

内存接口的可靠性、可用性和可维护性 (RAS)

2014年7月

作者

Marc Greenberg
Synopsys公司产品营
销总监

摘要

更小的工艺尺寸和更高双倍数据速率 (DDR) 动态随机存取存储器 (DRAM) 的接口速度正在推升对更加可靠的、能够防范、检测和修复内存错误的新技术的需求。其中一些技术是由最新DDR4和DDR3 RDIMM标准的功能实现的, 另一些技术适用于任何DRAM类型。这些技术共同提升了计算系统的可靠性、可用性和可维护性 (RAS)。本白皮书阐述DDR DRAM内存子系统中可能发生的错误, 并探讨用于应对错误、提高RAS现有和未来的方法。

前言

新的计算和存储基础设施设备正呈爆炸式增长, 这些设备被整合到数据中心中, 用于实现各种云计算、电子商务和客户端-服务器应用, 连接平板电脑、智能手机等瘦客户端设备。

与此同时, 更小的工艺尺寸和更高的运行速度正在催生新的错误可能性, 而用户需要更加可靠的能够防范、检测和修复这些错误的技术。

随着行业将计算能力和商业价值整合到数据中心中, 以下三个针对数据中心的要求开始出现:

- ▶ 可靠性 - 设备防范和纠正错误的能力。
- ▶ 可用性 - 设备的“正常运行时间”, 设备从错误恢复正常的的能力, 或者设备在错误发生期间或之后保持运行的能力。
- ▶ 可维护性 - 系统管理员诊断问题、检测可能失效的组件以及维修设备中失效组件的难易程度。

对于数据中心而言, RAS要求正变得越来越重要。与此同时, 更小的半导体功能尺寸和更高的内存传输速率正在提高内存子系统发生错误的概率。日益重要的数据完整性和不断升高的错误概率正迫使电子行业开发新的内存子系统功能, 以便提高设备的RAS。

那些用于提升企业系统的RAS功能也可用于提升汽车、军事、航空航天、医疗等应用的RAS。尽管如此, 那些能够容忍偶发错误、很少强行重启的消费产品通常并不要求很高的RAS, 而且很多消费产品根本就无法维修。

大多数RAS功能都有代价。对于内存而言, 这种代价可能是内存带宽的减少、内存时延的升高、功耗的升高、硅片面积的增加或者必需购买更多内存来支持RAS功能的直接代价。

DRAM错误介绍

DRAM通过将电荷存储在一个电容器中来存储数据, 该电容器在半导体制作过程中被集成到DRAM中。目前, 主流DRAM厂商在一个硅片上制造超过40亿位 (4Gb) 的DRAM, 每一个数据位单元的尺寸仅有几十纳米, 每个电容上所存储的电荷仅有几十毫微微库仑, 代表每个位的值[1]。随着工艺尺寸不断缩小, 翻转位单元的值比以前任何时候都要容易。

多年来, DRAM被认为是完美无缺的: 可从厂家无错发货, 具备支持无限数量的读写周期的弹性, 位单元之间无电荷耦合要求, 不会出错的数据保持, 以及CPU和DRAM之间数据传输的零误码率 (BER)。DRAM当然非常可靠, 非常适合在消费设备中使用。与Flash内存相比, DRAM的错误通常少几个数量级。但是, DRAM的完美假设并不适用于高可靠性应用。

DRAM错误的一些来源包括:

- ▶ 亚原子粒子撞击内存或SoC, 导致所存储值的损坏 — “软错误”或“单粒子翻转”(SEU)。
- ▶ DRAM结构中的物理错误可导致某些位不可编程。
- ▶ 与数据有关的耦合错误, DRAM某个部位的活动导致邻近单元内容的损坏。。
- ▶ 影响单个数据位、一组数据位或一条发送至DRAM命令传输的信号完整性错误。
- ▶ 整个DRAM完全失效。

理论上而言, DRAM厂商应该屏蔽掉上述很多错误, 但它们会在机器首次通电或正常使用期间偶尔出现。人们对早期发生错误的来源有争议, 有人认为它们是因用户对DRAM使用不当造成的(例如, 电磁放电或过高的焊接温度), 另一些人则认为它们是厂商屏蔽流程漏检造成的。使用中发生的错误可能来自各种不同的来源, 其中包括电迁移等硅片老化效应, 或者某次初始事件(例如, 一次损坏位单元的不当使用)后重复使用造成的错误。

由亚原子粒子和宇宙射线引入的软错误

一种记录在案长达30多年的DRAM错误模式[2]是: 亚原子粒子撞击某个DRAM位单元, 产生电子空穴对, 从而影响该DRAM位单元上所保持的电荷。

通常而言, 这些错误有两种形式: 来自遥远来源(“宇宙射线”)的高能粒子撞击, 或者来自DRAM以及封装内部偶尔的放射性衰变产生的阿尔法粒子撞击。

这些所谓的“软错误”或SEU是DRAM子系统中所使用的第一代纠错功能起源。

硬错误: 固定错误和转换错误

一种著名的DRAM错误就是“固定”错误, 即无论被如何编程, 某个DRAM位单元都会连续输出一个固定的值。另一种同样著名的错误是“转换错误”, 即某个位单元可以从0转换到1, 或从1转换到0(仅一次), 但却不能向相反的方向转换。

保持错误

有一种很难诊断的错误是由DRAM保持故障导致的。DRAM中电容器上的电荷随着时间的推移向外泄露, 因此, DRAM位单元需要通过定期刷新恢复它们的电荷。

DDR DRAM内存控制器每隔一个平均刷新间隔时间向DRAM发送一条刷新命令, 一次刷新一小部分内存。对于一个保持时间为典型的64ms/位单元的内存, 刷新操作被细分8,192个周期刷新命令, 间隔时间为7.8us, 将在64ms内刷新内存中的所有位置, 然后重复这个过程。在室温下[3], DRAM位单元的平均保持时间可能实际长达几秒, 但却有一个非常宽广、与温度和存取有关的分布, 以便让分布的尾部能够在最差条件下高于64ms。

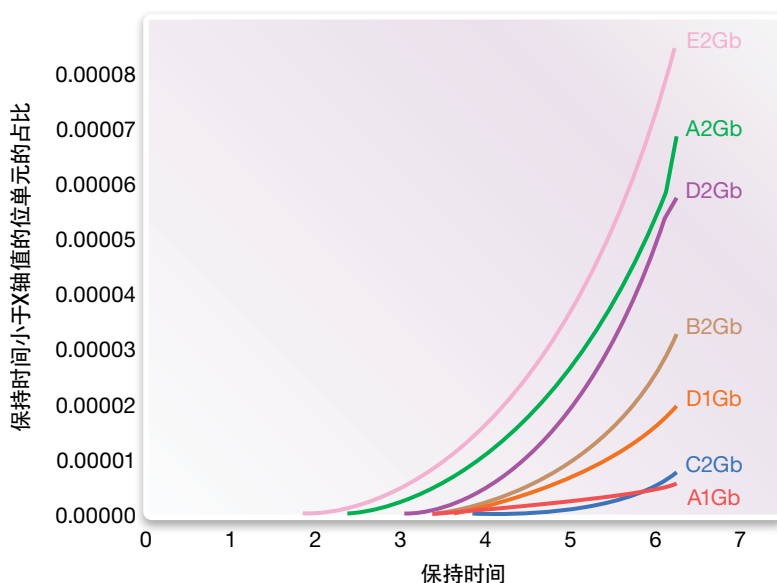


图1: 卡耐基梅隆大学和英特尔公司最近联合开展的一项研究成果: 几家DRAM厂商的产品(A-E)中拥有不同的保持时间(X轴)的位单元的占比(Y轴)。如图所示。DRAM中99.99%的位单元的保持时间超过6秒。DRAM厂商面临的挑战是使得分布的尾部100%高于64ms。

数据保持错误指的是某个DRAM位单元有可能在DRAM的正常刷新时间(图1中的64ms)过去之前失去其电荷。保持错误可能只影响DRAM上的一个位单元,在特定的电压和温度条件下,断续出现,也可能与数据模式有关[4]。存在保持错误的DRAM位单元(不能在整个刷新间隔时间内保持电荷的位单元)可能会断断续续地导致错误,因此,纠正此类错误好于防范它们。

耦合错误

当DRAM中一个或多个位单元的编程损坏邻近某个位单元的值时,由耦合导致的错误就会出现。由于这些错误已被发现很多年,大多数内存测试算法会生成针对大多数类型耦合错误的测试,以便在制造测试期间屏蔽掉它们。尽管如此,仍然存在制造测试漏检耦合错误的可能性。

Row hammering

最近发现的一种耦合错误是“row hammer”效应(图2),即针对某个侵略行的大量激活可能会导致某个邻近被侵略行中的电荷退化,从而导致被侵略中出现一个或多个误码[5]。与传统的耦合错误不同,row hammer错误不要求对任何位单元编程,也能影响被侵略行中的多个位单元。设备厂商公布了一些有关这种错误的信息,但目前发布的DDR DRAM规范没有禁止row hammering错误。由于这个问题很严重,一家厂商制作了一种能够检测row hammer错误的测试设备[6]。

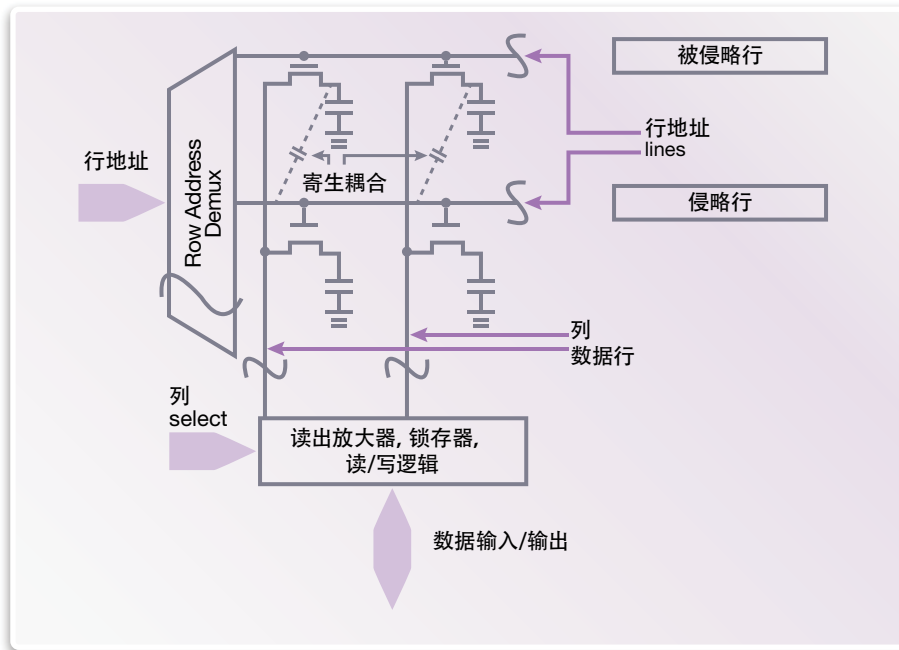


图2: 地址行与位单元电容器之间的寄生耦合是导致row hammering效应的一个可能原因

影响单个数据位、一组数据位或命令传输的信号完整性错误

由于高速DRAM能够在无需数据中嵌入式时钟的帮助下以GB/秒级别的速度并行传输数据,人们预测通道中的BER可能会大于零。某些信号完整性错误是由符号间干扰(ISI)或同时交换输出(SSO)等与数据有关的效应、数据行反射、训练偏移、或者来自系统中其它组件的噪声导致的。

信号完整性错误既有可能发生在DRAM的数据通道中,也有可能发生在命令/地址通道中。如果发生在数据通道,信号完整性错误可能会影响某一位,或者影响整个数据选通,从而影响DRAM的所有数据输出。如果发生在命令/地址通道,信号完整性错误可能会导致一条错误的命令被发送到DRAM—更改某条命令的地址部分,或者完全更改命令类型。

信号完整性错误曾被视为DRAM的主要错误模式,但一些研究显示。此类错误相对而言少于其它类型错误[7]。

DRAM完全失效

人们担心整个DRAM有可能失效，即某个DRAM芯片的所有输出有可能在一个数据突发期间或永久性地同时失效。整个DRAM芯片失效的原因与任何芯片完全失效的原因类似，由于原因太多，无法在此罗列。尽管如此，高可靠性系统应能抵御致命的DRAM错误。

芯片错误

当然，DRAM不是系统中唯一可能出错的部件。随着工艺尺寸日益变小，某个触发器有可能在没有外部输入的情况下改变状态，例如，在阿尔法粒子的撞击下改变状态。另一个可能的失效错误点位于异步时钟域的交界处，还有一个位于嵌入式RAM中。通过在芯片设计中集成特定的功能或技术，这些类型的错误通常都能被避免。这些功能和技术将在本白皮书的后面探讨。

DRAM的初始RAS: 汉明码ECC和BIST

有一些技术可以减轻DRAM错误在DRAM中出现时所产生的影响。

奇偶校验

最早的一种检错方法是对DRAM进行简单的奇偶校验。一条典型的64位内存通道配有一个额外的8位奇偶校验数据，与64位数据一同存储。每个字节的数据都有一个与之关联的奇偶校验数据位，这个奇偶校验位被设置为让数据+奇偶校验中的数字‘1’成为奇数或偶数。

奇偶校验功能速度快，只需很少的异或门就可实现，但却没有纠错能力，也不能检测出一个字节中的奇数数字。鉴于这些原因，奇偶校验让位给了更加强大的校验编码。

汉明码ECC

当今系统仍在使用的汉明纠错码 (ECC) 问世于1950年，比半导体DRAM问世早十年[8]。虽然有很多可能的实现方式，但汉明码ECC是企业设备中的一个常见方案。在该方案中，64位数据生成8个错误校验位。这个汉明码 (64,8) 方案可提供单纠错、双检错 (SECDED) 能力，即能够纠正64位数据中的1个错误位，检测2个错误位。如果有2个以上的错误位，则结果无效。

汉明纠错码所需的计算量大于奇偶校验，虽然大多数现代芯片能够运行这个计算，但它通常需要使用1个时钟周期来检查某个输入64位字的错误，并纠正所发现的单位错误。这个时延可能会略微影响系统性能。

很多类型的系统可能需要执行部分写操作，将新数据写入64位字节中的某些字节，并保持其它字节不变。常见的DDR DRAM配有一个数据遮罩输入，当未使用ECC时，它允许执行部分写操作；激活DRAM的数据遮罩输入和数据输入意味着该DRAM中的数据应保持不变。

奇偶校验或ECC功能被启用后，执行一次部分写操作通常需要一次读-修改-写 (RMW) 操作，读取底层数据，并将其与更改的字节合并，然后从合并后的数据生成一个新的奇偶校验或ECC字节 (图3)。这个RMW操作可能会耗费多个时钟周期；如果系统试图在奇偶校验或ECC功能被启用后尝试执行多次部分写操作，RMW操作会严重影响系统性能。

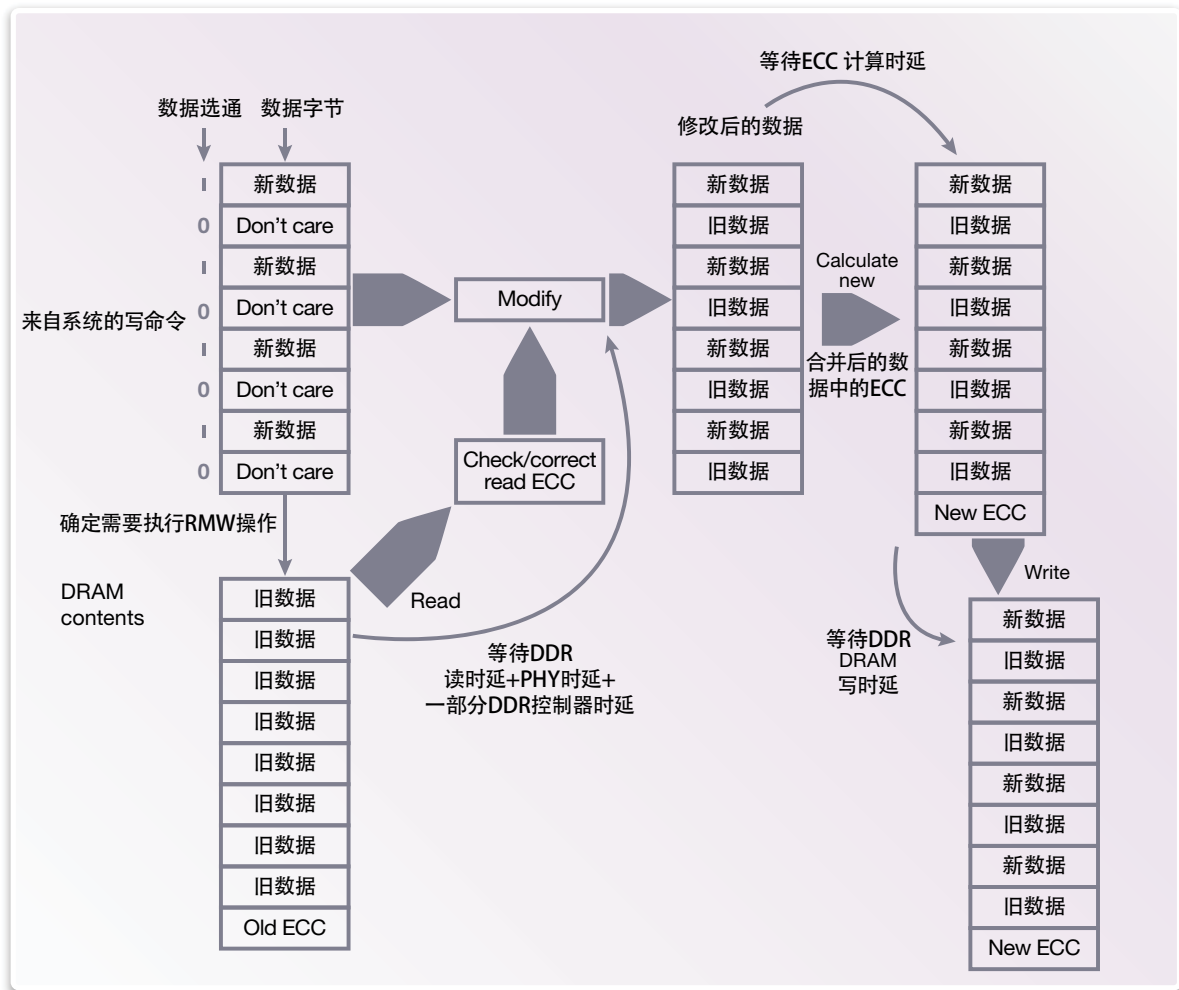


图3: RMW操作中的主要步骤。如果需要执行多次RMW操作, 该过程可能较慢。

内置自测

虽然纠错码能够检测出瞬时和永久故障导致的错误, 但大多数ECC只能纠正一个数据字中的一个错误。替换或维修存在永久故障的内存, 能够降低瞬时故障影响同一个内存字的概率, 因为永久故障可能导致一个不可纠正的错误。

内存BIST是测试内存永久故障的一种测试方法。有很多不同的BIST算法, 每种算法在复杂性、测试时间和测试覆盖率方面都有不同的权衡。

内存BIST可以是一个硬件模块, 也可以是系统CPU中的一个软件算法。通常而言, 硬件方法占用芯片面积, 但速度较快; 软件算法不占用逻辑面积, 但却占用程序空间, 而且速度较慢。

片上奇偶校验和片上ECC

如上所述, SEU或亚稳定性问题可能导致SoC内部、输入和输出驱动器之间以及片上总线中出现错误。有一些减轻片上错误的影响的方法。

用于检测片上错误的一种方法是引入片上奇偶校验功能, 即生成一个奇偶校验位 (每个字节或每个字), 然后将其与数据一同传送, 以此检测片上错误。该方法快速简单, 但缺点是只能检测错误, 不能纠正错误。

另一种方法是将DRAM ECC信号从DRAM一直延伸到片上总线接口, 甚至延伸到生成DRAM请求的主接口。该方法能够实现在主接口处纠正位错误, 但要求可能生成部分或屏蔽写操作的每个接口必需能够发起一次RMW操作, 并保持读和改写之间的内存位置不变。

还有一种采用片上ECC和片外ECC的混合方法。在该方法中, 一个ECC单元仍位于内存控制器中, 能够执行RMW操作。第二个ECC单元负责保护片上总线与内存控制器中ECC单元之间的数据。

所有这些片上奇偶校验或ECC解决方案只保护数据路径。内存控制器的数据路径一般含有最大数量的触发器，因此是最可能出错的地方；尽管如此，控制路径上仍可能出错。可靠性极高的系统需要更高的RAS，而设计人员可能选用抗辐射设计技术来降低控制路径出错的概率。

DRAM的高级RAS

当今最先进的高可靠性设计和未来5-10年的设计将利用多种创新技术提升内存接口的RAS。其中一些创新技术基于现有技术，另一些则是现有和未来内存中的新技术。每个系统都有不同的RAS要求，而且每个内存控制器和PHY也会拥有可能随时间增加的RAS功能。

高级ECC

高级ECC指的是一系列技术，它们能够实现比基于汉明码(64,8) ECC的SECDED更好的纠错效果。新的高可靠性应用必须解决“DRAM完全失效”问题，即系统中的某个DRAM芯片可能完全失效，造成DRAM接口上每个数据字的4或8位出错。纠正这些“DRAM完全失效”错误的方法被称为S4EC-D4ED，或采用很多专有名称，如IBM的“Chipkill”、英特尔的“Single Device Data Correction (SDDC)”、惠普的“Chipspare”以及Sun Microsystems的“Extended ECC”。

提供高级ECC功能的一种方法是通过扩展汉明纠错码提升纠错能力，但由于每64个数据位通常需要8个以上的ECC校验位，因此，需要创建带宽大于标准的72位(64位数据+8位ECC) DIMM的定制DIMM。创建更宽的DIMM并新增用于支持ECC的额外内存成本通常较高，并不适合大多数应用。

另一种高级ECC方法是从多个字节(或nibble)加扰数据，从而让来自任何DRAM的每个汉明码字最多包含1个位。这样一来，某个DRAM的完全失效仍能被纠正，因为每个汉明码字中仅有1个位受到影响。

高级ECC可能使用Reed-Solomon、BCH或LDPC等基于块的码字，能够纠正单个错误或“DRAM完全失效”错误，但可能会对性能、功耗和系统产生影响。使用基于块码字的高级ECC可能产生的影响包括：

- ▶ 延长实现纠错功能的时延
- ▶ 当需要时，进一步延长实现纠错功能的时延
- ▶ 减少内存带宽
- ▶ 增加芯片面积和功耗
- ▶ 限制内存系统配置，例如，要求使用多级DIMM或多个串联DIMM

使用ECC(奇偶校验、汉明码或某种高级ECC)的一个常见缺点是：当DDR控制器被要求执行一次部分或屏蔽写操作时，控制器必需执行一次RMW操作，以确保得到正确存储。这要求读取底层数据，并将其与正在更改的字节合并，然后从合并后的数据生成一个新的奇偶校验或ECC字节。RMW操作可能会耗费多个时钟周期；如果系统试图在奇偶校验或汉明码ECC功能被启用后尝试执行多次部分写操作，RMW操作会严重影响系统的性能。

ECC在外置内存和嵌入式SRAM中得到了广泛应用。最近的调查显示，单一位翻转已不再是当今工艺节点的主要软错误，多位翻转错误在嵌入式SRAM中发生的概率与单一位翻转近似。因此，汉明码等传统的ECC方法也被认为不足以支持嵌入式SRAM。用户需要新的多位纠错解决方案，让他们能够根据具体应用，从传统的单位、双位、三位和四位纠错率中选择或调整所需的多位纠错率。

并非每个系统都会选择高级ECC，那些选择高级ECC的系统可能会利用编程来指定是否在某个特定的应用中使用高级ECC。

内存备用

在内存备用方法中，系统中配有额外未使用(“备用”)的内存，让系统能够通过将数据从失效内存引导至此前未使用的内存，消除或减轻失效内存组件的影响。内存备用法可以防止错误在内存的同一个区域累积；当一个可纠正的SEU出现在某个字中，而该字中的一个不同的位已经出错时，这种错误累积情况就会发生；这些错误可以被单独纠正，但当多种错误同时出现在同一个字中时，它们将无法被纠正。

当使用内存的设备位于偏远地区、塔顶或太空等难以接近的地方时，内存备用法将会非常有用。内存备用法有很多种，其中一种是将现有内存的一部分划分为“备用”，让操作系统或其它系统将失效位标记为不可用，将备用内存和失效内存之间的地址重新映射。

某些系统的每条内存通道可能包含一个完整的备用DIMM，可用于替代失效的DIMM。某些高级ECC非常强大，不需要所有的ECC位也能实现ECC功能，从而腾出DIMM的某些位，用作备用[9]。

命令重试

当DDR4内存和某些DDR3在命令/地址总线(基于奇偶校验)或某个DDR4写命令中(使用一个循环冗余码)检测到一个错误时。它们能够维护一个“告警”信号。早期的DDR控制器会向系统发送一个中断,表示“有事发生”,系统可以选择忽略它、进入重启程序或采取其它措施。

更加先进的DDR控制器[10]存储一个最近执行的命令的日志,能够在命令失败和接收到告警信号期间重新执行失败的命令。

减轻Row hammering效应的影响

上述DRAM的“rowhammer”效应发现得较晚,因此用于应对它的技术正处在研发阶段。以下解决方案将会问世:或是防止出现可导致row hammering错误的内存使用模式,或是能够检测出哪些行可能会受row hammering影响,并利用目标行刷新[11]等未来可能的内存技术防止数据丢失。

温度时间参数调整

随着DRAM的温度不断上升,当芯片温度超过与器件阈值时,保持时间等特性将受到影响,从而增加保持失效的概率,除非刷新时间间隔等参数得到调整。最新的DDR可能内置一个温度传感器,它能够检测出温度阈值何时被超越,温度数据通常可以从DDR的模式寄存器读出,而某些DDR控制器 [10]能够根据这个数据自动调整与温度相关的时间参数。

片上奇偶校验和片上ECC

先进半导体工艺中的触发器,易受可导致DRAM错误的相同软错误机制的影响。鉴于此,某些系统利用奇偶校验和ECC保护片上主接口与DRAM I/O引脚之间的数据路径。

封装后维修

如果器件在组装后,或在使用期间出现了硬错误,制造商可以选择维修器件,而不是替换它。这有助于维修那些位于难以维护的器件,也能够纠正正在制造期间发现的错误,而非替换整个器件,从而降低制造成本。DRAM厂商正在开发各种技术,让用户能够在DRAM受到损坏时维修它们(Jong-Pil Son 2010)。

除了找出有缺陷的DRAM并维修它们之外,设计人员还必须对逻辑芯片和DRAM之间的互连做同样的保护。这要求使用一个智能引擎运行全速测试,并具有能够找出错误的互连或DRAM的诊断能力。此类IP需要在DDR PHY之前将嵌入到逻辑芯片中。

结束语

本白皮书描述了DDR DRAM可能出现的一些错误,以及可用于提升容易出现这些错误系统的RAS的技术,其中包括当前和未来的系统。

Synopsys可以提供包含一个卓越、不断扩展RAS功能集的DesignWare? DDR Controller和PHY IP、面向嵌入式SRAM的高级STAR ECC IP以及面向外置内存测试及诊断的DesignWare STAR Memory System? (SMS)。

更多有关Synopsys DesignWare DDR IP解决方案的信息,敬请访问: <http://www.synopsys.com/ddr>。

参考资料

- [1] Alchesky, Larry C. “Single-Bit-Errors: A Memory Module Supplier’s perspective on cause, impact and detection.” (单位错误: 提价内存模块供应商对于此类错误的原因、影响和检测的阐述) Smart Modular Corporation的 Web Site. n.d. http://www.smartm.com/files/salesLiterature/dram/smart_whitepaper_sbe.pdf.
- [2] Ziegler, James F and Lanford, William A. “The Effect of Sea Level Cosmic Rays on Electronic Devices.” (海平面宇宙射线对于电子器件的影响) ISSCC. 1980. 70-71.
- [3] J. Alex Halderman, Seth D. Schoen, Nadia Heninger, William Clarkson. “Lest We Remember: Cold Boot Attacks on Encryption Keys.” (以免我们记住: 冷启动对加密密钥的攻击) USENIX Security Symposium. 2008年。
- [4] Jamie Liu, Ben Jaiyen, Yoongu Kim, Chris Wilkerson, Onur Mutlu. “An Experimental Study of Data Retention Behavior in Modern DRAM Devices: Implications for Retention Time Profiling Mechanisms.” (现代DRAM器件中数据保持行为的实验研究: 保持时间分析机制的影响) ISCA ‘13. 以色列特拉维夫, 2013年。
- [5] Micheletti, Mike. “Tuning DDR4 for Power and Performance.” (优化DDR的功耗和性能) Memcon, 圣何塞, 2013年。
- [6] M. Micheletti, “Teledyne LeCroy Press Release,” (Teledyne LeCroy的新闻稿), 2013年9月4日 [在线]。网址: <http://cdn.teledynelecroy.com/files/pressreleases/09042013.pdf>。
- [7] Bianca Schroeder, Eduardo Pinheiro, Wolf-Dietrich Weber. “DRAM Errors in the Wild: A Large-Scale Field Study.” (野外发生的DRAM错误: 一次大规模现场调查) Sigmetrics, Seattle, 2009.
- [8] Hamming, R.W. “Error Detecting and Correcting Codes.” (错误检测和纠正码) The Bell System Technical Journal 29, no.2 (1950) : 147-160页。
- [9] Intel. “Intel® Xeon® Processor E7 Family: Reliability, Availability, and Serviceability.” (英特尔至强处理器E7系列: 可靠性、可用性和可维修性) 2011. <http://www.intel.com/content/dam/www/public/us/en/documents/white-papers/xeon-e7-family-ras-server-paper.pdf> (2013年12月5日访问)。
- [10] Synopsys. Synopsys DesignWare Cores Enhanced Universal DDR Controller (uMCTL2) Databook. (. Synopsys DesignWare内核增强型通用DDR控制器 (uMCTL2) 数据手册) Mountain View: Synopsys, 2013年。
- [11] Skinner, Dan. “LPDDR4 Moves Mobile - JEDEC Mobile Forum.” (LPDDR4走向移动 – JEDEC Mobile Forum) 2013年。 http://www.jedec.org/sites/default/files/D_Skinner_Mobile_Forum_May_2013_0.pdf