Using STT-RAM Based Buffers in Digital Circuits

Mohsen Imani, Shruti Patil, Tajana Rosing
Department of Computer Science and Engineering, UC San Diego
La Jolla, CA 92093, USA
Email: {moimani, patil, tajana}@ucsd.edu

***Abstract*— STT-RAMs are good candidates to replace conventional SRAM cache and DRAM in main memory, but their applicability in digital logic circuits is unclear. Our experiments explore the power benefit of utilizing non-volatile buffers in digital circuits, with a case study of Ripple Carry Adder and Carry-Skip Adder circuits. We design a low-overhead 2T1MTJ buffer and place it in the intermediate non-critical paths that hold data for long times. Use of NVMs in these paths allows us to turn off parts of the adder to save power. Our simulations show 22.4% and 10.8% power saving in 512-bit RCA and SCA structures respectively with minimal area overheads.**

I. Introduction

Spin-Torque Transfer Random Access Memories (STT-RAMs) have been extensively researched for their potential to replace SRAM and DRAM main memories in digital systems. STT-RAMs are built from Magnetic Tunnel Junction (MTJ) devices, which have nearly zero leakage power and are ~4X denser than SRAM. However, they suffer from long write latencies, therefore their suitability in logic circuits has not been explored much[1-4]. Emerging embedded always-on circuits with ultra-low power budgets, or low-specification mobile devices, may afford to relax performance requirements. Towards understanding the power-performance trade-offs of NVMs in digital circuits, we first design a low-overhead non-volatile buffer and then investigate its use in 16-nm CMOS-based circuits, through evaluations on Ripple Carry Adder (RCA) and Carry-Skip Adder (CSA).

RCA and CSA adder circuits are designed to reduce power and area at the expense of performance. Their designs utilize several buffers along non-critical paths with low activity, where data is held until all inputs of a gate become ready. Buffers consume area and energy in combinational systems such as adders and multipliers since these units are always active. Fine-grained power gating or Dynamic Voltage Scaling (DVS) in non-critical paths can help, but these introduce overheads. For instance, power gating has wake-up time and energy overhead. Moreover, in small systems the sleep time is too short to offer power improvement. Instead, we utilize non-volatile buffers in the intermediate nodes of combinational circuits with low activity and high hold times. To manage the change in state variable (voltage to resistance and vice versa), previous designs for MTJ-based buffers have used CMOS-based sense amplifier and write circuitry. In minimizing area overheads, we design an MTJ-based buffer using 2T1MTJ circuit. Three control terminals are added to enable read and store, and they can be shared across devices to minimize area and complexity of the adder. Once data is saved into the non-volatile buffer, some part of the circuit can be turned OFF for greater power savings. Data is read from the buffer when required, without recovering previous blocks. Our simulation results with 16nm PTM models for CMOS show 22.4% and 10.8% power improvement for 512-bit RCA and SCA.

II. Adder Design with NVM

Figure 1 shows a 4-bit adder circuit. Due to inherent dependency of the final sum on intermediate carry results, the output is generated only when all carry-out signals have been considered in the computation. The adder structure is typically implemented in three stages using *generate (G)* and *propagate (P)* blocks. The first stage produces P and G outputs of all inputs in parallel. However, data dependencies in the second stage create a long delay until the last stage bits can be calculated with XOR blocks. This second stage is implemented differently for RCA and SCA adder structures, presenting different overall latency.

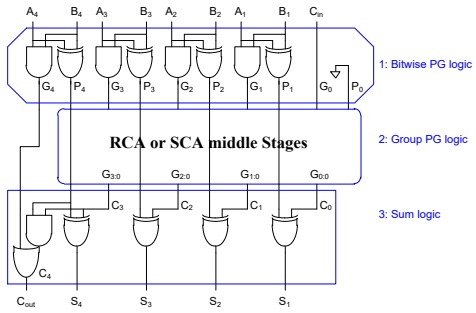

Fig. 1. 4-bit adder structure at gate-level

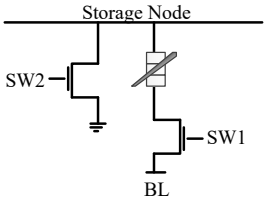

Fig. 2. MTJ based buffer for store and read operations

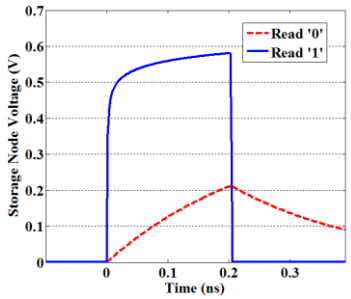

Fig. 3. MTJ buffer timing diagrams with 0.2ns pulse width

The total delay of RCA and SCA adders can be expressed as:

$$t_{RCA} = t_{PG} + (N-1)t_{AO} + t_{XOR} \quad (1)$$

$$t_{SCA} = t_{PG} + \sqrt{2N}t_{AO} + t_{XOR} \quad (2)$$

where T_{pg} is delay of first stage, t_{AO} is delay of each block in second stage and t_{xor} is delay of XOR gate. During propagation of carry bits through the circuit, the P/G outputs for most significant bits must be kept active till the low level P/G outputs become available. To achieve this in a power-efficient manner,

we add MTJ buffers at the output of the first stage. To save each output bit, we design a 2T1MTJ circuit shown in Figure 3. Three terminals SW1, SW2 and BL are suitably biased to enable store and read operations as follows:

STORE: For store operation, the transistor is activated by SW signal. The operation is done in two steps. First, BL is set to Vdd and in the second half stage it is set to zero. When the store value is zero, the output data is written in cell in the first phase (when BL is Vdd). On the other hand, when store value is one, the data gets written to the cell in the second phase.

READ: To read the state of the cell, the access transistor is activated and BL is set to Vdd. The ON transistor begins to charge the output node depending on stored resistance value. When reading one, MTJ is in low resistance state and BL voltage gets transferred to output node within the read cycle. When reading zero, the MTJ is in high resistance, so the output does not charge to logic-1 in time, and goes to high impedance state. Figure 3 shows the storage node voltage during read one and zero (0.2ns pulse width). Access transistor sizing can be tuned for the required charging levels. A second transistor is used to reset this charge to zero before another read or store operation is performed.

III. Evaluation and Results

Our simulations of CMOS blocks are performed with HSPICE 2011 using 16nm PTM [5]. MTJ circuit verification is performed using MTJ spice model [6]. The ON power and OFF current are calculated at Vdd=800mV and Vdd=200mV which represent the above and sleep regions for 16nm technology. To obtain latency and power consumption of MTJ for store and read operations, we use spice simulation. MTJ energy consumption in read and write operations is lower than the energy of a logical gate. The results indicate that MTJ only needs 1.02pJ/bit and 2.4pJ/bit respectively as a read and write energies. Similarly, read latency is about 0.18ns, while T_{Store} is close to 8.2ns (due to high write latency of MTJ).

From simulations of worst-case critical path delay of XOR, OR and AND gates, the maximum delay of each block in the second stage is T_{AO}= 200PS. For RCA and SCA the worst-case delay of second stage are respectively given by $(N-1)t_{AO}$ and $\sqrt{2N}t_{AO}$. As number of bits (N) increases, the hold time required from most significant bit buffers increases. We evaluate MTJs for *M* most significant blocks. For simpler control and minimal area overhead, we share the BL and SW lines among all MTJs. In case of large *M*, the MTJ power becomes dominant because low level bits do not present sufficient power savings. At low *M*, the saving is negligible since most of first level blocks are in ON mode. We experiment with varying M to determine the optimal number of lines with MTJ buffer. Figure 4 shows our results for a 512-bit RCA and SCA adders. Considering MTJ read and store latencies, MTJs are suitable for systems with N≥128. Adders for public key cryptography often require N≥512. In 512-bit RCA structures with MTJs in the last 300-bit blocks of first stage, we find 22.4% power saving. This improvement is about 31.8% when MTJs are used in the last 600 bits of 1024-bit adder.

Figure 5 shows the power advantages of using non-volatile buffers in RCA and SCA with different sizes. Small adders do not show any saving because of their short critical path, and high store energy and latency of MTJ. Note that in this graph each adder is optimized with the best value of *M*.

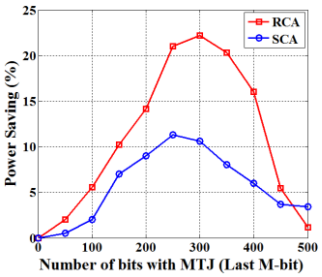

Fig. 4. Power saving for 512-point RCA with MTJs in last M-bits

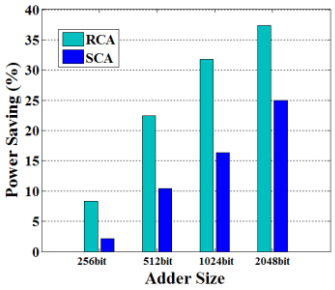

Fig. 5. Power saving in RCA and SCA with different size

IV. Conclusion

The low stand-by power of STT-RAM devices is particularly attractive for low power digital systems, however their application in current digital logic circuits has not been sufficiently explored. We design a non-volatile buffer and investigate its application in digital circuits with the example of Ripple Carry Adder and Carry-Skip Adder circuits. We use the buffers as intermediate nodes in paths with long hold times. This reduces leakage power along these paths. Due to the non-volatility of these devices, some parts of the adder circuit can be turned off to save additional power. Our simulations show that this improvement is 22.4% and 10.8% for 512-bit ripple carry adder and carry-skip adder respectively with potentially low area overhead. Our method is not limited to the application introduced. It is a general concept and can be adapted to other logic circuits with high latency and low activity.

References

[1] F. Ren and D. Markovic, "True energy-performance analysis of the MTJ-based logic-in-memory architecture (1-bit full adder)," Electron Devices, IEEE Transactions on, vol. 57, pp. 1023-1028, 2010.

[2] H. Meng, et al., "A spintronics full adder for magnetic CPU," Electron Device Letters, IEEE, vol. 26, pp. 360-362, 2005.

[3] S. Lee, et al., "The 3-bit gray counter based on magnetic-tunnel-junction elements," Magnetics, IEEE Transactions on, vol. 43, pp. 2677-2679, 2007.

[4] S. R. Patil, et al., "Design of a spintronic arithmetic and logic unit using magnetic tunnel junctions," in Proceedings of the 5th conference on Computing frontiers, 2008, pp. 171-178.

[5] "http://ptm.asu.edu/."

[6] X. Fong, et al., "SPICE Models for Magnetic Tunnel Junctions Based on Monodomain Approximation," ed, 2013.