Aging Effects of Leakage Optimizations for Caches

Aging Effects of Leakage Optimizations for Caches

Andrea Calimera[†], Mirko Loghi[‡], Enrico Macii[†], Massimo Poncino[†]
[†]Politecnico di Torino, 10129, Torino, ITALY
[‡]Universitå di Udine, 33100, Udine, ITALY

ABSTRACT

Besides static power consumption, sub-90nm devices have to account for NBTI effects, which are one of the major concerns about system reliability. Some of the factors that regulate power consumption also impact NBTI-induced aging effects; however, to which extent traditional low-power techniques can mitigate NBTI issues has not been investigated thoroughly.
This is especially true for cache memories, which are the target of this work. We show how leakage optimization techniques can also be leveraged to extend the lifetime a cache. Experimental analysis points out that, while achieving a total energy reduction up to 80%, managing static power can also provide a 5x factor on lifetime extension.
Categories and Subject Descriptors: B.3.2 [MEMORY STRUCTURES] : Design Styles.
General Terms: Design, Experimentation, Performance.
Keywords: Memory Hierarchy, Leakage Reduction, Aging.

1. INTRODUCTION

The long-term stability of a conventional six-transistor SRAM cell is strongly affected by temporal degradation of MOSFET parameters induced by Negative Bias Temperature Instability (NBTI) [1]. In particular, the increase over time of the threshold voltage of the PMOS transistors, under negative bias (i.e., a logic 0 at the input), results in a reduction in the Static Noise Margin (SNM) of the cell with consequences on the capability of the cell of reliably storing a value ([2]–[4]). Unlike random logic, these NBTI-induced effects cannot be tackled by forcing signal probabilities in order to reduce the occurrence probability of a logic 0; as a matter of fact, due to the symmetric structure of a cell, a SRAM cell ages whatever the value it stores.
Power (and in particular, static power) optimization techniques offer however some mitigation of NBTI effects in memories. The power management is implemented by either disconnecting a sub-block memory from the ground/supply network (*power gating*) or by reducing the supply voltage (*dynamic voltage scaling – DVS*). Both schemes have beneficial effects on NBTI-induced aging. Power gating has the effect of completely nullifying the aging effects [5, 6]. Similarly, but with a smaller impact, voltage scaling improves NBTI-induced aging because a reduced V_{dd} corresponds to a smaller bias voltage [7]. While these effects have been assessed on individual memory cells, the actual impact of their possible architectural embodiments in a multi-objective (performance, energy, aging) space has not been assessed in the literature. Objective of this work is to provide an explorative study of the complex interaction of these metrics as a function of the typical parameters of a memory hierarchy (namely, cache size and miss penalty), and of the power optimization paradigm adopted (power gating or DVS).
Results emphasize the fact that both power management schemes are extremely effective in lengthening the lifetime of a memory, even in the presence of significant miss penalties.

2. BACKGROUND

NBTI has emerged as the most relevant source of permanent, time-dependent variation of the transistor parameters for sub-90nm technologies. In particular, NBTI causes an increase over time of the threshold voltage of pMOS transistors, which in turn reduces the robustness of a SRAM cell. A conventionally accepted metric for the aging of a SRAM cell is the Static Noise Margin (SNM), defined as the minimum DC noise voltage necessary to change the state of an SRAM cell. NBTI impacts SNM because when the pull-up pMOS of the two cross-coupled inverters of the SRAM cell are negative biased, its V_{th} shifts over time, thus lowering the static characteristics of the two inverters. Therefore, after some time, the SNM falls below a (technology-dependent) value that allows safe storage of values.
A detailed treatment of NBTI effects and models is out of the scope of this paper, thus we refer the reader to classical tutorial papers on NBTI [1]. We summarize here the basic factors that impact NBTI effects:
Operating Conditions:. For a given set of technological parameters and physical dimensions (e.g., doping concentration, mobility, and oxide thickness) NBTI effects are mainly dependent on *(i) temperature* (delay degradation increases with increasing T), and *(ii) supply Voltage* (delay degradation increases with increasing V_{dd}).
Signal Statistics:. NBTI induced effects strongly depends on the actual amount of stress time (time in which the gate is negative biased). When stress is not applied (when $V_{gs} = 0V$), however, a partial *recovery* of of delay occurs.

Previous works on NBTI in the EDA domain have dealt with the issue of how NBTI impacts the SNM of an SRAM cell, for various technologies and operating conditions ([3]–[4]). The most relevant result was presented in [2]; based on the observation that an equal probability of storing a 0 or a 1 guarantees the minimum aging, they provide hardware and software schemes to periodically invert the entire content of a memory so as to guarantee a perfectly balanced probability. In [6], the authors assess the aging benefits provided by the application of power gating to a memory cell, observing that its impact can be much higher than the one obtained by controlling the value probability.

3. LEAKAGE OPTIMIZATIONS IN CACHES AND IMPACT ON AGING

Existing techniques for reducing leakage power in caches are essentially different embodiments of the power management problem. Based on the definition of (i) some granularity representing the unit that can be power-managed (e.g., one cell, one line, one set, one way), (ii) some metric of idleness, and (iii) some low-leakage state, these schemes simply *put a power-managed unit into a low-leakage state when idleness exceeds some threshold*.

The choice of granularity, idleness metric, and low-power state characterizes the various approaches. From the functional perspective, however, the most important dimension is the third one, i.e., the implementation of the low-leakage state. We can think of two main approaches, corresponding to two extremes of the leakage-performance space:

- *Non-state preserving* schemes, where by means of some form of *power gating*, the leakage of the power-managed unit is zeroed, but *its content is lost* ([10]–[13]).
- *State preserving* schemes, where, by means of some form of *dynamic voltage scaling*, the leakage of the power-managed unit is reduced (but not zeroed), while *keeping its content* ([14]).

Our objective is to assess how such solutions interact with the aging of memory cells; to this purpose, we must first characterize how the two basic mechanisms (namely, power gating and voltage scaling) impact the aging of a memory cell. Furthermore, we should remember that NBTI aging is value-dependent, thus we should also understand the role played by the content of a memory cell.

3.1 Impact of Values on SRAM Aging

As pointed out in previous NBTI characterization works for SRAM cells ([2]–[4]), a SRAM cell ages irrespective of the value it stores. The best case degradation happens when the value at the output of each inverter is 0 50% of the time, i.e., both PMOS degrade of the same amount. Otherwise, one of the PMOS transistors degrades faster than the other and the memory cell will fail earlier.

When considering an entire memory block from the functional standpoint, however, the situation is quite different. Consider, for example, a cache line as the unit of power-management. In a data cache, values in a line will change only if (i) there is a write in that line, or (ii) that line is replaced when fetching a block on a cache miss. Both events have low occurrence probability (writes are far less than reads, and miss rates are typically quite low). Furthermore, even when a line is overridden, very likely only a subset of the bits will toggle. Finally, the bit with the most skewed probability will determine the aging of the entire line it belongs to.

Based on these considerations, we can safely claim that the aging of a cache can be calculated based on the worst-case probability (i.e., fixed 0 or 1).

3.2 Impact of Power Gating on SRAM Aging

By power gating we denote the technique in which a footer transistor (*sleep transistor*) is used to disconnect a logic block from the ground. Power gating can be implemented using cell- or cluster-based approaches, or a distributed coarse-grained approach. The same holds when applying power gating to memory structures. Although the basic scheme [10, 11] uses one sleep transistor per cell, the transistor can be shared among multiple cells (e.g., a row of the memory [12, 13]). Whatever the granularity of the power gating, when the sleep transistor is on (active state), the cell will operate as usual, yet with a ground voltage equal to the virtual ground, therefore with a slightly worse performance (during read/write operations). Notice that the value of V_{VGND} does not affect the SNM, which is by definition is a DC quantity.

When the sleep transistor is off, the cell will be disconnected from the ground, and, both inverters' outputs will quickly reach the "1" value. Since this value corresponds to the recovery state, logic blocks in a stand-by state are naturally immune to NBTI-induced aging [5]. Notice that this is not a "logic" state: it is due to electrical reasons and cannot be forced by writing some value in the cell.

3.3 Impact of V_{dd} Scaling on SRAM Aging

The aging of a PMOS transistor is determined by the amount of negative bias voltage, i.e., gate-to-source voltage. Reducing V_{dd} (the source voltage) will therefore reduce the amount of bias accordingly. While the quantification of the V_{th} degradation as a function of bias voltage for a single PMOS device can be found in the literature ([1, 8, 9]), the analysis of its effect on the SNM of a memory cell was missing. We have thus run SPICE characterization on a custom-designed SRAM cell, mapped into a commercial 45nm technology by STMicroelectronics, using annotated netlist after parasitic extraction.

Results are reported in Table 1, which shows SNM degradation in % with respect to the nominal, time-zero value, for the nominal $V_{dd} = 1.1V$, and the "drowsy" $V_{dd} = 0.4V$ one. Values refer to the worst case of a fixed value 0 stored in the cell. The SNM degradation under the drowsy V_{dd} (36% of the nominal V_{dd}) is about 60% of that at the full V_{dd}.

	SNM Degradation [%]	
Years	$V_{dd} = 1.1V$	$V_{dd} = 0.4V$
3	20.41	12.77
6	24.08	15.06

Table 1: SNM Degradation as a Function of V_{dd}.

4. ARCHITECTURES AND MODELS

4.1 Low-Leakage Cache Architectures

We compare two generic cache leakage optimizations implementing one state-preserving and one non-state-preserving scheme, which are general implementations of approaches presented in the literature ([10]–[14]).

Both such schemes share two basic principles. First, *the granularity of the power management unit is a cache line*. Second, *the decision about whether to turn off a cache line is based on its usage*: lines that are not accessed since a given number of cycles (the *breakeven time*) are put into a low-leakage state. With power gating the line content is lost, so the line must be invalidated and a cache miss will occur. Conversely, using DVS, the content is preserved ("drowsy" state), and only a small time interval is required to restore the line back into the active state. The two conceptual architectures are shown in Figure 1. In both schemes, the block 'Control' implements the counting-based mechanism that triggers the de-activation of a line.

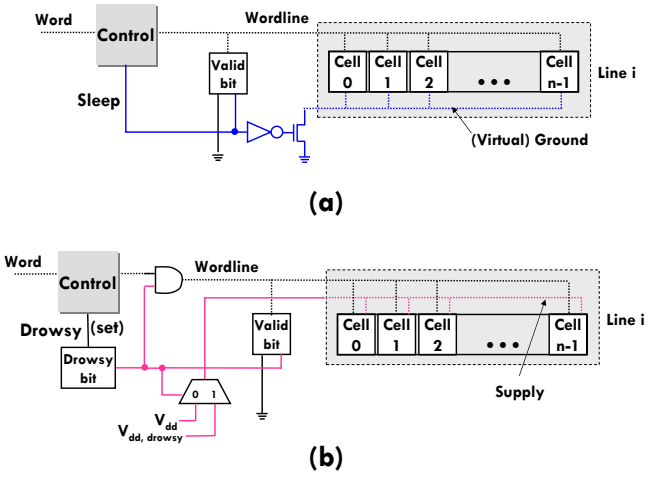

Figure 1: Reference Architectures: Power-Gated (a) and Drowsy Scheme (b).

The type of low-leakage state (gated or drowsy) affects the choice of the breakeven time B. States with lower leakage (gated) will have a longer breakeven time B. The latter is calculated as $B = \frac{E_T}{P_A - P_S}$, where E_T is the energy spent to put a cache line in the low leakage state and for the successive reactivation, while P_A and P_S are the leakage power spent by a cache line when in the standard (*active*) or in the low-leakage state respectively.

4.2 Aging Models and Metrics

4.2.1 Aging Model for the Power-Gated Scheme

For the power-gated scheme we can leverage the results presented in [5], where the probability P_{sleep} of the sleep signal (i.e., how often a cache line is the put into sleep) is used as a multiplicative factor of the stored value probability. In caches, since the dependence on the stored value is immaterial and we assume worst case, only P_{sleep} is relevant. The threshold voltage degradation can thus be compactly expressed as $\Delta V_{th}(t) = K \cdot ((1 - P_{sleep}) \cdot t)^{1/4}$, where K is a constant that lumps all the technological parameters (e.g., oxide electric field, thermal voltage, etc.) and t denote time. The term $(1 - P_{sleep}) \cdot t$ can be viewed as the effective stress time.

4.2.2 Aging Model for the Drowsy Scheme

For the drowsy scheme the evaluation of the aging is more elaborated. Since the aging curves are non-linear and since V_{dd} appears directly in the threshold degradation equation (the factor K shown above), the actual alternation of sleep and active interval matters (and not just their occurrence probability as for the power-gated case). This is pictorially explained in Figure 2.

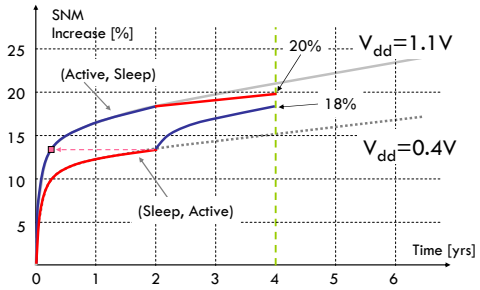

Figure 2: Effect of V_{dd} on Aging Depends on Temporal Sequence.

The plot shows the two SNM degradation curves over time corresponding to the full V_{dd} value of 1.1V (solid gray curve) and to the drowsy V_{dd} value of 0.4V (dotted gray curve). The plot shows the resulting SNM degradation corresponding to a four-year usage (two-years active mode, two years of sleep) for the two extreme cases of an active/sleep waveform with a 50% duty cycle: the worst case (Active, Sleep), and the best case (Sleep, Active). The former pattern yields a degradation of about 20% after four year, while the latter one results in a 18% degradation. Therefore, patterns where sleep intervals occurs earlier will have less aging and longer lifetimes. This is intuitive, since the curves grow quickly at the beginning and flatten out for larger time values.
In our methodology, in order to avoid the tracking of the detailed sleep/idle sequence, we assume the worst case pattern consisting of the *execution of all the active intervals first, and the sleep intervals then*. Although conservative, this analysis guarantees a safe lower bound to the lifetime of the cache.

4.2.3 Aging Metrics

SNM is the metric used for assessing the aging of a SRAM cell. As a compact metric of aging we define the **lifetime** of a cache as *the time at which the SNM decreases by 15% with respect to its nominal value*.
Extending lifetime to the entire cache is straightforward; since a line is the power management unit and we assume the worst case (always 0 or 1) for value probabilities, the lifetime of a line coincides with that of a single cell. By extension, the lifetime of the whole cache is the *shortest* lifetime among all the cache lines.
However, since absolute time has different performance interpretations for the two architectures (a given amount of time corresponds to different amounts of memory accesses, due to different miss rates), we lump performance and aging into a single metric, and we also express lifetime (ad defined above) in terms of *memory accesses*.

5. EXPERIMENTAL RESULTS

We have applied the above methodology on a set of traces obtained from the MediaBench suite [15]; these traces have been fed to a in-house cache simulator that was instrumented

Trace	Power Gating			Drowsy		
	LT [yrs]	LT [Pacc]	E [%]	LT [yrs]	LT [Pacc]	E [%]
adpcm.dec	1.25x	1.13x	75%	1.02x	1.02x	71%
adpcm.enc	1.19x	0.50x	41%	1.53x	1.53x	71%
cjpeg	3.85x	2.53x	65%	2.53x	2.53x	71%
CRC32	1.25x	0.50x	39%	1.41x	1.41x	73%
dijkstra	2.04x	1.49x	68%	1.71x	1.71x	71%
djpeg	3.37x	2.26x	65%	3.03x	3.03x	72%
fft_1	3.19x	1.77x	57%	3.33x	3.33x	72%
fft_2	3.88x	2.21x	58%	3.54x	3.55x	72%
gsmd	1.55x	0.73x	48%	1.53x	1.53x	72%
gsme	2.56x	1.30x	53%	1.86x	1.86x	72%
ispell	3.79x	2.53x	67%	3.33x	3.33x	73%
lame	5.20x	2.71x	54%	4.07x	4.07x	72%
mad	5.20x	3.19x	61%	4.60x	4.60x	71%
rijndael_i	1.52x	1.05x	65%	2.53x	2.53x	71%
rijndael_o	1.35x	1.00x	67%	2.29x	2.29x	70%
say	2.91x	1.55x	55%	2.55x	2.55x	71%
search	4.43x	3.02x	67%	2.29x	2.29x	73%
sha	4.05x	3.36x	74%	2.54x	2.54x	72%
tiff2bw	3.01x	2.89x	80%	2.18x	2.18x	73%
AVG	**2.9x**	**1.9x**	61%	**2.5x**	**2.45x**	72%

Table 2: Leakage, Performance and Aging Results for a 8K Unified Cache.

with aging models as described above and with energy models that have been obtained using 45nm technology data from STMicroelectronics.

The first set of results refers to the case of a 8K unified cache, for which we collected total (dynamic and static) energy of the whole memory hierarchy and aging values. Results are shown in Table 2, and refer to a miss penalty of 5 cycles. All figures are relative to the baseline case of a cache without any power management scheme; aging figures denote improvements, whereas energy figures reports percentage savings. Values are obtained by assuming an infinite repetition of the memory access pattern of the trace.

Notice that two values of lifetime are reported for each trace: one in absolute times, another one in memory accesses (specifically, 10^{15} accesses). This two units are necessary because, due to different miss rates and different breakeven time values, a given amount of time corresponds to different numbers of executed instructions.

The table offers a few interesting insights. As a first results, we see how both the power gating and the drowsy scheme provide significant improvements of lifetime (2.9x and 2.5x in terms of absolute times). Notice, however, that in the drowsy scheme a slightly shorter lifetime in years corresponds to an effective much higher number of useful accesses. This expresses the fact that much of the extra lifetime offered by power gating is spent in the transition from the low-power state and therefore not executing useful work. This is reflected by the performance penalty of the two schemes (not reported in the table): power gating results, as expected, in a large overhead (72.8%); conversely, the high exploitability of the drowsy state limits this overhead to 4.6% for the drowsy scheme.

Concerning energy, both schemes are obviously very effective, with the drowsy being best thanks to the smaller energy overhead. Although a sleep state using power gating results in zero aging, the longer breakeven times reduce the number of useful intervals; the drowsy architecture, while providing far less benefit in terms of aging reduction, can be triggered more frequently.

6. CONCLUSIONS

Reducing energy consumption of cache memories by exploiting a low-leakage state, is also effective in terms of lifetime extension. Analyses show that an aggressive approach provides good benefits, as in terms of energy saving as well as in terms of aging relief, if the miss penalty is reasonably small. On the contrary, whenever the miss penalty increases, more conservative techiques become more effective, because their smaller reactivation overhead.

The key difference between the idleness metrics that rule energy and aging (worst case for aging, average case for energy) suggests however that NBTI mitigation can be pushed further if specific architectural strategies are adopted.

7. REFERENCES

[1] M.A.Alam, "A critical examination of the mechanics of dynamic NBTI for PMOSFETs," *Proc. IEDM, 2003, pp.346-349*

[2] S.V. Kumar, K.H. Kim, S.S Sapatnekar, "Impact of NBTI on SRAM read stability and design for reliability," *ISQED'06: International Symposium on Quality Electronic Design* March 2006, pp. 213–218.

[3] K.Kang, H. Kufluoglu, K. Roy, M.A. Alam, "Impact of Negative-Bias Temperature Instability in Nanoscale SRAM Array: Modeling and Analysis," *IEEE Transactions on CAD*, Vol. 26, No. 10, pp. 1770-1781, Oct. 2008.

[4] V. Huard, et al., "NBTI Degradation: from Transistor to SRAM Arrays," *IEEE 46th Annual International Reliability Physics Symposium*, May 2008, pp. 289–300.

[5] A. Calimera, E. Macii, M. Poncino, "NBTI-Aware power gating for concurrent leakage and aging optimization", *ISLPED '09: International Symposium on Low power Electronics and Design*, August 2009, pp. 127-132.

[6] A. Calimera, E. Macii, M. Poncino, "Analasis of NBTI-Induced SNM Degradation in Power-Gated SRAM Cells," *ISCAS'10: International Symposium on Circuits and System*, MAy 2010, to be published.

[7] L. Zhang, R. P. Dick, "Scheduled Voltage Scaling for Increasing Lifetime in the Presence of NBTI," *ASPDAC'09: Asia & South Pacific Design Automation Conference*, pp. 492–497, Jan. 2009.

[8] R. Vattikonda, et.al. "Modeling and minimization of PMOS NBTI effect for robust nanometer design," *DAC-44: Design Automation Conference*, pp. 1047-1052, 2006.

[9] A. Calimera, E. Macii, M. Poncino, "NBTI-aware sleep transistor design for reliable power-gating," *GLS-VLSI'09: IEEE 19th Great Lakes Symposium on VLSI*, May 2009, pp. 333–338.

[10] K. Nii, et al., "A Low-Power SRAM using Auto-Backgate-Controlled MT-CMOS," *ISLPED'98: International Symposium on Low Power Electronics and Design*, August 1998, pp. 293–298.

[11] M. Powell, et al. "Gated-Vdd: A Circuit Technique to Reduce Leakage in Deep-Submicron Cache Memories," *ISLPED'00: International Symposium on Low Power Electronics and Design*, July 2000, pp. 90–95.

[12] S. Kaxiras, Z. Hu, and M. Martonosi, "Cache Decay: Exploiting General Behavior to Reduce Cache Leakage Power," *ISCA'01: IEEE/ACM International Symposium on Computer Architecture* June 2001, pp. 240–251.

[13] H. Zhou, M.C. Toburen, E. Rotenberg, T. M. Conte, "Adaptive Mode Control: A Static-Power-Efficient Cache Design," *ACM Transactions on Embedded Computing Systems*, Vol. 2, No. 3, August 2003, pp. 347–372.

[14] K. Flautner, N. Kim, S. Martin, D. Blaauw, T. Mudge, "Drowsy caches: Simple techniques for reducing leakage power," *ISCA'02: Int. Symp. on Computer Architecture*, May 2002, pp. 148–157.

[15] M. R. Guthaus et al., "MiBench: A free, commercially representative embedded benchmark suite", *IEEE 4th Annual Workshop on Workload Characterization*, pp. 3–14, Dec. 2001.