# Time Domain Constraints on Switching with Gated-GND Cache Structures for Static Power Reduction

Pan Yan
*National University of Singapore*
panyan@nus.edu.sg

Tay Teng Tiow
*National University of Singapore*
eletaytt@nus.edu.sg

## Abstract

*As semiconductor technology goes into the deep submicron region, the static power caused by leakage current is no longer negligible. Gated-GND circuit technique switches SRAM cells into standby mode to reduce the static power. In this paper, we studied the detailed current scenario in a Gated-GND circuit structure. Based upon the analysis, we employed a more accurate way to estimate the efficiency of the Gated-GND technique and found that the efficiency is dependant on the length of the standby mode period. Thus we further proposed a time-domain constraint for using this circuit technique in higher level switching-based power reduction algorithms. Simulation shows the standby mode length should be no less than 0.175ms for a TSMC 180nm technology so as to achieve significant power reduction.*

## 1. Introduction

Each generation of integrated circuits fabrication technology pushes the limit on the number of transistors that can be packed onto a single chip. This allows complex logic and massive memory to be integrated into a single chip in modern-day processors. Coupled with the higher density integration, power consumption and heat dissipation have become major concerns.

For many years, efforts toward power reduction are focused on reducing dynamic power consumption, due mainly to the extensive use of CMOS technology where leakage in the static state is many orders of magnitude smaller compared to power consumed as a result of dynamic switching of states. Various techniques are proposed [7]-[10], among which the Dynamic Voltage Scaling technique (DVS) has been successful and widely adopted in commercial products [8]. However, as CMOS fabrication technology goes into the deep submicron region, the static power consumption in CMOS circuits caused by leakage current is no longer negligible. According to the International Technology Roadmap for Semiconductors (ITRS) projection, leakage current is 0.01uA/um for the 130nm and is projected to be 3uA/um for the 45nm technology [11]. Leakage power is predicted to have a five-fold increase with each technology generation [15].

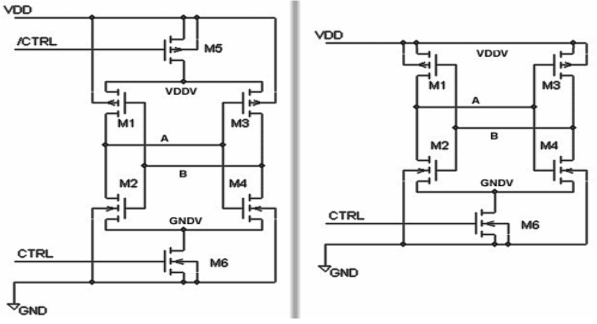
The shift is further amplified by the fact that for higher transistor density, there is a gradual, but steady change in the ratio of silicon usage for implementing on-chip memory compared to functional logic. This is especially prominent in high performance general purpose processors. Furthermore, for common memory structures, where access is a word or a small number of words at a time, the proportion of memory cells in the active switching mode compared to the static mode is small. For the high density memory portion of the chip, each small increase in static leakage per transistor has to be multiply by the corresponding large number of contributing transistors. On the other hand, dynamic power consumption tends to decrease as the capacitance is reduced with finer feature sizes.

It is noted that DVS is also effective in reducing leakage current in the static mode [16]. However, to cap the rapid rising static mode dissipation, other switching-based techniques [1]-[6] focusing on minimizing static power consumption, have been proposed and could be employed together with DVS. The Gated-GND technique [5] is one of the prominent solutions. While the method certainly is promising, we note that actual gain in a real operating environment has been over estimated. The efficiency of this method has always been estimated from a static view, that is, to compare the saturated leakage current in both the active and standby modes. The comparison however failed to account for two factors. Firstly, when a memory cell is gated off, the standby mode leakage current is not cut off abruptly. It decreases gradually after the cell is turned into standby mode. Secondly, upon turning a cell from standby mode to the active mode, a spike of current dissipation occurs at the standby-to-active switching point. In this paper, we proposed a novel way to accurately estimate the effectiveness of the Gated-GND technique. Furthermore, we also explored the limit to the application of this technique.

The remaining sections of this paper are organized as follows. In Section 2, we present the actual leakage scenario of the Gated-GND circuit. A more accurate way to estimate the efficiency of the Gated-GND circuit is proposed in Section 3. The limit to the application of the Gated-GND circuit technique is also explored.

## 2. Leakage Current in Gated-GND Circuits

The Gated-GND circuit technique employs an additional transistor to gate the power or ground lines. When switched off, the additional transistor can effectively reduce sub-threshold leakage [12], which plays a dominant role in the static leakage scenario of state-of-art CMOS circuits [6]. The anatomy of the circuit structure is shown in Figure 1.



(I) (II)

**Figure 1. Gated-GND Circuits**

By carefully choosing the dimension of M6 or M5 transistors, leakage current from $V_{DD}$ to GND can be dramatically reduced [2] due to the *stacking effect* [12]. As both M5 and M6 serially sit in the path from $V_{DD}$ to GND, employing only one of them is enough, and a single NMOS M6 is preferably used due to its smaller size for the same performance. In the following simulations and discussions, we will only focus on circuit type II in Figure 1.

In many research reports [1][2][4] discussing the efficiency of the Gated-GND circuit technique, a simple static view is adopted, which is, to compare the leakage current in both active and standby modes and to claim the difference as the savings. This is intuitive, as the leakage current constitutes the "static" power consumption. However, our detailed analysis of the behavior of the Gated-GND circuit shows that the leakage current in standby mode is not constant, but decreases gradually after the cell is switched into standby mode. To be more accurate in estimating the efficiency, we are interested in the leakage current profile right after the cell is switched off into standby mode. This requires transient analysis of the circuit. We approached it through simulation and support the results using simplified circuit models.

### 2.1 Transient Analysis Simulation

We simulated the circuit structure shown in Figure 3 (I). The dimension of each MOSFET is as tagged in the figure. The unit here is the minimum channel length of each technology. Initially, we assume Node A is low while Node B is high. We performed a transient analysis of the circuit with CTRL signal being a square wave of an appropriate frequency. The simulation was done with Spectre Simulator in the CADENCE IC Design Environment.

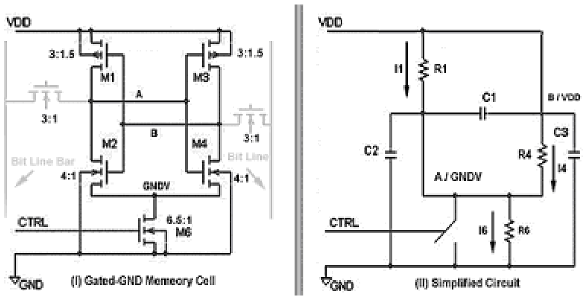


**Figure 3. Simplified Circuit for Gated-GND Memory**

Using TSMC 180nm SPICE models for the MOSFETs and a 1.8V VDD, simulation shows the results in Figure 3.

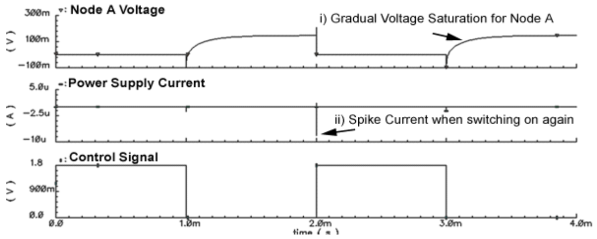


**Figure 2. Simulation Results for TSMC 180nm**

Statistics are listed in the table below.

**Table 1 Simulation Setting and Results**

| | |
|---|---|
| Power Supply ($V_{DD}$) | 1.8 V |
| SPICE Model Provider | TSMC |
| Channel Length | 180 nm |
| Active Mode Leakage | 17.79 pA |
| Saturated Standby Mode Leakage | 8.934 pA |
| Saturated Node A Voltage ($V_{Asat}$) | 146.3 mV |
| Node B Voltage (Stuck at VDD) | 1.8 V |
| Time for $V_A$ to reach 90% of $V_{Asat}$ | 324 uS |

### 2.2 Gradual Saturation

Figure 2 shows that the voltage of Node A increases gradually and saturates at some intermediate value between $V_{DD}$ and GND. This essentially means that asserting a low CTRL signal does NOT instantly send the memory cell into the stationary standby mode. Instead, it takes some time for the voltage of Node A to go up to the saturated value.

We employ a simplified circuit model to explain this effect. From Figure 3 (I), during active mode with CTRL is high, Node A is tied to GND by M2 and M6, while Node B is tied to VDD by M3. M1 and M4 are therefore shut-off. We use a high value non-linear resistor to represent these two off-state transistors (R1 for M1 and R4 for M4) and the on-state transistors (M2 and M3) are simply modeled as wires. As M6 is controlled by CTRL signal, it is a wire or high-value resistor in either mode. This is represented as a resistor put in parallel with a switch. We only consider the node-associated capacitors and the cross-node capacitor for the circuit, as they play the most significant role in the switching of the memory cell. Our simplified circuit model of the static memory cell is shown in Figure 3 (II).

In active mode, the voltage across C2 ($V_A$) is obviously zero, while in stationary standby mode, this voltage should be $V_A$=$V_{DD}$R6/(R1||R4+R6). Once the cell is switched into standby mode, $V_A$ could not be charged up instantly, and the leakage current (I1+I4) will be greater than I6, which is always given by I6=VA/R6. Thus, Node A will be charged up, that is, $V_A$ will go up. As $V_A$ gradually goes up, I1 and I4 will decrease while I6 will increase. The stationary state will be a certain value of $V_A$ letting (I1+I4) = I6. This charging is done by the leakage through two shut-off transistors (M2 and M4), so it will take some time. As shown in Table 1, it takes 324uS for $V_A$ to reach a value of 90% of the saturated value. This is a long time for processors.

On the other hand, the current leakage through this cell is actually given by (I1 + I4). Qualitatively, (I1+I4) = ($V_{DD}$-$V_A$)(1/R1 + 1/R4). This means, a lower $V_A$ gives a higher leakage. So the gradual increase of $V_A$ actually corresponds to the gradual reduction of leakage current. The leakage current is not reduced instantly after asserting a low CTRL signal. The actual leakage consumption in standby mode is a time-dependant value, decreasing with time. The accurate way to express the leakage energy consumption in standby mode should then be the power supply voltage multiplied by the integration of this time-dependant leakage current.

### 2.3 Spike Current at the Switching Point

Figure 2 also shows a clear, huge current spike at the point of switching from standby mode back into active mode. Again, we use the simplified circuit model shown in Figure 3 (II) to explain this effect.

In standby mode, Node A is gradually charged up, and the voltage across C1 is ($V_{DD}$ – $V_{A,standby}$). When switched into active mode again, Node A will be discharged almost instantly, but Node B is stuck at $V_{DD}$. So C1 will be charged from (VDD – VA,standby) to VDD nearly instantly. This will cost an amount of charge in a very short period of time, which constitutes the current spike.

Figure 4 confirms this notion as we find that the "size" of the spike is dependent on the standby time and in turn the value of $V_{A,standby}$. This current spike also represents a part of power consumption during the standby mode. The charges stored at Node A in standby mode lowered the voltage across C1, which should always be $V_{DD}$ in active mode. This voltage loss is made up nearly instantly when switched back to active mode. So even though rigorously speaking this spike appears in the active mode period, it represents power dissipation in the standby mode.

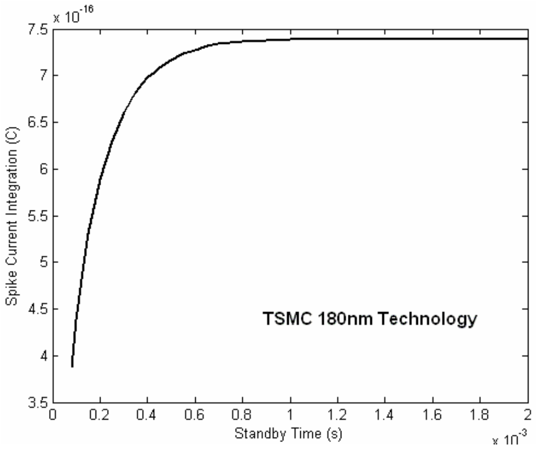


**Figure 4. Integration of the current spike**

## 3. Switching Frequency Limit

With the knowledge of the actual leakage current situation discussed in the previous section, it is clear that simply estimating the standby mode power consumption by a saturated leakage current ($I_{sat}$) multiplied by the power supply voltage is not sufficient. A more accurate way is to integrate the time-dependant standby leakage current and multiply the result by the power supply voltage. Equivalently, we can calculate the Averaged Standby Leakage current ($I_{ASL}$) for the whole period of standby mode, given by

$$I_{ASL} = \frac{\int_{T_{\text{Standby Period}}} I_{\text{standby}}(t)dt + \int_{T_{\text{Spike Period}}} I_{spike}(t)dt}{T_{\text{Standby Period}}}$$

Here $I_{standby}$ is the time-dependant leakage current in the standby mode, which is integrated over the whole standby period. $I_{spike}$ is, on the other hand, the spike current appearing at the switching point.

With this $I_{ASL}$, the power dissipation in standby mode is represented as $I_{ASL}$*$V_{DD}$, while the power dissipation in the active mode is $I_{active}$*$V_{DD}$. Here $I_{active}$ is the constant leakage in active mode. Thus the power saving by switching the memory cell into standby mode is given by:

$$P_{save} = (I_{active} - I_{ASL}) \times V_{DD}$$

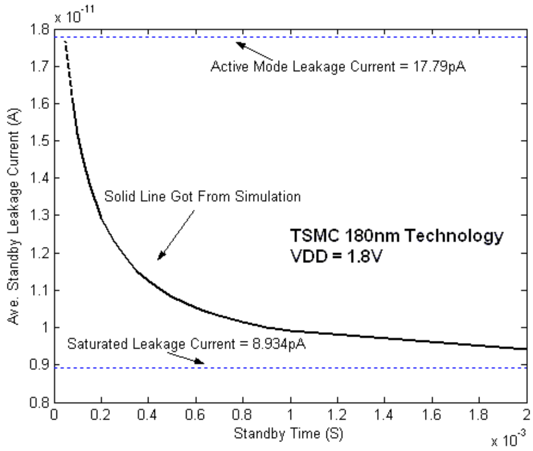


**Figure 5. Ave. Standby Leakage ($I_{ASL}$) Vs. Standby Time**

It is important to note that $I_{ASL}$ is also a time-dependant value, that is, it varies with the length of the standby mode period.

$I_{ASL}$ has two components: the current spike and the standby leakage. Figure 4 shows that the integration of the spike current is bounded, and according to the discussion in the previous section, the leakage current in standby mode will saturate to Isat. Thus, $I_{sat}$ is the lower bound of $I_{ASL}$, and $I_{ASL}$ will approach $I_{sat}$ when the standby period length is long enough. On the other hand, as the standby mode period goes shorter, $I_{ASL}$ will quickly approach $I_{active}$. This is also reasonable, because if the standby mode period is too short, the gated low voltage node (Node A in the previous example) will not be charged sufficiently to reduce the leakage current.

Figure 5 shows our simulated $I_{ASL}$ value with varied Standby-mode Period Length, with $I_{active}$ and $I_{sat}$ actually appearing as the upper and lower bounds.

At this point, we may conclude that switching a memory cell into standby mode for short periods does not effectively save energy. This is an important guide to higher level algorithms that seek to shut down circuits or memory cells that are not actively used.

Conventionally, scheduling algorithms focus on the percentage of time that a part of the cache can be sent into standby mode. To achieve better down-time ratio, it is advantageous to divide time slots into finer grains, so that more time slots could be identified as inactive and therefore sent into standby mode. For example, Roy et al proposed to send a memory cell in the cache into standby mode by the Gated-GND circuit technique once it is not accessed [4]. In such an algorithm, the percentage of time a cell is in standby mode is extremely high. However, such a design may at the same time result in very frequent switching, that is, each cell could be in standby mode for a very short period of time and then switched to active mode for an even shorter period. However, based on our analysis and simulation results, such short-term standby modes will not give noticeable power saving. In fact, the length of the standby mode period ($T_{standby}$) is also an important factor for estimating the efficiency of the scheduling algorithm working on a Gated-GND switching technique. With the simulated ASL curve, we can give a reference value of this factor ($T_{standby, typical}$), beyond which, switching into standby mode will not be efficient.

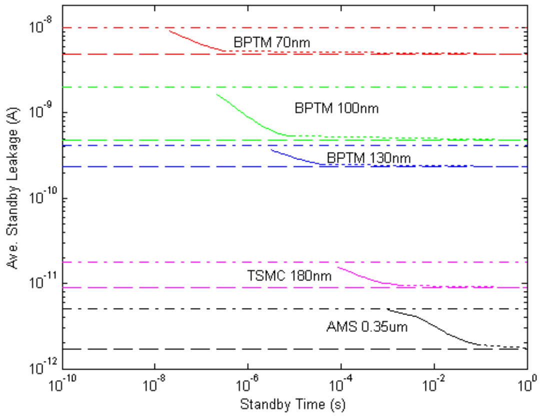


**Figure 6. $I_{ASL}$ versus Standby Time for different technologies**

Let us choose $T_{standby, typical}$ as the $T_{standby}$ at which the $I_{ASL}$ is midway between $I_{active}$ and $I_{sat}$. We simulated various technology models for MOSFETs to see the effect of technology progress on this reference factor. This is shown in Figure 6.

Table 2 gives the reference values for standby period length (Tstandby, typical).

**Table 2 Typical Standby Period Length**

| Technology | $T_{standby, typical}$ (s) |
|---|---|
| BPTM 70nm | 5.12e-8 |
| BPTM 100nm | 4.55e-7 |
| BPTM 130nm | 6.47e-6 |
| TSMC180nm | 1.75e-4 |
| AMS 0.35um | 8.3e-3 |

In Figure 6, BPTM is the Berkeley Prediction Transistor Model. The dash-dot lines are active mode leakages for each technology while the dash lines are saturated standby leakage current for each technology. It can be inferred that for each technology generation, the $I_{ASL}$ shows a similar dependence on the standby period length, and $T_{standby, typical}$ becomes shorter for advanced technologies. This is again reasonable, because the speed of a circuit is increased by minimizing the time to charge/discharge parasitic capacitors, and the speed of the charging/discharging of information carrying node (Node A in our example) in the SRAM cell by leakage decides the switching limit. The fundamental mechanism is similar. We can thus conclude that this standby-mode period length limit factor should be taken into account whenever strategic algorithms are employed to shut-down the processor or its associated memory structure so as to minimize energy.

## 4. Conclusion

In this paper we focused on the Gated-GND circuit technique and analyzed in detailed the leakage scenario in standby mode. Based upon the analysis, we proposed a novel way to precisely estimate the actual power saving of Gated-GND, and we showed that the standby mode period length is an important factor in the design of scheduling algorithms based on Gated-GND switching.

## 1. References

[1] Michael Powell, Se-Hyun Yang, Babak Falsafi, Kaushik Roy, "Reducing Leakage in a High-Performance Deep-Submicron Instruction Cache", *IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION (VLSI) SYSTEMS, VOL. 9, NO.1, FEBRUARY 2001*

[2] Michael Powell, Se-Hyun Yang, Babak Falsafi, Kaushik Roy, and T. N. Vijaykumar, "Gated-Vdd: A Circuit Technique to Reduce Leakage in Deep-Submicron Cache Memories", *ACM/IEEE International Symposium on Low Power Electronics and Design (ISLPED), 2000*

[3] Amit Agarwal, Hai Li, Kaushik Roy, "A Single-Vt Low-Leakage Gated-Ground Cache for Deep Submicron", *IEEE JOURNAL OF SOLID-STATE CIRCUITS, VOL. 38, NO. 2, FEBRUARY 2003*

[4] Amit Agarwal, Hai Li, Kaushik Roy, "DRG-Cache: A Data Retention Gated-Ground Cache for Low Power", in *Proc. Design Automation Conf.*, 2002, pp.473-478.

[5] Kaushik Roy, Saibal Mukhopadhyay and Hamid Mahmoodi-Meimand, " Leakage Current Mechanisms and Leakage Reduction Techniques in Deep-Submicrometer CMOS Circuits", Contributed Paper in *PROCEEDINGS OF THE IEEE, VOL. 91, NO. 2, FEBRUARY 2003.*

[6] Chris H. Kim and Kaushik Roy, "Dynamic Vt SRAM : A Leakage Tolerant Cache Memory for Low Voltage Microprocessors", *Int. Symp. Low Power Electronics and Design, Monterey, CA, Aug 2002.*

[7] Thomas D. Burd, Trevor A. Pering, Anthony J. Stratakos, and Robert W. Brodersen, "A Dynamic Voltage Scaled Microprocessor System", *IEEE JOURNAL OF SOLID-STATE CIRCUITS, VOL. 35, NO. 11, NOVEMBER 2000*

[8] Intel® Pentium® M Processor on 90nm Process with 2-MB L2 Cache Datasheet, June 2004.

[9] Bo Zhai, David Blaauw, Dennis Sylvester, Krisztian Flautner, "Theoretical and Practical Limits of Dynamic Voltage Scaling", *ACM/IEEE Design Automation Conference (DAC), June 2004*

[10] Shih Wei Sun; Tsui, P.G.Y., "Limitation of CMOS supply-voltage scaling by MOSFET threshold-voltage variation", *Solid-State Circuits, IEEE Journal of, August 1995*

[11] International Technology Roadmap for Semiconductors. [Online]. Available: http://public.itrs.net/

[12] Z. Chen, L. Wei, M. Johnson, K. Roy. "Estimation of Standby Leakage Power in CMOS Circuits Considering Accurate Modeling of Transistor Stacks", *Int. Symp. On Low Power Electronics and Design*, 1998, pp. 239-244.

[13] Y. Cao, T. Sato, D. Sylvester, M. Orshansky, and C. Hu, "New paradigm of predictive MOSFET and interconnect modeling for early circuit design," *Proc. of IEEE CICC, pp. 201-204, Jun. 2000.* URL of BPTM: http://www-device.eecs.berkeley.edu/~ptm

[14] TSMC SPICE Models, Got from http://www.mosis.org/Technical/Testdata/

[15] S. Borka. Design challenges of technology scaling. In IEEE Micro, pages 23029, Aug 1999

[16] Nam Sung Kim, Krisztian Flautner, David Blaauw, Trevo Mudge. "Drowsy Instruction Caches – Leakage Power Reduction using Dynamic Voltage Scaling and Cache Sub-bank Prediction", *35th Ann. IEEE/ACM Symp. Microarchitecture (MICRO-35)*, Nov. 2002, pp. 219-230.