JEDEC PUBLICATION

SYSTEM LEVEL ROWHAMMER MITIGATION

JEP301-1

MARCH 2021

JEDEC SOLID STATE TECHNOLOGY ASSOCIATION

NOTICE

JEDEC standards and publications contain material that has been prepared, reviewed, and approved through the JEDEC Board of Directors level and subsequently reviewed and approved by the JEDEC legal counsel.

JEDEC standards and publications are designed to serve the public interest through eliminating misunderstandings between manufacturers and purchasers, facilitating interchangeability and improvement of products, and assisting the purchaser in selecting and obtaining with minimum delay the proper product for use by those other than JEDEC members, whether the standard is to be used either domestically or internationally.

JEDEC standards and publications are adopted without regard to whether or not their adoption may involve patents or articles, materials, or processes. By such action JEDEC does not assume any liability to any patent owner, nor does it assume any obligation whatever to parties adopting the JEDEC standards or publications.

The information included in JEDEC standards and publications represents a sound approach to product specification and application, principally from the solid state device manufacturer viewpoint. Within the JEDEC organization there are procedures whereby a JEDEC standard or publication may be further processed and ultimately become an ANSI standard.

No claims to be in conformance with this standard may be made unless all requirements stated in the standard are met.

Inquiries, comments, and suggestions relative to the content of this JEDEC standard or publication should be addressed to JEDEC at the address below, or refer to www.jedec.org under Standards and Documents for alternative contact information.

Published by
©JEDEC Solid State Technology Association 2021
3103 North 10th Street
Suite 240 South
Arlington, VA 22201-2108

PRICE: Contact JEDEC

Printed in the U.S.A.

SYSTEM LEVEL ROWHAMMER MITIGATION

(From JEDEC Board Ballot JCB-21-06, formulated under the cognizance of the JC-42 Committee on Solid State Memories. Item 1866.02.)

1 Scope

A DRAM rowhammer security exploit is a serious threat to cloud service providers, data centers, laptops, smart phones, self-driving cars and IoT devices. Hardware research and development will take time. DRAM components, DRAM DIMMs, System-on-chip (SoC), chipsets and system products have their own design cycle time and overall life time.

1) Rowhammer vulnerability is a fundamental DRAM issue.
2) RFM is designed to alleviate rowhammer concerns, but cannot eliminate vulnerability to all possible forms of attacks.
3) System companies need the maximum amount of flexibility in monitoring rowhammer and deploying the appropriate mitigation measures (HW & SW):
    a) MAC (Maximum Active Count) and blast radius. Also known as DRAM victim row vulnerability to aggressor row activation count.
    b) Comprehensive ECC, BIST, BISR statistics
    c) The ability for the OS to report an aggressor to DRAM and ask DRAM to take mitigation actions
    d) The ability for DRAM to ask for pause (no new commands can be issued by SoC) if needed. It can be non-deterministic (Ready or Wait)

This publication recommends the following best practices to mitigate the security risks from rowhammer attacks.

1) At the product level (e.g., cloud, smart phones), each company has its reliability (bit error rate) and security (privilege escalation or denial of service) requirements. The following guidelines are recommended:
    a) Companies share rowhammer test patterns (e.g., Github[1]) and tools (e.g., Github[2]) with each other. Security is a shared industry concern, not an arena for competitive advantage.
    b) Companies invest in memory validation and a quality assurance process. The bit error rate metrics should be quantified clearly for the suppliers (e.g., DRAM package and DIMMs).

1 Scope (cont'd)

2) At the Operating System (OS) and FW level, add reliability, availability and serviceability (RAS) and security features to the most critical instruction and data structure. The following guidelines are recommended:
   a) A DRAM health monitor that takes as many signals as possible from SoC. The signals can be ECC error statistics, ECC logs, MBIST statistics, TRR tracking circuits, memory controller performance monitors. Heuristic-based DRAM health monitors can potentially detect a large-scale rowhammer attack.
   b) Isolate the user program from the most critical data structures (e.g., Page Table Entry) in the OS kernel. Protect the kernel in a fixed address space with memory encryption and integrity check (e.g., Hash-based Message Authentication Code). For example, carve out a 256MB enclave region in 8GB DRAM addressable space for the kernel. Some SoC vendors provide integrity for the enclave region.
3) At the SoC and memory controller level, invest in hardware circuits that improve RAS and security. The following guidelines are recommended:
   a) Built-in-self-test (BIST) and built-in-self-repair (BISR) engines for DRAM. Not all DRAMs have MBIST (e.g., LPDDR4X).
   b) Post-package-repair (PPR) support in the field. If the repair resource is exhausted, provide the bad DRAM address to the OS so that it can be taken offline.
   c) TRR tracking circuits to complement DRAM's mitigation scheme. SoC and memory controller to track the aggressor address. DRAM to figure out the victim address and the time needed to adequately perform the mitigation.
   d) System-level ECC. Detection is more important than correction. On-die SEC ECC is not good enough. 3-4 bit flips are observed in 128-bit data. The existing ECS/patrol mechanism (e.g., 24 hour) is not responsive enough.
   e) Additional buffer memory hierarchy to shield the main memory from rowhammer attack. Conceptually, it is a hardware managed cache. It will be harder (but still possible) for the hackers to instrument a rowhammer attack without cache flush support.
   f) High-performance memory encryption (ME) and Hash-based Message Authentication Code (HMAC) circuits.

2 Terms, Definitions, and Acronyms

ACT Command: The Activate command opens a row for a subsequent Read or Write. The Refresh command also activates a row, but it is aimed to refresh its content (a maintenance operation). It can cause a hammer effect to its neighbors as well.

Aggressor Row(s): A row or rows that receive excessive activations during a Rowhammer attack in an attempt to disturb data in nearby DRAM cells.

ATE (Automated Test Equipment): Generally reserved for component level tests before that component is made part of a larger system or assembly. ATEs are used for DRAM component and DIMM testing.

ECS (Error Check Scrub): A built-in function in DDR5 to track DRAM chip's error status and fix soft errors by writing back the corrected data. A system can also perform reads and write back corrected data.

Hammer Count (RowHammer Characterization ETH [5]**)**: DRAM's intrinsic rowhammer tolerance without any mitigation. If an aggressor row is activated (hammered) more than the hammer count, the cell contents of a victim row(s) can be lost. It can also be described as DRAM victim row vulnerability to aggressor row activation count.

MAC (Maximum Activation Count): The maximum number of activates that a single row can sustain within a refresh period before the adjacent rows need to be refreshed.

MBIST (Memory Built in Self Test): Test that can be implemented in the SoC or memory chip, DIMM.

PPR (Post Package Repair): A process to remove a ROW within the DRAM and replace it with a spare one.

RAA (Rolling Accumulated Activate count for RFM): Incremented and decremented by the Memory Controller on a per bank basis. Incremented by 1 when an ACT Command is issued and decremented by a REF/RFM command. The amount to decrement is specified by the RAA Counter Decrement which is also specified by the DRAM vendor.

RAADEC (RAA Counter Decrement): The decrement amount for RAA per REF or RFM command.

NOTE Specified by the DRAM vendor

RAAIMT (Rolling Accumulated Activate Initial Management Threshold): When the RAA reaches this value additional Refresh or RFM commands are needed.

NOTE Specified by the DRAM vendor

RAAMMT (Refresh Accumulated Activate Maximum Management Threshold): A value that the RAA can never exceed (per bank). If the RAA reaches the RAAMMT no more ACT commands should be issued by the Memory Controller to that bank until one or more REF or RFM commands have been issued such that the RAA drops below the RAAMMT.

Refresh postpone and pull-in: To allow for improved efficiency in scheduling, DRAM Refresh command can postpone and pull-in the refresh commands.

2 Terms, Definitions, and Acronyms (cont'd)

SDRAM/DRAM (Dynamic Random Access Memory): For the purpose of this document they are the same. The DDR DRAMs are synchronous.

System Companies: Companies that sell the complete integrated system or board vendors that sell a board containing an SoC/FPGA and DDR Memory.

tREFI: Average refresh Interval timing parameter in the JEDEC specification. 1x refresh tREFI is typically 3.9 µs in LPDDR4/DDR5/LPDDR5 DRAM. 1x refresh tREFI is 7.8 µs in DDR4 DRAM.

tREFW: Refresh window timing parameter in the JEDEC LPDDR (not in DDR4/5 DRAM) specification. 8192 refresh commands are required in tREFW. 1x refresh tREFW is typically 32 ms in LPDDR4 DRAM.

Victim Row(s): A row or rows with cells that are affected by Rowhammer activity.

RSA (**Rivest–Shamir–Adleman):** A public-key cryptosystem that is widely used for secure data transmission.

Sudo: A program for Unix-like computer operating systems that allows users to run programs with the security privileges of another user, by default the superuser.

2.1 Acroynms

AES, XTS: Types of encryption modes.

BIST: Built in Self Test.

BISR: Built in Self Repair.

ECC: Error Correction Code.

DED: Double Error Detection.

DPPM: Defects in Parts Per Million.

FW: Firmware. (Often BIOS or other boot related code that is stored in non-volatile memory.)

HMAC: Hash-based Message Authentication Code.

ME: Memory Encryption.

OS: Operating System.

PARA: Probabilistic Adjacent Row Activation.

PTE: Page Table Entry.

RFM: Refresh Management.

SEC: Single Error Correction.

SoC: System on a Chip.

TRR: Targeted Row Refresh.

RAS: Reliability, Availability and Serviceability.

3 System-level Mitigation Schemes

This publication provides additional details in each system-level mitigation scheme. These are recommendations, not specifications. it is up to each company to choose what to implement and to what degree.

1) At the product level, there are different grades or classes. For example, commercial, industrial, automotive and military. It is expected that a military grade product will require higher rowhammer immunity than a commercial grade product. A clear metric should be defined by the system companies. Rowhammer induced bit flips cause a transient failure like soft errors. For hard defects, it is recommended to use DPPM. For soft errors, it is recommended to use bit error rate. The rowhammer test program can have single-sided, double-sided, many-sided (Figure 12 in TRRespass [3]) and random fuzzer patterns. Rowhammer pattern searches by trial-and-error will be a trend in the future given the cloud computing scale.
    a) Maximum Activate Count (MAC) was a parameter defined in prior LPDDR4 TRR specification. In this publication, Hammer count is used to describe the intrinsic rowhammer tolerance without any mitigation scheme (e.g., PARA, ECC, TRR). The system companies can benchmark the hammer count to meet its product's reliability and security requirements. The researchers provide some useful examples to characterize the intrinsic hammer count with refresh control (Figure 12 in Microsoft [4], Algorithm1 in ETH [5]). DRAM may suffer from retention failures when the refresh is dynamically controlled (not within the specification). For example, 80K is the minimum rowhammer tolerance observed in the LPDDR4X hammer count characterization (conducted in March 2020).
    b) Rowhammer is a widespread DRAM issue. Most advanced node DRAMs are susceptible to bit flips from rowhammer (Figure 4 in ETH [5]). The difference is the effectiveness of DRAM's mitigation schemes. The rowhammer bit error rate is defined as error occurrences over device-time. For the known rowhammer test patterns (e.g., single-sided, double-sided), it may be a reasonable goal to have 1 bit flip per week per device (e.g., 8Gb 512Mx16 DDR4 chip). There can be waivers for new rowhammer test patterns (e.g., Github [1]). It is up to each company's QA (quality assurance) team to work with their DRAM suppliers on the root cause analysis and corrective action. ATE will be more efficient than the system-level test when conducting rowhammer tests at the DRAM level.
    c) If the SoC and memory controller companies choose to implement any of the suggested mitigation schemes, the SoC-level bit error rate is expected to improve. For the known rowhammer test patterns (e.g., single-sided, double-side), it may be a reasonable goal to have 1 bit flip per month per socket (e.g., 8 fully populated 16Gb 1Gx16 DDR4 chips). Similarly, the waivers may be applied. Bit error rate heavily depends on the test patterns and mitigation circuits. For a deterministic (e.g., TWiCE [6]) mitigation scheme, a test pattern that targets the known vulnerability can have a very high bit flip rate. For a probabilistic (e.g., PARA in CME [7]) mitigation scheme, a test program that exploits the vulnerability can have a very low bit flip rate in the beginning. Gradually, the test program can learn to be more effective.

3 System-level Mitigation Schemes (cont'd)

d) If the system companies choose to implement any of the kernel mitigation schemes, the product-level bit error rate is expected to improve. For the known security exploits (e.g., PTE, sudo, RSA; Table IV in TRRespass [3]), the time to exploit varies. The product requirement also varies on the number of privilege escalation, crash or denial of service per system over time (e.g., year). A security breach is more severe than denial of service or system downtime.

2) The OS kernel, system FW/BIOS and CPU/SoC should have a coordinated rowhammer mitigation scheme. Ideally, the operating system monitors DRAM's health and maintains DRAM's error statistics, logs and repair information even after the system is powered down. It should sample many signals and counters from the SoC to assess if the DRAM needs diagnostic service or if the DRAM is under rowhammer attack. If the DRAM is under rowhammer attack, the system may identify the offending user process, and apply the available rowhammer mitigation or counter measures. There can be different levels of severity and each mitigation has different levels of effectiveness and its associated performance or power overhead. A detailed security review will be required to prevent new side-channel vulnerabilities.

   a) The SoC and memory controller should provide CPU performance monitors, memory event counters and status registers (e.g., LLC cache miss; page conflict; rowhammer aggressor tracking; uncached memory access, cache flush instruction counters). There may not be definitive rowhammer error conditions, but combinations of forensic evidence can add up to make a plausible detection. ECC correctable DRAM bit flips are a first warning to the system of potential problems. Predicting rowhammer attacks without observing any DRAM bit flip is an idealized goal for future mitigation schemes.

   b) A system engineer usually needs to use an interposer to probe the DRAM CA bus during debug. An on-chip embedded DRAM CA bus trace analyzer through an external port (e.g., USB-C) to a monitoring system or analyzer can enable streaming debug visibility in the field without modifying the form factor board.

   c) Increasing the refresh rate is a common mitigation scheme and it is often a BIOS (FW) setup (versus dynamically enabled or configured at run-time). 2x refresh has certain implications to DRAM's power (e.g., 10%) and system's performance (e.g., 1%). While the increased refresh rate may help reduce the rate of bit flips, it will not eliminate the issue completely and the increased power and reduced system performance may be excessive relative to the potential gain. Ability to configure refresh behavior outside JEDEC specification (both ways - more and fewer refreshes) to enable comprehensive testing and to have more room for mitigations. Out of specification refresh settings are useful for experimentation or exploration, not for production.

   d) Identifying the offending user process that issues the rowhammer attacks (aggressor's address) will be useful for the OS. During the memory templating stage (VIII. Exploitation in TRRespass [3]) of the rowhammer attack, the SoC is expected to encounter more ECC errors than under typical workload. If the system implements a DRAM health monitor and SoC provides the system-level ECC. The scheduler can slow down the offending process or prompt the user/administrator to kill the process.

3 System-level Mitigation Schemes (cont'd)

e) A large system-level cache in front of main memory may offer rowhammer mitigation benefits in addition to its original performance and power enhancement. There are a few things to consider. The system-level cache or buffer should not be vulnerable to rowhammer attack itself. It is managed by the hardware and shields the main memory from direct user program access. It makes the rowhammer attack more difficult (another layer of indirection), but not impossible.

f) There are always certain weak (low hammer count tolerance) bits in the DRAM chip that are easier to flip than the rest. A malicious user program needs to control a few dozens of these bits to launch a successful attack. With a DRAM health monitor, it will be possible to use MBIST to diagnose the bits vulnerable to rowhammer attacks and apply PPR to repair those rows or mask off those DRAM addresses when the repair resource is exhausted. MBIST engine designs should be flexible to adopt new rowhammer test patterns.

g) The system-level ECC is different from DRAM's on-die ECC (e.g., SEC code to improve the DRAM reliability). It is the system companies' decision (cost-performance tradeoff) to choose the ECC code location. It is possible to use existing DRAM's capacity to store the check/parity bits, or to use a dedicated DRAM chip for the check/parity bits. It is also up to the system and SoC companies to collaborate and define the ECC algorithm's capability. For example, detecting multi-bit uncorrectable errors is more important than correcting a single-bit error. Silent data corruption is not acceptable.

h) Memory encryption (ME) is a known technique (e.g., AES) to protect the plain text. Hash-based message authentication code (HMAC) adds a tag to ensure the message or DRAM content has not been modified since it was written. These are potential hardware techniques against rowhammer attack. The system with ME and HMAC may alleviate the attack from privilege escalation exploit to denial of service, however, they require more computational logic and more data bits than ECC logic. It is up to the system and SoC companies to collaborate and define the requirement. For example, is it feasible to limit the kernel's critical data structure in a fixed address space for memory encryption and HMAC? What kind of encryption mode (e.g., AES or XTS) is useful? What kind of strength of the hash function (e.g., SHA-2, SHA-3) is needed?

i) While there will be additional cost, performance and power overhead to implement the above features at the system level, security can be a product differentiator based on market's demand and user's requirement.

4 References

NOTE JEDEC does not endorse these reference documents. References are limited to the specific patterns, figures, algorithms, and mitigation schemes identified in the text of this document that includes these references.

[1] https://github.com/vusec/trrespass

[2] https://github.com/google/hammer-kit

[3] TRRespass: Exploiting the Many Sides of Target Row Refresh. Frigo, P.; Vannacci, E.; Hassan, H.; van der Veen, V.; Mutlu, O.; Giuffrida, C.; Bos, H.; and Razavi, K. In *S&P*, May 2020.

[4] Are We Susceptible to Rowhammer? An End-to-End Methodology for Cloud Providers. Cojocar, L. , Kim J., Patel, M., Tsai, L., Saroiu, S., Wolman, A., and Mutlu, O in S&P, 2020.

[5] Revisiting rowhammer: An experimental analysis of modern DRAM devices and mitigation techniques. Kim, J., Patel, M., Yağlıkçı, A.G., Hassan, H., Azizi R., Orosa, L., and Mutlu, O. In International Symposium on Computer Architecture, 2020.

[6] TWiCe: Preventing Row-Hammering by Exploiting Time Window Counters. Lee, E., Lee S., Suh, G., and Ahn, J. in ISCA, 2019

[7] Flipping Bits in Memory Without Accessing Them: An Experimental Study of DRAM Disturbance Errors. Kim, Y., Daly, R., Kim, J., Fallin, C., Lee, J., Lee, D.,Wilkerson, C., Lai, K., and Mutlu, O. In ISCA, 2014.

Standard Improvement Form JEDEC _____

The purpose of this form is to provide the Technical Committees of JEDEC with input from the industry regarding usage of the subject standard. Individuals or companies are invited to submit comments to JEDEC. All comments will be collected and dispersed to the appropriate committee(s).

If you can provide input, please complete this form and return to:

JEDEC
Attn: Publications Department
3103 North 10th Street
Suite 240 South
Arlington, VA 22201-2107

Fax: 703.907.7583

1. I recommend changes to the following:

☐ Requirement, clause number _____

☐ Test method number _____ Clause number _____

The referenced clause number has proven to be:

☐ Unclear ☐ Too Rigid ☐ In Error

☐ Other _____

2. Recommendations for correction:

_____

_____

_____

_____

3. Other suggestions for document improvement:

_____

_____

_____

_____

Submitted by

Name: _____ Phone: _____

Company: _____ E-mail: _____

Address: _____

City/State/Zip: _____ Date: _____


JEDEC®