Introduction

Why is it that some engineering teams manage to validate and debug intermittent memory failures with relative ease, while other teams struggle through the process? What can you do to make sure you don’t lose valuable time when you are trying to determine the root cause intermittent memory failures?

Typical causes of memory failures include marginal timing relationships, protocol violations, clock integrity issues, signal integrity issues, errors from other buses, incorrect BIOS setting for on-die termination (ODT) and invalid Cas latency.

How do you determine which of these is the cause of the problems in your design?

This application note outlines a debug methodology and introduces tools and techniques that can save you time and give you greater insight into system performance when you are debugging memory failures in DDR and DDR2 systems (including the SDRAM side of fully buffered DIMM).

Debug Methodology

A four-phase methodology will help you determine the root cause of intermittent memory failures. With this approach, you will quickly eliminate unlikely causes using quick checks and automated tools, then move on to exploring likely causes with more thorough tests. The four phases:

1. Determine if the failure is repeatable. Try to duplicate the failure conditions. Duplicating a failure often provides valuable insight into the nature of the problem.

2. Connect a logic analyzer to the memory bus with a probe or interposer to gain rapid insight into timing relationships of the entire DDR2 bus, parts-per-million errors, clock quality, and bus protocol.

3. Run software tests using a logic analyzer to gain insight into smallest data-valid windows, protocol errors, refresh rates, and precharge intervals.

4. Make parametric measurements using a highperformance scope with high-bandwidth probing.
Phase 1:
Duplicate failure conditions

Duplicating conditions can be challenging. Keep in mind that the root cause of the problem can come from sub-systems or applications that are not directly connected to memory. When you are evaluating memory failures, it is important to pay attention to factors such as LAN access, power sequences of subsystems, entering and exiting sleep modes, and power cycles. Problems can be caused by cross talk and conflicting resources from a variety of sources.

The problem will be easier to evaluate if you can isolate it during a specific test or set of conditions. For example, failure during a specific test could point to the software routine or signal integrity issues, such as cross talk or inter-symbol interference. With a repeatable failure, you have the ability to take multiple measurements under the failure conditions.

It can be helpful to review error logs and identify what software was running at the time of the failure. Environmental variants can also impact system failures. What was the room temperature when the system failed? What about the airflow to the system?

Be sure to take a close look at hardware issues:

- Is the power to the system within specifications?
- Has a system of this same design ever passed validation tests?
- Do other systems fail or is this failure unit-specific?
- What are the revisions on the board, DIMM, processor, or other components of the failed system?
- How does the failed system differ from working systems?
- Have there been recent component changes in manufacturing?

If conditions are repeatable, run your tests under those conditions. If they are not repeatable, choose a robust memory test and vary the test conditions, such as temperature and power supply limits, in a methodical manner.

Phase 2:
Connect a logic analyzer to the memory bus with a probe or interposer

You can save time by narrowing down problem areas quickly with logic analyzer tools. Connecting a logic analyzer, either directly or with a DDR probe or interposer, provides rapid insight across the entire DDR bus.

A new probe, the FS2334 from FuturePlus Systems, lets you use your Keysight Technologies, Inc. logic analyzer as a DDR2 800 DIMM SDRAM bus analyzer to quickly identify problems in your design. The FS2334 is an interposer design as are earlier model probes from FuturePlus for DDR1 and DDR2 in standard DIMM or SODIMM form factors.

For a higher-level protocol view of the DDR2 bus, FuturePlus Systems offers a new transaction viewer (see Figure 1), which can be used only with Keysight logic analysis systems, that lets you quickly scan bus transactions, then drill down to see details.

Most logic analyzer systems offer state traces with protocol decode to translate commands for functional validation. Simultaneous to the state capture, there are 64 k-deep traces of high-resolution timing analysis across the entire DDR bus, see Figure 2.
Figure 1. FuturePlus Transaction Viewer provides a high-level, packetized view of bus transactions. Looking at data in this view allows you to see out-of-sequence events and other protocol anomalies.

Keysight logic analyzers offer unique high-resolution eye measurements that make it possible to identify parts-per-million errors as shown in Figure 3. Other logic analyzer features include global markers (up to 1024) that can be set automatically from search functions. The global markers track between waveform and listing windows to allow for different views of suspect areas.

Colorized filters on the logic analyzer can help you visualize problems. Color filtering enables you to use pattern recognition when you view waveforms to recognize areas that require further investigation.
Figure 3. Eye scan showed a bad clock line, which is indicated by the smaller eye for CK1. The clock trace had been cut during rework on the board. Eye scan showed the level of crosstalk for CK1 into #CK1 when the negative side of the differential clock was missing.

Figure 2. A logic analyzer provides rapid insight into timing relationships across the entire bus. Logic analyzer systems offer up to 64 M deep state traces with protocol decode to translate commands for functional validation. Global markers help you make quick measurements:

- Clock frequency
- Data valid windows
- CAS latency

Logic analyzer features like eye finder and eye scan make it possible to quickly identify parts-per-million errors. These features can help you see problems such as clock noise at a glance by providing information such as setup/hold of address and control lines and relative skew of address/control signals. See Figure 3.
Phase 3:
Run software tests using a logic analyzer

When you are debugging intermittent memory failures, you can save time either by using preexisting software tools or by writing your own code for automating measurements with an advanced customization environment like the Keysight B4606A, which works with the Keysight 16900 Series logic analyzer systems. The B4606A advanced customization environment makes it easy to customize your logic analyzer’s data analysis and displays so you can quickly sift through large amounts of data to get to the nuggets of insight you need. FuturePlus Systems offers a protocol checker tool that runs in the B4606A environment. The protocol checker tool analyzes previously acquired DDR2 traces for protocol errors and calculates a variety of statistics (for example, refresh rate) on the trace. The FuturePlus Protocol Checker was used to capture the example in Figures 4,
5, 6, and 7. Here, the tool was checking for functional errors in logic analyzer traces while running repetitive captures. Every time an error occurred, the trace was saved for review. The error report in Figure 4 indicates that 128,111 commands were analyzed in the trace. There were 25 errors recorded. All errors were Writes to a bank that was not active. A marker was placed at each error. You can click on any error and the global markers track in the listing and waveform windows to help you investigate the problem. Scrolling through the errors, we see that the errors were not bank specific.

Viewing the listing, shown in Figure 5, we see that Bank 3 was precharged prior to the Write without being activated. Further investigation of the errors using the listing and waveform windows showed that not all Writes were in error. Errors occurred intermit- tently on all banks.
In Figure 6, marker measurements in the waveform show a series of five evenly spaced errors followed by a long delay until the next error. The gray bar under the black waveform window indicates which section of the trace is being viewed (shown in black) and where the errors occur on the trace. Red markers are set at all errors. Exporting data from the protocol checking tool into a graphical application as shown in Figure 7 helps visualize problem areas.
The same protocol checker tool from FuturePlus also offers timing analysis of the data valid windows relative to each other from timing zoom traces as shown in Figure 8. For example, an automated measurement might show us that the average data valid window of data bursts on one system is 2.1 ns. However, if there were two data signals with average data valid windows of 500 ps, such a dramatic variation in data valid windows would be a concern. Instead of looking at all data signals with a scope, you could concentrate your effort on the data signals with the smallest data valid windows.

Figure 7. In the FuturePlus Systems protocol checking tool, you can export acquired data for use with graphical applications and other analysis functions of spreadsheets and other COM-enabled PC applications. Engineers often use graphical analysis to view bottlenecks in different software routines.

Figure 8. The FuturePlus Systems timing analysis tool takes a previously acquired trace and calculates timing statistics and performance benchmarks on that data.

The same protocol checker tool from FuturePlus also offers timing analysis of the data valid windows relative to each other from timing zoom traces as shown in Figure 8.
Phase 4:

Make parametric measurements using a high-performance scope with high-bandwidth probing

To make parametric measurements of high-speed signals you need a scope with flat response, high bandwidth, and high sampling rate. Accurate parametric measurements are like a chain, where the lowest-performance component in the measurement system will limit the bandwidth of the measurement. For the most accurate parametric measurements, choose your scope and probing combination to provide enough bandwidth to cover the 5th harmonic of your data rate. Sample rate also affects your measurement accuracy. A sample rate of 20 Gs/s is excellent for DDR and DDR2.

The Keysight 80000 Series high-bandwidth real-time oscilloscopes have the performance required for this type of testing.

Characterization measurements you might want to make include Ts/Th, rise time, clock overshoot, frequency, and jitter.

You can use a variety of jitter analysis packages, eye measurements, eye masks, and eye unfolding software to gain insight into signal behavior.

On the left side of Figure 9 you can see an eye mask with violations on the top side of the eye. When you unfold the eye (as shown on the right of Figure 9), you will see the trace at the point(s) of failure. From the trace we can see the data pattern preceding the failure and observe signal characteristics such as rise time, ringing, and overshoot.
Probe placement is critical to making accurate parametric measurements for signal characterization. Probes read data and strobes at the memory controller. Probes write data and strobes at the SDRAM.

Figure 10 is an eye measurement of DQS0 relative to the rising and falling edges of DQS5 at T=0. The measurement was taken at an interposer in the DIMM slot and illustrates the importance of probe placement for parametric measurements. The eye for Write strobes is large and well shaped. The probe location on the interposer is close enough to the SDRAM that the signal is clear of reflections.

Read strobes are degraded from reflections at the interposer. The eye is adequate for relative measurements of pulse width as seen with a logic analyzer. However, the position on the bus is inadequate for actual characterization of the Read traffic.

For an accurate view of the Read data as seen by the memory controller, miniature scope probe tips need to be placed at the memory controller.
Conclusion

Many memory technology leaders validate and debug high-speed memory systems using the tools and techniques described in this article. Engineers who use time-saving tools reap the rewards of faster debug and greater insight into system performance.

Related Keysight literature

Data sheets

Keysight 16900 Series
Logic Analysis System Mainframes
5989-0421EN

Infinium 80000 Series Oscilloscopes
InfiniiMax II Series Probes
5989-1487EN
Evolving Since 1939
Our unique combination of hardware, software, services, and people can help you reach your next breakthrough. We are unlocking the future of technology. From Hewlett-Packard to Agilent to Keysight.

myKeysight
www.keysight.com/find/mykeysight
A personalized view into the information most relevant to you.

www.keysight.com/find/emt_product_registration
Register your products to get up-to-date product information and find warranty information.

Keysight Services
www.keysight.com/find/service
Keysight Services can help from acquisition to renewal across your instrument’s lifecycle. Our comprehensive service offerings—one-stop calibration, repair, asset management, technology refresh, consulting, training and more—helps you improve product quality and lower costs.

Keysight Assurance Plans
www.keysight.com/find/AssurancePlans
Up to ten years of protection and no budgetary surprises to ensure your instruments are operating to specification, so you can rely on accurate measurements.

Keysight Channel Partners
www.keysight.com/find/channelpartners
Get the best of both worlds: Keysight’s measurement expertise and product breadth, combined with channel partner convenience.

This document was formally known as Application Note 1575.