SERESSA 2022

5th to 9th of December at CERN, Geneva

Analyzing data extracted from radiation tests in advanced SRAMs

Juan A. Clemente, Universidad Complutense de Madrid (UCM)



Agenda

- 1. Introduction and motivation
- 2. Extraction of simple events (SBUs) / multiple events (MCUs / MBUs)
- 3. Analysis of "false" Multiple Cell Upsets (MCUs) by accumulation
 - Birthday statistics
 - Correction of experimental data
- 4. Analysis of "false" Multiple Bit Upsets (MBUs) by accumulation
 - Error Correcting Codes (ECC)
 - Accumulation of events and ECC reliability
- 5. Conclusions

1-Introduction and motivation

Introduction

- Accelerated radiation tests on SRAMs are a common way of estimating the sensitivity of a device in harsh conditions.
- OK, we test a device against radiation and... what do we get?



SEU!! @address: 000D4F69; 46 != 42 SEU!! @address: 001DCA89; C3 != E3 SEU!! @address: 00030BC7; 80 != 84 SEU!! @address: 00030C25; 6C != 64 SEU!! @address: 000D5079; 16 != 14 SEU!! @address: 001DDC55; 7D != 6D @address: 00030EF4; FE != DE SEU!! @address: 000D50D9; 9E != 9A SEU!! @address: 001DDF75; 22 != 20 SEULI SEU!! @address: 000D5B55; 65 != 6D SEU!! @address: 001DE24C; 7B != 79 SEU!! @address: 00030FA2; A5 != A1 SEU!! @address: 000311A5; 93 != 9B SEU!! @address: 000D7153; 60 != 70 SEU!! @address: 001DE616; 51 != 41 SEU!! @address: 000311DA; 9D != 9C SEU!! @address: 000D72A0; E5 != A5 SEU!! @address: 001DEACA; 84 != 86 SEU!! @address: 000313BF; 92 != 82 SEU!! @address: 000D79E0; A8 != A9 SEU!! @address: 001DEE9C; AF != AE SEU!! @address: 000D7ADC; 20 != A0 SEU!! @address: 001DF0E2; AF != AD SEU!! @address: 0003147A; 51 != 11 SEU!! @address: 00032387; C4 != 84 SEU!! @address: 000D815F; 5A != 5B SEU!! @address: 001DF303; 89 != 09 SEU!! @address: 00032422; 1E != 5E SEU!! @address: 000D8D68; 44 != 45 SEU!! @address: 001DF84B; 72 != 7A SEU!! @address: 000D92A2; B1 != A1 SEU!! @address: 001DFAE1; AF != AB SEU!! @address: 00032D42; 7C != 7E SEU!! @address: 000D93E7; AA != BA SEU!! @address: 001DFC71; 3C != 2C SEU!! @address: 00032DB7; 8C != 84 SEU!! @address: 000D94ED; CB != CA SEU!! @address: 001DFD39; 7C != 7D SEU!! @address: 00032DB9; 82 != 83 SEU!! @address: 001E0148; 78 != 7C SEU!! @address: 00032DC2; 80 != 82 SEU!! @address: 000D9921; DC != 5C SEU!! @address: 000D9F70; 2E != 2F SEU!! @address: 001E07E9; BE != BF @address: 000334FB; E4 != F4 SEU!! @address: 000338F9; CE != EE SEU!! @address: 001E0C09; 1A != 1B SEU!! SEU!! @address: 000DA2AB; 81 != 91 SEU!! @address: 0003409F; A9 != A8 SEU!! @address: 000DA9B3; 07 != 87 SEU!! @address: 001E2117; 40 != 44 SEU!! @address: 000342D9; DA != 9A SEU!! @address: 000DB358; E8 != 68 SEU!! @address: 001E2337; 7D != 7C @address: 00034357; EA != 6A SEU!! @address: 000DB831; F6 != 76 SEU!! @address: 001E2988; E4 != E6 SEULI SEU!! @address: 00034F28; 6B != 69 SEU!! @address: 000DC4E2; A9 != AD SEU!! @address: 001E2B7F; 03 != 01 SEU!! @address: 001E2BFF; 02 != 00 @address: 000354E1; AA != AB SEU!! @address: 000DC6BF; C2 != 82 SEU!! SEU!! @address: 000355AD; 0E != 8E SEU!! @address: 000DC703; 29 != 09 SEU!! @address: 001E2DF5; E6 != E2 SEU! @address: 00035884; E2 != F2 SEU!! @address: 000DD705; 8F != 0F SEU!! @address: 001E2ED8; 88 != 98 SEU!! @address: 000359EC; 87 != C7 SEU!! @address: 000DD9DB; 8E != 9E SEU!! @address: 001E3005; 07 != 0F SEU!! @address: 001E3159; 47 != 67 SEU!! @address: 00035E70; 2D != 2F SEU!! @address: 000DDAB6; A5 != 85 SEU!! @address: 001E379F; B8 != A8 SEU!! @address: 000368CC; 8A != 88 SEU!! @address: 000DE196; AE != BE SEU!! @address: 00036BA8; 9E != 96 SEU!! @address: 000DE237; 74 != 7C SEU!! @address: 001E37D4; 93 != 92 SEU!! @address: 001E38B0; 0A != 8A SEU!! @address: 00036DBB; A2 != 82 SEU!! @address: 000DE304; 8C != 0C SEU!! @address: 00037421; DC != 5C SEU!! @address: 000DE396; FE != BE SEU!! @address: 001E3C2B; 6C != 6E @address: 00037789; E7 != E3 SEU!! @address: 001E43C0; C2 != 82 SEU! SEU!! @address: 000DE5CE; 88 != 8A SEU!! @address: 001E4CFD; 7A != FA SEU!! @address: 0003793D; 7C != 7E SEU!! @address: 000DE5FE; BD != FD SEU!! @address: 00037E09; 9B != 1B SEU!! @address: 000DEA11; 31 != 33 SEU!! @address: 001E4E0F; AD != 2D SEU!! @address: 00038175; 60 != 20 SEU!! @address: 000DEB76; 5D != 1D SEU!! @address: 001E4EF3; CB != DB SEU!! @address: 0003854D; 79 != 78 SEU!! @address: 000DF509; 3B != 1B SEU!! @address: 001E4FB2; 8A != 88 SEU!! @address: 00038775; 30 != 20 SEU!! @address: 000DF6C1; C2 != 82 SEU!! @address: 001E57AA; 13 != 93 SEU!! @address: 00038ED8; 99 != 98 SEU!! @address: 000DF883; F7 != F6 SEU!! @address: 001E5943; 3E != 7E SEU!! @address: 00038FBC; 80 != 82 SEU!! @address: 000DF93B; FE != 7E SEU!! @address: 001E6484; FA != F2 SEU!! @address: 00039402; 46 != 06 SEU!! @address: 000E0989; E2 != E3 SEU!! @address: 001E675E; 59 != 5D SEU!! @address: 00039984; F6 != F2 SEU!! @address: 001E68C1; 02 != 82 SEU!! @address: 000E0CF4; FE != DE SEU!! @address: 000399A6; 89 != 99 SEU!! @address: 000E0E25; E4 != 64 SEU!! @address: 001E6F10; 31 != 30



What are these: Single Bit Events (SBUs), Multiple Cell Upsets (MCUs)...?

Motivation

- Technology miniaturization (Moore's law) leads to more cell density.
 - Increase of the SER/device.
 - Also, increase of the % of the MCU SER contribution.
 - >+900% MCU SER contribution between 180-nm and 22-nm nodes.
- MCU understimations lead to wrong estimations of the total SER.
- A correct (or at least, accurate) MCU extraction is critical.



A. Neale, M. Jonkman and M. Sachdev, *Adjacent-MBU-Tolerant SEC-DED-TAEC-yAED Codes for Embedded SRAMs*, in **IEEE Transactions on Circuits and Systems II: Express Briefs**, vol. 62, no. 4, pp. 387-391, April 2015.

2-Extraction of simple events (SBUs) / multiple events (MCUs / MBUs)

Definition of "bit interleaving"

Bit interleaving

Manufacturing technique that physically **separates bits** belonging to the same word, so they are distant enough and they cannot be affected by the same particle.

2 types of n-bit multiple events:

- Multiple Bit Upsets (MBUs): n bits in the same word are flipped by the same particle. <u>Difficult to recover by</u> <u>standard Error Correcting Codes</u> (ECCs).
- Multiple Cell Upsets (MCUs): 1 bit is affected in n words. <u>Each single error</u> is easy to recover (just 1 bit per word).



MCU/MBU extraction with unscrambling

Unscrambling

Information about the internal organization of the memory, provided by the manufacturer, who makes possible to establish a **relationship between "logical addresses" and the physical positions** of those bits in the XY layout of the memory.



Example of internal organization of an SRAM (quads and blocks)

XY representation of the physical addresses affected by bitflips

MCU/MBU extraction without unscrambling

"Statistical" MBU/MCU extraction techniques

When **unscrambling is not available**, **many authors** have proposed techniques that identify MCUs by detecting **statistical anomalies** in the set of observed bitflips. For instance, XOR'ed values between addresses more abundant than they should be in a theoretical scenario where no MCUs can occur.







F. J. Franco et al., Statistical Deviations from the Theoretical only-SBU Model to Estimate MCU rates in SRAMs, in IEEE Transactions on Nuclear Science (TNS), vol. 64, no. 8, pp. 2152-2160, July 2017.

How to analyze data correctly?

- **1.** Initialize the memory with a known pattern (i.e., 0x55).
- 2. Expose the memory under the radiation beam for a given time.
- **3. Read the memory contents** to search errors provoked by radiation.
- **4. Group errors** by multiplicity:
 - Single Bit Upsets (SBUs): 1 particle → 1 error
 - <u>Multiple Cell Upsets (MCUs)</u>: 1 particle \rightarrow several errors in different data words.
 - <u>Multiple Bit Upsets (MBUs)</u>: 1 particle \rightarrow several errors in the same data word.
- 5. Give a metric for the SBU/MCU sensitivity:
 - "<u>Cross section</u>" (σ): Probability of a single particle (proton, neutron, heavy ion...) to provoke an error in a memory bit.

$$\sigma = \frac{\text{Number of events}}{\text{Particle fluence} \cdot \text{Memory size (bits)}}$$

$$\sigma = \frac{\sigma_{SBU}}{\sigma_{SBU}} = \frac{\text{Number of SBUs}}{\sigma_{MCU-2bit}} = \frac{\text{Number of 2} - \text{bit MCUs}}{\text{Particle fluence} \cdot \text{Memory size (bits)}}$$

$$\sigma_{MCU-2bit} = \frac{\text{Number of 2} - \text{bit MCUs}}{\text{Particle fluence} \cdot \text{Memory size (bits)}}$$

3-Analysis of "false" MCUs by accumulation

Birthday statistics

Correction of experimental data

Accumulation of "false" MCUs in a radiation-ground experiment



SRAM columns

Estimation of false MCU rates

□ Idea of the "Birthday paradox".

□ How many people we need to put in the same group so the probability of finding, at least 2 people with the same birthday, is greater than 50%?

 \cdot (365 – *n* + 1)

- Only 23 people.
- https://keisan.casio.com/exec/system/1223738282

$$P_{coincidence} = 1 - \frac{365 \cdot 364 \cdot 363 \cdot \dots \cdot 365^n}{365^n}$$

Z. E. Schnabel, *The estimation of the total fish population of a lake*, in **American Mathematical Monthly**, vol. 45, no. 6, pp. 348-352, June-July 1938.



Probability with same birthdays





One coincidence for the US presidents happened for the 28th president (W. G. Harding)

More on birthday statistics

How many people we need to put in the same group so the probability of finding, at least 2 people whose birthdays are k days apart is greater than 50%?

> Much less: for k=1 day, only 14 people.



M. Abramson and W. Moser, *More Birthday Surprises*, in **American Mathematical Monthly**, vol. 77, no. 8, pp. 856-858, October 1970.

More on birthday statistics

The previous idea can be used for **analyzing bitflips** observed in a memory.



Birthday statistics

- 1. Which is the probability of finding, at least, 2 people whose birthdays are k days apart in a group of n people?
- 2. Which is the probability of finding, at least, 2 bitflips that are k bitcells apart in a memory with n bitflips?
- Birthday statistics can be used for analyzing probability of occurrence of close bitflips (MBUs and MCUs) falsely attributed to the same particle.
- 1. In a group of n people, it's not that unlikely to find 2 birthdays being placed, at least, k days apart
- 2. In a set of n bitflips, it's not that unlikely to find 2 affected addresses being placed, at least, k bitcells apart

In other words, it's not that unlikely to find multiple events by accumulation (false MCUs).

Estimation of the number of 2-bit "false MCUs"

Manhattan Distance (MD)



3-bit MCUs can also be estimated, but equations are way more complex and **out of the scope of this discussion**.

Correction of experimental data

- Example. For a 16-Mbit memory and 2400 bitflips, NF_MCUs_2bit = 4. Does this mean that any time we find 2400 bitflips in a 16-Mbit memory, 4 false 2-bit MCUs will occur for sure?
 - NO. NF_MCUs_2bit is a false 2-bit MCU rate.
- Such false MBUs/MCUs are "rare events" and their stochastic occurrence can be modeled with the **Poisson distribution**.
- \Box Let λ be such an event rate:



Correction of experimental data

Alternative 1. Let an experiment be:

- Memory size = 1Mb (2²⁰ bits)
- Criteria: MD (threshold value = 3)
- *p* = 592 bitflips
- N_{F_MCUs_2bit} = <u>4 false 2-bit MCUs</u>
- N_{observed_MCUs_2bit} = <u>5 observed 2-bit MCUs</u>

What are those 5 MCUs? false, true...?

- \Box Let's find a value of k (k₀) such that CDF(k₀) > 99%
 - k₀ = 9. There is 99% probability that between 0 and 9 false 2-bit MCUs occur in that experiment.
 - 5 false MCUs are perfectly within that range, hence we can consider them as false.



Correction of experimental data

□ Alternative 2. Let another experiment be:

- Memory size = 16Mb (2²⁴ bits)
- Criteria: MD (threshold value = 3)
- *p* = 2439 bitflips
- $N_{F_MCUs_{2bit}} = 4$ false 2-bit MCUs
- N_{observed_MCUs_2bit} = <u>11 observed 2-bit MCUs</u>

□ The following methodology can be followed:

- Confidence margins are calculated around N_{events} = 11.
 - A good approach is: $=\frac{1}{2}\chi^2\left(\frac{\alpha}{2}, 2N_{events}\right) < N_{events} < \frac{1}{2}\chi^2\left(1 \frac{\alpha}{2}, 2(N_{events} + 1)\right)$
 - With 95% confidence, $N_{events} = [N_{events_LOW}, N_{events_HIGH}] = [5.49, 19.68]$
- The following correction is made:
 - [N_{events_LOW} N_{observed_MCUs_2bit}, N_{events_HIGH} N_{observed_MCUs_2bit}]
 - In this case, [5.49 4, 19.68 4] = [1.49, 15.68]
- We can say that, in that experiment, there is 95% probability that, between 1.49 and 15.68 actual 2-bit MCUs occurred.

J.L. Autran, D. Munteanu, P. Roche, G. Gasiot, *Real-time soft-error rate measurements: A review*, in **Microelectronics Reliability**, vol. 54, no. 8, pp. 1455-1476, August 2014.

4-Analysis of "false" MBUs by accumulation

Error Correcting Codes (ECC)

Accumulation of events and ECC reliability

Accumulation of "false MBUs"



Accumulation of "false MBUs" - ECC





Error Correcting Codes (ECC)

- Mechanism to add redundancy to the memory contents.
 - ✓ An M-bit word contains N=M+K bits.
- ✓ The "f" module generates the K redundancy bits.
- The "Comparator" reports if there has been an error in the word (ERR signal)
- ✓ The "Corrector" corrects the DOUT, but it does not correct the fault in the memory module.
- ✓ ECCs are **sensitive** to **accumulated errors**.

Types of ECC

Single Error Correction – Double Error Detection (SEC-DED)





Double Error Correction – Triple Error Detection (DEC-TED)





Triple Error Correction (TEC)





Double Adjacent Error Correction (DAEC)





Single Nibble Correction – Double Nibble Detection (SNC-DND)



Estimation of false MBU rates

- This is relevant for studying the **efficiency of Error Correction Codes (ECC)**.
- □ For instance, in a block of n bits, a **SEC-DED code** will be effective only if 2 bitflips do not occur in the same word:





Estimation of MBU rates

The most accurate estimation ever made in the literature:

Estimated number of false 2-bit MBUs:



 \Box Where $N_{\mu}(k)$ is the estimated number of addresses being hit k times:

$$V_H(k) = {\binom{m}{k}} \cdot (L_A)^{1-k} \cdot \left(1 - \frac{1}{L_A}\right)^{m-k}$$

W = Data width per address (bits)

m = Number of bitflips

 L_A = Total number of data addresses

Obtained by using the ideas of the "urn-andballs problem" (better see reference!!) J.A. Clemente, M. Rezaei and F. J. Franco, *Reliability of Error Correction Codes Against Multiple Events by Accumulation*, in **IEEE Transactions on Nuclear Science (TNS)**, vol. 69, no. 2, pp. 169-180, February 2022.

Estimation of false MBU rates

□ Similarly:

Estimated number of false 3-bit MBUs:

$$N_{FM}(3) \approx \frac{(W-1) \cdot (W-2)}{W^2} \cdot \left(N_H(3) + 10 \cdot \frac{W-3}{W^2} \cdot N_H(5)\right)$$

W = Data width per address (bits)

 $N_H(k)$ = same as previous slide

Estimated number of false 4-bit MBUs:

$$N_{FM}(4) \approx \frac{(W-1) \cdot (W-2) \cdot (W-3)}{W^3} \cdot \left(N_H(4) + 5 \cdot \frac{3W-8}{W^2} \cdot N_H(6)\right)$$

• The same can be done for 5-bit, ... n-bit MBUs.

J.A. Clemente, M. Rezaei and F. J. Franco, *Reliability of Error Correction Codes Against Multiple Events by Accumulation*, in **IEEE Transactions on Nuclear Science (TNS)**, vol. 69, no. 2, pp. 169-180, February 2022.

Estimation of MBU rates – ECC reliability

Single Error Correction – Double Error Detection (SEC-DED) is sensitive against any MBU of any multiplicity.

Therefore, the probability of failure of SEC-DED is the cumulated probability of seeing any MBU of any multiplicity.



Total number of bitflips (m)

Estimation of MBU rates – ECC reliability

Similarly, **DEC-TED is sensitive against >2-bit MBUs**

It tolerates MBUs with multiplicity 2.

For DEC-TED,
$$\lambda = N_{FM}^{3+} = \sum_{i=3}^{W} N_{FM}(i)$$



• And similarly: $P_{failure_DEC_TED} = 1 - e^{-\lambda} = 1 - e^{-N_{FM}^{3+}}$

Similar calculations can be proposed for events that "break" other ECC types: DAEC, SNC-DND, etc.

J.A. Clemente, M. Rezaei and F. J. Franco, *Reliability of Error Correction Codes Against Multiple Events by Accumulation*, in **IEEE Transactions on Nuclear Science (TNS)**, vol. 69, no. 2, pp. 169-180, February 2022.

Estimation of MBU rates – ECC reliability

The number of accumulated bitflips needed keep different ECC techniques under certain reliability can also be calculated.

	Total number of bitflips (m)										
ECC reliab.	SEC-DED				DEC-TED				SNC-DND		TEC
	(22, 16)	(39, 32)	(72, 64)	(137, 128)	(27, 16)	(45, 32)	(79, 64)	(145, 128)	(144, 128)	(152, 128)	(64, 45)
99%	1682	1178	828	584	106813	66250	41328	25886	590	598	147488
99.9%	531	372	262	185	49505	30705	19154	11997	187	189	99477
99.99%	169	118	83	59	22975	14250	8890	5568	60	61	66480
99.999%	54	38	27	19	10664	6615	4127	2585	20	20	43581
99.9999%	18	13	9	7	4951	3071	1916	1201	7	7	27638
99.99999%	6	5	4	3	2299	1426	890	558	3	3	16772
99.999999%	3	2	2	2	1068	663	414	260	2	2	9786

RELIABILITY OF DIFFERENT ECC TECHNIQUES AGAINST MBUS PROVOKED BY SBU ACCUMULATION, FOR A 256-MB MEMORY

J.A. Clemente, M. Rezaei and F. J. Franco, *Reliability of Error Correction Codes Against Multiple Events by Accumulation*, in **IEEE Transactions on Nuclear Science (TNS)**, vol. 69, no. 2, pp. 169-180, February 2022.

5-Conclusions

- Modern memories implement techniques such as **bit interleaving** and **Error Correcting Codes** (ECC) to increase reliability.
- Devices are increasingly sensitive to multiple events, therefore a correct SBU/MCU extraction and classification is very important.
- In radiation-ground experiments, analyzing data correctly involves:
 - **Classifying events** by multiplicity.
 - Estimating the **"false MCU" rates**.

In the real world, **SBUs** coincidentally affecting **bitcells in the same word** can **break the ECC**.

- Chances are not that low!! (remember "birthday statistics").
- **Equations** have been given to estimate false "multiple event rates".
 - False MCUs: Provide accurate results in tests.
 - False MBUs: Estimate the reliability of ECC techniques.

Thanks for your attention!