HBM3 RAS
The Journey to Enhancing Die-Stacked DRAM Resilience at Scale

Sudhanva Gurumurthi
RAS Architecture Group

PUBLIC
Acknowledgments

Vilas Sridharan
Aaron Nygren
Roy Chung
Yesin Ryu
Munseon Jang
Taekyun Kim
Kijun Lee
Kyomin Sohn
Eric Scott
HBM3 Overview

- HBM3 is the next-generation of the JEDEC High-Bandwidth Memory DRAM standard

- HBM3 is expected to be used in future SoCs, at scale, to accelerate data center and automotive workloads

- DRAM resilience is important at scale.
Talk Outline

• Introduction

• DRAM Reliability Overview

• The Case for a Strong HBM3 ECC

• The Road to a Practical HBM3 ECC

• Conclusions
Increased Heterogeneity

Tight Integration of Compute and Memory

Larger Systems

Enabled by Industry Standards
## Evolution of HPC System Configurations

<table>
<thead>
<tr>
<th>System</th>
<th>Jaguar</th>
<th>Titan</th>
<th>Summit</th>
<th>Frontier</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cores / CPU</td>
<td>6</td>
<td>16</td>
<td>22</td>
<td>64</td>
</tr>
<tr>
<td>CUs / GPU</td>
<td>0</td>
<td>14</td>
<td>80</td>
<td>220</td>
</tr>
<tr>
<td>CPUs / node</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>GPUs / node</td>
<td>0</td>
<td>1</td>
<td>6</td>
<td>4</td>
</tr>
<tr>
<td>Memory</td>
<td>DDR2</td>
<td>DDR3</td>
<td>DDR4</td>
<td>DDR4</td>
</tr>
<tr>
<td>Memory / CPU</td>
<td>8 GB</td>
<td>32 GB</td>
<td>512 GB</td>
<td>512 GB</td>
</tr>
<tr>
<td>Memory / GPU</td>
<td>n/a</td>
<td>6 GB</td>
<td>96 GB</td>
<td>128 GB</td>
</tr>
<tr>
<td>Node count</td>
<td>18,688</td>
<td>18,688</td>
<td>4,608</td>
<td>9,408</td>
</tr>
<tr>
<td>Year</td>
<td>2009</td>
<td>2012</td>
<td>2018</td>
<td>2022</td>
</tr>
</tbody>
</table>

The Need for Improved Resiliency

• There are many sources of non-idealities in silicon:
  • Particle induced transient faults
  • Permanent faults
  • Variable retention time faults
  • Lifetime reliability

• Errors that arise from these faults can disrupt normal execution

• Frequency of errors increase with scale
Resilience = R + A + S

- **Reliability**: The ability to provide correct service

- **Availability**: The ability to be ready to provide correct service

- **Serviceability**: The ability to diagnose and repair faults
Faults vs. Errors

- **Fault**: A condition that causes the inability to meet the specifications expected
- A fault may be transient, permanent, or intermittent
- **Error**: The manifestation of a fault
- From a hardware design perspective faults are more important to understand
Fault and Error Management in Servers

- **Fault management**: Pinpoint where the faults occur in hardware and take appropriate action.

- **Error Management**: Contain the impact of uncorrected errors by terminating or recovering from those errors at the software level.

- Both error and fault management rely on **transparency** of hardware error events.
DRAM Reliability Overview
Potential Fault Boundaries in a DRAM Device

- Bank Faults
- Single-Bit faults
- Single-Column faults
- Row Faults
Approximate Breakdown of DRAM Device Faults
[Sridharan et al., SC’12] [Sridharan et al., SC’13] [Sridharan et al., ASPLOS’15]

All DRAM Generations Experience Multi-Bit Faults
Chipkill ECC in DDRx DRAM

With Chipkill, any fault within a single DRAM device is correctable
DDR-style Chipkill is Inefficient in the HBM Architecture

• A cache line access is done from a single bank in a single layer of the HBM stack

• Chipkill and other “RAID-style” schemes require distributing cache line bits across multiple dies or channels

• This is impractical from power and performance viewpoints
The Case for a Strong HBM3 ECC
BASELINE: Single-Error Correction HBM2 ECC

[Gurumurthi et al., IEEE CAL’21]

- The HBM2 specification provides 32 check bits per 256-bit access for a host ECC
- Modeled the uncorrected memory error rate at the system level for an HPC-like node configuration
- DRAM field data used to for reliability modeling

**This node configuration is not intended to model any AMD product(s).**

ECC improvements can enhance resilience at scale
Emergence of DRAM Scaling Faults

• Variable Retention Time (VRT) faults
  • DRAM cell retention can change randomly from low- to high-retention states
  • Increasingly challenging to screen out weak cells during DRAM manufacturing
  • Variable-rate refresh schemes could help (e.g., “AVATAR” [Qureshi at al., DSN’15]), but are challenging for practical adoption

• Write recovery time (tWR) faults
  • After a write command is issued to DRAM, one has to wait for a certain amount of time for the write to be successful in the DRAM cells (tWR timing requirement)
  • tWR is increasing with DRAM technology scaling
  • Increasing parallelism within the DRAM die partly offsets this problem

• The memory industry’s solution to DRAM scaling faults is to use In-DRAM ECC (ID-ECC)
  • Example: DDR5 devices use a Single-Error Correct (SEC) In-DRAM ECC
Factoring In the Impact of DRAM Scaling Faults

• The original glidepath for HBM3 was to use a SEC ID-ECC, which requires 16 additional bits

• In the absence of DRAM scaling faults, Single-Bit and Single-Column faults are correctable with SEC ECC. Other multi-bit faults are uncorrectable

• Overlap between a correctable fault mode and a scaling fault will lead to an uncorrectable error with an SEC code
Impact of DRAM Scaling Faults

32K weak cells => 0.00019% of total bits of a 16Gb DRAM die.

A Stronger ECC in HBM3 can narrow this gap
The Road to a Practical HBM3 ECC
Original Glidepath for HBM3

SEC ID-ECC
Unused check bits for Host ECC
Host ECC
Host Memory Controller

HBM3

Corrected Data

SEC code does not meet Reliability requirement
Lack of transparency does not meet Availability and Serviceability requirements
ID-ECC and Host ECC overlap. Risk of aliasing.

48 bits of storage total for the ECC
16 bits for ID-ECC + 32 bits for Host ECC
Alternative ECC Architectures Explored

No ID-ECC. Host ECC Only

- HBM3
- Host ECC
- Host Memory Controller
- All 48 Check Bits to Host ECC
- Corrected Data

Detection @DRAM; Correction @Host

- "XED" [Nair et al., ISCA'16]
- HBM3
- Host ECC
- Host Memory Controller
- Detection-Only Code
- Unused Check Bits for Host ECC
- Signal Detected Error to Host
- Corrected Data

DRAM and Host Share Check Bits

- "DUO" [Gong et al., HPCA'18]
- HBM3
- Host ECC
- Host Memory Controller
- ID-ECC
- Corrected Data

- Low Cost + Strong ECC at Host
- DRAM scaling faults visible at host
- ID-ECC Transparency + Strong ECC
- DRAM scaling faults visible at host
- Challenging to standardize
- ID-ECC Transparency + Avoids aliasing
- DRAM scaling faults visible at host
- Reduced correction capability
What Does it Take to Get An Acceptable RAS for HBM3?

- **GOAL:** Get to a quantifiably better RAS design than HBM2

- The total ECC storage budget cannot exceed 48 bits

- An ID-ECC is required and must be able to correct at least single-bit (DRAM scaling) faults
  - A detection-only code is not acceptable
  - Getting worse RAS than HBM2 is not acceptable
  - ID-ECC transparency is important

- Cannot add DRAM pins for solely for RAS purposes

- The ECC architecture must allow for high detection coverage at scale
Shared Component Failures Underlie DRAM Fault Boundaries

Can we bound #bits impacted by a fault?

Bank Faults

Single-Bit faults
Single-Column faults
Row Faults
Re-Design DRAM Die With Explicit Fault Bounding

- Partition the DRAM die to limit the #bits impacted by the failure of any one shared component
- Entails area penalties
- Most row faults are SWL-Arm faults
- We found that SWL-Arm faults can be bounded to a 16-bit boundary at an acceptable area cost

Row Faults become Correctable with a Symbol-based ECC
Attaining High Error Detection Coverage

- The detection coverage in HBM3 depends on both the ID-ECC and the host ECC.
- ID-ECC and the host code should be orthogonal in their capabilities. Each code is also likely implemented and evolved by separate parties: ID-ECC by memory vendor; host code by SoC vendor.
- A detection-based code (e.g., CRC) at the host can enhance overall detection capability.

Collaborated with memory vendors to study interaction of various Reed-Solomon ID-ECC and Host CRC polynomials

Key Conclusions:
- CRC8 requires careful co-selection of the CRC polynomial and RS code implementation to achieve high detection coverage
- CRC16 does not require such co-selection of the ID-ECC and host codes

Provide 16 bits per fetch to the host
The HBM3 RAS Architecture

48 bits total for the ECC
32 bits for ID-ECC +
16 bits for Host ECC

In-DRAM ECC and Host ECC are orthogonal
DRAM and SoC architects can independently evolve their ECC
HBM3 can reduce the uncorrected memory error rate over HBM2.

The node configuration is not intended to model any AMD product(s).
Conclusions

• HBM3 is expected to provide a significant RAS improvement over HBM2

• Prior research on DRAM field studies, ECC architectures, and technology studies played a key role in shaping the HBM3 RAS architecture.

• Close collaboration with the memory industry was key to developing this design and standardizing it

• For More Information:
Disclaimer

• The information presented in this document is for informational purposes only and may contain technical inaccuracies, omissions, and typographical errors. The information contained herein is subject to change and may be rendered inaccurate for many reasons, including but not limited to product and roadmap changes, component and motherboard version changes, new model and/or product releases, product differences between differing manufacturers, software changes, BIOS flashes, firmware upgrades, or the like. Any computer system has risks of security vulnerabilities that cannot be completely prevented or mitigated. AMD assumes no obligation to update or otherwise correct or revise this information. However, AMD reserves the right to revise this information and to make changes from time to time to the content hereof without obligation of AMD to notify any person of such revisions or changes.

• THIS INFORMATION IS PROVIDED ‘AS IS.’ AMD MAKES NO REPRESENTATIONS OR WARRANTIES WITH RESPECT TO THE CONTENTS HEREOF AND ASSUMES NO RESPONSIBILITY FOR ANY INACCURACIES, ERRORS, OR OMISSIONS THAT MAY APPEAR IN THIS INFORMATION. AMD SPECIFICALLY DISCLAIMS ANY IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY, OR FITNESS FOR ANY PARTICULAR PURPOSE. IN NO EVENT WILL AMD BE LIABLE TO ANY PERSON FOR ANY RELIANCE, DIRECT, INDIRECT, SPECIAL, OR OTHER CONSEQUENTIAL DAMAGES ARISING FROM THE USE OF ANY INFORMATION CONTAINED HEREIN, EVEN IF AMD IS EXPRESSLY ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

• © 2022 Advanced Micro Devices, Inc. All rights reserved.
• AMD, the AMD Arrow logo, and combinations thereof are trademarks of Advanced Micro Devices, Inc. Other product names used in this publication are for identification purposes only and may be trademarks of their respective companies.