Memory Reliability Model for Accumulated and Clustered Soft Errors

Soonyoung Lee[1], Sanghyeon Baeg[2], and Pedro Reviriego[3]
[1,2]Hanyang University, Sa 3-Dong, Sangrok-Gu, Ansan-si, Gyeonggi-do, Korea
[3]Universidad Antonio de Nebrija, C/ Pirineos, 55 E-28040 – Madrid, Spain
Tel: +81-31-400-4674, Fax: +82-31-696-5562
E-mail: [1]leesak@hanyang.ac.kr, [2]bau@hanyang.ac.kr, [3]previrie@nebrija.es

ABSTRACT

The soft error rate of memories is increased by high-energy particles as technology shrinks. Single-error correction codes (SEC), scrubbing techniques and interleaving schemes are the most common approaches for protecting memories from soft errors. It is essential to employ analytical models to guide the selection of interleaving distance; relying on rough estimates may lead to unreasonable design choices. The analytic model proposed in this paper includes row clustering effects of accumulated upsets and was able to estimate the failure probability with only a difference of 0.41% compared to the test data for a 45nm SRAM design.

I. INTRODUCTION

Computer memory can be disturbed by high-energy neutron particles from the terrestrial atmosphere, or alpha particles resulting from IC package material. Since such particle hits do not cause permanent damage, they are called soft errors. A particle hit can induce multiple cell upset (MCU) or single cell upset (SCU). Previous studies showed that the soft error rate is closely related to critical charge [1][2], process [3][4], and architecture [5][6]. At the memory architecture level, the interleaving technique is quite effective in reducing soft errors, especially for MCUs when implemented with SEC codes [7].

Fig. 1 illustrates the design of interleaving distance (ID) in a memory. ID is the physical spacing in the number of bits intra logical word. The group of cells within the ID belongs to the same bit in a logical word. For example, the cells in region C_0 in Fig.1 belong to the same bit but belong to different logical words.

The cells at a row are accessed by enabling a word line driver. The selected cells are routed to local inputs and outputs (I/Os) through the local multiplexers. For example, one cell from the cells in C0 is mapped to a local I/O through mux-logic, which is controlled by the column addresses. As a result of ID design, the clustered upsets (MCU or combinations of MCUs and SCUs) can be effectively spread over multiple logical words. The SEC, therefore, can correct the MCU with the proper choice of ID.

The effects of ID selection has been widely recognized and practiced at industry. However, the analytical approaches in selecting ID are hardly found in conjunction with soft error. The previous study [5] proposed an analytical ID selection method based on the total number of upsets. The upsets were assumed to be independent of physical proximity. Therefore, [5] apparently demonstrated the upper bound of failure probability due to soft errors. This study advances the work in [5] by considering any accumulated error until a memory is refreshed by either scrubbing or reset operations and proposes the reliability model.

This work was supported by the GRRC program of Gyeonggi province. [GRRC Hanyang2010-A01, Developing Sensor Network SoC Equipped with Low Power and Reliability Characteristics]

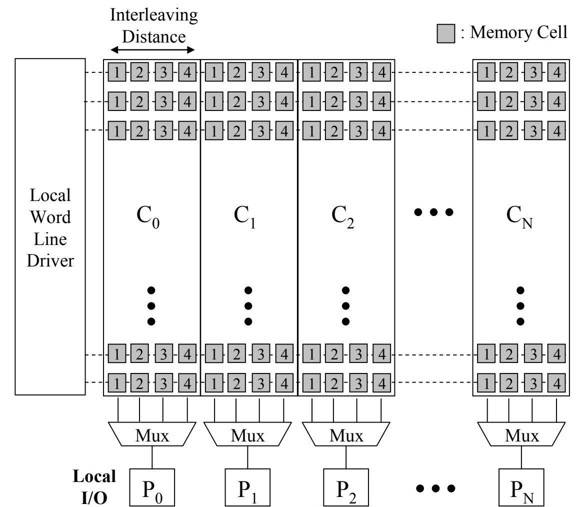

Fig. 1: Example of the interleaving distance architecture in memory tested in [5].

II. ROW GROUPING PROBABLITY

The soft error impact on SRAM reliability with interleaving distance is very sensitive to MCU because the upsets in MCU are geometrically clustered [3], [5]. Row depth, which is the maximum distance between upsets of MCU in a SRAM row, is the key factor in determining the interleaving distance. It was previously reported that the row depth follows a geometric distribution [5].

A scrubbing technique is also used to mitigate soft error-related reliability issue. Scrubbing interval cannot be always chosen within a soft error arrival rate for error detections and corrections. The reason is that excessively small scrubbing interval can overload the memory bandwidth. As a result, multiple errors can be accumulated.

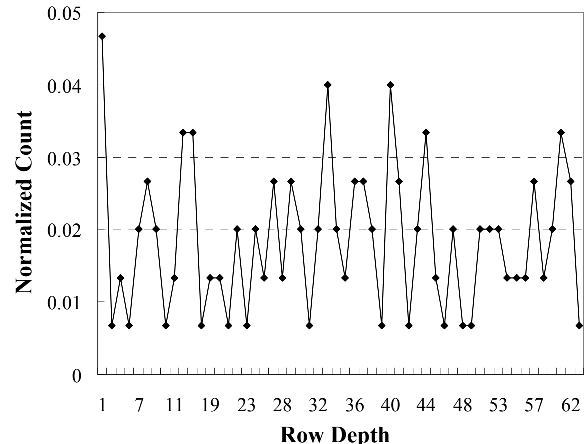

Fig. 2: Row depth for accumulated upset in a SRAM row.

In this paper, the row depth is plotted without the distinctions of MCU or SCU and is shown in Fig. 2. The accelerated test data is for 45nm SRAM under white beams up to 800MeV. The test did not stop when a word fails but let upsets accumulate further. Fig. 2 shows that the row depth does not follow geometric distribution when upsets are accumulated. This behavior matches our common-sense belief that the events and the locations due to the events are random in nature.

The same data used in Fig. 2 are shown in Fig. 3; however, they are plotted with variations in the number of upsets in a row. The asterisks in Fig. 3 are test data from 12 cases of three different chips; the plot is similar to the shape of the geometric distribution. For example, 38% of the rows in SRAM had a single upset on average.

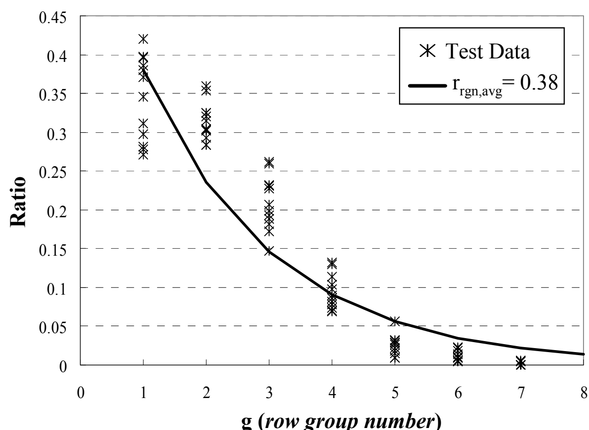

Fig. 3: Probability of the number of words with different row group numbers.

The number of upsets in a SRAM row is defined as *row-group number (rgn)*. The distribution of rgn is then expressed with the geometric distribution as in (1). $p_{rgn}(g)$ will refer to the row probability and is the probability for the number of rows with rgn equal to "*g*." r_{rgn} is a geometric constant.

$$p_{rgn}(g) = r_{rgn} \cdot (1 - r_{rgn})^{g-1}, \quad (1)$$

The solid line in Fig. 3 represents the geometric distribution with an average value of geometric constants, $r_{rgn,avg}$, from multiple test data. The r_{rgn} represents row clustering effects and can be dependent on the number of accumulated upsets. It is recommended to extract the parameter value from the accumulated upsets for scrubbing intervals for target technologies. Scrubbing interval is further discussed in Section IV.

As r_{rgn} approaches 1, the row probability with large rgn values is noticeably smaller than with smaller rgn values. On the contrary, the probabilities of all rgn values become similar with a geometric constant close to 0. Fig. 4 shows two examples of two different r_{rgn} values.

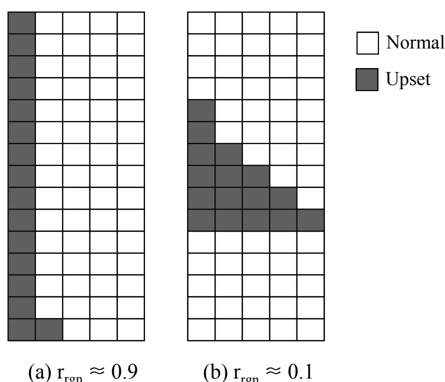

Fig. 4: Example for distribution of row group number.

Fig. 4 (a) is when the r_{rgn} is relatively large, 0.9. It is not too difficult to see the $P_{rgn}(1)$ is much larger than $P_{rgn}(2)$. Please note that the physical gap between upsets is not considered in determining row probability. Fig.4 (b) illustrates what happens when r_{rgn} is smaller than Fig.4 (a). The row probabilities of the five different rgn values are nearly the same.

III. FAILURE PROBABLITY MODEL WITH ID CONSIDERATIONS

As the row probability is available as discussed in the previous section, the failure probability needs to be developed with different choices of ID values. The failure probability for a row with different ID values was acquired from simulations. A 64-bit row was injected with randomly generated faults in random locations. The failure probabilities for different ID are shown in Fig. 5. The slope of the failure probability decreases as ID number increases. The failure probability with ID=16 is much smaller than with ID=2 for the same number of upsets. Such observations match our intuitive understanding of the issue.

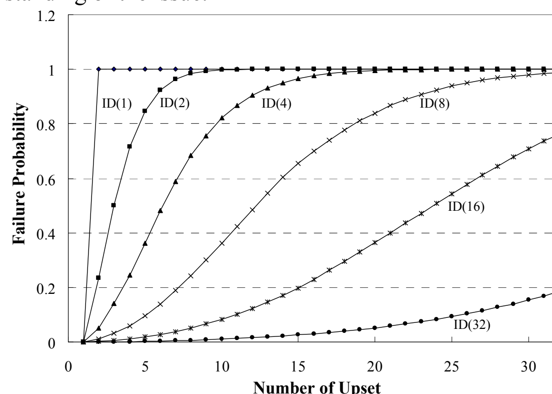

Fig. 5: Failure probability in a row with different ID values.

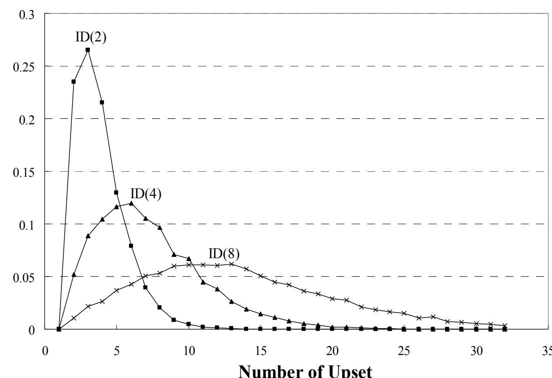

Fig. 6: The PDF of the failure probability in a row.

In order to derive the distribution, the probability in Fig. 5 is differentiated and shown in Fig. 6. The differentiated probability followed the shape of a Weibull distribution. The failure probability of ID=1 is always "1" when the number of upsets is more than one and is "0" when the number of upsets is less than equal to 1. It is assumed that the memory discussed in this paper is protected with a single error correction code (SEC) which can detect and correct a single-bit upset in a word. ID=1 means unprotected memory architecture for soft errors such as MCU. The failure model, therefore, is considered for ID is larger than 1.

The probability density function (PDF) for a row failure probability can be expressed as (2). α_{ID} and β_{ID} are shape and scale Weibull parameters. L is the number of upsets in a row.

$$f_{word}(L, ID) = \frac{\alpha_{ID}}{\beta_{ID}} \left(L / \beta_{ID}\right)^{\alpha_{ID}-1} \exp(-\left(L / \beta_{ID}\right)^{\alpha_{ID}}), \quad (2)$$

We observed that the number of upsets in a row with the highest density increases as ID values increase. In order to extract α_{ID} and β_{ID} values, the fitting algorithm Trust-Region was used [8]. Table 1 shows the results of parameter extractions. The error range was up to 5% in the worst case.

Table 1: Parameters of Weibull distributions in Fig. 6

	ID=2	ID=4	ID=8	ID=16	ID=32
α_{ID}	2.591	2.158	2.158	2.506	2.6
β_{ID}	3.891	7.753	14.93	28.1	58.75

The shape parameter α_{ID} ranged approximately between 2 and 2.5. The scale parameter β_{ID} can be roughly estimated as $2 \cdot ID$. Such a rough estimation method can increase the range of error, which was typically up to 15%.

The failure probability is reversely derived from the PDF in (2). $F_{word}(L,ID)$ is the failure probability for a word when there are L number of upsets in a row and shown in (3).

$$F_{word}(L, ID) = 1 - \exp\left(-(L / \beta_{ID})^{\alpha_{ID}}\right), \quad (3)$$

Let's consider the memory with R rows. The equation (4) shows the failure probability for the memory with R rows.

$$F_{mem}(X, ID) = \sum_{g=2}^{b} \frac{p_{rgn}(g) \cdot X}{g} \cdot \frac{F_{word}(g)}{R}, \quad (4)$$

$F_{mem}(X, ID)$ is the failure probability when there are X number of upsets for the memory with R rows and an interleaving distance of ID. The failure probability of a row with g upsets is same as $F_{word}(g,ID)$ because each word in a row has same failure probability.

The last term in (4) is the failure probability when a SRAM row is selected from R rows and the selected row has g number of upsets. When a memory with R rows has X upsets, $p_{rgn}(g) \cdot X$ is the number of upsets belonging to the rows with g upsets. It divided by g to get the total number of rows with g number of upsets. The value b is the upper limit in the summation and depends on the total number of upsets, X and the number of bits in a row, B. The value of "b" cannot be larger than B or X.

X is assumed to be less than or equal to X_{max} which is expressed in (5). The X_{max} is between R and $R \times B$ and its value depends on r_{rgn}. When X is larger than X_{max}, the number of rows with g upsets in (4) starts to be greater than R. In other words, the equation (4) is no longer valid once X becomes greater than X_{max}. It makes sense for distribution of row group number. It is also a valid assumption, as memory with more than X_{max} upsets would not be meaningful in reality. The scrubbing interval technique can guarantee this assumption by properly choosing the scrubbing interval based on event arrivals [6].

$$X_{max} = \sum_{g=1}^{B} p_{rgn}(g) \cdot g \cdot R \quad (5)$$

The total number of upsets X with MCU and SCU events was previously reported to follow the Compound Poisson (CP) model [9], which is shown in (6).

$$CP(X, t) = \sum_{Y=1}^{X} \frac{(\lambda t)^{Y} e^{-\lambda t}}{Y!} \binom{X-1}{Y-1} r^{X-Y} (1-r)^{Y} \quad (6)$$

The equation (4) can then be rewritten with the CP model as in (7). Again, it is assumed that the total number of upsets in a memory is no more than X_{max}, which can be achieved by the scrubbing interval adjustment.

$$F(t, ID) = \sum_{x=2}^{X_{max}} CP(x, t) \cdot F_{Block}(x, ID) \quad (7)$$

Lastly, the reliability function, $R(t)$, can be expressed in terms of the failure probability, $F(t)$, as shown (8).

$$R(t, ID) = 1 - F(t, ID) \quad (8)$$

IV. Results and Discussion

In order to validate the model proposed in this work, the failure probability was compared against the accelerated test data for a 45nm SRAM design. The parameters used for CP model were taken from the previous publication in [10]. The arrival rate λ was 0.0121863 and the geometric parameter r for CP was 0.3815. We used the time units directly from the accelerated test in the rest of the discussion for conveniences. The time unit can be translated into the normal environmental arrival rate.

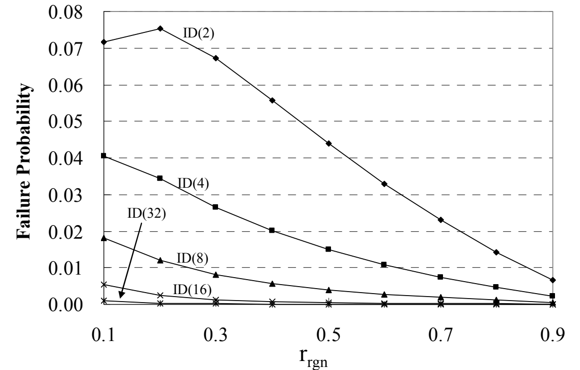

Fig. 7: Failure probability with different r_{rgn} values with scrubbing interval of 8000 sec.

The failure probabilities for the SRAM block with R=256 and B=64 are compared for various r_{rgn} values. When the scrubbing interval is assumed at 8000 sec, the failure probabilities of 9 r_{rgn} values were compared and shown in Fig. 7. As expected, the failure probabilities decrease as ID size increases.

It was generally observed that the failure probability decreases with increasing r_{rgn}. The observation makes sense because there are more probabilities in which the upsets fall in the same rows with low number of r_{rgn} values as in Fig. 4 (b), which in other words means that there is a fewer number of words failing and thus, fewer failure probabilities from the entire memory perspective. Such intuitive analysis leads to the observation that the failure probability with low r_{rgn} has higher ID values than the failure probability with high r_{rgn}. Please note that the failure probability with ID=2 initially increased with increasing r_{rgn} and decreased with further increases of r_{rgn} values. As previously discussed, the failure probability largely depends on two factors: the failure probability in a word and the number of failed words in a SRAM. In the case of ID=2, the failure probability in a word is high with r_{rgn}=0.1 with a fewer number of failed words. As r_{rgn} increased, the failure probability for a word is somehow maintained due to a smaller ID, and the number of failing rows increased, which resulted in an increased failure probability as r_{rgn} increased from 0.1 to 0.2. In the case of this example, the peaking behavior happened for ID=2. However, it could happen in other ID values with different X values.

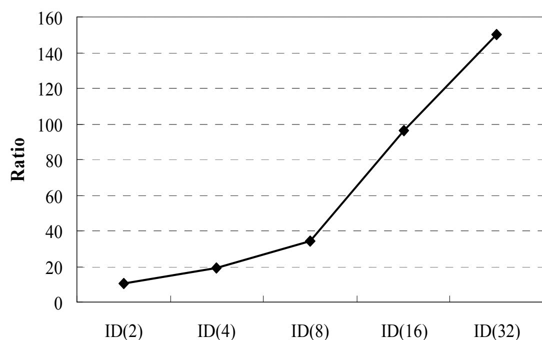

Fig. 8: The ratio of failure probability from the proposed model with two extreme cases of r_{rgn} values, 0.9 and 0.1.

Fig. 8 shows the ratios of failure probabilities with two extreme cases of r_{rgn} values, 0.1 and 0.9, to support the observations. The failure probability of r_{rgn}=0.1 is divided by the failure probability of r_{rgn}=0.9 and is shown in Fig. 8. The ratios of two different failure probabilities have far higher ID values.

The proposed model was also compared to the accelerated test data. The proposed model requires only one parameter, r_{rgn}, in addition to the CP parameters. The geometric constant value was extracted from the test data by a Trust-Region algorithm [8]. The extracted r_{rgn} value was 0.3329. The failure probabilities are shown in Fig. 9. The circled line shows the failure probability from the proposed model. The lines without circles represent the test data results. The proposed model showed similar trends as with the test data. The failure probabilities from the proposed model showed the biggest gap from the test results for ID=4 for a scrubbing interval. The maximum and average difference between proposed model and test data were 0.0206(2.06%) and 0.0145(1.45%), respectively.

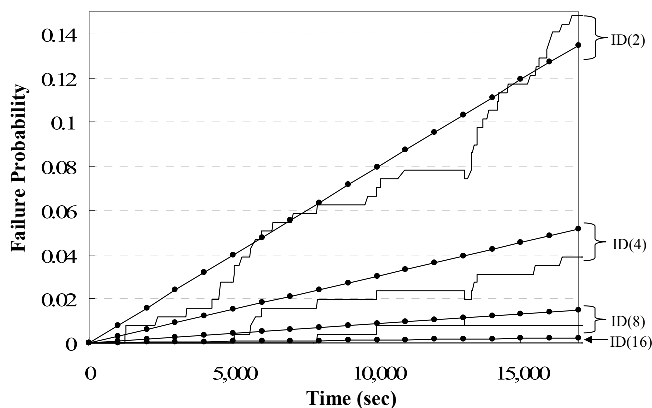

Fig. 9: Failure probability with r_{rgn}=0.3329 based on test data.

The failure probability of ID=2 is approximately 13% at the end of the scrubbing interval. ID design choices of 4, 8, and 16 showed 5.1, 1.5, and 0.2 %, respectively. It is therefore assumed that the proposed model can guide the ID selection of a SRAM by providing failure probability with ID variations. At the same time, an overly conservative design choice can be avoided by utilizing the analytical model. For example, the IDs of 8 and 16 are unnecessary if the failure probability requirement is less than 10%.

V. CONCLUSION

This study proposed the statistical failure model with embedded row clustering effects and interleaving schemes. The model could estimate the soft error rate with 1.45% gap for a sample 45nm SRAM design when compared to accelerated test data with interleaving schemes and scrubbing intervals considered. The proposed model was also able to demonstrate the differences of failure probabilities with different choices of three design parameters, row clustering effect(r_{rgn}), different ID parameters and scrubbing intervals. As a result, it can be effectively used for soft error estimations during early design stages.

ACKNOWLEDGMENTS

The author thanks Cisco Systems Inc., especially ShiJie Wen and Richard Wong for sharing the soft error test data.

REFERENCES

[1] E. Ibe, S. Chung, S.Wen, H. Yamaguchi, Y. Yahagi, H. Kameyama, S. Yamamoto, and T. Akioka, “Spreading diversity in multi-cell neutron induced upsets with device scaling,” in Proc. IEEE Custom Integrated Circuit Conf., pp. 437-444, 2006

[2] Y. Kawakami, M. Hane, H. Nakamura, T. Yamada, and K. Kumagai, “Investigation of soft error rate including multi-bit upsets in advanced SRAM using neutron irradiation test and 3D mixed-mode device simulation,” in Proc. IEDM, pp. 945-948, Dec. 2004

[3] D. Radaelli, H. Puchner, S. Wong, and S. Daniel, “Investigation of multi-bit upsets in a 150 nm technology SRAM device,” IEEE Trans. Nucl. Sci., vol. 52, no. 6, pp. 2433-2437, Dec. 2005.

[4] Y. Tosaka, H. Ehara, M. Igeta, T. Uemura, H. Oka, N. Matsuoka, and K. Hatanaka, “Comprehensive study of soft errors in advanced CMOS circuits with 90/130 nm technology,” in Proc. IEDM, pp. 941-944, Dec. 2004

[5] Sanghyeon Baeg, ShiJie Wen and Richard Wong, “SRAM Interleaving Distance Selection With a Soft Error Failure Model,” IEEE Trans. On Nuclear Science, Vol. 56, No. 4, pp. 2111-2118, Aug. 2009.

[6] Pedro Reviriego, Juan Antonio Maestro, Sanghyeon Baeg, "Optimizing Scrubbing Sequences for Advanced Computer Memories", IEEE Trans. on Device and Materials Reliability, Vol. 9, Issue 4, Dec. 2009

[7] T. P. Haraszti, CMOS Memory Circuits. Boston, MA: Kluwer, pp. 438-468, 2000.

[8] Wenyu Sun, Optimization Theory and Methods : Nonlinear Programming, Springer Publishing, pp. 300-316, 2009

[9] M. Sahinoglu, “Compound-Poisson Software Reliability Model,” IEEE Trans. on Software Engineering, vol. 18, no. 7, pp. 624-630, July 1992.

[10] Sanghyeon Baeg, ShiJie Wen, Richard Wong, “Minimizing Soft Errors in TCAM Devices: A Probabilistic Approach to Determining Scrubbing Intervals”, IEEE Trans. on Circuits and Systems – I, Vol. 57, No. 4, pp. 814-822, April 2010