Hybrid CMOS-STTRAM Non-Volatile FPGA: Design Challenges and Optimization Approaches

Somnath Paul
Department of EECS
Case Western Reserve University
Cleveland,OH
Email: sxp190@case.edu

Saibal Mukhopadhyay
Department of ECE
Georgia Institute of Technology
Atlanta, Georgia
Email: saibal@ece.gatech.edu

Swarup Bhunia
Department of EECS
Case Western Reserve University
Cleveland, OH
Email: skb21@case.edu

***Abstract*—Research efforts to develop a novel memory technology that combines the desired traits of non-volatility, high endurance, high speed and low power have resulted in the emergence of Spin Torque Transfer-RAM (STTRAM) as a promising next generation universal memory. However, the prospect of developing a non-volatile FPGA framework with STTRAM exploiting its high integration density remains largely unexplored. In this paper, we propose a novel CMOS-STTRAM hybrid FPGA framework; identify the key design challenges; and propose optimization techniques at circuit, architecture and application mapping levels. Simulation results show that a STTRAM based optimized FPGA framework achieves an average improvement of *48.38%* in area, *22.28%* in delay and *16.1%* in dynamic power for ISCAS benchmark circuits over a conventional CMOS based FPGA design.**

***Index Terms* - Emerging memory technologies, STTRAM, non-volatile FPGA.**

I. Introduction

Most of the research concerning emerging devices are directed towards the investigation of novel memory technologies that combine the best features of current volatile and non-volatile memories in a fabrication technology compatible with CMOS process flow. These efforts have culminated in the emergence of several alternatives such as FeRAM, PCRAM and STTRAM. The high speed of operation along with increased endurance and higher integration density makes STTRAM(Spin Torque Transfer Random Access Memory) one of the front-runner for the next generation universal memory [1]. Recent advances [2] in fabrication technology and novel memory architectures [3] have been reported for STTRAM arrays. One of the major challenge in integrating STTRAM as an embedded memory is its high write current [3]. A high write current requirement is however of lesser concern when the memory is reconfigured only infrequently. This makes STTRAM particularly suitable as a non-volatile storage in Field Programmable Gate Array (FPGA) frameworks which are configured only in a while.

Till date conventional SRAM has been the primary choice for storage in the CLBs (Configurable Logic Block) as well as for the configuration bits of the reconfigurable interconnects [4]. However, due to the volatile nature of the SRAM, the configuration is lost when the power is turned down. However, in an ever increasing FPGA market, the demand for non-volatile FPGAs is also on the rise. An existing solution is to use flash memory as a non volatile storage to hold the configuration even when the power is turned down. However, integration of flash technology all over the device raises technological constraints and increases the number of masks [5]. STTRAM is an fast-emerging non-volatile alternative that outperforms existing flash technologies due to its higher endurance, low access latencies and compatibility with existing CMOS fabrication flow. It is therefore worthwhile to investigate the circuit and architectural optimizations for a CMOS STTRAM hybrid FPGA framework.

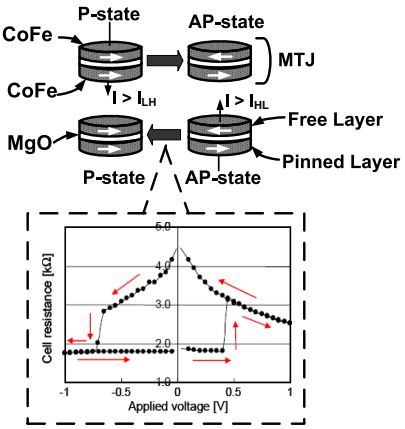


Fig. 1. **Change in magnetization states for a MTJ on application of current. Callout shows a representative R-V curve along with the dependence of the TMR ratio on the bias voltage [1].**

The basic building block of a STTRAM cell is the Magnetic Tunneling Junction (MTJ) (Fig. 1). Each MTJ consists of two ferromagnetic layers (typically CoFe) separated by a very thin tunneling dielectric film (typically crystallized MgO). Magnetization in one of the layers (referred as pinned layer) is fixed in one direction by coupling to an anti-ferromagnetic layer (such as PtMn) [1]. The other ferromagnetic layer (referred as free layer) is used for information storage. The direction of magnetization of free layer with respect to the pinned layer (i.e., anti-parallel or parallel) can be controlled by the injection of spin-polarized electrons. Hence the MTJ can be switched between two stable magnetic states with high (R_{AP} or R_H) or low (R_P or R_L) resistances and it retains the state without any applied power. The information stored in an MTJ is thus determined by sensing whether the resistance state is high or low. One of the quality metrics for a MTJ device is therefore the Tunneling Magneto-Resistance (referred as TMR) ratio [6], defined as (R_H-R_L)/R_L. It may be noted that recent advances in STTRAM technology has been able to achieve a TMR ratio as high as 470% [3]. As observed from Fig. 1, the write current for an MTJ cell is required to be larger than the Switching Threshold Current (I_{HL} or I_{LH}) in order to switch the magnetization of the *free layer* from anti-parallel to parallel spin or vice versa.

The fact that STTRAM stores the high or the low state in the form of resistances makes it suitable for large embedded arrays. However its use as a storage device with conventional sensing circuitry incurs significant overhead for the case of small 1D Look Up Tables (LUT)

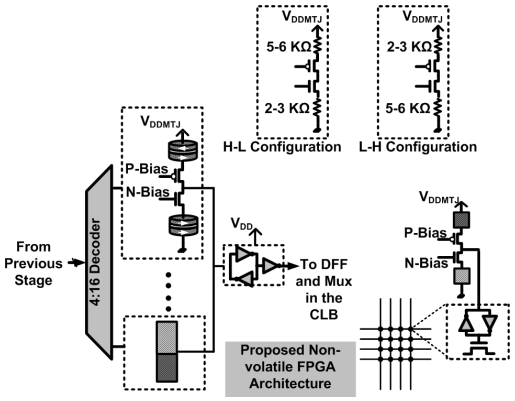


Fig. 2. **Proposed scheme for integration of STTRAM in the CLB and reconfigurable interconnects of a CMOS-STTRAM hybrid FPGA architecture**

present in FPGAs. A smarter scheme for sensing the resistance of the MTJs in case of FPGAs has been proposed in [5,7]. The principle idea is to use two MTJ elements for storing one-bit information and then employing a latch for reading out the stored configuration. This dynamic sensing scheme requires the latch to be pre-charged before evaluation. However, in a spatial computing framework such as a FPGA, the time for evaluation of a particular CLB is not fixed and depends on the path delay. Rather a static sensing approach as presented in [8] would be beneficial for the STTRAM based LUT implementation. However, the approach as presented in [8] does not provide rail to rail swings of the output voltage. This results in a static current dissipation in the following logic stages. In context of the above challenges, we have explored circuit/architectural optimizations and developed appropriate design mapping techniques for CMOS-STTRAM hybrid FPGA. In particular, the paper makes the following contributions:

1) We have presented a CMOS-STTRAM hybrid *non-volatile* FPGA framework that leverages on the high integration density of STTRAM process and preserves the spatial computing model of conventional FPGAs.
2) We have suggested a *preferential storage* based application mapping technique for reduction of dynamic and static power consumption.
3) We have also investigated an architecture for Shannon decomposition based dynamic supply gating for static power reduction in the proposed framework.

II. Background

A. Models for a CMOS-STTRAM hybrid framework

We have considered an integration of the STTRAM device with 130nm TSMC CMOS process. Conventionally the STTRAM device is modeled as high or low resistances to denote the two magnetization states. As demonstrated in [8, 9], the high and low resistance states may be assumed to have values in the range of $5-6K\Omega$ and $2-3K\Omega$ respectively. This corresponds to an achievable TMR ratio of 67%-200%. The read-operating voltage (V_{ddMTJ}) for the STTRAM is limited to 0.7V to prevent any undesired writes during read operation. With this model, we now seek to find an effective architecture for a CMOS-STTRAM hybrid FPGA.

B. Existing architectures for non-volatile FPGA

In a STTRAM cell, depending on whether the low or the high resistance is in contact with the V_{dd}, the output logic level would be either high or low respectively. However, even for a TMR of 200%, the maximum and minimum voltage levels attained at the output of the resistor divider for a supply of 0.7V are 0.525V and 0.175V respectively. A suitable modification suggested in [8] to increase the noise margin is by series insertion of P and N transistors, biased with 0 and V_{dd} respectively. Although, one may use STTRAM as a back up storage device for each CMOS latch, it fails to exploit the high integration density of the STTRAM process. Due to the smaller footprint of the MTJ device ($25F^2$) compared to the SRAM ($140F^2$, F being the minimum feature size), an architecture that reduces the contribution of the CMOS will therefore be able to save valuable die area.

TABLE I
Design overhead for CLB

Design Overhead	Stratix	Proposed Arch.	
		H-L	L-H
Area (μm^2)	556.23	287.07	287.07
Read Delay (ns)	0.193	0.151	0.149
Dynamic Power (μW)	51.93	37.14	69.24
Static Power (μW)	3.21	8.71	44.68

III. Circuit/Architectural optimizations for CMOS-STTRAM hybrid FPGA

A. Circuit optimizations for a static voltage sensing based logic evaluation architecture

The proposed architecture is illustrated in Fig. 2. The multiplexor inside the CLB of a conventional design is replaced with a decoder that applies proper P and N-bias values to only one MTJ leg depending on the inputs driving a given CLB. The output of the leg that is turned ON therefore determines the latch output. Hereafter we will denote the two resistor divider configuration as $H-L$ and $L-H$ depending on whether the higher or lower MTJ resistance is connected to V_{dd}. The $H-L$ configuration leads to an acceptable logic zero at the output of the selected MTJ leg ($0.014V$) and hence at the input of the latch. The configuration produces only a small static current through the MTJ leg which is required to be 'ON' over the entire clock period. Note that this static current ($\sim 8.7\mu A$) is much less compared to the minimum MTJ write current ($\sim 100\mu A$) reported to date [3]. The resistor divider however does not provide a rail to rail switching and for a $L-H$ configuration, the high output logic level is at a voltage value 0.7V. This is primarily due to the moderate TMR ratio (200%) used in our models. It is due to this degraded high logic level that a static 'ON' current flows through the CMOS latch over the entire clock period. The proposed architecture provides an 'Instant ON' non-volatile FPGA framework with higher integration density at the cost of increased power overhead. Table I provides a comparison of the design overheads for the CLBs in the present and the proposed architecture.

From Table I, we note that the proposed architecture improves the area and delay requirements for a single CLB by *48.38%* and *22.28%* respectively. However the dynamic power requirement for the $H-L$ and $L-H$ configurations of the proposed framework are smaller and greater than the conventional design by *28.48%* and *33.33%* respectively. In the proposed framework a static current flows through the MTJ leg that is turned ON. Moreover a degraded logic '1' output from the MTJ leg leads to an increased static power dissipation in the following latch. Following few standard circuit optimization techniques were employed to improve the logic '1'.

- Upsize the pmos transistor at the output.

- Since the series nmos cannot be downsized due to constraints for delivering the write current, the output logic level was further skewed towards '1' using a low V_t pmos device and a high V_t nmos device.

The above optimizations in the resistor divider circuit improves the logic '1' output level from $0.68V$ to $0.75V$.

B. Architecture for Shannon decomposition based dynamic supply gating

Supply gating has emerged as an extremely effective technique for reduction of dynamic and active leakage power in standard cell designs [11]. The savings in dynamic and leakage power becomes more substantial when the supply gating is dynamically applied to finer granularity of logic. To achieve this objective, a hypergraph partitioning of the original circuit followed by Shannon decomposition based synthesis [12] of the partitioned netlists has been proposed in [13]. For standard cell design, such an approach has proven to be extremely effective for dynamic and leakage power reduction. However, the same has not been extended to the realm of designs mapped to FPGA. The primary reason is that if the supply is gated off, the configuration stored in the SRAM present in the LUTs is lost. But, the non-volatile nature of CMOS-STTRAM hybrid FPGA presents us an opportunity to extend supply gating for power reduction in FPGA circuits. Here we present an architecture for Shannon decomposition based dynamic supply gating for reducing static power components in the proposed framework.

1) Hardware support for Shannon decomposition: The non-volatile nature of the proposed hybrid CMOS-STTRAM FPGA framework allows a dynamic supply gating of the decoder, the MTJ legs and the output latch without destroying the configuration inside the CLB. Fig. 3 shows the hardware support for the proposed supply gating scheme. Since the MTJ leg consumes a static current over the entire clock period, a fine-grained dynamic supply gating brings significant improvement to the power requirements for the mapped design.

2) Improvement in power consumption with dynamic supply gating: The steps for mapping a design to a framework with support for dynamic supply gating were adopted from [13]. A 2-level Shannon decomposition of the original circuit resulted in a substantial (*75.6%*) power improvement for some of the benchmarks. Although the Shannon decomposition resulted in significant savings of static power for all the benchmarks, it led to an increase in the total number of resources required for mapping all the co-factors. We have therefore only reported those benchmarks for which the area impact is minimal.

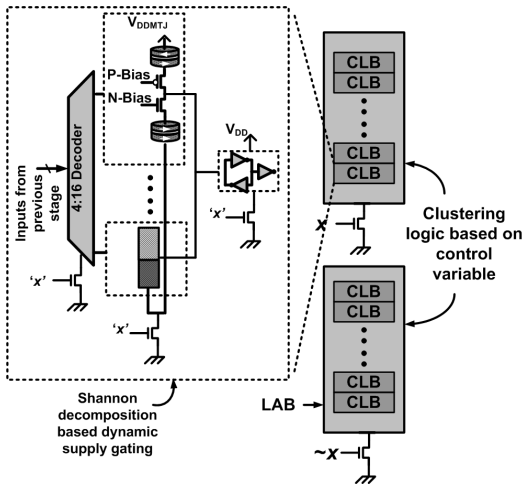


Fig. 3. **Hardware architecture for Shannon decomposition based dynamic supply gating in CMOS-STTRAM hybrid FPGA**

IV. Application mapping using preferential storage

The observation that the power requirement for the CLB operation in the original design is intermediate to that required for the $H-L$ and the $L-H$ configurations leads to the concept of *preferential storage*. For reducing the power consumption of the proposed architecture, it is desirable that the LUT inside the CLB contains more logic zeros rather than logic one values. In order to have a first hand estimate of the power saving through preferential storage, we have developed a simple software routine that optimizes the circuit after it has been technology mapped using standard algorithms such as Flowmap or Cutmap [10]. Table II summarizes the savings in average dynamic and static power compared to the un-optimized design. As seen from Table II, the proposed routine achieves on an average a *9.78%* improvement in average dynamic power per CLB and a *21.95%* improvement in average static power per CLB for a *103.23%* increase in the number of CLB outputs with '0' polarity.

V. Simulation Results

We have validated the effectiveness of the proposed design approaches at circuit/architecture/application level for the CMOS-STTRAM hybrid FPGA framework. Results on the delay, power and resource utilization for the benchmark circuits were obtained by mapping them to a Stratix FPGA using Altera-Quartus v7.0 software.

A. Improvement in Area

Fig. 4(a) shows the total logic area as required for mapping the benchmarks in the two cases: i) CMOS FPGA and ii) CMOS-STTRAM hybrid FPGA. From these results we note that the proposed scheme leads to an average area savings of *48.39%* compared to the conventional platform.

B. Improvement in Delay

Fig. 4(b) shows the logic delay of the chosen benchmark set for four different scenarios: i) Normal CMOS FPGA, ii) low power CMOS FPGA employing high-Vt transistors for the latches, iii) proposed CMOS-STTRAM hybrid FPGA and iv) proposed FPGA design with supply gating transistors. As seen from Fig. 4(b), compared to the conventional CMOS FPGA design, the proposed architecture achieves an average logic delay improvement of *22.28%*.

C. Improvement in Power

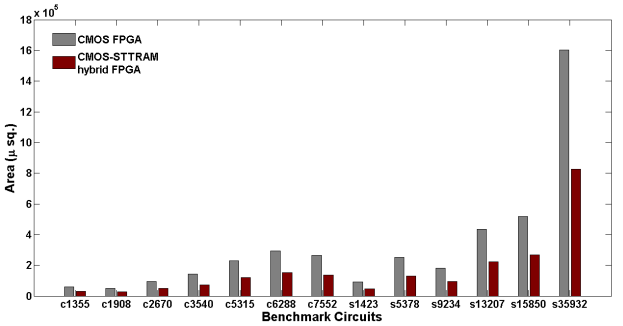
Fig. 4(c) compares the power requirement of the optimized and the un-optimized designs against the conventional CMOS based FPGA framework. From Fig. 4(c), we note that on an average the optimized STTRAM hybrid framework consumes *16.1%* less power than the CMOS design.

VI. Conclusion

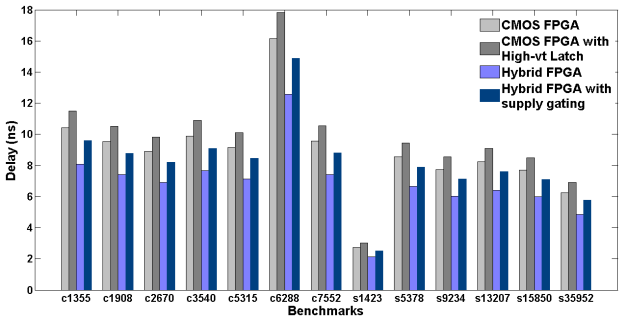
In this paper, we have presented a novel non-volatile CMOS-STTRAM hybrid FPGA architecture; identified key design challenges; and proposed optimization techniques at multiple levels of design abstraction. The proposed architecture leverages on the high integration density of emerging STTRAM devices and minimizes the total logic area by reducing the contribution of CMOS latches. The proposed architecture in its un-optimized form requires higher power than its CMOS counterpart. However an efficient resistor-divider design coupled with application mapping methodology based on

TABLE II
SAVINGS IN DYNAMIC AND STATIC POWER ACHIEVED THOUGH PREFERENTIAL STORAGE IN CMOS-STTRAM HYBRID LUTS

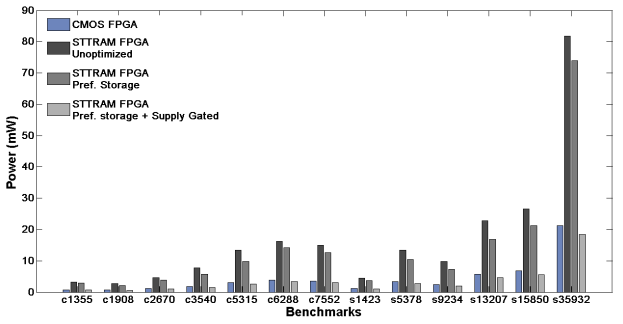
ISCAS Benchmarks	Ratio of 0's and 1's as CLB output		% Change in CLB output polarity	Avg. Dynamic power per CLB (μ W)		% Change in avg. dynamic power per CLB	Avg. Static power per CLB (μ W)		% Change in avg. static power per CLB
	Pre-opt.	Post-opt.		Pre-opt.	Post-opt.		Pre-opt.	Post-opt.	
c1355	1.03	1.39	34.52	52.94	50.58	4.45	26.41	23.77	10.00
c1908	0.97	2.16	122.95	53.43	47.28	11.50	26.96	20.08	25.53
c2670	1.28	2.39	86.86	51.22	46.61	9.01	24.49	19.32	21.12
c3540	0.93	2.37	153.66	53.75	46.67	13.16	27.32	19.39	29.01
c5315	0.77	2.08	169.03	55.23	47.55	13.91	28.98	20.38	29.69
c6288	0.92	1.42	54.61	53.88	50.41	6.44	27.47	23.58	14.15
c7552	0.86	1.52	75.75	54.38	49.90	8.23	28.03	23.01	17.90
s1423	1.39	2.46	76.55	50.56	46.42	8.18	23.75	19.11	19.51
s5378	1.06	2.36	123.34	52.74	46.68	11.48	26.19	19.40	25.90
s9234	0.97	2.43	151.80	53.47	46.50	13.05	27.01	19.19	28.94
s13207	1.09	2.78	155.31	52.52	45.64	13.10	25.95	18.24	29.71
s15850	1.16	2.34	101.12	51.97	46.75	10.05	25.33	19.47	23.12
s35932	1.20	1.64	36.54	51.70	49.28	4.69	25.03	22.31	10.85



(a) **Area**



(b) **Delay**



(c) **Power**

Fig. 4. **Comparison of design overheads between CMOS-FPGA and the CMOS-STTRAM hybrid FPGA using the proposed optimization approaches**

preferential storage helps in reducing the overall power consumption. Finally Shannon decomposition based power gating was applied to significantly reduce the power dissipation of the proposed non-volatile platform. Simulation results for a set of standard benchmark circuits show that the proposed non-volatile FPGA provides a promising reconfigurable computing platform in future technology generation.

REFERENCES

[1] M. Hosomi et al, "A Novel Nonvolatile Memory with Spin Torque Transfer Magnetization Switching: Spin-RAM", *IEDM Tech. Dig.*, pp. 473-476, Dec., 2006.

[2] J. Hayakawa et al, "Current-Driven Magnetization Swicthing in CoFeB/MgO/CoFeB Magnetic Tunnel Junctions", *Japanese Journal of Applied Physics*, 44(41), pp. L1267-L1270, 2005.

[3] T. Kawahara et al, "2Mb Spin-Transfer Torque RAM (SPRAM) with Bit-by-Bit Bidirectional Current Write and Paralilelizing-Direction Current Read", *isscc*, 2007.

[4] P. Chow et al, "The Design of an SRAM based Field-Programmable Gate Array, Part II: Circuit Design and Layout", *IEEE Transactions on VLSI*, Vol. 7, pp. 191-197, 1999.

[5] N.Bruchon et al, "New non-volatile FPGA concept using Magnetic Tunneling Junction", *ISVLSI*, 2006.

[6] S. Tehrani et al, "Magnetoresistive Random Access Memory Using Magnetic Tunnel Junctions", *Proceeding of The IEEE*, Vol. 91, No. 5, May 2003.

[7] W. Zhao et al, "Integration of Spin-RAM technology in FPGA circuits", *ICSICT*, 2006.

[8] W. Xu et al,"Spin-Transfer Torque Magnetoresistive Content Addressable Memory (CAM) Cell Structure Design with Enhanced Search Noise Margin",*ISCAS*, 2008.

[9] Z. Diao et al, "Spin-transfer torque switching in magnetic tunnel junction and spin-tansfer torque random access memory", *Journal of Physics: Condensed Matter*, vol. 19, 165209, April 2007.

[10] J. Cong and Y. Ding,"FlowMap: An Optimal Technology Mapping Algorithm for Delay Optimization in Lookup-Table based FPGA Designs," *IEEE Trans. on CAD*, Vol. 13, No. 1, pp. 1-12, Jan. 1994.

[11] J.W. Tschanz et al, "Dynamic Sleep Transistor and Body Bias for Active Leakage Power Control of Microprocessors", *JSSCC*, 2003.

[12] S. Bhunia et al,"A Novel Synthesis Approach for Active Leakage Power Reduction Using Dynamic Supply Gating", *DAC*, 2005.

[13] L. Leinweber and S. Bhunia, "Fine-Grained Supply Gating Through Hypergraph Partitioning and Shannon Decomposition for Active Power Reduction", *DATE*, 2008.