to configure the lookup tables (LUTs) and multiplexers (MUXs) in configurable logic blocks (CLBs) and routings [1], as shown in Fig. 1. Some proposals suggest using dynamic random-access memory (DRAM) cells instead [2], as it is not prone to soft errors, but SRAM cells are still widely used as they are faster than DRAM cells and require no refreshing. However, SRAM cells are volatile, and as such, after each power-down, configuration bits are serially received from an external nonvolatile (NV) memory. As a result, the setup and configuration of the FPGA is a timing bottleneck. One way to eliminate this timing bottleneck is to store the configuration bits locally (in NV cells) next to SRAM cells. This arrangement, which results in fast power-up, also enables power-saving techniques using deliberate power-downs.

To implement NV-FPGAs, previous works employ Flash memory [3]. However, Flash suffers from high program/erase voltages, low write endurance ( $10^5$ ), and high write access time (0.1–10 ms). Deploying other types of NV memory cells such as magnetoresistive random-access memory (MRAM) for NV-FPGAs resulted in works of [4]–[6]. Reference [4] is based on field-induced magnetic switching MRAM, whereas [5] and [6] use thermally assisted switching MRAM. These previous generations of MRAM are less scalable and require more

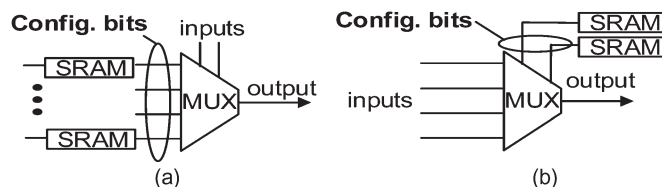


Fig. 1. CLB and routing elements. (a) Conventional LUT. (b) MUX. While the configuration bits set the content of the LUT in CLB, they determine which input (path) gets connected to the output routing.

switching current compared to the most recent generation, i.e., spin-transfer-torque (STT) MRAM.

The STT-MRAM is a prime candidate for a universal memory, as it accommodates high read/write endurance of  $10^{15}$  and low access time (less than 10 ns) [7]. In this regard, some hybrid CMOS-STTRAM structures are proposed to take advantage of STT-MRAM properties [8], [9]. These designs distribute STT-MRAM cells through the CLBs, but they do not maintain the functionality of individual SRAM cells. In addition, they cannot be used for storing the data directly from the cell before power-downs, and data should be stored in STT-MRAM cells every time it changes. On the other hand, previous NV-SRAM cells [10]–[13], either occupy a large area [10], [11] or cannot be reconfigured in the background [10]–[13]. More importantly, they are prone to variations, and restoring the data after power-up will fail in the presence of mismatch. To address these issues, we propose a new NV-SRAM cell, which we call an MRAM-backed SRAM cell. The proposed cell is suitable for fast wake-ups and power gating; it is variation tolerant and dynamically reconfigurable in the background.

This brief is organized as follows. Section II provides some background on the conventional STT-MRAM cell, the shadow structure, and the previous work on NV-SRAM. Section III describes the proposed cell and its different modes of operation. In addition, the transistor sizing is discussed, and an alternative store operation is presented. Section IV presents simulation results of the cell and the timings for different operation modes, as well as Monte Carlo simulation results. Furthermore, in Section V, the proposed cell is compared against the other NV-SRAM cells in detail. Section VI concludes this brief.

## II. BACKGROUND

The conventional STT-MRAM cell consists of a magnetic tunnel junction (MTJ) and an access transistor, as shown in Fig. 2. An MTJ, as the main element of the memory cell, consists of two ferromagnetic layers with a thin insulating layer in between. It includes one pinned magnetized layer and one free layer, whose magnetization can be changed by spin-polarized current in the process of writing to the memory. Depending

Manuscript received November 11, 2014; accepted January 13, 2015. Date of publication February 27, 2015; date of current version May 29, 2015. This work was supported by the Natural Sciences and Engineering Research Council of Canada. This brief was recommended by Associate Editor J. G. Delgado-Frias.

The authors are with the Department of Electrical and Computer Engineering, University of Toronto, Toronto, ON M5S 3G4, Canada.

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

Digital Object Identifier 10.1109/TCSIL.2015.2407711

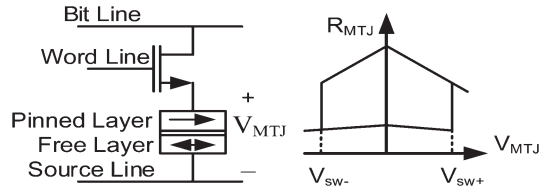


Fig. 2. Conventional STT-MRAM cell and MTJ characteristic.

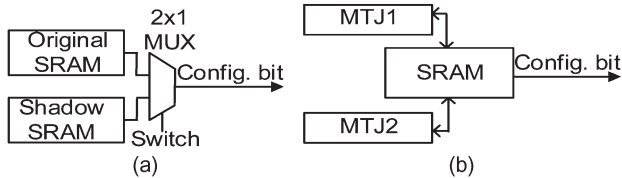


Fig. 3. (a) Conventional shadow structure [14]. (b) Proposed shadow structure.

on the direction of the current, magnetization of the free layer will be aligned either parallel or antiparallel to the pinned layer. The read operation involves measuring the effective resistance of the MTJ in two different states representing “0” and “1.” The resistance between the pinned layer and the free layer is higher when the layers have their magnetization in antiparallel state ( $R_{AP}$ ) than when they are in parallel ( $R_P$ ). The tunneling magnetoresistance ratio (TMR) is defined as  $(R_{AP} - R_P)/R_P$ .

**Shadow Structure:** Dynamic reconfiguration enables time-sharing of design functionalities. This reduces size and cost while increasing flexibility of the design. Changing functionality during run time is beneficial for multicontext FPGAs and video/image processing applications. A shadow structure, as shown in Fig. 3(a), enables selecting between two SRAM cells. In addition to original cells, shadow cells can be added to enable dynamic reconfiguration without latency [14]. With this, the normal operation will not be interrupted when shadow cells are being reconfigured. In our proposed design, we will use a shadow implementation that consists of an SRAM cell plus two MTJs, as illustrated in Fig. 3(b).

**Previous Work:** We briefly review and critique four previously proposed NV-SRAM cells using MTJs, as shown in Fig. 4. Spin-RAM structure [see Fig. 4(a)] [10] combines SRAM with MTJs. After equalizing the outputs, it uses SRAM to sense the resistance difference between the two MTJs and to latch the stored data. However, the restore operation is likely to fail due to the mismatch between the two transistors in the nMOS or the pMOS pairs, as well as the variation of the MTJ resistance. While the former reduces the sensing margin in the back-to-back inverters, the later reduces the TMR by making the  $R_{AP}$  of one MTJ and the  $R_P$  of the other one closer to each other. If no voltage difference is developed before the SRAM sensing kicks in, these mismatch effects will result in restore failure. This is the case in the design of Fig. 4(a), where the mismatch in the nMOS (pMOS) pair is present immediately at the beginning of the restore operation and could lead to restore error. Furthermore, the MTJs are active even during the read/write operation. A shadow structure cannot be implemented in this design since MTJs are not isolated from the SRAM cell. In addition, data of storage nodes cannot be used to store data in MTJs. As a result, data should be stored every time it changes (i.e., cannot be limited to before each power-down).

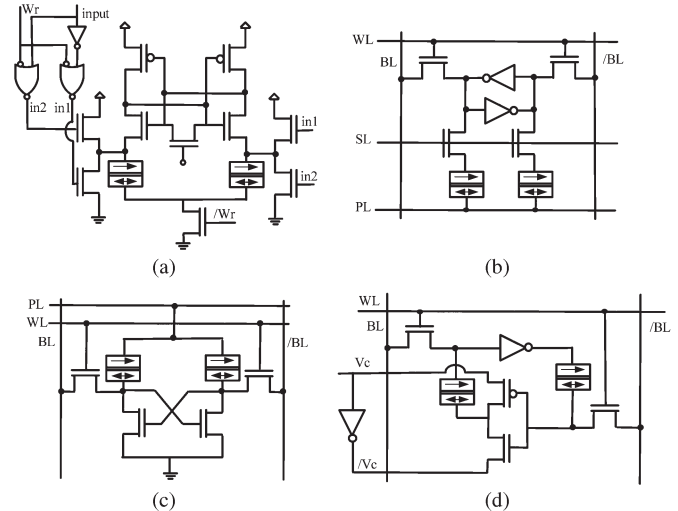


Fig. 4. Cell structure of (a) spin-RAM [10], (b) 8T2MTJ NV-SRAM [11], (c) 4T2MTJ NV-SRAM [12], and (d) 6T2MTJ NV-SRAM [13].

Fig. 4(b) illustrates an 8T2MTJ structure [11]. MTJs are isolated from the SRAM cell during read/write, and adding another pair of MTJs for shadow bits is feasible. However, the store operation is performed in two steps by setting the PL to VDD and then to GND to set the state of two MTJs. This increases the configuration time. To restore the data, while SL is high,  $V_{supply}$  of the inverters are ramped gradually to VDD. As a result, while the storage nodes are being charged by the pull-up transistors of the two inverters, they are discharged by the current through the MTJs; then, the storage nodes are set due to the difference in the current drivability of the two branches. There are two drawbacks in this implementation. First, the rate of the VDD ramp is susceptible to  $PVT$  variation and to the mismatch, which may both lead to restore failure. Second, since accessing the MTJs is through the storage nodes, dynamic reconfiguration of the MTJs is not possible during normal operation.

Fig. 4(c) shows a 4T2MTJ structure [12], which uses cross-coupled nMOS transistors instead of back-to-back inverters, reducing the number of transistors. Unlike the previous two designs, this design is more tolerant of the mismatch because PL is gradually increased to allow the development of voltage difference at the sense nodes before positive feedback fully kicks in. Similar to the structure shown in Fig. 4(b), the store operation has two steps. However, the store current is provided through BLs. In addition, similar to Fig. 4(a), implementation of a shadow structure is not feasible, and a store operation is necessary after each data change. Moreover, dynamic reconfiguration of the MTJs is not possible during normal operation.

Fig. 4(d) shows a 6T2MTJ NV-SRAM cell [13], which has its MTJ cell between the input and the output of the back-to-back inverters. Signals with short pulsewidths are applied across the lower inverter to make it behave like a resistor during the store operation. Restore is performed based on the difference in resistance of the MTJs. However, this is also prone to failure due to mismatch. This architecture can be augmented with another branch to provide a shadow structure, but dynamic reconfiguration of MTJs during run time is not possible.

In the next section, we propose an MRAM-backed SRAM cell that has a shadow structure, inactive MTJs during read/write,

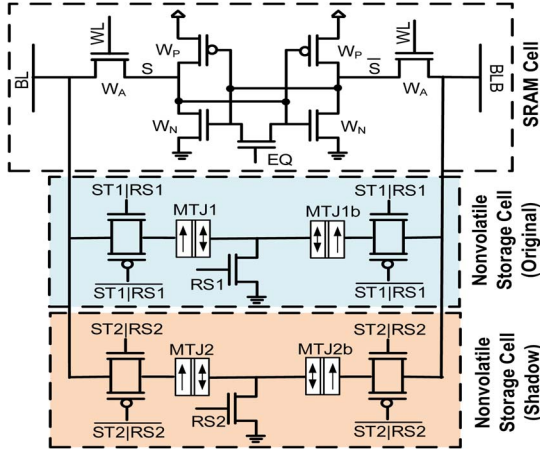


Fig. 5. Proposed MRAM-backed SRAM cell structure.

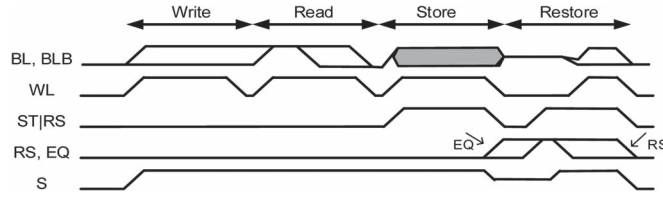


Fig. 6. Timing diagram for different modes of operation.

tolerance to mismatch during restore, as well as capability to dynamically reconfigure the MTJs in the background.

### III. PROPOSED MRAM-BACKED SRAM CELL

The proposed cell structure consists of a conventional SRAM cell and two STT-MRAM storage cells (one original and one shadow cell), as shown in Fig. 5. The SRAM part includes two back-to-back inverters along with access transistors. An EQ transistor is also included for equalization in the restore operation (more details will be provided in Section III-A). MTJ1 and MTJ1b are used to store an original bit and its complement, whereas MTJ2 and MTJ2b are used for shadow bits. At power-down, the SRAM cell state will be stored in either MTJ1 or MTJ2 pair, and at power-up, either MTJ1 or MTJ2 pair will restore data to the cell.

The cell operation in one of four modes is discussed next. Subsequently, we will discuss the sizing of the transistors and propose an alternative store operation.

#### A. Modes of Operation

The proposed cell operates in one of the four modes of read, write, store, and restore. While read and write operations occur with respect to the SRAM cell, the store and restore operations occur with respect to MTJs. The timing diagram shown in Fig. 6 (not to scale) illustrates these four modes of operation.

*Writing* the data to the cell and *reading* it from the cell are the same as those for the conventional SRAM cell. During the read/write operation, the MTJs are fully isolated from the bitlines (BL and BLB) and experience zero volts across.

*Storing* the state of the cell prior to power-down is achieved by setting the magnetization vectors of the MTJ pair to parallel and antiparallel according to the data. Control signals of

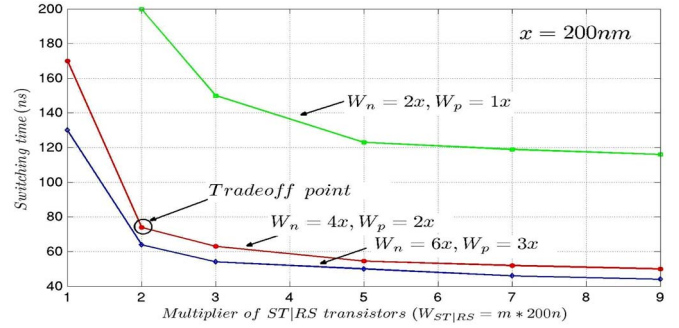


Fig. 7. Tradeoff between size and switching time of MTJs.

ST1|RS1 and ST2|RS2 decide whether the shadow or the original pairs will be used to store the data. The SRAM cell provides necessary current to set the MTJs to parallel/antiparallel states. In case of activation of the original pair, the data at  $S$  and  $\bar{S}$  will be stored in MTJ1 and MTJ1b, respectively. If  $S = "1"$ , the store operation results in an antiparallel MTJ1 and a parallel MTJ1b. Conversely,  $S = "0"$  results in a parallel MTJ1 and an antiparallel MTJ1b. To store the data in the shadow cell, we will activate ST2|RS2 instead of ST1|RS1, and the same procedure will take place for MTJ2 and MTJ2b.

To *restore*, both BL and BLB are first precharged to about  $VDD/2$ . Then, RS1 (or RS2) is activated to ground the node between the two MTJs. With this, two different resistance values will be presented to BL and BLB. The side with lower resistance, i.e.,  $R_P$ , will fall faster than the other side. When enough voltage difference is developed between BL and BLB [15], WL is activated to connect the already equalized storage nodes to BL and BLB. This will push the storage node of the side with lower resistance toward 0, whereas the other side with higher resistance ( $R_{AP}$ ) drives its storage node to VDD. Because of back-to-back inverters, the slight differential voltage between  $S$  and  $\bar{S}$  will grow to full VDD, setting the storage nodes to the stored data.

#### B. Sizing the Transistors

There are several factors to consider when sizing the transistors of the proposed cell. Although minimum size transistors are desired for small cell area, factors such as stability of the cell and necessary drive current of the MTJs for a certain store time set different requirements on sizing.

The 65-nm CMOS technology is used for simulations. To have a stable read/write operation, we set the size to  $W_N = 2W_P = 2W_A$  (see Fig. 5), where  $W$  refers to the transistor width. Read/write stability analyses confirm 0.15/0.25 V of read/write noise margin at the typical process corner (TT) and 0.1/0.25 V at the worst case process corner (FF). We size the transmission gates (ST|RS transistors) such that sufficient current pass through the MTJs during the store operation (to achieve certain store times). For three values of  $W_P$  (i.e.,  $1x$ ,  $2x$ , and  $3x$ , where  $x = 200$  nm), we sweep  $m$  in  $W_{ST|RS} = mx$  and plot in Fig. 7 the switching time as a function of  $m$ . There is a tradeoff in which a higher  $m$  results in a larger cell area but reduces the switching time (store time) due to increased current through the MTJs. We choose  $W_{ST|RS} = W_P = 2x$  to provide switching time of less than 80 ns.

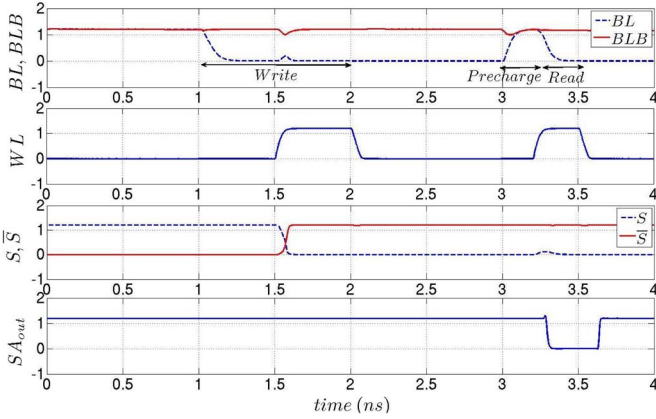


Fig. 8. Read/write operation.

The size of the proposed cell is more than twice the size of the conventional SRAM cell with the added benefit of nonvolatility. Next, we discuss the limitations and the tradeoff of the proposed cell and present an alternative store operation.

### C. Alternative Store Operation

A store operation via back-to-back inverters does not require storing the data in MTJs after each change, as storing the data in MTJs is only necessary before each power-down using the data of the storage nodes. This eliminates timing overhead and reduces the power consumption. However, the proposed cell is costly in terms of size unless an alternative store operation is used. This is because the store operation via back-to-back inverters (Store1) requires the cell to provide sufficient current for MTJ switching. To reduce the size, we propose a different store operation. The current to switch MTJ states is no longer provided by the back-to-back inverters of the cell but from BL and BLB through the shared column drivers (Store2). With this, upsizing the transistors of SRAM cell is no longer necessary, and the widths of the transistors can be halved. This results in a cell almost the same size of the conventional SRAM cell. We have laid out a DRC-clean version of each of SRAM cells with Store1 and Store2, and we observed a cell area of  $2.3x$  and  $1.5x$ , respectively, relative to the basic SRAM cell. However, having to store the data via drivers, and not the cell itself, requires storing the data in MTJs every time it is changed. This does not affect the reconfiguration time, as it is dynamically processed in parallel at the background while the system continues its normal operation, but it results in extra power consumption.

In summary, the desired store operation is the one through the back-to-back inverters of the SRAM cell, as it eliminates the need for frequent store of the data to MTJs. However, for this, MTJ properties should be improved to switch with lower current (eliminating the need for upsizing the SRAM cell transistors). Otherwise, the area penalty would be inevitable.

## IV. SIMULATION RESULTS

For simulations, we employ an MTJ model developed in Verilog-A [16] using the Landau–Lifshitz–Gilbert–Slonczewski equation along with Spectre. A  $50 \text{ nm} \times 50 \text{ nm}$  MTJ device with  $R_P = 4.2\text{K}$  and  $R_{AP} = 6.8\text{K}$  is assumed.

Fig. 8 shows signals for the *write* operation followed by *read*. Once BL and BLB are set to GND and VDD, respectively,

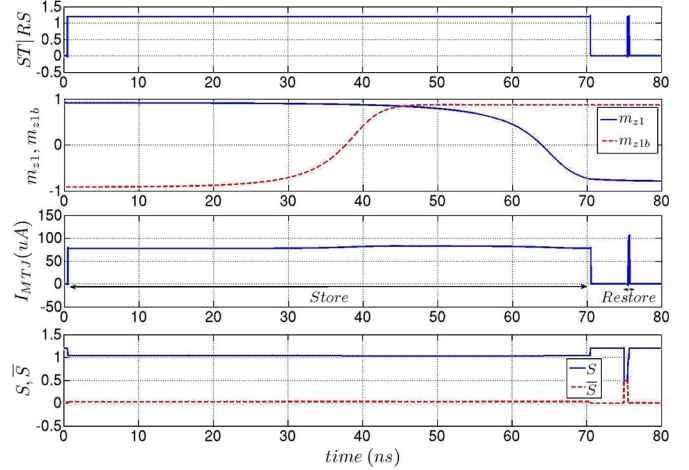


Fig. 9. Store/restore operation.

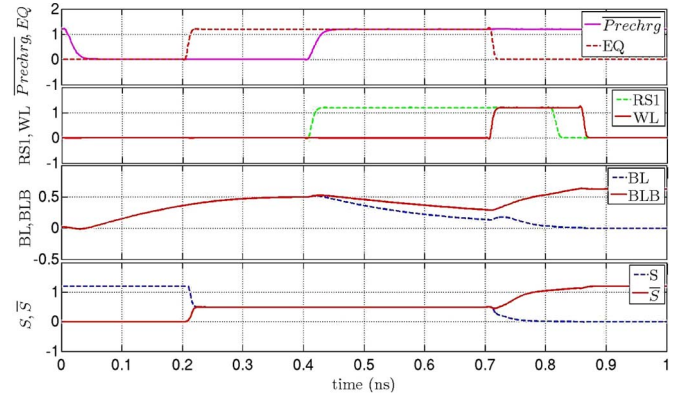


Fig. 10. Restore operation.

WL is pulled up to set the storage nodes ( $S$  and  $\bar{S}$ ) to their data accordingly. Then both BL and BLB are precharged to  $V_{DD}/2$ , and the read operation is performed by pulling up the WL again. The output of the sense amplifier is set according to the written data. Note that an array of  $64 \times 64$  cells is assumed, and parasitic capacitances are taken into account by adding the other 63 cells of the row/column to WL/BL.

Fig. 9 illustrates a *store* operation followed by a *restore* operation. During store, the stored data at  $S$  and  $\bar{S}$  are written into MTJ and MTJb, respectively. With a store current of  $80 \mu\text{A}$ , the magnetization vectors of MTJ pairs are switched from parallel ( $m_z = 1$ ) to antiparallel ( $m_z = -1$ ), and vice versa, when ST|RS is activated and current passes through the MTJs. The zoomed-in version of the restore operation is shown next.

Signals for *restore* are shown in Fig. 10. The store/restore time of  $80 \text{ ns}/1 \text{ ns}$  per row is more than  $10\times$  reduction in configuration time compared to a volatile FPGA with an external NV memory.

To test the proposed cell's robustness in restore operation under mismatch, we run Monte Carlo simulations. Effects of transistor pair mismatch on failure rate versus TMR for different  $\Delta V$  development times (the time from when VDD is raised to when the voltage difference between BL and BLB is developed) are studied in 1000 Monte Carlo runs for each point. In each run, the nMOS and pMOS thresholds are randomly chosen from a Gaussian distribution with a nominal threshold of  $0.355 \text{ V}$  for

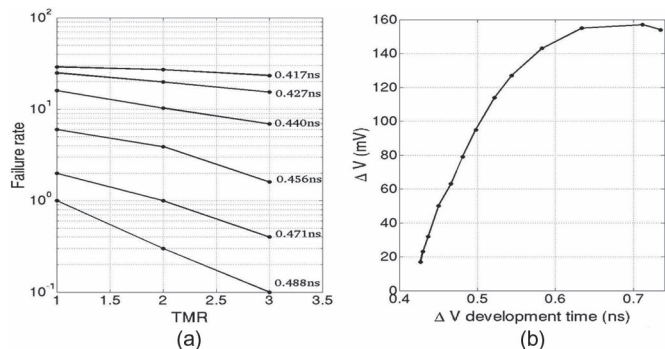


Fig. 11. Monte Carlo simulation results. (a) Failure rate (log scale) versus TMR for different  $\Delta V$  development times. (b)  $\Delta V$  versus its development time.

TABLE I  
COMPARISON OF NV-SRAM CELLS

Cells	Relative Size	Dynamic Reconfig.	Shadow Imp.	Store /change	Mismatch Tolerant
SRAM	1x	Yes	Yes	N/A	Yes
[10]	2.5x*	No	No	Yes	No
[11]	2.5x*	No	Yes	No	No
[12]	0.8-1.2x*	No	No	Yes	Yes
[13]	1.2x*	No	Yes	Yes	No
This work Store1,Store2	2.3,1.5x <sup>†</sup>	Yes, Yes	Yes, Yes	No, Yes	Yes, Yes

\*taken from [13]

<sup>†</sup>based on actual layout in 65nm CMOS

nMOS and  $-0.365$  V for pMOS and a sigma of  $0.02$  V for both cases. As shown in Fig. 11(a), the failure rate decreases as the TMR increases corresponding to smaller MTJ variation. The failure rate also decreases when we allow longer development times for  $\Delta V$ . This is because the larger development time results in a larger  $\Delta V$ , as shown in Fig. 11(b) for TMR = 3, which, in turn, overcomes the larger threshold mismatch between transistors. To increase the statistical confidence, we extract  $\sigma_{\Delta V}$  by curve fitting the results of failure rate versus  $\Delta V$  to a Gaussian distribution function. This results in  $\sigma = 24$  mV, which, in combination with a  $\Delta V = 156$  mV (obtained for TMR = 3), yields a confidence level of  $6.5\sigma$ . Therefore, with high enough TMR and/or long enough restore time, the failure rate can be reduced substantially.

### V. COMPARISON WITH PREVIOUS WORK

Table I compares the proposed cells against the previous NV-SRAM cells. The relative cell sizes for the previous works are taken from [13] and are defined with respect to an SRAM cell. While the proposed cell has comparable size to previous cells, it offers important features such as run-time dynamic reconfigurability of the MTJs, mismatch tolerance, and shadow implementation. In addition, the proposed cell does not require store operation after each data change (in Store1).

The proposed cell offers a restore time of 1 ns for restoring every 64 configuration bits. This should be compared against 50 ns for programming the same number of bits via external Flash, such as in Xilinx’s Virtex-7 product (see application note: xapp587). In addition, while the proposed cell requires only  $80 \mu\text{A}$  for store current, the corresponding store current for previous NV-FPGAs [4]–[6] (with similar architecture as [10]) is on the order of 1 mA.

### VI. CONCLUSION

An MRAM-backed SRAM cell has been proposed to replace the SRAM cells of FPGA to produce an NV-FPGA. Data can be stored in MTJs before power-down and restored after power-up. Due to nonvolatility and fast power-ups, selective power-down of cells can be utilized to eliminate the leakage current of SRAM cells during standby. With comparable cell size as SRAM, the proposed cell with an alternative store operation enables dynamic reconfiguration of the MTJs without interrupting operation of the system. With the same read/write operation as the SRAM cell, the proposed cell takes less than 80 ns for a store operation and less than 1 ns for a restore operation.

### ACKNOWLEDGMENT

The authors would like to thank CMC Microsystems for providing computer-aided design tools.

### REFERENCES

- [1] P. Chow *et al.*, “The design of a SRAM-based field-programmable gate array—Part II: Circuit design and layout,” *IEEE Trans. VLSI Syst.*, vol. 7, no. 3, pp. 321–330, Sep. 1999.
- [2] O. Goncalves, G. Prenat, G. Di Pendina, and B. Dieny, “Non-volatile FPGAs based on spintronic devices,” in *Proc. 50th ACM/EDAC/IEEE DAC*, May 2013, pp. 1–3.
- [3] K. JoonHan *et al.*, “A novel Flash-based FPGA technology with deep trench isolation,” in *Proc. 22nd IEEE NV Semicond. Memory Workshop*, Aug. 2007, pp. 32–33.
- [4] N. Bruchon, L. Torres, G. Sassatelli, and G. Cambon, “New non-volatile FPGA concept using magnetic tunneling junction,” in *Proc. IEEE Comput. Soc. Annu. Symp. Emerging VLSI Technol. Archit.*, Mar. 2006, p. 6.
- [5] W. Zhao, E. Belhaire, B. Dieny, G. Prenat, and C. Chappert, “TAS-MRAM based non-volatile FPGA logic circuit,” in *Proc. ICFPT*, Dec. 2007, pp. 153–160.
- [6] Y. Guilleminet, L. Torres, G. Sassatelli, N. Bruchon, and I. Hassoune, “A non-volatile run-time FPGA using thermally assisted switching MRAMS,” in *Proc. Int. Conf. FPL Appl.*, Sep. 2008, pp. 421–426.
- [7] D. D. Tang and Y. Lee, *Magnetic memory: Fundamentals and Technology*. Cambridge, U.K.: Cambridge Univ. Press, 2010.
- [8] W. Zhao, E. Belhaire, and C. Chappert, “Spin transfer torque (STT)-MRAM-based runtime reconfiguration FPGA circuit,” *ACM Trans. Embed. Comput. Syst.*, vol. 9, no. 2, pp. 14:1–14:16, Oct. 2009.
- [9] S. Paul, S. Mukhopadhyay, and S. Bhunia, “A circuit and architecture codesign approach for a hybrid CMOS–STTRAM nonvolatile FPGA,” *IEEE Trans. Nanotechnol.*, vol. 10, no. 3, pp. 385–394, May 2011.
- [10] W. Zhao *et al.*, “Integration of spin-RAM technology in FPGA circuits,” in *Proc. 8th ICSICT*, Oct. 2006, pp. 799–802.
- [11] Y. Shuto, S. Yamamoto, and S. Sugahara, “Nonvolatile static random access memory based on spin-transistor architecture,” *J. Appl. Phys.*, vol. 105, no. 7, pp. 07C933–07C933-3, Apr. 2009.
- [12] T. Ohsawa *et al.*, “A 1 Mb nonvolatile embedded memory using 4T2MTJ cell with 32 b fine-grained power gating scheme,” *IEEE J. Solid-State Circuits*, vol. 48, no. 6, pp. 1511–1520, Jun. 2013.
- [13] S. Fujita *et al.*, “Novel nonvolatile L1/L2/L3 cache memory hierarchy using nonvolatile-SRAM with voltage-induced magnetization switching and ultra low-write-energy MTJ,” *IEEE Trans. Magn.*, vol. 49, no. 7, pp. 4456–4459, Jul. 2013.
- [14] W. Zhang, N. K. Jha, and L. Shang, “Low-power 3-D nano/CMOS hybrid dynamically reconfigurable architecture,” *J. Emerg. Technol. Comput. Syst.*, vol. 6, no. 3, pp. 1–32, Aug. 2010.
- [15] S. J. Lovett, G. A. Gibbs, and A. Pancholy, “Yield and matching implications for static RAM memory array sense-amplifier design,” *IEEE J. Solid-State Circuits*, vol. 35, no. 8, pp. 1200–1204, Aug. 2000.
- [16] A. Vatankhahghadim, S. Huda, and A. Sheikholeslami, “A survey on circuit modeling of spin-transfer-torque magnetic tunnel junctions,” *IEEE Trans. Circuits Syst. I: Reg. Papers*, vol. 61, no. 9, pp. 2634–2643, Sep. 2014.