Whitepaper on
Soft Errors in Modern Memory Technology
By Charlie Slayman, Ph.D., Ops A La Carte LLC

INTRODUCTION – What are soft errors and what causes them?

As IC device technology scales to smaller and smaller feature sizes, the subject of soft errors is gaining importance in the design of reliable systems. A soft error is the upset in memory, static logic or combinational logic. The term “soft” is used because the upset is not due to any permanent damage of the circuit. If the error can be detected and corrected, the device can be returned to its proper operating condition without any service intervention.

Soft errors originate from charge generation and collection created by energetic particles striking the sensitive volume of the IC. If the particle generates a sufficient amount of charge to change the state or output of the circuit, this is called critical charge (Q_{crit}). The three sources of energetic particles in commercial environments are alpha particles (^{4}\text{He} nucleus), high-energy neutrons and thermal neutrons. Alpha particles come from material contamination in wafer fabrication and chip packaging. The peak energy of isotopes that are common contamination sources range from 1 to 10\text{MeV}. High-energy neutrons are the result of cosmic rays striking the earth’s upper atmosphere and generating a shower of high-energy atomic and sub-atomic particles. At terrestrial levels (i.e. sea level to 10,000 feet), these neutrons are the dominant cosmic ray particles that cause soft errors. Their energy can range from below 1 \text{MeV} to above 1 \text{GeV}. Thermal neutrons are the result of high-energy neutrons that lose energy by scattering off materials in the environment and ultimately reaching the thermal equilibrium energy (\sim 25\text{meV}). Thermal neutrons generally do not have enough energy to generate significant charge, but in special cases where the thermal neutron reacts with the material and generates nuclear fission, high-energy charged particles are created that result in soft errors. The ^{10}\text{B} isotope of boron has an extremely large thermal neutron capture cross-section and is commonly found in many IC processes. The result of the
interaction is a 1.87 MeV $^4$He (alpha) and 0.84 MeV $^7$Li particles along with a 0.48 MeV photon (gamma ray)[1].

Tezzaron did a very comprehensive study of soft errors in memory [2]. However, much of the data collected from various sources gives conflicting information and is now more than seven years out of date. This whitepaper is intended to compile more up to date soft error data on commercial electronics based on technical conference and journal publications.

**MYTHS AND MISCONCEPTIONS OF SOFT ERRORS**

*Myth 1 – DRAM soft error rates will increase as process technology shrinks and will dominate SRAM soft error rates.*

In reality, DRAMs demonstrate a much better error resistance because storage capacitance is remaining constant while cell size is decreasing. Therefore, the charge generated per volume of silicon must increase in order to upset and advanced DRAM design. But to first order, the charge generated per unit volume is a constant determined only by the particle energy and is not a function of device technology.

Figure 1 shows the cell upset trends for DRAM and SRAM from different manufacturers due to high-energy neutrons [3]. There is significant scatter in the data indicating that the details of the process technology have an impact on the soft error rate. The downward trend for DRAMs reflects the fact that the cell capacitance is not scaling, so it is more difficult for a neutron to generate enough critical charge in a shrinking volume. On the other hand, the SRAM soft error rate is roughly flat on a per bit basis, indicating that the sensitive volume and critical charge are scaling at roughly the same rate. Figure 2 shows the DRAM and SRAM cell upsets x (design rule)$^2$. The trend lines reflect what the expected soft error rate per chip or system will look if the memory density scales as 1/(design rule)$^2$. Note that the SRAM soft error rate (SER) is actually increasing with process technology scaling while the DRAM SER is decreasing.
FIG. 1 DRAM and SRAM cell upset rates vs. design rule from high-energy neutrons.

FIG. 2 Product of (DRAM and SRAM cell upset rates) x (Design Rule)$^2$ from high-energy neutrons.
Myth 2 - Alpha particle problems have been largely eliminated.

Alpha soft error trends are difficult to predict because they are dependent on process and package contamination factors as well as device scaling. The former is under complete control of the IC and package manufacturer and not subject to any trends. In general, IC manufacturers who understand soft error issues attempt to balance the contribution of alpha particles and neutrons to overall soft error rates rather than trying to completely eliminate alpha particle soft errors. Total elimination of alpha particle soft errors due to extreme material and process purity requirements would add significantly to the cost of the final product without eliminating neutron soft errors.

Myth 3 – Memory errors are dominated by hard fails and not soft errors.

In a recent study [4], the authors concluded that memory errors are dominated by hard errors, rather than soft errors. However, their method of collecting data did not allow for access to information on the address of the error. So they were unable to distinguish multiple error counts from repeated reads of a single failure vs. multiple errors counts from single reads of independent failures. As a result, their technique of counting errors was insufficient to truly distinguish between hard errors and soft errors.

Myth 4 – Multi-bit errors are fatal.

Several misconceptions exist around this myth. If a neutron or alpha particle generates enough charge that is spreads across a distance, then several memory cells can be upset. This is defined as a multi-cell (not multi-bit) upset in the JEDEC Standard - JESD89A Measurement and Reporting of Alpha Particle and Terrestrial Cosmic Ray-Induced Soft Errors in Semiconductor Devices [5]. If physical interleaving is used, data bits from the same word are spread out across the memory array. Therefore, nearest neighbors cells do not store data from the same word and multi-cell upsets lead to multiple single bit errors in different words. These are easily corrected by single error correct – double error detect (SEC-DED) codes used for many cache applications. In addition,
when words are wide enough (e.g. 128, 256 or 512 bits), as is the case with main memory, multi-bit error correction codes are implemented with minimum overhead (typically an 11% area or chip count penalty).

**TYPES OF ERRORS**

**Correctable Errors/Uncorrectable Errors/Silent Data Corruption**

There are three broad categories of memory errors. Those errors that are detected by the memory control system can either be correctable errors (CE) or uncorrectable errors (UE). Errors that are undetected can lead to silent data corruption (SDC). Examples of CEs are single bit errors in a SEC-DED design, four bit errors in a S4C-D4D design or an odd number of bit errors in a parity protected design where copies of the data exist and are uncorrupted. In the first two examples, the correct data can be re-assembled mathematically from the corrupted checkword. In the third example, the parity bit is used to flag the corrupted word so that it can be replaced by a correct copy. Examples of UEs are parity errors where a copy of the original word do not exist, two bit errors in a SEC-DEC design and >4 bit errors in a S4C-D4D design. Example of SDC are even bit errors in a parity protected design (since this will falsely give a correct parity signal), >2 bit error in SEC-DED designs and >8bit errors in S4C-D4D designs.

**Hard errors vs. Soft Errors**

Memory errors are further classified as hard errors and soft errors. A hard error indicates that there is some permanent damage to the memory element and it is no longer able to retain the correct data. A soft error indicates a temporary bit flip has occurred that can be corrected by simply rewriting the data. This is also called a single event upset (SEU). In addition, soft errors from energetic radiation can occur in logic elements. If these soft errors impact the functioning or state of the circuit (beyond simply storing a single bit of a data word), they are called single event functional interrupts (SEFI). In addition, an energetic particle can also drive the parasitic pnnp or npnp thyristor structures of a CMOS circuit into latch-up, known as single event latch-up (SEL). In
any case (i.e. SEU, SEFI and SEL), if there is no permanent damage and the proper data or state can be restored by a rewrite, reset or power sequence operation, the event is called a soft error.

**IMPACT OF SOFT ERRORS AT THE SYSTEM LEVEL**

A common unit to measure device reliability is the FIT (failure in time). 1 FIT is defined as 1 device fail per 10⁹ operating hours. (Note that for soft errors, an event and failure are synonymous, even though no device damage has occurred. A bit has been flipped and that is defined as a failure of the data even though there are architectural designs that can correct it.) A reciprocal unit of measure is mean time between event (MTBE). The two are related by

\[
\text{MTBE}_{\text{dev}} \ (\text{hr}) = \frac{10^9}{R_{\text{dev}}} \quad \text{(Equation 1)}
\]

where \( R_{\text{dev}} \) is device FIT. For a system with \( n \) devices of identical FIT values of \( R \), the system MTBE is

\[
\text{MTBE}_{\text{sys}} \ (\text{hr}) = \frac{10^9}{n \cdot R_{\text{dev}}} \quad \text{(Equation 2)}
\]

And for a system composed of different device, the system MTBE is

\[
\text{MTBE}_{\text{sys}} \ (\text{hr}) = \frac{10^9}{\sum n_i \cdot R_i} \quad \text{(Equation 3)}
\]

where \( n_i \) and \( R_i \) are the count and failure rate (in FIT) of device \( i \), respectively. Those devices with a (high part count x failure rate) product will dominate the MTBE of the system.

Not all events lead to system failure or downtime. How the system responds to these events is highly dependent upon the architecture. Most soft errors will come from either DRAM or SRAM memory due to the shear count \( n_i \) of those devices where interleaving and ECC techniques will virtually eliminate the soft error impact. Logic upsets are less frequent but difficult to protect against without resorting to high overhead techniques discussed later. However, not all logic upsets impact the outcome of a compute operation. Conversion of raw logic
soft error rates to system failure rates requires knowledge of architectural vulnerability factors (AVF) [6].

**Software**

Failures due to software reliability are dependent upon software maturity and bug fixes. By definition, software should not fail. There is nothing to break or wear out. But software does fail because of defects (or bugs) in the code. For newly developed software, bugs can dominate system fail rates compared to hardware reliability. As software matures and more bugs are fixed, it can become less significant for system reliability.

It might be tempting to compare software failure rates to soft error rates of components, since they are both events that can interrupt system operation. But it is more important to understand the role each of these failure mechanisms has on system performance and customer perception. Software failures are visible and frustrating to the customer. However, in most cases, the short term fix is to reboot the system and do an upgrade when a software patch is available with the bug fixed. In the case of soft fails, rebooting the system might eliminate the problem, but the soft fails will continue to occur at the same rate unless the components are replaced with parts that have a lower soft error rate. So the near-term fix is the same – reboot the system. But the long term fix is not. Therefore, it is important to determine if the system failure is from software or a soft error. The best techniques require detailed knowledge of the operating system and application codes. By careful examination of the error logs, one must determine if the faults could be generated by physical mechanism (e.g. bit flips in memory) or if the error logs are inconsistent and indicate a bug. In addition, soft errors are random in time and (address) location. Clustering of errors around specific locations or times would indicate other mechanisms are involved.

**DRAM**

From Figure 1, the cell upset rate of DRAMs are trending down with process technology. Cell upset rates are $< 10^{-8}$ FIT/bit for technology
65nm and below. A 1Gb DRAM chip will display a cell upset soft error rate of < 10 FIT/chip. But this is only part of the soft error picture for DRAM. Since cell upsets are trending downward and the on-board logic to support modern DRAM features is increasing, logic upsets are becoming more important. Figure 3 shows single event upset rate from neutrons (normalized to the neutron flux at NYC) vs. number of bits in error for 90 – 180nm DRAMs [7]. Bits in error from a single event of 16 or less are clusters of cell upsets. Bits in error from a single event in excess of 1000 represent neutrons hits on the control logic of the DRAM. Logic upset rates range from $10^{-10}$ to $10^{-8}$ FIT/bit. (This metric is somewhat misleading since it is logic circuits that are being upset and not cells. A more correct metric would be FIT/logic gate, but the logic gate count for the DRAMs from all the vendors in Figure 3 is not necessarily known. A reasonable assumption is that the number of logic gates scales with the density of the DRAM.) Some of these logic upsets are cleared by issuing new read or write instructions, but others affect the various state registers and can only be cleared by a re-initialization of the DRAM. While logic soft error rates are lower than cell upset rates, the number of bits corrupted per event is $10^3$ to $10^4$ times greater. The net effect is that the bit error rate is impacted by both cell and logic upsets. The bit error rate is a combination of the cell upsets and logic upsets:

$$B = \sum_{n=1}^{16} nR_n + \sum_{l=1000}^{10,000} lR_l$$

Equation 4

where n is the number of cells upset, $R_n$ is the n-cell upset rate, l is the number of bits corrupted in a logic upset and $R_l$ is the logic upset rate. Eq. 4 assumes the maximum cell upset cluster is 16 and logic upsets range from 1000 to 10,000 bits as shown in Figure 3. Figure 4 is a graph of the bit error rate for several different DRAM vendors as a function of design rule. The logic upset component of the bit error rate dominates the cell upset component in all cases. For a 1Gb DRAM chip at 65nm, the bit error rate is roughly 1000 FIT.
In the case of logic upsets, the entire output of a single DRAM chip can be in error - 4 bit errors for a x4 I/O DRAM or 8 bit errors for a x8 I/O DRAM. In addition, the entire burst output (burst length can be anywhere from 1 to 4), can be in error, leading to 16 to 32 bits in error. Fortunately, new server memory controllers feature chipkill [8] or single device data correction (SDDC) [9] designs that can correct these errors from a single DRAM chip.

FIG. 3 Single event upset rate vs. bits in error per event for three different DRAM process nodes. The events from 1 to 16 bits are single cell and multi-cell upsets. The events in excess of 1000 bits are actually upsets of the DRAM logic circuitry.
FIG. 4 DRAM bit error rate (as defined in Eq. 4) vs. design rule for six different vendors. Black curves are logic component of bit error rate and red curves are cell upset component.

**SRAM**

From Figure 1, a reasonable estimate of SRAM cell upset rates is $10^{-3}$ FIT/bit. The predominant amount of SRAM soft errors are single cell upsets. There is some data that suggests multi-cell upsets are becoming more dominant below 65nm. However, there is no indications that the SRAM bit error rate (i.e. soft error rate x number of bits upset) will be skewed the way DRAMs are. So for all practical purposes, the soft error rate and bit error rate for SRAMs are roughly the same.

CPU cache sizes typically range from 0.5 to 24MB. This corresponds to a soft error rate of 500 to 24,000 FIT per chip. While this number is quite high, much of the caches are protected with some form of error correction code.

**Logic**

For soft error estimates, logic can be broken into two categories – combinational and sequential. Combinational logic SER increases
linearly with frequency, whereas sequential logic is independent of frequency. Recent work at the 32nm process technology node indicates that the chip-level SER contribution of combinational logic is well below that of the sequential SER [10]. Since the trend to increase logic performance is no longer to increase frequency but rather to increase the number of cores, it is not likely that combinational logic SER will dominate sequential logic SER.

Logic elements follow design rules that are more relaxed than SRAM cells. This is because the chip area of modern CPU circuits is dominated by cache. Being more aggressive with SRAM geometries has a bigger impact on die size. With these more aggressive design rules comes more device marginality in terms of the charge required to flip the SRAM ($Q_{\text{crit}}$). Therefore, logic circuits will follow the SRAM SER trend shown in Figure 1, but will probably be an order of magnitude lower – $10^{-5}$ to $10^{-4}$ FIT/gate would be a good rule of thumb.

**Summary**

Table 1 summarizes the relative soft error rates of DRAM, SRAM and logic in a server system assuming a relative density of memory and logic elements. It gives a rough idea of how soft error events will be distributed, but actual system failure rates will depend upon how well each sub-system can detect and respond to these events.

<table>
<thead>
<tr>
<th>Device</th>
<th>Relative Density (bits or gates)</th>
<th>FIT / bit (or gate)</th>
<th>Total Relative Soft Error FIT</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOGIC</td>
<td>1</td>
<td>$10^{-4}$</td>
<td>$10^{-4}$</td>
</tr>
<tr>
<td>SRAM</td>
<td>10</td>
<td>$10^{-3}$</td>
<td>$10^{-2}$</td>
</tr>
<tr>
<td>DRAM - cell</td>
<td>$10^5$</td>
<td>$10^{-8}$</td>
<td>$10^{-3}$</td>
</tr>
<tr>
<td>DRAM - logic</td>
<td>NA</td>
<td>NA</td>
<td>$10^{-1}$</td>
</tr>
</tbody>
</table>

Table 1. Relative soft error rates of various system components. The impact of DRAM logic upsets are assumed to be 100x greater than cell upsets as discussed in the text.
CONCLUSIONS – What soft errors to tackle first?

There are many techniques to achieve high reliability, availability and serviceability (RAS) with respect to soft errors:

- various levels of error detection and correction codes
- temporal filtering of transient pulses from neutron or alpha strikes
- spatial circuit redundancy – triple modular redundancy (TMR), dual interlocked storage cell (DICE)
- hardening of SRAM cells by adding extra holding capacitance
- redundant or radiation hardened clock nodes to prevent clock jitter from neutron or alpha strikes
- memory mirroring

Extremely low system FIT is possible at great expense. For example, achieving high RAS on an IBM z900 series server is possible through use of highly customized IC designs and redundancy [11]. For optimization of system design where low cost is a driving factor and must be balanced against RAS features, it pays to work first on the sub-systems that are the largest soft error FIT contributors.

Fortunately, main memory and cache memory have the highest soft error rates and can be mitigated by techniques that require a minimum amount of overhead. Since ECC overhead decreases as word size increases, large word structures can afford the more powerful multi-bit error correction codes. This reduces the threat of DRAM logic upsets leading to system crashes. The use of simpler SEC-DED codes combined with interleaving (to prevent multi-cell errors from becoming multi-bit errors) is effective for large SRAM caches. Smaller memory arrays that have back-up copies can simply use parity protection. If an error is detected, the corrupted data can be flushed and a clean copy retrieved. These techniques are more cost effective than memory mirroring or hardening of cells.

For soft error protection of logic, the choices (i.e. temporal filtering and clock hardening) create much higher overhead unless they are implemented on a very selective basis. However, since the anticipated logic soft errors are orders of magnitude below memory soft errors, it is
likely that the system RAS goals can be taken care of by addressing all the memory soft error mechanism first.

REFERENCES


[9] Intel® 7300 Chipset Memory Controller Hub (MCH) Datasheet, September 2007. Available online at
http://www.intel.com/Products/Server/Chipsets/7300/7300-technicaldocuments.htm
