

A Platform to Analyze DDR3 DRAM's Power and Retention Time

Matthias Jung, Deepak M. Mathew, Carl C. Rheinländer, Christian Weis, and Norbert Wehn

University of Kaiserslautern

Editor's note:

The authors explore the intrinsic trade-off in a DRAM between the power consumption (due to refresh) and the reliability. Their unique platform allows tailoring to the design constraints depending on whether power consumption, performance or reliability has the highest design priority.

– Jörg Henkel, Karlsruhe Institute of Technology

■ **A DYNAMIC RANDOM ACCESS MEMORY** (DRAM) has to be refreshed regularly due to its charge-based bit storage property (capacitor). The retention time of a DRAM cell is defined as the amount of time that a cell can safely retain its data without being refreshed. It is a fact that 40% to 50% of power consumption in future 32- or 64-Gbit DRAMs has to be attributed to refresh commands [1]. Moreover, 3-D-integrated DRAMs such as Wide I/O or hybrid memory cube worsen the scenario with respect to increased cell leakage due to the much higher temperature in 3-D stacks. Therefore, the refresh frequency must be increased to avoid data corruption. In recent years, the paradigm of approximate computing has been extended to the DRAM subsystem, called *approximate DRAM* [2]. The underlying motivation of this approach is to reduce the increasing power consumption caused by unavoidable DRAM refresh commands by lowering the refresh frequency or even disabling the refresh completely while accepting the risk of data errors.

Digital Object Identifier 10.1109/MDAT.2017.2705144

Date of submission: 3 March 2016. Date of publication: 17 May 2017;

date of current version: 30 June 2017.

However, the amount of tolerable errors (resilience) strongly depends on the application. The degradation of the output quality of an application is directly related to the used DRAM device, the temperature, and the target application itself. Therefore, it is essential to truly investigate the DRAM

retention errors before deploying it as approximate memory in specific applications.

Furthermore, the analysis of the power consumption savings while using approximate DRAMs is additionally required. All of the aforementioned issues we address with DRAMMeasure, which is a low cost DRAM power consumption and retention time measurement platform that outperforms state-of-the-art measurement systems. It consists of a field-programmable gate array (FPGA)-board and two expansion printed circuit boards (PCBs), one as an adapter for the small outline dual inline memory module (SO-DIMM) slot and another for temperature control and current sensing.

With this measurement platform, we characterize the retention time behavior of different DRAM devices from several vendors. Similar to the timing and current (I_{DD}) values in DRAM's datasheets [3], we show that DRAM retention times specified in datasheets are very pessimistic. This is due to the high process margins added by the vendors to ensure correct functionality under worst-case conditions and a high-enough yield [3].

The refresh is initiated by the DRAM controller and is performed in a row-wise manner with the goal to refresh each cell every $t_{REF} = 64$ ms for a

temperature $T \leq 85^\circ\text{C}$ and $t_{\text{REF}} = 32\text{ms}$ for $T > 85^\circ\text{C}$. For example, if a DRAM bank has a number of rows $R = 8192$, then the time between two refreshes, called the *refresh interval*, is $t_{\text{REFI}} = \frac{t_{\text{REF}}}{R} = \frac{64\text{ ms}}{8192} \approx 7.8\text{ }\mu\text{s}$. The length of a refresh command is defined as t_{RFC} . In fact, the DRAM refresh intervals recommended by the vendors and the Joint Electron Device Engineering Council (JEDEC) (64 and 32 ms) adds a large guard band, as shown in Figure 1.

Approximate DRAMs such as [2], [4], and [5] exploit this fact by reducing the refresh rate or disabling the refresh completely in order to improve bandwidth and lower energy consumption. However, it is very difficult to characterize the weak cells in DRAMs, as they experience variable retention times (VRT) and data pattern dependencies (DPD) [6]. As shown in [7], mitigation techniques that rely on characterization are unable to ensure reliable operation even after many months of testing. Therefore, the authors of [8] try to overcome VRT issues by combining an online error-correcting code detection with row selective refresh.

However, for approximate DRAM it is key to conceive the DRAM as a stochastic model that includes both phenomena.

Figure 1 shows qualitatively the retention error behavior over time and the design space for approximate DRAM. The sphere around the curve represents the process variation, VRT, and DPD. In general, we have two key parameters for approximate DRAM: the lifetime of the data and the application robustness. Both of the parameters lead to three possibilities in the approximate DRAM design space.

- We can switch off refresh if the data lifetime in the DRAM is smaller than the actual required refresh period.
- We can switch off the refresh if the lifetime of the data is larger than

the required refresh period and the application provides resilience to the resulting number of errors at this working point.

- If the application only provides a certain robustness, we configure the refresh rate according to the resulting working point.

Studies like [5] have shown that many applications, such as communication, signal processing, and big data applications, which deal with noisy data, can tolerate the amount of errors induced by an approximate DRAM. It has been shown that even the complete refresh power could be saved for some applications.

To explore the previously presented design space, meaningful measurements for several temperatures and many devices from different vendors have to be conducted. Therefore, we run long-time retention time tests (1000s and more) that give us insights into the stability of the data in current devices. Moreover, we show the influence of data patterns on the bit error rate, caused by retention time failures, for different DRAM vendors.

Related Work

Kim and Lee presented a detailed study in 2009 on the data retention times of nanoscaled

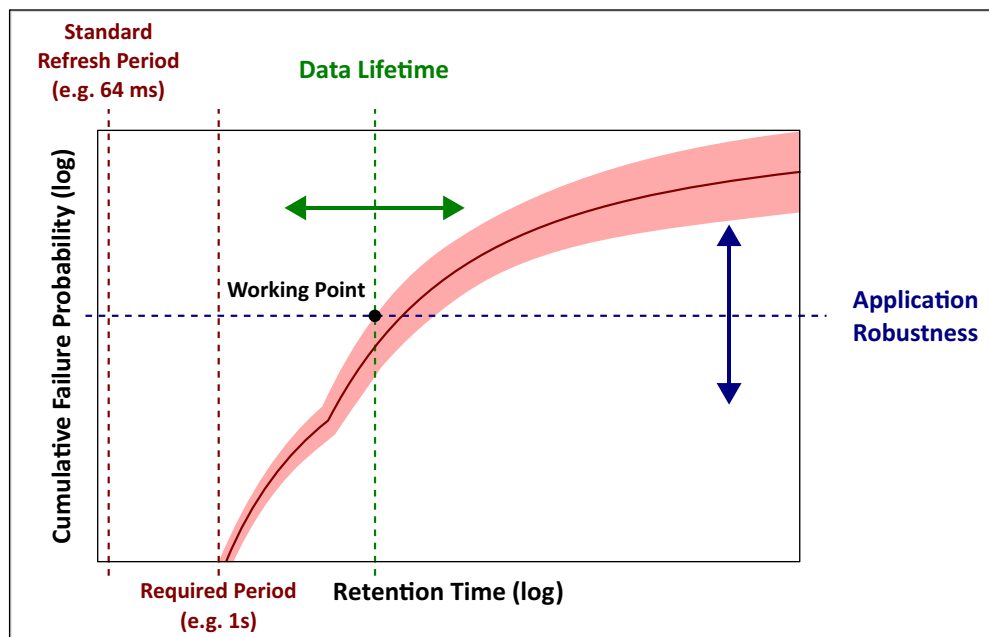


Figure 1. Qualitative retention error behavior—approximate DRAM design space.

DDR DRAMs [9]. Most of the following investigations, e.g., [1] and [10]–[12], are based on these results. However, in our previous measurements on Wide I/O DRAM, we observed that even at high temperatures the majority of cells in this device can hold data much longer than 10,000 s for a 0xFF... data pattern [2]. Therefore, we measured with DRAMMeasure a similar DDR3 DRAM Device (same vendor) from 2009 and confirmed that these presented numbers are way too pessimistic in the average case, as shown in Figure 2. In contrast to common belief, our measurements show that most of the DRAM cells can hold their value up to two orders of magnitude longer than assumed in [9]. Despite this, we confirm experimentally the scaling trends predicted by Kim and Lee [9] for recent nano-scaled DRAMs.

Liu et al. [6] analyze the data retention behavior with respect to temperature, DPD, and VRT. However, they measure the retention times only for several seconds (<10 s) and they use a thermal chamber, which has the following disadvantages:

- They are unable to raise the DRAM temperature higher than 70 °C.
- The fact that the entire measurement system is placed inside the thermal chamber might lead to unwanted effects to the system itself, which could influence the results.

- They use the DIMM's temperature sensor (TS), which is not mandatory for DDR3 DIMMs. Therefore, they are not able to measure DDR3 devices without a TS. Moreover, this sensor does not deliver the real chip temperature as it is not physically embedded into the DRAM chips. Thus, in the strict sense, an offset temperature must be estimated via thermal simulation.

In [13], the authors measure the number of faulty pages with respect to ambient temperature and time. They heat up the DRAM chip manually using a heat gun, which has several drawbacks:

- Stabilizing the temperature over several hours is impossible.
- Heating up all devices on an SO-DIMM to an evenly distributed temperature is infeasible.
- The results are not reliable, as the DRAM device can be damaged accidentally due to very hot spots (heat guns generate temperatures between 100 °C and 550 °C).

Rahmati et al. [14] also use a thermal chamber, but, moreover, they analyze a DRAM device from the 1990s. Therefore, their results have absolutely no relevance in today's DDR3/4 world. The authors of [15] measure only short retention times (<3 s) and only one bank of one single DDR3 DRAM. Therefore, their results do not consider process

variations. Moreover, they do not disclose details about their measurement setup, e.g., how they heat up the DRAM.

The Measuring Platform

We developed a custom platform¹ to measure the power consumption, retention errors, and to heat up the DRAM devices of DDR3 SO-DIMM modules.

The heating section consists of a mechanical

¹Further details of the measurement platform will be published in a detailed technical report at <http://www.uni-kl.de/3d-dram>.

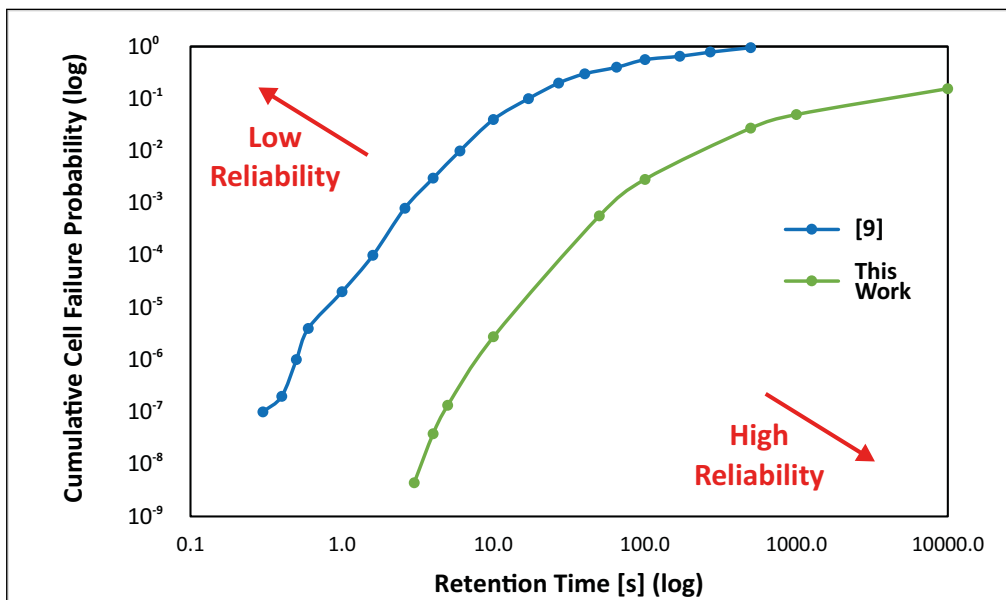


Figure 2. Comparison to [9]: single device measurements at room temperature.

setup, which is placed on the surface of the DRAM devices. A Peltier element (thermoelectric) is placed in the center, creating a hot and a cold side. The cold side is faced outward, equipped with aluminum cooling ribs. The hot side is faced toward the DRAM module and equipped with a temperature sensor for automatic heat control. To ensure fast and even distribution of the temperature among all covered DRAM devices, the hot side is made of a solid copper plane. Heat conducting foil is used to build a thermal bridge between the copper surface and the exterior surfaces of the DRAM packages on a DIMM.

For analyzing the current consumption of DRAMs, we designed an adapter PCB for DDR3 SO-DIMMs (six layers) that conforms to JEDEC standard requirements (Figure 3). The data, address, and control lines are passed through, and the power lines are routed across a precision 10 m Ω shunt resistor in order to measure the DRAM power consumption. A shunt monitor is used to amplify the low-voltage range of the shunt resistor.

The amplified voltage is forwarded to a computer-interfaced microcontroller board that also controls the power electronics for the heat control.

As user interface, a convenient custom PC software was designed for functional control, heat control, and current analysis. The temperature control provides a range of 25 °C to 90 °C. In advance, various thermal simulations have been conducted to construct the mechanical setup in a way to provide an accuracy of ± 2 °C among all covered devices. To guarantee reliable results the

system has been analyzed with calibrated precision measurement instruments by Keithley. The inspection record certifies a current measurement accuracy of ± 0.1 mA.

The DDR3 SO-DIMM adapter board is plugged into a Xilinx FPGA-based evaluation platform. The standard memory interface generator (MIG) memory controller from Xilinx is customized to generate the required command and data sequences for the measurements and to disable the refreshes. A virtual input/output (VIO) core connected to the custom MIG enables the real-time monitoring and control of the internal FPGA signals. Furthermore, we add the functionality to measure the error rates of single devices on an SO-DIMM by separating the data signals, which is important for the following measurements. The entire measurement setup is fully automated using the before mentioned software, which communicates to the FPGA board over the PCIe interface.

Experimental Results

By using the DRAMMeasure platform we run several analyses regarding DRAM retention time errors (bit flips) and DRAM power consumption (I_{DD} currents defined by JEDEC).

Retention Time Analysis

First, we present our measurement results with respect to retention time variations of ten identical 2-GB DDR3 SO-DIMMs, each based on four 4-Gbit DDR3 chips (devices) from Vendor-A. To obtain

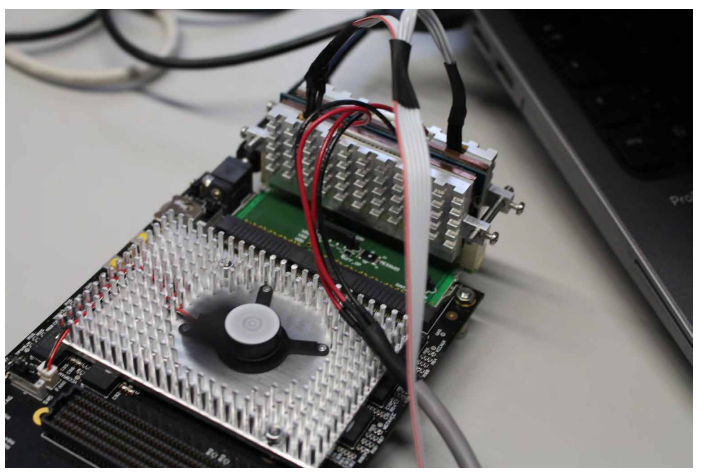
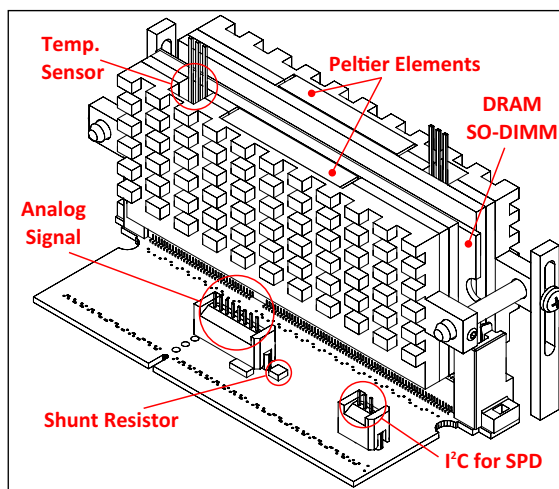


Figure 3. Mechanical construction and adapter PCB placed in a typical standard FPGA board.

the behavior of retention errors with respect to the device's process variation we measured the aforementioned total 40 Vendor-A devices. Each single device has been measured ten times at four different temperatures and five retention times, resulting in a total of 8000 measurement points, shown in Figure 4(a). We plot the retention times versus the cumulative cell failure probability obtained during each measurement step. The bars mark the minimum and the maximum measured number of errors, which represent the process variation depending on the temperature. We see here a quite prominent variation of the order of 20% (max. number of errors), which shows a large temperature dependency. This needs to be considered as a guardband in approximate computing platforms utilizing the approximate DRAM approach. Additionally, the figure shows a histogram of the absolute number of bit errors (between 1×10^6 and 4×10^6) measured at the data point with 100 s retention time and a temperature of 25 °C. Second, we revisit the analysis of how DRAM data is stored in actual devices. In principle, two flavors exist here:

- **True-cell** bit storage, where a logical one is always stored at bitline high voltage (e.g., 1.1 V) and a logical zero is stored at low voltage (0 V).
- **Anti-cell** bit storage, where a logical one or zero is stored either as bitline high (1.1 V) or low voltage (0 V) value depending on the address.

To extract the impact on the retention time of the two different storage methods we evaluate a device from Vendor-A (true-cells) and from Vendor-B (anti-cells) in order to compare the different behavior. Thus, we apply randomized patterns to the entire DRAM device containing a specific number of “1s” in the DRAM write burst. First, we generate a random bit string of length equal to the device's burst size, with a fixed number of “1s.” Second, the DRAM is written completely with the previous determined bit string. Third, we disable the refresh, keep the DRAM at room temperature, and read out the data after 100s. For each number of “1s” we repeat the above steps ten times with shuffled data patterns in order to obtain the graphs in Figure 4(b), which shows the average, minimum, and maximum number of errors measured in the experiment. The data width of Vendor-A is $\times 16$, which leads to a burst size of 128 bits (with a burstlength

of 8), whereas the Vendor-B device has a $\times 8$ organization and therefore a burst size of 64 bits, respectively. In this figure, we see the impact of the data pattern (DPD) on the retention time, while the device from Vendor-A (true-cells) shows increasing error counts for increasing number of “1s,” the device from Vendor-B (anti-cells) stays, besides some variations and a slight drop down at the end, nearly at the same level. This slight drop down can be explained in the following way: by sending a burst of 0x00... to the anti-cell device, it stores a 0x55... pattern in the memory array, while for a 0xFF... burst it stores a 0xAA... pattern. Thus, we see a difference due to DPD, which we will investigate in the future work.

Furthermore, we expected for the true-cell device a more linear behavior (Vendor-A). However, it is a more quadratic relationship between the number of “1s” and the number of bit errors. This can be explained by the observation that in the first half of the plot ($0 < \text{number of “1s”} < 64$) the number of “0s” dominates. This leads to a higher count of “1s” surrounded by mainly “0s” in the DRAM array. It has the consequence that in the first half of the graph the increase in bit errors is larger due to coupling effects during read-out. In the second half of the plot in Figure 4(b) the majority of storage data contains “1s.” Thus, the increase in bit errors saturates when the number of “1s” in a DRAM burst increases, due to reducing the coupling effects in the array. As can be seen by these results the impact of DPD has to be taken into account for approximate computing scenarios. Third, when looking into more details of retention time profiling, it is required to investigate the phenomenon of the so-called weak-cells and cells with VRT, which we also observed during our measurements. Many recent investigations try to identify cells with low retention time in order to decrease the refresh rate by excluding these cells from read/write operations. However, the effect of VRT imposes a large challenge to detect these cells. Thus, as mentioned before, the key for approximate DRAM is to conceive the DRAM device as a stochastic model that also includes process variation, VRT, and DPD.

Additionally, Kim and Lee investigated in [9] the further scaling down of DRAM and observed two overlapping distributions for a 30 nm DRAM technology. The main distribution of the retention time errors is due to the reverse pn-junction leakage, while

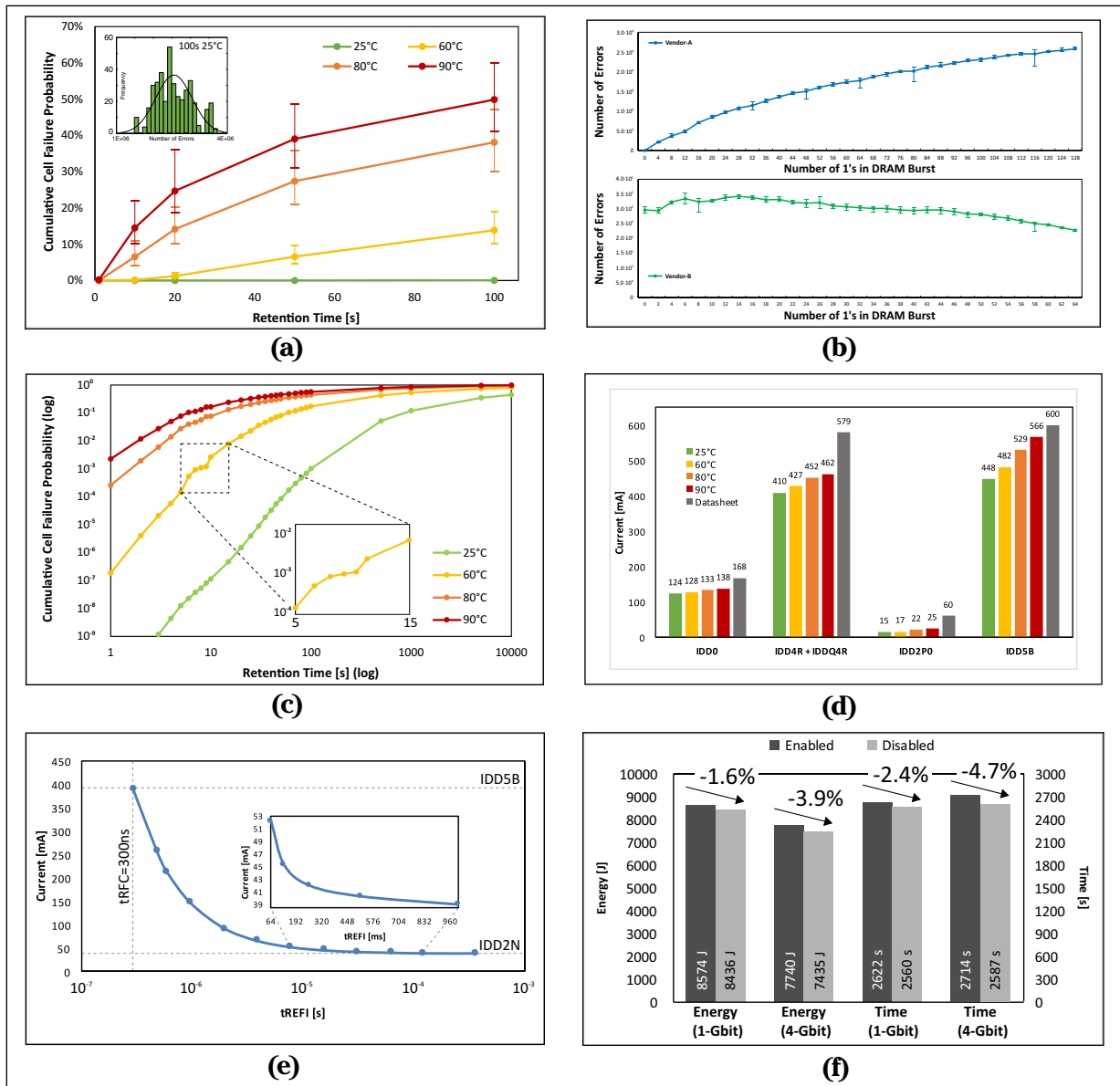


Figure 4. Measurements on DDR3 devices. (a) Process variation of 40 4-Gbit Vendor-A devices with a 0xFF... data pattern. (b) Random data pattern analysis for a single $\times 16$ device (true-cell) of Vendor-A and a single $\times 8$ device (anti-cell) of Vendor-B at 100 s and 25 °C. (c) Temperature dependence of the worst of the 40 measured 4-Gbit DRAM devices of Vendor-A with a 0xFF... data pattern. (d) Current consumption averaged over ten 4-GByte DDR3 Vendor-A SO-DIMMs. (e) Influence of refresh rate on currents for a single Vendor-A SO-DIMM with 1-Gbit devices. (f) Image processing application with enabled and disabled refresh for an SO-DIMM with 1-Gbit devices and an SO-DIMM with 4-Gbit devices from Vendor-B.

the tail distribution is due to trap-assisted tunneling drain-induced drain leakage (TAT-GIDL) based on permanent defects during high-temperature processing (high-power plasma etching, oxidation, and so on). We confirm by our experimental results plotted in Figure 4(c) the simulative prediction made by Kim

and Lee [9]. Especially for a temperature of 60 °C, we see a strong bend in the curve that is located in the range of 4 to 20 s. This finally shows that we are in the 30 nm range for the DRAM devices measured in Figure 4(c). Further scaling down of DRAM technology will potentially separate the two distributions

even more. We will investigate this in the future and we will publish the obtained results.

Current Consumption Measurement (IDDs)

As seen in the previous measurements, the retention times published by the DRAM vendors are very pessimistic. Similarly, this guardbanding in the range of five sigma (σ) also applies to the I_{DD} currents presented in datasheets. We ran standard I_{DD} test patterns (defined by JEDEC), such as I_{DD0} (activate/precharge cycling), I_{DD4R}/I_{DDQ4R} (continuous read), I_{DD2P0} (precharge power-down slow exit), and I_{DD5B} (refresh) to quantify the power consumption of recent DDR3 SO-DIMMs. Due to the limitations of the used FPGA boards, all the tests are executed with a frequency of 400 MHz (800 Mbit/s/pin data rate). We used the 6-6-6-18 ($t_{RCD} \cdot t_{CL} \cdot t_{RP} \cdot t_{RAS}$) setting given by JEDEC in clock cycles for this frequency. Each test is repeated several times to ensure stability and reproducibility with a temperature range from 25 °C to 90 °C. The results plotted in Figure 4(d) show an expected behavior. The values measured at the highest temperatures for I_{DD0} and I_{DD4R} are around 18% to 20% lower than the datasheet current values of the selected SO-DIMM. For the power-down current (I_{DD2P0}), which is leakage dominated, the difference is even larger and at 90 °C it still deviates by 60% from the value defined in the datasheet. In order to show how much refresh energy can be saved in the context of approximate DRAM we measured the refresh current under different retention times, as shown in Figure 4(e). When $t_{REFI} = t_{RFC}$ we measure the I_{DD5B} current. If we increase the refresh interval t_{REFI} , the measured current decreases and converges to I_{DD2N} , which is the precharge background current. Furthermore, we conducted experiments on an FPGA-based image processing application, where the DRAM is used as a framebuffer. The frame period is ≈ 8 ms, such that the refresh can be disabled safely according to the first possibility in the approximate DRAM design space. Figure 4(f) shows the measured savings in time and energy for this application, while processing approximately 600,000 frames on SO-DIMMs with 1 and 4-Gbit devices of Vendor-B. As shown in [1], the savings will further increase with growing DRAM device size.

WE PRESENTED DRAMMeasure, a versatile measurement platform to quantify today's DRAM's reliability with respect to temperature, its power consumption,

and process variation. To the best of our knowledge we confirmed, for the first time, with the measurement of recent DDR3 SO-DIMMs the TAT-GIDL effect of DRAMs predicted by [9]. Furthermore, we evaluated in detail the retention time behavior for different DRAM vendors. Approximate computing systems using approximate DRAMs will benefit from the accurate analysis results in terms of lower guardbanding and better prediction possibilities. In the future, we will present new contributions with respect to power modeling of DRAMs based on results of experiments conducted with DRAMMeasure. Furthermore, we have started to extend our platform to DDR4 and DDR4 SO-DIMMs implementations to derive the latest trends of DRAM scaling. ■

Acknowledgment

We thank Martin Schultheis and Vitor Kielsing for their support. This work was funded in part by the German Research Foundation (DFG) as part of the priority program Dependable Embedded Systems SPP 1500 (<http://spp1500.itec.kit.edu>), the DFG Grant WE2442/10-1, and Carl-Zeiss Stiftung.

References

- [1] J. Liu et al., "RAIDR: Retention-aware intelligent DRAM refresh," in *Proc. 39th Ann. Int. Symp. Comp. Archit., ISCA '12*, Washington, DC, 2012, pp. 1–12; <http://dl.acm.org/citation.cfm?id=2337159.2337161>.
- [2] M. Jung et al., "Efficient reliability management in SoCs—An approximate DRAM perspective," in *21st Asia and South Pacific Design Automation Conf. (Asp-Dac)*, 2016.
- [3] K. Chandrasekar et al., "Towards variation-aware system-level power estimation of DRAMs: An empirical approach," in *Proc. 50th Design Automation Conf.*, Austin, TX, 2013.
- [4] S. Liu et al., "Flikker: Saving DRAM refresh-power through critical data partitioning," *SIGPLAN Not.*, ACM, New York, USA, 2011, pp. 213–24.
- [5] M. Jung et al., "Omitting refresh—A case study for commodity and Wide I/O DRAMs," in *1st Int. Symp. Memory Syst. (Memsys 2015)*, Washington, DC, USA, 2015.
- [6] J. Liu et al., "An experimental study of data retention behavior in modern DRAM devices: Implications for retention time profiling mechanisms," *SIGARCH Comput. Archit. News*, ACM, New York, 2013, pp. 60–71.
- [7] S. Khan et al., "The efficacy of error mitigation techniques for DRAM retention failures: A comparative

- experimental study,” in *2014 ACM Int. Conf. Meas. Model. Comp. Syst.*, SIGMETRICS '14, ACM, New York, USA, 2014, pp. 519–32, doi: 10.1145/2591971.2592000.
- [8] M.K. Qureshi et al., “AVATAR: A variable-retention-time (VRT) aware refresh for DRAM systems,” *Memory*, vol. 2, p. 20, June 2015.
 - [9] K. Kim and J. Lee, “A new investigation of data retention time in truly nanoscaled DRAMs,” *IEEE Electron Device Letters*, vol. 30, no. 8, pp. 846–848, 2009.
 - [10] P.J. Nair, D.-H. Kim, and M.K. Qureshi, “ArchShield: Architectural framework for assisting DRAM scaling by tolerating high error rates,” in *proc. 40th Ann. Int. Symp. Comp. Archit.*, ISCA '13, ACM, New York, USA, 2013, pp. 72–83, doi: 10.1145/2485922.2485929.
 - [11] J. Lucas et al., “Sparkk: Quality-scalable approximate storage in DRAM,” in *Memory Forum*, 2014; www.redaktion.tu-berlin.de/fileadmin/fg196/publication/sparkk2014.pdf.
 - [12] C.-H. Lin et al., “SECRET: Selective error correction for refresh energy reduction in DRAMs,” in *2012 IEEE 30th Int. Conf. Comp. Design (ICCD)*, 2012, 67–74, doi: 10.1109/ICCD.2012.6378619.
 - [13] R.K. Venkatesan, S. Herr, and E. Rotenberg, “Retention-aware placement in DRAM (RAPID): software methods for quasi-non-volatile DRAM,” in *Proc. of HPCA*, 2006.
 - [14] A. Rahmati et al., “Refreshing thoughts on DRAM: Power saving vs. data integrity,” in *Workshop Approx. Comput. Syst. Stack (Wacas)*, 2014.
 - [15] C.-S. Hou et al., “An FPGA-based test platform for analyzing data retention time distribution of DRAMs,” in *2013 Int. Symp. VLSI Des. Autom. Test (VLSI-DAT)*, 2013, pp. 1–4, doi:10.1109/VLDI-DAT.2013.6533853.

Matthias Jung received a PhD degree in electrical engineering from the University of Kaiserslautern, Germany, in 2017. His research interests are SystemC-based virtual prototypes, especially with a focus on the modeling of memory systems. He is a Member of the IEEE and a member of the Association for Computing Machinery. Contact him at jungma@eit.uni-kl.de.

Deepak M. Mathew is currently pursuing a PhD degree with the Microelectronic Systems Design Research Group, University of Kaiserslautern, Germany. His research interests are heterogeneous memory systems, memory controller hardware design, and emerging nonvolatile memory technologies. He is a Member of the IEEE. Contact him at deepak@eit.uni-kl.de.

Carl C. Rheinländer is currently pursuing a PhD degree with the Microelectronic Systems Design Research Group, University of Kaiserslautern, Germany. His research interests are embedded ultralow power hardware designs for distributed interconnected systems. He is a Member of the IEEE. Contact him at rheinlaender@eit.uni-kl.de.

Christian Weis has been with the Microelectronic System Design Research Group, University of Kaiserslautern, Germany, since 2009. He received a PhD degree in electrical engineering from the University of Kaiserslautern, Germany, in 2014. His current research interests include 3-D integrated DRAMs, DRAM controller design, and heterogeneous memories and MPSoCs. He is a Member of the IEEE. Contact him at weis@eit.uni-kl.de.

Norbert Wehn holds the chair for Microelectronic System Design, Department of Electrical and Computer Engineering. His research interests are VLSI-architectures for mobile communication, forward error correction techniques, advanced SoC architectures and methodologies, 3-D integration, reliability issues in SoC, and high performance computing on embedded devices. He is an IEEE Senior Member and a member of the Association for Computing Machinery. Contact him at wehn@eit.uni-kl.de.

■ Direct questions and comments about this article to Matthias Jung, Microelectronic Systems Design Research Group, University of Kaiserslautern, Erwin-Schrödinger-Straße 12, D-67663 Kaiserslautern, Germany; e-mail: jungma@eit.uni-kl.de.