Drowsy Caches: Simple Techniques for Reducing Leakage Energy—A Retrospective

Krisztián Flautner, *Cisco*
Nam Sung Kim, *University of Illinois*
Steve Martin, *Broadcom*
David Blaauw, *University of Michigan*
Trevor Mudge, *University of Michigan*

I. Background

This paper was published at ISCA 2002 [1]. Most of the authors were at the University of Michigan. Nam Sung Kim and Steve Martin were PhD students, and David Blaauw and Trevor Mudge were faculty members. Krisztián Flautner was the Director of Advanced Research at Arm. The ideas were developed during the early 2000's. A paper trail of their development can be found in Drowsy Caches [1]–[4]. The impact of this work was recognized by an ACM SIGARCH/IEEE-CS TCCA Influential ISCA Paper Award in 2017. The paper proposed a technique for reducing the power consumption of L1 data caches. This was later extended to instruction caches in [4].

By 2002, metal–oxide–semiconductor (CMOS) logic had become the leading silicon technology. Its advantage over then competing technologies, such as transistor-transistor logic (TTL) and emitter coupled logic (ECL), was its lower power dissipation. It made large chips possible without exceeding the power constraint or power limit of a single chip. In the past when CMOS transistors did not switch, negligible amounts of power where dissipated. However, as the feature size of these devices decreased, their leakage (static) power dissipation became a growing and significant contributor. As process technology moved below 0.1μm, static power dissipation was on target to dominate the total power used by digital circuits [3]. Furthermore, leakage presents a tradeoff. On one hand, performance demands require the use of fast high-leakage transistors; on the other hand, new applications and cost issues favor designs that are energy efficient. Even in the technology of 2002, leakage power dissipation was becoming comparable to dynamic power dissipation. In fact, according to a projection from Intel [5] at that time, the off-state leakage component of the total power in a microprocessor was set to exceed active power as technology nodes moved below 65nm.

Leakage was becoming a problem for all transistors, but it was a particularly important problem in on-chip caches, because they were a growing fraction of the total number of microprocessor devices. For instance, 30% of Alpha 21264 (1998) and 60% of StrongARM are devoted to cache and memory structure [6]. A recent chip from Apple, the M2, now has 36 MB of shared L2 and a 48 MB system level cache (L3). The 2002 paper focused on reducing the power consumption of data caches by putting infrequently used cache lines into a low power state-retention drowsy state. They set the voltage of memory cells to a much lower level to reduce their leakage power while preserving their states.

II. The Drowsy Concept

A. Earlier Approaches

Cache designers were aware that reducing static cache power was an important goal. The first solutions simply turned off cache lines using the gated-V_{DD} techniques [7] have been described in [8], [9]. These approaches reduce leakage power by selectively turning off cache lines that contain data that is not likely to be reused. The drawback of this approach is that the state of the cache line is lost when it is turned off and reloading it from the L2 cache has the potential to negate any energy savings and have a significant impact on performance. To avoid these pitfalls, it is necessary to use complex adaptive algorithms and be conservative about which lines are turned off.

B. Drowsy Caches

To avoid the need to reload lines, significant leakage reduction can still be achieved by putting a cache line into a state-preserving low-power drowsy mode. The paper describes a drowsy mode in which the state in the cache line is preserved even though the supply voltage is reduced. It was still necessary to reinstated the line to a high-power mode before its contents could be accessed.

We proposed a design in which one can choose between two different supply voltages in each cache line, corresponding to normal supply voltage and a drowsy lower voltage. In effect we used a form of voltage scaling to reduce static power consumption. Due to short-channel effects in deep-submicron processes, leakage current reduces significantly with voltage scaling [10]. The combined effect of reduced leakage current and voltage yields a dramatic reduction in leakage power.

Drowsy caches do not reduce leakage energy as much as those that rely on gated-V_{DD}. However, we showed that for the total power consumption of the cache, drowsy caches can get close to the theoretical minimum. This is because the fraction of total energy consumed by the drowsy cache in low power mode tends to be so small that reducing it further may be possible but the pay-off is not great. Moreover, since the penalty for waking up a drowsy line is relatively small (it requires little energy and only 1 or 2 cycles) and there are less frequent accesses to the lower memory hierarchy than

GatedV_{DD} schemes, cache lines can be put into drowsy mode more aggressively to save more power.

The changes required in the circuitry of the cache line are detailed in [1], [4]. In brief, their is a drowsy bit to indicate when the line is in drowsy mode, circuitry to control the voltage to the memory cells, and a word line gating circuit to prevent state upset when the line is in a drowsy mode. Each cache line also requires two more transistors than a traditional memory circuit. The controller switches the supply voltage between high (active) and low (drowsy) depending on the state of the drowsy bit. If an access is made to a drowsy line, its drowsy bit is first cleared and the line is put into the active mode. The access can then go ahead. This wake up introduces a delay, which in our experiments were additional access delays of 1 and 2 cycles.

C. Drowsy Caches Wake-up Policies

Several policies for waking up drowsy lines were evaluated using a simulator based on SimpleScalar/Alpha 3.0 using the SPEC2000 benchmarks. The key difference between drowsy caches and caches that use gated-V_{DD} is that in drowsy caches, the cost of being wrong by putting a soon-to-be-accessed line into drowsy mode, is relatively small. The only penalty one must contend with is an additional delay and energy cost for having to wake up a drowsy line. One of the simplest policies that one might consider is that all lines in the cache, regardless of access patterns, are put into drowsy mode periodically and a line is awakened only when it is accessed again. This policy, referred to as the *simple* policy in [1], requires only a single global counter and no per-line statistics. In terms of the average leakage power reduction, roughly 85% of leakage power can be reduced with just 0.62% run-time increase when the average fraction of drowsy lines is 93% and a drowsy cache line consumes 10% leakage power of an awakened line.

III. Concluding Remarks

The work described in the 2002 paper was analyzed in the 70nm Berkeley Predictive Technology Model. That was 20 years ago. Since then there have been very significant changes to CMOS gate architectures. Notable improvements, among many, have been the introduction of FinFETS and recently Gate-All-Around gates. These have all allowed greater control of the basic transistor and, among other things, have prevented leakage power from getting worse with scaling. They have not eliminated it. Typically, in mobile devices it is kept as low as 10%, in desktop devices 20%, and in servers 30%. Static power still remains an issue but is one of many trade-offs that are factored into a design. If you look at today's Apple M2 chip, the various caches take a significant portion of the area, so leakage power still remains an issue and a good place to realize energy saving. Although the 2002 paper focused on L1 caches and proposed several advanced policies for setting lines into drowsy states, this was because added latency is very important for L1 caches. For lower level caches this is not so important—adding an extra 1-2 cycles when the latency is already 20+ cycles is less critical. The even simpler strategy of leaving all lines in a drowsy state but just waking them on demand and then placing them back into a drowsy state after the line is exited functions well.

References

[1] K. Flautner, N. Kim, S. Martin, D. Blaauw, T. Mudge."Drowsy Caches: Simple techniques for reducing leakage power", *29th Annual International Symposium on Computer Architecture*, May, 2002. Anchorage, Alaska. pp. 148–157.

[2] N. Kim, K. Flautner, D. Blaauw, T. Mudge. "Drowsy instruction caches: Leakage power reduction using dynamic voltage scaling and cache sub-bank prediction", *35th Annual IEEE/ACM Symposium on Microarchitecture*, November, 2002. Istanbul, Turkey, pp. 219-230.

[3] N. Kim, T. Austin, D. Blaauw, T. Mudge, K. Flautner, J. Hu, M. Irwin, M. Kandemir, N. Vijaykrishnan. "Leakage Current: Moore's Law Meets Static Power", *Computer*, vol 36, no 12. December, 2003. pp. 68-75

[4] N. Kim, K. Flautner, D. Blaauw, T. Mudge. "Circuit and microarchitectural techniques for reducing cache leakage power", *IEEE Transactions on VLSI*, vol 12, no 2. February, 2004. pp. 167-184.

[5] R. Doyle, R. Arghavani, D. Barlage, S. Datta, M. Doczy, J. Kavalieros, A. Murthy, and R. Chau. "Transistor elements for 30 nm physical gate lengths and beyond," *Intel Technology Journal*, vol 6, May 2002, pp. 42–54.

[6] S. Manne, A. Klauser, and D. Grunwald. "Pipeline gating: Speculation control for energy reduction," *25th Annual International Symposium on Computer Architecture*, July 1998, Barcelona, Spain, pp. 132–141.

[7] M. Powell, S-H. Yang, B. Falsafi, K. Roy, T. Vijaykumar ."Gated-Vdd: A circuit technique to reduce leakage in deep-submicron cache memories", *International Symposium on Low Power Electronics and Design*, 2000, Rapallo, Italy, pp. 90-95.

[8] S. Kaxiras, Z. Hu, and M. Martonosi. "Cache decay: Exploiting generational behavior to reduce cache leakage power," *28th Annual International Symposium on Computer Architecture*, 2001, Goteborg, Sweden, pp. 240-251.

[9] H. Zhou, M. Toburen, E. Rotenberg, T. Conte. Adaptive mode-control: A static-power-efficient cache design. *International Conference on Parallel Architectures and Compilation Techniques*, 2001, Barcelona, Spain pp. 61-70.

[10] S. Wolf. *Silicon Processing for the VLSI Era Volume 3—The Submicron MOSFET*. Sunset Beach, CA: Lattice Press, 1995, pp. 213–222.