A Technique to Calculate the MBU Distribution of a Memory under Radiation Suffering the Event Accumulation Problem

Pedro Reviriego and Juan Antonio Maestro

Abstract— **When radiating memories, SEUs/MBUs tend to accumulate, making difficult to characterize the number of events and error patterns that have affected the system. A technique to deal with this problem is presented in this paper.**

Index Terms— **Multiple Bit Upsets (MBUs), reliability, memories.**

I. INTRODUCTION

As technology advances, the effect of radiation on microelectronic circuits is becoming more important [1][2]. Memories, due to their broad use and large area, are specially sensitive to radiation effects [3][4]. That is especially true for Multiple Bit Upsets (MBUs) [5][6], whose impact accounts for a growing number of effects. Therefore, one critical issue for manufacturers is to categorize how a certain memory behaves within different radiation environments and study the sources of potential errors.

In this situation, it is important to characterize the distribution of the different events induced in the system. That is to say, how many errors have been generated in the memory by radiation, how many were SEUs and how many MBUs, how many errors were produced by each MBU, the topology of these MBUs (the error patterns), etc.

However, calculating this information is not a trivial problem. A memory radiated for a long time or with high energies presents a large number of errors, which would tend to group and even overlap. Obtaining merged error patterns would make difficult to differentiate them and find out the number of events that caused them. In order to deal with this event accumulation problem, a technique is presented in this paper to compute the number and pattern of SEUs/MBUs that have affected a memory system.

In Section II, the problem of the error-per-event calculation is presented; in Section III and IV, two different experiment scenarios are discussed and a solution for the problem is presented; some results are offered and commented in Section V; finally, conclusions and future work are presented in Section VI.

II. PROBLEM DEFINITION: ERROR-PER-EVENT CALCULATION

As it was mentioned in the previous section, the goal of this paper is to propose a method to calculate the MBU distribution (number and topology) produced in a certain memory by a radiation source. In this way, the characterization of the technology in that particular environment can be performed.

Formally, let us use the term "error-per-event distribution" to refer to this MBU distribution, and let us denote it with the set $\{p(1),p(2),p(3), \ldots, p(\mathrm{n})\}$, being $p(n)$ the probability that an event with n errors has happened. In this way, $p(1)$ would be the probability that a SEU happens (only one error per event), $p(2)$ the probability that a 2-error MBU happens, etc. Calculating this distribution is very useful to foresee how a certain technology would behave in a given radiation environment and the probability that this technology is affected by multi-bit upsets.

One way to achieve this would be to perform a radiation experiment on a memory (using a well-characterized source), and then to examine the final memory map. By comparing this final map with the initial data, the set of erroneous positions (affected by radiation) can be determined. However, the problem lies in estimating the nature of the events that produced those errors. This is because it is not possible to rebuild how events arrived in the memory, and therefore, if the final map shows e.g. 4 errors in adjacent positions (Figure 1.A), this could be due to 4 SEUs or to a 4-error MBU (or any other combination). This event accumulation problem has a very direct impact in the error-per-event distribution.

A possibility to perform this, addressed in [7], would be as follows: The experiment should be real-time controlled using a monitor platform, in such a way that periodical read-out/correct processes are performed in the memory. After the experiment starts, the first read-out process should happen at a pre-determined instant. In this process, the whole memory map is read and possible errors detected. If no errors, the experiment would continue until the next read-out process. If in one of these processes errors are detected, the topology of the errors is annotated and cleaned immediately afterwards. Then, when the experiment continues, the whole memory is correct again. If we can guarantee that the read-out processes are so frequent that only one event happens between every pair of them, then n errors observed at a certain instant would mean an n-error MBU, since events do not accumulate in the memory.

This work was supported by the Spanish Ministry of Science and Education under Grant ESP-2006-04163.

P. Reviriego and J.A. Maestro are with Universidad Antonio de Nebrija,C/Pirineos, 55 E-28040 Madrid, Spain (phone: +34 914521100; fax: +34 914521110; email: {previrie,jmaestro}@nebrija.es).

However, there may be situations in which the previous experimental methodology cannot be used. One possible scenario are environments where a real-time monitoring system is not feasible. For example, experiments to characterize radiation in remote areas, in-flight tests, etc. In this case, only the final memory map would be available, making the error-per-event calculation unfeasible. Another possible scenario are environments where a real-time monitoring system is feasible, but where it is not possible to guarantee the minimum read-out frequency so that events do not accumulate in the memory. This problem is also addressed in the literature [8], and it can be due to a very high event arrival rate (very intense radiation), which would make the read-out frequency unfeasible. In the next sections, both scenarios will be studied, and a methodology to deal with the problem of event accumulation will be described.

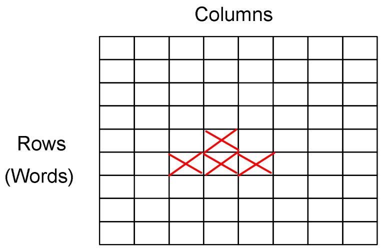

Figure 1.A: False MBU example

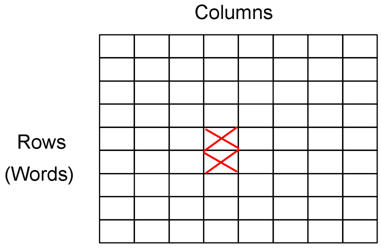

Figure 1.B: False MBU example

III. ERROR PER-EVENT CALCULATION WITH NO READ-OUT PROCESSES

In this section, a methodology to calculate the error-per-event distribution in an environment where no read-out processes are possible will be presented.

Since in this case only the final memory map is available, a methodology to deduce how the different errors came grouped into MBUs is required. Let us consider the memory map depicted in Figure 1.B, where the positions marked with an "x" denote erroneous positions affected by radiation. The first thought would be to think that the memory has been affected by a double MBU. However, it could also be a "false" MBU, due to 2 independent SEUs. Depending on the case, we would have a different MBU count for the error-per-event distribution: in other words, the observed MBU distribution does not have to be the real error-per-event distribution.

Having defined in the previous section the p(n) distribution (the probabilities of the different types of events), let us now define $\{p^*(2)|^k, p^*(3)|^k, \ldots, p^*(\mathrm{n})|^k\}$, being $p^*(n)|^k$ the probability of obtaining a false n-bit MBU (formed in reality by n independent SEUs) after k events in the memory. In this way, the observed MBU distribution would be formed by the actual p(n) plus the set $p^*(n)|^k$. Therefore, if we quantify the latter, we would be able to obtain the right value of p(n).

In order to do this, let us study, from a probabilistic point of view, how likely the obtention of a false MBU is. In the simple case of 2 adjacent errors (Figure 1.B would be one of the possibilities of this case), the second SEU needs to have hit the adjacent area of the first one. If the memory has M words of N bits, this probability would be $8/(M\cdot N)$, since there are 8 neighbors per bit position (for positions that are along the edges, this number is lower, but these cases are negligible if the memory is large enough). This would determine the probability that a second SEU creates a false MBU with a previous one. In the case that there were two isolated errors, the probability that a third SEU creates a false 2-bit MBU with any of the two previous would be $(8\cdot 2)/(M\cdot N)$, since now there are already two candidate erroneous positions to group with. Generalizing, the probability that the i-th SEU hits a position adjacent to any of the i-1 previous events (to form the false 2-bit MBU), can be bounded by (assuming $i << M\cdot N$):

$$p_{adj}(2)\big|^{i} \le \frac{8\cdot(i-1)}{M\cdot N} \tag{1}$$

Then, if an experiment with k events is designed, the average number of false 2-bit MBUs obtained in the system, $f(2)|^k$ would be:

$$f(2)\big|^{k} \le \sum_{i=2}^{k} p_{adj}(2)\big|^{i} = \frac{4\cdot k\cdot(k-1)}{M\cdot N} \tag{2}$$

And therefore, the probability of false 2-bit MBUs after k events (that is to say, $p^*(2)|^k$), will be:

$$p^*(2)\big|^{k} \le \frac{f(2)\big|^{k}}{k} = \frac{4\cdot(k-1)}{M\cdot N} \tag{3}$$

Analyzing expression (3), if k is much smaller than $M\cdot N$ (memory size), then $p^*(2)|^k$ will tend to 0. In other words, the effect of false MBUs is negligible, being the observed MBU distribution the actual error-per-event function.

Let us define the error margin of the experiment, e, as the fraction of MBUs that we tolerate to be false. Then, the maximum number of events that we can generate in the memory, according to (3), is:

$$p^*(2)\big|^{k} \le e \Rightarrow e \ge \frac{4\cdot(k-1)}{M\cdot N} \Rightarrow k \le \frac{e\cdot(M\cdot N)}{4}+1 \tag{4}$$

In this way, k is the threshold for the number of events: below that threshold the amount of false 2-bit MBUs is smaller than e. Therefore, the observed MBU distribution in the final memory map would be in fact the actual error-per-event distribution (with the error tolerance e). The same derivation could be done to characterize the rest of $p^*(n)|^k$ for $n>2$. However, if k is such that the effect of false 2-bit MBUs is negligible, the effect of MBUs with more bits will be even smaller.

At this point, two alternatives may be followed:

1) To keep the number of events below the value of k given by (4). That would imply that there is a constraint on the number of events. If the event arrival rate per word, λ, is known for the radiation source, that would imply that the experiment should last less than $k/(\lambda\cdot M)$ units of time. For

larger number of events, the weight of false MBUs would be noticeable, distorting the observed distribution.

2) A second possibility would be to use a larger number of events (radiation time). The error-per-event distribution is based on probabilities, which will work better for larger data samples, and sometimes a very high number of events may be required. In this case, we should have to eliminate the effect of the calculated $p^*(n)|^k$ from the observed distribution, in order to get the right value. How to compensate this effect is not trivial, and will not be explained in this paper, due to the space limitations.

Both approaches would give the right error-per-event distribution just studying the final memory map. In the next section, the scenario with read-out processes will be discussed.

IV. ERROR PER-EVENT CALCULATION WITH READ-OUT PROCESSES

As mentioned previously, one way to calculate the error-per-event distribution in a radiation experiment would be to perform numerous read-out processes [7][9], guaranteeing that no more than one event is generated in the system in each cycle. This last constraint presents some problems. The first problem is that the read-out frequency needed to avoid event accumulation may be unfeasible [8]. This is especially likely in intense radiation environments, where λ is high enough. The second problem is that even if the read-out frequency is feasible, performing too many of these processes implies handling a large amount of data, or even a delay in the experiment.

Therefore, we propose to apply the method described in section III to this scenario, what would lead to an important reduction in the read-out frequency. In this way, expression (4) determines the maximum number of events, k, to allow reconstruction of the error-per-event distribution based on the memory map. This means that the maximum number of events allowed in a single read-out cycle is k (instead of very few, as in the traditional method). If the event arrival rate of the experiment is λ, then we could allow a read-out period of up to $k/(\lambda \cdot M)$ units of time. After each cycle, the intermediate memory map would be studied, the error-per-event distribution would be updated, and the radiation process would continue after correcting the existing errors. With this, the event accumulation would be kept bounded and will not pose any problem to the error-per-event calculation.

With this method, *the total number of events is unbounded.* Once k is reached, the readout process cleans the memory, and the event count can grow. At the same time, the number of read-out cycles (frequency) is optimized, performing only the minimum number required to get accurate results.

V. EXPERIMENTAL RESULTS

In this section, the experimental work to put in perspective the presented technique will be described. The behavior of a memory with M = 512 K words and N = 12 bits has been simulated for a radiation associated to a constant event arrival rate per word λ = 1/400. In this way, the average number of events is a linear function of the radiation time (and vice versa). Notice that the value of λ would depend both on the memory technology and the radiation nature, and therefore it is not necessary to explicitly characterize them.

The purpose of the experiment is to determine how the number of false MBUs evolves with radiation time, and whether they are kept under the error tolerance e when the maximum number of events (radiation time) k is not reached. To perform the experiment, a value of e=0.001 has been chosen, which corresponds to a radiation time $t_s = k/(\lambda \cdot M)$ = 1.3 units of time (approx.). It can be seen in Figure 2 that while the radiation time is below that threshold, the probability of false 2-bit MBU is always below the selected e level. This means that since e is small, these false MBUs will have a negligible weight in the overall count of observed MBUs, and therefore these observed MBUs will mostly correspond to real MBUs (which would lead to the calculation of the per-event distribution $p(n)$). On the other hand, when k is exceeded, the false MBUs are higher than e, meaning that they will have a noticeable impact on the $p(n)$ calculation. Another thing to remark is when the probabilities of 3-bit MBUs and higher ($p(n)$ for $n \geq 3$) are smaller than $p(2)$, then the probabilities of false MBUs, $p^*(n)$ for $n \geq 3$ are also below e, as it can be seen in Figure 2.

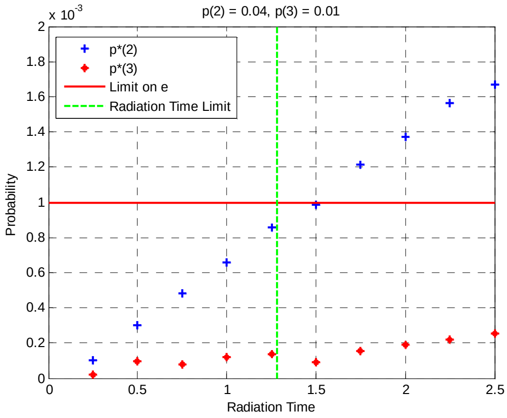

Figure 2: Probability of false MBUs in the presence of real MBUs

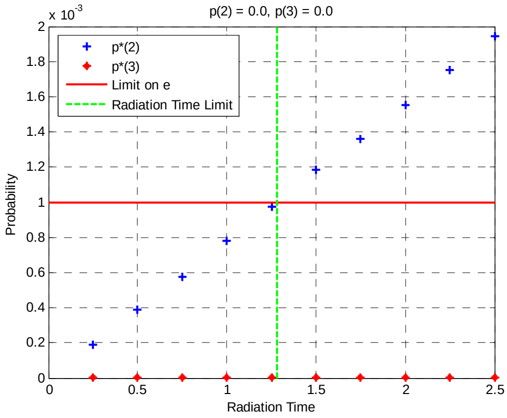

Figure 3: Probability of false MBUs in the absence of real MBUs

In Figure 3, the same experiment is reproduced, but now in the case that no real MBUs are present (only SEUs). It can be seen that the result trend is similar to the one in Figure 2. However, the value of the calculated false MBUs ($p^*(2)$ and $p^*(3)$) are more accurate than in the previous case (Figure 3). This is because due to the absence of real MBUs, there are no side effects for complex combinations of them (e.g. a 2-bit MBU combined with an SEU to form a false 3-bit MBU). This can be directly seen in Figure 3: the probability $p^*(3)$ is nearly 0 as expected (since there are no MBUs, the combination of 3 SEUs is needed to produce a false 3-bit MBU, which is unlikely). In the same way, the probability of $p^*(2)$ is also more accurate: the break point (where $p^*(2)$ meets the value of e) is exactly reached at the time threshold marked by k. However, although the first experiment is less accurate, the model is robust enough to keep all $p^*(n)$ under the e level when the radiation time is not exceeded (Figure 2).

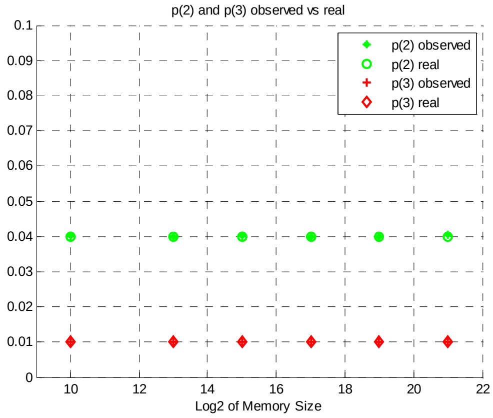

Figure 4: MBU probability for a number of events below the threshold

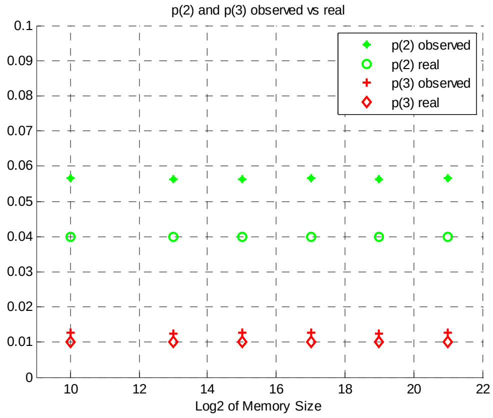

Figure 5: MBU probability for a number of events over the threshold

Finally, the quality of the calculated per-event distribution, $p(n)$, has been measured for the first experiment described in this section (the second experiment has trivial results, since $p(2)=p(3)=0$). Different memory sizes have been tested in order to verify that the method is size-independent for $k >> 1$. The observed values of $p(2)$ and $p(3)$ have been compared with the real ones in order to check if the radiation time limit effectively removes the presence of false MBUs. In Figure 4, the results are offered when the radiation time is below the threshold defined by k. It can be seen that the observed MBUs coincide with the real ones for all memory sizes, what implies that no false MBUs were accumulated in the memory. On the other hand, the same results are depicted in Figure 5, but now exceeding k. It can be seen that the real and observed $p(n)$ differ due to the false MBUs that have started to accumulate in the memory.

VI. CONCLUSIONS

In this paper, a method to derive the error-per-event distribution (MBU distribution) on a memory radiated with a certain source has been studied. The method presents two main contributions:

- It allows experiments with some degree of event accumulation, what enables larger radiation times (number of events). To deal with false MBUs, the probability of occurrence is derived and used to determine the maximum number of events in a single experiment.
- Even if the number of events needs to be unbounded, then read-out processes may be introduced. The method determines the optimal frequency to achieve this, avoiding unnecessary high read-out frequencies and large data manipulation.

About the future work, the second possibility mentioned in section III will be explored: instead of bounding the number of events using (4) or introducing read-out cycles, the effect of false MBUs may be estimated and subtracted from the observed MBU distribution, allowing more events per experiment. This calculation becomes complex, because when the number of events is high enough (compared to the memory size), other side effects similar to the ones described in section V appear and need to be considered (for example the probability of MBUs overlapping with previous ones).

REFERENCES

[1] R. D. Schrimpf, D. M. Fleetwood, “Radiation effects and soft errors in integrated circuits and electronic devices”, World Scientific Publishing, 2004. ISBN: 981-238-940-7.

[2] P. E. Dodd, LL. Massengill, “Basic Mechanisms and Modeling of Single-Event Upset in Digital Microelectronics”, IEEE Trans. on Nuclear Science, Vol. 50, No 3, June 2003, pp. 583 – 602.

[3] T. Sasada, S. Ichikawa, T. Kanai, “Measurement of Single-Event Effects on a Large Number of Commercial DRAMs”, IEEE Trans. on Nuclear Science, Vol. 53, Issue 4, Part 1, Aug. 2006, pp.1806 – 1812.

[4] R.M. Goodman, M. Sayano, “The reliability of semiconductor RAM memories with on-chip error-correction coding”, IEEE Trans. on Information Theory, Vol. 37, Issue 3, Part 2, May 1991, pp. 884-896.

[5] A. D. Tipton et al., “Multiple-Bit Upset in 130 nm CMOS Technology” IEEE Trans. on Nuclear Science, Volume 53, Issue 6, Part 1, Dec. 2006, pp. 3259 – 3264.

[6] A.M. Chugg, M.J. Moutrie, R. Jones, “Broadening of the variance of the number of upsets in a read-cycle by MBUs”, IEEE Trans. on Nuclear Science, Volume 51, Issue 6, Part 2, Dec. 2004, pp. 3701 – 3707.

[7] A.M. Chugg, M.J. Moutrie, A.J. Burnell, R. Jones, “A Statistical Technique to Measure the Proportion of MBU's in SEE Testing”, IEEE Trans. on Nuclear Science, Vol. 53, Issue 6, Part 1, Dec. 2006, pp. 3139 – 3144.

[8] M.A. Bajura, Y. Boulghassoul, R. Naseer et al., “Models and Algorithmic Limits for an ECC-Based Approach to Hardening Sub-100-nm SRAMs”, IEEE Trans. on Nuclear Science, Vol. 54, Issue 4, Part 2, Aug. 2007, pp. 935 – 945.

[9] K. Johansson, M. Ohlsson, N. Olsson, J. Blomgren, P.-U. Renberg, “Neutron induced single-word multiple-bit upset in SRAM”, IEEE Trans. on Nuclear Science, Vol. 46, Issue 6, Dec 1999, pp. 1427 – 1433.