Introduction

Although small, there is some likelihood that DRAM-based memory in a computer can fail. These failures are of two basic categories:

1. Hard fails, in which the nature of the failure repeats, and is basically permanent. Fixing these failures permanently may require replacement of some part of the memory hardware. Hard error rates are known as HERs.
2. Soft fails, non-permanent failures that may never reoccur, or occur at infrequent intervals (soft fails are effectively “fixed” by powering the system off and back on). Soft error rates are known as SERs.

SERs are higher than HERs. These errors stem from two sources: alpha particles and cosmic rays. Alpha particle SERs have been virtually eliminated in modern DRAM technology. No DRAM is entirely insensitive to cosmic rays, but IBM DRAMs have very low cosmic ray sensitivity because the IBM “inside-store” trench cell stores more charge. In addition, the inside-store trench stores the charge in a dielectric capacitor, minimizing the charge collection area.

All system designers should have a basic goal of providing as reliable a system as possible. Over time, methods have been developed (and discussed and debated endlessly!) to minimize the impact of system memory failures via improving system fault tolerance. This application note presents a comparison, from an advantage/disadvantage aspect, of the four basic levels of fault tolerance:

- Non-parity
- Parity
- ECC, or error-correction coding
- EOS, or ECC-on-SIMM

Non-parity

Basically, non-parity systems have no fault tolerance at all. The reason they are even used is because they have the lowest inherent cost. No additional memory is necessary as is the case with parity or ECC techniques. Since a parity-type data byte has nine bits versus eight for non-parity, memory cost is 12.5% higher. Also the non-parity memory controller is simplified since it does not need the logic gates to calculate parity or ECC check bits. Portable systems which place a premium on minimizing power might benefit from the reduction in memory power due to fewer DRAM chips. Finally, the memory system data bus is narrower which reduces the amount of data buffers. In modern memories, HERs and SERs are smaller than before, and as a result, the likelihood of memory errors occurring in a given system for its entire operating life has dropped to an extremely small level. The statistical probability of memory failures in a modern office desktop computer is now estimated at less than one in ten years. This error rate may be tolerable because:

- The error may result in a system hang, or doing something so anomalous as to be obvious to the user, in which case he/she turns off the system and reboots. This is the case with a soft error, for example.
- In the case of hard errors rendering a system inoperative, running the system diagnostics may lead to the problem source.
- In low-end systems, such as those intended for playing games, their extreme market cost sensitivity probably can’t justify the extra cost of parity memory.

At any rate, employing no fault tolerance in a system is simply gambling that memory errors are unlikely, and if they do occur, result in an inherent cost less than the additional hardware necessary for error detection. However, the disadvantage is that the errors can lead to a serious problem such as calculating the wrong value to go into a bank check, or in the case of a system being used as a server, a memory error forcing a system hang and bringing down all LAN-resident client systems with subsequent loss of productivity. Finally, with a non-parity memory system, problem traceability is difficult, which is not the case with parity or ECC. These techniques at least isolate a memory source as the culprit, thus reducing both the time and cost of problem resolutions.

Parity

Parity, as mentioned previously, results in increased initial system cost due primarily to the additional memory bits involved. Parity cannot correct system errors, but, since parity can detect errors, it can make the user aware of memory errors when they

occur. This has three basic benefits:

1. Guards against the consequences of faulty calculations based on incorrect data.
2. Pinpoints source of errors which assists in problem resolution, thus improving system serviceability.
3. Potential users may perceive system as higher-quality than non-parity systems, resulting in marketplace benefits.

Finally, DRAM module-based systems can easily be designed to function in both parity and non-parity environments. This enables the system manufacturer to offer their system purchasers the choice of parity if they feel the additional cost is justified for their particular application.

ECC

Since studies have indicated that approximately 98% of memory errors are single-bit, the most common type of ECC is one in which the attendant memory controller detects and corrects single-bit errors in an accessed data word (double bit errors can be detected, but not corrected). This type of ECC is known as SEC-DED and requires an additional seven check bits over 32 bits in a 4-byte system and eight check bits in an 8-byte system. ECC in a 4-byte system obviously costs more than non-parity or parity, but in an 8-byte system ECC and parity costs are equal.

ECC entails the memory controller calculating the check bits on a memory-write operation, and performing a compare between the read and calculated check-bits on a read operation and, if necessary, correcting bad bit(s). The additional ECC logic in the memory controller is not very significant in this age of inexpensive, high performance VLSI logic, but ECC actually affects memory performance on writes. This is because the operation must be timed to wait for the calculation of check bits and, when the system waits for corrected data, reads. On a partial-word write, the entire word must first be read, the affected byte(s) rewritten, and then new check bits calculated. This turns partial-word write operations into slower read-modify writes.

Most memory errors are of a single-bit nature, correctable by ECC. Incorporating this fault-tolerant technique provides high system reliability and attendant availability. An ECC-based system is a good choice for servers, workstations, or mission-critical applications in which the cost of a potential memory error outweighs the additional memory and system cost to correct it, and ensures it does not detract from system reliability.

By designing a system that allows the user to make the choice of ECC or parity, or non-parity for that matter, ultimate flexibility is provided. Several of IBM's personal computer systems provide user selectability of memory fault tolerance techniques.

EOS

These DRAM module types provide an upgrade path to ECC via replacing the previously installed modules with EOS-type SIMMS, thus entailing no processor or planar changes. The EOS SIMMs carry out ECC not on word, but on single-byte boundaries. Through the use of high speed DRAMs and fast on-board logic, performance degradation is effectively masked. The cost of ECC implementation using EOS is higher than a combination of ECC DRAM modules and an ECC-compatible memory controller. This is due primarily to the extra memory necessary to perform ECC on a per-byte basis (ECC on a byte requires four check bits, thus on a 32-bit word a total of 16 check bits are necessary) and the repetition of the ECC logic on each installed SIMM. However, EOS SIMMs are an excellent method to obtain the reliability-enhancing advantages of ECC without performance impacts. They also provide the system OEM with a method of providing the system purchaser a choice between ECC and parity-based memory.

For more information contact your IBM Microelectronics sales representative or
visit us on World Wide Web at http://www.chips.ibm.com

IBM Microelectronics manufacturing is ISO 9000 compliant.