Device and Circuit Optimizations for Spin Transfer Torque Reconfigurable Logic

Darya Almasi and Hamid Mahmoodi
School of Engineering
San Francisco State University
San Francisco, CA, USA
daryaa@mail.sfsu.edu,
mahmoodi@sfsu.edu

Houman Homayoun
Electrical and Computer Engineering
George Mason University
Fairfax, VA, USA
hhomayou@gmu.edu

Hassan Salmani
Electrical and Commuter Engineering
Howard University
Washington, DC, USA
hassan.salmani@howard.edu

***Abstract*— Spin Transfer Torque (STT) is a promising technology for storage in which the information is stored in the form of magnetic orientation of a Magnetic Tunnel Junction (MTJ) rather than electric charge. Besides memory applications, this technology is promising for non-volatile reconfigurable logic design. The major challenge in realizing this technology is the power and performance overhead associated with reading the state of MTJs (high and low resistance states) and converting it into high and low voltages for interface to next stage circuits. In this paper, methods are proposed to reduce this overhead by MTJ resistance optimization at the device level and mapping multiple low fan-in logic gates into a STT-based Look Up Table (STT-LUT) at the circuit level. The paper demonstrates that by optimally mapping logic gates to STT-LUTs, the power and performance of a reconfigurable design can be improved or even exceed that of a full-custom design. Our results on an arithmetic benchmark circuit shows that by optimal logic gate mapping into STT-LUTs, the power and performance of the design is improved by 58% and 50%, respectively, compared to the design where each individual logic gates is replaced by an STT-LUT.**

Keywords—look up table; low power; magnetic tunnel junction, reconfigurable logic; spin transfer torque

I. INTRODUCTION

Spin Transfer Torque RAM (STTRAM) is an emerging storage technology. STTRAM utilizes a Magnetic Tunnel Junction (MTJ) device to store information (Fig. 1) [1]. MTJ is composed of two ferromagnetic layers, one with a fixed orientation and the other with a programmable orientation, isolated by a thin insulator and creating a two-state binary non-charge based and non-volatile information storage cell [2]. The parallel and antiparallel states of the MTJ magnetic layers result in low (R_P or RL) and high resistances (R_{AP} or RH) that need to be sensed with a sense amplifier circuit to produce full-swing logic 0 and 1 voltages, respectively. The percentage of difference between R_{AP} and R_P is referred to as Tunneling MagnetoResistance (TMR). The write operation requires passing a current, higher than the critical write current, through the MTJ in one direction or the other to write binary states. This technology is low cost due to its CMOS compatibility, and very scalable due to its non-charge based storage, and offers very low leakage due to its non-volatility [1,2]. However, the main challenge is its relatively high write current [3].

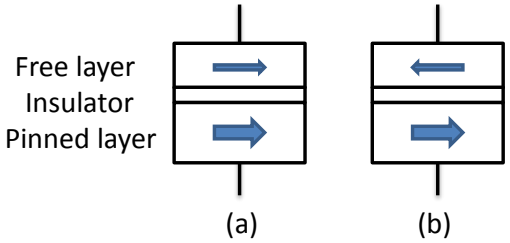


Fig. 1: Programmable MTJ: (a) Parallel (low resistance) and (b) anti-parallel (high resistance) states

This challenge makes this technology particularly suitable for applications where the read operation dominates and the write operation is not as frequent. An example of such an application is reconfigurable logic as in Field Programmable Gate Arrays (FPGA) or partially reconfigurable logic [4,5]. In such applications, an STT-LUT is used to implement logic functions.

There have been pervious works on logic design using STT-LUTs, demonstrating considerable dynamic power and performance overhead for such designs compared to the custom CMOS alternative, except for complex functions [6,7]. STT-LUTs have inherent advantage of low leakage and non-volatility that makes them desirable for wider applications, if the active mode overhead can be reduced.

It is known that LUT based logic implementations exhibit power and performance overhead compared to custom implementations [7]. This problem has been tackled in FPGA synthesis and LUT mapping algorithms [8]. The FPGA mapping optimizations deal with finding optimal scenarios for mapping multiple logic gates of a netlist into a single LUT, thereby reducing the number of required LUTs.

This paper first investigates optimization of TMR for improving performance of STT-LUTs. Then, mapping of multiple logic gates into a single LUT is explored as a way to reduce the reconfiguration overhead at the circuit level. The contributions of this paper are as follows:

This research is sponsored by the Defense Advanced Research Project Agency of the USA.

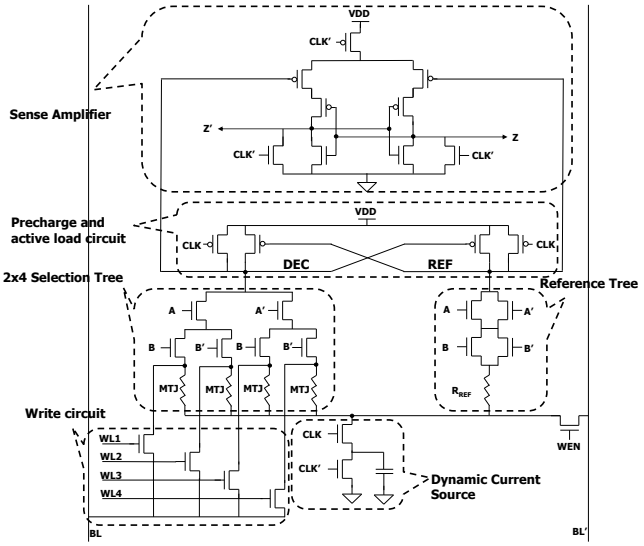


Fig. 2: 2-input STT-LUT

- The impact of TMR on power and performance of STT-LUTs is investigated and it is shown that higher TMR results in lower power and better performance for STT-LUTs
- Effectiveness of mapping multiple logic gates into a single STT-LUT for overhead reduction is explored and it is shown an optimal LUT mapping can reduce the power and performance overhead and/or lower the active power of STT-LUT based design below the custom CMOS design.

The remainder of this paper is organized as follows. Section II provides some background on the design of STT-LUTs. Section III presents the influents of TMR on performance of STT-LUTs. Comparison of STT-LUTs with custom CMOS gates is presented in section IV. Optimal mapping of logic gates into STT-LUTs for power and performance improvement is discussed in Section V. Finally conclusions are drawn in Section VI.

II. SPIN TRANFER TORQUE LOOK UP TABLE (STT-LUT)

Fig. 2 shows the schematic of a 2-input (4-bit) STT-LUT [4]. This is a dynamic circuit that operates in a precharge (CLK=0) and evaluate (CLK=1) fashion. For the read mode, the MTJ selection is performed via a pass-transistor decoder/mux (selection tree). To balance the transistor paths of the MTJs and the reference resistor (R_{REF}), similar transistors are inserted above the reference resistor. When CLK goes high, the current provided by the dynamic current source is divided between the selected MTJ and the reference resistor, resulting in a current differential that is drained from the nodes DEC and REF. This current differential is converted to a low swing voltage differential on the nodes DEC and REF by the two cross coupled PMOSes. This voltage differential is then amplified by a sense amplifier to produce full swing differential outputs (Z and Z'). In the write mode, the write differential voltage is applied to the bitlines (BL and BL') and the right MTJ is selected by the world-line signals (WLi) and the write enable (WEN) is activated.

For high read performance and enhanced noise margin, greater difference between the low and high resistances of the MTJ is desired. This resistance differential is quantified by the Tunnel Magneto Resistance (TMR), defined as:

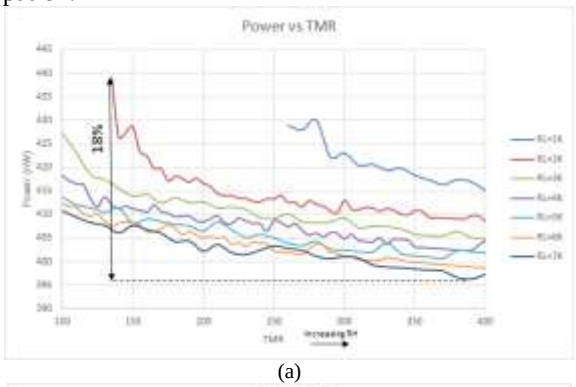
$$TMR = 100 \times (R_{AP} - R_P) / R_P \quad (1)$$

where R_P and R_{AP} are the resistances of the MTJ in the parallel and anti-parallel states, respectively. TMR is technology parameter dependent on the MTJ geometries and materials.

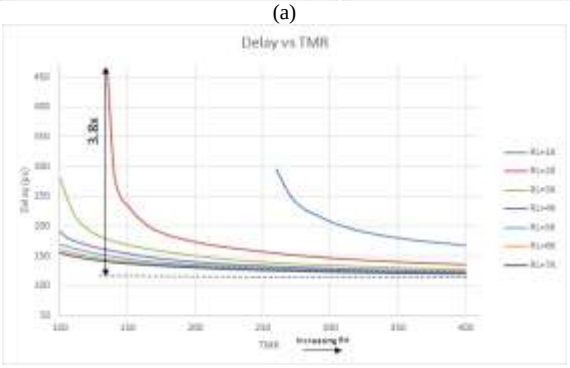
III. TMR OPTIMIZATION

Higher TMR is desirable for read performance. We have simulated the read power and performance of a 2-input STT-LUT in a predictive 16nm CMOS technology node [9].

As shown in Fig. 3, higher TMR results in reeducation in both read power and delay of the STT-LUT. Higher TMR results in large difference between the low and high resistances of the MTJ (Eq. (1)) and hence larger voltage differential produced at the input of the sense amplifier (Fig. 2) in the read mode resulting in reduced delay. Moreover, larger TMR increases the average resistance of the pull-down network (due to increase in RH (R_{AP}) and R_{REF} that is about the average of RH an RL (R_P)), and hence reduction in power dissipation.



(a)



(b)

Fig. 3: (a) Power and (b) Delay of STT-LUT vs TMR. RH and RL denote the high and low resistances (R_{AP} and R_P) of the MTJ.

Achieving very high TMR has manufacturing limitations in terms of material choice and physical limitations. A survey of published literature on manufactured MTJs show TMR values ranging from 100 to 600%. For the rest of our analysis, we use TMR of 400% which is in the range feasible by manufacturing.

The reference resistor (R_{REF}) can also be optimized for achieving best power and delay. A good starting point for R_{REF} would be average of RH and RL, but it could be optimized. The effective resistance of the two networks, reference and selection trees (Fig. 2), should be equal to achieve symmetric delay between the differential outputs. Moreover, the

capacitance of the dynamic current source (Fig. 2) can be jointly optimized with R_{REF} to minimize the power and delay product of the STT-LUT. Table I shows the characterization of STT-LUTs with fan-ins ranging from 2 to 8. With increase in fan-in the active power and delay generally increase, but the leakage power remains very low and almost insensitive to the fan-in, since the leakage is limited by the stacking effect, and saturated through the NMOS pulldown network gated by the clocked transistor in the dynamic current source (Fig. 2) [7].

Table I. Power and delay characterization of STT-LUTs (TMR=400% (RL=20KΩ and RH= 100KΩ), Clock frequency=500MHz, Vdd=0.7V, Temperature=100˚C, Process: 16nm CMOS). PDP=Power Delay Product

Fan -in	R_{REF} (KΩ)	C (fF)	Active Power(nW)	Delay (pS)	PDP (fJ)	Leakage Power(nW)
2	32.6	3	378.4	108.8	0.407	4.37
3	30.6	3	337.3	178.4	0.601	3.97
4	28.5	3	328.5	251.3	0.825	3.76
5	64.7	5	443.4	250.4	0.111	3.66
6	62.9	5	437.8	316.9	0.138	3.68
7	61	7	500.6	293.2	0.146	3.87
8	55	9	553.6	303.8	0.168	4.34

IV. COMARISION WITH CUSTOM CMOS

An n-input LUT can be programmed to implement any arbitrary logic function of n inputs. A custom circuit on the other hand is designed to deliver a fixed functionality, most likely in a more efficient manner for that specific function. Here, we compare the power and performance of STT-LUTs with custom logic circuits of same fan-in and designed in the static complementary CMOS logic style.

Fig. 4 shows comparison of active power between STT-LUTs and same fan-in custom CMOS gates. The power of a static CMOS gate depends on its output switching activity rate; whereas the active power of the STT-LUT is independent of the activity pattern as well as the function programmed into the LUT. That is due to the differential dynamic nature of the STT-LUT design that makes its outputs switch every cycle irrespective of its input pattern or functionality. It is observed that STT-LUTs show lower power for complex high fan-in gates with high activity factors.

Fig. 5 shows comparison of leakage power consumptions. The STT-LUTs show extremely low leakage due to the presence of significant transistor staking effect in their transistor network [7]. Fig. 6 shows the delay comparisons. STT-LUTs show considerable delay overhead as compared to the custom CMOS logic gates that could limit their application. In the next section, we propose a method to reduce the STT-LUT overhead by mapping multiple logic gates into a single LUT.

V. CIRCUIT OPTIMIZATION USING LUT MAPPING

Given that an n-input LUT can implement any arbitrary function of n inputs, mapping multiple logic gates into a single LUT is possible and results in reducing hardware complexity in LUT based designs. This concept has been used in logic synthesis and mapping algorithms for LUT-based FPGAs [8]. Here, we explore the effectiveness of this technique for reducing the overhead of STT-LUT based implementations.

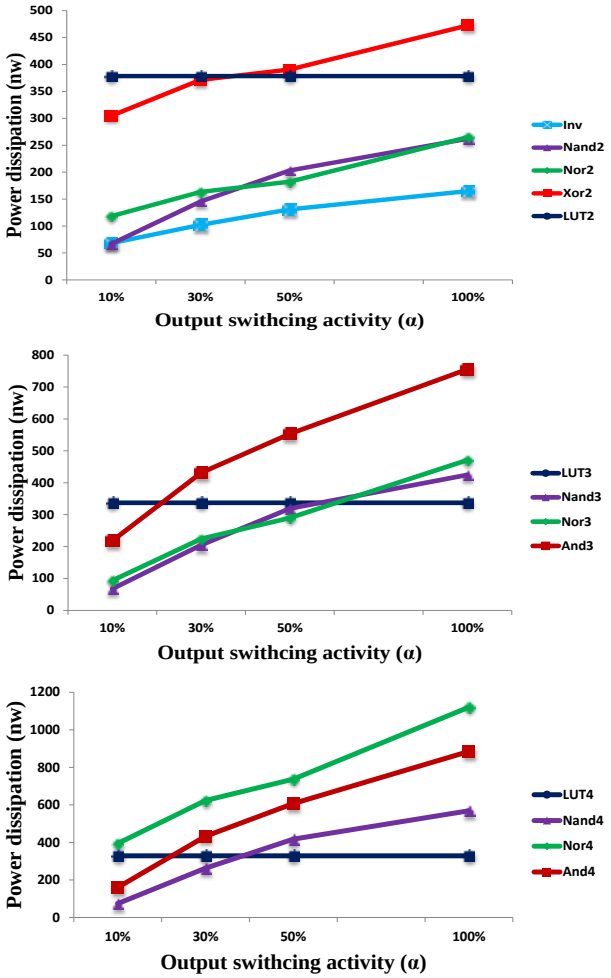


Fig. 4: Comparisons of active power of STT-LUTs and same fan-in custom logic gates implemented in static complementary CMOS logic style

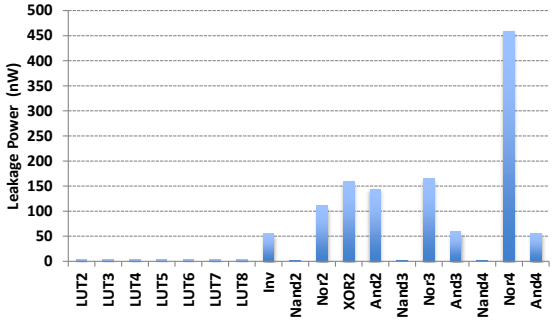


Fig. 5: Leakage Power of STT-LUTs and custom static CMOS gates

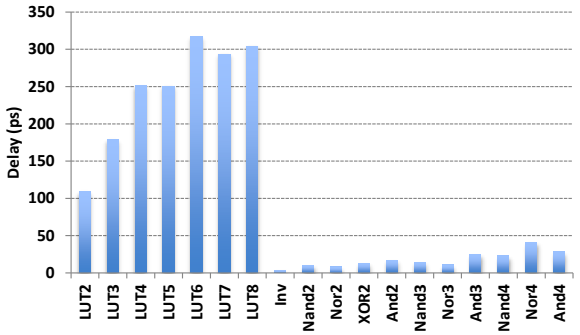


Fig. 6: Delay of STT-LUTs and custom static CMOS gates

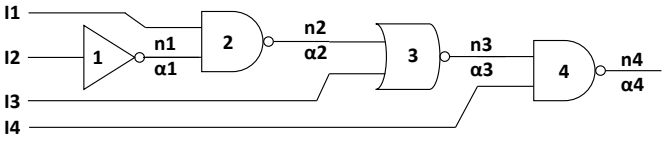


Fig. 7: A sample 4 input logic circuit

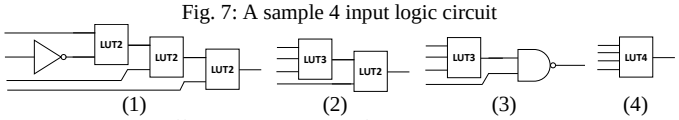


Fig. 8: Different LUT mapping for sample circuit in Fig. 7

Table II: Power and delay comparison of different LUT mappings in Fig. 8

Metric	LUT (1)	LUT (2)	LUT (3)	LUT (4)	FullCustom
Power (nW)	1368	794	435	461	471
Delay (pS)	340	307	203	239	21

Let's start by the simple example circuit shown in Fig. 7. Fig. 8 shows four different methods of mapping these logic gates to STT-LUTs of different fan-in. Approaches 1 and 3 show a partial mapping where some gates are left to be implemented in the custom form. The power and delay of these alternative implementations along with those of the custom CMOS design are shown in Table II. It is interestingly observed that by optimal STT-LUT mapping it is possible to achieve power consumption even less than that of the custom CMOS design. The third LUT mapping offers the least power consumption by an optimal use of STT-LUT and custom CMOS gates. LUT mapping also shows significant influence on delay. Although for such a simple design, the delays of STT-LUT based implementations still remain longer than the custom CMOS method, the third approach offers the least delay as well among the LUT mapping alternatives.

We expect the effectiveness of optimal LUT mapping to improve as the circuit grows in complexity. Let's consider the ISCAS 74283 benchmark circuit which is a fast 4-bit adder circuit composed of 36 logic gates (Fig. 9). The logic gates grouped by the dashed lines demonstrate possible mapping of multiple logic gates into a single STT-LUT. These groups must be single output as an LUT has a single output. In other words, no internal node within a group can be used outside the group.

Table III shows different approaches of full and partial STT-LUT mapping for the benchmark circuit and the resulting delay and power. The minimum power is achieved when LUT mapping is limited to multi-gate collapsing and the remaining individual logic gates are kept in custom CMOS (method #3). Notice that this power is even lower than the full-custom CMOS design (method#5). Method#3 shows 58% power reduction compared to method#1 and 42% compared to the full custom design. Among partial and full LUT mapped implementations, the minimum delay is shown by the method#4 that keeps the critical path logic gates in custom CMOS to avoid delay penalty. The delay of this method is 50% less than that of method#1. This minimum delay is still longer than that of the full custom method, because the critical path delay (87pS) is shorter than the minimum STT-LUT delay (108.8pS). Hence, by LUT insertion a new critical path is formed. As the length of the critical path grows in more complex circuits so that its delay exceeds the STT-LUT delay, this method would be able to avoid any delay penalty.

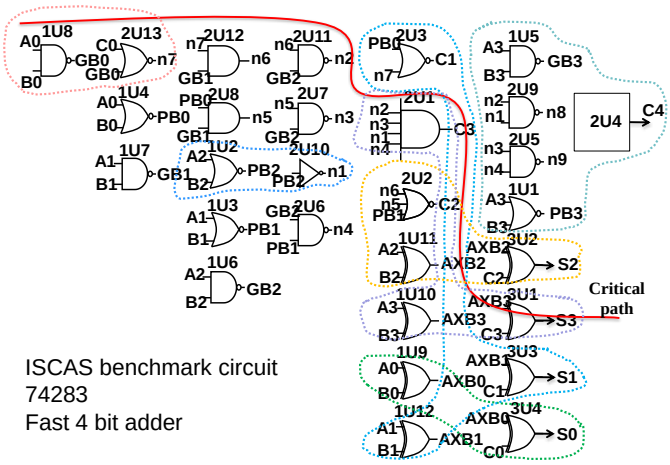


Fig. 9:Schematic of benchmark circuit highlighting critical path

Table III: Power and delay results of different STT-LUT mapping on 74283

LUT Mapping Method	Power (μW)	Delay (pS)
1. All individual gates except inverters replaced by same fan-in LUTs	10.6	795
2. All possible multi-gate LUT mapping and remaining gates replaced by same fan-in LUT	6.1	715
3. All possible multi-gate LUT mapping and remaining gates kept in custom CMOS	4.4	526
4. Same as 3, but keeping critical path gates in custom CMOS	5	396
5. Full custom CMOS (no LUT)	7.6	87

VI. CONCLUSION

STT-LUTs offer opportunities for low power logic design. However, the overhead associated with LUT reconfigurability needs to be mitigated. This paper has shown that the delay and power of STT-LUTs can be reduced via TMR enhancement at the device level and optimal multi-gate LUT mapping at the circuit level. Optimal multi-gate LUT mapping offers not only power reduction over custom CMOS, but also possibility of avoiding delay penalty in complex designs.

REFERENCES

[1] C. Augustine et. al., "Spin-Transfer Torque MRAMs for Low Power Memories: Perspective and Prospective," IEEE Sensors Journal, vol. 12, no. 4, pp. 756 – 766, 2012.

[2] N. Nishimura et. al."Magnetic tunnel junction device with perpendicular magnetization films for high-density magnetic random access memory," Journal of Applied Physics, vol. 91, no. 8, pp. 5246-5249, 2002.

[3] M. Rasquinha et. al., "An energy efficient cache design using Spin Torque Transfer (STT) RAM," International Symposium on Low-Power Electronics and Design, pp. 389-394, 2010.

[4] D. Suzuki et. al., "Fabrication of a nonvolatile lookup-table circuit chip using magneto/semiconductor-hybrid structure for an immediate-power-up field programmable gate array," Symposium on VLSI Circuits, pp. 80 –81, 2009.

[5] D. Suzuki et. al., "Fabrication of a 3000-6-input-LUTs embedded and block-level power-gated nonvolatile FPGA chip using p-MTJ-based logic-in-memory structure," Symposium on VLSI Circuits, pp. C172 – C173, 2015.

[6] X. Guo wt. al., "Resistive computation: avoiding the power wall with low-leakage, stt-mram based computing," International Symposium on Computer Architecture, pp. 371–382, 2010

[7] H. Mahmoodi, et. al. , "Resistive computation: a critique," IEEE Computer Architecture Letters, vol. 13, no. 2, pp. 89-92, 2014.

[8] J. Cong et. al., "Combinational logic synthesis for LUT based filed programmabel gate arrays," ACM Transactions on Design Automation of Electronic Systems, vol. 1, no. 2, pp. 145-204, April 1996.

[9] Predictive Technology Models. Online: http://ptm.asu.edu