Refreshing Thoughts on DRAM: Power Saving vs. Data Integrity

Amir Rahmati, Matthew Hicks, Daniel Holcomb, and Kevin Fu
University of Michigan

Abstract

To head-off the trend of increasing power consumption and throughput overheads due to refresh in DRAM, researchers are exploring ways to fine-tune refresh rate. Refresh management proposals range from temperature-aware refresh to partitioning data cells based on volatility, with each partition having its own refresh rate. One hurdle in this area is the lack of precise description of the evaluation setups used in current proposals. The incomplete description makes it difficult to reproduce reproduce results and compare approaches. Thus, it is common for researchers to evaluate their approach using mathematical models derived on the experimental results from a disparate set of platforms.

To aide in the evaluation of existing and future approaches for tuning DRAM refresh to save power, this paper provides a reproducible DRAM research platform. On this platform, we re-run commonly cited experiments from previous papers in the area. Our experimental results highlight the necessity of a reproducible platform by showing how both assumptions and experimental outcomes are sensitive to often omitted parameters like temperature and refresh duration.

1 Introduction

Dynamic Random Access Memory (DRAM) is a volatile memory: as soon as you write data it begins to disappear. To counteract this gradual decay of values, DRAM must constantly refresh its data by reading and rewriting data in each cell. The refresh must happen often enough that no data is lost between two refreshes. Traditionally, the refresh interval is set conservatively to prevent data loss in the worst case scenario: where fast decaying cells operate at high temperatures.

DRAM typically operates in a setting that is far from worst-case. In such a setting, the frequent refreshes dictated by worst-case conditions enforce unnecessarily high power overhead. To reclaim this wasted power, researchers have proposed several systems for managing the refresh rate of DRAM. Approaches tend to focus on three parameters voltage [11, 3, 4], temperature [11, 9], and cell volatility differences [12, 10, 21, 20]. While the knob used to control DRAM power differs, most of the approaches rely on a similar set of prior experiments for their evaluation. Proposals that require new measurements are unable to give a fair comparison against prior works that often omit critical information, such as temperature and DRAM chip identification. Highlighting the need for a common platform, we show that the viability of various refresh schemes depends on the specific properties of the target chip, as we show in Section 3.3.

This paper covers the design of a reproducible experimental platform to aide researchers in evaluating existing and future approaches to reducing DRAM power by managing refresh. We use this experimental platform to repeat prior experiments that previous papers in the area commonly cite. Finally, we re-analyze several previous proposal in light of our experimental results.

This analysis highlights the need for a reproducible

platform as it shows the range of interpretations possible given the level of descriptions of the experimental systems used in many papers on reducing DRAM power.

We make three contributions in this paper:

- We describe and implement a reproducible platform for future research on DRAM refresh.
- Using our experimental platform, we reproduce experiments central to DRAM refresh research.
- We analyze previous proposals in light of our experimental results.

2 Experimental Platform

We perform experiments using a 32KB KM41464A DRAM chip [17]. This DRAM chip stores data as 64K 4-bit words, arranged in 256 columns and 256 rows. The chip supports full-auto refresh as well as row-level refresh for experiments involving partial array refresh [12]. The chip also supports the ability to completely disable automatic refresh, a feature that we rely on in our experiments (Section 3).

Figure 1 shows a high-level block diagram of the setup used to perform experiments on the DRAM chip. The relevant blocks and their roles are,

- The MSP430-F2618 [19] microcontroller orchestrates the experiments. Its duties include writing and reading data to and from the DRAM, controlling the timing of refreshes, and analyzing the data from the DRAM for decay.
- The AQV103 solid-state relay [16] gates power to the DRAM according to control signals form the MSP430.
- The Data acquisition unit (DAQ) measures DRAM power consumption by monitoring the voltage across a 200Ω sense resistor placed in series with the DRAM power input. We configure the DAQ to take a measurement 1000 times a second. The DAQ also monitors two digital outputs from the MSP430 that encode the current mode of the test software.
- The Sun Electronics EC-12 thermal chamber [18] allows us to control temperature for the DRAM experiments. Our experimental results vary greatly with temperature, so the thermal chamber is essential for reproducibility.

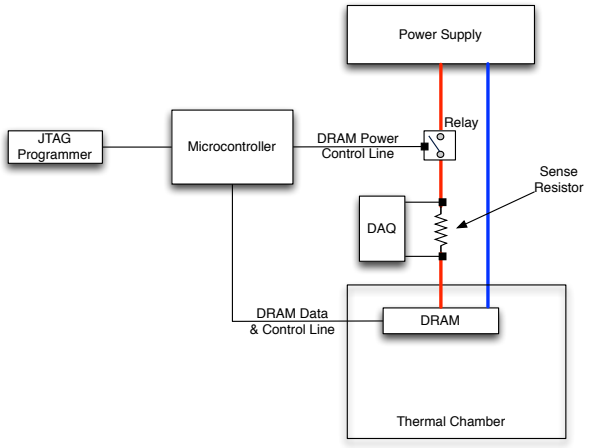

Figure 1: *DRAM research experimental platform. The MSP430 microcontroller controls the DRAM power using a relay. A Data Acquisition Unit reads the voltage across a sense resistor for power measurements. The target DRAM is placed inside a thermal chamber to ensure environment consistency across experiments.*

3 Evaluation

Using the experiment setup described above, we evaluated a range of parameters that affect the DRAM retention time. Specifically, we examine,

- The effect of power gating on data retention.
- The effect of temperature on data retention.
- The spatial locality of decay in the DRAM.
- The probability distribution of cell volatility.
- DRAM power consumption in different modes.

3.1 Effect of Power Gating

Power gating the memory refers to lowering or cutting power supply from DRAM chip between data refreshes. Previous work, such as Deng [4], considers dynamic change of memory voltage as a power saving method in DRAM. To examine the effect of

power gating on memory decay, we load the memory with high values, use the relay to gate power, and let the values decay for various time durations. After the prescribed delay, we close the relay, and read the data from the DRAM. For each delay and each DRAM cell, we record the number of bit flips between what we wrote and what we read after the delay. To provide an idea of scale, we repeated the experiment, but we maintained the specified voltage. We performed ten trials at each DRAM cell for each delay and each power configuration.

Figure 2 presents the results of these experiments at 30 °C. The results show that memory decays an order of magnitude faster when the power is completely removed than when power is maintained between refreshes.

Discussion: Although cutting power from memory between refreshes might appear as a great energy saving method, the order of magnitude increase in data volatility caused by power gating would require an order of magnitude decrease in power consumption to begin to compete with non-power-gated approaches. Experiments in Section 3.5 show that, in our system, the power saving of power gating is no where near enough. Current state of the art technology supports *Partial Array Refresh (PAR)* which enables coarse grained selective refresh of the parts of memory that contain data [2, 14]. More fine grained control over power gating in active parts of DRAM might provide additional power savings at the costs of increased circuitry.

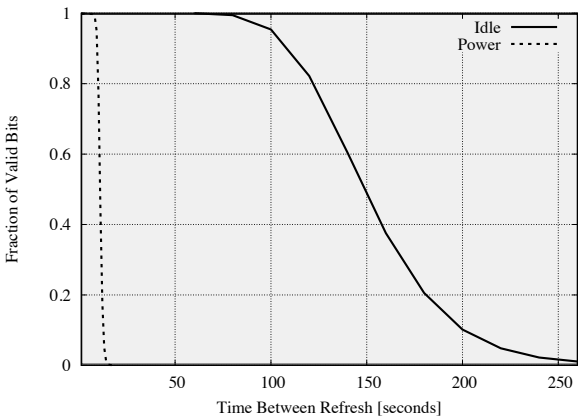

Figure 2: *The effect of power gating on DRAM decay: The DRAM chip decays an order of magnitude faster when the power is cut off between refreshes.*

3.2 Effect of Temperature

Temperature plays a large role in memory decay by affecting power leakage at the transistor level. To explore the chip-level impact of temperature, we ran the experiments described in the previous section, but varied the thermal chamber temperature between 30 °C and 60 °C, by 10 °C increments. As before, we repeated experiment 10 times (this time adding the temperature variable). Figures 3 and 4 show the results of this experiment for both normal and power gated mode, respectively. The effect of temperature on DRAM operation observed in our experiments matches previous experiments by Hamamoto *et al.* [6]. The figures show the temperature has a pronounced effect on DRAM data volatility: increases in temperature in both scenarios logarithmic increase the volatility of data. The results also show that the order-of-magnitude difference in volatility between power gated and idle modes is maintained across temperature.

Discussion: Memory manufacturers consider maximum operational temperature when calculating calculate the refresh rate of DRAM which explains much of the large error margin observed in the previous section. Some DRAM chips designed for mobile devices, now support *Temperature Compensated Refresh (TCR)* which enables them to offset some of this large error margin by modifying the refresh rate based on the temperature of DRAM chip [2, 14]. Other DRAM power energy saving schemes such as Flikker [12] and RAPID [20] which seek to reduce DRAM refresh rate to achieve energy saving similarly need to assume the operational upper bound for their systems: they do *not* compensate for temperature. Our results show that power and temperature have a dramatic impact on data volatility, thus DRAM refresh proposals need to include a knob for both to minimize power.

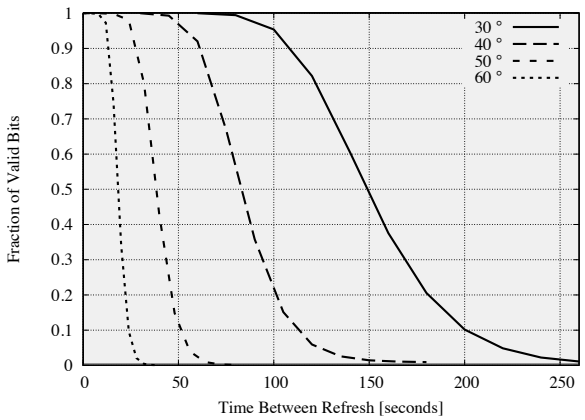

Figure 3: *The effect of temperature on data volatility in DRAM at different refresh rates. Power is maintained between refreshes.*

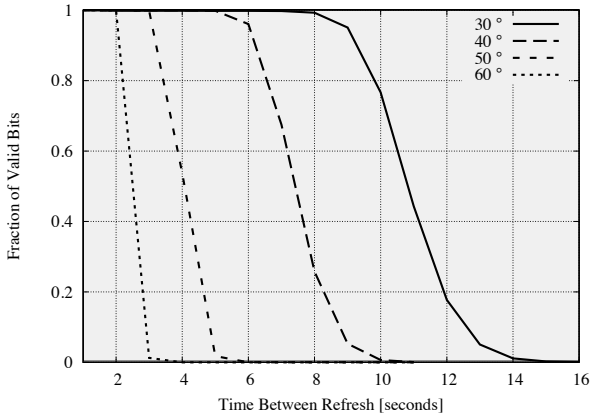

Figure 4: *The effect of temperature on data volatility in DRAM at different refresh rates. Power is gated between refreshes.*

3.3 Decay Pattern

To observe the decay pattern of memory, we repurpose the results from the previous section. Figure 5 shows the spatial relationship of DRAM data cell locality at both the chip level and at the cell level. The results shown in Figure 5 are from the power-gated experimental trials performed at 60 °C. While we only show one example, our data shows that the lack-of spatial relationship between cell volatility holds for all other experiments that we performed: cells tend to fail in the same order, regardless of power, refresh rate, or temperature. This results means that DRAM cell volatility relative to other cells is heavily tied to intrinsic properties of the cell (*e.g.,* manufacturing variances), and these intrinsic properties are not readily predictable. Our results are similar to observations made by Halderman *et al.* [5]: their is no spatial locality to DRAM cell volatility at any sub-chip granularity.

Discussion: Refresh management proposals by Venkatesan *et al.* [20] and Kim *et al.* [7] propose dividing DRAM into different partitions based on cell volatility. Our experimental results suggest that implementing such schemes effectively requires per cell basis control for refresh time which requires large amounts of additional circuitry and software-level control.

3.4 Volatility Distribution

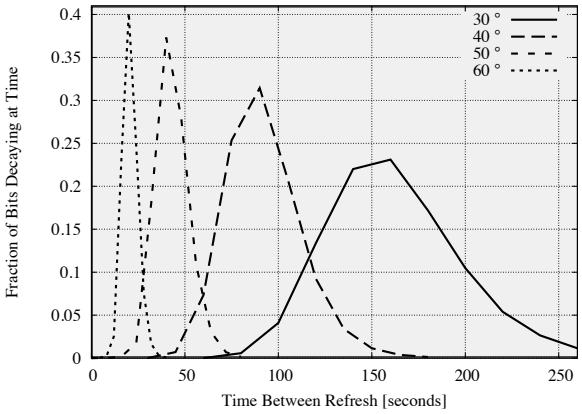

Figure 6: *A histograms of cell failures when the DRAM idles between refreshes. As temperature increases, data cells tend to behave the same (at a constant granularity).*

Figures 6 and 7 show another view of data from previously described experiments for both normal and power gated mode, respectively. The results show that cell volatility follows a standard probability distribution regardless of temperature, refresh rate, and power. The figures also show that as temperature increases, DRAM cells tend to behave more

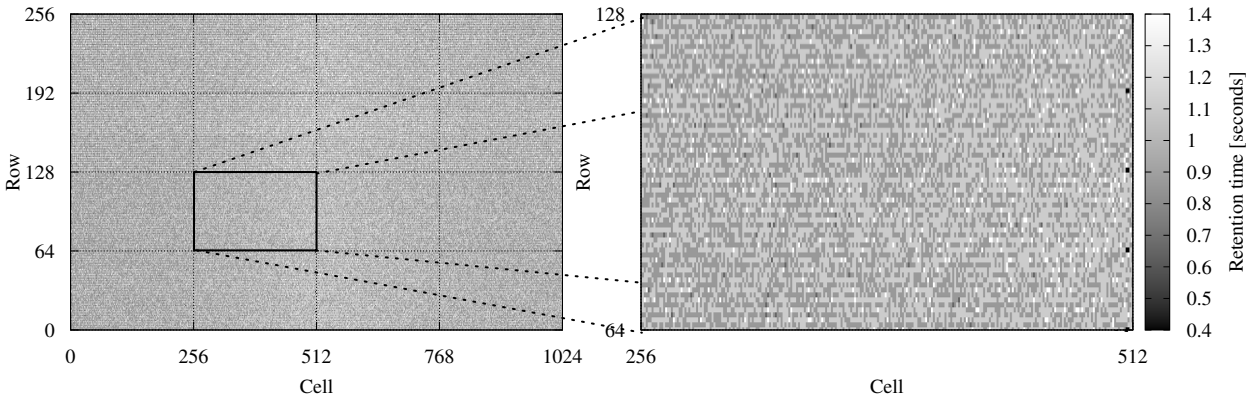

Figure 5: *Spatial locality of DRAM data volatility: there is no obvious relation between decay time of adjacent cells, at any sub-chip granularity. Darker colors indicate faster decay times.*

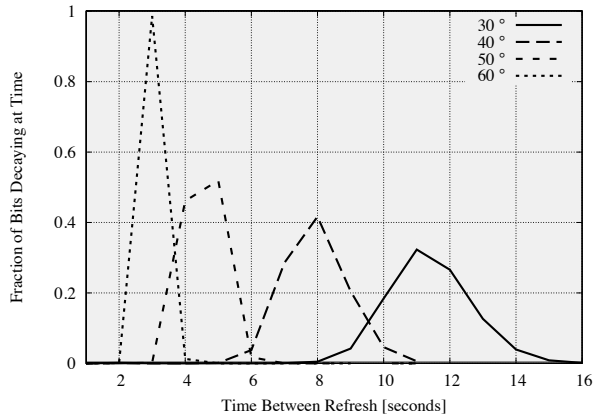

Figure 7: *A histograms of cell failures when power is gated between refreshes. As temperature increases, data cells tend to behave the same (at a constant granularity).*

similar, *i.e.,* temperature increases lead to volatility variance decreases.

Discussion: The results of this experiment are relevant to approaches for managing DRAM refresh that are based on the idea that a very few, highly volatile data cells dictate the refresh rate. For approaches like RAIDR [10] and Kim *et al.* [7] that seek to identify and isolate these tyrannical cells, thus allowing for a lower refresh rate, it is crucial that the histograms in Figures 6 and 7 have long tails in the direction of low refresh periods. Given our result of similarity to a standard probability distribution, the tail of the curve is directly related to the variance—95% of cells will be within 2 standard deviations of the mean. The implication of our results is that for approaches that seek to isolate highly volatile cells, thermal management is essential.

3.5 Power Usage

To measure power consumption of DRAM, we used a Data Acquisition Unit (DAQ) to record the voltage across a sense resistor. To determine the current consumption of the DRAM at a given moment, we measure the voltage across the sense resistor, which is in series with the voltage supply of the DRAM. We then divide the measured voltage by the measured Ohm value of the sense resistor to determine the current supplied to the DRAM. Table 1 shows the average current draw of our DRAM chip at 30 °C in both idle and refresh modes.

Given supply voltage V_{SS} and sense resistor voltage reading V_R for resistor R, the voltage and current of our memory in these conditions is $V_{SS} - V_R$ and $\frac{V_R}{R}$, respectively. Using this data, the average power usage of memory under different modes of operation

Mode	Avg. Current Draw	Avg. Power Usage
Refresh	$2.6mA$	$11.65mW$
Idle	$2.0mA$	$9.20mW$

Table 1: *Power usage of DRAM chip in different modes. In idle mode, the DRAM chip uses* 23% *less power than refresh mode.*

can be calculated using equation 1.

$$P_{avg} = \frac{1}{t}\int_{0}^{T} V_{(t)} I_{(t)} dt \quad (1)$$

Discussion: The high current draw of memory in idle mode compared to refresh mode challenges the premises of high energy saving possibility by reducing the refresh time of memory in active systems. Add to this the fact that refresh occurs infrequently and quickly. In our experiments, less than 1% of DRAM time is used for refreshing. As such, reducing the refresh rate of our chip leads to at most 0.2% of power saving. What about larger, more modern, DRAM? In the current state of the art DDR4 DRAM, refresh only occupies between $4.5\% - 8\%$ of memory idle time [15] and idle uses 12% less power. Given these observations, completely eliminating refresh from DRAM can only save between $0.5\% - 1\%$ energy in state of the art technologies. This is DRAM energy, not system energy. Previous research shows that DRAM can consume up to 6% of power in a commodity system [13], up to 10% of power in a smart phone [1], and up to 33% of power in a server [8]. Previous approaches that must model DRAM, such as Flikker [12] which decrease refresh rate to reduce power consumption, assume that refresh occurs 100% of the time and that system memory uses much of the energy of a system. For our platform and for modern DRAM, these are not valid assumptions. When ported to our platform, Flikker saves approximately .1% of DRAM energy.

4 Conclusion

Manufacturers select a conservative refresh rate for DRAM to eliminate data corruption. Reducing the refresh rate can result in substantial DRAM power savings. Researchers have attempted to seize this opportunity by tuning voltage, adjusting refresh based on temperature, and partitioning DRAM data cells based on volatility.

We support such research by building an experimental platform and re-running the most cited experiments from previous papers on reducing DRAM power. Our work coalesces a disparate set of experiments onto a single platform. We show that this enables a deeper analysis of existing proposals and makes it possible to compare approaches.

Acknowledgments

This work was supported in part by C-FAR, one of six centers of STARnet, a Semiconductor Research Corporation program sponsored by MARCO and DARPA, NSF CSR-1218586, and NSF CNS-0845874. Any opinions, findings, conclusions, and recommendations expressed in these materials are those of the authors and do not necessarily reflect the views of the sponsors

References

[1] A. Carroll and G. Heiser. An analysis of power consumption in a smartphone. In *USENIX Annual Technical Conference*, USENIXATC, pages 21–21, Berkeley, CA, USA, 2010. USENIX Association.

[2] CellularRAM. 128Mb/64Mb Async/Page/Burst CellularRAM 1.5 Memory. 2005.

[3] H. David, C. Fallin, E. Gorbatov, U. R. Hanebutte, and O. Mutlu. Memory power management via dynamic voltage/frequency scaling. In *International Conference on Autonomic Computing*, ICAC, pages 31–40, New York, NY, USA, 2011. ACM.

[4] Q. Deng, D. Meisner, L. Ramos, T. F. Wenisch, and R. Bianchini. Memscale: active low-power modes for main memory. *ACM SIGPLAN Notices*, 2011.

[5] J. A. Halderman, S. D. Schoen, N. Heninger, W. Clarkson, W. Paul, J. A. Calandrino, A. J. Feldman, J. Appelbaum, and E. W. Felten. Lest we remember: cold-boot attacks on encryption keys. *Communications of the ACM*, 2009.

[6] T. Hamamoto, S. Sugiura, and S. Sawada. On the retention time distribution of dynamic random access memory (DRAM). *IEEE Transactions on Electron Devices*, 1998.

[7] J. Kim and M. C. Papaefthymiou. Block-based multi-period refresh for energy efficient dynamic memory. In *ASIC/SOC Conference, 2001. Proceedings. 14th Annual IEEE International*. IEEE, 2001.

[8] C. Lefurgy, K. Rajamani, F. Rawson, W. Felter, M. Kistler, and T. Keller. Energy management for commercial servers. *Computer*, 36(12):39–48, Dec 2003.

[9] C.-H. Lin, C.-L. Yang, and K.-J. King. PPT: joint performance/power/thermal management of DRAM memory for multi-core systems. In *ISLPED '09: Proceedings of the 14th ACM/IEEE international symposium on Low power electronics and design*, Aug. 2009.

[10] J. Liu, B. Jaiyen, R. Veras, and O. Mutlu. Raidr: Retention-aware intelligent dram refresh. In *International Symposium on Computer Architecture*, ISCA, pages 1–12, Washington, DC, USA, 2012. IEEE Computer Society.

[11] S. Liu, B. Leung, A. Neckar, S. O. Memik, G. Memik, and N. Hardavellas. Hardware/-software techniques for DRAM thermal management. *IEEE 17th International Symposium on High Performance Computer Architecture (HPCA)*, 2011.

[12] S. Liu, K. Pattabiraman, T. Moscibroda, and B. G. Zorn. Flikker: saving DRAM refresh-power through critical data partitioning. In *ASPLOS XVI: Proceedings of the sixteenth international conference on Architectural support for programming languages and operating systems*, June 2012.

[13] A. Mahesri and V. Vardhan. Power consumption breakdown on a modern laptop. In *International Conference on Power-Aware Computer Systems*, PACS, pages 165–180, Berlin, Heidelberg, 2005. Springer-Verlag.

[14] Micron Technology Inc. Mobile DRAM Power-Saving Features and Power Calculations. 2005.

[15] J. Mukundan, H. Hunter, K.-h. Kim, J. Stuecheli, and J. F. Martínez. Understanding and mitigating refresh overheads in high-density DDR4 DRAM systems. In *Proceedings of the 40th Annual International Symposium on Computer Architecture*. ACM, 2013.

[16] Panasonic Corporation. PhotoMOS HF 1 Form A.

[17] Samsung Electronics. KM41464A NMOS DRAM.

[18] Sun Electronic Systems, Inc. *Model EC1X Environmental Chamber User and Repair Manual*, 2011.

[19] Texas Instruments Inc. MSP430F241x, MSP430F261x Mixed Signal Microcontroller. In *Texas Instruments Application Report*, Jun. 2007, revised Nov. 2012.

[20] R. K. Venkatesan, S. Herr, and E. Rotenberg. Retention-aware placement in DRAM (RAPID): software methods for quasi-non-volatile DRAM. In *High-Performance Computer Architecture, 2006. The Twelfth International Symposium on*. IEEE, 2006.

[21] D. Wu, B. He, X. Tang, J. Xu, and M. Guo. RAMZzz: Rank-aware DRAM power management with dynamic migrations and demotions. In *SC '12: Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis*. IEEE Computer Society Press, Nov. 2012.