Reliability, Availability, and Serviceability (RAS) for DDR DRAM interfaces

Marc Greenberg
Director, Product Marketing
DDR Controller IP
Synopsys
Oct 15, 2014
Abstract

• Abstract:
• Smaller process geometries, higher interface speeds, and new memory standards are driving demand for new and more robust techniques for preventing, detecting, and repairing errors in DDR DRAM interfaces. Collectively these techniques improve the Reliability, Availability, and Serviceability (RAS) of the computing system that adopts them. This session will review errors can occur in the DDR DRAM interface, and how Error Correcting Codes (ECC), parity, Cyclical Redundancy Checks (CRC), Targeted Row Refresh (TRR), and metastability prevention in the DDR memory controller and PHY can work together to protect the data stored in the DDR DRAM of a target system. Special attention will be given to the RAS capabilities of DDR4 devices.

• Abstract Takeaways
• New and increasingly common error modes, caused by higher speeds and smaller process geometries, that may cause DDR DRAM interfaces to experience more errors than current and past interfaces
• The techniques that may be used to mitigate errors in high-speed, small geometry DDR interfaces
• The capabilities of DDR4 devices to detect and prevent errors
Legal Notice

This presentation is intended as a DDR Memory technology guide and is not intended as a guide to Synopsys products.

Information contained in this presentation may or may not reflect Synopsys products or plans as of the date of this presentation. If plans exist, such plans are subject to completion and are subject to change.

Products may be offered and purchased only pursuant to an authorized quote and purchase order.
Introduction / Agenda

• What is RAS?
• What are we protecting against?
• A short summary of DRAM error and failure modes
• What are the techniques that have been used to protect DRAM in the past?
• What are the techniques that may be used to protect DRAM in the future?
What is RAS?

• **Reliability** – the ability of equipment to prevent or correct errors

• **Availability** – the “uptime” of the equipment, the ability of the equipment to recover from errors, or the ability of the equipment to remain operational during and after an error

• **Serviceability** – the ease with which system administrators can diagnose problems, detect components likely to fail, and repair failing components within the equipment
What Are We Protecting Against?

All a modern DRAM device has to do is…

Store ~4-8 billion unique bits of data
In leaky capacitors
That have dimensions of a few 10s on nanometers
And capacitance on the order of 10s of femtocoulombs
And transmit that data at billions of bits per second
In parallel – up to 72 bits at a time
And without the benefit of embedded clock in the data
And almost never lose a bit
At a cost of a few bucks

What could go wrong?
Some Error Types in DRAM

- **Soft errors** – occasional errors that are not reproducible
  - Bit cell discharged by subatomic particle strike
    - “Single Event Upset” or SEU
    - Radioactive decay in package, cosmic particles, etc
  - Signal integrity errors
- **Hard errors** – errors that can be reproduced
  - Short/open circuit on silicon
  - Short/open circuit in PCB or package
  - Silicon aging effects
  - Coupling faults between DRAM bit cells
- **Retention errors**
  - Bit cell changes state before refresh
  - Row Hammering
Example Error Types on SoC

• Soft errors – occasional errors that are not reproducible
  – Flipflop discharged by subatomic particle strike
    – “Single Event Upset” or SEU
    – Radioactive decay in package, cosmic particles, etc
  – Signal integrity errors
  – Metastability at Clock Domain Crossing (CDC)
• Hard errors – errors that can be reproduced
  – Short/open circuit on silicon
  – Short/open circuit in PCB or package
  – Silicon aging effects
What’s The Worst That Could Happen?

- Launch!
- Don’t Launch!
- Your bank balance is: MINUS $999,999,231.72
- Site busy, try again later
- Brakes busy, try again later
- RESET

SoC

DRAM

DDR Bus
## Some Examples

<table>
<thead>
<tr>
<th>Event</th>
<th>Possible Result If Not Corrected</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single soft data bit error in video traffic</td>
<td>Imperceptible glitch on video</td>
</tr>
<tr>
<td>Single soft data bit error in program data</td>
<td>Incorrect calculation in program</td>
</tr>
<tr>
<td>Single soft data bit error in program instruction</td>
<td>Program crash</td>
</tr>
<tr>
<td>Single soft data bit error in OS instruction</td>
<td>System crash</td>
</tr>
<tr>
<td>Single incorrectly captured data strobe (leads to 4-16 bytes of incorrect data)</td>
<td>Any of the above</td>
</tr>
<tr>
<td>Repeated incorrectly captured data strobe</td>
<td>System crash</td>
</tr>
<tr>
<td>Incorrectly captured address</td>
<td>Any of the above</td>
</tr>
<tr>
<td>Incorrectly captured command</td>
<td>System crash</td>
</tr>
<tr>
<td>Single hard data bit error</td>
<td>Any of the above, system instability, poor reliability</td>
</tr>
<tr>
<td>Whole DRAM chip error</td>
<td>Inability to boot</td>
</tr>
<tr>
<td>Row Hammer</td>
<td>Any of the above</td>
</tr>
</tbody>
</table>
Error Correcting Codes (ECC)

• Basic Operation: Use Hamming Codes for SECDED protection (Single-Error-Correct, Double-Error-Detect)
  – Protects against most kinds of single-bit errors:
    – SEU (Single Event Upset) due to radiation, signal integrity error, coupling or retention
    – Single bit hard error in DRAM
  – Typically adds one clock cycle latency
  – Reduces bandwidth for certain masked or unaligned transactions
  – Record corrected addresses and alert user / IT to replace DIMM after multiple corrections for better availability
Error Correcting Codes (ECC)

- Advanced Operation: Use Block-Based Codes for SxCyED protection (x-Error-Correct, y-Error-Detect)
  - Like “RAID for DRAM”
  - Choose X to protect against multi-bit errors from same DRAM device
    - Error in data strobe launch/capture
    - Error in command / address
    - Complete DRAM device failure
    - May additionally correct one SEU
  - Adds multiple clock cycles of latency
  - Some algorithms may require multiple ranks of DRAM in parallel and/or use of X4 DRAM
ECC Scrubbing

- Soft errors may accumulate in DRAM over time
- Data with long-term storage may be susceptible to accumulating multi-bit errors
- A single-bit error is correctable; a double-bit error is not correctable with Hamming ECC
- Scrubbing is the process of checking through the whole memory array and correcting any single-bit errors
- Periodic low priority process (may take hours)
Memory Sparing

Memory sparing adds additional DRAM in the system that can be used to replace failing memory on-the-fly.

1. Halt system or halt access to failing area
2. Copy data and ECC from failing column or rank of data into spare DRAM
3. Switch from failing DRAM to spare DRAM
4. Return to run condition
5. Plan for DRAM replacement (if device is in an accessible area)
Memory Sparing and Hot Swap

Memory sparing adds additional DRAM in the system that can be used to replace failing memory on-the-fly.

• Memory Sparing in accessible Location (without Hot Swap)
  1. Use spare DRAM to replace bad DRAM until controlled shutdown of machine
  2. Replace failing DIMM
  3. Restart machine

• Memory Sparing in accessible Location (with Hot Swap)
  1. Data is migrated out of failing DIMM at last step
  2. Replace failing DIMM
  3. New DIMM is now the spare DIMM to recover from the next hard error
On-Chip Datapath Parity and ECC

- Radiation affects more than DRAM bit-cells
- Any flop or SRAM bit-cell on small-geometry processes can be affected
- Protect on-chip memories with ECC
- Protect datapaths with parity or ECC
Metastability Mitigation

- DDR Interfaces often have clock domain crossings (CDC)
  - On-Chip Bus to memory controller, or between DRAM and DFI
- Consider CDC Synchronization with multiple stages
  - Improves Mean Time Between Failure (MTBF) / Failure in Time (FIT)
  - Makes latency worse
- Choose library cells with better metastability resistance
Command/Address Parity

- Signal integrity failure on command/address bus is bad
  - Command type incorrect or command sent to incorrect address
- DDR4 Devices and some DIMMs have command/address bus parity detection and alert
  - Allows system to detect error
  - Some systems may retry after error
  - DDR4 parity function adds Latency
Cyclical Redundancy Checks (DDR4)

- Signal integrity failures on Writes can be problematic
  - Single bit errors correctable by Hamming ECC unless a second error occurs in the same word due to other effects
- DDR4 Devices have optional Cyclical Redundancy Check (CRC) on write data
  - CRC Polynomial detects 1 bit, 2 bit, and some multi-bit errors
  - Allows system to detect error (and retry for bonus points)
Targeted Row Refresh

• Some DRAM devices may be susceptible to “Row Hammer” effect
  – Multiple Activate commands to aggressor rows may disturb stored charge on nearby victim rows
  – Value on victim row may be restored if Targeted Row Refresh command issued before charge is degraded to unreadable state
DDR4 DBI Improves Performance

- DDR3 SSO noise impact represented above
- DDR4 DBI limits how many data bits can switch the same way at the same time
- DBI limits the data eye shrinkage from SSO or crosstalk
- Significant timing margin gained in system timing budget
### DDR4 DBI Examples

#### DDR3

<table>
<thead>
<tr>
<th>Logic</th>
<th>Physical</th>
</tr>
</thead>
<tbody>
<tr>
<td>CKn</td>
<td>00000000000</td>
</tr>
<tr>
<td>CKn+1</td>
<td>11111111111</td>
</tr>
</tbody>
</table>

Worst Case SSO, 8 bits change simultaneously

#### DDR4 w/ DBI

<table>
<thead>
<tr>
<th>Logic</th>
<th>Physical</th>
</tr>
</thead>
<tbody>
<tr>
<td>CKn</td>
<td>111111111110</td>
</tr>
<tr>
<td>DBI_n</td>
<td></td>
</tr>
<tr>
<td>CKn+1</td>
<td>111111111111</td>
</tr>
<tr>
<td>DBI_n</td>
<td></td>
</tr>
</tbody>
</table>

DBI_n pin Minimizes transitions

<table>
<thead>
<tr>
<th>Logic</th>
<th>Physical</th>
</tr>
</thead>
<tbody>
<tr>
<td>CKn</td>
<td>1111011111</td>
</tr>
<tr>
<td>DBI_n</td>
<td></td>
</tr>
<tr>
<td>CKn+1</td>
<td>1111011111</td>
</tr>
<tr>
<td>DBI_n</td>
<td></td>
</tr>
</tbody>
</table>

DBI_n pin Minimizes transitions

---

Worst Case Xtalk, 1 victim, 7 aggressors
Data Scrambling

- A transfer of a repeated data pattern may cause spike in emitted RF spectrum and have ISI & SSO effects

<table>
<thead>
<tr>
<th>Address</th>
<th>Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>0x38</td>
<td>0xFFFFFFFFFFFFFFFFFFFFFFFFFFF</td>
</tr>
<tr>
<td>0x30</td>
<td>0xFFFFFFFFFFFFFFFFFFFFFFFFFFF</td>
</tr>
<tr>
<td>0x28</td>
<td>0x0000000000000000</td>
</tr>
<tr>
<td>0x20</td>
<td>0x0000000000000000</td>
</tr>
<tr>
<td>0x18</td>
<td>0xFFFFFFFFFFFFFFFFFFFFFFFFFFF</td>
</tr>
<tr>
<td>0x10</td>
<td>0xFFFFFFFFFFFFFFFFFFFFFFFFFFF</td>
</tr>
<tr>
<td>0x8</td>
<td>0x0000000000000000</td>
</tr>
<tr>
<td>0x0</td>
<td>0x0000000000000000</td>
</tr>
</tbody>
</table>

- May be reduced with DBI
- Some systems may scramble data by hashing data with address to improve data pattern dependent effects
On-DRAM ECC?

- JEDEC LPDDR4 standard allows for on-DRAM “data protection schemes”
- Method and benefit not described in standard
- Data protection schemes may not be designed for the direct benefit of the user!
- With on-die ECC, memory manufacturers can:
  - Ship memory with faulty bit-cells that will be corrected by ECC
  - Ship memory with some bit cells that have lower retention time, that will be corrected with ECC
  - Lower Idd6 (Self-refresh current) by allowing longer refresh intervals and correcting the subsequent errors
POST/BIST

• Power-on-Self Test (POST) and Built-in-Self-Test (BIST) allow the system to detect hard errors at power-on or at prescribed intervals.

• Regions that have developed hard errors may be mapped out of the usable memory area by the Operating System.

• System may indicate to user that maintenance is required.
# RAS Comparison by Memory Class

<table>
<thead>
<tr>
<th></th>
<th>Hamming ECC overhead</th>
<th>Memory Sparing overhead</th>
<th>On-Die ECC</th>
<th>CA Parity</th>
<th>Write CRC</th>
<th>DBI</th>
<th>TRR</th>
</tr>
</thead>
<tbody>
<tr>
<td>DDR4</td>
<td>12.5%</td>
<td>6.25%</td>
<td>?</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>?</td>
</tr>
<tr>
<td>DDR3</td>
<td>12.5%</td>
<td>6.25%</td>
<td>?</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Unlikely</td>
</tr>
<tr>
<td>LPDDR4</td>
<td>25%-100%</td>
<td>25%-100%</td>
<td>Likely</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>LPDDR3</td>
<td>25%-100%</td>
<td>25%-100%</td>
<td>Unlikely</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>DDR4 RDIMM/LRDIMM</td>
<td>12.5%</td>
<td>6.25%</td>
<td>?</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>?</td>
</tr>
<tr>
<td>DDR3 RDIMM/LRDIMM</td>
<td>12.5%</td>
<td>6.25%</td>
<td>Unlikely</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Unlikely</td>
</tr>
</tbody>
</table>
## DRAM Error Comparison by Technique

<table>
<thead>
<tr>
<th>Event</th>
<th>Hamming ECC</th>
<th>Block-based ECC</th>
<th>Command / Address Parity</th>
<th>Memory Sparing</th>
<th>BIST</th>
<th>Targeted Row Refresh</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single soft data bit error</td>
<td>Yes</td>
<td>Yes</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Single incorrectly captured data strobe (leads to 4-16 bytes of incorrect data)</td>
<td>Yes</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Repeated incorrectly captured data strobe</td>
<td>Yes</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Incorrectly captured address (on one DRAM)</td>
<td>Yes</td>
<td>Yes</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Incorrectly captured command (on one DRAM)</td>
<td>Yes</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Single hard data bit error</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>Whole DRAM chip error</td>
<td>Yes</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Row Hammer</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
</tbody>
</table>
Conclusion

• There are many types of errors that can occur in DRAM memory systems
• There are many techniques for improving the RAS of systems by preventing or correcting memory errors
• Choose the most appropriate techniques to provide the RAS required by your system

• Synopsys provides DDR Controller IP, DDR PHY IP, and other IP incorporating many of the features presented