Statistical Deviations from the Theoretical only-SBU Model to Estimate MCU rates in SRAMs

Francisco J. Franco, Juan Antonio Clemente, Maud Baylac, Solenne Rey, Francesca Villa, Hortensia Mecha, Juan A. Agapito, Helmut Puchner, Guillaume Hubert, and Raoul Velazco

Abstract

This paper addresses a well-known problem that occurs when memories are exposed to radiation: the determination if a bitflip is isolated or if it belongs to a multiple event. As it is unusual to know the physical layout of the memory, this paper proposes to evaluate the statistical properties of the sets of corrupted addresses and to compare the results with a mathematical prediction model where all of the events are SBUs. A set of rules easy to implement in common programming languages can be iteratively applied if anomalies are observed, thus yielding a classification of errors quite closer to reality (more than 80% accuracy in our experiments).

Index Terms

Multiple cell upsets, single bit upsets, single events, soft errors, SRAMs

I. Introduction

THE evolution of electronic technology has pushed transistors sub-10-nm dimensions. Thus, charge sharing between adjacent memory cells has gained in importance to understand the multiple cell upset (MCU) events. In general, the occurrence of MCUs in *Static Random Access Memories* (SRAMs) has done alike. In old generation SRAMs, bits belonging to the same data word were placed adjacent to each other, so multiple errors involving several bitflips in the same data word were likely to occur (*Multiple Bit Upsets*, MBUs). Since they cannot be corrected by standard mitigation techniques such as Error Correcting Codes (ECC), manufacturers started to use bit interleaving. The objective is to prevent multiple upset events in the same address (*Multiple Cell Upsets*, MCUs), and thereby enabling a simple Hamming code ECC [1] to recover from soft error events.

It is mandatory to distinguish between *Single Bit Upsets* (SBUs) and MCUs. The reason is that it is crucial to understand the nature of the effects provoked by radiation on modern devices. In general terms, having a deep understanding of said effects is imperative to design effective protection mechanisms against them. Thus, if all the observed errors are counted as SBUs but MCUs occur in the devices, the sensitivity of the device against radiation can be significantly overestimated. It can certainly impede designers from implementing adequate mitigation techniques to address this problem.

This is a challenge for researchers as, unless knowing the SRAM structure, it is not possible to relate logic addresses with physical ones. Unfortunately, this information is usually protected by the manufacturers. Some techniques have been developed to obtain bit maps from laser screening [2] but they require an appropriate decapsulating of the SRAM samples. Other authors irradiate devices with very low flux to separate SBUs from MCUs [3], but this approach may take too much beam time to get

This work was supported in part by the Spanish MCINN projects TIN2013-40968-P and FPA2015-69120-C6-5R, by UCM-BSCH, and by the "José Castillejo" mobility grant for professors and researchers.

F. J. Franco and J. A. Agapito are with the Departamento de Física Aplicada III, Facultad de Físicas, Universidad Complutense de Madrid (UCM), Spain, e-mail: fjfranco@fis.ucm.es, agapito@fis.ucm.es.

J. A. Clemente and H. Mecha are with the Computer Architecture Department, Facultad de Informática, Universidad Complutense de Madrid (UCM), Spain, e-mail: ja.clemente@fdi.ucm.es, horten@fis.ucm.es.

M. Baylac, S. Rey, and F. Villa are with Laboratoire de Physique Subatomique et de Cosmologie LPSC, Université Grenoble-Alpes & CNRS/IN2P3, Grenoble, France, e-mail: baylac@lpsc.in2p3.fr, solenne.rey@lpsc.in2p3.fr, francesca.villa@lpsc.in2p3.fr.

H. Puchner is with Cypress Semiconductor, Technology R&D, 3901 San Jose, CA, USA. e-mail: hrp@cypress.com

G. Hubert is with the French Aerospace Laboratory (ONERA), Toulouse, France, e-mail: guillaume.hubert@onera.fr.

R. Velazco is with the Université Grenoble-Alpes & CNRS, TIMA, Grenoble (France), e-mail: raoul.velazco@univ-grenoble-alpes.fr.

significant amount of data. Therefore, several strategies based on statistical anomalies in the set of addresses affected by the radiation have been developed to group addresses in MCUs from mathematical properties.

In a previous work [4], the authors demonstrated the feasibility of the detection of addresses involved in multiple events using the XOR operation, firstly proposed by Falguere et al. [5]. The affected addresses were combined in pairs, XORed and finally, the values that appeared more often than expected from random events were selected. The key point was that the XOR operation takes advantage of the modularity of SRAMs [6], [7]. This constituted a great difference from previous works used as inspiration for the method presented here [8], [9], where the addresses were subtracted rather than XORed.

In this work, the study done in [4] has been generalized for any operation (XOR, subtraction, etc.) and an algorithm is proposed to find memory addresses involved in multiple events. Also, the the human dependency is minimized and only a threshold value, ϵ_R, must be chosen to set a boundary between randomness and causality. Additionally, it is necessary that the number of affected cells is much lower than the total number of cells. This avoids adjacent flipped cells from different events.

This work is an extension of a paper presented at the RADECS conference in 2016 [10]. That paper discussed the pertinence of the use of the positive subtraction ($P.S.$ in the remainder of the manuscript), to identify MCUs. This work further extends that idea, explores the possibility of using both XOR and $P.S.$, incorporates new rules for MCU extraction (depicted in Section IV) and presents a new systematic methodology to detect MCUs from the bulk of SEUs in a radiation test campaign. Part of the presented experimental results analyze new data not published elsewhere.

II. Mathematical Foundations

The strategy to characterize systems with MCUs requires a good understanding of the opposite: An ideal system where only SBUs occur. Let us suppose that we are checking an SRAM with N address bits and datawidth of N_W bits. Let an experiment consist in writing a known pattern in the SRAM, irradiating, and finally reading the content from the lowest address (0) to the highest one ($L_N = 2^N - 1$). This experiment is supposed to yield N_E affected addresses. Two facts are postulated: First, only SBUs can occur. Secondly, addresses with bitflips are randomly distributed between 0 and L_N and they are equally probable. Thus, a set of N_E elements is obtained, not repeated and distributed between 0 and L_N.

From the set of addresses, a new set, called "Difference Vector", DV, can be created combining addresses in pairs and operating with the bitwise XOR, $P.S.$, etc. Several mathematical properties can be deduced but, due to their technical content, they are presented in the Appendix. The most important properties elaborated in said appendix are the number of elements in DV, (Eq. 8), and the expected number of repeated elements in this set (Eqs. 11-12).

III. Validation of the only-SBU system model

A. Monte Carlo tests

A Monte Carlo study[1] was performed to validate the ideas introduced in Section II and developed in the appendix. 100 addresses were randomly selected 1000 times from a pool of 2^{21} values to generate DVs of 4950 elements using the XOR ($XORDV$) and the $P.S.$ ($PSDV$). The mean values ($\bar{x}$) and standard deviations (σ) of all the DVs were evaluated. Then, the mean values of the 1000 $\bar{x}$'s and σ's were calculated. The results are shown in Table I, which confirm the good agreement between theory (see Appendix, Section A, second-to-last paragraph) and simulations.

Another interesting property that can be studied is the number of 1's contained in the elements of DV when these are written in binary (called "*trace*" in [4]). The physical meaning of this parameter is further described in Section IV-C. It is possible to determine that, if the $P.S.$ is used, the expected number of elements in DV containing m ones is:

$$N_{1,PS}\left(m, N, N_{DV}\right) \approx \frac{N_{DV}}{2^{N-1}} \cdot \left(\begin{array}{c} N-1 \\ m \end{array} \right) \tag{1}$$

The proof is too long to be included in this paper. It is worth to indicate that, in the case of using the XOR [4], the equivalent result is:

[1]Calculations exposed throughout the manuscript were performed in the Julia Language [11] for speed, efficiency, comprehension, and portability.

Table I
COMPARISON BETWEEN MONTE CARLO SIMULATIONS AND THEORETICAL PREDICTIONS IN AN ONLY-SBU SCENARIO

	$XORDV$		$PSDV$	
	Meas.	Theor.	Meas.	Theor.
$\bar{x}/L_N$	0.4998...	0.5	0.3332...	0.3333...
σ/L_N	0.2887...	0.2887...	0.2349...	0.2357...
Elements appearing...				
Once	4938.24	4938.33	4934.49	4934.39
Twice	5.878	5.827	7.739	7.760
Three times	0.001	0.005	0.01	0.009

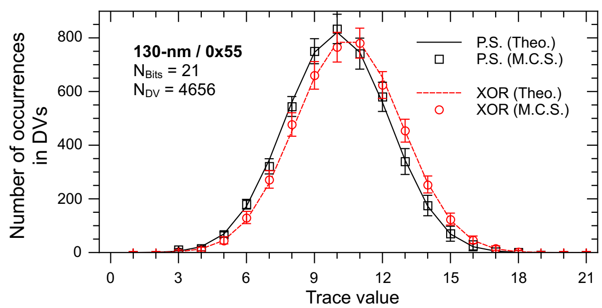

Figure 1. Average number of occurrences of elements with k 1's in binary format for different DVs (XOR and $P.S.$) after 1000 simulations. Error bars are also included. The curves show the theoretical predictions (Eqs. (1) and (2)), whereas the dots and circles represent the Monte-Carlo simulations.

$$N_{1,XOR}(m,N,N_{DV}) = \frac{N_{DV}}{2^N-1} \cdot \begin{pmatrix} N \\ m \end{pmatrix} \quad (2)$$

Fig. 1 compares the trace predictions (Eqs. 1-2), in an environment where no MCUs occur, with the simulation results. The agreement is almost perfect. The overrepresentation of DV elements in a real environment where MCUs occur will be discussed in Section IV.

B. Results from radiation test experiments

This subsection validates the predictions of the only-SBU model presented in the previous subsection with experimental data issued from radiation ground tests.

In a previous work [4], two SRAMs manufactured by Cypress Semiconductor, in 90 & 130 nm CMOS technologies (CY62167EV and CY62167DV, both with $N = 21$, $N_W = 8$), were irradiated with 14-MeV neutrons at the GENEPI2 facility, located at the *Laboratoire de Physique Subatomique et Cosmologie*, in Grenoble (France) [12], [13]. These memories were tested with different patterns (0x00, 0x55, 0xFF) to obtain more than 100 bitflips in each reading round. Data were then classified into SBUs/MCUs by using proprietary information from the manufacturer (Table II). The criteria consisted in grouping in the same MCU those events that were located at a Manhattan distance [14] lower than 5. These MCUs were removed from the set of errors, and hence, the remaining addresses are supposed to be constituted by SBUs. These SBUs were used to build the two DV sets for $P.S.$ and XOR. Table III compares the actual values of $\bar{x}$ and σ of the elements of both sets vs. the theoretical predictions, both in good agreement with each other.

It is also interesting to investigate the relative abundance of trace values in the DV sets. This is shown in Fig. 2, which analyzes the data obtained from the 130-nm memory with 0x55 pattern, which is the largest set in Table III. Eqs. 1-2 are in perfect agreement with the experiments. It is worth to point out the fact that Eq. 2 has been used by some authors [9] as an approximation for the $P.S.$. Fig. 2 demonstrates that this approximation is close to the experimental results but not as accurate as Eq. 1.

The last statistical parameter, but the most important one for practical applications, is the expected number of elements repeated m times in the DVs for only-SBUs scenarios ($N_{R,PS}$, Eq. 11). Table IV shows the results of studying the three

Table II
EVENTS OBSERVED IN PREVIOUS EXPERIMENTS, CLASSIFIED BY USING UNSCRAMBLING PROPRIETARY INFORMATION FROM CYPRESS

				MCU size					
Tech.	Pattern	N_E	N_{DV}	SBU	2b	3b	4b	5b	6b
90 nm	0x00	131	8515	92	12	1	3	0	0
	0x55	120	7140	86	12	2	1	0	0
	0xFF	108	5778	81	9	3	0	0	0
130 nm	0x00	115	6555	62	10	5	2	2	0
	0x55	146	10585	97	13	3	2	0	1
	0xFF	129	8256	81	13	2	4	0	0

Table III
STATISTICAL DATA OF THE STUDIED DV SETS IN L_N UNITS, FOR THE EXPERIMENTS OF TABLE II, AND REMOVING THE MCUS (TABLE II IN [10])

			XOR		$P.S.$	
Technology	Pattern	SBUs	$\bar{x}$	σ	$\bar{x}$	σ
90 nm	0x00	92	0.503	0.288	0.345	0.242
	0x55	86	0.499	0.285	0.330	0.235
	0xFF	81	0.502	0.286	0.342	0.240
130 nm	0x00	62	0.506	0.290	0.368	0.259
	0x55	97	0.500	0.287	0.337	0.238
	0xFF	81	0.505	0.293	0.365	0.259
Theoretical			0.500	0.289	0.333	0.236

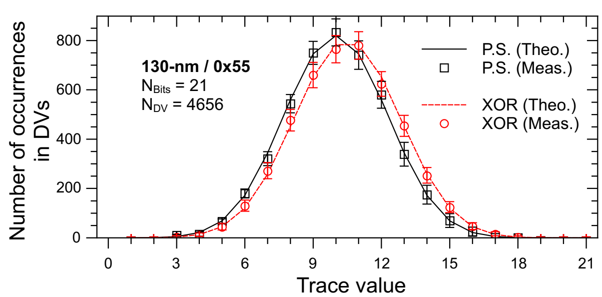

Figure 2. Histogram of elements for different trace values in the $PSDV$ set ($N_{DV} = 4656$, Eq. 10), for an only-SBU scenario. Error bars were calculated by means of the inverse-χ^2 function with 95%-confidence [15].

experiments of Table III that issued the highest numbers of SBUs. According to the study, most of the DV values (in the following, named $PSDV$ values) appeared only once and some of them twice. No values appeared three or more times. From the number of occurrences it is possible to determine the range where the expected number of values is with 95%-confidence (also shown in Fig. 2) and observe that they are in agreement with the theoretical predictions issued from Eq. 11. Similar conclusions can be drawn from Table V and Eq. 12, where XORing was used to generate the DV (in the following, named $XORDV$ values).

IV. RULES PROPOSED TO EXTRACT ANOMALOUS DV VALUES AND TO IDENTIFY MCUS

As the mathematical foundations described in Section II and in the Appendix are appropriate to describe a scenario with SBUs, deviations with respect to the predictions observed in the experiments are hints of the MCU presence.

In this paper, we postulate that an observed phenomenon did not happen by chance if the predicted number of occurrences is lower than $\epsilon_R = 0.05$. This threshold is based on the standard 95%-confidence used in many fields of physics, and it means that the phenomenon occurs in 1 experiment out of 20. The value of ϵ_R is a choice made by the authors.

A. Excessive repetitions and self-consistency

Let us assume that we have irradiated a memory and reported bitflips in N_E different addresses. This set of addresses has been used to create DV sets for XOR and $P.S.$ and evaluates the number of times that every value between 0 and L_N is

Table IV
REPETITION OF ELEMENTS IN SEVERAL ONLY-SBUS DV SETS WITH $P.S.$ (TABLE III IN [10])

	N_{DV}	Rep.	Occur.	95%-Conf.	Theo.
130 nm 0x55	4656	1	4632	4496-4768	4642.2
		2	12	6.2-20.9	6.87
		3	0	0-3.7	0.008
90 nm 0x00	4186	1	4178	4049-4307	4174.8
		2	4	1.1-10.2	5.55
		3	0	0-3.7	0.005
90 nm 0x55	3655	1	3651	3530-3772	3646.5
		2	2	0.2-7.2	4.23
		3	0	0-3.7	0.004

Table V
REPETITION OF ELEMENTS IN SEVERAL ONLY-SBUS DV SETS WITH XOR

	N_{DV}	Rep.	Occur.	95%-Conf.	Theo.
130 nm 0x55	4656	1	4644	4507-4780	4645.7
		2	6	2.2-13.1	5.16
		3	0	0-3.7	0.004
90 nm 0x00	4186	1	4174	4045-4303	4177.6
		2	6	2.2-13.1	4.17
		3	0	0-3.7	0.003
90 nm 0x55	3655	1	3655	3534-3776	3648.6
		2	0	0-3.7	3.18
		3	0	0-3.7	0.002

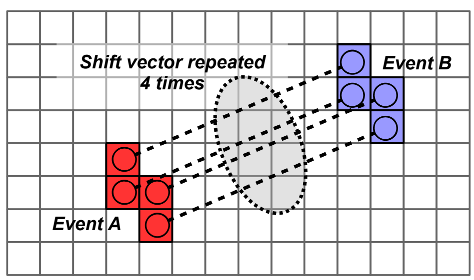

Figure 3. Interaction between MCUs. Each square of the grid symbolizes a memory cell. The interaction between distant MCUs with identical shape can introduce anomalously repeated values in the DV set.

repeated in each DV set. As L_N and N_{DV} are known, numerical calculations on Eqs. 11 & 12 allow determining the threshold value of m (m_0) from which the expected number of repeated elements is lower than ϵ_R. In consequence, elements appearing in the DV sets more than m_0 times are not compatible with only-SBU systems and must be attributed to the occurrence of multiple events.

Although anomalous DV values only link two addresses, they can recreate MCUs of larger size. Thus, if two addresses A_1 & A_2 are linked by an anomalous value DV_0, but A_1 is also linked to another address A_3 by another value DV_1, it is evident that both pairs must be merged into an MCU involving three addresses: $\{A_1, A_2\} \cup \{A_1, A_3\} = \{A_1, A_2, A_3\}$.

Unfortunately, not all of the values are appropriate to detect MCUs due to a particular phenomenon: the interaction between multiple events [4]. This phenomenon is described in Fig. 3: when the addresses of cells of two MCUs are combined to create the DV set, the occurrences of the shift vectors relating cells are anomalously high. To avoid this problem, in our previous work [4] we proposed that more than 15 anomalous DV values never should be chosen. This was obtained after some trial-and-error tests. In the present work, it is proposed a new strategy to avoid the selection of false MCU indicators. We have called the idea *"self-consistency"*. In Fig. 3, one can see that the shift vector appears 4 times, exactly the size of the MCUs. This gives a clue to reject false positives: *Anomalously repeated values are useful if and only if they are repeated more times than the*

Table VI
ELEMENTS IN EXCESS IN THE 130NM/0X00 DVS

XOR		Positive Subtraction	
Value	Occurrences	Value	Occurrences
0x010001	22	0x0$FFFF$	14
0x010101	14	0x000100	13
0x000100	13	0x010001	8
25 elements	4	0x0$FEFF$	5
		0x025$B7E$ 0x09C382	4
90 elements	3	13 elements	3

Table VII
APPLICATION OF SELF-CONSISTENCY RULE FOR THE 130NM/0X00

			Event sizes vs. occurrences						
Step	Value	Rep.	1	2	3	4	5-7	8	S.C.?
0	N/A	N/A	115	0	0	0	0	0	N/A
1	0x0$FFFF$	14	87	14	0	0	0	0	Yes
2	0x00100	13	73	11	4	2	0	0	Yes
3	0x10001	8	66	9	5	4	0	0	Yes
4	0x0$FEFF$	5	66	9	5	4	0	0	Yes
5	0x25$B7E$ 0x9C382	**4**	66	9	5	0	0	**1**	NO

size of the largest proposed MCU.

This rule is valid both for the $P.S.$ and the XOR. The rule is iterative: First of all, the most repeated element in the DV set is selected and its involved addresses are identified. Then, the second most repeated element is selected, added to the preliminary set of anomalous DV values, and the addresses reclassified. This process continues until the principle of self-consistency is violated.

This rule is democratic. In other words, if several DV elements appear the same number of times, provided that the self consistency is not violated, all of them must be added as a whole to the preliminary set of anomalous DV values.

Let us illustrate the procedure with the results of the 130nm/0x00 test of Table II, where 115 addresses were corrupted. In this case, $N_{DV} = 6555$. Hence, Eq. 12 predicts that, in the $XORDV$ set, ~ 10.57 elements appear 2 times, and 0.011 elements appear 3 times. For the $P.S.$, Eq. 11 predicts 14.07 and 0.022 times, respectively. Using the ϵ_R threshold, there should not be events repeated 3 or more times in both DV sets. However, as Table VI shows, many elements are repeated more than 3 times. Hence, there must be something beyond the SBUs: the occurrence of multiple events.

Table VII illustrates this procedure step by step, for the data regarding the $P.S.$ in Table VI. Initially, the set of anomalous values is empty (Step 0). After every step, new MCUs with different multiplicities are found (Columns *SEUs (Event sizes vs. occurrences)*) and the self-consistency is verified. Only in Step #5 something works wrong: An 8-bit MCU is proposed, but the DV values that identify said MCU only appear four times. Therefore, the self-consistency is violated. Thus, these elements are removed from the set of anomalous values, the set of MCUs is recalculated and the procedure stops. The self-consistency turned out to be more accurate than the first proposal of taking no more than 15 values [4]. For example, if the latter criteria had been used, for the 130nm/0x00 test, 0x25$B7E$ and 0x9C382 would have been accepted, but the addresses involving those DV values do not belong to the same MCU. This point was verified with proprietary unscrambling information provided by the manufacturer.

This rule is applied separately to the XOR and $P.S.$ DV sets and the resulting sets of MCUs must be combined in order to obtain a larger set of DV elements. In our previous works [4], [10], both $P.S.$ and XOR (separately) were not completely efficient. In some cases, some MCUs were detected by $P.S.$ and were not immediately detected by the XOR, and vice-versa. None of these MCUs was a false positive, so a combined classification turned out to be significantly more accurate than the separate ones. For instance, for the 90nm/0x55 test, the XOR operation allowed classifying 120 addresses into 96 SBUs and 12 2-bit MCUs. The $P.S.$ led to 99 SBUs, 7 2-bit MCUs, 1 3-bit MCU, and 1 4-bit MCU. After the combination, the proposal was much more accurate: 89 SBUs, 12 2-bit MCUs, 1 3-bit MCU, and 1 4-bit MCU.

B. Combination of data from different experiments and the Pattern Rule

In [4], it was observed that the occurrence of anomalous DV values strongly depended on the written pattern. Some values hardly recognizable in one test can occur several times in other rounds after changing the pattern. Therefore, the use of several patterns in different tests was convenient to obtain a more complete set of anomalous DV values. For instance, for the 130-nm memory, the following anomalous DV values were observed for the $P.S.$: {0x256, 0x65279, 0x65535, 0x65537} for the 0x00 pattern, {0x65535, 0x524032, 0x589567} for 0x55, and {0x65535, 0x65537, 0x262400, 0x524032} for 0xFF. Clearly, the union of these sets, {0x256, 0x65279, 0x65535, 0x65537, 0x262400, 0x524032, 0x589567}, is better than the partial ones separately.

Additionally, it is very common that experiments are repeated several times in similar conditions for a number of reasons: Verification of the test repeatability, changes in the kind of particle or its energy, etc. In this situation, anomalous DV values issued from different experiments, not necessarily only from different patterns, can also be combined to improve the classification of the events. It is clear that the anomalous DV values issued from one experiment are valid for the other ones since they are only related to the internal structure of the SRAM and not to the kind of radiation. Therefore, we propose to merge both sets of values whenever it is possible.

The accuracy of the extraction technique can be improved with the combination of DV values from different experiments (either issued from XOR or $P.S.$) before identifying the anomalies. The reason is that the data processing relies on the assumption that, in each experiment, the DV elements are picked up from the set of all the possible ones (for more details, see the Appendix). If N_{DV1} and N_{DV2} elements are selected in different rounds, the union of both sets contains $N_{DV1} + N_{DV2}$ randomly selected elements that accomplish the conditions described in Section II and in the Appendix[2]. Therefore, the histogram with the number of occurrences of this new set is the addition of the partial histograms.

Such combination of DV values has an additional advantage: Random fluctuations in the number of occurrences of DV elements are mitigated so it is less likely to accidentally select false positives. If the number of repetitions in a unique experiment is distributed between $m \cdot (1 \pm \Delta\alpha)$, statistical theory shows that the addition of n similar experiments leads to $\sim n \cdot m \cdot (1 \pm \Delta\alpha/\sqrt{n})$ [16]. Meanwhile, the number of anomalous DV occurrences is proportional to n so they are easier to locate. Also, anomalously abundant DV values caused by the interaction between MCUs (Fig. 3) are mitigated if data from different experiments are combined.

However, said combination of DV values is not advisable whatsoever if those values were issued from experiments with different patterns. In other words, the idea that was discussed above, and also in the last paragraph of the previous subsection (combination of DV values) must not be applied in case of having experiments with different patterns. The reason is that some DV values appear many times with a pattern but few with the others, in such a way that the sum of occurrences is compensated, thereby making impossible to discern the excess of occurrences from random fluctuations if both sets are merged. Hence, in a nutshell, the *Pattern Rule* establishes that the sets of addresses issued from different patterns should be analyzed separately (i.e., not combining their DV values), in order to obtain their respective anomalous DV values, and after that, these anomalous values can be safely merged to extract MCUs from the bulk of SEUs. However, if the patterns are identical, their DV sets must be merged prior to obtaining the anomalous ones. Finally, other parameters such as the incidence angle or SRAM orientation might behave like the pattern so new experiments must be performed.

C. The Trace Rule

Typically, memories are modularly designed using blocks that are multiplexed by the address bits. This means that a large part of them are shared by adjacent cells. When these addresses are XORed, the resulting DV element will have a lot of 0's and very few 1's (trace) in binary format. In an environment where MCUs can occur, these values, especially those with no

[2]It is important to note that sets of addresses must not be merged, but the derived DV ones. If sets of addresses are merged, false MCUs may appear.

Table VIII
130-NM SRAM: ESTIMATED VS. ACTUAL EVENTS

	SBU		2-bit MCU		3-bit MCU	
Test	Estim.	Actual.	Estim.	Actual	Estim.	Actual
A	369	355	34	41	0	0
B	322	312	29	34	0	0
C	243	241	19	20	1	1
D	273	271	21	22	0	0
E	256	248	26	30	0	0
F	260	252	33	37	0	0

more than 3 1's out of 21 bits, are usually anomalously over-represented. Hence, it is interesting to study them because they are evidences of the occurrence of MCUs.

This leads us to the Trace Rule, which is quite simple: *It is necessary to look for elements with low trace in the $XORDV$ set and verify if they appear more often than expected*. In our experiments, we decided to look for elements with trace equal to or lower than 3 and appearing m_0 times or more in the $XORDV$ set. m_0 was defined in Section IV-A.

D. The XORing rule

Another interesting consequence of the modular organization of SRAMs is that new anomalous $XORDV$ values can be extracted by XORing other confirmed anomalous $XORDV$ values [4]. In that paper, the following scenario was depicted: After an experiment, a DV set was built as well a set of anomalous $XORDV$ values, $AXORDV$, after applying previous rules. Then, if an element $K \in XORDV$, $K \notin AXORDV$, and one of the following conditions occured:

1) It can be expressed as $K = C_1 \oplus C_2$, with $C_1, C_2 \in AXORDV$.
2) $K \oplus C_1 = C_2$, with $C_1, C_2 \in AXORDV$.
3) $K \oplus K_2 = C_1$, with $C_1 \in AXORDV$, and $K_2 \in XORDV$.

Then, K was added to the $AXORDV$ set. The problem in this initial postulation was that there was not a way to determine if the occurrence was just issued from randomness or not. Hence, in the methodology presented in this paper, we propose to accept only those elements discovered by these operations and, additionally appearing m_0 times or more.

E. The Preliminary MCUs Rule

In a p-bit MCU, it is possible to determine $0.5 \cdot p \cdot (p-1)$ DV values relating close cells for each operation. Thus, for example, in a 4-bit MCU it is possible to calculate 6 $XORDV$ values and other 6 $PSDV$ ones from the involved addresses. It is possible that, after applying the previous rules, some of these $XORDV$ or $PSDV$ values are not discovered (for instance, as a consequence of the self-consistency rule being previously violated). Thus, this rule proposes to include them in the set of anomalous $PSDV$ or $XORDV$ values, but only if they appear more than m_0 times.

In consequence, after a preliminary identification of the MCUs, they must be further analyzed in order to possibly accumulate more anomalous DV values. Unfortunately, the application of this rule in actual irradiation sets on the $PSDV$ set led to false positives relating addresses not physically close. This rule only worked correctly in the detection of new elements in $XORDV$.

This rule is iterative. After every search of new $XORDV$ values, the organization of multiple events can change: appearance of new 2-bit MCUs, growth of some MCUs, merging of two small MCUs to yield a largest one, etc. After a new MCU has been identified, the new potential $XORDV$ values must be analyzed until no new elements are discovered. Once this happens, the execution of this rule finishes.

V. RESULTS AND DISCUSSION

The rules depicted in Section IV have been used to elaborate a methodology to detect MCUs (Algorithm 1). If other anomalous DV values are known (e.g., from the literature or from experiments with another pattern), they must be included in the *XCandidates* and *PCandidates* sets before *#Step 6*.

Algorithm 1 Proposed methodology to extract MCUs

Input: Sets of affected addresses: $Addr_1$, $Addr_2$, ..., $Addr_n$
Output: Set of MCUs: $MCUs$

#Step 1: Create DVs for all the address sets and initialize MCUs
$MCUs$ = $\emptyset$;
for each $Addr_i$ **do**
  $XORDV_i$ = CreateXORDV ($Addr_i$, $XORDV_i$);
  $PSDV_i$ = CreatePSDV ($Addr_i$, $PSDV_i$);
end for;

#Step 2: Merge DV sets if issued with = pattern (pattern rule)
pat_rule (&$XORDV_1$, &$PSDV_1$, ..., &$XORDV_n$, &$PSDV_n$);

#Step 3: Iterate on the DV sets obtained in the previous steps
$AXORDV$ = $\emptyset$; *// Set of anomalous values issued from XOR*
$APSDV$ = $\emptyset$; *// Set of anomalous values issued from P.S.*
for each pair ($XORDV_i$, $PSDV_i$) **do**
  #Step 3.1: Study of DVs by applying Equation (10)
  $XCandidates$ = InExcess ($XORDV_i$, 0.05);
  $PCandidates$ = InExcess ($PSDV_i$, 0.05);

  #Step 3.2: Apply Self-consistency in $XORDV_i$
  $AXORDV$ = $\emptyset$;
  $violatedSelfConsistency$ = $FALSE$;
  while ($violatedSelfConsistency$ == $FALSE$) **do**
    $newCandidate$ = getMostRepeated ($XCandidates$);
    updateSet (&$AXORDV$, $newCandidate$);
    $updated\_MCUs$ = updateMCUs ($MCUs$, $AXORDV$);
    if (numberOfOccurrences ($newCandidate$, $XORDV_i$) > sizeLargestMCU ($updated\_MCUs$) **then**
      $MCUs$ = $updated\_MCUs$;
    else
      $violatedSelfConsistency$ = $TRUE$;
    end if;
  end while;

  #Step 3.3: Self-consistency in $PSDV_i$: Similar to Step 3
  #Creates/updates $APSDV$ and updates $MCUs$ again
  #Steps 3.2 & 3.3 combine DV values from XOR and P.S.

  #Step 3.4: Trace rule in $XORDV_i$
  $newCandidates$ = extractLowTrace ($XCandidates$);
  updateSet (&$AXORDV$, $newCandidates$);
end for;

#Step 4: XORing rule
$newCandidates$ = XOR_Rule ($AXORDV$, $XCandidates$);
updateSet (&$AXORDV$, $XCandidates$);

#Step 5: Preliminary MCU Rule
for each MCU_i in $MCUs$ **do**
  $all\_XValues$ = extractXValues (MCU_i);
  updateSet (&$AXORDV$, $all\_XValues$, $XCandidates$);
end for;

#Step 6: Update MCUs with the final AXORDV and APSDV sets
$MCUs$ = updateMCUs ($MCUs$, $AXORDV$);
$MCUs$ = updateMCUs ($MCUs$, $APSDV$);
return $MCUs$;

To verify the efficacy of this strategy, data sets issued from irradiations of 90 & 130-nm SRAMs at the GENEPI2 14-MeV neutron facility were used. In these experiments, the memories were written with 0x55 pattern and irradiated in different rounds (Tables VIII and IX). The data in Table VIII have not been published elsewhere, whereas those of Table IX have been presented and analyzed in another work [17].

Both tables compare the estimated number of MCUs vs. the actual ones. The actual distributions of MCUs were deduced

Table IX
90-NM SRAM: ESTIMATED VS. ACTUAL EVENTS

	SBU		2-bit MCU		3-bit MCU		4-bit MCU	
Test	Est.	Act.	Est.	Act.	Est.	Act.	Est.	Act.
A	1621	1645	112	96	14	12	8	8
B	1354	1385	105	89	11	10	2	3
C	1201	1215	105	96	11	13	5	3
D	1047	1065	109	97	14	15	5	4
E	879	876	95	99	15	12	3	4
F	727	734	93	77	11	16	4	5
G	623	623	70	69	5	7	1	0
	5-bit MCU		6-bit MCU		7-bit MCU		≥8 bit	
A	0	2	1	0	0	0	0	1
B	1	0	0	1	0	0	0	0
C	0	1	0	0	0	0	0	0
D	0	0	0	0	0	1	0	0
E	0	0	0	1	0	0	0	0
F	0	0	0	0	0	0	0	0
G	0	0	0	0	0	0	0	0

by using proprietary information from Cypress, as we also did for the validations of Section III-B. In Table VIII, the number of events is underestimated. In other words: The algorithm did not manage to identify all of the anomalous values. However, in Table IX, the case is exactly the opposite: Multiple events are overestimated. The reason is that the physical location of the cells depends on the bit position in the word. In this case, some bitflips were close enough to mislead the algorithm because the addresses are quite similar but they did not belong to MCUs (after incorporating the bit position, cells turned out to be too far from each other). A possible improvement of the algorithm would consist in incorporating the bit position to the affected address. Therefore, 8-bit words would require 3 additional bits to codify the address. The problem is that the L_N increases by a factor of 8, making computations heavier.

Also, the algorithm fails at detecting unusual large events. In any case, the non-detected events are on the order of 15% of the total, which is a quite good estimation taking into account that no knowledge of the physical layout was required and that this margin is on the order of the experimental error (more or less, twice the square root of the number of events). Finally, the computation time of the program that has been coded for Algorithm 1 is not high at all. Thus, the classification of the whole data shown in Table IX only required $\sim$45 seconds on a PC with a 4-core Intel Xeon processor, 8-Gb RAM, working at 3.39 GHz, and running Xubuntu GNU/Linux 16.04/64-bits.

The algorithm can be improved in some ways. Timestamp is easy to include. Some tests consist in periodically reading and writing the memory during the irradiation and are called *"pseudostatic"* tests [4]. In this case, every round of reading-writing must be thought as an independent test with its own DV sets, and the combination of all of them will lead to the global DV to be studied, as explained in Section IV-B.

Public knowledge about the internal SRAM structure can be successfully used to improve the algorithm. For example, the tested memories can be used either as 1Mx16 or 2Mx8. That means that the memory is internally divided into two blocks: one of them containing addresses from 0 to $2^{20}-1$; the other one, from 2^{20} to $2^{21}-1$. The most significant bit, A_{20} is used to control a multiplexer in the case of the 2Mx8 configuration, and it is unused in the case of the 1Mx16 configuration. It is evident that MCUs can only involve addresses inside the same block. Therefore, it is nonsense the calculation of DV elements relating addresses belonging to different blocks.

The SRAMs studied in this paper contain 2^{21} addresses, but they are divided into two blocks of 2^{20} addresses each. If, for instance, we record 100 errors equally distributed between both blocks (50 errors each), the original problem would lead to investigate 4950 DV elements. However, taking into account this physical division into two blocks, these data can be studied separately, thereby yielding two sets of only 1225 DV elements. Hence the total number of DV elements to be studied is only 1225 + 1225 = 2450. Obviously, values relating MCUs are the same in any case.

Table X
DA FOR $N = 3$ AND POSITIVE SUBTRACTION (TABLE I IN [10])

	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**
0	...	1	2	3	4	5	6	7
1	...	...	1	2	3	4	5	6
2	...	...	...	1	2	3	4	5
3	...	...	...	...	1	2	3	4
4	...	...	...	...	...	1	2	3
5	...	...	...	...	...	...	1	2
6	...	...	...	...	...	...	...	1
7	...	...	...	...	...	...	...	...

The presented technique is effective but of course, it has some limitations. One of them is that, in case of long and intense irradiations, SBUs can occur in quite close cells and be mistaken for multiple events [18]. However, this problem occurs even in the case of knowing the physical layout of the memory. Finally, it is worth to indicate that our technique may not work in experiments where events other than SBUs, MBUs or MCUs occur, such as those depicted in [6], where huge clusters of errors dominate SBUs. In this case, the experiment cannot be depicted as an only-SBU scenario with second-order perturbations related to the occurrence of MCUs.

VI. CONCLUSIONS AND FUTURE WORK

In this paper, it has been demonstrated that the deviations from the statistical properties of an only-SBU scenario for SRAMs can be used to provide a quite exact picture of the distribution of MCUs. Error rates for SBUs and MCUs can be estimated with a accuracy of 15% without needing information about the physical structure of the SRAM. Also, the proposed algorithm is not difficult to implement, nor it requires long computation times in typical personal computers. Only the unusual largest events, which appear too seldom to extract statistical information, were not detected in the tests. As future work, this approach will be further validated with dynamic tests on the same SRAMs, DRAMs or FLASH memories.

APPENDIX

This appendix has been taken mostly from the previous RADECS work [10], so this paper is totally self-contained.

A. Statistics for the positive subtraction, XOR, etc.

Let A be a set containing every natural number between 0 and L_N. Therefore, it contains $L_N + 1$ elements. Now, in this set a binary operation $d : A \times A \to A$ is defined with the following properties: 1) Symmetry: $d(a, b) = d(b, a)$, and 2) $d(a, a)$ is not defined for any $a \in A$. Examples are the $P.S.$ ($d(a, b) = \max(a, b) - \min(a, b)$), XORing ($d(a, b) = a \oplus b$), etc., both with the prohibition of combining two identical elements. Now, a new set, called DA, associated with A, is created such that:

$$DA = \{d(a_i, a_j), \forall a_i, a_j, \backslash a_i < a_j,\ a_{i,j} \in A\} \tag{3}$$

Table X shows the DA-set for $N = 3$ ($L_N = 2^3 - 1 = 7$) with $P.S.$. From this example some interesting features can be observed:

- 0 never appears among the 8 possible values, result of the prohibition of subtracting two identical values.
- This set has 28 elements and 7 possible values. Obviously, several values are repeated but the number of times that they appear are not identical.

Indeed, two fundamental facts can be deduced from the principle of mathematical induction: First, if A contains M elements, ($[0, 1, \ldots, M-1]$), there are N_{DA} elements in DA:

$$N_{DA} = \begin{pmatrix} M \\ 2 \end{pmatrix} = \frac{1}{2} \cdot M \cdot (M - 1) \tag{4}$$

Every element is created combining two different elements of A, without repetition and regardless of order, a well-known problem in combinatorics [16]. In our scenario, $M = 2^N = L_N + 1$. Secondly, the number of times that an element $k \in [1, 2, \ldots, M-1]$ appears in DA is:

$$N_k = M - k, \tag{5}$$

deduced from the principle of mathematical induction. The classical approach to probability establishes that the probability of an event is the number of favorable cases divided by the total number of possible cases [16]. Therefore, the probability of obtaining $d(a_1, a_2) = k$, $0 \leq k \leq L_N$ after choosing two different elements, a_1 and a_2, from A is:

$$P_{DA}(k) = N_k / N_{DA} \tag{6}$$

This result is valid for any operation and, in the case of the $P.S.$, the equation becomes:

$$P_{DA,PS}(k) = \frac{N_k}{N_{DA}} = \frac{2 \cdot (L_N + 1 - k)}{L_N \cdot (L_N + 1)} \tag{7}$$

In the case of using the XOR operation, it can be demonstrated that $P_{DA,XOR}(k) = L_N^{-1}$ *if and only if* M *is a natural power of* 2 [4]. From Eq. 7, one can demonstrate that the mean value, $\overline{k}$, and the standard deviation of DA, σ, with $P.S.$ are[3] $\bar{k}_{PS} = \frac{1}{3}(L_N + 1)$ and $\sigma_{PS} \approx \frac{1}{\sqrt{2}} \cdot \bar{k}$. If the XOR had been used instead, the values of the parameters would have been $\bar{k}_{XOR} = \frac{1}{2}(L_N + 1)$ and $\sigma_{XOR} \approx \frac{1}{\sqrt{3}} \cdot \bar{k}$.

Other operations can be defined but they usually lead to expressions difficult to work with.

B. *The DA SET and the irradiation experiment*

As previously stated, obtaining N_E addresses with SBUs is formally equivalent to randomly selecting N_E elements out of $[0, 1, \ldots, L_N]$. From this subset $V \subset A$ with N_E elements, a new set, DV, is generated exactly as DA was: combining every element in V with those higher than it and then applying the binary operation. This new set will have N_{DV} elements:

$$N_{DV} = 0.5 \cdot N_E \cdot (N_E - 1) \tag{8}$$

The probability of occurrence is determined by Eq. 6, as well as by versions particularized for each specific operation.

C. *Statistical parameters observable in DV*

According to the classic approach, the probability of randomly extracting N_{DV} elements from a set A and obtaining m times an element k is:

$$P(k, m, N_{DV}) = \left(\begin{array}{c} N_{DV} \\ m \end{array} \right) \cdot p_k^m \cdot (1 - p_k)^{N_{DV} - m} \tag{9}$$

p_k being the probability of obtaining k in only one attempt. Clearly, in the studied case, $p_k \equiv P_{DA}(k)$. The expected number of elements that appear m times is just the addition of all the individual probabilities:

$$\begin{aligned} N_R(m, N_{DV}) &= \sum_{k \in A} P(k, m, N_{DV}) = \\ &= \left(\begin{array}{c} N_{DV} \\ m \end{array} \right) \cdot \sum_{k \in A} p_k^m \cdot (1 - p_k)^{N_{DV} - m} \end{aligned} \tag{10}$$

Regarding the case of positive subtraction, $N_{R,PS}$ is calculated by replacing p_k by $P_{DA,PS}(k)$ from Eq. 7. If $N_{DV} \ll L_N$, a simple approach for N_R arises:

$$N_{R,PS}(m, N_{DV}) \approx \frac{2^m}{m+1} \cdot \left(\begin{array}{c} N_{DV} \\ m \end{array} \right) \cdot L_N^{1-m}.$$

[3]These parameters are defined as $\bar{k} = \sum_{k \in DA} k \cdot P_{DA}(k)$, $\sigma^2 = \overline{k^2} - \left(\overline{k}\right)^2$, with $\overline{k^2} = \sum_{k \in DA} k^2 \cdot P_{DA}(k)$. Values are calculated using the identities $\sum_{k=0}^{n} 1 = n + 1$, $\sum_{k=0}^{n} k = \frac{1}{2} n \cdot (n+1)$, and $\sum_{k=0}^{n} k^2 = \frac{1}{6} n \cdot (n+1) \cdot (2n+1)$.

$$\cdot \left(1 + \frac{3 \cdot m - 2 \cdot N_{DV}}{L_N + 1}\right) \cdot \left(1 + \frac{2\left(N_{DV} - m\right)}{\left(L_N - 2\right) \cdot \left(m + 2\right)}\right) \tag{11}$$

It is worth to compare this expression with that derived from the XOR operation [4], which is exact and much simpler:

$$N_{R,XOR}\left(m, N_{DV}\right) = \left(\begin{array}{c} N_{DV} \\ m \end{array}\right) \cdot \frac{\left(L_N - 1\right)^{N_{DV} - m}}{L_N^{N_{DV} - 1}} \tag{12}$$

References

[1] R. W. Hamming, "Error Detecting and Error Correcting Codes," *Bell Syst. Tech. J*, vol. 29, no. 2, pp. 147–160, Apr. 1950.

[2] S. P. Buchner, F. Miller, V. Pouget, and D. P. McMorrow, "Pulsed-Laser Testing for Single-Event Effects Investigations," *IEEE Tran. Nucl. Sci.*, vol. 60, no. 3, pp. 1852–1875, Jun. 2013.

[3] A. Manuzzato, S. Gerardin, A. Paccagnella, L. Sterpone, and M. Violante, "On the Static Cross Section of SRAM-Based FPGAs," in *2008 IEEE Radiation Effects Data Workshop*, Jul. 2008, pp. 94–97.

[4] J. A. Clemente, F. J. Franco, F. Villa, M. Baylac, S. Rey, H. Mecha, J. A. Agapito, H. Puchner, G. Hubert, and R. Velazco, "Statistical Anomalies of Bitflips in SRAMs to Discriminate SBUs from MCUs," *IEEE Tran. Nucl. Sci.*, vol. 63, no. 4, pp. 2087–2094, Aug. 2016.

[5] D. Falguere and S. Petit, "A Statistical Method to Extract MBU Without Scrambling Information," *IEEE Tran. Nucl. Sci.*, vol. 54, no. 4, pp. 920–923, Aug. 2007.

[6] G. Tsiligiannis, L. Dilillo, V. Gupta, A. Bosio, P. Girard, A. Virazel, H. Puchner, A. Bosser, A. Javanainen, A. Virtanen, C. Frost, F. Wrobel, L. Dusseau, and F. Saigné, "Dynamic Test Methods for COTS SRAMs," *IEEE Tran. Nucl. Sci.*, vol. 61, no. 6, pp. 3095–3102, Dec. 2014.

[7] A. L. Bosser, V. Gupta, G. Tsiligiannis, C. D. Frost, A. Zadeh, J. Jaatinen, A. Javanainen, H. Puchner, F. Saigné, A. Virtanen, F. Wrobel, and L. Dilillo, "Methodologies for the Statistical Analysis of Memory Response to Radiation," *IEEE Tran. Nucl. Sci.*, vol. 63, no. 4, pp. 2122–2128, Aug. 2016.

[8] M. Wirthlin, D. Lee, G. Swift, and H. Quinn, "A Method and Case Study on Identifying Physically Adjacent Multiple-Cell Upsets Using 28-nm, Interleaved and SECDED-Protected Arrays," *IEEE Tran. Nucl. Sci.*, vol. 61, no. 6, pp. 3080–3087, Dec. 2014.

[9] A. Hands, P. Morris, C. Dyer, K. Ryden, and P. Truscott, "Single Event Effects in Power MOSFETs and SRAMs Due to 3 MeV, 14 MeV and Fission Neutrons," *IEEE Tran. Nucl. Sci.*, vol. 58, no. 3, pp. 952–959, Jun. 2011.

[10] F. J. Franco, J. A. Clemente, M. Baylac, S. Rey, F. Villa, H. Mecha, J. A. Agapito, H. Puchner, G. Hubert, and R. Velazco, "Some Properties of only-SBUs Scenarios in SRAMs Applied to the Detection of MCUs," in *2016 IEEE Conference on Radiation Effects on Components and Systems (RADECS)*, Sep. 2016, p. (pending of publication).

[11] J. Bezanson, A. Edelman, S. Karpinski, and V. B. Shah, "Julia: A Fresh Approach to Numerical Computing," Online: http://arxiv.org/abs/1411.1607, Nov. 2014.

[12] F. Villa, M. Baylac, S. Rey, O. Rossetto, W. Mansour, P. Ramos, R. Velazco, and G. Hubert, "Accelerator-Based Neutron Irradiation of Integrated Circuits at GENEPI2 (France)," in *2014 IEEE Radiation Effects Data Workshop (REDW)*, Jul. 2014, pp. 1–5.

[13] F. Villa, M. Baylac, A. Billebaud, P. Boge, T. Cabanel, E. Labussière, O. Méplan, and S. Rey, "Multipurpose Applications of the Accelerator-Based Neutron Source GENEPI2," *Nuovo Cimento C-Colloquia and Communications in Physics*, no. 38, Article ID: 182, pp. 1–8, May 2016.

[14] S. Craw, *Manhattan Distance*. Boston, MA: Springer US, 2010, pp. 639–639. [Online]. Available: http://dx.doi.org/10.1007/978-0-387-30164-8_506

[15] J. L. Autran, D. Munteanu, P. Roche, and G. Gasiot, "Real-Time Soft-Error Rate Measurements: A Review," *Microelectron. Reliab.*, vol. 54, no. 8, pp. 1455–1476, Aug. 2014.

[16] J. Schiller, M. R. Spiegel, and R. A. Srivanasan, *Schaums Outline of Theory and problems of Statistics*, 3rd ed. McGraw-Hill, Aug. 2008.

[17] J. A. Clemente, G. Hubert, F. J. Franco, F. Villa, M. Baylac, H. Mecha, H. Puchner, and R. Velazco, "Sensitivity Characterization of a COTS 90-nm SRAM at Ultra Low Bias Voltage," *IEEE Tran. Nucl. Sci.*, vol. (pending of publication), 2017.

[18] J. A. Maestro and P. Reviriego, "A Method to Eliminate the Event Accumulation Problem from a Memory Affected by Multiple Bit Upsets," *Microelectron. Reliab.*, vol. 49, no. 7, pp. 707–715, Jul. 2009.